

Mirko Nentwig

Untersuchungen zur Anwendung von computer- generierten Kamera- bildern für die Ent- wicklung und den Test von Fahrer- assistenzsystemen

Audi-Dissertationsreihe, Band 88

Untersuchungen zur Anwendung von computergenerierten
Kamerabildern für die Entwicklung und den Test von
Fahrerassistenzsystemen

Der Technischen Fakultät
der Friedrich-Alexander-Universität
Erlangen-Nürnberg

zur
Erlangung des Grades Dr.-Ing.
vorgelegt von
Dipl.-Ing. (FH) Mirko Nentwig
aus Wolfsburg

Bibliografische Information der Deutschen Nationalbibliothek

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

1. Aufl. - Göttingen : Cuvillier, 2014

Zugl.: Erlangen-Nürnberg, Univ., Diss., 2013

978-3-95404-625-6

Als Dissertation genehmigt von
der Technischen Fakultät
der Friedrich-Alexander-Universität Erlangen-Nürnberg
Tag der mündlichen Prüfung: 17.12.2013

Vorsitzende des Promotionsorgans: Prof. Dr. Ing. habil. Marion Merklein

Gutachter: Prof. Dr. Marc Stamminger
Prof. Dr. Rüdiger Westermann

© CUVILLIER VERLAG, Göttingen 2014
Nonnenstieg 8, 37075 Göttingen
Telefon: 0551-54724-0
Telefax: 0551-54724-21
www.cuvillier.de

Alle Rechte vorbehalten. Ohne ausdrückliche Genehmigung des Verlages ist es nicht gestattet, das Buch oder Teile daraus auf fotomechanischem Weg (Fotokopie, Mikrokopie) zu vervielfältigen.

1. Auflage, 2014

Gedruckt auf umweltfreundlichem, säurefreiem Papier aus nachhaltiger Forstwirtschaft.

978-3-95404-625-6

Kurzbeschreibung

In dieser Dissertation werden Methoden zur Erprobung von kameraunterstützten Fahrerassistenz- und aktiven Sicherheitssystemen entwickelt. Es wird die Definition „kameraunterstützt“ eingeführt, da viele Systeme neben einer Kamera weitere Sensoren zur Wahrnehmung des Umfelds verwenden. Der Schwerpunkt der Anwendung der Methoden soll die Hardware-in-the-Loop Erprobung von vernetzten Fahrerassistenzsystemen sein. Dieser Anwendungsfall sieht einen End-To-End Test der Funktionen vor, d.h. die Sensordaten sollen möglichst am Beginn der Signalverarbeitung eingespeist werden. Im Falle, dass ein Kamerasystem stimuliert werden soll, bietet sich die Erzeugung eines synthetischen Kamerabilds mit Mitteln der Computergrafik an. Da im Vergleich zu anderen Sensoren, wie z.B. Radar, der Aufwand geringer ist. Durch die Einschränkung, dass ein Modell immer eine Vereinfachung der Realität ist, ist es von Wichtigkeit zu untersuchen, welchen Realismusgrad die Simulation für eine plausible Stimulation der Bildverarbeitung aufweisen muss. Weiterhin müssen mögliche Anwendungsfelder für die Simulation in der Erprobung und Entwicklung definiert werden.

Zur Stärkung der methodischen Arbeit in der Computergrafik wurde ein Hardware-in-the-Loop Prüfstand mit der Virtual Test Drive-Toolkette zur Erprobung von Fahrerassistenz- und aktiven Sicherheitssystemen bei der AUDI AG entwickelt. Diese Systeme werden seit 2010 entwicklungsbegleitend eingesetzt. Mittels der Erfahrungen aus der Anwendung konnte die methodische Entwicklung gestärkt werden.

Inhaltsverzeichnis

1. Einleitung	1
1.1. Motivation	1
1.2. Zielsetzung	4
1.3. Aufbau der Arbeit	5
2. Fahrerassistenz- und aktive Sicherheitssysteme	7
2.1. Aktuelle Fahrerassistenzsysteme und aktive Sicherheitssysteme	7
2.1.1. Adaptive Cruise Control	8
2.1.2. Spurverlassenswarnung/Spurhalteunterstützung	9
2.1.3. Lichtassistenzsysteme	10
2.1.4. Verkehrszeichenerkennung	11
2.1.5. Kollisionsminderung/-vermeidung	12
2.2. Sensorik zur Wahrnehmung des Fahrzeugumfelds	12
2.2.1. Sensorfusion	13
2.2.2. Radar	13
2.2.3. Ultraschallsensor	14
2.2.4. Navigationssystem/prädiktive Streckendaten	14
2.2.5. Kamera	14
2.3. Bildverarbeitung im Fahrzeug	24
2.3.1. Signalverarbeitung in der Bildverarbeitung	24
2.3.2. Spurerkennung	25
2.3.3. Fahrzeugdetektion	27
2.3.4. Verkehrszeichenerkennung	30
2.3.5. Herausforderungen	31
2.3.6. Zusammenfassung	31
3. Entwicklungsmethoden für Fahrzeugelektronik	33
3.1. Entwicklungsprozess in der Automobilindustrie	33
3.2. Erprobung von Fahrerassistenzsystemen	34
3.3. Simulationsumgebungen für Fahrerassistenz- und aktive Sicherheitssysteme	37
3.3.1. Wissenschaftliche Simulationen	37
3.3.2. Kommerzielle Simulationen	40
3.3.3. Vergleich der Simulationsumgebungen und Zusammenfassung	44
4. Evaluierung von Konzepten der Computergrafik	49

4.1.	Anforderungsanalyse	49
4.1.1.	Stimulation der Signalverarbeitung	50
4.1.2.	Stimulation der Bildverarbeitung	50
4.1.3.	Anforderungen an die Qualität der Simulation	51
4.2.	Beschreibung der Simulationsumgebung	53
4.2.1.	Virtual Test Drive-Framework	53
4.2.2.	Architektur der Simulationsumgebung der Kamera	54
4.2.3.	OpenSceneGraph basierte Grafik-Engine	55
4.3.	Simulation der Beleuchtung	57
4.3.1.	Beleuchtungsmodelle in der Computergrafik	57
4.3.2.	Schattenberechnung	58
4.3.3.	Atmosphärische Simulation	62
4.3.4.	Zusammenfassung	75
4.4.	Geometrie und Oberflächeneigenschaften von Objekten	76
4.4.1.	Oberflächenmodelle	76
4.4.2.	Fahrzeuge	77
4.4.3.	Straße	88
4.4.4.	Verkehrsschilder	107
4.4.5.	Zusammenfassung	109
4.5.	Synthese von Kamerabildern	113
4.5.1.	Bildsyntheseprozess der Computergrafik	113
4.5.2.	Steigerung der Realitätsnahe eines synthetischen Bilds	115
4.5.3.	Anti-Aliasing-Filter	116
4.5.4.	Modell des Strahlungsaufnehmers	118
4.5.5.	Rauschen	123
4.5.6.	Tiefenschärfe	125
4.5.7.	Verzeichnung	129
4.5.8.	Zusammenfassung	131
4.6.	Aufbau von Szenen und Szenarien	132
4.6.1.	Prozess zur Rekonstruktion von Testfahrten	132
4.6.2.	Extraktion von Daten aus der realen Testfahrt	133
4.6.3.	Verfahren zur Generierung von Straßenverläufen	136
4.6.4.	Gestaltung der Szene	141
4.6.5.	Zusammenfassung	142
4.7.	Zusammenfassung	143
5.	Entwicklung eines generischen Verfahrens zur Validierung und Kalibrierung der Simulation	145
5.1.	Stand der Technik	145
5.2.	Metrik zum Vergleich der Bildinhalte	149
5.2.1.	Anforderungen	149
5.2.2.	Struktur der Metrik	150
5.2.3.	Strategien zum Vergleich der Bildinhalte	150

5.2.4.	Merkmale für den Vergleich der Bildinhalte	153
5.2.5.	Untersuchung der Vergleichsstrategien und der Merkmale für den Vergleich der Bildinhalte	158
5.2.6.	Gesamt-Maß der Metrik	161
5.2.7.	Anwendungsbeispiel	162
5.3.	Kalibrierung der Simulationsumgebung durch Anwendung der Metrik	163
5.3.1.	Integration der Metrik in die Simulationsumgebung	164
5.3.2.	Verfahren zur Optimierung	165
5.3.3.	Anwendungsbeispiel	166
5.4.	Zusammenfassung	169
6.	Untersuchung der Anwendbarkeit einer Kamerasimulation im Entwicklungsprozess	171
6.1.	Entwicklung eines Fahrzeugerkennungssystems	171
6.1.1.	Evaluierung	172
6.1.2.	Zusammenfassung	173
6.2.	Untersuchung des Verhaltens der Signalverarbeitungskette in der Simulation	174
6.2.1.	Vergleich der Kundenfunktion Active Lane Assist	174
6.2.2.	Vergleich der Spur- und Fahrzeugerkennungsfunktion	176
6.2.3.	Vergleich der Hypothesenverifikation der Fahrzeugerkennung	179
6.2.4.	Vergleich der Hypothesengenerierung der Fahrzeugerkennung	182
6.3.	Zusammenfassung	186
7.	Hardware-in-the-Loop Simulator zur Erprobung von Fahrerassistenzsystemen	189
7.1.	Systemaufbau	189
7.2.	Stimulation der Fahrzeugkamera	190
7.2.1.	Abbildungsfehler	191
7.3.	Weitere Sensormodelle	193
7.3.1.	Radar	194
7.3.2.	Prädiktive Streckendaten	194
7.3.3.	Ultraschall	194
7.4.	Anwendungsbeispiele	195
7.4.1.	Active Lane Assist	195
7.4.2.	Tempolimitanzeige	195
7.4.3.	Akzeptanz der virtuellen Erprobung	196
7.5.	Zusammenfassung	196
8.	Zusammenfassung und Ausblick	199
8.1.	Zusammenfassung	199
8.2.	Ausblick	205
9.	Publikationen, Patente und betreute Arbeiten	207
9.1.	Publikationen	207
9.2.	Patentanmeldungen	208

9.3. Betreute Arbeiten	208
----------------------------------	-----

Literaturverzeichnis **209**

A. Anhang	i
A.1. Vergleich des Simulationslösungen	i
A.2. Koordinatensysteme	i
A.2.1. Weltkoordinatensystem (WKS)	i
A.2.2. Umfeldsensorikkoordinatensystem (USK)	i
A.2.3. Kamerakoordinatensystem (KOS)	ii
A.2.4. Straßenkoordinatensystem	ii
A.2.5. OpenGL-Koordinatensystem	ii
A.3. Bewertung der Fahrzeuge	iii
A.4. Anforderungen an die Simulation	iv
A.4.1. Geometrie- und Oberflächeneigenschaften	iv
A.4.2. Verkehrszeichen	v
A.4.3. Beleuchtungssimulation	vi
A.4.4. Szenen und Szenario Design	vi
A.4.5. Bildentstehungsprozess	vii
A.4.6. Weitere Anforderungen an das Testsystem	vii
A.5. Szenengraphen	vii
A.5.1. Fahrzeug	vii
A.6. Herleitung der Emission von Wasserpartikeln	vii

Abbildungsverzeichnis

1.1. Entwicklung der Verunglückten und Verkehrstoten in Deutschland von 1953 bis 2011 basierend auf den Daten von [11]	1
1.2. Sensorik Audi A6 [19]	3
2.1. Sensoren des Adaptive Cruise Control [19]	8
2.2. Schema Spurhalteunterstützung	9
2.3. Lichtassistenzsysteme	10
2.4. Verkehrszeichenerkennung	11
2.5. Audi PreSense [19]	12
2.6. Schematische Darstellung des optischen Pfads und wichtiger Kamerabestandteile	15
2.7. Strahlendiagramm eines optischen Systems, nach [141]	16
2.8. Definition des Seh winkels einer Kamera, nach [141]	17
2.9. Strahlendiagramm zur Definition der Schärfentiefe, nach [141]	18
2.10. Schärfefehler einer Linse, nach [141]	20
2.11. Linsenverzeichnungen (a) unverzeichnetes Abbildung, (b) Kissenförmige radiale Verzeichnung, (c) tonnenförmige radiale Verzeichnung, (d,e) tangentielle Verzeichnung, nach [141]	20
2.12. Häufige Farbfilter a) Monochromer-Filter b) RGB-Bayer-Filter c) RCCC-Farbfilter	22
2.13. Ablauf der Bildverarbeitung, nach [130]	24
2.14. Ablauf der Spurerkennung, nach [108]	25
2.15. Ablauf Fahrzeugerkennung, nach [179]	27
2.16. Schema der Verkehrszeichenerkennung, nach [71]	30
3.1. Das V-Modell, nach [187]	34
3.2. Erprobung von kamerabasierten Fahrerassistenzsystemen	35
3.3. Model-, Software- und Hardware-in-the-Loop	36
4.1. Screenshot der Simulationsumgebung	53
4.2. Komponenten des Virtual Test Drive Frameworks	54
4.3. Architektur der Simulationsumgebung der Kamera	55
4.4. Architektur der Grafik-Engine	55
4.5. Shadow Mapping Algorithmus	58
4.6. Vergleichende Darstellung der Shadow-Maps	60
4.7. Auswertung der Schattenverfahren	61
4.8. Koordinatensystem zur Bestimmung der Sonnenposition, nach [146]	69
4.9. Ergebnisse der Simulation des Sonnenlichts im RGB-Farbraum	70

4.10. Vergleich des Hoffmann- und des Preetham-Modells	73
4.11. Wetterbedingungen und Parametrierung des atmosphärischen Modells	74
4.12. Fahrzeugmodelle	77
4.13. Oberflächenmodelle der Fahrzeug-Karosserie	78
4.14. Aufbau Fahrzeuglack	78
4.15. Vergleich von Oberflächenmodellen zur Simulation von Fahrzeuglack	81
4.16. Regionen für den Vergleich der Oberflächenmodelle	82
4.17. Inhalt des Geometry-Buffers [109]	83
4.18. Verfahren zur Simulation der Ausleuchtung einer Szene bei Dunkelheit [109]	84
4.19. Umsetzung von Scheinwerfern in der Simulation, Bildinhalte aus [109]	85
4.20. Links: Physikalisch basierte Lichtstreuung aus [109], Rechts: Anwendung der Billboard- Technik zur Simulation der Lichtkränze bei Tag und Nacht im Vergleich zu einer realen Szene	85
4.21. Vergleich zweier realer Testfahrten mit der Simulation durch Anwendung einer Schwellwert- Segmentierung, Einzelbilder von [109]	86
4.22. Schema Parallax-Mapping	90
4.23. Auswirkung der Visualisierung auf verschiedene Straßenbeläge	91
4.24. Auswertung der Straßenbeläge	91
4.25. Vergleich der Oberflächenmodelle	96
4.26. Vergleich der Oberflächenmodelle bei tief stehender Sonne	97
4.27. Darstellung von Markierungen	99
4.28. Beispiel für eine Regenszene	100
4.29. Reflexionsverhalten auf nassen Asphalt, nach [90]	101
4.30. Simulation einer nassen Straßenoberfläche	103
4.31. Zusammenhänge der Emission von Wasserpartikeln	103
4.32. Vergleich des Alpha-Werts	105
4.33. Erzeugung von Reifenspuren in der Simulation	106
4.34. Vergleich der optischen Qualität der Simulation einer nassen Straße	106
4.35. Gegenüberstellung realer und simulierter Verkehrszeichen in verschiedenen Entfernungen zur Kamera	108
4.36. Kreuzkorrelation des Grauwerte-Bilds der Schilder über die Distanz	109
4.37. Kreuzkorrelation des Gradienten-Bilds der Schilder über die Distanz	110
4.38. Bildgenerierungsprozess: synthetische vs. reale Kamera	113
4.39. Definition des Viewing-Frustrums, nach [14]	114
4.40. Umsetzung der Kamerasimulation	116
4.41. Verwendete Szene und Messfenster für Vergleich der Anti-Aliasing Techniken	117
4.42. Vergleich der Anti-Aliasing-Filter für definierte Messfenster	117
4.43. Auswertung der Anti-Aliasing Techniken: links Messfenster 1 Zeile 20 und rechts Messfens- ter 2 Zeile 10	118
4.44. Simulation der Farbfilter einer Kamera	120
4.45. Einfluss der Belichtungssteuerung auf das Bild	122
4.46. Vergleich verschiedener Antwortfunktionen	123

4.47. Einfluss der Rauschmodelle auf das finale Bild	126
4.48. Einfluss der Tiefenschärfe auf die Gradientenbeträge im Bild	128
4.49. Auswertung der Verzeichnungssimulation	130
4.50. Prozess zur Rekonstruktion von Testfahrten	132
4.51. Extraktion von Messdaten für die Rekonstruktion der Testfahrt	134
4.52. Szenario: Messfahrzeug schließt zu EGO-Fahrzeug auf	136
4.53. Geschätzte Geschwindigkeit des EGO-Fahrzeugs (links) und Abstand zum CO-Fahrzeug (rechts) über die Simulationszeit	137
4.54. Prozess zur Abbildung von realen Straßenverläufen in der Simulation	137
4.55. Ablauf der Segmenterzeugung	138
4.56. Vergleich der Simulation mit realen Messdaten, aus [72]	140
4.57. Die Trainingsdatensätze und Ergebnisse im Vergleich [117]	141
5.1. Struktur der Metrik	150
5.2. Ansätze zum Vergleich der Bildinhalte	151
5.3. Eigenschaften verschiedener Bildmerkmale hinsichtlich ihrer Bedeutung in der Bildver- arbeitung, nach [171]	153
5.4. Eigenschaft Mittelwert und Varianz	154
5.5. Verwendete Daten für die Bewertung der Vergleichsstrategien	158
5.6. Auswirkung der Teilmaße	160
5.7. Anwendung der Metrik auf gesamte Bilder im Vergleich zum SSIM	162
5.8. Anwendung der Metrik auf die Region um das Fahrzeug und Vergleich zum SSIM	163
5.9. Optimierungsalgorithmus	164
5.10. GPU-Implementierung	164
5.11. Ergebnisse des Anwendungsbeispiels zur Berechnung auf der Grafikkarte	165
5.12. Links: Maske zur Optimierung der Straße, Rechts: Maske zur Optimierung der Fahrzeug- farbe rot = strukturelle Differenz, grün = geometrische Differenz, blau = photometrische Differenz	166
5.13. Links: Entwicklung des Distanzmaßes über Permutation, Rechts: Werte der Parameter des Straßenmodells über Permutation	167
5.14. Links: Entwicklung des Distanzmaßes über Permutation, Rechts: Wert des Parameters der Fahrzeugfarbe über Permutation	168
5.15. Vergleich der Parametersätze der Straße Links: Startparametrierung, Rechts: Optimierte Parametrierung der Straße und des Fahrzeugs rot = strukturelle Differenz, grün = geome- trische Differenz, blau = photometrische Differenz	169
6.1. Klassifikatorkaskade nach Viola und Jones [185]	171
6.2. Ergebnis der Evaluierung der Fahrzeugklassifikators unter verschiedenen Witterungsbedin- gungen	173
6.3. Testaufbau für den Test eines Lane Departure Warnings	174
6.4. Auswertung des NHTSA-Versuchs [18]	175

6.5. Szenarien für die Untersuchung der Ähnlichkeit der Modellgrößen der Bildverarbeitung Links: Reale Szene, Rechts: Simulation	177
6.6. Links: Vergleich der Prädiktionsweite des Straßenverlaufs, Rechts: Vergleich der Spurbreite	177
6.7. Links: Vergleich der Abstandschätzung, Rechts: Vergleich der Fahrzeugbreite	178
6.8. Links: reale Szene, Rechts: simulierte Szene	179
6.9. Messaufbau für die Auswertung	180
6.10. Links: Auswertung der gesamten Szene, Rechts: Auswertung ab Frame 420	181
6.11. Auswertung von Keyframes	182
6.12. Ergebnisse der Kalibrierung der Simulation	183
6.13. Wahre Hypothesen	184
6.14. Falsche Hypothesen	185
7.1. Systemtopologie der virtuellen Erprobungsfahrt	190
7.2. Aufbau der Dunkelkammer	191
7.3. Links: Vergleich der Grauwerte verschiedener Monitore, Rechts: Vergleich der Histogramme reale Szene vs. abgefilmte reale Szene	192
7.4. Links: Interferenz zwischen Monitor und Kamera	193
7.5. Rechts: Diskretes Motion-Blur	193
7.6. Radarmodell und Reaktion des Adaptive Cruise Control	194
7.7. Stimulation des Navigationssystems in der Simulation	194
7.8. Sensormodell für Ultraschall: Links Szene in Virtual Test Drive, Rechts: Szene des Sensor- modells	195
7.9. Testfall für das Audi Active Lane Assist: System muss bei einer stark verengter Spur von aktiv auf passiv wechseln	196
7.10. Testfall für die Tempolimitanzeige: Erprobung des Systemverhaltens bei bestimmter Schil- derabfolge	197
7.11. Entwicklung der Erprobungsumfänge mit Einführung der virtuellen Probefahrt	198
A.1. Vergleich der Simulationslösungen hinsichtlich der Erprobung von kameraunterstützten Fah- rerassistenzsystemen	ix
A.2. Weltkoordinatensystem (WKS)	x
A.3. Umfeldsensorikkoordinatensystem (USK)	x
A.4. Kamerakoordinatensystem (KOS)	x
A.5. Straßenkoordinatensystem	x
A.6. OpenGL-Koordinatensystem (Canoical View Volume)	xi
A.7. Szenengraph der Fahrzeugmodelle	xi
A.8. Berechnung des Deklinationwinkels	xi

Tabellenverzeichnis

4.1. Parametrierung der Heney-Greenstein Funktion	74
4.2. Scores der Messbereiche	82
4.3. Vergleich Visualisierungsverfahren für Asphalt	91
4.4. Vergleich Visualisierungsverfahren für Kopfsteinpflaster	92
4.5. Vergleich Visualisierungsverfahren für Schotter	92
4.6. Parametrierungen der Oberflächen-Modelle	96
4.7. Parametrierung des Rauschmodells	125
5.1. Vergleich der Blockbildungs-Verfahren	159
5.2. Analyse der Eigenschaften	160
5.3. Startwerte der Optimierung	166
5.4. Ergebnisse der Optimierung der Straßenparameter	167
6.1. Ergebnisse der Hypothesengenerierung in verschiedenen Szenarien	183
6.2. Verteilung der falschen Hypothesen auf die Klassen	184
A.1. Weltkoordinatensystem	i
A.2. Umfeldsensorikkoordinatensystem	ii
A.3. Kamerakoordinatensystem	ii
A.4. Straßenkoordinatensystem	ii
A.5. OpenGL-Koordinatensystem (Canoical View Volume)	iii

Danksagung

An dieser Stelle bedanke ich mich bei allen Personen, die mich während der Anfertigung dieser Dissertationsschrift begleitet haben. Mein besonderer Dank gilt Prof. Dr. Marc Stamminger von der Friedrich-Alexander-Universität Erlangen-Nürnberg für die großartige Betreuung und die wertvollen Diskussionen während der Promotionszeit. Für die Übernahme des Zweitgutachtens gilt mein besonderer Dank Prof. Dr. Rüdiger Westermann von der Technischen Universität München.

Ich bedanke mich bei meinem Chef Dr. Maximilian Miegler für die Möglichkeit zur Durchführung meiner Promotion in der Abteilung Hardware-in-the-Loop Funktionserprobung der AUDI AG und die hervorragende langjährige Unterstützung. Herrn Reinhard Schieber danke ich für die großartige Betreuung und Unterstützung während meiner dreijährigen Zeit als Doktorand. Herrn Norbert Strupf danke ich ebenso für die Unterstützung meines Promotionsprojekts.

Meinen Kollegen Dr. Andreas Jörg, Dr. Frank Derichsweiler, Dr. Sebastian Thiel und Dirk Zitterell danke ich für die vielen wertvollen fachlichen Diskussionen. Weiterhin bedanke ich mich bei meinen aktuellen und ehemaligen Kollegen der Hardware-in-the-Loop Funktionserprobung für die gute langjährige Zusammenarbeit und das Interesse an meinem Promotionsprojekt: Richard Maubach, Sebastian Fritsch, Sven Strauchmeier, Markus Ritzer, Frank Stocker, Bernhard Speth, Nicole Lenk, Thomas Kluitmann, Peter Hoffmann, Robert Walesch, Richard Bergmann, Christian Rehm, Martin von der Grün, Constantin Brückner, Patricia Lorber und Martin Irion.

Herrn Thomas Ganslmeier und Herrn Andreas Kern von der Audi Electronics Venture GmbH danke ich für die gute Zusammenarbeit bei der Realisierung von gemeinsamen Ideen, welche nachhaltig die Erprobung von Fahrerassistenzsystemen im Volkswagen Konzern verändert haben.

Ich bedanke mich herzlich bei meinen ehemaligen studentischen Mitarbeitern Herrn Rayan Naal, Herrn Daniel Melzer, Herrn Andreas Aspeleiter und Herrn Sebastian Königer.

Bei Prof. Dr. Paolo Mercorelli bedanke ich für die außergewöhnliche Förderung und wissenschaftliche Arbeit während meines Studiums.

Mein größter Dank gilt meinen Eltern für die immerwährende Unterstützung, welche es mir ermöglicht hat diesen Weg erfolgreich zu meistern. Nicht weniger danke ich meiner Lebensgefährtin Jutta Alt für die Motivation und moralische Unterstützung dieses Projekt erfolgreich zum Abschluss zu bringen.

Abschließend bedanke ich mich bei meiner Familie und meinem Freundeskreis, welcher mich in den letzten Jahren verständnisvoll und wohlwollend begleitete.

1. Einleitung

1.1. Motivation

Individuelle Mobilität ist das Grundbedürfnis einer modernen Gesellschaft. Ein effizientes und sicheres Verkehrssystem ist deshalb die Grundlage einer modernen Volkswirtschaft und genießt höchste wirtschaftliche Bedeutung [5]. Aufgrund der immensen gesellschaftlichen Kosten von über 160 Milliarden €, bedingt durch 1 500 000 Verkehrsunfälle, 54 000 Verkehrstote und 2 000 000 Verletzte wurde im Jahr 2001 das ambitionierte Ziel ausgegeben, die Zahl der Verkehrstoten bis 2010 zu halbieren. Diese Ziel wurde im Weißbuch zur europäischen Verkehrspolitik [4] formuliert. Im Rahmen dieser Anstrengungen wurden Maßnahmen zur Fahrerqualifikation, Verbesserung der Straßeninfrastruktur und der Fahrzeugtechnologie auf den Weg gebracht. In Abbildung 1.1 ist die Entwicklung der verunglückten Personen, mit Verletzungen und der Verkehrstoten in Deutschland von 1953 bis 2011, dargestellt. Wie ersehen werden kann hat sich

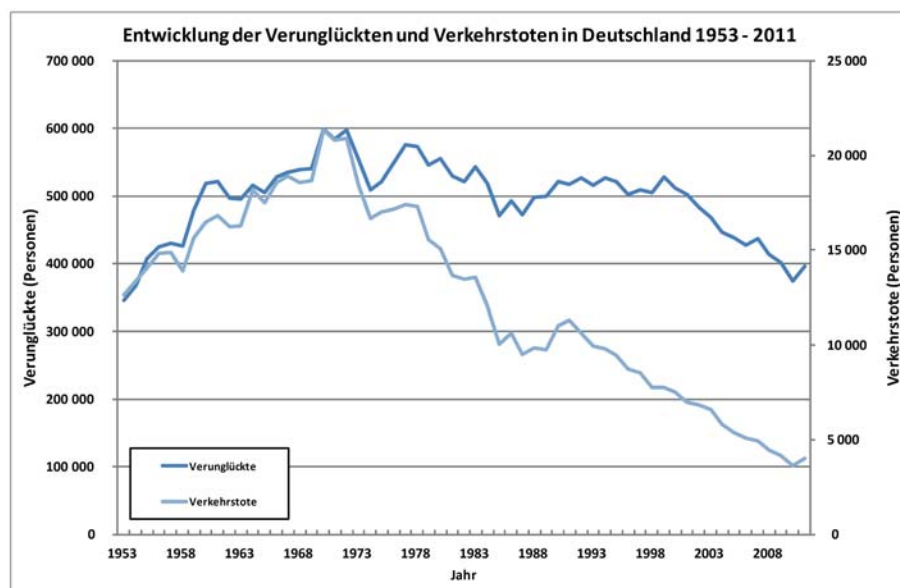


Abbildung 1.1.: Entwicklung der Verunglückten und Verkehrstoten in Deutschland von 1953 bis 2011 basierend auf den Daten von [11]

seit dem Jahr 2000 die Anzahl der Verunglückten von 511 577 auf 395 934 im Jahr 2011 Personen reduziert. Dies entspricht einem Rückgang von 22,6%. Im gleichen Zeitraum hat sich die Anzahl der getöteten Personen, welche eine Teilmenge der Verunglückten darstellen, von 7 503 Personen auf 4 002 Personen reduziert, was einem Rückgang von 46,6% entspricht. Nach Aussage des statistischen Bundesamts trat im

Jahr 2011 eine Stagnation aufgrund der schlechten Witterung zum Jahresbeginn ein. Bei einer Auswertung des Fehlverhaltens von Fahrzeugführern im Zusammenhang von Unfällen mit Personenschäden [5] folgt, dass von 520 702 der beteiligten Fahrzeugführern 350 323 ein Fehlverhalten zugeschrieben werden kann. Die häufigsten Ursachen sind: nicht angepasste Geschwindigkeit, Missachtung der Vorfahrt und Unterschreitungen des Abstands. In [10] wird angeführt, dass fast 95% der Unfälle zumindest zum Teil, auf den Faktor Mensch zurückzuführen sind. Drei Viertel der Unfälle sind ausschließlich auf menschliches Fehlverhalten zurückzuführen. In diesem Zusammenhang hat eine im Jahr 2002 eingesetzte Experten-Gruppe, mit Vertretern der Automobilindustrie und der Legislative, das größte Potential zur Lösung der Sicherheitsprobleme im Straßenverkehr durch den Einsatz von intelligenten Fahrzeugsicherheitssystemen identifiziert [6]. Seit 1970 hatte sich gezeigt, dass durch Einsatz von passiven Sicherheitssystemen, wie z.B. Sicherheitsgurte, Airbags und Fahrgastzellen, die Zahl der Verletzten um 20 % und die Zahl der getöteten Unfallbeteiligten um 50 % reduziert werden konnte. Obwohl die Verkehrsumfälle während dieser Zeit um ca. 40 % anstiegen. Jedoch ist das Potential der passiven Systeme weitestgehend ausgeschöpft, so dass es immer schwieriger wird weitere Verbesserungen der Sicherheit mit akzeptablen Aufwand zu erreichen [6].

Daraus entstand der Bedarf nach aktiven Sicherheitssystemen, welche den Fahrer unterstützen. Diese Fahrerassistenzsysteme (FAS) informieren und warnen den Fahrer, erhöhen den Komfort und reduzieren die Arbeitsbelastung durch eine aktive Stabilisierung und das Manövrieren des Fahrzeugs [8]. Die Systeme übernehmen nicht die Fahraufgabe im Ganzen, daher bleibt die Verantwortung zur Steuerung des Fahrzeugs immer beim Fahrer. Die bekanntesten Beispiele für diese Systeme sind das Anti-Blockier-System (ABS) und die elektronische Stabilitätskontrolle (ESC/ESP) [23], zur Reduktion von Schleuder- oder Überschlagsunfällen.

Zur Vermeidung von Kollisionen nach einem Fahrspurwechsel oder dem unbeabsichtigten Verlassen der Fahrspur/Fahrbahn sind diese Fahrzeugstabilisierungssysteme nicht ausreichend. Aus diesem Grund wurden Fahrerassistenzsysteme mit maschineller Wahrnehmung oder auch Advanced Driver Assistance Systems (ADAS) [8] entwickelt. Im Gegensatz zu konventionellen Fahrerassistenzsysteme, zeichnen sich diese durch die Wahrnehmung und Bewertung des Umfelds außerhalb des Fahrzeugs, mit entsprechenden Sensoren, wie z.B. Kameras, Radar, Ultraschall und die Verwendung externer Informationsquellen aus. Weiterhin wird eine komplexe Signalverarbeitung entsprechend der zu unterstützenden Fahraufgabe ausgeführt. Durch diese Fähigkeiten können angrenzende Fahrspuren und Objekte wahrgenommen werden. Damit ist eine Unterstützung des Fahrers bei einer drohenden Kollision mit einem anderen Fahrzeug oder bei einem Verlassen der Fahrbahn möglich. In Abbildung 1.2 sind die Sensoren eines aktuellen Audi A6 dargestellt. Die Systeme werden in Kapitel 2 näher beschrieben. Die erweiterte Funktionalität der Systeme spiegelt sich in einer höheren Komplexität in Entwurf und Entwicklung wieder. Zudem sind die Haftungsfragen bei einer Markteinführung intelligenter Fahrzeugsicherheitssysteme sehr komplex. Hierzu gehören neue Risiken für die Verbraucher, die Gesellschaft und vor allem für die Hersteller im Bereich der Produkthaftung, z.B. in Zusammenhang mit Rückrufaktionen. Die Risiken in Bezug auf die Produkthaftung sind nicht nur technologischer Natur, sondern erstrecken sich auch auf menschliche Faktoren wie Zuverlässigkeit, Beherrschbarkeit, Verständlichkeit, Berechenbarkeit und Widerstandsfähigkeit gegen Missbrauch. Deshalb muss die Zuverlässigkeit und funktionale Sicherheit von integrierter Elektronik und Software besonderes berücksichtigt werden. Ein guter Überblick über diese Problematik ist in [107] gegeben. Basierend auf den Erfahrungen mit der Norm DIN EN 61508 [73], wurde mit der ISO 26262 [12] eine

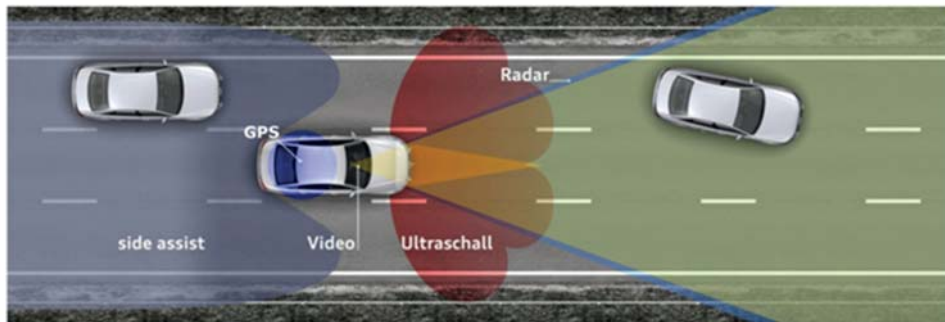


Abbildung 1.2.: Sensorik Audi A6 [19]

automobilspezifische Richtlinie geschaffen, welche einen definierten Entwicklungs- und Absicherungsprozess für Fahrzeugelektronikkomponenten vorschreibt. Mit den Erfahrungen aus den Response-Projekten [8] wurde ein Zusatz zur ISO 26262 geschaffen, welcher insbesondere auf Aspekte für ADAS eingeht.

Eine Absicherung oder Erprobung der Fahrzeugelektronik findet auf Testfahrten und in der Simulation statt. Als Methoden für die simulationsbasierte Erprobung sind Modell-, Software-, Prozessor und Hardware-in-the-Loop seit langem etablierte Methoden. Durch die Einführung von ADAS werden gänzlich neue Anforderungen an diese Simulationsmethoden gestellt. Während es bei einem konventionellen System ausreicht die zur Überwachung des Fahrzeugzustands vorhandene Sensorik zu stimulieren oder zu simulieren, muss für die Erprobung von ADAS die Sensorik zur Umfeldwahrnehmung stimuliert oder simuliert werden. Die Varianz hinsichtlich der benötigten Qualität der Sensordaten reicht von Objekten bis zu Rohdaten eines Kamerasystems. Ein Objekt beschreibt z.B. die Position und Größe des Fahrzeugs, Rohdaten stellen die einzelnen Pixel eines Bilds dar. Weiterhin ist es notwendig das Verhalten des Fahrzeugumfelds durch eine Verkehrs- und Umfeldsimulation zu berechnen. Durch das Konzept der Datenfusion ist es außerdem nicht ausreichend nur einen Sensor zu stimulieren, es müssen alle Sensoren des Gesamtsystems stimuliert werden. Eine Kompensation dieser Lücke durch eine stärkere Durchführung der Tests mit realen Testfahrten ist ebenfalls keine Lösung. Durch die hohe Komplexität der ADAS und die Gefährlichkeit der notwendigen Testsituationen, z.B. die aktive Unfallvermeidung, ist eine Validierung der Funktion in der Realität nur mit großem technischen Aufwand möglich. Durch den großen Testraum, welcher aus einer Vielzahl von Parametern definiert wird, ist die Erprobung wirtschaftlich nicht darstellbar.

Deshalb ist es für die Automobilindustrie von höchster Wichtigkeit verlässliche simulationsbasierte Methoden für die Erprobung von Fahrerassistenz- und aktiven Sicherheitssystemen zu schaffen. Aus dieser Notwendigkeit sind in der jüngsten Vergangenheit verschiedene kommerzielle Simulationsmethoden entstanden. Diese unterscheiden sich hinsichtlich der Modellgenauigkeit deutlich. Durch Schmidt [170] wurden parallel zur dieser Arbeit Anstrengungen unternommen eine Toolkette für die Hardware-in-the-Loop Erprobung von kamerabasierten Fahrerassistenzsystemen zu schaffen.

In dieser Dissertation werden Methoden zur Erprobung von kameraunterstützten Fahrerassistenz- und aktiven Sicherheitssystem entwickelt. Es wird die Definition „kameraunterstützt“ eingeführt, da viele Systeme neben einer Kamera weitere Sensoren zur Wahrnehmung des Umfelds verwenden. Der Schwerpunkt der Anwendung der Methoden soll die Hardware-in-the-Loop Erprobung von vernetzten Fahrerassis-

tenzsystemen sein. Dieser Anwendungsfall sieht einen End-To-End Test der Funktionen vor, d.h. die Sensordaten sollen möglichst am Beginn der Signalverarbeitung eingespeist werden. Im Falle, dass ein Kamerasystem stimuliert werden soll, bietet sich die Erzeugung eines synthetischen Kamerabilds mit Mitteln der Computergrafik an. Da im Vergleich zu anderen Sensoren, wie z.B. Radar, der Aufwand geringer ist. Durch die Einschränkung, dass ein Modell immer eine Vereinfachung der Realität ist, ist es von Wichtigkeit zu untersuchen, welchen Realismusgrad die Simulation für eine plausible Stimulation der Bildverarbeitung aufweisen muss. Weiterhin müssen mögliche Anwendungsfelder für die Simulation in der Erprobung und Entwicklung definiert werden.

Zur Stärkung der methodischen Arbeit in der Computergrafik wurde ein Hardware-in-the-Loop Prüfstand mit der Virtual Test Drive-Toolkette zur Erprobung von Fahrerassistenz- und aktiven Sicherheitssystemen bei der AUDI AG entwickelt. Diese Systeme werden seit 2010 entwicklungsbegleitend eingesetzt. Mittels der Erfahrungen aus der Anwendung konnte die methodische Entwicklung gestärkt werden.

1.2. Zielsetzung

In der Motivation wurde dargelegt, dass die verstärkte Auslagerung von Testfahrten in die Simulation der einzige Weg zur wirtschaftlichen Entwicklung und Erprobung von komplexen Fahrerassistenz- und aktiven Sicherheitssystemen ist. Besonders die Stimulation von Kamerasystemen am Beginn der Wirkungskette, d.h. ab der Bildverarbeitung, ist für tiefgehendes Testkonzept von hoher Wichtigkeit. Daher gilt es vor allem zu klären, welche Genauigkeit simulierte Bilddaten aufweisen müssen, um einen End-To-End Test zu ermöglichen. Daher sollen im Rahmen dieser Arbeit die folgenden Leitfragen diskutiert werden:

- In welchem Maße kann eine Simulation für die Entwicklung und den Test verwendet werden?
- Welchen Grad an Komplexität muss die Simulation aufweisen?
- Welche Aufwände sind zur Erreichung dieser Zielsetzung notwendig?

Zur Lösung dieser Fragestellungen werden im Rahmen dieser Arbeit die folgenden Zielsetzungen im Vorfeld bearbeitet:

- Ziel 1: Analyse und Bewertung von Simulationslösungen für die Erprobung von Fahrerassistenzsystemen.
- Ziel 2: Definition von Anforderungen an eine Simulationsumgebung zur Stimulation von kameraunterstützten Fahrerassistenzsystemen.
- Ziel 3: Experimentelle Evaluierung der Komplexität der Simulation hinsichtlich:
 - Ziel 3.1: Simulation der Beleuchtung
 - Ziel 3.2: Geometrie und Oberflächeneigenschaften von Objekten
 - Ziel 3.3: Synthese von Kamerabildern

- Ziel 3.4: Aufbau von Szenen und Szenarien
- Ziel 4: Entwicklung eines generischen Verfahrens zur Validierung und Kalibrierung der Simulation.
- Ziel 5: Experimenteller Nachweis zur Anwendbarkeit von synthetischen Bilddaten.
- Ziel 6: Aufbau eines Systems für die Hardware-in-the-Loop Erprobung von Fahrerassistenzsystemen.
- Ziel 7: Definition von sinnvollen Anwendungsgebieten und von zukünftigen Verbesserungspotentialen.

1.3. Aufbau der Arbeit

Als Einleitung in die Thematik werden die betrachteten Fahrerassistenz- und aktiven Sicherheitssysteme in Kapitel 2 vorgestellt. Weiterhin werden die Sensoren zur Wahrnehmung des Fahrzeugumfelds beschrieben. Als besonders komplex erweist sich die Signalverarbeitung der Kamerasysteme, da die eigentlichen Messgrößen durch Bildverarbeitungsalgorithmen ermittelt werden. Daher gilt es für einen End-To-End Test die Bildverarbeitung zu stimulieren. Aus diesem Grund werden die zur Realisierung der betrachteten Fahrerassistenzsysteme- und aktiven Sicherheitsfunktionen verwendeten Bildverarbeitungsalgorithmen analysiert. Im Rahmen dieser Analyse werden die von den Algorithmen verwendeten Merkmale diskutiert. Außerdem werden Störgrößen welche zu einer Abschwächung oder Auslöschung dieser Merkmale führen beschrieben.

In Kapitel 3 werden die in der Automobilindustrie verwendeten Entwicklungsmethoden beschrieben. Dazu wird an erster Stelle der Entwicklungsprozess nach dem V-Modell erläutert. Darauf folgend werden die Vor- und Nachteile einer Erprobung im Fahrzeug, einer Wiedergabe von Messdaten und der simulationsbasierten Methoden Model-, Software- und Hardware-in-the-Loop betrachtet. Abschließend wird der Stand der Technik der simulationsbasierten Methoden vorgestellt und eine Bewertung der aktuellen Verfahren durchgeführt.

Auf Basis der vorher definierten Grundlagen werden im dritten Kapitel Anforderungen an eine Simulationsumgebung für den End-to-End Test von kameraunterstützten Systemen definiert. Zur Ableitung der Anforderungen werden die notwendigen Merkmale und Herausforderungen an die Bildverarbeitung zusammengefasst. Diese Anforderungen stellen die Grundlage zur Betrachtung notwendigen Komplexität einer Simulationsumgebung für die Stimulation der Signalverarbeitung dar. Zur Bearbeitung werden diese in die Schwerpunkte Beleuchtungssimulation, Geometrie und Oberflächeneigenschaften von Objekten, Synthese von Kamerabildern und Szenen- und Szenario-Design zusammengefasst.

In Kapitel 4 sind die Ergebnisse der Untersuchungen hinsichtlich der Komplexitätsanforderungen der vorher definierten Schwerpunkte zusammengefasst. Als erstes wird die für die Evaluierung verwendete Simulationsumgebung beschrieben. Anschließend werden Verfahren zur Simulation einer realitätsnahen Beleuchtung betrachtet. Weiterhin werden die Aufwände zur Simulation der Straße, Fahrzeuge und Verkehrsschildern untersucht. In der Computergrafik werden Bilder mittels geometrischer Abbildungen synthetisiert, dieser Prozess unterscheidet sich von der Bildaufnahme mit einer realen Kamera. Daher gilt es den Prozess zur Erzeugung eines möglichst realitätsnahen Bilds zu untersuchen. Abschließend werden

die Anforderungen an den Aufbau von Szenen und Szenarien für die Durchführung von Tests in der Simulation untersucht.

Eine genaue Parametrierung der Modelle ist für die Durchführung von simulationsbasierten Erprobungen von hoher Wichtigkeit. Die Durchführung einer Kalibrierung und Validierung der Modelle ohne eine Rückmeldung für den Anwender gestaltet sich schwierig. In Kapitel 5 wird auf Basis der in dieser Arbeit erlangten Erkenntnisse eine generische Metrik zur Messung der Abweichung zwischen der Simulation und der realen Referenz definiert. Zur Demonstration der Fähigkeit zur Verwendung der Metrik für eine automatische Optimierung von Modellparametern wurde die Metrik in die Simulationsumgebung integriert und eine beispielhafte Optimierung von Parametern durchgeführt.

Zur weiteren Bewertung der Anwendbarkeit einer Kamerasimulation werden in Kapitel 6 die Ergebnisse von experimentellen Untersuchungen vorgestellt. Im ersten Teil werden die bei der Entwicklung eines Fahrzeugererkennungssystems mit simulierten Bilddaten gesammelten Erkenntnisse präsentiert. Darauf folgend wird das Verhalten der Signalverarbeitungskette mit simulierten Bilddaten untersucht.

In Kapitel 7 wird der entwickelte Hardware-in-the-Loop Prüfstand für die Durchführung von End-to-End Tests präsentiert. Dazu wird an erster Stelle das technische Konzept beschrieben. Danach wird die Stimulation der Kamera mittels einer optischen Übertragung untersucht und die Anwendungsmöglichkeiten bewertet. Abschließend werden die mit dem System durchgeführten Tests beschrieben und der Trend der Erprobung dargestellt.

Zum Abschluss dieser Arbeit werden die zu Beginn formulierten Leitfragen mit den in den vorherigen Kapiteln gesammelten Erkenntnissen diskutiert. Abschließend wird ein Ausblick auf Potentiale und zukünftige Entwicklungsrichtungen gegeben.

2. Fahrerassistenz- und aktive Sicherheitssysteme

2.1. Aktuelle Fahrerassistenzsysteme und aktive Sicherheitssysteme

Moderne Straßenfahrzeuge, wie ein Audi A8, sind mit Fahrerassistenz- und aktiven Sicherheitssystemen ausgestattet. Diese Systeme unterstützen den Fahrzeugführer bei der Navigation/Information, Manövrierung und Stabilisation des Fahrzeugs [196]. Die Verantwortung zur Führung des Fahrzeugs bleibt weiterhin beim Fahrer. Die Navigations- oder Informationssysteme unterstützen den Fahrer, z.B. durch eine Routenplanung zur Erreichung des Reiseziels, Darstellung von aktuellen Verkehrskehrzeichen, Visualisierung des Parkvorgangs oder Verbesserung der Nachtsicht. Systeme die den Fahrer bei der Manövrierung des Fahrzeugs in der Längs- und Querrichtung unterstützen werden entsprechend der Definition von [8] als Advanced Driver Assistance Systems (ADAS) oder Fahrerassistenzsysteme mit maschineller Wahrnehmung [177] bezeichnet. Beispiele für diese Systeme sind das Adaptive Cruise Control und die Spurhalteunterstützung. Nach [8] muss ein ADAS die folgenden Eigenschaften erfüllen:

„ADAS sind charakterisiert durch sämtliche der folgenden Eigenschaften:

- *Den Fahrer in der primären Fahraufgabe unterstützen*
- *Aktive Unterstützung zu bieten bei seitlicher und/oder Längskontrolle mit oder ohne Warnung*
- *Erkennen und Bewerten der Fahrzeugumgebung*
- *Verwendung von komplexer Signalverarbeitung*
- *Direkte Interaktion zwischen Fahrer und System“*

Im Folgenden werden die in dieser Arbeit betrachteten Fahrerassistenz- und aktiven Sicherheitssysteme beschrieben. Diese Systeme sind als Serien- oder Mehrausstattung erhältlich. Ein allgemeiner Überblick über Fahrerassistenz- und aktive Sicherheitssysteme aus Forschung oder Industrie kann in [196], [154] und [177] erlangt werden.

2.1.1. Adaptive Cruise Control

Das Adaptive Cruise Control (ACC) oder Abstandsregeltempomat regelt den Abstand zu einem vorausfahrenden Fahrzeug. Falls sich kein Fahrzeug voraus befindet wird eine Wunschgeschwindigkeit geregelt. Während bis vor einigen Jahren die untere Regelgrenze bei 30 km/h lag, ist es inzwischen möglich eine Stop-and-Go Funktion im Stau zu realisieren. Diese Funktionalitäten stellen enorme Anforderungen an

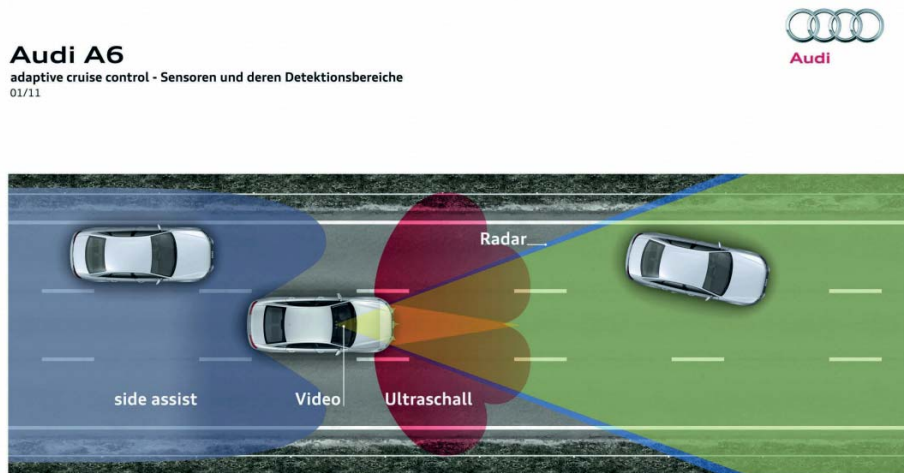


Abbildung 2.1.: Sensoren des Adaptive Cruise Control [19]

die Wahrnehmung des Fahrzeugumfelds, siehe Abbildung 2.1. Für die Erfassung weit entfernter Verkehrsteilnehmer werden Radarsysteme mit einer Reichweite von bis zu 200m verwendet. Für die Erkennung eines nahen Spurwechslers kommen Kamerasysteme zum Einsatz. Für den automatischen Stop & Go-Betrieb ist es notwendig, vor dem Anfahren zu erkennen, ob sich vor dem Fahrzeug Hindernisse befinden. Diese Erkennung wird durch Verwendung der Ultraschallsensoren und der Kamera durchgeführt. Zur Verbesserung der Regelungsfunktionen werden Straßeninformationen auf Basis der Navigationsdaten und zur Erkennung von überholenden Fahrzeugen wird das Heckradar verwendet.

2.1.2. Spurverlassenswarnung/Spurhalteunterstützung

Kollisionen nach einem Fahrspurwechsel oder dem unbeabsichtigten Verlassen der Fahrspur/Fahrbahn zählen zu den häufigsten Unfallursachen. Zur Prävention des unbeabsichtigten Verlassens der Fahrbahn wurden Systeme entwickelt, welche den Fahrer vor dem Verlassen der Fahrbahn warnen oder unterstützend bei der Spurhaltung wirken. Zur Realisierung der Systemfunktionalität werden mit einem Ka-

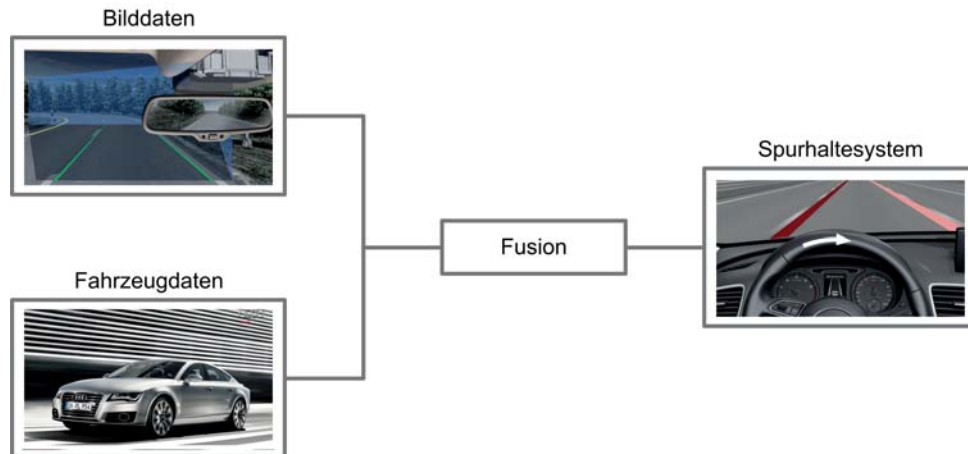


Abbildung 2.2.: Schema Spurhalteunterstützung

merasystem die Fahrspurmarkierungen oder der Übergang zwischen Fahrbahnen und/oder Grünstreifen erkannt, siehe Abbildung 2.2. Durch die Verwendung von Sensorinformationen über den Fahrzeugzustand, können mittels Modellannahmen die Zeit bzw. Distanz bis zum Verlassen der Fahrbahn bestimmt und entsprechende Gegenmaßnahmen ergriffen werden.

2.1.3. Lichtassistenzsysteme

Lichtassistenzsysteme unterstützen den Fahrer durch die Bereitstellung einer optimalen Straßenausleuchtung, siehe Abbildung 2.3. Die einfachste Ausprägung erkennt Lichtquellen und schaltet zwischen Abblend- und Fernlicht um wenn ein vorausfahrendes Fahrzeug erkannt wird. Fortgeschrittene Systeme erkennen mittels einer Kamera die relative Position eines Fahrzeugs und passen den Winkel der Scheinwerfer entsprechend an. Damit wird ein gleitender Übergang zwischen Fern- und Abblendlicht möglich. Zur Anpassung der Lichtverteilung an die aktuelle Straßenart werden prädiktive Streckendaten verwendet, so wird z.B. für eine verbesserte Ausleuchtung an Kreuzungen gesorgt.

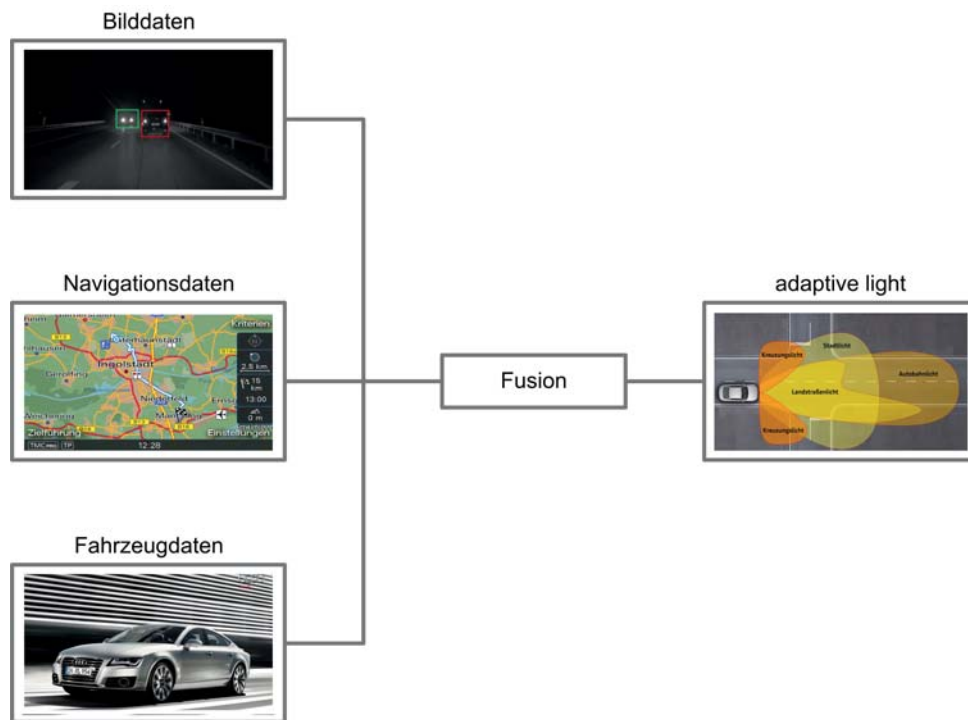


Abbildung 2.3.: Lichtassistenzsysteme

2.1.4. Verkehrszeichenerkennung

Verkehrszeichenanzeigen haben vor allem durch Aftersale Navigationssysteme eine hohe Marktdurchdringung erhalten. Die angezeigten Verkehrszeichen sind in den Navigationsdatenbanken hinterlegt. Daher sind diese nicht immer aktuell und richtig. Im Gegensatz dazu führen die von den Fahrzeugherstellern

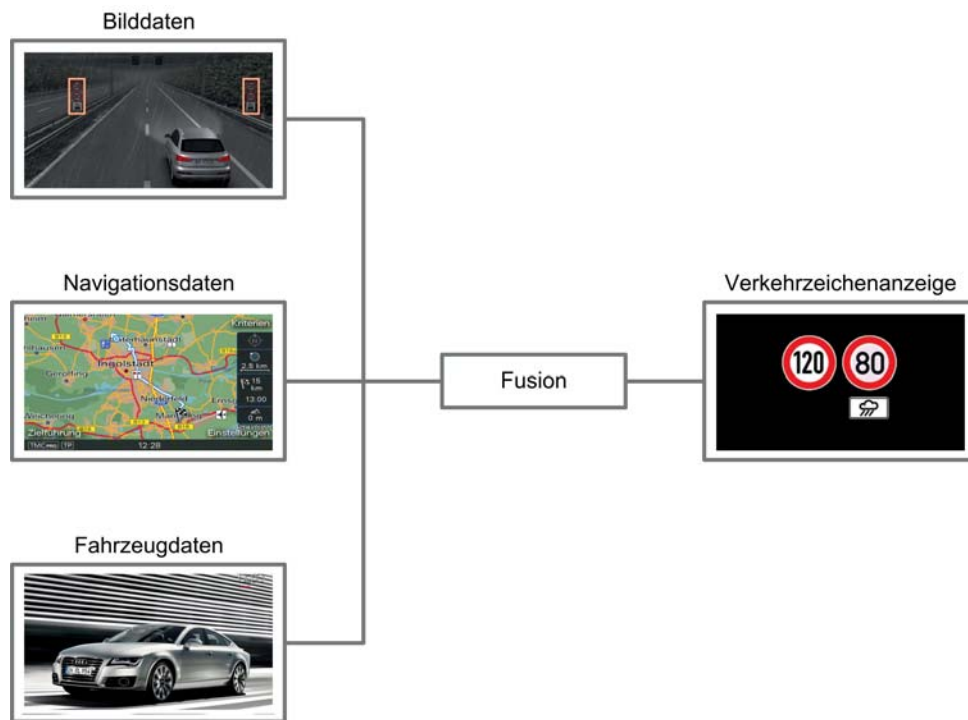


Abbildung 2.4.: Verkehrszeichenerkennung

(OEM's) eingesetzten Systeme eine Datenfusion zwischen Bildern einer Kamera und den Informationen aus der Navigationsdatenbank, siehe Abbildung 2.4, durch. Mittels der Kamera werden Verkehrszeichen im Videobild erkannt. Zur Stabilisierung und Plausibilisierung der Erkennung wird eine Fusion mit prädiktiven Streckendaten durchgeführt.

2.1.5. Kollisionsminderung/-vermeidung

Einen wichtigen Beitrag zur Steigerung der Verkehrssicherheit leisten Systeme zur Kollisionsminderung und -vermeidung. Wird eine kritische Situation erkannt, wird zunächst der Fahrer sensibilisiert und falls dieser nicht reagiert eine Bremsung eingeleitet, um die Unfallschwere zu senken oder Kollision zu vermeiden, siehe Abbildung 2.5. Zur Wahrnehmung des Umfelds wird häufig die Sensorik des Adaptive

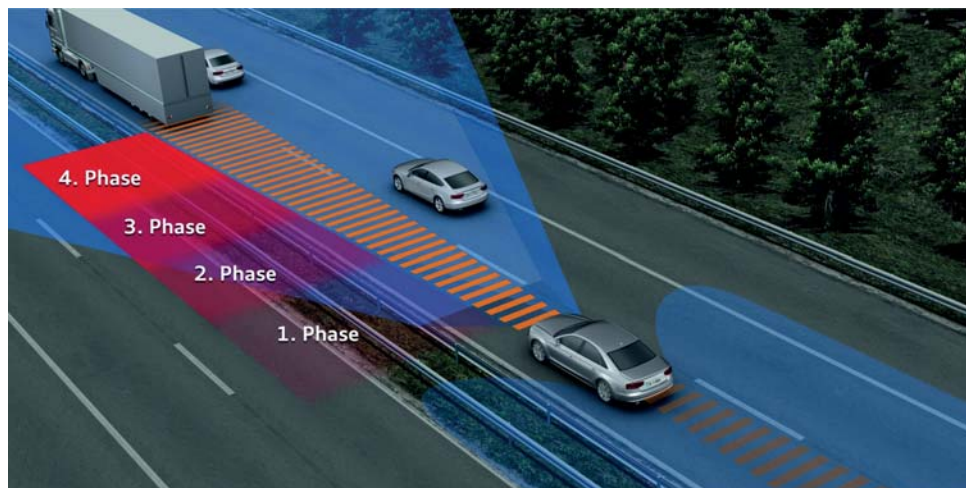


Abbildung 2.5.: Audi PreSense [19]

Cruise Controls verwendet. Der Volvo V60 verwendet ausschließlich den CV-Sensor [43], für die City-Safety Funktion, zur Messung von Relativgeschwindigkeiten und kann Kollision auf Standziele bis 30 km/h komplett vermeiden. Zum Vergleich greifen die Modelle Audi A6/A7/A8 auf ein ausgereifteres Sensor-Set zurück [19]. Neben der Sensoren für den Frontbereich Kamera, Radar und Ultraschall können mittels des Heckradars auch Gegenmaßnahmen bei einem drohenden Auffahrunfall getroffen werden, siehe Abbildung 2.1.

2.2. Sensorik zur Wahrnehmung des Fahrzeugumfelds

Wie im vorherigen Abschnitt herausgearbeitet wurde, werden verschiedenartige Sensoren zur Wahrnehmung des Umfelds durch Fahrerassistenzsysteme verwendet. Häufig werden die Sensordaten einzelner Sensoren mittels einer Sensorfusion aggregiert, um eine hochwertige Wahrnehmung des Umfelds zu erhalten. In diesem Abschnitt wird zunächst das Konzept der Sensorfusion beschrieben. Anschließend werden das Radar, der Ultraschallsensor und die prädiktiven Streckendaten einführend erläutert. Weiterführende Informationen erhält der interessierte Leser in [196] und [177]. Da sich diese Arbeit im Schwerpunkt mit der Entwicklung und Erprobung von kameraunterstützten Fahrerassistenzsystemen beschäftigt, wird der Aufbau einer CMOS-Kamera ausführlich beschrieben.

2.2.1. Sensorfusion

In [98] wird die Sensorfusion als mehrstufiger und vielseitiger Prozess bezeichnet, welcher die automatische Erkennung, Verbindung, Korrelation, Bestimmung und Kombinationen von Daten und Informationen aus mehreren Datenquellen beinhaltet. Im Bereich automotiver Anwendungen wird nach [51] die Sensorfusion verwendet, um Informationslücken eines einzelnen Sensortyps zu schließen, d.h. um eine zuverlässigere und genauere Umfeldinformation zu generieren. Konkret bedeutet dies, wenn verschiedene Sensoren ein Szenario beobachten ist es, aufgrund von verschiedenen Sensortechnologien und Verbauorten möglich, ergänzende, redundante, konträre oder qualitätsverbessernde Informationen zu erhalten. Beispielsweise verwendet der Audi A6 zur Realisierung der Pre-Sense Front Funktion ein Sensor-Set aus zwei Long-Range-Radaren und einer Kamera. Durch dieses System ist es möglich ein mit dem Radar detektiertes Objekt visuell durch die Kamera zu klassifizieren und zu verifizieren. Weiterhin ist es durch die Verwendung von zwei Radaren möglich redundante Messdaten zu erhalten, was die Reduktion von Messfehlern zulässt und die Fehlertoleranz des Gesamtsystems erhöht.

2.2.2. Radar

Die Umfeldwahrnehmung im mittleren Bereich bis $70m$ und im Fernbereich bis ca. $200m$ wird auf Basis von Radar-Systemen (Radio Detection and Ranging) realisiert [196]. Als Messgrößen werden die der Entfernung und der Winkel zwischen Sender und Objekt und die Geschwindigkeit des gemessenen Objekts ermittelt [196]. Der Sensor verwendet elektromagnetische Wellen als Messmedium, welche über Antennen emittiert und empfangen werden. Die elektromagnetischen Wellen werden durch elektrischleitende oder dielektrische Medien reflektiert. Am Häufigsten werden $24GHz$ -Systeme für den Nahbereich und $77GHz$ -Systeme für den Fernbereich verbaut. Die Wahl der Trägerfrequenz hat eine direkte Auswirkung auf die Robustheit gegenüber der atmosphärischen Dämpfung und der realisierbaren Ortsauflösung [196]. Der Öffnungswinkel beträgt in der Regel zwischen 10° und 20° .

Zur Abstands- und Geschwindigkeitsmessung werden in der Fahrzeugtechnik zwei unterschiedliche Messverfahren verwendet. Diese basieren auf der Amplituden-Modulation (Puls-Verfahren) und der Frequenz-Modulation. In beiden Fällen wird eine feste Modulationsfrequenz als Trägersignal verwendet. Im Falle einer pulsbasierten Modulation wird der Abstand zum Objekt anhand einer Time-of-Flight Messung der elektrischen Welle ermittelt, d.h. es wird die Signallaufzeit gemessen. Im Gegensatz dazu wird bei einer Frequenz-Modulation nach dem FMCW-Verfahren (Frequency Modulated Continuous Wave) eine kontinuierliche Welle ausgesendet, deren Wellenlänge linear über die Zeit verändert wird. Durch die Auswertung der Phasenverschiebung zwischen dem gesendeten und dem empfangenen Signal kann auf die Entfernung und die relative Geschwindigkeit geschlossen werden. Das Puls-Verfahren erfordert aufgrund der Notwendigkeit nach einer hohen Abtastung des Signals hochwertige Komponenten, daher sind vermehrt frequenzmodulierte Radare im Einsatz. Die Bestimmung der Geschwindigkeit ist bei beiden Messverfahren auf Basis des Doppler-Effekts möglich [196]. Zur Winkelmessung sind verschiedene Verfahren denkbar. Bei Anwendung des Scanning-Verfahrens wird der Radar-Emitter innerhalb eines Messzyklus durch den Messbereich mechanisch geschwenkt. Die Auflösung der Bewegung beträgt in der Regel 1° . Im Vergleich dazu

verwendet, z.B. das Monopuls-Verfahren einen Radar-Emitter und zwei Radar-Empfänger. Der Winkel kann somit über eine Triangulation der gemessenen Abstände bestimmt werden.

2.2.3. Ultraschallsensor

Ultraschallsensoren werden für die Wahrnehmung des Fahrzeugumfelds bis zu einer Distanz von $5m$ verwendet. Die Sender/Empfänger-Elemente des Ultraschallsensors verwenden zur Distanzmessung Schallwellen mit einer Frequenz von ca. $40\text{ kHz} - 50\text{ kHz}$. Diese breiten sich in der Umgebungsluft des Fahrzeugs aus. Nach [196] sollte zur Realisierung einer lückenlosen Überdeckung der Fahrzeugbreite, jeder Sensor einen horizontalen Öffnungswinkel von ca. 120° bis 140° aufweisen. Der vertikale Öffnungswinkel sollte danach so gering ausgelegt sein, dass Reflektionen von der Fahrbahn zu keinen Pseudo-Hindernissen führen. Daher hat sich in der Praxis ein effektiver vertikaler Öffnungswinkel von ca. 60° bis 70° bewährt. Die Bestimmung des Abstands zu einem Objekt erfolgt auf Basis einer einfachen Laufzeitmessung, welche mit dem Aussenden des Impulses beginnt und mit Empfang der Reflexion endet. Mit Hilfe der bekannten Schallgeschwindigkeit kann der Abstand zum Reflektor bestimmt werden.

Durch den Umstand, dass jeder Sensor nur den Abstand zu einem Reflektor ermittelt, ist dieser als Radius im Messbereich zu interpretieren. Weiterhin ergeben sich geometrische Messungenauigkeiten, welche durch Position, Dimension, Geometrie und Position der Reflektoren beeinflusst werden [196]. Zur Reduktion dieser Fehlerquellen werden, zum einen mehrere Wandler auf jeder Fahrzeugseite verbaut und außerdem eine Trilateration durchgeführt. Das Prinzip einer Trilaterationsmessung besteht darin neben dem aktiven Wandler die benachbarten Wandler in den Empfangsmodus zu schalten und so weitere Abstände zu erhalten. Durch Bestimmung der Schnittpunkte der Abstandsradien zweier benachbarter Sensoren (Trilateration) kann eine genauere Abstandsschätzung durchgeführt werden.

2.2.4. Navigationssystem/prädiktive Streckendaten

Zur Verbesserung der Funktionalität von Fahrerassistenzsystemen werden straßen- oder routenbezogene Daten aus der Datenbank des Navigationssystems verwendet. Diese Daten beinhalten Informationen über den aktuellen Typ der Straße, z.B. Geschwindigkeitsbeschränkungen, Kurvenradien und Steigungen. Als Grundlage zur Erzeugung dieser Daten werden die mittels GPS ermittelte Fahrzeugposition und eine evtl. geplante Route verwendet. Den Assistenzsystemen stehen somit Informationen über den Straßenverlauf zur Verfügung. Weiterhin können Messdaten anderer Sensoren in Abhängigkeit von der Position des Fahrzeugs auf ihre Richtigkeit geprüft werden.

2.2.5. Kamera

Aufgrund der vielseitigen Einsatzmöglichkeiten werden Kamerasysteme für eine Vielzahl von Erkennungsaufgaben verwendet. Diese sind, z.B. Fahrspur-, Fahrzeug-, Personen- und Verkehrszeichenerkennung. Im Vergleich zu aktiv messenden Sensoren wie Radar oder Ultraschallsensoren arbeitet die Kamera auf Basis einer passiven Messung der Bestrahlungsstärke $E \left[\frac{W}{m^2} \right]$. Der Sensor liefert somit keine direkte Messgröße,

welche Rückschlüsse auf den Abstand oder die Geschwindigkeit eines Objekts erlaubt. Das aufgezeichnete Bild setzt sich aus einer Vielzahl von Bildpunkten „Pixel“ zusammen. Der Inhalt eines Pixels ergibt sich aus der einfallenden Bestrahlungsstärke und Wellenlänge des Lichts. Auf Basis eines hochauflösenden Kamerabilds und mit Verfahren der Bildverarbeitung können in den aufgezeichneten Bildern, Objekte erkannt und geometrische Größen ermittelt werden. Durch Verwendung von strukturgebenden Verfahren ist es möglich ein 3D-Bild zu rekonstruieren [166].

In Abbildung 2.6 ist der schematische Aufbau einer Kamera dargestellt [178]. Die Datenspeicher werden in dieser Abbildung ausgelassen. Die Kamera setzt sich im Wesentlichen aus zwei Bestandteilen zusammen, der Optik und einem hochintegrierten Bildsensor. Die von der Umgebung abgestrahlte Strahldichte L_{amb} wird durch das optische System auf den Strahlungsempfänger fokussiert. Dieser erzeugt eine zu der Bestrahlung H und Wellenlänge des Lichts λ proportionale analoge Spannung für jeden Bildpunkte U_{Pixel} . Die Bestrahlung H ergibt sich aus dem Produkt der einfallenden Bestrahlungsstärke E und Belichtungsdauer $t_{Belichtung}$. Die Belichtungsdauer ergibt sich anhand einer intelligenten, auf der Intensität der Pixel basierenden, Regelung. Zur Realisierung einer Nachverstärkung ist häufig ein programmierbarer Verstär-

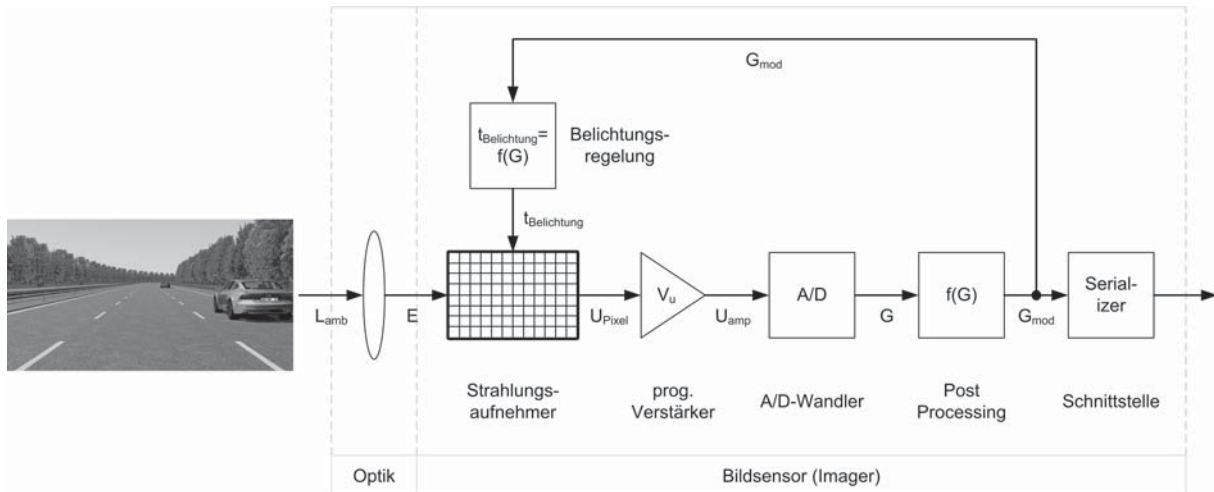


Abbildung 2.6.: Schematische Darstellung des optischen Pfads und wichtiger Kamerabestandteile

ker nachgeschaltet, welcher die eingehende Spannung U_{Pixel} in die verstärkte Spannung U_{amp} wandelt. Anschließend erfolgt eine Wandlung der analogen Spannung in einen diskreten Grauwert g , häufig in einer Auflösung von 10-Bit bzw. 12-Bit. Dieser Wert stellt die Helligkeit eines späteren Pixels dar. Häufig ist nach der Wandlung noch eine einfache Nachverarbeitung geschaltet, diese führt z.B. eine Anpassung des Schwarzwerts oder eine Korrektur von Pixelfehlern durch. Abschließend werden die Grauwerte parallel- oder serialisiert und über eine Schnittstelle an die Bildverarbeitungsfunktionen übertragen. Im Folgenden werden das optische System und der Strahlungsaufnehmer näher beschrieben.

2.2.5.1. Optisches System

Die Transformation eines realen Objekts aus der Objektebene O auf die Bildebene O' des Strahlungsaufnehmers findet durch ein optisches System statt. Ein optisches System bezeichnet nach [141] eine belie-

bige Anzahl von reflektierenden und/oder brechenden Oberflächen beliebiger Krümmung und Brechzahl, die die Richtung von Strahlen, die den Objektpunkt O verlassen, ändern. Die Abbildung erfolgt nach den physikalischen Gesetzen der geometrischen Optik und Wellenoptik. Die Hauptaufgabe des optischen Systems ist die Fokussierung der Objekte im Objektraum und die Steuerung der einfallenden Bestrahlungsstärke E . Nach [60] und [141] kann eine Betrachtung eines optischen Systems als dünne Linse mit hinreichender Genauigkeit durchgeführt werden, wenn die Objektweite ein Vielfaches der Brennweite des optischen Systems ist. Diese Bedingung trifft für den Anwendungsfall im Fahrzeug zu. Daher werden im Folgenden die Abbildungseigenschaften eines optischen Systems mit einer dünnen Linse behandelt [141].

Abbildungseigenschaften

Die Abbildungseigenschaften eines optischen Systems werden mit Hilfe eines Strahlendiagramms, siehe Abbildung 2.7, beschrieben. Die Bezeichnungen wurden aus [141] übernommen und basieren auf der

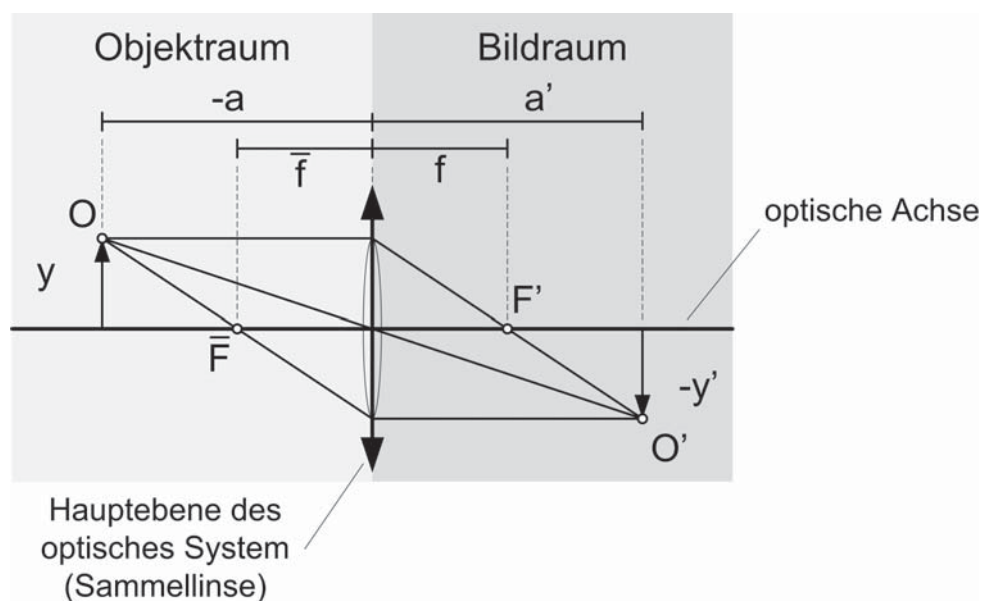


Abbildung 2.7.: Strahlendiagramm eines optischen Systems, nach [141]

DIN1335. Wie ersehen werden kann, werden die im Objektpunkt O divergierenden Strahlen durch die Hauptebene der Sammellinse umgelenkt, so dass diese in der Bildebene O' konvergieren und eine Abbildung des Objekts auf dieser erzeugen. Die senkrecht zur Linse einfallenden Strahlen werden im Brennpunkt F' gebündelt. Durch den Brennpunkt \bar{F} im Objektraum verlaufende Strahlen verlaufen nach der Linse parallel zur optischen Achse. Die durch den Hauptpunkt verlaufenden Strahlen werden nicht verändert. Die Strecke f wird als Brennweite der optischen Linse bezeichnet. Die Strecke a wird als Gegenstandsweite und die Strecke a' als Bildweite bezeichnet. Die Objektgröße ist mit y definiert die Bildgröße mit y' . Für eine scharfe Aufnahme des Objektpunkts O muss sich dieses auf der durch den Abstand a definierten Bildebene befinden. Diese Bedingung wird durch die Linsengleichung Glg. 2.1 definiert:

$$\frac{1}{a'} - \frac{1}{a} = \frac{1}{f} \quad (2.1)$$

Für den Maßstab der Abbildung auf der Bildseite β' folgt somit nach Glg. 2.2.

$$\beta' = \frac{y'}{y} = \frac{a'}{a} \quad (2.2)$$

Auf Basis der Maße kann eine erste Abschätzung zur Positionierung und notwendigen Dimension des Strahlungsaufnehmers getroffen werden.

Sehwinkel

Der Sehwinkel ω einer realen Kamera definiert sich durch die Brennweite f des Objektivs, den Abstand zwischen Objektiv und Strahlungsaufnehmer a und der horizontalen d_h und vertikalen Größe d_v

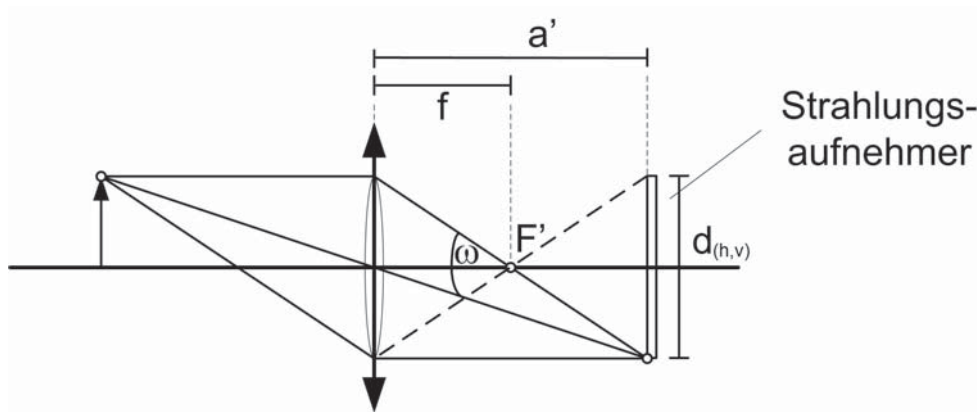


Abbildung 2.8.: Definition des Sehwinkels einer Kamera, nach [141]

des Strahlungsaufnehmers, siehe Abbildung 2.8. Danach ergibt sich für den horizontalen und vertikalen Sehwinkel der Kamera $\omega_{(h,v)}$ mit Glg. 2.3:

$$\omega_{(h,v)} = 2 \cdot \arctan \left(\frac{a' - f}{\frac{d_{(h,v)}}{2}} \right) \quad (2.3)$$

Blende

Durch die Verwendung einer Blende im optischen System kann die eintreffende Bestrahlungsstärke E auf der Bildebene kontrolliert werden. Die Bestrahlungsstärke ist proportional zur Fläche der Blende und umgekehrt proportional zur Fläche des Bilds. Für eine kreisförmige Blende mit dem Durchmesser d ergibt sich unter der Bedingung eines parallelen Strahlenbündels auf der Linse, einer homogenen Lichtverteilung und einer zur Brennweite proportionalen Bildgröße die folgende Bestrahlungsstärke auf der Bildebene.

$$E = \left(\frac{d}{f}\right)^2 \cdot E_0 \quad (2.4)$$

Eine weitere Angabemöglichkeit ist die Verwendung der Blendenzahl k , welche sich aus der Brennweite f und dem Durchmesser der Eintrittspupille d_{EP} definiert:

$$k = \frac{f}{d_{EP}} \quad (2.5)$$

Durch die Blende kommt es zu einer Vignettierung des Bilds, d.h. die Bildbereiche am Rand erscheinen dunkler.

Tiefenschärfe

Während der Bildaufnahme mit einem optischen System tritt der Effekt der Tiefenschärfe auf. Die Erklärung wird auf Basis von [141] erfolgen. Wie in Abbildung 2.9 ersehen werden kann, ist das Objektiv auf den Objektpunkt O im Abstand a_0 fokussiert. Durch diese Annahme werden alle Objektpunkte auf dieser Ebene scharf auf die Bildebene abgebildet. Objektpunkte welche sich nicht auf dieser Ebene befinden, z.B. sich näher oder weiter entfernt zum Objektiv befinden, werden vor oder nach der Bildebene abgebildet. Dieser Sachverhalt ist mit den Strahlengängen der Abstände a_v und a_h dargestellt. Die

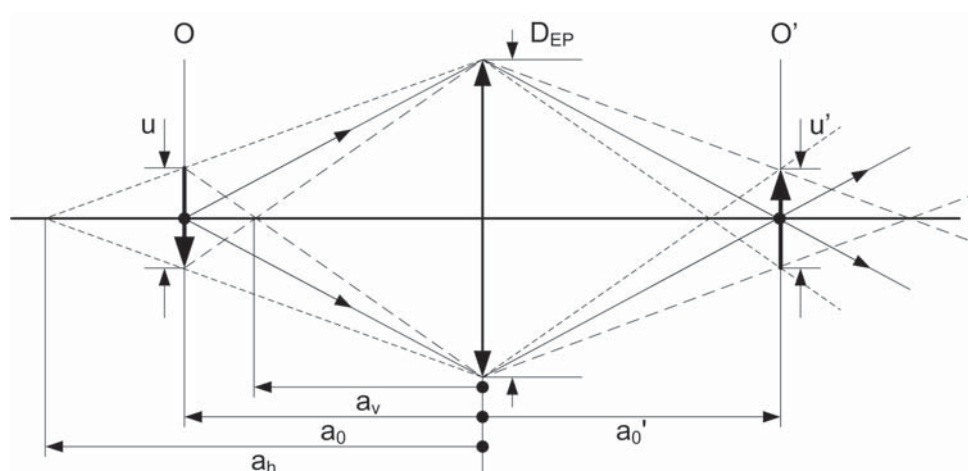


Abbildung 2.9.: Strahlendiagramm zur Definition der Schärfentiefe, nach [141]

Abbildung erfolgt nicht als ein einzelner Bildpunkt sondern als Kreisflächen (Zerstreuungskreise), deren Durchmesser u' sich in Abhängigkeit der Objektpunktdistanz zur Linse verändert. Der Durchmesser u' des Zerstreuungskreises ist solange unproblematisch, solange dieser nicht die Größe eines Bildpunkts auf dem Strahlungsaufnehmer überschreitet. Die Bestimmung des Durchmessers u' kann auf Basis der Abbildungsgleichung, Glg. 2.2, durchgeführt [141] werden. Danach folgt für den Durchmesser des Zerstreuungskreises u'

$$\beta' = \frac{f}{a_0 + f} = \frac{u'}{u} \quad (2.6)$$

und mit,

$$\frac{d_{EP}}{-u} = \frac{-a_v}{a_v - a_0}$$

folgt schließlich für u' , bei einem Objekt vor a_0 :

$$u'_v = \frac{f}{a_0 + f} \left(-d_{EP} + \frac{d_{EP}a_0}{a_v} \right) \quad (2.7)$$

für ein Objekt nach a_0 folgt:

$$u'_h = \frac{f}{a_0 + f} \left(d_{EP} - \frac{d_{EP}a_0}{a_h} \right) \quad (2.8)$$

Abbildungsfehler

Durch reale Linsen werden Abbildungsfehler (Aberrationen) verursacht. Da diese auf den Brechungs- und Reflexionsgesetz basieren sind diese im Randbereich des Bilds stärker ausgeprägt. Dieser Umstand ergibt sich daraus, dass die Sehstrahlen in einem größeren Winkel auftreffen. Durch diese Abweichungen werden die Strahlen nicht genau in einem Zentrum der Linse fokussiert. Eine mathematische Herleitung der Aberrationen ist in [141] vorzufinden. Stiller et. al. [178] und Pedrotti [141] nennen für die Bildverarbeitung folgende wichtige Fehler: Astigmatismus (Punktlosigkeit), Koma (Asymmetriefehler), chromatische Abberation (Farblängsfehler) und Verzeichnungen (Öffnungsfehler). Während die Verzeichnung durch die Bildverarbeitung korrigiert werden kann, müssen alle anderen Abbildungsfehler durch eine gute Konstruktion des optischen Pfads gelöst werden.

Schärfefehler Es gibt drei Arten von Schärfefehlern welche zu einer unscharfen Abbildung führen, siehe Abbildung 2.10:

- (a) Sphärische Aberration (Öffnungsfehler): Bei einem auf der optischen Achse liegenden Objektpunkt werden achsenferne Strahlen stärker gebrochen als achsennahe Strahlen. Daher wird der Punkt als Zerstreungskreis, anstelle eines Bildpunkts abgebildet
- (b) Koma (Asymmetriefehler): Bei einem nicht auf der optischen Achse liegenden Objektpunkt wird das bildseitige Strahlenbündel asymmetrisch. Aufgrund der Erscheinung als Schweif wird dieser Fehler häufig als Koma bezeichnet.
- (c) Astigmatismus (Zweischalenfehler): Befindet sich ein Objektpunkt nicht auf der optischen Achse, so erfolgt eine asymmetrische Abbildung. Der Fehler drückt sich dadurch aus, dass die Strahlen schief auf die Linse treffen.

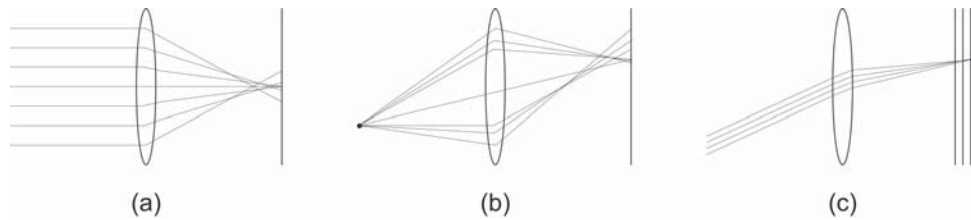


Abbildung 2.10.: Schärfefehler einer Linse, nach [141]

Lagefehler Lagefehler haben eine direkte Auswirkung auf die Geometrie des Bilds, d.h. die Abbildung von Maßstäben. Die Schärfe des Bilds wird nicht verändert.

- (a) Bildwölbung: Durch das optische System wird ein nicht auf der optischen Achse liegender Punkt auf eine gewölbte Fläche abgebildet.
- (b) Verzeichnungen: Diese werden durch Veränderungen des Abbildungsmaßstabs entlang der optischen Achse hervorgerufen. Bei einer Zunahme des Abbildungsmaßstabs ergibt sich der in (b) dargestellte kissenförmige Effekt und bei einer Abnahme des Abbildungsmaßstabs die in (c) dargestellte tonnenförmige Verzeichnung. Weiterhin können die in (d) und (e) dargestellten tangentialen Verzeichnungen auftreten.

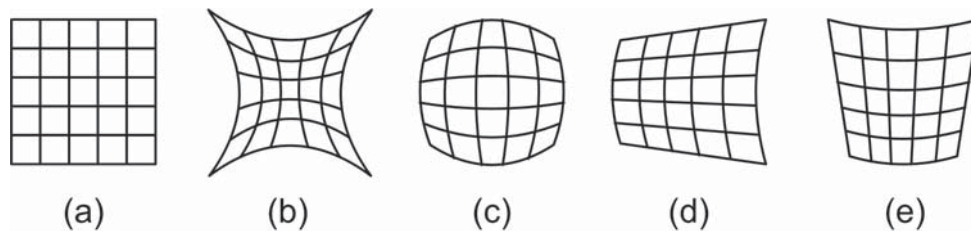


Abbildung 2.11.: Linsenverzeichnungen

(a) unverzeichnetes Abbildung, (b) Kissenförmige radiale Verzeichnung, (c) tonnenförmige radiale Verzeichnung, (d,e) tangentiale Verzeichnung, nach [141]

Farbfehler (chromatische Aberration) Die chromatische Aberration folgt aufgrund einer Dispersion, d.h. die Brechzahl nimmt mit zunehmender Wellenlänge ab. Dies führt dazu, dass die Abbildungen einzelner Wellenlängen der sichtbaren Strahlung nicht mehr auf den gleichen Punkt auf der Bildebene zusammenfallen. Als Ergebnis treten z.B. Farbsäume am Bildrand auf, welche als Farbquerfehler bezeichnet werden. Bei dem Farblängsfehler oder der chromatischen Schnittweitendifferenz liegen die Bilder in verschiedenen Bildebenen.

2.2.5.2. Strahlungsaufnehmer

Zur Wandlung der eingehenden Bestrahlungsstärke E in eine analoge Spannung werden als Strahlungsaufnehmer Halbleitermatrizen mit CCD- (charge-coupled device) oder CMOS- (complementary metal oxide semiconductor) Technik [104] verwendet. Die Wandlung des sichtbaren Lichts geschieht auf Basis des

inneren Photoeffekts [60]. Bei diesem werden durch auftreffende Photonen Elektronen vom Valenz- in das Leitungsband des Halbleitermaterials verschoben. Diese Ladungsträger werden wie bei einem Kondensator in der elektrischen Schaltung gespeichert. Während der Bestrahlung mit Licht werden diese Ladungen integriert. Die Ladungsmenge ist proportional zum Lichteinfall. Die Bauteile weisen üblicherweise eine Sensitivität für Wellenlängen von $\lambda = 400nm$ bis $\lambda = 1100nm$ auf.

CMOS-Bildsensor

In der Fahrzeugtechnik werden zunehmend CMOS-Sensoren verwendet. Daher werden CCD-Sensoren nicht weiter ausgeführt. Ein wichtiger Vorteil von CMOS-Sensoren ist, dass diese nicht am „Blooming“-Effekt leiden. Dieser Effekt tritt bei einer lokalen Überbelichtung auf. Die gesammelten Ladungen eines Pixels breiten sich durch einen Überlauf auf das Nachbar-Pixel aus. Ein wahlfreies Auslesen der Pixel ist möglich, daher können bestimmte Bildbereiche mit einer höheren Frequenz ausgelesen werden (Windowing). Viele Sensoren sind als Aktive Pixel Sensoren (APS - active pixel sensors) realisiert, dies ist ein integrierender Photosensor mit Spannungsauslese [60]. Die Bildaufnahme läuft in drei Stufen ab:

1. Reset der Pixel durch setzen einer definierten Anfangsspannung.
2. Belichtung der Pixel.
3. Auslesen der Spannung der Pixel

Zur Verbesserung des Kontrastes werden häufig Bildsensoren mit einer logarithmischen Abbildung der Bestrahlung H in eine Spannung U verwendet. Das Antwortverhalten ist dem Weber-Fechner-Gesetz des menschlichen Auges nachempfunden [178].

$$U = c \cdot \ln \frac{H}{H_0}$$

Nach [178] ist es damit möglich Strahlungsaufnehmer zu realisieren, welche mehrere Helligkeitspotenzen mit einer konstanten Kontrastaufösung differenzieren. Diese Fähigkeit ist besonders für Fahrzeuganwendungen in Szenen mit, z.B. mit Gegenlicht oder stark schwankendem Licht wichtig.

Farbfilter

Ein Strahlungsaufnehmer misst nicht die Wellenlänge des Lichts sondern die Lichtstärke und besitzt daher eine monochromatische Lichtempfindlichkeit im sichtbaren Bereich, zwischen $\lambda = 350nm$ und $\lambda = 1050nm$. Zur Realisierung von farblichen Aufnahmen werden daher Farbfilter gesetzt, siehe Abbildung 4.44. Diese Farbfilter sind nur für bestimmte Wellenlängen durchlässig. In (a) ist ein normaler monochromatischer Sensor dargestellt. Der häufig für die Erzeugung von Farbbildern verwendete Bayer-Filter ist in (b) abgebildet. Zur Erzeugung eines RGB-Bilds werden für jeden farblichen Pixel die nicht vorhandenen Farben aus den Nachbarpixeln interpoliert. Da das menschliche Auge gegenüber der Farbe grün eine höhere Sensitivität aufweist, sind doppelt so viele Pixel vorhanden. Zur Interpolation des

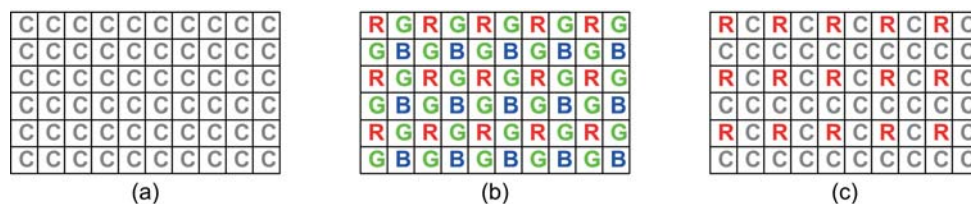


Abbildung 2.12.: Häufige Farbfilter
a) Monochromer-Filter b) RGB-Bayer-Filter c) RCCC-Farbfilter

finalen Bilds stehen eine Vielzahl von möglichen Algorithmen zur Verfügung, welche in der Regel ein intellektuelles Eigentum der Kamerahersteller darstellen. Die reale Ortsauflösung des Bilds ist durch diesen Farbfilter um den Faktor drei geringer als bei einem monochromen Bild.

Im Bereich des maschinellen Sehens haben sich RCCC-Farbfilter (c) durchgesetzt. Diese bieten aufgrund der drei monochromen Pixel eine gute Ortsauflösung. Außerdem können direkt auf dem Roh-Bild Bildverarbeitungsoperationen durchgeführt werden. Durch den Rot-Kanal ist es möglich in einem gewissen Maße farbliche Informationen über die Szene zu erhalten. Damit können z.B. Front- von Rückleuchten oder die Farbe von Schildern oder Markierungen unterschieden werden.

Quantenausbeute

Zur Beschreibung der Umsetzungsfähigkeit der einfallenden Bestrahlung durch den Imager in eine analoge Spannung, wird das Verhältnis der einfallenden Photonen N_{ph} zu erzeugen Elektronen N_e angegeben. Diese Größe wird als Quanteneffizienz bezeichnet und hängt von der Wellenlänge λ ab.

$$\eta(\lambda) = \frac{N_e}{N_{ph}(\lambda)} \quad (2.9)$$

Zur Quantifizierung der Empfindlichkeit wird diese Größe häufig in Datenblättern von Strahlungsaufnehmern angegeben.

Rauschquellen

Während der Bildaufzeichnung gibt es eine Vielzahl von Quellen für ein Bildrauschen [88] [178]:

- Die Photo Response Non-Uniformity (PRNU) resultiert aus Fertigungstoleranzen der verwendeten CMOS-Transistoren und den Aufbau der Bildmatrix. Nicht jeder Transistor bringt bei gleicher Bestrahlung die gleiche Spannung auf. Da die Eigenschaften des PRNU auf physikalischen Prozessen basieren, kann dieses schwer eliminiert werden. Nach [88] ist die PRNU von der Bestrahlung abhängig und besonders bei einer hohen Bestrahlung sehr groß.
- Das Photonen-Rauschen (Shot-Noise) wird durch eine ungleichmäßige Beaufschlagung des Bildsensors mit Photonen verursacht. Nach [88] kann es durch eine Poisson-Verteilung beschrieben werden. Diese Rauschart ist besonders bei kurzen Belichtungszeiten von Relevanz.

- Das Dunkelstrom-Rauschen (DSNU - Dark Signal Non-Uniformity) ist von der Temperatur des Bildsensors abhängig. Durch thermische Energie werden Elektronen in das Leistungsband des Halbleiters angehoben. Dieser Effekt ist vor allem in dunklen Bildbereichen zu sehen und den Helligkeitswerten überlagert. Das Rauschen ist von Quadrat des Dunkelstroms und der Beleuchtungszeit abhängig.
- Das Reset/Read-Out Rauschen (kTC-Noise) entsteht aufgrund einer verbleibenden Spannung im Schaltkreis zwischen Sensor und ADC-Schaltung. Das Read-Out Rauschen verändert sich ausschließlich in Abhängigkeit der Zeit und beinhaltet das Reset-Rauschen, thermisches Rauschen und 1/f-Rauschquellen. Das Reset-Rauschen ist eine besondere Art des thermischen Rauschens und wird durch eine zufällige Bewegung der Ladungsträger verursacht.
- Zur Wandlung der analogen Spannung werden in der Regel 10-Bit oder 12-Bit Analog-Digital-Wandler verwendet. Bei der Konvertierung tritt häufig ein Quantisierungsrauschen auf. Die ist dadurch bedingt, dass die Spannung nur in diskrete Grauwerte umgerechnet werden kann. Liegt die Spannung zwischen zwei Grauwerten wird zum Unteren abgerundet.

2.2.5.3. Belichtungssteuerung

Da eine Kamera nur einen begrenzten Helligkeitsbereich auflösen kann ist ein wichtiger Bestandteil des Strahlungsaufnehmers die Belichtungssteuerung. Durch eine Veränderung der Belichtungszeit $t_{\text{Belichtung}}$ kann vom die Strahlungsaufnehmer integrierte Bestrahlung H verändert werden:

$$H = \int_0^{t_{\text{Belichtung}}} E dt \approx E \cdot t_{\text{Belichtung}}$$

Die physikalische Größe E die auf den Strahlungsaufnehmer auftreffende Belichtungsstärke nach dem optischen System. In den vergangenen Jahren sind eine Vielzahl von Belichtungsstrategien entwickelt worden, welche häufig durch die Systemlieferanten unter Verschluss gehalten werden. In der frei zugänglichen Literatur basiert die Bestimmung der Integrationszeit oder Belichtungszeit häufig auf den Mittelwerten und/oder dem Histogramm des aufgezeichneten Bilds [101] [103] [173].

Bewegungsunschärfe

Durch die endlichen Belichtungszeiten tritt bei einem hohen optischen Fluss die Bewegungsunschärfe oder Motion-Blur auf. Ein in Bildebene abzubildender Objektpunkt bewegt sich während der Integrationszeit über mehrere Positionen in der Bildebene und wird daher mehrfach abgebildet. Da der Abbildungszeitraum dieses Punkts nur einen kurzen Zeitraum innerhalb der Integrationszeit einnimmt erscheint dieser als halb-transparent und wird mit anderen Farbwerten vermischt. Die Kanten von Objekten erscheinen daher als unscharf und auslaufend.

2.3. Bildverarbeitung im Fahrzeug

Bei Betrachtung der Sensorik zur Wahrnehmung des Umfelds wurde festgestellt, dass die Stimulation der Signalverarbeitung für einen End-To-End Test von hoher Wichtigkeit ist. Im Vergleich zu einem Radar oder Ultraschallsystem können die Informationen an den Beginn der Kette eingespeist werden, d.h. in die Optik oder in den Speicher der Kamera. Daher ist es von hoher Wichtigkeit zielgerichtet zu analysieren welche Merkmale ein simuliertes Bild beinhalten muss, damit dieses die Signalverarbeitung erfolgreich durchläuft und es zu einer richtigen Entscheidung in späteren Stufen der Bildverarbeitung kommt. Deshalb werden im folgenden Abschnitt der aktuelle Stand der Technik im Bereich der Wahrnehmung von Fahrspuren, Fahrzeugen und Verkehrszeichen betrachtet und die getroffenen Annahmen für die Erkennung untersucht. Basierend auf dieser Analyse werden Anforderungen an die Modellgüte der Simulation definiert. Einleitend wird die Signalverarbeitung in der Bildverarbeitung beschrieben.

2.3.1. Signalverarbeitung in der Bildverarbeitung

Zur Erlangung eines allgemeinen Verständnisses der späteren beschriebenen speziellen Abläufe, wird anhand der in [130] und [92] dargelegten Beschreibungen, eine kurze Einführung in den Prozess der Signalverarbeitung gegeben. Zur Erlangung eines erweiterten Verständnisses sei auf die bereits erwähnten Standardwerke verwiesen. In Abbildung 2.13 ist der funktionale Ablauf der Bildverarbeitung dargestellt,

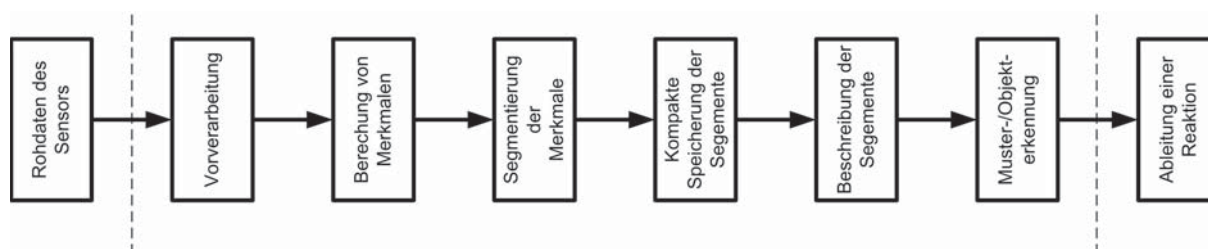


Abbildung 2.13.: Ablauf der Bildverarbeitung, nach [130]

nach [130]. Nach der Aufzeichnung der Sensordaten ist eine Vorverarbeitung notwendig. Im Rahmen dieser werden Operationen zur Bildverbesserung und Fehlerreduktion durchgeführt, wie z.B. Reduktion von Bildrauschen, Entfernung von Verzeichnungen, Kontrastverbesserung und Helligkeitsanpassungen [130]. Anschließend gilt es eine Segmentierung der relevanten Bildinhalte anhand von Merkmalen durchzuführen. Diese lassen sich in bildpunktbezogene Merkmale (Helligkeit und Farbton), lokale Merkmale (Kontrast, Gradient oder Texturmaße) oder temporale Merkmale (z.B. optischen Fluss) differenzieren. Aus den segmentierten Merkmalsbildern gilt es wichtige Merkmale für die spätere Klassifikation zu extrahieren. Im Falle einer Spurerkennung können dies z.B. Kanten mit einem bestimmten Winkel oder zusammenhängende Liniensegmente sein. Im Falle einer Erkennung von Fahrzeugen sind dies, die Symmetrie des Objektes oder auch das Verhältnis von Breite zu Höhe. Im letzten Schritt wird anhand der extrahierten Merkmale eine Muster- oder Objekterkennung durchgeführt, welche die Grundlage für Funktionen außerhalb der Bildverarbeitung ist. Zur Durchführung der Klassifikation kommen häufig heuristische Verfahren, Verfahren des maschinellen Lernens oder Expertensysteme zum Einsatz, vgl. [92].

2.3.2. Spurerkennung

In der Einleitung wurde beschrieben, dass eine Vielzahl der Unfälle durch ein Verlassen der Fahrbahn geschieht. Daher hat sich die Forschung und Entwicklung in den letzten Jahrzehnten auf die Detektion von Fahrspuren konzentriert. Die Zielsetzung ist es dem Fahrer Funktionen bereitzustellen welche durch Warnungen und aktive Regeleingriffe diesen vor einem Verlassen der Fahrbahn bewahren. In Abbildung 2.14 ist der funktionale Ablauf der Spurerkennung nach [108] dargestellt.

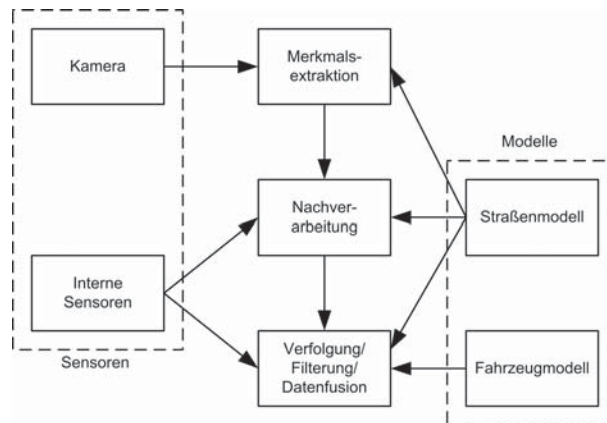


Abbildung 2.14.: Ablauf der Spurerkennung, nach [108]

2.3.2.1. Extraktion von Merkmalen

Nachdem das vorausbefindliche Verkehrsumfeld mittels eines Kamerasystems aufgezeichnet wurde, werden aus dem Bild Merkmale zur Beschreibung der Fahrspuren extrahiert. Die Algorithmen müssen eine hohe Robustheit aufweisen, um mit verschiedenen und wechselnden Fahrbahnoberflächen, Markierungsarten und -farben arbeiten zu können. Nach den Ausführungen von [41] und [108] können die Verfahren in kanten-, region- und frequenzbasierte Ansätze eingeteilt werden.

Kantenbasierte Ansätze

Dieser Ansatz arbeitet sehr gut in Situationen in welchen die Markierung gut vom Hintergrund trennbar ist, z.B. bei durchgezogenen und gestrichelten Linien. Die Qualität der Kante wird maßgeblich durch den Kontrast zwischen Markierung und Fahrbahn beeinflusst. Daher wird die Erkennungsleistung durch Abschattungen negativ beeinflusst und es können Pseudo-Kanten erzeugt werden. Außerdem können Oberflächenübergänge in vertikaler oder diagonaler Richtung zu ähnlichen Fehlerkennungen führen. Dennoch arbeiten eine Vielzahl der Algorithmen aufgrund einfacher Implementierung und Interpretierbarkeit der Ergebnisse mit kantenbasierten Merkmalen. Die Berechnung der Merkmale findet z.B. mit einem Sobel-Filter [130] statt.

Regionbasierte Ansätze

Als Annahme für diesen Ansatz wird von der Straße als eine zusammenhängende Fläche mit einer ähnlichen Erscheinung ausgegangen. Im innerstädtischen Bereich können asphaltierte Bürgersteige als Fahrbahn erkannt werden [41]. Weiterhin besteht die Möglichkeit, dass die Verfahren bei Straßen mit unterschiedlichen Oberflächen ein Fehlverhalten aufweisen. Zur Berechnung der Fläche kann ein einfacher Seed-Fill-Algorithmus verwendet werden [174].

Frequenzbasierte Ansätze

Zur Erkennung von Fahrspuren wird von der Annahme ausgegangen, dass die Fahrspur im Frequenzraum charakteristische Merkmale aufweist. In der Praxis haben sich die Verfahren als besonders robust im Umgang mit fehlerhaften Kanten gezeigt, aber leiden unter Schwächen in komplexen Schatten Situationen. Als Technik für die Transformation in den Frequenzraum wird die schnelle Fourier-Transformation (FTT) oder Kosinus-Transformation verwendet [130].

2.3.2.2. Segmentierung

Im Rahmen der Nachverarbeitung gilt es die extrahierten Merkmale durch das Einbringen von Annahmen über das Straßenumfeld und Fahrzeug zu verbessern, und die relevanten von den nicht relevanten Merkmalen zu trennen. Nach McCall und Trivedi [108] kommen dafür häufig Verfahren wie, z.B. die Hough-Transformation [130] und Kantenorientierungsfiler zum Einsatz. Weiterhin werden Annahmen über die Straße zur Auswahl der relevanten Features gemacht, nach Pätzold et. al [148] und McCall und Trivedi [108] folgt:

- Fahrbahnmarkierungen sind im Bild in der Regel zusammenhängende oder unterbrochene Linien.
- Fahrbahnmarkierungen laufen auf einen gemeinsamen Fluchtpunkt zu und sind parallel.
- Markierungen weisen eine ähnliche Farbe und Helligkeit auf.
- Die Geometrie der Fahrbahn ist auf Autobahnen vorwiegend durch Klothoiden beschreibbar. Im Rahmen des Stadtverkehrs ist die Fahrbahnführung willkürlich und aktueller Gegenstand der Forschung [41].
- Die Straßen- und Spurbreiten sind kausal und lokal konstant.
- Die Erscheinung der Markierungen ist durch den lokalen Gesetzgeber definiert.
- Die Oberfläche der Straße kann als Ebene angenommen werden.

2.3.2.3. Spurverfolgung

Zur Verbesserung der Leistungsfähigkeit, auch unter schwierigen Bedingungen, wird anhand der ermittelten Linienkorrespondenzen eine Spurverfolgung (Tracking) durchgeführt. Zur Realisierung werden Kalman- oder Partikel-Filter in Kombination mit einem Straßen- oder Fahrzeugmodell verwendet, siehe [41] und [108]. Das bekannteste Modell wurde von Dickmanns und Mysliwetz [50] eingeführt und beschreibt die Straßen mittels einer definierten Klothoide. Diese Annahme ist für Autobahnen und gut ausgebaute Landstraßen akzeptabel. Im innerstädtischen Bereich sind andere Beschreibungsformen für den Straßenverlauf relevant, häufig werden Splines verwendet [41].

2.3.3. Fahrzeugdetektion

Sun et. al [179] geben einen umfassenden Überblick über Ansätze zur Fahrzeugdetektion. Im Rahmen ihrer Ausführungen beschreiben sie den Klassifikationsprozess als zweistufiges Erkennungsproblem, siehe Abbildung 2.15. Während der Erzeugung von Hypothesen wird nach möglichen Fahrzeugen im Videobild

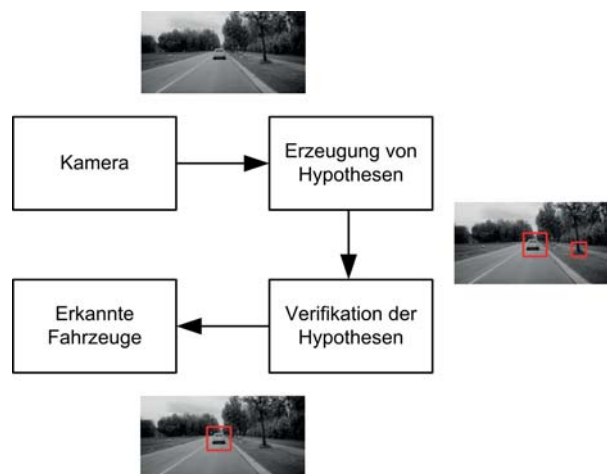


Abbildung 2.15.: Ablauf Fahrzeugerkennung, nach [179]

gesucht. Die anschließende Verifikation dient dazu, Hypothesen anhand von Thesen weitergehend zu prüfen und einzuordnen.

2.3.3.1. Hypothesenerzeugung bei Tag

Im Folgenden werden die verwendeten Verfahren zur Erzeugung von Hypothesen während des Tages dargestellt. Nach [179] ist eine Einteilung in wissens-, stereo- und bewegungsbasierte Ansätze möglich.

Wissensbasierte Ansätze

Zur Erzeugung einer Hypothese kann bekanntes Vorwissen über das zu erkennende Objekt verwendet werden. Die folgenden Merkmale werden in der Literatur für die Erzeugung einer Hypothese herangezogen:

- Symmetrie: Fahrzeuge weisen in der Regel eine hohe Symmetrie in der Hochachse auf.
- Farbe: Fahrzeuge haben in der Regel große einfarbige Flächen.
- Schatten: einige Ansätze verwenden Schatten unter dem Fahrzeug als Merkmal. Als Begründung wird angeführt, dass in der Regel die Straßenoberfläche unter dem Fahrzeug dunkler ist als der Rest der Szene.
- Ecken: Es wird die Annahme verwendet, dass ein Fahrzeug vier Eckpunkte besitzt.
- Vertikale und horizontale Kanten: reale Fahrzeuge besitzen in der Regel viele horizontale oder vertikale Strukturen.

Stereo- und bewegungsbasierte Ansätze

Zur stereobasierten Fahrzeugerkennung werden zwei Kameras eingesetzt deren intrinsische Kalibrierung und Position bekannt sind. Zur Erkennung von Objekten werden die Disparitäten [171] zwischen dem linken und rechten Kamerabild verwendet. Anhand dieser wird ein Disparitätsbild mit Tiefeninformationen erzeugt. Dafür gilt es allerdings eine Korrespondenz zwischen einzelnen Pixeln herzustellen. Ein weiterer Ansatz besteht in der Anwendung einer inversen perspektivischen Abbildung auf eine Ebene mit $z = 0$. Durch die Rückprojektion des ersten Bilds in das zweite Bild ist es möglich, Objekte mit einer Höhe größer Null zu finden, da durch die Höhe eine Differenz im fusionierten Bild entsteht. Dieser Ansatz ist problematisch bei sich wiederholenden Flächen. Bewegungsbasierte Ansätze verwenden den optischen Fluss [171] zur Detektion von Relativbewegungen zur Kamera. Die Idee ist Intensitätsschwankungen im Bild einen Vektor auf der Bildebene zuzuordnen welcher die Bewegungsrichtung darstellt. Dieser stellt die Relativbewegung zwischen der Szene und der Kamera dar. Bei entgegenkommenden Objekten zeigen die Vektoren in Richtung der Kamera, bei sich entfernenden Objekten in Fahrtrichtung des eigenen Fahrzeugs.

2.3.3.2. Hypothesenerzeugung bei Dunkelheit

Als Hauptmerkmal für die Erzeugung von Hypothesen zur Detektion von Fahrzeugen bei Dunkelheit, werden die Frontscheinwerfer oder Rückleuchten der Fahrzeuge verwendet. Der Erkennungsprozess ist meistens zweistufig. In einem ersten Schritt wird das Eingangsbild durch Definition von plausiblen Bildregionen und mittels eines adaptiven Schwellwertverfahrens hinsichtlich Lichtquellen segmentiert. Anschließend werden die folgenden Annahmen zur Generierung von Hypothesen verwendet [42] [15] [16] [134]:

- Lichtquellen von Fahrzeugen gehören zu den hellsten Objekten in der Szene.

- Die Frontscheinwerfer sind weiß und die Heckleuchten sind rot.
- Durch die Brechung des Lichts in der Linse besitzen Lichtquellen eine Korona.
- Die Fahrzeuglichter haben eine bestimmte Form, Größe und Position.
- Lichtquellen treten bei einem intakten Fahrzeug immer paarweise auf.
- Reflexionen durch Schilder oder Leitpfosten haben eine ähnliche Intensität wie Fahrzeuglichter.

2.3.3.3. Verifikation der Hypothesen

Zur Verifikation der erzeugten Hypothesen werden template- und erscheinungsbasierte Ansätze verwendet. Dabei gilt es, die Hypothesen in „Fahrzeuge“ und „Nichtfahrzeuge“ zu unterscheiden.

Templatebasierte Verfahren

Diese nutzen Referenzmuster (Templates) der Fahrzeugklasse und führen eine Korrelation zwischen der Hypothese und dem Template aus. Einige Ansätze verwenden das Kennzeichen oder die Heckscheiben als Merkmale andere gehen von der Annahme aus, dass das Fahrzeug in der Front- bzw. Heckansicht die Form eines „U“ besitzt. Die Templates werden häufig aus der visuellen Erscheinung oder Geometrie des Fahrzeugs abgeleitet. Da ein Fahrzeug generell symmetrisch ist, kann es z.B. durch eine rechteckige Bounding-Box beschrieben werden, welche ein bestimmtes Höhen- und Breitenverhältnis aufweist.

Erscheinungsbasierte Verfahren

Im Gegensatz dazu verwenden erscheinungsbasierte Verfahren Ansätze des maschinellen Lernens, Neuronale Netze oder Support Vector Machines. Die Merkmale einer Fahrzeugklasse werden anhand eines Trainingsdatensatzes erlernt. Bei der Auswahl des Trainingsdatensatzes ist auf eine ausgewogene Zusammensetzung aller möglichen Fahrzeugerscheinungsformen und nicht Fahrzeuge zu achten. Die Qualität und Aussagekraft von verwendbaren Merkmalen ändert sich durch die Ausrichtung der Kamera, Position und Ausrichtung der Fahrzeuge und Beleuchtung der Szene.

2.3.3.4. Tracking

Zur Erhöhung der Robustheit wird ein Tracking verwendet. Dies ist möglich, da davon ausgegangen werden kann, dass Fahrzeuge nicht plötzlich im Bild erscheinen und wieder verschwinden. Als Tracker werden, z.B. Kalman-Filter verwendet [130].

2.3.4. Verkehrszeichenerkennung

Die Detektion und Interpretation von Verkehrszeichen ist im Vergleich zu den bereits vorgestellten Verfahren zur Fahrzeugdetektion deutlich komplexer. Es gilt nicht nur, eine Klassifizierung in Verkehrsschild und kein Verkehrsschild durchzuführen, es muss auch die Bedeutung erkannt werden. In den Arbeiten von Brkic [36] und Fu et. al. [71] ist ein allgemeiner und umfassender Überblick über den Stand der Technik gegeben. Der Ablauf ist in Abbildung 2.16 dargestellt.

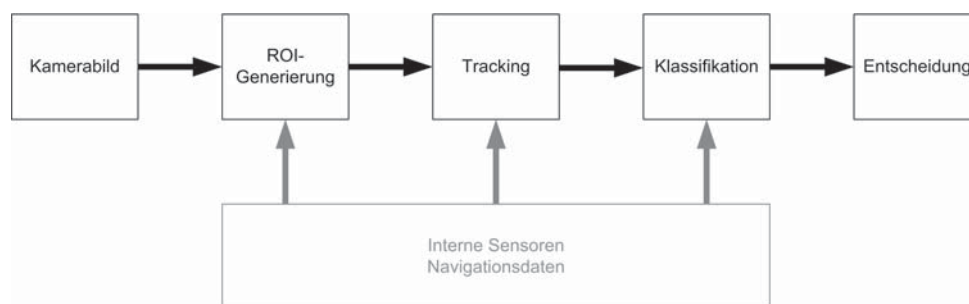


Abbildung 2.16.: Schema der Verkehrszeichenerkennung, nach [71]

Nachdem das Kamerabild aufgezeichnet wurde, werden Hypothesen (ROI - Region of Interest) generiert welche häufig temporal verfolgt werden. Im Vergleich zur Fahrzeugerkennung ist dieser Trackingprozess der Klassifikation vorgeschaltet, da die Klassifikation deutlich komplexer ist. Nach Durchführung der Klassifikation wird das Ergebnis an die weiteren Funktionsbestandteile übergeben. Zur Stabilisierung der einzelnen Verarbeitungsschritte werden häufig Informationen der internen Fahrzeugsensorik oder streckenbezogene Daten aus dem Navigationssystem einbezogen.

2.3.4.1. Hypothesen Generierung & Klassifikation

Durch Brkic [36] werden die Verfahren zur Hypothesen Generierung in farb-, form- und lernbasierte Ansätze eingeteilt. Diese werden häufig zur Steigerung der Robustheit der Klassifikation kombiniert. Weiterhin werden diese Techniken auch für die Klassifikation der Verkehrsschilder verwendet [71].

Farbbasierte Erkennung

Der Ansatz der farbbasierten Erkennung besteht in der Annahme, dass die Farben eines Verkehrszeichens sehr gut von der Umgebung getrennt werden können. Daher wird eine Segmentierung anhand der Farbe des Verkehrsschildes durchgeführt. Der einfachste Ansatz besteht in einer Grenzwertsegmentierung. Dieses Verfahren ist jedoch sehr sensitiv hinsichtlich Belichtungsschwankungen, Schatten und Wetterbedingungen. Daher wird dieser Ansatz häufig als Vorverarbeitungsschritt für weitere Verfahren verwendet.

Formbasierte Erkennung

Zur formbasierten Erkennung von Verkehrszeichen wird häufig eine Hough-Transformation, Eckendetektion oder ein Template-Matching verwendet. Die Hough-Transformation verwendet ein Kantenbild als Ausgangsbasis. Jedes Pixel einer Kante stimmt für ein Objektzentrum zu welchem es zu gehören würde, wenn es auf der Kante des Schilds liegt. Das eckenbasierte Verfahren sucht ausgehend von einer Startecke nach weiteren Ecken in vordefinierten Bereichen einer Region. Die templatebasierten Verfahren verwenden eine Referenzform, z.B. einen Kreis, zur Suche im aktuellen Bild. Zur Auffindung der optimalen Ähnlichkeit wird das Template rotiert, skaliert und über das Bild bewegt.

Lernbasierte Ansätze

Verfahren mit lernbasierten Ansätzen verwenden trainierte Support Vector Machines oder mit Ada-Boost trainierte Kaskaden an Klassifikatoren. Die Verfahren sind auf die spezifische Form des Verkehrsschildes trainiert.

2.3.5. Herausforderungen

In der betrachteten Literatur nannten die Autoren eine Reihe von Herausforderungen für die Bildverarbeitung.

Eine große Herausforderung an die Segmentierung und Klassifikation sind Hintergründe mit einer ähnlichen Intensität und Struktur, wie die des zu erkennenden Objekts. Durch diese werden störende Kanten und Ecken erzeugt.

Ein weiterer nicht vernachlässigbarer Aspekt sind die Lichtbedingungen, welche sich durch die Tageszeit und die Wetterbedingungen verändern. Durch Auftreten von starken Schatten kann sich die Erkennungsleistung verschlechtern. Die sich ändernden Anteile der spektralen Strahlung des Sonnenlichts, durch die Dämpfung und Streuung der Atmosphäre, können zu einer anderen Farbwahrnehmung führen als ursprünglich erwartet.

Weiterhin können Verdeckungen durch andere Objekte, Verkehrsteilnehmer oder Wetterbedingungen auftreten.

Die Erscheinung eines bekannten Objekts kann sich durch eine Deformation oder Alterung verändern. Im Falle von Verkehrsschildern können länderspezifische Abweichungen in der Erscheinung auftreten.

Durch den Umstand, dass die Kamera aus einem sich bewegenden Fahrzeug Bilder aufzeichnet, kann bei langen Belichtungszeiten Motion-Blur entstehen. Weiterhin können Vibrationen oder Wank- und Nickbewegungen einen negativen Einfluss auf die Qualität des Bilds haben.

2.3.6. Zusammenfassung

In diesen Abschnitt wurde der funktionale Ablauf von Verfahren zur Spur-, Fahrzeug-, Verkehrszeichenerkennung betrachtet. Dafür wurden die Ausführungen aus [108], [41], [179], [36] und [71] herangezogen.

Für die spätere Erstellung der Simulation sind besonders die von der Bildverarbeitung verwendeten Merkmale von großer Relevanz. Für die Spurerkennung sind die Geometrie, die Oberfläche der Straße und der Straßenmarkierungen von Wichtigkeit. Im Vergleich dazu müssen Objekte (PKW, LKW, Fußgänger, Verkehrsschilder) eine möglichst realitätsnahe Kontur, Symmetrie und Textur aufweisen. Weiterhin muss eine der Realität entsprechende Beleuchtung des Objekts sichergestellt sein.

Dementsprechend haben verändernde atmosphärische Bedingungen oder schlechtes Wetter einen störenden Effekt auf die Leistungsfähigkeit, da diese häufig die relevanten Merkmale beeinflussen. Die Erscheinung von Objekten wird maßgeblich durch ungünstige Lichtbedingungen oder Schatten beeinflusst. Verdeckungen des Objekts und ungünstige Strukturen im Bild treten durch die Anordnung der Szene während des Tests auf. Weiterhin wird die Wahrnehmung durch eine inkonsistente Bewegung der Bildinhalte erschwert. Diese kann das Resultat der Bewegung des eigenen Fahrzeugs sein, z.B. durch Bodenwellen.

3. Entwicklungsmethoden für Fahrzeugelektronik

In diesem Abschnitt werden die Methoden für die Entwicklung- und Erprobung von Fahrzeugelektronik vorgestellt. Als Einleitung in die Thematik wird der Software-Entwicklungsprozess in der Automobilindustrie nach [187] [85] [172] dargestellt. Anschließend werden die für die System-Integration von Fahrerassistenzsystemen eingesetzten technischen Lösungen vorgestellt und Vor- und Nachteile aufgezeigt. Abschließend werden Simulationsumgebungen für die X-in-the-Loop Erprobung dargestellt und bewertet.

3.1. Entwicklungsprozess in der Automobilindustrie

Die Entwicklung von Elektronikkomponenten in der Automobilindustrie orientiert sich an dem V-Modell [54], welches seine Wurzeln im Bereich der Luft- und Raumfahrttechnik hat. Mit der Implementierung der Norm ISO26262 [12] wird dieses Modell in der Automobilindustrie vorgeschrieben. Auf der linken Seite des V sind die Anforderungen/Entwicklungsumfänge dargestellt, auf der rechten Seite befindet sich ein korrespondierender Prüfschritt, siehe Abbildung 3.1.

Die Grundlage der Entwicklung eines elektrischen Fahrzeugsystems stellen die im Lastenheft formulierten Anforderungen dar. Auf Basis dieser wird zunächst eine funktionale Analyse ausgeführt. Die funktionale Analyse ist die Grundlage für den Entwurf des Systems und die Durchführung der Spezifikation für die zu entwickelnde Software. Anschließend werden die Anforderungen an die einzelnen Module beschrieben. Mittels der Modulspezifikation wird die Implementierung der einzelnen Softwaremodule durchgeführt.

Nachdem die Implementierung der Softwaremodule abgeschlossen ist, wird deren Verhalten hinsichtlich der Erfüllung der in der Modulspezifikation definierten Umfänge getestet. Wenn die einzelnen Softwaremodule erfolgreich getestet wurden, werden diese entsprechend des Softwareentwurfs auf dem entsprechenden Steuergerät zu einer gesamtheitlichen Software integriert. Zur Verifikation des Steuergeräteverhaltens wird dieses auf Basis einer Software-Testspezifikation getestet. Nach erfolgreicher Verifikation wird die Integration des Systems zur Realisierung einzelner Fahrzeugsysteme ausgeführt. Anschließend gilt es den aufgebauten Steuergeräteverbund hinsichtlich der Systemfunktionalität zu verifizieren. Nach erfolgreicher Verifikation, werden die Fahrzeugsysteme in die Prototypenfahrzeuge integriert und eine Applikation und ein Abnahmetest hinsichtlich der Erfüllung der Anforderungen an das Gesamtfahrzeug durchgeführt.

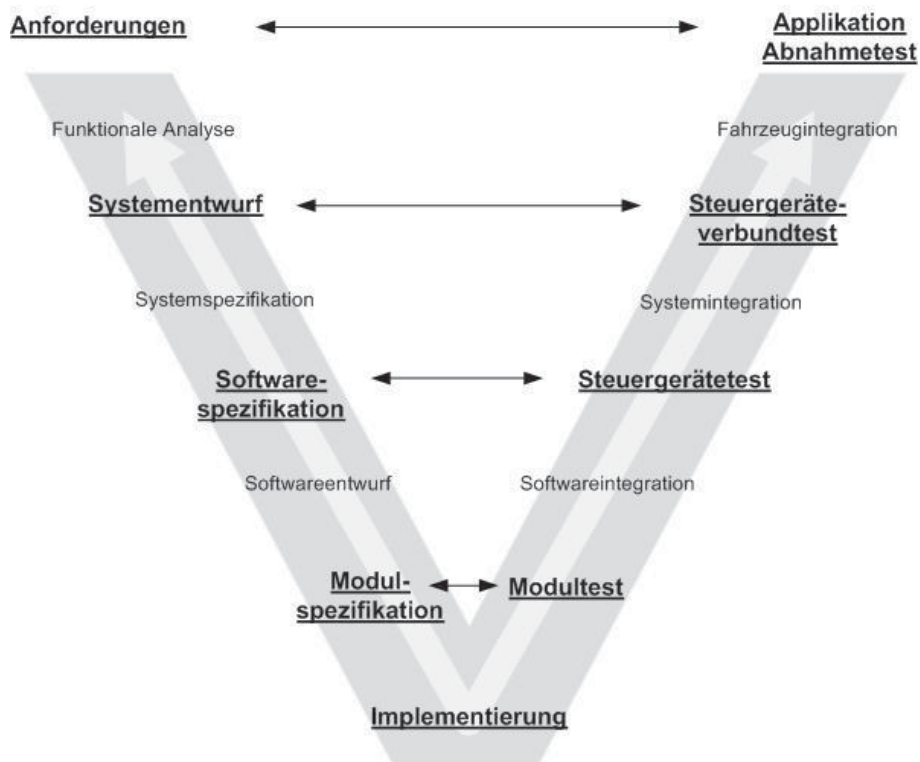


Abbildung 3.1.: Das V-Modell, nach [187]

3.2. Erprobung von Fahrerassistenzsystemen

Nachdem im vorherigen Abschnitt der Entwicklungsprozess von Elektronikkomponenten in der Automobilindustrie nach dem V-Modell beschrieben wurde, werden Methoden zur Erprobung von Fahrerassistenzsystemen erklärt. In Abbildung 3.2 sind Methoden für System-Integration von Fahrerassistenzsystemen mit Vor- und Nachteilen dargestellt.

Fahrzeugerprobung

Die historisch gewachsene Fahrzeugerprobung wird heutzutage sowohl für die Applikation, die Abnahmetests und den Test von Steuergeräten und Software-Modulen eingesetzt. Die sich ergebenden Vorteile einer Fahrzeugerprobung liegen in dem realitätsnahen Testsystem, dem Fahrzeug, und dem Verkehrsumfeld. Nachteilig erweist es sich, dass nicht kontrollierend in die Testfahrten eingegriffen werden kann. Bei einer realen Testfahrt können komplexe Regelsysteme in Echtzeit untersucht werden.

Durch den Umstand, dass sich die Szenarien einer realen Testfahrt nicht einfach verändern lassen, muss bei der Gestaltung und Durchführung der Testfahrten ein hoher Ressourcen Aufwand aufgebracht werden. Es kann notwendig sein für einen bestimmten Test mehrere tausend Kilometer zu reisen. Viele der Fahrerassistenz- und Sicherheitsfunktionen greifen aktiv und präventiv in kritischen Situationen ein. Daher muss ein großer Aufwand zur Umsetzung, von für den Straßenverkehr zu gefährlichen Szenarien auf

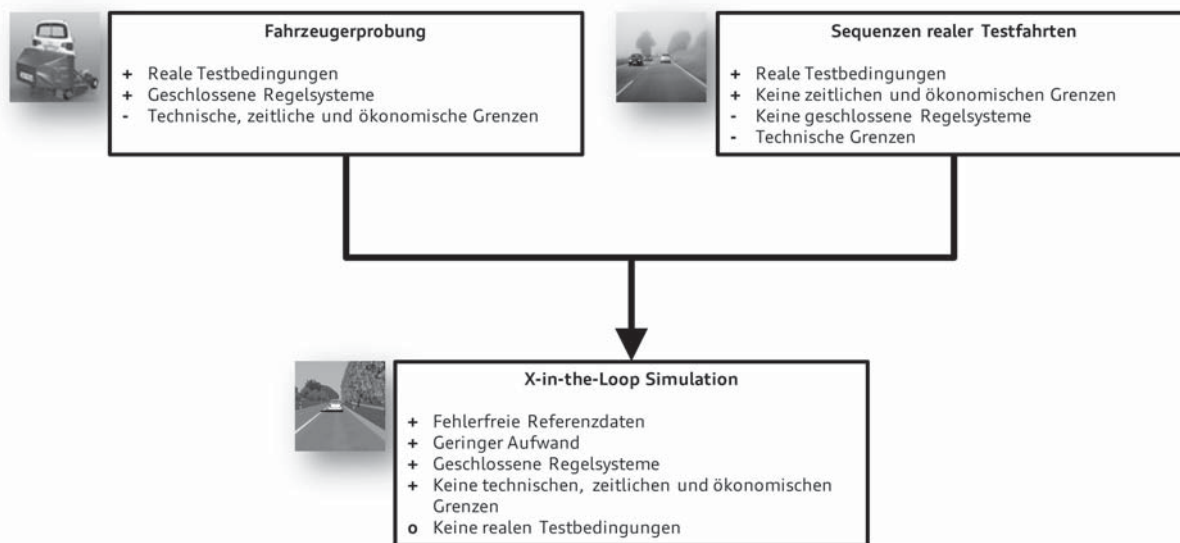


Abbildung 3.2.: Erprobung von kamerabasierten Fahrerassistenzsystemen

einem Prüfgelände aufgebracht werden. Daimler nutzt zum Beispiel für das reproduzierbare Ausführen von Szenarien von Roboter gesteuerte Fahrzeuge [46] und fahrbare Softziele. Audi hat für die Applikation der Pre-Crash Funktionen ein fahrbares Zielobjekt entwickelt [112]. Ein innovativer Ansatz zur Kombination von Realität und Simulation stellt das Vehicle-in-the-Loop Konzept von [26] dar. Dieses zeichnet sich dadurch aus, dass ein reales Fahrzeug mit einer Simulationsumgebung gekoppelt wird. Damit ist es möglich, auf einem Testgelände unter Verwendung von Augmented-Reality Techniken die Simulation und die Realität zu vereinen. Die Sensorsignale werden in der Simulation erzeugt. Ein wichtiger Aspekt bei realen Testfahrten ist es, dass entsprechend des Fahrmanövers ein gewisses Risiko entsteht, welches nicht immer vernachlässigt werden kann.

Sequenzen realer Testfahrten

Zur Integration der kamerabasierten Wahrnehmungsanteile und Funktionen werden häufig Videoaufzeichnungen von realen Testfahrten verwendet. Zur Aufzeichnung der Videos sind reale Testfahrten, mit den bereits erwähnten Vor- und Nachteilen notwendig. Die Videos werden in den Speicher der Kamera zur weiteren Verarbeitung durch Signalverarbeitung eingespeist und die Bildverarbeitungsalgorithmen werden wie bei einer realen Erprobungsfahrt ausgeführt.

Da es sich um statische Videosequenzen handelt, welche mit einer bestimmten Kamera-Hardware aufgezeichnet wurden, können diese bei einer Änderung der Kamera, Optik oder Belichtungssteuerung nicht weiterverwendet werden. Eine Konvertierung der Videos zur Verwendung mit einem neuen Kamerasystem ist nicht möglich. Außerdem ist es nicht möglich Systeme in ihrer gesamten Komplexität zu erproben welche einen regelnden oder steuernden Einfluss auf die Längs- oder Querdynamik des Fahrzeugs haben. Im Falle eines Regelungseingriffs eines Fahrzeugsystems müsste sich das Video bei einer Nachsimulation entsprechend anpassen.

Als Vorteile lassen sich reale optische Testbedingungen und die nicht vorhandenen zeitlichen und ökonomischen Einschränkungen anführen. Letzteres ergibt sich durch die Wiederverwendbarkeit der aufgezeichneten Daten und der Automatisierbarkeit der Tests.

X-in-the-Loop

Unter dem Begriff X-in-the-Loop, werden hier Software-, Model- und Hardware-in-the-Loop zusammengefasst. In Abbildung 3.3 sind verschiedene simulationsbasierte Ausprägungen zur Verifikation der Modul-, Software- und Systemspezifikationen dargestellt. Die Model-in-the-Loop Erprobung kommt vor allem für

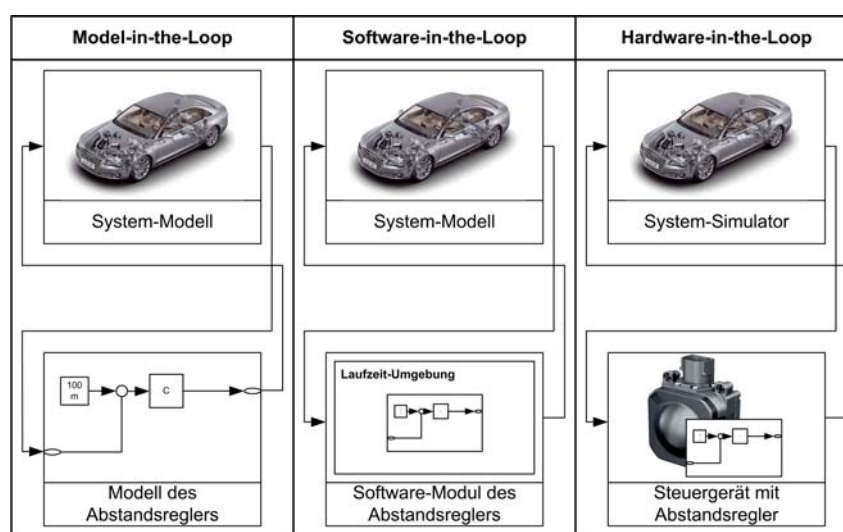


Abbildung 3.3.: Model-, Software- und Hardware-in-the-Loop

den Test von einzelnen Software-Modellen zum Einsatz, eignet sich aber auch für die Bewertung von gesamten Systemen. Dieser Ansatz wird häufig verwendet, wenn die Softwaremodule durch eine grafische Modellierung erstellt werden. Für die Untersuchung des Verhaltens wird das erstellte Software-Modell mittels eines Simulationsmodells des Rest-Systems oder Gesamtfahrzeugs evaluiert. Dafür kann Model-in-the-Loop Test in einem Echtzeit-Kontext ausgeführt werden.

Ein ähnlicher Ansatz wird bei der Verwendung von Software-in-the-Loop Techniken für die Integration von Softwaremodulen oder Softwaresystemen verfolgt. Im Vergleich zur Model-in-the-Loop Erprobung wird anstelle eines Modells der Software, die kompilierte Software erprobt. Dies kann durch Autocode-Generierung auf Basis des vorher erstellten Modells geschehen.

Zur Erprobung eines einzelnen oder eines Verbunds von Steuergeräten wird die Hardware-in-the-Loop Technologie verwendet, vgl. Abbildung 3.3. Dieser Ansatz zeichnet sich dadurch aus, dass das zu testende Steuergerät oder System mit einem Simulatorsystem verbunden wird. Der Simulator stellt dem realen Steuergerät oder Steuergeräteverbund ein in Echtzeit simuliertes Fahrzeug bereit. Als Schnittstelle zum Datenaustausch werden die im Fahrzeug vorgesehenen Steckverbindungen und Bussysteme verwendet. Das Simulatorsystem stellt alle Größen des Restsystems in Echtzeit bereit. Weiterhin wird häufig das

Verhalten des realen Fahrzeugs und aller beteiligten Aktuatoren, welche durch das reale Steuergerät angesteuert werden, simuliert. Ein wichtiger Anspruch bei der Durchführung von Hardware-in-the-Loop-Tests ist es, simulierte Daten am Beginn der funktionalen Wirkkette einzuspeisen um End-To-End Test durchführen zu können. Ein End-To-End Test prüft die gesamte Wirkkette der Funktion, von der Quelle bis zur Senke.

Die Einschränkungen der realen Fahrzeugerprobung und nicht Verwendbarkeit aufgezeichneter Testfahrten für eine Erprobung von Fahrerassistenz- und aktiven Sicherheitssystem in kritischen Fahrsituationen haben zu einem Paradigmenwechsel geführt. In den letzten Jahren wurden für die X-in-the-Loop Erprobung verschiedene Lösungen entwickelt [44] [79] [169] [111] [124]. Diese Simulationssysteme sind eine vielversprechende Alternative zu realen Testfahrten oder aufgenommenen Videosequenzen. Durch die Kombination mit einem Hardware-in-the-Loop Prüfstand, können außerdem geschlossene Regelschleifen in Echtzeit untersucht werden.

3.3. Simulationsumgebungen für Fahrerassistenz- und aktive Sicherheitssysteme

Ziel 1: Analyse und Bewertung von Simulationslösungen für die Erprobung von Fahrerassistenzsystemen.

Zur Umsetzung einer X-in-the-Loop Testumgebung für Fahrerassistenzsysteme wurden eine Vielzahl von Softwarelösungen zur Generierung synthetischer Sensorsignale und Kamerabilder entwickelt. Im Folgenden werden bekannte Simulationslösungen vorgestellt und Anwendungen beschrieben. Abschließend wird eine Bewertung der Simulationsgüte durchgeführt. Die Bewertung basiert auf bekannten wissenschaftlichen Veröffentlichungen und frei zugänglichen Informationen der Hersteller.

3.3.1. Wissenschaftliche Simulationen

Im Bereich der universitären Forschung wurden durch die folgenden Institutionen Lösungen für die Erprobung von Fahrerassistenzsystemen und insbesondere kameraunterstützte Fahrerassistenzsysteme entwickelt:

- Ohio State University
- RWTH Aachen
- Universität Kassel
- Universität Auckland

3.3.1.1. Ohio State University

Der erste bekannte Anwendung einer Fahrsimulation für die Bewertung der Robustheit eines Spurerkennungsalgorithmus fand im Jahr 2000 durch Redmill et. al. [150] statt. Zur Bewertung der Funktionen wurde der durch die Ohio State University entwickelte „Vehicle and Traffic Simulator“ (VATSIM) verwendet. Dieser wurde speziell zur Simulation von automatisierten Fahrzeugen mit verschiedenen Sensoren, Regelstrukturen und Verkehrsnetzwerken entwickelt [152] [102]. Das Simulatorsystem verfügte über eine 3D-Umfeldsimulation und Szenariogenerierung für die Erzeugung von Eingangsgrößen für bildverarbeitende Systeme. Weiterhin sind Fahrzeugdynamikmodelle mit sechs Freiheitsgraden zur Beschreibung einer kohärenten Fahrzeugbewegung hinterlegt. Der Simulator ist in der Lage aus GIS-Daten abgeleitete Beschreibungen für das Umfeld und Straßenverläufe zu importieren. Damit wurden in [151] die Anwendungsmöglichkeiten für die Parameteroptimierung und Bewertung von Sensordatenfusionsalgorithmen mit GPS-Daten untersucht.

Anhand der Ausführungen und Darstellungen von Kasnakoglu [96] kann festgestellt werden, dass die Grafik des Simulators dem damaligen Stand der Technik entsprechend war. Die Texturen der Straßenoberfläche und der Markierung waren sehr niedrig aufgelöst und wurden nicht interpoliert. Die Modellierung der Straßenoberfläche wies einen kontinuierlichen Verlauf mit einer angemessenen Tessellierung auf. Es wurden keine atmosphärischen Modelle oder Wettermodelle umgesetzt. Die Fahrzeugmodelle waren sehr einfach in der geometrischen Ausprägung. Der Lack der Fahrzeuge wurde mittels Flat-Shading [175] umgesetzt. Die Objekte wiesen keinen Schatten in der Simulation auf.

Mit Hilfe des Simulators wurde die Auswirkung von Rauschen auf die Erkennungsleistung einer Spurerkennung [150] untersucht. Zur Bewertung des Verhaltens wurden sechs verschiedene Straßenoberflächen erzeugt. Als Testszenario wurde eine vierspurige Straße, mit durchgezogenen und gestrichelten Markierungen verwendet. Der Straßenbelag wurde in den Tests durch Rauschen gestört. Der Anteil des Rauschens wurde zunehmend erhöht. Dies führte zu einer negativen Beeinflussung der Leistungsfähigkeit der Spurerkennung. Da durch das Rauschen das Signal-zu-Rausch-Verhältnis zwischen der Markierung und der Straßenoberfläche reduziert wurde.

3.3.1.2. RWTH Aachen

In [34] und [70] wird das durch die RWTH Aachen entwickelte System PELOPS für die Software- und Hardware-in-the-Loop Erprobung von Fahrerassistenzsystemen vorgestellt. Der Fokus des Simulators liegt nicht auf einer Stimulation der Bildverarbeitung. Das System beinhaltet eine Kombination auf Fahrzeugs- und Verkehrsmodellen um die Wechselwirkung von Fahrer, Fahrzeug und Verkehr zu betrachten [69]. Die Beschreibung des Straßenverlaufs erfolgt auf Basis von Krümmungsradien und Klothoiden. In der Simulation können Verkehrszeichen, Spurmarkierungen und Umweltbedingungen (Nässe, Nebel und Glätte) platziert werden. Zur Beschreibung des Verkehrsumfelds werden Szenarien mit anderen Fahrzeugen definiert.

Die grafische Simulation erfüllt grundlegende Anforderungen an den Realismus. Durch geometrische Beschreibung der Straße ist die Qualität dieser sehr gut. Die Texturen der Straße weisen eine ent-

3.3. SIMULATIONSUMGEBUNGEN FÜR FAHRERASSISTENZ- UND AKTIVE SICHERHEITSSYSTEME

sprechende Genauigkeit auf. Zur Simulation der atmosphärischen Bedingungen wird auf die in OpenGL implementierten Funktionen zurückgegriffen. Es findet keine Simulation von dynamischen Schatten statt.

3.3.1.3. Universität Kassel

Im Rahmen des DECOS-Projekts wurde durch die Universität Kassel die Simulationsumgebung PRAETORIA für den Test von Fahrerassistenzsystemen entwickelt. Mit dieser soll die Funktionsfähigkeit von sicherheitskritischen Systemen nachgewiesen werden. Das System setzt sich aus einer Fahrdynamiksimulation, einer Verkehrssimulation und Sensormodellen zur Überwachung des Umfelds zusammen [180]. Die Beschreibung des Straßenverlaufs erfolgt ähnlich zum OpenDrive-Format.

Durch Schmidt [168] wurde die eine graphische Visualisierung für den Anwender und zur Stimulation von Kamerasystemen entwickelt. Die 3D-Visualisierung verwendet qualitativ hochwertig aufgelöste Modelle für die Fahrzeuge und die Straße. In der Simulation wird eine dynamische Schattenberechnung auf Basis von Shadow-Volumes durchgeführt. Nach Tellmann et. al. [180] ist es weiterhin möglich das Verkehrsszenario, d.h. Umfeld und Objekte, mit verschiedenen Sensoren abzutasten. In der Simulation sind Modelle für einen Laserscanner, Radarsensor und Ultraschall umgesetzt.

Nach [169] wird die Simulationsumgebung für die Untersuchung eines radarbasierten Stauassistenten verwendet. Die Erprobung einer Kurvenlichts, Distanzregelung und Spurhaltesystems ist in [168] beschrieben.

3.3.1.4. Universität Auckland

Durch die Universität Auckland werden auf Basis der Raytracer POV-Ray und LuxRenderere erzeugte Videosequenzen für die Erprobung von Structure-from-Stereo Algorithmen verwendet. Die Erzeugung der Bilddaten fand nicht in Echtzeit statt. Durch das aufwändige Raytracing weisen die Videos eine hohe optische Qualität auf

Durch Vaudrey [184] wurden die Unterschiede bei der Verwendung einer synthetischen Szene im Vergleich zur einer realen Szene für Structure-from-Stereo Algorithmen untersucht. Als Erkenntnis bei der Anwendung verschiedener Algorithmen wurde festgestellt, dass Algorithmen welche für die Simulation optimiert wurden eine schlechtere Erkennungsqualität in einer realen Szene aufweisen. Das gleiche Verhalten konnte umgekehrt mit synthetischen Bilddaten verifiziert werden, wenn ein für reale Szenen optimierter Algorithmus angewendet wurde. Vaudrey gibt als Erklärung für das Verhalten an, dass die Kanten von Objekten in synthetischen Szenen sehr genau sind. In einer realen Szenen sind die Kanten verschmiert. Weiterhin wird angemerkt, dass es in der Simulation keine Schwankungen in der Belichtung einer Bildfolge gibt.

In [115] wird von Morales die Simulation für den Vergleich von verschiedenen Structure-from-Stereo Algorithmen verwendet. Es werden Auswirkungen auf die Leistungsfähigkeit der Algorithmen bei Störungen untersucht. Die Störungen sind Helligkeitsschwankungen, Weichzeichnung und Rauschen. Als vorteilhaft erweist sich die Möglichkeit bei der Auswertung den ermittelten optischen Fluss und die

3.3. SIMULATIONSUMGEBUNGEN FÜR FAHRERASSISTENZ- UND AKTIVE SICHERHEITSSYSTEME

geschätzten Distanzwerte gegen ideale Referenzdaten aus der Simulation zu vergleichen. Die Leistungsfähigkeit der Algorithmen wurde mittels einer Rangfolge festgehalten. Es stellte sich heraus, dass diese bei einer Bewertung der Algorithmen in der Simulation eine andere ist als bei Verwendung von realen Bilddaten.

Ein zusammenfassender Überblick über die Untersuchungen ist in [100] gegeben. In dieser Veröffentlichung werden verbesserte Szenen einer Kreuzung präsentiert. Zur Erstellung wird ein realistisches globales Beleuchtungsmodell verwendet, insbesondere Schatten werden genau simuliert. Nach [100] ist die Simulation von Glanzlichtern wichtig, da diese häufig zu Schwierigkeiten bei der maschinellen Wahrnehmung führen.

3.3.2. Kommerzielle Simulationen

Die folgenden kommerziellen Simulationsumgebungen werden zur Zeit für die Entwicklung und Erprobung von Fahrerassistenzsystemen verwendet:

- IPG Automotive GmbH: CarMaker
- dSpace GmbH: Automotive Simulation Models-Toolkette
- Tesis GmbH: Dyna4
- LIVIC: SiVIC
- MB-Tech: Provetech:VL
- TASS: PreScan
- Vires GmbH: Virtual Test Drive

3.3.2.1. IPG Automotive GmbH: CarMaker

Die CarMaker-Toolkette von IPG Automotive GmbH besteht aus Komponenten für eine Fahrdynamik-, Verkehrs-, Sensor- und Kamerasimulation. Nach Angaben des Herstellers können mit der Software Tests von aktuellen Fahrerassistenzsystemen durchgeführt werden. In der Software kann eine beliebige Anzahl von Verkehrsobjekten und statischen Hindernissen abgebildet werden. Die Beschreibung des Straßenverlaufs und der Straßenmarkierungen kann auf Basis von geometrischen Elementen erfolgen.

Die grafische Visualisierung des IPG-Movie ist für den Anwender zur Begutachtung der Testfälle vorgesehen. Inzwischen wird dieses Programm auch für die Stimulation von Kamerasystemen verwendet. Die Qualität der Oberflächeneigenschaften der Fahrzeuge und Objekte ist mit einem Blinn-Phong-Shading Ansatz einfach gehalten. Die Simulation verfügt über eine lokale Beleuchtungsberechnung, d.h. es werden keine Schatten, berechnet.

In [44] ist die Verwendung des IPG-Movie zur Erprobung einer Spurerkennung in der Software- und Hardware-in-the-Loop Simulation beschrieben. Die Leistungsfähigkeit der Funktion wurde in Szenarien

3.3. SIMULATIONSUMGEBUNGEN FÜR FAHRERASSISTENZ- UND AKTIVE SICHERHEITSSYSTEME

mit Geraden und scharfen Kurven, Verdeckungen durch Fahrzeuge und verschiedenen Spurmarkierungen untersucht. Die ermittelte Leistungsfähigkeit lag bei 98.7%. Durch [114] wird die Verwendung des CarMaker für die Hardware-in-the-Loop (HIL) Erprobung eines ACC, eines Parksystems und einer kamerabasierten Spurhalteunterstützung beschrieben.

3.3.2.2. dSpace: Automotive Simulation Models-Toolkette

Die durch dSpace entwickelte Automotive Simulation Models-Toolkette bietet mit den Werkzeugen MotionDesk, ModelDesk und ASM-Traffic die Möglichkeit ADAS in einem simulierten Umfeld abzusichern [55]. Das System wird im Hardware-in-the-Loop Betrieb verwendet. Die Erstellung der Straße und Definition der Fahrhinweise findet mit ModelDesk statt. Die Definition der Verkehrssituation mit ASM-Traffic. Eine grafische Visualisierung für ein Kamerasystem kann mit MotionDesk durchgeführt werden.

Die Qualität der grafischen Aussage erfüllt grundlegende Ansprüche zur Durchführung von funktionalen Tests. Es werden Straßenoberflächen und Spurmarkierungen simuliert. Weiterhin haben Fahrzeuge eine hinreichend genaue geometrische Qualität. Eine globale Beleuchtungssimulation findet nicht statt. Die Simulation von Sensoren erfolgt auf Objektebene durch die Angabe eines Öffnungswinkels für die Sensormodelle. Durch die Simulation können maximal 15 Fahrzeuge berechnet werden.

Nach [55] wird das ASM-Traffic bei Denso für die Absicherung von ACC-Systemen eingesetzt. Ford verwendet die Toolkette für die Absicherung der kamerabasierten Spur- und Verkehrszeichenerkennung [191].

3.3.2.3. Tesis GmbH: Dyna4

Eine weitere echtzeitfähige Plattform für die Erprobung von Fahrerassistenzsystemen stellt die Dyna4-Toolumgebung der Tesis GmbH dar. Das System kann für Software- und Hardware-in-the-Loop Tests verwendet werden.

Die Verkehrssimulation kann maximal 40 Verkehrsteilnehmer simulieren. Die Straßenverläufe werden durch eine Aneinanderreihung von Straßensegmenten oder Navigationsdaten erzeugt. Die Eigenschaften der Fahrspurmarkierungen können frei definiert werden. Es ist möglich Verkehrsschilder am Rand der Straße zu positionieren.

Zur Erprobung von kamerabasierten Systemen wird DYNAanimation verwendet. Die Straßenoberfläche ist mittels Texturen dargestellt und wird mittels eines Phong-Shadings beschrieben. Die Verkehrszeichen sind als Billboard-Texturen ohne Oberflächeneigenschaften realisiert. Die dargestellten Fahrzeuge weisen eine hohe geometrische Genauigkeit auf. Der Lack der Fahrzeuge ist sehr einfach gehalten. Atmosphärische Eigenschaften und Wettereinflüsse werden nicht simuliert. Weiterhin werden keine statischen oder dynamischen Schatten erzeugt.

3.3. SIMULATIONSUMGEBUNGEN FÜR FAHRERASSISTENZ- UND AKTIVE SICHERHEITSSYSTEME

Die Verwendung der Simulation zur Erprobung von kamerabasierten Assistenzsystemen ist in [21] beschrieben. Mit Hilfe des Systems sollen eine Fahrspurerkennung, Fahrspurverlassenswarnung, Fahrzeugerkennung, Fernlichtassistent, Kollisionswarnung und Verkehrszeichenerkennung erprobt werden. In der Veröffentlichung werden keine Ergebnisse präsentiert.

Durch Gunia et. al. [80] [81] wird die Verwendung der Simulation für die Erprobung eines Spurhalteassistenten vorgestellt. Im Rahmen der Anwendung wurden der Fahrbahntyp und -verlauf und die Fahrbahnmarkierungen verändert. Damit konnten schwierige Situationen in der Simulation werden. Als nachteilig hat sich erwiesen, dass nicht alle Umgebungsbedingungen durch die synthetischen Daten abgebildet wurden. Daher wurde die Absicherung in der Simulation als Ergänzung betrachtet. In den Testaufbauten von [21] und [80] wird das Bild von einem Projektionsmedium durch die Fahrzeugkamera abgefilmt.

3.3.2.4. LIVIC: SiVIC

Die von LIVIC (Laboratory on the Interactions between Vehicle-Infrastructure-Driver) des INRETS entwickelte SiVIC-Simulationsumgebung wurde speziell für die Stimulation der Bildverarbeitung von Fahrerassistenzsystemen entwickelt [79] [183]. Die Zielsetzung ist Algorithmen mit genauen Referenzdaten zu testen und zu erproben. Die Simulationsumgebung beinhaltet das Straßenumfeld, Fahrzeuge, Fußgänger und Infrastruktur.

In der von Gruyer et. al. [78] dargestellten Ausprägung weist die Kamerasimulation eine sehr hohe Güte auf. Die Simulation setzt sich aus einem physikalischen Renderer und einem Kameramodell zusammen. Die Simulation ist modular aufgebaut, so dass Module für die Simulation von Regen und Nebel eingebunden werden können. Zur Simulation der Kameraeigenschaften können Module für Tiefpass-Filter, Rauschen, Schärfentiefe, Farbfilter integriert werden. Weiterhin ist eine Beleuchtungssteuerung der Kamera enthalten.

Die SiVIC-Simulationsumgebung wurde in einer Reihe von Projekten verwendet. Durch Glaser et. al. [76] wird die Simulationsumgebung zur Untersuchung der Trajektorienplanung auf Autobahnen in Abhängigkeit der Verkehrssituation verwendet. Durch Bossu et. al. [29] wird die Simulation zur Validierung und Bewertung einer Fußgängererkennung verwendet. In [31] wird die Simulation zur Entwicklung eines Algorithmus zur Sichtverbesserung bei Nebel und in [78] bei Projekten zur Spur- und Objekterkennung eingesetzt.

3.3.2.5. MB-Tech: Provetechn:VL

Für den Test von aktiven kamerabasierten Systemen wurde durch MB-Tech das Werkzeug Provetechn:VL entwickelt. In der 2012 veröffentlichten Dissertation von Schmidt [170] werden Informationen und eine wissenschaftliche Motivation zur Entwicklung des Tools gegeben. Das Tool basiert auf der Unity Grafik-Engine. Bei der Entwicklung wurde von der Annahme ausgegangen, dass das zu testende Bildverarbeitungssystem eine Black-Box ist, d.h. keine Informationen über die notwendige Simulationsqualität und die Funktionalität zur Verfügung stehen. Die Simulation soll für alle bildverarbeitenden Systeme

3.3. SIMULATIONSUMGEBUNGEN FÜR FAHRERASSISTENZ- UND AKTIVE SICHERHEITSSYSTEME

funktionieren. Aufgrund dieser Annahmen ist eine Validierung Simulation gegen die Anforderungen des Systems nicht möglich. Die Qualität der Simulation wurde nach Aussage von Schmidt in einer von ihm betreuten Diplomarbeit untersucht. Die Qualität einer betrachteten Referenzszene betrug demnach 95% zu einem realen Vorbild.

Nach Aussage von [170] ist es möglich Streckenverläufe auf Basis von OpenStreetMap-Daten zu erzeugen. Als vorteilhaft erweist sich die Größe der vorhandenen Objekt-Datenbank. Weiterhin sind für Oberflächen verschiedene Alterungsstufen von Texturen vorhanden. Deformationen werden nicht betrachtet.

In der Arbeit von Schmidt [170] werden drei Anwendungsbeispiele für das Simulationssystem genannt: ein Aftersale Spurerkennungssystem, eine Aftersale Verkehrszeichenerkennung und eine Structure-from-Stereo Rekonstruktion [197]. In jedem Testaufbau filmt die Kamera einen Monitor ab. Als Ergebnis für die Spurerkennung und Verkehrszeichenerkennung wird angemerkt, dass diese Systeme eine schlechtere Leistungsfähigkeit bei einer Verschlechterung der Wetterbedingungen in der Simulation aufwiesen. Bei der Auswertung wurden keine Bilddaten der Kamera betrachtet. Es wird festgestellt, dass die Spurerkennung robuster als die Verkehrszeichenerkennung gegenüber Störungen ist. Bei der Erprobung der Structure-from-Stereo Technik, verweist der Autor auf Probleme mit periodischen oder „gekachelten“ Texturen. Eine Structure-from-Stereo Rekonstruktion führt eine Suche nach Korrespondenzen in benachbarten Bildpaaren durch [171], daher ist es nicht verwunderlich, dass es bei periodischen Mustern zu Schwierigkeiten kommen kann.

3.3.2.6. TASS: PreScan

Durch TNO wurde das Tool PreScan entwickelt. Die Simulation beinhaltet das Verkehrsumfeld und die Infrastruktur. Durch die Simulation können verschiedene Umweltbedingungen abgebildet werden: Tag, Nacht, Regen, Schnee und Nebel. Hinsichtlich der Qualität der Abbildung werden keine Aussagen getroffen. Zur Simulation einer globalen Beleuchtung werden Schatten in Echtzeit simuliert. Zur Generierung von hochqualitativen Videos steht eine Anbindung an POV-Ray zur Verfügung [181]. Laut Produktdatenblatt [182] stehen Sensormodelle für Ultraschall, Radar, Lidar, PMD und Kamera zur Verfügung. Zur Qualität der Modelle wird keine Aussage getroffen. Die Geometrie des Straßenmodells weist keine hohe Qualität auf, es werden keine Klothoiden verwendet. Ein Import von Straßennetzen aus GPS-Daten ist möglich.

Nach Aussage von TNO verwendet Denso PreScan zur Absicherung des Lane Departure Warnings und der Lane Keeping Assistance [30]. SIAC Motor verwendet PreScan zur Konzeptprüfung von neuen aktiven Sicherheitssystemen [110].

3.3.2.7. Vires: Virtual Test Drive

In Zusammenarbeit mit dem Volkswagen Konzern wurde durch Vires die Virtual Test Drive-Toolkette entwickelt. Diese wird für die Entwicklung und Erprobung von Fahrerassistenzsystemen und aktiven Sicherheitsfunktionen im Software- und Hardware-in-the-Loop Umfeld verwendet [57] [53]. In der Simulationsumgebung können verschiedene Sensoren, auf Objekt und Rohdatenebene, realisiert werden. Zur

3.3. SIMULATIONSUMGEBUNGEN FÜR FAHERASSISTENZ- UND AKTIVE SICHERHEITSSYSTEME

Generierung eines Kamerabilds steht eine entsprechende Grafik-Engine zur Verfügung. Weiterhin ist eine Integration von auf nVidia-Optix basierenden Ray-Tracing-Modulen möglich.

Das Straßenmodell wird auf Basis des OpenDRIVE-Standards [136] beschrieben und beinhaltet Geraden, Kreisbögen und Klothoiden. Durch diese geometrischen Funktionen können realitätsnahe Straßenverläufe definiert werden. Die Texturen der Straße können ebenfalls frei festgelegt werden. Zur Abbildung von Umwelteigenschaften kann die Uhrzeit variiert werden. Weiterhin können verschiedene Wetterbedingungen wie Nebel und Regen dargestellt werden. Bei Regen werden die Siegelungen von Objekten auf der Straße simuliert. Eine globale Beleuchtung wird durch dynamische Schatten approximiert. Die Fahrzeugmodelle sind hoch aufgelöst und besitzen eine photorealistische Erscheinung. Zur Erprobung einer Verkehrszeichenerkennung können Schilder positioniert werden. Weiterhin ist eine Integration von Fußgängern möglich.

Im Bereich der Anwendung der Toolkette sind eine Reihe von Publikationen entstanden. In [127] wird die Verwendung von Virtual Test Drive für die Optimierung von Regelparametern eines Spurmodells beschrieben. Die Parameter werden auf Basis einer Kamera geschätzt. Durch Roth et. al. [159] wird ein Prozess zur Verifikation von Sensormodellen dargestellt. Die Simulation eines PMD-Sensors ist in [158] beschrieben.

3.3.3. Vergleich der Simulationsumgebungen und Zusammenfassung

Im diesem Abschnitt wurden verschiedene Simulationsumgebungen zur Simulation des Fahrzeugumfelds, von Sensordaten und zur Generierung von synthetischen Kamerabildern vorgestellt. Im Folgenden sollen die Simulationsumgebungen verglichen und bewertet werden. Eine Gegenüberstellung, in Form einer Tabelle, kann im Anhang A.1 vorgefunden werden.

3.3.3.1. Vergleichs des Realismusgrads

Als erstes soll der Grad des Realismus der verschiedenen Simulationsumgebungen verglichen werden. In der Literatur werden drei Varianten des Realismus in der Computergrafik definiert [68]:

- physikalischer Realismus: Das Bild beinhaltet die gleiche visuelle Stimulation wie die reale Szene.
- Photorealismus: Das Bild ruft beim Betrachter die gleiche visuelle Erscheinung wie die reale Szene hervor.
- funktionaler Realismus: Das synthetisierte Bild bietet für den Betrachter die gleiche Information wie die reale Szene.

Jede dieser Varianten stellt unterschiedliche Anforderungen an den Bildgenerierungsprozess.

Physikalischer Realismus

Im Falle des physikalischen Realismus muss das Bild eine genaue 1:1-Abbildung der spektralen Lichtstärke der realen Referenz an jedem Punkt der Szene aufweisen. Daraus folgt, dass die Szene genaue Beschreibungen der Form und Materialien von Objekten und Beleuchtung einer realen Szene beinhalten muss. Eine detaillierte physikalische Bildsynthese ist extrem aufwendig und nicht in interaktiven Anwendungen verwendbar. Bei der Bilderzeugung werden keine Einschränkungen durch die Wahrnehmung betrachtet.

Photorealismus

Das Ziel einer photorealistischen Simulation ist es, für den menschlichen Betrachter ein Bild zu erzeugen, welches nicht von einem Foto unterscheidbar ist. Das Bild muss die gleiche visuelle Erscheinung wie die reale Szene beim Betrachter hervorrufen. Daher wird zur Optimierung der Bildgenerierung der menschliche Betrachter miteinbezogen.

Funktionaler Realismus

In Echtzeitanwendungen der Computergrafik wird häufig der funktionale Realismus verwendet. Die Zielsetzung ist es für den Betrachter der Simulation die gleichen visuellen Informationen bereitzustellen, wie bei Betrachtung der realen Szene. Es soll eine verlässliche Ausführung von Aufgaben und Bewertungen mit der Simulation sein. Als Beispiel dafür nennt Ferwerda [68] Simulator-Systeme.

3.3.3.2. Realismusgrad der Simulationsumgebungen

Alle echtzeitfähigen Simulationsumgebungen weisen mindestens einen funktionalen Realismus auf. Hinsichtlich der Anwendung wird eine Einteilung in Basis- und Robustheitsuntersuchungen vorgenommen. Bei der ersten Ausprägung sollen die Wahrnehmungsfunktionen, vgl. Abschnitt 2.3, korrekt stimuliert werden um die eigentliche Assistenzfunktion zu erproben. Bei einer Robustheitsuntersuchung soll die Güte der Wahrnehmungsfunktionen unter dem Einfluss von Störgrößen, z.B. Schlagschatten, betrachtet werden. Dazu wird ein fast photorealistischer Realismusgrad in den Simulationslösungen erreicht. Dieser hat vor allem Auswirkungen auf die Detaillierung der Geometrie und Oberflächeneigenschaften.

3.3.3.3. Anwendungsfelder

Der umgesetzte Realismusgrad hat direkte Auswirkungen auf die Anwendungsfelder. Software-Lösungen für die Durchführung von Basisuntersuchungen werden ausschließlich in der Hardware-in-the-Loop Integration des Gesamtsystems oder des Steuergerätes eingesetzt. Diese Vorgehensweise spiegelt auch das V-Modell wieder, in welchem vorausgesetzt wird, dass die Wahrnehmungsmodule bereits im Modul-Test abgesichert wurden. Im Gegensatz dazu werden Software-Lösungen mit Fähigkeiten zur Durchführung

3.3. SIMULATIONSUMGEBUNGEN FÜR FAHERASSISTENZ- UND AKTIVE SICHERHEITSSYSTEME

von Robustheitsuntersuchungen im gesamten Entwicklungsprozess beginnend, ab dem Model-in-the-Loop auf Modul-Test-Ebene eingesetzt. Durch die Schaffung einer durchgängigen Anwendbarkeit der Simulationsumgebungen profitieren die folgenden Verifikationsstufen vom bereits investierten Aufwand zur Validierung der Simulation, Verifikation der Modelle und Definition von Testszenarien.

3.3.3.4. Validierung der Simulationsumgebungen

In der vorliegenden Literatur wurde keine der Simulationslösungen umfassend validiert. Es wurden keine Grenzen oder Gültigkeitsbereiche bzgl. der Anwendbarkeit der Simulationslösungen definiert bzw. kommuniziert.

Per Definition ist ein Modell immer nur eine beschränkte Abbildung der Realität. Daher ist die Validierung der Anwendbarkeit von höchster Wichtigkeit. Besonders bei Simulationslösungen mit der Zielsetzung von Robustheitsuntersuchungen ist dies eine notwendige Voraussetzung, um die Zweckmäßigkeit und Anwendbarkeit einzelner Teilmodelle und des Gesamtmodells zu rechtfertigen. Damit gilt es auch die Wirtschaftlichkeit der Anwendung einer Simulation zu belegen. Ein Beispiel für dieses Dilemma sind die vorhandenen Implementierungen oder Parametrierungen verschiedener Tageszeiten und Wetterbedingungen. Diese führen zu einer Veränderung der Bildeigenschaften. Eine Übertragbarkeit auf eine Situation in der Realität ist jedoch nicht möglich. Ebenso verhält es sich mit Alterungserscheinungen. In diesem Zusammenhang spielt die Modellierung der Kamera eine übergeordnete Rolle, da die Abbildung einer Szene, maßgeblich durch die verwendete Integrationszeit (Belichtungszeit), das Antwortverhalten des CMOS und auftretende Aberrationen durch das optische System beeinflusst werden, vgl. Abschnitt 2.2.5.

3.3.3.5. Anwendungsfälle

Die Anwendbarkeit der einzelnen Simulationslösungen wurde in den entsprechenden Publikationen durch Anwendungsfälle dargestellt. Die publizierten Anwendungen im Bereich der Basisuntersuchungen betreffen vorwiegend die Erprobung von Spurerkennungssystemen [150] [44] [114] [81] oder einer Verkehrszeichenerkennung [191]. Als Begründung für keine bekannten Anwendungsbeispiele im Bereich der kamerabasierten Objekterkennung z.B. von Fahrzeugen kann angeführt werden, dass die Simulationslösungen über keine Approximation einer globalen Beleuchtung verfügen und damit das maßgebliche Merkmal eines Schattens nicht vorhanden ist. Im Vergleich dazu umfassten die Anwendungsfälle der verbesserten Simulationsumgebungen ein breiteres Spektrum. Die Anwendungsbereiche bezogen sich auf die Erprobung von Objekterkennungsverfahren [127] [76] [29] [78], Structure-from-Stereo [197] [100] oder Sichtverbesserung [31]. In keiner der Anwendungen ist eine Verwendung der Nacht- oder Wettersimulation, mit Ausnahme von Nebel, beschrieben.

3.3.3.6. Schlussfolgerung

Die Qualität der Simulationsumgebungen umfasst eine große Bandbreite, welche maßgeblich durch die Anwendungsfelder bestimmt wird.

3.3. SIMULATIONSUMGEBUNGEN FÜR FAHERASSISTENZ- UND AKTIVE SICHERHEITSSYSTEME

Zur durchgängigen Anwendung einer Simulation ist ein großer Aufwand hinsichtlich einer Validierung der Simulationsumgebungen für die definierten Anwendungsfälle notwendig. Eine Betrachtung des Assistenzsystems als Black-Box ist daher noch nicht zulässig. Die durch den Testaufbau oder die Simulation verursachten Wechselwirkungen mit dem zu testenden System werden damit vernachlässigt und die durchgeführten Tests verlieren an Gültigkeit.

Durch die Vielzahl der dokumentierten Anwendungen kann die jedoch die Sinnhaftigkeit einer Simulationsumgebung für die Erprobung von kameraunterstützten Assistenzsystemen nicht mehr bestritten werden. In diesem Zusammenhang gilt es vor allem sinnvolle Anwendungsbereiche und Gültigkeitsgrenzen der Simulation zu definieren.

4. Evaluierung von Konzepten der Computergrafik

Ziel 2: Definition von Anforderungen an eine Simulationsumgebung zur Stimulation von kameraunterstützten Fahrerassistenzsystemen.

Ziel 3: Experimentelle Evaluierung der Komplexität der Simulation hinsichtlich:

- *Ziel 3.1: Simulation der Beleuchtung*
- *Ziel 3.2: Geometrie und Oberflächeneigenschaften von Objekten*
- *Ziel 3.3: Synthese von Kamerabildern*
- *Ziel 3.4: Aufbau von Szenen und Szenarien*

Im Rahmen dieses Kapitels werden verschiedene Konzepte der Computergrafik für die Realisierung einer Simulationsumgebung zur Erprobung von kameraunterstützten Fahrerassistenzsystemen betrachtet. Als erstes werden Anforderungen an die Simulationsumgebung definiert und vier Schwerpunkte für die Untersuchung von Konzepten herausgearbeitet. Einleitend wird die zur Durchführung der Studien verwendete Simulationsumgebung beschrieben. Anschließend wird eine experimentelle Bewertung von Konzepten zur Simulation der Beleuchtung, der Geometrie und Oberflächeneigenschaften von Objekten, Synthese von Kamerabildern und der Gestaltung von Szenen und Szenarien durchgeführt.

4.1. Anforderungsanalyse

Anhand der Definition von ADAS [8] können die Anforderungen für ein System zur Erprobung der Systeme im Hardware-in-the-Loop Kontext getroffen werden, vgl. Kapitel 2. Das Testsystem muss in der Lage sein:

- Die Fahrzeugumgebung in einer ausreichend genauen Qualität zu simulieren.
- Die komplexe Signalverarbeitung plausibel und kausal zu stimulieren.
- Den Einfluss der aktiver Unterstützung des Fahrers, durch die Fahrerassistenzsysteme auf die Fahrdynamik abzubilden.

Bei Betrachtung dieser Anforderungen kann festgestellt werden, dass es durch den Hardware-in-the-Loop Ansatz möglich ist den Einfluss der aktiven Unterstützung der Fahrerassistenzsysteme in der Simulation umzusetzen. Die Herausforderung ist die Simulation der Fahrzeugumgebung in einer hinreichenden Genauigkeit und die kausale Stimulation der komplexen Signalverarbeitung. Die Simulation der Fahrzeugumgebung kann durch die Anwendung einer Umfeld- und Verkehrsimulation wie Virtual Test Drive

realisiert werden. Die Umsetzung dieser Anforderungen in einem Testsystem für die Hardware-in-the-Loop Erprobung ist in Kapitel 7 beschrieben.

4.1.1. Stimulation der Signalverarbeitung

Bei der Betrachtung der Radar- und Ultraschallsensorik konnte festgestellt werden, dass die Messgrößen durch vektorielle oder skalare Größen beschrieben werden können, vgl. Kapitel 2. Die Sensoren liefern als Messergebnis den Abstand, die Geschwindigkeit und den Winkel zurück. Eine Simulation von reflektiertem Schall oder elektromagnetischen Wellen ist für den gesetzten Anwendungsfall nicht zweckmäßig. Daher bietet es sich an simulierte Daten in die Signalverarbeitung, nach dem Empfang durch den Sensor, einzuspeisen.

Im Vergleich dazu sind die Messgrößen einer Kamera deutlich komplexer, jedes Pixel beschreibt die empfangene Bestrahlungsstärke. Ein einzelner Pixel lässt im Vergleich zu den Ausgangsdaten des Ultraschalls oder des Radars keine Rückschlüsse auf den Abstand oder Geschwindigkeit zu. Diese Informationen werden erst in einem nachgelagerten Schritt mittels der Bildverarbeitung gewonnen. Daher müssen zur Stimulation der Wirkkette, für einen End-to-End Test, die Bilddaten in den Speicher des Bildverarbeitungssystems übertragen werden. Die Einspeisung kann über eine optische oder elektrische Verbindung stattfinden. Da auf diesen Daten eine Mustererkennung ausgeführt wird, müssen die simulierten Bilder eine hinreichend genaue Qualität aufweisen und alle wichtigen Merkmale für die Bildverarbeitung beinhalten. Daher ist es notwendig die von der Bildverarbeitung ausgewerteten Merkmale zu analysieren und in die Simulation zu integrieren.

4.1.2. Stimulation der Bildverarbeitung

Zur erfolgreichen Stimulation der Bildverarbeitung muss analysiert werden welche Merkmale ein simuliertes Bild beinhalten muss. Bei Betrachtung der in Abschnitt 2.3 beschriebenen Algorithmen wurde zwischen Merkmalen für die Erkennung und Störgrößen unterschieden. Als Merkmale für die Spurerkennung sind beispielsweise besonders die Geometrie und die Oberfläche der Straße, sowie die Straßenmarkierungen von Relevanz. Im Vergleich dazu müssen Objekte eine möglichst realitätsnahe Kontur, Symmetrie und Textur aufweisen. Weiterhin muss eine der Realität entsprechende Beleuchtung des Objekts sichergestellt sein.

Während die Merkmale für eine erfolgreiche Detektion überschaubar sind, sind die Störgrößen ungleich komplexer. Eine große Herausforderung an die Segmentierung und Klassifikation stellen strukturierte Hintergründe mit einer ähnlichen Intensität und Textur wie des zu erkennenden Objekts dar. Durch diese können fälschlicherweise Kanten und Ecken erzeugt werden.

Ein weiterer nicht vernachlässigbarer Aspekt sind die Lichtbedingungen, welche sich durch die Tageszeit und die Wetterbedingungen verändern. Durch Auftreten von starken Schatten kann sich die Erkennungsleistung verschlechtern. Die sich ändernden Anteile der spektralen Strahlung des Sonnenlichts, durch die

Dämpfung und Streuung der Atmosphäre, können zu einer anderen Farbwahrnehmung führen, als ursprünglich erwartet.

Weiterhin können Verdeckungen durch andere Objekte, Verkehrsteilnehmer oder Wetterbedingungen auftreten.

Die Erscheinung eines bekannten Objekts kann sich weiterhin durch eine Deformation oder Alterung verändern. Im Falle von Verkehrsschildern können länderspezifische Abweichungen die Erscheinung verändern.

Durch den Umstand, dass die Kamera aus einem sich bewegenden Fahrzeug Bilder aufzeichnet, kann bei langen Belichtungszeiten Motion-Blur entstehen. Weiterhin können Vibrationen oder Wank- und Nickbewegungen einen negativen Einfluss auf die Qualität des Bilds haben.

4.1.3. Anforderungen an die Qualität der Simulation

Auf Basis der Ergebnisse der funktionalen Analysen von Algorithmen zur Spur-, Fahrzeug- und Verkehrszeichendetektion werden Anforderungen für die Darstellung eines funktionalen Tests und erweiterte Anforderungen für die Simulation von Störgrößen hergeleitet. Die einzelnen Anforderungen können im Anhang A.4 vorgefunden werden. Im weiteren Verlauf dieses Kapitels werden die folgenden Schwerpunkte untersucht.

Beleuchtungssimulation

Eine Kamera ist ein passiver Sensor, welcher reflektierte oder emittierte Strahlung misst. Daher ist die Untersuchung von Beleuchtungsmethoden von hoher Wichtigkeit. In den Anforderungen wurde herausgearbeitet, dass der Schatten eines Objekts ein wichtiges Merkmal oder eine Störgröße sein kann. Daher werden verschiedene Verfahren zur Simulation eines realitätsnahen Schattens betrachtet. Zur Simulation der spektralen Zusammensetzung des Sonnenlichts wird ein physikalisch basiertes Modell untersucht. Ein wichtiger Aspekt ist die Möglichkeit zur Simulation von verschiedenen atmosphärischen Bedingungen. Daher werden die Möglichkeiten zur Realisierung eines effizienten Modells für die Fahrsimulation untersucht.

Geometrie und Oberflächeneigenschaften

In der Computergrafik werden die Konturen von Objekten mit einer begrenzten Anzahl von Dreiecken und die Materialien dieser mit Modellen beschrieben. Die Kontur eines realen Objekts ist nicht kontinuierlich, selbst glatte Oberflächen bestehen aus vielen kleinen Unebenheiten. Durch diese Abweichungen kann die Erscheinung eines Objekts in der Simulation deutlich von der Realität abweichen. Daher gilt es nachzuweisen welche Oberflächenmodelle für Fahrzeuge, Straßen und Verkehrsschilder am besten geeignet sind. Weiterhin muss der Aufwand für die Erstellung eines 3D-Modells betrachtet werden. Als weiterer Schwerpunkt wird die Simulation einer nassen Straßenoberfläche mit Interaktion zur Umwelt untersucht.

Bildsynthese

Zur Erzeugung eines Bilds in der Computergrafik wird ein einfaches Lochkamera-Modell verwendet. Daher gilt es ein möglichst realitätsnahes Kamerabild zu erzeugen. Dies ist vor allem dann wichtig, wenn das Kamerabild direkt in den Speicherbereich der Kamera transferiert werden soll. Daher werden in diesem Abschnitt Techniken zur Simulation eines möglichst realitätsnahen Kamerabilds untersucht.

Szenen- und Szenario-Design

Als Ergebnis der durchgeführten Analyse von Algorithmen zeigte sich, dass der Aufbau der Szene und des Testszenarios einen großen Einfluss auf die Qualität des Tests haben kann. Durch einen irregulären Hintergrund in der Simulation kann das Ergebnis des Tests von der Realität abweichen. Weiterhin kann durch die Anordnung von Objekten in der Szene das eigentliche Objekt verdeckt sein. Daher gilt es die Auswirkungen des Szenen-Designs auf die Leistungsfähigkeit der Algorithmen zu untersuchen.

4.2. Beschreibung der Simulationsumgebung

In diesem Abschnitt wird die für die Durchführung der simulationsbasierten Studien verwendete Simulationsumgebung vorgestellt. Der Hauptgrund für die Entwicklung einer eigenen Visualisierung war die Möglichkeit ohne Restriktionen Konzepte zu evaluieren und einzubringen. Die Visualisierung wurde basierend auf dem OpenGL [137] entwickelt und ist durchgängig zum Virtual Test Drive-Framework verwendbar. In Abbildung 4.1 ist die Simulationsumgebung dargestellt.

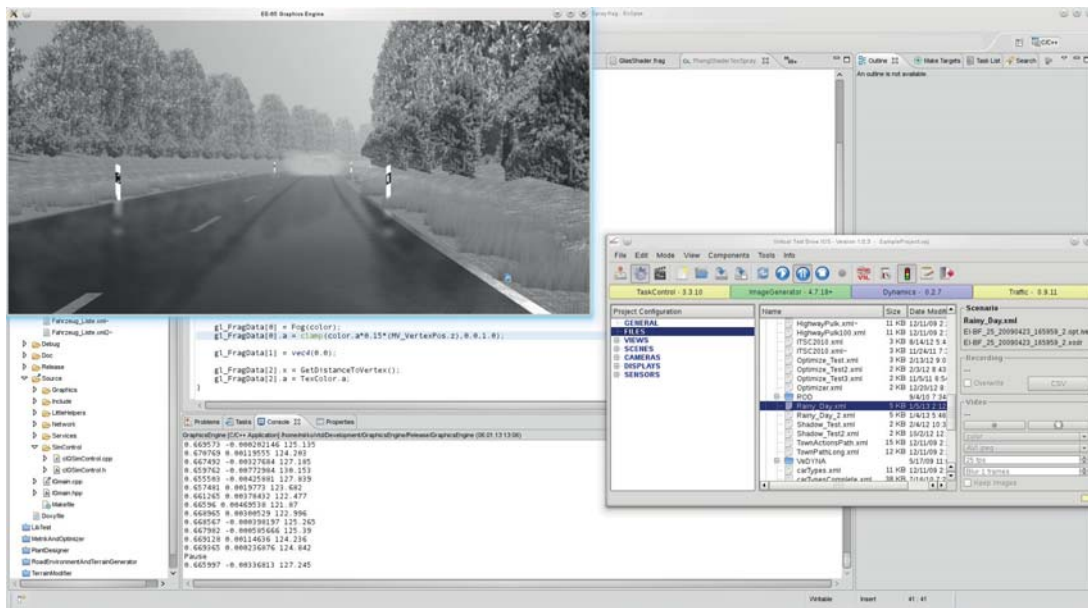


Abbildung 4.1.: Screenshot der Simulationsumgebung

4.2.1. Virtual Test Drive-Framework

Als Grundlage der Simulationsumgebung wird das Virtual Test Drive-Framework in der Version 1.0 verwendet, siehe Abbildung 4.2. Das Framework besteht aus fünf Komponenten. Die Hauptkomponente für die Ablaufsteuerung der Simulation und Kommunikation mit dem Umfeld ist die TaskControl. Im Dynamics-Manager wird die Fahrdynamik für alle Fahrzeuge berechnet. Zur Steuerung der anderen Verkehrsteilnehmer und des eigenen Fahrzeugs wird das Traffic-Modul verwendet. Die Simulation von idealen Sensoren, in der Form eines Öffnungskegels wird mittels des Sensormanagers durchgeführt. Zur Erzeugung eines Bilds für die Simulation von Bildverarbeitungssystemen ist ein Bildgenerator (Image Generator) vorhanden. Eine ausführliche Beschreibung kann in [57] vorgefunden werden. Durch offene Netzwerkschnittstellen kann eine Kommunikation mit externen Applikationen hergestellt werden. Als Datenformate stehen das Generic Simulation Interface (GSI) und das Simulation Control Protocol (SCP) zur Verfügung. Die GSI ist eine Schnittstelle, welche alle dynamischen und statischen Simulationsdaten, z.B. die Position, die Geschwindigkeit und die Beschleunigung anderer Verkehrsteilnehmer, beinhaltet.

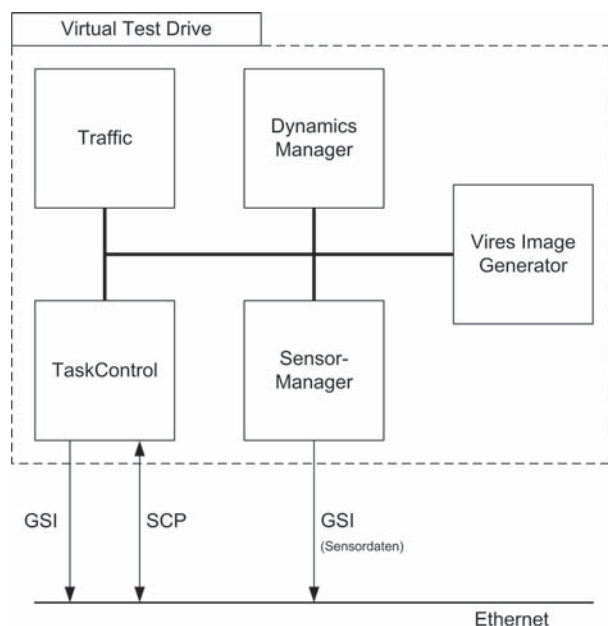


Abbildung 4.2.: Komponenten des Virtual Test Drive Frameworks

Zur Herstellung einer bidirektionalen Kommunikation zwischen Virtual Test Drive und weiteren Applikationen steht das SCP-Interface zur Verfügung. Durch diese Schnittstelle ist es möglich Kommandos an Virtual Test Drive zu versenden. Weiterhin werden Status-Informationen von Virtual Test Drive durch das SCP verteilt. Für eine ausführliche Beschreibung der Schnittstellen und des Frameworks sei auf das Handbuch von Virtual Test Drive verwiesen [186].

4.2.2. Architektur der Simulationsumgebung der Kamera

In Abbildung 4.3 ist die Architektur der Simulationsumgebung der Kamera dargestellt, welche an Stelle des Vires Image Generators verwendet wurde. Diese ist über ein GSI- und ein SCP-Interface mit dem Virtual Test Drive-Framework verbunden. Die empfangenden Daten werden in einem Shared-Memory Bereich gespeichert. Durch dieses Konzept kann die Grafik-Engine asynchron auf die Daten zugreifen und unabhängig von der Gesamtsimulation getaktet werden. Zur Stabilisierung der Bilderzeugung werden die empfangenden Daten der Fahrzeuge extrapoliert. Zur Steuerung der Simulation wurde eine Simulationssteuerung entwickelt, welche basierend auf einer Konfigurationsdatei eine Vorkonditionierung der Simulation durchführt. Die visuelle Datenbasis und die Konfiguration der Fahrzeuge werden in Abhängigkeit des in Virtual Test Drive ausgewählten Szenarios geladen. Zur Realisierung eines Replay-Modus können die Netzwerkdaten aufgezeichnet werden. Diese werden bei einer Wiedergabe zeitsynchron in den Shared-Memory Bereich kopiert. Die Erzeugung des Kamerabilds findet in einer eigenen Instanz des OpenSceneGraphs statt, welcher mit einer von Virtual Test Drive unabhängigen Taktfrequenz betrieben werden kann.

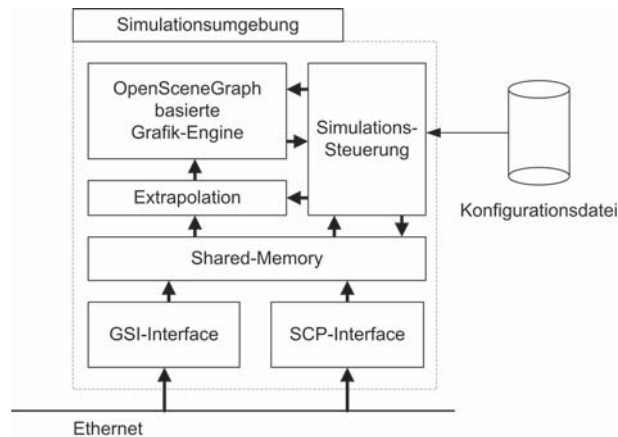


Abbildung 4.3.: Architektur der Simulationsumgebung der Kamera

4.2.3. OpenSceneGraph basierte Grafik-Engine

Zur Durchführung der Untersuchungen wurde eine auf OpenSceneGraph [137], OpenGL [175] und GLSL [157] basierte Grafik-Engine auf einem Linux Betriebssystem implementiert. Durch die Verwendung des OpenSceneGraphs kann die Geometrie der Datenbanken von Virtual Test Drive wiederverwendet werden. Zur Implementierung eigener Shader und Texturen wurde in der Grafik-Engine ein Modifikator integriert, welcher nach jedem Laden eines Szenarios die vorhandenen Oberflächenmaterialien gegen eigene Modelle austauscht. Zur Integration von Fahrzeugmodellen wurde eine eigene Beschreibungsform entwickelt. Die Architektur der Grafik-Engine ist der Realität nachempfunden, in welcher ein Umfeldmodell durch einen Sensor beobachtet wird. Daher teilt sich die Simulation in zwei gesonderte Simulationsmodelle. Das Umfeldmodell beinhaltet die Beleuchtungssimulation, Szenengeometrie und Oberflächenmodelle, vgl. die Abschnitte 4.3 und 4.4. Das Sensormodell oder Kameramodell beschreibt den Bildsyntheseprozess, vgl. Abschnitt 4.5. In Abbildung 4.4 sind die physikalischen Zusammenhänge vereinfacht dargestellt, die durch die Grafik-Engine beachtet werden sollen. Das Ziel der Simulation ist es eine hinreichend ge-

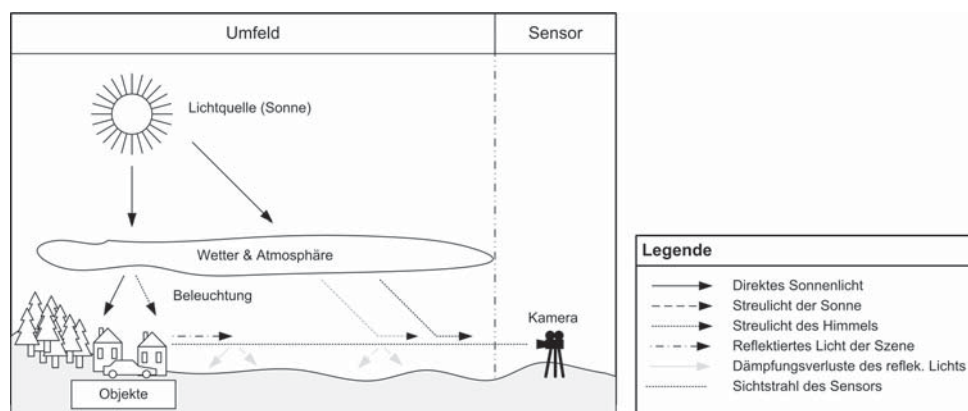


Abbildung 4.4.: Architektur der Grafik-Engine

naue Abbildung der Realität unter dem Aspekt der Interaktivität zu erreichen. Der Sensor, in diesem Fall die Kamera, empfängt die Lichtstärke welche über den Sichtstrahl in die Kamera einfällt. Die Strahlung des Strahls setzt sich aus dem reflektierten Licht der Szene, Streulicht und direkten Sonnenlicht zusammen. Weiterhin treten auf dem Übertragungsweg Dämpfungsverluste durch die Atmosphäre und die Wetterbedingungen auf. Das Umfeld besteht aus einer Lichtquelle zur Simulation des Sonnenlichts und Modellen für die atmosphärischen Bedingungen und das Wetter. Die atmosphärischen Modelle simulieren eine Dämpfung und Streuung des Sonnenlichts. Die von der Kamera aufgezeichnete Szene setzt sich aus einer beliebigen Anzahl von Objekten zusammen, deren Oberflächeneigenschaften hinreichend genau simuliert werden.

4.3. Simulation der Beleuchtung

In diesem Abschnitt werden Verfahren zur Erzeugung einer möglichst realitätsnahen Beleuchtung und zur Approximation von Schatten untersucht. Dabei wird besonderer Wert auf die Simulation der Sonnenstrahlung und von atmosphärischen Prozessen gelegt.

4.3.1. Beleuchtungsmodelle in der Computergrafik

In der Computergrafik kann im Wesentlichen zwischen zwei verschiedenen Beleuchtungsmodellen unterschieden werden, der globalen und lokalen Beleuchtung. Diese können mittels der von Kajiya [93] aufgestellten Rendergleichung beschrieben werden:

$$L(\mathbf{x}, \mathbf{x}') = g(\mathbf{x}, \mathbf{x}') \left(L_e(\mathbf{x}', \mathbf{x}'') + \int_{\Omega} f_r(\mathbf{x}', \mathbf{x}'') L(\mathbf{x}', \mathbf{x}'') d\mathbf{x}'' \right) \quad (4.1)$$

Die wesentliche Aussage dieser Gleichung ist, dass die zu dem Beobachter im Punkt \mathbf{x} reflektierte Strahlung eines Objekts am Punkt \mathbf{x}' im wesentlichen vom Material des Objekts f_r und dem Integral der aus allen Raumrichtungen eintreffenden Strahldichte L abhängt. Die eintreffende Strahlung kann von einer Lichtquelle stammen oder von anderen Objekten reflektiert werden. Der Fall, dass das Objekt selbst Strahlung emittiert wird durch den Term L_e behandelt. Zur Behandlung einer Absorption des Lichtvektors durch andere Geometrie ist die Funktion g enthalten, diese beschreibt die Durchlässigkeit einer Geometrie in Richtung der Reflexion. Im Falle, dass das komplette Licht absorbiert ist der Wert der Funktion Null.

Bei der globalen Beleuchtungsberechnung wird die Rendergleichung mittels numerischer Methoden in ihrer Gesamtheit approximiert. Die Lösung findet häufig auf Basis einer Integration basierend auf dem Monte-Carlo Ansatz mit einer bestimmten Rekursionstiefe statt. Beim Path-Tracing werden vom Punkt des Beobachters aus durch die Bildmatrix hindurch Strahlen emittiert und die Rendergleichung rekursiv von jedem Schnittpunkt aus gelöst [93]. Eine andere Möglichkeit ist das Light Ray-Tracing [17] bei welchem von der Lichtquelle die Rendergleichung gelöst wird. Häufig wird auch eine Kombination aus Path- und Light Ray-Tracing verwendet [91]. Bedingt durch rekursive Lösung der Rendergleichung und die daraus resultierende mehrfache Verdeckungsprüfung weisen die globalen Beleuchtungsverfahren eine hohe Komplexität auf. Zwar wurden in der Vergangenheit eine Vielzahl von Beschleunigungsverfahren entwickelt [174], dennoch ist es nach dem aktuellen Stand der Technik nicht möglich auf einem High-End PC eine komplexe Szene mit Bildwiederholungsraten von $f > 30Hz$ zu berechnen.

Aus diesem Grund hat sich in der Echtzeit-Computergrafik die Rasterisierung mit einem lokalen Beleuchtungsmodell etabliert [14] [175]. Das Verfahren der Rasterisierung wird in Abschnitt 4.5 und in [174] beschrieben. Als Ergebnis der Rasterisierung werden die Polygone der 3D-Modelle auf die Bildebene projiziert und in Fragmente, später Pixel, zerlegt. Anstelle die Rendergleichung in ihrer Gesamtheit zu lösen, wird bei der lokalen Beleuchtung die Rendergleichung nur für das aktuelle Fragment gelöst. Damit dies möglich ist wird die Verdeckungsfunktion g , eine Reflexion und Einstreuung von Licht vernachlässigt. In der Rendergleichung werden somit nur das Material der Oberfläche f_r und die Lichtquellen L_n berücksichtigt.

$$L(\mathbf{x}, \mathbf{x}') = \sum \frac{1}{r^2} f_r(\mathbf{x}', \mathbf{x}'') L_n(\mathbf{x}', \mathbf{x}'') \quad (4.2)$$

Durch diese Vereinfachungen kann mit Rasterisierungsverfahren keine direkte Schattenberechnung durchgeführt werden.

4.3.2. Schattenberechnung

Wie in Abschnitt 4.1.3 hergeleitet wurde sind besonders Schatten als Merkmale oder Störgröße für die Bildverarbeitung von großer Wichtigkeit. Daher werden verschiedene Techniken für die Approximation von Schatten untersucht. In der Literatur wird zwischen texturbasierten, Shadow-Maps, [194] und geometriebasierten Ansätzen, Shadow-Volumes, [45] unterschieden.

Shadow-Map Verfahren

In dieser Arbeit wurden texturbasierte Ansätze hinsichtlich einer Verwendbarkeit betrachtet. Durch Williams wird in der ursprünglichen Implementierung [194] der in der Grafikkarte integrierte Tiefenpuffer, Z-Buffer, für die Schattenberechnung verwendet. Zur Generierung der Shadow-Map wird die Szene aus der Position der Lichtquelle berechnet und die Tiefeninformationen des Z-Buffers in eine Textur gespeichert. Die Orientierung der Kamera ist durch die Ausrichtung der Lichtquelle definiert.

Die Berechnung des Schattens wird während der Berechnung des lokalen Beleuchtungsmodells ausgeführt, siehe Abbildung 4.5. Dazu wird jeder sichtbare Eckpunkt eines Polygons, Vertex, mittels der für die Erzeugung der Shadow-Map verwendeten Projektions- und View-Matrix in das Koordinatensystem Shadow-Map transformiert. Die x- und y-Komponente des transformierten Eckpunkts entsprechen einer Texturkoordinate \mathbf{t} in der Shadow-Map. Für den Vergleich wird die Shadow-Map an dieser Texturkoordinate ausgelesen. Der ausgelesene Tiefenwert $d_{Blocker}$ wird mit dem Tiefenwert des Eckpunkts d_{Object} verglichen. Wenn die Distanz des aktuellen Vertex zur Lichtquelle größer ist als die gespeicherte, so befin-

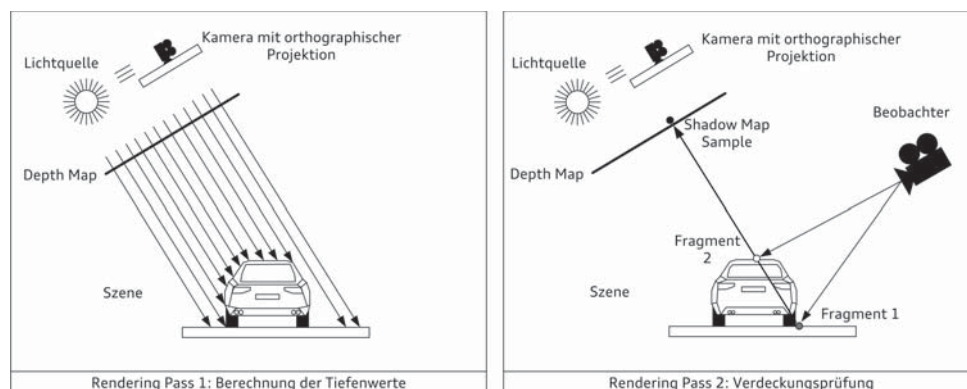


Abbildung 4.5.: Shadow Mapping Algorithmus

det sich das Fragment im Schatten, siehe Fragment 1. Wenn die Distanz des aktuellen Fragments kleiner oder gleich ist, so befindet sich das Fragment nicht im Schatten, siehe Fragment 2. Damit folgt für die Verdeckungsfunktion:

$$g(d_{Blocker}, d_{Object}) = \begin{cases} d_{Blocker} \leq d_{Object} & g(d_{Blocker}, d_{Object}) = 1 \\ d_{Blocker} > d_{Object} & g(d_{Blocker}, d_{Object}) = 0 \end{cases} \quad (4.3)$$

Bei Verwendung dieses Ansatzes liegt ein Fragment entweder im Schatten oder nicht. Daher entstehen keine weichen Übergänge an den Schattenkanten. Weiterhin treten durch die konstante Auflösung der Textur Aliasing-Effekte auf, der Schatten besitzt häufig eckige und harte Kanten.

Zur Erzeugung von weichen Schatten, Soft-Shadows, wurden in der Vergangenheit eine Vielzahl von Techniken entwickelt [83]. In dieser Arbeit wurde die Filterung mit einem Percentage Closer Filter [153] und eine Kombination aus einer Cascaded Shadow-Map (CSM) [52] und dem Percentage Closer Soft-Shadow (PCSS) [116] Algorithmus untersucht.

Durch den vorgeschlagen Percentage Closer Filter (PCF) von Reeves [153] wird die Shadow-Map mit einem konstanten Filter-Kernel gefaltet. Bei dieser Technik werden um den ursprünglichen Sampling-Punkt der Shadow-Map mehrere Samples $d_{Blocker,n}$ abgefragt. Die Samples sollten poissonverteilt sein. Für jeden Sample $d_{Blocker,n}$ wird mit $g(d_{Blocker,n}, d_{Blocker})$ geprüft, ob sich dieser näher zur Lichtquelle befindet als das aktuelle Fragment des Objekts, falls dies nicht der Fall ist wird der Wert des PCF um eins erhöht. Nach Abschluss des Samplings wird die Summe durch die Anzahl der Samples N dividiert und so die Verdeckung g_{PCF} ermittelt.

$$g_{PCF} = \frac{1}{N} \sum^N g(d_{Blocker,n}, d_{Blocker}) \quad (4.4)$$

Mittels dieser Technik ist es möglich weiche Schatten zu erhalten. Der Einfluss von Streulicht oder die Entfernung vom Boden wird vernachlässigt. Damit ergibt sich für jedes Objekt eine gleiche Schattenkante. Eine Möglichkeit zur Verbesserung der Erscheinung ist die Anwendung der Percentage Closer Soft-Shadow Technik nach Fernando [67] mit der Implementierung nach Myers [116]. Bei dieser Technik wird eine Skalierung der Abtastweite des PCF durch Berechnung der Penumbra vorgenommen. Die Größe der Penumbra wird unter der Annahme berechnet, dass die Oberfläche des Objekts und der verdeckenden Geometrie parallel zur Lichtquelle sind. Da in der Simulation die Verdeckung von der Sonne betrachtet werden soll, ist diese Annahme erfüllt. In Abhängigkeit von der Größe der Lichtquelle w_{Licht} und der Distanz zwischen dem verdeckenden Fragment $d_{Blocker}$ und dem aktuellen Fragment d_{Objekt} folgt für die Größe der Penumbra $w_{Penumbra}$:

$$w_{Penumbra} = \frac{(d_{Objekt} - d_{Blocker}) \cdot w_{Licht}}{d_{Blocker}} \quad (4.5)$$

Vergleich

In Abbildung 4.6 sind die drei beschriebenen Shadow-Map Verfahren im Vergleich zu einem realen Referenzbild dargestellt. Nahe dem Betrachter wurde ein weiteres Fahrzeug eingefügt, um die Auswirkungen der Schattentechnik auf Objekte in Bodennähe besser bewerten zu können. In allen Beispielen wurde eine

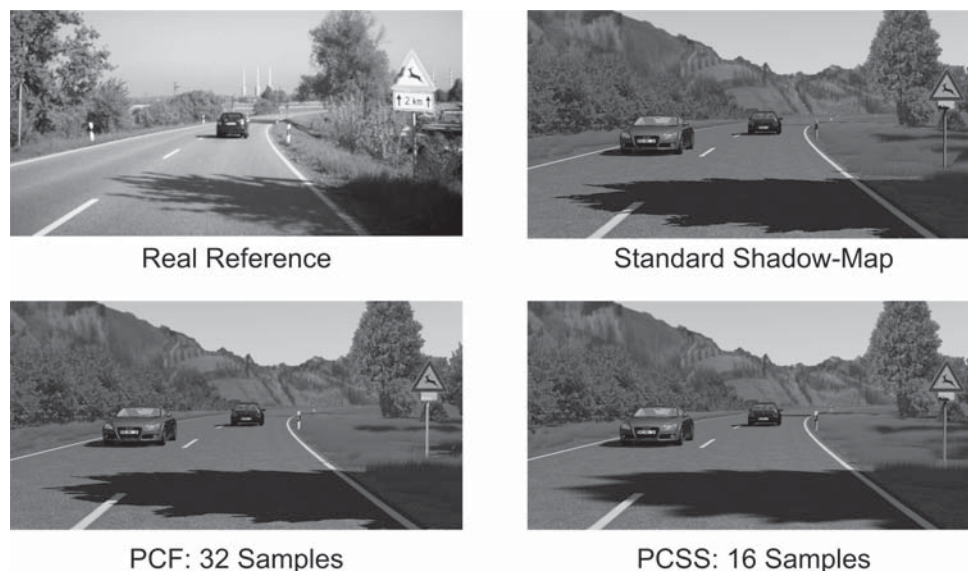


Abbildung 4.6.: Vergleichende Darstellung der Shadow-Maps

Shadow-Map mit 4096^2 Punkten verwendet, diese deckt eine Fläche von $1600m^2$ ab. Zur Erzeugung des PCF Soft-Shadows wurde ein Kernel mit Größe von 32×32 Pixeln verwendet. Abschließend wurde das Percentage Closer Soft Shadows (PCSS) Verfahren mit 16×16 Samples angewendet.

Zur Auswertung werden zwei Samples zwischen dem realen und den synthetischen Bildern verglichen, siehe Abbildung 4.7. Das erste Sample beinhaltet eine Schattenkante des Fahrzeugs. Das zweite Sample ist die Schattenkante eines von der Straße entfernt stehenden Baums. Weiterhin wurden zwei weitere Samples eingefügt. In Sample 3 ist die Verdeckung einer Straßenmarkierung durch einen Baum und in Sample 4 ist die Schattenkante eines Fahrzeugs dargestellt. Zur Auswertung werden die Helligkeiten und Gradientenbeträge der Schattenkante betrachtet. Bei Vergleich des ersten Samples kann festgestellt werden, dass die Schattenkante eine ähnliche Erscheinung wie in der Realität aufweist. Ein Unterschied bei Anwendung der verschiedenen Verfahren hinsichtlich der Qualität ist nicht feststellbar.

Im Gegensatz dazu kann bei Betrachtung der Schattenkante in Sample 2 festgestellt werden, dass mittels der PCSS-Technik eine der Realität sehr ähnliche Schattenkante erzeugt wird. Bei Anwendung der Standard-Shadow- oder PCF-Technik entstehen Schatten mit einer härteren Kante. Ein ähnliches Ergebnis kann auch in Sample 3 festgestellt werden. Weiterhin entstehen bei der Standard- und PCF-Shadow-Map unerwünschte Artefakte durch den Schatten an der Spurmarkierung. Bei Betrachtung des vierten Samples kann festgestellt werden, dass sich für den Schatten keine Abweichungen in Abhängigkeit des Verfahrens ergeben.

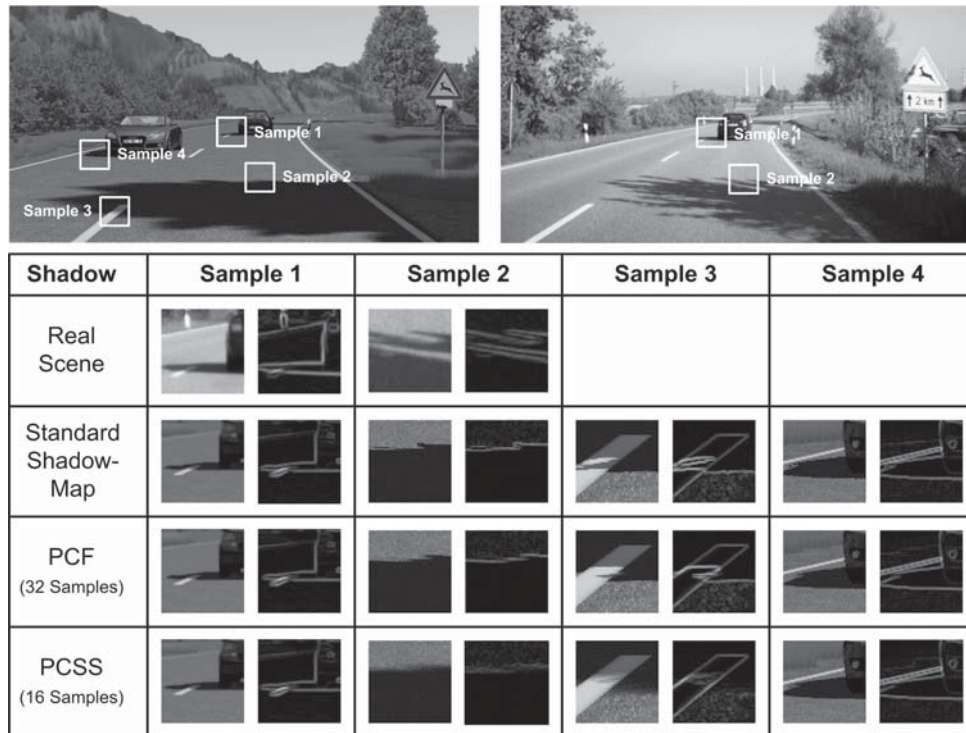


Abbildung 4.7.: Auswertung der Schattenverfahren

Schlussfolgerung

Im diesem Abschnitt wurden verschiedene Schatten-Techniken aus der Perspektive der Fahrzeugkamera hinsichtlich der Qualität untersucht. Bei der Auswertung wurde festgestellt dass es möglich ist für Objekte nahe der Straßenoberfläche, wie z.B. Fahrzeuge, eine normale Shadow-Map zu verwenden. Aufgrund des flachen Betrachtungswinkels und des kleinen Abstands zwischen Straßenoberfläche und verdeckenden Objekt ist keine große Penumbra feststellbar, siehe Sample 2 und 4.

Zur Integration von Schatten, welche durch Objekte weit über den Boden erzeugt werden, muss ein Verfahren mit variabler Penumbrabreite eingesetzt werden. Damit ist es außerdem möglich das Auftreten von Artefakten zu reduzieren, vgl. Sample 3. Zur Verbesserung der Schattenqualität wurde im Rahmen dieser Arbeit die Cascaded Shadow Map Technik von [52] verwendet, bei welcher das Viewing-Frustrum in mehrere Shadow-Maps unterteilt wird. Damit es möglich nahe dem Beobachter eine hohe Ortsauflösung in der Shadow-Map zu gewährleisten und für weiter entfernte Objekte eine niedriger aufgelöste Shadow-Map zu verwenden.

4.3.3. Atmosphärische Simulation

Die Erdatmosphäre ist in vier Schichten aufgeteilt: Troposphäre, Stratosphäre, Mesosphäre und Thermosphäre. Die unteren Schichten sind die Troposphäre und die Thermosphäre. Diese Schichten beinhalten 99% der atmosphärischen Gase und besitzen eine Ausdehnung von bis 12 km und 53 km [176]. Die Zusammensetzung der Atmosphäre besteht zu 78% aus Stickstoff und 21% aus Sauerstoff. Weiterhin beinhaltet die Atmosphäre Wasserdampf und Staubpartikel. Die Gasmoleküle und Partikel absorbieren bestimmte Wellenlängen des Sonnenlichts, z.B. absorbieren die Sauerstoff- und Ozon-Moleküle Licht im ultravioletten Spektrum. Durch Wasserdampf, Methan, Ozon, und Kohlenstoffdioxid wird infrarotes Licht absorbiert. Neben der Absorption tritt eine Streuung des Lichts auf. Daher kann Sonnenlicht ein- oder mehrfach in den Sichtstrahl eines Beobachters eingestreut werden [97].

4.3.3.1. Grundlagen zur atmosphärischen Simulation

Bevor auf die atmosphärische Simulation eingegangen wird, werden die drei Hauptprozesse für die Ausbreitung von Strahlung in einer Atmosphäre erläutert. Dazu wird auf die Definitionen und Ausführungen von [143] zurückgegriffen:

- Absorption: Dieser Prozess beschreibt die Reduktion der Lichtstrahlung durch Wandlung des Lichts in eine andere Energieform.
- Streuung: Die Streuung führt zu einer Richtungsänderung des Lichts durch Kollision mit Partikeln im Medium. Die Eigenschaften der Streuung hängen von der Größe und dem Brechungsindex des Partikels ab.
- Emission: Die Partikel geben Energie an die Umgebung ab.

Zur Simulation einer gashaltigen Atmosphäre sind das Absorptions- und Streuungsverhalten von Wichtigkeit. Eine Emission von Strahlung kommt in der Regel nicht vor. Daher werden die Absorption und die Streuung im Folgenden näher beschrieben.

Absorption

Die Absorptionsfähigkeit eines Materials wird durch den Absorptionsquerschnitt σ_{ab} definiert. Dieser drückt die Wahrscheinlichkeit der Absorption des Lichts durch das Medium aus. Das Absorptionsverhalten bestimmter Stoffe ist von der Wellenlänge λ abhängig [84], [143], daher absorbieren einige Stoffe stärker als andere bestimmte Wellenlängen. Zur Beschreibung des Absorptionsverhaltens eines Materials wird die optische Dichte T verwendet [84]. Diese definiert sich aus den Absorptionskoeffizienten β_{ab} und der Distanz x zwischen Lichtquelle oder Reflektor und dem Beobachter.

$$T = \beta_{ab}x \tag{4.6}$$

Der von der Wellenlängen abhängige Absorptionskoeffizient β_{ab} definiert sich wie folgt,

$$\beta_{ab}^\lambda = \rho_{ab} \sigma_{ab}^\lambda \quad (4.7)$$

dabei ist ρ_{ab}^λ die Dichte des absorbierenden Mediums. Die Absorptionskoeffizienten verschiedener Moleküle oder Partikel verhalten sich additiv. Auf Basis dieser Grundlagen kann die Abnahme der Strahldichte L bei der Durchquerung eines Mediums, anhand der folgenden Differentialgleichung beschrieben werden [84]:

$$-\frac{dL}{L} = \beta_{ab} dx \quad (4.8)$$

Durch das Lösen der Differentialgleichung folgt für die empfangende Strahldichte nach der Absorption $L_{ab}(x)$ der Lichtquelle L_0 nach der Wegstrecke x :

$$L_{ab}(x) = L_0 e^{-\int_0^x \beta_{ab}(x') dx'} \quad (4.9)$$

Mit der in Bodennähe gültigen Annahme, dass β_{ab} über die gesamte Strecke x konstant ist folgt:

$$L_{ab}(x) = L_0 e^{-\beta_{ab} x} \quad (4.10)$$

Streuung

Durch die Partikel und Gase in der Atmosphäre wird das Licht auf unterschiedliche Weise gestreut, die wichtigsten Streuungsprozesse sind die Rayleigh- und Mie-Streuung [135]. Weiterhin muss zwischen einer Aus- und Einstreuung in den Sichtstrahl des Beobachters unterschieden werden. Diese Begriffe sollen zuvor definiert werden, bevor auf die Streuungsprozesse eingegangen wird. Das Verhalten der Streuungsprozesse wird mittels einer Phasenfunktion $\Phi(\theta)$ beschrieben.

Streuung aus dem Sichtstrahl Der Prozess bei dem die Strahlung auf dem Weg von der Lichtquelle oder dem reflektierenden Objekt zum Beobachter den Sichtstrahl, bedingt durch die Berechnung an einem Molekül oder Partikel, verlässt wird als Aus-Streuung bezeichnet [84]. Das Streuungsverhalten des Mediums wird durch den Streuungsquerschnitt σ_{sc} quantifiziert. Basierend darauf wird analog zur Absorption der Streuungskoeffizient $\beta_{sc} = \rho_{sc} \sigma_{sc}$ mit der Dichte des Mediums ρ_{sc} definiert. Damit folgt für die empfangende Strahlung nach der Aus-Streuung L_{sc} auf Basis der gleichen Annahmen wie bei der Absorption,

$$L_{sc}(x) = L_0 e^{-\beta_{sc} x} \quad (4.11)$$

Dabei ist L_0 die Strahlungsdichte der Lichtquelle oder des reflektierten Objekts, β_{sc} der Streuungskoeffizient und x die Distanz zwischen Objekt und Beobachter.

Ein-Streuung Die Ein-Streuung beschreibt im Gegensatz zur Aus-Streuung, eine Erhöhung der Strahlung auf dem Sichtstrahl des Beobachters durch eingestreuete Strahlung. Der Streuungskoeffizient β_{sc} beschreibt nur den Anteil des gestreuten Lichts, aber nicht wie viel Licht in eine Richtung gestreut wird [143]. Zur Beschreibung dieses Zusammenhangs wird die Phasenfunktion $\Phi(\theta, \phi)$ verwendet. Diese beschreibt die Wahrscheinlichkeit einer Streuung des Lichts in die Richtung des Beobachters. Unter der Annahme, dass die Partikel sphärisch oder sehr klein im Vergleich zur Wellenlänge sind, hängt die Phasenfunktion ausschließlich vom Winkel zwischen der ursprünglichen und neuen Richtung θ ab. Daher kann die Phasenfunktion zu $\Phi(\theta)$ vereinfacht und der winkelabhängige Streuungskoeffizient $\beta(\theta)$ definiert werden.

$$\beta_{sc}(\theta) = \beta_{sc}\Phi(\theta) \quad (4.12)$$

Zur Bestimmung der eingestreuerten Strahlung in den Sichtstrahl über den gesamten Weg dx muss die Wahrscheinlichkeit eines Ein-Streuungsereignis betrachtet werden [84]. Mit der einfallenden Strahlung L_i und einen unendlich kleinen Raumwinkel $d\omega$ folgt,

$$dL_{In} = \beta_{sc}dx \int_{\Omega} L_i(\theta, \phi)\Phi(\theta)d\omega = dx \int_{\Omega} L_i(\theta, \phi)\Phi(\theta)d\omega \quad (4.13)$$

Da die eingestreuete Strahlung auch der Extinktion unterliegt, muss diese mit betrachtet werden. Diese wird an späterer Stelle definiert.

$$\frac{dL_{in}}{dx} = -\beta_{ex}L + \int_{\Omega} L_i(\theta, \phi)\Phi(\theta)d\omega \quad (4.14)$$

Damit folgt,

$$L_{in}(x) = L_0 e^{-\int_0^x \beta_{ex}(x')dx'} + \int_0^x \left(e^{-\int_{x'}^x \beta_{ex}(x'')dx''} \int_{\Omega} L_i(\theta, \phi)\beta_{sc}(\theta)d\omega \right) dx' \quad (4.15)$$

Rayleigh-Streuung Das Rayleigh-Modell beschreibt die Streuung von sehr kleinen Molekülen in der Erdatmosphäre. Wenn die Radien r der Partikel kleiner als die Wellenlänge des Lichts λ ($\frac{r}{\lambda} < 0.05$) sind ist das Rayleigh-Modell eine hinreichend genaue Annahme [97]. Die Rayleigh-Streuung ist vor allem für das blaue Erscheinungsbild des Himmels verantwortlich. Ebenso ergibt sich damit das orange Erscheinungsbild des Sonnenuntergangs, da das Licht einen weiteren Weg durch die Erdatmosphäre zurücklegen muss und somit das blaue und grüne Licht gestreut werden, bevor es den Beobachter erreicht. Die Phasenfunktion für die Rayleigh-Streuung ist [146]:

$$\Phi(\theta)_{Rayleigh} = \frac{3}{16\pi} (1 + \cos^2\theta) \quad (4.16)$$

Nach [147] wird der Streuungs-Koeffizient $\beta_{Rayleigh}$ wie folgt definiert,

$$\beta_{Rayleigh} = \frac{8\pi^3(n^2 - 1)^2}{3N\lambda^4} \left(\frac{6 + 3p_n}{6 - 7p_n} \right) \quad (4.17)$$

Wobei n der Brechungsindex der Luft, N die Anzahl der Moleküle und p_n der Depolarisationsfaktor der Luft ist.

Mie-Streuung Die Mie-Streuung wird durch größere Partikel (Aerosole), z.B. Staub, verursacht. Die Mie-Streuung sorgt dafür, dass an einem bewölkten Tag der Himmel grau erscheint und die Sonne eine weiße Korona besitzt. Zur Beschreibung wird häufig die Henyey und Greenstein Phasenfunktion Φ_{HG} verwendet [143]. Diese Phasenfunktion ist an Messdaten anpassbar und besitzt die Erscheinung einer Ellipse [84]. Die Streuung des Lichts wird durch den Asymmetrieparameter g beschrieben.

$$\Phi_{HG}(\theta) = \frac{1}{4\pi} \frac{1 - g^2}{(1 + g^2 - 2g(\cos(\theta)))^{3/2}} \quad (4.18)$$

Der gültige Wertebereich für g beträgt $(-1,1)$. Negative Werte für g beschreiben eine Rückstreuung, d.h. das meiste Licht wird in die Einfallsrichtung zurückgeworfen, und positive Werte beschreiben die Vorwärtsstreuung. Der Streukoeffizient β_{Mie} ist definiert durch folgende Gleichung:

$$\beta_{Mie} = 0.434c\pi \left(\frac{2\pi}{\lambda}\right)^{v-2} K \quad (4.19)$$

Der Parameter $c = (0.6544T - 0.6510) \cdot 10^{-16}$ beschreibt den Konzentrationsfaktor, v stellt Junges Exponent dar mit einem Standardwert von 4 für das Himmelsmodell [146]. Der Wert K schwankt von 0.656 bei $\lambda = 400nm$ und 0.69 bei $\lambda = 770nm$ [146].

Extinktion/Auslöschung Da sowohl die Absorption und die Aus-Streuung die Strahlungsleistung reduzieren wird dieses in der Literatur häufig unter dem Begriff Extinktion oder Auslöschung zusammengefasst. Die Strahlung nach der Auslöschung L_{ex} definiert sich wie folgt,

$$L_{ex}(x) = L_0 (e^{-\beta_{ab}x} \cdot e^{-\beta_{sc}x}) = L_0 e^{-(\beta_{ab} + \beta_{sc})x} \quad (4.20)$$

In der Veröffentlichung von Hoffmann [84] werden Absorptions- und Ausstreuungskoeffizient zum Extinktionskoeffizienten β_{ex} zusammengefasst.

$$\beta_{ex} = \beta_{ab} + \beta_{sc} \quad (4.21)$$

4.3.3.2. Übersicht zu Simulationsmodellen

Nachdem die grundlegenden Prozesse der Lichtausbreitung in einem nicht Vakuum beschrieben wurden, sollen im Folgenden Modelle zur Simulation einer Atmosphäre beschrieben werden. Durch Preetham [147] wird eine Einteilung in simulationsbasierte und empirische Modelle vorgenommen. Im Folgenden sollen einige Modelle vorgestellt werden.

Simulationsbasierte Modelle

Zur Simulation der Lichtübertragung in der Atmosphäre werden in der Regel die im vorherigen Abschnitt definierten Grundlagen der Haupt- und Streuungsprozesse verwendet. Die Lösung der Differentialgleichungen findet numerisch statt. Die Qualität der Modelle unterscheidet sich in der Abbildungsgenauigkeit der atmosphärischen Dichte, Zusammensetzung und Parametrierung [147]. Durch Sloup [176] wurde eine Einteilung der Modelle hinsichtlich einfacher und mehrfacher Simulation der atmosphärischen Streuung getroffen.

Als Grundlagenarbeit wird in allen betrachteten Veröffentlichungen die Arbeit von Klassen [97] als Referenz geführt. In dieser wird auf die Berechnung der Farbe der Sonne und des Himmels eingegangen. Weiterhin wird ein Verfahren zur Simulation von Nebel auf einer ebenen Fläche beschrieben. Die Schichten der Atmosphäre werden durch zwei Radien definiert. Der Abstand vom äußeren bis zum inneren Radius beschreibt die äußere Atmosphäre, der innere Radius bis zur Oberfläche definiert die Atmosphäre mit Dunst. Die Atmosphäre innerhalb eines Radius besitzt eine konstante Dichte. Eine Verschleierungsperspektive wird nicht behandelt. Zur Beschreibung der Himmelsfarbe wird eine einfache Streuung des Lichts berücksichtigt mit der Annahme, dass der Hauptteil der Streuung beim ersten Streuungsereignis passiert.

In Kaneda et. al. [94] wurde ein Ansatz zur Simulation von Partikeln der Atmosphäre unter verschiedenen Wetterbedingungen vorgestellt. Dabei wurde die spektrale Zusammensetzung der Sonnen- und Himmelslichts betrachtet. Die Rayleigh-Streuung wird in Abhängigkeit der Höhe durch eine Anpassung der atmosphärischen Dichte berechnet. Im Vergleich zu dem Modell von Klassen wurde eine sphärische Atmosphäre für die Mie-Streuung angenommen. Die Berechnung des Himmelslichts findet mittels numerischer Integration statt.

Durch Nishita [131] werden zwei Techniken zur Berechnung der Farbe des Himmels mit einfacher und mehrfacher Streuung für einen Betrachter am Boden vorgeschlagen. In dem Modell werden die Rayleigh- und die Mie-Streuung berücksichtigt. Zur Beschleunigung der Berechnungen wird eine Vorberechnung der optischen Dichte an jedem Abtastpunkt durchgeführt und in einer Look-Up Table gespeichert. Die Abbildung einer mehrfachen Streuung wird mittels einer zweistufigen Methode durchgeführt. Im ersten Schritt wird die Atmosphäre in eine bestimmte Anzahl von Volumen-Elementen, Voxel, aufgeteilt. Anschließend wird der Himmel in acht Richtungen zur Abtastung der Strahlung segmentiert. Für jedes dieser Voxel wird die Strahlung in Richtung der Abtastung bestimmt und in einer 3D-Look-Up Table gespeichert. Im zweiten Schritt wird die Strahlung entlang des Sichtvektors integriert und damit der Anteil am Streulicht bestimmt.

Durch ONeil [135] wird eine Umsetzung der Gleichungen von Nishita et. al. [131] auf einer GPU durch die Verwendung von Look-Up Tables beschrieben. Die Streuungs-Integrale werden mittels des Euler-Verfahrens durch fünf Stützstellen approximiert. Bei der Berechnung der Ein- und Aus-Streuung, wird eine Betrachtung des Druckabfalls zur Bestimmung der optischen Dichte durchgeführt. Weiterhin wird die Rückstreuung von der Oberfläche betrachtet. Zur Verbesserung der Echtzeitfähigkeit wurde die Abhängigkeit der optischen Dichte von der Höhe und dem Winkel zur Sonne mittels eines Polynoms approximiert. Eine weitere Verbesserung der Simulation ist in [106] beschrieben. Die Autoren stellen ein echtzeitfähiges Verfahren zur Darstellung der Atmosphäre in Echtzeit vor, welches vom Boden bis in den Weltraum

Gültigkeit besitzt und eine mehrfache Rayleigh- und Mie-Streuung berücksichtigt. Das Modell kann Tageslicht, Zwielflicht und die Verschleierungsperspektive darstellen. Der Ansatz verwendet eine für alle Sichtpunkte, Sichtrichtungen und Sonnenstände vorberechenbare Lichttransportgleichung.

Durch Hoffmann [84] wurde ein einfaches Modell zur Simulation der Streuung in Echtzeit entwickelt, dies beinhaltet eine Ein- und Aus-Streuung erster Ordnung durch die Rayleigh- und Mie-Streuung. Das Modell kann ohne großen Aufwand in einem programmierbaren Shader implementiert werden. Nach Aussage der Autoren kann es für die Simulation der Verschleierungsperspektive, Sonnenlicht und Himmelsfarbe verwendet werden. Wenn gleich die Simulation von Sonnenlicht und der Himmelsfarbe nur eingeschränkt möglich ist, da das Modell eine konstante atmosphärische Zusammensetzung annimmt.

Analytische & empirische Modelle

Durch Preetham et. al. [146] wurde ein analytisches Modell vorgeschlagen, welches das volle Spektrum des Tageslichts für verschiedene atmosphärische Bedingungen darstellen kann. Das Modell wurde mittels der Anpassung von einfachen Basisfunktionen an die Simulationsdaten des Nishita et. al.-Modells entwickelt. Die Parametrisierung des Modells findet mittels der Sonnenposition und eines Trübheitsparameters statt. In [199] wurde das Preetham-Modell kritisch betrachtet und Einschränkungen bzgl. der Parametrierbarkeit und Anwendbarkeit aufgezeigt, welche bei einer Anwendung beachtet werden müssen.

4.3.3.3. Modell für die Fahrsimulation

Für Betrachtung des Nutzens einer atmosphärischen Simulation wird der Ansatz nach Hoffmann [84] für die Verschleierungsperspektive und eine Kombination der Modelle von Preetham [146] und Hoffmann [84] für das Sonnen- und Himmelslicht umgesetzt. Diese Modelle sind für die Fahrsimulation ausreichend, da sich der Beobachter immer auf dem Bodenniveau befindet. Daher kann eine konstante Luftdichte angenommen werden. Aus Sicht der Bildverarbeitung sind besonders die spektrale Verschiebung des Sonnenlichts und die Verschleierungsperspektive von Interesse.

Simulation der Sonnenstrahlung

Eine Simulation des Tageslichts ist für eine simulationsbasierte Bewertung von Bildverarbeitungsfunktionen von Wichtigkeit. In den Anforderungen wurde herausgearbeitet, dass eine spektrale Verschiebung des Sonnenlichts Schwierigkeiten in der Wahrnehmung verursacht. Diese Verschiebung kann auf die Dämpfungs- und Streuungsprozesse in der Erdatmosphäre zurückgeführt werden.

Die von der Erdoberfläche empfangene Sonnenstrahlung hängt von der Position der Sonne am Himmel und den vorherrschenden atmosphärischen Bedingungen ab [132]. Daher gliedert sich die Simulation der Sonnenstrahlung in folgende Punkte:

- Berechnung der Sonnenstrahlung im RGB-Farbraum
- Berechnung der Sonnenposition in Abhängigkeit von der Tageszeit und dem Datum

Berechnung der Sonnenstrahlung im RGB-Farbraum

Als Grundlage zur Berechnung der Sonnenstrahlung im RGB-Farbraum in Abhängigkeit des Sonnenstands und der Erdatmosphäre dient die spektrale Strahlungsverteilung $E(\lambda)$ aus Tabelle 2 der Veröffentlichung von Preetham [146]. Zur Verifikation der Annahmen wurde die Arbeit von [75] herangezogen. Im ersten Schritt wurde die Strahlung in den CIE-Tristimulus Farbraum konvertiert, vgl. [174], und mit dem Raumwinkel der Sonne Ω [sr] multipliziert. Dieser definiert sich wie folgt:

$$\Omega = \frac{A}{r^2} \quad (4.22)$$

Dabei ist A der Durchmesser der Sonne und r der Abstand zwischen Erde und Sonne. Das Sonnenlicht wird auf dem Weg zum Boden durch Extinktion vermindert. Wenn sich die Sonne dem Horizont nähert erhöht sich der Anteil der Rayleigh-Streuung und das Sonnenlicht nimmt eine rötliche Färbung an [84]. Der Anteil der Streuung ist proportional zur Dichte der Atmosphäre. Daher wird das Sonnenlicht in der Atmosphäre mehrfach durch Kollisionen mit anderen Partikeln gestreut. Zur Berechnung der Sonnenstrahlung welche die Oberfläche erreicht, werden die Gleichungen von Preetham [146] verwendet. Diese beschreiben den Transmissionsgrad einzelner Schichten der Atmosphäre in Abhängigkeit der Wellenlänge, Luft-Moleküle τ_r , Aersole τ_a , Ozon τ_o , sonstige Gase τ_g und Wasserdampf τ_{wa} . Die Definition der Funktionen kann aus [146] entnommen werden. Für den gesamten Transmissionsgrad τ_{ges} der Erdatmosphäre folgt:

$$\tau_{ges}(\lambda) = \tau_r(\lambda)\tau_a(\lambda)\tau_o(\lambda)\tau_g(\lambda)\tau_{wa}(\lambda) \quad (4.23)$$

Damit ergibt sich für die Berechnung der Sonnenstrahlung im CIE-Tristimulus-Farbraum.

$$X = 683\Omega \int_{\lambda=380nm}^{780nm} (\bar{x}(\lambda)\tau_{ges}(\lambda)E(\lambda)) d\lambda \quad (4.24)$$

$$Y = 683\Omega \int_{\lambda=380nm}^{780nm} (\bar{y}(\lambda)\tau_{ges}(\lambda)E(\lambda)) d\lambda \quad (4.25)$$

$$Z = 683\Omega \int_{\lambda=380nm}^{780nm} (\bar{z}(\lambda)\tau_{ges}(\lambda)E(\lambda)) d\lambda \quad (4.26)$$

Als nächstes wird die Konvertierung in den RGB-Farbraum mittels Gleichung 20.5 aus [174] durchgeführt.

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 2.5623 & -1.1661 & -0.3962 \\ -1.0215 & 1.9778 & 0.0437 \\ 0.0752 & -0.2562 & 1.1810 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (4.27)$$

Danach wurde eine Skalierung der RGB-Werte mittels einer Normierung durchgeführt.

$$\begin{bmatrix} R_{norm} \\ G_{norm} \\ B_{norm} \end{bmatrix} = \begin{bmatrix} \frac{R}{R+G+B} \\ \frac{G}{R+G+B} \\ \frac{B}{R+G+B} \end{bmatrix} \quad (4.28)$$

Abschließend werden diese mit $Y = 127500 \left[\frac{lm}{m^2}\right]$ multipliziert, um die Sonnenstrahlung als Beleuchtungsstärke L_{Sonne} im RGB-Farbraum zu erhalten.

$$\mathbf{E}_{Sonne} = Y \begin{bmatrix} R_{norm} \\ G_{norm} \\ B_{norm} \end{bmatrix} \quad (4.29)$$

Berechnung der Sonnenposition in Abhängigkeit von der Tageszeit und dem Datum

Zur Berechnung der auf der Erde auftreffenden Sonnenstrahlung wird die Position der Sonne in Abhängigkeit von der Zeit, dem Datum und der Position auf der Erde benötigt. Die Position der Sonne wird durch den Polar- θ_s und Azimutwinkel ϕ_s beschrieben, siehe Abbildung 4.8. Die Berechnung der Sonnenposition

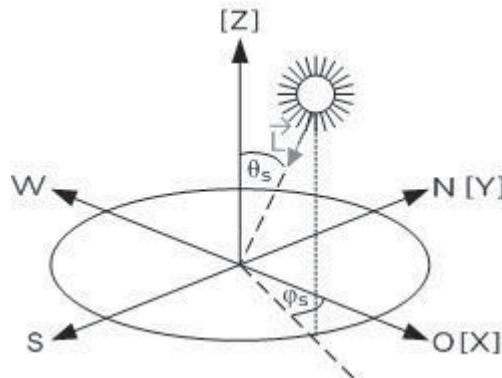


Abbildung 4.8.: Koordinatensystem zur Bestimmung der Sonnenposition, nach [146]

wird auf Basis des in [146] vorgeschlagenen Modells durchgeführt.

Anwendungsbeispiel

In Abbildung 4.9 sind die Ergebnisse der Simulation des Sonnenlichts dargestellt. Wie ersehen werden kann nimmt das Sonnenlicht während der Morgen- und Abendröte eine rötliche Färbung an. Während des Auf- und Untergangs der Sonne verschiebt sich das Spektrum in Richtung orange. Am Tag nimmt das Licht eine weiße Färbung an. Weiterhin kann festgestellt werden, dass sich die Schatten entsprechend der Position der Sonne kausal verändern. Im Rahmen der Fallstudien in Kapitel 6 wurde verifiziert, dass die Schatten entsprechend der Realität geworfen werden. Als vorteilhaft erweist sich, dass die Position der Sonne in Abhängigkeit der Uhrzeit und des Datums definiert werden kann. Außerdem verändert

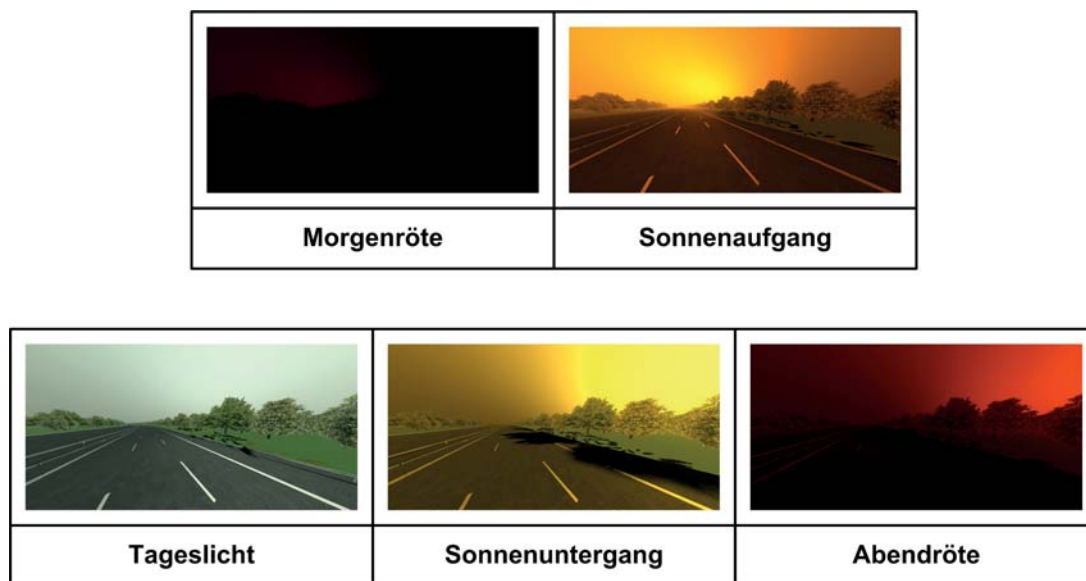


Abbildung 4.9.: Ergebnisse der Simulation des Sonnenlichts im RGB-Farbraum

sich die spektrale Zusammensetzung in Abhängigkeit von der Sonnenposition und den atmosphärischen Bedingungen. Als nachteilig kann festgestellt werden, dass es für gewisse Anwendungsfälle notwendig sein kann, die Position der Sonne und die Farbe der Sonnenstrahlung direkt festzulegen. Daher sollte diese Möglichkeit in der Simulation gegeben sein.

Verschleierungsperspektive „Aerial Perspective“

Die Verschleierungsperspektive beschreibt die Veränderung der Erscheinung von Objekten durch die Interaktion mit Partikeln der Atmosphäre in Abhängigkeit von der Entfernung zum Beobachter [84]. Wie in Abbildung 4.4 dargestellt, wird das durch Objekte reflektierte Licht auf den Weg zur Kamera teilweise ausgelöscht. Außerdem findet eine Einstreuung von Sonnen- und Himmelslicht in den Sichtstrahl statt [146]. Nach [84] werden bei besonders klarer Luft durch die Rayleigh-Streuung vor allem kürzeren Wellenlängen ausgelöscht. Während des Tages tritt ein nicht vernachlässigbarer Lichtanteil von Einstreuung auf, welcher vor allem aus kurzen Wellenlängen besteht. Durch Preetham et. al. [147] wird angeführt, dass der Hauptanteil des eingestreuten Lichts direkt von der Sonne stammt. Daher kann Streulicht zweiter Ordnung, ohne einen Qualitätsverlust in der Simulation, vernachlässigt werden. Zur Beschreibung schlagen Hoffmann et. al. [84] das folgende Modell zur Berechnung der Verschleierungsperspektive L_{Aerial} vor:

$$L_{Aerial}(x, \theta, \lambda) = L_0 f_{ex}(x, \lambda) + L_{in}(x, \theta, \lambda) \quad (4.30)$$

In der Gleichung stellen x den Abstand des Reflektors zur Kamera, θ den Winkel zwischen Sichtvektor und Lichtvektor, L_0 das reflektierte Licht und L_{in} das eingestreute Licht dar. Für den Extinktionsfaktor $f_{ex}(\lambda)$ folgt,

$$f_{ex}(\lambda) = e^{-(\beta(\lambda)_{Rayleigh} + \beta(\lambda)_{Mie})x} \quad (4.31)$$

Für das eingestreuete Licht $L_{in}(\lambda)$

$$L_{in}(x, \theta, \lambda) = \frac{\beta_{Rayleigh}(\theta, \lambda) + \beta_{Mie}(\theta, \lambda)}{\beta_{Rayleigh}(\lambda) + \beta_{Mie}(\lambda)} E(\lambda)_{Sonne} (1 - f(\lambda)_{ex}) \quad (4.32)$$

mit den winkelabhängigen Streuungsfunktionen.

$$\beta_{Rayleigh}(\theta, \lambda) = \beta_{Rayleigh}(\lambda) \Phi_{Rayleigh}(\theta, \lambda) \quad (4.33)$$

und

$$\beta_{Mie}(\theta, \lambda) = \beta_{Mie}(\lambda) \Phi_{Mie}(\theta, \lambda) \quad (4.34)$$

Im Rahmen dieser Arbeit wurde eine spektrale Berechnung der Verschleierungsperspektive in einem programmierbaren Shader implementiert. Dazu werden die Gleichungen zur Berechnung der Sonnenstrahlung während der Laufzeit der Simulation berechnet.

Himmelslicht

Die Farbe des Himmels ergibt sich aus der Einstreuung von Sonnenlicht in den Sichtstrahl. Dabei führt Rayleigh-Streuung zur blauen Farbe des Himmels während des Tages und die Mie-Streuung zur der grauen Farbe eines bewölkten Himmels. Zur Simulation des Himmelslichts sind nach [84] sehr genaue Dichtefunktionen der Atmosphäre notwendig, da das Licht auf den Weg zum Boden mehrfach in der Atmosphäre gebrochen und aus verschiedenen Richtungen in den Sichtstrahl des Beobachters eingestreuert wird. Eine physikalische Betrachtung dieses Sachverhalts macht eine numerische Integration notwendig [147]. Nach [147] ist die Anwendung von analytischen Modellen eine Alternative.

Im Rahmen dieser Arbeit wurden das Modell von Preetham [146] und das Modell von Hoffman [84] hinsichtlich ihrer Anwendbarkeit untersucht.

Preetham-Modell Durch Preetham et. al. [146] wurde ein empirisches Modell basierend auf dem Perez et. al.-Modell [142] zur Simulation von wolkenlosen und bewölkten Himmel entwickelt:

$$F(\theta, \gamma) = (1 + Ae^{B/\cos(\theta)})(1 + Ce^{D\gamma} + E\cos^2\gamma) \quad (4.35)$$

Wobei die Koeffizienten A , B , C , D und E die Verteilung der Helligkeit darstellen. Die Helligkeit des Himmels Y ergibt sich mit dem Perez-Modell wie folgt:

$$Y = Y_z \frac{F(\theta, \gamma)}{F(0, \theta_s)} \quad (4.36)$$

Mit Y_z der Helligkeit am Zenit, θ dem Winkel zwischen Zenit und Beobachter, θ_s dem Winkel zwischen Sicht- und Lichtvektor. Die Bedatung des Modells wurde mit Werten aus dem Nishita et. al.-Modell [131] durchgeführt. Als Parameter zur Anpassung der Bewölkung dient der „Turbidity“-Parameter T . Das Modell liefert die Farbe des Himmels im CIE-Tristimulus Farbraum.

$$Y = Y_z \frac{F(\theta, \gamma)}{F(0, \theta_s)} \quad (4.37)$$

$$x = x_z \frac{F(\theta, \gamma)}{F(0, \theta_s)} \quad (4.38)$$

$$y = y_z \frac{F(\theta, \gamma)}{F(0, \theta_s)} \quad (4.39)$$

Die Modellstruktur ist dem Perez-Modell nachempfunden. Die Konvertierung aus dem CIE-Tristimulus Farbraum kann in [146] nachgelesen werden.

Hoffmann-Modell Durch Hoffmann [84] wird angeführt, dass mit dem Modell zur Simulation der Einstrahlung von Licht in den Sichtstrahl des Beobachters ein Himmelsmodell dargestellt werden kann. In der Veröffentlichung weist Hoffmann darauf hin, dass es zu Abweichungen in der Erscheinung des Himmels kommt. Als Grund wird angeführt, dass das Modell eine starke Vereinfachung der atmosphärischen Prozesse darstellt. Für die Farbe des Himmels folgt mit dem Hoffmann-Modell:

$$L_{Sky}(x, \theta) = L_{in}(\lambda)(x, \theta) \quad (4.40)$$

mit $L_{in}(\lambda)$

$$L_{in}(x, \theta, \lambda) = \frac{\beta_{Rayleigh}(\theta, \lambda) + \beta_{Mie}(\theta, \lambda)}{\beta_{Rayleigh}(\lambda) + \beta_{Mie}(\lambda)} E(\lambda)_{Sonne} (1 - f(\lambda)_{ex}) \quad (4.41)$$

Vergleich der Modelle Zur Bewertung der Anwendbarkeit der Modelle wurde die Erscheinung des Himmels über einen gesamten Tag von 6 Uhr bis 17 Uhr bewertet. Als Werte für die Turbidity wurde $T = 1.1$ und für die Asymmetrie wurde $g = 0.2$ verwendet. Es kann festgestellt werden, dass die Modelle während der Morgenröte eine ähnliche Erscheinung erzeugen. Während des Sonnenaufgangs erstrahlt beim Hoffmann-Modell der gesamte Himmel in orange. Das Preetham-Modell hingegen erzeugt eine grünliche Färbung. Nach dem Sonnenaufgang um 8 Uhr besitzt der Himmel beim Hoffmann-Modell eine orange Färbung, das Preetham nimmt eine türkise Farbe an. Bei beiden Modellen ist die Färbung des Himmels von 9 Uhr bis 15 Uhr konstant. Im Hoffmann-Modell ist der Himmel gräulich gefärbt, während hingegen das Preetham-Modell die türkise Farbe beibehält. Bei der Abenddämmerung nimmt der Himmel beim Hoffmann-Modell eine gelbliche Färbung an, das Preetham-Modell behält die gelbliche Färbung. Während des Sonnenuntergangs nimmt der Himmel erneut eine rötliche Färbung beim Hoffmann-Modell an. Das Preetham-Modell erzeugt an der Position der Sonne eine rötliche Färbung im weiteren Bereich des Himmels ist es lila. Abschließend kann festgestellt werden, dass das Hoffmann-

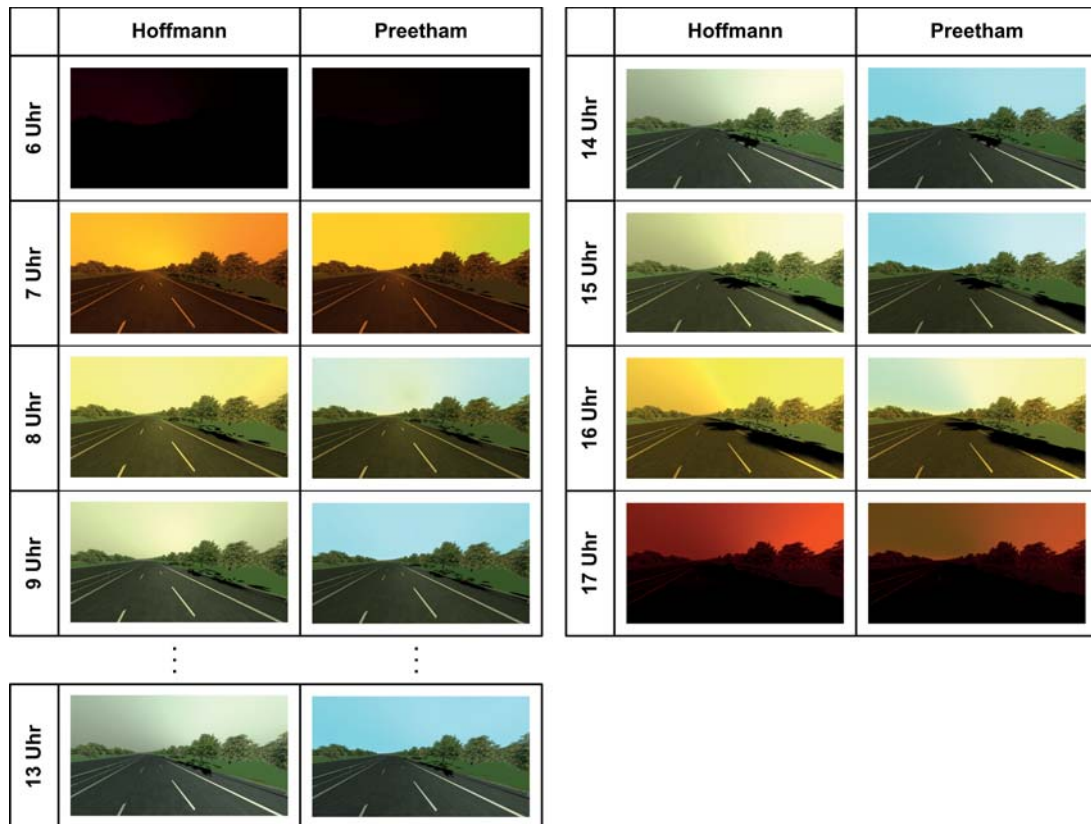


Abbildung 4.10.: Vergleich des Hoffmann- und des Preetham-Modells

Modell vor allem während Dämmerungsphasen eine deutliche bessere Anmutung des Himmels erzielt als das Preetham-Modell. Während des Tages erzeugt das Preetham-Modell eine sehr realistische Färbung des Himmels. Die gräuliche Farbe des Hoffmann-Modells ergibt sich aus der Parametrierung der Turbidity. Diese sollte während des Tages reduziert werden. Wie in der Einleitung zum diesem Kapitel beschrieben wurde, ist am Vormittag der Anteil an gelösten Aerosolen in der Atmosphäre geringer. Am Abend steigt dieser durch verdunstetes Wasser an. Daher erscheint vor allem der Sonnenuntergang mit einer großen Korona. Aus technischer Sicht muss dem Hoffmann-Modell dem Vorzug gegeben werden. Durch die freie Parametrierbarkeit kann die Erscheinung des Himmels frei modifiziert werden. Weiterhin ist die optische Erscheinung der Verschleierungsperspektive durchgängig, da gleiche Modelle verwendet werden. Bei Anwendung des Preetham-Modells kann es zu einer Inkonsistenz zwischen Himmel und Boden kommen.

4.3.3.4. Anwendungsbeispiel

Zur Bewertung der Anwendbarkeit des atmosphärischen Modells wurden verschiedene Parametrierungen untersucht. Nach Narasimhan und Nayar [118] können verschiedene Wetterbedingungen durch die Art,

4.3. SIMULATION DER BELEUCHTUNG

die Größen und die Anzahl von Partikeln im Raum beschrieben werden. Die Autoren definieren die Wetterbedingungen: Dunst, Nebel und Regen. Dunst soll als Aerosol definiert werden in welchem Partikel, z.B. Asche oder Ruß, gelöst sind. Die Partikel reagieren schnell auf Änderungen der Luftfeuchtigkeit und können als Keim für die Entstehung von Wassertropfchen dienen. Dunst kann als grauer Schleier oder bläuliche Färbung wahrgenommen werden und beeinflusst die Sichtweite.

Nebel entsteht in mit Feuchtigkeit gesättigter Luft unter der Bedingung, dass die als Keime dienenden Dunstpartikel auf die Größe eines Wassertropfens anwachsen. Während Dunst bis einer Höhe von mehreren Kilometern wahrnehmbar ist, ist Nebel auf eine Höhe von wenigen hundert Metern beschränkt.

Im Vergleich zu Dunst oder Nebel ist Regen keine statische Wetterbedingung, da sich die Verteilung und Intensität der Regentropfen zeitlich verändert. Daher ist die Simulation von Regen deutlich komplexer. Nach [118] entsteht Regen in einer mit Wasserdampf gesättigten Luft durch Kondensation. Wenn die kondensierten Tröpfchen schwerer als der in der Atmosphäre wirkende Auftrieb sind, beginnen diese als Regen zu fallen.

Durch [119] werden Werte für die Parametrierung der Henyey-Greenstein Funktion vorgeschlagen. Die Tabelle wurde mit den Partikelgrößen von Narasimhan [118] erweitert. Diese Tabelle wurde als Grundlage für die Bedatung des vorher hergeleiteten Modells verwendet.

Wetterbedingung	Partikel-Art	Radius [μm]	Konzentration [m^3]	Asymmetrie (g)
Luft	Molekül	10^{-4}	10^{25}	0.0 – 0.2
Dunst	Aerosol	$10^{-2} - 1$	$10^9 - 10^6$	0.2 – 0.8
Nebel	Wasser-Tröpfchen	1 – 10	$10^8 - 10^7$	0.8 – 0.9
Regen	Wasser-Tropfen	$10^2 - 10^4$	$10^4 - 10$	0.9 – 1.0

Tabelle 4.1.: Parametrierung der Henyey-Greenstein Funktion

Ergebnisse

In Abbildung 4.11 sind die Ergebnisse der Simulation von verschiedenen Wetterbedingungen dargestellt. Durch Anpassung der Parametrierung können verschiedene atmosphärische Bedingungen in der Simula-





	Sonnenschein	Bewölkung	Nebelig	Regenerisch
				
Turbidity	T = 1.0	T = 1.4	T = 6.4	T = 1.7
Asymmetrie	g = 0.2	g = 0.6	g = 0.6	g = 0.4

Abbildung 4.11.: Wetterbedingungen und Parametrierung des atmosphärischen Modells

tion hergestellt werden. Bei Anwendung der Daten aus Tabelle 4.1 muss angemerkt werden, dass diese für Nebel und Regen nur eine bedingte Gültigkeit haben. Das atmosphärische Modell geht von einer konstanten Dichte der Atmosphäre aus. Da die Konzentration der Partikel bei Nebel und Regen deutlich

geringer ist als bei Dunst, musste für visuell ansprechende Ergebnisse eine andere Parametrierung der Asymmetrie gewählt werden.

4.3.4. Zusammenfassung

In diesem Kapitel wurden verschiedene Techniken zur Realisierung einer realitätsnahen Beleuchtung in der Fahrsimulation untersucht.

Da in der Echtzeit-Computergrafik nur lokale Beleuchtungsmodelle verwendet werden ist eine Approximation von Schatten notwendig. Deshalb wurden verschiedene Schatten-Techniken aus der Perspektive der Fahrzeugkamera hinsichtlich der Qualität untersucht. Bei der Auswertung wurde festgestellt dass es möglich ist für Objekte nahe der Straßenoberfläche, wie z.B. Fahrzeuge, eine normale Shadow-Map zu verwenden. Aufgrund des flachen Betrachtungswinkels und des kleinen Abstands zwischen Straßenoberfläche und verdeckenden Objekt ist keine große Penumbra feststellbar.

Zur Integration von Schatten welche durch Objekte weit über dem Boden erzeugt werden, muss ein Verfahren mit variabler Penumbrabreite eingesetzt werden. Damit ist es außerdem möglich das Auftreten von Artefakten zu reduzieren. Zur Verbesserung der Schattenqualität wurde im Rahmen dieser Arbeit die Cascaded Shadow Map Technik von [52] verwendet, bei welcher das Viewing-Frustrum in mehrere Shadow-Maps unterteilt wird. Damit es möglich nahe dem Beobachter eine hohe Ortsauflösung in der Shadow-Map zu gewährleisten und für weiter entfernte Objekte eine niedriger aufgelöste Shadow-Map zu verwenden.

Zur Simulation der Lichtausbreitung und Atmosphäre wurde ein Modell für die Fahrsimulation, basierend auf den Ausführungen von Preetham [146] und Hoffmann [84] entwickelt.

Einleitend wurden die atmosphärischen Grundlagen beschrieben und ein Überblick der Simulationsmethoden gegeben. Zur Berechnung der Sonnenstrahlung wurde die Position der Sonne in Abhängigkeit der Uhrzeit und des Datums bestimmt. Die spektrale Verschiebung der Sonnenstrahlung ist in Abhängigkeit der Uhrzeit definierbar. Die Simulation der Himmelsfarbe wurde mit den Modellen von Preetham und Hoffmann untersucht. Bei einem Vergleich der Modelle wurde festgestellt, dass das einfachere Hoffmann-Modell den besten Kompromiss für den gesetzten Anwendungsfall darstellt. Weiterhin ist die Erscheinung der Verschleierungsperspektive durchgängig. Bei Anwendung des Preetham-Modells kann es zu einer Abweichung zwischen der Himmelsfarbe und dem Nebel kommen.

Abschließend wurde die Parametrierung der atmosphärischen Simulation für die Realisierung von verschiedenen Witterungen betrachtet. Es zeigte sich, dass das Modell besonders für die Simulation von Dunst oder klaren Himmel geeignet ist. Für eine Simulation von Nebel oder Regen müssen die Parametrierungen empirisch ermittelt werden.

4.4. Geometrie und Oberflächeneigenschaften von Objekten

Zur Stimulation der Signalverarbeitung müssen die zu detektierenden Objekte eine ausreichende Qualität in der geometrischen Modellierung und in der Oberflächenmodellierung aufweisen. Daher werden für die Oberflächenmodelle von Fahrzeugen und der Straße verschiedene Qualitätsgrade hinsichtlich der Auswirkung für die Bildverarbeitung betrachtet. Weiterhin wird die Qualität von texturierten Verkehrsschildern in der Simulation und in der Realität verglichen.

4.4.1. Oberflächenmodelle

Im Folgenden sollen die Grundlagen zur Beschreibung der visuellen Eigenschaften von Oberflächen in der Computergrafik vorgestellt werden. Zur Beschreibung dieser gilt die Annahme, dass eine auftreffende Bestrahlungsstärke E entweder durch das Medium reflektiert oder absorbiert wird. Die Reflexions- und Absorptionseigenschaften eines Materials werden mit bidirektionalen Reflektionsverteilungsfunktionen (BRDF) f_r beschrieben. Diese sind durch das Dokument „Geometrical Considerations and Nomenclature for Reflectance“ [129] des National Bureau of Standards definiert. Nach [14] folgt:

$$f_r(\theta_i, \phi_i, \theta_r, \phi_r) = f_r(\mathbf{l}, \mathbf{v}) = \frac{dL_o(\mathbf{v})}{dE(\mathbf{l})} \quad (4.42)$$

Die Variablen θ_i und ϕ_i entsprechen dem Polar- und Azimutwinkel zwischen dem Lichtvektor \mathbf{l} und der Normalen der Oberfläche \mathbf{n} am Bestrahlungspunkt. Der Polar- und Azimutwinkel der Reflexion wird mit θ_r und ϕ_r bezeichnet. Diese definieren sich aus der Normalen \mathbf{n} und dem Sichtvektor \mathbf{v} . Mit Hilfe dieser Eingangsgrößen wird der Reflexionsquotient der BRDF f_r definiert. Dieser beschreibt das Verhältnis der einfallenden differentiellen Bestrahlungsstärke dE und der in Richtung des Betrachters reflektierten differentiellen Leuchtdichte dL_o . Für weiterführende Beschreibungen sei auf [14] verwiesen. Zur Darstellung von Glanzeffekten wurde von Phong [144] ein empirisches Modell vorgeschlagen, welches sich wie folgt definiert:

$$f_{r,Phong} = k_s \cdot \frac{\cos(\phi)^{n_s}}{\cos(\theta)} = k_s \cdot \frac{(\mathbf{r} \cdot \mathbf{v})^{n_s}}{(\mathbf{n} \cdot \mathbf{l})} \quad (4.43)$$

Wobei k_s den Reflektionskoeffizienten des Glanzlichts darstellt. Der Beitrag des Koeffizienten zum Reflexionsverhalten definiert sich auf Basis des Quotienten aus dem Skalarprodukt des Reflexionsvektors \mathbf{r} und dem Sichtvektor \mathbf{v} und dem Skalarprodukt der Normalen \mathbf{n} und des Richtungsvektors des Lichts \mathbf{l} . Zur Skalierung der Ausprägung des Glanzlichtpunkts dient der Exponent n_s . Kleine Werte für n_s führen zu einem unscharfen Glanzlicht, große Werte für n_s zu einem scharfen Glanzlicht. Zur Beschreibung einer Oberfläche wird das Oberflächenmodell um den lambertschen Term erweitert, damit folgt für die BRDF:

$$f_{r,Phong} = \frac{k_d}{\pi} + k_s \cdot \frac{(\mathbf{r} \cdot \mathbf{v})^{n_s}}{(\mathbf{n} \cdot \mathbf{l})} \quad (4.44)$$

Durch Blinn [24] wurde eine effizientere Version des Oberflächenmodells vorgeschlagen. In diesem wird der Reflexionsvektor \mathbf{r} durch den sog. Halfway-Vektor $\mathbf{h} = \frac{\mathbf{l}+\mathbf{v}}{|\mathbf{l}+\mathbf{v}|}$ approximiert. Neben einer Verbesserung der Berechnungsgeschwindigkeit soll damit auch die optische Erscheinung verbessert werden. Damit folgt für das Oberflächen-Modell:

$$f_{r, \text{Blinn-Phong}} = \frac{k_d}{\pi} + k_s \cdot \frac{(\mathbf{n} \cdot \mathbf{h})^{n_s}}{(\mathbf{n} \cdot \mathbf{l})} \quad (4.45)$$

Die Blinn-Phong-BRDF ist das in der Fixed-Function Pipeline von Open-GL implementierte Modell zur Darstellung von Oberflächen [175].

4.4.2. Fahrzeuge

Im Folgenden wird auf die Simulation von Fahrzeugen zur Gewährleistung einer hinreichend genauen Darstellung in der Fahrsimulation eingegangen. Dazu wird die Geometrie der Fahrzeugmodelle und die Simulation von Fahrzeuglack betrachtet. Weiterhin wird auf die Simulation von Frontscheinwerfern und Heckleuchten eingegangen.

4.4.2.1. Geometrische Modellierung

Die geometrische Definition der Fahrzeugmodelle findet auf Basis von frei verfügbaren Modellen realer Fahrzeuge mit einer Polygonanzahl zwischen 30000 und 40000 Dreiecken statt. In Abbildung 4.12 sind fertig modellierte Fahrzeuge dargestellt. Zur Realisierung von Animationen und der späteren Definition



Abbildung 4.12.: Fahrzeugmodelle

von Oberflächeneigenschaften durch programmierbare GLSL-Shader erfolgt eine Zerlegung des Modells in verschiedene Komponenten, siehe Anhang A.7. Der Aufbau des Szenengraphen orientiert sich an realen Fahrzeugen, die geometrischen Elemente werden in Aufbau und Chassis aufgeteilt. Der Aufbau beinhaltet alle starr mit dem Fahrzeug verbundenen geometrischen Elemente. Diese können eine Nick-, Wank- und Gierbewegung des Fahrzeugs ausführen. Dies sind der Rahmen, alle Glas-Elemente, die Karosserie und das Kennzeichen. Alle Elemente des Fahrwerks werden dem Chassis zugeschrieben, in diesem Fall sind dies die Räder und die Bremsen. Diese Elemente befinden sich in der Visualisierung immer auf der Bodenebene. Daher werden der Gierwinkel des Fahrzeugs und die Gierwinkel und Rotationswinkel der Räder berücksichtigt. Ein Sturzwinkel wird nicht betrachtet. Die Transformation der Fahrzeugposition wird ebenfalls global betrachtet.

4.4.2.2. Oberflächenmodelle

Die äußerliche Erscheinung von Fahrzeugen wird maßgeblich von zwei Materialien beeinflusst: Glas und Fahrzeuglack. Während sich die Simulation von Glas durch Anwendung der physikalischen Gesetze von Reflexion und Berechnung umsetzen lässt [175], ist die Simulation von Fahrzeuglack eine Herausforderung [77]. Da die Erscheinung der Karosserie des Fahrzeugs eines der wichtigsten Merkmale für die Fahrzeugerkennung ist, werden verschiedene Oberflächenmodelle hinsichtlich Qualität bewertet. In Abbildung 4.13 ist die Erscheinung verschiedener Lackmodelle gleicher Grundfarbe dargestellt.



Abbildung 4.13.: Oberflächenmodelle der Fahrzeug-Karosserie

Fahrzeuglackmodelle

Nach [160] setzt sich der Fahrzeuglack aus mehreren Schichten zusammen. Die Basis-Schicht wird als Substrat bezeichnet. Bei modernen Fahrzeugen ist dies eine galvanisierte Schicht Zinn. Diese ist mit einer aus weißen oder hellgrauen Puder bestehenden Grundierung abgedeckt. Licht welches auf diese Schicht auftrifft wird diffus gestreut. Auf diese Grundierung wird die Hauptschicht des Lacks aufgebracht. Diese Schicht besteht aus Bindemittel und Farbpigmenten, welche eine Streuung und Absorption des Lichts verursachen. Im Falle von metallischen Lacken ist in dieser ebenfalls eine große Anzahl an

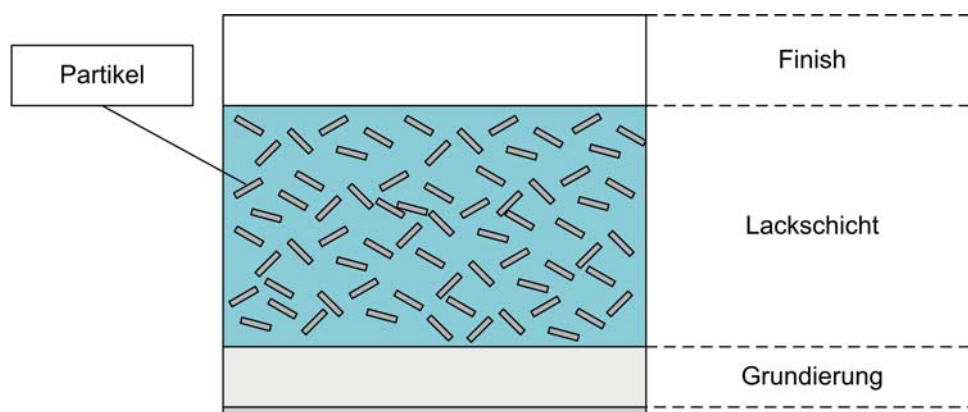


Abbildung 4.14.: Aufbau Fahrzeuglack

metallischen Partikeln enthalten. Insbesondere im Sonnenlicht sind diese Partikel über Meter hinweg sichtbar. Weiterhin verursachen diese eine gerichtete Reflexion des Lichts, welche als Glitzern sichtbar

ist. Bei Perlglanzlacken sind die metallischen Partikel mit einer halbtransparenten Schicht bestehend aus metallischen Oxiden bedeckt. Diese Beschichtung führt zu einer Auslöschung von bestimmten Wellenlängen in Abhängigkeit des Blickwinkels. Daraus resultieren Farbveränderungen in Abhängigkeit von Sicht- und Lichtrichtungen. Viele Lacke besitzen eine zweite Schicht, welche als Finish bezeichnet wird. Diese Schicht führt zu einer spiegelnden und glänzenden Reflexion.

Zur Simulation einer realitätsnahen Erscheinung von Fahrzeuglacken wurden in den vergangenen Jahren eine Reihe von Modellen für die Entwicklung von virtuellen Prototypen und zur Präsentation in Show-Rooms entwickelt.

Durch Ershov et. al. [63] wurde die Simulation der Glitzer-Effekte in metallischen und Perleffektlacken untersucht. Dafür werden nur Lackpartikel berücksichtigt, welche Licht direkt zur Position des Beobachters streuen. Die Beschreibung findet über eine Wahrscheinlichkeitsfunktion für jeden Pixel statt, welche die Ausrichtung des Pixels beschreibt. Eine BRDF zur Beschreibung von Perleffektlacken wurde in [62] entwickelt. Der Ansatz basiert auf der Zerlegung der realen Lackschichten in unendlich kleine Zwischenschichten. Die Dicke der Schichten ist so gewählt, dass eine mehrfache Streuung zwischen den Schichten betrachtet werden kann. Basierend auf den physikalischen Eigenschaften der enthaltenen Materialien werden die Reflexions- und Transmissions-Eigenschaften der Schichten bestimmt. Die BRDF für jede Schicht werden zur einer BRDF für die Beschreibung des Fahrzeuglacks zusammengefasst. In Durikovic und Martens [59] wird die Simulation der Reflexion von Licht auf Oberflächen mit Glitzer- und Tiefeneffekten für Metalllacke beschrieben. Dafür wird die Geometrie der Glitzer-Partikel modelliert. Durch die Streuung des Lichts innerhalb der Lackschichten wurde so das Glitzern und Funkeln in Form von Schlieren erzeugt. Durch Durikovic et. al. [58] und Ershov et. al. [61] werden Verfahren zur Parametrierung einer BRDF für die Darstellung von Metallic- oder Perleffektlacken auf Basis von Messungen beschrieben. Als Ansatz wird ein zweischichtiges Farbmodell verwendet. Es werden zwei Arten von Pigmenten, Flocken und Mikropartikeln in verschiedene Schichten aufgeteilt. Damit sind Einflussgrößen für die Erscheinung wie Glanz, Glitzern und Schattierung direkt mit dem physikalischen Modell verbunden. Durch Günther et. al. [77] wird ein Framework zur Aufnahme und realistischen Rendering von realen Fahrzeuglack vorgestellt. Aufgenommene Messdaten werden an ein Cook-Torrance BRDF-Modell zur Darstellung des Fahrzeuglacks angepasst. Zur Simulation der Glitzer-Effekte wird das von Ershov et. al. [63] entwickelte Modell verwendet. Durch Rump [160] wurde ein Modell zur Simulation von Fahrzeuglack entwickelt, welches in der Lage ist Glanz, räumlich variierender Glitzer-, Tiefen- und Flip-Flop-Effekte darzustellen. Zur Realisierung dieser Effekte wurde das Modell in zwei Teile aufgeteilt. Zur Beschreibung des homogenen Teils wird das Cook-Torrance Modell verwendet. Dieses beschreibt die Reflexion der Grundierung und obersten Lackschicht. Der räumliche veränderliche Teil des Lacks, welcher durch Effektpartikel im Lack ausgedrückt wird, wird durch eine BTF (Bidirectional Texture Function) beschrieben.

Wenngleich diese Modelle zur Beschreibung der Lackeigenschaften eine hohe Genauigkeit besitzen, so sind diese aufgrund ihrer Komplexität, hinsichtlich Parametrierung und Berechnungszeit nicht in einer interaktiven Simulation mit mehreren Fahrzeugen und weiteren Effekten verwendbar. Durch Isidoro [89] wurde ein empirisches Modell zur Beschreibung eines Perleffektlacks mittels programmierbarer Shader vorgeschlagen. Das entwickelte Modell ermöglicht die Darstellung eines Fahrzeuglacks mit zwei verschiedenen Farbkomponenten und Glitzern durch Flocken. Die Spiegelung auf dem Finish wird durch die Verwendung einer Cube-Map [14] simuliert. Als nachteilig erweist sich, dass keine Referenzparametrierungen oder eine Definition der einzelnen Lackparameter vorliegen. Daher müssen diese empirisch ermittelt werden.

Im Folgenden sollen die mathematischen Zusammenhänge des Oberflächenmodells diskutiert werden. Als Approximation einer BTF zur Beschreibung der Ausrichtung der Lackpartikel für den Flip-Flop- und Glitzer-Effekt des Lacks wird in [89] eine Normalen-Textur verwendet. Die Komponenten der Vektoren wurden mit einem Zufallsgenerator erzeugt. Die Ausrichtung der Normalen der Lackbestandteile wird auf Basis der Normalen des Vertexes \mathbf{n} und der berechneten Zufalls-Normalen \mathbf{n}_n ermittelt. Anhand dieser Normalen werden zwei verdrehte Normalen zur Beschreibung des zweifarbigen Lacks \mathbf{n}_{Lack} und der Flocken $\mathbf{n}_{\text{Partikel}}$ bestimmt.

$$\mathbf{n}_{\text{Lack}} = \frac{a \cdot \mathbf{n}_n + b \cdot \mathbf{n}}{|a \cdot \mathbf{n}_n + b \cdot \mathbf{n}|} \quad (4.46)$$

$$\mathbf{n}_{\text{Partikel}} = \frac{c \cdot \mathbf{n}_n + d \cdot \mathbf{n}}{|c \cdot \mathbf{n}_n + d \cdot \mathbf{n}|} \quad (4.47)$$

Die Koeffizienten a , b , c und d beschreiben den Anteil der verrauschten Normalen \mathbf{n}_n an den verdrehten Normalen. Für die Definition der Koeffizienten gilt, $a \ll b$ und $c = d$. Der Betrag der Normalen definiert die Sichtbarkeit der Flocken.

Durch [89] wird eine aus vier Termen bestehende BRDF vorgeschlagen Glg. 4.48. Der Beitrag der einzelnen Lackschichten $\{\mathbf{c}_1 \dots \mathbf{c}_3\}$ zur gesamtheitlichen Erscheinung wird durch den Polarwinkel der Reflexion an der Oberfläche ($\theta_{r,Lack} = \mathbf{n}_{\text{Lack}} \cdot \mathbf{i}$) und den zugehörigen Exponenten gewichtet. Auf ähnliche Weise wird der Beitrag der metallischen Partikel $\mathbf{c}_{\text{Partikel}}$ beschrieben. Der Polarwinkel definiert sich hier durch ($\theta_{r,Partikel} = \mathbf{n}_{\text{Partikel}} \cdot \mathbf{i}$). Damit folgt für die BRDF:

$$f_{r,Lack}(\theta_r) = \mathbf{c}_1 \cos(\theta_{r,Lack}) + \mathbf{c}_2 \cos(\theta_{r,Lack})^2 + \mathbf{c}_3 \cos(\theta_{r,Lack})^4 + \mathbf{c}_{\text{Partikel}} \cos(\theta_{r,Partikel})^{16} \quad (4.48)$$

Der Beitrag des Finishs zur Beschreibung des Glanz-Faktors ergibt sich durch folgenden Zusammenhang:

$$f_{r,Finish}(\theta_r) = 1 - \frac{1}{2} \cdot \cos(\theta_r) \quad (4.49)$$

Für die Shading Gleichung des Fahrzeug-Lacks folgt somit:

$$L_{Lack}(\theta_i, \theta_r) = (E \cdot \overline{\cos}(\theta_i) + L_{amb}) \cdot f_{r,Lack}(\theta_r) + f_{r,Finish}(\theta_r) \quad (4.50)$$

Bei Betrachtung von Glg. 4.48 erweist sich der Umstand, dass die Beiträge der Lackfarben ausschließlich vom Betrachtungswinkel der Oberfläche abhängen als nachteilig. Bei $\theta_r > 85^\circ$ geht der Kosinus gegen Null und damit auch der Beitrag der Farbpigmente. Daher erscheint eine Modifikation der BRDF um einen diffusen Term als sinnvoll, damit folgt:

$$f_{r,LackMod}(\theta_r) = (k) \frac{\mathbf{c}_1}{\pi} + (k-1) f_{r,Lack}(\theta_r) \quad (4.51)$$

Der Parameter k beschreibt das Verhältnis zwischen konstanter Lackfarbe und der Abhängigkeit vom Blickwinkel.

Vergleich

In Abbildung 4.15 sind die Ergebnisse verschiedener Oberflächenmodelle zur Simulation von Fahrzeuglack dargestellt: Blinn-Phong [24], Blinn-Phong mit Addition einer Cube-Map [14], Isidoro et. al. [89] und die verbesserte Version des Isidoro-Shaders. Als Referenzszene zur Betrachtung der Qualität der Simulation wird ein Ausschnitt einer realen Testfahrt verwendet. Die simulierte Szene beinhaltet ein Fahrzeug mit ähnlicher Geometrie, weiterhin wurde das Umfeld der Straße nachempfunden. Um eine Vergleichbarkeit der Ergebnisse zu gewährleisten, wurden die gleichen Lichtbedingungen, d.h. Position der Sonne und Beleuchtungsstärke, festgelegt. Die Parametrierung des Lacks wurde zur Realisierung einer hohen Ähnlichkeit angepasst.

Bei einer visuellen Betrachtung der Bilder der Grauwerte und Gradienten festgestellt werden kann, dass die Auswahl und Parametrierung des Oberflächenmodells massive Auswirkungen auf die Erscheinung des Fahrzeugs hat. Durch unterschiedliche Oberflächenmodelle werden bei ähnlicher Beleuchtung verschiedene Bereiche des Fahrzeugs unterschiedlich stark betont. Dieses macht sich vor allem in starken strukturellen Unterschieden im Heckbereich der Fahrzeuge bemerkbar.

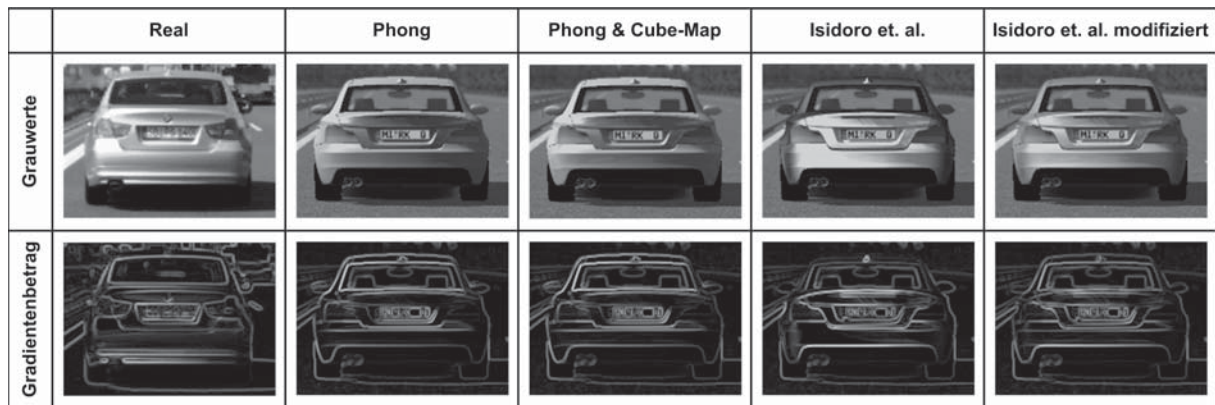


Abbildung 4.15.: Vergleich von Oberflächenmodellen zur Simulation von Fahrzeuglack

Zur Durchführung der Vergleichsbetrachtungen werden die in Abschnitt 4.1.3 definierten Anforderungen an die Simulation und die Ergebnisse der Analyse der Fahrzeugerkennungsverfahren herangezogen. Zur Erkennung von Fahrzeugen werden Textur- und Farbmerkmale ausgewertet, weiterhin geometrische Eigenschaften welche auf der Form basieren, wie Symmetrie und Ecken. Zur Bestimmung der Ähnlichkeit zwischen Simulation und Realität wird zum einen ein Bild mit Grauwerten und zum anderen ein Bild mit Gradientenbeträgen verwendet. Die Berechnung des Gradientenbilds wird mittels des Sobel-Operators [92] durchgeführt. Neben der Bestimmung der globalen Ähnlichkeit in der gesamten Bildregion, wurden Messbereiche definiert, vgl. Abbildung 4.16. Zur Bestimmung der Ähnlichkeit werden der Mittelwert [171] und die Varianz [171] in den Bildern bestimmt. Das reale Bild dient als Referenz mit einer Abweichung von Null. Weiterhin wird eine Kreuzkorrelation [171] zwischen der Simulation und der Realität durchgeführt. Vor der Berechnung der Merkmale wurden die Intensitäten im Bild mittels des Mittelwerts normalisiert. Zur Generierung einer globalen Aussage über die Ähnlichkeit der Oberflächenmodelle bezüglich der realen Referenz wurde für die sechs Messungen in den Bildbereichen ein Score ermittelt. Die mathematische Definition kann im Anhang A.3 vorgefunden werden.

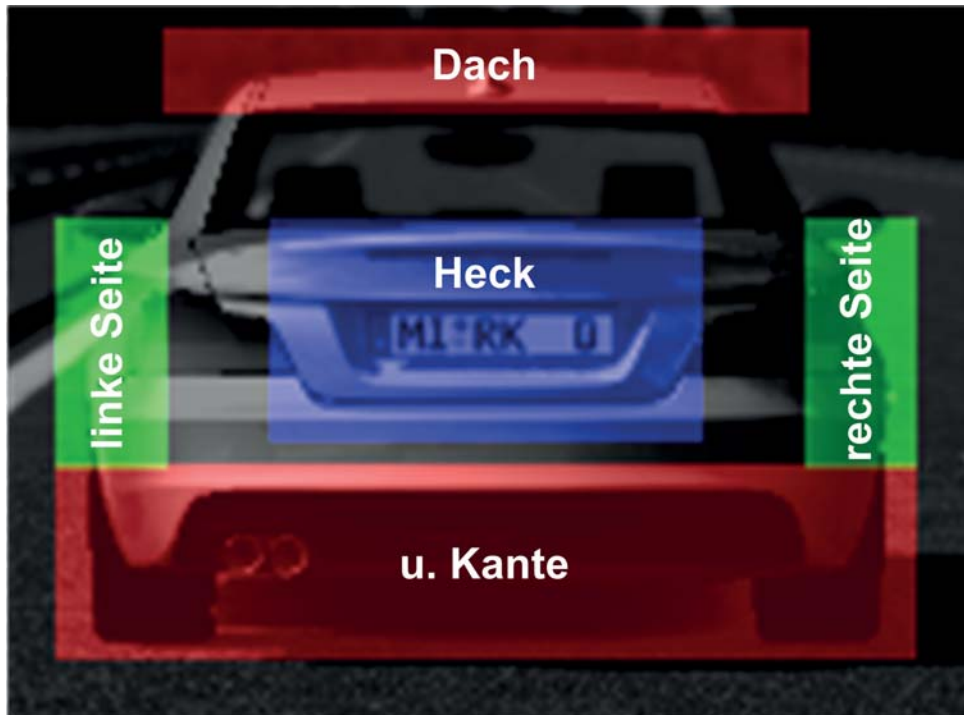


Abbildung 4.16.: Regionen für den Vergleich der Oberflächenmodelle

In Tabelle 4.2 sind die Scores der sechs Messungen aufgetragen. Wie ersehen werden kann schneidet das modifizierte Isidoro et. al. Modell am Besten ab und zeigt die beste Leistungsfähigkeit in jeden Messbereich. Besonders bei Vergleich der Ähnlichkeit bezogen auf das gesamte Bild ist das Modell deutlich überlegen. An zweiter Stelle liegt das Blinn-Phong & Cube-Map - Modell, welches knapp gegen das normale Phong und Isidoro et. al.-Modell gewinnt. Besonders interessant ist das bessere Abschneiden der Phong-Modelle in der Heck- und rechten Seitenregion gegenüber den aufwendigeren Modellen. Das Isidoro et. al. schneidet besonders schlecht aufgrund der starken Abhängigkeit vom Blickwinkel bei dem Vergleich ab. Der Winkel zwischen der Normalen und dem Sichtvektor ist an den Seiten nahe 180°. Durch den konstanten Lack-Term des modifizierten Modells schneidet dieses besser ab. Im Rahmen der durchgeführten

Lack-Modell	Dach	Heck	r. Seite	l. Seite	u. Kante	ges. Bild	Σ
Phong	38.000	15.000	38.000	16.000	19.000	21.000	147.000
Phong & Env-Map	18.000	45.000	37.000	20.000	26.000	23.000	169.000
Isidoro et. al	19.000	22.000	22.000	20.000	49.000	19.000	151.000
Isidoro et. al mod.	45.000	38.000	23.000	64.000	26.000	57.000	253.000

Tabelle 4.2.: Scores der Messbereiche

Betrachtung wird deutlich, dass die Wahl eines geeigneten Oberflächenmodells einen massiven Einfluss auf die Qualität der relevanten Merkmale besitzt. Weiterhin ist die Parametrierung des Modells von großer Wichtigkeit. Anhand der Ergebnisse lässt sich feststellen, dass eine realitätsnahe Simulation des Fahrzeuglacks zu bevorzugen ist. Entsprechend des gewählten Anwendungsfalls der Simulation kann es ebenso sinnvoll sein nur ein Blinn-Phong-Modell zu verwenden und evtl. mit einer einfachen Cube-Map zu ver-

bessern. Wie durch den Vergleich des Isidoro et. al.-Modells mit dem modifizierten Modell gezeigt werden konnte, sollte vor allem der Kontrast zwischen Fahrzeugkante und Umgebung möglichst genau abbildbar sein. Dieser ist vor allem für die Trennung des Objekts vom Hintergrund entscheidend.

4.4.2.3. Simulation der Frontscheinwerfer und Heckleuchten

Während des Tages sind die Lichtquellen anderer Fahrzeuge als Störgrößen zu sehen. Durch den hohen Kontrast zwischen Lichtquelle und Hintergrund entstehen Kanten, welche die Kontur des vorausfahrenden Fahrzeugs beeinflussen. Bei Dunkelheit dienen diese als Merkmale zur Erkennung von Fahrzeugen, vgl. Abschnitt 2.3. Durch Melzer [109] wurde im Rahmen seiner Diplomarbeit die Simulation von Lichtquellen für die Simulation einer Nachtfahrt untersucht. Dazu wurde die bestehende Multi-Pass Grafik-Engine, um eine Deferred-Shading Pipeline erweitert [14]. Bei einem Deferred-Shading-Ansatz wird die geometrische Berechnung von der Berechnung der Ausleuchtung getrennt. In einem ersten Schritt werden die Normalen, die Tiefenwerte, und Parameter der Oberflächenmodelle (Diffuse-, Glanzlicht- und Texturfarbe) in einem Geometry-Buffer gespeichert, vgl. Abbildung 4.17. Durch Anwendung des Geometry-Buffers wird die

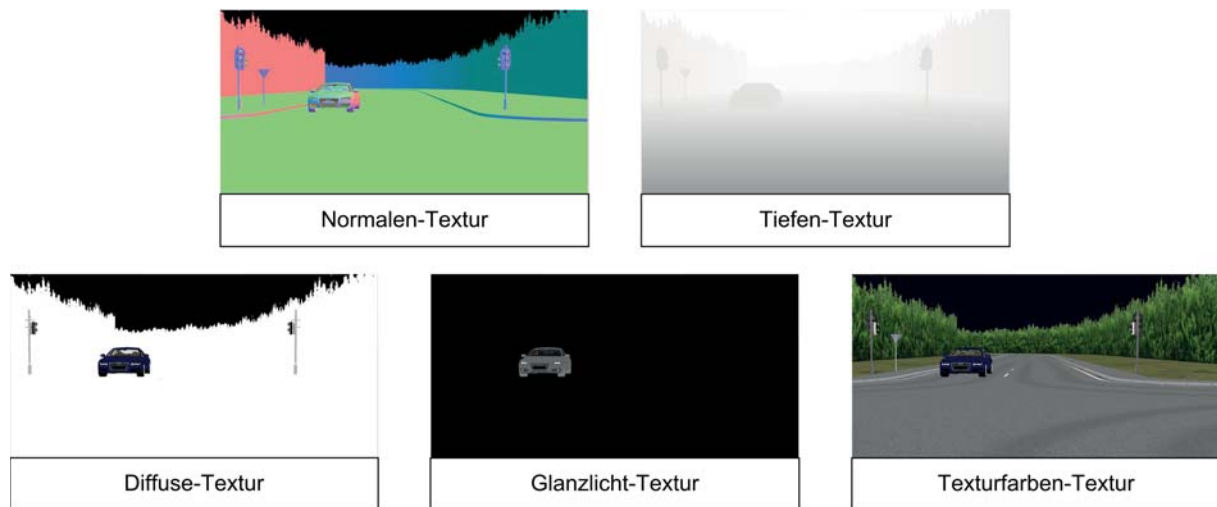


Abbildung 4.17.: Inhalt des Geometry-Buffers [109]

Ausleuchtung der Szene in einem nachfolgenden Schritt für das gesamte Bild bestimmt. Die Berechnung der Ausleuchtung erfolgt für jede Lichtquelle sequentiell. Durch die Eigenschaft, dass sich die Beleuchtungsstärke additiv verhält kann diese während der Berechnung integriert werden. Die Eigenschaften der verschiedenen Lichtquellen werden mittels projektiver Texturierung definiert. Die Berechnung der Lichtkränze findet in einem weiteren Berechnungsschritt statt. Für die Prüfung der Verdeckung durch Objekte werden die Tiefenwerte aus dem Geometry-Buffer verwendet. In Abbildung 4.18 ist das Verfahren dargestellt. Im Folgenden werden die Techniken beschrieben und die Auswirkungen für die Bildverarbeitung betrachtet.

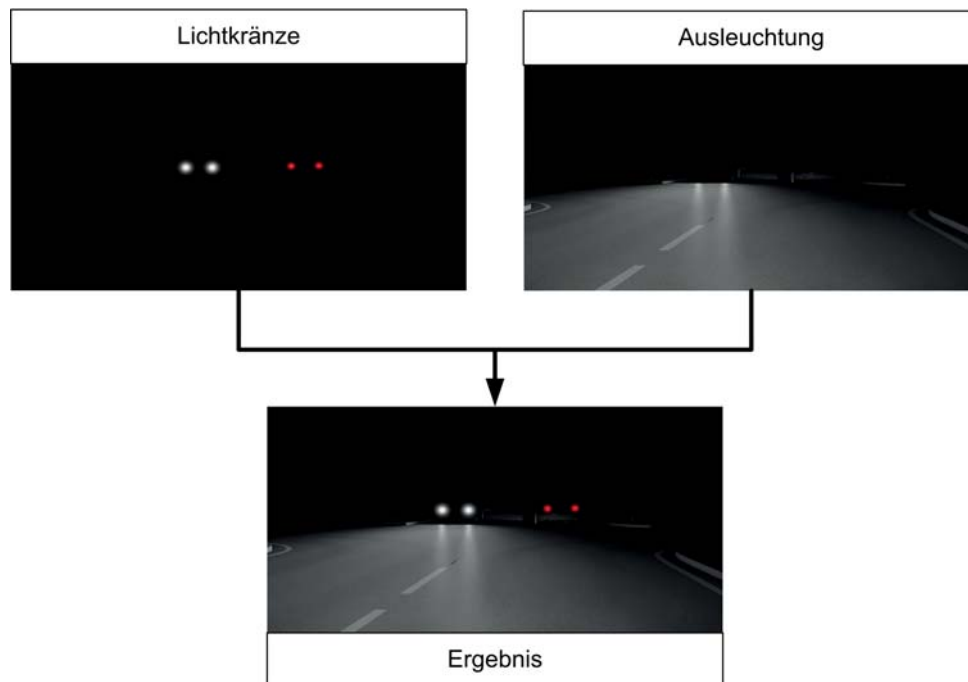


Abbildung 4.18.: Verfahren zur Simulation der Ausleuchtung einer Szene bei Dunkelheit [109]

Scheinwerfer

Die Umsetzung des Ausleuchtungsmodells der Scheinwerfer erfolgte mittels perspektivischer Projektion einer Beleuchtungsstärken oder Lichtstärken-Textur [175]. Dazu wird eine Textur, ähnlich wie bei einem Dia-Projektor auf die Szene geworfen. Die Textur beschreibt die Abstrahlungscharakteristik eines Scheinwerfers. Die Ermittlung dieser erfolgt durch Messungen im Lichtlabor. Der Scheinwerfer oder das Fahrzeug wird dafür in einem definierten Abstand vor eine Messwand positioniert. Anschließend wird die Beleuchtungsstärke oder Lichtstärke der Lichtquelle aufgezeichnet. Im linken Teil Abbildung 4.19 ist beispielsweise die Verteilung der Lichtstärke für das Fernlicht über den Abstrahlungswinkel dargestellt. Durch Verwendung dieser Textur wurde das im rechten Teil der Abbildung dargestellte Ergebnis für das Fernlicht erreicht. Das gleiche Verfahren wird auch für das Abblendlicht verwendet.

Lichtkränze

Die Erscheinung der Lichtquellen in der Kamera resultiert aus den Streuungs- und Absorptionsprozessen in der Atmosphäre, der Frontscheibe und der Kameralinse und zum anderen durch die Eigenschaften des Bildsensors [119]. Zur Simulation des Verhaltens können physikalische und erscheinungsbasierte Techniken eingesetzt werden. Im Rahmen der Diplomarbeit von Melzer wurde ein Verfahren basierend auf der Simulation der Streuung nach [113] und ein weiteres Verfahren basierend auf Texturen untersucht. Mit diesem Post-Processing Verfahren kann die Streuung von Lichtstrahlen über die Kanten von verdeckenden Objekten hinaus simuliert werden. Die Abschwächung der Leuchtkraft eines Pixels wird durch numerische Integration, über die Distanz des Pixels zum Zentrum der Lichtquelle bestimmt.

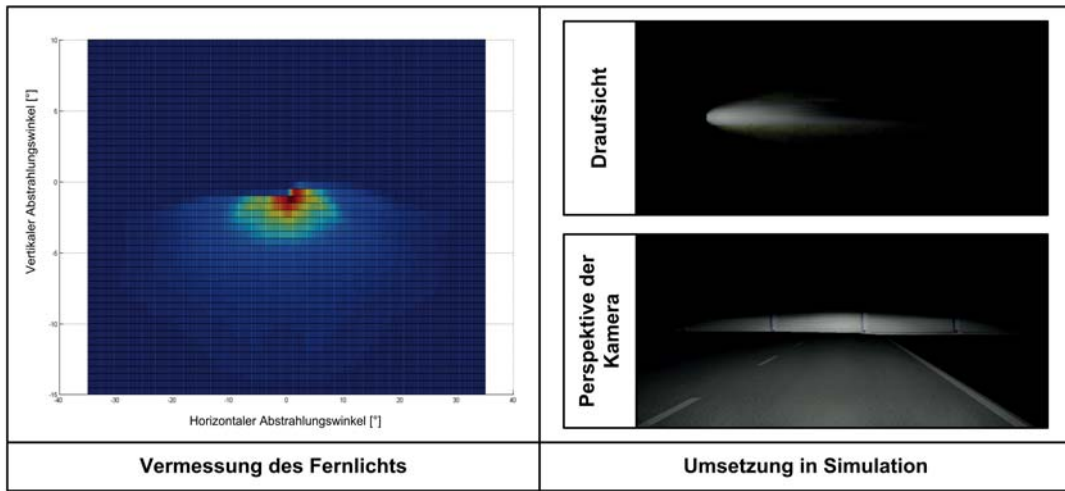


Abbildung 4.19.: Umsetzung von Scheinwerfern in der Simulation, Bildinhalte aus [109]

Eine einfachere und effektivere Simulation der Lichtkränze ist durch die Verwendung von Billboards möglich. Billboards sind texturierte Quadrate. Zur Simulation der Lichtquellen werden immer dem Beobachter zugewandte Billboards verwendet, daher werden diese auch Screen-Aligned-Billboards bezeichnet [14]. Die Billboards werden an der Position der Scheinwerfer jedes Fahrzeug in der Szene platziert und mittels Alpha-Blending [175] der bestehenden Szene überlagert. Diese Technik kann aufgrund ihrer Einfachheit und Effektivität auch in der normalen Multi-Sampling Pipeline für Simulation von Tag-Szenen verwendet werden. Die Billboards werden dafür direkt während der Forward-Rendering Passes berechnet.

Die Ergebnisse der Methoden sind in Abbildung 4.20 dargestellt. Im linken Teil ist die prototypische Umsetzung der Streusimulation und im rechten Teil die Billboard-Technik bei Tag und Nacht im Vergleich zu einer realen Referenz dargestellt. Da das Verfahren der Streusimulation in einer Echtzeit-

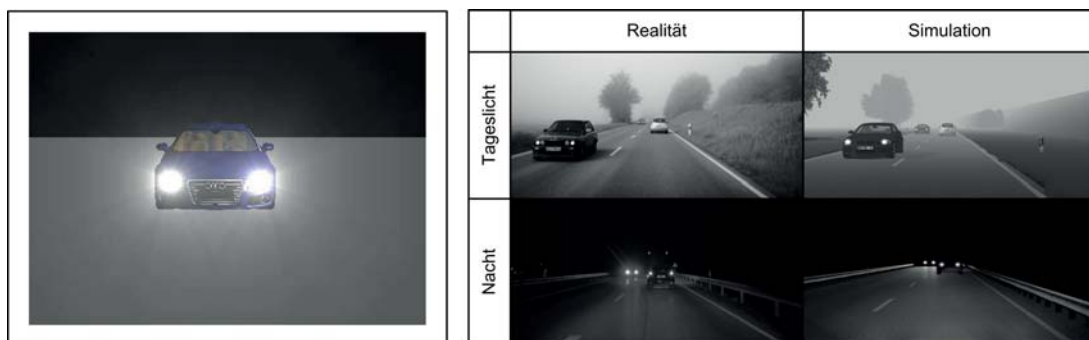


Abbildung 4.20.: Links: Physikalisch basierte Lichtstreuung aus [109], Rechts: Anwendung der Billboard-Technik zur Simulation der Lichtkränze bei Tag und Nacht im Vergleich zu einer realen Szene

Applikation, bedingt durch die numerische Integration nicht leistungsfähig genug ist, wurde der empirische Ansatz weiterverfolgt. Durch Melzer [109] wird weiter angeführt, dass die meisten Algorithmen bei der

Segmentierung der Lichtobjekte, z.B. mittels Schwellwertverfahren, viele Informationen über die Lichtkränze auslöschen. Im segmentierten Bild bleibt nur das Zentrum des Lichtkranzes erhalten. Daher stellt das Billboard-Verfahren eine hinreichend genaue Approximation dar.

Auswertung

Als Nachweis zur Anwendbarkeit wurde durch Melzer die Billboard-Technik in zwei Szenarien mittels einer Segmentierung des Ursprungsbilds, anhand von zwei verschiedenen Schwellwerten th , untersucht. In Abbildung 4.21 sind die Ergebnisse des Vergleichs dargestellt. Wie ersehen werden kann treten bei

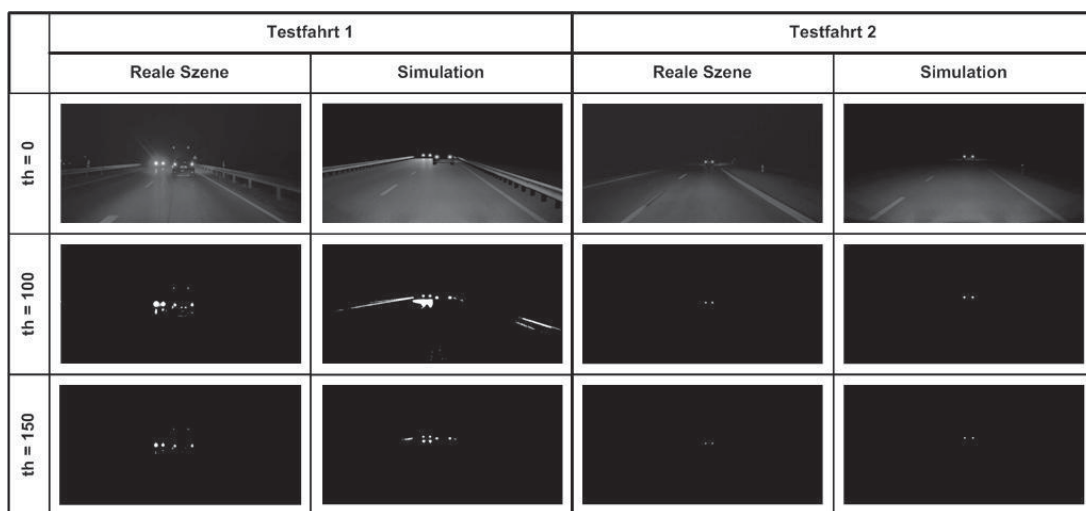


Abbildung 4.21.: Vergleich zweier realer Testfahrten mit der Simulation durch Anwendung einer Schwellwert-Segmentierung, Einzelbilder von [109]

einem Schwellwert von $th = 100$ Grauwerten bei der ersten realen Testfahrt störende Reflexionen durch die Spiegelung der Scheinwerfer auf der Straße auf. In der Simulation sind diese durch die gewählte Parametrierung der Oberflächenmodelle für Straße und Leitpfosten stärker ausgeprägt. Die Form der Scheinwerfer ist in beiden Bildern ähnlich, wenngleich die Lichtkränze des entgegenkommenden Fahrzeugs durch die starke Streuung in der realen Szene größer sind. Bei einem Schwellwert von $th = 150$ Grauwerten sind die Störungen deutlich reduziert.

In der zweiten Szene gibt es nur ein vorausfahrendes Fahrzeug. Daher entstehen keine Reflexionen auf der Straße. Nach einer Segmentierung der Bilder kann festgestellt werden, dass die Verwendung der Billboards in der Szene eine hinreichende Genauigkeit für die Simulation der Heckleuchten darstellt.

Wie durch Melzer gezeigt wurde stellt die Verwendung von Billboards für die Simulation der Scheinwerfer und Heckleuchten eine hinreichend genaue Approximation dar. Für Frontleuchten kann diese mit einer von der Luftfeuchtigkeit abhängigen Skalierung der Lichtkränze verbessert werden.

4.4.2.4. Schlussfolgerung

Im Rahmen dieses Abschnitts wurde die notwendige Modellgenauigkeit für Fahrzeuge betrachtet. Dazu wurde als erstes die geometrische Modellierung von Fahrzeugen beschrieben. Die Modelle mit einer Polygonanzahl von 30000 bis 40000 wiesen eine ausreichende Genauigkeit auf und sind in offenen Datenbanken wie in Google 3D-Warehouse erhältlich.

Anschließend wurde auf die Simulation des Fahrzeuglacks eingegangen, da die Erscheinung des Fahrzeugs ein wichtiges Merkmal für die Fahrzeugerkennung während des Tages ist. Zur Bewertung des Modellaufwands wurde ein Vergleich zwischen verschiedenen Shadern mit einer realen Referenz durchgeführt. Als Merkmale für den Vergleich wurden Textur- und Farbmerkmale, Symmetrie und Ecken ausgewertet. Zur Bestimmung der Ähnlichkeit zwischen Simulation und Realität wurden Bilder mit Grauwerten und mit Gradientenbeträgen verwendet. Das modifizierte Isidoro et. al.-Modell zeigte die beste Leistung. Besonders bei einem Vergleich der Ähnlichkeit bezogen auf das gesamte Bild ist das Modell deutlich überlegen. An zweiter Stelle liegt das Blinn-Phong & Cube-Map - Modell, welches knapp gegen das normale Phong- und Isidoro et. al.-Modell gewinnt. Besonders interessant ist das bessere Abschneiden der Phong-Modelle in den Messbereichen für die Heck- und die rechte Seitenbereich gegenüber den aufwendigeren Modellen. Das Isidoro et. al. schneidet besonders schlecht aufgrund der starken Abhängigkeit von Blickwinkel bei dem Vergleich ab. Es konnte gezeigt werden, dass das Oberflächenmodell einen massiven Einfluss auf die relevanten Merkmale der Bildverarbeitung besitzt. Anhand der Ergebnisse lässt sich feststellen, dass eine realitätsnahe Simulation des Fahrzeuglacks zu bevorzugen ist. Entsprechend des gewählten Anwendungsfalls der Simulation kann es sinnvoll sein nur ein Blinn-Phong-Modell zu verwenden und evtl. mit einer einfachen Cube-Map zu verbessern.

Ein weiterer wichtiger Aspekt zur Detektion von Fahrzeugen sind die Front- und Heckleuchten der Fahrzeuge. Dieses Merkmal wird bei Dunkelheit zur Erkennung verwendet, während des Tages kann das Licht als Störgröße fungieren. Diese Thematik wurde im Rahmen der Diplomarbeit von Melzer [109] ausführlich betrachtet. Die Ausleuchtung der Szene wurde mit einem Deferred-Shading-Ansatz und projektiver Texturierung gelöst. Die Simulation der Lichtquellen entgegenkommender Fahrzeuge wurde auf Basis von Billboards umgesetzt. Die Qualität der Simulation wurde durch einen Abgleich mit zwei realen Referenzszenen validiert. Melzer [109] konnte nachweisen, dass die Verwendung von Billboards eine hinreichend genaue Approximation von Front- und Heckleuchten zur Erkennung von Fahrzeugen darstellt. Weiterhin war es möglich störende Reflexionen, durch die Spiegelung der Scheinwerfer auf der Straße und Leitpfosten in der Simulation darzustellen. Diese traten auch in der realen Referenzszenen auf.

Wie im Rahmen dieses Abschnitts gezeigt wurde ist es möglich mit Mitteln der Computergrafik eine zur Realität ähnliche Erscheinung von Fahrzeugen zu erhalten. Die geometrische Erscheinung ist vorwiegend durch die Anzahl der Polygone definiert, Modelle in der notwendigen Qualität sind frei verfügbar. Als wichtigeres Kriterium erweist sich die Simulation der Fahrzeugoberfläche. Dafür ist das Blinn-Phong genauso wie das Isidoro et. al.-Modell geeignet. Weiterhin wurde die Möglichkeit zur Simulation von Fahrzeugerkennungsalgorithmen in der Dunkelheit durch die Verwendung von einfachen Billboards untersucht.

4.4.3. Straße

Die Erscheinung der Straße in der Stimulation ist ebenso wie die eines Fahrzeugs von der geometrischen Modellierung und der Wahl des Oberflächenmodells abhängig. Während die geometrische Erscheinung durch die Erhöhung der Polygonanzahl verbessert werden kann, sind die Auswirkungen bei der Wahl eines Oberflächenmodells auf die Segmentierung der Spurmarkierungen zu betrachten. Zur Bewertung des Aufwands für die Umsetzung von Wettermodellen wurde eine nasse Straßenoberfläche betrachtet.

4.4.3.1. Geometrische Modellierung

Die geometrische Beschreibung des Straßenverlaufs in der Simulation geschieht auf Basis des OpenDRIVE-Standards [136]. Der Standard erlaubt die Definition des Straßenverlaufs durch eine Verkettung der geometrischen Grundfunktionen: Gerade, Kreisbogen und Klothoide. Eine Klothoide [136] ist eine spiralförmige Übergangsfunktion von einer Geraden in einem Kreisbogen. Eine wichtige Eigenschaft der Klothoide ist, dass sich die Krümmung linear über die Wegstrecke verändert. Weiterhin können für jedes Straßenelement Attribute, z.B. die Anzahl der Fahrspuren, die Breite und die Markierungen definiert werden. Mittels des Formats ist eine Definition von Fahrspuren und Bürgersteige möglich.

Zur Erzeugung der graphischen Datenbasis wird der Vires RoadDesigner verwendet. Mit diesem ist es möglich verschiedene Markierungsarten, Strich- und Lückenverhältnisse zu definieren. Weiterhin können verschiedene Oberflächen durch Texturen definiert werden, z.B. Spurrillen. Während der Erzeugung der Datenbasis werden die geometrischen Grundfunktionen durch eine Tessellierung in lineare Polygon-Modelle zerlegt. Durch diese lokale Linearisierung kann es bei einer zu groß gewählten Schrittweite der Tessellierung zu einer „eckigen“ Kurve führen. Daher ist auf eine hinreichende Auflösung zu achten.

4.4.3.2. Straßenbelag

Die Erscheinung der Straßenoberfläche wird maßgeblich durch eine makroskopische Textur und mesoskopische Reflexionseigenschaften beschrieben. Zur Beschreibung der Straßenoberfläche sind verschiedene Strategien denkbar, welche im Folgenden auf ihre Anwendbarkeit untersucht werden sollen:

- 2D-Textur
- 2D-Textur und Normal-Mapping
- 2D-Textur und Normal-Mapping mit Parallax Mapping

Weiterhin werden die Auswirkungen bei der Anwendung von verschiedenen BRDF-Modellen untersucht:

- Blinn-Phong [24]
- Ward [190]
- Schlick [167]

Straßenoberfläche

Im einfachsten Falle kann die gesamte Straßenoberfläche durch eine homogene Farbe, z.B. grau, beschrieben werden. Zur Darstellung von komplexeren Straßenbelägen, ohne die Erhöhung der geometrischen Komplexität, eignen sich Texturen. Eine Textur stellt die visuelle Erscheinung der Oberfläche im RGB-Farbraum unter diffusen Lichtbedingungen in einem Bereich des Objekts dar [14]. Bei der Verwendung von Texturen auf großen Oberflächen, z.B. einer Straße, weisen diese in der Regel eine periodische Wiederholung der Oberfläche auf. Durch diesen Umstand können Probleme bei der Lösung von Korrespondenz-Problemen auftreten.

Eine weitere Möglichkeit zur Verbesserung der Erscheinung der Oberfläche stellt die Verwendung einer Normalen-Textur (Normal-Map) in Verbindung mit einer 2D-Textur zur Modifikation der Lichtberechnung dar. Die Normalenvektoren des geometrischen Modells werden durch die Eckpunkte des Primitives definiert. Daher gibt es in jedem Punkt auf dem Dreieck die gleiche Normale. Zur Verbesserung der Belichtungsberechnung für jeden Texel wurde durch Blinn [25] die Idee eingeführt relative Verschiebungskoeffizienten für die Normalen in einem 2D-Array zu speichern. Heutzutage ist der bevorzugte Weg direkt die Normalen-Vektoren in einer Textur zu speichern. Diese besitzt die gleiche Auflösung wie die optische Textur. Die 8-Bit jedes Farbwerts werden auf einen Intervall von $[-1, 1]$ aufgeteilt. Die Normalen liegen in dem Tangenten-Koordinatensystem der Textur vor. Für die Verwendung der Normal-Map in einer 3D-Anwendung müssen alle für das Shading notwendigen Vektoren in den Tangenten-Raum, wie z.B. der Lichtvektor, transformiert werden [14]. Anschließend kann die Beleuchtung für jedes Texel individuell berechnet werden.

Durch Akenine und Moeller [14] wird angeführt, dass durch die alleinige Verwendung eines Normal-Mappings bestimmte optische Erscheinungen nicht ausreichend dargestellt werden können. Als Beispiel wird durch die Autoren der Fall genannt, dass bei Betrachtung einer realen Backsteinmauer aus einem bestimmten Winkel der Mörtel zwischen den Steinen nicht mehr sichtbar ist. Durch [95] und [192] wurde zur Lösung dieser Schwierigkeit das Parallax-Mapping vorgeschlagen. Parallaxe bedeutet, dass die Positionen von Objekten sich relativ zueinander verändern wenn sich der Betrachter bewegt. Die Idee des Parallax-Mappings ist es daher eine Abschätzung darüber zu treffen, was unter Einbeziehung einer Parallaxe in diesem Bildpunkt sichtbar sein sollte. Als Grundlage dafür dient die Höhe der Oberfläche in einem benachbarten Bildpunkt. Daher wird zur Realisierung des Parallax-Mappings neben der Normal-Map eine Heightfield-Textur benötigt, welche jedem Texel eine Höhe zuordnet. Bei Berechnung eines Texels der Oberfläche wird an dieser Stelle der Höhenwert abgefragt und für eine Verschiebung der Texturkoordinaten \mathbf{t} verwendet um einen anderen Teil der Textur anstelle des ursprünglichen Werts auszulesen. Die Verschiebung basiert auf der erhaltenen Höhe h und dem Winkel zwischen Oberfläche und Augpunkt. In Abbildung 4.22 ist das Parallax-Mapping dargestellt. Auf der linken Seite wird der Schnittpunkt zwischen der Strecke Sicht-Vektor und Texel und dem Heightfield bestimmt. Auf der rechten Seite ist die Approximation durch einen konstanten Offset, basierend auf dem Heightfield dargestellt. Daraus ergibt sich die neue Texturcoordinate der Parallaxe \mathbf{t}_p . Dieser Zusammenhang kann durch die folgende Gleichung formuliert werden:

$$\mathbf{t}_p = \mathbf{t} + \frac{h \cdot \mathbf{v}_{x,y}}{v_z} \quad (4.52)$$

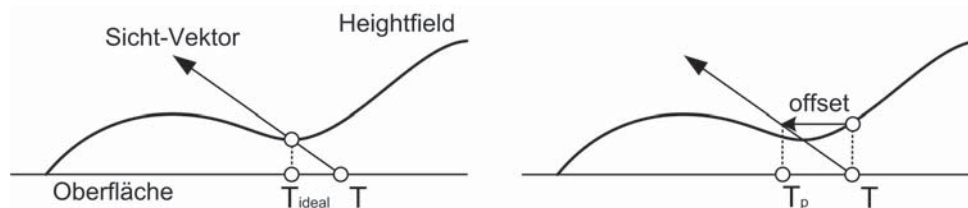


Abbildung 4.22.: Schema Parallax-Mapping

Durch Welsh [192] wurde folgende Gleichung mit einer höheren numerischen Robustheit eingeführt:

$$\mathbf{t}_p = \mathbf{t} + h \cdot \mathbf{v}_{x,y} \quad (4.53)$$

Anwendungsbeispiel

Im Folgenden werden die verschiedenen Ansätze hinsichtlich der Anwendbarkeit zur Visualisierung der Straßenbeläge Asphalt, Kopfsteinpflaster und Schotter betrachtet. Die Vergleichsbetrachtung wird aus der Perspektive der Fahrzeugkamera und aus einem Winkel von 22.5° durchgeführt. Die Kamera befindet sich jeweils $1.30m$ über dem Boden.

Ergebnis In Abbildung 4.23 sind die Straßenbeläge unter einem Betrachtungswinkel von 22.5° dargestellt. Wie ersehen werden kann ergibt sich bei Verwendung einer reinen zweidimensionalen Textur für alle drei Straßenbeläge im Vergleich zum Normal oder Parallax-Mapping ein kontrastärmeres Erscheinungsbild. Die Strukturen der Oberflächen werden nicht hervorgehoben, das Kopfsteinpflaster und der Schotter erscheinen flach. Durch die Verwendung einer Normal-Map, welche für jedes Texel eine individuelle Normale bereitstellt, erhalten die Oberflächen eine strukturierte Erscheinung. Dies ist in der mittleren Spalte dargestellt. Das Kopfsteinpflaster erhält im Vergleich zur 2D-Textur einen stärkeren Kontrast zwischen den Fugen und den Pflastersteinen. Weiterhin erscheinen die Strukturen der Steine deutlicher. Ebenso verhält es sich mit den Strukturen des Asphalts und des Schotters. Die Berücksichtigung einer Parallaxe erlaubt es auf einfache Weise den Steinen eine virtuelle Höhe zu geben. Daraus resultiert eine weitere Verbesserung des Kontrasts zwischen Steinen und Fugen.

Auswertung Die vorgestellten Verfahren zur Visualisierung von Straßenbelägen werden aus der Perspektive einer Spurerkennungsfunktion bzgl. der Auswirkungen auf die Bildqualität betrachtet. Dafür wird zunächst eine Projektion des zweidimensionalen Bilds auf die Bodenebene durchgeführt, nach [175]. Die notwendige in- und extrinsische Kalibrierung der Kamera liegt vor. Nach der Projektion werden mittels des Sobel-Filters [92] Gradienten im Bild als Merkmale berechnet. Der Ablauf ist in Abbildung 4.24 dargestellt. Zur Durchführung des Vergleichs werden die Gradientenbeträge eines homogenen einfarbigen Straßenbelags der gleichen Szene als Referenz verwendet. Als Merkmale für den Vergleich werden auf Basis der Gradientenbeträge der Mittelwert, die Varianz, die mittlere quadratische Abweichung (MSE) und das Spitzen-Signal-Rausch-Verhältnis (PSNR) [32] bestimmt.

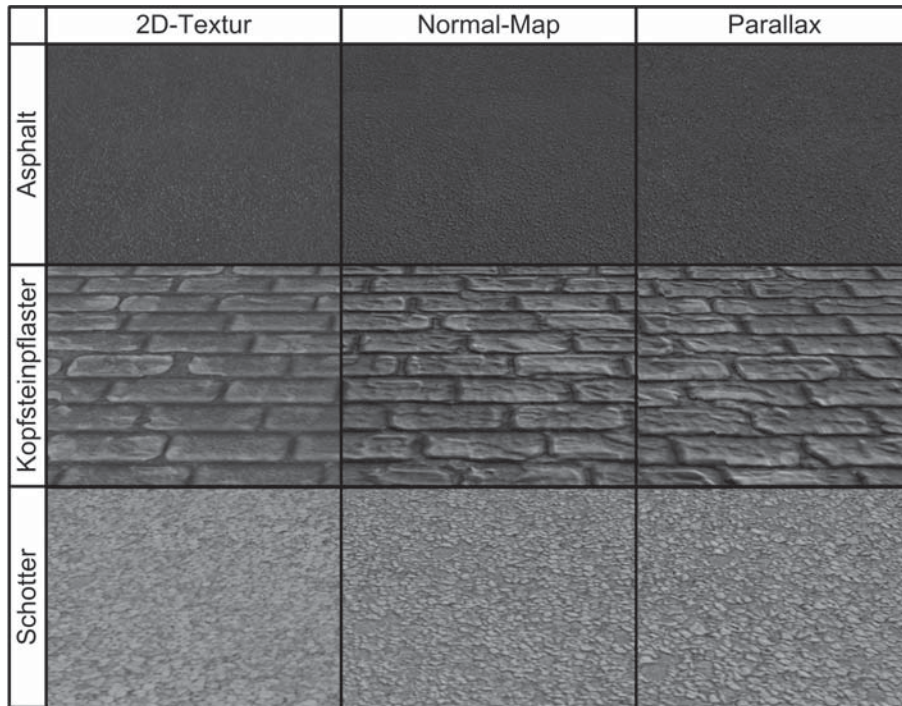


Abbildung 4.23.: Auswirkung der Visualisierung auf verschiedene Straßenbeläge

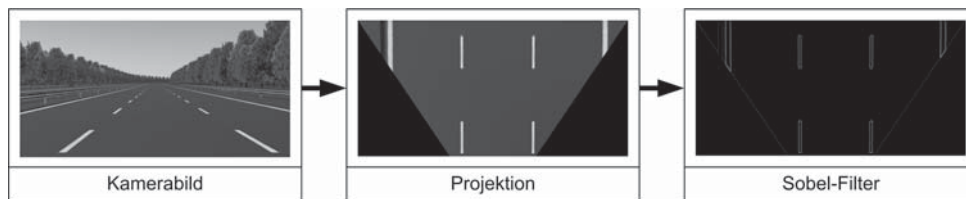


Abbildung 4.24.: Auswertung der Straßenbeläge

In Tabelle 4.3 sind die Ergebnisse der Visualisierungsmethoden für eine Asphaltoberfläche dargestellt. Wie ersehen werden kann nimmt durch Anwendung eines Normal- oder Parallax-Mapping das PSNR auf ein gleiches Niveau ab, bzw. der MSE nimmt zu. Weiterhin ist festzustellen, dass aus dem Winkel der Fahrzeugkamera, welche einen Schnittwinkel mit der Bodenebene von kleiner als 5° besitzt komplexere Visualisierungsmethoden keinen großen Einfluss besitzen. Die Strukturen im Asphalt sind sehr fein, so dass diese kaum ins Gewicht fallen.

Asphalt	\bar{x}	σ	MSE	PSNR
Texture-Mapping	8.4900	0.03730	2057.4000	27.0400
Normal-Mapping	15.1400	0.03870	2296.6600	26.5600
Parallax-Mapping	15.5300	0.03890	2313.1600	26.5300

Tabelle 4.3.: Vergleich Visualisierungsverfahren für Asphalt

Der Einfluss der Visualisierungsmethoden auf einer Kopfsteinoberfläche ist in Tabelle 4.4 dargestellt. Bedingt durch die deutliche Ausprägung der Oberflächenstruktur, bestehend aus Steinen und Fugen, hat das Parallax-Mapping einen großen Einfluss auf das PSNR. Der MSE verdoppelt sich bei Anwendung des Verfahrens im Vergleich zum Normal-Mapping. Beispielhaft wurde der Kontrast der 2D-Textur so modifiziert, dass mit dem Texture-Mapping ähnliche Ergebnisse erreicht wurden. Als Modifikator wurde in der ursprünglichen 2D-Textur die Helligkeit um 50% reduziert und mit zwei multipliziert.

Kopfsteinpflaster	\bar{x}	σ	MSE	PSNR
Texture-Mapping	25.6500	0.04720	3309.7200	24.9700
Normal-Mapping	28.8500	0.05040	3771.6400	24.4200
Parallax-Mapping	44.4800	0.06500	6288.1400	22.1900
Texture-Mapping mod. Kontrast	69.7904	0.1015	14665.5200	18.5100

Tabelle 4.4.: Vergleich Visualisierungsverfahren für Kopfsteinpflaster

Bei Betrachtung der Ergebnisse für eine Schotteroberfläche in Tabelle 4.5 kann festgestellt werden, dass das Visualisierungsverfahren keine große Auswirkungen auf das PSNR, den MSE, den Mittelwert und die Varianz haben.

Schotter	\bar{x}	σ	MSE	PSNR
Texture-Mapping	11.9123	0.0339	1703.3500	27.8600
Normal-Mapping	13.9890	0.0346	1793.1800	27.6400
Parallax-Mapping	14.1409	0.0347	1802.2500	27.6100

Tabelle 4.5.: Vergleich Visualisierungsverfahren für Schotter

Schlussfolgerung

Der Einfluss verschiedener Methoden für Simulation von Straßenbelägen aus dem Blickwinkel einer Fahrzeugkamera wurde bewertet. Es wurde gezeigt, dass das Normal- und Parallax-Mapping keinen signifikanten Einfluss auf die Bildqualität zur Durchführung von Bildverarbeitungsoperationen für die Spurerkennung bei einer Asphalt oder Schotteroberfläche haben. Das Parallax-Mapping zeigte sich am Besten für die Visualisierung eines Kopfsteinpflasters geeignet, wenngleich ähnliche Ergebnisse durch die Modifikation des Kontrasts der Straßen-Textur erreicht wurden. Der Implementierungs- und Berechnungsaufwand der Kontrastmodifikation ist während der Simulationslaufzeit im Vergleich zum Parallax-Mapping deutlich geringer. Basierend auf diesen Erkenntnissen kann durch entsprechende Anpassungen der Textur der Straßenoberfläche eine Verwendung eines Texture-Mappings für den Test von Spurerkennungsfunktionen ausreichen.

BRDF-Modelle

Zur Approximation des Reflexions- und Absorptionsverhaltens ist die Verwendung verschiedener BRDF denkbar. In der Computergrafik wird häufig die Blinn-Phong-BDRF aufgrund ihrer Einfachheit verwendet. Weiterhin wurden mit der Ward- und Schlick-BRDF zwei physikalische Modelle für die Echtzeitsimu-

lation entwickelt. Im Folgenden sollen diese drei BRDF hinsichtlich der Anwendbarkeit für die Simulation des Reflexionsverhaltens von Asphalt untersucht werden.

Ward-BRDF Die BRDF nach Ward [190] ist ein physikalisch empirisches Modell zur Beschreibung von aniso- und isotropischen Oberflächen. Diese anisotropischen Oberflächen besitzen eine richtungsabhängige Reflexionsrichtung, z.B. wie gebürstetes Metall. In der zugehörigen Veröffentlichung wird darauf hingewiesen, dass es eine hinreichende mathematische Genauigkeit besitzt und schnell zu berechnen ist. Weiterhin soll das Modell durch empirische Messwerte applizierbar sein. Die Herleitung wird mit einem Modell für eine isotropische Oberfläche durchgeführt. Das Modell vernachlässigt fresnelsche und geometrische Dämpfungskoeffizienten, diese werden mittels des Normalisierungsfaktors $\frac{1}{4\pi\alpha^2}$ approximiert. Dieser besitzt eine hinreichende Genauigkeit solange die Standardabweichung der Reflexion $\alpha < 0.2$ ist. Für die isotropische BRDF folgt:

$$f_{r,Ward,iso} = \frac{k_d}{\pi} + \frac{k_s}{4\pi\alpha^2\sqrt{(\mathbf{l}\cdot\mathbf{n})(\mathbf{v}\cdot\mathbf{n})}} e^{-\frac{\left(\frac{1-(\mathbf{n}\cdot\mathbf{h})^2}{(\mathbf{n}\cdot\mathbf{h})^2}\right)}{\alpha^2}} \quad (4.54)$$

Wobei k_d der diffuse und k_s der spekulare Reflexionskoeffizient ist. Diese können auch vom Betrachtungswinkel der Oberfläche abhängen, um den Fresnel-Effekt zu modellieren. Dafür gilt die Bedingung, dass $k_d + k_s < 1.0$. Der Vorteil des Modells ist es, dass es im Vergleich zu Blinn-Phong-Modell symmetrisch bezüglich des Einfallswinkels und des Reflexionswinkels und physikalisch normiert ist. Zur Beschreibung von anisotropischen Oberflächen wird das Modell um zwei unkorrelierte Standardverteilungen α_x und α_y erweitert. Diese beschreiben das Reflexionsverhalten in Richtung der Vektoren \mathbf{x} und \mathbf{y} . Der Vektor \mathbf{x} beschreibt, dabei die gebürstete Richtung der Oberfläche und kann ein beliebiger Einheitsvektor sein. Nach Ward kann mit diesen Parametern das Verhalten jeder Oberfläche einfach und physikalisch basierend beschrieben werden. In dem Paper werden durch den Autor Messwerte für Referenzoberflächen bereitgestellt.

$$f_{r,Ward,aniso} = \frac{k_d}{\pi} + \frac{k_s}{4\pi\alpha_x\alpha_y\sqrt{(\mathbf{l}\cdot\mathbf{n})(\mathbf{v}\cdot\mathbf{n})}} e^{-\frac{\left(\frac{(\mathbf{h}\cdot\mathbf{x})}{\alpha_x}\right)^2 + \left(\frac{(\mathbf{h}\cdot\mathbf{y})}{\alpha_y}\right)^2}{(\mathbf{h}\cdot\mathbf{n})^2}} \quad (4.55)$$

Schlick-BRDF Aus den Erfahrungen mit dem Cook-Torrance- und dem Ward-Modell schlägt Schlick [167] ein empirisch physikalisches Modell zur hinreichend genauen Beschreibung von Reflexionen vor. Das Modell verzichtet auf die Gewichtung eines diffusen und spekularen Anteils. Das Modell erlaubt es zwischen homogenen (Metall, Glas, Papier) und heterogenen Materialien (Plastik, Haut, lackierte Oberflächen) zu unterscheiden. Letztere bestehen üblicherweise aus einer durchsichtigen und einer nicht durchsichtigen Schicht einzelner Materialien. Das Modell kann zur Beschreibung von iso- und anisotropischen Oberflächen verwendet werden. Zur Parametrierung des Modells werden drei Parameter verwendet:

$$C_\lambda \in [0, 1] \quad (4.56)$$

Reflexionsfaktor bei Wellenlänge λ , dieser kann auch als die Reflektanz des Materials verstanden werden.

$$r \in [0, 1] \quad (4.57)$$

Rauheitsfaktor ($r = 0$: perfekt spiegelnd Reflexion, $r = 1$ perfekt diffuse Reflexion).

$$p \in [0, 1] \quad (4.58)$$

Isotropiefaktor ($p = 0$: perfekt anisotropisch, $p = 1$: perfekt isotropisch)

Nach [167] kann das Verhalten der BRDF in einen spektralen S_λ und richtungsabhängigen Faktor D aufgeteilt werden. Der Autor schlägt entsprechend der Zusammensetzung des Materials zwei Modelle vor. Für eine Oberfläche welche sich aus einem Material zusammensetzt folgt:

$$f_{r,Schlick} = S_\lambda(u)D(t, v, v', w) \quad (4.59)$$

Diese Gleichung kann entsprechend für eine Oberfläche mit zwei Materialien wie folgt erweitert werden:

$$f_{r,Schlick} = S_\lambda(u)D(t, v, v', w) + [1 - S_\lambda(u)]S'_\lambda(u)D'(t, v, v', w) \quad (4.60)$$

Mit den Parametern $u = \mathbf{h} \cdot \mathbf{v}$, $t = \mathbf{h} \cdot \mathbf{n}$, $w = \mathbf{t}(\mathbf{h} - \mathbf{n})$, $v = \mathbf{i} \cdot \mathbf{n}$ und $v' = \mathbf{l} \cdot \mathbf{n}$.

Spektraler Faktor Im einfachsten Fall kann S_λ als konstante Funktion behandelt werden:

$$S_\lambda(u) = C_\lambda \quad (4.61)$$

Da dieser nach [167] vom Einfallswinkel θ abhängt soll dieser Sachverhalt mit dem Fresnelgesetz beschrieben werden:

$$S_\lambda(u) = C_\lambda + (1 - C_\lambda)(1 - u)^5 \quad (4.62)$$

Dabei stellt die C_λ die Reflektanz der Oberfläche dar.

Gerichteter Faktor Zur Beschreibung eines richtungsabhängigen Faktors D wird die folgende Definition vorgeschlagen. Die Abhängigkeit vom Polar- und Azimutwinkel wird durch die Faktoren $Z(t)$ und $A(w)$ beschrieben.

$$D(t, v, v', w) = \frac{1}{4\pi v v'} Z(t) A(w) \quad (4.63)$$

Für $Z(t)$ und $A(w)$ gilt,

$$Z(t) = \frac{r}{(1 + rt^2 - t^2)^2} \quad (4.64)$$

$$A(w) = \sqrt{\frac{p}{p^2 - p^2w^2 + w^2}} \quad (4.65)$$

Durch die Einführung eines geometrischen Abschwächungsfaktors $G(v)G(v')$ kann die Streuung innerhalb des Materials simuliert werden. Dabei beschreiben $G(v)$ das Verhältnis des reflektierten zum nicht abgeblockten Licht und $G(v')$ das Verhältnis des einfallenden zum nicht abgeblockten Lichts. Zur Beschreibung von reemittierten Licht kann der Term $1-G$ verwendet werden. Damit folgt:

$$D(t, v, v', w) = \frac{G(v)G(v')}{4\pi vv'} Z(t)A(w) + \frac{1 - G(v)G(v')}{4\pi vv'} \quad (4.66)$$

mit

$$G(v) = \frac{v}{r - rv + v} \quad (4.67)$$

$$G(v') = \frac{v'}{r - rv' + v'} \quad (4.68)$$

Zur Beschreibung eines stufenlosen Übergangs zwischen einer diffusen und spiegelnden Erscheinung wird durch Schlick eine Erweiterung des Modells vorgeschlagen. Es existiert keine Parametrierung die einen lambertschen Reflektor darstellen kann. Zur Simulation dieses Übergangs wird eine aus einem lambertschen, anisotropischen und frenselschen Modell zusammengesetzte Gleichung vorgeschlagen. Die Modellterme werden mit den Gewichten a , b und c und der Bedingung $a+b+c = 1$ gewichtet:

$$D(t, v, v', w) = \frac{a}{\pi} + \frac{b}{4\pi vv'} B(t, v, v', w) + \frac{c}{v'dV'} \Delta \quad (4.69)$$

Im Rahmen der Vergleichsbetrachtungen, wird Gleichung 4.70 verwendet, bei welcher der Fresnel-Term vernachlässigt wird.

$$D(t, v, v', w) = \frac{a}{\pi} + \frac{b}{4\pi vv'} B(t, v, v', w) \quad (4.70)$$

Dabei kann $B(t, v, v', w)$ der gerichtete Faktor aus Gleichung 4.63 oder Gleichung 4.66 sein.

Vergleich

Im Folgenden werden die Blinn-Phong-, Ward- und Schlick-BRDF hinsichtlich der Anwendbarkeit zur Simulation der Straßenoberfläche verglichen. Zur Beschreibung der mikroskopischen Oberflächeneigenschaften wird im Folgenden das Parallax-Mapping verwendet. Zur Durchführung des Vergleichs werden die BRDFs in verschiedenen Szenen mit einer realen Referenz verglichen. Der Vergleich wird in einer Szene mit diffusen Licht und in einer Szene mit Schlagschatten parallel zur Fahrbahn durchgeführt. Die anhand dieser Szenen kalibrierten BRDFs werden anschließend in einer ähnlichen Szene mit einer direkten

4.4. GEOMETRIE UND OBERFLÄCHENEIGENSCHAFTEN VON OBJEKTEN

Blendung der Kamera getestet. In Tabelle 4.6 sind die verwendeten Parametrierungen der Oberflächen dargestellt. Die Parameter wurden mittels einer manuellen Optimierung ermittelt. Der Vergleich wird

Schotter	Diffuse	Specular	Parameter
Blinn-Phong	0.7	0.3	$\alpha = 10$
Ward-BRDF,aniso	0.5	0.5	$A_x = 0.45, A_y = 0.45$
Ward-BRDF,iso	0.5	0.5	$A = 0.45$
Schlick-BRDF	0.5	0.5	$C_\lambda = 0.1, r = 0.1, p = 1.0$
Schlick-BRDF (Abschwächung)	0.5	0.5	$C_\lambda = 0.1, r = 0.1, p = 1.0$

Tabelle 4.6.: Parametrierungen der Oberflächen-Modelle

Anhand zweier Szenen durchgeführt. Der Sonnenstand wurde in beiden Szenen mittels des Datums und der Uhrzeit der realen Aufzeichnung parametrisiert. Beide Szenen wurden am 2. April aufgezeichnet. In Szene 1 befindet sich die Sonne auf der rechten Seite der Straße, die Uhrzeit ist 10:38 Uhr, in Szene 2 befindet sich die Sonne auf der linken Seite um 11:18 Uhr. In Szene 1 kommt es in der rechten unteren Ecke des Bilds zu einer stärkeren Reflexion des Sonnenlichts, gleiches tritt in der Szene 2 in der linken unteren Ecke des Bilds auf.

Ergebnisse In Abbildung 4.25 sind die Ergebnisse der verschiedenen Oberflächenmodelle dargestellt.

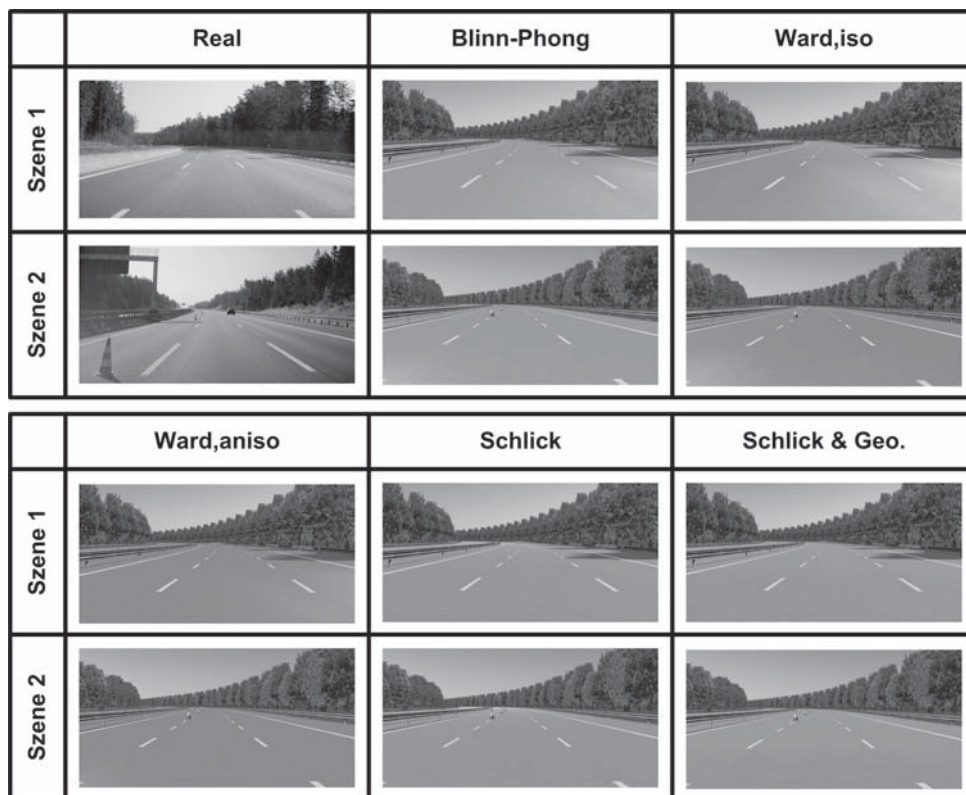


Abbildung 4.25.: Vergleich der Oberflächenmodelle

Auswertung Die Blinn-Phong-BRDF stellt in beiden Szenen die entsprechenden Reflexionen dar. Es ist eine Erhöhung der Helligkeit des Asphalts in den Randbereichen erkennbar, dieser weist sonst eine homogene Erscheinung auf. Die Ward-BRDF mit isotropischen Oberflächeneigenschaften zeigt ein ähnliches Verhalten. In Szene 1 ist der Asphalt im linken Bildbereich deutlich heller. Die Reflexion besitzt eine geringere Homogenität. Eine ähnliche Erscheinung ist in Szene 2 zu erkennen. Im Vergleich wird bei der Ward-BRDF mit einer anisotropischen Oberflächenstruktur das Licht stärker gestreut, was zu einer reduzierten Glanzlichtbildung führt. Im Gegensatz zu der Blinn-Phong und der Ward-BRDF weisen die Schlick-BRDFs, ohne und mit Abschwächungs-Term, keine Glanzlichtbildung in den Randbereichen auf. Diese werden durch den Fresnel-Term gänzlich ausgelöscht. Das Verhalten konnte durch das Hinzufügen des Fresnel-Terms in die Ward-BRDF nachgewiesen werden. Durch den Einfluss des Blickwinkels nimmt bei der Schlick-BRDF die Reflexion mit einem flacheren Blickwinkel zur Oberfläche zu.

In Abbildung 4.26 wurde in Szene 2 die Uhrzeit auf 17:30 Uhr verändert, so dass die Sonne einen größeren Winkel zwischen der Normalen und dem Sichtvektor besitzt. Daraus resultiert eine Verstärkung

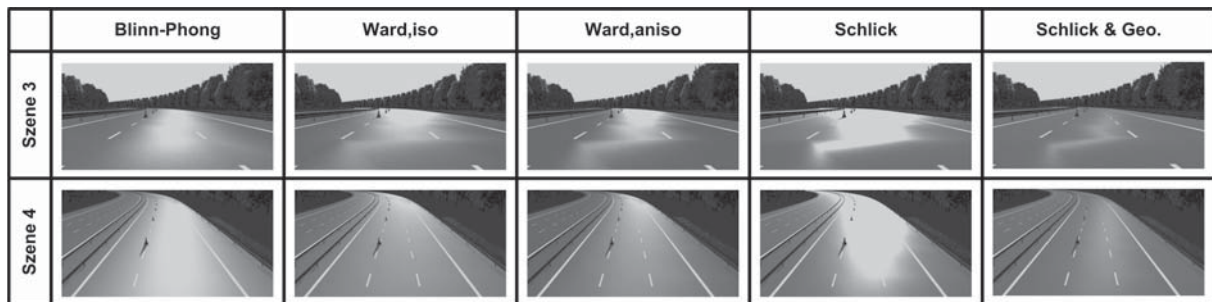


Abbildung 4.26.: Vergleich der Oberflächenmodelle bei tief stehender Sonne

des Glanzlichteffekts. In der ersten Szene kommt es bei einer Kamerahöhe von $h = 1.3m$ und einem Betrachtungswinkel von $\theta_i = 2^\circ$ zu Artefakten. Die Blinn-Phong-BRDF erzeugt eine homogene Reflexion, welche trotz des niedrigen Exponenten des Glanzlichtbestandteils eine starke Fokussierung erhält. Im Vergleich erzeugt die Ward-BRDF eine heterogene Reflexion, welche ein breiteres Zentrum aufweist und stärker fokussiert ist. Als auffällig erweist sich die Bildung von Zacken in den Glanzlichtern. Weiterhin lässt sich feststellen, dass die isotropische BRDF ein breiteres Glanzlicht, als die anisotropische BRDF aufweist. Die Schlick-BRDF erzeugt, ohne den geometrischen Abschwächungs-Term, eine starke Überreflexion der Oberfläche, welche unnatürlich wirkt. Bei Hinzufügen des geometrischen Abschwächungs-Terms verschwindet die Reflexion fast.

In Szene 4 ist Szene 3 aus einer Höhe von $h = 7m$ und einem Betrachtungswinkel von $\theta_v = 12^\circ$ dargestellt. Die Blinn-Phong-BRDF erzeugt in diesem Blickwinkel eine starke Reflexion, welche den Eindruck einer Taschenlampe auf der künstlichen Oberfläche wie Plastik vermittelt. Im Gegensatz dazu erzeugt die Ward-BRDF eine natürlichere Reflexion, welche nur bei einem spitzen Reflexionswinkel zwischen Lichtvektor und Normale der Oberfläche sichtbar ist. Die Schlick-BRDF, ohne geometrischen Abschwächungs-Term, erzeugt ein breites Glanzlicht. Mit geometrischem Abschwächungs-Term ist kaum ein Glanzlicht möglich. Alle BRDFs weisen aus diesem Blickwinkel keine Zacken auf.

Schlussfolgerung

In diesem Abschnitt wurden die Blinn-Phong-, Ward- und Schlick-BRDF hinsichtlich der Anwendbarkeit für die Beschreibung von Straßenoberflächen bewertet. Anhand der durchgeführten Vergleichs kann festgestellt werden, dass sich die Blinn-Phong-BRDF und die Ward-BRDF zur Simulation der Straßenoberfläche eignen. Die Schlick-BRDF kann keine zufriedenstellenden Ergebnisse bereitstellen und besitzt im Vergleich zu den anderen BRDFs einen erhöhten Parametrierungsaufwand. Die Blinn-Phong-BRDF erweist sich bei einem flachen Lichtvektor und den resultierenden großen Schnittwinkel zwischen Oberflächennormale und Sichtvektor als nachteilig. Da die Reflexion eine unnatürliche Erscheinung aufweist. Weiterhin besitzt die BRDF keine physikalische Basis. Als vorteilhaft erweist sich die einfache Berechenbarkeit.

Die Ward-BRDF erzeugt auch bei einer tief stehenden Sonne ein realitätsnahes Glanzlicht. Allerdings müssen weiterhin die Auswirkungen und Gründe für die Zacken untersucht werden. Zusammenfassend erweist sich die Ward-BRDF durch die physikalische Grundlage und einfache Parametrierbarkeit als bessere Wahl für die Beschreibung der Reflexionseigenschaften.

4.4.3.3. Straßenmarkierungen

Im Folgenden soll auf Möglichkeiten zur Darstellung von Straßenmarkierungen eingegangen werden, als Techniken können die für den Straßenbelag verwendeten Verfahren herangezogen werden. In der Realität werden die Markierungen durch das Auftragen von Farbe auf den Asphalt dargestellt, daher weisen diese deutliche Konturen auf. Daher ist es denkbar die Markierung mittels eines einfarbigen Polygons darzustellen. Eine weitere Möglichkeit ist die Farbe der Markierung mittels einer Textur darzustellen. Diese ermöglicht eine unregelmäßigere Kontur zu erzeugen. Zur Darstellung von Oberflächenstrukturen kann eine Normal-Map verwendet werden. Straßenmarkierungen besitzen ein sehr diffuses Reflexionsverhalten um eine sehr gute Wahrnehmbarkeit der Markierung, auch bei wenig Licht, zu gewährleisten. Auf Basis der zur Simulation des Straßenbelags gesammelten Erkenntnisse erscheint es sinnvoll die Ward-BRDF als Grundlage zu verwenden.

Vergleichsbetrachtung

In Abbildung 4.27 ist ein Vergleich zwischen den verschiedenen Arten zur Visualisierung von Fahrspuren dargestellt. In der ersten Zeile ist die Sicht auf die Szene aus der Perspektive der Fahrzeugkamera dargestellt. Die zweite Zeile zeigt einen Ausschnitt aus dem unteren Bildbereich der Fahrzeugkamera. Das Bild der Fahrzeugkamera wurde, wie in Abschnitt 4.4.3.2, auf die Bodenebene projiziert. Zum Vergleich der Auswirkungen wurde auf diesen Bildausschnitt die Berechnung von Gradienten mittels des Sobel-Operators ausgeführt. Bei Verwendung einer einzelnen Farbe für die Straßenmarkierungen weisen diese eine homogene Erscheinung auf. Bei einer Rückprojektion in die Bodenebene und einer anschließenden Sobel-Filterung kann ersehen werden, dass die Markierungen eine starke Kontur aufweisen. Durch diese ist eine Trennung der Markierung von der Straßenoberfläche sehr einfach möglich. Bei Verwendung

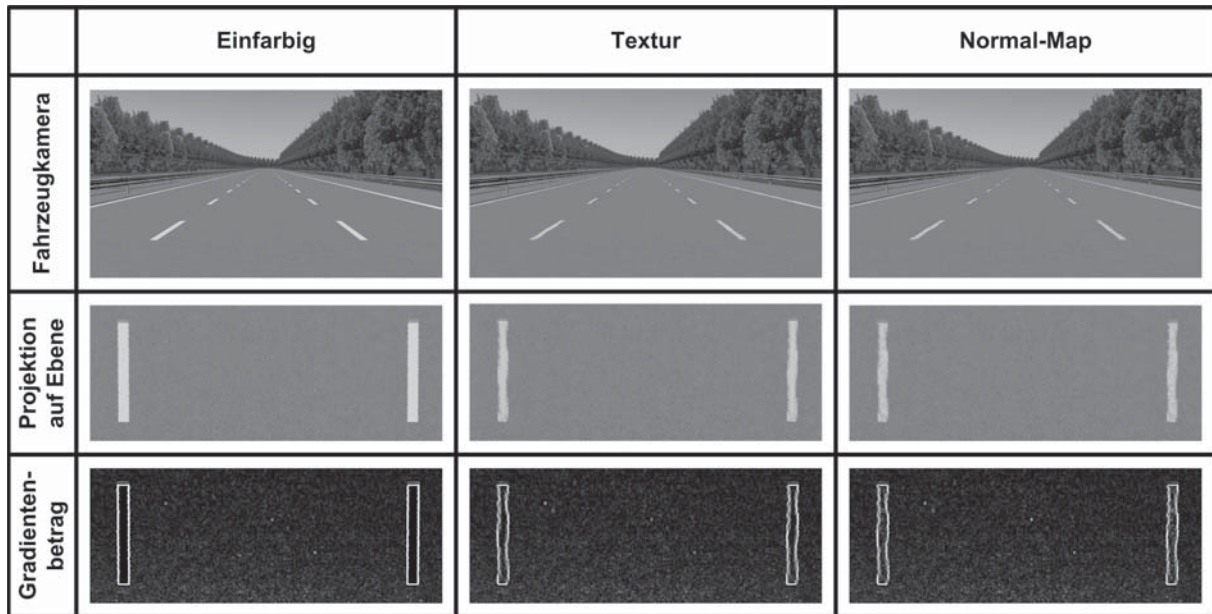


Abbildung 4.27.: Darstellung von Markierungen

einer Textur ist es möglich die Straßenmarkierung mit einer nicht kontinuierlichen und unterschiedlich starken Kontur zu versehen. Bei Betrachtung der Gradientenbeträge kann festgestellt werden, dass die Kontur im Vergleich zur einfarbigen Markierung eine deutlich irreguläre Erscheinung aufweist. Eine Segmentierung und anschließende Verfolgung der Konturen durch eine Wahrnehmungsfunktion muss daher robuster gestaltet werden. Durch Hinzunahme einer Normal-Map werden auf der Straßenmarkierung Unebenheiten sichtbar, welche aus der Struktur der Normal-Map resultieren. Wie in Abschnitt 4.4.3.2 nachgewiesen wurde, reduziert sich das Signal-zu-Rausch-Verhältnis und die Segmentierung wird erschwert.

Schlussfolgerung

Im Rahmen dieses Abschnitts wurde die Darstellung von Straßenmarkierungen in der Simulation betrachtet als BRDF wurde die bereits eingeführte Ward-BRDF verwendet. Zur Beschreibung der Erscheinung wurden eine einfarbige, eine texturierte und einer mit einer Normal-Map verbesserte Straßenmarkierung verglichen. Bei Vergleich der Markierungen zeigte sich, dass für die Durchführung von einfachen Tests zur Stimulation eines Spurerkennungsalgorithmus die einfarbige Straßenmarkierung eine ausreichende Güte besitzt. Durch Reduktion des Kontrasts zwischen Markierung und Straßenoberfläche, können auch Grenzfälle betrachtet werden. Die Applikation einer Textur auf die Straßenmarkierung ermöglicht es auf einfache Weise die geometrische Erscheinung der Markierung zu verändern um die Testtiefe zu erhöhen. Durch die Anwendung einer Normal-Map kann die Erscheinung der Markierung für den menschlichen Betrachter verbessert werden, für das maschinelle Sehen kann damit ein Rauschen simuliert werden. Dieses Rauschen oder andere Störungen in der Oberfläche können auch durch eine Modifikation der 2D-Textur erreicht werden. Daher kann die Verwendung einer Normal-Map als optional angesehen werden und hat keinen Einfluss auf die Qualität der Simulation.

4.4.3.4. Nasse Straßenoberfläche

Eine besondere Herausforderung stellt die Simulation von nassen Oberflächen dar. Wie in Abbildung 4.28 dargestellt treten eine Reihe von Störungen im Bild auf: Reflexion des Himmels auf der Straße, Reifenspuren und aufgewirbeltes Wasser anderer Fahrzeuge.



Abbildung 4.28.: Beispiel für eine Regenszene

Nasser Asphalt

Im Rahmen der Arbeit von [163] wurde eine Visualisierung von nassen Asphalt nach Jensen et. al. [90] betrachtet. Danach wird eine mit Wasser bedeckte Oberfläche durch den Übergang zwischen Luft und Wasser glänzender. Dieses Übergangsverhalten kann durch die Fresnel-Gleichungen [167] für dielektrische Medien beschrieben werden. Wie in Abbildung 4.29 dargestellt, wird das Licht beim Übergang von Luft in Wasser teilweise reflektiert und ein anderer Teil wird in Richtung der Oberfläche der Straße transportiert. Dieser Teil dient zur Beleuchtung der Oberfläche. Nach [90] ist ein Hauptgrund für die Abdunklung der Oberfläche die mögliche mehrfache Reflexion an der Wasser-Luft Grenze. Durch diese wird ein gewisser Anteil des Lichts zurück auf die Oberfläche der Straße geworfen, welches wieder absorbiert und reflektiert wird. Auf Basis dieser mehrfachen Absorption reduziert sich die beim Augpunkt auftreffende Strahlung. In der ursprünglichen Veröffentlichung behandeln die Autoren auch die optischen Eigenschaften von Sand oder Ton. Bei diesen porösen Oberflächen verdrängt das Wasser die Luft in den Poren des Materials. Dadurch verändern sich die Streuungseigenschaften des Materials und es folgt eine stärkere Absorption und verringerte Reflexionswirkung des Materials.

Nach [90] folgt für die ausgehende Strahlungsdichte an einem Oberflächenpunkt L_o :

$$L_o = L_r + L_t \tag{4.71}$$

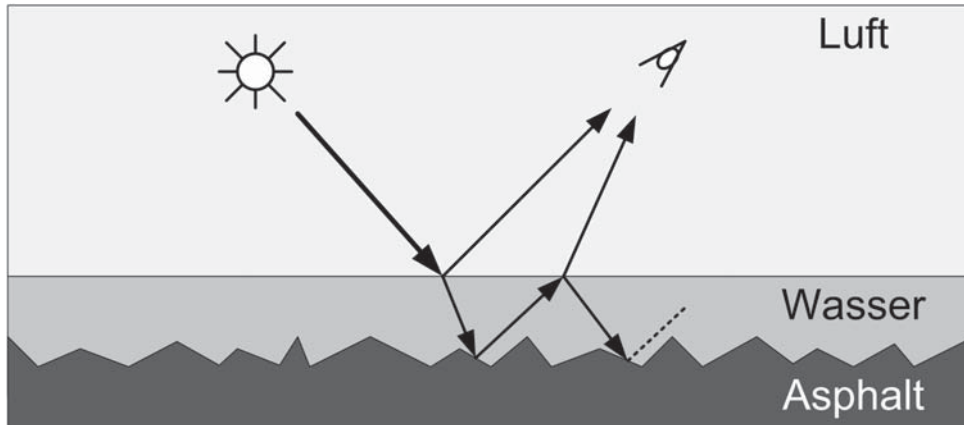


Abbildung 4.29.: Reflexionsverhalten auf nassen Asphalt, nach [90]

Diese setzt sich wie in Abbildung 4.29 dargestellt aus der direkt an der Wasseroberfläche der reflektierten L_r Strahlung und der übertragenden L_t Strahlung zusammen, welche durch die Oberfläche des Objektes reflektiert wurde. Zur Beschreibung des Reflexions- und Transmissionsverhaltens wird im Folgenden die Approximation nach Schlick [167] verwendet. Für den Reflexionsgrad ϱ , welcher den Anteil an reflektierten Licht durch die Oberfläche beschreibt folgt:

$$\varrho(\theta) = \varrho_{orth} + (1 - \varrho_{orth})(1 - \cos(\theta))^5 \quad (4.72)$$

Wobei ϱ_{orth} den Reflexionsgrad für einen senkrechten Einfallswinkel und θ den Einfallswinkel darstellt. Für den Reflexionsgrad bei einem senkrechten Einfallswinkel folgt:

$$\varrho_{orth} = \left(\frac{n_1 - n_2}{n_1 + n_2} \right)^2 \quad (4.73)$$

Der Transmissionsgrad τ ergibt sich damit zu:

$$\tau = 1 - \varrho; \quad (4.74)$$

Durch Verwendung dieser Definition kann Gleichung 4.71 für die ausgehende Strahlungsdichte wie folgt dargestellt werden:

$$L_o = (\varrho(\theta_{Light}) \cdot \mathbf{n} \cdot \mathbf{r} + (1 - \varrho(\theta_{Light})) * f_{r,Road}(\mathbf{n}, \mathbf{l}_{refract})) \cdot L_{In} \cdot \mathbf{n} \cdot \mathbf{l} \quad (4.75)$$

Dabei ist θ_{Light} der Einfallswinkels des Lichts, \mathbf{n} die Normale, \mathbf{r} der Reflexionsvektor, $f_{r,Road}$ die BRDF der Straße, $\mathbf{l}_{refract}$ Richtungsvektor des Lichts nach der Brechung, \mathbf{l} Richtungsvektor der Lichts und L_{in} die Strahlungsdichte der Sonne. Im Vergleich zur der ursprünglichen Gleichung von [90] wurde der Übergang zwischen Wasser und Asphalt vernachlässigt. Weiterhin wird als BRDF für den Asphalt $f_{r,Road}$ eine Ward-BRDF verwendet. Damit ähnelt die BRDF der von Schlick zur Beschreibung von zweischichtigen Oberflächen [167]. Die Brechungsindizes $n_{1,2}$ können aus [13] abgerufen werden. Für die Berechnungsindizes folgt: $n_{Wasser} = 1.33$, $n_{Luft} = 1.0$ und $n_{Asphalt} = n_{Granit} = ca.1.5$. Die von einem Betrachter

wahrgenommene Erscheinung der Oberfläche setzt wie bei der Beleuchtungsrechnung aus der von der Oberfläche ausgehenden Strahlung L_o und einem reflektierten Anteil L_{ref} auf der Oberfläche in Abhängigkeit des Betrachtungswinkels θ_{View} zusammen. Damit folgt:

$$L(\theta_v) = \varrho(\theta_{View}) \cdot L_{ref} + (1 - \varrho(\theta_{View}))L_o \quad (4.76)$$

Zur Beschreibung des reflektierten Anteils des Sonnenlichts L_{Ref} wurde das Verfahren nach Sauer [163] verwendet. Die Idee ist die Reflexion auf der Straße als planare Reflexion zu betrachten. Dieser Ansatz setzt voraus, dass sich die Elevation der Straße in einem Bildausschnitt nur sehr gering verändert. Diese Einschränkung ist im Rahmen der späteren Vergleichsbetrachtung akzeptabel, für eine breitere Verwendung erscheint eine Cube-Map sinnvoller. Zur Durchführung der Reflexionsberechnung nach Sauer wird die gesamte Szene an einer Ebene, welche parallel zur X- und Y-Achse des Weltkoordinatensystems liegt und die Höhe h besitzt, gespiegelt. Die so transformierte Szene wird anschließend in eine Textur gerendert. Während der späteren Berechnung von L wird diese Textur auf die Straße projiziert. Da in der Realität die Straße eine raue Oberfläche aufweist ist die Reflexion nicht spiegelnd, sondern diffus. Daher wird aus der Textur nicht nur ein Wert ausgelesen, sondern eine Mittlung basierend auf der Rauigkeit und den Wasserstand der Straße durchgeführt. Im Paper von [90] wird dies durch eine Veränderung der Streuungseigenschaften durchgeführt. Ein ähnlicher Effekt kann durch eine Skalierung der z-Komponente der Normale \mathbf{n} erreicht werden. Bei Oberflächen mit viel Wasser kann die Normale als senkrecht nach +z betrachtet werden. Daher ist es sinnvoll die z-Komponente mit einem Skalar zu modifizieren und danach einen neuen Einheitsvektor zu bestimmen. Diese modifizierte Normale kann weiterhin mit dem Reflexionsfaktor multipliziert werden, bei einer ebenen Fläche nimmt danach der Reflexionsfaktor den maximalen Wert an. Im Gegensatz zur Umsetzung von Sauer werden diese Schritte während der Berechnung von L ausgeführt.

In Abbildung 4.30 sind Ergebnisse für die Visualisierung der nassen Straßenoberfläche dargestellt. In jeder Szene wurde der Wasserstand um 20% erhöht. Wie ersehen werden kann ist durch die Einbeziehung der fresnelschen Reflexion der nasse Asphalt dunkler als der trockene Asphalt. Durch eine Anpassung des Wasserstands wird die Reflexion auf der Fahrbahnoberfläche verstärkt und die Streuung reduziert. Damit kann durch diesen Parameter die Stärke und die Schärfe der Reflexion gesteuert werden. Weiterhin hat dieser einen direkten Bezug zur Straßenoberfläche, glatte Oberflächen reflektieren stärker als raue bei gleichem Wasserstand. Durch die Anwendung der Fresnel-Gleichung ist erkennbar, dass die Reflexion des Himmels bei einem flacheren Betrachtungswinkel zunimmt, d.h. je weiter entfernt die Straße ist desto stärker ist die Reflexion vom Himmel.

Wasserpartikel

Durch die Reifen von vorausfahrenden Fahrzeugen werden Wasserpartikel aufgewirbelt und sind als Wolke sichtbar. Dieser Wasserdampf streut und absorbiert das Licht, als Ergebnis werden Konturen verwischt und der Kontrast reduziert.

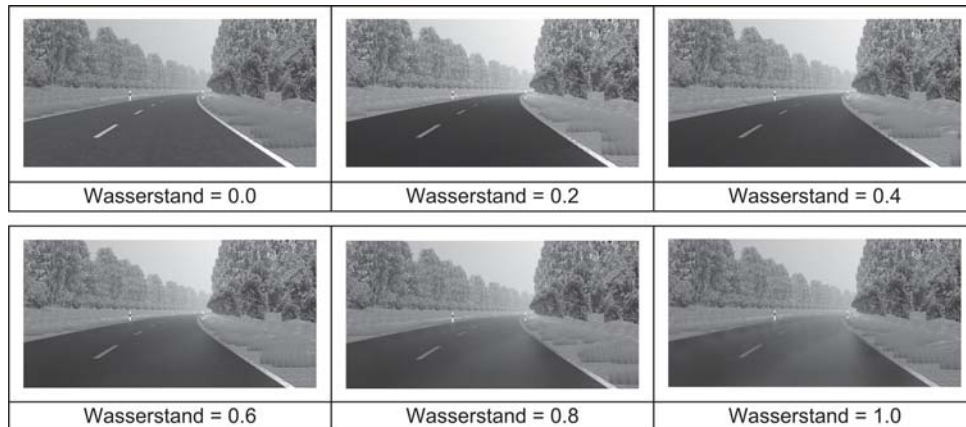


Abbildung 4.30.: Simulation einer nassen Straßenoberfläche

Modellbildung Die Stärke und Dichte des Nebels hängt hauptsächlich von der Geschwindigkeit des Fahrzeugs, dem Reifentyp und dem Wasserstand ab. Weiterhin kann der Wind eine Rolle spielen, wird in diesem Modell aber vernachlässigt. Der Emissionswinkel ω der Partikel hängt zu einem von dem Lenkwinkel der Räder und dem Profil der Reifen ab. In Abbildung 4.31 sind die Zusammenhänge visualisiert.

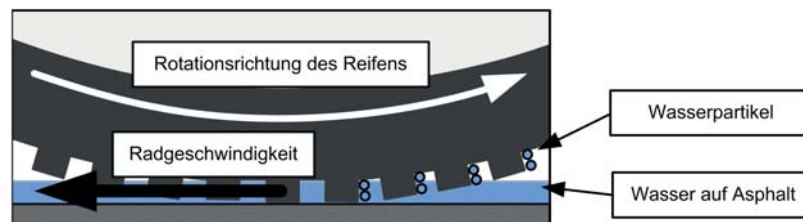


Abbildung 4.31.: Zusammenhänge der Emission von Wasserpartikeln

Zur Erstellung des Modells wird die Annahme getroffen, dass eine Wasserhöhe von 1mm einem Partikel entspricht. Auf Basis dieser Annahme kann für einen Schlitz des Reifenprofils die Anzahl der beinhalteten Partikel berechnet werden. Bei einer Reifenbreite $b = 255\text{mm}$ und einer Profillänge $l_{\text{Profil}} = 5\text{mm}$ und der Annahme, dass ein Stollen 10 Mal so groß wie die Lücke ist folgt eine Länge von $l_{\text{Luecke}} = 45\text{mm}$. Damit ergibt sich eine Länge von $l = 50\text{mm}$ für einen Stollen und eine Lücke. Für einen Reifen in der Bauweise 255/32 R 19 folgt ein statischer Reifenradius von $r_{\text{stat}} = 0.6456\text{m}$.

$$r_{\text{stat}} = 0.255 \cdot 0.32 \cdot 2 + 19 \cdot 0.0254 = 0.6456\text{m} \quad (4.77)$$

Für den Umfang des Reifens folgt damit:

$$u = 2 \cdot \pi \cdot r_{\text{stat}} = 2.028\text{m} \quad (4.78)$$

Daraus ergibt sich eine gesamte Freifläche von $A = \frac{l_{\text{Profil}}}{l} \cdot u \cdot b = 1.0488m^2$, daher werden bei einer Umdrehung des Reifens $N_{0,Partikel} = 1048$ Partikel bei einem Wasserstand von 1 mm emittiert. Bei einer Geschwindigkeit $v = 50m/s$ führt das Rad,

$$n = \frac{v}{u} = \frac{50}{2.028} = 24,64 \frac{1}{s} \quad (4.79)$$

, Umdrehungen n aus. Daraus ergibt sich die Anzahl der zu emittierenden Partikel in Abhängigkeit der Geschwindigkeit v zu:

$$N(v)_{Partikel} = n(v) \cdot N_{0,Partikel} = 24.64 \cdot 1048 = 25822 \quad (4.80)$$

Die Emission der Partikel wird im Folgenden als schiefer Wurf in einer gashaltigen Atmosphäre betrachtet. Die Herleitung der physikalischen Sachverhalte kann im Anhang A.6 vorgefunden werden. Zur Simulation der Emission von Wasserpartikeln wird die in OpenSceneGraph implementierte Partikelsimulation mit einem modifizierten Fragment-Shader verwendet. Der Fragment-Shader graduiert den Alpha-Wert der Partikel in Bodennähe um eine unnatürliche Kante der Partikel zu vermeiden. Auf Basis der vorher hergeleiteten Ausführungen wurde die Parametrierung für die Partikelsimulation bestimmt. Als Werte für die Fläche eines Partikels wurden $A = 0,00314m^2$ angenommen, was einem Radius des Partikel von $r = 0.0005m$ entspricht. Als Gewicht des Partikel wurde eine Masse von $m = 0.0001kg$ angenommen.

In der vorherigen Betrachtung wurde die Annahme getroffen, dass pro Umdrehung eines Reifens ca. 1 000 Partikel emittiert werden müssen. Bei einer Geschwindigkeit von $v = 50m/s$ sind dies ca. $N_{Partikel} = 25000$ pro Sekunde pro Rad, pro Fahrzeug ergeben sich somit 100 000 Partikel. Als Nebenbedingung gilt, dass ein Partikel nicht gelöscht wird solange dieser über der Straße liegt. Da weiterhin mehrere Fahrzeuge in einer Szene aktiv sind ergibt sich eine nicht in Echtzeit zu lösende Komplexität. Daher wird die Anzahl der Partikel auf 100 pro Sekunde reduziert und die visuelle Größe auf $A = 0.25m^2$ erhöht. Die Wasserpartikel werden durch texturierte Billboards dargestellt. Die Auflösungserscheinungen der Wolke werden durch einen über die Entfernung vom Fahrzeug variierenden Alpha-Wert realisiert.

In Abbildung 4.32 ist die Qualität der Partikelsimulation im Vergleich zu einer realen Referenzszene dargestellt. In den Szenen wurde der Alpha-Wert der Partikel von 0.025 bis auf 0.15 erhöht. In der ersten Zeile der Abbildung sind die Grauwerte dargestellt in der zweiten Zeile die Beträge der Gradienten. In den Abbildungen kann ersehen werden, dass durch die Anpassung des Alpha-Werts die Verdeckungswirkung der Partikel variiert werden kann. Bei einem Alpha-Wert von 0.025 sind die Konturen des Fahrzeugs noch sehr gut erkennbar. Bei einem Alpha-Wert von 0.05 sind diese weniger stark wahrnehmbar. Die sichtbare Kante unter dem Fahrzeug entsteht durch den Schatten des Fahrzeugs. Dieses kann auch im Gradienten-Bild wahrgenommen werden. Bei einem Alpha-Wert von 0.1 nimmt die Verdeckung durch die Partikel eine unnatürlich starke Ausprägung an.

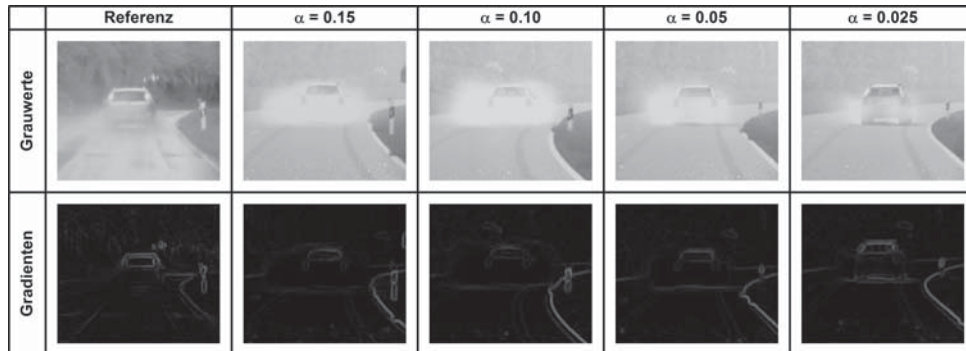


Abbildung 4.32.: Vergleich des Alpha-Werts

Reifenspuren

Auf der Straße entstehen durch verdrängtes Wasser sichtbare Reifenspuren, siehe Abbildung 4.28. Diese können fälschlicherweise als Fahrspurmarkierungen durch ein Spurerkennungsverfahren erkannt werden.

Zur Erzeugung der Reifenspuren in der Simulation wird ein Polygon für jedes Rad erzeugt, welches insgesamt 1000 Punkte besitzt. Das Polygon beschreibt die Wegstrecke jedes Rad über eine Distanz von 1500 m. Dieses wird bei jedem Berechnungsschritt der Simulation aktualisiert und kann als Ring-Puffer betrachtet werden. Die älteste Position wird gelöscht und an dem Beginn der Liste ein neues Punktepaar an der Position des Reifens eingefügt, wenn die Distanz größer als 1,5m ist. Der seitliche Abstand der Punkte entspricht der Reifenbreite. Zur Darstellung einer heterogenen Wasserverdrängung ist die Geometrie mit einer entsprechenden Textur versehen.

Zur Erzeugung der Reifenspuren wird die Geometrie in eine Textur gerendert, ein ähnliches Verfahren wird von [163] für Reifenspuren im Schnee verwendet. Die Parametrierung der Kamera entspricht der Fahrzeugkamera. Während der Berechnung der Reflexionseigenschaften des nassen Asphalts wird die erzeugte Textur auf die Straßenoberfläche projiziert. Mittels der in der Textur gespeicherten Transmissionsgrade ϱ_{mod} , wird der Reflexionsgrad der Straßenoberfläche modifiziert:

$$\varrho_{mod} = \varrho \cdot (1 - \tau_{Reifenspuren}) \quad (4.81)$$

In Abbildung 4.33 ist die Realisierung des Effekts dargestellt. Im linken Bereich sind die erzeugten Fahrspuren dargestellt. Auf der rechten Seite sind die Auswirkungen der Anpassung des Reflexionsgrads erkennbar. Wie ersehen werden kann verringert sich die Reflexion in den Fahrspuren.

Anwendungsbeispiel

Im Folgenden werden die Fähigkeiten der Verfahren zur Reproduktion einer nassen Straßenoberfläche untersucht. Dazu wurde eine reale Szene in der Simulation nachmodelliert und eine Anpassung der Modellparameter durchgeführt.

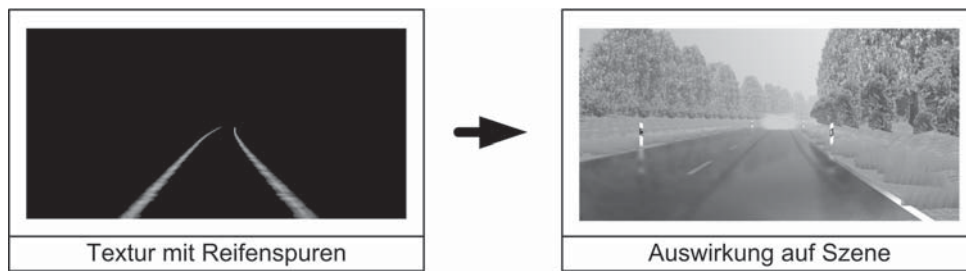


Abbildung 4.33.: Erzeugung von Reifenspuren in der Simulation

Ergebnis In Abbildung 4.34 ist der Vergleich zwischen Realität und Simulation dargestellt. Im oberen

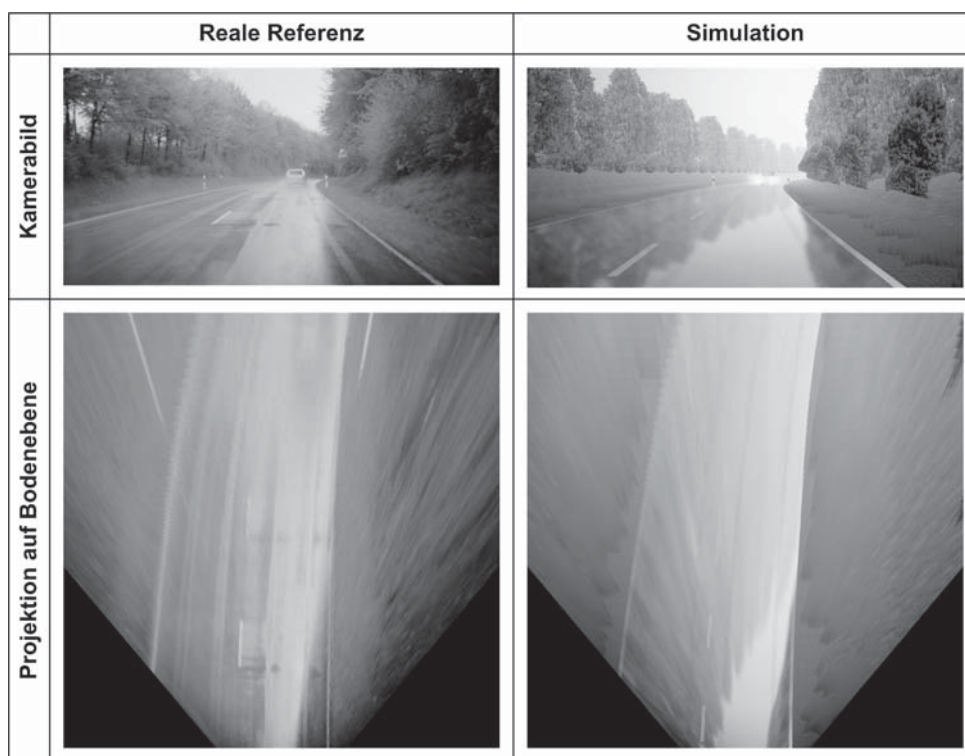


Abbildung 4.34.: Vergleich der optischen Qualität der Simulation einer nassen Straße

Teil der Abbildung ist die Szene aus der Perspektive der Kamera und im unteren Teil die Projektion der zwei-dimensionalen Kamerabilds auf die Bodenebene dargestellt. Die Projektion wurde mittels des gleichen Verfahrens wie in Abschnitt 4.4.3.2 durchgeführt.

Auswertung Bei Vergleich der Reflexion auf der Straße kann festgestellt werden, dass diese einen anderen Verlauf als in der realen Szene hat. Es ist erkennbar, dass mit zunehmender Entfernung die Sichtbarkeit der Fahrbahnmarkierungen abnimmt und die Stärke der Reflexion zunimmt. Weiterhin sind die Spuren anderer Fahrzeuge auf der Straße sichtbar, die Eigenschaften entsprechen in der Simulation qualitativ der Realität. Das vom vorausfahrenden Fahrzeug aufgewirbelte Wasser hat eine ähnliche Erscheinung wie

in der Realität, durch die Belichtung der Kamera wirkt dieses eher weißlich. Anhand des Vergleichs der Bildinhalte kann festgestellt werden, dass mit den Mitteln der Computergrafik ähnliche Ergebnisse wie bei einer realen Testfahrt erzielt werden können. Ein Handlungsfeld ist die Parametrierung der Simulationsmodelle. In Abschnitt 6 wird das Verhalten einer Spurerkennung und einer Fahrzeugdetektion in einer virtuellen Regenszene weiter untersucht.

4.4.4. Verkehrsschilder

Die Erkennung und Interpretation von Verkehrsschildern ist im Vergleich zur Spur- und Fahrzeugerkennung eine größere Herausforderung. So ist es notwendig den Inhalt des Verkehrsschildes richtig zu erkennen, um z.B. ein 100 km/h-Gebot von einem 50 km/h-Gebot zu unterscheiden.

4.4.4.1. Modellierung

In Simulationsumgebungen werden Verkehrsschilder mittels texturierter Polygone dargestellt. Die Texturen sind photographische oder künstlerische Visualisierungen des Verkehrszeichens. Diese stellen das Verkehrszeichen nur in einem bestimmten Abstand scharf dar. Bei einer Vergrößerung der Textur kommt es zu einer Verpixelung des Bilds. Eine Möglichkeit zur Lösung des Problems ist die Interpolation zwischen zwei benachbarten Texeln der Textur. Bei der Interpolation entstehen unscharfe Übergänge und die Kanten werden unscharf oder verschwimmen. Bei der Verkleinerung der Textur kommt es durch die Unterabtastung zu einer irregulären Auswertung und Darstellung des Bilds, z.B. kann bei einem Verkehrsschild die Schrift verschwinden.

Zur Verminderung dieser Effekte werden häufig Mip-Maps verwendet. Dies sind Texturen des Bilds in verschiedenen Auflösungen, welche in Abhängigkeit der Distanz zum Objekt umgeschaltet werden. Damit ist es theoretisch möglich ein Texel einem Pixel im finalen Bild zuzuordnen. Befindet sich der Betrachter zwischen zwei Mip-Map Stufen wird zwischen diesen interpoliert, d.h. es findet eine zweimalige Filterung des Bilds statt. Erstens bei der Reduktion der ursprünglichen Texturgröße zur Erzeugung der Mip-Map Stufe und zweitens bei der Interpolation zwischen verschiedenen Mip-Map Stufen. In der Arbeit von Breithecker [33] konnte dieser nachweisen, dass sich die Anwendung von Mip-Maps positiv auf die Wahrnehmung von Verkehrszeichen bei dem Menschen auswirkt. Hinsichtlich der Auswirkungen auf das maschinelle Sehen liegen keine Erkenntnisse vor. Daher wird dieser Sachverhalt im Folgenden betrachtet.

4.4.4.2. Anwendungsbeispiel

Zur Untersuchung der Auswirkungen auf das maschinelle Sehen, werden im Folgenden deutsche Verkehrsschilder in verschiedenen Distanzen zum Betrachter platziert und mit einer realen Referenz verglichen. Die betrachteten Verkehrsschilder sind 30 km/h, 50 km/h, 80 km/h, 100 km/h, das Überholverbot und das Stoppschild. Als BRDF für das Verkehrszeichen kommt die Ward-BRDF mit der gleichen Parametrierung, wie für die Straße, zum Einsatz. Verkehrsschilder haben in der Regel eine Reflexionsbeschichtung, so dass die Reflexion sehr homogen ist. In Abbildung 4.35 sind die Ergebnisse des Vergleichs dargestellt. Die Schilder wurden aus einer Distanz von 5 m, 10 m, 15 m, 20 m, 25 m und 30 m aufgezeichnet. Zur

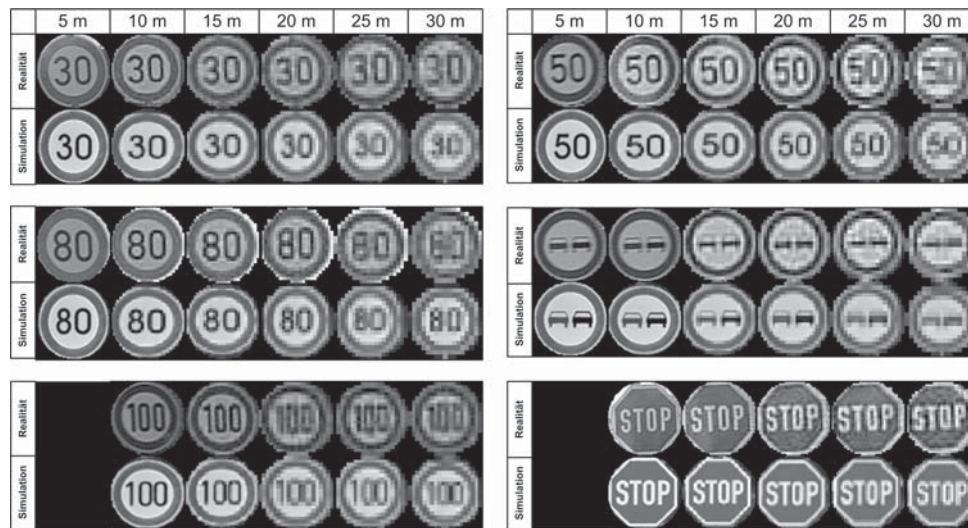


Abbildung 4.35.: Gegenüberstellung realer und simulierter Verkehrszeichen in verschiedenen Entfernungen zur Kamera

Auswertung der visuellen Qualität werden die Intensitäten und Gradienten mittels einer Kreuzkorrelation R [171] zwischen dem simulierten und dem realen Verkehrsschild analysiert. Die Bilder der Gradienten enthalten den normierten Betrag der mit dem Sobel-Filter berechneten Ableitungen der Kanten des Grauwerte. Für das 100 km/h-Gebot und das Stoppschild lagen bei einer Distanz von 5 m keine Messwerte aus der Realität vor.

4.4.4.3. Ergebnisse

Die Ergebnisse des Vergleichs sind in Abbildung 4.36 und in Abbildung 4.37 dargestellt. Dafür wurden die Resultate der Kreuzkorrelation für die verschiedenen Verkehrsschilder über die Distanz aufgetragen. Wie bei Betrachtung der Ergebnisse der Kreuzkorrelation für die Grauwerte festgestellt werden kann, ist die Korrelation bei einem Abstand von 5 m für alle Schilder am Geringsten. Bei einer Vergrößerung der Distanz erhöht sich die Korrelation der realen und simulierten Schilder mit Geschwindigkeitsgeboten. Durch die Reduktion der Schildgröße über die Distanz, welche als Tiefpassfilterung verstanden werden kann, haben kleine Abweichungen im Aussehen der Schilder eine geringere Auswirkung auf das Ergebnis. Weiterhin muss angemerkt werden, dass die Schrift bei den simulierten Geschwindigkeitsgeboten kleiner als bei der realen Referenz ist. Die Schildergröße des Überholverbots und Stoppschilds entspricht der realen Referenz. Bei diesen Schildern nimmt die Ähnlichkeit über die Distanz ab. Bei einer Distanz von 30 m erhöht sich die Ähnlichkeit der Schilder, bzw. nimmt den maximal Wert an. In Abbildung 4.37 ist das Ergebnis der Kreuzkorrelation der Gradienten-Bilder dargestellt. Bei Auswertung der Gradienten-Bilder wird die Auswirkung des Mip-Mappings auf die Strukturen des Verkehrsschilders betrachtet. Bei einer Distanz von 5 m findet keine Filterung der Textur durch die Mip-Maps statt. Wie ersehen werden kann ist die Ähnlichkeit der Verkehrsschilder zur Realität geringer, im Vergleich zu den Ergebnissen mit den Grauwerte-Bildern. Das Überholverbot weist die höchste Ähnlichkeit zur Realität auf. Weiterhin kann festgestellt werden,

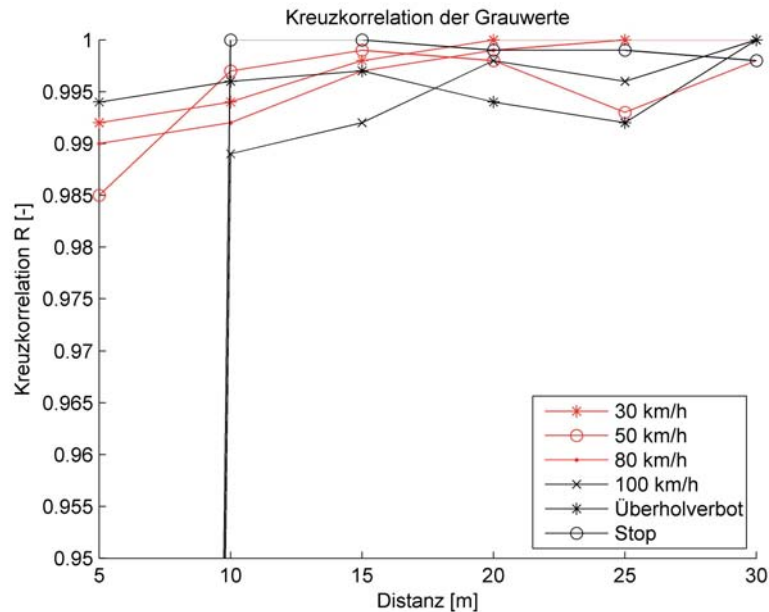


Abbildung 4.36.: Kreuzkorrelation des Grauwerte-Bilds der Schilder über die Distanz

dass die Korrelation für alle Schilder über die Distanz zunimmt. Wenngleich die Korrelation bis auf Ausnahme des 100km/h-Gebots einer starken Schwankung unterworfen ist.

4.4.4.4. Schlussfolgerung

Als Erkenntnis kann festgehalten werden, dass die Verwendung von texturierten Quadraten für die Simulation von Verkehrsschildern als hinreichend genau angesehen werden kann. Durch das Mip-Mapping und die Filterung entstehen keine negativen Effekte im Vergleich zur Realität. Die Abweichungen in der Ähnlichkeit resultieren vorwiegend aus einer zu klein gewählten Schriftart für die Verkehrsschilder.

4.4.5. Zusammenfassung

In diesem Abschnitt wurde der notwendige Realismus für die Simulation von Fahrzeugen, Straßen und Verkehrszeichen untersucht. Im Folgenden werden die Ergebnisse der einzelnen Abschnitte zusammengefasst.

Fahrzeuge

Bei der Betrachtung der notwendigen Modellgenauigkeit für Fahrzeugmodelle stellte sich heraus, dass Modelle aus frei verfügbaren Datenbanken eine ausreichende Qualität aufweisen.

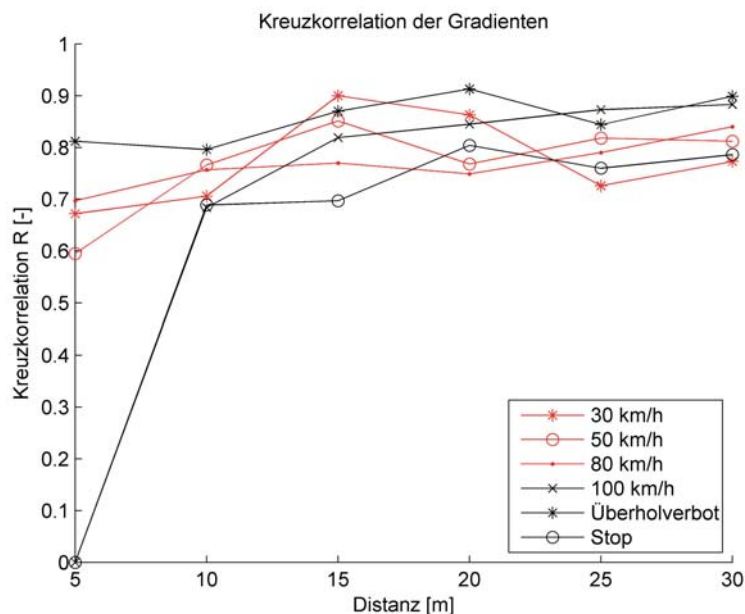


Abbildung 4.37.: Kreuzkorrelation des Gradienten-Bilds der Schilder über die Distanz

Als wichtigeres Kriterium erwies sich die Simulation der Fahrzeugoberfläche. Zur Bewertung der notwendigen Qualität wurden verschiedene Shader mit Mitteln der Bildverarbeitung objektiv verglichen. Das modifizierte Isidoro et. al.-Modell zeigte die beste Leistung. Besonders bei einem Vergleich der Ähnlichkeit bezogen auf das gesamte Bild ist das Modell deutlich überlegen. An zweiter Stelle lag das Blinn-Phong & Cube-Map - Modell, welches knapp gegen das normale Blinn-Phong und Isidoro et. al.-Modell gewinnt. Anhand der Ergebnisse konnte festgestellt werden, dass eine realitätsnahe Simulation des Fahrzeuglacks zu bevorzugen ist. Entsprechend des gewählten Anwendungsfalls der Simulation kann es sinnvoll sein nur ein Blinn-Phong-Modell zu verwenden und evtl. mit einer einfachen Cube-Map zu verbessern.

Ein weiterer wichtiger Aspekt zur Detektion Fahrzeugen sind die Front- und Heckleuchten der Fahrzeuge. Diese Thematik wurde im Rahmen der Diplomarbeit von Melzer [109] ausführlich betrachtet. Die Qualität der Simulation wurde durch einen Abgleich mit zwei realen Referenzszenen validiert. Melzer [109] konnte nachweisen, dass die umgesetzte Simulation hinreichend genau zur Erkennung von Fahrzeugen ist. Weiterhin wurden störende Reflexionen, durch die Spiegelung der Scheinwerfer auf der Straße und Leitposten betrachtet.

Straße

Es wurden die Möglichkeiten zur Simulation einer möglichst realitätsnahen Straßenoberfläche und Markierungen untersucht. Weiterhin wurde die Simulation von nassem Asphalt betrachtet.

Zur Simulation der Strukturen von Asphalt, Pflastersteinen und Sand wurden 2D-Texturen, Normal- und Parallax-Mapping untersucht. Es wurde gezeigt, dass Normal- und Parallax-Mapping keinen signifikanten

Einfluss auf die Bildqualität für die Durchführung von Bildverarbeitungsoperationen zur Spurerkennung bei einer Asphalt oder Schotteroberfläche haben. Das Parallax-Mapping zeigte sich am besten für die Visualisierung eines Kopfsteinpflasters geeignet. Ähnliche Ergebnisse können durch die Modifikation des Kontrasts der Straßen-Textur erreicht werden. Der Implementierungs- und Berechnungsaufwand während der Simulationslaufzeit ist im Vergleich zum Parallax-Mapping deutlich geringer. Basierend auf diesen Erkenntnissen kann durch entsprechende Anpassungen der Straßen-Textur eine Verwendung eines normalen Texture-Mappings für den Test von Spurerkennungsfunktionen ausreichen.

Zur Beschreibung der Reflexion der Straßenoberfläche wurde die Blinn-Phong-, Ward- und Schlick-BRDF bewertet. Anhand der durchgeführten Vergleiche kann festgestellt werden, dass sich die Blinn-Phong- und Ward-BRDF zur Simulation eignen. Die Schlick-BRDF kann keine zufriedenstellenden Ergebnisse bereitstellen und besitzt im Vergleich zu den anderen BRDFs einen erhöhten Parametrierungsaufwand. Die Blinn-Phong BRDF erweist sich bei einem flachen Lichtvektor und damit großen Schnittwinkel zwischen Oberflächennormalen und Sichtvektor als nachteilig, da die Reflexion eine unnatürliche Erscheinung aufweist. Die Ward-BRDF erzeugt auch bei einer tief stehenden Sonne ein realitätsnahes Glanzlicht.

Zur Beschreibung von Straßenmarkierungen wurden eine einfarbige, texturierte und mit einer Normal-Map verbesserte Straßenmarkierung verglichen. Bei Vergleich der Markierungen zeigte sich, dass für die Durchführung von einfachen Tests eines Spurerkennungsalgorithmus die einfarbige Straßenmarkierung eine ausreichende Güte besitzt. Durch Reduktion des Kontrasts zwischen Markierung und Straßenoberfläche, können auch Grenzfälle betrachtet werden. Durch die Applikation einer Textur auf die Straßenmarkierung ist es möglich die geometrische Erscheinung der Markierung zu verändern. Durch die Anwendung einer Normal-Map kann die Erscheinung der Markierung für den menschlichen Betrachter verbessert werden, für das maschinelle Sehen kann damit ein Rauschen simuliert werden. Das Rauschen kann auch durch eine Modifikation der 2D-Textur erreicht werden. Daher kann die Verwendung einer Normal-Map als optional angesehen werden und hat keinen Einfluss auf die Qualität der Simulation.

Zur Simulation einer nassen Straßenoberfläche wurde ein Modell hergeleitet. Dieses ist in der Lage nassen Asphalt mit Reflexionen der Umgebung zu simulieren. Das Modell wurde basierend auf dem Fresnel-Gesetz für die Brechung an einem Grenzübergang von einem Medium in ein anderes definiert. Als Parameter für die Stärke der Reflexion wird der Wasserstand verwendet. Zur Simulation von aufgewirbeltem Wasser eines vorausfahrenden Fahrzeugs wird ein physikalisch motiviertes Partikelsystem verwendet. Die Dichte der Partikel nimmt mit dem Abstand vom Fahrzeug und der Höhe über dem Boden ab. Weiterhin werden Reifenspuren simuliert, diese reduzieren das Wasser auf der Straße. Die Verwendbarkeit der Simulation wurde durch den Vergleich einer realen Szene mit einer simulierten Szene nachgewiesen. Als schwierig erweist sich die Parametrierung der Modelle, wenn eine identische Szene hergestellt werden soll.

Verkehrsschilder

Als Modelle werden häufig Bilder realer Verkehrsschilder als Texturen verwendet. Da die Texturen nur in definierten Auflösungen vorliegen, galt es zu klären welche Auswirkung Mip-Maps auf das Bild haben.

Dieser Sachverhalt wurde mit simulierten und realen Verkehrsschildern in verschiedenen Abständen zur Kamera untersucht. Als Vergleichsmaß wurde die Kreuzkorrelation zwischen den simulierten und dem realen Verkehrsschild bestimmt. Die Auswertung wurde auf Grauwerten- und Gradienten-Bildern durchgeführt.

Als Ergebnis der Kreuzkorrelation auf den Grauwerte-Bildern wurde festgestellt, dass die Ähnlichkeit über die Distanz zur Kamera zunimmt. Die Überholverbot- und Stoppschilder wiesen über die Distanz eine Schwankung um den Mittelwert des Ähnlichkeitsmaßes auf. Ein ähnliches Ergebnis folgt aus der Betrachtung der Kreuzkorrelation der Gradientenbilder. Wenngleich festgestellt werden muss, dass die Ähnlichkeit der Gradienten eine geringere Qualität als bei Betrachtung der Grauwerte-Bilder aufweist.

Als Ergebnis kann festgehalten werden, dass die Verwendung von texturierten Quadraten für die Simulation von Verkehrsschildern als hinreichend genau angesehen werden kann. Durch das Mip-Mapping und die Filterung entstehen keine negativen Effekte im Vergleich zur Realität. Die Abweichungen in der Ähnlichkeit resultieren vorwiegend aus einer zu klein gewählten Schriftart für die Verkehrsschilder.

4.5. Synthese von Kamerabildern

Zur Stimulation der Signalverarbeitung der Kamera ist es denkbar das virtuelle Bild durch eine optische Übertragung (Abfilmen) oder durch direktes beschreiben des Bildspeichers in die Kamera zu übertragen. In beiden Fällen ist es notwendig ein für die Bildverarbeitung möglichst realitätsnahes Bild zu erzeugen und nachteilige Effekte der Computergrafik zu vermeiden. Wenn das Bild direkt in die Kamera eingespeist wird, muss außerdem der optische Pfad simuliert werden. In diesem Abschnitt wird auf die Erzeugung von synthetischen Kamerabildern eingegangen. Weiterhin wird die Simulation von Effekten zur Steigerung des Realismus beschrieben.

4.5.1. Bildsyntheseprozess der Computergrafik

Die Erzeugung eines computergenerierten Bilds wird als Bildsynthese bezeichnet. Bei dieser wird aus geometrischen Modellen, Licht- und Oberflächenbeschreibungen ein Abbild der Szene erzeugt. In der Echtzeit-Computergrafik wird als Syntheseverfahren die Rasterisierung verwendet.

Im Folgenden werden die wichtigsten Schritte des Prozesses dargestellt. Die Abbildungen und Beschreibungen legen Wert auf einfache Darstellung. Für eine detaillierte Erläuterung sei auf die Dokumentation von OpenGL [175] oder [14] verwiesen. In Abbildung 4.38 ist der Bildgenerierungsprozess einer synthetischen Kamera der Bildaufnahme einer realen Kamera gegenübergestellt. Die reale Bildaufnahme ist in Abschnitt 2.2.5 ausführlich beschrieben. Zur Realisierung einer Bildaufnahme in einer virtuellen Szene

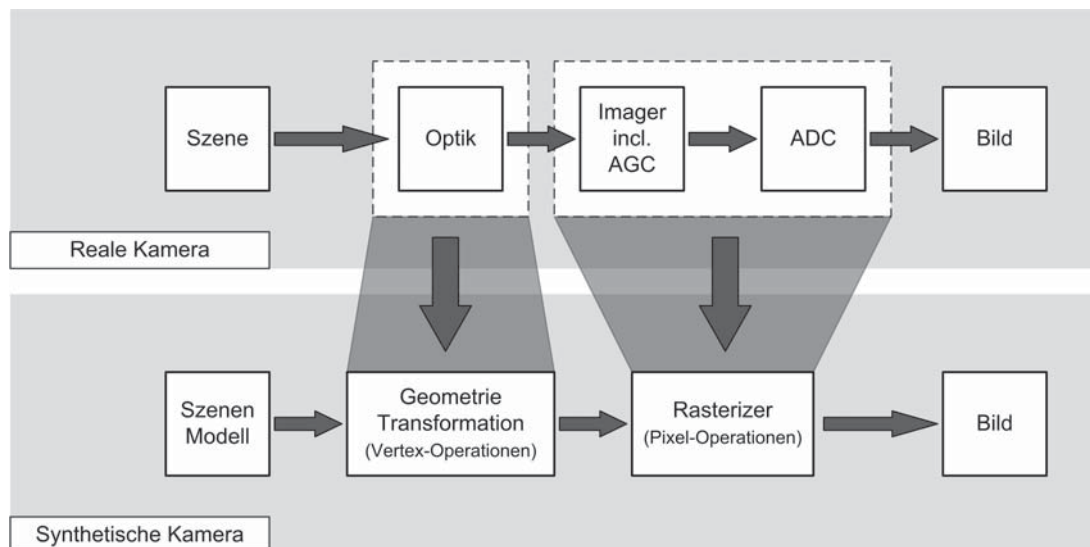


Abbildung 4.38.: Bildgenerierungsprozess: synthetische vs. reale Kamera

werden die Abbildungseigenschaften einer realen Optik durch geometrische Transformationen und mittels des Lochkameramodells beschrieben.

Dazu werden als erstes die Vertex-Koordinaten aus dem Objektkoordinatensystem in das Weltkoordinatensystem transformiert. Im nächsten Schritt findet eine Transformation in das Koordinatensystem der

Kamera, den View-Space, statt. Dies ist vergleichbar mit der Ausrichtung des Objektivs auf die Szene. Die Ausrichtung des Kamera-Koordinatensystems variiert zwischen OpenGL und DirectX. In diesem haben alle Eckpunkte einen relativen Bezug zum Ursprung der Kamera, dem Augpunkt. Die finale Abbildung der Eckpunkte, Vertexe, auf einen 3D-Punkt in der Bildebene findet mittels einer perspektivischen Transformation statt. In der programmierbaren Grafik-Pipeline können die geometrischen Transformationen mittels eines Vertex-Shaders modifiziert werden.

Die Abbildungseigenschaften der perspektivischen Transformation der Lochkamera werden mittels eines Frustrums (Pyramidenstumpf) beschrieben, [14]. Das Viewing-Frustrum ist in Abbildung 4.39 dargestellt. Dieses wird mit der Projektionsmatrix für OpenGL P_{OpenGL} wie folgt definiert [14]:

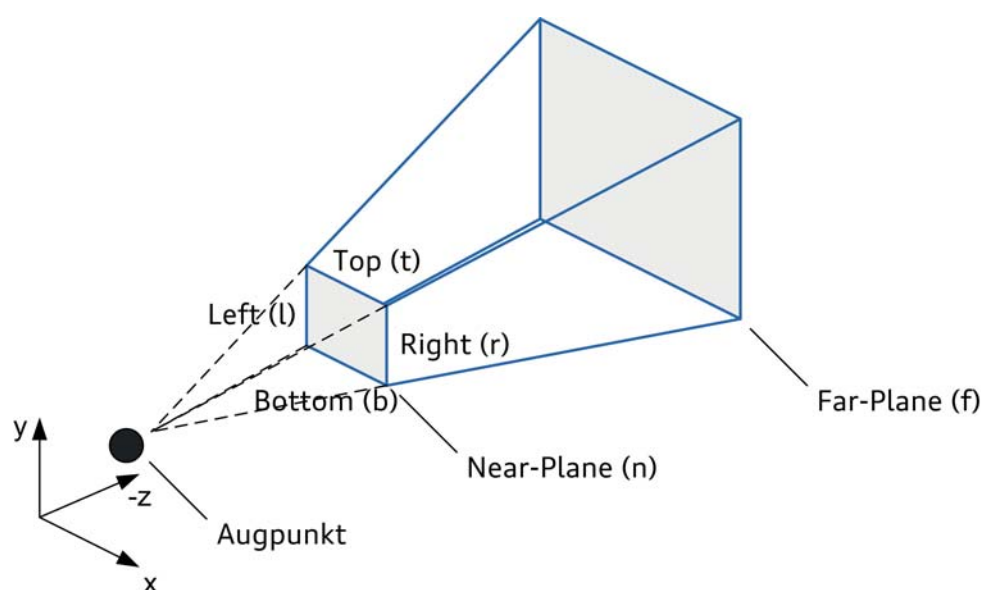


Abbildung 4.39.: Definition des Viewing-Frustrums, nach [14]

$$P_{OpenGL} = \begin{bmatrix} \frac{2|n|}{r-l} & 0 & \frac{r+l}{r-l} & 0 \\ 0 & \frac{2|n|}{t-b} & \frac{t+b}{t-b} & 0 \\ 0 & 0 & -\frac{|f|+|n|}{|f|-|n|} & -\frac{2|f||n|}{|f|-|n|} \\ 0 & 0 & -1 & 0 \end{bmatrix} \quad (4.82)$$

Die Definition der Öffnungswinkel der Kamera erfolgt durch das Verhältnis aus der Near-Plane und der Left-, Right-, Top- und Bottom-Kante. Nach der Transformation liegen die Vertex-Punkte als Normalized Device Coordinates vor. Die Koordinaten beinhalten neben der x- und y-Position des Vertices einen Tiefenwert. Dieser wird bei der Rasterisierung zur Verdeckungsprüfung verwendet.

Die Near- und Far-Plane und Kanten des Pyramidenstumpfs werden weiterhin für das Clipping verwendet. Das Clipping wird nach der perspektivischen Transformation im Canonical View Volume [14] durchgeführt. Während des Clippings wird geprüft, welche Teile des Polygons im Canonical View Volume liegen. Im letzten Schritt werden die NDC der Vertexe in Fragmente der Bildschirmkoordinaten transformiert. In der Fixed-Function Pipeline werden die Fragmente direkt an den Rasterisierer übertra-

gen. In der programmierbaren Pipeline kann mittels eines Fragment-Shader die Berechnung der Farbe der Fragmente modifiziert werden. Bei der abschließenden Rasterisierung werden die Fragmente, Farbinformationen und Alpha-Werte aus der vorherigen Stufe zur Generierung des Bilds verwendet. Als Speicher für die Rasterisierung dient der Framebuffer. Die Fragmente werden mittels des Edge-Fill oder des Scanline-Algorithmus [174] [14] ausgefüllt. Da die Fragmente der Objekte sequentiell abgearbeitet werden ist es möglich, dass mehr als ein Fragment der gleichen Bildschirmkoordinate zugeordnet ist. Daher muss geprüft werden, welches Fragment sichtbar ist. Diese Prüfung wird mittels der z-Komponente des Fragments durchgeführt, welche den Abstand des Fragments zur Bildebene beschreibt. Die Tiefenwerte der im Framebuffer liegenden Fragmente werden dazu im Depth-Buffer (Tiefenpuffer) gespeichert. Wenn sich das aktuelle Fragment näher an der Bildebene befindet, d.h. die z-Komponente kleiner ist als die im Puffer gespeicherte, wird der alte durch den neuen Bildpunkt ersetzt, ansonsten wird der das Fragment verworfen.

4.5.2. Steigerung der Realitätsnahe eines synthetischen Bilds

Durch die Abbildung einer Szene mittels geometrischer Transformationen in ein Bild treten im Vergleich zu einer realen Bildaufnahme bei der Bildsynthese die in Abschnitt 2.2.5 beschriebenen Abbildungsfehler nicht auf. Außerdem werden Abbildungseigenschaften des Strahlungsaufnehmers oder der Optik nicht berücksichtigt. Daher gilt es zu prüfen, wie diese Effekte sinnvoll in den Bildgenerierungsprozess integrierbar sind, um die Realitätsnähe des Bilds zu steigern. Zur Realisierung dieser Effekte wird ein Modell des Strahlungsaufnehmers und der Optik mit den folgenden Eigenschaften erstellt:

- Anti-Aliasing-Filter
- Belichtungssteuerung
- Antwortfunktion
- Farbfilter
- Rauschen
- Tiefenschärfe
- Verzeichnungen

Das Kameramodell erhält als Eingangsdaten zwei Texturen, die Bestrahlungsstärke der Szene und das Tiefenbild, vgl. Abbildung 4.40. Als Vorverarbeitung für den Tiefenschärfefeffekt wird die Auflösung der Bestrahlungsstärke-Textur halbiert. In der Simulations-Stufe wird das eigentliche Kameramodell mit Verzeichnungen, Tiefenschärfe, Rauschen, Antwortfunktion, Farb- und Anti-Aliasing-Filter simuliert. Die Belichtungszeit wird mittels einer Belichtungsregelung auf dem Ausgangsbild bestimmt.

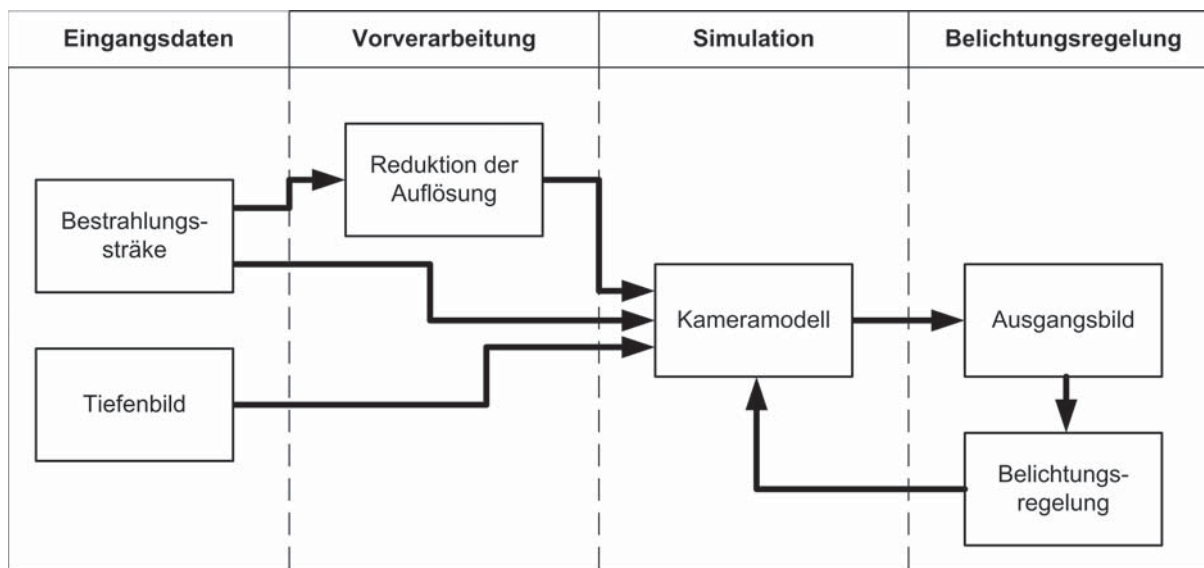


Abbildung 4.40.: Umsetzung der Kamerasimulation

4.5.3. Anti-Aliasing-Filter

Wie bei der Vorstellung des Bildgenerierungsprozess beschrieben, wird bei der Rasterisierung jedem Pixel nur ein Farbwert zugeordnet. Daher erscheinen eigentlich gerade Kanten in der Form einer "Treppe". Zur Reduktion dieser „Treppeneffekte“ wurden eine Vielzahl von Tiefpassfiltern entwickelt. Die Filter unterscheiden sich vorwiegend in der Anzahl und die Positionierung der Abtastpunkte im Ursprungsbild [14].

4.5.3.1. Untersuchung

Im Folgenden soll die Auswirkung von Anti-Aliasing-Filtern auf Kanten im Bild untersucht werden. Dafür wurde eine typische Szene aus der Fahrsimulation in achtfacher Auflösung berechnet und anschließend mittels verschiedener Anti-Aliasing-Filter nach [14] bearbeitet, vgl. Abbildung 4.41. In der Abbildung sind die zwei für die Auswertung verwendeten Messfenster farblich markiert. Zur Untersuchung der Auswirkungen wurden mittels des Sobel-Operators Kantenbilder erzeugt. Der Betrag der Gradienten im Messfenster wurde zur Durchführung des Vergleichs normiert.

4.5.3.2. Ergebnisse

In Abbildung 4.42 sind die Ergebnisse der Filterung dargestellt. Die Kanten der realen Spurmarkierung sind bedingt durch das Rauschen der Straßenoberfläche weniger stark ausgeprägt. Die mittels Computergrafik erzeugten Kanten unterscheiden sich bei einer visuellen Betrachtung nur minimal. Zur Durchführung eines quantitativen Vergleichs wurde der Gradientenbetrag über die Pixel aufgetragen, siehe Abbildung 4.43. Aus der ersten Messung wurde Zeile 20 verwendet und aus Messung 2 die erste Zeile.

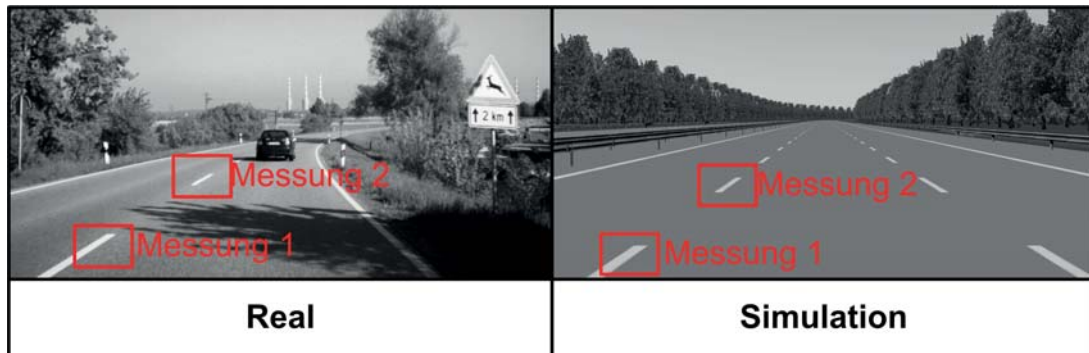


Abbildung 4.41.: Verwendete Szene und Messfenster für Vergleich der Anti-Aliasing Techniken

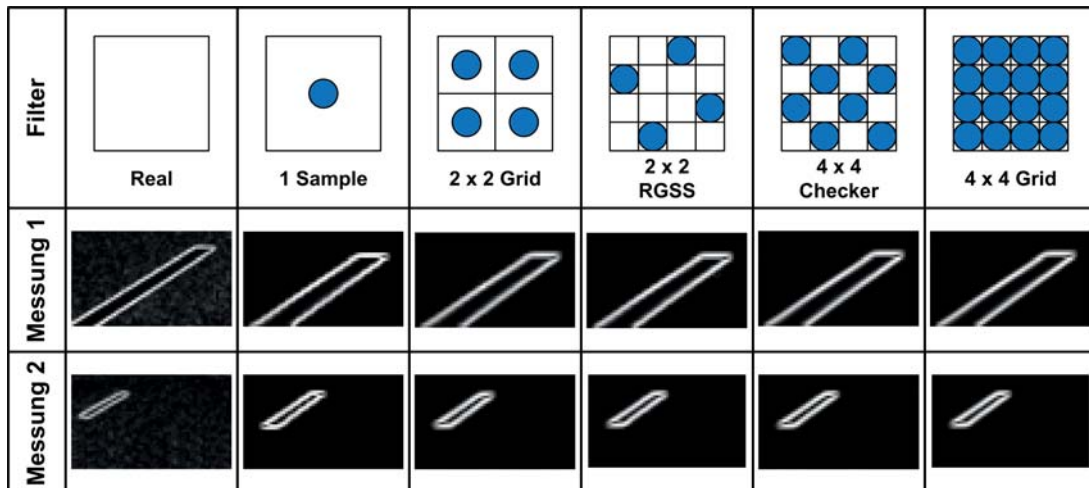


Abbildung 4.42.: Vergleich der Anti-Aliasing-Filter für definierte Messfenster

Um einen besseren Vergleich zu ermöglichen, wurden die Gradienten auf die erste steigende Flanke synchronisiert. Wie in der Abbildung gesehen werden kann ist die Amplitude des Gradienten in der realen Messung geringer als in der Simulation. Bei Auswertung von Messfenster 1 kann festgestellt werden, dass das ungefilterte Bild am schlechtesten abschneidet da die Form des Gradienten nicht der Referenz entspricht. Bei der ersten Kante der Markierung führen alle Filter-Techniken, bis auf Ausnahme des 4x4 Grids zur einer Verstärkung der Amplitude im Vergleich zur ungefilterten Referenz. Ein ähnliches Verhalten ist auch bei der zweiten Kante beobachtbar. Die Erscheinung des Gradienten wird von allen Techniken gleich gut abgebildet.

Im zweiten Messfenster kann festgestellt werden, dass die reale Kante gedämpft wird. Die ungefilterte Kante spiegelt die Realität am wenigsten wieder. Der RGSS-Filter führt zu einer Inversion des Vorzeichens des Gradienten.

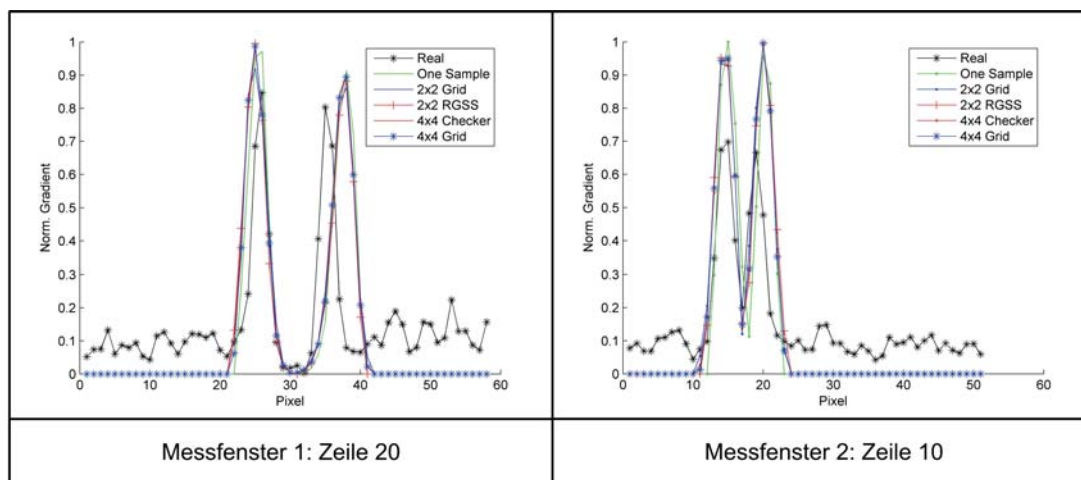


Abbildung 4.43.: Auswertung der Anti-Aliasing Techniken: links Messfenster 1 Zeile 20 und rechts Messfenster 2 Zeile 10

4.5.3.3. Auswertung

Zusammenfassend lässt sich feststellen, dass keiner der Anti-Aliasing-Filter einen Vorteil oder Nachteil bei der Kantenglättung in dem untersuchten Anwendungsfall bietet. Daher kann auf Grundlage der vorliegenden Erkenntnisse ein 2x2 Grid als ausreichend zur Reduktion der Aliasing-Effekte angesehen werden.

4.5.4. Modell des Strahlungsaufnehmers

In diesem Abschnitt wird auf die Erstellung des Modells eines Strahlungsaufnehmers nach CMOS-APS-Bauart eingegangen. Als Grundlage für die Bedatung des Modells soll das Datenblatt eines Bildsensors verwendet werden.

Die Erstellung eines Modells einer Digitalkamera ist durch Farrel et. al [65], basierend auf Ray-Tracing, beschrieben. Klein [99] beschreibt eine einfache Möglichkeit zur Simulation von Kameras für Augmented-Reality Anwendungen. Das Übertragungsverhalten des Modells soll mit den Gleichungen von [65] motiviert werden.

4.5.4.1. Gemessene Beleuchtungsstärke des Strahlungsaufnehmers

Im ersten Schritt wird die Strahldichte der Szene L_{Szene} in die vom Sensor empfangene Beleuchtungsstärke E_{Sensor} umgerechnet, nach [65] folgt:

$$E_{Sensor} = \frac{\pi T_{Optik}(\lambda)}{4k^2} L_{Szene} \quad (4.83)$$

Wobei k die Blendenzahl und $\tau_{Optik}(\lambda)$ der Transmissionsfaktor bei einer bestimmten Wellenlänge ist. Zur Beschreibung des Transmissionsverhaltens wird der RGB-Farbraum verwendet. Daher muss der Trans-

missionsfaktor für die Basiswellenlängen des Farbmodells ermittelt werden [174]. Der Transmissionsfaktor für die Linsen der Optik wird auf Basis des Fresnel-Terms für einen senkrechten Lichteinfall bestimmt [167]. Die Transmission des Lichts zum Sensor findet durch die Übergänge von der Luft in das Glas der Linse und durch die Linse zurück in die Luft auf den Sensor statt, damit folgt für den Transmissionsgrad der Linse τ_{Optik} :

$$\tau_{Optik}(\lambda) = (1 - \varrho_{orth}(\lambda))^2 = 1 - 2 \cdot \varrho_{orth}(\lambda) + \varrho_{orth}(\lambda)^2 \quad (4.84)$$

Diese Gleichung besitzt nur für die optische Achse Gültigkeit [65]. Zur Beschreibung der Strahlungsabfalls außerhalb der optischen Achse verwenden die Autoren das Cos4-Gesetz [141]:

$$R(\theta) = \cos(\theta)^4 \quad (4.85)$$

Dabei beschreibt θ den Winkel zwischen dem Hauptpunkt und dem Bildpunkt auf der Bildachse. Damit ergibt sich für die gesamte Beleuchtungsstärke auf der Sensoroberfläche:

$$E_{Sensor} = R(\theta) \frac{\pi T_{ges}(\lambda)}{4k^2} L_{Szene} \quad (4.86)$$

In der Veröffentlichung von [65] wird eintreffende Strahlung weiterhin durch eine Unschärfefunktion gefiltert. Dieser Schritt wird bei dem angewendeten Modell vernachlässigt.

4.5.4.2. Antwortverhalten des Strahlungsaufnehmers

Das Antwortverhalten eines Detektors, wie einer Kamera, kann nach [174] wie folgt formuliert werden:

$$Response = k \int w(\lambda) L(\lambda) d(\lambda) \quad (4.87)$$

Dabei stellt $w(\lambda)$ die spektrale Empfindlichkeit des Detektors für die Wellenlänge dar und $L(\lambda)$ die empfangene Leuchtdichte. Zur Simulation des Antwortverhaltens der Kamera verwenden Farrel et. al. [65] die folgende Gleichung:

$$e_i = T \int_T \int_{\mathbf{x}} S_i(\lambda) A_i(\mathbf{x}) E_{Sensor}(\lambda, \mathbf{x}) d\lambda dx \quad (4.88)$$

In der Gleichung beschreibt T die Beleuchtungszeit, S die Quantenausbeute, A die Blendenfunktion. Danach ist die Anzahl der gemessenen der Elektronen durch den Photodetektor, die Summe der Elektronen über die Öffnung der Blende und die Wellenlänge.

Im Rahmen dieser Arbeit wurde die folgende Gleichung zur Berechnung der gemessenen Bestrahlung durch Sensor H für jeden Pixel im RGB-Farbraum angewendet:

$$H(\mathbf{x})(R, G, B) = t_{Belichtung} \cdot \eta(R, G, B) \cdot \mathbf{C}(\mathbf{x})(R, G, B) \cdot E_{Sensor}(R, G, B) \quad (4.89)$$

Mit der Belichtungszeit $t_{\text{Belichtung}}$, der Quantenausbeute η , dem Farbfilter \mathbf{C} und Beleuchtungsstärke des Sensors E_{Sensor} .

Farbfilter

Für die Konvertierung des RGB-Farbraums in den Farbraum der Kamera wird die folgende Transformationsmatrix \mathbf{C} , nach [99], angewendet:

$$\mathbf{C} = \begin{bmatrix} r_{\text{Farbfilter}} \\ g_{\text{Farbfilter}} \\ b_{\text{Farbfilter}} \end{bmatrix} = \begin{bmatrix} c_{rr} & c_{rg} & c_{rb} \\ c_{gr} & c_{gg} & c_{gb} \\ c_{br} & c_{bg} & c_{bb} \end{bmatrix} \quad (4.90)$$

Die Matrix \mathbf{C} beschreibt die Zusammensetzung des Kamerafarbraums auf Basis des RGB-Farbraums. Die erste Zeile beschreibt die Komposition der Farbe rot, die zweite Zeile die Farbe grün und die dritte Zeile die Farbe blau.

Die Elemente der Matrix c können aus dem Datenblatt eines Bildsensors bestimmt werden. Dazu muss die Antwort des Sensors auf die Wellenlängen des RGB-Farbraums, $\lambda_r = 700\text{nm}$, $\lambda_g = 546\text{nm}$ und $\lambda_b = 438\text{nm}$, ermittelt werden. Die Summe der Antworten für jeden Farbkanal des Farbfilters muss normiert werden. Bei dem monochromen Sensor [3] vereinfacht sich die Matrix zu:

$$\mathbf{C}_m = \begin{bmatrix} c_{mr} & c_{mg} & c_{mb} \\ c_{mr} & c_{mg} & c_{mb} \\ c_{mr} & c_{mg} & c_{mb} \end{bmatrix} = \begin{bmatrix} 0.204081633 & 0.346938776 & 0.448979592 \\ 0.204081633 & 0.346938776 & 0.448979592 \\ 0.204081633 & 0.346938776 & 0.448979592 \end{bmatrix} \quad (4.91)$$

Anwendungsbeispiel In Abbildung 4.44 sind die Ergebnisse der Simulation verschiedener Farbfilter dargestellt. Zur besseren Darstellung der Auswirkungen einer Anwendung der Filter wurden im Bild nur die vom Farbfilter beeinflussten Kanäle verwendet. Als Referenz ist das Ausgangsbild ohne die Anwen-

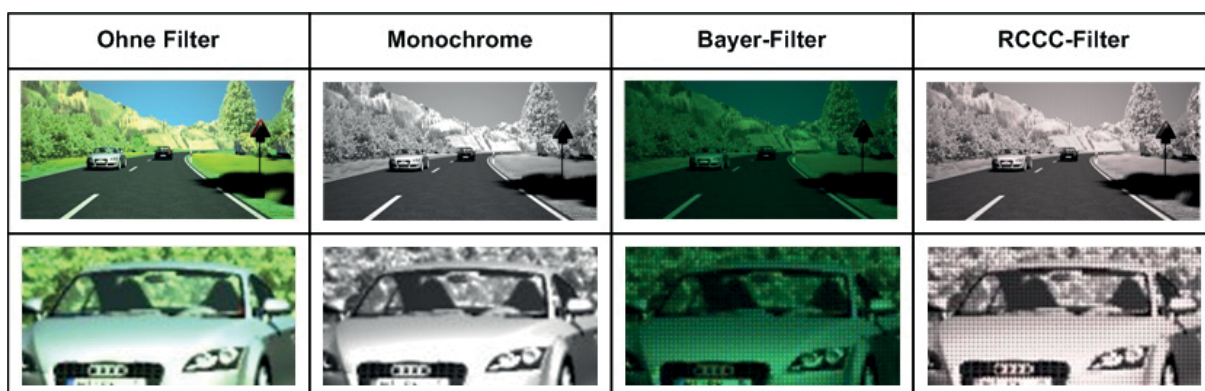


Abbildung 4.44.: Simulation der Farbfilter einer Kamera

dung eines Filters dargestellt. Im unteren Bereich der Abbildung wurde eine Vergrößerung des Fahrzeugs vorgenommen um die Auswirkung auf einzelne Pixel darzustellen. Als zweites wurde ein monochromer

Filter angewendet, welcher durch die Matrix C_m definiert wurde. In der dritten Abbildung wurde ein Bayer-Filter einer Farbkamera modelliert. Die Demosaikierung, d.h. Vermischung der Pixel, wurde nicht durchgeführt. Da der Bayer-Filter die doppelte Anzahl grüner Pixel im Vergleich zu blauen oder roten Pixel verwendet erscheint das Bild grünlich. Im unteren Bereich der Abbildung sind deutlich die Pixel erkennbar. Im vierten Teilbild ist die Anwendung eines RCCC-Filters dargestellt. Durch den Rotkanal erhält das Bild eine rötliche Färbung.

Quantenausbeute

Zur Simulation der Quantenausbeute η wird aus dem Datenblatt der Kamera in Abhängigkeit der Wellenlänge des RGB-Farbraums, $\lambda_r = 700nm$, $\lambda_g = 546nm$ und $\lambda_b = 438nm$, die Quantenausbeute $\eta(\lambda)$ für die einzelnen Farbkanäle ermittelt. Für den im Datenblatt [7] dokumentierten Bildsensor folgt eine Quantenausbeute für den RGB-Farbraum $\eta(R, G, B)$:

$$\eta = \begin{bmatrix} \eta_r \\ \eta_g \\ \eta_b \end{bmatrix} = \begin{bmatrix} 0.48 \\ 0.51 \\ 0.50 \end{bmatrix} \quad (4.92)$$

Belichtungssteuerung

Zur Anpassung der Belichtungszeit $t_{Belichtung}$ kann eine beliebige Belichtungsregelung in die Simulation implementiert werden. In der Computergrafik wird die Anpassung der Helligkeitswerte hinsichtlich eines optimalen Kontrasts als Tone-Mapping bezeichnet. Dazu schlägt Reinhard et. al. vor, die Helligkeitswerte L_w des gesamten Bilds zu summieren und den Mittelwert \hat{L}_w zu bilden [155]. Anstelle einer linearen Addition wird dieses mit einer dem Weber-Fechner Gesetz nachempfundenen Funktion durchgeführt:

$$\hat{L}_w = exp \left(\frac{1}{N} \sum_{x,y} \log(\delta + L_w(x, y)) \right) \quad (4.93)$$

Dabei sind x und y die Koordinaten der Bildpunkte, N die Anzahl der Bildpunkte und δ ein Stabilitätsterm. Der Mittelwert dient als Grundlage zur Adaption aller Helligkeitswerte im Bild, z.B. mittels eines einfachen linearen Skalierungs-Terms a [14]:

$$L(x, y) = \frac{a}{\hat{L}_w} \cdot L_w(x, y) \quad (4.94)$$

Ein ähnlicher Ansatz wird zur Berechnung der Belichtungszeit in einer realen Kamera verwendet. Da in der Realität die Schätzung der neuen Belichtungszeit auf dem zuletzt aufgenommenen Bild basiert ändert sich der Algorithmus. Nach [101] kann die neue Belichtungszeit $t_{Belichtung}(k+1)$ nach folgendem Ansatz bestimmt werden. Dabei gilt die Annahme $H = E \cdot t_{Belichtung}(k)$, d.h. dass die Helligkeit im Bild linear mit der Belichtungszeit zusammen hängt. Damit folgt für den Algorithmus:

1. Nehme ein Bild mit einer bekannten Belichtungszeit $t_{Belichtung}(k)$ auf. Definiere einen gewünschten Mittelwert des Bilds zu $\hat{H}_{Des} = 0.5$
2. Berechne in diesem Bild den Mittelwert \hat{H}
3. Berechne eine neue Belichtungszeit $t_{Belichtung}(k+1)$ so, dass $\hat{H} = 0.5$ gilt. $t_{Belichtung}(k+1) = t_{Belichtung}(k) \frac{\hat{H}_{Des}}{\hat{H}(k)}$
4. Nehme mit der neuen Belichtungszeit ein neues Bild auf

Anwendungsbeispiel In Abbildung 4.45 ist der Einfluss der beschriebenen Belichtungssteuerung auf eine Bildfolge dargestellt. Im ersten Bild wurde die Szene mit der ursprünglichen Belichtungszeit aufgenommen. Auf Basis des Mittelwerts wurde für das zweite Bild eine neue Belichtungszeit bestimmt. Wie im Frame 1 ersehen werden kann ist dieses deutlich besser ausgeleuchtet. In Frame 2 und Frame 3 ist keine Verbesserung der Ausleuchtung zu erkennen.

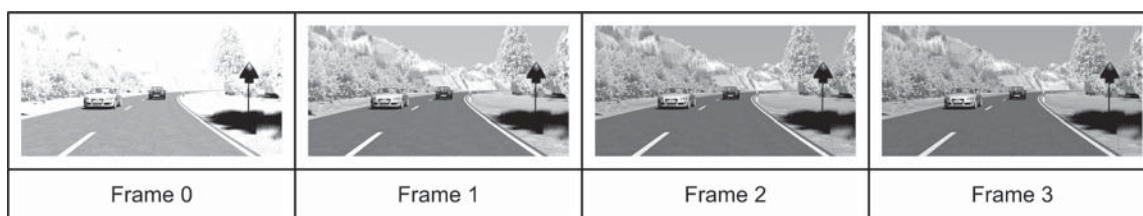


Abbildung 4.45.: Einfluss der Belichtungssteuerung auf das Bild

Konvertierung der Bestrahlung in Grauwerte

Abschließend wird die einfallende Bestrahlung H mittels einer Kennlinie in Grauwerte umgerechnet. Die Funktion zur Umrechnung einer Belichtung in einen entsprechenden Grauwert ist von der Beschaffenheit des Strahlungsaufnehmers abhängig. Zur Quantifizierung der Qualität des Imagers ist diese Antwortfunktion $g(H)$ im zugehörigen Datenblatt dokumentiert [3]. Basierend darauf kann eine Look-Up Table zur Umrechnung erstellt werden.

Eine weitere Möglichkeit ist die in Abschnitt 2.2.5 beschriebene Abbildungsfunktion durch eine logarithmische Umrechnungsfunktion zu approximieren. Dazu kann der Dynamikbereich und die Sensitivität der Kamera aus dem Datenblatt verwendet werden. Für den Pegel L [dB] einer Feldstärke folgt,

$$L = 20 \log_{10} \frac{H_2^2}{H_1^2} \quad (4.95)$$

Damit folgt zum Beispiel für den Sensor LM9617 aus dem Datenblatt [3],

$$L_{max} = 20 \log_{10} \frac{H_2^{2,max}}{H_1^2} = 57dB \quad (4.96)$$

Mit dem Maximalwert der Spannung $U_{2,max}$ welche bei einer vollen Ausnutzung des Dynamikbereichs auftritt folgt

$$H_{2,max} = \frac{BitsProPixel}{Sensitivitaet} = \frac{4096}{28700} lux \cdot s = 0.142 \quad (4.97)$$

damit kann H_1^2 wie folgt bestimmt werden:

$$H_1^2 = \frac{H_{2,max}^2}{10^{L_{max}/20}} = \frac{0.142^2}{10^{57/20}} = 2.84824 \cdot 10^{-5} lux \cdot s \quad (4.98)$$

Als letzte Möglichkeit kann die Funktion als OECF (Opto-Electronic Conversion Function) nach ISO14524 [9] empirisch ermittelt werden. Diese beschreibt direkt die Umrechnung der Helligkeitswerte in Grauwerte.

Anwendungsbeispiel Zur Untersuchung der Anwendbarkeit sind in Abbildung 4.46 das Modell einer OECF und eine vermessene OECF einer realen Fahrzeugkamera dargestellt. Als Referenz wurde die Szene ohne die OECF abgebildet. Wie beim Vergleich der Bilder festgestellt werden kann stellt das Modell der OECF, welches aus dem Datenblatt der Kamera bedatet wurde, eine gute Approximation der Realität dar.

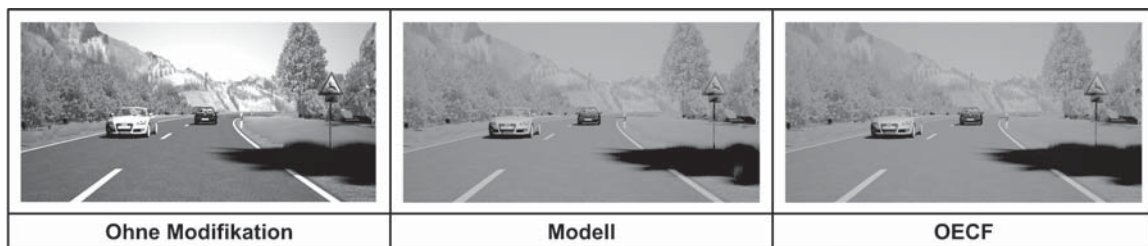


Abbildung 4.46.: Vergleich verschiedener Antwortfunktionen

4.5.5. Rauschen

Im Folgenden wird ein physikalisch motiviertes Modell zur Simulation von Bildrauschen auf Basis des Modells von Irie et. al. [88] [87] beschrieben. Die Leuchtdichte des verrauschten Bilds $L_{v,Noise}$ setzt sich wie folgt zusammen:

$$L_{v,Noise} = (L_v + L_v \cdot N_{PRNU}(L_v) + N_{SN_{ph}}(L_v) + N_{DSNU} + N_{read}) \cdot N_D \cdot N_{filt} + N_Q(L_{filt}) \quad (4.99)$$

Dabei stellt L_v die Leuchtdichte des Ursprungsbilds dar. Anschließend folgen die additiven und multiplikativen Anteile der Rauschquellen: N_{PRNU} - Photo Response Non-Uniformity, $N_{SN_{ph}}$ - Photon Shot Noise, N_{FPN} - Offset fixed-pattern noise, N_{read} - Readout Noise, N_D - Demosaicing Noise, N_{filt} - Digital Filtering und N_Q Quantization Noise.

Die physikalischen Hintergründe der Rauschquellen werden in Abschnitt 2.2.5 behandelt. Das Rauschen

bei der Demosaikierung [88] tritt bei der Interpolation der einzelnen Farbkanäle eines Bayer-Musters auf. Das Rauschen der digitalen Filterung ergibt sich bei einer Anpassung der Helligkeiten und Farbangleichung durch den Analog-Digital-Wandlung oder durch eine Nachverstärkung.

Die Amplitude des Rauschens wird mittels folgender Funktion berechnet:

$$N_{Quelle}(I) = 3\sigma_{Quelle}(I) \cdot n_{Noise,Quelle}(\sin(t), \cos(t)) \quad (4.100)$$

Dabei ist σ die Standardabweichung der Rauschquelle. Die Variable n_{Noise} ist das Ergebnis eines Textur-Samplings aus einer Textur mit weißem Rauschen. Die Koordinaten für den Zugriff werden als eine Funktion der Zeit t mittels trigonometrischer Gleichungen ermittelt. Die Koeffizienten für die Standardabweichung der Rauschquelle ergeben sich nach [88] wie folgt:

$$\sigma_{PRNU} = k_{1,PRNU} \cdot L_v + k_{0,PRNU} \quad (4.101)$$

$$\sigma_{SN_{ph}} = k_{1,SN} \sqrt{L_v} + k_{0,SN} \quad (4.102)$$

$$\sigma_{DSNU} = k_{0,DSNU} \quad (4.103)$$

$$\sigma_{N_{Read}} = k_{0,Read} \quad (4.104)$$

$$\sigma_{N_D} = k_{0,N_D} \cdot L_v \quad (4.105)$$

$$\sigma_{N_Q} = k_{0,N_Q} \cdot L_v \quad (4.106)$$

Für eine Anwendung in der Grafik-Engine müssen die Koeffizienten des Rauschens auf den Dynamikbereich der Kamera skaliert werden. Die Wahl der Koeffizienten kann als Skalar oder Vektor erfolgen, falls z.B. eine spektrale Empfindlichkeit des Rauschens abgebildet werden soll. Durch Brouk et. al. [37] wird das Rauschverhalten eines CMOS-Strahlungsaufnehmers betrachtet. Danach setzt sich das Rauschen wie folgt zusammen:

$$\sigma_{Output}^2 = (\sigma_{SN}^2 + \sigma_{CT}^2) L_v^2 + \sigma_{Read}^2 \quad (4.107)$$

Das DSNU- und PRNU-Rauschen beschreibt in der Definition das Rauschen während des Ladungstransfers σ_{CT}^2 . Mittels des Modells aus Gleichung 4.99 kann diese wie folgt transformiert werden:

$$L_{v,Noise} = L_v + (N_{SN_{ph}} + L_v \cdot *N_{PRNU} + N_{DSNU} + N_{Read}) \quad (4.108)$$

4.5.5.1. Anwendungsbeispiel

Zur Demonstration der Anwendbarkeit der Simulation des Rauschens wurde das Verfahren in einen programmieren Shader implementiert und in Realzeit in der Simulation ausgeführt. Im Rahmen des Anwendungsbeispiels wurde die folgende Parametrierung verwendet, siehe Tabelle 4.7. In Abbildung 4.47

Parameter	Grauwert		Parameter	Grauwert
$k_{0,PRNU}$	0.2		$k_{1,PRNU}$	0.05
$k_{0,SN}$	0.07		$k_{1,SN}$	0.015
$k_{0,DSNU}$	0.013			
$k_{0,Read}$	1.3			

Tabelle 4.7.: Parametrierung des Rauschmodells

ist der Einfluss der Shot-Noise, der PRNU und des Fixed-Pattern Rauschens dargestellt. In der Abbildung wird das Fixed-Pattern Rauschen als Summe aus DSNU und Read-Out Noise definiert. In den oberen Abbildungen ist zu erkennen, dass die Shot-Noise auf das ganze Bild Auswirkungen hat. Die PRNU ist vor allem im Bereich des Himmels sichtbar. Das Fixed-Pattern Rauschen spielt eine untergeordnete Rolle in der Zusammensetzung des Rauschens.

4.5.6. Tiefenschärfe

Wie in Abschnitt 2.2.5 beschrieben weisen reale Kamerasysteme den Effekt der Tiefenschärfe auf. Dieser Effekt führt dazu, dass durch den endlichen Schärfbereich der Kamera ein Punkt aus dem Objektraum auf mehrere Pixel im Bildraum abgebildet wird. Daraus resultiert eine unscharfe Abbildung. Wie zu Beginn dieses Abschnitts hergeleitet wurde, wird in der Computergrafik ein Bild mittels des Lochkamera-Modells erzeugt. Die Abbildung eines Objektpunkts erfolgt durch eine affine Transformation auf die Bildebene der virtuellen Kamera und einer anschließenden Rasterisierung des Bildpunkts. Daher entspricht ein Punkt im Objektraum einem Bildpunkt. In einem synthetischen Bild besitzt deshalb ein weit entferntes Objekt die gleiche Schärfe wie ein Objekt im Fokusbereich der Kamera. Daher ist der Hintergrund eines synthetischen Bildes strukturierter, was die Segmentierung von Objekten erschwert.

Zur Realisierung des Effekts der Tiefenschärfe in der Computergrafik führt Demers [48] eine Einteilung von verschiedenen Ansätzen entsprechend der technischen Realisierung durch:

- Ray-Tracing Ansätze: Strahlen werden ausgehend von der gesamten Fläche der Linse versendet.
- Akkumulation-Buffer Techniken: Die Szene wird mehrfach mit Lochkamera-Modellen aus verschiedenen Positionen berechnet.

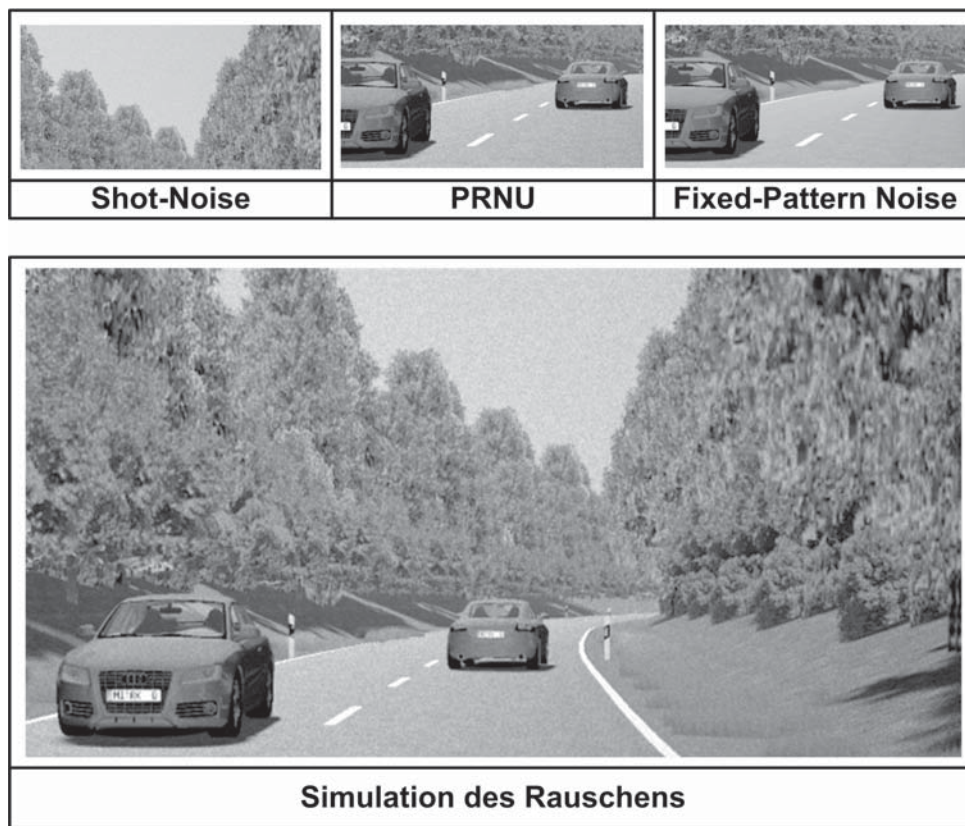


Abbildung 4.47.: Einfluss der Rauschmodelle auf das finale Bild

- Forward-Mapped Z-Buffer-Ansätze: Diese verteilen die ursprüngliche Pixelfarbe entsprechend des Zerstreungskreises auf die Nachbarpixel.
- Backward-Mapped Z-Buffer-Ansatz: Bei diesen wird die Pixelfarbe durch Vermischung mit den Nachbarpixeln des Ursprungsbilds auf Basis des Zerstreungskreises der Kamera realisiert.

In der Echtzeit-Computergrafik sind Variationen der Backward-Mapped Z-Buffer-Technik der am häufigsten angewendete Ansatz zur Simulation der Tiefenschärfe.

Durch Riguer et. al. [156] werden zwei Ansätze zur Simulation des Tiefenschärfe-Effekts vorgeschlagen. Bei dem ersten Ansatz wird die Szene in einem ersten Rendering-Schritt normal gerendert. Weiterhin werden der Tiefenpuffer und ein Unschärfe-Faktor in einen weiteren Puffer gespeichert. Der Unschärfe-Faktor definiert, wie stark ein Pixel verwischt wird. Im nächsten Rendering-Schritt wird das Bild auf Basis des Unschärfe-Faktors, welcher zur Skalierung eines Filter-Kernels dient gefiltert. Der zweite Ansatz basiert auf der Idee die Szene in einem Off-Screen-Buffer zu rendern und die Tiefeninformation für jedes Pixel zu sichern. Als nächstes wird das ursprüngliche Bild auf ein viertel der Größe reduziert und mittels zwei Rendering-Schritten erst in X- und dann in Y-Richtung gefiltert. Abschließend wird mittels Alpha-Blending zwischen dem ursprünglichen und den gefilterten Bild das Bild mit Tiefenschärfe-Effekt berechnet. Die Wahl der Mischanteile basiert auf der Distanz des jeweiligen Pixels zur Brennpunktebene. Durch Zhou et. al. [198] wird ein Verfahren zur Simulation der Tiefenschärfe vorgeschlagen, welches

direkt den Tiefenpuffer der Grafikkarte verwendet, um das in einer Textur gespeicherte Ursprungsbild zu verzeichnen. Zur Gewichtung der Samples führen die Autoren Parameter für die Überlappung-, die Lichtintensität und das Pixel-Bleeding ein. Der erste Parameter beschreibt eine Gewichtung entsprechend des Abstands des Pixels vom Ursprungspixel, der zweite Parameter definiert die Dämpfungsfunktion auf Basis des Zerstreuungskreises. Der Bleeding-Term gewichtet das Pixel entsprechend des Abstands von der Hyperfokalebene. Das Ergebnis der Gewichtung liegt im Intervall von 0 bis 1. Die Pixel werden entsprechend des Gewichtungsfaktors überlagert.

Hammon [82] wählt einen Ansatz, welcher den Zerstreuungskreis für jeden Bildpunkte vor der Hyperfokalebene berechnet und in einer Textur speichert. Für Bildpunkte auf oder nach der Hyperfokalebene ist der Zerstreuungskreis 0. Anschließend wird das Bild mit den Zerstreuungskreisen gefiltert und auf ein Viertel der Größe reduziert. Im nächsten Schritt wird eine Korrektur der Zerstreuungskreise durchgeführt und der größere der beiden Zerstreuungskreise an einer Kante verwendet. Im letzten Schritt wird das vorab berechnete Unschärfefeld auf das Ursprungsbild mit einem variablen Filter angewendet, welcher entsprechend des Zerstreuungskreises skaliert wird. Als Artefakt tritt bei diesem Ansatz das „Einbluten“ von fokussierten in nicht fokussierte Objekte auf. Weiterhin wird die Transparenz nicht explizit behandelt. Alle Variationen führen in der Regel die folgenden Schritte aus:

1. Berechnen und Speichern des unverfälschten RGB-Bilds mit der Lochkamera
2. Berechnen und Speichern eines (linearen) Tiefenbilds mittels der Lochkamera
3. Berechnen von mindestens eines gefilterten Bilds der Lochkamera
4. Berechnen der Zerstreuungskreise auf Basis der Kameraparameter und des Tiefenbilds
5. Blenden oder Vermischen der Bilder in Abhängigkeit des Zerstreuungskreises

Zur Untersuchung der Anwendbarkeit des Tiefenschärfe-Effekts wird der folgende Algorithmus verwendet:

1. Rendern des Kamera- und des Tiefenbilds in eine Textur
2. Vorfilterung des gesamten Bilds mit einem 3x3 Box-Filter
3. Berechnung der Tiefenunschärfe in einem Fragment-Shader

Zur Berechnung der Zerstreuungskreises eines Bildpunkts u' in Bildpunkte wird auf die folgende Gleichung aus [48] zurückgegriffen:

$$u'(a_v) = \left| \frac{d_{EP} \cdot (f \cdot (a_v - a_0))}{(a_v (a_0 - f))} \right| / d_{Pixel} \quad (4.109)$$

Diese verwendet den Tiefenwert des Bildpunkts a_v , den Durchmesser der Eintrittspupille d_{EP} , den Durchmesser eines Pixels d_{Pixel} , die Brennweite f und den Abstand der Hyperfokalebene a_0 . Bei einer Berechnung des Zerstreuungskreises einer Kamera für die Fahrzeuganwendung konnte festgestellt werden, dass der Zerstreuungskreis bis 150 m unter der Breite eines Pixels liegt. Daher wird zur Abbildung einer von der Distanz abhängigen Bildunschärfe die Gleichung wie folgt angepasst:

$$u'(a_v) = \begin{cases} u'(a_v) & a_v < k_1 \\ u'(a_v) + k_0(a_v - k_1) & a_v \geq k_1 \end{cases} \quad (4.110)$$

Nachdem der Zerstreungskreis u' für den Ursprungspixel g_u berechnet wurde, werden maximal neun weitere Samples aus dem Bild bestimmt und gewichtet addiert. Das Ursprungssample besitzt die Gewichtung $w = 1.0$. Damit ergibt sich der Grauwert des Bildpunkts \mathbf{x} zu:

$$g(\mathbf{x}) = \frac{(g_u + \sum w(i, j) \cdot g(i, j))}{\sum w + 1.0} \quad (4.111)$$

Dabei sind i und j die Indizes der Samplingpunkte. Das Gewicht für den Bildpunkt \mathbf{x} ergibt sich mit den Abstand des Samples zum Ursprungspixel $d_{\mathbf{x}(i,j)}$ und dem Stabilisierungsterm δ zu:

$$w = \frac{1}{d_{\mathbf{x}(i,j)}} \cdot \frac{1}{u'(\mathbf{x}) + \delta} \quad (4.112)$$

4.5.6.1. Anwendungsbeispiel

In Abbildung 4.48 sind die Ergebnisse der Tiefenschärfe dargestellt. In der linken oberen Seite des Bilds ist die Szene ohne Tiefenschärfe und auf der rechten Seite mit Tiefenschärfe abgebildet. Im unteren Bereich der Abbildung ist die Differenz der Gradientenbeträge zwischen der Szene mit und ohne Tiefenschärfe für den oberen Teil der Szene visualisiert. Die größeren Differenzen sind weiß dargestellt. Wie ersehen

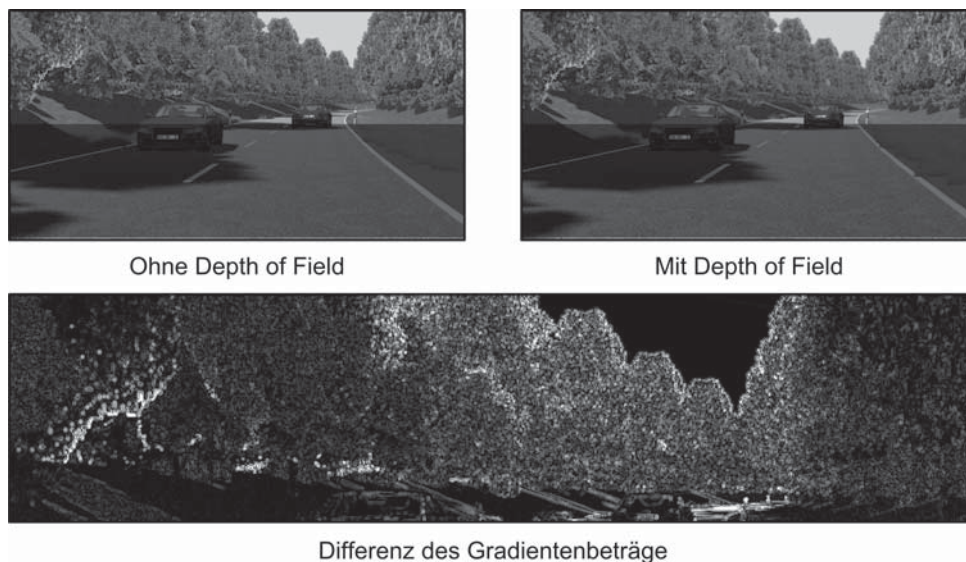


Abbildung 4.48.: Einfluss der Tiefenschärfe auf die Gradientenbeträge im Bild

werden kann erhöht sich für weiter entfernte Bildpunkte die Differenz der Gradienten im Vergleich zum Ursprungsbild. Daher kann festgestellt werden, dass mittels einer Simulation der Tiefenschärfe der Einfluss

von weiter entfernten Bildpunkten auf die Kanten und Texturen im Bild reduziert werden kann. Damit ist es möglich die Realitätsnähe der Simulation zu steigern.

4.5.7. Verzeichnung

Durch den optischen Pfad der Kamera, vgl. Abschnitt 2.2.5, entstehen Verzeichnungen im Bild. Diese Verzeichnungen basieren auf dem Brechungs- und Reflexionsgesetz und sind im Randbereich des Bilds stärker ausgeprägt. Durch diese Abweichungen werden die Strahlen nicht genau in einem Zentrum fokussiert. Daher ist es notwendig Korrekturterme zur Reduktion der Verzeichnungen auf dem Bild anzuwenden, dieses ist in [178], [171], [49], [193] und [130] beschrieben. Diese Terme führen in der Nachverarbeitung eine Verschiebung einzelner Bildpunkte aus. Deshalb ist es notwendig, falls das zu testende Bildverarbeitungssystem eine solche Korrektur durchführt, diese Verzeichnungen in das synthetische Bild einzurechnen. In dieser Arbeit wurde dies mittels einer Inversion der Korrekturterme durchgeführt.

Für die Kompensation einer radialen δ^r und tangentialen δ^t Verzeichnung kann nach [178] die folgende Funktionen verwendet werden. Für einen Bildpunkt \mathbf{x} ergibt sich der reale Bildpunkt \mathbf{x}^{Real} zu:

$$\mathbf{x}^{Real} = \mathbf{x} + \delta^r(\mathbf{x}) + \delta^t(\mathbf{x}) \quad (4.113)$$

mit der symmetrischen radialen Verzeichnung δ^r ergibt sich folgende Gleichung,

$$\delta^r(\mathbf{x}_R) = \begin{pmatrix} x \\ y \end{pmatrix} (\kappa_1 \cdot r^2 + \kappa_2 \cdot r^4 + \dots) \quad (4.114)$$

Nach der Fachliteratur ist es ausreichend die ersten zwei Terme zu betrachten. Der Grad der Verzeichnung kann mittels der Parameter κ bestimmt werden. Wobei $r = \sqrt{(x - x_p)^2 + (y - y_p)^2}$ den euklidischen Abstand des Bildpunkts vom Hauptpunkt \mathbf{x}_p beschreibt. Eine radial-asymmetrische und tangentiale Verzeichnung wird mittels des Terms δ^t definiert:

$$\delta^t(\mathbf{x}_R) = \begin{pmatrix} 2 \cdot \tau_1 \cdot x \cdot y + \tau_2 \cdot (r^2 + 2 \cdot x^2) \\ \tau_1 \cdot (r^2 + 2 \cdot y^2) + 2 \cdot \tau_2 x \cdot y \end{pmatrix} \quad (4.115)$$

Wie bei Betrachtung der Gleichungen feststellbar ist, ist es nicht möglich eine analytische Inversion durchzuführen. Daher wurde ein anderer Ansatz gewählt:

- Berechnung der Pixelposition \mathbf{x}^{Real} für jeden Bildpunkt \mathbf{x} .
- Speichern von \mathbf{x} als Look-Up Table \mathbf{M} an Position von \mathbf{x}^{Real} .
- Durchführung einer Interpolation bis alle Lücken in der Look-Up Table geschlossen sind.
- Kopieren der Ergebnisse in eine Textur und Zuweisung zu einem Shader.

Mittels der gesicherten Koordinaten wird während der Post-Processing Phase die Adressierung des Kamerabilds durchgeführt.

4.5.7.1. Anwendungsbeispiel

Zur Demonstration der Verzeichnungssimulation wurden die Modelle aus [178] in der Simulationsumgebung umgesetzt. In Abbildung 4.49 sind die Ergebnisse dargestellt. Zur Verifikation des Modells wurde das verzeichnete Bild mittels der Korrekturterme in Matlab korrigiert. Abschließend ist die Differenz zwischen dem Ursprungsbild aus der Simulation, ohne Verzeichnung, und dem korrigierten Bild dargestellt. In der ersten Zeile wurde eine kissenförmige und in der zweiten Zeile eine tonnenförmige Verzeichnung simuliert. In der dritten und vierten Zeile sind tangentielle Verzeichnungen mit verschiedenen Parametrierungen dargestellt. Wie in der ersten Spalte der verzeichneten Bildern ersehen werden kann, kommt



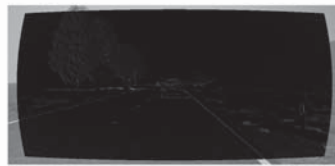









	Simulation Verzeichnung	Korrigiertes Bild	Differenz
Kissenförmige Verzeichnung			
Tonnenförmige Verzeichnung			
Tangentiale Verzeichnung			
Tangentiale Verzeichnung			

Abbildung 4.49.: Auswertung der Verzeichnungssimulation

es durch die Anwendung der Verzeichnung zu einem schwarzen Rand im Bild. Dieses liegt darin begründet, dass der entzerrte Öffnungswinkel der Kamera geringer als der verzerrte Öffnungswinkel ist. Durch die Korrektur des Bilds in Matlab wurden in jedem Bild die Verzeichnungen korrigiert. Bildbereiche außerhalb des verzeichneten Bilds wurden schwarz eingefärbt. Bei der Auswertung der Differenzbilder kann festgestellt werden, dass die Bilder eine sehr hohe Überdeckung mit der unverzeichneten Referenz aufweisen.

4.5.8. Zusammenfassung

In diesem Kapitel wurden Techniken zur Realisierung eines an der Realität angelehnten Bildsyntheseprozesses untersucht. Dafür wurden zu Beginn die Unterschiede zwischen einer realen Bildaufnahme und einer synthetischen Bildgenerierung betrachtet. Die Bildsynthese verwendet in der Echtzeit-Computergrafik eine perspektivische Abbildung mit Rasterisierung. Daher erscheinen die Kanten von Objekten in der Form von Treppen. Zur Lösung dieser Aliasing-Effekte werden verschiedene Filter-Techniken verwendet, welche die Kanten der Objekte beeinflussen. Deshalb wurden zu Beginn verschiedene Filter-Techniken hinsichtlich der Auswirkungen auf das Bild untersucht. Es zeigte sich, dass die Anwendung eines 2x2-Grid-Filters ausreichend ist.

Zur Simulation des Strahlungsaufnehmers einer realen Kamera wurde ein Modell hergeleitet, welches mit Angaben aus einem Datenblatt parametrierbar ist. Das Modell setzt sich aus Funktionen zur Simulation der Bestrahlungsstärke, des Antwortverhaltens, des Farbfilters, der Quantenausbeute, der Belichtungssteuerung und einer Konvertierung der Bestrahlung in Grauwerte zusammen.

Während der Bildaufnahme tritt ein Rauschen in den Pixeln durch vielfältige Quellen auf. Zur Integration dieses Rauschens in den Bildgenerierungsprozess wurde das Modell von [88] in einem Shader auf der Grafikkarte umgesetzt. Die Wirksamkeit des Modells wurde durch eine Veränderung der Belichtungszeit bewiesen.

Durch die Optik tritt der Effekt der Tiefenschärfe auf. Bei dieser werden Punkte von Objekten außerhalb des Schärfebereichs der Kamera auf mehrere Bildpunkte abgebildet. In der Computergrafik tritt dieser Effekt nicht auf. Daher erscheinen Objekte außerhalb des Fokus der Kamera genauso scharf, wie fokussierte Objekte. Der Hintergrund einer Szene ist daher stärker als in der Realität strukturiert. Zur Lösung dieser Schwierigkeit wurde der Tiefenschärfe-Effekt in der Simulation umgesetzt. Die Wirksamkeit konnte durch ein Beispiel nachgewiesen werden.

Der reale optische Pfad einer Kamera führt zu Verzeichnungen im Bild, welche in der Bildvorverarbeitung einer realen Kamera mittels Korrekturtermen entfernt werden. Daher ist es notwendig, diese für eine direkte Einspeisung des simulierten Bilds in den Bildspeicher der Kamera zu integrieren. Als Lösungsmöglichkeit wurde die Inversion der Korrekturterme durchgeführt. Mit diesen Daten wurde das unverzerrte Bild der Lochkamera verzerrt. Die Wirksamkeit dieses Ansatzes wurde durch eine Vor- und Rücktransformation nachgewiesen.

Abschließend kann festgestellt werden, dass es mit überschaubaren Mitteln möglich ist den optischen Pfad und den Strahlungsaufnehmer einer realen Kamera zu simulieren. Weiterhin wurden Verfahren zur Reduktion von Artefakten der Computergrafik untersucht. Die Anwendbarkeit des Kameramodells wurde mittels Werten aus Datenblättern untersucht.

4.6. Aufbau von Szenen und Szenarien

Zur Umsetzung von vergleichenden Tests für die Bewertung der Simulation ist es notwendig eine einfache und effiziente Möglichkeit zur Rekonstruktion von Testfahrten in der Simulation zu schaffen. Im Rahmen dieses Abschnitts wird auf die Erstellung von realitätsnahen Straßenverläufen im OpenDrive-Format basierend auf Navigationsdaten eingegangen [72]. Durch die Abbildung von realen Szenarien in der Simulation kann mittels eines GPS-Sensormodells das reale Navigationsgerät bedatet werden, vgl. 7. Weiterhin wird auf die Rekonstruktion des Verhaltens anderer Verkehrsteilnehmer in der Simulation eingegangen. Abschließend werden Erkenntnisse zur Gestaltung einer möglichst realitätsnahen Szene aus der Diplomarbeit von Naal [117] vorgestellt.

4.6.1. Prozess zur Rekonstruktion von Testfahrten

Im Rahmen der Arbeit wurde ein Verfahren zur Rekonstruktion von Testfahrten für einen vergleichenden Test von Bildverarbeitungssystemen im Software- und Hardware-in-the-Loop Test entwickelt, vgl. [121] [125] [122]. In Abbildung 4.50 sind die wesentlichen Arbeitsschritte dargestellt. Im ersten Schritt werden

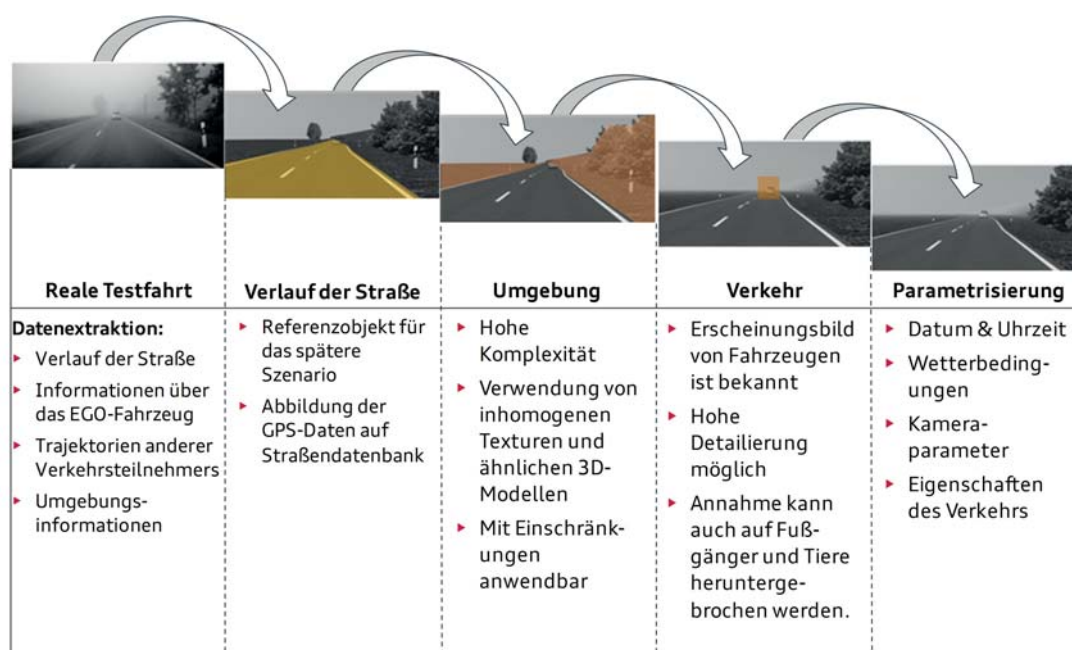


Abbildung 4.50.: Prozess zur Rekonstruktion von Testfahrten

aus Aufzeichnungen realer Testfahrten wesentliche Daten zur Rekonstruktion des Szenarios extrahiert. Darauf folgend wird automatisiert der Verlauf der Straße erzeugt. Als Grundlage kann die Fahrstrecke des Messfahrzeugs (EGO-Fahrzeug) auf eine Straßendatenbank abgebildet werden oder eine Filterung der GPS-Positionsdaten vorgenommen werden. Anschließend werden die Stützstellen in das OpenDRIVE-Format konvertiert. Der Verlauf der Straße dient als Referenzobjekt für die spätere Erstellung der Datenbasis und des Testszenarios.

Für die Erstellung der Umgebung werden ähnliche 3D-Objekte in dem Testszenario platziert und Oberflächen mit ähnlichen Texturen belegt. Durch die hohe Komplexität der Szene ist eine Abbildung der realen Umgebung nur mit Einschränkungen möglich.

Die Positionierung von anderen Verkehrsteilnehmern wird mit Hilfe bei der Extraktion gesammelter Informationen über die Trajektorien der Fahrzeuge durchgeführt. Bei Kraftfahrzeugen kann, wie in Abschnitt 4.4.2 nachgewiesen wurde, eine hohe Abbildungsgenauigkeit erreicht werden. Wenn Fußgänger oder Tiere eine Rolle in dem Szenario spielen sollten, ist dieses nur mit Einschränkungen möglich. Im letzten Schritt gilt es das Szenario zu parametrieren. Das Datum und die Uhrzeit werden anhand des bekannten Datums der Testfahrt festgelegt. Die Parametrierung der Wetterbedingungen und Oberflächeneigenschaften gestaltet sich ungleich schwerer, als Hilfsmittel wird die in Abschnitt 5 beschriebene Metrik verwendet.

4.6.2. Extraktion von Daten aus der realen Testfahrt

Als Grundlage zur Erstellung der Datenbasis und des Szenarios liegen aufgezeichnete Testfahrten vor. Es werden folgenden Daten aus der Aufzeichnung der Testfahrt verwendet:

- Das Bild der Frontkamera des EGO-Fahrzeugs
- Die GPS-Positionsdaten der Referenz-Messtechnik (DGPS)
- Die Geschwindigkeit und Winkelgeschwindigkeiten des Fahrzeugs

Als Vorbereitungsschritt muss jeder relevante Verkehrsteilnehmer im Videobild mit einem Label markiert werden. Diese Markierung dient zur späteren Positionsbestimmung des Verkehrsteilnehmers relativ zum EGO-Fahrzeug. Das Labeln der Messdaten wird häufig durch die Funktionsentwickler für die Auswertung der Bildverarbeitung durchgeführt, so dass kein Mehraufwand entsteht. Zur Extraktion der Daten wurde mit ADTF [164] die in Abbildung 4.51 dargestellte Verarbeitungskette aufgebaut. Im ersten Schritt werden aus den Messdaten die Geschwindigkeit und die Winkelgeschwindigkeiten des EGO-Fahrzeugs extrahiert. Parallel wird die GPS-Position des EGO-Fahrzeugs aus den Messdaten extrahiert und in das UTM-Koordinatensystem transformiert und für die spätere Erstellung des Straßenverlaufs gesichert. Die extrahierten Daten des EGO-Fahrzeugs und die UTM-Koordinaten werden im nächsten Schritt zur Bestimmung der relativen Fahrzeugposition, seit Beginn der Aufzeichnung verwendet. Im letzten Schritt werden mit den Daten des EGO-Fahrzeugs die relativen Abstände zwischen den Verkehrsteilnehmern und dem Messfahrzeug berechnet. Weiterhin werden die Zeitbasen synchronisiert bevor die extrahierten Daten gesichert werden.

4.6.2.1. Bestimmung der Position des EGO-Fahrzeugs

Zur Bestimmung der Position des EGO-Fahrzeugs werden die Längsgeschwindigkeit und die Winkelgeschwindigkeit des Fahrzeugs verwendet. Für die Messdaten gilt die Annahme, dass diese von Messrauschen und Offsetfehlern befreit sind. Weiterhin gilt die Annahme, dass sich das Fahrzeug in Fahrtrichtung in der Spur während des ersten Abtastschritts $k = 0$ befindet und daher der Gier- ψ , Nick- θ und Wankwinkel ϕ im Straßenkoordinatensystem Null sind. Außerdem muss sich das Fahrzeug während der gesamten

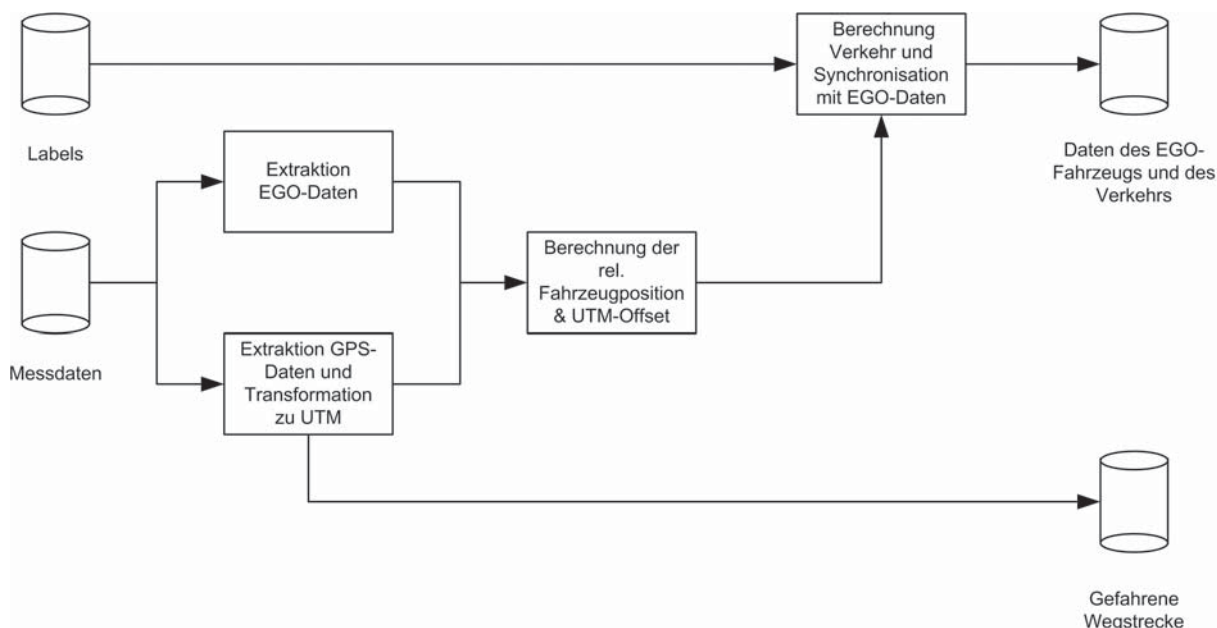


Abbildung 4.51.: Extraktion von Messdaten für die Rekonstruktion der Testfahrt

Testfahrt auf der Straße befinden. Damit folgt für die Fahrzeugdaten zum Zeitpunkt k und der Abtastzeit dt :

$$\psi(k) = \dot{\psi}(k) \cdot dt + \psi(k-1) \quad (4.116)$$

$$\theta(k) = \dot{\theta}(k) \cdot dt + \theta(k-1) \quad (4.117)$$

$$\phi(k) = \dot{\phi}(k) \cdot dt + \phi(k-1) \quad (4.118)$$

$$\dot{\mathbf{x}} = v_{Fzg} \cdot \begin{bmatrix} \cos(\theta(k)) \cdot \cos(\psi(k)) \\ \cos(\theta(k)) \cdot \sin(\psi(k)) \\ -\sin(\theta(k)) \end{bmatrix} \quad (4.119)$$

Für die relative Position des Fahrzeugs seit Beginn der Simulation folgt somit:

$$\mathbf{x} = \dot{\mathbf{x}}(k) \cdot dt + \dot{\mathbf{x}}(k-1) \quad (4.120)$$

Weiterhin wird an dieser Stelle die gemessene GPS-Position in das in der Simulation verwendete Koordinatensystem transformiert, so dass eine Offsetposition ermittelt werden kann. Zur Rekonstruktion

der Bewegung des EGO-Fahrzeugs stehen somit die Geschwindigkeit, Beschleunigung, Wegstrecke und Offset-Position zur Verfügung. Durch die Definition, dass die Trajektorie des EGO-Fahrzeugs dem Straßenverlauf entspricht, beziehen sich die Größen auf die Straßenkoordinaten.

4.6.2.2. Bestimmung der Trajektorie der CO-Fahrzeuge

Die Schätzung einer Trajektorie eines Fahrzeugs aus dem Videobild ist eine komplexe Aufgabe. Durch die Struktur der Simulation kann die Komplexität deutlich reduziert werden. Als Erstes soll die Annahme gelten, dass sich das CO-Fahrzeug während der gesamten Testfahrt auf der Straße befunden hat. Damit muss ausschließlich der Abstand des Objekts zur Kamera betrachtet werden. Weiterhin soll gelten, dass die Dimension eines Labels einer realen Höhe und Breite des Fahrzeugs von $d_{\mathbf{Fzg}} = 1.5m$ entspricht. Diese Annahme ist für PKWs hinreichend genau. Außerdem sind die Objekte in einem Bild kausal mit Label versehen. Durch Anwendung dieser Annahmen kann der Abstand des Objekts zur Kamera x mittels der Dimension eines Labels bestimmt werden [125]:

$$x = \sqrt{\left(\left(x_{Label} - \frac{d_{Bild,x}}{2} \right) \cdot \frac{\frac{d_{Bild,x}}{d_{Label,x}} \cdot d_{Fzg,x}}{d_{Bild,x}} \right)^2 + \left(\frac{d_{Bild,x}}{d_{Label,x}} \cdot d_{Fzg,x} \cdot \frac{z_{NearClip}}{d_{NearClip,x}} \right)^2} \quad (4.121)$$

Die Größen $d_{Label,x}$ und $d_{Bild,x}$ beschreiben die Breite des Labels bzw. des Bilds in *Pixel* und $d_{Fzg,x}$ die Breite des Fahrzeugs in *m*. Der Parameter x_{Label} definiert das Zentrum eines Labels. Die Größe $z_{NearClip}$ beschreibt den Abstand des NearClips vom Imager und $d_{NearClip,x}$ die Breite der NearClipping-Ebene. Zur Stabilisierung der Schätzung wird jedes Label in ein Objekt konvertiert, d.h. für jedes Objekt im aktuellen Frame wird nach einem korrespondieren im Objekt-Label im vorherigen Frame des Videos der Messfahrt gesucht. Zur Ermittlung der Ähnlichkeit, wird für alle Objekte des vorherigen Frames das Pascal-Maß [64] mit den Labels des aktuellen Frames bestimmt. Das Label mit der höchsten Score, unter der Bedingung das die $Score > 0.3$, wird dem aktuell geprüften Objekt hinzugefügt. Falls für ein Label kein bestehendes Objekt gefunden wird, wird ein neues Objekt erstellt. Alte Objekte ohne einen Nachfolger werden gelöscht.

4.6.2.3. Anwendungsbeispiel

In Rahmen der Veröffentlichung [121] wurden aus der in Abbildung 4.52 dargestellten Testfahrt die Daten für das EGO-Fahrzeug und der Abstand vom CO-Fahrzeug zum EGO-Fahrzeug ermittelt, siehe Abbildung 4.53. Bei dieser Testfahrt schließt das Messfahrzeug auf einen vorausfahrenden PKW auf. Die extrahierten Daten für die Geschwindigkeit des EGO-Fahrzeugs und dem Abstand zum CO-Fahrzeug wurden in Matlab mittels Regressions-Polynomen gefiltert. Aus diesen Polynomen wurden anschließend Stützstellen für jede Sekunde des Testlaufs berechnet. Diese Stützstellen beinhalten die Geschwindigkeit des EGO-Fahrzeugs und den Abstand des vorausfahrenden Fahrzeugs zum EGO-Fahrzeug. Anhand dieser Daten wird der Ablauf der Simulation im Testlauf gesteuert, dazu wird die Geschwindigkeit des EGO-Fahrzeugs und der Sollabstand des CO-Fahrzeugs zum EGO-Fahrzeug geregelt.



Abbildung 4.52.: Szenario: Messfahrzeug schließt zu EGO-Fahrzeug auf

4.6.3. Verfahren zur Generierung von Straßenverläufen

Zur Durchführung von vergleichenden Tests mit einer realen Fahrscene und für die Erzeugung von prädiktiven Streckendaten ist es notwendig reale Straßenverläufe in der Simulation abzubilden. In Abbildung 4.54 ist das von Nentwig und Ganslmeier entwickelte Verfahren [123] zur Erzeugung von Straßenverläufen dargestellt. Als Grundlage für die Erstellung der Straßenverläufe dienen 3D-Geländedaten und Navigationsdaten aus einer Datenbank. Aus den Navigationsdaten werden Informationen über den Verlauf der Straße und die Attribute der Fahrspuren, z.B. Geschwindigkeitsbegrenzungen, entnommen. Im Vergleich zu anderen Interpolationsverfahren [40] wird die Trajektorie durch Basiselemente des Straßenbaus beschrieben und ist damit zum OpenDRIVE-Standard konform.

4.6.3.1. Algorithmus zur Abbildung eines Straßenverlaufs

Das Verfahren bildet eine geordnete Punktfolge auf geometrische Elemente des Straßenbaus ab. Als Elemente für die Approximation des Straßenverlaufs werden Geraden und Kreisbögen verwendet.

In einem ersten Schritt ist es notwendig die Stützstellen der Navigationsdaten aus dem WGS84- Koordinatensystem [74] in ein kartesisches Koordinatensystem zu transformieren. In der verwendeten Im-

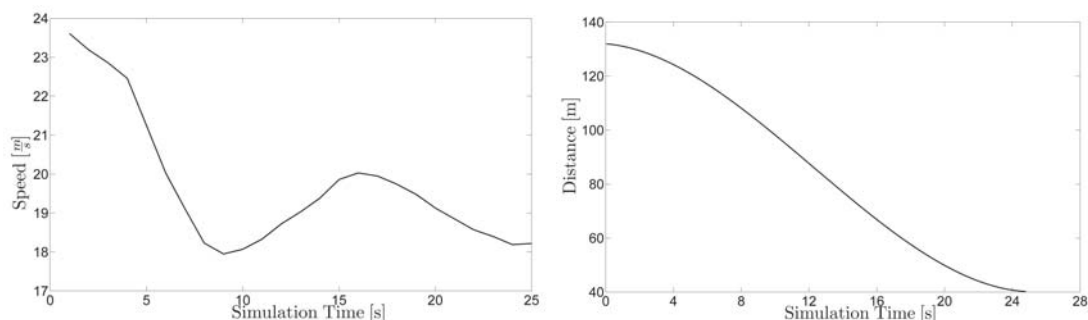


Abbildung 4.53.: Geschätzte Geschwindigkeit des EGO-Fahrzeugs (links) und Abstand zum CO-Fahrzeug (rechts) über die Simulationszeit

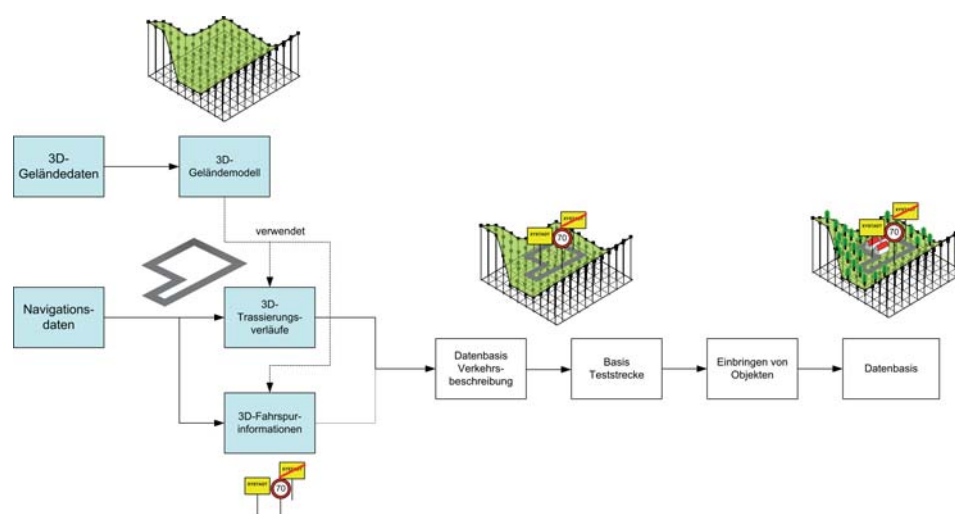


Abbildung 4.54.: Prozess zur Abbildung von realen Straßenverläufen in der Simulation

plementierung wird eine Transformation in das UTM- Koordinatensystem [2] nach [165] vorgenommen. Die North- und Eastern-Achsen innerhalb einer UTM-Zone entsprechen der x- und y-Achse eines kartesischen Koordinatensystems. Zur Reduktion der Kartengröße wird im nächsten Schritt eine Anpassung der North- und Eastern-Koordinaten vorgenommen. Dafür wird der Koordinatenursprung aller Punkte relativ zur kleinsten North- und Eastern-Koordinate verschoben.

Anschließend wird mittels des in [123] beschriebenen Algorithmus der Verlauf der Straße berechnet. Dafür wird die Annahme getroffen, dass eine Folge von drei Punkten durch zwei Geraden und einen Kreisbogen verbunden ist. Der Ablauf ist in Abbildung 4.55 veranschaulicht. Der Ablauf wird rekursiv über die gesamte Menge der Stützstellen angewendet. Zur Konstruktion der Geometrie wird im ersten Schritt ein gemeinsamer Schwerpunkt für alle drei Punkte bestimmt. Dazu wird die Annahme getroffen, dass sich die Punkte eines Kreisbogens auf einem Kreis mit gleichem Radius befinden. Daher ist es notwendig das durch die drei Punkte aufgespannte Dreieck in ein gleichschenkliges Dreieck zu transformieren. Dazu werden die Strecken $\overline{p_2p_1}$ und $\overline{p_2p_3}$ bestimmt und die Länge berechnet:

$$\overline{p_2p_1} = \|p_2 - p_1\| \quad (4.122)$$

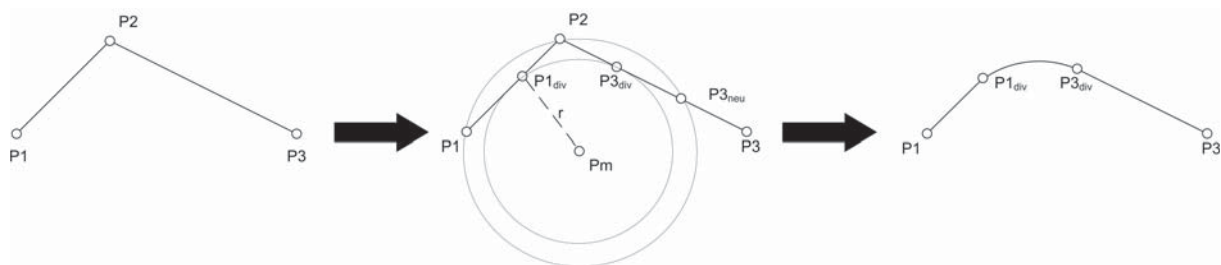


Abbildung 4.55.: Ablauf der Segmenterzeugung

und

$$\overline{p_2 p_3} = \|p_2 - p_3\| \quad (4.123)$$

mit den normalisierten Richtungsvektoren:

$$dp_2 p_1 = \frac{p_2 - p_1}{\overline{p_2 p_1}} \quad (4.124)$$

und

$$dp_2 p_3 = \frac{p_2 - p_3}{\overline{p_2 p_3}} \quad (4.125)$$

Anschließend werden neue Punkte $p_{1,neu}$ und $p_{3,neu}$ des gleichschenkligen Dreiecks bestimmt:

$$p_{1,neu} = p_2 - dp_2 p_1 \cdot \min(\overline{p_2 p_1}, \overline{p_2 p_3}) \quad (4.126)$$

und

$$p_{3,neu} = p_2 - dp_2 p_3 \cdot \min(\overline{p_2 p_1}, \overline{p_2 p_3}) \quad (4.127)$$

Danach werden mittels der Richtungsvektoren der Strecken $\overline{p_2 p_1}$ und $\overline{p_2 p_3}$ neue Punkte mit der Entfernung der kürzesten Streckenlänge von p_2 platziert. Der Mittelpunkt des Kreises $p_m(x_m, y_m)$ und der Radius r kann mit der Gleichung für einen verschoben Kreis bestimmt werden:

$$(x - x_m)^2 + (y - y_m)^2 = r^2 \quad (4.128)$$

Durch Auflösen und Umstellen der Gleichung folgt:

$$x_m^2 + y_m^2 - r^2 - 2x_m x - 2y_m y = -(x^2 + y^2) \quad (4.129)$$

Da für alle drei Punkte des Dreiecks $p_{1,neu}$, p_2 und $p_{3,neu}$ die Annahme gilt, dass diese auf einem Kreis liegen, können die Parameter durch Lösen des folgenden Gleichungssystem bestimmt werden:

$$\mathbf{A} = \begin{bmatrix} 1 - x_{p_{1,neu}} - y_{p_{1,neu}} \\ 1 - x_{p_2} - y_{p_2} \\ 1 - x_{p_{3,neu}} - y_{p_{3,neu}} \end{bmatrix} \quad (4.130)$$

und

$$\mathbf{b} = \begin{bmatrix} -(x_{p_{1,neu}}^2 - y_{p_{1,neu}}^2) \\ -(x_{p_2}^2 - y_{p_2}^2) \\ -(x_{p_{3,neu}}^2 - y_{p_{3,neu}}^2) \end{bmatrix} \quad (4.131)$$

Damit ergibt sich für den Kreismittelpunkt:

$$p_m = (x(2)/2, x(2)/2) \quad (4.132)$$

Im letzten Schritt wird der Anteil des Kreises an dem gesamten Streckensegment bestimmt. Dazu wird ein Parameter für die Rundheit $k_{curve} < 0.5$ des Segments eingefügt, mit Hilfe des Parameters wird der Beitrag des Kreises skaliert. Damit ergeben sich die folgenden Punkte zu Beschreibung des Segments:

$$p_{1,div} = p_2 - k_{curve} \cdot dp_2 p_1 \cdot \min(\overline{p_2 p_1}, \overline{p_2 p_3}) \quad (4.133)$$

$$p_{3,div} = p_2 - k_{curve} \cdot dp_3 p_2 \cdot \min(\overline{p_2 p_1}, \overline{p_2 p_3}) \quad (4.134)$$

Für den Radius des Kreissegments folgt damit,

$$r_{Curve} = \sqrt{(x_{p_{1,div}} - x_{p_m})^2 + (y_{p_{1,div}} - y_{p_m})^2} \quad (4.135)$$

Mit dem Radius des Kreisbogens und von den Strecken $\overline{p_2 p_{3,div}}$ und $\overline{p_2 p_{1,div}}$ aufgespannten Winkel α [rad] ist es möglich die Länge des Kreissegments l_{Curve} zu berechnen:

$$l_{Curve} = r_{Curve} \cdot \alpha \quad (4.136)$$

Die Bestimmung der Richtungsvektoren am Beginn des Segments für die Berechnung der Ausrichtung der Segmente muss entsprechend des OpenDRIVE-Standards [136] erfolgen. Das Segment kann somit durch drei geometrische Elemente beschrieben werden:

$$Seg_{Line} = Linie(p_1, p_{1,div}) \quad (4.137)$$

$$Seg_{Curve} = Kreis(p_{1,div}, p_{3,div}, r_{Curve}, l_{Curve}) \quad (4.138)$$

$$Seg_{Line} = Linie(p_{3,div}, p_{1,div}) \quad (4.139)$$

4.6.3.2. Anwendungsbeispiel

In [72] wurde die Genauigkeit des Algorithmus gegenüber realen GPS-Messungen betrachtet, vgl. Abbildung 4.56. Die Genauigkeit der generierten Straße hängt nach den durchgeführten Untersuchungen

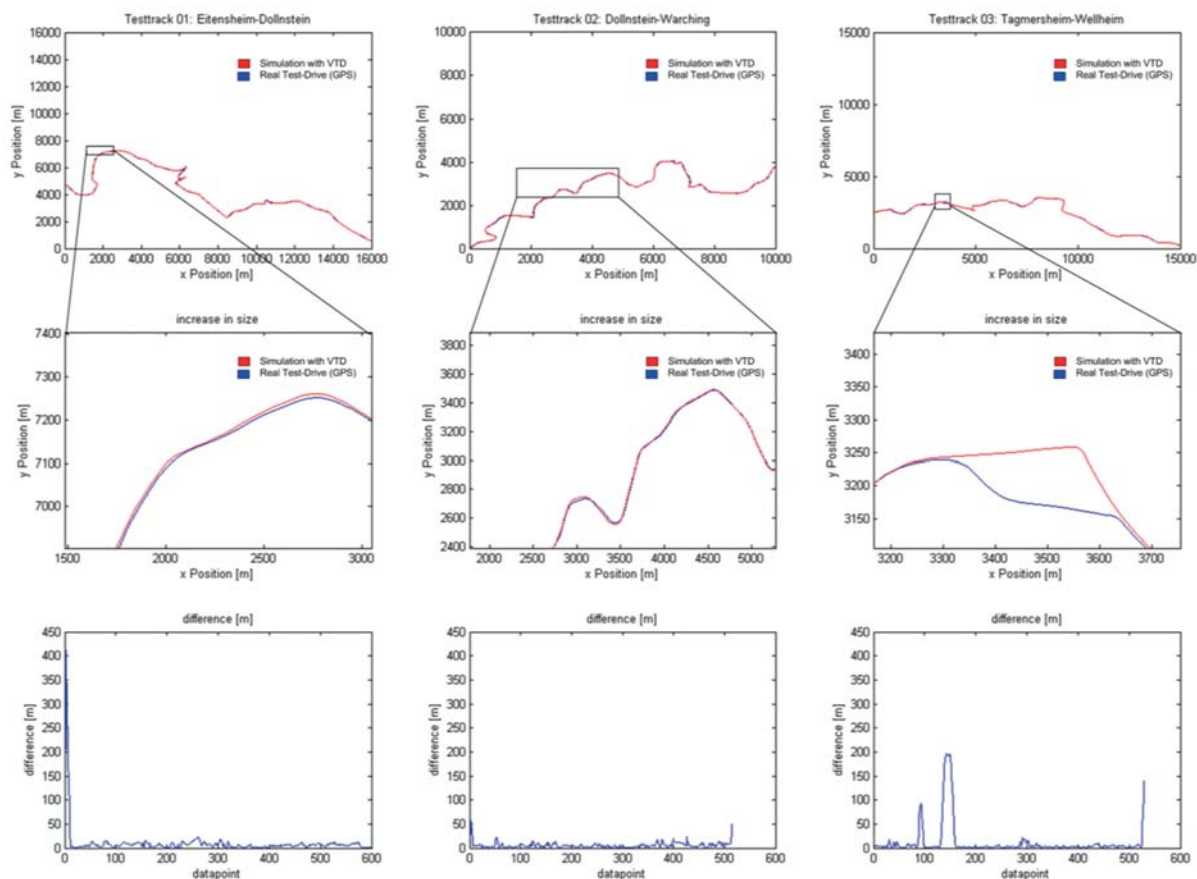


Abbildung 4.56.: Vergleich der Simulation mit realen Messdaten, aus [72]

davon ab, wie gut die Qualität der vorliegenden Navigationsdatenbank ist. Als Ergebnis der Tests kann festgehalten werden, dass die Abweichung zwischen der Position in der Simulation und Realität kleiner als $e = 20m$ ist. Die mittlere Abweichung zwischen der Simulation und der realen Szene beträgt $e = 6.072m$ mit einer Standardabweichung von $\sigma = 4.42m$. Die Ungenauigkeit ist in erster Linie der fehlerhaften GPS-Messungen während der realen Testfahrt und der Qualität der Navigationsdatenbank geschuldet. Die übliche Abweichung einer GPS-Messung beträgt in der Regel 20 m. Aus Sicht der gewünschten Anwendung ist eine Abweichung von 20 m akzeptabel.

4.6.4. Gestaltung der Szene

Durch Naal wurden in seiner Diplomarbeit [117] Erkenntnisse über die Qualität der Randbebauung für die Entwicklung von Bildverarbeitungssystemen auf Basis von synthetischen Bilddaten gewonnen. Im Kern der Arbeit wurde ein Haar-Wavelet [185] basierter Fahrzeugklassifikator mit synthetischen Heckansichten von Fahrzeugen entwickelt und die Leistungswerte mit einem Fahrzeugerkennungssystem für den Serieneinsatz verglichen. Die für das Training des Klassifikators verwendeten Datensätze an Heckansichten wurden automatisch aus der Simulation erzeugt. Nachdem das Training mit dem aktuellen Datensatz abgeschlossen war, wurde die Leistungsfähigkeit des Klassifikators in mehr als 300 realen Sequenzen geprüft. Basierend auf den Erkenntnissen der Validierung wurde das Szenario-Design in der virtuellen Welt angepasst und ein neuer Datensatz erzeugt welcher erneut validiert wurde.

4.6.4.1. Ergebnisse

Die Auswirkungen des Szenario-Designs auf die Erkennungsrate der Klassifikators werden anhand von fünf Datensätzen beschrieben. In Abbildung 4.57 sind die Datensätze und die zugehörigen Ergebnisse im Vergleich dargestellt. Im ersten Datensatz wurden die Heckansichten auf einer geraden Straße im freien

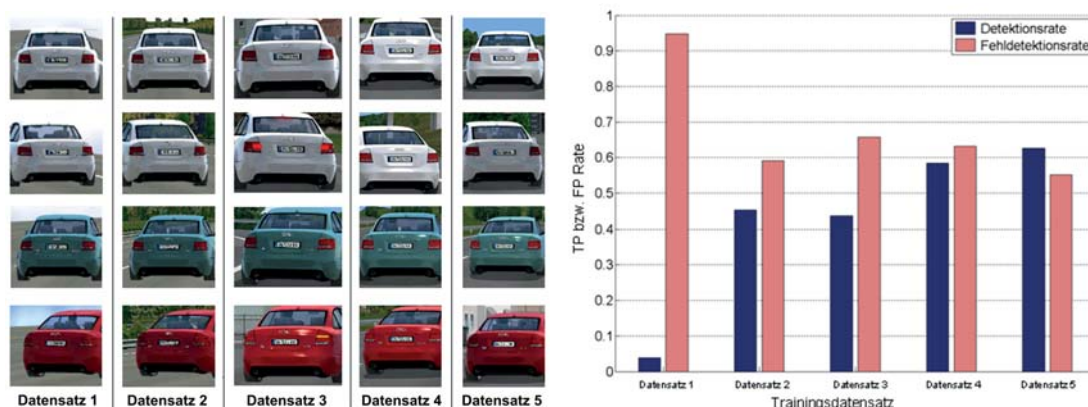


Abbildung 4.57.: Die Trainingsdatensätze und Ergebnisse im Vergleich [117]

Raum erzeugt. Daher hatte der Hintergrund eine homogene Farbe und es entstand in jedem Bild eine harte Kante zwischen der Straße und dem Horizont. Diese Kante führte beim Training des Klassifikators dazu, dass diese als Merkmal mit hohem Gewicht in die Klassifikatorkaskade aufgenommen wurde. In den realen Videos zur Bewertung des Klassifikators wurden eine Vielzahl von Fahrzeugen nicht als solche erkannt. Als Grund wurde festgestellt, dass sich nichterkannten Fahrzeuge meist vor einem strukturierten Hintergrund befanden. Daher lag keine harte Kante zwischen der Straße und dem Horizont vor.

Basierend auf diesen Erkenntnissen wurde im zweiten Datensatz die Strecke aus dem ersten Datensatz um einen strukturierten Hintergrund bestehend aus Wald erweitert. Durch das Hinzufügen des strukturierten Hintergrunds konnte die Detektionsrate des Fahrzeugklassifikators verbessert werden. Trotzdem wiesen die Trainingsdaten nach Naal [117] eine zu geringe Varianz auf, da die Heckansichten der Fahrzeuge limitiert waren und die Straße einen geraden Verlauf aufwies.

Der dritte Datensatz wurde in einem komplexen Szenario erzeugt, welches verschiedene Umgebungen, Autobahnen, Landstraßen, Ortschaften, Kurven und Kreuzungen, beinhaltete. Die Vergrößerung der Varianz und der Heckansichten resultierte in einer leicht geringeren Erkennungsrate. Naal führt als Erklärung an, dass es durch viele gleiche Merkmale in den Heckansichten es zu einem Over-Fitting des Klassifikators kam da und in einer realen Szene eine größere Varianz dieser Merkmale vorliegt.

Als Schlussfolgerung wurden im nächsten Datensatz diese Merkmale abgeschwächt. Die Label zur Erzeugung der Heckansichten wurden mit einer Varianz in der Position erzeugt. Daher lagen die Kanten des Fahrzeug-Hecks nicht immer an der gleichen Position im Bild. Damit konnte ein Over-Fitting des Klassifikators vermieden werden. Durch diese Änderung konnte die Erkennungsrate des Klassifikators deutlich verbessert und eine Reduktion der Fehlerkennungen erreicht werden. Im fünften Datensatz wurde die Varianz der Trainingsdaten weiter erhöht in dem die Position der virtuellen Kamera anders positioniert wurde. Durch diese Änderung konnte die Leistungsfähigkeit des Klassifikators weiter gesteigert werden.

4.6.4.2. Schlussfolgerung

Als Ergebnis der Untersuchungen von Naal [117] kann festgehalten werden, dass bei der Durchführung einer virtuellen Entwicklung von Bildverarbeitungssystemen eine hohe Varianz in der Randbebauung einen positiven Einfluss auf die Erkennungsgenauigkeit des Systems hatte. Durch diese kann der Ada-Boost-Algorithmus [185] die falschen von guten Merkmalen segmentierten und somit die Gewichte entsprechend adaptieren. Ein ähnlicher Einfluss konnte durch die Anpassung der Labelposition und die Variation der Kameraeinstellungen hergestellt werden. Daher ist bei der Erstellung von Testszenarien für die Erprobung von Objekterkennungssystemen auf eine Variation im Hintergrund zu achten. Für die Durchführung von Robustheitstests kann ein Szenario ohne Hintergrund verwendet werden.

4.6.5. Zusammenfassung

In diesem Abschnitt wurde der angewendete Prozess zur Erstellung von Szenarien für die Erprobung von kameraunterstützten Fahrerassistenzsystemen vorgestellt. Dieser sieht es vor für die Validierung der Simulation reale Szenen in der Simulation abzubilden. Daher wurde in einem ersten Schritt ein Verfahren zur Extraktion von Daten des Testfahrzeugs und weiterer Fahrzeuge aus Messdaten entwickelt. Die Wirksamkeit des Verfahrens wurde im Rahmen verschiedener Veröffentlichungen nachgewiesen. Als weiterer Bestandteil wurde eine Möglichkeit zur Abbildung von realen Straßenverläufen in der Simulation geschaffen [123]. Bei dem Prozess zu Erstellung des Szenarios dient die Straße als Referenz. Durch den realen Straßenverlauf ist es möglich Navigationsdaten zu erzeugen.

Zur Bewertung der Auswirkungen hinsichtlich der Gestaltung von Simulationsszenarien wurden durch Naal [117] verschiedene Komplexitätsgrade untersucht. Als Ergebnis wurde festgestellt, dass eine hohe Varianz in der Randbebauung einen positiven Einfluss auf die Erkennungsgenauigkeit eines Fahrzeugerkennungssystems hatte. Bei der Erstellung von Testszenarien für die Erprobung muss daher auf eine Variation im Hintergrund geachtet werden. Ein monotoner Hintergrund kann umgekehrt als Szenario für einen Robustheitstest verwendet werden.

4.7. Zusammenfassung

In diesem Kapitel wurde eine Anforderungsanalyse an die Simulation durchgeführt. Diese definiert Anforderungen an die Simulation der Beleuchtung, Geometrie und Oberflächeneigenschaften von Objekten, Synthese von Kamerabildern und den Aufbau der Szene.

Bei der Analyse von Methoden zur Beleuchtungssimulation konnte festgestellt werden, dass für grundlegende Tests zur Erkennung von Fahrzeugen in Bodennähe eine einfache Schattensimulation ausreichend ist. Für erweiterte Tests müssen weiche Schatten in die Simulation integriert werden. Mit diesen können Schatten von weiter über den Boden entfernten Objekte dargestellt werden. Die Simulation der Sonnenstrahlung im RGB-Farbraum und Position ermöglicht eine einfache Parametrierung der Szene. Die Parametrierung der atmosphärischen Simulation erweist sich als komplex. Mittels des umgesetzten Modells können verschiedene Witterungsbedingungen dargestellt werden.

Im zweiten Abschnitt konnte festgestellt werden, dass die Geometrie von Fahrzeugen, der Straße und Verkehrsschildern im Rahmen der Untersuchungen eine ausreichende Genauigkeit aufwies. Bei einem Vergleich der Oberflächenmodelle wurde festgestellt, dass das Blinn-Phong-BRDF eine für viele Tests ausreichende Qualität aufweist. Zur Realisierung von erweiterten Tests kann auf komplexere Modelle wie die Ward-BRDF zurückgegriffen werden. Die Simulation einer nassen Straßenoberfläche erwies sich als komplexer Prozess, in welchem Abweichungen zur Realität auftraten. Bei einer Verwendung von Wettereffekten in der Simulation muss daher auf eine Validierung der Parametrierung des Modells geachtet werden.

Die Generierung eines möglichst realitätsnahen Kamerabilds wurde im dritten Abschnitt bearbeitet. Dazu wurde an erster Stelle der Bildsyntheseprozess betrachtet. Da in der Computergrafik Kanten als Treppen erscheinen wurden Verfahren zur Reduktion dieser Aliasing-Effekte betrachtet. Bei dieser Analyse zeigte sich, dass ein einfacher 2x2 Grid-Filter zur Reduktion der Artefakte ausreicht.

Da bei der Rasterisierung immer ein Objektpunkt auf einen Bildpunkt abgebildet wird, besitzt der Hintergrund eine ungewöhnlich starke Struktur. Zur Lösung dieser Problematik wurde eine Simulation der Tiefenschärfe integriert. Wenn Bilddaten digital die zu testende Kamera eingespeist werden, kann es notwendig sein optischen Verzeichnungen in die Simulation zu integrieren und einen Bildsensor mit Rauschen zu simulieren. Zur Lösung diese Herausforderung wurde ein mittels Datenblätter parametrierbares Kameramodell untersucht.

Im letzten Teil der Untersuchung wurde der Aufbau der Szenen und Szenarien betrachtet. Für die Umsetzung von realen Testfahrten in der Simulation wurde ein semi-automatischer Prozess entwickelt. Die rekonstruierten Szenarien wurden zur Validierung der Simulation eingesetzt. Im Rahmen von zwei Beispielen wurde die Rekonstruktion von Fahrzeugbewegungen und des Straßenverlaufs betrachtet. Weiterhin konnte durch die Untersuchungen von Naal [117] nachgewiesen werden, dass eine hohe Variation in der Randbebauung der Szene notwendig ist, um auf die Realität übertragbare Ergebnisse zu erhalten.

5. Entwicklung eines generischen Verfahrens zur Validierung und Kalibrierung der Simulation

Ziel 4: Entwicklung eines generischen Verfahrens zur Validierung und Kalibrierung der Simulation.

Zur Durchführung von simulationsbasierten Studien ist es notwendig die Genauigkeit der verwendeten Modelle im definierten Anwendungsbereich zu validieren. Ein häufig angewendetes Verfahren zur Validierung ist ein Abgleich des Modellverhaltens mittels simulierter und realer Messdaten [198] [105] [128] [138]. Ein guter Überblick über die Verfahren kann in [22] und [133] vorgefunden werden. Weiterhin ist es im Vorfeld notwendig eine adäquate Parametrierung der Modelle zu finden, deshalb wird häufig eine Kalibrierung anhand von realen Daten durchgeführt. In der Computergrafik werden häufig Metriken oder Maße zur Steuerung des Bildgenerierungsprozess bei strahlenbasierten Verfahren verwendet [27] [28] [149] [66]. In diesem Kapitel wird ein Verfahren zur Validierung und Kalibrierung einer Kamerasimulation nach [124] beschrieben.

5.1. Stand der Technik

In den letzten Jahrzehnten wurde eine Vielzahl von Verfahren zur Validierung von synthetischen Bilddaten für den menschlichen Betrachter entwickelt. Die wichtigsten Verfahren, welche einen Referenzabgleich mit einem Beispielbild durchführen, werden im Folgenden beschrieben.

Durch Baddeley [20] wird eine Metrik zur Erkennung von Unterschieden zwischen binären Bildern vorgeschlagen. Diese soll zur Bewertung der Leistungsfähigkeit von Bildverarbeitungsalgorithmen, wie z.B. Kantenerkennung, dienen. Die Metrik definiert sich aus der Differenz zwischen begrenzten Distanztransformationen zweier Bilder \mathbf{A} (Referenz) und \mathbf{B} (Schätzung). Das Maß beschreibt die Häufigkeit einer fehlerhaften Klassifikation einzelner Pixel. Zur Bestimmung der Distanz wird die Hausdorff-Distanz $H(\mathbf{A}, \mathbf{B})$ verwendet. Diese ist über den Abstand eines Elements aus einer Menge zum nächsten Element einer anderen Menge definiert. Nach [20] ist dieses Maß sehr rauschempfindlich. Daher wird eine Mittelwertbildung über alle Distanzen, anstelle der Verwendung des Maximalwerts im Bild, durchgeführt. Durch Wilson [195] wurde die Metrik auf Bilder mit Grauwerten erweitert.

Eine weitere Anwendung der Hausdorff-Distanz wird durch Huttenlocher [86] vorgeschlagen. Die Hausdorff-Distanz wird verwendet um ein Modell (Template) in einem Binärbild aufzufinden. Zur Durchführung des Vergleichs wird um den Ursprungspunkt eine Schwankungsbreite eingeführt, in deren Bereich nach dem Objekt gesucht wird. Das Suchfenster mit dem Template wird dazu über das Bild verschoben (Sliding-Window). Die kleinste Distanz wird als die wahrscheinlichste Position für das gesuchte Objekt

im Bild erachtet. Weiterhin wird durch eine Skalierung oder eine Rotation des Templates versucht die Distanz weiter zu minimieren. Das Verfahren wird am Beispiel des Findens von Strukturen in Binärbildern dargestellt. Die Methode erweist sich als sehr tolerant gegenüber kleinen Positionsfehlern.

Durch Rushmeier et. al. [161] wurden verschiedene Metriken für den Vergleich von synthetischen mit realen Bildern, hinsichtlich photometrischer Ähnlichkeit, in einer genau definierten Laborumgebung untersucht. Ziel des Vergleichs soll die Bewertung von Simulationssoftware für die globale Beleuchtung sein. Durch die Autoren wurde weiterhin der generelle Aufbau einer Metrik betrachtet. Nach [161] sollte sich eine Metrik für das menschliche Sehen nur auf Helligkeitsunterschiede, anstelle von absoluten Differenzen, beziehen. Durch die Autoren wurden drei Metriken betrachtet, eine frequenzbasierte MSE-Metrik, eine frequenzbasierte Metrik und eine auf den Visual-Difference-Predictor basierte Metrik [47]. Vor der Berechnung wird zuerst eine Normalisierung des Bilds hinsichtlich der Helligkeit durchgeführt. Bei der ersten Metrik wird die Nichtlinearität der menschlichen Wahrnehmung durch die Quadratwurzel der normalisierten Helligkeiten realisiert. Danach wird eine Fast-Fourier-Transformation (FFT) [35] und der Mean-Squared-Error (MSE) [92] zwischen den Referenz- und Ist-Bild bestimmt. Im zweiten Ansatz wird auf die normalisierten Helligkeiten direkt eine FFT angewendet, eine Amplituden-Matrix gebildet und danach eine anisotropische Filterung durchgeführt. Zur Differenzbildung wird der MSE berechnet. Bei der dritten Metrik wird der erste Teil des Visual-Difference-Predictor angewendet, welcher nicht die Wahrnehmung des Menschen modelliert. Dabei wird jede Helligkeit zu einer nichtlinearen Amplitude transformiert und eine FFT durchgeführt. Anschließend wird das Bild mit einer kontrastbasierten Antwortfunktion gefiltert, welche die menschliche Wahrnehmung simuliert. Als Ergebnis wird der MSE zwischen den Amplituden beider Bilder bestimmt. Als Referenz-Maß für den Vergleich von Bildinhalten wird direkt der Mean-Squared-Error zwischen dem Referenz-Bild und der simulierten Szene bestimmt. Bei dem Experiment schneidet die direkte Anwendung des Mean-Squared-Errors am schlechtesten ab. Als Gründe dafür sind geometrische Abweichungen und die Positionen der Lichter zwischen Simulation und Realität zu sehen. Daher schlagen die Autoren einen Vergleich der Erscheinung, anstelle eines direkten pixelbasierten Vergleichs vor.

In [140] wird ein auf Histogrammen basierendes Verfahren für den Vergleich von Bildinhalten verwendet. Histogramme weisen generell eine gute Robustheit gegenüber leichten Abweichungen in der Szene auf, beinhalten aber keine räumlichen Informationen. Da Bilder mit unterschiedlichen Inhalten ein gleiches Histogramm besitzen können, wird die Bildung eines Histogramms über Farb-Kohärenz-Vektoren vorgeschlagen. Diese definieren zu welchem Maß ein Pixel zu einer geschlossenen Farbregion zugehörig ist. Zur Bestimmung der Ähnlichkeit wird die Differenz der Histogramme zwischen Messung und Referenz gebildet. Das Verfahren ist flächenbasiert und ist auf das gesamte Bild oder einzelne Regionen anwendbar. Durch Pass und Zabih wurden in [139] Verbund-Histogramme verwendet, um Bilder mit bestimmten Inhalt aus einer Datenbank anhand einer Referenz zu finden. Diese Histogramme beinhalten mehrere lokale Bildmerkmale: Farbe, Dichte der Kanten, Texturierung, Gradientenbetrag und Rank der Pixel. Zur Berechnung wird das Bild in fünf sich überschneidende Regionen zerteilt und es werden Momente auf Basis der Farbverteilung im Bild berechnet. Weiterhin werden die Regionen gewichtet. Die Regionen am Rand haben weniger Gewicht als im Zentrum des Bilds. Die Autoren begründen dies damit, dass sich die wichtigen Inhalte häufig im Zentrum des Bilds befinden. Die Distanz zwischen zwei Bildern ist die Summe aus der Distanz der Zentrumsregionen und der Region im Randbereich mit der kleinsten Distanz. Nach [139] reduzieren die Verbund-Histogramme die Wahrscheinlichkeit für einen Fehlvergleich.

In [162] schlagen die Autoren eine quantitative ortsbezogene Metrik vor, welche sich aus mehreren Einzelmaßen zusammensetzt. Die Metrik soll die Ungleichheit zwischen verschiedenen Bildern bestimmen. Die Anwendung des Verfahrens soll dem Anwender helfen Unterschiede im Modell aufzuzeigen und eine einfache Bewertung der Modellgüte erlauben. Die Autoren begründen die Verwendung von mehreren Teilmaßen damit, dass kaum ein einzelnes numerisches Maß robust gegenüber allen Fehlern ist. Daher wird ein Satz von Teilmetriken bereitgestellt, welche bestimmte Abweichungen aufzeigen. Als erstes wird ein globaler RMSE (Root-Mean-Squared-Error) [92] zwischen Bildpunkten der Messung und Referenz bestimmt. Im zweiten Teilmaß werden Bildregionen verglichen. Der Beitrag der Region zur Differenz wird durch die Größe der Region gewichtet. Zur Bestimmung des Maßes für hochfrequente Regionen wird eine Filterung mit einer Aktivitätsmaske, welche Kanten detektiert, durchgeführt. Der Fehler wird anschließend in den angrenzenden Pixeln der Kante bestimmt. Zur Bestimmung des Maßes für Regionen mit geringer Frequenz wird diese Filterung invertiert. Nach Abschluss der Berechnung werden alle Teilmaße summiert. Entsprechend des Anwendungsfalls ist eine Gewichtung der Teilmaße möglich.

In [188] stellen Wang et. al. einen Qualitätsindex vor, welcher sich aus drei Komponenten zur Abdeckung der folgenden Störungen zusammensetzt, Verlust der Korrelation, Belichtungsstörungen und Kontrastschwankungen. Die Autoren verwenden zur Definition des Maßes statistische Größen aufgrund der Tatsache, dass ein Modell des menschlichen Wahrnehmungssystems keine bessere Leitungsfähigkeit bzgl. des Qualitätsvergleichs zeigte. Als Vorteil der Verwendung von statistischen Maßen wird angeführt, dass diese einfach und effektiv zu berechnen sind. Weiterhin sind diese unabhängig von der Wahrnehmung durch den Betrachter. Das Maß basiert auf dem Mittelwert μ und der Varianz σ zweier Bilder \mathbf{A} und \mathbf{B} :

$$M = \frac{\sigma_{AB}}{\sigma_A \sigma_B} \cdot \frac{2\mu_A \mu_B}{\mu_A^2 + \mu_B^2} \cdot \frac{2\sigma_A \sigma_B}{\sigma_A^2 + \sigma_B^2} \quad (5.1)$$

Nach [188] stellt die erste Komponente einen Korrelationskoeffizient zwischen Bild \mathbf{A} und \mathbf{B} dar. Diese misst den Grad an Korrelation zwischen \mathbf{A} und \mathbf{B} . Die zweite Komponente bestimmt die Ähnlichkeit in der Helligkeit zwischen \mathbf{A} und \mathbf{B} . Die Varianz σ_A und σ_B kann als eine Schätzung des Kontrasts verstanden werden. Die Ähnlichkeit des Kontrasts wird durch das dritte Maß definiert. Die Autoren führen an, dass es aufgrund von lokalen Qualitätsschwankungen sinnvoll ist die Maße in lokalen Bereichen zu bestimmen und am Ende zusammenzufassen. Durch die Autoren wird ein Sliding-Window Ansatz [171] verwendet. Abschließend wird die durchschnittliche Qualität über das gesamte Bild, auf Basis der Bereiche, berechnet. Durch die Autoren von [145] wird eine Erweiterung der Metrik vorgestellt. Als Neuerung wird ein lokaler Gewichtungsfaktor λ für die aktuelle Region eingeführt. Dieser kann vom Kontrast, der Schärfe, den Kanteninformationen und der Entropie des Bilds abhängig sein. Durch den Gewichtungsfaktor wird ein Qualitätsindex erzeugt, welcher den Informationsgehalt einer Region in die Bildung des Maßes einbezieht. Damit ist es möglich Bildregionen mit höherer Informationsdichte stärker zu gewichten. Als Differenzmaß wird der Mean-Squared-Error zwischen den Regionen verwendet. Die Anwendbarkeit der Metrik wurde auf Graustufenbildern verifiziert.

In [189] wird der Universal Quality Index zur allgemeinen Structural Similarity Index erweitert. An erster Stelle wird der Einfluss der Helligkeit reduziert. Nach Ansicht der Autoren ist Struktur der Objekte von der Helligkeit unabhängig. Daher wird das Maß von den Mittelwerten des Kontrasts und der Helligkeit befreit und der lokale Kontrast und die lokale Helligkeit verwendet. Das Maß wird in einen Helligkeits-

Kontrast- und Struktur-Vergleich aufgeteilt. Die einzelnen Teilmaße sind multiplikativ und können mittels exponentieller Faktoren gewichtet werden. Als erstes wird die Helligkeit l jedes Signals verglichen, dazu wird die mittlere Helligkeit jedes Eingangsbilds bestimmt. In den Gleichungen sind $C_{1,2,3}$ Stabilitäts-Terme.

$$l(\mathbf{A}, \mathbf{B}) = \frac{2\mu_A\mu_B + C_1}{\mu_A^2 + \mu_B^2 + C_1} \quad (5.2)$$

Zur Betrachtung der Kontrasts c wird die Standardabweichung des Signals verglichen

$$c(\mathbf{A}, \mathbf{B}) = \frac{2\sigma_A\sigma_B + C_2}{\sigma_A^2 + \sigma_B^2 + C_2} \quad (5.3)$$

Zum Vergleich der Struktur s wird das Signal durch die eigene Standardabweichung normalisiert.

$$s(\mathbf{A}, \mathbf{B}) = \frac{\sigma_{AB} + C_3}{\sigma_A\sigma_B + C_3} \quad (5.4)$$

Für die Metrik *SSIM* folgt:

$$SSIM(\mathbf{A}, \mathbf{B}) = l(\mathbf{A}, \mathbf{B})^\alpha \cdot c(\mathbf{A}, \mathbf{B})^\beta \cdot s(\mathbf{A}, \mathbf{B})^\gamma \quad (5.5)$$

Die Gewichtungsfaktoren α , β und γ müssen größer Null sein. Der Algorithmus wird in lokalen Fenstern angewendet, welche das Gesamtbild unterteilen. Die Autoren begründen dies damit, dass Inhalte und Störungen im Bildbereich häufig örtlich veränderlich sind. Als Ergebnis wird für jede Region der mittlere Fehler bestimmt. Die Anwendung einer lokalen Berechnung ermöglicht die Erstellung einer Bildqualitätskarte welche direkt Rückschlüsse auf die Bildqualität erlaubt.

Ergebnisse

Es kann festgestellt werden, dass sich ein Vergleich begrenzter Regionen als vorteilhaft gegenüber einem Vergleich des gesamten Bilds oder einzelner Bildpunkte erwiesen hat. Zur Schaffung einer gewissen Unschärfe und Robustheit gegenüber geometrischen Abweichungen erweist sich ein Sliding-Window Ansatz, d.h. eine Verschiebung des Vergleichsfensters als mögliche Lösung. Außerdem hat sich die Verwendung von statistischen Merkmalen, z.B. Mittelwert und Varianz, als vorteilhaft erwiesen. Diese sollten die Helligkeit, den Kontrast oder die Struktur der Vergleichsbereiche beschreiben. Die Bildung des Qualitätsmaßes kann durch eine direkte Bildung der Differenz, des Verhältnisses oder eine Auswertung von Histogrammen der Merkmale zwischen der Referenz und dem erzeugten Bild geschehen.

Schlussfolgerung

Die Metriken nach dem aktuellen Stand der Technik dienen zur Bewertung von Bildinhalten für den menschlichen Betrachter und von Kompressionsverfahren. Daher sind diese Maße nicht direkt auf den Anwendungsfall einer Anpassung von Simulationsparametern für den Test von Bildverarbeitungsfunktionen anwendbar. Anhand der betrachteten Literatur hat der SSIM für die Bewertung von Bildinhalten für

technische Anwendungsfälle die größte Akzeptanz in der Wissenschaft. Daher soll dieser als Referenz für die Entwicklung einer eigenen Metrik dienen.

5.2. Metrik zum Vergleich der Bildinhalte

Im Folgenden Abschnitt wird eine Metrik für den Vergleich von Bildinhalten mit der Zielsetzung einer Validierung und Kalibrierung einer Simulationsumgebung für das menschliche Sehen anhand von Referenzbildern definiert. Auf Basis des Stands der Technik und der Anforderungen an die Simulation werden zu Beginn Anforderungen an die Metrik präsentiert. Anschließend wird auf die Struktur der Metrik eingegangen. Danach werden Strategien und Merkmale für den Vergleich der Bildinhalte vorgestellt. Darauf folgend werden diese hinsichtlich der Anwendbarkeit untersucht. Nach dem Beweis der Anwendbarkeit wird das Gesamt-Maß der Metrik definiert. Abschließend wird durch ein Beispiel die Anwendung der Metrik demonstriert.

5.2.1. Anforderungen

Die Metrik M soll zur Validierung von synthetischen Kamerabildern einer Simulationsumgebung durch Bilder einer realen Testfahrt dienen. Das Ergebnis M soll die Distanz zwischen dem simulierten und dem realen Kamerabild darstellen. Weiterhin sollen eine Visualisierung des Ergebnisses und eine Anwendung im Rahmen einer Kalibrierung der Simulation möglich sein. Die Berechnung des Differenzmaßes soll durch Merkmale der Bildverarbeitung und der in Kapitel 4.1.3 definierten Anforderungen motiviert sein.

Die Metrik muss aufgrund von Abweichungen in der Beschreibung der Geometrie und Vereinfachung der Oberflächenmodelle in der Simulation eine gewisse Unschärfe aufweisen. Weiterhin muss sich diese Merkmalen für die Textur, Beleuchtung und geometrische Erscheinung der Objekte zusammensetzen. Entsprechend des Anwendungsfalls der Simulation, z.B. den Test von Fahrzeug- oder Spurerkennungsfunktionen, soll es möglich sein die einzelnen Maße zu gewichten.

Eine Metrik M definiert sich durch die folgenden Eigenschaften [20] [161]. Im Folgenden seien \mathbf{G}_{Ist} und \mathbf{G}_{Soll} zwei zu vergleichende Bilder oder Grauwerte-Matrizen. Die Distanz zwischen den Bildpaaren \mathbf{G}_{Ist} und \mathbf{G}_{Soll} ist Null, wenn \mathbf{G}_{Ist} und \mathbf{G}_{Soll} identisch sind.

$$M(\mathbf{G}_{\text{Ist}}, \mathbf{G}_{\text{Soll}}) = 0, \mathbf{G}_{\text{Ist}} = \mathbf{G}_{\text{Soll}} \quad (5.6)$$

Das Ergebnis der Metrik darf sich nicht verändern, wenn \mathbf{G}_{Ist} und \mathbf{G}_{Soll} vertauscht werden,

$$M(\mathbf{G}_{\text{Ist}}, \mathbf{G}_{\text{Soll}}) = M(\mathbf{G}_{\text{Soll}}, \mathbf{G}_{\text{Ist}}) \quad (5.7)$$

Weiterhin gilt für den Vergleich mit einem weiteren Bild $\mathbf{G}_{\text{Ist},2}$. Wenn das Bild \mathbf{G}_{Ist} dem Bild \mathbf{G}_{Soll} ähnlicher ist als das Bild $\mathbf{G}_{\text{Ist},2}$ dem Bild \mathbf{G}_{Soll} .

$$\frac{M(\mathbf{G}_{\text{Soll}}, \mathbf{G}_{\text{Ist},2})}{M(\mathbf{G}_{\text{Soll}}, \mathbf{G}_{\text{Ist}})} \gg 1 \quad (5.8)$$

Wenn \mathbf{G}_{Ist} und $\mathbf{G}_{\text{Ist},2}$ ähnlich sind,

$$\frac{M(\mathbf{G}_{\text{Soll}}, \mathbf{G}_{\text{Ist}})}{M(\mathbf{G}_{\text{Soll}}, \mathbf{G}_{\text{Ist},2})} \approx 1 \quad (5.9)$$

Darauf folgt weiterhin die fünfte Bedingung, wenn der Abstand zwischen Bild \mathbf{G}_{Soll} und Bild \mathbf{G}_{Ist} , ähnlich dem Abstand zwischen Bild $\mathbf{G}_{\text{Soll},2}$ und Bild $\mathbf{G}_{\text{Ist},2}$ ist,

$$\frac{M(\mathbf{G}_{\text{Soll}}, \mathbf{G}_{\text{Ist}})}{M(\mathbf{G}_{\text{Soll},2}, \mathbf{G}_{\text{Ist},2})} \approx 1 \quad (5.10)$$

5.2.2. Struktur der Metrik

In Abbildung 5.1 ist die Struktur der Metrik dargestellt. Als Eingangsgrößen für die Berechnung des Abstands dienen ein Soll-Bild \mathbf{G}_{Soll} und ein Ist-Bild \mathbf{G}_{Ist} . Auf Basis der Bilddaten werden geometrische und pixelbasierte Merkmale als Eingangsgrößen für die Metrik berechnet. Während die pixelbasierten Merkmale direkt aus den Bilddaten \mathbf{G} abgeleitet werden, müssen für die Erzeugung der geometrischen Merkmale lokale Operationen im Bildbereich ausgeführt werden. Als Erstes wird eine Filterung zur Reduktion von Bildrauschen mittels eines Gaußfilters durchgeführt. Anschließend werden mittels des Sobel-Kantenextraktors der Betrag und die Richtung der Kantenpunkte bestimmt. Zur Unterdrückung von Rauschen wird im nächsten Schritt eine Segmentierung der Gradientenbeträge und Gradientenrichtungen anhand des zweifachen Mittelwerts der Gradientenbeträge durchgeführt. Anschließend wird eine Blockbildung auf die segmentierten geometrischen und pixelbasierten Merkmale durchgeführt und die Teilmaße jeder Region berechnet. Auf Basis dieser Teilmaße wird abschließend das Gesamt-Maß gebildet. Die durchgeführten Operationen zur Berechnung der Metrik sollen im Folgenden näher beschrieben und hergeleitet werden.

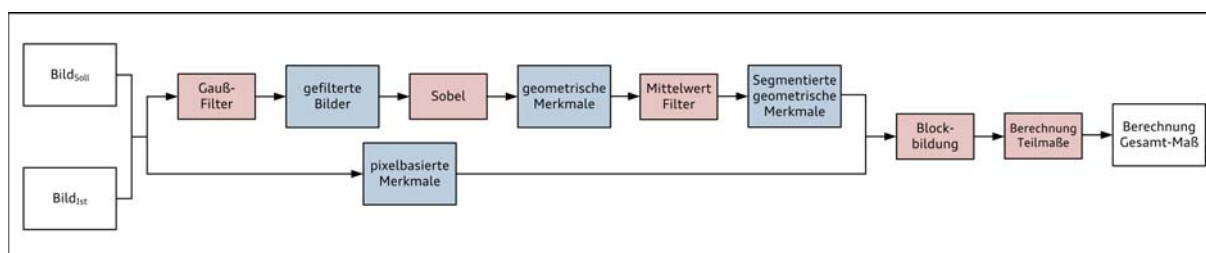


Abbildung 5.1.: Struktur der Metrik

5.2.3. Strategien zum Vergleich der Bildinhalte

In den Anforderungen wurde definiert, dass die Metrik eine gewisse Robustheit gegenüber Abweichungen in der Geometrie und in den Oberflächenmodellen aufweisen muss. Weiterhin soll die Vergleichsstrategie eine räumliche Interpretation der Abweichungen im Bild erlauben. Wie im Stand der Technik zu diesem Kapitel dargelegt, ist es nicht sinnvoll einzelne Pixel miteinander zu vergleichen, da diese zu

sensitiv gegenüber geometrischen Abweichungen sind. Durch die Berechnung von Vergleichsmerkmalen über Bildregionen kann bei einer ausreichenden Größe dieser Regionen eine leichte Verschiebung von Pixeln kompensiert werden. Daher werden hier Strategien für den Vergleich von Bildregionen diskutiert. Diese Ansätze können als Derivate des Block-Matchings aufgefasst werden [171]. Als vorteilhaft bei der Durchführung eines Block-Vergleichs, anstelle eines Vergleichs des kompletten Bilds, erweist sich die Wahrung einer gewissen räumlichen Interpretation der Ergebnisse. Zur Bewertung der am besten geeigneten Vergleichsstrategie zur Abdeckung der Anforderungen sollen die folgenden Konzepte betrachtet werden:

- Sliding-Window
- Rasterisierung
- Rasterisierung mit Überlappung
- Objektvergleich

In Abbildung 5.2 sind die verschiedenen Ansätze hinsichtlich der Auswirkungen auf das Bild dargestellt.

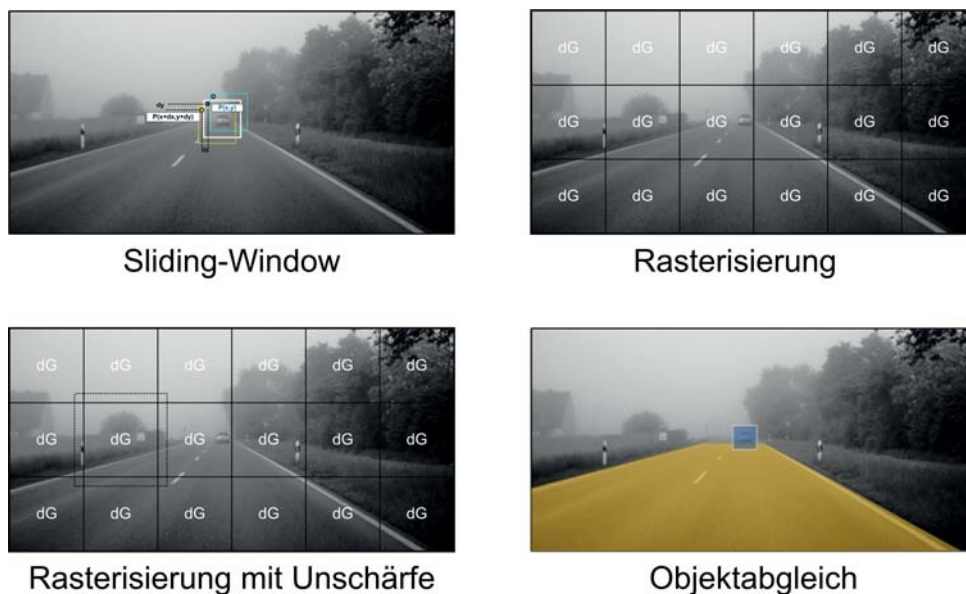


Abbildung 5.2.: Ansätze zum Vergleich der Bildinhalte

5.2.3.1. Definition eines Blocks

Bevor die einzelnen Ansätze diskutiert werden, soll eine Definition eines Blocks im Bild \mathbf{G} erfolgen. Zur Kompensation von Abweichungen in der örtlichen Position einzelner Pixel g wird der Vergleich mittels Bildregionen $\delta\mathbf{G}$ durchgeführt, welche um den Punkt $\mathbf{p}(\mathbf{i}, \mathbf{j})$ definiert werden. Die Größe der verwendeten Regionen wird durch die Breite δ_x und die Höhe δ_y definiert. Aus der Division der Bildbreite durch die Breite der Regionen ergibt sich die Anzahl der Regionen in X- und Y-Richtung $n_x = d_{x,max}/\delta_x$ und

$n_y = d_{y,max}/\delta_y$. Die Anzahl aller Regionen ist mit $N = n_x \cdot n_y$ definiert. Dabei müssen n_x und n_y ganze Zahlen sein.

$$\delta\mathbf{G}(\mathbf{i}, \mathbf{j}) = \begin{bmatrix} g((i-1)*\delta_x+1, (j-1)*\delta_y+1) & g((i-1)*\delta_x+1, (j-1)*\delta_y+2) & \cdots & g((i-1)*\delta_x+1, (j-1)*\delta_y+\delta_y) \\ \vdots & \vdots & \ddots & \vdots \\ g((i-1)*\delta_x+\delta_x, (j-1)*\delta_y+1) & g((i-1)*\delta_x+\delta_x, (j-1)*\delta_y+2) & \cdots & g((i-1)*\delta_x+\delta_x, (j-1)*\delta_y+\delta_y) \end{bmatrix} \quad (5.11)$$

Durch die Größe des Blocks kann die Sensitivität der Metrik gegenüber der Abweichungen gesteuert werden. Die Wahl einer kleinen Blockgröße erhöht die Genauigkeit der Metrik. Bei Ausweitung der Größe eines Blocks auf die Bildgröße wird der komplette Bildinhalt verglichen. Der Block sollte für den Vergleich größer als die erwartete Abweichung der Geometrie sein.

5.2.3.2. Sliding-Window

Unter der Annahme, dass sich die zu vergleichenden Inhalte der Soll-Region im Umfeld der Ist-Region befinden, kann der Sliding-Window Ansatz herangezogen werden. Beim hier angewendeten Sliding-Window Ansatz [171] wird eine Soll-Region $\delta\mathbf{G}_{\text{Soll}}$ aus dem Soll-Bild \mathbf{G}_{Soll} mit einer Reihe von Kandidatenregionen $\delta\mathbf{G}_{\text{Ist}, [1...m]}$ aus dem Ist-Bild \mathbf{G}_{Ist} verglichen. Zur Generierung der Kandidatenregionen wird die Ursprungs-Region, welche um den Punkt $\mathbf{p}(\mathbf{x}, \mathbf{y})$ definiert wurde an verschiedene Positionen verschoben. Die Verschiebung einer weiteren Kandidatenregionen zum Ursprung wird durch p_{dx} und p_{dy} definiert. Die Kandidatenregion mit der kleinsten Abweichung zur Soll-Region beschreibt die Abweichung zwischen realer und simulierter Szene in diesem Bildbereich. In Abbildung 5.2 ist der Ansatz mit zwei Kandidatenregionen dargestellt. Durch die Notwendigkeit der Bestimmung der Abweichung zwischen Soll-Region und den Kandidatenregionen ist dieser Ansatz sehr aufwendig zu berechnen und eignet sich nur bedingt für den gesetzten Anwendungsfall. Daher werden im Folgenden optimierte Varianten des Sliding-Window Ansatzes betrachtet.

5.2.3.3. Rasterisierung

Die Idee zur Rasterisierung stellt eine Vereinfachung des Sliding-Window Ansatzes dar. Bei dem Verfahren wird eine Bildregion aus dem Soll-Bild mit einer korrespondieren Region aus dem Ist-Bild verglichen. Dafür wird das Bild in feste Regionen für den Vergleich unterteilt vgl. Abbildung 5.2. Bei der Rasterisierung wird nur eine Region um den Punkt $\mathbf{p}(\mathbf{x}, \mathbf{y})$ definiert. Der Abstand der Regionen zueinander entspricht $\delta_x/2$ und $\delta_y/2$. Es wird angenommen, dass sich die zu vergleichenden Bildinhalte in der entsprechenden Region befinden. Durch die ausschließliche Prüfung einer Kandidaten- mit einer Referenzregion ist eine Reduktion der Rechenzeit möglich.

5.2.3.4. Rasterisierung mit Unschärfe

Zur Steigerung der Robustheit gegenüber einer Verschiebung von Kanten außerhalb der Rasterisierungsregion $\delta\mathbf{G}$ kann es sinnvoll sein, einen gewissen Rand bestehend aus benachbarten Pixeln der aktuellen Region mit in die Berechnung der Merkmale mit einzubeziehen. Die Definition der Region ergibt sich

daher wie in Abbildung 5.2 dargestellt. Durch die Größe des Randbereichs kann die Unschärfe für den Vergleich definiert werden. Es ist denkbar die Unschärfe in Abhängigkeit der Distanz des Objekts von der Kamera zu definieren.

5.2.3.5. Objekt-Vergleich

Zur Reduktion des Einflusses von geometrischen Abweichungen im Randbereich und zur Beschleunigung der Berechnungszeit kann der Vergleich auf im Vorfeld markierte Bildregionen (Objekte oder ROI-Regions of Interest) beschränkt werden. Zur Anwendung ist es notwendig, die Objekte in der realen Testfahrt und im simulierten Bild zu markieren. Weiterhin kann es notwendig sein, die Größen der Regionen anzupassen. Der Berechnungsaufwand steigt linear mit der Anzahl der Objekte.

5.2.4. Merkmale für den Vergleich der Bildinhalte

5.2.4.1. Einteilung der Bildmerkmale

Nach [171] können Bildmerkmale mittels pixel- oder merkmalsbasierter Verfahren analysiert werden. Bei Ersteren werden die Ähnlichkeiten auf Bildpunktebene unterschieden. Als Merkmale jedes Bildpunkts stehen die Intensität oder die Farbe zur Verfügung. Zur Steigerung der Aussagekraft werden häufig Regionen oder Blöcke um die Bildpunkte betrachtet. Wie in Abbildung 5.3 ersehen werden kann, ist für die Durchführung der Operation keine weitere Vorverarbeitung der Bilddaten notwendig. Die merkmalsbasierten Verfahren verwenden auf Basis der Pixel berechnete Merkmale, z.B. Kantenpunkte, Linien, Konturen und Objekte. Durch den gesteigerten Informationsgehalt ist es möglich eine bessere Zuordnung der Merkmale zu definierten Klassen zu erreichen. Diese Merkmale stellen eine Obermenge der in Ab-

Stufe 6	Objekte	↓	↑	↑	↑
Stufe 5	Regionen				
Stufe 4	Konturen				
Stufe 3	Linien				
Stufe 2	Kantenpunkte				
Stufe 1	Pixel				
		Anzahl	Eindeutigkeit	Detektionsaufwand	Informationsgehalt

Abbildung 5.3.: Eigenschaften verschiedener Bildmerkmale hinsichtlich ihrer Bedeutung in der Bildverarbeitung, nach [171]

schnitt 4.1.3 beschriebenen Merkmale zur Realisierung von Fahrerassistenzfunktionen dar. Daher ist es naheliegend eine Metrik basierend auf diesen Merkmalen zu definieren.

5.2.4.2. Statistische Bildmerkmale für den Vergleich

Zur Durchführung des Vergleichs werden der Mittelwert μ und die Varianz σ^2 der Grauwerte-Matrix \mathbf{G} bestimmt. Diese Merkmale können auch über eine definierte Teilmenge der Grauwerte-Matrix (Region) $\delta\mathbf{G}$ berechnet werden. Eine wichtige Eigenschaft dieser statistischen Merkmale ist die Invarianz gegenüber einer Rotation oder Skalierung des Bilds [92]. Die Dimension der Grauwerte-Matrix ist in der Breite mit d_x und in der Höhe d_y definiert.

Mittelwert

Zur Bestimmung der durchschnittlichen Ausprägung eines Merkmals innerhalb des Bildes \mathbf{G} oder Region $\delta\mathbf{G}$ wird der Mittelwert $\mu(\mathbf{G})$ verwendet.

$$\mu(\mathbf{G}) = \frac{1}{d_x d_y} \sum_{i=1}^{d_x} \sum_{j=1}^{d_y} g_{i,j} \quad (5.12)$$

Da der Mittelwert nur einen Durchschnittswert eines Bildbereichs darstellt ist es denkbar, dass komplett unterschiedliche Bildinhalte den gleichen Mittelwert aufweisen. In Abbildung 5.4 sind zwei Bildbereiche mit identischen Mittelwert ($\mu = 0.5$) dargestellt. Anhand dieses Beispiels wird deutlich, dass der Mittelwert nicht als alleiniges Merkmal ausreichend ist.



Abbildung 5.4.: Eigenschaft Mittelwert und Varianz

Varianz

Die Verwendung des Mittelwerts erlaubt keine Aussage über die statistische Verteilung der Messwerte innerhalb der Stichprobe. Daher ist es möglich, dass ein Rauschsignal den gleichen Mittelwert wie ein strukturiertes Bild aufweist. Zur Analyse der Merkmale wird daher außerdem die lokale Varianz $\sigma(\mathbf{G})^2$ verwendet [92]. Die lokale Varianz beschreibt die Schwankung der Messwerte um den Mittelwert $\mu(\mathbf{G})$. Für die Varianz $\sigma(\mathbf{G})^2$ folgt:

$$\sigma(\mathbf{G})^2 = \frac{1}{d_x d_y - 1} \sum_{i=1}^{d_x} \sum_{j=1}^{d_y} (g_{i,j} - \mu(\mathbf{G}))^2 \quad (5.13)$$

5.2.4.3. Allgemeine Definition des Vergleichsmaßes

Zur Bestimmung des Differenz-Maßes werden für die korrespondierenden Blöcke im Soll- und Ist-Bild der Mittelwert $\mu(\delta\mathbf{G})$ und die Varianz $\sigma(\delta\mathbf{G})^2$ bestimmt. Der Fehler e ergibt sich aus der Summe der Beträge des auf den Sollwert bezogenen Fehlers, der Mittelwerte und der Varianzen des Soll- und des Ist-Bilds. Für den Fehler des Mittelwerts folgt,

$$e(\delta\mathbf{G}_{\text{Ist}}, \delta\mathbf{G}_{\text{Soll}})_{\mu} = \frac{|\mu(\delta\mathbf{G}_{\text{Soll}}) - \mu(\delta\mathbf{G}_{\text{Ist}})|}{\mu(\delta\mathbf{G}_{\text{Soll}})} \quad (5.14)$$

und für die Varianz folgt,

$$e(\delta\mathbf{G}_{\text{Ist}}, \delta\mathbf{G}_{\text{Soll}})_{\sigma^2} = \frac{|\sigma(\delta\mathbf{G}_{\text{Soll}})^2 - \sigma(\delta\mathbf{G}_{\text{Ist}})^2|}{\sigma(\delta\mathbf{G}_{\text{Soll}})^2} \quad (5.15)$$

für den Fehler gelte, $e \in \mathfrak{R}[0.0 \ 1.0]$, wenn $e > 1.0$, dann gelte $e = 1.0$ und damit ergibt sich für den Fehler des Merkmals eines Bilds

$$e(\delta\mathbf{G}_{\text{Ist}}, \delta\mathbf{G}_{\text{Soll}}) = \frac{e(\delta\mathbf{G}_{\text{Ist}}, \delta\mathbf{G}_{\text{Soll}})_{\mu} + e(\delta\mathbf{G}_{\text{Ist}}, \delta\mathbf{G}_{\text{Soll}})_{\sigma^2}}{2} \quad (5.16)$$

Durch die Verwendung des Fehlers des Mittelwerts ist es möglich einzelne Merkmale in einer Region zu vergleichen. Die Varianz erlaubt eine Aussage zur Schwankungsbreite des entsprechenden Merkmals. Die Summe aus den Fehlern des Mittelwerts und Varianz ermöglicht eine genaue Beschreibung der Abweichung. Durch die Addition der Fehler sind diese nicht direkt korreliert und durch die Division normiert.

5.2.4.4. Berechnung des photometrischen Maßes einer Region

Das photometrische Maß ist durch den Mittelwert und die Varianz der Intensitäten der Bildpunkte der Region oder des Blocks definiert:

$$e_{\text{Photo}}(\delta\mathbf{G}_{\text{Ist}}, \delta\mathbf{G}_{\text{Soll}}) = e(\delta\mathbf{G}_{\text{Ist}}, \delta\mathbf{G}_{\text{Soll}}) \quad (5.17)$$

5.2.4.5. Berechnung der geometrischen Merkmale

Zur Berechnung der geometrischen und der strukturellen Maße werden im Vorfeld geometrische Merkmale in der Form von Kantenpunkten und Kantenrichtung verwendet. Im Vergleich zu den pixelbasierten Merkmalen sind diese von der Rotation und der Skalierung der Objekte abhängig.

Zur Generierung der geometrischen Merkmale wird auf Teile des Canny-Algorithmus zurückgegriffen [39]. Der Ablauf gestaltet sich wie folgt:

1. Reduktion von Rauschen durch Faltung mit einem Gaußglocken ähnlichen Filterkern
2. Berechnung des Betrags und der Richtung des Gradienten

3. Segmentierung des Betrags und der Richtung der Gradienten mit einem Schwellwert

Reduktion von Rauschen durch Faltung mit einem Gaußlocken ähnlichen Filterkern

Zur Reduktion von Rauschen, welches bei photographischen Aufnahmen entsteht wird die Grauwerte-Matrix \mathbf{G} im Vorfeld mit der Filtermaske \mathbf{H} gefaltet. Die Filtermaske ist für einen 3x3 Ortsbereich wie folgt definiert:

$$\mathbf{H} = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \quad (5.18)$$

Damit folgt für die gefilterte Graustufen-Matrix \mathbf{G}_{filt} :

$$\mathbf{G}_{\text{filt}} = \mathbf{H} \cdot \mathbf{G} \quad (5.19)$$

Berechnung der geometrischen Merkmale mittels des Sobel-Operators

Zur Berechnung der geometrischen Merkmale der gefilterten Graustufen-Matrix \mathbf{G}_{filt} wird der Sobel-Operator [92] verwendet. Das Ergebnis der Operationen sind die Matrizen des Gradientenbetrags ∇ und der Orientierung der Gradienten Θ im Bild. Die Orientierung eines Gradienten Θ beschränkt sich auf $\Theta = \{0^\circ, 45^\circ, 90^\circ, 135^\circ\}$ und ist durch die folgende Definition gegeben, entsprechend [39]:

$$\Theta = \begin{cases} 0 & 0 < \theta \leq 22.5^\circ \vee 147.5^\circ < \theta \leq 190^\circ \\ 45^\circ & 22.5^\circ < \theta \leq 57.5^\circ \\ 90^\circ & 57.5^\circ < \theta \leq 112.5^\circ \\ 135^\circ & 112.5^\circ < \theta \leq 147.5^\circ \end{cases} \quad (5.20)$$

Die Berechnung der Kanten geschieht durch Faltung des gefilterten Bilds G_{filt} mit dem horizontalen \mathbf{H}_x und dem vertikalen \mathbf{H}_y Sobel-Filter. Die Graustufen-Matrix \mathbf{G}_{filt} wird daher einer zweimaligen Faltungsoperation unterzogen.

$$\mathbf{H}_x = \frac{1}{8} \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad \mathbf{H}_y = \frac{1}{8} \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad (5.21)$$

Damit ergibt sich für die Kantenpunkte in horizontaler und vertikaler Richtung die Matrizen ∇_x und ∇_y :

$$\nabla_x = \mathbf{H}_x \cdot \mathbf{G}_{\text{filt}} \quad \nabla_y = \mathbf{H}_y \cdot \mathbf{G}_{\text{filt}} \quad (5.22)$$

Auf Grundlage der horizontalen und vertikalen Kantenpunkte folgt für den Betrag ∇ in Zeile i und Spalte j :

$$\nabla(i, j) = \sqrt{\nabla_x(i, j)^2 + \nabla_y(i, j)^2} \quad (5.23)$$

Für die Matrix der Gradientenbeträge ∇ der Grauwerte-Matrix \mathbf{G} folgt somit:

$$\nabla = \begin{bmatrix} \nabla(1, 1) & \cdots & \nabla(1, d_{x,max}) \\ \vdots & \ddots & \vdots \\ \nabla(d_{y,max}, 1) & \cdots & \nabla(d_{y,max}, d_{x,max}) \end{bmatrix} \quad (5.24)$$

Für Richtung Θ des Gradienten der Kante in Zeile i und Spalte j folgt:

$$\Theta(i, j) = \arctan\left(\frac{\nabla_x(i, j)^2}{\nabla_y(i, j)^2}\right) \quad (5.25)$$

Damit folgt für die Richtungs-Matrix Θ der gesamten Grauwerte-Matrix G :

$$\Theta = \begin{bmatrix} \Theta(1, 1) & \cdots & \Theta(1, d_{x,max}) \\ \vdots & \ddots & \vdots \\ \Theta(d_{y,max}, 1) & \cdots & \Theta(d_{y,max}, d_{x,max}) \end{bmatrix} \quad (5.26)$$

Unterdrückung von Rauschen

Zur Unterdrückung fehlerhaft durch Rauschen gebildeter Kanten wird eine Segmentierung auf den zweifachen Mittelwert der Gradienten-Matrix ∇ durchgeführt. Für ein einzelnes Element $\nabla(i, j)$ der Matrix ∇ folgt somit:

$$\nabla(i, j) = \begin{cases} 0 & | \nabla(i, j) < 2 \cdot \mu(\nabla) \\ \nabla & | \nabla(i, j) \geq 2 \cdot \mu(\nabla) \end{cases} \quad (5.27)$$

Die gleiche Operation wird auch auf die Matrix der Gradientenrichtungen Θ angewendet, damit ergibt sich für ein Element $\Theta(i, j)$:

$$\Theta(i, j) = \begin{cases} 0 & | \nabla(i, j) < 2 \cdot \mu(\nabla) \\ \Theta(i, j) & | \nabla(i, j) \geq 2 \cdot \mu(\nabla) \end{cases} \quad (5.28)$$

Strukturelles-Maß

Zur Schaffung einer Aussage über die Ausprägung der Strukturen werden die Beträge der Kantenpunkte zur Bestimmung eines Maßes einzelner Regionen verwendet.

$$e_{Strukt}(\nabla(\delta\mathbf{G}_{Ist}), \nabla(\delta\mathbf{G}_{Soll})) = e(\nabla(\delta\mathbf{G}_{Ist}), \nabla(\delta\mathbf{G}_{Soll})) \quad (5.29)$$

Geometrisches-Maß

Das geometrische Maß wird durch die Verwendung der Richtung des Kantenspektrums der geometrischen Merkmale der Regionen bestimmt.

$$e_{Geo}(\Theta(\delta\mathbf{G}_{Ist}), \Theta(\delta\mathbf{G}_{Soll})) = e(\Theta(\delta\mathbf{G}_{Ist}), \Theta(\delta\mathbf{G}_{Soll})) \tag{5.30}$$

5.2.5. Untersuchung der Vergleichsstrategien und der Merkmale für den Vergleich der Bildinhalte

Im Folgenden werden die Vergleichsstrategien und Merkmale für den Vergleich der Bildinhalte hinsichtlich der Erfüllung der Anforderungen und der Eigenschaften untersucht.

5.2.5.1. Betrachtung der Vergleichsstrategien

Die verschiedenen Ansätze werden anhand einer statischen Szene betrachtet. Dazu werden der Sliding-Window-, der Rasterisierungs- und der Rasterisierungsansatz mit Unschärfe auf die in Abbildung 5.5 dargestellte Szene angewendet. Für den Vergleich auf Objektebene, werden die ROI-Fahrzeug und ROI-Straße verwendet. Als Ergebnis werden die Differenzen der einzelnen Teilmaße, die Gesamtdifferenz, die






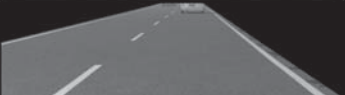
	Realität	Simulation
Gesamte Szene		
ROI-Fahrzeug		
ROI-Straße		

Abbildung 5.5.: Verwendete Daten für die Bewertung der Vergleichsstrategien

Bildgröße, die Anzahl der Regionen und die Berechnungszeit in Matlab t_{Comp} erfasst. In Tabelle 5.1 sind die Ergebnisse der Teilmaße bei Verwendung der verschiedenen Ansätze dargestellt. Wie ersehen werden kann ist der Sliding-Window Ansatz am Robustesten gegenüber Abweichungen zwischen dem Referenzdaten und den simulierten Bildinhalten. Die Berechnungszeit ist durch die Mehrberechnung der Regionen

Ansatz	m _{Photo}	m _{Strukt}	m _{Geo}	M	t _{Comp}	Bildgröße	Regionen
Sliding-Window	0.4832	0.6191	0.6000	0.5674	299.2188	1024x512	2048
Rasterisierung	0.6316	0.6421	0.6127	0.6288	65.11s	1024x512	2048
Raster. (10px Rand)	0.6097	0.7252	0.7086	0.6812	66.09s	1024x512	2048
ROI-Fahrzeug	0.4892	0.6637	0.4955	0.5495	0.64s	80x67	56
ROI-Straße	0.4064	0.4849	0.5090	0.4668	30.47s	950x260	200
ROI-Gesamt	K/A	K/A	K/A	K/A	31.11s	K/A	256

Tabelle 5.1.: Vergleich der Blockbildungs-Verfahren

fünf Mal so hoch wie die des einfachen Rasterisierungsansatzes. Die Rasterisierung ist deutlich sensitiver gegenüber photometrischen Abweichungen. Die Anwendung einer überlappenden Region weist in diesem Beispiel keinen Vorteil auf.

Während mittels der generischen Sliding-Window- oder Rasterisierungsansätze das globale Bild betrachtet wird, ist es möglich auf Objektebene gezielt bestimmte Objekte zu betrachten. Wie ersehen werden kann ist es möglich durch einen direkten Vergleich von Objekten Berechnungszeit einzusparen. In dem Fall, dass durch das Fahrzeugsystem die Generierung von Bounding-Boxen im realen Bildmaterial möglich ist, kann das manuelle Markieren der Objekte der in den realen Videodaten vermieden werden. Durch das Vorliegen von fehlerfreien Referenzdaten in der Simulation kann dies dort gänzlich vermieden werden. Damit ist es möglich einen automatischen Abgleich auf Objektebene durchzuführen.

Aufgrund der einfachen Umsetzbarkeit und guter Ergebnisse des Rasterisierungsansatzes wird dieser im Folgenden weiterverfolgt. Der Algorithmus bietet den besten Kompromiss aus Genauigkeit und Empfindlichkeit gegenüber Änderungen. Für gezielte Auswertungen von Objekten ist der Vergleich von Regionen von Interesse.

5.2.5.2. Betrachtung der Teilmaße

Im Folgenden werden die Eigenschaften der einzelnen Teilmaße genauer beleuchtet. Als Grundlage wurde [198] verwendet. Die Teilmaße wurden für die Verifikation auf die in Abbildung 5.6 dargestellten Bilder angewendet. Die Struktur innerhalb der Bilder ist für den Vergleich einfach gehalten um eine gute Analysierbarkeit der Ergebnisse zur gewährleisten. Das Bild besitzt eine Größe von 64 x 64 Pixeln. Die Testobjekte sind Variationen des Referenzbilds hinsichtlich Veränderungen in, Helligkeit, Kontrast, Translation von Bildpunkten und Spiegelung/Rotation des Bilds. Weiterhin werden die Maße durch ein Bild mit Rauschen um den Mittelwert des Referenzbilds und ein Bild des Mittelwerts der Referenz-Region getestet. Die durchgeführten Modifikationen des Ursprungsbildes sind wie folgt definiert:

- Test 1: Die Helligkeit jedes einzelnen Pixels wurde um einen festen Graustufenwert $g_{fixed} = 0.0784$ erhöht.
- Test 2: Der Kontrast $K = g_{max}/g_{min}$ wurde von $K = 0.6546$ auf $K = 2.5500$ erhöht.
- Test 3: Die Linien wurden um sieben Pixel in der X-Achse nach links verschoben.
- Test 4: Das Referenzbild wurde um die Y-Achse von links nach rechts gespiegelt.






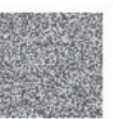

							
Bezeichnung	Referenz	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6
Veränderung	keine	Helligkeit	Kontrast	Translation	Spiegelung / Rotation	Rauschen	Mittelwert

Abbildung 5.6.: Auswirkung der Teilmaße

- Test 5: Der Inhalt wurde mittels einer Rauschfunktion $r(t)$ erzeugt. Die Amplitude ergibt sich aus der Varianz des Referenzbilds, welche um den Mittelwert schwankt.
- Test 6: Der Inhalt ist der Mittelwert μ des Referenzbild \mathbf{G}_{ref} .

Ergebnisse

In Tabelle 5.2 sind die Ergebnisse der einzelnen Maße mit Hinblick auf das modifizierte Eingangsbild dargestellt.

Datensatz	Modifikation	m_{photo}	m_{Strukt}	m_{Geo}
Test 1	Helligkeit	0.0777	0.0	0.0
Test 2	Kontrast	0.7093	1.0	0.0
Test 3	Translation	0.0	0.0	0.0
Test 4	Spiegelung oder Rotation	0.0	0.0	1.0
Test 5	Rauschen	0.5073	0.3905	0.3688
Test 6	Mittelwert	0.5012	1.0	1.0

Tabelle 5.2.: Analyse der Eigenschaften

Auswertung

Test 1: Es kann ersehen werden, dass das pixelbasierte Maß die Erhöhung der Intensität der Bildpunkte sicher detektiert. Die Abweichung beträgt 0.0777. Hinsichtlich der Geometrie und Struktur wurden keine Abweichungen festgestellt.

Test 2: Für den zweiten Test wurde der Kontrast angepasst. Aufgrund der Helligkeitsabweichungen reagiert das photometrische Maß. Es wird eine Abweichung von 0.7093 detektiert. Durch den stärkeren Helligkeitsunterschied an den Kanten der Linien wird ebenso das strukturelle Maß mit einer entsprechenden Abweichung bestimmt.

Test 3: Wie im dritten Test ersehen werden kann ist Metrik robust gegenüber einer Translation der Linien innerhalb der Region.

Test 4: Wie ersehen werden kann reagieren die photometrischen und strukturellen Maße nicht auf eine Rotation oder Spiegelung des Bildinhalts innerhalb der Region. Das geometrische Maß detektiert innerhalb der Region eine entsprechende Abweichung.

Test 5: Im Rahmen des fünften Tests wurde die Maße gegenüber eines Bilds mit den vorher definierten Rauscheigenschaften erprobt. Wie ersehen werden kann, ist es möglich das Bild mit Hilfe der strukturellen und geometrischen Maße sicher zu differenzieren.

Test 6: Die Summe der Teilmaße ist robust gegenüber einem Bild mit dem gleichen Mittelwert. Durch die nicht vorhandene Struktur und Ausprägung der Geometrie werden in den entsprechenden Teilmaßen Differenzen bestimmt.

Zusammenfassung

Durch die Beispiele konnte gezeigt werden, dass die Einzelmaße zur Detektion von Abweichungen der Helligkeit und dem Kontrast innerhalb einer Bildregionen eingesetzt werden können. Weiterhin wurde gezeigt, dass die Maße gegenüber einer lokalen Translation von Bildpunkten innerhalb der betrachteten Region invariant sind. Durch die Verwendung des geometrischen Maßes ist es möglich Rotationen oder Spiegelungen von Pixeln zu detektieren. Abschließend wurde die Robustheit gegenüber eines Rauschens, um den Mittelwert und mittels eines Mittelwertbilds demonstriert.

5.2.6. Gesamt-Maß der Metrik

Das Gesamt-Maß setzt sich aus den drei einzelnen Maßen jeder Region zusammen:

$$m_{Photo} = \frac{1}{N} \sum_{i=1}^{i \leq n_x} \sum_{j=1}^{j \leq n_y} e_{Photo}(\delta \mathbf{G}(\mathbf{i}, \mathbf{j})_{Ist}, \delta \mathbf{G}(\mathbf{i}, \mathbf{j})_{Soll}) \quad (5.31)$$

$$m_{Strukt} = \frac{1}{N} \sum_{i=1}^{i \leq n_x} \sum_{j=1}^{j \leq n_y} e_{Strukt}(\delta \nabla(\mathbf{i}, \mathbf{j})_{Ist}, \delta \nabla(\mathbf{i}, \mathbf{j})_{Soll}) \quad (5.32)$$

$$m_{Geo} = \frac{1}{N} \sum_{i=1}^{i \leq n_x} \sum_{j=1}^{j \leq n_y} e_{Geo}(\delta \Theta(\mathbf{i}, \mathbf{j})_{Ist}, \delta \Theta(\mathbf{i}, \mathbf{j})_{Soll}) \quad (5.33)$$

Die Konstante n_x definiert die Anzahl der Regionen entlang der X-Richtung des Bilds und n_y die Anzahl der Variablen entlang der Y-Richtung des Bilds. Das Maß wird auf die Anzahl der Regionen $N = n_x \cdot n_y$ gewichtet. Zur Gewichtung der einzelnen Maße entsprechend der Relevanz für den aktuellen Anwendungsfall, werden die Koeffizienten c eingeführt. Mittels dieser Koeffizienten kann der Beitrag eines Teilmaßes für das Gesamtergebnis gewichtet werden. Bei der Erprobung eines Spurerkennungsalgorithmus kann z.B. die Wichtigkeit der pixelbasierten Merkmale reduziert werden, und dafür eine stärkere Gewichtung der

Gradientenbeträge und der Orientierung der Kanten vorgenommen werden. Damit folgt für das Gesamtmaß der Metrik M :

$$M = c_{Photo}m_{Photo} + c_{Strukt}m_{Strukt} + c_{Geo}m_{Geo} \quad (5.34)$$

5.2.7. Anwendungsbeispiel

Zur Demonstration der Fähigkeiten der Metrik wird diese auf fünf Testbilder angewendet und mit dem SSIM [189] verglichen:

- Ein Referenz-Bild mit identischem Inhalt, zur Prüfung der Identitäts-Eigenschaft
- Eine der Realität nachempfundene Szene mit Nebel
- Eine der Realität nachempfundene Szene ohne Nebel
- Ein Bild mit zufälligem Inhalt
- Ein Bild mit Rauschen gleichen Mittelwerts und Varianz wie das Referenz-Bild.

Der Vergleich der Bilder wird mittels 64×32 Regionen durchgeführt. Die Metrik wurde für die Auswertung, ohne eine Überdeckung der Regionen angewendet. In Abbildung 5.7 sind die Ergebnisse der Metrik bei Anwendung auf komplette Bilder dargestellt. Wie ersehen werden kann erfüllen beide Maße







Referenzbild	Selbst-Test	Simulation mit Nebel	Simulation ohne Nebel	Zufallsbild	Rauschen
					
Photometr. Maß	0.0	0.5834	0.6083	0.7289	0.7237
Strukturelles Maß	0.0	0.5512	0.5651	0.7290	0.9155
Geometrisches Maß	0.0	0.4920	0.5066	0.6660	0.8660
Differenz-Maß	0.0	0.5422	0.5600	0.7079	0.8351
(1-SSIM)	0.0	0.6876	0.7212	0.7494	0.8124

Abbildung 5.7.: Anwendung der Metrik auf gesamte Bilder im Vergleich zum SSIM

die Identitätseigenschaft. Bei Betrachtung der Ergebnisse der simulierten Szene mit und ohne Nebel kann festgestellt werden, dass beide Maße die Szene mit Nebel näher an der Referenz sehen als die Szene ohne Nebel. Das definierte Differenz-Maß weist, im Vergleich zum SSIM, eine geringere Differenz auf. Bei Vergleich der Ergebnisse des SSIM für das Zufallsbild und das Rausch-Bild fällt auf, dass diese sehr nah an den Bildern mit formal richtigem Inhalt sind. Die Differenz des Zufallsbildes zum Bild mit Sonnenschein beträgt nur 0.0367 und des Rausch-Bild zum Sonnenschein-Bild 0.0906. Im Vergleich dazu weist die neue Metrik eine Differenz von 0.1479 zwischen dem Zufallsbild und der Simulation ohne Nebel auf. Die Distanz zum Rausch-Bild beträgt 0.2751. Damit sind die Distanzen mindestens doppelt so groß. Für die Bildung des Distanz-Maßes erweist sich die Verwendung der Gradientenbeträge und Gradientenrichtungen als positiv.







Referenzbild	Selbst-Test	Simulation mit Nebel	Simulation ohne Nebel	Zufallsbild	Rauschen
					
Photometr. Maß	0.0	0.4326	0.4951	0.7898	0.4971
Strukturelles Maß	0.0	0.6151	0.6274	0.8623	0.8135
Geometrisches Maß	0.0	0.5289	0.5055	0.7452	0.8661
Differenz-Maß	0.0	0.5255	0.5426	0.7991	0.7257
(1-SSIM)	0.0	0.4751	0.4672	0.7772	0.4563

Abbildung 5.8.: Anwendung der Metrik auf die Region um das Fahrzeug und Vergleich zum SSIM

In einem weiteren Beispiel wurde die Metrik auf eine Region um das Fahrzeug herum angewendet. Die Region hat eine Dimension von 70 x 60 Pixeln. Durch diese Anwendung, soll der Einfluss des Hintergrunds reduziert werden und nur das für die Bildverarbeitung wichtige Fahrzeug betrachtet werden. Bei Auswertung der Ergebnisse fällt auf, dass der SSIM das Fahrzeug in der Szene ohne Nebel näher an der Referenz sieht, als in der Szene mit Nebel. Die neue Metrik sieht mit einer geringen Differenz die Szene mit Nebel näher an der Realität, als die Szene ohne Nebel. Bei Auswertung der „falschen“ Bilder wird von beiden Metriken das Zufallsbild mit der größten Differenz belegt. Das Bild mit Rauschen wird von beiden Metriken näher an der Referenz, als das Zufallsbild gesehen. Der SSIM weist dem Bild mit Rauschen sogar eine größere Ähnlichkeit zu als den simulierten Szenen der realen Testfahrt.

Wie bei Auswertung der Ergebnisse festgestellt werden kann, weist die neue Metrik eine robuste Leistung gegenüber falschen Bildinhalten auf und ist sogar in der Lage kleinere Abweichungen in der Parametrierung der Simulation zu erkennen. Als positiv erweist sich die Einführung der Gradientenbeträge und Gradientenrichtungen. Weiterhin ist der Berechnungsaufwand reduziert gegenüber dem SSIM. Der SSIM dagegen erweist sich nicht robust gegenüber Szenen mit Rauschen. Weiterhin wurde die Szene mit strukturiertem Hintergrund näher zur realen Referenz zugeordnet als die simulierte Szene mit Nebel. Für den Anwendungsfall, der Validierung einer Simulation für die Stimulation von Kamerasystemen ist daher die neue Metrik dem SSIM vorzuziehen.

5.3. Kalibrierung der Simulationsumgebung durch Anwendung der Metrik

Im Folgenden soll die Verwendung der Metrik zur Kalibrierung der Simulation anhand von zwei Beispielen dargestellt werden. Als Voraussetzung dafür wurde die Metrik in die Simulationsumgebung integriert. Die Zielsetzung der Kalibrierung ist es durch automatische Modifikation eines oder mehrerer Parameter die Abweichung der Simulation zur realen Referenz zu reduzieren.

Zur Durchführung der Optimierung wurde die Metrik mit der Kamerasimulation gekoppelt. In Abbildung 5.9 ist der schematische Aufbau dargestellt. Während der Optimierung der Simulationsparameter wird das Distanzmaß der Metrik als zu minimierende Fehlerfunktion betrachtet. Die Differenz kann an einem

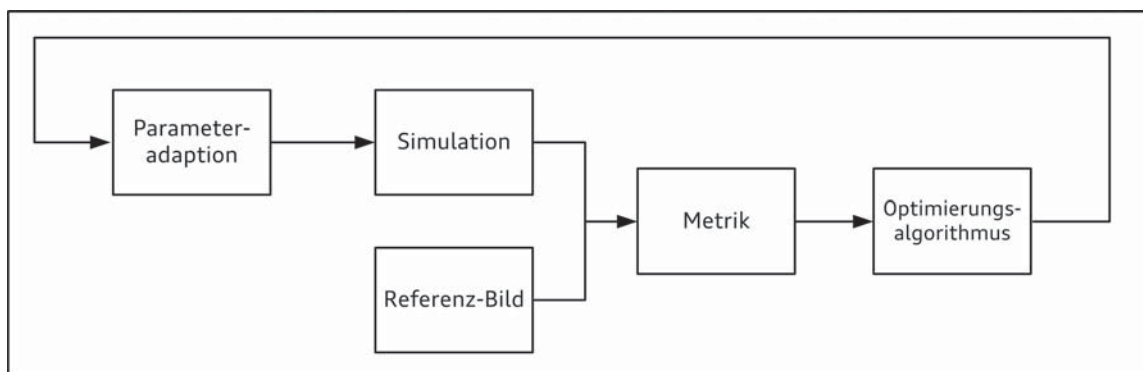


Abbildung 5.9.: Optimierungsalgorithmus

Optimierungsalgorithmus übergeben werden. Dieser führt eine Variation der Simulationsparameter durch. Nach Erreichen des Optimierungsziels oder nach Ablauf der maximal erlaubten Durchläufe werden die Ergebnisse der Optimierung in eine Datei geschrieben.

5.3.1. Integration der Metrik in die Simulationsumgebung

Zur Integration der Metrik in die Simulationsumgebung wurde diese mittels OpenGL-Shadern auf die Grafikkarte portiert.

5.3.1.1. Implementierung

Wie in Abbildung 5.10 ersehen werden kann, wurde der in Abbildung 5.1 dargestellte Ablauf in vier Berechnungsstufen auf der Grafikkarte (GPU) und eine Berechnung auf der CPU aufgeteilt:

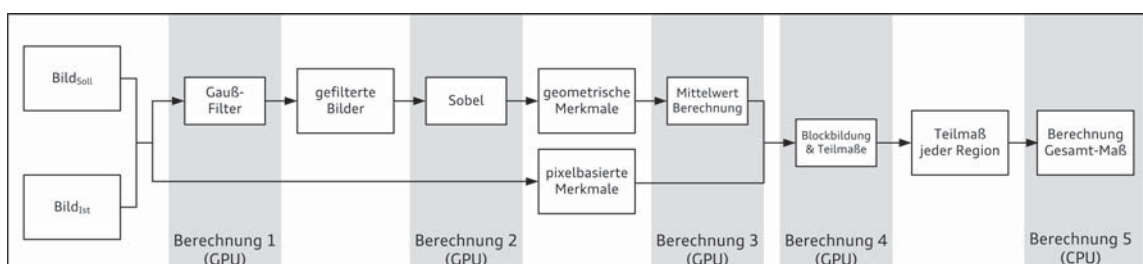


Abbildung 5.10.: GPU-Implementierung

1. Filtern der Eingangs-Bildern mit einem Gauß-Filter zur Reduktion von Rauschen. (GPU)
2. Berechnung der geometrischen Merkmale mittels des Sobel-Filters. (GPU)
3. Berechnung der Mittelwerte der Gradienten für die Unterdrückung von Rauschen. (GPU)
4. Segmentierung der geometrischen Merkmale, Bildung der Vergleichsregionen und Blöcke und Berechnung der Teilmaße für jede Region (GPU)

5. Addition der Teilmaße und Bildung des Gesamt-Maßes. (CPU)

5.3.1.2. Anwendungsbeispiel

Zum Abgleich der Implementierung mit der Matlab-Referenz, wurden die gleichen Bilder wie in Abschnitt 5.2.7 geprüft. Die Berechnungszeit konnte deutlich reduziert werden. In Abbildung 5.11 sind die Ergebnisse der Berechnung und die Abweichung zur Matlab-Implementierung dargestellt. Wie ersehen werden kann







Referenzbild	Selbst-Test	Simulation mit Nebel	Simulation ohne Nebel	Zufallsbild	Rauschen
					
Photometr. Maß	0.0	0.5834	0.6084	0.7292	0.7237
Strukturelles Maß	0.0	0.5512	0.5651	0.7290	0.9155
Geometrisches Maß	0.0	0.4920	0.5066	0.6676	0.8660
Differenz-Maß (GPU)	0.0	0.5420	0.5600	0.7086	0.8351
Differenz-Maß (Matlab)	0.0	0.5422	0.5600	0.7079	0.8351
Abweichung	0.0	0.0002	0.000	0.0007	0.000

Abbildung 5.11.: Ergebnisse des Anwendungsbeispiels zur Berechnung auf der Grafikkarte

tritt bei der Berechnung auf GPU im Szenario mit Nebel und dem Zufallsbild eine geringe Abweichung zur Matlab-Implementierung auf. Bei einem genauen Vergleich der Ergebnisse konnte festgestellt werden, dass diese Abweichungen vorwiegend durch die reduzierte Gleitkommagenauigkeit der Grafikkarte bei der Rundung verursacht werden. Durch die Summenbildung des Ergebnisses der Metrik verstärkt sich dieser Effekt. Dennoch konnte nachgewiesen werden, dass die Anwendung der Metrik auf der Grafikkarte möglich ist und verlässliche Ergebnis erreicht werden.

5.3.2. Verfahren zur Optimierung

Zur Lösung eines nichtlinearen Optimierungsproblems existieren eine Vielzahl von Algorithmen. In diesem Anwendungsbeispiel wurde der Sintflut-Algorithmus nach Dück [56] zur Demonstration der Anwendbarkeit der Metrik als Fehlerfunktion für eine Optimierung verwendet. Bei dieser Optimierungsmethode wird eine Modifikation der Parameterwerte als zielführend erachtet, wenn der Wert der Zielfunktionen unter dem aktuellen Grenzwert w liegt.

In diesem Beispiel ist die Zielfunktion die Metrik und es gilt die Distanz zwischen den Bildpaaren zu minimieren. Bei der Optimierung gilt die Bedingung, dass $M < w$ sein muss und keine lokalen Verschlechterungen der Lösung zugelassen werden.

Der Ablauf gestaltet sich wie folgt:

1. Wähle einen Startpunkt für den zu optimierenden Parameter p_0 aus dem gültigen Parameterraum \mathbf{P} .

2. Wähle einen zufällige Variation des Parameters p_1 in der Nachbarschaft von p_0 .
3. Falls $M(p_1) \leq w$, wähle p_1 sonst behalte p_0 und fahre mit (2) fort.

5.3.3. Anwendungsbeispiel

Zur Demonstration der Fähigkeiten des Differenzmaßes wurden in zwei Versuchen die Parameter der Straße und die Farbe des vorausfahrenden Fahrzeugs anhand des Referenz-Bilds der realen Testfahrt optimiert.

5.3.3.1. Versuchsaufbau

In der ersten Stufe wurde eine Optimierung der Parameter der Straßen-BRDF vorgenommen. Die Schlick-BRDF besitzt insgesamt vier Parameter. Der erste Parameter dient zur Skalierung der Texturfarbe. Die Parameter C_λ , r und p gehören zur normalen Schlick-BRDF, vgl. Abschnitt 4.4.3.2. Jeder der vier Para-

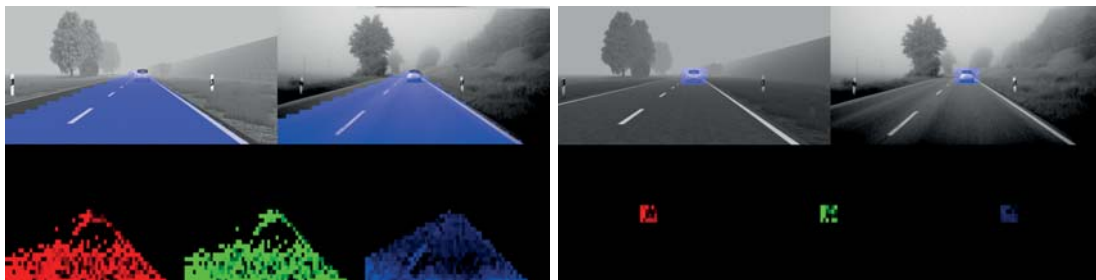


Abbildung 5.12.: Links: Maske zur Optimierung der Straße, Rechts: Maske zur Optimierung der Fahrzeugfarbe
 rot = strukturelle Differenz, grün = geometrische Differenz, blau = photometrische Differenz

meter wurde parallel optimiert. Zur Beschleunigung der Berechnung wurde eine Maskierung der Straße vorgenommen, siehe Abbildung 5.12. Im unteren Bereich der Abbildung sind die Differenzen der Teilmaße visualisiert: rot = strukturelle Abweichung, grün = geometrische Abweichung und blau = photometrische Abweichung. Das gleiche Vorgehen wurde auch zur Bestimmung der optimalen Lackfarbe gewählt, siehe rechte Seite der Abbildung 5.12. Die Optimierung der Fahrzeugfarbe wurde getrennt von der Straßenoberfläche durchgeführt. Als Startwerte wurden folgende Festlegungen getroffen:

Parameter-Name	Startwert
<i>Kontrastfaktor</i>	1.0
<i>Schlick – BRDF : C_λ</i>	1.0
<i>Schlick – BRDF : r</i>	0.1
<i>Schlick – BRDF : p</i>	0.1
<i>Fahrzeugfarbe</i>	0.0

Tabelle 5.3.: Startwerte der Optimierung

5.3.3.2. Ergebnisse der Optimierung

In Abbildung 5.13 und 5.14 sind die Ergebnisse der Optimierung der Straße und des Fahrzeugs dargestellt. Im linken Bereich der Abbildungen sind die Ergebnisse des Distanzmaßes über jede Permutation des Parameterwerts im Vergleich zum aktuellen Referenz-Maß, dargestellt. Im rechten Teil sind die optimalen Werte der Parameter über die Permutation dargestellt.

Parameter des Straßenmodells

Wie in Abbildung 5.13 ersehen werden kann, besitzt die Basisparametrierung des Straßenmodells eine Differenz von 0.5961. Nach 11 Permutationen kann diese auf 0.5870 reduziert werden. Nach 72 Permutationen beträgt die Differenz 0.5632. Nach 433 Permutationen wird der beste Kandidat des Optimierungslaufs gefunden. In Tabelle 5.4 ist das Resultat den Startwerten gegenüber gestellt.

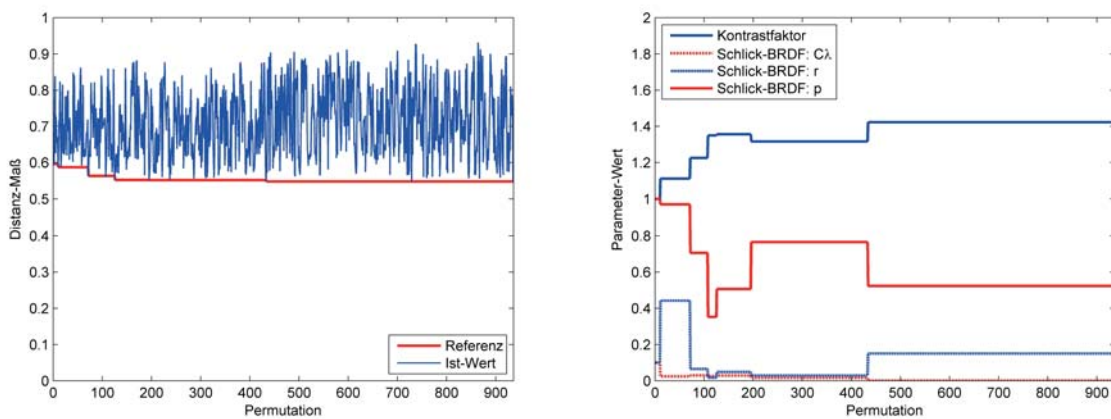


Abbildung 5.13.: Links: Entwicklung des Distanzmaßes über Permutation, Rechts: Werte der Parameter des Straßenmodells über Permutation

Parameter-Name	Startwert	Endwert
<i>Kontrastfaktor</i>	1.0	1.42331
<i>Schlick – BRDF : C_λ</i>	1.0	0.000374
<i>Schlick – BRDF : r</i>	0.1	0.150271
<i>Schlick – BRDF : p</i>	0.1	0.522973

Tabelle 5.4.: Ergebnisse der Optimierung der Straßenparameter

Der Kontrastfaktor des Straßenbelags wurde um 0.42331 erhöht. Hingegen wurden der Fresnel-Faktor der Straße C_λ gesenkt. Die Rauheit der Straße r und p wurden erhöht.

Fahrzeugfarbe

Das Ergebnis der Optimierung der Fahrzeugfarbe ist in Abbildung 5.14 dargestellt. Nach fünf Optimierungsschritten reduziert sich die Distanz des Fahrzeugs von 0.5950 auf 0.5584 mit einem Grauwert von 0.2974. In der sechsten Permutation wird eine Fahrzeugfarbe von 0.4994 erzeugt, welche zu einer Distanz

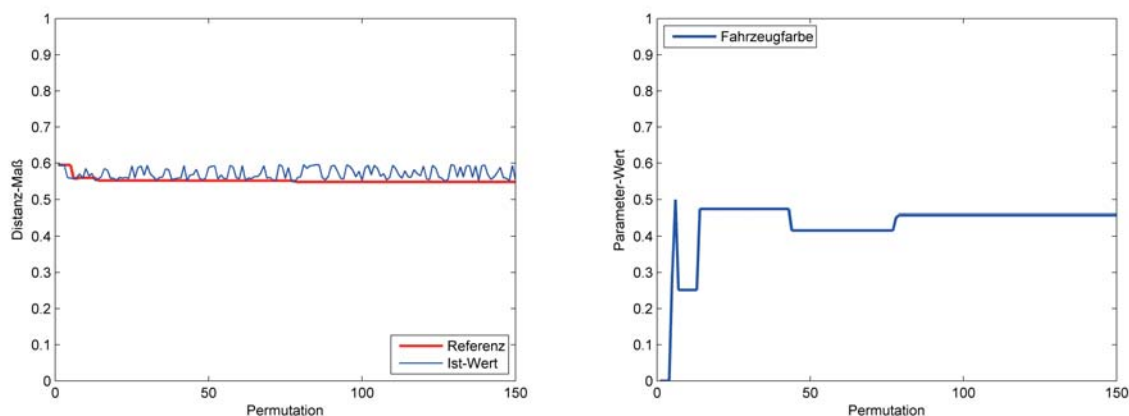


Abbildung 5.14.: Links: Entwicklung des Distanzmaßes über Permutation, Rechts: Wert des Parameters der Fahrzeugfarbe über Permutation

von 0.5572 führt. Diese wird von einem Grauwert von 0.2510 gefolgt mit einer Distanz von 0.5596. Bei der 14. Permutation wird eine Fahrzeugfarbe von 0.4737 erzeugt. Nach 30 weiteren Permutationen ergibt eine Fahrzeugfarbe von 0.4158, eine Differenz von 0.5517. Die finale Parametrierung von 0.4578 mit einer Distanz von 0.5483 wird nach den 78 Permutationen gefunden.

5.3.3.3. Auswertung

In Abbildung 5.15 ist das Ergebnis der Startparametrierung der Straße dem Ergebnis der Optimierung der Straße gegenübergestellt. Bei einem Vergleich der Straßenoberfläche kann festgestellt werden, dass sich diese deutlich an die reale Referenz angenähert hat. Die helle Reflexion im oberen Bereich der Straße wurde deutlich reduziert. Weiterhin kann festgestellt werden, dass durch die Anpassung der Helligkeit des Fahrzeugs eine höhere Übereinstimmung erzielt wird. Bei Betrachtung der Bilder der Teilmaße ist ersichtlich dass, mittels der Optimierungen der strukturelle und geometrische Fehler der Straße reduziert werden konnte. Weiterhin wurde die photometrische Abweichung deutlich reduziert. Die Gesamt-Differenz konnte von 0.6189 auf 0.5355 reduziert werden. Als Gründe für die hohe geometrische Abweichung kann gesehen werden, dass Abweichungen in der Geometrie der Szene auftreten. Weiterhin werden durch das aktivierte Normal-Mapping eine Vielzahl von Konturen in der Straße erzeugt, welche in dieser Feinheit in der Realität nicht vorlagen.



Abbildung 5.15.: Vergleich der Parametersätze der Straße

Links: Startparametrierung, Rechts: Optimierte Parametrierung der Straße und des Fahrzeugs

rot = strukturelle Differenz, grün = geometrische Differenz, blau = photometrische Differenz

5.4. Zusammenfassung

In diesem Abschnitt wurde eine Metrik für die Validierung und die Kalibrierung einer Simulationsumgebung entwickelt. Als Referenz für die Validierung der Simulation wird ein reales Bild verwendet. Die Metrik setzt sich aus drei Teilmaßen zusammen, welche die photometrische, geometrische und strukturelle Differenz der Simulation zur Referenz beschreiben. Die photometrische Differenz beschreibt die Abweichung der Pixelintensitäten, die geometrische Differenz die Abweichung der Richtung der Kantenpunkte und die strukturelle Differenz den Betrag der Gradienten der Kantenpunkte. Zur Steigerung der Robustheit wird kein Pixel-zu-Pixel Vergleich durchgeführt. Als Maße werden die Varianz und der Mittelwert aus Regionen verglichen.

Die Metrik wurde in Matlab umgesetzt. Die Leistungsfähigkeit wurde anhand eines Vergleichs mit der Structural Similarity Index (SSIM) [189] untersucht. Als Stichprobe wurde eine reale Szene, eine simulierte Szene mit hoher Ähnlichkeit, eine simulierte Szene mit gezielter Abweichung, ein Zufallsbild und ein Bild mit Rauschen betrachtet. In den Ergebnissen konnte festgestellt werden, dass die neue Metrik dem SSIM überlegen ist. Besonders bei der Unterscheidung des Zufalls- und des Rauschbilds ist die neue Metrik in der Lage diese effektiv zu differenzieren. Die Distanzen sind mindestens drei Mal so hoch wie bei Anwendung der SSIM.

Abschließend wurde die Metrik beispielhaft als zu minimierende Fehlerfunktionen für die Optimierung von Simulationsparametern verwendet. Als Optimierungsalgorithmus wurde der Sintflut-Algorithmus von Dück [56] verwendet. Im Anwendungsbeispiel wurde die Optimierung der Straßenparameter einer Schlick-BRDF, vgl. Kapitel 4.4.3.2, und des Straßenkontrasts vorgenommen. In einem weiteren Experiment wurde die Farbe des Fahrzeugs automatisiert angepasst. Als Ergebnis kann festgestellt werden, dass die Metrik zur Anpassung von Simulationsparameter verwendet werden kann.

Zusammenfassend kann daher festgestellt werden, dass die neue Metrik sich für den definierten Anwendungsfall eignet. Weiterhin könnte die Metrik für die Untersuchung der Eigenschaften von erkannten Objekten einer Objekterkennung verwendet werden.

6. Untersuchung der Anwendbarkeit einer Kamerasimulation im Entwicklungsprozess

Ziel 5: Experimenteller Nachweis zur Anwendbarkeit von synthetischen Bilddaten.

Zum Nachweis der Anwendbarkeit von synthetischen Bildern wurden zwei Anwendungsfälle betrachtet. Im ersten Anwendungsfall wurde eine Bildverarbeitungsfunktion mit synthetischen Bilddaten entwickelt. Die Erprobungsmöglichkeiten mit synthetischen Bilddaten wurden im zweiten Anwendungsfall auf Basis der Signalverarbeitungskette untersucht.

6.1. Entwicklung eines Fahrzeugerkennungssystems

In der Diplomarbeit von Naal [117] wurden die Möglichkeiten zur Verwendung einer Simulationsumgebung für die Entwicklung eines kamerabasierten Fahrzeugklassifikators untersucht. Der Klassifikator basiert auf dem Ansatz nach Viola und Jones [185] und verwendet zur Erkennung von Objekten eine Haar-Wavelet basierte Klassifikatorkaskade. Jede Stufe der Kaskade dient als Filter zur Verfeinerung der getroffenen

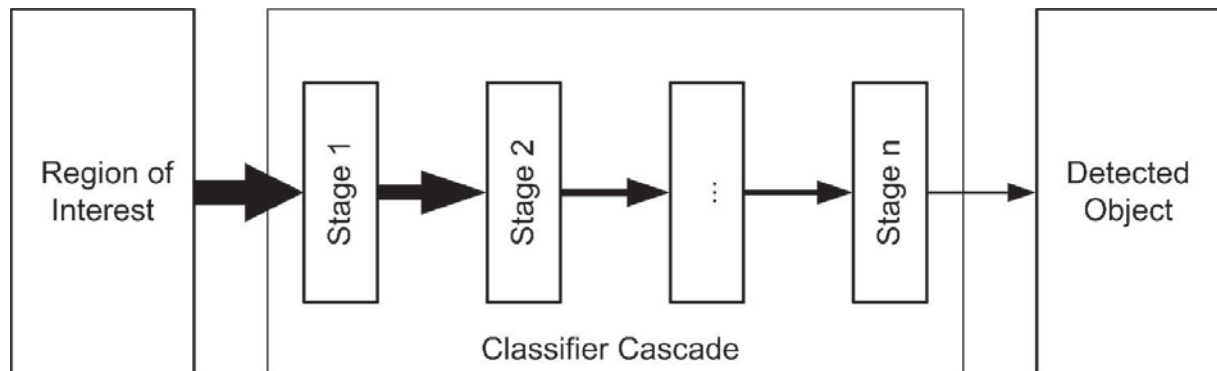


Abbildung 6.1.: Klassifikatorkaskade nach Viola und Jones [185]

Hypothesen. In der ersten Stufe wird ein schwacher Klassifikator verwendet, die Komplexität des Klassifikators erhöht sich mit der Stufe. Nach der Theorie bleibt am Ende nur ein Fahrzeug übrig. Dieser Sachverhalt ist in Abbildung 6.1 visualisiert. Der Klassifikator erlernt die notwendigen Merkmale zur Unterscheidung von Fahrzeugen und Nichtfahrzeugen mittels maschineller Lernverfahren, anhand bekannter Trainingsbilder. In der Diplomarbeit von Naal wurde der Klassifikator mit synthetischen Heckansichten der Fahrzeuge trainiert. Diese Trainingsdaten wurden automatisiert aus der Simulation extrahiert. Das Ziel war es bei der Generierung der Daten einen großen Merkmalsraum abzudecken. Dies wurde durch

eine Variation der Kameraperspektive, die Modifikation der Helligkeit und des Kontrasts sowie einer Veränderung der Szenen und Fahrzeuge erreicht. Die verwendeten Datensätze wurden in Abschnitt 4.6.4 vorgestellt. Als negative Daten wurden Bilder aus öffentlichen Datenbanken und Bilder realer Testfahrten ohne Fahrzeuge verwendet. Zur Stabilisierung der Erkennungsleistung wurden ein Kalman-Filter und ein Clustering-Verfahren angewendet.

Als Maße zur Bewertung der Leistungsfähigkeit des Klassifikators werden die True-Positive $Rate_{TP}$ und False-Negative Rate $Rate_{FP}$ verwendet. Da aus Sicht der Bildverarbeitung vor allem True-Positives, d.h. richtig erkannte Fahrzeuge, und False-Positives, d.h. fehlerhaft als Fahrzeug erkannte Muster, von Wichtigkeit sind. In [117] werden die Maße wie folgt definiert:

$$Rate_{TP} = \frac{N_{TP}}{N_{Fahrzeuge}} \quad (6.1)$$

$$Rate_{FP} = \frac{N_{FP}}{N_{FP} + N_{TP}} \quad (6.2)$$

Dabei sind N_{TP} die Anzahl der richtig erkannten Fahrzeuge, $N_{Fahrzeuge}$ die Summe der zu erkennenden Fahrzeuge und N_{FP} die Anzahl der fälschlicherweise als Fahrzeug erkannten Muster. Zur Bewertung der Erkennungsgenauigkeit des Klassifikators wird die Überdeckung zwischen der getroffenen Hypothese des Klassifikators und dem Referenz-Label des Fahrzeugs bestimmt. Als Label wird eine automatische und manuelle Referenzmarkierung um das zu erkennende Objekt eingezeichnet. Die Erkennungsgenauigkeit folgt aus der normierten Überdeckung der generierten Region durch den Klassifikator \mathbf{A}_{ROI} und dem Label \mathbf{A}_{Label} :

$$p = \frac{\mathbf{A}_{ROI} \cap \mathbf{A}_{Label}}{\mathbf{A}_{ROI} \cup \mathbf{A}_{Label}} \quad (6.3)$$

Dieses Maß wird in der Literatur als Pascal-Maß bezeichnet [64]. Wenn das Pascal-Maß größer als 50% ist wird das Objekt als True-Positive gewertet, ansonsten wird die Region als False-Positive gewertet.

6.1.1. Evaluierung

Durch Naal wurde eine Evaluierung der Erkennungsleistung mittels 305 Aufnahmen realer Testfahrten bei verschiedenen Witterungen und Straßenklassen durchgeführt. Es wurden Szenen der Wetterbedingungen, optimale Lichtverhältnisse (diffuses Licht), schwierige Lichtverhältnisse (Gegenlicht & Schlagschatten), Regen, Schnee, Bewölkung und Sonnenschein verwendet. Weiterhin wurde der Klassifikator in 13 simulierten Szenen erprobt. Die Ergebnisse der Evaluierung sind in Abbildung 6.2 abgebildet. Wie ersehen werden kann werden die besten Ergebnisse in Szenen mit diffusem Licht, z.B. bei bewölktem Himmel, erreicht. Ebenfalls sind die Ergebnisse bei Sonnenschein sehr passabel. Bei einem Vergleich mit den Ergebnissen aus der Simulation fällt auf, dass die Erkennungsleistung in der Simulation sehr hoch ist. In diesen Bildern lagen weder Rauschen noch sonstige Störeffekte vor. Daher entsprechen die simulierten

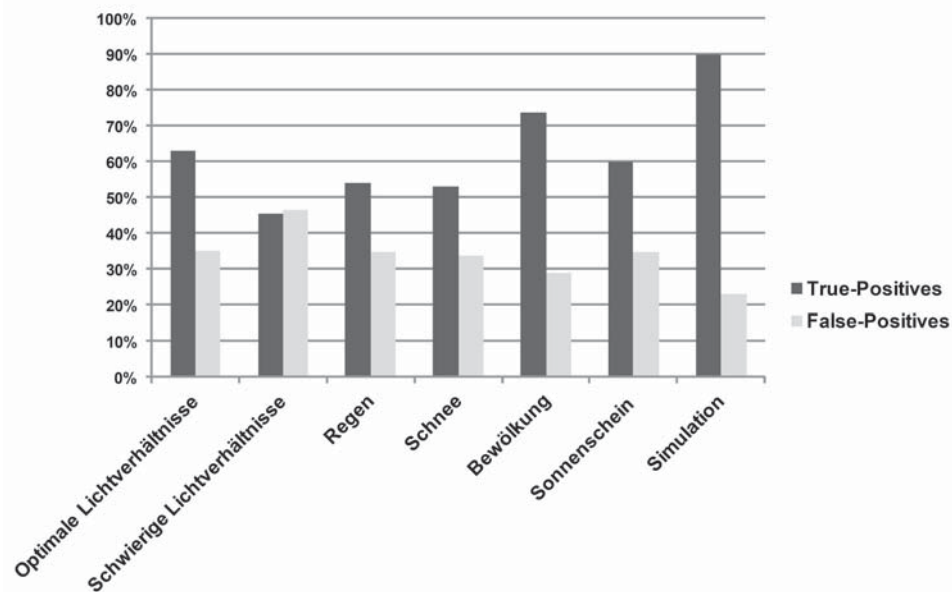


Abbildung 6.2.: Ergebnis der Evaluierung der Fahrzeugklassifikators unter verschiedenen Witterungsbedingungen

Szenen von der Merkmalsverteilung Szenen mit diffusem Licht. Eine Szene mit diffusem Licht kann daher gut mit lokaler Beleuchtungsrechnung approximiert werden. Bei Szenen mit Regen oder Schnee nimmt durch die starken Störeffekte die Erkennungsleistung deutlich ab. Ähnlich verhält es sich bei Szenen mit Gegenlicht, in welchem der Kontrast im Bild deutlich reduziert ist. Naal führt in seiner Arbeit weiter an, dass ein großer Teil der False-Positives in den realen Szenen auf ungenaue Label zurückzuführen war [117]. Die vom Klassifikator ermittelten Regionen waren zu groß oder zu klein. Durch Anwendung des Pascal-Maßes wurden diese False-Positives zugeordnet. Die hohen Fehlererkennungsraten unter schwierigen Wetterbedingungen, konnten auf einen niedrigen Kontrast im Bild, z.B. durch Regen, Schnee oder Gischt, zurückgeführt werden. Durch die Gischt der Fahrzeuge wurden weiterhin die Räder oder die Fahrzeugunterkanten verdeckt. Abschließend stellt Naal fest, dass der entwickelte Klassifikator bei diffusen Lichtbedingungen hohe Detektionsraten erzielt. Damit kommt er zu der Schlussfolgerung, dass simulierte Fahrzeuge in der Computergrafik ähnliche Merkmale wie in der Realität bei diffusen Lichtbedingungen aufweisen.

6.1.2. Zusammenfassung

Naal konnte im Rahmen seiner Diplomarbeit nachweisen, dass es möglich ist mittels synthetischer Bilddaten einen erscheinungsbasierten Klassifikator für die Fahrzeugerkennung zu trainieren. Die Trainingsdaten wurden mit Virtual Test Drive erzeugt. Bei der Untersuchung der Leistungsfähigkeit zeigte sich, dass diese unter diffusen Lichtbedingungen am besten ist. Durch einen Test des Klassifikators in der Simulation konnte gezeigt werden, dass die Erkennungsrate in Szenen mit diffusem Licht und in der Simulation korrelieren. Daher wurde durch Naal nachgewiesen, dass die Anwendung einer einfachen lokalen Beleuch-

tungsberechnung in der Simulation einer realen Testfahrt unter diffusen Lichtbedingungen hinsichtlich der Merkmalsverteilung entspricht.

6.2. Untersuchung des Verhaltens der Signalverarbeitungskette in der Simulation

Zur Bewertung der Anwendbarkeit von simulierten Bilddaten für die Stimulation der Bildverarbeitung wurde eine Untersuchung an verschiedenen Stufen der Signalverarbeitung, vgl. Abschnitt 2.3, durchgeführt. Es wurden die folgenden Schritte der Signalverarbeitung betrachtet:

- Verhalten der Kundenfunktion Audi Active Lane Assist [19]
- Leistungsfähigkeit einer Spurkennungs- und Fahrzeuigerkennungsfunktion
- Vergleich der Hypothesenverifikation der Fahrzeuigerkennung
- Vergleich der Hypothesengenerierung der Fahrzeuigerkennung

Als Grundlage wurden reale Testfahrten in der Simulation nachgebildet und die Ausgangsgrößen der Systeme in der Simulation mit den realen Messdaten verglichen.

6.2.1. Vergleich der Kundenfunktion Active Lane Assist

Durch Aspeleiter [18] wurde das Verhalten des Active Lane Assist Systems an einem Hardware-in-the-Loop Prüfstand mit den Ergebnissen einer realen Testfahrt verglichen. Als Grundlage wurde ein von der National Highway Traffic Safety Administration (NHTSA) festgelegtes Fahrmanöver verwendet, vgl. Abbildung 6.3 aus [1].

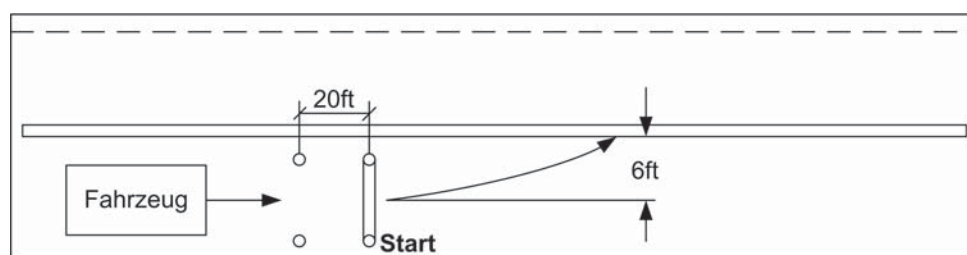


Abbildung 6.3.: Testaufbau für den Test eines Lane Departure Warnings

Zur Durchführung des Tests wird das Fahrzeug auf eine Geschwindigkeit von $v = 45\text{mph}$ beschleunigt. Weiterhin bewegt sich das Testobjekt entlang einer geraden Straßenmarkierung mit einem Abstand von $d = 6\text{ft}$, zur Fahrzeugsachse. Nach dem Erreichen einer bestimmten Markierung muss sich das Fahrzeug mit einer definierten Gierrate der Markierung nähern. Anschließend wird der Abstand zur Markierung, bei welchem die Spurhalteunterstützung einen Lenkeingriff ausführt, ausgewertet. Nach Ausführung dieses Eingriffs ist das Testszenario beendet.

6.2.1.1. Ergebnisse

Zur Auswertung des Systemverhaltens standen zehn Datensätze realer Testfahrten zur Verfügung. Als Vergleichsmaße wurden der Abstand zur Spurmarkierung, die Zeit bis Überschreiten der Markierung, die Gierrate und der Status der Spurhalteunterstützung aufgezeichnet. Die Auswertung wurde auf Grundlage des Mittelwerts und der Standardabweichung für den Abstand zur Markierung, der Gierrate und der Zeit bis zur Überschreitung der Linie durchgeführt. Die Messungen wurden mittels des Eingriffszeitpunkts der Spurhalteunterstützung synchronisiert und die Zeitbasen angeglichen.

In Abbildung 6.4 sind die Ergebnisse der realen Testfahrt und der drei simulierten Versuche dargestellt. Der Mittelwert der realen Messungen wurde als dicke schwarze Linie und die Standardabweichung als gestrichelte Linie eingezeichnet.

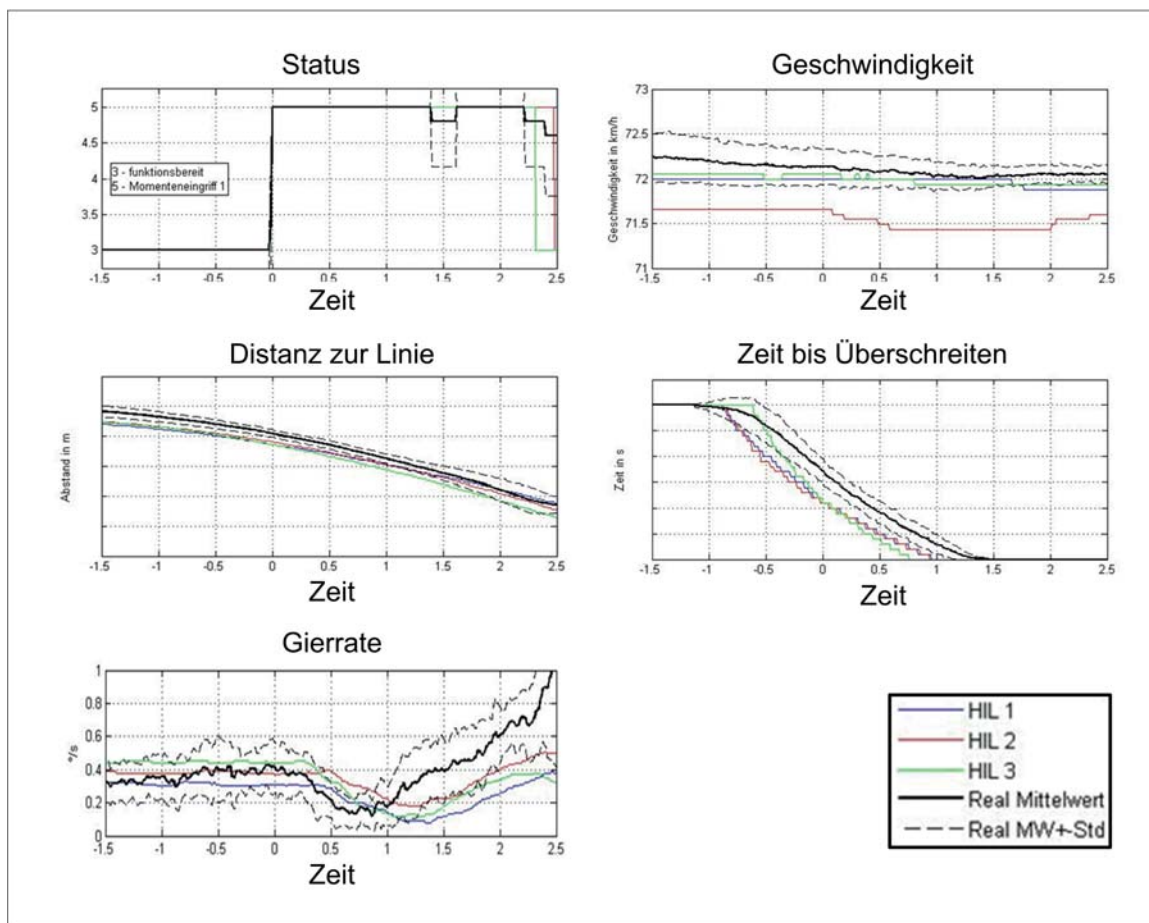


Abbildung 6.4.: Auswertung des NHTSA-Versuchs [18]

6.2.1.2. Auswertung

Beim Vergleich der Geschwindigkeit des Fahrzeugs in der Realität und der Simulation kann festgestellt werden, dass die Geschwindigkeit während des ersten und dritten Testlaufs innerhalb der Schwankungsbreite der realen Testfahrt liegt und deutlich konstanter ist. Ein ähnliches Bild kann auch bei Betrachtung der Gierrate festgestellt werden. Der Verlauf ist deutlich konstanter als in der Realität. Bei Vergleich der Distanz zur Linie und der Zeit zum Überschreiten der Markierung kann festgestellt werden, dass in der Simulation von Beginn an der Abstand zur Linie geringer ist als in der Realität. Dieser befindet sich bei jedem Test an der unteren Grenze der Schwankungsbreite. Im Vergleich zur Realität ist die Schwankungsbreite in der Simulation verschwindend klein. Aus dem geringeren Abstand zur Markierung folgt entsprechend auch eine geringere Zeit bis zur Überschreitung der Markierung. Der Phasenversatz zwischen Simulation und dem Mittelwert der realen Szene entspricht in etwa 700 ms.

6.2.1.3. Zusammenfassung

Durch die Untersuchungen von Aspeleiter konnte gezeigt werden, dass das Systemverhalten einer Spurhalteunterstützung im NHTSA-Szenario ähnlich dem Verhalten bei einer realen Testfahrt ist. Durch die Laborbedingungen und die Reproduzierbarkeit der Szenarien konnte die Schwankungsbreite bei der Testdurchführung deutlich reduziert werden. Als Grund für die Phasenverschiebung bzgl. der Zeit bis zum Überschreiten der Spurmarkierung kann der geringere Abstand zur Linie bei Beginn des Szenarios herangezogen werden. Das Verhalten einer Spurunterstützung kann somit unter Laborbedingungen untersucht werden.

6.2.2. Vergleich der Spur- und Fahrzeugerkennungsfunktion

In [125] und [122] wurden die Ausgangsgrößen einer Spur- und Fahrzeugerkennungsfunktion in der Simulation durch den Abgleich mit realen Testfahrten untersucht. Die betrachteten Szenarien sind mit der realen Referenz in Abbildung 6.5 dargestellt. Die Szenarien wurden mittels der in Abschnitt 4.6 beschriebenen Technik erstellt und parametrisiert.

6.2.2.1. Ergebnisse

Als Messwerte wurden die geschätzte Spurbreite, die Voraussicht der Spurerkennung, der Abstand zum vorausfahrenden Fahrzeug und die geschätzte Fahrzeugbreite aufgezeichnet. Im Rahmen der Auswertung wurden diese Werte mit den Referenzdaten aus der realen Szene verglichen. Da nicht die Leistungsfähigkeit der Bildverarbeitungssysteme bewertet werden soll, sondern die Qualität der Simulation, wird ein Ähnlichkeitsmaß $s = (ySim/yReal - 1.0)$ gebildet. Dabei sind $ySim$ und $yReal$ die gemessenen Größen in der Simulation und in der Realität. In den Abbildungen wurde das Ähnlichkeitsmaß über die Simulationsdauer in Prozent aufgetragen, da die Dauer der Szenarien zwischen $t = 15s$ und $t = 30s$ schwankte.

6.2. UNTERSUCHUNG DES VERHALTENS DER SIGNALVERARBEITUNGSKETTE IN DER SIMULATION

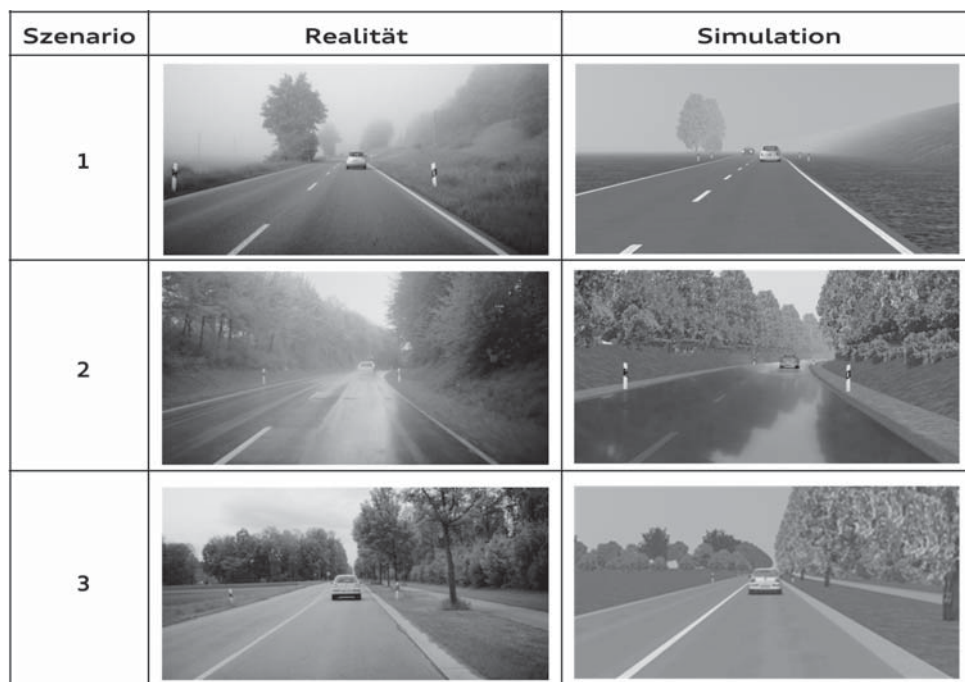


Abbildung 6.5.: Szenarien für die Untersuchung der Ähnlichkeit der Modellgrößen der Bildverarbeitung
Links: Reale Szene, Rechts: Simulation

Spurerkennung

Die Validierung der Simulation für den Test der Spurerkennung wird durch den Vergleich der ermittelten Prädiktionsweite des Straßenverlaufs und der Straßenbreite für die aktuelle Fahrspur durchgeführt. In den Abbildungen 6.6 sind die Ergebnisse des Vergleichs zwischen Simulation und realer Testfahrt dargestellt. Wie in der Abbildung ersehen werden kann, ist die Prädiktionsweite der Spurerkennung einer

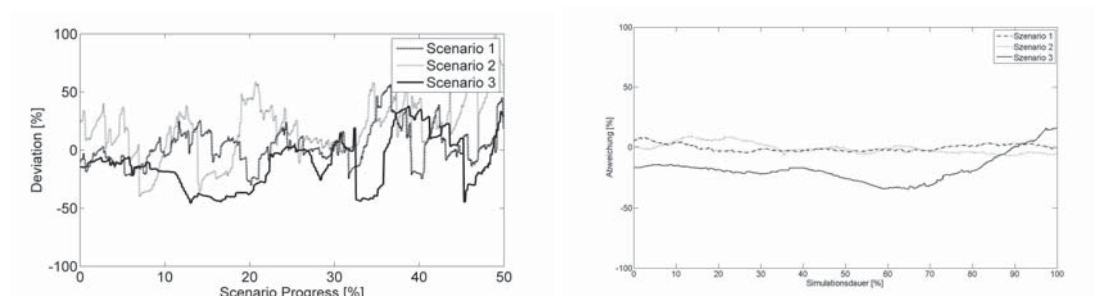


Abbildung 6.6.: Links: Vergleich der Prädiktionsweite des Straßenverlaufs, Rechts: Vergleich der Spurbreite

hohen Schwankungsbreite im Vergleich zur Realität unterworfen. Während des ersten Szenarios beträgt die durchschnittliche Abweichung 3,19% mit einer maximalen Abweichung von 56,48%. Auf der regennassen Straße im zweiten Szenario beträgt die durchschnittliche Abweichung der Prädiktionsweite 18,8% mit einer maximalen Abweichung von 101,4%. Dort stimmen die Positionen und die Abstände der Markierun-

gen nicht genau mit der Realität überein. Im dritten Szenario beträgt die durchschnittliche Abweichung 11,4% mit einer maximalen Abweichung von 44,4%.

Bei Vergleich der Schätzung der Spurbreite kann festgestellt werden, dass diese im ersten und zweiten Szenario eine hohe Abbildungsgenauigkeit von 99,6 % und 99,3 % aufweist. Im dritten Szenario beträgt die Schätzung der Genauigkeit 82,2 % mit einer maximalen Abweichung von 34,5 %.

Fahrzeugerkennung

Für den Vergleich werden die geschätzte relative Position und die Dimension des Fahrzeugs in der Simulation und in der Realität verwendet. In Abbildung 6.7 sind Ergebnisse für die Distanz- und Dimensionschätzung von Szenario eins und drei dargestellt.

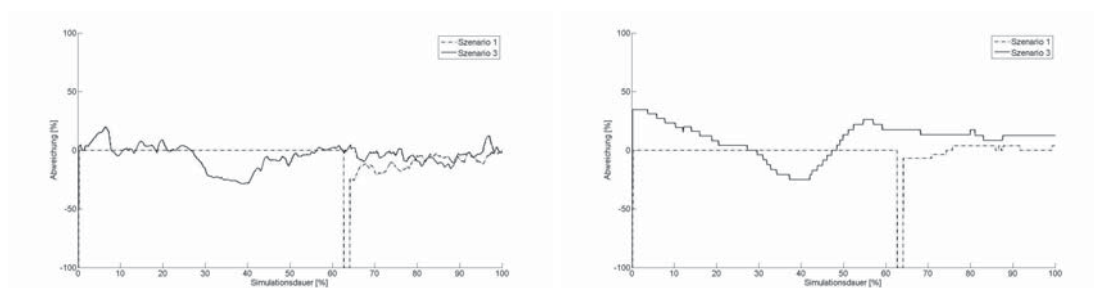


Abbildung 6.7.: Links: Vergleich der Abstandschätzung, Rechts: Vergleich der Fahrzeugbreite

Im ersten Szenario fährt das Messfahrzeug auf ein vorausfahrendes Fahrzeug bei Nebel auf. Bei einem Simulationsfortschritt von 62,7 % wird das Fahrzeug in der Realität detektiert, in der Simulation erst bei 64,1 %. Dieses resultiert in einer kurzzeitigen Abweichung in der Dimensions- und Positionsschätzung. Anschließend beträgt die Übereinstimmung bei der Distanzschätzung durchschnittlich $s = 90,7\%$, mit einem Minimum von 74,1 %. In Szenario drei folgt das Messfahrzeug einem vorausfahrenden Fahrzeug, die durchschnittliche Übereinstimmung mit der realen Szene beträgt $s = 94,9\%$. Bei einem Simulationsfortschritt von 28% bis 43 % fällt die Übereinstimmung auf 71 %. In Abbildung 6.7 kann ersehen werden, dass in diesem Zusammenhang die Abmessungen des Fahrzeugs zu klein geschätzt werden. Im zweiten Szenario wurde in der Realität und in der Simulation das Fahrzeug nie erkannt, daher wurde die Auswertung ausgelassen.

Auswertung

Bei Abgleich der durch die Spurerkennung geschätzten Voraussicht wurde festgestellt, dass die Prädiktionssweite stark schwankt. Als mögliche Ursachen können ein anderes Strich-/Lückenverhältnis der Markierungen und Abweichungen in der Parametrierung der Szene angesehen werden. Im Vergleich dazu lieferte die Schätzung der Spurbreite einen konstanten Fehler, was auf eine Abweichung in der geometrischen Modellierung der Straße hindeutet.

6.2. UNTERSUCHUNG DES VERHALTENS DER SIGNALVERARBEITUNGSKETTE IN DER SIMULATION

Die Fahrzeugdetektion lieferte im ersten Szenario eine leichte Abweichung, da das Fahrzeug in der Simulation später erkannt wird. Als Grund kann eine unterschiedliche Parametrierung der Simulation angesehen werden. Weiterhin wurde im dritten Szenario die Dimension des Fahrzeugs falsch eingeschätzt, daraus resultierte ein Fehler in der Abstandsschätzung zu Beginn des Szenarios.

Zusammenfassung

Im Rahmen dieser Untersuchung konnte gezeigt werden, dass die von einer realen Kamera zurückgelieferten Bussignale eine Übereinstimmung zwischen Simulation und Realität aufweisen.

Zur Durchführung der Tests wurden drei reale Testfahrten bei Nebel, Regen und diffusen Licht in der Simulation umgesetzt. Als Ergebnis kann festgehalten werden, dass die Voraussicht der Spurerkennung eine hohe Schwankungsbreite aufweist, aber die mittlere Abweichung gering war. Die Schätzung der Spurbreite wies ebenfalls eine hohe Ähnlichkeit zur Realität auf. Als komplex erweist sich die Umsetzung eines ähnlichen Verhaltens einer Fahrzeugerkenung in nicht idealen Szenen. Eine identische Parametrierung der Modelle erscheint sehr komplex. Die Genauigkeit der Simulationsumgebung lag in beiden Tests bei über 90%.

Im Rahmen der Tests konnte damit nachgewiesen werden, dass sich eine Simulation zum Abgleich von geschätzten Größen der Bildverarbeitung eignet. Die Ergebnisse sind auf das reale Verhalten übertragbar.

6.2.3. Vergleich der Hypothesenverifikation der Fahrzeugerkenung

In [125] wurde ein Vergleich der Hypothesenverifikation eines Fahrzeugerkenungssystems durchgeführt. Dafür wurde eine Szene einer realen Testfahrt in die Simulation transferiert. In der Szene schließt das Messfahrzeug auf ein vorausfahrendes Fahrzeug auf. Die verwendeten Geschwindigkeits- und Abstandprofile sind in Abschnitt 4.6 dargestellt. Für einen Vergleich der visuellen Qualität der Szene ist ein Screenshot der realen Testfahrt und der Simulation in Abbildung 6.8 dargestellt. In Abbildung 6.9 ist



Abbildung 6.8.: Links: reale Szene, Rechts: simulierte Szene

der verwendete Messaufbau dargestellt. Das aufgezeichnete Video der Testfahrt und das erzeugte Video aus der Simulation werden in das Fahrzeugerkenungssystem eingespeist. Zur Auswertung werden die

erzeugten Objekte des Klassifikators gesichert. Zur Erzeugung eines aussagekräftigen Ergebnisses ist es nicht ausreichend die True- oder False-Positives Rates in beiden Szenen zu vergleichen. Daher wird die

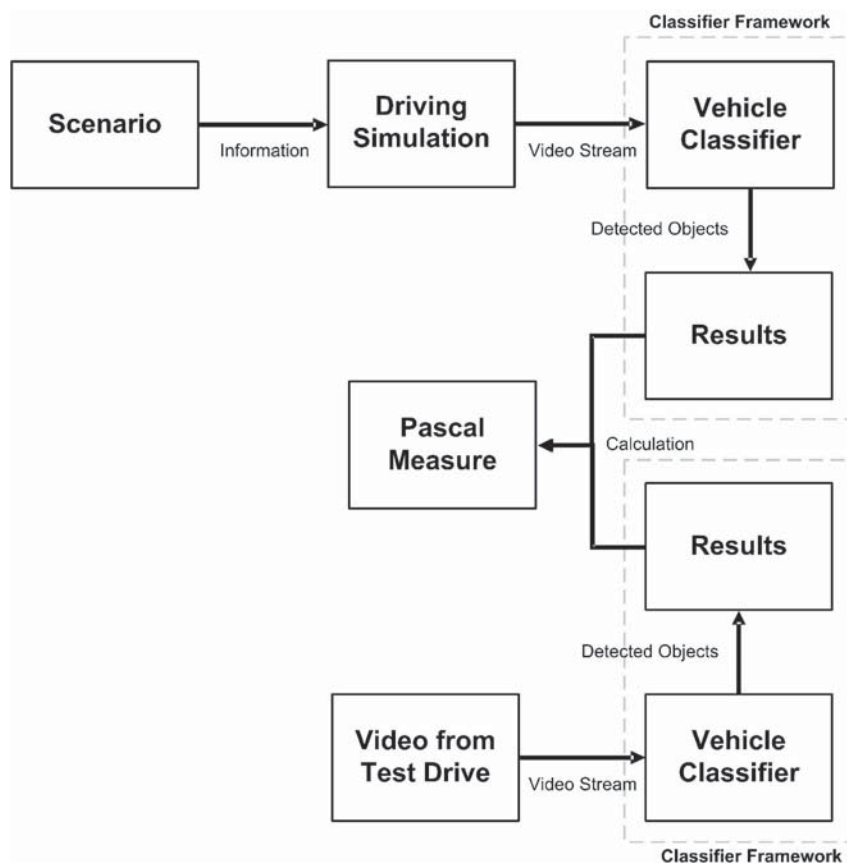


Abbildung 6.9.: Messaufbau für die Auswertung

Überlappung der generierten Regionen miteinander verglichen. Dafür wird das Pascal-Maß [64] bildweise angewendet, vgl. Abschnitt 6.1. In der ursprünglichen Form ist das Maß nur auf eine Region pro Bild anwendbar. Zur Auswertung eines Bilds mit mehreren Regionen wird das Maß als Produkt von allen n einzelnen Maßen definiert:

$$P = \prod_{i=0}^n p_i \quad (6.4)$$

Durch diese Definition gewinnt das Maß an Stärke, da schon bei einer nicht korrekt zugeordneten Region das gesamte für das Frame gültige Maß Null wird.

Ergebnisse

In Abbildung 6.10 ist das Pascal-Maß für den durchgeführten Test aufgetragen. Wie ersehen werden kann gibt es zu Beginn eine hohe Übereinstimmung zwischen der Simulation und realen Szene. Das Pascal-Maß fällt von Frame 401 bis Frame 450 auf Null. Da der Klassifikator in der simulierten Szene

das Fahrzeug bereits erkennt. Im rechten Teil der Abbildung ist eine Vergrößerung der Ergebnisse von Frame 400 bis Frame 570 dargestellt. Die Ausreißer in dem Diagramm können dadurch erklärt werden, dass in der Simulation das Fahrzeug nicht in jedem Frame richtig erkannt wird. Ab Frame 450 wird das Fahrzeug in der Simulation und Realität wahrgenommen. In diesem Zeitraum ist das Pascal-Maß im Schnitt 50% mit einem Maximalwert von 80%. Wengleich einige Fehlererkenntnisse zu einem Abfall des Maßes führen. Zwischen Frame 523 bis 547 gibt es große Abweichungen, das Pascal-Maß erreicht aber einen Maximalwert von 60%. Danach beträgt der Wert 62% und reduziert sich auf 33% bis zum Ende der Simulation.

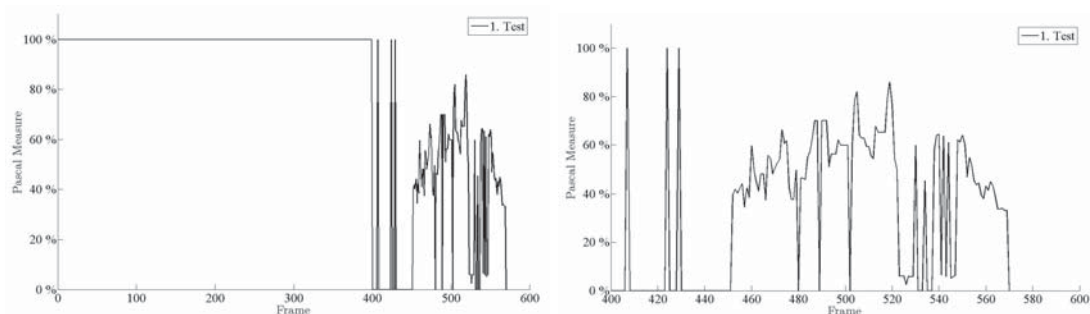


Abbildung 6.10.: Links: Auswertung der gesamten Szene, Rechts: Auswertung ab Frame 420

Auswertung

Im Folgenden sollen interessante Key-Frames für die Bildverarbeitung näher untersucht werden. Daher werden die Grauwerte und Gradientenbeträge in einer Bounding-Box um das Fahrzeug, zwischen Simulation und Realität verglichen, vgl. Abbildung 6.11. Auf der linken Seite sind die realen Grauwerte und Gradientenbeträge dargestellt. Auf rechten Seite ist eine entsprechende Visualisierung für das synthetische Bild dargestellt. In Bild 400 ist der Wert der Gradientenbeträge deutlich geringer als in der Simulation und beinhaltet daher weniger Details. Da sich das Fahrzeug ca. 64,5 m entfernt von der Kamera befindet, ist die Ortsauflösung des Fahrzeugs im Bild deutlich geringer. Damit kann erklärt werden, warum der Klassifikator das Fahrzeug in der Simulation früher erkennt. Bei Bild 450 wird das Fahrzeug auch im realen Video erkannt. Bei diesem Bild besitzen beide Key-Frames eine hohe Ähnlichkeit, dennoch gibt es einige Abweichungen im unteren Bereich des Fahrzeugs. Weiterhin kann ersehen werden, dass das synthetische Fahrzeug ähnliche Gradientenbeträge von Bild 437 bis Bild 537 aufweist.

Ab Frame 537 wird das Fahrzeug in der realen Szene nicht mehr erkannt. Im linken Bereich des realen Grauwerte und Gradientenbetrag-Bilds sind Hintergrundobjekte und ein Scheinwerfer erkennbar. Dieser Scheinwerfer wurde in der Simulation nicht modelliert. Der Lichtkranz führte zu einer Abweichung der Gradientenbeträge zwischen Simulation und Realität. Ab Frame 550 wurde das Fahrzeug wieder erkannt.

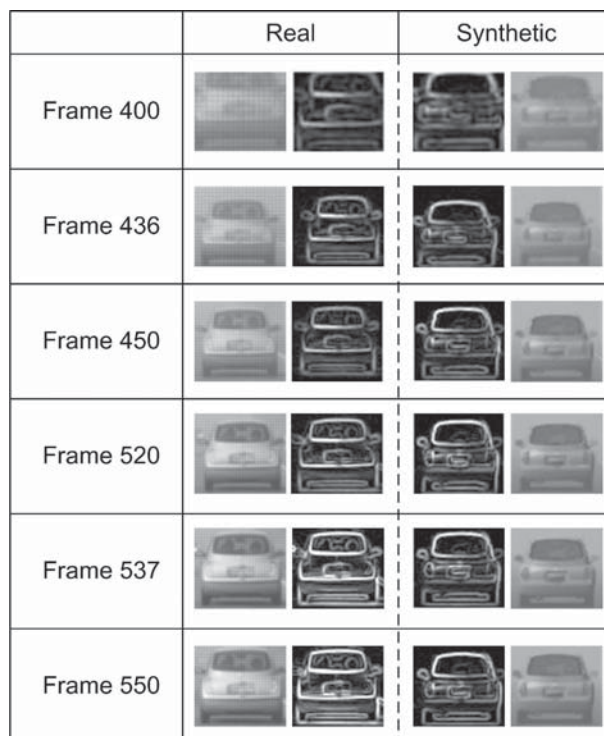


Abbildung 6.11.: Auswertung von Keyframes

Zusammenfassung

Bei diesem Experiment wurde gezeigt, dass bei Anwendung eines Fahrzeugerkennungsalgorithmus auf ein reales und ein synthetisches Video ähnliche Ergebnisse erreicht werden. Bei Betrachtung der Qualität, d.h. der Erkennungsleistung, kann festgestellt werden, dass es eine hohe Übereinstimmung gibt. Anhand der Hintergrundobjekte, z.B. Scheinwerfer, kann der Einfluss von nicht vorhandenen Hintergrunddetails in einer simulierten Szene nachgewiesen werden. Daher ist es naheliegend bei einer weiteren Untersuchung auch den Hintergrund der Szene zu betrachten. Als weiteren Punkt konnte festgestellt werden, dass durch Abweichungen in der Parametrierung der Szene das synthetische Fahrzeug früher als in der Realität erkannt wurde.

6.2.4. Vergleich der Hypothesengenerierung der Fahrzeugerkennung

In [120] wurde der gleiche Klassifikator aus [125] in einer verbesserten Szene getestet. Im Vergleich zur Auswertung der Position des Objekts wurden in diesem Fall die generierten Hypothesen verglichen. Vor Durchführung der Simulation wurde eine manuelle Kalibrierung der Simulation unter Anwendung der Metrik aus Abschnitt 5 durchgeführt. Es wurden die folgenden Parametrierungen und Modelle angepasst: Datum, Uhrzeit, Wetterbedingungen, BRDF der Straße, BRDF des Fahrzeugs, Qualität und Kontrast der Spurmarkierungen und Helligkeit und Kontrast der Vegetation. Die Ergebnisse der Anpassung des








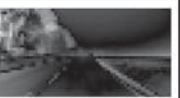








	Gesamt Maß	Strukturelles Maß	Geometrisches Maß	Photometrisches Maß
Anpassung 3	 0.5318	 0.5314	 0.5068	 0.5573
Anpassung 2	 0.5411	 0.5475	 0.5105	 0.5653
Anpassung 1	 0.5779	 0.5931	 0.5516	 0.5888
Basis	 0.6094	 0.6246	 0.5799	 0.6238

Abbildung 6.12.: Ergebnisse der Kalibrierung der Simulation

Testszenarios sind in Abbildung 6.12 dargestellt. Die Differenz ist größer, je heller die Farbe ist. Wie ersehen werden kann konnte durch die Anpassung der Parameter die Distanz der Simulation zur Realität deutlich reduziert werden.

Ergebnisse

Die Auswirkung der verschiedenen Parametrierungen auf die Hypothesengenerierung ist in Tabelle 6.1 dargestellt. Als Ergebnisse werden die Größe der kleinsten wahren Hypothese *KleinsteROI*, die Summe der wahren $\sum TH$ und falschen $\sum FH$ Hypothesen, die Summe aller Hypothesen $\sum H$ und das Verhältnis der wahren an allen Hypothesen betrachtet.

Datensatz	Kleinste ROI	$\sum TH$	$\sum FH$	$\sum H$	$\frac{TH}{H}$ -Verhältnis
<i>Reale Szene</i>	24 x 24	103	337	440	0.2341
Basis	24 x 24	79	6679	6758	0.0012
Anpassung 1	32 x 32	213	3337	3550	0.0650
Anpassung 2	29 x 29	357	3115	3472	0.1028
Anpassung 3	24 x 24	324	790	1114	0.2908

Tabelle 6.1.: Ergebnisse der Hypothesengenerierung in verschiedenen Szenarien

In der realen Szene wurde das Fahrzeug bei einer Größe der Hypothese von 24x24 Pixel erkannt. Insgesamt wurden 440 Hypothesen erzeugt, davon waren 338 wahre und 103 falsche Hypothesen. Das Verhältnis der wahren Hypothesen betrug 0.2318. Im Vergleich dazu werden mit der Basis-Parametrierung 6758 Hypothesen erzeugt, davon waren 79 wahre und 6679 falsche Hypothesen. Es wurden 15-Mal mehr Hypothesen

6.2. UNTERSUCHUNG DES VERHALTENS DER SIGNALVERARBEITUNGSKETTE IN DER SIMULATION

als in der realen Szene erzeugt. Mit dem Parametersatz der dritten Anpassung wurde das Ergebnis der Simulation weiter verbessert. Insgesamt wurden 1114 Hypothesen erzeugt, davon waren 324 wahr und 790 falsch. Damit ergibt sich ein Verhältnis der wahren Hypothesen von 0.2908. Im Vergleich zur realen Szene wurden 2.5 mal mehr Hypothesen auf Basis der Simulationsdaten erzeugt. Die Größe der kleinsten wahren ROI war 24x24 Pixel.

Auswertung

Zur Auswertung werden die erzeugten Hypothesen der realen Szene und der Simulation mit dem besten Parametersatz verglichen. Die richtigen Hypothesen wurden in sechs Klassen und die falschen Hypothesen in zehn Klassen gruppiert. Die wahren Hypothesen sind in Abbildung 6.13 dargestellt. Wie ersehen werden



Abbildung 6.13.: Wahre Hypothesen

kann besteht eine große Ähnlichkeit zwischen den in der realen Szene und der Simulation generierten Hypothesen. Der Bildinhalt und die Erscheinung sind sehr ähnlich. Hinsichtlich der Auswertung sind die falschen Hypothesen von größeren Interesse, vgl. Abbildung 6.14.

	Reale Szene	Simulation
Hypothese 1	7.42%	11.14%
Hypothese 2	0.89%	2.41%
Hypothese 3	48.37%	27.85%
Hypothese 4	0.89%	12.03%
Hypothese 5	5.34%	3.42%
Hypothese 6	0.30%	1.01%
Hypothese 7	11.87%	7.72%
Hypothese 8	7.12%	7.97%
Hypothese 9	1.78%	1.27%
Hypothese 10	8.31%	18.48%
Gesamt Abdeckung	92.28%	93.29%

Tabelle 6.2.: Verteilung der falschen Hypothesen auf die Klassen

In Tabelle 6.2 ist der Beitrag jeder Klasse von falschen Hypothesen an der gesamten Menge der falschen Hypothesen dargestellt. Die so klassifizierten Hypothesen decken in der realen Szene 92.28% und in der

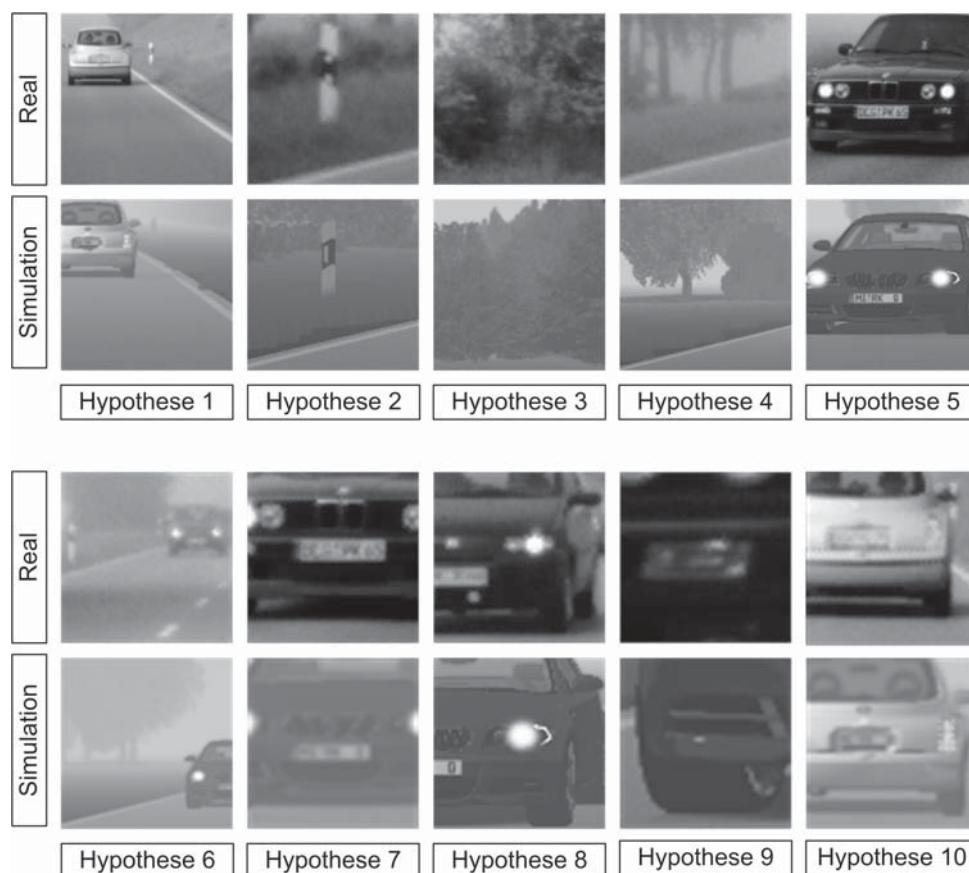


Abbildung 6.14.: Falsche Hypothesen

Simulation 93.29% aller falschen Hypothesen der Szene ab. Bei Betrachtung der Tabelle kann festgestellt werden, dass die Verteilung der Hypothesen unterschiedlich ist. Im Folgenden wird auf die größten Abweichungen eingegangen. In der realen Szene gehört jede zweite Hypothese der dritten Klasse an, während in der Simulation nur jede dritte Hypothese zu dieser Klasse gehört. In der Simulation gehören 12.03% zur vierten Klasse, in der realen Szene sind dies nur 0.89%. Weiterhin werden 18.48% der Hypothesen 10 von der Rückseite des Fahrzeugs in der Simulation erzeugt, in der realen Szene sind dies 8.31%

Als Gründe für die unterschiedliche Verteilung der Hypothesen kann angeführt werden, dass es Unterschiede im Design des Szenarios gibt, z.B. sind in der realen Szene evtl. mehr Büsche am Rand der Straße verteilt. Damit können die Abweichungen für die Hypothesen drei und vier begründet werden. Die Abweichung in Klasse zehn kann damit begründet werden, dass sich die Erscheinung des Fahrzeugs von der realen Referenz unterscheidet.

Zusammenfassung

Dieser Test betrachtet die Hypothesengenerierung in einer simulierten Szene im Vergleich zu einer realen Referenz. Zur Erreichung von ähnlichen Ergebnissen in der Simulation musste eine Optimierung der Parameter durchgeführt werden. In der besten Parametrierung erzeugte der Klassifikator ähnliche Hypo-

thesen wie in der realen Referenzszene. Die Anzahl der in der Simulation erzeugten Hypothesen ist mehr als doppelt so hoch. Zur Auswertung wurden die falschen Hypothesen in der Simulation und Realität in Klassen eingeteilt. Die Verteilung der Hypothesen unterscheidet sich zwischen Simulation und Realität. Abschließend kann festgehalten werden, dass der Klassifikator simulierte Bilder mit ähnlichem Inhalt wie in der Realität als gleiche wahre oder falsche Hypothesen erkennt.

6.3. Zusammenfassung

Die Untersuchung der Anwendbarkeit einer Simulation zur Stimulation der Bildverarbeitung wurde mittels der Entwicklung eines Fahrzeugerkennungsalgorithmus und durch den Abgleich der Ergebnisse von Bildverarbeitungsalgorithmen an verschiedenen Stufen der Signalverarbeitung durchgeführt.

Bei der Validierung des mit simulierten Bilddaten entwickelten Algorithmus zur Fahrzeugdetektion wurde festgestellt, dass die besten Leitungswerte bei bewölktem Himmel oder Sonnenschein erreicht wurden. Diese Ergebnisse korrelieren zu einem Test mit simulierten Bilddaten. Daher kann festgestellt werden, dass die Merkmale in der Simulation sehr ähnlich sind. Bei Szenen mit komplexen Umweltbedingungen nimmt die Leistungsfähigkeit ab. In diesen Szenen ist der Kontrast geringer. Daher schlägt Naal einer Verbesserung der Simulation hinsichtlich komplexer Wetterbedingungen vor.

Die Signalverarbeitungskette wurde an vier Stufen untersucht: Vergleich der Kundenfunktion Active Lane Assist, Vergleich der Spur- und Fahrzeugerkennung, Vergleich der Hypothesenverifikation und Hypothesengenerierung der Fahrzeugerkennung. Bei einem Abgleich der Active Lane Assist Funktion mit einem NHTSA-Versuchsaufbau wurde festgestellt, dass das Systemverhalten einer Spurhalteunterstützung im NHTSA-Szenario, ähnlich dem Verhalten bei einer realen Testfahrt ist. Durch die Laborbedingungen und die Reproduzierbarkeit des Szenarios konnte die Schwankungsbreite bei der Testdurchführung deutlich reduziert werden. Das Verhalten einer Spurunterstützungsfunktion, kann somit unter Laborbedingungen untersucht werden.

Im nächsten Schritt wurden die Ausgangsgrößen einer Spur- und Fahrzeugerkennungsfunktion in drei Szenarien untersucht. Es wurde gezeigt, dass die von einer realen Kamera bereitgestellten Objektinformationen eine hohe Übereinstimmung zwischen Simulation und Realität aufweisen. Als Ergebnis kann festgehalten werden, dass die Voraussicht der Spurerkennung eine hohe Schwankungsbreite aufweist, aber die mittlere Abweichung gering war. Die Schätzung der Spurbreite wies ebenfalls eine hohe Ähnlichkeit zur Realität auf. Als komplex erweist sich die Umsetzung eines ähnlichen Verhaltens einer Fahrzeugerkennung in nicht idealen Szenen, da eine identische Parametrierung der Modelle schwierig erscheint. Die Genauigkeit des Tests betrug über 90%. Im Rahmen dieser Versuche konnte nachgewiesen werden, dass sich die Simulation zum Abgleich von geschätzten Größen der Bildverarbeitung in der Simulation und in der Realität eignet.

Zur weiteren Betrachtung der Objekterkennung auf simulierten Bilddaten wurde die Verifikation und Generierung von Hypothesen der Fahrzeugerkennung näher betrachtet. Dazu wurde die Objektgenerierung der Fahrzeugerkennung in einer Szene mit Nebel validiert. Ähnlich dem Verhalten bei Vergleich der Fahrzeugerkennungsfunktion im vorherigen Abschnitt war eine hohe Übereinstimmung erkennbar. Im

Rahmen des Tests konnte festgestellt werden, dass zur Erreichung von ähnlichen Ergebnissen es notwendig ist, die reale Szene im Detail nachzubilden. Dies wurde durch einen in der Simulation nicht vorhandenen Scheinwerfer eines Fahrzeugs deutlich. Dieser verursachte in der realen Szene Schwierigkeiten bei der Objekterkennung. Weiterhin wurde aufgrund von Abweichungen in der Parametrierung das synthetische Fahrzeug früher als in der Realität erkannt.

Auf Basis der vorher gesammelten Erkenntnisse wurde zur Verifikation der Hypothesengenerierung eine iterative Optimierung der Simulationsparameter durchgeführt. Als Hilfsmittel wurde die in Abschnitt 5 beschriebene Metrik eingesetzt. Durch die Optimierung der Parameter erzeugte der Klassifikator ähnliche Hypothesen wie während der realen Testfahrt. Zur Auswertung wurden die falschen Hypothesen aus der Simulation und aus der Realität in Klassen eingeteilt. Bei dieser Klassifikation wurde weiterhin deutlich, dass durch Unterschiede im Szenen-Design die Hypothesen eine andere Verteilung als in der realen Szene aufweisen. Als Ergebnis kann festgestellt werden, dass der Klassifikator ein ähnliches Klassifikationsergebnis wie in einer realen Szene liefert, weil Bilder mit ähnlichem Inhalt als Fahrzeuge oder nicht Fahrzeuge erkannt wurden.

Zusammenfassend kann festgestellt werden, dass der Aufwand für eine realitätsnahe Stimulation der Bildverarbeitung steigt, je früher in der Signalverarbeitung eine Erprobung stattfinden soll. Bei der Bewertung einer Kundenfunktion oder von durch die Bildverarbeitung geschätzten Objekteigenschaften können mit geringem Aufwand aussagekräftige Ergebnisse erreicht werden. Bei der Entwicklung des Fahrzeugerkennungssystems und den Tests zur Hypothesengenerierung und Hypothesenverifikation zeigte sich, dass die Zusammenstellung der Szene und die genaue Parametrierung der Modelle und der Atmosphäre eine notwendige Voraussetzung für eine erfolgreiche Untersuchung darstellt. Durch Anwendung der Metrik konnten im Rahmen der Hypothesengenerierung ähnliche Ergebnisse wie in der Realität erreicht werden. Abschließend kann festgestellt werden, dass es möglich ist eine Simulationsumgebung in jeder Stufe der Signalverarbeitung für die Entwicklungs- und den Tests von kameraunterstützten Funktionen einzusetzen. Wenngleich der Aufwand für die Simulation und die Parametrierung steigt.

7. Hardware-in-the-Loop Simulator zur Erprobung von Fahrerassistenzsystemen

Ziel 6: Aufbau eines Systems für die Hardware-in-the-Loop Erprobung von Fahrerassistenzsystemen.

Der entwickelte Hardware-in-the-Loop Simulator stellt ein System dar, welcher um eine Simulation des Verkehrsumfelds und entsprechender Sensormodelle für die Stimulation von Fahrerassistenzsysteme eines aktuellen Audi A6 und Audi A8 erweitert wurde. In der jüngeren Vergangenheit und Gegenwart wurden ähnliche Systeme durch andere Automobilhersteller vorgestellt. In [114] ist die Verwendung eines Simulationssystems zur Validierung der Fahrerassistenzfunktionen im Passat CC beschrieben, es werden Kamerabilder, Radarobjekte und Ultraschallobjekte simuliert. In [191] wird ein ähnlicher Ansatz verfolgt, in welchem ebenfalls Kamerabilder, Radarobjekte und Ultraschallobjekte simuliert werden. Durch [197] wird die Verwendung eines HIL-Prüfstands zur Validierung eines Stereokamerasystems beschrieben. Weiterhin wird erwähnt, dass eine Fusion mit Radarobjekten durchgeführt wird. In [21] wird eine Simulationslösung für die Erprobung von kamerabasierten Fahrerassistenzsystemen vorgestellt. Im Vergleich zu den vorher beschriebenen Ansätzen verfolgt das System einen ganzheitlichen Ansatz, d.h. es soll jedes Fahrerassistenz- und aktives Sicherheitssystem mit realitätsnahen Sensordaten aus einem simulierten Fahrzeugumfeld versorgt werden [111] [122].

7.1. Systemaufbau

In Abbildung 7.1 ist die Systemtopologie der virtuellen Erprobungsfahrt mit Signalflüssen zwischen den einzelnen Komponenten dargestellt. Im linken Bereich der Abbildung ist der klassische Hardware-in-the-Loop Prüfstand abgebildet, bestehend aus: Fahrzeugmodell, Fahrermodell, realen und simulierten Aktuatoren und Steuergeräten. Das Fahrzeugmodell stellt den realen Steuergeräten alle notwendigen internen Sensorsignale bereit und verhält sich entsprechend eines realen Fahrzeugs. Die realen Steuergeräte führen gemäß den Eingangsgrößen die hinterlegten Funktionen aus und greifen steuernd oder regelnd auf die realen oder simulierten Aktuatoren zu. Diese beeinflussen das Fahrzeugmodell und schließen die Regelschleife. Zur Steuerung des Fahrzeugs kann ein Fahrermodell über simulierte Mensch-Maschine-Schnittstellen, Lenkrad, Gas-, Kupplungs- und Bremspedal, auf die Aktuatoren einwirken und das Fahrzeug steuern. Die Erweiterungen zur Erprobung von modernen Fahrerassistenzsystemen sind im rechten Bereich der Abbildung dargestellt. Die Grundlage stellt die Verkehrs- und Umfeldsimulation Virtual Test Drive, vgl. Abschnitt 3.3, dar. Das simulierte Erprobungsfahrzeug wird mittels eines in Virtual Test Drive

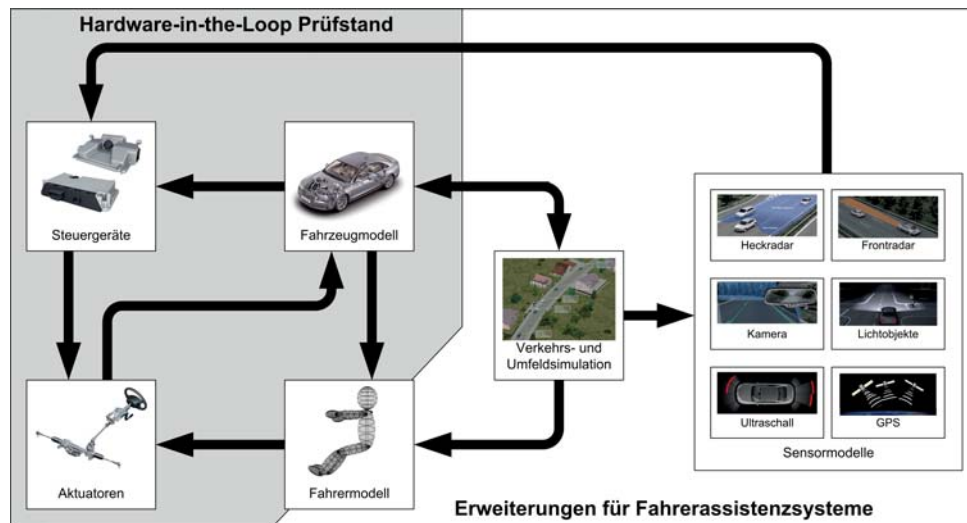


Abbildung 7.1.: Systemtopologie der virtuellen Erprobungsfahrt

vorhandenen Fahrermodells kontrolliert. Der Fahrer ist in der Lage das Fahrzeug entsprechend in Test-szenarien definierter Fahranweisungen reproduzierbar zu steuern. Die Umfeldsimulation ist durch eine echtzeitfähige Schnittstelle mit dem Hardware-in-the-Loop Prüfstand verbunden, als Middleware kommt das ADTF (Automotive Data and Time Triggered Framework) [164] zum Einsatz. Durch die Kommunikationsschnittstelle werden Positions- und Straßendaten zwischen Fahrzeugmodell und Virtual Test Drive ausgetauscht. Zur Stimulation von Sensoren zur Wahrnehmung des Fahrzeugumfelds kommen verschiedenartige Modelle zum Einsatz. Auf Basis einer Simulationsdatenschnittstelle werden beispielsweise Stimuli für ein Radarsystem erzeugt. Durch die ausschließliche Verwendung dieser Schnittstelle als Eingangsgröße für die Generierung aller Sensordaten ist eine Kohärenz der simulierten Messdaten zueinander gewährleistet.

7.2. Stimulation der Fahrzeugkamera

Die Generierung des Kamerabilds wird in Abhängigkeit der Fahrzeugbewegung mit Virtual Test Drive [126] durchgeführt. Die Simulation besitzt durch die hohe geometrische Güte und Qualität der Oberflächenmodelle die Voraussetzungen für die Stimulation der Signalverarbeitung. Die Übertragung der Bilddaten in die Kamera findet auf Basis einer optischen Übertragung statt. Zur Abschirmung der Kamera gegen Streulicht wird die in Abbildung 7.2 dargestellte Dunkelkammer verwendet. Zur Aufnahme der Kamera wurde eine spezielle Halterung entwickelt, welche in einem definierten Abstand zur Projektionsfläche aufweist.

In einem Kraftfahrzeug verbaute Kameras besitzen eine Hyperfokalebene in einem Bereich von $a_0 = 5m - 11m$, d.h. ein Punkt auf dieser Ebene wird scharf auf das Bild abgebildet. Der Abstand zwischen dem Monitor und der Kamera in einer Dunkelkammer beträgt in der Regel zwischen $d = 1m - 2m$. Daher ist es notwendig zur Realisierung einer scharfen Abbildung eine auf die Brennweite der Kamera angepasste

Korrekturlinse zu verwenden. Eine schnelle Bestimmung der Brennweite der Linse kann mit hinreichender Genauigkeit anhand der in [141] aufgeführten Gleichungen für ein System bestehend aus zwei dünnen Linsen berechnet werden. Die Auswirkung der Verwendung einer Dunkelkammer auf ein Lochkameramodell wurde in [38] betrachtet. In einem abschließenden Schritt wird die Kamera auf den Monitor und die

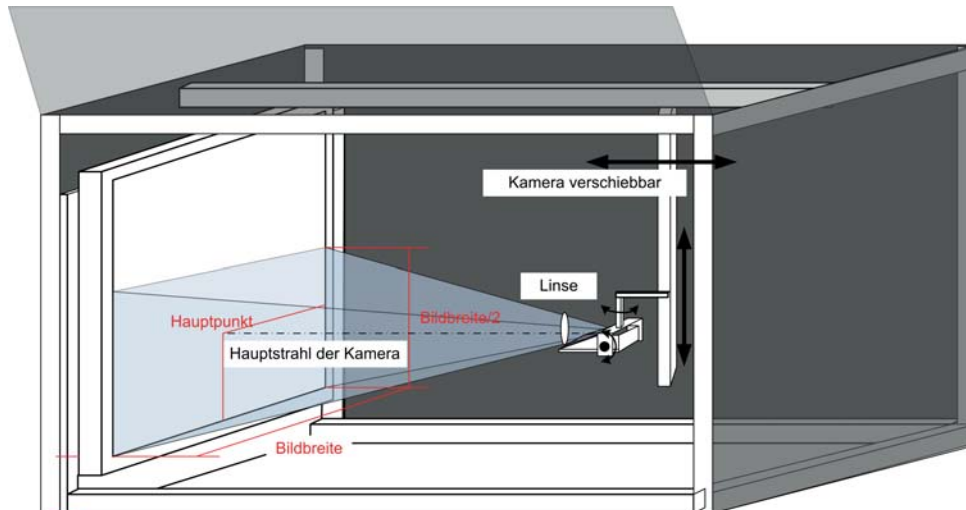


Abbildung 7.2.: Aufbau der Dunkelkammer

virtuelle Umgebung kalibriert. Dafür wird die für die Kalibrierung der Kamera im Fahrzeug verwendete Kalibriertafel in Virtual Test Drive positioniert. Die Dimension, Abstand und Ausrichtung der Tafel in der Simulation entsprechen der Realität.

7.2.1. Abbildungsfehler

Die Verwendung einer Dunkelkammer führt zu einer Reihe von Abbildungsfehlern. Aus Sicht der Bildverarbeitung sind die Veränderung der Grauwerte, Interferenzen und diskretes Motion-Blur am schwerwiegendsten.

Veränderung der Grauwerte

Monitore oder Fernseher können in der Regel nur einen gewissen Helligkeits- und Kontrastbereich abbilden. Die Leuchtdichte eines handelsüblichen Monitors liegt zwischen $300 - 360 \frac{cd}{m^2}$ und kann als eine Flächenlichtquelle mit homogener Ausleuchtung betrachtet werden. Im Vergleich dazu besitzt ein realer Scheinwerfer eine Helligkeit von $70000lx$. Daher ist es mit einem Monitor nicht möglich Scheinwerfer zu simulieren. Dieser Sachverhalt konnte empirisch nachgewiesen werden. Weiterhin reduziert sich der Kontrast im Bild. Die Bildmatrixelemente eines TFT-Monitors wirken als dimmende Elemente, welche in Abhängigkeit von der Spannung die Lichtdurchlässigkeit verändern. Selbst bei einem vollständig gesperrten Element wird noch ein gewisser Anteil an Restlicht hindurch gelassen.

In Abbildung 7.3 ist der von einer realen Kamera gemessene Grauwert über den im Monitor abgebildeten Grauwert dargestellt. Der Grauwert des Monitors wurde von Null auf den Maximalwert erhöht. Keiner der gewählten Monitore kann ein echtes Schwarz darstellen, da im Bild eine Resthelligkeit vorzufinden ist. Bei Betrachtung der Abbildungskennlinie von Monitor 1 und Monitor 2 kann festgestellt werden, dass die Antwort einen logarithmischen Verlauf annimmt. Die starke Schwankung der Grauwerte von Monitor 1 ist Interferenzen mit der Kamera geschuldet. Die logarithmische Abbildung kann mithilfe einer Invertierung der Gammakorrektur entfernt werden, wenngleich mit einem Verlust an Kontrast zu rechnen ist. Monitor 3 war ohne Korrekturmaßnahmen einen verbesserten Grauwerteverlauf darzustellen. Bei allen Monitoren ist zu beachten, dass ab einem vorgegebenen Grauwert von $g > 200$ der gemessene Grauwert sich asymptotisch einem Endwert nähert und nicht die Sättigung der Kamera erreicht. Im rechten Teil

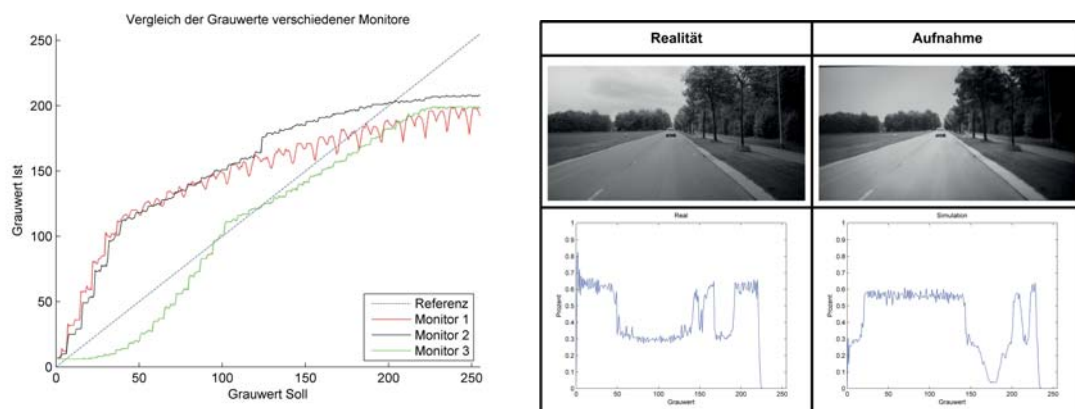


Abbildung 7.3.: Links: Vergleich der Grauwerte verschiedener Monitore, Rechts: Vergleich der Histogramme reale Szene vs. abgefilmte reale Szene

der Abbildung 7.3 ist das Histogramm eines realen Kamerabilds im Vergleich zu einem abgefilmten Kamerabild aus der Dunkelkammer dargestellt. Wie im abgefilmten Bild und im Histogramm des Bilds zu sehen ist, findet eine Verschiebung der Grauwerte statt. Das Bild weist eine höhere Helligkeit im Mittel auf. Die größeren Grauwerte sind weniger stark belegt. In Gesamtheit resultiert daraus eine Reduktion des Kontrasts im abgefilmten Bild. Für die Bildverarbeitung wichtige Merkmale, wie Kanten oder Texturen sind weniger stark ausgeprägt.

Interferenzen

Bei TFT-Monitoren wird die Farbe eines Bildpunkts dadurch erzeugt, dass jedes TFT-Element der Bildmatrix als Shutter der Hintergrundbeleuchtung fungiert. Bedingt durch die am TFT-Element angelegte Spannung kann die Lichtdurchlässigkeit jedes TFT-Elements gesteuert werden. Da die TFT-Elemente nicht selbstleuchtend sind wird eine aktive Hintergrundbeleuchtung benötigt. Durch Anpassung dieser Hintergrundbeleuchtung, kann die Helligkeit gesteuert werden. Da gesperrte TFT-Elemente keine perfekte Sperrung des Lichts ermöglichen, haben schwarze Elemente eine relative Helligkeit. Als Hintergrundbeleuchtung kommen entweder CCFL-Röhren (Cold Compact Fluorescent Lamp) oder LED (Light Emitter Diodes) zum Einsatz. Zur Steuerung der Helligkeit wird die sog. Pulsweitenmodulation verwendet [23],

bei welcher der Effektivwert der Spannung durch ein schnelles Aus- und Einschalten der Spannungsversorgung realisiert wird.

Bedingt durch das schnelle Ein- und Ausschalten der Beleuchtung oder Pixel kann es im Zusammenhang mit der Abtastfrequenz der Kamera bzw. dem zeilenweise Auslesen zu Interferenzen kommen. Im Bild kann dies als stehende oder bewegende Welle wahrgenommen werden. In der Praxis kann dieser Effekt reduziert werden, indem die Helligkeit des Anzeigeräts auf ein Maximum erhöht wird. Als Auswirkung reduziert sich der Schwarz-Wert im Bild. In Abbildung 7.4 ist die Interferenz eines Monitors dargestellt.

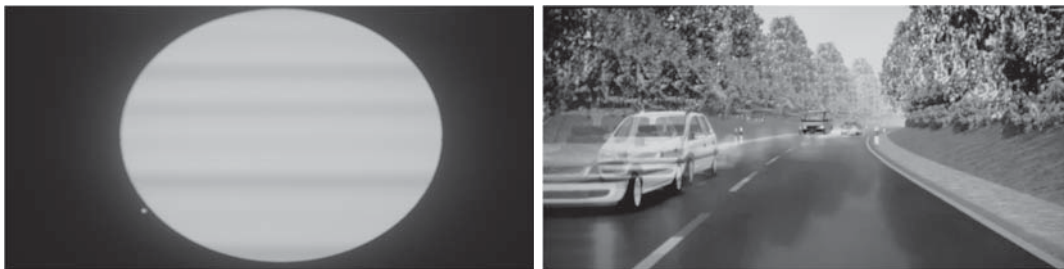


Abbildung 7.4.: Links: Interferenz zwischen Monitor und Kamera
Abbildung 7.5.: Rechts: Diskretes Motion-Blur

Motion-Blur

Bei der Aufnahme eines bewegten Bilds mit einem hohen optischen Fluss kommt es durch die endlichen Belichtungszeiten zum Motion-Blur Effekt, vgl. 2.2.5. Bei Aufnahme einer Bildfolge, wie es beim Abfilmen von einem Anzeigerät vorliegt, kann es passieren, dass zwei überlagernde Bilder aufgezeichnet werden. Dies tritt dann auf, wenn das Bild auf dem Anzeigerät während des Integrationszeitraums der Kamera aktualisiert wird. In Abbildung 7.5 ist dieser Sachverhalt abgebildet. Als Grund ist eine nicht vorhandene Synchronisation zwischen Anzeigerät und Kamera zu sehen. Als Lösungsmöglichkeit kann die von nVidia angebotene G-Sync-Technologie herangezogen werden. Bei dieser Technologie kann die Umschaltung des V-Sync des Monitors durch eine externe Quelle getaktet werden. Damit kann eine Aktualisierung des Bilds während der Belichtungszeit unterbunden werden.

7.3. Weitere Sensormodelle

Wie in Kapitel 2 erläutert, werden für die Realisierung von Fahrerassistenzsystemen verschiedene Sensoren verwendet. Daher ist es notwendig für die Erprobung dieser Systeme neben den Kamerabildern auch Daten für das Radar, die Ultraschallsensoren und das Navigationssystem bereitzustellen. Durch das Navigationssystem werden vorausschauende Daten des Straßenverlaufs bereitgestellt, diese werden bei Audi als prädiktive Streckendaten bezeichnet. Der Aufwand zur Stimulation dieser Systeme richtet sich nach den vorhandenen Testschnittstellen und kann von der Bereitstellung einfacher Objektlisten bis hin zu physikalischen Rohdaten führen.

7.3.1. Radar

Zur Simulation des Front- und Heckradars werden dynamische Radarobjekte aus der Simulation generiert, siehe Abbildung 7.6. Das verwendete Sensormodell basiert auf einem einfachen Kegelstumpf mit parametrierbaren Öffnungswinkel und Reichweite. Als Ausgabegröße werden Objekte im Sensorkoordinatensystem zurückgegeben.



Abbildung 7.6.: Radarmodell und Reaktion des Adaptive Cruise Control

Abbildung 7.7.: Stimulation des Navigationssystems in der Simulation

Diese werden mit einem Radarmodell des Zulieferers in entsprechende Reflexionen umgerechnet. Anschließend werden die Daten über die Datenschnittstellen in die Signalverarbeitung der Software eingespeist.

7.3.2. Prädiktive Streckendaten

Zur Stimulation des Navigationssystems ist es notwendig in der Simulation der Realität entsprechende Straßenverläufe abbilden zu können. Zur Abbildung der realen Straßenverläufe in der Simulation wurde der in Abschnitt 4.6 vorgestellte Algorithmus [123] verwendet. Auf Basis der virtuellen Straßen wird mittels eines Sensormodells eine GPS-Position erzeugt und über eine Schnittstelle in das reale Navigationssystem übertragen. Dieses berechnet daraufhin die entsprechende Position des Fahrzeugs, wie bei einer realen Testfahrt und stellt die prädiktiven Streckendaten zur Verfügung.

7.3.3. Ultraschall

Das umgesetzte Sensormodell ist in der Lage ein Parksystem mit zwölf Sensoren in Echtzeit mit Abstandsinformationen zu versorgen, siehe Abbildung 7.8. Zur Berechnung der Abstände werden entsprechend des Öffnungswinkels jedes Sensors Strahlen nach dem Ray-Tracing Verfahren [174] vom Sensorursprung ausgesendet und der kürzeste Abstand zu einem Hindernis zurückgegeben. Die simulierten Sensoren sind entsprechend der realen Verbaulage im virtuellen Fahrzeug positioniert und ausgerichtet.

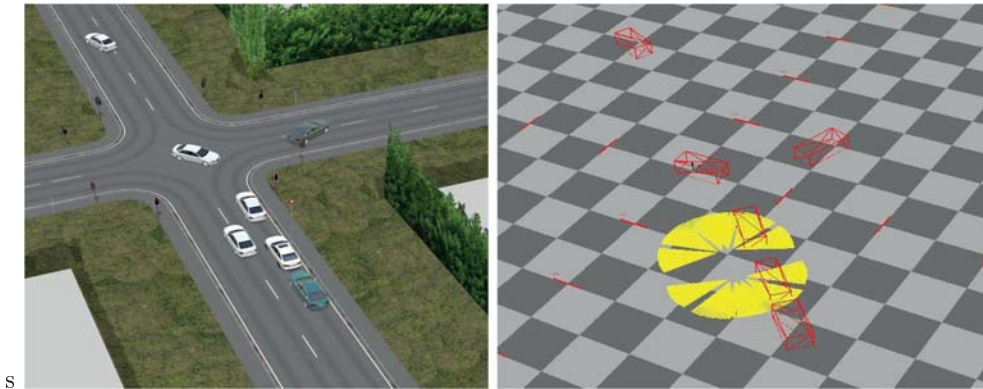


Abbildung 7.8.: Sensormodell für Ultraschall: Links Szene in Virtual Test Drive, Rechts: Szene des Sensormodells

7.4. Anwendungsbeispiele

Das System wird seit dem Jahr 2010 zur funktionalen Erprobung von Fahrerassistenzsystemen in der System-Integration eingesetzt. Die Anwendungsmöglichkeiten sollen anhand eines Testfalls für das Audi Active Lane Assist und Tempolimitanzeige dargestellt werden. Weiterhin soll die Akzeptanz der Methodik anhand der Entwicklung der Erprobungsumfänge für die Funktionen belegt werden.

7.4.1. Active Lane Assist

In Abbildung 7.9 ist ein Testfall für die Validierung des Audi Active Lane Assist dargestellt. Das funktionale Verhalten des Systems sieht es vor, dass das System bei Unterschreiten einer gewissen Spurbreite von dem aktiven in den passiven Betrieb wechseln muss. In der Realität ist dieser Testfall nur sehr aufwendig darstellbar, da eine entsprechende Straßenkonfiguration gefunden und bei jeder funktionalen Änderung erneut aufgesucht werden muss. In Abbildung 7.9 sind die für die Durchführung des Tests relevanten Bestandteile der virtuellen Erprobungsfahrt gekennzeichnet. Die Steuerung des Fahrzeugs während des Szenarios wird durch den Fahrer ausgeführt. Zur Erkennung der Fahrbahnmarkierungen ist das Sensormodell der Kamera von Relevanz. Der notwendige Straßenverlauf wird durch die Verkehrs- und Umfeldsimulation bereitgestellt.

7.4.2. Tempolimitanzeige

Die Validierung des Verhaltens der Tempolimitanzeige ist in Abbildung 7.10 beschrieben. Wie zu erkennen ist müssen für den Testfall prädiktive Streckendaten zur Laufzeit bereitgestellt werden. Daher ist es notwendig dem Navigationssteuergerät gültige Positionsangaben auf Basis eines realen Autobahnabschnitts bereitzustellen. In dem Szenario werden zum Test der Verkehrszeichenerkennung Geschwindigkeitsbegrenzungen in einer bestimmten Sequenz platziert. Diese werden durch die Kamera wahrgenommen und mit den Navigationsdaten fusioniert.

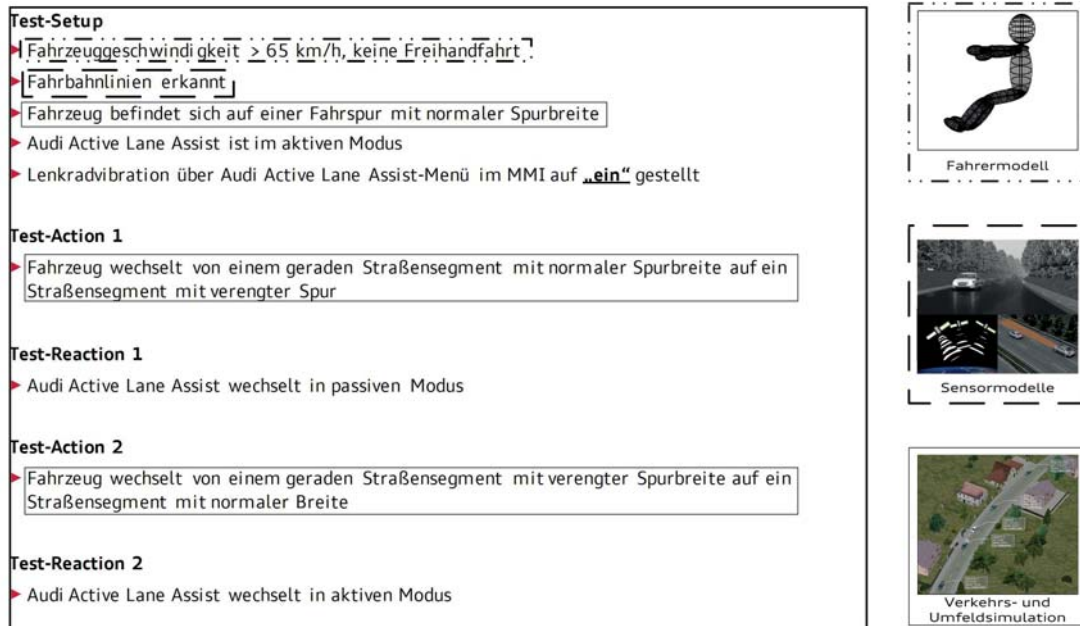


Abbildung 7.9.: Testfall für das Audi Active Lane Assist: System muss bei einer stark verengter Spur von aktiv auf passiv wechseln

7.4.3. Akzeptanz der virtuellen Erprobung

Durch Einsatz der virtuellen Erprobungsfahrt wurde die Absicherungsdichte an Hardware-in-the-Loop Prüfständen deutlich gesteigert. In Abbildung 7.11 sind die steigenden Erprobungsumfänge, exemplarisch für Spurhaltesysteme und die Tempolimitanzeige, dargestellt. Der Erprobungsschwerpunkt wurde in Richtung einer stärkeren funktionalen Erprobung verschoben. Weiterhin wird das System zur Verifikation von Auffälligkeiten eingesetzt, welche bei realen Testfahrten aufgetreten sind. Der Simulationsverbund wird inzwischen für jede Baureihe aufgebaut und stetig hinsichtlich Genauigkeit und Leistungsfähigkeit erweitert.

7.5. Zusammenfassung

In diesem Abschnitt wurde der aufgebaute Hardware-in-the-Loop Simulator für die Erprobung von Fahrerassistenzsystemen vorgestellt. Es wurde ein bestehender Hardware-in-the-Loop Prüfstand um eine ganzheitliche Simulation des Fahrzeugumfelds erweitert, welche sicherstellt, dass alle Sensorinformationen aus einer Datenquelle stammen. Als Simulationsumgebung wurde das Virtual Test Drive Framework verwendet.

Die graphische Visualisierung dient zur Stimulation der Bildverarbeitungsfunktionen. Die Übertragung des Bilds erfolgt durch eine optische Übertragung. Die optische Stimulation der Bildverarbeitungsfunktionen auf dem Kamerasteuergerät unterliegt starken Einschränkungen. Anhand der durchgeführten Untersuchungen konnte gezeigt werden, dass das Abfilmen eines Bilds von einem Monitor zu hohen Einbußen in

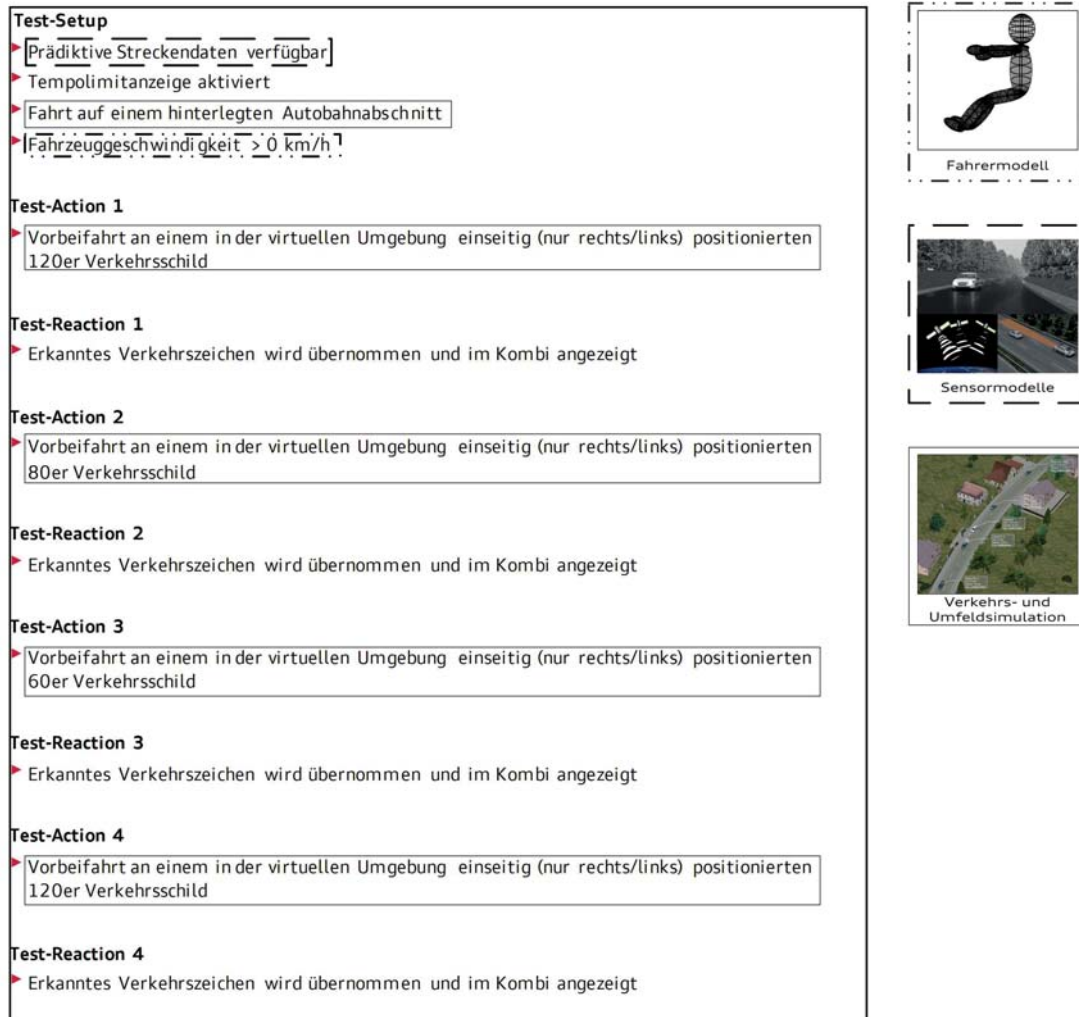


Abbildung 7.10.: Testfall für die Tempolimitanzeige: Erprobung des Systemverhaltens bei bestimmter Schilderabfolge

der Bildqualität führt, welche nicht oder nur mit hohem Aufwand kompensierbar sind. Daher gehen viele Informationen im Bild unwiederbringlich verloren. Außerdem ist es nicht möglich reale Lichtbedingungen zu erreichen. Aus diesem Grund ist die Verwendung dieses Aufbaus für die Untersuchung einer Hypothesenverifikation oder Hypothesengenerierung nicht möglich. Auf Basis der gesammelten Erkenntnisse ist es möglich das System für eine System-Integration oder den Abgleich von geschätzten Objektgrößen zu verwenden, unter der Bedingung, dass das Bildverarbeitungssystem das gesuchte Objekt in der definierten Szene richtig erkennt. Da dieser Hardware-in-the-Loop Prüfstand für die Validierung des Systemverhaltens genutzt wird ist diese Einschränkung akzeptabel. Zur Stimulation des Radars, der Ultraschallsensoren und des Navigationssystems wurden weitere Sensormodelle entwickelt. Für die Verwendung des Navigationssystems werden automatisiert reale Straßen in der Simulation nachgebildet.

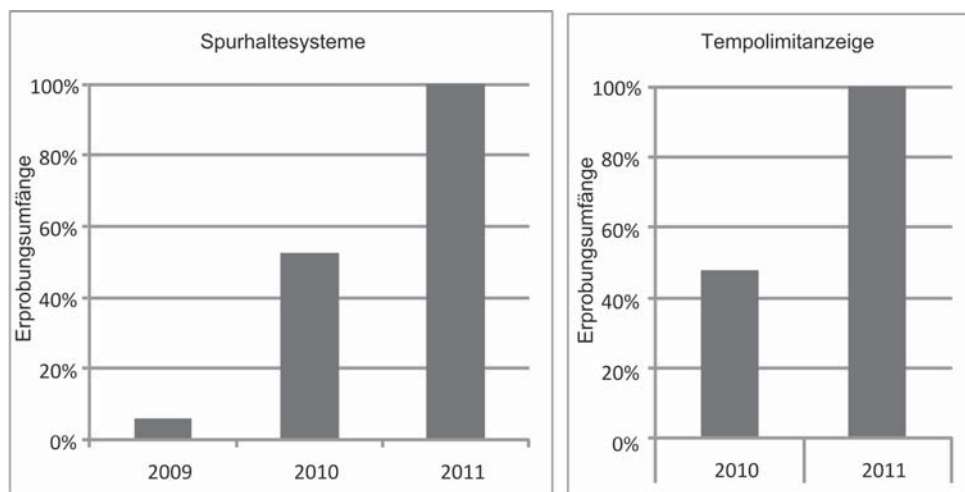


Abbildung 7.11.: Entwicklung der Erprobungsumfänge mit Einführung der virtuellen Probefahrt

Die Anwendbarkeit des Systems wurde anhand der Darstellung von zwei Testfällen dargelegt. Durch die Anwendung der Simulation ist es möglich diese Tests reproduzierbar auszuführen. Die Akzeptanz der entwickelten Methodik kann durch die Entwicklung der Erprobungsumfänge nachgewiesen werden welche seit der Einführung stetig zu nehmen.

Abschließend kann daher festgestellt werden, dass es mit dem Testsystem möglich ist Fahrerassistenzsysteme und aktive Sicherheitssysteme mit maschineller Wahrnehmung ganzheitlich in der Simulation zu betrachten. Im Hinblick auf die zunehmende Komplexität der Fahrerassistenz- und aktiven Sicherheitsfunktionen und die damit steigenden Anforderungen an Erprobungsfahrten, im physikalischen und funktionalen Grenzbereich, werden in Zukunft immer mehr Integrationsumfänge in der Simulation durchgeführt werden.

8. Zusammenfassung und Ausblick

Ziel 7: Definition von sinnvollen Anwendungsgebieten und von zukünftigen Verbesserungspotentialen.

Im Rahmen der Zusammenfassung wird das letzte Ziel dieser Dissertation bearbeitet. Als Grundlage für die Definition von sinnvollen Anwendungsgebieten und Verbesserungspotentialen sollen die bei der Bearbeitung der vorherigen Kapitel dieser Arbeit gesammelten Erkenntnisse dienen. Daher werden als erstes die Ergebnisse bei Bearbeitung der vorherigen Ziele zusammengefasst. Darauf folgend die sinnvollen Anwendungsgebiete definiert. Abschließend werden zukünftige Verbesserungspotentiale an das Themenfeld im Ausblick formuliert.

8.1. Zusammenfassung

Im Rahmen dieser Dissertation wurden Untersuchungen zur Verwendung einer Simulationsumgebung für die Erprobung von Fahrerassistenzsystemen durchgeführt. Einleitend wurden die Fahrerassistenz- und aktiven Sicherheitssysteme in modernen Fahrzeugen vorgestellt. Im Folgenden werden die bei der Bearbeitung der Ziele gesammelten Erkenntnisse zusammengefasst.

Ziel 1: Analyse und Bewertung von Simulationslösungen für die Erprobung von Fahrerassistenzsystemen

Bei der Bearbeitung des ersten Ziels in Abschnitt 3.3 konnte festgestellt werden, dass alle echtzeitfähigen Simulationsumgebungen einen funktionalen Realismus aufweisen. Einige Anwendungen verwenden einen funktionalen Photorealismus, welche im gesamten V-Modell eingesetzt werden. Die Anwendungsfälle können in eine Stimulation der Bildverarbeitung für die Erprobung der Kundenfunktion oder System-Integration und in eine Stimulation für die Erprobung der Wahrnehmungsfunktionen unterschieden werden. Der Grad des erwarteten Realismus hat häufig auch Auswirkungen auf die Detaillierung der Geometrie und Oberflächeneigenschaften. Softwarelösungen für den Test der Kundenfunktion oder System-Integration werden ausschließlich beim Hardware-in-the-Loop Test eingesetzt. Im Gegensatz dazu werden Softwarelösungen mit Fähigkeiten zur Durchführung von Tests der Wahrnehmungsfunktionen im gesamten Entwicklungsprozess beginnend, ab dem Model-in-the-Loop auf Modul-Test-Ebene eingesetzt.

Durch die Vielzahl von bekannten Anwendungsfällen kann die Sinnhaftigkeit einer Simulationsumgebung für die Erprobung von kameraunterstützten Assistenzsystemen nicht bestritten werden. In der vorliegenden Literatur wurde jedoch keine der kommerziellen Simulationslösungen umfassend verifiziert und

validiert. Es wurden keine Grenzen oder Gültigkeitsbereiche bzgl. der Anwendbarkeit der Simulationslösungen definiert bzw. kommuniziert. Da ein Modell per Definition immer nur eine beschränkte Abbildung der Realität ist, ist dieses von höchster Wichtigkeit. Besonders bei Simulationslösungen mit der Zielsetzung einer Stimulation der Wahrnehmung ist dies eine notwendige Voraussetzung, um die Zweckmäßigkeit und die Anwendbarkeit einzelner Teilmodelle und des Gesamtmodells zu rechtfertigen.

Ziel 2: Definition von Anforderungen an eine Simulationsumgebung zur Erprobung von Bildverarbeitungsfunktionen

Zur Definition der Anforderungen an die Simulationsumgebung wurde auf die Herausforderung der Stimulation von Sensorsystemen für die Hardware-in-the-Loop Erprobung eingegangen. Als besonders komplex erweist sich die Stimulation der Kamerasysteme, da es für einen End-to-End Test notwendig ist die Bildverarbeitung zu stimulieren. Auf Basis der Ergebnisse der funktionalen Analysen von Algorithmen zur Spur-, Fahrzeug- und Verkehrszeichenerkennung wurden Anforderungen für die Darstellung eines funktionalen Tests und erweiterte Anforderungen für die Simulation von Störgrößen hergeleitet. Die Anforderungen wurden in die Kategorien Beleuchtungssimulation, Geometrie und Oberflächeneigenschaften, Synthese von Kamerabildern und Szenen- und Szenario-Design eingeteilt. Die Umsetzbarkeit dieser Kategorien wurde in Ziel drei untersucht.

Ziel 3: Experimentelle Evaluierung der Komplexität der Simulation

Die Bewertung der Komplexität der Simulation wurde mittels einer eigenen Simulationsumgebung durchgeführt. In dieser wurden verschiedene Konzepte aus den einzelnen Anforderungsfeldern umgesetzt und bewertet.

Simulation der Beleuchtung Zur Simulation einer realitätsnahen Beleuchtung in der Fahrsimulation wurden Techniken zur Erzeugung von Schatten, Sonnenstrahlung und atmosphärischen Bedingungen untersucht. Zur Approximation von Schatten wurden Techniken aus der Perspektive der Fahrzeugkamera betrachtet. Als Ergebnis wurde festgestellt, dass für Objekte in der Nähe der Straßenoberfläche keine komplexe Schattenberechnung notwendig ist. Zur Integration von Schatten weiter vom Boden entfernter Objekte muss eine Technik mit weichen Schatten verwendet werden.

Die Simulation der Sonnenstrahlung wurde auf Basis einer physikalischen Bestimmung der Sonnenposition und der spektralen Zusammensetzung des Sonnenlichts durchgeführt. Zur Realisierung der atmosphärischen Dämpfung wurde ein physikalisches Modell für das Himmelslicht und die Verschleierungsperspektive umgesetzt. Mit diesem Modell ist es möglich physikalisch basierte Wettereffekte darzustellen. Die Untersuchung hinsichtlich der Anwendung einer atmosphärischen Simulation ergab, dass diese Effekte als optional anzusehen sind. Die spektrale Verschiebung des Sonnenlichts kann ebenso durch eine direkte Modifikation der Lichtquellenfarbe definiert werden.

Geometrie und Oberflächeneigenschaften von Objekten Bei der Betrachtung der Simulationsqualität von Fahrzeugen, Straßenoberflächen, Straßenmarkierungen und Verkehrszeichen wurde festgestellt, dass einfachere Oberflächenmodelle für eine Simulation der Algorithmen ausreichend sind. Bei Auswertung der Oberflächenmodelle zeigte sich, dass die Blinn-Phong BRDF für Fahrzeuge, Straßenoberflächen und Verkehrszeichen eine hinreichend genaue Approximation darstellt. Durch genauere Modelle ist es möglich die Abweichung zur Realität zu minimieren. Bei der Simulation einer nassen Straßenoberfläche zeigte sich, dass die Umsetzung dieser, trotz physikalisch motivierter Modelle, eine Herausforderung darstellt. Bei einem Vergleich zur realen Referenz lagen Abweichungen zwischen der realen und der simulierten Szene vor. Daher ist gerade für die Simulation von komplexeren Umfeldbedingungen eine genaue Validierung der Modelle notwendig.

Synthese von Kamerabildern Zur Bewertung von Verfahren zur Bildsynthese wurden Techniken zur Realisierung eines an der Realität angelehnten Prozesses zur Bildsynthese untersucht. An erster Stelle wurden Verfahren zum Anti-Aliasing betrachtet. Es zeigte sich, dass ein 2x2 Grid-Filter für die Verbesserung der Kantenqualität ausreicht. Zur Simulation der Abbildungseigenschaften einer Kameraoptik wurden Verfahren zur Simulation der Tiefenschärfe und optischer Verzeichnungen betrachtet. Durch die Simulation der Tiefenschärfe kann der Einfluss des Hintergrunds auf das Bild reduziert werden. Die Verzeichnungen durch die Optik wurden durch eine Inversion der Korrekturterme der Bildverarbeitung durchgeführt. Die Wirksamkeit der Modelle wurde durch Anwendungsbeispiele nachgewiesen.

Zur Simulation des Strahlungsaufnehmers einer realen Kamera wurde ein Modell hergeleitet, welches mit Angaben aus einem Datenblatt parametrierbar ist. Das Modell setzt sich aus einer Funktion zur Bestimmung der Bestrahlungsstärke, des Antwortverhaltens, des Farbfilters, der Quantenausbeute, der Belichtungssteuerung und der Konvertierung der Bestrahlung in Grauwerte zusammen. Zur Integration von Rauschen während der Bildgenerierung wurde das Modell von [88] in einem Shader auf der Grafikkarte umgesetzt.

Als Ergebnis kann festgestellt werden, dass durch Anwendung dieser Techniken eine Simulation des optischen Pfads und der Strahlungsaufnehmers in Echtzeit realisiert werden kann. Die Reduktion von Artefakten der Bildsynthese ist durch Anwendung von Anti-Aliasing und Tiefenschärfe-Effekten möglich.

Aufbau von Szenen und Szenarien Zur Erstellung von Szenarien wurde ein halbautomatischer Prozess hergeleitet. Dieser ist die Grundlage zur Rekonstruktion von Testfahrten in der Simulation. Dazu werden in einem ersten Schritt Daten zur Beschreibung der Bewegung des Messfahrzeugs und weiterer Fahrzeuge bestimmt. Die Genauigkeit des Ansatzes wurde in verschiedenen Veröffentlichungen nachgewiesen. Weiterhin wurde eine Möglichkeit zur Abbildung von realen Straßenverläufen in der Realität geschaffen. Durch die Abbildung von realen Straßenverläufen in der Simulation können Navigationsdaten erzeugt werden. Weiterhin wurden durch Naal [117] die Auswirkungen der Randbebauung in der Simulation auf die Fahrzeugerkennung betrachtet. Bei der Analyse stellte sich heraus, dass eine hohe Varianz der Randbebauung einen positiven Einfluss auf die Leistungsfähigkeit der Bildverarbeitung hat. Ein monotoner Hintergrund wirkte sich negativ auf die Leistungsfähigkeit aus.

Ziel 4: Entwicklung eines generischen Verfahrens zur Validierung und Kalibrierung der Simulation

Im Rahmen dieser Arbeit wurde eine Metrik für die Validierung und die Kalibrierung einer Simulationsumgebung entwickelt. Als Referenz für den Vergleich wird ein Bild der Sollszene verwendet. Die Metrik setzt sich aus drei Teilmaßen zusammen, welche die photometrische, die geometrische und die strukturelle Differenz beschreiben. Zur Steigerung der Robustheit wird kein Vergleich einzelner Pixel durchgeführt. Der Vergleich wird mittels statistischer Merkmale aus definierten Regionen ausgeführt. Die Regionen für den Vergleich wurden mittels der Rasterisierung definiert.

Die Metrik wurde in Matlab umgesetzt und mit der Structural Similarity Index Metric (SSIM) als Referenz-Maß verglichen. Als Stichprobe wurde eine reale Szene, eine simulierte Szene mit hoher Ähnlichkeit, eine simulierte Szene mit gezielter Abweichung, ein Zufallsbild und ein Bild mit Rauschen betrachtet. Als Ergebnis konnte festgestellt werden, dass die neue Metrik der SSIM überlegen ist. Besonders bei dem Zufalls- und dem Rausch-Bild ist die neue Metrik in der Lage diese effektiv zu differenzieren. Die Distanzen sind mindestens drei Mal so hoch wie bei Anwendung der SSIM. Die Metrik wurde als zu minimierende Fehlerfunktion für die Optimierung von Simulationsparametern verwendet. Im Anwendungsbeispiel wurde die automatische Optimierung der Straßenparameter einer Schlick-BRDF, vgl. 4.4.3.2, und des Straßenkontrasts vorgenommen. In einem weiteren Experiment wurde der Lack des Fahrzeugs automatisiert angepasst. Als Ergebnis kann festgestellt werden, dass die Metrik zur Optimierung von Simulationsparametern verwendet werden kann.

Ziel 5: Experimenteller Nachweis zur Anwendbarkeit von synthetischen Bilddaten

Die Untersuchung der Anwendbarkeit einer Simulation zur Stimulation der Bildverarbeitung wurde mittels der Entwicklung eines Fahrzeugerkennungsalgorithmus und durch den Abgleich der Ergebnisse von Bildverarbeitungsalgorithmen an verschiedenen Stufen der Signalverarbeitung durchgeführt.

Bei der Validierung des mit simulierten Bilddaten entwickelten Algorithmus zur Fahrzeugdetektion wurde festgestellt, dass die besten Leistungswerte bei bewölktem Himmel oder Sonnenschein erreicht wurden. Diese Ergebnisse korrelieren zu einem Test mit simulierten Bilddaten. Bei Szenen mit komplexen Umweltbedingungen nimmt die Leistungsfähigkeit ab. In diesen Szenen ist der Kontrast geringer. Daher kann festgehalten werden, dass synthetische Bilder die gleiche Merkmalsverteilung wie Szenen mit diffusem Licht aufweisen.

Die Signalverarbeitungskette wurde an vier Stufen untersucht. Die Untersuchung beinhaltete den Vergleich der Kundenfunktion Active Lane Assist, Vergleich der Spur- und Fahrzeugerkennung, Vergleich der Hypothesenverifikation und Hypothesengenerierung der Fahrzeugerkennung. Bei einem Abgleich der Active Lane Assist Funktion mit einem NHTSA-Versuchsaufbau wurde festgestellt, dass das Systemverhalten einer Spurhalteunterstützung im NHTSA-Szenario ähnlich dem Verhalten bei einer realen Testfahrt ist. Durch die Laborbedingungen und die Reproduzierbarkeit der Szenarien konnte die Schwankungsbreite bei der Testdurchführung deutlich reduziert werden. Durch diesen Test wurde gezeigt, dass ein vergleichender Test von Kundenfunktion in der Simulation möglich ist.

Weiterhin wurden die Ausgangsgrößen einer Spur- und Fahrzeugerkennungsfunktion in drei Szenarien untersucht. Es wurde gezeigt, dass die von der Bildverarbeitung ermittelten Größen eine hohe Übereinstimmung zwischen Simulation und Realität aufweisen. Als Ergebnis kann festgehalten werden, dass die Voraussicht der Spurerkennung eine hohe Schwankungsbreite aufweist, aber die mittlere Abweichung gering war. Die Schätzung der Spurbreite wies ebenfalls eine hohe Ähnlichkeit zur Realität auf. Als komplex erweist sich die Umsetzung eines ähnlichen Verhaltens einer Fahrzeugdetektion in nicht idealen Szenen, da eine identische Parametrierung der Modelle entsprechend der realen Referenz als schwierig erscheint. Im Rahmen der Tests konnte nachgewiesen werden, dass sich die Simulation zum Abgleich von geschätzten Größen der Bildverarbeitung in der Simulation und in der Realität eignet.

Die Herausforderungen der Objekterkennung auf simulierten Bilddaten wurden durch die Analyse der Hypothesenverifikation und Hypothesengenerierung der Fahrzeugerkennung näher betrachtet. Dazu wurde die Hypothesen-Verifikation der Fahrzeugerkennung in einer Szene mit Nebel validiert. In der Auswertung war eine hohe Übereinstimmung erkennbar. Im Rahmen des Tests konnte festgestellt werden, dass es zur Erreichung von ähnlichen Ergebnissen in der Simulation es notwendig ist, die reale Szene im Detail nachzubilden. Aufgrund von Abweichungen in der Parametrierung der Modelle wurde das Fahrzeug früher als in der realen Testfahrt erkannt.

Zur Verifikation der Hypothesengenerierung wurde eine Optimierung der Simulationsparameter durchgeführt. Als Hilfsmittel wurde die in Abschnitt 5 beschriebene Metrik eingesetzt. Durch die Optimierung der Parameter erzeugte der Klassifikator ähnliche Hypothesen wie während der realen Testfahrt. Als Ergebnis kann festgestellt werden, dass der Klassifikator eine ähnliche Klassifikationsleistung wie in einer realen Szene aufwies. Bei dem Test der Hypothesen-Generierung konnte festgestellt werden, dass simulierte Bilder mit einem ähnlichen Inhalt oder der Struktur zu gleichen Hypothesen wie während der realen Testfahrt führten.

Zusammenfassend kann festgestellt werden, dass der Aufwand für eine realistische Simulation steigt, je früher die Signalverarbeitung stimuliert werden soll. Durch die Tests konnte nachgewiesen werden, dass in der Simulation eine Bewertung einer Kundenfunktion oder der geschätzten Objekteigenschaften möglich ist. Dafür musste die Simulation eine grundlegende Parametrierung aufweisen. Auf der anderen Seite wurde bei der Entwicklung und Test von Bildverarbeitungsfunktionen festgestellt, dass die Zusammenstellung der Szene und die genaue Parametrierung der Modelle und Atmosphäre eine notwendige Voraussetzung für eine erfolgreiche Untersuchung darstellt. Durch Anwendung der Metrik konnten im Rahmen der Hypothesengenerierung ähnliche Ergebnisse wie in der Realität erreicht werden. Damit kann abschließend festgestellt werden, dass es möglich ist eine Simulationsumgebung in jeder Stufe der Signalverarbeitung für die Entwicklung und den Test einzusetzen.

Ziel 6: Aufbau eines Systems für die Hardware-in-the-Loop Erprobung von Fahrerassistenzsystemen

Für die Erprobung von Fahrerassistenzsystemen wurde ein Hardware-in-the-Loop System entwickelt. Dafür wurde ein bestehender Hardware-in-the-Loop Prüfstand, um eine ganzheitliche Simulation des Fahrzeugumfelds erweitert. Als Simulationsumgebung wurde das Virtual Test Drive-Framework verwendet. Die Bildverarbeitung wird mit der graphischen Ausgabe von Virtual Test Drive durch eine optische Übertragung stimuliert. Die optische Stimulation der Bildverarbeitungsfunktionen auf dem Kamerasteuergerät

unterliegt starken Einschränkungen. Anhand der durchgeführten Untersuchungen konnte gezeigt werden, dass das Abfilmen eines Bilds von einem Monitor zu hohen Einbußen in der Qualität führt. Aus diesem Grund ist die Verwendung dieses Aufbaus für die Untersuchung einer Hypothesenverifikation oder Hypothesengenerierung nicht möglich. Auf Basis der gesammelten Erkenntnisse ist es möglich das System für eine System-Integration oder Abgleich von geschätzten Objektdaten zu verwenden. Da dieser Hardware-in-the-Loop Prüfstand für die Validierung des Systemverhaltens genutzt wird ist diese Einschränkung akzeptabel.

Zur Stimulation des Radars, der Ultraschallsensoren und des Navigationssystems wurden weitere Sensormodelle entwickelt. Zur Stimulation des Navigationssystems werden realen Straßen in der Simulation nachgebildet.

Die Anwendbarkeit des Systems wurde anhand der Darstellung von zwei durchgeführten Testfällen unter Beweis gestellt. Durch die Anwendung der Simulation ist es möglich die Tests reproduzierbar auszuführen. Die Akzeptanz der entwickelten Methodik kann durch die Entwicklung der Erprobungsumfänge nachgewiesen werden.

Daher kann abschließend festgestellt werden, dass es möglich ist Fahrerassistenzsysteme und aktive Sicherheitssysteme mit maschineller Wahrnehmung ganzheitlich in der Hardware-in-the-Loop Erprobung zu betrachten. Im Hinblick auf die zunehmende Komplexität der Fahrerassistenz- und aktiven Sicherheitsfunktionen und die daraus resultierenden steigenden Anforderungen an Erprobungsfahrten werden in Zukunft mehr Integrationsumfänge in der Simulation durchgeführt.

Ziel 7: Definition von sinnvollen Anwendungsgebieten und von zukünftigen Verbesserungspotentialen

Auf Basis der in dieser Dissertation erlangten Erkenntnisse werden sinnvolle Anwendungsgebiete und Verbesserungspotentiale definiert. Weiterhin werden die in der Zielsetzung definierten Leitfragen diskutiert:

- In welchem Maße kann eine Simulation für die Entwicklung und den Test verwendet werden?
- Welchen Grad an Komplexität muss die Simulation aufweisen?
- Welcher Aufwand ist zur Erreichung dieser Zielsetzung notwendig?

Im Rahmen der Untersuchung von Konzepten in der Computergrafik zur Realisierung einer Simulation für die Stimulation von Bildverarbeitungsalgorithmen konnte festgestellt werden, dass mittels elementarer Techniken eine hinreichend genaue Qualität der Simulation erreicht werden kann. Die Simulation von Schatten kann mittels einfacher Techniken realisiert werden. Für die Simulation von Fahrzeugen, Straße und Verkehrszeichen zeigte sich, dass die Merkmale der Bildverarbeitung maßgeblich von der geometrischen Qualität der Objekte abhängen. Als Oberflächenmodelle können einfache Ansätze verwendet werden. Bei der Erstellung von Szenen wurde festgestellt, dass vor allem eine hohe Vielfalt im Hintergrund von Wichtigkeit ist. Für die Durchführung von Tests zur Prüfung der Robustheit der Bildverarbeitung wurden verschiedene Techniken zur Steigerung des Realismus untersucht. Durch die spektrale Berechnung der Sonnenstrahlung in Abhängigkeit der Position und der atmosphärischen Bedingungen kann die Beleuchtung verbessert werden. Durch die Anwendung von verbesserten Oberflächenmodellen kann

bei Fahrzeugen die Erscheinung verbessert werden. Die Anwendung von komplexeren Straßenoberflächen bringt keinen Vorteil. Als besonders komplex erwies sich die Parametrierung der Modelle. Daher wurde eine Metrik zur Messung der Differenz zwischen Simulation und Realität entwickelt.

Anhand der Komplexität der Modelle und Ergebnisse der Untersuchungen zur Anwendbarkeit einer Kameratelesimulation im Entwicklungsprozess kann auf die Anwendungsgebiete geschlossen werden. Die Erprobung einer Kundenfunktion wie das Active Lane Assist stellt relativ geringe Anforderungen an die Qualität der Simulation, da der Schwerpunkt auf die nachgelagerten Funktionalitäten gelegt wird. Ähnliche Anforderungen können für die Bewertung von durch die Bildverarbeitung geschätzten Größen, wie Abstand und Dimension gelten. Diese Annahmen können für die Hardware-in-the-Loop Erprobung herangezogen werden. Wenn das Anwendungsgebiet hinsichtlich der Bewertung von Bildverarbeitungsfunktionen verlagert werden soll, muss größerer Aufwand in Modellqualität und Parametrierung investiert werden. Durch Optimierung der Parametrierung mittels der Metrik konnten bei einem Vergleich der Hypothesengenerierung zwischen Realität und Simulation ähnliche Ergebnisse erzielt werden. Daher kann festgestellt werden, dass der Aufwand für eine Simulation der Bildverarbeitung steigt, je früher in der Signalverarbeitung eine Erprobung stattfindet.

8.2. Ausblick

Im Rahmen des Ausblicks sollen die Verbesserungspotentiale und zukünftige Forschungsrichtungen beschrieben werden. An erster Stelle muss die Qualität und Parametrierungsfähigkeit von Oberflächenmodellen verbessert werden. Bei allen durchgeführten Untersuchungen zeigte sich, dass die Parametrierung den höchsten Aufwand verursacht. Weiterhin wies die Genauigkeit der Modelle zum Teil Abweichungen auf. In Zukunft gilt es weiterhin die Erscheinung der Randbebauung zu verbessern. Dazu können z.B. verbesserte 3D-Modelle verwendet werden. Die Simulation von atmosphärischen Bedingungen und Wetter erwies sich als komplex. Daher müssen in Zukunft weitere Bestrebungen zur Umsetzbarkeit von Wettereffekten in der Simulation getätigt werden. Dies zeigte sich bei der Umsetzung einer nassen Straßenoberfläche in der Simulation. Während der Durchführung der Tests zeigt sich, dass es notwendig ist Szenen und Szenarien auf Basis von realen Testfahrten zu erzeugen. Daher sollten die entwickelten Verfahren für die Rekonstruktion von Szenarien, Trajektorien und Straßenverläufe weiterentwickelt werden.

9. Publikationen, Patente und betreute Arbeiten

9.1. Publikationen

Im Zusammenhang mit dieser Dissertation sind die folgenden Publikationen entstanden:

- Miegler, Schieber, Kern, Ganslmeier und Nentwig: Hardware-in-the-Loop-Test von vorausschauenden Fahrerassistenzsystemen, ATZelektronik, Ausgabe 2009-05.
- Miegler, Schieber, Nentwig und Ganslmeier: Virtuelle Erprobung, Automobilelektronik A8 Sonderausgabe, 2010.
- Ganslmeier, Nentwig, Roth und von Neumann-Cosel: Vehicle Environment Simulation Using Realistic Road Networks for Predictive Driver Assistance Systems. FISITA 2010 - World Automotive Congress, Budapest, 2010. Ausgezeichnet mit den Manuel Junoy Memorial Prize.
- Nentwig und Stamminger: A Method for the Reproduction of Vehicle Test Drives for the Simulation Based Evaluation of Image Processing Algorithms, 13th International IEEE Conference on Intelligent Transportation Systems, Madeira, 2010.
- Nentwig, Schieber, Miegler und Stamminger: Möglichkeiten und Grenzen der simulationsbasierten Validierung von kamerabasierten Fahrerassistenzfunktionen, 15. VDI-Tagung Erprobung und Simulation in der Fahrzeugentwicklung, Baden-Baden, 2010.
- Nentwig, Stamminger, Hardware-in-the-Loop Testing of Computer Vision Based Driver Assistance Systems, Intelligent Vehicles Symposium 2011, Baden-Baden, 2011.
- Nentwig, Schieber, Miegler, Hardware-in-the-Loop-Test für vernetzte Fahrerassistenzsysteme, ATZ Elektronik, Ausgabe 04/2011.
- Nentwig, Schieber, Miegler, Die virtuelle Erprobungsfahrt, Elektronik im KfZ 2011, Baden-Baden, 2011.
- Nentwig, Stamminger und Miegler: Concerning the Applicability of Computer Graphics for the Evaluation of Image Processing Algorithms, 2012 International IEEE Conference on Vehicular Electronics and Safety, Istanbul, 2012.

9.2. Patentanmeldungen

Im Zusammenhang mit den Ergebnisse dieser Arbeit sind folgende Patentanmeldungen entstanden:

- Nentwig, Ganslmeier, Verfahren zum Gewinnen eines Modells für einen Verlauf einer Fahrbahn, DE102009022941A1, Anmeldedatum: 27.05.2009.
- Nentwig, Verfahren und Vorrichtung für eine Funktionsprüfung einer Objekterkennungseinrichtung eines Kraftwagens, DE102010013943A1/EP2402827A1, Anmeldedatum: 06.04.2010/01.04.2011
- Nentwig, Verfahren zum Evaluieren einer Objekterkennungseinrichtung eines Kraftfahrzeugs, EP2546778A2, Anmeldedatum: 15.07.2011

9.3. Betreute Arbeiten

Im Rahmen der Dissertation wurden die folgenden Diplomarbeiten und Praktika betreut:

- Naal, Rayan: Simulationsbasierte Analyse und Validierung eines Verfahrens zur automatischen bildbasierten Fahrzeugerkennung, Diplomarbeit, 2009.
- Melzer, Daniel: Nachtfahrtsimulation für den Test von Algorithmen zur Fahrzeugdetektion bei Dunkelheit, Diplomarbeit, 2011.
- Aspeleiter, Andreas: Validierung von Sensormodellen am HIL, Praktikumsbericht, 2012.
- Königer, Sebastian: Modellierung von Fahrzeugmodellen, Praktikumsbericht, 2010.

Literaturverzeichnis

- [1] NATIONAL HIGHWAY TRAFFIC SAFETY ADMINISTRATION: Lane Departure Warning System Confirmation Test. – Forschungsbericht
- [2] DEFENSE MAPPING AGENCY: The Universal Grids - Universal Transverse Mercator (UTM) and Universal Polar Stereographic (UPS). 1989. – Forschungsbericht
- [3] NATIONAL SEMICONDUCTOR: LM9617 Monochrome CMOS Image Sensor VGA 30 FPS. 2001. – Forschungsbericht
- [4] *Weißbuch: Die europäische Verkehrspolitik bis 2010: Weichenstellungen für die Zukunft*, KOM(2001) 370. KOMMISSION DER EUROPÄISCHEN GEMEINSCHAFTEN, 2001
- [5] *Aktionsprogramm für die Straßenverkehrssicherheit (2003-2010)*. KOMMISSION DER EUROPÄISCHEN GEMEINSCHAFTEN, 2003
- [6] *Informations- und Kommunikationstechnologien für sichere und intelligente Fahrzeuge (KOM(2003) 542)*. KOMMISSION DER EUROPÄISCHEN GEMEINSCHAFTEN, 2003
- [7] APTINA: MT9V125 Data Sheet. 2005. – Forschungsbericht
- [8] *Code of Practice for the Design and Evaluation of ADAS V 5.0*. RESPONSE 3 Arbeitsgruppe, 2009
- [9] *ISO 14524 - Methods for measuring opto-electronic conversion functions (OECFs)*. International Organization for Standardization, 2009
- [10] *Ein europäischer Raum der Straßenverkehrssicherheit: Leitlinien für die Politik im Bereich der Straßenverkehrssicherheit 2011-2020*. KOMMISSION DER EUROPÄISCHEN GEMEINSCHAFTEN, 2010
- [11] : Verkehrsunfälle - Zeitreihen. Version: 2011. https://www.destatis.de/DE/Publikationen/Thematisch/TransportVerkehr/Verkehrsunfaelle/VerkehrsunfaelleZeitreihenXLS_5462403.xls?__blob=publicationFile. 2011. – Forschungsbericht
- [12] *ISO 26262 - Road vehicles – Functional safety*. International Organization for Standardization, 2012
- [13] : Refractive Index Database. Version: 2012. <http://www.refractiveindex.info>. 2012. – Forschungsbericht
- [14] AKENINE-MÖLLER, Tomas ; HAINES, Eric ; HOFFMANN, Naty: *Real-Time Rendering*. Third Edition. A K Peters, Ltd., 2008

- [15] ALCANTARILLA, P. F. ; BERGASA, L. M. ; JIMÉNEZ, P. ; SOTELO, M. A. ; PARRA, I. ; FERÁNDEZ, D.: Nighttime Vehicle Detection for Driving Assistance Lightbeam Controller. In: *IEEE Intelligent Vehicles Symposium 2008*, 2008, S. 291–296
- [16] ALT, Nicolas ; CLAUS, Chritsopher ; STECHELE, Walter: Hardware/Software Architecture of an Algorithm for Vision-Based Real-Time Vehicle Detection in Dark Environments. In: *Conference of Design, Automation and Test in Europe*, 2008, S. 176–181
- [17] ARVO, James: Backward Ray Tracing. In: *In ACM SIGGRAPH '86 Course Notes - Developments in Ray Tracing*, 1986, S. 259–263
- [18] ASPELEITER, Andreas: Praktikumsbericht: Validierung vom Sensormodellen am HIL / AUDI AG. 2012. – Forschungsbericht
- [19] AUDI: Audi Technology Portal. (2012). www.audi-technology-portal.de
- [20] BADDELEY, A. J.: An Error Metric for Binary Images. In: *International Workshop on Robust Computer Vision*, 1992, S. 59–78
- [21] BADERSCHNEIDER, M. ; MERTKE, T.: Testsystem für kamerabasierte Fahrerassistenzsysteme. In: *Betrandt FAS Konferenz Ingolstadt*, 2010
- [22] *Kapitel Verification, Valdiation and Testing*. In: BALCI, Osman: *Handbook of Simulation*. John Wiley & Sons, 1998, S. 335 – 393
- [23] BAUER, Horst (Hrsg.): *Kraftfahrtechnisches Taschenbuch*. Bd. 25. Robert Bosch GmbH, 2003
- [24] BLINN, James F.: Models of light reflection for computer synthesized pictures. In: *SIGGRAPH Comput. Graph.* 11 (1977), Juli, Nr. 2, S. 192–198. – ISSN 0097–8930
- [25] BLINN, James F.: Simulation of wrinkled surfaces. In: *SIGGRAPH Comput. Graph.* 12 (1978), August, Nr. 3, S. 286–292. – ISSN 0097–8930
- [26] BOCK, Thomas.: *Vehicle in the Loop - Test- und Simulationsumgebung für Fahrerassistenzsysteme*. Cuvillier Verlag, 2008
- [27] BOLIN, Mark R. ; MEYER, Gary W.: A perceptually based adaptive sampling algorithm. In: *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM, 1998 (SIGGRAPH '98). – ISBN 0–89791–999–8, S. 299–309
- [28] BOLIN, Mark R. ; MEYER, Gary W.: Visual difference metric for realistic image synthesis. (1999), S. 106–120
- [29] BOSSU, Jérémie ; GRUYER, Dominique ; SMAL, Jean C. ; BLOSSEVILLE, Jean M.: Validation and Benchmarking for Pedestrian Video Detection based on a Sensors Simulation Platform. In: *IEEE Intelligent Vehicles Symposium*, 2010
- [30] BOURS, Roy: Introduction to PreScan. Version: 2011. <http://home.agh.edu.pl/~kridm/files/PreScanGeneral.pdf>. 2011. – Forschungsbericht

-
- [31] BOUSSARD, Clément ; HAUTIERE, Nicolas ; NOVEL, Brigitte d'Andrea: Vehicle Dynamics Estimation for Camera-based Visibility Distance Estimation. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008
- [32] BREDIES, Kristian ; LORENZ, Dirk: *Mathematische Bildverarbeitung - Einführung in Grundlagen und moderne Theorie*. Vieweg+Teubner Verlag, 2011
- [33] BREITHECKER, Marc: *Die Bewertung und Optimierung der visuellen Wahrnehmung in der Fahrsimulation*, Universität Erlangen-Nürnberg, Diss., 2008
- [34] BREUER, Karsten ; FREDERIC, Christen: Entwicklung von Fahrerassistenzsystemen unter Verwendung von SIL- und HIL-Techniken im Verkehrsflusssimulationsprogramm PELOPS. In: *11. Aachener Kolloquium Fahrzeug- und Motorentechnik*, 2002
- [35] BRIGHAM, Elbert O.: *FFT-Anwendungen*. R. Oldenbourg Verlag, 1997
- [36] BRKIC, Karla: An Overview of Traffic Sign Detection Methods / Department of Electronics, Microelectronics, Computer and Intelligent Systems Faculty of Electrical Engineering and Computing Unska 3, 10000 Zagreb, Croatia. – Forschungsbericht
- [37] BROUK, I. ; NEMIROVSKY, A. ; NEMIROVSKY, Y.: ANALYSIS OF NOISE IN CMOS IMAGE SENSOR. In: *IEEE International Conference on Microwaves, Communications, Antennas and Electronic Systems 2008*, 2008
- [38] BURDESSA, C. ; KREIS, C. ; GREUL, R. ; TORSTEN, B.: Eine Kamera zur Spurerkennung in einer Virtuellen Umgebung. In: *15. VDI-Tagung Erprobung und Simulation in der Fahrzeugentwicklung*, 2010
- [39] CANNY, J.: A Computational Approach To Edge Detection. In: *IEEE Trans. Pattern Analysis and Machine Intelligence* 8 (1986), Nr. 6, S. 679–698
- [40] CATMULL, E. ; ROM, R. ; BARNHILL, R. E. (Hrsg.) ; RIESENFELD, R. F. (Hrsg.): *A class of local interpolating splines*. New York, NY, USA : Academic Press, 1974. – 317–326 S.
- [41] CECH, Markus: *Fahrspurschätzung aus monokularen Bildfolgen für innerstädtische Fahrerassistenzanwendungen*, Institut für Mess- und Regelungstechnik - Universität Karlsruhe (TH), Diss., 2009
- [42] CHEN, Yen-Lin ; CHEN, Yuan-Hsin ; CHEN, Chao-Jung ; WU, Bing-Fei: Nighttime Vehicle Detection for Driver Assistance and Autonomous Vehicles. In: *International Conference on Pattern Recognition*, 2006, S. 687–690
- [43] CONTINENTAL, AG: Continental-Sensorsystem vermeidet Auffahrunfälle – Weltpremiere im neuen Volvo XC60. Version: 2008. http://www.conti-online.com/generator/www/com/de/continental/presseportal/themen/pressemitteilungen/3_automotive_group/chassis_safety/press_releases/volvo_xc60_31_03_08_de.html. 2008. – Forschungsbericht
-

- [44] COSKUN, F. ; TUNCER, Ö. ; ELIF KARSLIGIL, M. ; GÜVENC, L.: Real Time Lane Detection and Tracking System Evaluated in a Hardware-in-The-Loop Simulator. In: *13th International IEEE Conference on Intelligent Transportation Systems*, 2010, S. 1336 – 1343
- [45] CROW, F.C.: Shadow algorithms for computer graphics. In: *ACM SIGGRAPH Computer Graphics* 11 (1977), Nr. 2, S. 242–248
- [46] DAIMLER, AG: Automatisiertes Fahren präzisiert Prüfmethodik. Version: 2010. <http://media.daimler.com/dcmedia/0-921-656548-49-1294063-1-0-1-0-0-0-11701-614316-0-1-0-0-0-0-0.html>. 2010. – Forschungsbericht
- [47] DALY, Scott: The visible differences predictor: an algorithm for the assessment of image fidelity. In: WATSON, Andrew B. (Hrsg.): *Digital images and human vision*. Cambridge, MA, USA : MIT Press, 1993. – ISBN 0–262–23171–9, S. 179–206
- [48] *Kapitel* Depth of Field: A Survey of Techniques. In: DEMERS, Joe: *GPU Gems*. Addison-Wesley Longman, Amsterdam, 2004
- [49] DEVERNAY, Frédéric ; FANGERAS, Olivier: Automatic Calibration and removal of distortion from scenes of structured environments. In: *SPIEs 1995 International Symposium on Optical Science, Engineering and Instrumentation*, 1995
- [50] DICKMANN, D. E. ; MYSLIWETZ, D. B.: Recursive 3-D road and relative ego-state recognition. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14 (1992), S. 199–213
- [51] *Kapitel* Fusionsarchitekturen zur Umfeldwahrnehmung für zukünftige Fahrerassistenzsysteme. In: DIETMAYER, Klaus ; KIRCHNER, Alexander ; KÄMPCHEN, Nico: *Fahrerassistenzsysteme mit maschineller Wahrnehmung*. Springer-Verlag Berlin Heidelberg, 2005, S. 59 – 88
- [52] DIMITROV, Rouslan: Cascaded Shadow Maps / NVIDIA Corporation. 2007. – Forschungsbericht
- [53] DIRNDORFER, Tobias ; ROTH, Erwin ; NEUMANN-COSEL, Kilian von ; WEISS, Christian ; KNOLL, Alois: Simulation environment for the development of predictive safety systems. In: *FISITA 2010 - World Automotive Congress*, 2010
- [54] DRÖSCHEL, W (Hrsg.): *Das V-Modell 97*. Oldenbourg Verlag, München, 1999
- [55] dSPACE: Viel Verkehr auf virtuellen Straßen. In: *dSpace-Magazin* 02 (2008), S. 38 – 41
- [56] DUECK, Gunter: New Optimization Heuristics The Great Deluge Algorithm and the Record-to-Record Travel. In: *Journal of Computational Physics* 104 (1993), S. 86–92
- [57] DUPUIS, Marius ; NEUMANN-COSEL, Kilian von ; WEISS, Christian: Virtual Test Drive – Vereinheitlichung der Simulationsumgebung für SiL-, HiL-, DiL- und ViL-Tests bei der Entwicklung von Fahrerassistenz- und aktiven Sicherheitssystemen. In: *AutoTest 2010*, 2010
- [58] DURIKOVIC, Roman ; ERSHOV, Sergey ; KOLCHIN, Konstantin ; MYSZKOWSKI, Karol: Solution of an Inverse Problem in Rendering Metallic and Pearlescent Appearance. In: *Journal of Three Dimensional Images* 18 (2004), S. 54–60

-
- [59] DURIKOVIC, Roman ; MARTENS, William L.: Simulation of sparkling and depth effect in paints. In: *Proceedings of the 19th spring conference on Computer graphics*. New York, NY, USA : ACM, 2003 (SCCG '03). – ISBN 1-58113-861-X, S. 193–198
- [60] ERHARDT, Angelika: *Einführung in die Digitale Bildverarbeitung - Grundlagen, Systeme und Anwendungen*. Vieweg+Teubner | GWV Fachverlage GmbH, 2008
- [61] ERSHOV, Sergey ; DURIKOVIC, Roman ; KOLCHIN, Konstantin ; MYZKOWSKI, Karol: Reverse engineering approach to appearance-based design of metallic and pearlescent paints. In: *The Visual Computer* 20 (2004), Nr. 8-9, S. 586–600. – ISSN 0178-2789
- [62] ERSHOV, Sergey ; KOLCHIN, Konstantin ; MYZKOWSKI, Karol: Rendering Pearlescent Appearance Based on Paintcomposition Modelling. In: *Eurographics 2001*, 2001
- [63] ERSHOV, Sergey V. ; KHODULEV, Andrei B. ; KOLCHIN, Konstantin V.: Simulation of sparkles in metallic paints. In: *Proceedings of Graphicon'99*, 2001, S. 121–128
- [64] EVERINGHAM, Mark ; WILLIAMS, Christopher K. I. ; ZISSERMAN, Andrew ; VAN GOOL, Luc ; ALLAN, Moray ; BISHOP, Christopher M. ; CHAPELLE, Olivier ; DALAL, Navneet ; DESELAERS, Thomas ; DORK, Gyuri ; DUFFNER, Stefan ; EICHHORN, Jan ; FARQUHAR, Jason D. R. ; FRITZ, Mario ; GARCIA, Christophe ; GRIFFITHS, Tom ; JURIE, Frederic ; KEYSERS, Daniel ; KOSKELA, Markus ; LAAKSONEN, Jorma ; LARLUS, Diane ; LEIBE, Bastian ; MENG, Hongying ; NEY, Hermann ; SCHIELE, Bernt ; SCHMID, Cordelia ; SEEMANN, Edgar ; SHAW-TAYLOR, John ; STORKEY, Amos ; SZEDMAK, Sandor ; TRIGGS, Bill ; ULUSOY, Ilkay ; VIITANIEMI, Ville ; ZHANG, Jianguo: The 2005 pascal visual object classes challenge. In: *In Selected Proceedings of the First PASCAL Challenges Workshop*, 2006, S. 14
- [65] FARRELL, Joyce ; CATRYSSSE, Peter B. ; WANDELL, Brian: Digital Camera Simulation. In: *Optical Society of America: Applied Optics* 51 (2012), February, Nr. 4, 80 - 90. http://scien.stanford.edu/jfsite/Papers/ImageCapture/DigitalCameraSimulation_092908.pdf
- [66] FARRUGIA, Jean-Philippe ; PÉROCHE, Bernard: A Progressive Rendering Algorithm Using an Adaptive Perceptually Based Image Metric. In: *Computer Graphics Forum* 23 (2004), Nr. 3, S. 605–614. – ISSN 1467-8659
- [67] FERNANDO, Randima: Percentage-Closer Soft Shadows. In: *ACM SIGGRAPH 2005 Sketches*, 2005
- [68] FERWERDA, James A.: Three Varieties of Realism in Computer Graphics. In: *SPIE Human Vision and Electronic Imaging '03*, 2003, S. 290–297
- [69] FKA: PELOPS - White Paper / Forschungsgesellschaft Kraftfahrwesen mbH Aachen. 2012. – Forschungsbericht
- [70] FRIEDRICH, A. ; GROSSE-KAPPENBERG, S. ; HAPPE, J.: Erprobung von Fahrerassistenzsystemen mit dem Interactive Driving Simulator. In: *VDI-Berichte Nr. 1900*, 2005, S. 37 – 55
- [71] FU, MENG-YIN ; HUANG, YUAN-SHUI: A Survey of Traffic Sign Recognition. In: *International Conference on Wavelet Analysis and Pattern Recognition*. Qingdao, July 2010, S. 119–124
-

- [72] GANSLMEIER, Thomas ; NENTWIG, Mirko ; NEUMANN-COSEL, Kilian von ; ROTH, Erwin: Vehicle Environment Simulation using Realistic Road Networks for Predictive Driver Assistance Systems. In: *FISITA 2010 World Automotive Congress*. Budapest, 2010
- [73] GEK-GREMIUM (Hrsg.): *DIN EN 61508*. International Electrotechnical Commission, 2010
- [74] GEODESY ; DEPARTMENT, Geophysics: World Geodetic System 1984 / National Imagery and Mapping Agency. 2000. – Forschungsbericht
- [75] GILLETTE, Gray ; PIERPOINT, William ; TREADO, Stephen: A general illuminance model for daylight availability. In: *Journal of IES* (1984), S. 330 – 340
- [76] GLASER, Sébastien ; VANHOLME, Benoit ; MAMMAR, Saïd ; GRUYER, Dominique ; NOUVELIÈRE, Lydie: Maneuver-Based Trajectory Planning for Highly Autonomous Vehicles on Real Road With Traffic and Driver Interaction. In: *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS* 11 (2010), Nr. 3, S. 589 – 606
- [77] GÜNTHER, Johannes ; CHEN, Tongbo ; GOESELE, Michael ; WALD, Ingo ; SEIDEL, Hans-Peter: Efficient Acquisition and Realistic Rendering of Car Paint. In: *Vision, Modeling, and Visualization 2005*, 2005, S. 487–494
- [78] GRUYER, D. ; HIBLOT, N. ; SOUZA, P. D. ; SAUER, H. ; MONNIER, B.: A new generic virtual platform for cameras modeling. In: *Vision 2010*, 2010
- [79] GRUYER, Dominique ; GLASER, Sébastien ; VANHOLME, Nicolas Benoit a. Benoit andHiblot ; MONNIER, Bertrand: SiVIC, a virtual platform for ADAS and PADAS prototyping, test and evaluation. In: *Fisita 2010*, 2010
- [80] GUNIA, D. ; PLAUMERT, R. ; TATSCHKE, T.: Der Spurhalteassistent von Ford – vom Konzept bis zur Verifikation. In: *Aachener Kolloquium Fahrzeug- und Motorentechnik*, 2010
- [81] GUNIA, Dirk ; SCHÜLING, Jürgen: Modellbasiertes Testen des Fahrspur-Assistenten von Ford. In: *ATZ* 11 (2011), S. 844 – 848
- [82] *Kapitel* Practical Post-Process Depth of Field. In: HAMMON, Earl: *GPU Gems 3*. Addison-Wesley Professional, 2007, S. 583–606
- [83] HASENFRATZ, J. M. ; LAPIERRE, M. ; HOLZSCHUCH, N. ; SILLION, F. ; GRAVIR/IMAG-INRIA, Artis: A Survey of Real-time Soft Shadows Algorithms. In: *Computer Graphics Forum* 22 (2003), Nr. 4, S. 753–774. – ISSN 1467–8659
- [84] HOFFMAN, Naty ; PREETHAM, Arcot J.: Rendering Outdoor Light Scattering in Real Time. In: *Game Developers Conference*, 2002
- [85] *Kapitel* Anwendungsorientierte Übersicht kommerzieller Fahrzeug-Simulations-Systeme. In: HOLZMANN, Henning: *Fahrdynamik-Regelung - Modellbildung, Fahrerassistenzsysteme, Mechatronik*. Friedr. Vieweg & Sohn Verlag, GWV Fachverlage GmbH, 2006, S. 93 – 116

-
- [86] HUTTENLOCHER, Daniel P. ; KLANDERMAN, Gregory A. ; RUCKLIDGE, William J.: Comparing Images Using the Hausdorff Distance. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 15 (1993), S. 850–863
- [87] IRIE, K. ; MCKINNON, A. E. ; UNSWORTH, K. ; WOODHEAD, I. M.: A model for measurement of noise in CCD digital-video cameras. In: *MEASUREMENT SCIENCE AND TECHNOLOGY* 19 (2008)
- [88] IRIE, Kenji ; MCKINNON, Alan E. ; UNSWORTH, Keith ; WOODHEAD, Ian M.: A Technique for Evaluation of CCD Video-Camera Noise. In: *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY* 18 (2008), S. 280–284
- [89] *Kapitel Layered Car Paint Shader*. In: ISIDORO, John ; OAT, Chris ; TATARCHUK, Natalya: *ShaderX2: Shader Programming Tips & Tricks with DirectX 9*. Wordware Publishing, Inc., 2003, S. 293–298
- [90] *Kapitel Rendering of Wet Materials*. In: JENSEN, H. W. ; LEGAKIS, J. ; DORSEY, J.: *Rendering Techniques*. Springer Verlag, 1999, S. 273 – 282
- [91] JENSEN, Henrik W.: *Realistic image synthesis using photon mapping*. Natick, MA, USA : A. K. Peters, Ltd., 2001. – ISBN 1–56881–147–0
- [92] JÄHNE, Bernd: *Digitale Bildverarbeitung - und Bildgewinnung*. Bd. 7. Springer-Verlag Berlin Heidelberg, 2012
- [93] KAJIYA, James T.: The Rendering Equation. In: *13th annual conference on Computer graphics and interactive techniques* Bd. 4, 1986, S. 143–150
- [94] KANEDA, Kazufumi ; OKAMOTO, Takashi ; NAKAMAE, Eihachiro ; NISHITA, Tomoyuki: Photorealistic image synthesis for outdoor scenery under various atmospheric conditions. In: *The Visual Computer* 7 (1991), S. 247–258. – ISSN 0178–2789
- [95] KANEKO, Tomomichi ; TAKAHEI, Toshiyuki ; INAMI, Masahiko ; KAWAKAMI, Naoki ; YANAGIDA, Yasuyuki ; MAEDA, Taro ; TACHI, Susumu: Detailed shape representation with parallax mapping. In: *In Proceedings of the ICAT 2001*, 2001, S. 205–208
- [96] KASNAKOGLU, Cosku: *Developments on a Virtual Environment System for Intelligent Vehicle Applications*, The Ohio State University, Diplomarbeit, 2003
- [97] KLASSEN, R. V.: Modeling the effect of the atmosphere on light. In: *ACM Trans. Graph.* 6 (1987), Juli, Nr. 3, S. 215–237. – ISSN 0730–0301
- [98] KLEIN, A L: *Sensor and Data Fusion Concepts and Applications*. SPIE Optical Engineering Press, 1999
- [99] KLEIN, Georg ; MURRAY, David W.: Simulating Low-Cost Cameras for Augmented Reality Compositing. In: *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS* 16 (2010), Mai/Juni, Nr. 3, S. 369 – 380
-

- [100] KLETTE, Reinhard ; KRÜGER, Norbert ; VAUDREY, Tobi ; PAUWELS, Karl ; HULLE, Marc van ; MORALES, Sandino ; KANDIL, Farid I. ; HAEUSLER, Ralf ; PUGEAULT, Nicolas ; RABE, Clemens ; LAPPE, Markus: Performance of Correspondence Algorithms in Vision-Based Driver Assistance Using an Online Image Sequence Database. In: *IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY* 60 (2011), June, Nr. 5, S. 2012 – 2026
- [101] KUNO, T. ; SUGIURA, H. ; MATOBA, N.: A new automatic exposure system for digital still cameras. In: *Consumer Electronics, IEEE Transactions on* 44 (1998), feb, Nr. 1, S. 192 –199. – ISSN 0098–3063
- [102] LEI, Jia ; REDMILL, K.A. ; OZGUNER, U.: VATSIM: a simulator for vehicles and traffic. In: *IEEE Intelligent Transportation Systems Conference Proceedings*, 2001, S. 686 – 691
- [103] LIANG, JiaYi ; QIN, Yajie ; HONG, Zhiliang: An Auto-exposure algorithm for detecting high contrast lighting conditions. In: *ASIC, 2007. ASICON '07. 7th International Conference on*, 2007, S. 725 –728
- [104] LITFIN, Gerd: *[Kamera] Technische Optik in der Praxis*. Bd. 3. Springer Berlin Heidelberg New York, 2005
- [105] LYU, S. ; FARID, H.: How realistic is photorealistic? In: *Signal Processing, IEEE Transactions on* 53 (2005), feb., Nr. 2, S. 845 – 850. – ISSN 1053–587X
- [106] MARSCHNER, Steve ; WIMMER, Michael: Precomputed Atmospheric Scattering. In: *Eurographics Symposium on Rendering 2008* Bd. 27, 2008
- [107] *Kapitel* Entwurf und Test von Fahrerassistenzsystemen. In: MAURER, Markus: *Handbuch Fahrerassistenzsysteme - Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort*. Vieweg+Teubner | GWV Fachverlage GmbH, 2009, S. 43 – 54
- [108] MCCALL, J.C. ; TRIVEDI, M.M.: Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation. In: *IEEE Transactions on Intelligent Transportation Systems* Bd. 7, 2006, S. 20 – 37
- [109] MELZER, Daniel: *Nachtfahrtsimulation für den Test von Algorithmen zur Fahrzeugdetektion bei Dunkelheit*, Universität Erlangen-Nürnberg, Diplomarbeit, 2011
- [110] MIAO, Q. ; TANG, X. ; WANG, D. ; LIU, Q. ; TIDEMAN, M. ; LIU, X. ; LI, J. ; ZHU, X.: Application of PRESCAN in the Concept Development of Active Safety System. In: *INFATS Conference 2011*, 2011
- [111] MIEGLER, M. ; SCHIEBER, R. ; NENTWIG, M. ; GANSLMEIER, T.: Virtuelle Erprobung. In: *Automobilelektronik A8 Sonderausgabe*, 2010, S. 95 – 108
- [112] MIELICH, W ; GOLOWKO, K: Erprobung von Assistenzsystemen zur Minderung von Auffahrunfällen. In: *Automotive Engineering Partner* 06 (2009), S. 32 – 35
- [113] *Kapitel* Volumetric Light Scattering as a Post-Process. In: MITCHELL, Kenny: *GPU Gems 3*. Addison-Wesley Professional, 2007, S. 275–292

- [114] MÜLLER, Sven-Oliver ; BRAND, Marcus ; WACHENDORF, Sven ; SCHRÖDER, Henning ; SZOT, Thomas ; SZOT, Thomas ; KREMER, Birgit: Integration vernetzter Fahrerassistenz-Funktionen mit HiL für den VW Passat CC. In: *Automotive Engineering Partners* 06 (2009), S. 60 – 65
- [115] MORALES, S. ; VAUDREY, T. ; KLETTE, R.: Robustness evaluation of stereo algorithms on long stereo sequences. In: *IEEE Intelligent Vehicles Symposium 2009*, 2009, S. 347 – 352
- [116] MYERS, Kevin ; FERNANDO, Randima ; BAVOIL, Louis: Integrating Realistic Soft Shadows into Your Game Engine / nVidia Corporation. 2008. – Forschungsbericht
- [117] NAAL, Rayan: *Simulationsbasierte Analyse und Validierung eines Verfahrens zur automatischen bildbasierten Fahrzeugerkennung*, Universität Dortmund, Diplomarbeit, 2009
- [118] NARASIMHAN, Srinivasa G. ; NAYAR, Shree K.: Vision and the Atmosphere. In: *International Journal of Computer Vision* 48 (2002), S. 233–254
- [119] NARASIMHAN, Srinivasa G. ; NAYAR, Shree K.: Shedding Light on the Weather. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2003*, 2003
- [120] NENTWIG, M. ; MIEGLER, M. ; STAMMINGER, M.: Concerning the applicability of computer graphics for the evaluation of image processing algorithms. In: *Vehicular Electronics and Safety (ICVES), 2012 IEEE International Conference on*, 2012, S. 205 –210
- [121] NENTWIG, M. ; STAMMINGER, M.: A Method for the Reproduction of Vehicle Test Drives for the Simulation Based Evaluation of Image Processing Algorithms. In: *13th International IEEE Conference on Intelligent Transportation Systems*, 2010, S. 1307 – 1312
- [122] NENTWIG, M. ; STAMMINGER, M.: Hardware-In-The-Loop Testing of Computer Vision Based Driver Assistance Systems. In: *IEEE Intelligent Vehicles Symposium*, 2011, S. 339 – 344
- [123] NENTWIG, Mirko ; GANSLMEIER, Thomas: *Verfahren zum Gewinnen eines Modells für einen Verlauf einer Fahrbahn*. 05 2009
- [124] NENTWIG, Mirko ; SCHIEBER, Reinhard ; MIEGLER, Maximilian: Die virtuelle Erprobungsfahrt. In: *15. Internationaler Kongress Elektronik im Kraftfahrzeug*, 2011
- [125] NENTWIG, Mirko ; SCHIEBER, Reinhard ; MIEGLER, Maximilian ; STAMMINGER, Marc: Möglichkeiten und Grenzen der simulationsbasierten Validierung von kamerabasierten Fahrerassistenzfunktionen. In: *15. VDI-Tagung Erprobung und Simulation in der Fahrzeugentwicklung*, 2010
- [126] NEUMANN-COSEL, Kilian von ; DUPUIS, Marius ; WEISS, Christian: Virtual test drive - provision of a consistent tool-set for [d,h,s,v]-in-the-loop. In: *Proceedings of the Driving Simulation Conference Monaco*, 2009
- [127] NEUMANN-COSEL, Kilian von ; NENTWIG, Mirko ; LEHMANN, Daniel ; SPETH, Johannes ; KNOLL, Alois: Preadjustment of a Vision-Based Lane Tracker. In: *Driving Simulator Conference DSC*, 2009

- [128] NG, Tian-Tsong ; CHANG, Shih-Fu ; HSU, Jessie ; XIE, Lexing ; TSUI, Mao-Pei: Physics-motivated features for distinguishing photographic images and computer graphics. In: *Proceedings of the 13th annual ACM international conference on Multimedia*. New York, NY, USA : ACM, 2005 (MULTIMEDIA '05). – ISBN 1–59593–044–2, S. 239–248
- [129] NICODEMUS, F.E. ; RICHMOND, J.C. ; HSIA, J.J. ; GINSBERG, I.W. ; LIMPERIS, T.: Geometrical Considerations and Nomenclature for Reflectance / National Bureau of Standards. 1977. – Forschungsbericht
- [130] NISCHWITZ, Alfred ; FISCHER, Max ; HABERÄCKER, Peter: *Computergrafik und Bildverarbeitung - Alles für Studium und Praxis*. 2. Auflage. Friedr. Vieweg & Sohn Verlag | GWV Fachverlage GmbH, Wiesbaden, 2007
- [131] NISHITA, Tomoyuki ; DOBASHI, Yoshinori ; KANEDA, Kazufumi ; YAMASHITA, Hideo: Display Method of the Sky Color Taking into Account Multiple Scattering. In: *Proceedings of Pacific Graphics 1996* Bd. 8, 1996, S. 117–132
- [132] NURICK, A.: A Model for Upper Bound Clear Sky a Vailability of Solar Illuminance. In: *Transaction on Solar Energy* 2 (2011)
- [133] OBERKAMPF, William L.: Verification and Validation in Computational Simulation. In: *2004 Transport Task Force Meeting*, 2004
- [134] O'MALLEY, R. ; GLAVIN, M. ; JONES, E.: Vehicle Detection at Night Based on Tail-Light Detection. In: *1st International ICST Symposium on Vehicular Computing Systems*, 2008
- [135] *Kapitel Accurate Atmospheric Scattering*. In: O'NEIL, Sean: *GPU Gems 2*. Addison-Wesley, 2005
- [136] OPENDRIVE, Core-Team: OpenDRIVE V 1.3. Version: 2010. <http://www.opendrive.org>. 2010. – Forschungsbericht
- [137] OSG, Community: OpenSceneGraph. Version: 2012. <http://www.openscenegraph.org>. 2012. – Forschungsbericht
- [138] PAN, Feng ; CHEN, JiongBin ; HUANG, JiWu: Discriminating between photorealistic computer graphics and natural images using fractal geometry. In: *Science in China Series F: Information Sciences* 52 (2009), S. 329–337. – ISSN 1009–2757
- [139] PASS, Greg ; ZABIH, Ramin: Comparing images using joint histograms. In: *Multimedia Systems* 7 (1999), S. 234–240. – ISSN 0942–4962
- [140] PASS, Greg ; ZABIH, Ramin ; MILLER, Justin: Comparing images using color coherence vectors. In: *Proceedings of the fourth ACM international conference on Multimedia*. New York, NY, USA : ACM, 1996 (MULTIMEDIA '96). – ISBN 0–89791–871–1, S. 65–73
- [141] PEDROTTI, F. ; PEDROTTI, L. ; BAUSCH, W. ; SCHMIDT, H.: *Optik für Ingenieure: Grundlagen*. Springer-Verlag Heidelberg, 2005

-
- [142] PEREZ, R. ; SEALS, R. ; MICHALSKY, J.: All-weather model for sky luminance distribution – Preliminary configuration and validation. In: *Solar Energy* 50 (1993), Nr. 3, S. 235 – 245. – ISSN 0038–092X
- [143] PHARR, Matt ; HUMPHREYS, Greg: *Physically Based Rendering*. Elsevier Science Publishers B. V., 2004
- [144] PHONG, Bui T.: Illumination for computer generated pictures. In: *Commun. ACM* 18 (1975), Juni, Nr. 6, S. 311–317. – ISSN 0001–0782
- [145] PIELLA, G. ; HEIJMANS, H.: A new quality metric for image fusion. In: *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on* Bd. 3, 2003. – ISSN 1522–4880, S. III – 173–6 vol.2
- [146] PREETHAM, A. J. ; SHIRLEY, Peter ; SMITS, Brian: A practical analytic model for daylight. In: *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press/Addison-Wesley Publishing Co., 1999 (SIGGRAPH '99). – ISBN 0–201–48560–5, S. 91–100
- [147] PREETHAM, Arcot J.: Modeling Skylight and Aerial Perspective. In: *Siggraph 2003 Course Notes*, 2003
- [148] PÄTZOLD, Frank ; FRANKE, Uwe: Road Recognition in Urban Environment. In: *IEEE International Conference on Intelligent Vehicles* (1998)
- [149] RAMASUBRAMANIAN, Mahesh ; PATTANAIK, Sumanta N. ; GREENBERG, Donald P.: A perceptually based physical error metric for realistic image synthesis. In: *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press/Addison-Wesley Publishing Co., 1999 (SIGGRAPH '99). – ISBN 0–201–48560–5, S. 73–82
- [150] REDMILL, K.A. ; MARTIN, J.I. ; OZGÜNER, U.: Virtual Environment Simulation for Image Processing Sensor Evaluation. In: *IEEE Intelligent Transportation Systems Conference Proceedings*, 2000, S. 64 – 70
- [151] REDMILL, K.A. ; MARTIN, J.I. ; OZGUNER, U. ; TAMURA, K.: Sensor and data fusion design and evaluation with a virtual environment simulator. In: *IEEE Intelligent Vehicles Symposium*, 2000, S. 668 – 674
- [152] REDMILL, K.A. ; OZGUNER, U.: VATSIM: A Vehicle and Traffic Simulator. In: *Proceedings of ITS Conference*, 1999, S. 656–661
- [153] REEVES, William T. ; SALESIN, David H. ; COOK, Robert L.: Rendering Antialiased Shadows with Depth Maps. In: *SIGGRAPH '87 Proceedings of the 14th annual conference on Computer graphics and interactive techniques* Bd. 21, 1987
- [154] REIF, Konrad (Hrsg.): *Fahrstabilisierungssysteme und Fahrerassistenzsysteme*. Vieweg+Teubner Verlag | Springer Fachmedien Wiesbaden GmbH, 2010
-

- [155] REINHARD, Erik ; STARK, Michael ; SHIRLEY, Peter ; FERWERDA, James: Photographic tone reproduction for digital images. In: *ACM Trans. Graph.* 21 (2002), Juli, Nr. 3, S. 267–276. – ISSN 0730–0301
- [156] *Kapitel* Real-Time Depth of Field Simulation. In: RIGUER, Guennadi ; TATARCHUK, Natalya ; ISIDORO, John: *ShaderX2: Shader Programming Tips and Tricks with DirectX 9.0*. Wordware Publishing, Inc., 2003, S. 529–556
- [157] ROST, Randi J. ; LICEA-KANE, Bill ; GINSBURG, Dan ; KESSENICH, John M. ; LICHTENBELT, Barthold ; MALAN, Hugh ; WEIBLEN, Mike: *OpenGL Shading Language*. Addison-Wesley Longman, 2009
- [158] ROTH, Erwin ; CALAPOGLU, Tugkan: Advanced Driver Assistance System Testing Using OptiX. In: *In Proceedings of GPU Technology Conference 2012*, 2012
- [159] ROTH, Erwin ; DIRNDORFER, Tobias ; NEUMANN-COSEL, Kilian v. ; KNOLL, A. ; GANSLMEIER, Thomas ; KERN, Andreas ; WEIS, Christian: Analyse und Validierung vorausschauender Sensormodelle in einer integrierten Fahrzeug- und Umfeldsimulation. In: *VDI Berichte 2104*, 2010
- [160] RUMP, Martin ; MÜLLER, Gero ; SARLETTE, Ralf ; KOCH, Dirk ; KLEIN, Reinhard: Photo-realistic Rendering of Metallic Car Paint from Image-Based Measurements. In: *EUROGRAPHICS 27* (2008), Nr. 2
- [161] RUSHMEIER, H. ; WARD, G. ; PIATKO, C. ; P.SANDERS ; RUST, B.: Comparing Real and Synthetic Images: Some Ideas About Metrics. In: *1995 Eurographics Workshop on Rendering*. Springer-Verlag, 1995, S. 82–91
- [162] SAHASRABUDHE, N. ; WEST, J.E. ; MACHIRAJU, R. ; JANUS, M.: Structured spatial domain image and data comparison metrics. In: *Visualization '99. Proceedings*, 1999. – ISSN 1, S. 97 –515
- [163] SAUER, Sebastian: *Simulation von Witterung und dynamischen Effekten in der Echtzeit-Visualisierung der Fahrsimulation*, Universität Erlangen-Nürnberg, Diplomarbeit, 2007
- [164] SCHADENBERGER, R.: ADTF: Framework for Driver Assistance and Safety Systems. In: *International Congress of Electronics in Motor Vehicles 13 Baden-Baden*, 2007
- [165] SCHÖDLBAUER, A.: *Rechenformeln und Rechenbeispiele zur Landesvermessung Teil 2*. Herbert Wichmann Verlag Karlsruhe, 1981
- [166] *Kapitel* Kamerabasierte Fußgängerdetektion. In: SCHIELE, Bernt ; WOJEK, Christian: *Handbuch Fahrerassistenzsysteme - Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort*. Vieweg+Teubner | GWV Fachverlage GmbH, 2009, S. 223–236
- [167] SCHLICK, Christophe: An Inexpensive BRDF Model for Physically-Based Rendering. In: *Comput. Graph. Forum* 13 (1994), Nr. 3, S. 233–246
- [168] SCHMIDT, Carsten: *Hardware-in-The-Loop gestützte Entwicklungsplattform für Fahrerassistenzsysteme: Modellierung und Visualisierung des Fahrzeugumfelds*, Universität Kassel, Diss., 2011

-
- [169] SCHMIDT, Christian: *Hardware-in-The-Loop gestützte Entwicklungsplattform für Fahrerassistenzsysteme: Analyse und Generierung kritischer Verkehrsszenarien*, Universität Kassel, Diss., 2010
- [170] SCHMIDT, Florian: *Funktionale Absicherung kamerabasierter Aktiver Fahrerassistenzsysteme durch Hardware-in-the-Loop-Tests*, Universität Kaiserslautern, Diss., 2012
- [171] SCHREER, Oliver: *Stereoanalyse und Bildsynthese*. Berlin ; Heidelberg ; New York : Springer, 2005
- [172] SCHÄUFFELE, Jörg ; ZURAWKA, Thomas: *Automotive Software Engineering - Grundlagen, Prozesse, Methoden und Werkzeuge effizient einsetzen*. Vieweg+Teubner, Wiesbaden, 2010
- [173] SHIMIZU, S. ; KONDO, T. ; KOHASHI, T. ; TSURATA, M. ; KOMURO, T.: A new algorithm for exposure control based on fuzzy logic for video cameras. In: *Consumer Electronics, IEEE Transactions on* 38 (1992), aug, Nr. 3, S. 617–623. – ISSN 0098–3063
- [174] SHIRLEY, Peter ; ASHIKHMIN, Michael ; GLEICHER, Michael ; MARSCHER, Spethan R. ; REINHARD, Erik ; SUNG, Kelvin ; THOMPSON, William B. ; WILLEMSSEN, Peter ; SHIRLEY, Peter (Hrsg.): *Fundamentals of Computer Graphics*. Second Edition. A K Peters, Ltd., 2005
- [175] *Kapitel The Framebuffer*. In: SHREINER, Dave ; WOO, Mason ; NEIDER, Jackie ; DAVIS, Tom: *OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 2.1*. Add, 2007, S. 465–504
- [176] SLOUP, Jaroslav: A survey of the modelling and rendering of the earth’s atmosphere. In: *Proceedings of the 18th spring conference on Computer graphics*. New York, NY, USA : ACM, 2002 (SCCG ’02). – ISBN 1–58113–608–0, S. 141–150
- [177] STILLER, Christoph (Hrsg.): *Fahrerassistenzsysteme mit maschineller Wahrnehmung*. Springer-Verlag Berlin Heidelberg, 2005
- [178] *Kapitel Maschinelles Sehen*. In: STILLER, Christoph ; BACHMANN, Alexander ; DUCHOW, Christian: *Handbuch Fahrerassistenzsysteme - Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort*. Vieweg+Teubner | GWV Fachverlage GmbH, 2009, S. 198 – 222
- [179] SUN, Zehang ; BEBIS, George ; MILLER, Ronald: On-Road Vehicle Detection: A Review. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28 (2006), May, Nr. 5, S. 694 – 710
- [180] TELLMANN, D. ; SCHMIDT, Ca. ; SCHMIDT, Ch. O. ; AYEYB, M. ; THEUERKAUF, H.: Echtzeitfähige Simulationsumgebung zum Test von Fahrerassistenzsystemen. In: *VDI-Berichte*, 2006, S. 271 – 280
- [181] TIDEMAN, Martijn: Scenario-based Simulation Environment for Assistance Systems. In: *ATZ Autotechnology* 01 (2010), S. 32 – 36
- [182] TNO: Pre Scan 5.4 Datenblatt. http://www.tass-safe.com/public/files/112011/PreScan_datasheet_v3.pdf. – Forschungsbericht
- [183] VANHOLME, Benoit ; GRUYER, Dominique ; GLASER, Sebastien ; MAMMAR, Saïd: Fast prototyping of a Highly Autonomous Cooperative Driving System for public roads. In: *IEEE Intelligent Vehicles Symposium*, 2010
-

- [184] VAUDREY, Tobi ; RABE, Clemens ; KLETTE, Reinhard ; MILBURN, James: Differences Between Stereo and Motion Behavior on Synthetic and Real-World Stereo Sequences. In: *23rd International Conference of Image and Vision Computing New Zealand (IVCNZ '08)*, 2008, S. 1–6
- [185] VIOLA, Paul ; JONES, Michael: Robust Real-time Object Detection. In: *Second International Workshop on Statistical and Computational Theories of Vision - Modeling, Learning, Computing and Sampling*. Vancouver, Canada, July 2001
- [186] VIRES: *Virtual Test Drive Manual*. I, August 2010
- [187] *Kapitel Software-Entwicklung*. In: WALLENTOWITZ, Henning (Hrsg.) ; REIF, Konrad (Hrsg.): *Handbuch Kraftfahrzeugelektronik - Grundlagen, Komponenten, Systeme, Anwendungen*. Friedr. Vieweg & Sohn Verlag, GWV Fachverlage GmbH, 2006, S. 651 – 684
- [188] WANG, Zhou ; BOVIK, A.C.: A universal image quality index. In: *Signal Processing Letters, IEEE* 9 (2002), march, Nr. 3, S. 81 –84. – ISSN 1070–9908
- [189] WANG, Zhou ; BOVIK, A.C. ; SHEIKH, H.R. ; SIMONCELLI, E.P.: Image quality assessment: from error visibility to structural similarity. In: *Image Processing, IEEE Transactions on* 13 (2004), april, Nr. 4, S. 600 –612. – ISSN 1057–7149
- [190] WARD, Gregory J.: Measuring and modeling anisotropic reflection. In: *SIGGRAPH Comput. Graph.* 26 (1992), Juli, Nr. 2, S. 265–272. – ISSN 0097–8930
- [191] WEINRAUCH, David ; FREESE, Holm ; GIERSIEFER, Andreas: Einsatz von Hardware-in-the-Loop zur Verifikation von Fahrerassistenzsystemen in der Praxis. In: *3. AutoTest*, 2010
- [192] *Kapitel Parallax Mapping*. In: WELSH, Terry: *Shaderx3 Advanced Rendering with DirectX and OpenGL*. Course Technology, 2004, S. 89–96
- [193] WENG, Juyan ; MARC HERNIOU, Paul C.: Camera Calibration with Distortion Models and Accuracy Evaluation. In: *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE* 14 (1992), S. 965–980
- [194] WILLIAMS, Lance: Casting curved shadows on curved surfaces. In: *ACM SIGGRAPH Computer Graphics* 12 (1978), Nr. 3, S. 270–274
- [195] WILSON, DaleL. ; BADDELEY, AdrianJ. ; OWENS, RobynA.: A New Metric for Grey-Scale Image Comparison. In: *International Journal of Computer Vision* 24 (1997), S. 5–17. – ISSN 0920–5691
- [196] WINNER, Hermann (Hrsg.) ; HAKULI, Stephan (Hrsg.) ; WOLF, Gabriele (Hrsg.): *Handbuch Fahrerassistenzsysteme - Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort*. Vieweg+Teubner | GWV Fachverlage GmbH, 2009. – 43 – 54 S.
- [197] WOHLFAHRT, Christoph ; WEIZENEGGER, Florian ; SMUDA, Peer: HiL-Testtechnologie für kamerabasierte Fahrerassistenzsysteme. In: *3. AutoTest*, 2010

- [198] ZHOU, Hualin ; CHEN, Min ; WEBSTER, Mike F.: Comparative evaluation of visualization and experimental results using image comparison metrics. In: *Proceedings of the conference on Visualization '02*. Washington, DC, USA : IEEE Computer Society, 2002 (VIS '02). – ISBN 0–7803–7498–3, S. 315–322
- [199] ZOTTI, Georg ; WILKIE, Alexander ; PURGATHOFER, Werner: A Critical Review of the Preetham Skylight Model. In: *WSCG 2007 Short Communications Proceedings I*, 2007, S. 23–30

A. Anhang

A.1. Vergleich des Simulationslösungen

In Tabelle A.1 sind die Ergebnisse des Vergleichs der Simulationslösungen hinsichtlich der Erprobung von kameraunterstützten Fahrerassistenzsystemen dargestellt.

A.2. Koordinatensysteme

Im folgenden Abschnitt werden die verwendete Koordinatensysteme definiert, welche auf der DIN7000 basieren. Die lokalen Koordinatensysteme besitzen eine Translations- oder Rotationsbeziehung zum ortsfesten Weltkoordinatensystem (WKS).

A.2.1. Weltkoordinatensystem (WKS)

Das Ursprungskoordinatensystem stellt das ortsfeste Weltkoordinatensystem (WKS) dar. Aus diesen sind alle weiteren Koordinatensysteme abgeleitet. Das Koordinatensystem ist daher invariant gegenüber einer Translation und Rotation, siehe Abbildung A.2.

Ursprung	frei definierbar
x-Achse	frei definierbar
y-Achse	Orthogonal zu x-Achse, positiv nach links
z-Achse	Orthogonal zu x- und y- Achse, positiv nach oben
Winkel	Orientierung im Uhrzeigersinn in Vektorrichtung

Tabelle A.1.: Weltkoordinatensystem

A.2.2. Umfeldsensorikkoordinatensystem (USK)

Das Umfeldsensorikkoordinatensystem ist ein fahrzeugfestes Koordinatensystem, welche relativ zum WKS definiert ist.

X, Y- und Z-Achse formen ein Rechtssystem. X- und Y-Achse liegen in der Ebene der Straße. Alle drei Achsen sind somit nicht fest am Fahrzeug verankert. Nick-, Gier- und Wankbewegungen des Fahrzeugs beeinflussen nicht den Verlauf der Achsen, siehe Abbildung A.3.

Ursprung	Lotfußpunkt Mitte Hinterachse auf die Straße
x-Achse	Parallel zur dynamischen Fahrachse, positiv in Fahrtrichtung
y-Achse	Orthogonal zur x-Achse, somit quer zur Fahrachse, nach links positiv
z-Achse	Orthogonal zu x- und y- Achse, positiv nach oben
Winkel	Orientierung im Uhrzeigersinn in Vektorrichtung

Tabelle A.2.: Umfoldsensorikkoordinatensystem

A.2.3. Kamerakoordinatensystem (KOS)

Das Umfoldsensorikkoordinatensystem ist ein fahrzeugfestes Koordinatensystem, welche relativ zum WKS definiert ist. X, Y- und Z-Achse formen ein Rechtssystem. X- und Y-Achse liegen in der Ebene der optischen Achse. Alle drei Achsen sind somit nicht fest am Fahrzeug verankert. Nick-, Gier- und Wankbewegungen des Fahrzeugs beeinflussen den Verlauf der Achsen, siehe Abbildung A.4.

Ursprung	Optische Zentrum (Ursprung der Blendenöffnung)
x-Achse	Orthogonal zur z-Achse, somit quer zur optischen Achse, nach rechts positiv
y-Achse	Orthogonal zu x- und y- Achse, positiv nach unten
z-Achse	optischen Achse, positiv in Blickrichtung.
Winkel	Orientierung im Uhrzeigersinn in Vektorrichtung

Tabelle A.3.: Kamerakoordinatensystem

A.2.4. Straßenkoordinatensystem

Bei dem Straßenkoordinatensystem handelt es sich um ein ortsfestes Koordinatensystem, welches am Ursprung der Straße beginnt. s, t- und h-Achse formen ein Rechtssystem, siehe Abbildung A.5. X- und Y-Achse liegen auf der Straße.

Ursprung	Am Beginn des Straßensegments, ausgerichtet an Mittelinie der Straße
s-Achse	Entlang der Mittelinie der Straße
t-Achse	Orthogonal zur s-Achse, somit quer zur Mittelinie der Straße, nach links positiv
h-Achse	Orthogonal zu t- und h- Achse, positiv nach oben
Winkel	keine

Tabelle A.4.: Straßenkoordinatensystem

A.2.5. OpenGL-Koordinatensystem

Das OpenGL-Koordinatensystem (Canoical View Volume) Koordinatensystem ist relativ zum WKS definiert, vgl. Abbildung A.6. X, Y- und Z-Achse formen ein Rechtssystem. X- und Y-Achse liegen in der Ebene der optischen Achse. Alle drei Achsen sind somit nicht fest am Fahrzeug verankert.

Ursprung	Optische Zentrum (Ursprung der Blendenöffnung)
x-Achse	Orthogonal zur z-Achse, somit quer zur optischen Achse, nach rechts positiv
y-Achse	Orthogonal zu x- und y- Achse, positiv nach oben
z-Achse	optischen Achse, negativ in Blickrichtung.
Winkel	Orientierung im Uhrzeigersinn in Vektorrichtung

Tabelle A.5.: OpenGL-Koordinatensystem (Canoical View Volume)

A.3. Bewertung der Fahrzeuge

Zur Generierung einer globalen Aussage über die Ähnlichkeit der $i = 4$ Oberflächenmodelle $f_{r,j}$ bezüglich der realen Referenz wurde für die Menge $n = 6$ der Messungen in Bildbereichen $X_{n,i}$ ein Score S_n für jeden Messbereich ermittelt. Dafür wird zunächst für jedes der Oberflächenmodelle mittels der $j = 4$ berechneten Merkmale $m_{n,i,j}$ eine Rangfolge bzgl. der Scores $S_{n,i}$ in jeder Messung ermittelt.

- $m_{i,1}(X_{n,j}) = x_{\bar{Gray}}(X_{n,i}) + \sigma_{Gray}(X_{n,i})$
- $m_{i,2}(X_{n,j}) = 1 - R_{x,y,Gray}(X_{n,i})$
- $m_{i,3}(X_{n,j}) = x_{\bar{Grad}}(X_{n,i}) + \sigma_{Grad}(X_{n,i})$
- $m_{i,4}(X_{n,j}) = 1 - R_{x,y,Grad}(X_{n,i})$

Für den Merkmalsvektor $\mathbf{m}_{n,i}$ einer Messreihe n und dem Oberflächenmodell j folgt,

$$\mathbf{m}_{n,i} = [m_{n,i,1}(X_{n,i}), m_{n,i,2}(X_{n,i}), m_{n,i,3}(X_{n,i}), m_{n,i,4}(X_{n,i})]$$

Damit folgt für die Merkmalsmatrix M aller Oberflächenmodelle,

$$\mathbf{M}_n = \begin{bmatrix} \mathbf{m}_{n,1} \\ \vdots \\ \mathbf{m}_{n,i} \end{bmatrix}$$

Für den Score S_n jeder Messung folgt,

$$S_n(M_n) = \begin{bmatrix} s_{n,1,1}(m_{n,1,1}, \mathbf{m}_{n,1..i,1}) & s_{n,1,2}(m_{n,1,2}, \mathbf{m}_{n,1..i,2}) & \dots & s_{n,1,j}(m_{n,1,j}, \mathbf{m}_{n,1..i,j}) \\ s_{n,2,1}(m_{n,2,1}, \mathbf{m}_{n,1..i,1}) & s_{n,2,2}(m_{n,2,2}, \mathbf{m}_{n,1..i,2}) & \dots & s_{n,2,j}(m_{n,2,j}, \mathbf{m}_{n,1..i,j}) \\ \vdots & \vdots & \ddots & \vdots \\ s_{n,i,1}(m_{n,i,1}, \mathbf{m}_{n,1..i,1}) & s_{n,i,2}(m_{n,i,2}, \mathbf{m}_{n,1..i,2}) & \dots & s_{n,i,j}(m_{n,i,j}, \mathbf{m}_{n,1..i,j}) \end{bmatrix}$$

Die Funktion $s_{n,i,j}m, \mathbf{m}$ liefert als Ergebnis die Position des Elements m in der geordneten Menge \mathbf{m} zurück. Für den gesamtheitlichen Score eines Oberflächenmodells i in der Messung n folgt somit,

$$Sg_{n,i}(S_n) = \frac{1}{j} \sum_{p=1}^j s_{n,i,p}^2$$

Zur Verbesserung der Aussagekraft wird das Ergebnis quadratisch gewichtet.

A.4. Anforderungen an die Simulation

Basierend auf den in Kapitel 2.3 durchgeführten funktionalen Analysen verschiedener Algorithmen zur Spur-, Fahrzeug-, Verkehrszeichen- und Fußgängerdetektion wurden Eingangs-Anforderungen an die Simulationsumgebung definiert. Wenn gleich sich diese Dissertation im Kern mit der Spur- und Fahrzeugdetektion beschäftigt. Diese gliedern sich notwendige Anforderungen für die Darstellung eines funktionalen Tests und in erweiterte Anforderungen an die Simulation von Störgrößen für die Darstellung von Tests zur Robustheit der Systeme. Die Anforderungen werden in folgende Klassen eingeteilt, welche auch Untersuchungsobjekte darstellen:

- Geometrie und Oberflächeneigenschaften, definieren die geometrische Erscheinung eine Objekte und die Art-und-Weise der Simulation der Oberflächenqualität.
- Licht- & atmosphärische Simulation, definieren die Anforderungen die Simulation der Lichtausbreitung und das Verhalten der Atmosphäre
- Bildentstehungsprozess, Definiton von Anforderungen an die Bildgenerierung, welche ein Kamera-ähnliches Verhalten ermöglichen.
- Szenen und Szenario-Design, beinhaltet die Position von Objekten/3D-Geometrie und die Steuerung von Szenarien.

Diese Anforderungen werden als Grundlage für die Umsetzung und Bewertung von Modellen für die Abbildung der später definierten Eigenschaften verwendet. Die Definitionen der einzelnen Anforderungen zu den Überschriften kann im Anhang vorgefunden werden.

A.4.1. Geometrie- und Oberflächeneigenschaften

- Verschiedene Witterungsbedingungen sollen einen Einfluß auf die Oberflächeneigenschaften auswirken.
- Deformation und Alterung von Oberflächen.

A.4.1.1. Straßenmodell

- Beschreibung des Straßenverlaufs muss auf Basis von geometrischen Grundfunktionen erfolgen: Gerade, Kreisbogen, Klothoide.
- Die Breite der Straße und der Fahrspuren muss definierbar sein. Weiterhin muss diese kausal und lokal konstant sein.
- Die Erscheinung der Markierungen muss hinsichtlich Farbe und Helligkeit parametrierbar sein.
- Die Markierung muss einen hohen Kontrast zwischen Straße und Fahrbahn besitzen.

- Die Markierungen müssen verschiedene Breiten darstellen können
- die Markierungen müssen verschiedene Strich/-Lückenverhältnisse darstellen können.
- Die Straßenoberfläche muss unterschiedliche Beläge/Texturen besitzen können. (Variation)
- Die Straßenoberfläche soll eine Rauheit besitzen
- Die Straßenoberfläche kann ähnlich einer realen Straßen reflektieren (Diffuse & Spekular)
- Eine Nasse Straßenoberfläche soll spiegeln.
- Eine nasse Straßenoberfläche soll Fahrspuren aufweisen.
- eine Straße soll Spurrillen besitzen können.
- Es sollen Bürgersteige vorhanden sein können.
- Es müssen Bürgersteinkanten erstellbar sein können.
- Die Straße muss Leitplanken besitzen können.
- Die Oberfläche der Straße kann wellig sein.
- Es soll möglich sein Sprünge in der Straßenbreite zu simulieren.

A.4.1.2. Verkehrsobjekte (Objekterkennung)

- Die Geometrie von Fahrzeugen soll an reale Fahrzeuge angelehnt sein.
- Die Farbe der Fahrzeuge soll einfarbig ohne Reflexionen sein.
- Die Farbe der Fahrzeuge kann realitätsnah sein mit Reflexionen sein.
- Fahrzeuge sollen ein Kennzeichen aufweisen
- Fahrzeuge können Lichtquellen aufweisen tag
- Fahrzeuge müssen Lichtquellen aufweisen nacht
- Hoher Kontrast zwischen Fahrbahn und Straßen soll hoch sein.

A.4.2. Verkehrszeichen

- Die Symbole sollen die realen Vorbild in den entsprechenden Ländern entsprechen
- Farben und Intensitäten sollen der Realität entsprechen.
- Die Geometrie der Verkehrsschilder soll erhalten bleiben.

A.4.2.1. Umgebung/Vegetation

- Die Umgebung ist Modellierbar
- Die Umgebung ist positionierbar
- Das Shading der Umgebung ist veränderbar
- Der Kontrast der Umgebung ist veränderbar
- Die Helligkeit der Umgebung ist veränderbar

A.4.3. Beleuchtungssimulation

- Jedes Verkehrsobjekt muss einen Schatten aufweisen.
- Jedes Szeneobjekt muss durch Schatten verdeckt sein können.
- Jeder Szenenobjekt kann einen Schatten enthalten.
- Eine hinreichende Simulation des Himmels- und Umgebungslicht besitzen.
- Die Simulationsumgebung kann eine spektrale Verschiebung besitzen. Es muss eine Veränderungen der Farben darstellbar sein.
- Der Sonnenstand muss reproduzierbar sein.
- Fahrzeuge müssen mindestens einen Drop-Shadow beinhalten.
- Die Lichtbedingungen sollen veränderbar sein
- Die Lichtberechnung soll veränderbar sein.

A.4.4. Szenen und Szenario Design

- Der Straßenverlauf kann der Realität nach empfunden sein
- Es ist eine Repositionierung von Fahrzeugen in der Simulation möglich
- Die Eigenbewegung des Fahrzeug soll kausal und plausibel sein
- Es sollen verschiedene Witterungsbedingungen (Sonne, Bewölkung und Regen) darstellbar sein. Die Objekte sollen sich entsprechend Verhalten.
- Es sollen Tag/Nachtfahrten möglich sein.
- Die Bewegungen soll reproduzierbar sein.
- Es soll Verkehrsschilder frei positionierbar sein.
- Die Hintergründe sollen frei definierbar sein.
- Verdeckungen darstellbar

A.4.5. Bildentstehungsprozess

- Die Kamera muss eine intrinsische Kalibrierung verarbeiten können
- Die Kamera muss eine extrinsische Kalibrierung verarbeiten können
- Die Kamera muss eine konfigurierbare Auflösung besitzen
- Die Kamera soll ein Tiefenunschärfe aufweisen
- Die Kamera soll eine Abbildungskennlinie aufweisen
- Die Kamera soll die gleiche farbliche Abbildung aufweisen
- Die Kamera soll ein Rauschen beinhalten
- Die Belichtungsdauer sollen verstellbar sein.
- Die Belichtungsdauer soll eine endlich Zeit besitzen

A.4.6. Weitere Anforderungen an das Testsystem

- Das Testsystem soll eine hinreichend genaue Fahrdynamik besitzen.
- Die Fahrdynamik muss Nick- und Wankwinkel darstellen.

A.5. Szenengraphen

A.5.1. Fahrzeug

In Abbildung A.7 ist der Szenengraph der Fahrzeuge dargestellt.

A.6. Herleitung der Emission von Wasserpartikeln

Die Emission der Partikel kann als schiefer Wurf in einer gashaltigen Atmosphäre betrachtet werden. Mit Hilfe von geläufigen Überlegen kann das Problem wie folgt für eine numerische Lösung für den Zeitpunkt k formuliert werden. Als Ansatz wird die Kräftegleichung für einen Wassertropfen im freien Fall in \mathbb{R}^3 im Weltkoordinatensystem verwendet:

$$0 = F_L(k) + F_G - F_{Tropfen}(k) \tag{A.1}$$

Wobei F_L den Luftwiderstand, F_G die Gewichtskraft und $F_{Tropfen}$ die Kraft des Tropfens beschreibt. Ausformuliert folgt für diese Gleichung:

$$m \cdot \ddot{\mathbf{x}}(k) = -\frac{1}{2} \cdot \rho \cdot c_w \cdot A \cdot \dot{\mathbf{x}}(k-1)^2 - m \cdot \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} \quad (\text{A.2})$$

wobei m die Masse der Partikel, $c_w = 1$ den cw-Wert, ρ die Dichte der Luft, A die Fläche des Tropfens und g die Gravitationskraft darstellen. Durch Umstellen kann daraus die auf den Partikel einwirkende Beschleunigung $\ddot{\mathbf{x}}(k)$ formuliert werden:

$$\ddot{\mathbf{x}}(k) = -\frac{1}{2m} \cdot \rho \cdot c_w \cdot A \cdot \dot{\mathbf{x}}(k-1)^2 - \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} \quad (\text{A.3})$$

Daraus folgt für die Geschwindigkeit $\dot{\mathbf{x}}(k)$:

$$\dot{\mathbf{x}}(k) = \ddot{\mathbf{x}}(k) + \dot{\mathbf{x}}(k-1) \quad (\text{A.4})$$

und schließlich für den zurückgelegten Weg $\mathbf{x}(k)$

$$\mathbf{x}(k) = \ddot{\mathbf{x}}(k) + \dot{\mathbf{x}}(k-1) + \mathbf{x}(k-1) \quad (\text{A.5})$$

Als Startbedingungen für die Partikel werden die folgenden Werte verwendet,

$$\mathbf{x}(0) = \mathbf{x}_{\text{Reifen}} \quad (\text{A.6})$$

$\mathbf{x}_{\text{Reifen}}$ stellt die Position des Reifens in Weltkoordinaten bei $k = 0$ dar.

$$\dot{\mathbf{x}}(0) = \dot{\mathbf{x}}_{\text{Reifen}} \cdot \mathbf{R}_{\text{Partikel}} \quad (\text{A.7})$$

Die Geschwindigkeit der Partikel bei $k = 0$ definiert, sich aus der Geschwindigkeit des Rads in Laufrichtung $\dot{\mathbf{x}}_{\text{Reifen}}$. Im Folgenden wird diese mit der Fahrzeuggeschwindigkeit als identisch angekommen. Eine Substitution gegen Geschwindigkeit jedes Rads ist denkbar. Die Emissionsrichtung in Weltkoordinaten des Partikel $\text{vect}R_{\text{Partikel}}$ folgt aus der Multiplikation Rotationsmatrizen des Fahrzeugs, des Rads und Emissionswinkels des Partikels. Für den Azimutwinkel ϕ wird ein Intervall von $[-22.5^\circ 22.5^\circ]$ angenommen, da Reifen das meiste Wasser zur Seite ableiten. Zur Berechnung des Polarwinkels wird der hintere Übergang um $0.5m$ verlängert und die Höhe der Unterkante der Stoßstange verwendet, siehe Abbildung A.8. Durch die Verlängerung soll der Umstand berücksichtigt werden, dass die Partikel bei $\theta < 45^\circ$ die Oberfläche verlassen. Mittels des Arkustangens folgt ein $\theta = 75.96^\circ$ für die obere Grenze, der Intervall für den Polarwinkel definiert sich damit zu $\theta = [75.96^\circ 90^\circ]$. Die Beschleunigung der Partikel wird bei diesem Modell als Null angekommen.

$$\ddot{\mathbf{x}}(0) = \mathbf{0} \quad (\text{A.8})$$

	Ohio State		RWTH Aachen	Kassel	IPG Automotive			
Entwicklungsstand	VAT SIM 2000		Pelops 2002	Decos 2010	CarlMaker 2011	SIVIC 2011	MB-Tech Provetech:VL 2011	
Bildsynthese								
Renderingmethode	Rasterisierung	Rasterisierung	Rasterisierung	Rasterisierung	Rasterisierung	Rasterisierung	Rasterisierung	Rasterisierung
Echtzeitfähigkeit	Ja	Ja	funktional	funktional	Ja	Ja	Ja	Ja
Realismusgrad	funktional	funktional	Nein	funktional	funktional	realitätsnah	realitätsnah	realitätsnah
Simulation der Kamera	Nein	Nein		Nein	Nein	Ja	Nein	Nein
Beleuchtungs simulation								
Atmosphärische Simulation	Nein	Nein	Nein	Nein	Nein	Ja	Ja	Ja
Schattenberechnung	Nein	Nein	Nein	Ja	Nein	Ja	Ja	Ja
Geometrie								
Straßenmodell	eingeschränkt	Nein	Nein	Ja	Ja	unbekannt	Ja	Ja
Fahrzeugeometrie	Nein	Nein	Nein	Nein	Ja	Ja	Ja	Ja
	TASS	Vires	dSpace	Tesis GmbH	Daimler			
Entwicklungsstand	PreScan 2011	Virtual Test Drive 2012	ASM Traffic 2011	Dynad 2011	Eisats 2011			
Bildsynthese								
Renderingmethode	Rasterisierung/ Ray-Tracing	Rasterisierung/ Ray-Tracing	Rasterisierung	Rasterisierung	Raytracing			
Echtzeitfähigkeit	Ja/Nein	Ja/Nein	Ja	Ja	Nein			
Realismusgrad	realitätsnah	realitätsnah	funktional	funktional	photorealismus			
Simulation der Kamera	Ja	Nein	Nein	Nein	Ja			
Beleuchtungs simulation								
Atmosphärische Simulation	Ja	Ja	Nein	Nein	Ja			
Schattenberechnung	Ja	Ja	Nein	Nein	Ja			
Geometrie								
Straßenmodell	Ja	Ja	Ja	Ja	Nein			
Fahrzeugeometrie	Ja	Ja	Ja	Ja	Ja			

Abbildung A.1.: Vergleich der Simulationslösungen hinsichtlich der Erprobung von kameraunterstützten Fahrerassistenzsystemen

A.6. HERLEITUNG DER EMISSION VON WASSERPARTIKELN

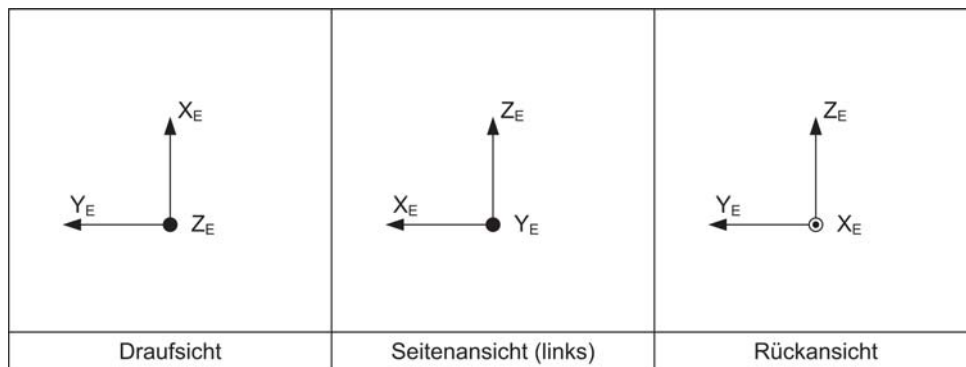


Abbildung A.2.: Weltkoordinatensystem (WKS)

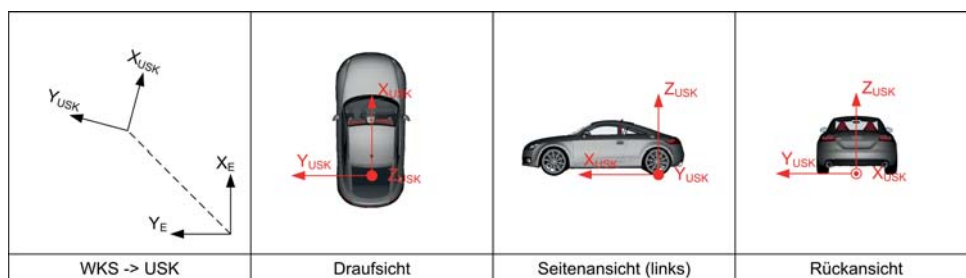


Abbildung A.3.: Umfeldsensorikkordinatensystem (USK)

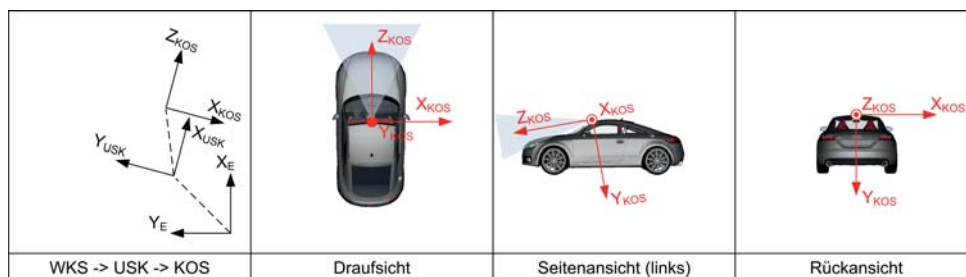


Abbildung A.4.: Kamerakordinatensystem (KOS)

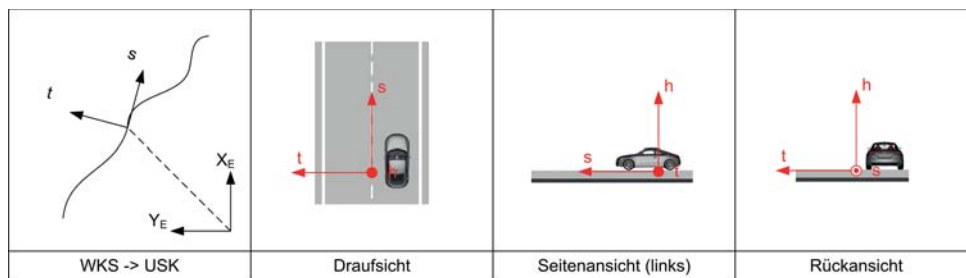


Abbildung A.5.: Straßenkoordinatensystem

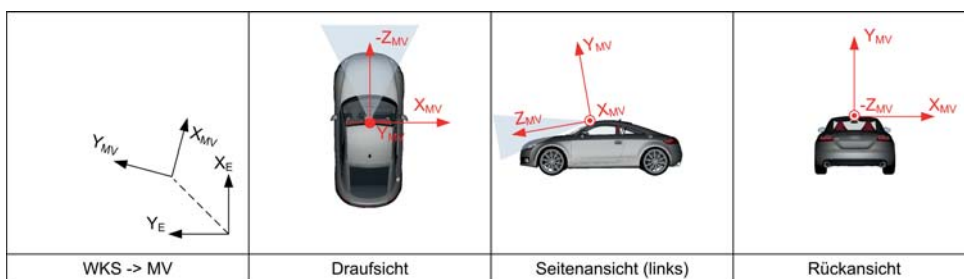


Abbildung A.6.: OpenGL-Koordinatensystem (Canoical View Volume)

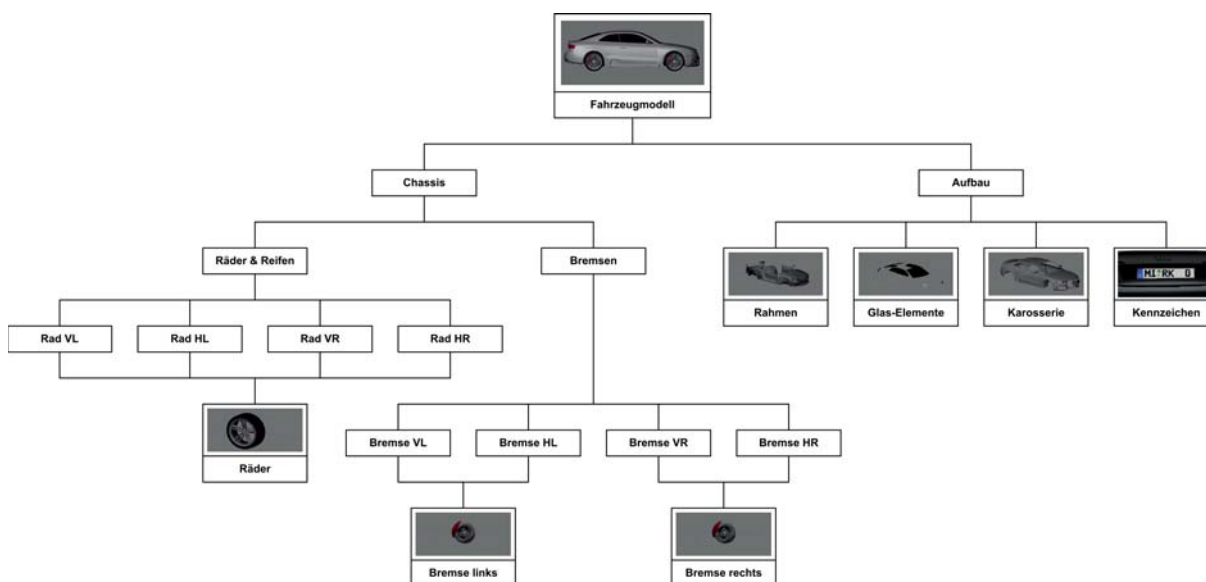


Abbildung A.7.: Szenengraph der Fahrzeugmodelle

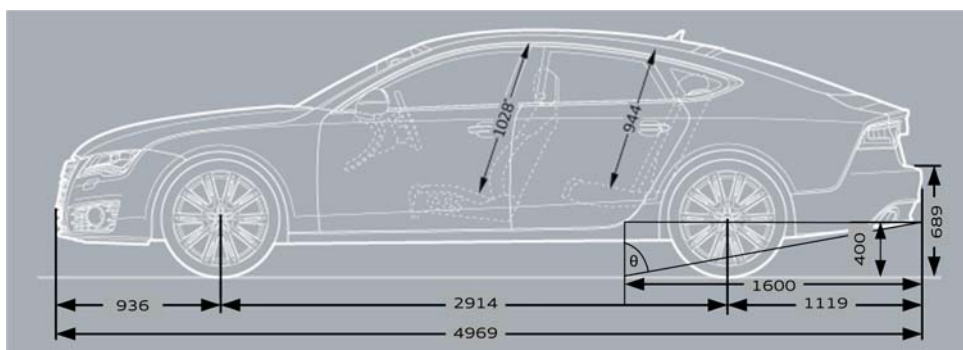


Abbildung A.8.: Berechnung des Deklinationswinkels

