

Analyse des situativen Fahrerverhaltens zur benutzeradaptiven Assistenz

Zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften

von der Fakultät für Informatik
des Karlsruher Instituts für Technologie (KIT)

genehmigte

Dissertation

von

Dipl.-Inform Tobias Bär

aus Offenburg

Tag der mündlichen Prüfung: 21. April 2016

Erster Gutachter: Prof. Dr.-Ing. J. Marius Zöllner

Zweiter Gutachter: Prof. Dr.-Ing. Rainer Stiefelhagen

Danksagung

Eine Promotion gelingt nicht im Alleingang. Man ist auf die Hilfe und Unterstützung vieler Personen angewiesen, bei denen ich mich an dieser Stelle herzlich bedanken möchte.

Ganz herzlich bedanke ich mich bei meinen Schwiegereltern Elena und Alexander sowie meinen Eltern Walter und Luitgard. Sie haben an zahlreichen Wochenenden den weiten Weg nach Karlsruhe auf sich genommen um auf Larissa und Kevin aufzupassen während meine Frau arbeitete und ich am FZI meine Dissertation schrieb.

Meine fleißigen Studenten Martin Pfeffer, Oleg Löwen, Jan Aidel, Jan-Felix Reuter, Denys Linke, Christian Voigt und Florian Richter haben durch Ideen in ihren Diplom-, Master und Bachelorarbeiten einen immensen Beitrag zum Gelingen dieser Arbeit geleistet.

Projekte in der Informatik sind in den seltensten Fällen Einzelprojekte. Sie bauen auf anderen Arbeiten und Ergebnissen auf, benutzen Softwarekomponenten und Bibliotheken anderer. Für eine prima Zusammenarbeit, sowohl bei der Softwareentwicklung, als auch im Austausch von Ideen und *Best-Practices* möchte ich mich beim IDS/TKS-Team des FZIs als auch der Uni-Gruppe des IAIM bedanken: Ralf Kohlhaas, Dennis Nienhüser, Thomas Gump, Thomas Schamm, Marcus Strand, Arne Rönnau, Steffen Rühl, Andreas Herrmann, Jan Oberländer, Thilo Kerscher, Mark Zofka, Christian Regenstein, Clemens Birkenhöfer, Georg Heppner, Lars Pforzer, Sebastian Klemm, Florian Kuhnt, Marco Ziegenmeier Steffen Rühl, Xhixing Xue Michael Göller, Stefan Scherzinger, Michael Weber, Klaus Uhl, Sonja Göttl, Manuel Yguel, Diana Reiter, Martin Lösch, Tobias Gindele und Sebastian Brechtel.

Für das Korrekturlesen dieser Arbeit bedanke ich mich bei meinem Vater, meiner Schwester Pamela und Dennis Nienhüser. Besonderes Lob gilt Dennis Nienhüser, der den kompletten Text akribisch Korrektur gelesen hat. Für seine wertvolle Hilfe und seine zahlreichen Verbesserungsvorschläge, Hinweise und Gedanken bin ich sehr dankbar.

Vielen Dank an Herrn Stiefelhagen für die Übernahme des Korreferats und die nützlichen Ratschläge im Themengebiet der Blickrichtungserkennung.

Bei meinem Doktorvater J. Marius Zöllner möchte ich mich außerordentlich bedanken. Er hatte trotz chronisch überfülltem Schreibtisch immer ein offenes Ohr für meine Promotion und gab durch sein visionäres Denken meiner Arbeit immer wieder neue Impulse.

Zuletzt möchte ich mich bei meiner Frau Jenia und meinen Kindern Kevin und Larissa bedanken. Gerade in der Endphase der Promotion mussten das ein oder andere Wochenende ohne mich verbringen. Sie hatten für mein Vorhaben immer Verständnis und boten mir die größtmögliche Unterstützung. Danke. Ich liebe Euch.

Karlsruhe, im Juni 2015

Inhaltsverzeichnis

Danksagung	i
Symbolverzeichnis	vii
1 Einleitung	1
1.1 Ziel- und Aufgabenstellung	2
1.2 Wissenschaftlicher Beitrag	3
1.3 Aufbau der Arbeit	3
2 Der Fahrer im Automobil der Zukunft - Aktuelle Forschung	5
2.1 Blickrichtungserkennung - Schätzung von Kopf- und Augenstellung	8
2.1.1 CCD-Kamera in Kombination mit IR-Beleuchtung	8
2.1.2 CCD-Kamera	9
2.1.3 RGB-D-Kamera	11
2.1.4 Zusammenfassung und Diskussion	14
2.2 Analyse des Blickverhaltens	16
2.2.1 Wohin blickt der Fahrer während der Fahrt?	16
2.2.2 Analyse der Blickrichtung zur Erkennung von Gefahren	18
2.2.3 Fahraufgabenspezifische Aufmerksamkeitsschätzung über Salienzkarten	21
2.2.4 Einzelarbeiten zur Bestimmung von Aufmerksamkeit und kognitiver Belastung	22
2.2.5 Zusammenfassung und Bewertung	24
2.3 Die Analyse des Fahrverhaltens	25
2.3.1 Fahrstil und Risikobereitschaft	25
2.3.2 Komfortparameter	26
2.3.3 Zusammenfassung und Bewertung	28
2.4 Prognose des Fahrverhaltens	28
2.4.1 Prognose von Stopp- und Bremsmanövern	29
2.4.2 Prognose der Abbiegerichtung an einer Abzweigung	31
2.4.3 Prognose von Fahrspurwechseln auf der Autobahn	31
2.4.4 Generische Manöverprognose	32
2.4.5 Zusammenfassung und Bewertung	33
2.5 Modellierung von Fahrerverhalten - Wissensrepräsentation und Inferenz	34
2.5.1 Fahrerverhalten modelliert über Dynamic Belief Networks (DBNs)	34
2.5.2 Fahrerverhalten modelliert über die ACT-R Architektur	35
2.5.3 Zusammenfassung und Bewertung	36
2.6 Zusammenfassung und Identifikation des Forschungsbedarfs	37

3	Blickrichtungserkennung im Automobil	39
3.1	Anforderungen an die Blickrichtungserkennung im Automobil	39
3.2	Blickrichtungserkennung über eine RGB-D-Kamera	41
3.2.1	Eignung der RGB-D-Kamera zur Blickrichtungserkennung im Automobil	41
3.2.2	Bestimmung der Kopfpose über das Iterative Closest Point Verfahren	48
3.2.3	Gesichtsmasken Management	55
3.2.4	Schätzung der Kopfpose bei überbelichtetem oder fehlendem Tiefenbild	57
3.2.5	Bestimmung der Pupillenstellung	63
3.3	Blickrichtungserkennung über CCD-Kameras	65
3.3.1	Lokalisierung markanter Gesichtsmerkmale	66
3.3.2	Schätzung der Kopfpose über die erkannten Gesichtsmerkmale	67
3.4	Abbildung der Blickrichtung auf die Geschehnisse innerhalb und außerhalb des Fahrzeugs	69
3.4.1	Abbildung des Blicks auf Entitäten innerhalb und außerhalb des Fahrzeugs	71
3.4.2	Bewusste Wahrnehmung - Aufmerksamkeitskonfidenz	75
3.4.3	Bildung von prädikatenlogischen Aussagen	77
3.5	Zusammenfassung	78
4	Analyse des Fahrerverhaltens	81
4.1	<i>Markov Logik Netze</i> für die Modellierung des Fahrerverhaltens	83
4.2	Fahrstilerkennung	85
4.2.1	Fahrstilerkennung über ein <i>Fuzzy-Inferenzsystem</i>	89
4.2.2	Fahrstilerkennung über <i>Markov Logik Netze</i>	92
4.2.3	Vergleich der Fahrstilerkennung über <i>Fuzzylogik</i> und <i>Markov Logik Netze</i>	96
4.3	Statistische Modellierung des Fahrverhaltens	97
4.3.1	Schätzung von Komfortparametern	97
4.4	Analyse des Blickrichtungsverhaltens	99
4.5	Prognose zukünftiger Manöver und Aktionen	101
4.6	Zusammenfassung	104
5	Evaluation und Ergebnisse	107
5.1	Blickrichtungserkennung hinsichtlich den Ansprüchen der Automobilindustrie	107
5.1.1	Genauigkeit und Erkennungsbereich der RGB-D-Kopfposenschätzung	108
5.1.2	Schnelle Kopfbewegungen	109
5.1.3	Bestimmung des Aufmerksamkeitsfelds im Fahrzeuginnenraum	110
5.1.4	Kompensation der Überbelichtung durch Kopfposenschätzung aus dem RGB-Farbbild	113
5.1.5	Schätzung der Kopfpose über CCD-Kamerabilder	115
5.2	Fahrstil	120
5.2.1	Fahrstilerkennung über Fuzzyregeln	121
5.2.2	Fahrstilerkennung über <i>MLNs</i>	124
5.2.3	Vergleich zwischen der Fahrstilklassifikation über <i>MLNs</i> und <i>Fuzzylogik</i>	128
5.3	Verbesserung der Reichweitenprognose über fahrerindividuelle Komfortparameter	130
5.3.1	Fahrerindividuelle Verbrauchsschätzung auf einer Autobahnfahrt	131

5.3.2	Verbrauchsprognose für unterschiedliche Fahrer	133
5.4	Zustandserkennung und Prognose des Fahrverhaltens	134
5.4.1	Konzeption des <i>MLNs</i>	136
5.4.2	Trainieren des Netzes	137
5.4.3	Bestimmung der aktuellen Handlungsphase - Inferenz	138
5.4.4	Prognose des Fahrspurwechsels	140
5.5	Zusammenfassung	141
6	Schlussbetrachtung	143
6.1	Zusammenfassung	143
6.2	Diskussion	144
6.3	Ausblick	145
A	Quellennachweise	147
A.1	Literatur	147

Symbolverzeichnis

Konventionen

Skalare	klein, nicht fett, kursiv: x, y, \dots
Vektoren	klein, fett, nicht kursiv: $\mathbf{x}, \mathbf{y}, \dots$
Mengen	groß, nicht fett, kursiv: X, Y, \dots
Matrizen	groß, fett, nicht kursiv: $\mathbf{X}, \mathbf{Y}, \dots$
Punktwolken als spezielle Mengen	groß, nicht fett, kursiv: P
Punkte als spezielle Vektoren	klein, fett, nicht kursiv: \mathbf{p}
Transformationen als spezielle Matrizen	groß, fett, nicht kursiv mit tiefgestelltem <i>von</i> A und hochgestelltem <i>nach</i> B : ${}^A\mathbf{T}_B$
Koordinatensysteme als spezielle Matrizen [$\mathbf{R} \mathbf{t}$] relativ zu einem anderen Koordinatensystem	groß, nicht fett, Blockschrift A
Bilder als spezielle Matrizen	\mathbf{B}
Funktionen	$F()$ oder $f()$
Klassen	kursiv, in eckigen Klammern: $[positiv]$

Zur Verbesserung der Lesbarkeit werden Mengenklammern nur für Mengen einer Kardinalität von zwei oder mehr geschrieben; für Mengen geringerer Kardinalität werden sie ausgespart, wenn aus dem Kontext ersichtlich ist, dass es sich um eine Menge handelt.

In Konfusionsmatrizen wird die (absolute oder relative) Häufigkeit von Klassifikationsergebnissen eingetragen. Pro Klasse wird jeweils eine Spalte und eine Zeile verwendet. Ein Eintrag m_{ij} in der Konfusionsmatrix \mathbf{M} gibt die Häufigkeit an, mit der Objekte der Klasse in Spalte j als Objekte der Klasse in Reihe i klassifiziert werden. Freie Einträge in der Matrix stehen für die Häufigkeit 0. Beispielsweise kann der Konfusionsmatrix

	ω_0	ω_1	ω_2
ω_0	23		3
ω_1		42	
ω_2			7

entnommen werden, dass 42 Objekte der Klasse ω_1 korrekt als ω_1 , 3 Objekte der Klasse ω_2 als ω_0 und keine Objekte der Klasse ω_0 als ω_1 klassifiziert wurden.

Symbole und Operatoren

\mathbf{x}	Vektor \mathbf{x} mit Komponenten $\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$
$\mathbf{M}^{m,n}$	Matrix mit m Zeilen und n Spalten und Komponenten (m_{ij})
$\mathbf{M}(a : b; c : d)$	Blockmatrix zu \mathbf{M}
2^X	Potenzmenge zu X
\emptyset	leere Menge
$P(x)$	Wahrscheinlichkeitsmaß
$P(x y)$	Bedingte Wahrscheinlichkeit
$\mathcal{N}_p(\mathbf{x}, \mathbf{Y})$	Dichtefunktion der p -dimensionalen Normalverteilung mit Erwartungswert \mathbf{x} und Kovarianzmatrix \mathbf{Y}
$x^{(t)}$	x zum Zeitpunkt t
H	Koordinatensystem
\mathbf{x}_H	\mathbf{x} im Koordinatensystem H
$\text{sgn}(x)$	Vorzeichenfunktion mit $\text{sgn}(x) := \begin{cases} -1, & x < 0 \\ +1, & x \geq 0 \end{cases}$
$\exp(x)$	Exponentialfunktion mit $\exp(x) := e^x$
$ x $	Betrag von x
$ M $	Anzahl der Elemente der Menge M
$ \mathbf{M} $	Determinante der Matrix \mathbf{M}
$\ \mathbf{x}\ $	Euklidische Norm von \mathbf{x}
$\langle \mathbf{x}, \mathbf{y} \rangle$	Skalarprodukt von \mathbf{x} und \mathbf{y}
$\mathcal{O}(f)$	Funktionenklasse asymptotischer oberer Schranken (O-Kalkül, Landau-Symbol)
$x \gg y$	x ist deutlich (mehrere Größenordnungen) größer als y
$x \bmod y$	Rest der Division von x und y mit $x \bmod y = x - y \lfloor \frac{x}{y} \rfloor$
$\mathbf{A} * \mathbf{B}$	Faltung von \mathbf{A} und \mathbf{B}
\top	Wahrheitswert TRUE
\perp	Wahrheitswert FALSE

1. Einleitung

Bei der Entwicklung zukünftiger Fahrerassistenzsysteme (FAS) kommt der individuellen Anpassung des Assistenzsystems an den Fahrer eine immer größer werdende Bedeutung zu. Systeme, deren Warnhinweise und Entscheidungen darauf beruhen, wie sich ein durchschnittlicher Fahrer verhalten würde, verschenken Potenzial in puncto Sicherheit und Kundenakzeptanz. Warnt ein Kollisionswarnsystem den Fahrer unberücksichtigt seiner Fähigkeiten permanent zu früh, werden die Warnungen als unwichtig empfunden und der *entscheidende Hinweis* möglicherweise ignoriert. Warnt ein Kollisionswarnsystem den Fahrer hingegen zu spät, kann ein Unfall möglicherweise nicht mehr vermieden werden.

Doch nicht nur unter dem Gesichtspunkt der Sicherheit ergibt sich die Notwendigkeit Fahrerassistenzsysteme fahreradaptiv zu gestalten. Nach den Vorstellungen der Automobilindustrie soll ein Fahrerassistenzsystem der nächsten Generation die Funktion eines *intelligenten Beifahrers* übernehmen, welcher auf den Fahrer abgestimmte Fahrhinweise in Bezug auf Sicherheit, Komfort und Energieeffizienz gibt. Was zeichnet aber einen intelligenten Beifahrer aus?

Ein intelligenter Beifahrer muss selbst ein Verständnis für die momentane Verkehrssituation haben, antizipationsfähig sein und die Fähigkeiten des Fahrers, wie beispielsweise seine Reaktionszeit, seine Risikobereitschaft oder sein Bremsverhalten, einschätzen können. Nur so kann der Beifahrer entscheiden, ob sich der Fahrer noch innerhalb der Normalität bewegt oder ob die Verkehrssituation einer außerordentlichen Handlung oder eines Warnhinweises bedarf.

Neben dem intelligenten Beifahrer wird von Forschungsgruppen und Industrie schon länger die Vision von *autonom agierenden Fahrzeugen* vorangetrieben. Fahrzeuge sollen fahrerlos Wegstrecken abfahren und ihre Verhaltensentscheidungen dabei gestützt durch Umfeldsensorik und gelernten Interaktionsregeln treffen. Doch rücken, bedingt durch die Substitution des Fahrers durch Algorithmen und Regelkreise, Themen wie die Verhaltensschätzung des Fahrers in den Hintergrund? Im Gegenteil: Mehr denn je werden die Verhaltensmuster des Fahrers und dessen Kriterien zur Entscheidungsfindung erforscht. Betrachtet man den von namhaften Automobilherstellern und -zulieferern aufgestellten Meilensteinplan [Aut14] zum *Autonomen Fahren* findet man darauf die Evolutionsschritte *Teilautomatisiertes Fahren*, *Hochautomatisiertes Fahren* und *Vollautomatisiertes Fahren*. In den ersten beiden Stufen übernimmt das Fahrzeug die monotonen, einfachen Fahraufgaben und gibt komplexe Fahraufgaben an den Fahrer ab. Fahrer und Fahrzeug üben die Fahraufgabe vermehrt kooperativ aus, was intuitive Übergabestrategien zwischen Mensch und Fahrzeug sowie eine maschinelle Einschätzung der Verkehrssituation und des Situationsbewusstseins des Fahrers fordert.

Um die fahrerindividuelle Einschätzung einer Verkehrssituation maschinell umzusetzen, muss verstärkt auf die Beobachtung des Fahrers, die Erkennung seines Zustands und seiner Intention sowie die Modellierung des Fahrerverhaltens gesetzt werden. Nur so können die Evolutionsschritte in der Automatisierung des Fahrens überhaupt umgesetzt werden.

1.1. Ziel- und Aufgabenstellung

Die primäre Aufgabenstellung und Zielsetzung der Arbeit lässt sich als Forschungsfrage formulieren:

Wie kann das Verhalten des Fahrers beobachtet und analysiert werden, so dass Fahrerassistenzsysteme der Zukunft individuell auf die Eigenschaften des Fahrers abgestimmt werden können?

Das Ziel der Arbeit ist die Erarbeitung von Konzepten zur Optimierung zukünftiger Fahrerassistenzsysteme in Bezug auf individuelle Eigenschaften und Fähigkeiten des Fahrers. Die Optimierungsmöglichkeit soll sich dabei nicht auf ein spezielles FAS beschränken; infolgedessen ergibt sich als Zielsetzung die maschinelle Bereitstellung von Wissen in Form einer Verhaltens- und Charakterbeschreibung des Fahrers basierend auf Modellen und Parametern.

Als Aufgabenstellung steht einerseits die *sensorielle Erfassung des Fahrers (Beobachtung oder auch Perzeption)* und andererseits die *Analyse des Fahrerverhaltens* im Vordergrund.

Sensorielle Erfassung des Fahrers (Perzeption) Elementar für die gestellte Aufgabe ist die sensorielle Erfassung des Fahrers, um dessen Informationsaufnahme sowie dessen Interaktion mit dem Fahrzeug und seinem lokalen Fahrzeugumfeld zu detektieren.

Einen besonderen Aufgabenschwerpunkt ist die Entwicklung einer speziell auf die Gegebenheiten der dynamischen Domäne „Automobil“ optimierten Blickrichtungserkennung, über die das Blickverhalten des Fahrers in Bezug zu den Bedien- und Anzeigekomponenten des Fahrzeugs, den Geschehnissen im lokalen Fahrzeugumfeld und der Verkehrsinfrastruktur gesetzt wird.

Analyse des Fahrerverhaltens Der zweite Aufgabenschwerpunkt ist die Analyse des Fahrerverhaltens in Form von Verhaltensmodellen und Algorithmen. Ein Modell im Sinne der Informatik reduziert einen komplexen Sachverhalt auf das Wesentliche, um an diesem beschränkten Abbild der Wirklichkeit Aspekte darzustellen, zu erklären oder zu schlussfolgern.

In dieser Arbeit soll das individuelle und nicht-deterministische Verhalten des Menschen (Fahrer) modelliert werden, um aus dem aufgestellten Modell die Eigenschaften des Fahrers sowie zukünftiges Verhalten oder Fehlverhalten zu schließen. Bedingt durch die Verarbeitung sensoruell erfasster Informationen müssen die entwickelten Modelle imstande sein mit Unsicherheiten in der Wahrnehmung und einer nur partiell erfassten Umwelt umzugehen.

Randbedingungen Randbedingung der Aufgabenstellung ist die technische Eignung der entwickelten Algorithmen und Modelle für den Automobilbereich. Dies bedeutet, dass die gewählten Sensoren zur Fahrer- und Umwelterfassung den Fahrer während der Fahrt nicht beeinflussen oder behindern dürfen. Die entwickelten Algorithmen und Verfahren müssen mit den Gegebenheiten einer Fahrt, wie wechselnden Lichtverhältnissen, Vibrationen oder einer eingeschränkten sensoruellen Erfassung der Umwelt, umgehen können.

Trotz des Bestrebens nach für die Automobilindustrie geeigneten Lösungen wird von einer marktreifen

Produktentwicklung Abstand genommen. Bewusst spielen bei den entwickelten Verfahren Produktionskosten, Zertifizierungen oder die Marktreife von Sensoren eine untergeordnete Rolle. Auch eine Evaluation der entwickelten Algorithmen über mehrere tausend Testkilometer, wie oft in Zertifizierungen für den Straßenverkehr gefordert, ist im Rahmen dieser Arbeit nicht möglich.

1.2. Wissenschaftlicher Beitrag

Die vorliegende Arbeit leistet einen wissenschaftlichen Beitrag zur computergestützten Modellierung von Fahrerverhalten. Diese Verhaltensmodellierung beschreibt und prognostiziert die Interaktion des Fahrers mit den im nahen Umfeld befindlichen Verkehrsteilnehmern unter Berücksichtigung der Fahrzeugdynamik und der lokalen Verkehrsinfrastruktur. Die Arbeit trägt somit im Forschungsbereich *Intelligenter Transportsysteme und der Mensch-Technik-Interaktion* bei, Informationen über die Komponenten Fahrer, Fahrzeug, Infrastruktur und lokaler Umwelt zu fusionieren und so Hypothesen über die zukünftige Entwicklung der Verkehrssituation abzuleiten.

Die Verhaltensmodellierung fordert eine quantitative Beschreibung der individuellen Eigenschaften des Fahrers, welche durch wissenschaftliche Beiträge im Bereich der maschinellen Fahrstilerkennung und der fahrertypischen Schätzung von Komfortparametern erreicht wird.

Grundlage der Verhaltensmodellierung ist eine aussagekräftige Beobachtung des Fahrers, sowie seiner Handlungen und Reaktionen in Bezug auf die Umwelt. Die Arbeit liefert einen Beitrag zur nicht-intrusiven sensorischen Erfassung des Fahrers mit besonderem Augenmerk auf in der Anwendungsdomäne „Automobil“ verwendbaren Sensoren. Ein besonderer Fokus liegt auf der Erkennung der Blickrichtung, d.h. der maschinellen Erkennung der Kopfposition und der Pupillenstellung, welche in Automobilanwendungen besonderen Ansprüchen unterliegt.

Zusammenfassend leistet diese Arbeit die folgenden wissenschaftlichen Beiträge im Forschungsfeld *Intelligenter Transportsysteme und der Mensch-Technik-Interaktion*:

- Steigerung der Mensch-Technik-Interaktionsfähigkeiten intelligenter Automobile durch eine auf die Domäne Automobil optimierte sensorische Erfassung des Fahrers und der damit verbundenen Möglichkeit, das Blickverhalten des Fahrers in Bezug zum situativen Verkehrskontext zu analysieren.
- Verbesserung von Sicherheit und Effizienz zukünftiger Fahrerassistenzsysteme durch Modellierung von nicht-deterministischem Fahrerverhalten unter der Berücksichtigung der sich wechselseitig beeinflussenden, nur partiell beobachtbaren und mit Unsicherheiten behafteten Komponenten *Fahrer, Fahrzeug und Fahrzeugumfeld*.

1.3. Aufbau der Arbeit

Kapitel 2 gibt einen Überblick des aktuellen Stands der Forschung hinsichtlich der Erfassung und Analyse des Fahrerverhaltens im Kontext der vorherrschenden Verkehrssituation. Einen Grundstein der Verhaltensanalyse bildet die sensorische Erfassung des Fahrers (Perzeption), in der die *Blickrichtungserfassung* einen bedeutenden Stellenwert einnimmt. Arbeiten welche die Blickrichtung des Fahrers über optische

Sensoren schätzen werden deshalb vertieft vorgestellt. Einen weiteren Schwerpunkt bildet die fahrerbezogene Situationsinterpretation, welcher im gegebenen Forschungsüberblick in die vier Kategorien *Analyse des Blickverhaltens*, *Analyse des Fahrverhaltens*, *Prognose des Fahrverhaltens* und *Modellierung des Fahrverhaltens* strukturiert ist.

Die Konzeptkapitel 3 und 4 beschreiben die Beiträge, welche diese Arbeit zur besseren Einbindung des Fahrers in das intelligente Fahrzeug von morgen leistet. Kapitel 3 beschreibt die Entwicklungen zur Blickrichtungserkennung über optische Sensoren und stellt dabei speziell die Optimierungen der Beiträge an die Eigenheiten und Herausforderungen der Domäne Automobil heraus. Neben der Erkennung der Blickrichtung steht deren Zuordnung zu den Entitäten der dynamischen Umgebung im Fokus.

Kapitel 4 umfasst die Beiträge zur maschinellen Analyse des Fahrerverhaltens. In diesem Kapitel bilden die *Erkennung des Fahrstils*, die *Ableitung von Komfortparametern*, die *Analyse des Blickverhaltens* und die *Manöverprognose* die Schwerpunkte.

Eine Validierung der entstandenen Konzepte findet im Evaluationskapitel 5 statt. Die entworfene Blickrichtungserkennung wird dabei an den im Konzeptkapitel definierten Anforderungen des Automobilbereichs gemessen. Hinsichtlich der Fahrstilerkennung wird eine überwachte und eine unüberwachte Klassifikation des Fahrstils verglichen und deren Vor- und Nachteile werden bewertet. Am Anwendungsfall der fahrerindividuellen Energieprognose wird quantitativ belegt, wie über die Schätzung fahrerindividueller Komfortparameter Modelle zur Verbrauchsprognose präzisiert werden können. Des Weiteren wird am Beispiel des Spurwechselverhaltens während einer Autobahnfahrt untersucht, wie gut der aktuelle Zustand des Fahrers erkannt werden kann und wie verlässlich aus dieser Erkennung zukünftige Handlungen prognostizierbar sind.

Kapitel 6 blickt zusammenfassend auf die geleisteten Beiträge und bewertet sie im Hinblick auf den aktuellen Forschungsstand. Das Kapitel schließt mit einem Ausblick auf zukünftige Forschungsfelder, welche den Fahrer im Automobil der Zukunft in den Mittelpunkt stellen.

2. Der Fahrer im Automobil der Zukunft - Aktuelle Forschung

Ein modernes Fahrzeug ist ein *Kognitives System*, welches sich durch die Komponenten *Perzeption*, *Interpretation*, *Planung*, *Aktion* und *Wissensbasis* auszeichnet. Abbildung 2.1 zeigt die abstrakte Grundstruktur eines *Kognitiven Automobils* und beinhaltet gleichzeitig die Strukturierung des in diesem Kapitel beschriebenen Stands der Technik.

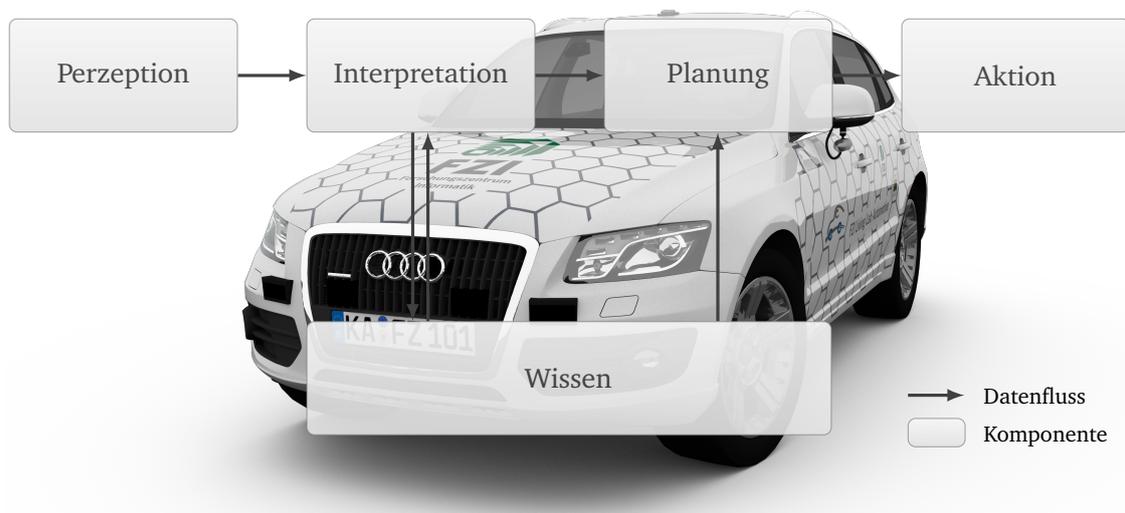


Abb. 2.1.: Schematische Darstellung des *Kognitiven Systems* „Automobil“ und Grundstruktur des in diesem Kapitel beschriebenen Stands der Technik.

Im Folgenden werden die dargestellten Komponenten im Hinblick auf den Fahrer und dessen Interaktion mit dem Fahrzeug erläutert:

Perzeption Eine Vielzahl von Arbeiten findet sich in der Literatur zur sensoriiellen Erfassung des Fahrers in seiner Arbeitsdomäne Automobil. Spezialisiert auf bildverarbeitende Systeme geben Cuong und Trivedi einen weitreichenden Überblick über die Fahrererkennung in [CT11]. Die wichtigsten Gebiete der maschinellen Erfassung bilden:

Vitalität: Eine Übersicht über die Erfassung der Vitalität mittels physiologischer Sensoren gibt Begum in [Beg13]. Sahayadhas, Sundaraj und Murugappan geben in [SSM12] einen Überblick über Methoden zur Erfassung von Müdigkeit. Sie vergleichen dabei Methoden, welche die Müdigkeit des Fahrers anhand von Fahrzeugdaten, physiologisch erfassten Daten und dem Fahrverhalten schätzen [SSM12]. Barr, Popkin und Howarth bewerten in [BPH05] auf dem Markt erhältliche Systeme zur Erfassung von Müdigkeit. In der Dissertation von Hargutt wird das Lidschlagverhalten über elektrische Miniaturspulen erfasst und über einen dreistufigen Entscheidungsbaum in vier ordinal abgestufte Müdigkeitsklassen klassifiziert [Har03].

Mirwaldt et al. vergleichen Sensoren und Algorithmen zur kontaktlosen Erfassung der Herzfrequenz des Fahrers [Mir+13]. Chamadiya et al. überwachen den Vitalitätszustand des Fahrers über im Sitz angebrachte kapazitive Sensoren [Cha+09]. Poh, McDuff und Picard messen die Herzfrequenz des Menschen (nicht spezialisiert auf den Autofahrer) über eine auf das Gesicht gerichtete CCD-Kamera [PMP10]. Healey und Picard leiten über die physiologischen Sensordaten des Elektrokardiogramms, der Elektromyographie und der elektrodermalen Aktivität den Stress des Fahrers ab [HP00; HP05].

Emotionen: Zeng et al. geben in [Zen+09] einen Überblick zur Erfassung von menschlichen Emotionen via Audio- und Bilddaten. Tawari und Trivedi leiten über eine Audioaufzeichnung im Fahrzeug Audiomerkmale ab und klassifizieren die Emotionen des Fahrers über eine Support Vector Machine (SVM) in die drei Klassen [*positiv*], [*negativ*] und [*neutral*] [TT10]. Cohen, Garg und Huang klassifizieren bildbasiert sechs verschiedene Emotionen über *Hidden Markov Modelle (HMMs)* [CGH00]. Kolli et al. kombinieren eine Wärmebildkamera mit einer CCD-Kamera, um in den Kamerabildern sechs verschiedene Emotionen des Fahrers abzuleiten [Kol+11]. Einen Überblick über die bildbasierte Erfassung von Emotionen (nicht auf das Automobil beschränkt) geben Pantic und Rothkrantz in [PR00].

Körperstellung: Cheng und Trivedi detektieren die Stellung der Hände des Fahrers über eine Wärmebildkamera [CT06; CPT07; Tri+04]. In einer weiteren Veröffentlichung klassifizieren sie bildbasiert die Stellung der Hände des Fahrers und des Beifahrers, um zu identifizieren wer mit dem Infotainmentsystem des Autos interagiert [CT10]. Tran und Trivedi extrahieren aus mehreren Kamerabildern die Stellung des Oberkörpers (Arme, Hände und Kopf). Dazu wird der *XMOB Body-Tracker* benutzt [TT09a; TT09b]. Datta, Sheikh und Kanade schätzen die Gelenkwinkelstellungen des Oberkörpers eines Autofahrers über eine Erkennung von beweglichen Ebenen in einem Kamerabild [DSK08]. Zhao et al. klassifizieren vier verschiedene Körperstellungen des Fahrers über ein *Multilayer-Perceptron (MLP)* [Zha+12b] und über *Random Forest Trees* [Zha+12a]. Die klassifizierten Körperstellungen sind [*beide Hände am Lenkrad*], [*eine Hand am Lenkrad und eine Hand an der Gangschaltung*], [*essend*] und [*telefonierend*].

Ohn-Bar et al. detektieren mit einer am Armaturenbrett angebrachten CCD-Kamera, ob sich die Hände des Fahrers am Lenkrad, am Gangwahlhebel oder am Infotainment-System befinden. Zusammen mit einer Blickrichtungserkennung (ebenfalls realisiert über eine CCD-Kamera) klassifizieren sie auch die momentane Aktion des Fahrers. Sie unterscheiden die Klassen [*Lenkrad-Aktion*], [*Infotainment-Aktion*] und [*Gangwahl-Aktion*] [Ohn+14].

Kopf- und Augenstellung: Eine besondere Rolle kommt der Erkennung der Blickrichtung über

die Schätzung von Kopf- und Augenstellung des Fahrers zu, da beim Fahren circa 90% aller Informationen visuell aufgenommen werden [BB07; BD13]. Die Blickrichtungserkennung wird deshalb in dieser Arbeit vertieft behandelt. Speziell im Automobilbereich unterliegt die Blickrichtungserkennung besonderen Ansprüchen. Auf keinen Fall darf der Fahrer in seinem Sichtfeld und seinem Arbeitsraum beeinträchtigt werden. Die Blickrichtungserkennung muss schnell und genau genug sein, um einen kurzen Blick in den Innenspiegel zu detektieren, der oftmals nur durch eine Augenbewegung ausgeführt wird. Darüber hinaus muss der Erkennungsbereich groß und die Algorithmik schnell genug sein, um bei starken und schnellen Kopfdrehungen während eines Schulterblicks, Einpark- oder Wendemanövers die Kopfposition bestimmen zu können. Erschwerend kommen ständig wechselnde Lichtverhältnisse hinzu. Diese Ansprüche, aber auch Limitierungen seitens eingesetzter Hardware oder aus wirtschaftliche Erwägungen heraus, machen die Blickrichtungserkennung weiterhin zum Bestandteil heutiger Forschung. Abschnitt 2.1 geht vertieft auf die kamerabasierte Erfassung von Kopf- und Augenstellung ein.

Interpretation Vergleichsweise wenige Arbeiten nehmen die *Situationsanalyse* und damit die holistischen Interpretation der Informationsquellen *Umwelt*, *Fahrer* und *Fahrzeug* in den Blick. Stellt man den Fahrer und dessen Verhalten in den Mittelpunkt, so fokussiert sich die Situationsanalyse auf die *Analyse des Fahrerhaltens*. Die aktuellen Forschungen zur Analyse des Fahrerhaltens wird in dieser Arbeit in drei Kategorien unterteilt:

- Arbeiten, die das *Blickverhalten* des Fahrers in Bezug zu den Ereignissen im Fahrzeugumfeld setzen (Abschnitt 2.2).
- Arbeiten, welche das *Fahrverhalten*, d.h. die Art und Weise der Bedienung des Fahrzeugs in Interaktion mit der Umwelt analysieren (Abschnitt 2.3).
- Arbeiten, die zukünftige Aktionen und Manöver des Fahrers prognostizieren. Diese werden in Abschnitt 2.4 unter dem Begriff *Verhaltensprognose* vorgestellt.

Wissensbasis Die Interpretationsgrundlage der *Situationsanalyse* bildet in einem *Kognitiven System* die *Wissensbasis*. Eine Wissensbasis kann aus Datenbanken gefüllt mit statistischem Wissen bestehen, aber auch aus komplexen Verhaltensmodellen, die das Verhalten des Fahrers modellieren. Dabei liegt der Fokus im Gegensatz zu den in den Abschnitten 2.2 bis 2.4 vorgestellten Arbeiten zur *Analyse des Fahrerhaltens* nicht auf der Interpretation, sondern auf der Verhaltensmodellierung. Oft werden im Automobilbereich (kognitive) Architekturen, welche menschliches Verhalten modellieren, wiederverwendet und auf den Fahrer bzw. die Domäne Automobil angepasst [Sal06]. Abschnitt 2.5 stellt Arbeiten vor, die Wissen über den Fahrer modellieren, bereithalten und über Inferenz neues Wissen ableiten.

Planung und Aktion Die Planung und die eigentliche Aktionsausführung sind (fahrerassistenz-) systemspezifisch und werden in dieser Arbeit nicht näher beschrieben.

Eine abschließende Bewertung des Stands der Technik mit Ermittlung des Forschungsbedarfs findet sich in Abschnitt 2.6.

2.1. Blickrichtungserkennung - Schätzung von Kopf- und Augenstellung

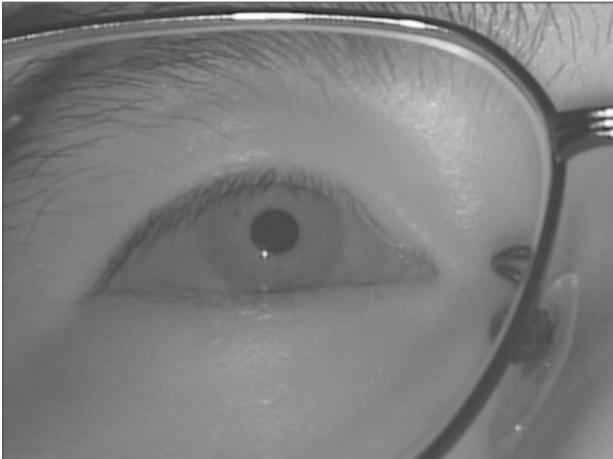
Die Blickrichtung des Fahrers kann über eine Vielzahl von Sensorsystemen ermittelt werden. Beschränkt auf nicht-intrusive Sensorsysteme, welche den Fahrer in seinem Handeln nicht stören, behindern, oder spezielle Montagen am Kopf fordern, reduziert sich die Sensorwahl auf fest im Fahrzeug angebrachte, auf den Fahrer gerichtete Kamerasysteme.

Abschnitt 2.1.1 stellt Methoden vor, die die Blickrichtung des Fahrers über Infrarotbestrahlung der Augen bei gleichzeitiger Bildaufnahme durch eine CCD-Kamera ermitteln. In Abschnitt 2.1.2 werden Blickrichtungserkennungssysteme vorgestellt, welche eine Monokularkamera nutzen. Blickrichtungserkennungssysteme basierend auf RGB-D-Kameras werden in Abschnitt 2.1.3 vertieft behandelt. Sensorsysteme, die über den Einsatz von strukturiertem Licht ein Tiefenbild generieren, werden in Industrie und Forschung schon länger eingesetzt. Dennoch gibt es vor 2011 kaum Veröffentlichungen zur Blickrichtungserkennung im Automobilbereich über RGB-D-Kameras. Die RGB-D-Technologie wird nach 2011 durch den Vertrieb von preisgünstigen Komplettsystemen wie der Microsoft® Kinect™ Kamera, welche ein RGB-Farbbild mit registriertem Tiefenbild liefert, zunehmend interessanter. Abschnitt 2.1.3 ist aus diesem Grund nicht auf Systeme zur Blickrichtungserkennung im Automobilbereich beschränkt. Eine Zusammenfassung der aktuellen Forschung im Themenbereich der Blickrichtungserkennung im Automobil gibt Abschnitt 2.1.4.

2.1.1. CCD-Kamera in Kombination mit IR-Beleuchtung

Besonders in Anwendungsszenarien mit hohen Genauigkeitsanforderungen ist die Bestrahlung des Benutzers durch eine oder mehrere Infrarot-Scheinwerfer in Kombination mit einer auf den Benutzer ausgerichteten Kamera eine verbreitete Methode [HNL06; Pér+03; OMY02]. Durch die auf der Hornhaut auftretende Reflexion des Infrarotlichts kann die Pupillenstellung unmittelbar aus dem Kamerabild bestimmt werden. Eine separate Bestimmung der Kopfposition des Fahrers ist nicht nötig. Mit einer präzise durchgeführten Kalibrierung können Systeme mit aktiver Bestrahlung des Auges die Pupillenstellung des Benutzers mit einer Genauigkeit von unter einem Grad Abweichung bestimmen.

Aufgrund der kleinen Reflektionsfläche im Kamerabild setzt diese Methode hochaufgelöste Bilder der Augenpartie voraus (siehe Abbildung 2.2a). Der Erkennungsbereich des Kopfs wird somit sehr eingeschränkt. In Automotive-Anwendungen wird deshalb in vielen Fällen auf mehrere Kameras und Scheinwerfer zurückgegriffen, was zusätzliche Kosten verursacht und erhöhte Rechenleistung beansprucht. Ein im Automobilbereich oft eingesetztes System mit aktiver Bestrahlung ist das in Abbildung 2.2b dargestellte System von *SmartEye* [AB14], welches mit vier Kameras eine Genauigkeit von unter $0,5^\circ$ Abweichung zur exakten Kopfstellung verspricht.



(a) Aufgrund der kleinen Reflektionsfläche im CCD-Kamerabild (kleiner weißer Punkt unterhalb der Pupille) setzt die Methode mit Infrarotbestrahlung hochaufgelöste Bilder der Augenpartie voraus und wird deshalb hauptsächlich in Schreibtisch- und Arbeitsplatzszenarien eingesetzt. Quelle: [OMY02]

(b) Das Blickrichtungserkennungssystem von SmartEye mit aktiver Bestrahlung und vier infrarotsensitiven Kameras. Die Genauigkeit des Blickrichtungsvektors wird beim Betrieb mit vier Kameras im Datenblatt mit unter $0,5^\circ$ Abweichung angegeben. Quelle: [AB14]

Abb. 2.2.: Besonders bei Systemen mit hohen Genauigkeitsanforderungen ist die Bestrahlung des Benutzers durch eine oder mehrere Infrarot-Scheinwerfer in Kombination mit einer auf den Benutzer ausgerichteten Kamera eine verbreitete Methode. Die Voraussetzung von hochaufgelösten Bildbereichen der Augenpartien schränkt den Erkennungsbereich von IR-belichteten Systemen.

2.1.2. CCD-Kamera

Speziell im Automobilbereich gibt es eine Vielzahl an Arbeiten, welche die Blickrichtungsschätzung über eine auf den Fahrer gerichtete Kamera realisieren [Ish+04; Pet+06; WT10]. Gründe sind preiswerte Kameras, die Möglichkeit einer kompakten Integration, aber auch die Wiederverwendbarkeit von Software-Frameworks aus der kamerabasierten Umfelderkennung.

Murphy-Chutorian und Trivedi geben 2009 einen umfassenden Überblick über den aktuellen Stand der Forschung im CCD-Kamerabereich [MT09]. Insgesamt werden in ihrer Studie 79 Arbeiten verglichen, welche die Kopfpose (und manche auch die Blickrichtung) über ein Kamerabild schätzen. Im wesentlichen werden darin acht verschiedene Methoden zur Kopfposenbestimmung unterschieden (siehe Abbildung 2.4, Seite 12).

Weit weniger Systeme zur Kopfposenbestimmung sind für Automotive-Anwendungen konzipiert. Bereits 1998 bestimmen Pappu und Beardsley von *Mitsubishi Electric Research Laboratories* die Kopfdrehung des Fahrers über *Appearance Template Methods*. Abbildung 2.4a veranschaulicht diese Methodik. Sie projizieren dafür ein frontal aufgenommenes Gesicht auf einen Ellipsoid und generieren über eine virtuelle Drehung des Ellipsoiden in 7° -Schritten weitere 186 synthetische Ansichten des Gesichts. Zur Laufzeit wird über die Gradienten des Graustufenbilds die ähnlichste Ansicht bestimmt und deren virtuelle Drehung als Kopfpose ausgegeben.

Das Verfahren ist dabei auf 187 Kopfposen beschränkt und macht implizit die Annahme, dass die Person zur Laufzeit ähnlich der eintrainierten Person ist. Weiterhin ist der Erkennungsbereich auf $\pm 35^\circ$ in Nick- und $\pm 56^\circ$ in Gierrichtung beschränkt.

Baker et al. verwenden *Active Appearance Models (AAM)* um die Kopfpose des Fahrers zu bestimmen

[Bak+04]. In ihrer Arbeit gehen sie davon aus, dass der Fahrer mehr oder weniger frontal in die Kamera blickt. Ihre Evaluation beschränkt sich auf die Echtzeitfähigkeit des Verfahrens und macht keine Angaben über die Genauigkeit oder den Erkennungsbereich des Kopfs. Die Stellung der Pupillen als Indikator für die eigentliche Blickrichtung wird nicht weiter bestimmt.

Quo et al. extrahieren aus einem 32×32 Pixel großen Ausschnitt des Gesichts einen 64-dimensionalen Merkmalsvektor über *Bilateral-Projection Matrix Principal Component Analysis (BMPCA)* und lernen Fahrerkopfpasen mit einer Diskretisierungsschrittweite von 5° ein [Quo+07]. Zur Laufzeit wird die Kopfpose über die euklidische Distanz bestimmt. Die Richtigkeit der Klassifikation wird auf einer Datenbank, welche 60 Personen beinhaltet, mit 95,5% angegeben. Das von Quo et al. verwendete Verfahren kann den Verfahren der *Detector Arrays* (siehe Abbildung 2.4b für eine schematische Darstellung des Verfahrens) zugeordnet werden. Auch Quo et al. gehen mit ihrem Verfahren davon aus, dass der Fahrer unmittelbar in die Kamera blickt. Die Stellung der Pupillen wird auch hier nicht weiter bestimmt. Die Genauigkeit des Systems ist auf die Schrittweite der Diskretisierung beschränkt.

Smith et al. ermitteln die Kopffrotation über die perspektivische Verschiebung der Augen und der Mundwinkel relativ zur Gesichtsfläche (siehe Abbildung 2.3). Dazu bestimmen sie über einen Hautfarbenfilter



Abb. 2.3.: Smith et al. realisieren ein sehr schnelles Verfahren für eine grobe Schätzung der Kopffrotation, indem sie die perspektivische Verschiebung der Augen und der Mundwinkel relativ zur Gesichtsfläche ermitteln. Als Kopfpose wird die grobe Orientierung über die Klassen *[rechts]*, *[links]*, *[geradeaus]*, *[oben]* und *[unten]* bestimmt. Quelle: [SSd00]

ein Rechteck, welches auf dem Kamerabild das Gesicht des Fahrers umschließt. Danach werden die 2-D-Positionen der Augen und der Mundwinkel bestimmt. Aus der relativen Positionierung der Augen und Mundwinkel innerhalb des Rechtecks wird die grobe Kopffrotation als *[rechts]*, *[links]*, *[geradeaus]*, *[oben]* und *[unten]* ermittelt. Einen ähnlichen Ansatz wählen Katahara und Aoki [KA01]. Sie bestimmen einzig die 2-D-Position der Nasenlöcher, welche sie mit einer Referenzposition vergleichen, um so eine grobe Schätzung der Kopffrotation zu ermitteln.

Murphy-Chutorian schätzen die Kopffrotation des Fahrers in den Dimensionen Nicken und Gieren [Mur07]. Sie lokalisieren hierfür das Gesicht des Fahrers im 2-D-Bild, skalieren dieses auf einen 32×32 Pixel großen Bildausschnitt und berechnen daraus das *Local Gradient Orientation Histogram (LGOH)* als Merkmalsvektor. Über diesen Merkmalsvektor wird jeweils eine *Support Vector Regression Machine (SVR)* für den Gier- und den Nickwinkel trainiert (siehe Abbildung 2.4c, *Nonlinear Regression Methods*). Der

durchschnittliche Fehler der Kopffrotationsschätzung wird mit $3,99^\circ$ in Nick- und $9,28^\circ$ in Gierrichtung während Tagesfahrten und $5,18^\circ$ bzw. $7,74^\circ$ während Nachtfahrten angegeben. Die Evaluation setzt voraus, dass der Kopf des Fahrers bereits erfolgreich lokalisiert wurde, was in 87% der Bilder ihrer Messdaten der Fall ist.

In [MT08] bzw. [MT10] erweitern Murphy-Chutorian die Kopffrotationsschätzung zu einer Kopfposeerkennung. Die Kopfpose wird durch einen Partikelfilter bestimmt, bei dem jeweils ein Partikel eine hypothetische Kopfpose repräsentiert. Der Partikelfilter wird dabei durch die Rotationschätzung initialisiert und iterativ korrigiert. Zur Partikelbewertung wird für jeden Partikel über ein 3-D-Modell des Gesichts eine theoretische 2-D-Ansicht des Gesichts berechnet und dessen Differenz zum Kamerabild bewertet. Da die Bestimmung der neuen Kopffrotation immer von der Bestimmung der letzten Kopffrotation abhängt, kann das Verfahren den *Trackingverfahren* (siehe Abbildung 2.4g) zugeordnet werden.

Der durchschnittliche Fehler wird mit $3,39^\circ$ in Nick- und $4,67^\circ$ in Gierrichtung angegeben. Die Standardabweichung beträgt dabei $4,71^\circ$ in Nick- und $6,89^\circ$ in Gierrichtung. Messungen, bei denen der Fehler größer 30° war, wurden von der Evaluation ausgeschlossen. Die Evaluation bezieht sich dabei auf die CVRR LISAP-14 Datenbank, welche 14 circa achtminütige Fahrten verschiedener Fahrer zu Tages- und Nachtzeiten beinhaltet [MT09; MT10].

2.1.3. RGB-D-Kamera

Neuere Arbeiten erforschen die Erkennung der Blickrichtung über RGB-D-Kameras und verwenden so zusätzlich zum RGB-Farbbild noch ein Tiefenbild zur Bestimmung von Kopfpose und Augenstellung.

Schätzung der Kopfpose über Random Forests Fanelli et al. verwenden *Random (Regression) Forests (RFs)* um die Kopfpose in einem 3-D-Tiefenbild zu bestimmen [FGG11; Fan+11]. Die Schätzung der Kopfpose wird somit auf ein *Regressionsproblem* abgebildet und über den Einsatz eines maschinellen Lernverfahrens gelöst. RFs werden in der Bildverarbeitung und im Gebiet des maschinellen Lernens oft eingesetzt, um eine sehr große Menge von Trainingsdatensätzen effizient zu klassifizieren. Sie bieten den Vorteil der Parallelisierbarkeit und zeichnen sich durch eine vergleichsweise schnelle Trainingsphase aus [Bre01].

Zum Trainieren der RFs wird ein mit der Kopfpose annotiertes Tiefenbild in kleinere Ausschnitte (*Patches*) aufgeteilt und ausgehend davon ein Merkmalsvektor generiert (siehe Abbildung 2.5). Die mit der Kopfpose annotierten Merkmalsvektoren bilden die Lerndatenbank. Da RFs eine große Anzahl von Lerndaten benötigen und deren Annotation aufwendig und teuer ist, ziehen Fanelli et al. synthetisch generierte Daten zur Lerndatenbank hinzu.

Im Detektionsschritt werden wieder zufällig Ausschnitte aus dem Tiefenbild extrahiert und über die RFs auf eine Kopfposition abgebildet. Die endgültige Kopfpose wird anschließend über ein Clusterverfahren aus den ermittelten Kopfposen bestimmt. Durch die zufällig und bewusst klein gewählten Tiefenbildausschnitte reagiert der Algorithmus robust auf Verdeckungen oder teilweise fehlende Bilddaten.

Trotzdem bleiben die Nachteile des ganzheitlichen, ansichtsbasierten Trainings bestehen: Neben hohem Annotationsaufwand muss die Messdatenbank sämtliche Gesichtsformen, Lichtverhältnisse, Kopfstellungen und -positionen beinhalten.

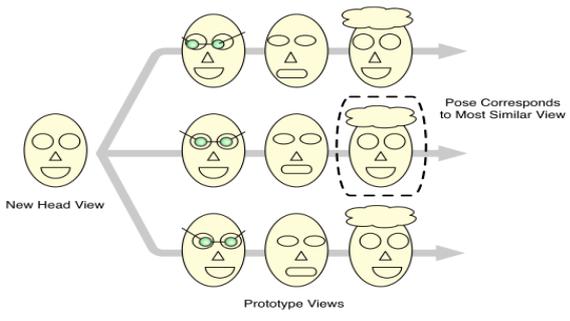
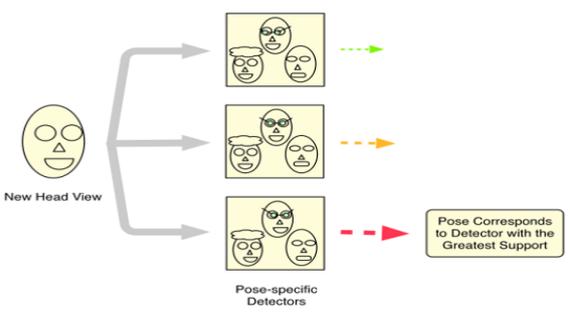
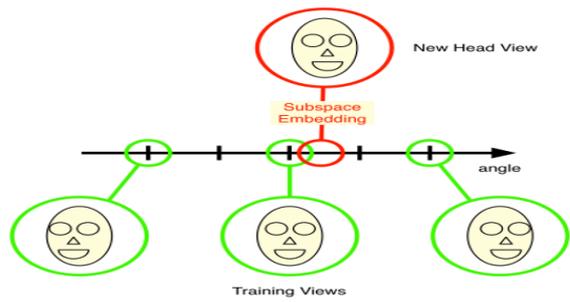
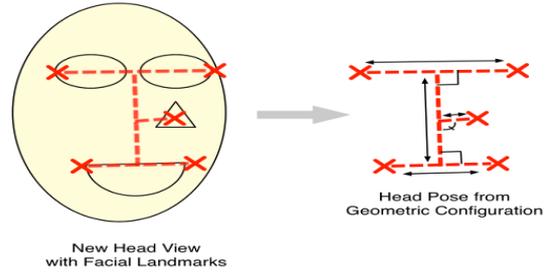
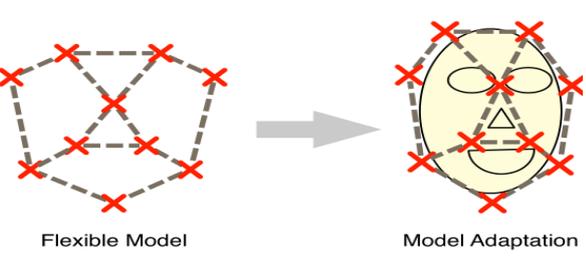
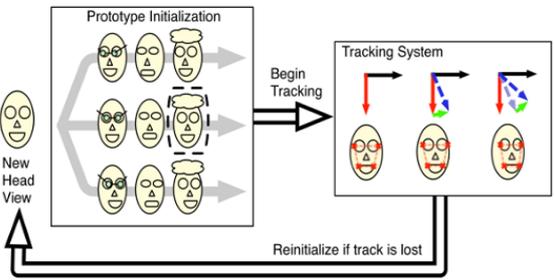
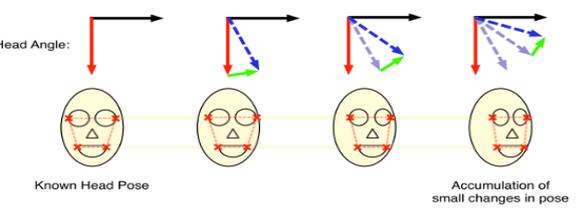
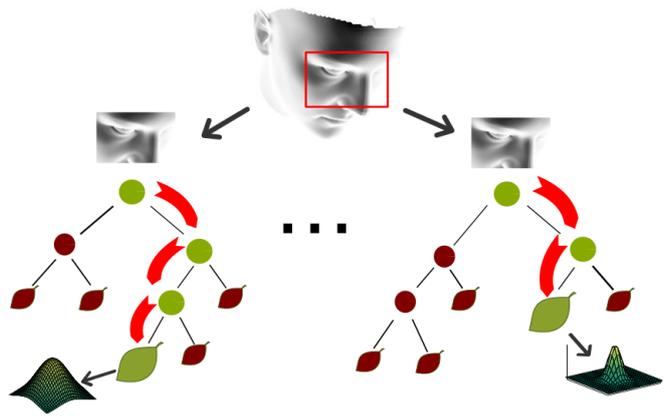
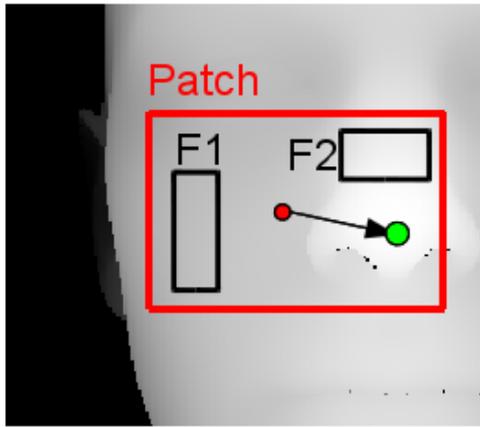
- (a) In **ansichtsbasierten Verfahren** wird aus einer Referenzdatenbank mit annotierten Kopfposen das zum Eingabebild am ähnlichsten erscheinende Bild bestimmt und dessen Kopfpose als Ergebnis ausgegeben. Die Methode der Zuordnung zum ähnlichsten Bild kann dabei variieren: Mean-Squared-Error [NF96], Normalized Cross-Correlation [Bey94], SVMs [NG02; NG99], Gabor-Wavelets [SGO01]. *Quelle: [MT09]*
- 
- (b) Beim **Detector-Array-Verfahren** werden aus einer Referenzdatenbank mit annotierten Kopfposen gleiche Kopfposen zu einer Gruppe zusammengefasst und für jede dieser Gruppen ein Klassifikator trainiert. Als Klassifikationsverfahren werden u.a. verwendet: SVMs [HSW98], FloatBoost Classifier [Zha+07], AdaBoost Cascades [JV03], Neuronal Networks [RBK98]. *Quelle: [MT09]*
- 
- (c) In **nichtlinearen Regressionsverfahren** werden aus einer Referenzdatenbank Bilder mit ihrer Kopfpose annotiert. Anhand von auf dem Bild generierten Merkmalen wird über ein Regressionsverfahren die annotierte Kopfpose gelernt. Als Regressionsverfahren werden unter anderem SVR [Li+04], Random Forests ([Dan+12]) oder Neuronale Netze [Bis95] benutzt. *Quelle: [MT09]*
- 
- (d) Der hochdimensionale Bildraum wird in **Manifold Embedding Models** auf einen sechsdimensionalen Raum der Kopfpose reduziert. Als dimensionsreduzierende Algorithmen werden Techniken wie PCA, K-PCA, Isometric feature Mapping (Isomap), Locally Linear Embedding (LLE) oder Laplacian Eigenmaps (LE) eingesetzt [MT09]. *Quelle: [MT09]*
- 
- (e) **Flexible Modelle** teilen das Gesicht in verschiedene Regionen auf und speichern das Aussehen der Region für verschiedenen Kopfposen. Die Schätzung der Kopfpose wird so zum Optimierungsproblem, indem durch Bildtransformation die Differenz der Texturen des Eingabebilds und den Modellansichten minimiert werden. *Quelle: [MT09]*
- 
- (f) In **geometrischen Modellen** werden Gesichtsmerkmale wie Nasenlöcher oder Mundwinkel im Kamerabild bestimmt und über die Zuordnung zu einem 3-D-Kopfmodell die Kopposition geschätzt. Diese Methode wird unter anderem in der Arbeit von Pfeffer et al. umgesetzt [PBZ10]. *Quelle: [MT09]*
- 
- (g) Bei **Trackingmethoden** werden die Positionsveränderung des jetzigen Eingabebilds zum Vorgängerbild ermittelt. Dies kann beispielsweise über Gesichtsmerkmale oder an Hand eines Kopfmodells realisiert sein. *Quelle: [MT09]*
- 
- (h) **Hybride Methoden** kombinieren die Ansätze aus 2.4a bis 2.4g, um die Kopfposenbestimmung in ihrer Qualität zu verbessern. Oft wird die Trackingmethode mit einer absoluten Methode kombiniert, um ein Driften der Kopfpose zu verhindern. *Quelle: [MT09]*

Abb. 2.4.: Murphy-Chutorian und Trivedi unterscheiden acht Methoden zur Schätzung der Kopfstellung in einem CCD-Kamerabild [MT09].



(a) Zum Bilden eines Merkmalsvektors wird aus dem Tiefenbild ein zufälliger Ausschnitt (120×80 Pixel) herausgegriffen und der lokale Normalenvektor für jedes Pixel des Ausschnitts berechnet. Für einen $w \times h$ großen Bildausschnitt entsteht durch Tiefenwert und Normalenvektor ein $n = 4wh$ dimensionaler Merkmalsvektor.

Quelle: [FGG11]

(b) Die inneren Entscheidungsknoten bilden Binärklassifikatoren, welche zufällig parametrisiert werden. Ein Binärklassifikator ist dabei definiert durch zwei Bildausschnitte F_1 und F_2 innerhalb des Patches, ein Schwellwert τ sowie einem Kanalselektor c . Die Binärklassifikatoren sind im Baum gemäß ihres Informationsgewinns sortiert. Die Blätter des Baums beinhalten die Kopfposition und deren Varianz als multivariate Normalverteilung.

Quelle: [FGG11]

Abb. 2.5.: Fanelli et al. erkennen die Stellung des Kopfs über *Random (Regression) Forests (RFs)*. Jeder *Regressionsbaum* schätzt die Kopfstellung in Form einer multivariaten Normalverteilung. Die endgültige Kopfpose wird anschließend mit einem Clusterverfahren aus den ermittelten Kopfposen bestimmt [FGG11].

Der wesentliche Nachteil des Ansatzes ist jedoch die für Automotive-Applikationen unzureichende Genauigkeit. Mit einer tolerierten Abweichung von 10° liegt die Richtigkeit der Schätzung bei 90,4% (95,4% bei 15° und 95,9% bei 20°).

Schätzung der Kopfpose bei gleichzeitiger Generierung eines Kopfmodells Manuel, van de Camp und Stiefelhagen veröffentlichen 2014 einen Algorithmus, welcher die Kopfpose anhand eines simultan generierten Kopfmodells schätzt [Mar13; MvS14].

Zur Erstellung des Kopfmodells findet zunächst eine Segmentierung der Punktwolke zur Trennung von Kopf und Hintergrund statt. Das Kopfmodell wird als Graph aufgebaut, welcher in den Knoten die segmentierten Punktwolken und deren Transformation zu einem initial bestimmten Koordinatensystem beinhaltet. Das initiale Koordinatensystem wird über die Detektion der Nasenspitze und die Einpassung eines Zylinders über den *RANSAC-Algorithmus* in die segmentierte Punktwolke geschätzt (siehe Abbildung 2.6a). Mit jeder neuen Punktwolke des Kamerabilds wird der Graph des Kopfmodells aktualisiert, indem über einen ICP-Algorithmus die Transformation der neuen Punktwolke zum bestehenden Modell berechnet wird. Die Kombination aus Punktwolke und Transformation wird im Graph hinterlegt.

Abbildung 2.6c zeigt die Graphstruktur des Kopfmodells. Abbildung 2.6b zeigt das Kopfmodell als Punktwolke. Knoten sind Nachbarn, wenn ihre Transformationen einen ähnlichen Rotationswinkel aufweisen. Weisen einzelne Knoten eine starke Inkonsistenz zum Gesamtmodell auf, werden sie aus dem Graphen entfernt.

Die Bestimmung der Kopfpose findet über die Erkennung von 3-D-Merkmalen auf dem Kopfmodell und der berechneten Transformation der aktuellen Punktwolke zum Kopfmodell statt.



- (a) Beispiel einer segmentierten Punktwolke des Kopfbereichs. Die grün annotierten Punkte sind Stützpunkte der Zylindereinpassung. Der pinke Punkt repräsentiert die erkannte Nasenposition. *Quelle: [Mar13]*
- (b) Punktwolkenrepräsentation des Kopfmodells. Das simultan generierte Kopfmodell setzt sich aus den Punktwolken der Einzelmessungen zusammen. *Quelle: [Mar13]*
- (c) Das Kopfmodell ist als Graph repräsentiert. Ein Knoten beinhaltet die Punktwolke und ihre Transformation zum Ursprungsknoten. Inkonsistente Knoten werden aus dem Graphen entfernt. *Quelle: [Mar13]*

Abb. 2.6.: Martin erstellt simultan zur Schätzung der Kopfpose ein Kopfmodell des Benutzers. Mit jedem neuen Eingabebild wird das Kopfmodell, repräsentiert durch einen Graph aus Punktwolken und Transformationen, iterativ verbessert. Die Kopfpose wird anhand von 3-D-Merkmalen des Kopfmodells geschätzt.

2.1.4. Zusammenfassung und Diskussion

Auf dem Gebiet der sensoruellen Erfassung des Fahrers (Perzeption) gibt es bereits zahlreiche Veröffentlichungen, welche die Blickrichtung des Fahrers schätzen. Macht man sich jedoch die hohen Ansprüche im Forschungsbereich der *Intelligenten Automobile* im Zusammenhang mit den Vor- und Nachteilen der verwendeten Kamerasystemen bewusst, wird der Bedarf weiterer Forschung deutlich:

CCD-Kamera in Kombination mit IR-Beleuchtung: Systeme, welche die Pupillenstellung über IR-Beleuchtung des Fahrers detektieren, sind sehr akkurat, decken jedoch nur einen beschränkten Erkennungsbereich ab. Durch die Reduktion der Erkennung auf die Reflektionspunkte der Pupille treffen diese Systeme implizit die Annahme, dass der Fahrer mehr oder weniger geradeaus blickt. Ist der Reflektionspunkt der Pupille nicht bestimmbar, weil der Fahrer beispielsweise gerade einen Schulterblick ausführt oder eine Sonnenbrille trägt, kann keine Schätzung über die Blickrichtung erfolgen. Eine Erweiterung des Erkennungsbereichs kann durch den Einsatz mehrerer Scheinwerfer und Kameras erreicht werden. Dies führt jedoch zu erhöhten Kosten und steigert die benötigte Rechenleistung.

CCD-Kameras: CCD-Kameras, welche preisgünstig und kompakt im Automobil verbaut werden können, decken ebenfalls nur einen eingeschränkten Erkennungsbereich ab. Sie sind je nach verwendeter Algorithmik entweder zu ungenau für Automotive-Anwendungen oder zu sehr auf einen einzelnen Benutzer spezialisiert.

Verfahren, welche die Blickrichtung anhand von Gesichtsmerkmalen bestimmen, gehen stark davon

aus, dass der Fahrer in die Richtung der Kamera blickt und dadurch die einzelnen Merkmale wieder auffindbar sind. Diese Annahme ist jedoch nicht erfüllt, wenn

- Gesichtsmerkmale verdeckt sind oder sich der Fahrer beispielsweise mit der Hand ins Gesicht greift,
- Gesichtsmerkmale durch eine starke Kopffrotation nicht im Kamerabild zu sehen sind,
- Gesichtsmerkmale sich durch Rotation des Kopfs in ihrem Aussehen verändern. Eine Nase sieht zum Beispiel in der Seitenansicht anders aus als in der Frontalansicht.

Der Erkennungsbereich von gesichtsmerkmalsbasierten Verfahren ist dadurch auf etwa $\pm 40^\circ$ in Gierrichtung beschränkt [Jim11].

Verfahren, welche die Kopfpose ansichtsbasiert lernen haben einen Erkennungsbereich von etwa $\pm 45^\circ$ in Gierrichtung [Jim11] und generalisieren nur sehr schlecht andere Benutzer. Darüber hinaus gehen ansichtsbasierte Verfahren von der nicht immer richtigen Annahme aus, dass ähnliche Bilder auf eine ähnliche Kopfpose zurückzuführen sind. Betrachtet man zwei Bilder einer gleichen Person mit leicht unterschiedlichen Kopfposen können diese Bilder sich jedoch ähnlicher sein, als zwei Bilder mit gleicher Kopfpose und verschiedenen Personen (siehe auch [Mur07]).

Großzügig geschätzt sind Blickrichtungserkennung auf Basis einer CCD-Kamera für einen Erkennungsbereich von $\pm 50^\circ$ in Gierrichtung ausgelegt. Wirtschaftliche Erwägungen und die sehr gute Integrierbarkeit der Sensorik qualifizieren CCD-Kameras jedoch weiterhin für den Einsatz im Automobilbereich.

RGB-D-Kameras: Der Einsatz von RGB-D-Kameras zur Schätzung der Blickrichtung klingt vielversprechend. Allerdings eignen sich weder [FGG11] noch [Mar13; MvS14] für den Einsatz im Automobilbereich, da beide Systeme nur die Kopfpose, nicht aber die Augenstellung schätzen. Gerade Blicke in den Spiegel zeichnen sich während der Fahrt oftmals nur durch die Veränderung der Pupillenstellung aus. Ein Blickrichtungserkennung, anhand dessen Ausgabedaten entschieden werden soll, ob der Fahrer ein umfassendes Verständnis für sein lokales Fahrzeugumfeld hat oder nicht, muss zwingend die Pupillenstellung des Fahrers berücksichtigen.

Es ist festzustellen, dass gegenwärtig der Bedarf einer echtzeitfähigen, nicht-intrusiven Schätzung der Blickrichtung unter stark wechselnden Lichtverhältnissen durch keines der veröffentlichten Blickrichtungserkennungssysteme gedeckt ist. Erschwerend kommt hinzu, dass die Blickrichtungserkennung sowohl für schnelle als auch für weiträumige Kopfdrehungen zuverlässige Ergebnisse liefern muss. Die Stellung der Augen, welche der Indikator für die eigentliche Blickrichtung des Fahrers ist, muss ebenfalls Berücksichtigung finden. Eine zusätzliche Forderung ist die Integrierbarkeit der Sensorik und deren wirtschaftliche Eignung für den Automobilbereich.

2.2. Analyse des Blickverhaltens

Beim Fahren werden circa 90% aller Informationen visuell aufgenommen [BD13; BB07]. Oft wird deshalb über die Kombination von Blick- und Umgebungsbetrachtung die Aufmerksamkeit des Fahrers geschlussfolgert. Zusätzlich rücken aufkommende Forschungsthemen wie *Kooperatives Fahren* das Blickverhalten des Fahrers in Bezug zu den Ereignissen der Umwelt besonders in den Fokus der Forschung.

Abschnitt 2.2.1 „Wohin blickt der Fahrer während der Fahrt?“ stellt zunächst die Grundlagenforschung über das Blickverhalten des Fahrers im Straßenverkehr vor. Abschnitt 2.2.2 beschreibt die Arbeiten von Fletcher et al., welche den Blick des Fahrers in Bezug zu den von der Umfeldsensorik (Stereokamera) erkannten Verkehrszeichen, Fahrspuren und Verkehrsobjekten setzen, um Gefahrensituationen zu erkennen.

In der in Abschnitt 2.2.3 vorgestellten Arbeit von Doshi et al. wird die Blickrichtung des Fahrers in Relation zu fahraufgabenspezifischen Salienzkarten gesetzt, um so die Aufmerksamkeit des Fahrers abzuleiten. Weitere Einzelarbeiten werden in Abschnitt 2.2.4 vorgestellt. Abschließend gibt Abschnitt 2.2.5 eine Zusammenfassung und Bewertung der Arbeiten.

2.2.1. Wohin blickt der Fahrer während der Fahrt?

Land and Lee untersuchen bereits 1994 die Blickbewegungen und Blickmuster des Fahrers während der Fahrt [LL94] und fassen 2006 ihre Ergebnisse (neben Erkenntnissen zum Blickverhalten in anderen Domänen) zusammen [Lan06]. Nach ihren Beobachtungen fixiert der Fahrer zu circa 80% der Zeit den Tangentialpunkt der Kurve. Dies lässt sich folgendermaßen erklären:

- Bei konstanter Kurvenkrümmung bleibt der Tangentialpunkt relativ zum Fahrer immer an der gleichen Stelle. Bewegt sich der Tangentialpunkt, so wird die Kurve enger oder weiter. Eine Veränderung des Tangentialpunkts erzwingt demnach unmittelbar eine Lenkhandlung.
- Hindernisse, welche unsichtbar hinter der Kurve liegen könnten, tauchen als erstes am Tangentialpunkt auf. Der Tangentialpunkt bietet also den weitesten Blick auf die gekrümmte Straße.

In Experimenten im Fahrsimulator untersuchen Land und Lee die Auswirkungen eines verdeckten Tangentialpunkts auf die Spurhaltefähigkeiten des Fahrers.

Zeigt man dem Fahrer nur den Tangentialpunkt der Kurve, regelt der Fahrer den Kurvenverlauf relativ gut. Das Halten des Fahrzeugs mittig in der Ego-Fahrspur gelingt jedoch nur schwer. Zeigt man dem Fahrer nur den unmittelbaren Nahbereich, wird das Fahrzeug zwar sehr gut in der Mitte der Fahrspur gehalten, der Fahrer regelt am Lenkrad aber permanent nach. Der Lenkwinkel weicht also stark vom Kurvenwinkel ab. Zeigt man dem Fahrer nur die Mitte der Straße, sind das Halten der Spur und die Antizipation des Kurvenverlaufs (gemessen an den Lenkbewegungen) nur durchschnittlich.

Die besten Lenkergebnisse werden erzielt, wenn der Fahrer Sicht auf das Nahfeld des Fahrzeugs *und* die Ferne hat. Messungen in denen der Nah- und der Fernbereich zu sehen waren konnten in ihrer Lenkqualität nicht von den Messungen unterschieden werden, in denen der Fahrer Blick auf die gesamte Straße hatte.

Interessanterweise wurde das Nahfeld des Fahrzeugs aber kaum bewusst beobachtet (foveal fixiert). *Land and Lee* kommen deshalb zum Schluss, dass das Halten der Spur über peripheres Sehen¹ ausgeführt wird. Mourant und Rockwell bemerken dazu, dass Fahranfänger dazu neigen, das nahe Fahrzeugumfeld aktiv anzusehen (foveales Sehen), da sie mit dem Spurhalten des Fahrzeugs noch weniger geübt sind [MR70].

Der Blick auf den Tangentialpunkt der Kurve reicht für ein unfallfreies und sicheres Fahren nicht aus. Schließlich muss der Fahrer Verkehrsschilder lesen und Fremdfahrzeuge wahrnehmen. Typisch ist ein wiederholtes Erkunden der Umgebung von einer halben bis zu einer Sekunde lang, je nach Kurvenkrümmung, gefahrener Geschwindigkeit und momentaner Kurvenposition.

Land misst, wie sich der Blick des Fahrers in einer Linkskurve mit einem am rechten Fahrbahnrand fahrenden Radfahrer verhält (siehe Abbildung 2.7). Ihre Aufzeichnungen zeigen, dass der Blick des Fahrers kontinuierlich zwischen dem Radfahrer und dem Tangentialpunkt alterniert. Land untersucht in

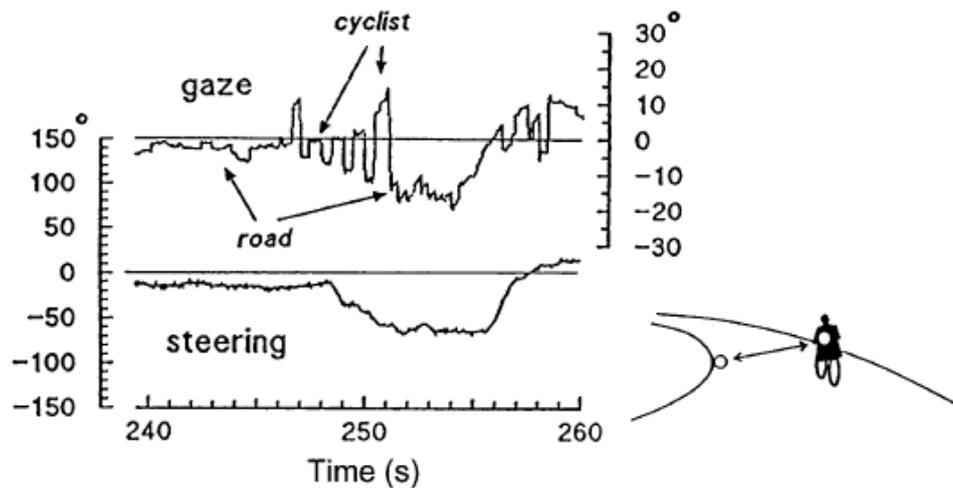


Abb. 2.7.: Befindet sich während der Kurvenfahrt ein Fremdobjekt in der Szene, wandert der Blick des Fahrers kontinuierlich zwischen dem Fremdobjekt (Radfahrer) und dem Tangentialpunkt hin und her.

Quelle: [Lan06]

einer weiteren Studie ebenfalls, wie sich beim Durchfahren einer 90 Grad Kurve das Blickverhalten von Fahranfängern, erfahrenen Fahrern und Rennfahrern unterscheidet.

Ein erfahrener Fahrer kompensiert mit seinen Kopfdrehungen die Rotation des Autos. Gleichzeitig verschiebt sich der Fokus der Augen bei der Kurveneinfahrt vom Nahbereich auf den Fernbereich. Erfahrene Fahrer antizipieren die Lenkbewegungen schon bei der Kurveneinfahrt und planen schon früh das Verlassen der Kurve. Bei Fahranfängern hingegen ist die Kopfdrehung mehr oder weniger gleich der Fahrzeugorientierung; Fahranfänger sind mit dem Lenkvorgang als solches beschäftigt. Abbildung 2.8 stellt das Blickverhalten eines Fahranfängers dem eines erfahrenen Fahrers gegenüber. Die Kopfbewegung eines Rennfahrers eilt interessanterweise ungefähr eine Sekunde dem Kurvenverlauf der Rennstrecke voraus. Land folgert, dass ein erfahrener Rennfahrer die Strecke im Kopf schon abgespeichert hat und antizipierende Kopfbewegungen ausübt.

¹ Beim peripheren Sehen wird nicht die zentrale Stelle der Netzhaut (Fovea) zum Fixieren eines Objekts benutzt. Das periphere Sehen steht im Gegensatz zum fovealen Sehen, bei dem die Blicklinie des Auges exakt auf das Objekt ausgerichtet ist. Über das periphere Sehen nimmt der Mensch sehr effizient Bewegungen auf [Wik14b].

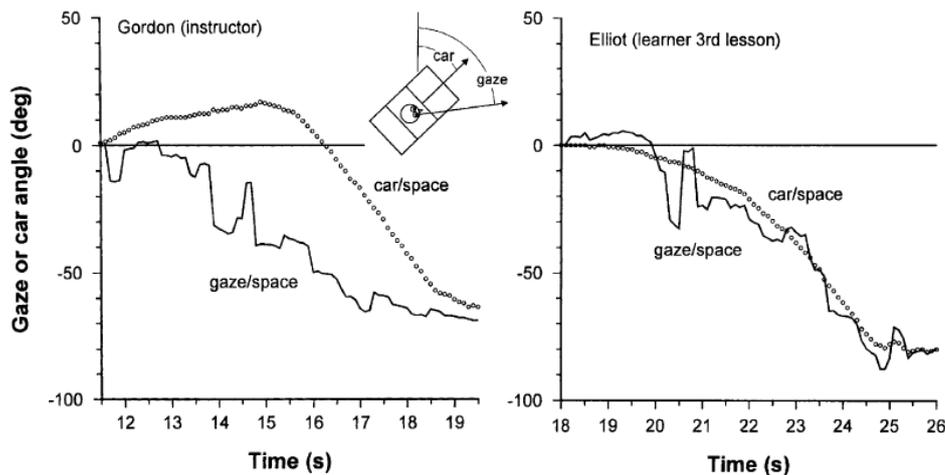


Abb. 2.8.: Vergleich des Blickverhaltens eines erfahrenen und eines unerfahrenen Fahrers. Während der erfahrene Fahrer die Kurve antizipiert und schon früh das Ausfahren aus der Kurve plant, ist der unerfahrene Fahrer mit dem Lenkvorgang ausgelastet. Quelle: [Lan06]

Vergleichbare Arbeiten, die sich in beobachtender Art und Weise mit dem Blickverhalten des Fahrers auseinandersetzen, wurden von Mourant und Rockwell in [MR70] und Shinar in [Shi08] durchgeführt.

2.2.2. Analyse der Blickrichtung zur Erkennung von Gefahren

Fletcher, Petersson, Zelinsky² et al. widmen sich der Entwicklung eines Fahrerassistenzsystems, welches sowohl den Fahrer als auch die Umgebung und das Fahrzeug berücksichtigt [FZ09; Fle+03; Fle08; Fle+05a; Fle+05b; Pet+04; PFZ05; PAZ03; Pet+06].

Aus der Kombination von Fahrspurerkennung und Blickrichtungserkennung untersuchen Fletcher et al. 2003, wie die Blickrichtung des Fahrers mit der Straßenführung korreliert [Fle+03; PAZ03]. Um eine Zuordnung des Blicks bezüglich der Straße herzustellen, werden beziehungsweise auf die erkannte Fahrspur vier (bzw. drei) laterale Regionen und vier longitudinale Regionen definiert. In lateraler Richtung sind das der linke Straßenrand, die eigene Fahrspur, der entgegenkommende Verkehr (entfällt bei einspurigen Straßen) und der rechte Fahrbahnrand. In longitudinaler Richtung wird der Nahbereich, der Fernbereich und der Horizontbereich unterschieden. Abbildung 2.10a zeigt die von Fletcher et al. definierten Sichtfelder. Die Blickrichtung des Fahrers wird über das auf einer Stereokamera basierende Blickrichtungserkennungssystem *FaceLab* ermittelt.

Bei Autobahn- und Landstraßenfahrten wurde ermittelt, wie viel Zeit der Fahrer auf die jeweilige Region blickt. Die Verteilung des Fahrerblicks auf die einzelnen Regionen zeigt Abbildung 2.10b. Fletcher et al. schlussfolgern, dass Fahrer auf einer Strecke die mehr Rechtskurven enthält, häufiger den Gegenverkehr beobachten als auf einer Strecke mit überwiegend Linkskurven. Da die Untersuchungen von Fletcher et al. im Linksverkehr (Australien) stattfanden, befindet sich der Gegenverkehr in ihren Versuchen auf der rechten Seite [PAZ03]. Der zeitliche Anteil, in dem der Fahrer die komplette Straße betrachtet (d.h. die eigene Fahrspur sowie die Gegenseite, falls vorhanden), bleibt in allen Versuchen etwa gleich. Auch der

² Dr. Zelinsky ist Mitbegründer der Firma Seeing Machines Pty Ltd, welche unter anderem das Blickrichtungserkennungssystem *FaceLab* vertreibt.

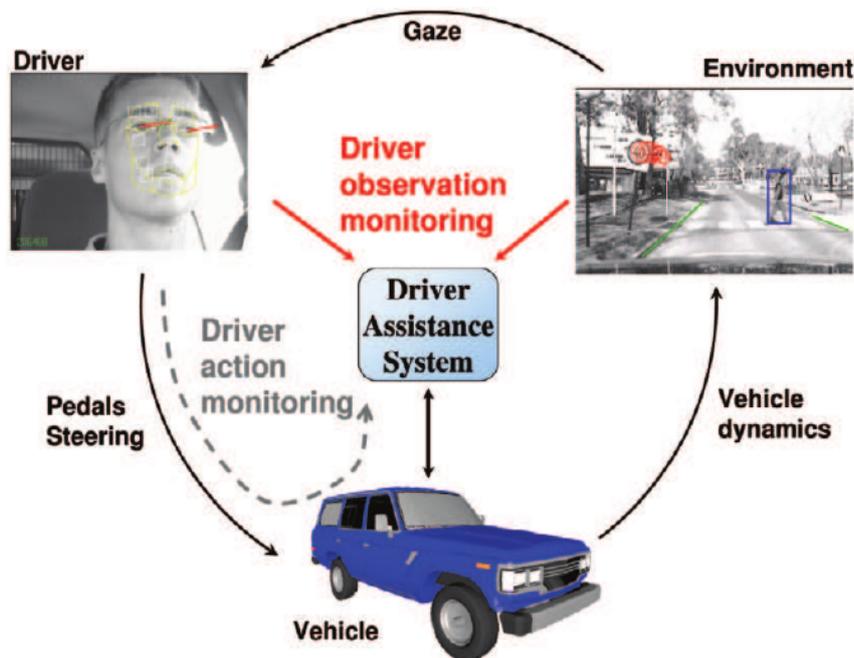
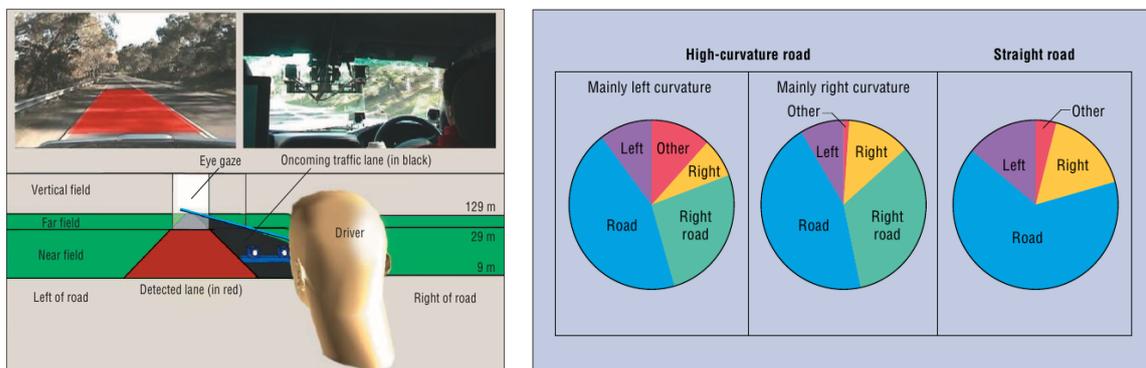


Abb. 2.9.: Die Systemarchitektur von *Fletcher, Petersson, Zelinsky et al.* berücksichtigt den Fahrer, das Fahrzeug und das Fahrzeugumfeld, um, abhängig von der Blickrichtung des Fahrers, Warnhinweise zu geben.

Quelle: [FZ09]



(a) Fletcher et al. teilen den Blick des Fahrers in vier laterale und drei longitudinale Regionen, um experimentell zu bestimmen, wohin der Fahrer während der Fahrt blickt. Quelle: [Fle+03]

(b) Zeitliche Verteilung des Blicks nach Fletcher et al. in drei verschiedenen Experimenten: Landstraße mit mehr Rechtskurven, Landstraße mit mehr Linkskurven und Autobahnfahrt [Fle+03; PAZ03]. Quelle: [Fle+03]

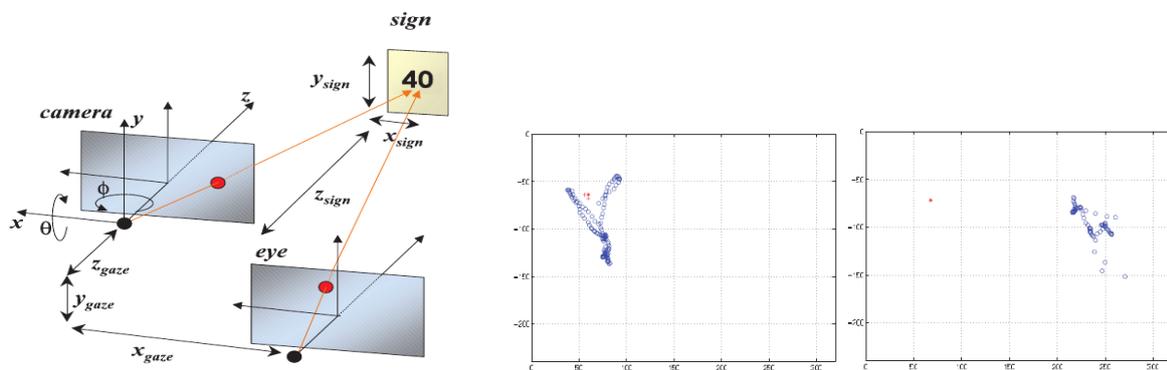
Abb. 2.10.: Aus der Information des Spurverlaufs und der Blickrichtung erstellen Fletcher et al. eine Statistik, wohin der Fahrer während der Fahrt blickt. Sie stellen fest, dass in rechtskurvigen Strecken der Gegenverkehr öfter beobachtet wird als in linkskurvigen. Sie führen ihre Experimente im Linksverkehr (Australien) durch.

Verkehrszeichen gesehen	Geschwindigkeit OK	Hinweis
ja	ja	kein Hinweis
ja	nein	Information im Display
nein	ja	Information im Display
nein	nein	Warnton

Tab. 2.1.: Hinweisgebung auf Verkehrsschilder nach [FZ09; Pet+04].

zeitliche Anteil, der auf die Fahrbahnränder entfällt bleibt gleich, wenn auch in linkskurvigen Strecken ein größerer Anteil auf den linken Fahrbahnrand und in rechtskurvigen Strecken ein größerer Anteil auf den rechten Fahrbahnrand entfällt.

In 2004 beziehen Petersson et al. in ihre Forschungsarbeiten erstmals auch Hindernisse (nicht näher spezifiziert) und Verkehrsschilder mit ein und geben aktiv Hinweise an den Fahrer. Wird festgestellt, dass ein Hindernis vom Fahrer nicht erkannt wurde, wird ein Kollisionshinweis ausgegeben. Wurde das Hindernis vom Fahrer gesehen, wird der Hinweis unterdrückt. Hinweise auf Verkehrsschilder werden dem Fahrer abhängig von seiner Geschwindigkeit, der erkannten Geschwindigkeitsbegrenzung und der Schätzung, ob das Verkehrschild als gesehen gilt, gegeben. Tabelle 2.1 zeigt die Hinweisgebung abhängig von der Fahrzeuggeschwindigkeit, Geschwindigkeitsbegrenzung und dem Erkennen des Verkehrszeichens durch den Fahrer. Ob ein Verkehrszeichen als gesehen oder nicht gesehen gilt, wird über Epipolargeometrie berechnet. Abbildung 2.11a zeigt das zu Grunde liegende Berechnungsmodell. Kennt



(a) Der Blick des Fahrers bildet eine Epipolarlinie im Kamerabild der Szenekamera. Liegt das erkannte Verkehrszeichen in einem Toleranzbereich entlang dieser Epipolarlinie, gilt es als gesehen. Quelle: [FZ09; PFZ05]
 (b) Messung eines gesehenen (links) und eines nicht gesehenen Verkehrszeichens (rechts). Quelle: [FZ09]

Abb. 2.11.: Ob ein Verkehrszeichen als gesehen bzw. nicht gesehen gilt, berechnen Petersson et al. über Epipolargeometrie [PFZ05].

man die Blickrichtung des Fahrers und dessen Kopfposition, lassen sich Epipolarlinien im Kamerabild der nach vorne gerichteten Kamera (Szenekamera) abbilden. D.h. zu der ermittelten Blickrichtung ergibt sich eine Epipolarlinie im Kamerabild der Szenekamera, auf der sich jene Objekte befinden die der Fahrer gerade fokussiert. Auf welcher Position das Verkehrszeichen auf dieser Epipolarlinie im Szenekamerabild liegt, hängt vom Abstand des Verkehrszeichens zur Szenekamera ab. Abbildung 2.11b zeigt jeweils eine Messung eines gesehenen und eines nicht gesehenen Verkehrsschildes. Liegt das Verkehrszeichen innerhalb

eines Toleranzbereichs entlang der Epipolarlinie, welcher durch eine Worst-Case-Betrachtung (Verkehrszeichen unmittelbar nahe am Fahrzeug versus Verkehrszeichen unendlich weit weg) definiert wurde, gilt das Verkehrszeichen als gesehen. *Fletcher, Petersson, Zelinsky et al.* setzen in ihren Forschungsarbeiten im Wesentlichen folgende Anwendungsfälle um:

Warnung bei Unaufmerksamkeit: Der Fahrer wird gewarnt, wenn er die Fahrspur längere Zeit nicht mehr betrachtet hat.

Warnung bei ungewolltem Verlassen der Fahrspur: Der Fahrer wird gewarnt, sobald er die Spur ungewollt verlässt. Ein gewollter Spurwechsel zeichnet sich dabei durch die Betätigung des Blinkers oder durch einen Blick in Richtung der Zielspur aus.

Warnungen bei Unaufmerksamkeiten bzgl. Verkehrereignissen: Der Fahrer wird auf übersehene Verkehrszeichen hingewiesen (siehe Tabelle 2.1). Außerdem wird der Fahrer durch einen Warnhinweis auf übersehene Hindernisse aufmerksam gemacht.

2.2.3. Fahraufgabenspezifische Aufmerksamkeitsschätzung über Salienzkarten

Doshi und Trivedi bestimmen die Aufmerksamkeit des Fahrers mit Hilfe von *Salienzkarten (Saliency Maps)*. Die vorwiegend in der Neurobiologie und Psychologie erforschten Salienzkarten geben Aufschluss darüber, wie der Blick des Menschen beim Anblick einer Szene wandert und welche Bereiche vom Auge verstärkt fixiert werden. Die Untersuchungen beziehen sich hierbei oft auf das Betrachten von Bildern oder Videos. In Abbildung 2.12 sind zwei Salienzkarten zu sehen, die beim Betrachten einer Fotografie aus dem Blickverhalten erstellt wurden.

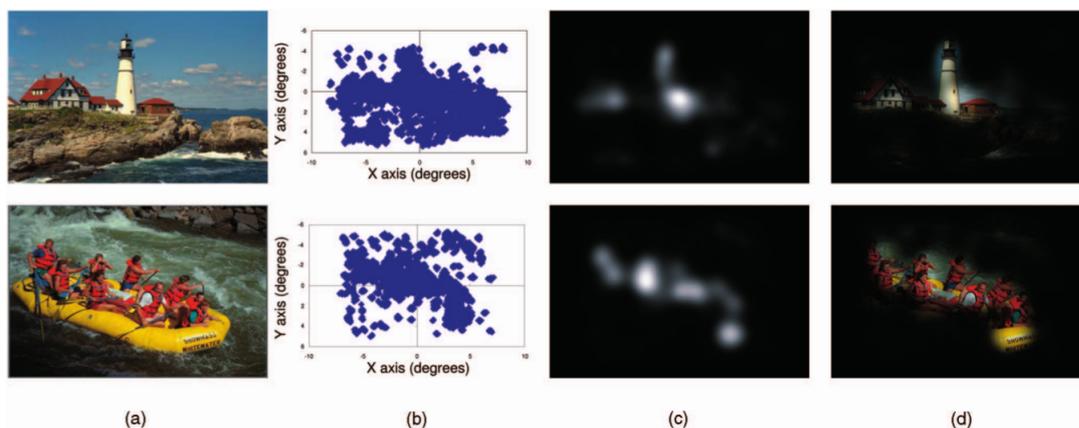
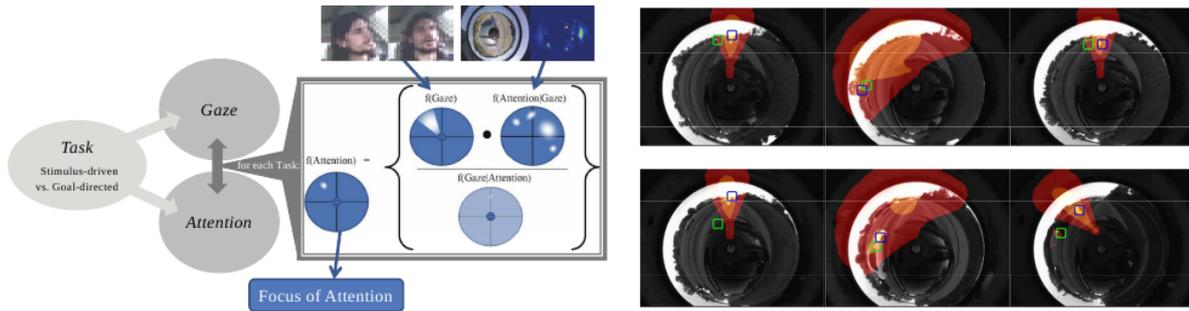


Abb. 2.12.: Abbildung (a) zeigt das dem Betrachter gezeigte Bild. In Abbildung (b) ist die Verteilung des Blicks bei 14 Sekunden Betrachtung zu sehen. Abbildung (c) zeigt die Blickhäufigkeit der einzelnen Pixel und Abbildung (d) zeigt die Blickhäufigkeit überlagert mit dem gezeigten Bild. *Quelle: [Le +06]*

In vielen wissenschaftlichen Arbeiten wird untersucht, wie für ein vorgegebenes Bild, Video oder eine vorgegebene Live-Szene algorithmisch eine Salienzkarte erstellt werden kann. Viele der Ansätze basieren



(a) Illustrative Darstellung der verwendeten Formel zur Bestimmung der Aufmerksamkeit. Die Aufmerksamkeit wird durch eine Multiplikation der erstellten Salienzkarte mit dem gemessenen Blickrichtungsvektor bestimmt. *Quelle: [DT10]*

(b) Erstellte Salienzkarten für die Fahraufgaben [Spurhalten], [Spurwechseln] und [keine Aufgabe]. *Quelle: [DT10]*

Abb. 2.13.: Zur Berechnung der Aufmerksamkeit nach Doshi und Trivedi wird eine fahraufgabenspezifisch erstellte Salienzkarte mit dem durch ein Gaußglocke modellierten Blickrichtungsvektor multipliziert.

auf der Suche nach Farb- oder Kontrastmerkmalen oder dem optischem Fluss (zum Beispiel [HPS11]). Doshi und Trivedi erstellen eine fahraufgabenspezifische Salienzkarte (siehe Abbildung 2.13b). Begründet wird dies durch die Arbeiten von Land bzw. Rothkopf, Ballard und Hayhoe aus denen hervorgeht, dass die Muster des menschlichen Blickverhaltens stets aufgabenorientiert sind [Lan92; RBH07]. Aus Komplexitätsgründen beschränken sie sich bei der Wahl der Fahraufgaben auf Autobahnsszenarien und identifizieren die Aufgaben [Spurhalten], [Spurwechseln] und [keine Aufgabe]. Für die Spurhalten- und Spurwechselaufgaben werden feste, von der Umgebung unabhängige Salienzkarten gebildet. Die Salienzkarte für das Spurhalten bildet den Blick des Fahrers durch einen geraden Sichtkegel nach vorne ab, die Karte für den Spurwechsel zeigt einen leicht nach rechts gedrehten Sichtkegel (siehe Abbildung 2.14). Einzig die Salienzkarte für die Fahraufgabe „keine Aufgabe“ zieht die Geschehnisse der Umgebung mit ein. Hierfür wird das Bild einer auf dem Dach montierten 360°-Omnikamera über optischen Fluss auf Bewegung analysiert. Bereiche mit viel Bewegung gehen mit höheren Werten in die Salienzkarte ein. Die Aufmerksamkeit des Fahrers wird durch eine Multiplikation von erstellter Salienzkarte mit dem gemessenen Blickrichtungsvektor, welcher durch einen 10°-gaußglockenartigen Sichtkegel abgebildet wird, bestimmt. Die Berechnungsformel wird illustrativ in Abbildung 2.13a gezeigt. Abbildung 2.14 zeigt das Omnikamerabild der lokalen Fahrzeugumgebung (Spalte 1), die für die Fahraufgabe spezifisch erstellte Salienzkarte multipliziert mit dem Blickrichtungskegel (Spalten 2-4) und die daraus resultierende Aufmerksamkeitsschätzung in der letzten Spalte.

2.2.4. Einzelarbeiten zur Bestimmung von Aufmerksamkeit und kognitiver Belastung

Schätzung der Aufmerksamkeit über Blickfelder Mori et al. bewerten die Aufmerksamkeit des Fahrers mit acht im Fahrzeugumfeld definierten Blickfeldern [Mor+12]. Für jedes Blickfeld wird über die *Time-To-Collision (TTC)* des nächsten Objekts ein Risikopotential berechnet. Der Blick des Fahrers wird in sieben Zustände diskretisiert und über eine 7×8 Transitionsmatrix eine Konfidenz für jedes der acht Blickfelder berechnet. Die Aufmerksamkeit für jedes Feld ergibt sich aus der Multiplikation der

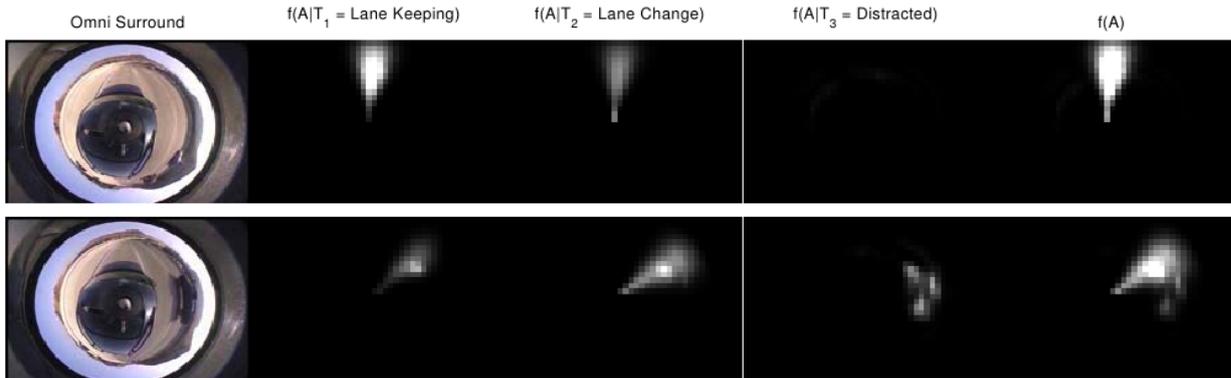
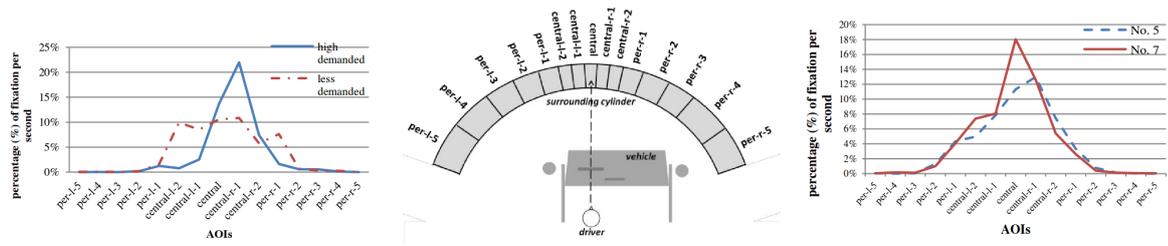


Abb. 2.14.: Die erste Spalte zeigt das Omnikamerabild der Umgebung des Fahrzeugs. Die für die Fahraufgabe spezifisch erstellte Salienzkarte multipliziert mit dem Blickrichtungskegel sind in den Spalten zwei bis vier zu sehen. Die daraus resultierende Aufmerksamkeitsschätzung zeigt sich in der letzten Spalte.
Quelle: [DT10]

Konfidenz und dem berechneten Risikopotential. Ein ähnlicher Ansatz findet sich in der Diplomarbeit von Pfeffer et al. [PBZ10].

Erkennen einer Verkleinerung des visuellen Blickfelds (Tunnelblick) Ein Effekt, der sich vor allem bei langen Fahrten im Blickverhalten des Fahrers beobachten lässt, ist die Verkleinerung des Blickbereichs (Tunnelblick). Dieser Effekt tritt verstärkt bei kognitiver Beanspruchung oder visuell sehr monotonen Fahrten ein [BD13]. Der Fahrer nimmt während des „Tunnelblicks“ nur noch Objekte in der Mitte wahr und übersieht vermehrt Objekte im Randbereich. Barby und Deml erforschen diesen Zusammenhang, indem sie die Blickrichtung des Fahrers in Bezug zu dessen gemessenem Puls setzen. Szenen, in denen der Fahrer einen hohen Pulsschlag aufweist, werden als [beanspruchend] klassifiziert. Szenen mit normalem Pulsschlag werden entsprechend als [nicht beanspruchend] klassifiziert. Abbildung 2.15 zeigt die periphere Aufteilung des Blickfelds nach Barby und Deml und die Verengung des Sichtfelds bei monotonen Fahraufgaben. In ihren Experimenten konnten Barby et al. den beschriebenen Effekt nur bedingt nachweisen. Sie begründen dies vor allem damit, dass die gewählten Situationen nicht anspruchsvoll genug waren und Kurven- oder Kreuzungssituationen ausgeschlossen werden sollten.

Ableitung einer kognitiven Belastung durch Analyse der Blickrichtung Zhang, Owechko und Zhang klassifizieren in einem Fahrsimulatorexperiment über einen Entscheidungsbaum die kognitive Belastung des Fahrers in die Klassen [High-Workload] und [Low-Workload] [ZOZ04]. Die Klassifizierung basiert auf den Merkmalen des Fahrverhaltens und des Blickverhaltens. Als Merkmale des Fahrverhaltens fließen die Fahrzeuggeschwindigkeit, die laterale Fahrzeugposition, der Lenkradwinkel und die Fahrzeugbeschleunigung in die Klassifikation mit ein. Um Merkmale für das Blickverhalten des Fahrers zu generieren, wird der Blickbereich des Fahrers in acht Felder unterteilt. Als Klassifikationsmerkmale dienen die Zeiten, in denen der Blick des Fahrers in einem Feld verweilt, sowie die Anzahl der Wechsel des Blicks in das entsprechende Feld. Außerdem fließt die Pupillengröße als Merkmal in die Klassifikation ein. Bei der Annotation der Daten wird angenommen, dass immer dann eine erhöhte kognitive Belastung besteht, wenn der Fahrer eine Zweitaufgabe ausführt. Mit der Kombination aus Blick- und Fahrverhal-



(a) Beispielszene (Landstraßenfahrt 80km/h) bei der die Verengung des Blickfelds zu beobachten war. *Quelle: [BD13]*

(b) Aufteilung des Blickfelds des Fahrers in unterschiedlich große periphere und zentrale Felder. *Quelle: [BD13]*

(c) Vergleich der Aufteilung des Blicks in einer anspruchsvollen Szene (5) und einer weniger beanspruchenden Szene (7). Der Verengungseffekt ist kaum bemerkbar. *Quelle: [BD13]*

Abb. 2.15.: Barby und Deml erforschen die Verengung des Blickfelds bei anspruchsvollen und monotonen Verkehrsszenen. Szenen, in denen der Fahrer einen hohen Pulsschlag aufweist, werden als [beanspruchend] klassifiziert. Szenen mit normalem Pulsschlag werden entsprechend als [nicht beanspruchend] klassifiziert.

tensmerkmalen können Zhang, Owechko und Zhang 81% der Daten korrekt klassifizieren. Beschränkt sich der Entscheidungsbaum auf die Fahrverhaltensmerkmale, so sinkt die Korrektklassifikationsrate auf 60%. Alleine mit Merkmalen des Blickverhaltens können 80% der Messdaten richtig klassifiziert werden.

2.2.5. Zusammenfassung und Bewertung

Fahrer und Fahrzeug sind Teil einer komplexen Verkehrsszene. Um den Fahrer zu unterstützen, muss ein FAS Objekte und Verkehrsteilnehmer wahrnehmen und ihre Rollen und Interaktionen in Relation zum Fahrer beurteilen. Dazu gehört die Erkennung von Objekten, die Einordnung ihrer Relevanz in Bezug auf die Fahraufgabe und die Feststellung der objektspezifischen Aufmerksamkeit des Fahrers. Dieses Wissen ermöglicht im Rahmen einer Situationsinterpretation komplexe Hilfestellungen.

Arbeiten, welche die Blickrichtung des Fahrers in Bezug zum lokalen Fahrzeugumfeld setzen, gibt es nur wenige.

Fletcher et al. warnen den Fahrer bei übersehenen Verkehrszeichen oder übersehenen Hindernissen. Auch wenn der Fahrer länger nicht mehr auf die Straße schaut, erfolgt eine Warnung. Da für ein Objekt über Epipolargeometrie bestimmt wird, ob es gesehen oder übersehen wurde, ist ihr Ansatz ohne weitere Modifikation auf Kamerasysteme beschränkt. Auch eine Differenzierung zwischen relevanten und nicht relevanten Objekten findet bei Fletcher et al. nicht statt. Gerade in unübersichtlichen Verkehrssituationen mit vielen Verkehrsteilnehmern ist es dem Fahrer nahezu unmöglich, alle Verkehrsobjekte wahrzunehmen. Eine Beurteilung, welche Objekte der Fahrer in Bezug auf den Situationskontext gesehen haben müsste, wäre für ein Assistenzsystem, welches eine Fahrzeugübergabe realisiert, zwingend notwendig, um die Situation einzuschätzen.

Auch Veränderungen von Objekten werden nicht weiter verfolgt. Ändert ein einmal gesehenes Fahrzeug unbemerkt vom Fahrer den Kurs, kann es unter Umständen nicht mehr als gesehen gelten.

Doshi et al. berücksichtigen den Situationskontext in Form von der momentan identifizierten Fahraufgabe des Fahrers. Für jede Fahraufgabe ist hinterlegt, welches Blickverhalten vom Fahrer erwartet wird. Entspricht das beobachtete Blickmuster dem erwarteten Blickmuster, gilt der Fahrer als „aufmerksam“. Dies

setzt voraus, dass es endlich viele Fahraufgaben gibt und dass ein geeigneter Algorithmus die momentan vorliegende Fahraufgabe korrekt bestimmen kann. Die Aufmerksamkeit wird in den Arbeiten von Doshi et al., Barby et al. und Zhang et al. als globale Größe ermittelt. Der Fahrer ist entweder [*aufmerksam*] oder [*unaufmerksam*]. Eine Differenzierung auf Objektebene findet nicht weiter statt.

Geometrische Ansätze, welche wie Mori et al. die Aufmerksamkeit des Fahrers über vorgegebene Blickfelder bestimmen, verlieren über die geometrische Diskretisierung die Aussagekraft gegenüber Einzelobjekten.

Will ein FAS dem Fahrer dediziert Objekte augmentieren, die relevant, aber ungesehen sind, ist eine objektspezifische Aufmerksamkeitsbestimmung zwingend nötig. Erst die Interpretation des Blickverhaltens hinsichtlich der Geschehnisse innerhalb und außerhalb des Fahrzeugs erlaubt eine situative Differenzierung zwischen einerseits normalem und andererseits unüblichem Blickverhalten. Dieses Wissen ermöglicht im Rahmen einer Situationsinterpretation komplexe Hilfestellungen, wie etwa eine selektive Warnung vor ungesehenen Verkehrsteilnehmern in Rechts-Vor-Links-Situationen.

2.3. Die Analyse des Fahrverhaltens

Unter dem Begriff *Fahrverhalten* wird in dieser Dissertation die Art und Weise verstanden, wie der Fahrer das Fahrzeug in Interaktion mit seiner Umwelt bedient. Eine Definition des Begriffs *Fahrverhalten* gibt die Definition 4.2 auf Seite 81. In der Literatur veröffentlichten Arbeiten, die das Fahrverhalten des Fahrers maschinell bewerten, können dabei in zwei Kategorien aufgeteilt werden: Zum einen in Arbeiten, die sich mit der maschinellen Klassifikation des Fahrstils (oft gleichgesetzt mit der Risikobereitschaft) des Fahrers beschäftigen und zum anderen in Arbeiten, welche fahrerindividuelle Komfortparameter wie den üblicherweise gewählten Bremsdruck oder die übliche Zentrifugalkraft schätzen. Abschnitt 2.3.1 beschreibt Arbeiten, welche die Fahrstilerkennung behandeln. Abschnitt 2.3.2 stellt Arbeiten vor, welche sich mit der maschinellen Schätzung von fahrerindividuellen Kennwerten (in dieser Arbeit als *Komfortparameter* bezeichnet) beschäftigen. Abschnitt 2.3.3 fasst die Arbeiten im Themenbereich der Analyse des Fahrerhaltens zusammen und bewertet den Stand der Forschung.

2.3.1. Fahrstil und Risikobereitschaft

Die Erkennung des Fahrstils ist in der Literatur üblicherweise mit der Erkennung der Risikobereitschaft des Fahrers gleichgesetzt, erkennbar, dass sich die Fahrstilklassen, in welche der Fahrer maschinell eingeordnet wird, üblicherweise über die Risikobereitschaft des Fahrers definieren.

Higgs und Abbas ermitteln den Fahrstil des Fahrers über ein zweistufiges Verfahren bestehend aus Segmentierung und Clusteranalyse [HA13]. Sie spezialisieren sich dabei auf das Verhalten bei einer Folgefahrt. Ihre Klassifikation unterscheidet die Klassen [*Low-Risk*], [*Medium-Risk*] und [*High-Risk*]. Zur Erkennung des Fahrstils wird ein achtdimensionaler Merkmalsvektor gebildet, welcher über *K-Means* dem ähnlichsten Cluster zugeordnet wird. Der Merkmalsvektor besteht aus *Inertialsystem (IMU)*-Werten und der lateralen Fahrzeugposition auf der Ego-Fahrspur. Die Erkennungsrate der Fahrstilklassifikation wurde nicht evaluiert.

Sundbom, Falcone und Sjoberg ordnen den Fahrstil des Fahrers in die zwei Klassen [*aggressiv*] und [*normal*] [SFS13]. Um den Fahrstil zu klassifizieren, parametrisieren Sundbom et al. zwei Lenkmodelle (aggressiv und normal) in Form von zwei Regressionsvektoren. Für jede Veränderung des Lenkwinkels wird die Wahrscheinlichkeit der Lenkmodellzugehörigkeit geschätzt. Die *Maximum-Likelihood-Annahme* klassifiziert den Fahrstil abschließend. Sundbom, Falcone und Sjoberg geben die Erkennungsrate ihrer Klassifikation auf kurvigen Fahrtabschnitten mit 90,3% für die Klasse [*aggressiv*] und 88,6% für die Klasse [*normal*] an. Auf der gesamten Fahrt, welche hauptsächlich aus geraden Abschnitten besteht und deshalb wenig Lenkbewegungen beinhaltet, wird die Klasse [*aggressiv*] mit einer Erkennungsrate von 50,8% und die Klasse [*normal*] mit einer Erkennungsrate von 91,7% erkannt.

Constantinescu, Marinoiu und Vladoiu extrahieren aus GPS-Daten einer Feldstudie mit 23 Fahrern Geschwindigkeits- und Beschleunigungsinformationen. Sie identifizieren über ein unüberwachtes Clusterverfahren sechs verschiedene Fahrstile aufsteigend nach Aggressivitätslevel [CMV10].

Neuere Arbeiten gebrauchen auch das Smartphone als Sensorplattform für eine Fahrstilbewertung. Zwar ist die Güte und der Umfang der Sensordaten eingeschränkter als würden Daten von äquivalenten Fahrzeugsensoren genutzt, dafür steht mit dem Smartphone jedoch ein vielseitiger und vom Fahrzeug unabhängiger Sensorträger zur Verfügung. Bereits auf dem Markt erhältliche Systeme zur Fahrstilklassifikation über das Smartphone sind *AVIVIA RateMyDrive* [Avi14], *State Farm[®] Driver Feedback* [Sta15] oder *GreenRoad* [Gre15] (letzteres fokussiert das Flottenmanagement).

Johnson und Trivedi erkennen mit einem Smartphone drei Fahrmanöver und unterscheiden darin bei zwei der Manöver die Fahrstilklassen [*aggressiv*] und [*nicht aggressiv*] [JT11]. Für ihre Manövererkennung klassifizieren sie die Sensordaten des Smartphone-Beschleunigungssensors und Gyroskops über *Dynamic Time Warping (DTW)*. Von 201 Fahr Szenen, von denen 50 als aggressiv eingestuft wurden, konnten die aggressiven Fahr Szenen zu 97% richtig erkannt werden. Castignani, Frank und Engel nutzen ebenfalls ein Smartphone, um das Fahrerverhalten einzuschätzen [CFE13a; CFE13b]. Sie bewerten dazu über ein Fuzzy-Inferenzsystem den Fahrer auf einer Aggressivitätsskala von 0 bis 100, aufgeteilt in die drei Fahrstilklassen [*normal*], [*moderat*] und [*aggressiv*]. Als Eingabe für das Fuzzy-Inferenzsystem werden insgesamt zwölf Eingabegrößen extrahiert. Dazu gehört die Anzahl an Geschwindigkeitsüberschreitungen, die Durchschnittsgeschwindigkeit der Überschreitung, die maximale Überschreitung, die Anzahl von moderaten und aggressiven Beschleunigungen und Lenkbewegungen pro Kilometer. Eine Beurteilung der Richtigkeit der Fahrstilklassifikation fand nicht statt.

2.3.2. Komfortparameter

Das individuelle Verhalten des Fahrers zeichnet sich durch die Wahl von physikalischen Kennwerten und Intervallen aus, in denen sich der Fahrer sicher und wohl fühlt. Diese Kennwerte des Fahrers werden in dieser Arbeit als *Komfortparameter* bezeichnet. Typische Komfortparameter sind der übliche Sicherheitsabstand zum vorausfahrenden Fahrzeug oder die üblicherweise gewählten (Kurven-) Geschwindigkeiten, Beschleunigungen oder Bremsverzögerungen.

Schätzung der Reaktionszeit des Fahrers Zhang und Bham bestimmen die Reaktionszeit des Fahrers aus dem Abstand zum vorausfahrenden Fahrzeug sowie den Geschwindigkeiten und Beschleunigungen des eigenen und des vorausfahrenden Fahrzeugs [ZB07]. Die Reaktionszeit bestimmen sie als Zeitspanne zwischen der Überschreitung eines Beschleunigungsschwellwerts des vorausfahrenden Fahrzeugs (Aktion) bis zur Überschreitung eines Beschleunigungsschwellwerts für das Ego-Fahrzeug (Reaktion). Voraussetzung für die Messung ist, dass beide Fahrzeuge vorher annähernd die gleichen Geschwindigkeiten haben, keine Beschleunigungen aufweisen und in definierten Abstandsgrenzen liegen.

Rakhshan, Pishro-Nik und Ray gehen davon aus, dass der Fahrer beim Umschalten einer Ampel von grün nach gelb eine Reaktion in Form einer Beschleunigung oder eines Abbremsens zeigt. Sie messen deshalb die Zeitspanne zwischen Umschalten der Ampel und Reaktion des Fahrers und definieren mit dieser Zeitspanne die Reaktionszeit des Fahrers [RPR14]. Chang et al. bemerken dazu, dass der Fahrer umso schneller auf das Umschalten der Ampel reagiert, je geringer der Abstand zur Ampel ist. Ist die gefahrene Geschwindigkeit hoch, lässt sich eine verzögerte Reaktion auf das Umschalten der Ampel beobachten [CMS85].

Sicherheitsabstand beim Folgeverhalten Rosenfeld und Bareket schätzen über persönliche Informationen des Fahrers den für den Fahrer komfortablen Sicherheitsabstand [RB12]. Dazu werden in einer Datenbank persönliche Informationen des Fahrers wie Alter, Bildungsstand, Geschlecht, Einkommen und Fahrertyp mit dem vom Fahrer über das *Adaptive Cruise Control (ACC) System* gewählten Sicherheitsabstand (sechs Stufen) annotiert. Über einen auf dieser Datenbank trainierten *C4.5-Entscheidungsbaum* lässt sich die gewählte Stufe in 70% der Fälle vorhersagen.

Schätzung der üblichen Geschwindigkeitswahl Um gefährliche und ungefährliche Verkehrssituationen maschinell zu unterscheiden, bestimmen Trautmann et al. die vom Fahrer üblicherweise gewählten Geschwindigkeiten, Beschleunigungen und Fliehkräfte [Tra+10]. Mit der Annahme, dass im Straßenverkehr circa 3% aller Verkehrssituationen gefährlich sind, bestimmen sie die individuellen Schwellwerte durch das 97%-Quantil des jeweiligen Messsignals.

Bauer, Gonter und Rojas bestimmen die übliche Geschwindigkeits- oder Abstandswahl des Fahrers, um Fahrmanöver besser erkennen zu können und normales Verhalten von ungewöhnlichem Verhalten zu unterscheiden [BGR10]. Die einzelnen Fahrmanöver werden durch *Fuzzylogik* erkannt, wobei die Fuzzymengen für die Eingangsgrößen auf die übliche Geschwindigkeits- oder Abstandswahl des Fahrers angepasst werden. Dazu werden für eine „Normalfahrt“ die Geschwindigkeiten gespeichert und nach den *Coverage-* und *Semantic-Soundness-Kriterien* aus [PG07] Fuzzymengen gebildet. Auf einer Datenbank von sechs Probanden schneidet ein individuell parametrisiertes Erkennungssystem für Stoppmanöver für jeden Fahrer in Bezug auf die Erkennungs- und Fehlerrate besser ab, als ein unparametrisiertes System. Durch das Schätzen der üblicherweise gewählten Geschwindigkeit kann die Erkennungsrate bei manchen Fahrern um bis zu 10% gesteigert werden.

2.3.3. Zusammenfassung und Bewertung

Arbeiten zur Analyse des Fahrstils klassifizieren den Fahrer nach seiner Risikobereitschaft. Der Situationskontext ist in den Arbeiten dabei streng vorgegeben oder bleibt unberücksichtigt. Der Fahrstil wird oft an den gefahrenen Beschleunigungen und Geschwindigkeiten festgemacht. Lässt man den Situationskontext außer Acht, kann ein stark beschleunigender Fahranfänger, der sein Auto nicht richtig einschätzen kann, nicht von einem sportlichen Fahrer, der stark beschleunigt, um die volle Fahrzeugdynamik auszunutzen, differenziert werden. Da die menschliche Einschätzung des Fahrstils subjektiv ist, stellt sich generell die Frage, wie die Bestimmung des Fahrstils evaluiert werden kann.

Forschungsbedarf besteht in der situativen Einschätzung des Fahrstils, welche den Kontext der aktuellen Verkehrssituation berücksichtigt und das Fahrverhalten in Bezug auf diesen bewertet. Des Weiteren wird der Fahrstil in allen Arbeiten als Maß für die Risikobereitschaft des Fahrers verstanden. Aspekte wie die Klassifizierung des Fahrstils in Bezug auf die Energie wurden bisher nicht behandelt.

Arbeiten, welche Komfortparameter des Fahrers quantifizieren, sind generell auf die Ziel-Applikation ausgerichtet und auf ein eingeschränktes Szenario fokussiert. Zhang et al. und Rakhshan et al. beschränken sich bei ihrer Reaktionszeitbestimmung auf die Autobahn bzw. auf Anfahrten an eine Ampel. Rosenfeld et al. schätzen die Komfortparameter des Fahrers ausschließlich auf statistischer Basis und ziehen keinerlei Merkmale des Fahrerverhaltens hinzu. Die Komfortparameterschätzung kann sich demnach nur am Durchschnittsfahrer orientieren. Allgemeiner gehalten sind die Komfortparameterschätzungen von Bauer et al. und Trautmann et al., wobei auch bei Trautmann et al. keinerlei Information über den Verkehrskontext zur Schätzung hinzugezogen wird.

Generell besteht Forschungsbedarf hinsichtlich einer quantitativen Beschreibung des Fahrerverhaltens und Untersuchungen, wie diese das Systemverhalten des Fahrzeugs effizienter gestalten können.

2.4. Prognose des Fahrverhaltens

Arbeiten zur Prognose des Fahrerverhaltens unterscheiden sich in ihrem Design in folgenden Gesichtspunkten:

Aktions-/Manövrierprognose vs. Intentionserkennung Tabelle 2.2 verdeutlicht den Unterschied zwischen Intentionserkennung und Aktions- bzw. Manövrierprognose. Eine Intention besteht, wenn sich der Fahrer *entschließt*, eine Aktion durchzuführen. Daraus folgt nicht zwangsläufig die *Ausführung* der Aktion. Beispielsweise kann der Fahrer die Absicht zeigen einen Spurwechsel auszuführen (Intention), ihn aber nach einem Blick in den Seitenspiegel abbrechen (keine Aktion) (siehe Fall 2 in Tabelle 2.2). Will man den Fahrer in seinen zukünftigen Handlungen unterstützen (Komfort) oder die Durchführbarkeit der Aktion prüfen (Sicherheit), so steht die Erkennung der Intention im Vordergrund. Will man den Fahrer warnen oder in ein unabsichtliches Handeln eingreifen, so steht die Erkennung der Aktion im Vordergrund.

Strategische, Manöver- oder Kontrollebene Die Handlungen des Fahrers (bzw. des Menschen) werden nach ihrer Komplexität üblicherweise in drei Hierarchieebenen unterteilt. Dabei hat sich die Hierarchie nach Michon, der eine Kontroll-, eine Manöver- und eine strategische Ebene unterschei-

det, in der Literatur als Unterscheidung für die Komplexität einer Handlung durchgesetzt [Mic85]. Auf der untersten Ebene, der Kontrollebene (oft auch operative Ebene), führt der Fahrer Regelungshandlungen wie das Abstandhalten zum Vorausfahrenden oder das Spurhalten des Fahrzeugs aus. Üblicherweise führt der Fahrer diese Handlungen unterbewusst aus. Die Dauer von Aktionen auf der Kontrollebene liegen im Millisekundenbereich.

In die mittlere Ebene, die Manöverebene (oft auch taktische Ebene), werden die Handlungen des Fahrers kategorisiert, welche der Fahrer bewusst ausführt und plant. Typische Vertreter dieser Hierarchieebene sind ein Spurwechsel oder ein Rechtsabbiegen. Die Handlungsdauer auf der Manöverebene liegt im Sekundenbereich.

Auf der strategischen Ebene, der obersten Ebene, stehen Entscheidungen wie die Routenplanung. Hier kann der Fahrer durch zum Beispiel durch Navigationsgeräte unterstützt werden.

Generische vs. spezifische Manövererkennung Beschränkt sich die Prognose des Fahrerverhaltens auf eine Aktion bzw. ein Manöver, so spricht man von einer spezifischen Prognosemethode. Ist die Prognose des Fahrverhaltens nicht auf eine Aktion bzw. ein Manöver beschränkt, spricht man von generischer Prognose.

		Tatsächliche Aktionsausführung	
		ja	nein
Intention	ja	Fall 1: Fahrer hat die Intention eine Aktion durchzuführen und führt diese auch aus.	Fall 2: Fahrer hat zwar den Willen etwas zu machen, verwirft die Aktion jedoch.
	nein	Fall 3: Fahrer führt unabsichtlich eine Aktion aus.	Fall 4: Fahrer plant keine Aktion und es findet auch keine Aktion statt.

Tab. 2.2.: Intention vs. Aktionserkennung.

Einen Schwerpunkt dieser Arbeit bildet die Analyse des Fahrverhaltens, dessen Aktionen der taktischen und der Kontrollebene zugeordnet werden. Dementsprechend werden im Folgenden Arbeiten vorgestellt, welche Aktionen auf diesen Ebenen präzisieren. Die in den Abschnitten 2.4.1 bis 2.4.4 beschriebenen Forschungsarbeiten sind nach ihrer Zielsetzung strukturiert. Abschnitt 2.4.1 beschäftigt sich mit Arbeiten zur Erkennung von Stopp- und Bremsmanövern, Abschnitt 2.4.2 mit Arbeiten, die die Abbiegerichtung an einer Abzweigung vorhersagen und Abschnitt 2.4.3 mit Arbeiten, die einen Spurwechsel prognostizieren. Abschnitt 2.4.4 führt Arbeiten auf, deren Prognoseverfahren generisch und nicht auf eine bestimmte Zielsetzung beschränkt sind. In Abschnitt 2.4.5 werden zusammenfassend die Forschungsergebnisse diskutiert.

2.4.1. Prognose von Stopp- und Bremsmanövern

Kumagai et al. präzisieren die Wahrscheinlichkeit eines Stoppmanövers in naher Zukunft [Kum+03]. Sie vergleichen hierzu zwei Ansätze: Die Prognose über *Hidden Markov Modelle (HMMs)* und die Prognose über ein *Switching Linear Dynamic System (SLDS)*. Als Datengrundlage fließt die Fahrzeuggeschwindigkeit, die Bremspedalstellung und die Gaspedalstellung in die Prädiktion ein. Ein Wahrscheinlichkeitsanstieg über 50% lässt sich bei beiden Verfahren circa eine Sekunde vor dem absoluten Stillstand des Fahrzeugs

beobachten. In ihrem Testszenario bremst ein Fahrzeug von 20 km/h zum Stillstand ab. Ein Überschreiten der 50%-Wahrscheinlichkeit wäre damit beim Unterschreiten einer Fahrzeuggeschwindigkeit von circa 3 km/h zu erkennen. McCall und Trivedi bestimmen über die CAN-Daten des Fahrzeugs, einer auf das Gesicht des Fahrers gerichteten Kamera und einer auf die Füße des Fahrers gerichteten Nahinfrarot-Kamera die Wahrscheinlichkeit eines zukünftigen Abbremsmanövers [MT07]. Zur Berechnung dieser Wahrscheinlichkeit wird jeweils eine *Relevance Vector Machine (RVM)* für einen Vorhersagehorizont von einer, zwei und drei Sekunden trainiert. Die Trainingsdaten sind automatisch über den Bremsdruck (CAN-Bus) in [Bremsvorgang] und [Normalfahrt] unterteilt. Unterscheidungskriterium der Trainingsdaten ist somit die Aktion (nicht die Intention). Folglich werden Manöverabbrüche (siehe Tabelle 2.2, Fall 3) falsch gelernt. McCall und Trivedi nehmen diesen Fehler in Kauf und merken an, dass das Annotieren der eigentlichen Intention mit erheblichem Arbeitsaufwand verbunden wäre. Ein auf einen Vorhersagehorizont von drei Sekunden trainierter Klassifikator kann ein Bremsmanöver in etwa 75% der Fälle mit einer Fehlerquote von 20% vorhersagen. Verringert man den Vorhersagehorizont auf zwei bzw. eine Sekunde, verbessert sich die Erkennungsrate bei gleichbleibendem Fehler zu 80% bzw. 90%.

Parallel zur Erkennung der Bremsintention bestimmen McCall und Trivedi über Laserscanner und die Fahrzeugdaten auch die Wahrscheinlichkeit, dass die momentane Verkehrssituation eines Bremsvorgangs bedarf. Aus der Prognose des zukünftigen Bremsverhaltens und der Situationseinschätzung über die Umfeldsensorik wird eine Dringlichkeit ermittelt, den Fahrer zu warnen. Abbildung 2.16 zeigt, wie die Kombination von Fahrer- und Situationsanalyse zur Bestimmung der Notwendigkeit einer Warnung führt.



(a) McCall und Trivedi leiten die Notwendigkeit einer Warnung über das Betrachten des lokalen Fahrzeugumfelds und der Bestimmung einer Wahrscheinlichkeit einer zukünftigen Bremsaktion des Fahrers ab. Die Absichtserkennung basiert auf Fahrzeugdaten sowie der Beobachtung des Fußraums und des Gesichts.
 Quelle: [MT07]

(b) Nach Auswertung der Laserscanner- und Fahrzeugdaten bedarf es in der links dargestellten Verkehrssituation eines Bremsmanövers. Das Verhalten des Fahrers weist jedoch keine Anzeichen eines Bremsmanövers auf. Folglich besteht eine Dringlichkeit, den Fahrer zu warnen.
 In der rechts dargestellten Situation wird die Intention des Fahrers zu bremsen erkannt. Eine Warnung ist hier nicht nötig.
 Quelle: [MT07]

Abb. 2.16.: McCall und Trivedi bestimmen über eine *Relevance Vector Machine (RVM)* die Wahrscheinlichkeit eines zukünftigen Abbremsmanövers und die Wahrscheinlichkeit, dass die momentane Verkehrssituation einer Bremsung bedarf. Die beiden Wahrscheinlichkeiten werden zu einem Notwendigkeitsmaß für eine Warnung kombiniert.

2.4.2. Prognose der Abbiegerichtung an einer Abzweigung

Eichhorn et al. sagen über eine Kostenfunktion die Abbiegerichtung an einer Abzweigung nach rechts voraus [Eic+13]. Für eine Abzweigung nach rechts werden zwei hypothetische Fahrziele, jeweils eins für jede Möglichkeit, bestimmt und jeweils eine virtuelle Trajektorie zum hypothetischen Zielpunkt aufgebaut. Die Hypothese mit den geringsten Kosten wird als wahrscheinlichstes Fahrziel betrachtet und dient so als Prognose der Abbiegerichtung des Fahrers. Mit einer Trefferquote von 98,8% kann die Abbiegerichtung in einer Offline-Analyse zwei Sekunden vor dem Abbiegevorgang bestimmt werden.

Liebner et al. prognostizieren die Abbiegerichtung auf einer Kreuzung [Lie+12; Lie+13a]. Sie beschränken sich dabei auf die Möglichkeiten geradeaus zu fahren und rechts abzubiegen. Ihre Prognose basiert auf der gemessenen Geschwindigkeit während des Überfahrens der Kreuzung. Generell lässt sich ein Rechtsabbiegen von einem Geradeausfahren durch eine niedrigere Geschwindigkeit unterscheiden. Fährt jedoch ein Fahrzeug voraus, kann diese Unterscheidung fehlerhaft sein.

Um den durch das vorausfahrende Fahrzeug verursachten Klassifikationsfehler zu minimieren, generieren Liebner et al. über das Parametrisieren des *Intelligent Driver Models (IDM)* von Treiber et al. [THH00] Hypothesen über das Beschleunigungsverhalten des Fahrers. Unter der Annahme, dass Hypothese H_i gilt, kann so die Wahrscheinlichkeit der Abbiegerichtung bei beobachtetem Fahrverhalten O über den *Satz von Bayes* bestimmt werden. Dazu muss die Wahrscheinlichkeit $P(O|H_i)$ bestimmt werden, was über die Betrachtung des Fahrerverhaltens in den letzten Zeitschritten erfolgt.

In Szenarien ohne vorausfahrendes Auto gelingt die Unterscheidung zwischen Rechtsabbiegen und Geradeausfahren an der Kreuzung nahezu immer. Am schwierigsten ist die Unterscheidung dann, wenn das vorausfahrende Fahrzeug selbst nach rechts abbiegt. Mit einem vorausfahrenden Fahrzeug wird das Rechtsabbiegen bei der Forderung von hundertprozentiger Genauigkeit in nur 50% der Fälle erkannt. Lässt man eine Fehlerrate von circa 25% zu, so werden circa 75% der Rechtsabbiegevorgänge erkannt. 2013 erweitern Liebner et al. ihre Algorithmen und betrachten neben der Geschwindigkeit des Ego-Fahrzeugs auch die Blickrichtung des Fahrers und die Bedienung des Blinkers [Lie+13b]. Die Erkennungsrate verbessert sich dadurch deutlich. Unter Berücksichtigung von Blinker, Blick und Geschwindigkeit können bei einer Fehlerrate von 3% nahezu 98% aller Rechtsabbiegevorgänge drei Sekunden vor der eigentlichen Handlung erkannt werden.

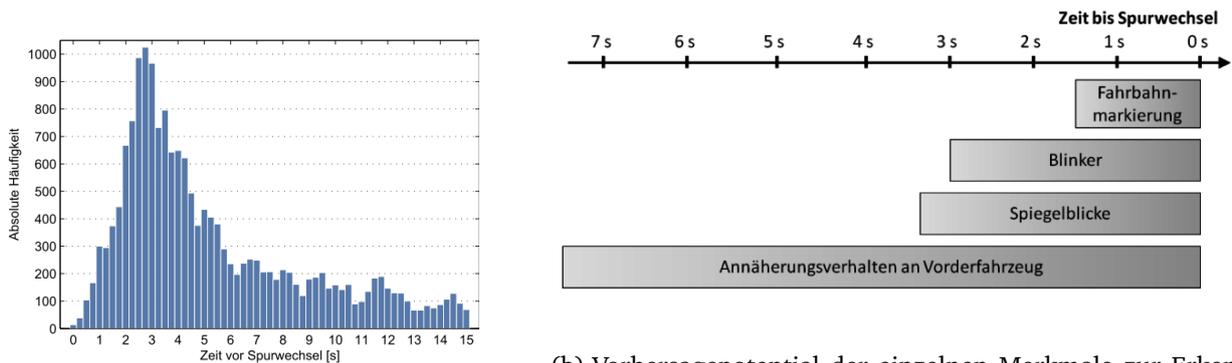
2.4.3. Prognose von Fahrspurwechseln auf der Autobahn

McCall et al. schließen aus Daten des Fahrspurerkenners, Fahrzeug-CAN-Daten und der Kopfpose des Fahrers auf die Absicht des Fahrers, die Fahrspur zu wechseln [McC+07]. Ein aus diesen drei Informationsquellen generierter Merkmalsvektor, welcher Daten der letzten Sekunde enthält, wird über *Sparse Bayesian Learning* auf die Wahrscheinlichkeit eines sich anbahnenden Spurwechsels abgebildet. Zur Klassifikation wird diese Wahrscheinlichkeit über einen Schwellwert auf die zwei Klassen [*Spurwechsel*] und [*Spur halten*] abgebildet. Ein in 2,5 Sekunden stattfindender Fahrspurwechsel kann so mit einer Fehlalarmrate von 10% und mit einer Trefferquote von 95% prädiziert werden. Schaut man drei Sekunden in die Zukunft, sinkt die Trefferquote bei 10% Fehlalarm auf 78%.

Doshi et al. untersuchen anhand des beschriebenen Fahrspurwechselekkenners von McCall et al., wie

sich Informationen über die Kopfpose bzw. die Blickrichtung des Fahrers auf die Klassifizierung auswirken [DT09]. Ein Klassifikator, welcher nur Fahrzeug- und Spurinformatoren verarbeitet, erkennt den Fahrspurwechsel in nur 49% der Fälle bei 2% Fehlerrate. Ein Klassifikator, der nur anhand der Kopf- und Blickinformation auf Spurwechsel entscheidet, erkennt 71% der Fahrspurwechsel bei einer Fehlerrate von 1%.

Ebenfalls untersuchen Liebner, Felix und Stiller die einzelnen Merkmale zur Vorhersage des Fahrspurwechsels [LFS12]. Das früheste Merkmal für einen Spurwechsel ist das Annäherungsverhalten an den Vorfahrenden gefolgt von der Blickrichtung in Form von gemessenen Blicken in den Spiegel. Abbildung 2.17 zeigt das Vorhersagepotential der einzelnen Merkmale und die Häufigkeit von Blicken in den Spiegel in den Sekunden vor einem Spurwechsel.



(a) Häufigkeit der Blicke in den Spiegel vor einem Spurwechsel. *Quelle: [LFS12]*

(b) Vorhersagepotential der einzelnen Merkmale zur Erkennung eines Spurwechsels auf der Autobahn. *Quelle: [LFS12]*

Abb. 2.17.: Vorhersagepotential der einzelnen Merkmale und die Häufigkeit von Blicken in den Spiegel in den Sekunden vor einem Spurwechsel [LFS12].

2.4.4. Generische Manöverprognose

Oliver und Pentland konzentrieren sich bei ihrer Manövererkennung auf sieben Manöver auf der Manöverebene. Diese sind [Überholen], [Abbiegen nach links], [Abbiegen nach rechts], [Spurwechsel nach links], [Spurwechsel nach rechts], [Anfahren] und [Anhalten] [OP00a; OP00b].

Zur Erkennung werden *Coupled Hidden Markov Models (CHMM)*, eine Erweiterung der *Hidden Markov Models (HMM)*, verwendet. Die bei *HMMs* implizite Annahme, dass sich ein System in *nur einem* aktuellen Zustand $s(t)$ befinden kann und dass sich die Historie des Systems auf den aktuellen Zustand beschränkt (Markovannahme), führt oft zu Einschränkungen, wenn sich gegenseitig beeinflussende Prozesse betrachtet werden.³ Zur Reduktion des Zustandsraums modellieren Oliver und Pentland das *eigene Fahrzeugverhalten* und das *Verhalten der Umgebung* als sich gegenseitig beeinflussende Prozesse und koppeln diese in einem *CHMM*.

In einem *CHMM* werden mehrere *HMMs* gekoppelt, indem der Nachfolgezustand eines einzelnen *HMMs*

³ Typischerweise wird bei der Beobachtung von zwei voneinander abhängigen Prozessen in einem *HMM* der Zustandsraum als Kreuzprodukt der Prozesszustände *A* und *B* verwendet. Dies führt jedoch schnell zu einem sehr großen Zustandsraum, für den sich die Übergangswahrscheinlichkeiten nur schwer bestimmen lassen. Durch den durch das Kreuzprodukt vervielfachten Zustandsraum werden viel mehr Lerndaten benötigt, um die Übergangswahrscheinlichkeiten statistisch aussagekräftig zu bestimmen.

nicht nur vom aktuellen Modellzustand und der Beobachtung abhängt, sondern auch von den momentanen Zuständen der anderen HMMs. Abbildung 2.18 zeigt ein zeitlich ausgerolltes *HMM* in Vergleich zu einem *CHMM*.

Zur Erkennung bilden Oliver und Pentland ein *CHMM* für jedes zu erkennende Manöver. Als Eingangs-

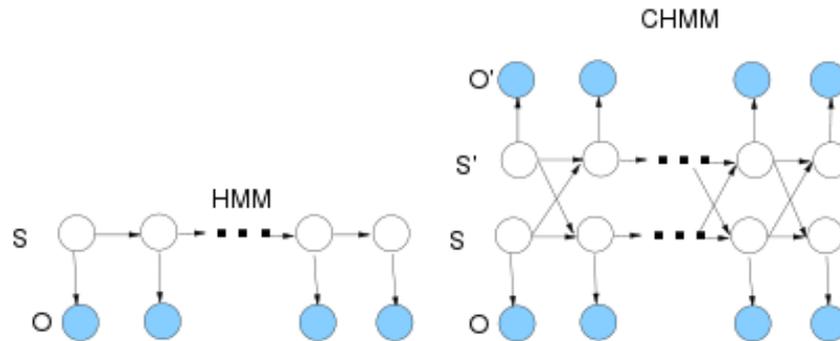


Abb. 2.18.: Ein zeitlich ausgerolltes *HMM* (links) im Vergleich zu einem zeitlich ausgerollten *CHMM* (rechts). Während beim *HMM* der Nachfolgezustand $s(t+1)$ nur vom aktuellen Zustand $s(t)$ abhängig ist, ist der nächste Zustand des *CHMMs* abhängig vom aktuellen Zustand $s(t)$ und dem aktuellen Zustand der anderen *HMMs* $s'(t)$. Quelle: [OP00a]

größen verarbeiten *CHMMs* Fahrzeugdaten, Daten über die aktuelle Strecke, Daten über die sich im lokalen Umfeld befindlichen Verkehrsteilnehmer und Daten über die Blickrichtung des Fahrers, annotiert in den sechs verschiedenen Blickfeldern.

Die von Oliver und Pentland trainierten *CHMMs* sind laut Autor in der Lage, Manöver im Durchschnitt eine Sekunde vor der Aktionsausführung zu erkennen. Am besten schneidet hierbei das Manöver [Anhalten] mit 2,4 Sekunden Vorhersagezeit ab. Das Erkennen des Überholvorgangs ist mit 1,24 Sekunden Vorhersagezeit angegeben. Die geringste Vorhersagezeit hat das Manöver [Rechtsabbiegen] mit einer Erkennung von 0,5 Sekunden vor Eintritt. Fehlalarmraten werden nicht angegeben.

Berndt und Dietmayer erkennen Spurwechsel- und Wendemanöver über HMMs [BD09]. *Hidden Markov Modelle* sind in ihrem Design für Erkennung, weniger für die Vorhersage, ausgelegt. Üblicherweise nimmt ein *HMM* eine Menge von Observationen auf und bewertet, wie gut diese Observationen zu dem gegebenen Modell passen (*Forward-Algorithmus*). Um mit *HMMs* eine Vorhersage zu realisieren, modellieren Berndt und Dietmayer ein lineares *Left-to-Right HMM* und betrachten zur Vorhersage nur die ersten Zustandsknoten des *HMMs*. Die Erkennungsraten werden mit 76% für den Spurwechsel nach links, 74% für den Spurwechsel nach rechts und 100% für ein Wendemanöver nach rechts angegeben. Die Spurwechsel- und die Wendemanöver wurden getrennt voneinander evaluiert. Berndt und Dietmayer geben an, dass die modellierten Manöver schon in einem frühen Zustand erkannt werden. Eine quantitative Vorhersagezeit wird nicht angegeben.

2.4.5. Zusammenfassung und Bewertung

Derzeitige Arbeiten zur Prognose des eigenen Fahrverhaltens beschränken sich auf die Manöverebene, wobei die Verhaltensprognosen auf bestimmte Manöver reduziert sind. Lediglich Oliver und Pentland und Berndt und Dietmayer stellen generische Methoden vor, die auf andere Manöver erweiterbar sind.

Beide nutzen *HMMs* zur Erkennung des Manövers, welche in der Regel dazu verwendet werden, eine Beobachtungskette *nach* dem Eintreffen der letzten Beobachtung einem Modell zuzuordnen. Der Vorhersagehorizont liegt in allen Arbeiten bei zwei bis drei Sekunden. Die Vorhersagezeit ist stark von der Art und Weise der Annotation des Fahrspurwechsels abhängig. Neuere Arbeiten beziehen die Blickrichtung des Fahrers in die Prognose mit ein.

Ist die Absichtserkennung nicht auf ein bestimmtes Manöver beschränkt (generische Absichtserkennung), gestaltet sich die Modellierung neuer Manöver oft als komplex (*CHMMs*). Fahrerspezifische Informationen, wie der Fahrstil des Fahrers oder aus dem Fahrverhalten bestimmte Komfortparameter, werden implizit durch das Trainieren (beispielsweise eines *HMMs*) gelernt, wodurch die Verhaltensprognose nachträglich nur schwer auf andere Fahrertypen adaptiert werden kann.

2.5. Modellierung von Fahrerverhalten - Wissensrepräsentation und Inferenz

Die in diesem Abschnitt vorgestellten Arbeiten beschäftigen sich mit der Modellierung von Fahrerverhalten und unterscheiden sich so in der Zielsetzung von den Arbeiten aus den vorherigen Abschnitten. Während die Arbeiten aus den Abschnitten 2.2 bis 2.4 primär die Zielsetzungen hatten, eine Kopfpose, den Fahrstil oder das nächste Manöver des Fahrers zu schätzen, widmen sich diese Arbeiten nun der Meta-Ebene der Modellierung. In allen bisher vorgestellten Arbeiten wurde in irgendeiner Form Wissen über den Fahrer gespeichert, fusioniert oder abgeleitet. Dies geschah jedoch als *Mittel zum Zweck*, um die eigentliche Aufgabe zu bewältigen und stand nicht im Fokus der bisher vorgestellten Arbeiten. Hier liegt im Fokus, wie Wissen über den Fahrer in einem Modell repräsentiert werden kann und wie über das abgelegte Wissen neues Wissen oder bestehende Zusammenhänge abgeleitet werden können (Inferenz). Vertiefend werden die Arbeiten von Dagli et al. und Salvucci in den Abschnitten 2.5.1 bzw. 2.5.2 vorgestellt. Abschnitt 2.5.3 gibt eine Bewertung und eine Zusammenfassung der vorgestellten Modellierungsverfahren.

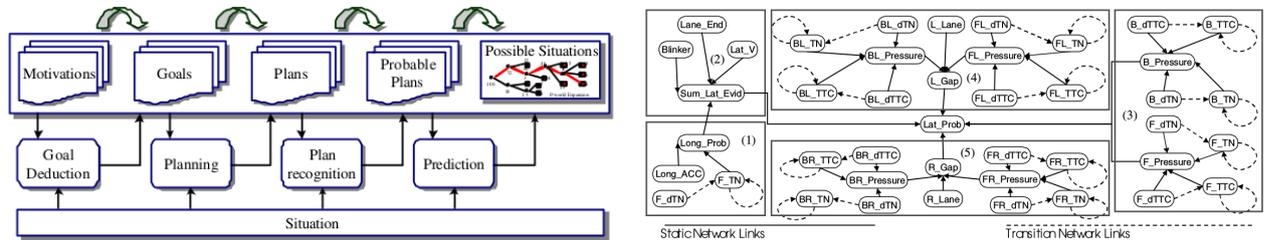
2.5.1. Fahrerverhalten modelliert über Dynamic Belief Networks (DBNs)

Dagli, Brost und Breuel modellieren Fahrerverhalten in einem *Dynamic Belief Network (DBN)* [DBB03; DR02]. In einer Situation mit vielen Verkehrsobjekten ist die Prädiktion aller möglichen zukünftigen Situationen eine kombinatorisch komplexe Aufgabe. Ihre Arbeit zielt deshalb darauf ab, die Motivation des Fahrers zu erkennen, um so den Hypothesenraum für die Prognose zukünftiger Verkehrssituationen reduzieren zu können.

Sie treffen dabei die Grundannahme, dass der Fahrer die Fahraufgabe von der Motivation getrieben ausführt und aus seiner Motivation Ziele und Pläne ableitet. Abbildung 2.19a zeigt den Zusammenhang von Motivationen, Zielen und Plänen nach dem Verständnis von Dagli et al. Im von ihnen entworfenen *DBN* repräsentieren die Knoten Zustände des eigenen Verhaltens und des lokalen Fahrzeugumfelds. In den Kanten werden die bedingten Abhängigkeiten zwischen den Beobachtungen modelliert.

Knoten ohne Vorgängerknoten repräsentieren dabei direkt beobachtbare Zustände und bilden die Eingangs-

beknoten des DBNs. Ein direkt beobachtbarer Zustand ist nach Dagli et al. beispielsweise ein gesetzter Blinker (eigenes Verhalten) oder ob die Ego-Fahrspur endet. Knoten ohne Nachfolgerknoten enthalten die Ausgabe des Netzes (Query-Nodes). Sie beinhalten die geschätzte Wahrscheinlichkeit einer longitudinalen bzw. lateralen Aktion. Knoten mit Vorgänger- und Nachfolgerknoten repräsentieren nicht beobachtbare Zustände, deren Wahrscheinlichkeit sich aus den bedingten Abhängigkeiten der Eingabeknoten ableitet. Abbildung 2.19b zeigt das entworfene Dynamic Belief Network. Dagli et al. beziehen ihre Eingabedaten



(a) Dagli et al. gehen davon aus, dass der Fahrer getrieben von Motivationen ist, aus denen sich Ziele und Pläne ableiten lassen.
 Quelle: [DBB03]

(b) Die Modellierung des Fahrerhaltens findet in einem Dynamic Belief Network (DBN) statt, welches Zustände und ihre bedingten Abhängigkeiten beinhaltet.
 Quelle: [DBB03]

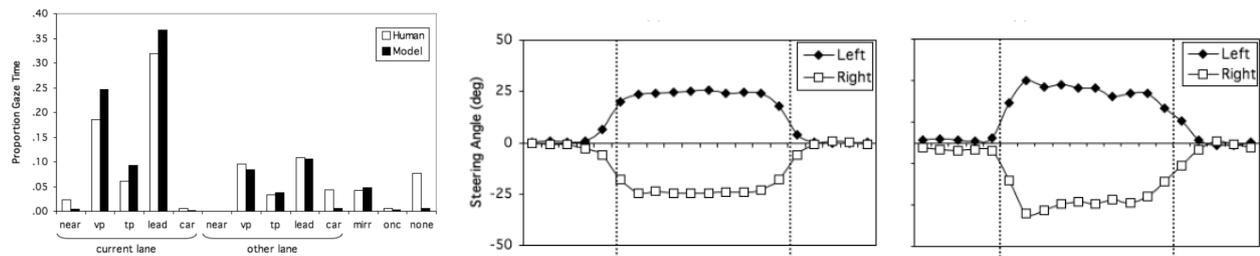
Abb. 2.19.: Dagli et al. modellieren das Fahrerverhalten in einem Dynamic Belief Network (DBN) in der Annahme, dass der Fahrer die Fahraufgabe von seiner Motivation getrieben ausführt.

aus einer Verkehrssimulation. Ihre Modellierung beschränkt sich dabei nicht auf ein bestimmtes Fahrzeug, sondern wird global auf alle Fahrzeuge der Simulation angewandt. Mit dem entwickelten DBN kann ein Überholvorgang circa 1,5 Sekunden vor dem Eintreffen detektiert werden.

2.5.2. Fahrerverhalten modelliert über die ACT-R Architektur

Salvucci bildet das Problem der Modellierung des Fahrerhaltens auf das Modellieren von menschlichem Verhalten ab [Sal06]. Sie teilen die Tätigkeiten des Fahrers in *primäre* und *sekundäre* Tätigkeiten. Primäre Tätigkeiten sind dabei alle Handlungen des Fahrers, die zur Bewältigung der Fahraufgabe dienen. Sekundäre Tätigkeiten, wie beispielsweise das Bedienen des Radios oder der Klimaanlage, verbessern den Reisekomfort und das Wohlbefinden des Fahrers. Je nach Ausstattung des Fahrzeugs können sich, bedingt durch die Anbringung der Bedienelemente, die sekundären Tätigkeiten sehr unterscheiden. Eine geeignete Modellstruktur für das Modellieren von Fahrerverhalten sieht Salvucci in der kognitiven *ACT-R-Architektur (Adaptive Control of Thought-Rational)*, die im Wesentlichen vom Kognitionspsychologen *John R. Anderson* entwickelt wurde. Kognitive Architekturen werden verwendet, um menschliches Verhalten computergestützt zu simulieren. Die *ACT-R Architektur* beschränkt sich dabei nicht auf das Fahrerverhalten, sondern ist konzipiert um menschliches Verhalten im Allgemeinen zu modellieren. Wissen wird in der *ACT-R-Architektur* über *Chunks* und *Produktionsregeln* repräsentiert. Die Bezeichnung *Chunk* (engl. *Brocken, Stückchen*) wird benutzt, weil auch der Mensch sein Wissen im Gedächtnis in kleineren „Wissensstückchen“ abspeichert, um sie unkomplizierter und schneller abrufen zu können. Produktionsregeln operieren auf dem Faktenwissen und können auch neues Faktenwissen erzeugen. In der *ACT-R-Architektur* sind Produktionsregeln nach ihrer Zuverlässigkeit bewertet. Je öfter eine Produktionsregel ausgeführt und das erwartete Ergebnis bestätigt wurde, desto zuverlässiger wird die

Produktionsregel eingestuft. Weiter modelliert die *ACT-R-Architektur*, analog zum menschlichen Erinnern bzw. Vergessen, ein falsches bzw. fehlendes Abrufen von deklarativem und prozeduralem Wissen. Wird ein *Chunk* oft abgerufen, ist er schneller und zuverlässiger verfügbar; wird er hingegen nur selten gebraucht, steht er langsamer zur Verfügung und wird öfter auch einmal „vergessen“. Um das entworfene Fahrermodell zu verifizieren, vergleicht Salvucci das virtuelle Verhalten mit dem Fahrverhalten von elf verschiedenen Fahrern im Fahrsimulator [Sal06]. Sie bewerten die Gleichheit des Verhaltens über den Korrelationskoeffizienten nach *Pearson* und stellen für den Lenkradwinkel eine Übereinstimmung von 98% fest. Auch die Blickverteilung des Modells entspricht der Blickverteilung eines echten Fahrers. Abbildung 2.20 visualisiert die Ergebnisse.



(a) Die relativen Häufigkeiten, mit der Mensch und Modell die jeweiligen Regionen anblicken, sind nahezu identisch. (b) Das Lenkverhalten des menschlichen Fahrers (links) entspricht dem Lenkverhalten des Modells (rechts) mit einem Korrelationskoeffizienten zu 98%.
Quelle: [Sal06]

Quelle: [Sal06]

Abb. 2.20.: Salvucci vergleicht das Lenkverhalten und das Blickverhalten beim Fahrspurwechsel des modellierten Fahrerhaltens mit dem Verhalten eines menschlichen Fahrers im Fahrsimulator.

2.5.3. Zusammenfassung und Bewertung

Die Wissensmodellierung von Salvucci zielt mit ihrer ACT-R-Architektur darauf ab, den *Menschen* nachzubilden. So wird beispielsweise ein „Vergessen“ modelliert. Wissensmodellierung soll in dieser Arbeit nicht als *Nachbildung* des Fahrerhaltens verstanden werden, sondern als möglichst detaillierte Informationsquelle über den Fahrer und dessen Verhalten. Das *Dynamic-Belief-Netz* von Dagli, Brost und Breuel entspricht in ihrer Wissensmodellierung dieser Philosophie. Allerdings ist das entworfene Netz stark auf die Erkennung eines Überholvorgangs ausgelegt. Mitunter wird die Arbeit stark von der Datenquelle „Verkehrssimulation“ geprägt. Es fließen keine fahrerspezifischen Parameter, wie die Blickrichtung oder die individuellen Komfortparameter des Fahrers, in die Verhaltensmodellierung mit ein. Es ist zu erwarten, dass die Komplexität, und damit die Rechenzeit für die Inferenz des Netzes, durch Hinzunahme von fahrerspezifischen Kennwerten stark steigt. Forschungsbedarf besteht vor allem in der Modellierung des Fahrerhaltens in probabilistischer Form. In der Modellierung von Dagli, Brost und Breuel über *DBNs* wird davon ausgegangen, dass eine Beobachtung zweifelsfrei ist. Dies mag in einer Verkehrssimulation gelten, in der realen Welt sind Beobachtungen der Umwelt über Sensoren aber immer verrauscht. Eine probabilistische Modellierung kann damit am ehesten umgehen. Eine Modellierung des Fahrerhaltens sollte deshalb die Möglichkeit bieten, Beobachtungen probabilistisch abzubilden und fehlende Beobachtungen zu kompensieren. In der Modellierungsmethode sollte in transparenter Art und Weise

Faktenwissen definierbar und gleichzeitig ein Lernen von Zusammenhängen möglich sein. Über die Modellierung abgeleitete Zusammenhänge (Inferenz) müssen darüber hinaus in Echtzeit berechenbar sein.

2.6. Zusammenfassung und Identifikation des Forschungsbedarfs

Um den Fahrer im Automobil der Zukunft ganzheitlich einzubinden, bedarf es einer ausgeprägten sensorischen Erfassung des Fahrers (Perzeption), einer Analyse des Verhaltens des Fahrers im Zusammenhang mit den Geschehnissen im lokalen Fahrzeugumfeld (Situationsanalyse) und einer Prognose des Fahrverhaltens in naher Zukunft (Situationsprognose).

Das Themenfeld der Perzeption ist dabei am weitesten erforscht. Es gibt zahlreiche Veröffentlichungen zur Erfassung von Vitalität, Emotionen sowie der Körperstellung. Insbesondere die Stellung des Kopfs und der Augen (die Blickrichtungserkennung) liegen im Zentrum der Forschung, da beim Fahren 90% aller Informationen visuell aufgenommen werden. Macht man sich die hohen Ansprüche der Automobilindustrie an eine Blickrichtungsschätzung bewusst, wird jedoch verständlich, warum in diesem Bereich immer noch Forschungsbedarf besteht. Systeme mit IR-Belichtung sind sehr akkurat, aus wirtschaftlicher Sicht zum heutigen Zeitpunkt jedoch in Serie nicht integrierbar. CCD-Kamerasysteme sind auf einen Erkennungsbereich von etwa $\pm 60^\circ$ beschränkt und scheitern häufig an den während der Fahrt ständig wechselnden Lichtverhältnissen. RGB-D-Kamerasysteme sind derzeit für den Automobilbereich nur wenig erforscht.

Konkret wurde für diese Arbeit folgender Forschungsbedarf im Bereich der Perzeption identifiziert:

- Konzeption einer echtzeitfähigen, nicht-intrusiven Schätzung der Blickrichtung unter stark wechselnden Lichtverhältnissen. Die Blickrichtungserkennung muss für schnelle und weiträumige Kopfdrehungen ausgelegt sein und zur Erkennung von Blicken in den Spiegel insbesondere die Augenstellung berücksichtigen. Die Sensorik muss für den Automobilbereich sowohl in wirtschaftlicher, als auch aus Sicht der Integrationsmöglichkeit im Fahrzeuginnenraum, geeignet sein.

Die Interpretation der Informationsquellen *Umwelt*, *Fahrer* und *Fahrzeug* (im Kontext *Kognitiver Automobile* oft als *Situationsanalyse* bezeichnet) wurde in dieser Arbeit in drei Forschungsfelder strukturiert: *Analyse des Blickverhaltens*, *Analyse des Fahrverhaltens* und *Prognose des Fahrverhaltens*.

Arbeiten, welche die Analyse des Blickverhaltens untersuchen, beschränken sich auf Einzelanwendungen (zum Beispiel der Warnung bei einem übersehenen Verkehrsschild) oder bestimmen einen *globalen* Aufmerksamkeitswert des Fahrers. Gerade Zukunftsvisionen wie der *Intelligente Beifahrer* oder *Kooperatives Fahren* wecken Forschungsbedarf in der

- Ableitung einer objektspezifischen Aufmerksamkeit des Fahrers, welche eine situative Differenzierung zwischen einerseits normalem und andererseits unüblichem Blickverhalten erlaubt.

In der aktuellen Forschung rund um die Analyse des Fahrverhaltens sind Verbesserungen in der Integration des situativen Verkehrskontexts möglich. Forschungsbedarf besteht vor allem in der

- Schätzung des Fahrstils im gegenwärtigen Situationskontexts zur Abschätzung des Risiko- oder Energieverhaltens des Fahrers.
- Schätzung fahrerindividueller Komfortparameter zur quantitativen Beschreibung des Fahrerverhaltens im situativen Kontext.

Die Prognose von Fahrverhalten ist, verglichen mit der Fahrerbeobachtung, nur wenig erforscht. Die vorgestellten Ansätze beschränken sich auf einfache Szenarien und sind nur schwer auf andere Fahrertypen und andere Verkehrssituationen übertragbar. Folgender Forschungsbedarf wurde für diese Arbeit im Themenfeld der Prognose des Fahrverhaltens identifiziert:

- Entwicklung einer generischen, erweiterbaren Methode zur Prognose des Fahrverhaltens unter Berücksichtigung des fahrerspezifischen Verhaltens.

Das Forschungsgebiet der *Modellierung des Fahrerverhaltens* richtet sich auf Untersuchungen die behandeln, wie Wissen über den Fahrer in einem Modell repräsentiert werden kann und wie aus diesem neues Wissen und neue Zusammenhänge hinsichtlich des Fahrverhaltens abgeleitet werden können (Inferenz). Aufgrund der nur partiell beobachtbaren Umwelt und fehlerbehafteter Messungen der Perzeptionskomponenten muss eine Modellierung des Fahrerverhaltens die Möglichkeit bieten, Beobachtungen probabilistisch zu schließen und fehlende Beobachtungen zu kompensieren. Derartige Modelle sind derzeit kaum veröffentlicht. Für diese Arbeit wurde daher folgender Forschungsbedarf im Themenfeld der Modellierung des Fahrverhaltens identifiziert:

- Modellierung des Fahrerverhaltens über ein probabilistisches, inferenz- und prädiktionsfähiges Modell, in welchem Faktenwissen transparent definierbar ist und Zusammenhänge gelernt werden können.

Zahlreiche Veröffentlichungen tragen zur besseren Einbindung des Fahrers in das intelligente Fahrzeug der Zukunft bei. Die fahrerfokussierte Situationsanalyse und damit die maschinelle Einschätzung des individuellen Fahrerverhaltens hinsichtlich der sich wechselseitig beeinflussenden Komponenten *Fahrer*, *Fahrzeug* und *Fahrzeugumfeld* stellt für die Wissenschaft eine wesentliche Herausforderung dar.

3. Blickrichtungserkennung im Automobil

Die Blickrichtung spielt eine zentrale Rolle in der Analyse des Fahrerverhaltens. Dennoch sind Blickrichtungserkennungssysteme (Stand 2015) noch immer nicht serienmäßig im Automobil integriert. Selbst in der Wahl der Sensorik scheint kein Konsens gefunden zu sein. Ein zentraler Grund sind die hohen Anforderungen, welche die dynamische Umgebung „Fahrzeug“ einer Blickrichtungserkennung stellt. Abschnitt 3.1 stellt die Anforderungen an ein Blickrichtungserkennungssystem im Automobil zusammen. Gemäß den Schlussfolgerungen aus dem Stand der Technik (Seite 8 ff.), wird in dieser Arbeit die Schätzung der Blickrichtung über CCD-Kameras und RGB-D-Kameras untersucht. Abschnitt 3.2 beschreibt die Realisierung eines Blickrichtungserkennungssystems basierend auf Eingabedaten einer RGB-D-Kamera. In Abschnitt 3.3 wird die Implementierung zur Blickrichtungsbestimmung über eine hochaufgelöste CCD-Kamera beschrieben. Der Systementwurf und die Algorithmik der Erkennungssysteme ist speziell auf die Erfüllung der in Abschnitt 3.1 formulierten Anforderungen im Automobilbetrieb ausgelegt. Neben der erkannten Blickrichtung sind für eine weitergehende Situationsanalyse (zum Beispiel zur Manöverprädiktion in Abschnitt 5.4, Seite 134) vorverarbeitete Informationen, welche den Blick des Fahrers schon in Bezug zur dynamischen Umgebung interpretieren, erforderlich. Abschnitt 3.4 beschreibt, wie im Zusammenspiel mit umfelderfassender Sensorik Objekte des Straßenverkehrs, Fahrspuren und Blickfelder als gesehen oder ungesehen bewertet werden. Abschließend fasst Abschnitt 3.5 die Beiträge auf dem Gebiet der Blickrichtungserkennung im Kontext *Intelligenter Automobile* zusammen.

3.1. Anforderungen an die Blickrichtungserkennung im Automobil

Die Domäne „Automobil“ stellt an ein System zur Blickrichtungserkennung höchste Ansprüche: Zum einen müssen wechselnde Lichtverhältnisse oder Verdeckungen des Gesichts kompensiert werden, zum anderen muss die Blickrichtungsschätzung sicher und genau sein, sowie den schnellen und weiträumigen Kopfbewegungen des Fahrers nachkommen. Im Speziellen muss eine Blickrichtungsbestimmung im Automobil folgenden Ansprüchen genügen:

Sicherheit: Die primäre Anforderung stellt ohne Frage die Gewährleistung der Sicherheit des Fahrers dar. Die gewählte Sensorik muss „nicht-intrusiv“ sein, d.h. sie darf den Fahrer in seinem Handeln nicht stören, die Sicht behindern oder spezielle Montagen am Kopf des Fahrers fordern.

Genauigkeit: Über die Blickrichtungserkennung muss es möglich sein Objekte des nahen Fahrzeugumfelds als gesehen oder übersehen einzuordnen. Die Blickrichtungserkennung muss darüber hinaus in der Lage sein, Blicke auf den Spiegel, den Tachometer oder das Infotainmentsystem, welche nur durch kurze Augenbewegungen ohne eigentliche Bewegung des Kopfs getätigt werden,

zu erkennen. Die Blickrichtungserkennung kann demnach nicht auf das Erkennen der Kopfpose reduziert werden. Abbildung 3.10 auf Seite 63 zeigt, wie sich die Kopfstellung während eines Blicks nach vorne und eines Blicks in den Innenspiegel kaum unterscheiden.

Schnelligkeit der Kopfbewegung: Der Mensch bewegt seinen Kopf mit Geschwindigkeiten von bis zu $250^\circ/\text{s}$ [Wik14a]. Im Automobil treten beim Einparken oder Schulterblick durchaus schnelle Kopfbewegungen auf. Zwar scheint die Maximalgeschwindigkeit als Systemanforderung übertrieben, trotzdem sollte das System für schnelle Kopfrotationen ausgelegt sein.

Echtzeitfähigkeit: Die Blickrichtungserkennung muss für Eingabedaten ein Ergebnis unterhalb einer fest definierten Zeitschranke liefern. Durch eine geforderte Echtzeitfähigkeit ist die Laufzeitverzögerung vom Zeitpunkt der (Bild-)Aufnahme bis zum Zeitpunkt des Ergebnisses wohldefiniert. Als Ergebnis kann dabei auch gelten, dass zu den aktuellen Eingabedaten keine Blickrichtung geschätzt werden kann.

Für FAS, welche die Blickrichtung als Informationsquelle in fest getakteten Regelkreisen verarbeiten, kann so gewährleistet werden, dass die Blickrichtungsinformation zu einem festen Zeitpunkt weiterverarbeitet werden kann.

Erkennungsbereich: Die Blickrichtungserkennung muss den Fahrer für seinem kompletten Bewegungsfreiraum erfassen. Speziell für den Gierwinkel bedeutet dies einen Wertebereich von einem Schulterblick nach links bei circa -100° bis zu einem Blick nach hinten beim Rückwärtsfahren von circa $+180^\circ$ (mit 0° als nach vorne gerichteter Blick).

Reproduzierbarkeit: Die Reproduzierbarkeit eines Ergebnisses bei gegebenem Eingabebild spielt in der Automobilindustrie für Zertifizierungen und Integrationstests eine wichtige Rolle.

Kompensation wechselnder Lichtverhältnisse: Die Robustheit bei wechselnden Lichtverhältnissen ist eine der wichtigsten Anforderungsunterschiede gegenüber einer statischen Laborumgebung. Da sich während der Fahrt die Umgebung ständig ändert sind Gegenlicht, plötzliche Dunkelheit oder Überblendungen und dynamische Schattenwürfe keine Seltenheit.

Kompensation von Verdeckungen: Gesichtsverdeckungen, wie durch (Sonnen-)Brillen, in das Gesicht fallende Haare oder ein Griff ins Gesicht darf keine Falschschätzung der Blickrichtung auslösen.

Hardware-Integration: Die Sensorik muss für den Automobilbereich sowohl aus wirtschaftlicher, als auch aus Sicht der Integrationsmöglichkeit im Fahrzeuginnenraum geeignet sein.

Mehrbenutzersystem: Moderne Verkehrskonzepte wie *CarSharing* fordern von einer im Fahrzeug integrierten Blickrichtungserkennung einen Umgang mit mehreren und unbekanntem Benutzern ein.

3.2. Blickrichtungserkennung über eine RGB-D-Kamera

Die Schätzung der Blickrichtung leitet sich aus der Erkennung der Kopfpose über das Tiefenbild mit anschließender Analyse der Augenregionen im RGB-Farbbild ab [BRZ12c; BRZ12a]. Die Schätzung der Kopfpose wird über ein *Multi-Template-ICP-Verfahren* realisiert. Um den Ansprüchen Echtzeitfähigkeit, Genauigkeit und Robustheit zu genügen wurden neben anderen Modifikationen zwei Schritte des ICP-Verfahrens optimiert: Zum einen wurde eine schnelle Korrespondenzsuche im Tiefenbild implementiert und zum anderen wechselt die Fehlerfunktion entsprechend des Iterationsfortschritts von einem Punkt-zu-Punkt-Abstandsmaß auf ein Punkt-zu-Ebene-Abstandsmaß. Eine Beschreibung des verwendeten ICP-Verfahrens, sowie die speziell für die Automobilanwendung benötigten Modifikationen findet sich in Abschnitt 3.2.2.

Das verwendete ICP-Verfahren verwendet zur Bestimmung der Kopfpose speziell für den Fahrer optimierte Gesichtsmasken. Zum Laden der fahrerindividuellen Gesichtsmasken wird der Fahrer über seine Silhouette identifiziert. Die Identifikation des Fahrers über seine Silhouette wird im Abschnitt 3.2.3 beschrieben.

RGB-D-Kamerasysteme bestrahlen die Szene mit strukturiertem Licht um Tiefeninformationen zu extrahieren. Starke Sonneneinstrahlung sowie für den Belichtungsregler zu schnell wechselnde Lichtverhältnisse haben einen Verlust der Tiefeninformation zur Folge. Die Kopfposenbestimmung über das ICP-Verfahren schlägt mit dem Störung des Tiefenbilds fehl. Um die Kopfpose trotz fehlendem Tiefenbild akkurat zu bestimmen „lernt“ das entworfene System im Normalbetrieb Relationen von Gesichtsmerkmalen und Kopfposen über *Random Forests*. Tritt eine Tiefenbildstörung auf, wird die Kopfpose des Fahrers allein aus dem RGB-Farbbild über den im Normalbetrieb trainierten *Random Forest* bestimmt.

Abschnitt 3.2.4 beschreibt die Vorgehensweise zur Kopfposenbestimmung bei fehlerhafter Tiefeninformationen.

Die Ermittlung der Pupillenstellung über eine orthographische Projektion basierend auf der erkannten Kopfpose und einem Augenmodell wird in Abschnitt 3.2.5 beschrieben.

Einführend geht Abschnitt 3.2.1 auf die positiven als auch negativen Eigenschaften von RGB-D-Kameras für die Blickrichtungserkennung im Fahrzeug ein.

3.2.1. Eignung der RGB-D-Kamera zur Blickrichtungserkennung im Automobil

RGB-D-Kameras kombinieren Farb- und Tiefenbild in einem Sensorsystem. Die Definition eines *Bilds* wird aus [Nie14, Seite 35ff] übernommen:

Definition 3.1. Ein *Bild* $\mathbf{B} \in \mathfrak{F}^{h,w}$ mit Breite w und Höhe h ist eine $h \times w$ Matrix über dem Farbraum \mathfrak{F} . Seine Komponenten $b_{i,j} \in \mathfrak{F}$ werden *Bildpunkte* oder *Pixel* genannt. Für die Bildpunkte eines Bildes gilt die Notation

$$b(u, v) := b_{u,h-v} \quad [3.1]$$

mit $(u, v) \in \mathbb{N}_w \times \mathbb{N}_h$ als den *Bildkoordinaten* des Bilds. Ein *RGB-Farbbild* ist ein Bild im RGB-Farbraum $\mathfrak{R} := \mathbb{N}_{255}^3$.

Ein *Tiefenbild* ist eine spezielle Form eines Bildes, welches statt der sonst üblichen Farbinformation Tiefeninformation beinhaltet.

Definition 3.2. Ein *Tiefenbild* ist eine $h \times w$ Matrix über dem Raum \mathbb{R} .

Die Tiefeninformation wird gewonnen indem ein Lichtmuster außerhalb des für den Menschen sichtbaren Frequenzbereichs auf die Szene projiziert wird und über eine im gleichen Spektralbereich sensitive Kamera aufgenommen wird. Durch die Verzerrung des projizierten Lichtmusters in der Szene kann durch Korrespondenzsuche Tiefeninformation gewonnen werden. Tiefenbildkameras stellen zunächst ein *Tiefenrohwertbild* als $h \times w$ Matrix über dem Farbraum \mathbb{N}_{255}^1 bereit. Durch eine aus den technischen Daten der Kamera bekannte Funktion $\kappa : \mathbb{N}_{255}^1 \rightarrow \mathbb{R}$ können die Tiefenrohwerte der Kamera in Distanzen umwandelt werden. Ist in dieser Arbeit von einem *Tiefenbild* die Rede, so ist das durch die Funktion κ transformierte Tiefenrohwertbild gemeint. Sind die intrinsischen Kameraparameter der Kamera bekannt, kann jeder Bildpunkt einem Szenepunkt zugeordnet werden und aus dem Tiefenbild eine Punktwolke P im Sensorkoordinatensystem generiert werden (siehe Formel 3.10).

Definition 3.3. Ein *3-D-Punkt, Szenepunkt oder Objektpunkt* $\mathbf{p} \in \mathfrak{P}$ ist ein Vektor, welcher mindestens die Elemente x, y, z beinhaltet. Die Menge aller Punkträume \mathfrak{R} kann als

$$\mathfrak{R} = \{ \mathbb{R}^3 \times \mathbb{M}^k \mid k \in \mathbb{N}_0 \} \quad [3.2]$$

formuliert werden. Der an den dreidimensionalen Raum \mathbb{R}^3 angehängte Raum \mathbb{M}^k steht für eine beliebige Vektorerweiterung des Punktes. In einem Punktraum $\mathfrak{P} \in \mathfrak{R}$ folgen somit alle Punkte der Struktur

$$\begin{bmatrix} x \\ y \\ z \\ \vdots \end{bmatrix} \quad [3.3]$$

Anmerkung 1: Ein Punkt ist immer relativ zu einem Koordinatenursprung eines Koordinatensystems (siehe Definition 3.5).

Anmerkung 2: Wird aus dem Kontext klar, dass es sich um einen 3-D-Punkt handelt, wird auf den „3-D“ Präfix verzichtet.

Definition 3.4. Eine *Punktvolke* P ist eine endliche Menge von Punkten $\mathbf{p}_i \in \mathfrak{P}$.

$$P := \{ \mathbf{p}_0, \dots, \mathbf{p}_N \mid \mathbf{p}_i \in \mathfrak{P} \} \quad [3.4]$$

Anmerkung 1: Die Punkte einer Punktvolke sind immer relativ zu einem Koordinatenursprung des selben Koordinatensystems (siehe Definition 3.5).

Definition 3.5. Ein *Koordinatensystem* dient zur eindeutigen Bezeichnung der Position von Punkten und Objekten in einem geometrischen Raum. In dieser Arbeit werden stets orthogonale, rechtshändige Koordinatensysteme verwendet. Mit der Einschränkung auf orthogonale, affine Koordinatensysteme kann ein Punkt $\mathbf{p} \in \mathfrak{P}$ in Bezug auf ein Koordinatensystem A über eine affine Transformation in einen Punkt $\mathbf{p}' \in \mathfrak{P}$ in Bezug auf ein Koordinatensystem B transformiert werden. Abbildung 3.1 stellt die für diese Arbeit relevanten Koordinatensysteme als Graphen dar.

Definition 3.6. Eine *affine Koordinaten-(Transformation)* überführt einen Punkt über eine Rotationsmatrix $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ und einem Translationsvektor $\mathbf{t} \in \mathbb{R}^3$ von einem *Koordinatensystem* in ein anderes Koordinatensystem.

$$\mathbf{p}' = \mathbf{R}\mathbf{p} + \mathbf{t} \quad [3.5]$$

Über die homogene Erweiterung kann eine Transformation anhand einer Matrix \mathbf{T} dargestellt werden:

$$\begin{bmatrix} \mathbf{p}' \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix} = \mathbf{T} \begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix} \quad [3.6]$$

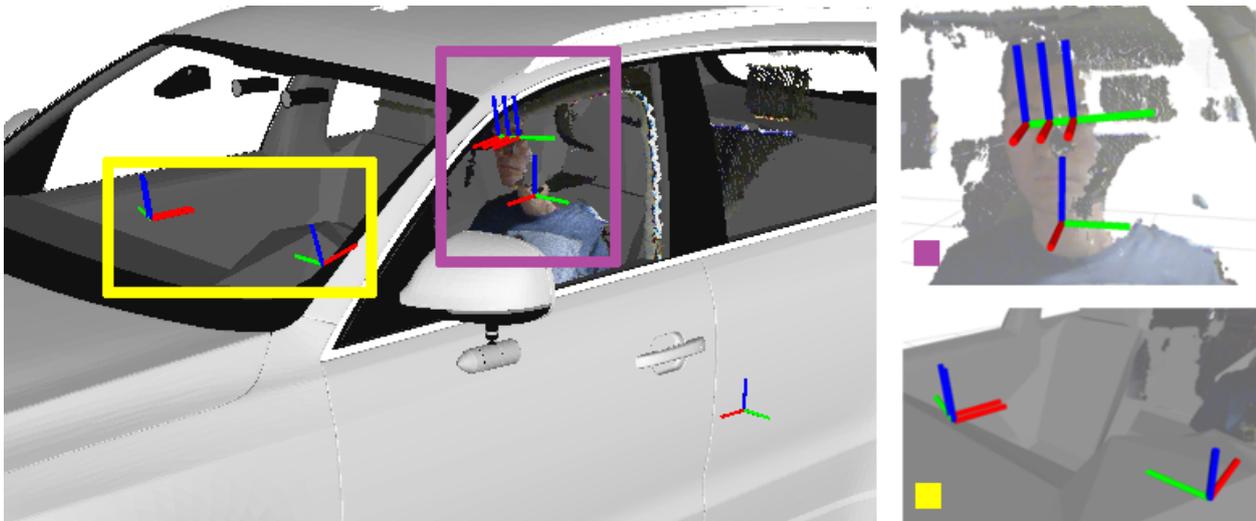
Unter dem Term Transformation wird in dieser Arbeit immer eine affine Koordinatentransformation verstanden.

Anmerkung 1: Um darzustellen, dass ein Punkt vom Koordinatensystem A in das Koordinatensystem B transformiert wird, wird die Notation ${}^B\mathbf{T}_A$ verwendet. Ein Punkt \mathbf{p} relativ zum Koordinatensystem A wird über die Multiplikation von Links mit der Transformationsmatrix ${}^B\mathbf{T}_A$ in einen Punkt \mathbf{p}' relativ zum Koordinatensystem B transformiert.

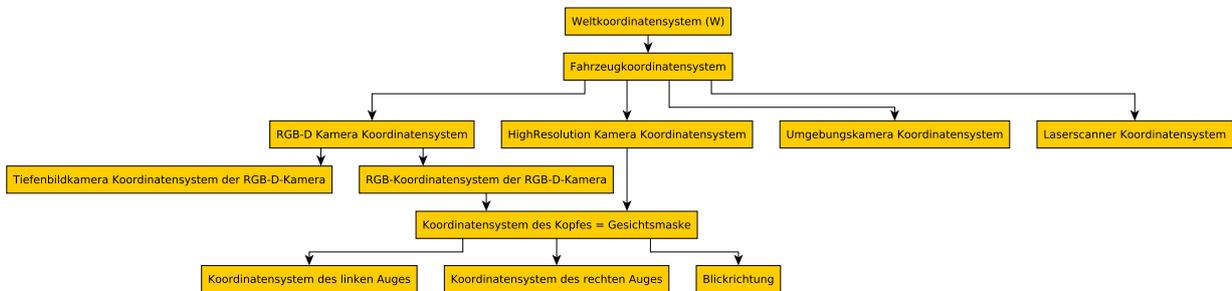
$$\mathbf{p}' = {}^B\mathbf{T}_A \mathbf{p} \quad [3.7]$$

Anmerkung 2: Zu einer Transformation ${}^A\mathbf{T}_B$ gibt es stets eine inverse Transformation ${}^B\mathbf{T}_A$.

Technische Eigenschaften von RGB-D-Kameras können Datenblättern und Fachliteratur entnommen werden (zum Beispiel [Lit12; And+12; KE12; Asu14]). Im Folgenden werden die positiven wie negativen Eigenschaften in Bezug auf eine Blickrichtungserkennung im Fahrzeug vorgestellt:



(a) Räumliche Visualisierung der für diese Arbeit verwendeten Koordinatensysteme.



(b) Graphische Darstellung von Koordinatensystemen als Knoten und Koordinatensystemtransformationen als Kanten. Eine Transformation zwischen im Graphen nicht direkt verbundenen Knoten A, B kann abgeleitet werden, indem ein Pfad zwischen A und B gefunden wird und die im Pfad enthaltenen Transformationen verkettet werden. Soll beispielsweise ein Szenepunkt vom RGB-D-Kamera-Koordinatensystem ins Laserscanner-Koordinatensystem transformiert werden, wird dieser in das Fahrzeugkoordinatensystem und anschließend in das Laserscanner-Koordinatensystem transformiert.

Abb. 3.1.: Räumliche und topologische Visualisierung der im Kontext der Blickrichtungserkennung Koordinatensysteme.

Vorteil Tiefenbild Abgesehen vom Lidschlag wird die vom Mund aus gesehene obere Gesichtshälfte vom Menschen kaum bewegt. Sie bildet sich als rigider, skalierungsinvarianter Körper in einer aus dem Tiefenbild gewonnenen Punktwolke ab. Die Schätzung der Transformation dieses rigiden, skalierungsinvarianten Körpers zur Kamera liefert direkt die Kopfpose des Fahrers.

Vorteil Tiefenbildregistrierung RGB-D-Kameras bestehen im Wesentlichen aus einer CCD-Kamera (RGB-Farbbild), einer Infrarot-Lichtquelle, welche strukturiertes IR-Licht emittiert und einer Infrarotkamera, welche das projizierte Lichtmuster aufnimmt. Abbildung 3.2a zeigt diese drei Komponenten am Beispiel einer Microsoft[®] Kinect[™] Kamera. Durch die feste Verbauung und die so bekannten extrinsischen und intrinsischen Kameraparameter dieser Komponenten kann das RGB-Farbbild zum Tiefenbild registriert werden. Ein *registriertes Tiefenbild* ist das Tiefenbild einer RGB-D-Kamera registriert zu ihrem RGB-Farbbild. Aus einem registrierten Tiefenbild und dem zugehörigen RGB-Farbbild kann eine Punkt-

wolke mit Farbinformation gewonnen werden.

Unter der Annahme eines Lochkameramodells wird ein Objektpunkt $(x, y, z) \in \mathbb{R}^3$ auf ein Bildpixel mit den Koordinaten $(u, v) \in \mathbb{N}_w \times \mathbb{N}_h$ nach dem folgenden Prinzip abgebildet:

$$s \begin{bmatrix} u & v & 1 \end{bmatrix}^\top = \mathbf{K} \mathbf{P} \begin{bmatrix} x & y & z & 1 \end{bmatrix}^\top \quad [3.8]$$

Der Skalierungsfaktor s wird für eine Darstellung als homogene Matrizen benötigt.

Die aus den extrinsischen Kameraparametern gebildete Projektionsmatrix \mathbf{P} bildet einen Punkt aus dem globalen Koordinatensystem der Szene in das lokale Kamerakoordinatensystem über eine Translation und eine Rotation ab.

$$\mathbf{P} := \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix}$$

Die aus den intrinsischen Kameraparametern gebildete Kameramatrix \mathbf{K} bildet über die Brennweite f und den Hauptpunkt \mathbf{c} einen Punkt des lokalen Kamerakoordinatensystems auf einen Bildpixel (u, v) ab.

$$\mathbf{K} := \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

Bedingt durch die reale Optik entsteht im aufgenommenen Kamerabild ein Verzeichnungseffekt. Eine Korrektur dieser Verzeichnung findet in der Regel über eine Polynomialfunktion vierten oder fünften Grads statt. Die Polynomkoeffizienten (Distortionskoeffizienten) sind Bestandteil der intrinsischen Kameraparameter.

Für ein zum RGB-Farbbild registriertes Tiefenbild gilt, dass Tiefenbildpixel und RGB-Bildpixel mit den gleichen Bildkoordinaten (u, v) vom gleichen 3-D-Objektpunkt (x, y, z) entstammen. Die durch die Tiefenbildkamera erfasste Distanz zum Szenepunkt macht diese Registrierung möglich (für ein RGB-Farbbild ist die Rekonstruktion des Szenepunkts (x, y, z) aus den Pixelkoordinaten (u, v) nicht möglich, da die Distanz zum Szenepunkt nicht bekannt ist).

Nach Formel 3.8 wird ein Szenepunkt (x_T, y_T, z_T) im Kamerakoordinatensystem der Tiefenbildkamera T auf ein Bildpixel (u_T, v_T) im Tiefenbild über

$$s \begin{bmatrix} u_T & v_T & 1 \end{bmatrix}^\top = \mathbf{K}_T \begin{bmatrix} x_T & y_T & z_T \end{bmatrix}^\top \quad [3.9]$$

abgebildet, da für die Projektionsmatrix \mathbf{P}_T für das Kamerakoordinatensystem $\mathbf{P}_T = \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix}$ gilt. Der unterstellte Index T verdeutlicht, dass es sich um die Kameramatrix bzw. die Bildpixel der Tiefenbildkamera handelt. Der unterstellte Index T kennzeichnet das Koordinatensystem der Tiefenbildkamera.

Für die Rekonstruktion eines 3-D-Objektpunkts (x_T, y_T, z_T) im Kamerakoordinatensystem der Tiefen-

bildkamera aus einem Tiefenbildpixel (u_T, v_T) gilt demnach

$$\mathbf{K}_T^{-1} \begin{bmatrix} u_T \\ v_T \\ 1 \end{bmatrix} s = \begin{bmatrix} x_T \\ y_T \\ z_T \end{bmatrix} \quad [3.10]$$

mit

$$\mathbf{K}_T^{-1} := \begin{bmatrix} \frac{1}{f_{T,x}} & 0 & -\frac{c_{T,x}}{f_{T,x}} \\ 0 & \frac{1}{f_{T,y}} & -\frac{c_{T,y}}{f_{T,y}} \\ 0 & 0 & 1 \end{bmatrix}$$

Der Objektpunkt (x_T, y_T, z_T) kann eindeutig bestimmt werden, da die Entfernung des Punkts zur Kamera $z_T = s = b(u_T, v_T)$ aus dem Tiefenbild bekannt ist.

Durch die bekannte Transformation ${}^T\mathbf{T}_F$ zwischen Tiefenbildkamerakoordinatensystem T und RGB-Kamerakoordinatensystem F lässt sich (x_T, y_T, z_T) in einen Punkt (x_F, y_F, z_F) lokal zum RGB-Kamerakoordinatensystem F überführen.

Die neuen Bildkoordinaten (u_F, v_F) für den Szenepunkt (x_F, y_F, z_F) werden mit den Kameraparametern der RGB-Kamera über Formel 3.8 bestimmt (der unterstellte Index F bzw. F verdeutlicht jetzt den Bezug zur RGB-Farbkamera bzw. deren Koordinatensystem).

$$s \begin{bmatrix} u_F & v_F & 1 \end{bmatrix}^\top = \mathbf{K}_F \begin{bmatrix} x_F & y_F & z_F \end{bmatrix}^\top$$

Für die Projektionsmatrix gilt in diesem Fall ebenfalls $\mathbf{P}_F = [1 \ 0]$, da sich der Szenepunkt wieder im kameraeigenen Koordinatensystem befindet.

Abbildung 3.2a zeigt die Hardwareintegration von CCD-Kamera, Infrarotlichtquelle und Infrarotkamera zu einem RGB-D-Kamerasystem an einer Microsoft[®] Kinect[™] Kamera. Da sich aus dem Tiefenbild die 3-D-Punkte der Szene rekonstruieren lassen, kann eine RGB-Punktwolke der Szene erstellt werden. Abbildung 3.2b zeigt dies. Die Abbildungen 3.2c bis 3.2e zeigen ein RGB-Farbbild der RGB-Kamera, ein Tiefenbild und ein zum RGB-Farbbild registriertes Tiefenbild der Kamera.

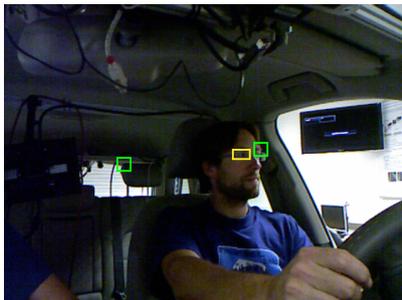
Nachteil Mindestabstand zur Kamera Da sich das RGB-D-Ausgabebild der Kamera über drei versetzt zueinander platzierte Hardwarekomponenten unterschiedlicher Emission- bzw. Sichtkegel gewonnen wird, entstehen im registrierten Tiefenbild undefinierte Bildbereiche. Der Abstand von IR-Lichtquelle und IR-Kamera führt für die Erfassung des Tiefenbilds zu einem erforderlichen Mindestabstand zwischen der Kamera und dem Kopf des Fahrers. Für die Microsoft[®] Kinect[™] Kamera, bei der die IR-Lichtquelle und die IR-Kamera 7,5 cm voneinander entfernt platziert sind, beträgt dieser Mindestabstand circa 0,5 Meter. In Abbildung 3.2d und 3.2e wird dieser Effekt im Tiefenbild an der nicht erfassten Hand des Fahrers deutlich.

Besonders für die Schätzung der Pupillenstellung wirkt sich dies nachteilig aus, da durch den Mindestab-

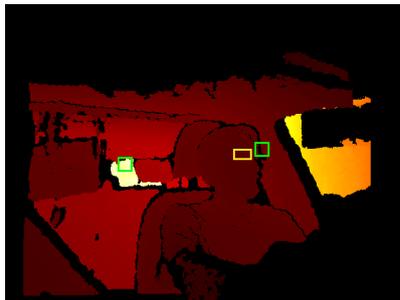


(a) Hardwareintegration von CCD-Kamera, Infrarot-Lichtquelle und Infrarot-Kamera zu einem RGB-D-Kamerasystem am Beispiel einer Microsoft® Kinect™ Kamera.
Quelle: [Hof10]

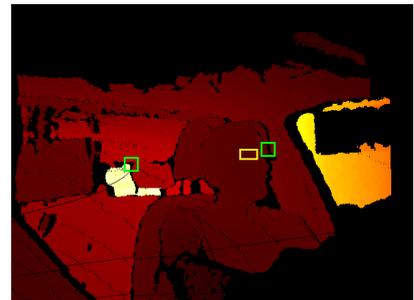
(b) Da sich aus dem Tiefenbild die 3-D-Punkte der Szene rekonstruieren lassen, kann eine RGB-Punktwolke der Szene erstellt werden.



(c) RGB-Farbbild



(d) Tiefenbild



(e) registriertes Tiefenbild

Abb. 3.2.: Über die bekannten intrinsischen und extrinsischen Kameraparameter der RGB-D-Kamera lässt sich das Tiefenbild zum RGB-Farbbild registrieren. Für ein zum RGB-Farbbild registriertes Tiefenbild gilt, dass Tiefenbildpixel und RGB-Bildpixel mit den gleichen Bildkoordinaten (u, v) vom gleichen 3-D-Objektpunkt (x, y, z) entstammen. Die grünen Rechtecke markieren dieselben Bildpixel in allen drei Bildern um die Registrierung zu verdeutlichen. Das gelbe Rechteck umfasst die Augenregion mit einer Größe von 24×12 Pixeln.

stand zwischen Kopf und Kamera die Region um die Augen nur wenige Bildpixel umfassen. In Abbildung 3.2 besteht die gelb markierte Augenregion aus 24×12 Pixeln. Durch Hardwaremodifikation kann der Mindestabstand der Kamera auf den Abstand vom Fahrer zum Kamerasystem optimiert werden.

Nachteil Gegenlicht und Störungen durch Spiegelungen Starkes Gegenlicht, welches entweder direkt in die Linse der Tiefenbildkamera oder auf die belichtete Szene fällt, verursacht ein lückenhaftes Tiefenbild. Eine technische Abhilfe würde das Bedampfen der Windschutzscheibe mit einem für das Spektrum der Infrarotkamera undurchlässigen Infrarotfilter schaffen. Abbildung 3.3 zeigt, wie durch wechselnde Lichtverhältnisse während der Fahrt das Tiefenbild verfälscht wird. Auch spiegelnde Objekte wie Sonnenbrillen verfälschen das Projektionsmuster und damit das Tiefenbild. Der entworfene Algorithmus schaltet bei einem fehlerhaften Tiefenbild auf eine Erkennung der Kopfpose über das RGB-Farbbild anhand von zur Laufzeit gelernten Gesichtsmerkmalen um (siehe Abschnitt 3.2.4).

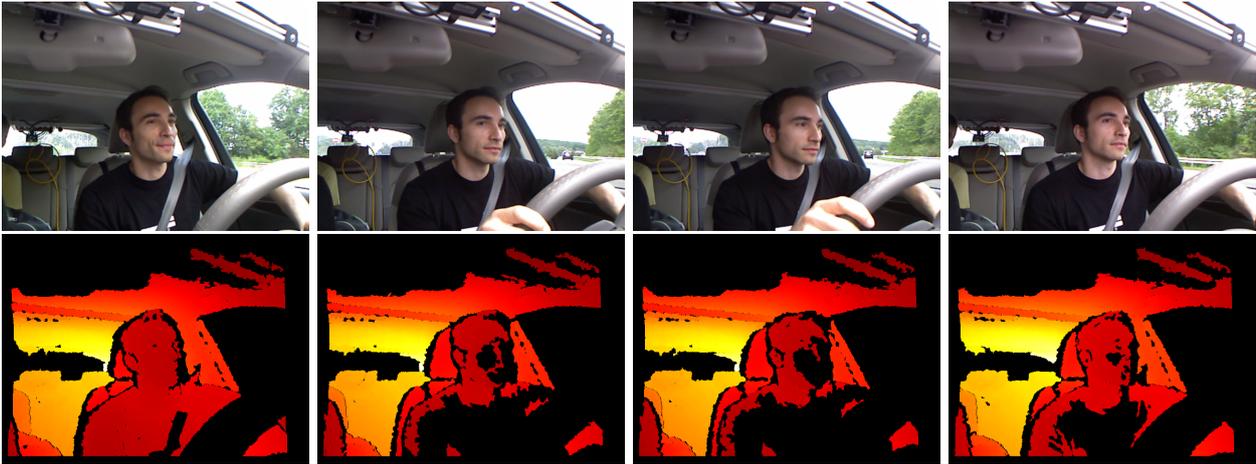


Abb. 3.3.: Das von der RGB-D-Kamera projizierte Lichtmuster ist zu schwach für die Rekonstruktion der Tiefendaten, wodurch ein lückenhaftes Tiefenbild entsteht. Physikalisch kann das Problem durch das Bedampfen der Windschutzscheibe durch einen IR-Filter gelöst werden. Der entworfene Algorithmus schaltet bei einem lückenhaften Tiefenbild auf eine Erkennung anhand von gelernten Gesichtsmerkmalen über das RGB-Farbbild um.

3.2.2. Bestimmung der Kopfpose über das Iterative Closest Point Verfahren

Bei intaktem Tiefenbild wird die Stellung des Kopfs über ein *Iterative-Closest-Point (ICP)* Verfahren bestimmt. Das *Iterative-Closest-Point-Verfahren (ICP)* ist in der Robotik weit verbreitet um eine geometrische Transformation zwischen Punktwolken verschiedener Sensoren oder verschiedener Messzeitpunkte zu finden [BM92; Zha92; Zha94].

In der vorliegenden Arbeit wird das ICP-Verfahren verwendet um eine Transformation zwischen den fahrerindividuellen 3-D-Gesichtsmasken des Fahrers und den Tiefendaten der RGB-D-Kamera zu bestimmen.

In einer der ersten Veröffentlichungen zum ICP-Algorithmus formuliert Zhang das mathematisch zu lösende Optimierungsproblem für das Finden einer bestmöglichen Transformation zwischen zwei Punktwolken als Minimierung von

$$F(\mathbf{R}, \mathbf{t}) = \sum_{i=1}^m \sum_{j=1}^{N_i} p_{j,j} d^2(\mathbf{R}x_{i,j} + \mathbf{t}, C'_k) + \sum_{k=1}^n \sum_{l=1}^{N_k} q_{k,l} d^2(\mathbf{R}^\top x'_{k,l} - \mathbf{R}^\top \mathbf{t}, C_i) \quad [\text{Formel 1 aus [Zha92]}]$$

bei gleichzeitiger Maximierung von

$$\sum_{i=1}^m \sum_{j=1}^{N_i} p_{j,j} + \sum_{k=1}^n \sum_{l=1}^{N_k} q_{k,l} \quad [\text{Formel 2 aus [Zha92]}]$$

mit

\mathbf{R} und \mathbf{t}	als Rotationsmatrix bzw. Translationsvektor zwischen den beiden Punktwolken C und C' . Mit $\mathbf{R} \in \mathbb{R}^{3 \times 3}$, $\mathbf{t} \in \mathbb{R}^3$
C und C'	als Punktwolken. Zhang formuliert eine Punktwolke als Menge von Kurven $C = \{C_i\} = \{C_0, \dots, C_m\}$ und $C' = \{C'_i\} = \{C'_0, \dots, C'_n\}$. Mit $C \in \mathbb{R}^{3 \times (mN_i)}$ bzw. $C' \in \mathbb{R}^{3 \times (nN_k)}$
\mathbf{x} und \mathbf{x}'	als Punkt der Punktwolke C bzw. C' mit der Indizierung $x_{i,j}$ als den i -ten Punkt der Kurve C_j . Mit $x_{i,j} \in \mathbb{R}^3$.
N_i und N_k	als die Anzahl der Punkte der Kurve C_i bzw. C'_k mit N_i bzw. $N_k \in \mathbb{N}^+$.
$\mathbf{p}_{i,j}$ und $\mathbf{q}_{k,l}$	als $\mathbf{p}_{i,j} := 1$ falls sich zum Punkt $x_{i,j}$ ein korrespondierender Punkt auf der Kurve C'_k findet und $\mathbf{p}_{i,j} := 0$ sonst ($\mathbf{q}_{k,l}$ definiert sich analog auf C'_k).
d^2	als Abstandsfunktion zweier Punkte im Raum \mathbb{R}^3 mit $d^2 : \mathbb{R}^3 \times \mathbb{R}^3 \rightarrow \mathbb{R}$.

Um die Formel von Zhang im Original wiederzugeben, weicht an dieser Stelle die Abkürzungskonvention von der in dieser Arbeit verwendeten Konvention ab.

Der linke Summand der Funktion $F(\mathbf{R}, \mathbf{t})$ summiert den Fehler der Korrespondenzpunkte (definiert durch $\mathbf{p}_{i,j}$) der Punktwolke C zu C' bei gegebener Transformation definiert durch \mathbf{R} und \mathbf{t} . Der rechte Summand summiert den Fehler der Korrespondenzpunkte (definiert durch $\mathbf{q}_{k,l}$) der Punktwolke C' zu C bei der inversen Transformation definiert durch \mathbf{R}^\top und $-\mathbf{t}$.

Unter der Annahme, dass eine Korrespondenz zwischen C und C' auch eine Korrespondenz zwischen C' und C ist kann mit $K \subseteq C \times C'$ eine Menge von Korrespondenzen zwischen den Punktwolken C und C' gebildet werden:

$$K := \{(\mathbf{p}_{i,j}, \mathbf{q}_{k,l}) \mid \mathbf{p}_{i,j} \mathbf{q}_{k,l} = 1\} \subseteq C \times C' \quad [3.11]$$

Nimmt man weiterhin eine kommutative Abstandsfunktion δ an, so kann das von Zhang formulierte Minimierungsproblem zur Minimierung von

$$F'(\mathbf{R}, \mathbf{t}) = \sum_{(\mathbf{p}_i, \mathbf{q}_i) \in K} \delta(\mathbf{p}_i - [\mathbf{R}, \mathbf{t}]\mathbf{q}_i) \quad [3.12]$$

vereinfacht werden.

Ein Iterationsschritt des ICP-Algorithmus beinhaltet damit das Finden einer neuen (besseren) Transformation $[\mathbf{R}', \mathbf{t}']$ basierend auf einer Menge von Korrespondenzen K gebildet über die aktuelle Transformation $[\mathbf{R}, \mathbf{t}]$.

$$[\mathbf{R}', \mathbf{t}'] \leftarrow \underset{\mathbf{R}, \mathbf{t}}{\operatorname{argmin}} F'(\mathbf{R}, \mathbf{t}) = \sum_{(\mathbf{p}_i, \mathbf{q}_i) \in K} \delta(\mathbf{p}_i - [\mathbf{R}, \mathbf{t}]\mathbf{q}_i) \quad [3.13]$$

Das von Zhang zur Formel 3.12 vereinfachte Minimierungsproblem macht die Kernkomponenten des ICP-Algorithmus deutlich:

- Finden von Korrespondenzen
- Wahl einer Fehlerfunktion

- Minimierung der Fehlerfunktion

Weiterhin benötigt der ICP-Algorithmus eine initiale Transformationsschätzung zum Bilden der ersten Menge von Korrespondenzpunktpaaren. Da jede dieser Kernkomponenten individuell und an die aktuelle Problemstellung angepasst werden kann gibt es viele ICP-Varianten. Rusinkiewicz und Levoy vergleichen und bewerten verschiedene ICP-Varianten auf Basis der genannten Kernkomponenten [RL01].

In dieser Arbeit werden zwei typische Schritte des ICP-Verfahrens auf die Problemstellung „Blickrichtungserkennung im Fahrzeug“ optimiert:

- Eine schnelle Korrespondenzsuche im registrierten Tiefenbild ermöglicht das Finden von Korrespondenzen in $\mathcal{O}(N_Q)$, mit N_Q als der Anzahl der Punkte in der zu registrierenden Gesichtsmaske.
- Ein entsprechend des Iterationsfortschritts wechselnde Fehlerfunktion garantiert eine akkurate und schnelle Terminierung des ICP-Algorithmus.

Die initiale Transformationsschätzung wird in dieser Arbeit entweder aus den letzten beobachteten Kopfpositionen des Fahrers, der fahrerindividuellen Standard-Kopfposition bei gegebener Identifikation oder einer aus dem RGB-Farbbild geschätzten Kopfpose bestimmt.

Finden von Korrespondenzen

Beim Finden von Korrespondenzen werden in den beiden Punktwolken Korrespondenzpunktpaare gesucht, welche den gleichen Szenepunkt abbilden. Ein Korrespondenzpunktpaar $(\mathbf{p}, \mathbf{q}) \in K \subseteq P \times Q$ besteht aus jeweils einem Punkt der zu registrierenden Punktwolken P und Q (vgl. Formel 3.11). Perfekte Korrespondenzen haben bei korrekter Registrierung einen Abstand von $\delta(\mathbf{p}, \mathbf{q}) = 0$.

In der Standardvariante werden die Korrespondenzen über eine *Nearest-Neighbour (NN)* Suche mit euklidischem Abstandsmaß ermittelt. Als Abstandsmaß kann aber auch die Farbe des Punkts [GRB94] oder die Regressionsebene der umliegenden Punkte einfließen [Pul99]. Da die Wahl des euklidischen Abstandsmaßes nicht zwingend ist haben einige Autoren angemerkt, dass *Iterative Corresponding Point* eine bessere Bezeichnung für das Verfahren sei [RL01; RBZ12].

In vielen Implementierungen wird ein Schwellwert für die maximale Distanz zwischen zwei korrespondierenden Punkten definiert. Einige ICP-Varianten führen eine Qualität (oder Gewicht) einer Korrespondenz ein, welches ein Maß dafür liefert inwiefern der Punkt \mathbf{p} der Punktwolke P auch tatsächlich dem Punkt \mathbf{q} der Punktwolke Q entspricht.

Ist eine der beiden Punktwolken (zum Beispiel Q) in einer Baumstruktur organisiert, kann eine Komplexität von $\mathcal{O}(N_p \log(N_Q))$ für die Korrespondenzsuche erreicht werden [Ben75; BM92]. Die Baumstruktur muss dabei eine *Nearest-Neighbour (NN)* Suche in $\mathcal{O}(\log(N_Q))$ erlauben. Diese Funktionalität bietet beispielsweise ein ausbalancierter *k-D-Baum*. Im schlechtesten Fall liegt die Korrespondenzsuche jedoch in der Komplexitätsklasse $\mathcal{O}(N_p N_Q)$, da der Abstand von jedem Punkt $\mathbf{p} \in P$ zu jedem Punkt $\mathbf{q} \in Q$ berechnet werden muss.

Nutzt man die Tatsache, dass ein Tiefenbild mit bekannten intrinsischen Kameraparametern vorliegt, kann jeder 3-D-Punkt aus der Gesichtsmaske über die intrinsischen Kameraparameter mit Aufwand $\mathcal{O}(1)$ in das Tiefenbild projiziert werden (siehe [BS99; BL95]). Die Korrespondenzsuche kann so auf die

Komplexitätsklasse $\mathcal{O}(N_Q)$, mit Q als die Punktwolke der zu registrierenden Gesichtsmaske, reduziert werden.

Mit einem Punkt $\mathbf{q} \in Q$ auf der Gesichtsmaske kann der korrespondierende Punkt $\mathbf{p} \in P$ im Tiefenbild der RGB-D-Kamera direkt berechnet werden. Mit Formel 3.9 kann für den Punkt \mathbf{q} der Gesichtsmaske direkt die Tiefenbildkoordinate (u, v) bestimmt werden. Mit Formel 3.10 kann aus dem Tiefenbildpixel der Korrespondenzpunkt $\mathbf{p} \in P$ in der Punktwolke des Sensors bestimmt werden. Abbildung 3.4 verdeutlicht die Berechnung der Korrespondenzen über die intrinsischen Kameraparameter der Tiefenbildkamera.

Da keinerlei Suche durchgeführt wird, verursachen schon kleinere Ungenauigkeiten der intrinsischen

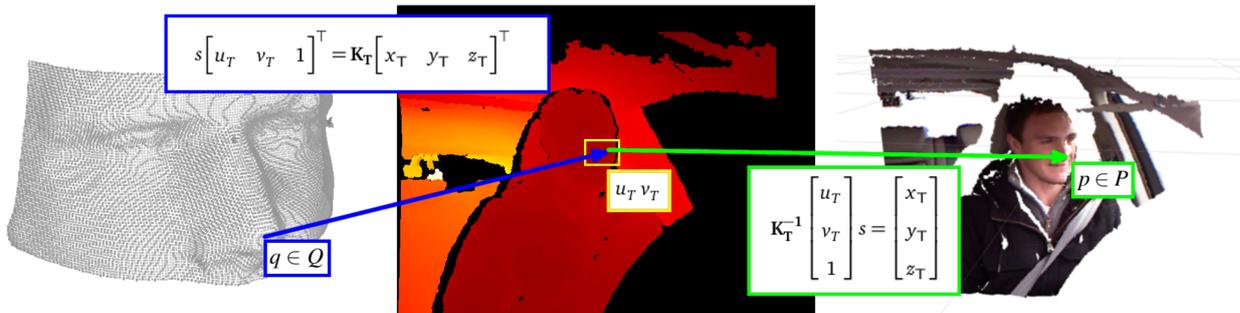


Abb. 3.4.: Für die Korrespondenzsuche kann jeder 3-D-Punkt der Gesichtsmaske über die intrinsischen Kameraparameter mit Aufwand $\mathcal{O}(1)$ in das Tiefenbild projiziert werden (blauer Pfad - Formel 3.9). Der korrespondierende Punkt aus der Sensorpunktwolke ergibt sich aus der Transformation des Bildpunkts (u_T, v_T) in das 3-D-Koordinatensystem (grüner Pfad - Formel 3.10).

Ungenauigkeiten der intrinsischen Kameraparameter führen zu ungenauen Korrespondenzen. Um die Genauigkeit zu verbessern wird in einem Bereich um den Projektionspunkt nach dem am naheliegendsten Punkt gesucht (gelbes Rechteck). Die Korrespondenzsuche kann so auf die Komplexitätsklasse $\mathcal{O}(N_Q)$ reduziert werden.

Kameraparameter eine falsche Zuordnung von Korrespondenzpunktepaaren. Um die Genauigkeit zu verbessern wird deshalb im Nachgang in einem Bereich um den Projektionspunkt nach dem naheliegendsten Punkt gesucht. Theoretisch definiert sich der Suchbereich um den Projektionspunkt als eine Ellipse, abhängig von den 3-D-Koordinaten des Projektionspunkts. In der Praxis wird ein rechteckiger Bereich um den Projektionspunkt abgesucht, was die Implementierung vereinfacht und die Genauigkeit vernachlässigbar beeinträchtigt. Mit einem Suchbereich fester Größe bleibt die Korrespondenzsuche in der Komplexitätsklasse $\mathcal{O}(N_Q)$.

Auswahl und Minimierung der Fehlerfunktion

Im Iterationsschritt des ICP-Verfahrens wird eine Fehlerfunktion, welche über die gefundenen Korrespondenzpunkte definiert ist, minimiert. Die Minimierung der Fehlerfunktion hinsichtlich der freien Parameter \mathbf{R} und \mathbf{t} bestimmt die optimale Transformation der gegebenen Menge der Korrespondenzpunkte K (vgl. Formel 3.13). Je nach gewählter Fehlerfunktion kann diese Minimierung in geschlossener Form formuliert werden. Ist dies nicht der Fall werden Näherungslösungen oder numerische Approximationsverfahren zur Minimierung der Fehlerfunktion gewählt.

Die in dieser Arbeit verwendete ICP-Methode benutzt eine Punkt-zu-Punkt-Fehlerfunktion für die grobe Registrierung und eine Punkt-zu-Ebene-Fehlerfunktion für die feine Registrierung. Abbildung 3.5 skizziert die unterschiedlichen Abstandsmaße der Fehlerfunktionen. Beide Fehlerfunktionen haben Vor- und

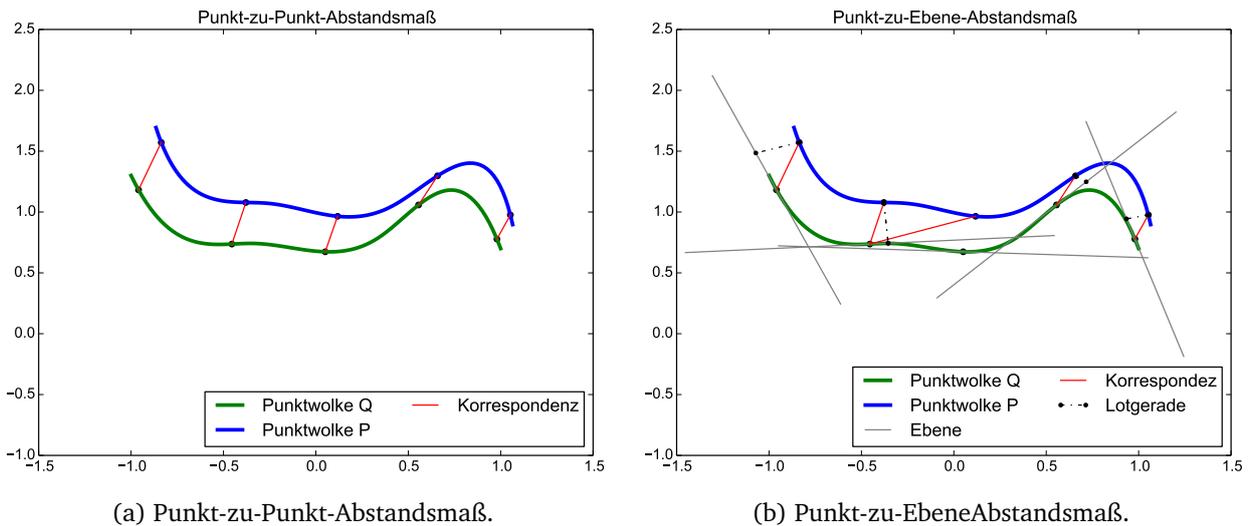


Abb. 3.5.: Vergleich von Punkt-zu-Punkt- und Punkt-zu-Ebene-Abstandsmaß. Basiert das Fehlermaß auf einem Punkt-zu-Punkt-Abstand, kann die Konvergenz des ICP-Schritts nachgewiesen werden [BM92]. Die Registrierung wird allerdings eine gewisse Qualität nicht überschreiten. Basiert das Fehlermaß auf einem Punkt-zu-Ebene-Abstand, ist die Konvergenz des ICP-Schritts nicht garantiert. Die Registrierung der Punktwolken über ein Punkt-zu-Ebene-Abstandsmaß liefert eine genauere Transformationen bei bereits gut registrierten Punktwolken.

Nachteile:

Punkt-zu-Punkt: Die Punkt-zu-Punkt-Fehlerfunktion, dargestellt in Gleichung 3.14, eignet sich sehr gut für noch schlecht registrierte Punktwolken und nähert sich in den ersten Iterationsschritten schnell dem lokalen Minima an. Basiert der ICP-Algorithmus auf einer Punkt-zu-Punkt-Fehlerfunktion kann seine Konvergenz nachgewiesen werden [BM92]. Die Registrierung wird allerdings eine gewisse Qualität nicht überschreiten.

$$F_p(\mathbf{R}, \mathbf{t}) = \sum_{(\mathbf{p}_i, \mathbf{q}_i) \in K} \|\mathbf{q}_i - \mathbf{T} \cdot \mathbf{p}_i\|^2 \quad [3.14]$$

Minimierung: Die Minimierung der Punkt-zu-Punkt-Fehlerfunktion kann in geschlossener Form über die Singulärwertzerlegung (SWZ oder SVD für *Singular Value Decomposition*) berechnet werden. Dazu werden die Punktwolken P und Q in mittelwertfreie Punktwolken P' und Q' transformiert.

$$P' = \{\mathbf{p}_i - \mu_P\} = \{\mathbf{p}'_i\} \quad \text{mit} \quad \mu_P = \frac{1}{N_P} \sum_{i=0}^{N_P} \mathbf{p}_i$$

$$Q' = \{\mathbf{q}_i - \mu_Q\} = \{\mathbf{q}'_i\} \quad \text{mit} \quad \mu_Q = \frac{1}{N_Q} \sum_{i=0}^{N_Q} \mathbf{q}_i$$

Danach wird für die mittelwertfreie Punktwolken P' und Q' das äußere Produkt (dyadisches

Produkt) W gebildet

$$W = \sum_{i=0}^{N_Q} \mathbf{p}_i \mathbf{q}_i^\top$$

welches über die Singulärwertzerlegung in die Matrizen

$$W = U \Sigma V^\top = U \begin{bmatrix} \sigma_0 & 0 & 0 \\ 0 & \sigma_1 & 0 \\ 0 & 0 & \sigma_2 \end{bmatrix} V^\top$$

zerlegt werden kann. Über diese Zerlegung lässt sich die beste Transformation $[\mathbf{R}, \mathbf{t}]$, sowie der Fehler dieser Transformation auf den gegebenen Korrespondenzen $F(\mathbf{R}, \mathbf{t})$ direkt bestimmen:

$$\begin{aligned} \mathbf{R} &= UV^\top \\ \mathbf{t} &= \mu_P - \mathbf{R} \mu_Q \\ F(\mathbf{R}, \mathbf{t}) &= \sum_{i=0}^{N_P} (\|\mathbf{p}'_i\|^2 + \|\mathbf{q}'_i\|^2) - 2(\sigma_0 + \sigma_1 + \sigma_2) \end{aligned}$$

Punkt-zu-Ebene: Die Punkt-zu-Ebene-Fehlerfunktion, dargestellt in Gleichung 3.15, liefert genauere Transformationen für bereits gut registrierte Punktwolken. Sind die Punktwolken jedoch noch stark voneinander entfernt, ist die Konvergenz des Iterationsschrittes nicht sicher.

$$F_E(\mathbf{R}, \mathbf{t}) = \sum_{(\mathbf{p}_i, \mathbf{q}_i) \in K} ((\mathbf{q}_i - \mathbf{T} \cdot \mathbf{p}_i) \cdot \mathbf{n}_i)^2 \quad [3.15]$$

Sind die registrierenden Punktwolken sehr flach, kommt es zu einem „Übereinandergleiten“ [CM91].

Minimierung: Zur Minimierung der Punkt-zu-Ebene-Fehlerfunktion werden Vereinfachungen getroffen (eine gängige Approximation ist $\cos(\alpha) \approx \alpha$ für kleine α , welche beispielsweise von Low verwendet wird [Low04]) oder numerische Optimierungsalgorithmen wie das *Levenberg-Marquardt* Verfahren verwendet [Fit03]. Eine Lösung in geschlossener Form ist (zumindest dem Autor) nicht bekannt.

Die in dieser Arbeit verwendete Minimierung basiert auf der Umwandlung der Fehlerfunktion in eine *Mahalanobis-Distanz*

$$F'_E(\mathbf{R}, \mathbf{t}) = \sum_{(\mathbf{p}_i, \mathbf{q}_i) \in K} (\mathbf{q}_i - (\mathbf{R}\mathbf{p}_i + \mathbf{t}))^\top \mathbf{n} \mathbf{n}^\top (\mathbf{q}_i - (\mathbf{R}\mathbf{p}_i + \mathbf{t}))$$

aus der zur Minimierung ein *Lagrange Optimierungsproblem* mit Nebenbedingungen formuliert werden kann [Ygu08].

$$L(F'_E(\mathbf{R}, \mathbf{t})) = F'_E(\mathbf{R}, \mathbf{t}) + \text{tr}(\mathbf{L}[\mathbf{R}^\top \mathbf{R} - \mathbf{I}]) + \lambda(\det(\mathbf{R}) - 1)$$

Die Minimierung des *Lagrange Optimierungsproblem* folgt numerisch über das *Gauß-Newton-Minimierungsverfahren*. Die mathematische Herleitung findet sich in [Ygu08].

Um eine schnelle und trotzdem genaue Konvergenz zu ermöglichen werden in dieser Arbeit die ersten Iterationsschritte der Kopfposesbestimmung mit Punkt-zu-Punkt als Abstandsmaß der Fehlerfunktion verwendet. Haben die Punktwolken nur noch wenig Abstand zueinander (sie sind gut registriert), werden die übrigen Iterationsschritte mit dem Punkt-zu-Ebene-Abstandsmaß durchgeführt. Dies resultiert in einer schnellen und gleichzeitig genauen Kopfposesbestimmung.

Initialisierung der Kopfpose

Der in Abschnitt 3.2.2 vorgestellte ICP-Algorithmus benötigt eine initiale Transformationsschätzung $[\mathbf{R}_0, \mathbf{t}_0]$ zum Bilden der ersten Menge von Korrespondenzpunktpaaren. Eine initiale Transformation kann über das RGB-Farbbild mit Hilfe der Gesichtserkennung über das *Viola/Jones-Verfahren* geschätzt werden. Das *Viola/Jones-Verfahren* liefert für ein RGB-Farbbild eine rechteckige Region in Bildkoordinaten, welches das Gesicht des Fahrers auf dem RGB-Farbbild umschließt. Mit der sehr vereinfachten Annahme, dass sich im Mittelpunkt dieser Region die Nasenspitze befindet, kann mit bekannter Nasenposition eine initiale Translation der Gesichtsmaske $\mathbf{t} \in \mathbb{R}^3$ über das Tiefenbild und den intrinsischen Kameraparametern bestimmt werden. Geht man weiter vereinfacht davon aus, dass der Fahrer gerade in die Kamera blickt ($\mathbf{R} = \mathbf{I}$), kann eine grobe, initiale Transformation bestimmt werden.

$$\mathbf{T}_0 = \begin{bmatrix} \mathbf{I} & \mathbf{t} \\ 0 & 1 \end{bmatrix}$$

Die Bestimmung der initialen Transformation über das RGB-Farbbild ist im Vergleich zur Initialisierung anhand früheren Kopfstellungen jedoch langsamer und ungenauer. Zur Steigerung der Performanz werden deshalb zunächst frühere Kopfstellungen und geringe Abweichungen derer als initiale Transformationen benutzt. Die initiale Transformationsschätzung folgt somit dem folgenden Prinzip:

Ist die letzte Kopfpose des Fahrers bekannt, wird ...

- die **letzte Kopfpose** des Fahrers als initiale Transformation verwendet. Kann so die Kopfpose nicht bestimmt werden, werden aus der ...
- letzten bekannten Kopfpose neue initiale Transformationen mit jeweils **leicht veränderter Rotation** um Roll, Pitch und Yaw generiert. Kann so die Kopfpose nicht bestimmt werden, wird aus ...
- dem **Median** der letzten bekannten Kopfposes eine neue initiale Transformation und bei Misserfolg neue Transformationen mit jeweils leicht veränderter Rotation um die Roll-, Nick- und Gierachse generiert. Kann so die Kopfpose nicht bestimmt werden, wird ...
- unter der Voraussetzung einer erfolgreichen **Fahreridentifikation, die fahrertypische Kopfstellung** und bei Misserfolg jeweils leicht veränderter Rotation um die Roll-, Nick- und Gierachse als initiale Transformationen verwendet. Konnte der Fahrer nicht identifiziert werden oder liegt keine Information über frühere Kopfstellungen vor oder kann die Kopfstellung so nicht bestimmt werden,

wird ...

- die initiale Transformation **über das RGB-Farbbild** mit Hilfe des **Viola/Jones-Verfahrens** bestimmt. Konnte die Kopfstellung so nicht bestimmt werden, schlägt die Kopfposenerkennung über den ICP-Algorithmus für dieses RGB-D-Bild fehl. Es wird eine Schätzung der Kopfstellung über das RGB-Farbbild initiiert (siehe Abschnitt 3.2.4).

3.2.3. Gesichtsmasken Management

Die in Abschnitt 3.2.2 beschriebene ICP-Methode bestimmt die Kopfpose über die Berechnung einer Transformation zwischen den fahrerindividuellen 3-D-Gesichtsmasken und den Tiefendaten der RGB-D-Kamera. Eine Gesichtsmaske modelliert die rigiden Teile des Gesichts von der Stirn bis zur Oberlippe in Form einer Punktwolke mit zugehörigem Normalenvektor pro Punkt P_{\perp} (siehe Definition 3.4). Abbildung 3.6b zeigt die linke Gesichtshälfte als Punktwolke mit zugehörigen Normalenvektoren.

Insgesamt werden für jeden Fahrer zwei rigide Gesichtsmasken erstellt (siehe Abbildung 3.6a). Die Aufteilung in zwei Gesichtshälften erwirkt einen größeren Erkennungsbereich um die Z-Achse (Gierbewegung). Dreht der Fahrer beim Einparken oder Wenden den Kopf stark um die Gierachse, ist immer noch eine Gesichtshälfte der Kamera zugewandt, für die sich die Transformation über den ICP-Algorithmus bestimmen lässt. Die Transformation mit dem niedrigsten ICP-Fehler wird als Kopfposenschätzung des Fahrers verwendet.

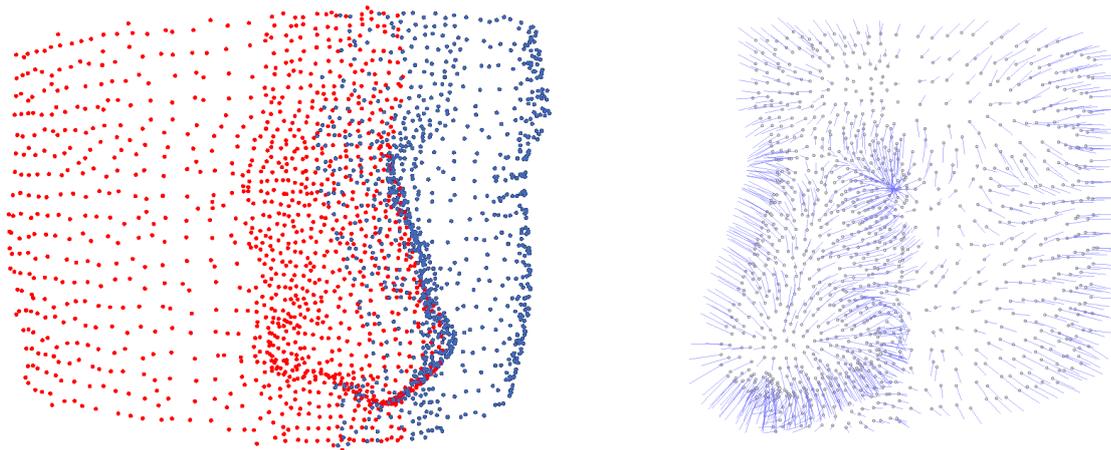
Die Gesichtsmasken sind individuell auf den Fahrer angepasst und werden nach erfolgreicher Identifikation verfügbar.

Identifikation des Fahrers

Die Identifikation des Fahrers bringt bei der Anpassung des Fahrzeugs und der im Fahrzeug integrierten Algorithmik an den individuellen Fahrer erhebliche Vorteile mit sich. Verhaltensmodelle des Fahrers können geladen werden und zum Beispiel (Reichweiten-)Berechnungen aussagekräftiger und akkurater kalkuliert werden [BNZ14] (siehe Abschnitt 5.3 im Evaluationskapitel auf Seite 130). Für die Blickrichtungserkennung wird die Identifizierung des Fahrers genutzt um eine personalisierte Gesichtsmaske zu laden.

Die Identifikation des Fahrers ist rein formal eine Funktion, welche anhand von Merkmalen aus einer Menge von potentiellen Fahrern den *richtigen* Fahrer auswählt. Die Identifikation kann demnach als *multinomiales Klassifikationsproblem* verstanden werden, für das es im Bereich des *Maschinellen Lernens* der Informatik schon zahlreiche Lösungskonzepte gibt. Mit einer begrenzten Menge an potentiellen Fahrern, kann eines der nach heutigem Stand der Forschung leistungsfähigsten Klassifikationsverfahren für die Fahreridentifikation gewählt werden: die *Support-Vektor-Maschine (SVM)*, oft auch als *Stützvektormaschine* oder *Kernel-Maschine* bezeichnet. Sie teilt in ihrer Mehrklassenvariante Objekte, repräsentiert durch Merkmalsvektoren, einer von N verschiedenen Klassen zu. Nienhäuser gibt einen Überblick über verschiedene SVM-Varianten und formuliert die mathematischen Zusammenhänge [Nie14].

Die Anwendung der SVM zur Fahreridentifikation fordert das Bilden eines möglichst aussagekräftigen Merkmalsvektors. Als Merkmalsvektor wird das Seitenprofil des Kopfs (Silhouette) verwendet, welche die



(a) Die rigiden Teile des Gesichts werden in einer linken und rechten Gesichtsmaske modelliert. Die Aufteilung der Gesichtsmasken trägt zur Vergrößerung des Erkennungsbereichs bei, da nicht immer das ganze Gesicht von der Kamera erfasst werden kann.

(b) Die Gesichtsmaske der linken Gesichtshälfte mit den entsprechenden Normalenvektoren der Punkte. Die gewölbten Teile der Maske (Nasen- und Augenbereich) weisen eine höhere Punktdichte auf, was die Genauigkeit des ICP-Ergebnisses verbessert.

Abb. 3.6.: Zur Berechnung der relativen Transformation des Kopfs zur Kamera wird sowohl die linke, als auch die rechte Gesichtsmaske des Fahrers auf die Punktwolke der RGB-D-Kamera eingepasst. Aus der Transformation mit dem niedrigsten ICP-Fehler bestimmt sich abschließend zur Kopfpose des Fahrers. Die Aufteilung in zwei Gesichtshälften dient zur Vergrößerung des Erkennungsbereichs. Eine höhere Punktdichte in den stark gekrümmten Teilen des Gesichts (Augen- und Nasenregion) verbessert die Genauigkeit der ICP-Schätzung. Die Gesichtsmasken sind individuell auf den Fahrer angepasst.

Gesichtsform in eindeutiger Weise beschreibt. Abbildung 3.7 zeigt die Silhouette des Fahrers, extrahiert aus der Punktwolke der RGB-D-Kamera. Die Silhouette des Fahrers kann bei bekannter Kopfstellung aus

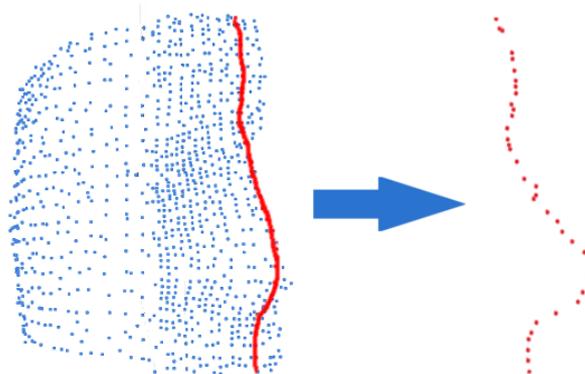


Abb. 3.7.: Als Merkmalsvektor zur Identifikation des Fahrers dient die Silhouette des Gesichts. Sie beschreibt den Fahrer in eindeutiger Art und Weise.

der Sensorpunktwolke extrahiert werden.

Für die Identifikation des Fahrers entsteht hier ein Dilemma: Für das Erstellen der Silhouette wird die erkannte Kopfpose vorausgesetzt. Für das Erkennen der Kopfpose wird eine personalisierte Gesichtsmaske vorausgesetzt. In der Praxis kann mit einer Standardgesichtsmaske jedoch eine hinreichend genaue Kopfstellung geschätzt werden, um die Silhouette des Fahrers aus der Sensorpunktwolke zu extrahieren.

Mit einer durch die ICP-Methode bestimmten Kopfstellung (siehe Abschnitt 3.2.2) ist die Transformation ${}^H\mathbf{T}_F$ zwischen dem Koordinatensystem der RGB-D-Kamera und der Gesichtsmaske bekannt, und die 3-D-Punktwolke P_F der RGB-D-Kamera lassen sich in das Koordinatensystem der Gesichtsmaske H transformieren.

$$P_H = \{{}^H\mathbf{T}_F \mathbf{p}_F \mid \mathbf{p}_F \in P_F\}$$

Mit $\epsilon \approx 0$ als der Breite der Silhouette in y -Richtung und z_{\min} und z_{\max} als die Abmaße der Silhouette in z -Richtung, lässt sich aus der Punktwolke P_H eine Silhouettenpunktwolke extrahieren.

$$S = \{\mathbf{p} \in P_H \mid (z_{\min} < p_z \leq z_{\max}) \wedge (|p_y| < \epsilon)\}$$

Zur Stabilisierung der Silhouettenpunktwolke werden in der Praxis die letzten N Silhouettenpunktwolken akkumuliert. Um einen Merkmalsvektor \mathbf{s} der festen Größe K zu erhalten, wird die Silhouette S in Untermengen S_i partitioniert

$$i \in [0 : K[\\ S_i := \left\{ \mathbf{p} \in S \mid \frac{i(z_{\max} - z_{\min})}{K} < p_z - z_{\min} \leq \frac{(i+1)(z_{\max} - z_{\min})}{K} \right\} \subseteq S$$

und für jede nicht leere Partition der Median gebildet. Hat eine Partition keine Elemente, so wird ein Wert \mathbf{s} aus den naheliegendsten Medianpunkten $\tilde{\mathbf{s}}_a$ und $\tilde{\mathbf{s}}_b$ interpoliert.

$$\tilde{\mathbf{s}}_i := \underset{z}{\text{median}}(\{S_i\}) \quad , S_i \neq \emptyset \\ \hat{\mathbf{s}}_i := \tilde{\mathbf{s}}_a + \frac{(\tilde{\mathbf{s}}_b - \tilde{\mathbf{s}}_a)(i - a)}{b - a}, S_i = \emptyset$$

Über die Interpolation werden Punktlücken in der Silhouette aufgefüllt. In der Praxis bricht die Identifikation für dieses Tiefenbild ab, wenn die Punktlücken zu groß sind ($a \ll b$) und damit eine sinnvolle Interpolation nicht gewährleistet ist. Löcher am Rand werden dem naheliegendsten, existierenden Medianpunkt gleichgesetzt.

Über das Zusammenführen von $\tilde{\mathbf{s}}_i$ und $\hat{\mathbf{s}}_i$ lässt sich der Merkmalsvektor \mathbf{s} , als Eingabevektor für die SVM bilden:

$$s_i := \begin{cases} \tilde{\mathbf{s}}_i & , S_i \neq \emptyset \\ \hat{\mathbf{s}}_i & , S_i = \emptyset \end{cases}$$

3.2.4. Schätzung der Kopfpose bei überbelichtetem oder fehlendem Tiefenbild

Die Fremdlichtkompensation ist eine der wesentlichen Herausforderungen, die die Domäne Automobil, mit den während der Fahrt ständig wechselnden Lichtverhältnissen, grundlegend von Labor- oder Innenraumumgebungen unterscheidet.

Starke Sonneneinstrahlung oder Spiegelungen führen zu Störungen im Tiefenbild. Als Folge lässt sich die Kopfpose über die ICP-Methode nicht mehr bestimmen (siehe Abbildung 3.3). Der Ausfall des

Tiefenbilds kommt in der Praxis recht selten vor (siehe Abschnitt 5.1.4, Seite 113). Zusätzlich lässt sich der Ausfall durch das Bedampfen der Windschutzscheibe mit einem Spektralfilter im Infrarotbereich der RGB-D-Kamera minimieren. Mit dem Anspruch einer robusten und lichtwechselbeständigen Blickrichtungserkennung darf der Ausfall des Tiefenbilds jedoch nicht zum Fehlerfall führen und muss algorithmisch kompensiert werden.

Lässt sich die Kopfpose mit der ICP-Methode nicht schätzen, wird sie ansichtsbasiert aus dem RGB-Farbbild über *Random Regression Forests* (Regressions-schätzung) bestimmt. In einer ansichtsbasierten Schätzung der Kopfpose sind keine bildverarbeitenden Schritte notwendig um Gesichtsmerkmale im RGB-Farbbild zu extrahieren. Der Gesichtsausschnitt des Fahrers wird direkt und mit nur wenig Vorverarbeitung als Merkmalsvektor gelernt (übliche Vorverarbeitungsschritte sind eine Konvertierung in Graustufen, ein Histogrammausgleich und eine Skalierung). In der Kopfposenschätzung sind ansichtsbasierte Verfahren deshalb besonders schnell. Sie unterliegen jedoch zwei grundlegenden Nachteilen: Zum einen wird eine große Trainingsmenge annotierter Messdaten zum Training der Regressionsmaschine benötigt in denen alle Kopfposen des Fahrers in (möglichst) ausgewogener Art und Weise unter verschiedenen Lichtverhältnissen vertreten sein müssen. Zum anderen ist die gelernte Ansicht einer Kopfpose vorwiegend von den (Gesichts-)Merkmalen des eintrainierten Fahrers geprägt. Eine Generalisierung des Gelernten auf andere Fahrer führt zu Ungenauigkeiten in der Kopfposenbestimmung.

Im umgesetzten Verfahren wird die Messdatenbank während der Kopfposenschätzung bei intaktem Tiefenbild aufgebaut und damit nebenläufig ein *Random Regression Forest* trainiert. Die Nachteile des ansichtsbasierten Verfahrens werden so kompensiert:

- Während der Kopfposenschätzung bei intaktem Tiefenbild liegt ein RGB-Farbbild mit bekannter Kopfposition vor (Label). Während des Fahrens lässt sich damit nebenläufig eine umfangreiche und ausgewogene Messdatenbank aufbauen.
- Durch die Generierung der Lerndaten zur Laufzeit ist die Messdatenbank auf den momentan fahrenden Fahrer und damit auf dessen Gesichtsmerkmale optimiert.

Über diese Art der Messdatenbildung ist keine Messung mit lückenhaftem Tiefenbild in der Lernmenge enthalten. Eine starke Generalisierung vom RGB-Farbbild mit zugehörigem intaktem Tiefenbild auf ein RGB-Farbbild mit zugehörigem lückenhaftem Tiefenbild wird vorausgesetzt und bestätigt sich in der Praxis. Der Abschnitt „*Random Regression Forests*“ führt kurz in die Grundlagen der Regressions-schätzung über *Random Regression Forests* ein. Im Abschnitt „Nebenläufiges Trainieren der *Random Regression Forests* über eine zur Laufzeit aufgebaute Trainingsdatenbank“ wird das Bilden des Merkmalsvektors und das Trainieren der *Random Regression Forests* über eine zur Laufzeit aufgebaute Trainingsdatenbank beschrieben. Der Abschnitt „Schätzung der Kopfpose über *Random Regression Forests*“ beschreibt die Schätzung der Kopfpose über *Random Regression Forests*.

Random Regression Forests

Random Forests erweitern das Prinzip der *Entscheidungsbäume (Decision Trees)* auf einen *Wald*, bestehend aus zufällig aufgebauten *Entscheidungsbäumen*. *Entscheidungsbäume* klassifizieren Messdaten anhand von Entscheidungsregeln. Stellt man die Entscheidungsregeln in einem Graphen dar, erhält man einen

geordneten, gerichteten Baum, welcher in den inneren Knoten die Entscheidungsregeln und in den Blättern das Klassifikationslabel trägt.

Die Entscheidungsregeln des Baums sind nach ihrer Trenngüte (*Information Gain*) $IG(S)$ geordnet. Umso besser eine Regel die Messdaten S in die verschiedenen Klassen $c \in C$ trennt, umso höher steht sie im Baum. Die Trenngüte einer Regel lässt sich über die durch die Trennung entstandene Reduzierung der Entropie $H(S)$ definieren.

$$H(S) = \sum_{c \in C} \frac{|S_c|}{|S|} \log\left(\frac{|S_c|}{|S|}\right) \quad [3.16]$$

$$IG(S) = H(S) - \sum_{c \in C} H(S_c) \quad [3.17]$$

Mit Entscheidungsbäumen kann schon mit wenigen Trainingsdaten eine aussagekräftige und schnelle Klassifikation in der Komplexitätsklasse $\mathcal{O}(\log(n))$ erzielt werden. Widersprüchliche oder verrauschte Messdaten oder solche mit sehr vielen Attributen führen zu allerdings zu Problemen: Der ursprünglich von Quinlan vorgestellte Entscheidungsbaum *ID3* [Qui86] kann nicht mit widersprüchlich annotierten Trainingsdaten umgehen, da gleiche Entscheidungsregeln auf nur ein Blatt und damit gezwungenermaßen auf eine Klassenzugehörigkeit abbilden. In verrauschten Trainingsdaten entsteht für jeden Ausreißer ein eigener Zweig. Der Entscheidungsbaum erreicht eine große Höhe und spezialisiert sich dabei stark auf die Trainingsdaten. Kann in den Messdaten ein Attribut sehr viele Attributwerte annehmen, führt dies zu sehr breit gefächerten Bäumen. Für jedes Attribut wird ein Sonderfall generiert.

In beiden Fällen sinkt die Generalisierungsfähigkeit des Baums und die Daten werden auswendig gelernt (*Overfitting*). Eine der bekanntesten Implementierungen, der *C4.5* von Quinlan, behilft sich mit dem Zurechtschneiden (*Prunen*) des Entscheidungsbaums und einer zusätzlichen Bevorzugung von Entscheidungsregeln welche die Daten in nur wenige Gruppen trennt [Qui93].

Für die Implementierung der *Random Forests* gibt es zahlreiche Varianten, die sich im Typ des zu Grunde liegenden Entscheidungsbaums, dem Umgang mit *Overfitting* eines Einzelbaums oder in Beschränkungen der Baumtiefe unterscheiden. Die Randomisierung der Bäume fließt über zwei Aspekte ein:

- Jeder Baum wird aus einer zufällig gewählten Untermenge der Trainingsdaten erstellt.
- Die Entscheidungsregeln eines einzelnen Baums werden aus einer zufällig zusammengestellten Untermenge aller Entscheidungsregeln gezogen.

Die Klassifikation geschieht durch eine Mehrheitsentscheidung nach der Auswertung aller Bäume.

Gegenüber der Klassifikation über eine *Support Vektor Maschine* (ein oft gewähltes Standard-Klassifikationsverfahren) haben *Random Forests* den Vorteil einer sehr kurzen Trainingsphase, die darüber hinaus parallelisierbar ist. Da in dieser Arbeit die Bäume zur Kopfposenbestimmung zur Laufzeit gelernt werden erweist sich dies als entscheidender Vorteil.

Random Regression Forests lernen im Gegensatz zu *Random Forests* einen kontinuierlichen Werteverlauf (Regression) und keine Klassifikation. In den Blätterknoten eines *Random Regression Forests* stehen Regressionswerte oder Verteilungen an Stelle von Klassen. Die in dieser Arbeit umgesetzte Form der *Random Regression Forests* hält Normalverteilungen in ihren Blätterknoten. Die Varianz der Normalverteilung kann im Kehrwert als Gewichtung des Regressionswerts interpretiert werden kann.

Nebenläufiges Trainieren der *Random Regression Forests* über eine zur Laufzeit aufgebaute Trainingsdatenbank

Aufbau einer ausgewogenen Messdatenbank Liefert die Blickrichtungsschätzung über das Tiefenbild eine Kopfposenschätzung, wird das zugehörige RGB-Farbbild mit der Kopfposenschätzung als Label in einer Lerndatenbank abgelegt. Die Lerndatenbank baut so zur Laufzeit sukzessive auf. Da der Fahrer während einer normalen Autofahrt die meiste Zeit geradeaus schaut, sind die Lerndaten in Bezug auf die Kopfposen sehr inhomogen. Ein Trainieren der *Random Regression Forests* auf einer inhomogenen Trainingsmenge würde einen Bias zu nach vorne gerichteten Kopfstellungen zur Folge haben. Um homogene Verteilung der Kopfposen in der Trainingsmenge zu erreichen, werden die Lerndaten in einem zweidimensionalen Gitter entsprechend ihres Nick- und Gierwinkels abgelegt. Eine Zellengröße von 10×10 Grad und eine Speicherung der letzten 1000 Bilder pro Zelle hat sich als praxistauglich erwiesen.

Vorverarbeitung Um von der Kopfpose unabhängige Einflussfaktoren wie die Belichtung oder wechselnden Hintergrund zu reduzieren und gleichzeitig markante Gesichtsmerkmale, welche auf die Kopfpose hinweisen, hervorzuheben, wird das RGB-Farbbild folgenden Vorverarbeitungsschritten unterzogen: Zunächst wird das RGB-Farbbild auf einen rechteckigen Bereich um die bekannte Mitte des Kopfs (in der Trainingsphase ist die Kopfpose bekannt) beschränkt. Der entstandene Bildausschnitt wird auf eine feste Größe skaliert um auf einer einheitlichen Bildgröße zu operieren. Zur Reduzierung des Datenvolumens wird das Bild in Graustufen umgewandelt. Alle Eigenschaften des Gesichts sind auch im Graustufenbild erkennbar. Um Helligkeitsunterschiede auszugleichen wird der Bildausschnitt abschließend normalisiert. Danach werden über den Gabor- und den Sobelfilter weitere Merkmalsbilder erzeugt [BB15]. Diese Texturfilter heben die Kanten des Graustufenbilds hervor und sind gleichzeitig weitestgehend unabhängig von Lichtverhältnissen. Nach der Erzeugung besteht das Bild aus folgenden Kanälen:

- ein Graustufenbildkanal (normalisiert, skaliert)
- 35 Gaborfilterkanäle (in 7 Richtungen mit je 5 verschiedenen Wellenlängen)
- zwei Sobelfilterkanäle (in zwei Richtungen)

Trainieren der *Random Regression Forests* Die zum Schätzen der Kopfpose verwendeten *Random Regression Forests* bestehen aus einer Menge von Regressionsbäumen. Jeder dieser Regressionsbäume besteht aus inneren Knoten, welche gemäß einer Trennregel Daten ihrem linken oder ihrem rechten Teilbaum zuordnen. In den Blättern der Regressionsbäume stehen die Regressionswerte, repräsentiert durch eine Normalverteilung.

Das Festlegen der Trennregeln und die Bestimmung der Normalverteilung in den Blättern basierend auf Trainingsdaten wird als „Trainieren“ oder „Einlernen“ der *Random Regression Forests* bezeichnet.

Zum Trainieren des *Random Regression Forests* wird für jeden Regressionsbaum durch randomisiertes Ziehen ohne Zurücklegen eine zufällige Untermenge der 38-kanaligen Bildausschnitte gewählt.

Aus jedem dieser Bildausschnitte werden wiederum zufällig 25 Bildausschnitte (*Patches*) ausgewählt, welche, zusammen mit einem Label, dem Regressionsbaum als Trainingsdaten dienen. Ein Trainingsdatum

umfasst somit den Patch, sowie als Label die Nasenposition relativ zum Patch in 2-D-Pixelkoordinaten (u_n, v_n) . Außerdem die Kopfrichtung θ als 3-D-Richtungsvektor.

Nach dem Generieren der Trainingsdaten wird eine Trennregel für den Wurzelknoten des Baums gesucht, welche die Trainingsdaten entsprechend ihres Informationsgewinns (siehe Formel 3.17) trennt. Für die Berechnung des Informationsgewinns ist ein Maß für die Entropie erforderlich. Mit der Annahme, dass es keine Korrelation zwischen der Nasenposition relativ zum zufällig gewählten Patch und dem Drehwinkel des Kopfs gibt, lässt sich die Entropie aus der Varianz der relativen Nasenposition (u_n, v_n) bzw. der Kopfdrehung θ ableiten:

$$E = -\log(\text{Var}(u_n, v_n) + \text{Var}(\theta))$$

Als Trennregel wird in einem Patch für einen Bildkanal a die Intensitätsdifferenz I_a zweier Rechtecke R_1 und R_2 bestimmt und durch einen Schwellwert τ getrennt.

$$\frac{1}{|R_1|} \sum_{q \in R_1} I_a(q) - \frac{1}{|R_2|} \sum_{q \in R_2} I_a(q) < \tau$$

Eine Trennregel besteht damit aus dem 4-er Tupel (R_1, R_2, a, τ) .

Das Generieren *einer* Trennregel erfolgt in zwei Stufen: In einem ersten Schritt werden die Parameter (R_1, R_2, a) zufällig gewählt, wodurch der Wertebereich von τ eingegrenzt werden kann. In einem zweiten Schritt wird der Schwellwert τ bestimmt, welcher die Daten gemäß des Informationsgewinns maximal trennt.

Für die Suche nach der besten Trennregel für den Wurzelknoten wird eine Menge von Trennregeln generiert und, wiederum gemäß des Informationsgewinns, die beste Trennregel gewählt. Anschließend werden die Trainingsdaten gemäß der gefundenen Trennregel auf Unterbäume verteilt und gleichermaßen Trennregeln für die entstandenen Unterbäume bestimmt. Ist eine maximale Baumtiefe erreicht terminiert das Verfahren und die Normalverteilungen für die Blätterknoten werden errechnet. Das Trainieren gleicht dem im Stand der Technik (Abschnitt 2.1.3) vorgestellten Verfahren von [FGG11] mit dem Unterschied, dass der hier vorgestellte Ansatz auf RGB-Farbbildern und nicht auf Tiefenbildern basiert.

Schätzung der Kopfpose über *Random Regression Forests*

Zur Schätzung der Kopfpose wird, analog zur Trainingsphase, der Gesichtsausschnitt des Bilds gemäß der zuvor beschriebenen Vorverarbeitung aufbereitet. Die Bestimmung des Gesichtsausschnitts basiert auf der letzten bekannten Kopfposition, da die aktuelle Kopfposition durch das gestörte Tiefenbild nicht ermittelbar ist. Im Gegensatz zum Training erfolgt die Erzeugung der Patches nicht mehr zufällig, da dies keinen Mehrwert schaffen würde. Stattdessen werden die Patches einem festen Raster entnommen um eine möglichst vollständige Abdeckung des Gesichts zu erzielen. Die Dichte des Rasters kann je nach Optimierungsziel angepasst werden: Eine zu dichte Rasterung erzeugt viele Patches und steigert die Genauigkeit. Eine grobe Rasterung steigert die Geschwindigkeit.

Über die Trennregeln werden die Patches einem Blatt *eines* Regressionsbaums (einen Regressionswert)

zugeordnet. Abbildung 3.8 zeigt wie ein Patch auf einen Regressionswert abgebildet wird. Durch diese

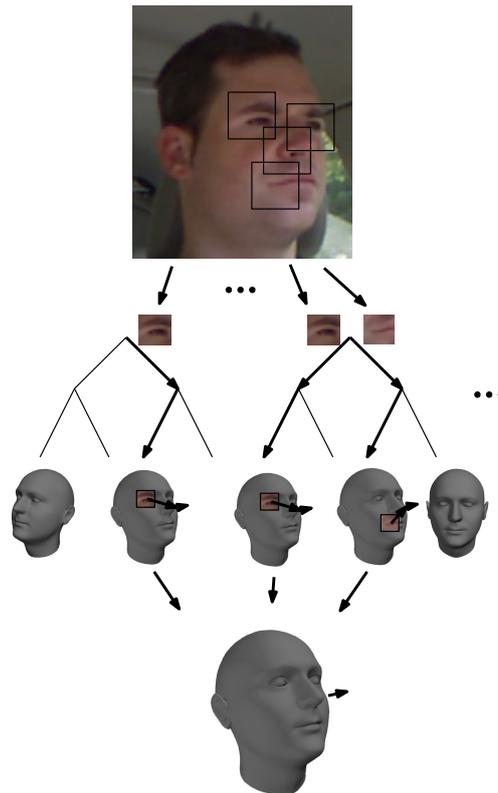


Abb. 3.8.: Aus dem Gesichtsausschnitt des Eingabebilds werden Patches entnommen, aus denen über die *Random Regression Forests* Hypothesen über die Kopfposition abgeleitet werden. Jeder Baum liefert zu einem Patch eine Hypothese. Aus den vielen Hypothesen wird über den *Mean-Shift-Algorithmus* ein Datenmittelpunkt bestimmt, welcher das Ergebnis der Regressionsschätzung durch die *Random Regression Forests* darstellt. Aus der Regressionsschätzung wird anschließend die Kopfpose in 6-D-Koordinaten berechnet.

Zuordnung entsteht eine Hypothese in Form einer Normalverteilung über die 2-D-Nasenposition und der Kopffrotation. Die Varianz der Normalverteilung kann als Gewichtung der Hypothese interpretiert werden. Abbildung 3.9 zeigt durch die *Random Regression Forests* generierte Hypothesen der Nasenposition eingezeichnet in das als Eingabe zur Verfügung stehende Graustufenbild.



Abb. 3.9.: Über die *Random Regression Forests* entstehen Hypothesen über die relative Position der Nasenspitze im Bildausschnitt. Über den *Mean-Shift-Algorithmus* wird aus den vielen Hypothesen ein Datenmittelpunkt bestimmt, der die geschätzte Position der Nasenspitze im Bildausschnitt repräsentiert. Im rechten Bild sind zusätzlich die Iterationsschritte des *Mean-Shift-Algorithmus* dargestellt.

Da die Nasenposition gegenüber der Kopffrotation das zuverlässigere Merkmal bildet, wird in einem

ersten Schritt die endgültige Nasenposition mittels *Mean-Shift-Algorithmus* aus den Hypothesen der *Random Regression Forests* bestimmt. Ausgehend von einer initialen Schätzung bestimmt der *Mean-Shift-Algorithmus* iterativ den gewichteten Mittelpunkt aller Nasenpositionen. Über eine Kernelfunktion K werden Datenpunkte nahe der Initialisierung \hat{x} stärker gewichtet als weiter entfernte Datenpunkte.

$$K(x_i - \hat{x}) := e^{-c\|x_i - \hat{x}\|^2}$$

Der Faktor c bestimmt dabei die Gewichtung der Entfernung. Zur Initialisierung des *Mean-Shift-Algorithmus* wird die letzte bestimmte Nasenposition verwendet.

Im Iterationsschritt wird durch den Kernel und die Varianz ω_i der Hypothesen der Nasenpositionen ein neuer Datenmittelpunkt bestimmt. Das Verfahren terminiert, wenn sich der Datenmittelpunkt \hat{x} nur noch geringfügig ändert oder einer bestimmte Anzahl Iterationsschritte erreicht ist.

Nach der Bestimmung der Nasenposition wird die Kopffrotation bestimmt. Dabei werden nur Stimmen in Betracht gezogen die innerhalb eines bestimmten Radius um die Nasenposition liegen. Diese Beschränkung verbessert die Genauigkeit der Rotationsbestimmung des Kopfs deutlich, da Stimmen, die schon die Position der Nase falsch bestimmt haben auch mit hoher Wahrscheinlichkeit keine korrekte Aussage über die Orientierung des Kopfs machen. Aus den verbleibenden Stimmen wird die Kopfpose über ein gewichtetes Mittel bestimmt.

3.2.5. Bestimmung der Pupillenstellung

Für die Schätzung der Blickrichtung muss neben der Kopfstellung auch die Stellung der Augen berücksichtigt werden. Blicke in den Rückspiegel oder den Tachometer werden vom Fahrer oft nur durch Augenbewegungen durchgeführt. Abbildung 3.10 zeigt einen Blick des Fahrers in den Rückspiegel. Die Kopfpose verändert sich beim Blick kaum.

Da die Zentren der Augäpfel fest im menschlichen Kopf verankert sind, lässt sich die 3-D-Positionen

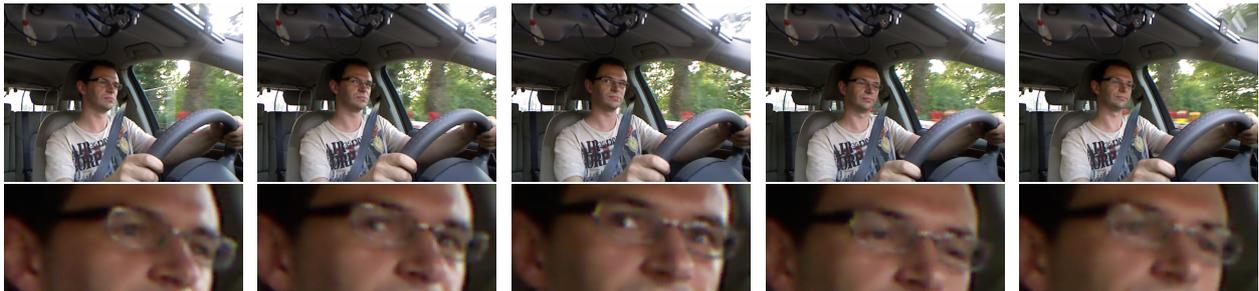


Abb. 3.10.: Beim Blick in den Rückspiegel ändert sich die Kopfpose des Fahrers kaum. Der hier dargestellte Blick in den Spiegel dauerte 0,5 Sekunden und erstreckt sich in der Aufnahme über 15 Bilder.

der Augäpfel im Koordinatensystem der personalisierten Gesichtsmaske des Fahrers definieren. Mit einer exakten Kopfposenbestimmung kennt man somit auch unmittelbar die exakten 3-D-Positionen der Zentren der Augäpfel. Mit einem weiteren 3-D-Punkt $b \in \mathbb{R}^3$, dem Mittelpunkt der Pupille auf der Hornhaut des Auges, ist die Blickrichtung des Fahrers eindeutig über eine Gerade bestimmt (siehe Abbildung 3.11a). Der Pupillenmittelpunkt wird über das RGB-Farbbild des Fahrers extrahiert, indem



- (a) Modell des Auges. Die Blickrichtung (violetter Pfeil) ergibt sich aus einer Geraden zwischen dem Zentrum $o \in \mathbb{R}^3$ des Auges und dem Mittelpunkt der Pupille $b \in \mathbb{R}^3$ auf der Hornhaut des Auges. *Quelle: Nachbearbeitet von [Clk]*
- (b) Rechter Augenextraktor dargestellt auf der Gesichtsmaske des Fahrers. Das Bild der Augenregion (Abbildung 3.11c) wird durch die Umwandlung jedes Punkts des Augenextraktors in ein RGB-Pixel gewonnen (Orthogonalprojektion).
- (c) Aus der Projektion der 3-D-Punkte des Augenextraktors gewonnenes Bild der Augenregionen. Die Kopfpose des Fahrers ist kompensiert. Das Auge stellt sich dar, als würde der Fahrer frontal in die Kamera blicken. Der rot markierte Pixel zeigt den gefundenen Mittelpunkt der Pupille.

Abb. 3.11.: Die Blickrichtung des Fahrers kann über zwei 3-D-Punkte bestimmt werden: Dem Zentrum des Augapfels und dem Mittelpunkt der Pupille auf der Hornhaut des Auges. Durch die bekannte Kopfpose des Fahrers ist implizit auch das Zentrum des Augapfels bekannt. Der Mittelpunkt der Pupille wird über das RGB-Farbbild der Kamera bestimmt. Dazu werden die Augenregionen orthogonal auf eine Bildebene projiziert.

in der Augenregion nach dem Mittelpunkt der dunkelsten Scheibe (Pupille) gesucht wird. Die vorliegende Implementierung nimmt als RGB-Farbbild das Bild der RGB-D-Kamera. Setzt man eine bekannte extrinsische und intrinsische Kalibrierung voraus, kann die Pupillenstellung aber auch aus einer zweiten, hochauflösenden CCD-Kamera wie sie in Abschnitt 3.3 verwendet wird, gewonnen werden.

Die perspektivische Projektion der Kamera, die Wölbungen des Augapfels und die veränderliche Kopfpose führen zu einer perspektivischen Verzerrung der Pupille im RGB-Farbbild, welche sich durch einfache Bildverarbeitungsoperationen nicht korrigieren lassen. Durch die bekannte Oberflächenstruktur der Augenregion, welche in der Gesichtsmaske des Fahrers verzeichnet ist, kann eine orthogonale Projektion der Augenregionen auf eine virtuelles Augenbild erfolgen. Die orthogonale Projektion erfolgt über einen Augenextraktor bestehend aus 3-D-Punkten. Die 3-D-Punkte des Augenextraktors ergeben sich aus einer orthogonalen Projektion von in der (Y, Z) -Ebene äquidistanten Punkten auf die Gesichtsmaske. Als Mitte des Augenextraktors in der (Y, Z) -Ebene wird das (y, z) -Zentrum des Augapfels gewählt. Abbildung 3.11b zeigt in rot die 3-D-Punkte des Augenextraktors, der in der Frontalansicht rechteckig wäre. Über Formel 3.9 (Seite 45), welche Szenepunkte auf Bildpixel abbildet, kann aus den 3-D-Punkten des Augenextraktors ein rechteckiges, orthogonales Augenbild generiert werden. Abbildung 3.11c zeigt ein über den Augenextraktor generiertes, orthogonales Augenbild. Der Einfluss der Kopfpose, der perspektivischen Verzerrung der Kamera und der Augenwölbung ist auf dem orthogonalen Augenbild eliminiert. Das Auge stellt sich im orthogonalen Augenbild dar, als würde es von einer frontal auf den Fahrer gerichteten Kamera erfasst.

Bei starken Kopfdrehungen zeichnen sich nicht immer beide Augenbereiche auf dem Kamerabild ab. Das orthogonale Augenbilds wird deshalb nur für Kopfstellungen generiert in denen das Auge der Kamera zugewandt ist.

Die Iris des menschlichen Auges hat in etwa einen Durchmesser von 12 mm. Unter dieser Annahme lässt sich ein Kreisfilter erstellen, der besonders stark auf dunkle, kreisförmige Regionen anspricht. Um den Mittelpunkt der Pupille zu finden wird das generierte Bild der Augenregion in Graustufen umgewandelt und mit einem Kreisfilter gefaltet. Als Mittelpunkt der Pupille wird das Minimum der Faltung gewählt. Mit $a \in \mathbf{B}$ als das Graustufenbild der Augenregion und f als dem generierten Kreisfilter, ergibt sich dies mathematisch zu

$$b := a * f$$

$$(u, v) := \underset{(u, v)}{\operatorname{arg\,min}}(b_{(u, v)})$$

Aus den Bildkoordinaten des Mittelpunkts lässt sich über den Augenextraktor direkt der Mittelpunkt der Pupille auf der Hornhaut des Auges als 3-D-Punkt $\mathbf{b} \in \mathbb{R}^3$ bestimmen. Zusammen mit dem bekannten Zentrum des Augapfels $\mathbf{o} \in \mathbb{R}^3$ kann die Blickrichtung eines Auges als Geradengleichung $g_\lambda : \mathbb{R} \rightarrow \mathbb{R}^3, \lambda \in \{l, r\}$ bestimmt werden:

$$g_\lambda(s) = \mathbf{o}_\lambda + s \|\mathbf{b}_\lambda - \mathbf{o}_\lambda\| \Rightarrow \mathbf{T}_\lambda$$

Die resultierende Blickrichtung wird als Mittel beider Blickrichtungsgeraden bestimmt:

$$g(s) = \frac{\mathbf{o}_l + \mathbf{o}_r}{2} + s \|(\mathbf{b}_l + \mathbf{b}_r) - (\mathbf{o}_l + \mathbf{o}_r)\| \Rightarrow \mathbf{T}$$

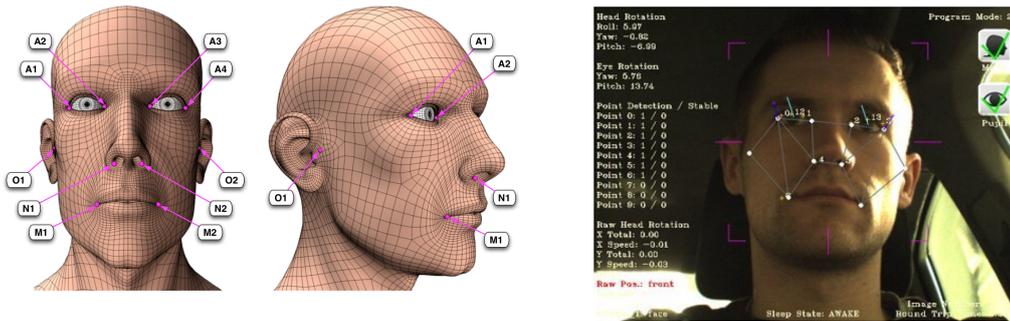
Der Ursprung der Blickrichtung wird als die mittlere Position zwischen beiden Augen angenommen, einem Punkt der einige Millimeter unter der Hautoberfläche in der Region des Nasenbeins liegt. Kann nur für ein Auge der Blickrichtungsvektor durch die Pupillenstellung bestimmt werden, ergibt sich der Richtungsvektor der Blickrichtung aus dem Richtungsvektor dieses Auges. Kann für kein Auge der Blickrichtungsvektor bestimmt werden, wird der Richtungsvektor der Kopfpose als Blickrichtungsvektor übernommen.

3.3. Blickrichtungserkennung über CCD-Kameras

Zur Bestimmung der Blickrichtung mittels einer hochauflösenden CCD-Kamera werden Gesichtsmerkmale im Kamerabild gesucht und auf ein 3-D-Kopfmodell des Fahrers abgebildet. Das 3-D-Kopfmodell beschreibt die dreidimensionale Anordnung der Merkmalspunkte und muss im Vorfeld erstellt und auf die jeweilige Person angepasst werden. Das 3-D-Kopfmodell besteht aus insgesamt acht 3-D-Punkten, welche Gesichtsmerkmalen zugeordnet sind.

Abschnitt 3.3.1 beschreibt die Lokalisierung von Gesichtsmerkmalen im RGB-Farbbild.

Durch die Abbildung gefundener Gesichtsmerkmale auf ein 3-D-Kopfmodell lässt sich die Kopfpose des Fahrers über den *POSIT-Algorithmus* [DD95] bestimmen. Abschnitt 3.3.2 geht auf die Bestimmung der Kopfpose über den *POSIT-Algorithmus* ein. In einem der Kopfposenerkennung nachgelagerten Schritt werden die Positionen der Pupillen gesucht und daraus die eigentliche Blickrichtung bestimmt. Die Ermittlung der Augenstellung wird über ein Augenmodell, vergleichbar zu dem im Abschnitt 3.2.5



(a) Gesichtsmerkmale, welche im CCD-Kamerabild lokalisiert werden.
Quelle: [PBZ10]

(b) Erkennung der Gesichtsmerkmale im CCD-Kamerabild.
Quelle: [PBZ10]

Abb. 3.12.: Um die Kopfstellung zu ermitteln werden aus dem CCD-Kamerabild insgesamt acht Gesichtsmerkmale lokalisiert.

beschriebenen Augenmodell (siehe Abbildung 3.11a), ermittelt. Auf die Bestimmung der Augenstellung wird deshalb nicht weiter eingegangen.

3.3.1. Lokalisierung markanter Gesichtsmerkmale

Um die Kopfstellung zu ermitteln werden im Kamerabild insgesamt acht Gesichtsmerkmale lokalisiert. Abbildung 3.12 zeigt die zu erkennenden Gesichtsmerkmale am Kopfmodell (Abbildung 3.12a) und stellt sie deren Erkennung im CCD-Kamerabild gegenüber (Abbildung 3.12b). Für eine vertiefende Beschreibung der Lokalisierung der Gesichtsmerkmale sei auf die Diplomarbeit von Pfeffer et al. [PBZ10] verwiesen. Zusammenfassend werden die einzelnen Gesichtsmerkmale wie folgt lokalisiert:

Pupille: Mit einer kreisförmigen Suchmaske wird zunächst nach dem dunkelsten Bereich in der Augenregion gesucht. Anschließend wird die Position verfeinert, indem der kreisförmige Übergang zwischen Iris und weißer Lederhaut lokalisiert wird. Der Mittelpunkt dieses Kreises kann schließlich als Pupillenmittelpunkt angenommen werden. Abbildung 3.13 zeigt die Lokalisierung des Pupillenmittelpunkts, sowie ein Diagramm der Intensitätsunterschiede, berechnet auf den Kreislinien um den markierten Mittelpunkt.

Augenwinkel: Aufbauend auf der erkannten Pupillenposition wird in einem kleinen Suchbereich eine Eckendetektion durchgeführt, wobei der verwendete Filter speziell an die Form der Augenwinkel angepasst wird. Abbildung 3.14 zeigt dieses Vorgehen schematisch.

Nasenlöcher: Die beiden Nasenlöcher werden über ein iteratives Schwellenwertverfahren lokalisiert. In einem Segmentierungsschritt wird dabei der Schwellwert zur Filterung der Nasenregion so lange erhöht, bis sich zwei dunkle Bereiche abzeichnen. In den segmentierten Nasenlochflächen wird der von der Nasenspitze, detektiert über die *Viola-Jones-Methode*, am weitesten entfernte Punkt als Lokalisierung gewählt. Abbildung 3.15 zeigt diesen Ablauf.

Mundwinkel: Mit Hilfe der durch die Lippen bedingten horizontalen Kanten in der Mundregion und der dunklen Linie zwischen Ober- und Unterlippe können die Positionen der Mundwinkel über

einen Kantenfilter mit anschließender zeilen- bzw. spaltenweiser Summierung ermittelt werden. Abbildung 3.16 zeigt die Lokalisierung der Mundwinkel an Hand eines Graustufenbilds der Lippenregion.

Ohren: Anhand der spezifischen Konturen der Ohrmuschel und der Farbgebung in und um diesen Bereich wird ein Referenzpunkt im Bereich der Ohren ausfindig gemacht. Abbildung 3.17 zeigt die Kantenbilder der Ohren und den Lokalisationspunkt.

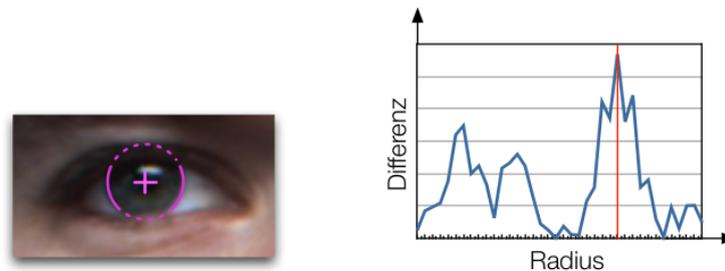


Abb. 3.13.: Lokalisierung der Iris über eine kreisförmige Suchmaske (links) und Intensitätsunterschiede, berechnet auf den Kreislinien um den markierten Mittelpunkt. Quelle: [PBZ10]

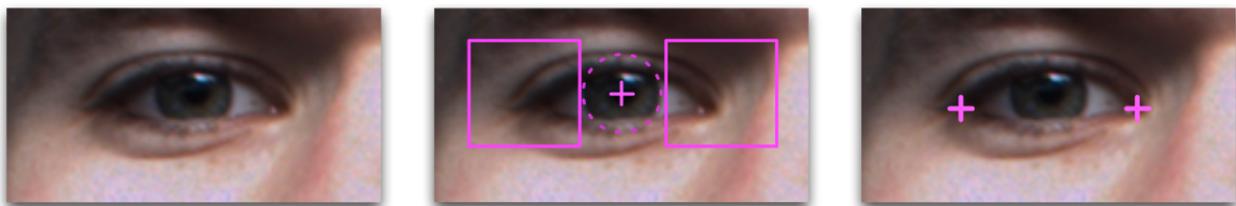


Abb. 3.14.: Aufbauend auf der erkannten Pupillenposition werden die Augenwinkel über eine Kombination aus Kanten- und Eckenfilter lokalisiert. Quelle: [PBZ10]

3.3.2. Schätzung der Kopfpose über die erkannten Gesichtsmerkmale

Der *POSIT-Algorithmus* (*Pose from Orthography and Scaling with Iteration*) schätzt die Pose eines Objekts im dreidimensionalen Raum über gegebene Bildpunkte mit zugeordneten 3-D-Positionen [DD95]. Abbildung 3.18 stellt den *POSIT-Algorithmus*, wie er in dieser Arbeit Verwendung findet, schematisch dar. Voraussetzung für die Anwendung des *POSIT-Methoden* ist, dass mindestens vier 2-D-Merkmalpositionen bekannt sind deren 3-D-Korrespondenzpunkte nicht auf einer geometrischen Ebene liegen. Der Algorithmus besteht aus zwei Schritten:

Zunächst wird durch das Lösen eines linearen Gleichungssystems die Rotationsmatrix und der Translationsvektor der Kopfstellung bestimmt. In diesem Schritt wird die perspektivische Projektion der Kamera durch eine orthographische Projektion angenähert. Weiter wird vereinfachend angenommen, dass alle gefundenen Merkmalspunkte in einer Ebene liegen, die zur Bildebene parallel ist. So kann eine Größenänderung des 3-D-Modells auf eine Skalierung abgebildet werden, die vom Abstand der Kamera bestimmt ist.

Im zweiten Schritt wird diese Skalierung ermittelt und auf den Abstand zur Kamera abgebildet.

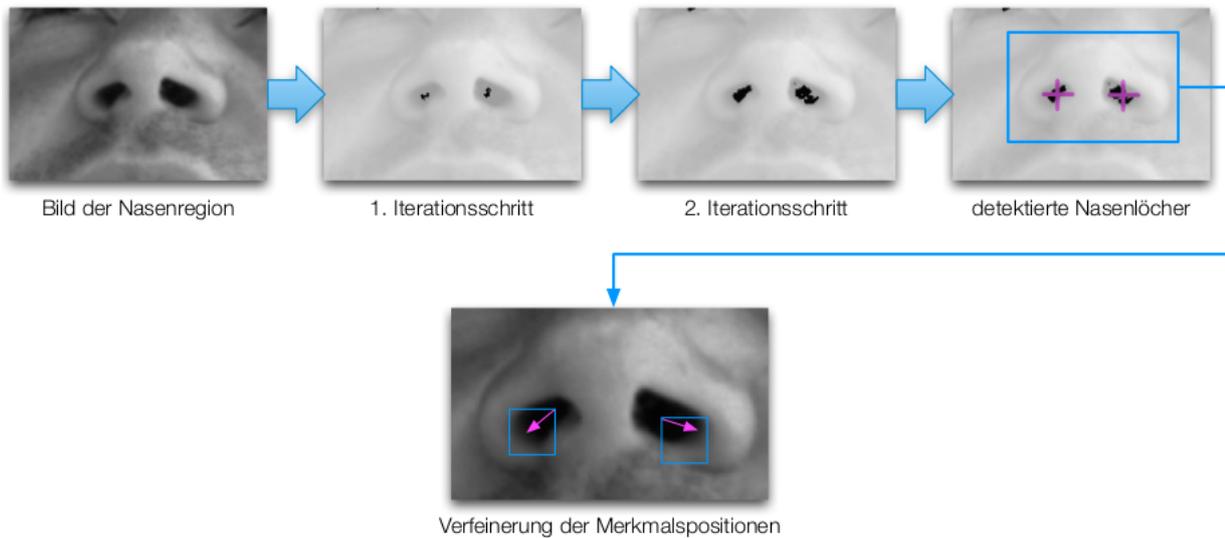


Abb. 3.15.: Die Nasenlochflächen werden über ein iteratives Schwellenwertverfahren aus dem Graustufenbild segmentiert. Als Lokalisierung wird der zur Nasenspitze am weitesten entfernte Bildpixel gewählt. *Quelle: [PBZ10]*

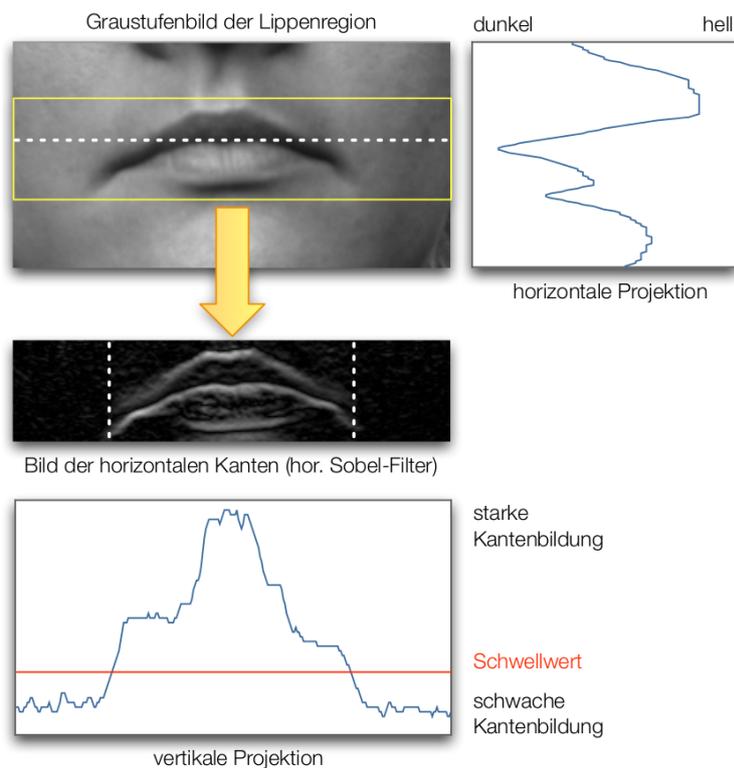


Abb. 3.16.: Über einen horizontalen Sobel-Filter [BB15] werden die Kanten in der Lippenregion bestimmt. Mit einer spalten- bzw. zeilenweisen Summierung lassen sich die Mundwinkel lokalisieren. *Quelle: [PBZ10]*

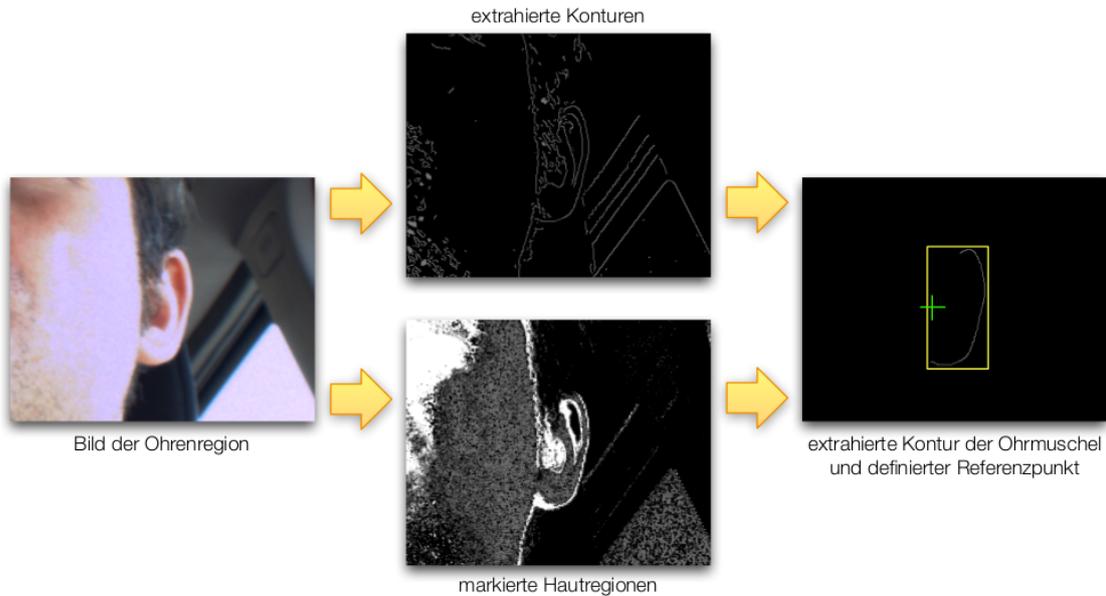


Abb. 3.17.: Die spezifischen Konturen der Ohrmuschel und der Farbgebung im Bereich der Ohren ermöglichen die Lokalisierung des *Tragus* (kleine Knorpelmasse an der Ohrmuschel). *Quelle: [PBZ10]*

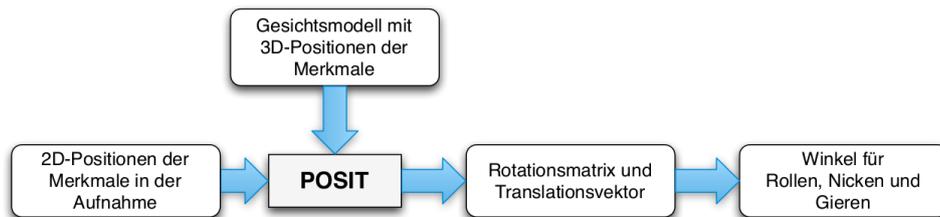


Abb. 3.18.: Über den *POSIT-Algorithmus* wird aus den 2-D-Positionen der extrahierten Gesichtsmerkmale und einem Gesichtsmodell, welches aus den 3-D-Positionen der Gesichtsmerkmale besteht, die Kopfpose des Fahrers bestimmt.

Mit der ermittelten Rotationsmatrix, dem Translationsvektor und der Skalierung lässt sich die Pose des Objekts (hier die Pose des Gesichtsmodells) eindeutig bestimmen. Eine vergleichbare Vorgehensweise zur Lokalisierung der Gesichtsmerkmale sowie zur anschließenden Kopfposenbestimmung über die *POSIT-Methode* findet sich bereits 1996 in der Arbeit von Stiefelhagen, Yang und Waibel [SYW96].

3.4. Abbildung der Blickrichtung auf die Geschehnisse innerhalb und außerhalb des Fahrzeugs

Die geschätzte Blickrichtung wird erst durch die Bezugnahme auf die Geschehnisse im nahen Fahrzeugumfeld zur wertvollen Information für Fahrerassistenzsysteme. Eine in Bezug auf die aktuelle Verkehrssituation interpretierte Einordnung der Blickrichtung wird als *assoziierte Blickrichtungsinformation* bezeichnet. Während die Blickrichtungsinformation die Blickrichtung als Geradengleichung beinhaltet, enthält die assoziierte Blickrichtungsinformation bereits verarbeitetes Wissen über die Fahrzeuggeometrie, Fahrspuren, die Objekte des nahen Umfelds oder Wissen über temporale Blickmuster des

Fahrers.

Beispiel 3.1. Soll für ein nachgeschaltetes FAS nicht nur der Blickrichtungsvektor, sondern auch die Information „Fahrer hat die Autobahnausfahrt gesehen: wahr oder falsch“ bereitgestellt werden, setzt dies zusätzliche Informationen voraus:

- **Fahrspurinformation**
 - Wo befinden sich die Indikatoren der Autobahnausfahrt (Fahrspurverlauf, Ankündigungsschilder, etc.) relativ zum Ego-Fahrzeug oder dem Welt-Koordinatensystem?
 - Welches geometrische Feld spannen die Indikatoren der Autobahnausfahrt relativ zum Fahrzeug auf?
 - Auf welcher Fahrspur befindet sich das Ego-Fahrzeug?
- **Geolokalisierung des Ego-Fahrzeugs**
 - Wo befindet sich das eigene Fahrzeug im Welt-Koordinatensystem?
- **Fahrzeuggeometrie**
 - Wie ist die den Fahrer beobachtende Kamera extrinsisch zum Fahrzeug kalibriert?
 - Wo sind die Fensterflächen räumlich angeordnet?
- **Kognitives Wissen**
 - Wann gilt die Autobahnausfahrt oder allgemein ein Objekt als gesehen?
- **Fremdobjekte**
 - Verdecken andere Objekte, wie große LKWs, die Sicht auf die Indikatoren der Autobahnausfahrt?
- **Temporale Zusammenhänge**
 - Waren die Indikatoren der Autobahnausfahrt sichtbar während der Fahrer in deren Richtung blickte?

Abschnitt 3.4.1 beschreibt die Abbildung des Blicks auf die Entitäten innerhalb und außerhalb des Fahrzeugs. Konkret wird erläutert, wie die vom Blickrichtungserkennungssystem erkannte Blickrichtung auf *Objekte*, *fahrzeugstationäre Sichtfelder (Regionen von speziellem Interesse)* und *Fahrspurinformation* abgebildet wird.

Da ein flüchtiger Blick nicht unbedingt mit der tatsächlichen Wahrnehmung von Geschehnissen gleichgesetzt werden kann, definiert Abschnitt 3.4.2 ein Konfidenzmaß, welches quantitativ die Sicherheit einer bewussten Wahrnehmung modelliert. Das Bewusstsein des Fahrers wird dabei nicht über Sensoren gemessen. Eine bewusste Wahrnehmung einer Entität wird über die Assoziation der Blickrichtung geschlossen.

Definition 3.7. Eine Entität gilt in dieser Arbeit als *geschnitten*, wenn die Blickrichtungsgerade die Entität räumlich schneidet. Eine Entität gilt in dieser Arbeit als *wahrgenommen*, wenn der Fahrer die Entität bewusst wahrgenommen hat. Eine bewusste Wahrnehmung wird über die Assoziation der Entität mit der Blickrichtung geschlossen.

Probabilistische Expertensysteme wie *Markov Logik Netze (MLNs)*, *Case-Based-Reasoning (CBR)* oder *Bayessche Netze (BNs)* schließen korrelierende oder (inter)kausale Zusammenhänge aus angenommenen Zuständen und Beobachtungen der Umwelt.

Beispiel 3.2. Aus der Beobachtung, dass der Fahrer durch sein Blickverhalten den rückwärtigen Verkehr auf der Autobahn verfolgt und der Tatsache, dass sich ein vorausfahrendes, langsames Auto auf der Ego-Fahrspur befindet, kann geschlossen werden, dass ein Überholen in den nächsten Sekunden sehr wahrscheinlich ist.

In *MLNs*, welche diese Arbeit zur Analyse des Fahrverhaltens verwendet (siehe Kapitel 4), werden Aussagen über einen Zusammenhang oder eine Situation als „prädikatenlogische Formeln“ formuliert. Abschnitt 3.4.3 beschreibt das Bilden von prädikatenlogischen Aussagen bezüglich des Blickverhaltens im Kontext der vorherrschenden Verkehrssituation.

3.4.1. Abbildung des Blicks auf Entitäten innerhalb und außerhalb des Fahrzeugs

Abschnitt „Abbildung auf das unterliegende Straßennetz“ beschreibt wie die erkannte Blickrichtung auf das Straßennetz abgebildet wird. Zur Abbildung wird die von Bender et al. definierte geometrische Beschreibung einer Fahrspur als *Lanelet* verwendet [BZS14]. Abschnitt „Abbildung auf fahrzeugstationäre Sichtfelder“ beschreibt die Abbildung des Blicks auf Sichtfelder innerhalb und außerhalb des Fahrzeugs. Die Abbildung des Blicks auf fahrzeugstationäre Sichtfelder wird in erster Linie für die Beschreibung des Blicks im Fahrzeuginnenraum verwendet.

Abschnitt „Objekte“ beschreibt ein sensorunabhängiges Konzept zur Abbildung der Blickrichtung auf Umweltobjekte. Zur nachfolgenden Validierung dieses Konzepts wird beispielhaft seine Umsetzung anhand der Objektliste einer Laserscanner-ECU vorgestellt. Des Weiteren wird in diesem Abschnitt speziell auf die Zuordnung des Blicks zu sich im Spiegel befindlichen Objekten eingegangen.

Abbildung auf das unterliegende Straßennetz

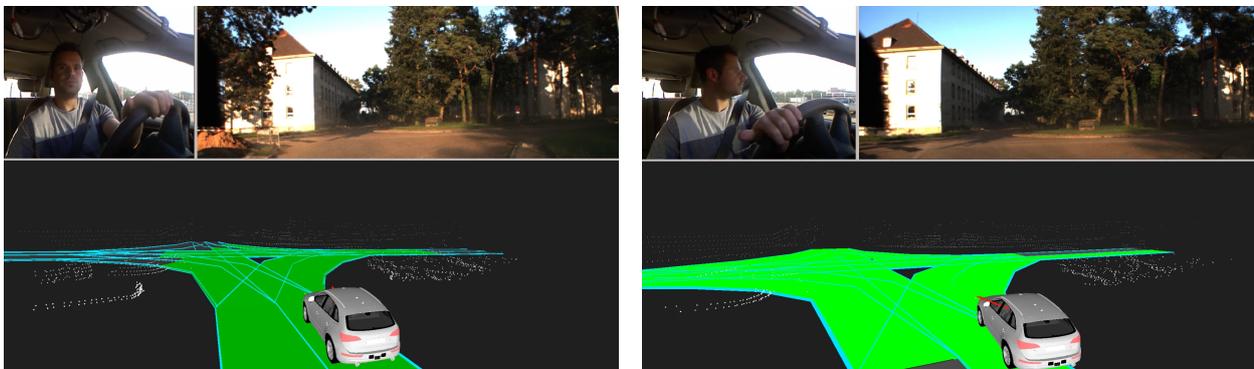
Im *OpenStreetMap-Projekt (OSM)* wird die Topologie des Straßennetzes mit Knoten (geografischen Punkten), Wegen (geordnete Listen von Knoten) und Relationen modelliert [HW08a]. Der Schwerpunkt wird in OSM-Projekt auf die Erstellung von Straßenkarten für Menschen gelegt und lokale Geometrien werden typischerweise ungenau dargestellt. Dies geschieht aus ästhetischen Gründen, zur Reduktion des notwendigen Speicherbedarfs und mangels ausreichend genauer Information über lokale Spurverläufe. Bender et al. stellen mit dem Konzept der „Lanelets“ eine OSM basierende Modellierung vor, die für autonome Fahrzeuge optimiert ist und die detaillierte Abbildung lokaler Geometrien vereinfacht.

Definition 3.8. Ein *Lanelet* ist ein durch seinen rechten und linken Rand definiertes Fahrspursegment mit einer beliebigen Länge. Die Ränder des Lanelets bestehen jeweils aus Wegen, die einen Polygonzug bilden. Die Fahrtrichtung eines Lanelets ergibt sich durch eine explizite Unterscheidung von rechtem und linkem Rand, annotiert durch eine Relation. Analog zu Wegen in OSM können sich Lanelets kreuzen oder ineinander übergehen. Aufeinander folgende Lanelets haben gemeinsame Anfangs- und Endpunkte.

Die Krümmung zwischen ihren Rändern darf am Übergang einen Schwellwert nicht überschreiten. Für eine ausführlichere Definition sei auf [BZS14] verwiesen.

Die Zuordnung der Blickrichtung zu den Lanelets erlaubt eine Aussage darüber, auf welche Bereiche der Straße der Fahrer seine Aufmerksamkeit richtet. Zur genaueren Bestimmung der betrachteten Straßenbereiche werden die Lanelets im lokalen Fahrzeugumfeld anhand der Punkte des rechten und linken Rands in Dreiecksfelder aufgeteilt. Dabei liegen stets zwei Eckpunkte des Dreiecks auf dem einen Rand und ein Punkt auf dem anderen Rand des Lanelets.

Um vom Blick des Fahrers *geschnittene* Lanelets zu identifizieren, wird die Blickrichtungsgerade als auch die geometrischen Punkte der Lanelets ins Fahrzeugkoordinatensystem transformiert. Anhand des Gierwinkels der Blickrichtung wird ein Sichtkegel aufgespannt, in dem alle sich darin befindlichen Dreiecke als *geschnitten* gelten. Abbildung 3.19 zeigt die Abbildung des Fahrerblicks auf die Lanelets im nahen Fahrzeugumfeld.



(a) Abbildung des Fahrerblicks auf Lanelets einer T-Kreuzung. Der Fahrer blickt zunächst auf die Zielspur. Auf von links kommenden Verkehr wurde noch nicht geprüft.

(b) Im späteren Verlauf prüft der Fahrer ob von links kommender Verkehr ein Kollisionsrisiko darstellt.

Abb. 3.19.: Abbildung des Fahrerblicks auf Lanelets. Die vom Fahrer *wahrgenommenen* Lanelets sind in grün dargestellt. Lanelets modellieren eine Fahrspur in geographischer Form [BZS14].

Abbildung auf fahrzeugstationäre Sichtfelder

Mit einer bekannten Fahrzeuggeometrie lassen sich Sichtfelder im Fahrzeuginnenraum mittels Triangulierung durch ein Dreiecksnetz im 3-D-Raum repräsentieren. Blickt der Fahrer auf einen Sichtbereich, schneidet sich die Gerade der Blickrichtungserkennung $g_C(\lambda)$ mit dem als Dreiecksnetz modellierten Sichtbereich. Auf die Indizierung des Koordinatensystems wird zu Gunsten der Lesbarkeit im Folgenden verzichtet.

Mit der Geradengleichung $g(\lambda)$ des Blickrichtungserkennungssystems im Fahrzeugkoordinatensystem

$$g(\lambda) = \mathbf{o} + \lambda \mathbf{b}$$

und einer Ebenengleichung $d(\kappa, \rho)$, definiert durch die Eckpunkte $(\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3)$ des Dreiecks

$$d(\kappa, \rho) = \mathbf{p}_1 + \kappa(\mathbf{p}_2 - \mathbf{p}_1) + \rho(\mathbf{p}_3 - \mathbf{p}_1)$$

schneidet die Gerade das Dreieck genau dann, wenn für die Gleichsetzung von Gerade und Ebene gilt, dass κ und ρ im Bereich $[0, 1]$ liegen und $\kappa + \rho \leq 1$ ist.

$$g(\lambda) \stackrel{!}{=} d(\kappa, \rho)$$

$$(\mathbf{o} - \mathbf{p}_1) = \begin{pmatrix} -\mathbf{b} & (\mathbf{p}_2 - \mathbf{p}_1) & (\mathbf{p}_3 - \mathbf{p}_1) \end{pmatrix} \begin{pmatrix} \lambda \\ \kappa \\ \rho \end{pmatrix} \quad [3.18]$$

Ist dies der Fall gilt das Sichtfeld als *geschnitten*.

Abbildung 3.20 zeigt die in dieser Arbeit verwendeten Sichtfelder. Unter Betrachtung von in realen Fahrversuchen gewonnenen Messdaten haben sich folgende Sichtfelder als praxistauglich erwiesen:

- Innenspiegel
- Außenspiegel rechts
- Außenspiegel links
- Armaturenbrett (Tachometer, Benzinanzeige, Drehzahlmesser, ...)
- Mittelkonsole (Radio-/Navigationsbereich)
- Handschuhfach
- Windschutzscheibe
- linkes Fenster
- rechtes Fenster

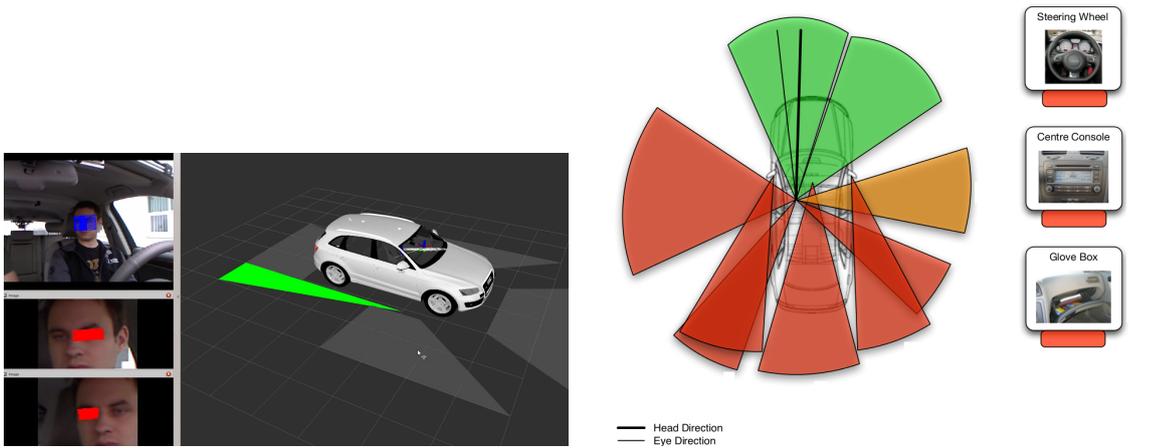
Die Sichtbereiche Armaturenbrett, Mittelkonsole und Handschuhfach befinden sich im Fahrzeuginnenraum und geben dem Fahrer keine Information über das Verkehrsgeschehen.

Abbildung auf Objekte und Objekte im Spiegel

Objekte Im Gegensatz zu fahrzeugstationären Sichtfeldern haben Objekte der Umwelt, wie Stoppschilder, Ampeln, andere Fahrzeuge oder Fußgänger relativ zum Fahrzeug dynamischen Charakter. Ihre Position relativ zum Ego-Fahrzeug ist variabel.

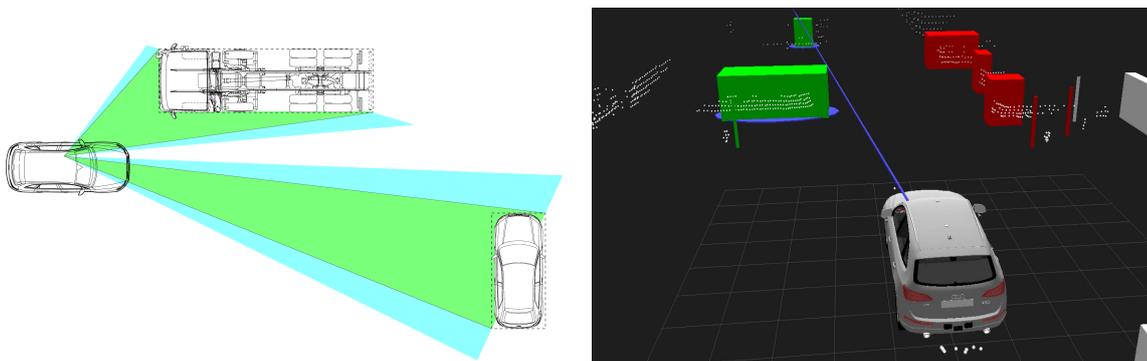
Auch dynamische Objekte können als Netz von Dreiecken modelliert werden um den Schnitt mit der Blickrichtungsgerechten entsprechend der Formel 3.18 zu berechnen. Für Umfelderkennungssysteme wie die Laserscanner des FZI-Versuchsträgers CoCar, die direkt die Abmaße des objektumhüllenden Quaders liefern bietet sich jedoch die Abstraktion der Objekte als solchen an. Der objektumhüllende Quader spannt zur erkannten Kopfposition einen Winkelbereich in allen drei Achsen auf und erlaubt so eine schnelle Schnittberechnung mit der Blickrichtungsgerechten. Abbildung 3.21a zeigt durch Fremdobjekte aufgespannte Winkelbereiche in der Drauf- und Seitenansicht einschließlich eines additiven Toleranzbereichs.

Ein Objekt gilt als *geschnitten*, wenn die Blickrichtungsgerechte sich mit dem objektumhüllenden Quader (zuzüglich eines Toleranzbereichs) über festen Zeitraum schneidet. Praxistauglich hat sich ein Zeitraum von mindestens 500 ms erwiesen.



(a) Dreidimensionale Ansicht der um das Fahrzeug definierten Sichtbereiche. Der Fahrer blickt in den rechten Außenspiegel und der zugeordnete Sichtbereich gilt als *wahrgenommen*. (b) Vogelperspektive auf die definierten Sichtbereiche mit Darstellung der Konfidenz (siehe Abschnitt 3.4.2) als Farbe.

Abb. 3.20.: Die im Fahrzeuginnenraum definierten Sichtbereiche sind das Armaturenbrett, die Mittelkonsole und das Handschuhfach. Außerhalb des Fahrzeugs wurden das Blickfeld nach vorne, nach links und nach rechts definiert. Blicke in den Spiegel werden auf nach hinten gerichtete Sichtkegel abgebildet. Jedes Sichtfeld wird als Dreiecksnetz im 3-D-Raum repräsentiert. Schneidet der Blick des Fahrers das Sichtfeld, gilt das Sichtfeld als *geschnitten*.



(a) Fremdobjekte spannen ausgehend von der Kopfposition des Fahrers einen Sichtkegel entsprechend ihrer Größe auf. (b) Abbildung des während der Fahrt erkannten Blicks auf die Objektliste der Laserscannerdaten. Gesehene Objekte sind grün markiert. Relevante, übersehene Objekte (Abschnitt 4.4) sind rot markiert.

Abb. 3.21.: Objekte werden als als Quader repräsentiert, welche zur erkannten Kopfposition einen Sichtkegel in allen drei Achsen aufspannen. Befindet sich der Blickrichtung des Fahrers in diesem Sichtkegel gilt das Objekt als *gesehen*.

Objekte im Spiegel Die Bestimmung des im Spiegel beobachteten Objekts setzt eine bekannte Orientierung der Spiegelfläche sowie die Bekanntheit dessen Wölbung voraus. Kleinere Ungenauigkeiten in der Blickrichtungsschätzung oder der Optik des Spiegels haben eine große Winkelabweichung im gespiegelten Blick zur Folge. Der Einfachheit halber wird für jeden Spiegel ein Sichtkegel definiert und ein Blick in den Spiegel über die im Abschnitt „Abbildung auf fahrzeugstationäre Sichtfelder“ definierten Sichtfelder erkannt. Abbildung 3.20 (Seite 74) zeigt die durch den Innenspiegel und die Außenspiegel aufgespannten Sichtkegel. Blickt der Fahrer in den Spiegel, wird genau ein Objekt, nämlich das zur Spiegelfläche am nächstliegende Objekt im Sichtkegel, als *geschnitten* erachtet.

3.4.2. Bewusste Wahrnehmung - Aufmerksamkeitskonfidenz

Die vorherigen Abschnitte (3.4.1 und 3.4.1) spezifizieren, wie sich die Geradengleichung des Blickrichtungsvektors auf Sichtfelder und Objekte abbilden lässt. Schneidet die Blickrichtungsgerade $g(t)$ ein Sichtfeld, eine Fahrspur oder ein Objekt $S_i(t)$ zu einem Zeitpunkt t , gilt es als *geschnitten*. Formel $\zeta_i(t)$ definiert diesen Zusammenhang:

$$\zeta_i(t) := \begin{cases} \top & , g(t) \cap S_i(t) \neq \emptyset \\ \perp & , g(t) \cap S_i(t) = \emptyset \end{cases}$$

Die Zeichen \top bzw. \perp stehen für die Wahrheitswerte *wahr* bzw. *falsch*. Aus dem Schnitt der Blickrichtungsgeraden mit einem Objekt oder Sichtbereich lässt sich nicht direkt eine bewusste Wahrnehmung eines Objekts oder eines Sichtfelds folgern. Um einen flüchtigen Blick (oder ein Messrauschen) von einer bewussten Wahrnehmung zu unterscheiden, wird eine Aufmerksamkeitskonfidenz gebildet. Beim Bilden dieser Aufmerksamkeitskonfidenz müssen folgende Aspekte berücksichtigt werden:

Attentive Fixation Um ein Objekt oder eine Information in einem Sichtfeld bewusst wahrzunehmen muss es mit den Augen attentiv fixiert werden. Fixationen der Augen können zwei verschiedenen Typen unterteilt werden [Met09]: Präattentive Fixationen führen noch nicht zur bewussten Wahrnehmung eines Objekts. Sie dienen der Exploration der Umwelt. Während attentiven Fixationen können Objekte bewusst wahrgenommen werden.

Da attentive Fixationen eine Mindestdauer von circa 500 ms haben [Met09], kann ein Sichtfeld erst als bewusst wahrgenommen gelten, wenn der Blick diese Mindestdauer im Sichtfeld verweilt.

Aktualität der Information Informationen in einem Sichtfeld wie dem Armaturenbrett oder der Mittelkonsole sind nur zum Zeitpunkt der Betrachtung aktuell. Wendet der Fahrer den Blick von einem Sichtfeld ab, kann dieser nur noch eine gewisse Zeitspanne als bewusst wahrgenommen gelten.

Dynamik der Umwelt Verkehrsteilnehmer wechseln ihre Fahrwege und Geschwindigkeiten. Auch Ampeln oder dynamische Straßenschilder ändern ihre Zustände. Ändert ein Verkehrsteilnehmer beispielsweise seine Geschwindigkeit oder seine Fahrtrichtung sehr stark, kann er von einem ungefährlichen zu einem gefährlichen Verkehrsteilnehmer werden.

Begrenztes Kurzzeitgedächtnis Die meisten im Straßenverkehr erkannten Geschehnisse bleiben dem

Fahrer nur kurze Zeit im Gedächtnis. Je länger ein Blick in einen Sichtbereich ausbleibt, desto sicherer ist es, dass die dort auftretenden Ereignisse dem Fahrer nicht mehr bewusst sind.

Messrauschen Einzelne Ausreißer der Blickrichtungsschätzung dürfen nicht zu einer Fehleinschätzung der Aufmerksamkeit führen.

Unaufmerksamkeitsblindheit bzw. Veränderungsblindheit Wenn größere Veränderungen in der Umgebung nicht wahrgenommen werden, weil sich der Betrachter auf einen anderen Teilaspekt der Szene oder eine Aufgabe konzentriert, spricht man von Veränderungsblindheit.¹ Unaufmerksamkeitsblindheit hingegen bezeichnet das Phänomen, dass der Fahrer ein Objekt betrachtet, es aber trotzdem nicht bewusst wahrnimmt (oft auch als „Tagträumen“ bezeichnet).

Mit diesen Überlegungen lässt sich die Aufmerksamkeitskonfidenz des Fahrers definieren:

Der Aspekt der geforderten *attentiven Augenfixation* wird in der Aufmerksamkeitskonfidenz berücksichtigt, indem die Aufmerksamkeitskonfidenz für einen Sichtbereich erst erhöht wird, wenn der Fahrer mit seinem Blick eine Mindestdauer von 500 ms verweilt. Die Forderung der Mindestverweildauer, vermeidet zusätzlich Ungenauigkeiten durch *Messrauschen*. Der Zeitpunkt τ der letzten *attentiven Fixation* eines Sichtfelds durch den Fahrer, definiert sich zu einem Zeitpunkt t_k wie folgt:

$$\tau := \begin{cases} t_k & , \zeta_i(t_k) \wedge \dots \wedge \zeta_i(t_{k-500ms}) = \top \\ t_{k-1} & , \text{sonst} \end{cases}$$

Der Aspekt des *begrenzten Kurzzeitgedächtnis* des Fahrers und das Veralten von Informationen innerhalb eines Sichtfelds kann durch eine zeitlich bedingte Verringerung der Aufmerksamkeitskonfidenz modelliert werden.

$$\beta_i(t_k) := \left(\frac{t_k - \tau}{T}\right)^4$$

T ist dabei die Zeitspanne, nach der dem Sichtbereich keinerlei Aufmerksamkeit mehr zugeordnet wird. Für Objekte findet eine Reduzierung der Aufmerksamkeitskonfidenz auch dann statt, wenn sich die Geschwindigkeit $v_i(t)$ oder die Bewegungsrichtung $\omega_i(t)$ eines Objekts ungesehen vom Fahrer ändert (Aspekt *Dynamik der Umwelt*). Das Maß der Reduzierung hängt dabei von der Stärke der Geschwindigkeits- bzw. Richtungsänderung ab.

$$\gamma_i(t_k) := \sum_{t=\tau}^{t_k} \left(\left| \frac{\dot{v}_i(t)}{\dot{v}_{\max}} \right| + \left| \frac{\omega_i(t)}{\omega_{\max}(v_i(t))} \right| \right)$$

Verzögerungen stellen die betragsmäßig größten Beschleunigungen im Straßenverkehr da. Ein gewählter Maximalwert von $\dot{v}_{\max} := 11 \text{ m/s}^2$ entspricht einer Vollbremsung.

Die maximale Richtungsänderung ω_{\max} in einem Zeitschritt ist abhängig von der momentanen Geschwindigkeit des Fahrzeugs. Ein schnell fahrendes Fahrzeug auf der Autobahn kann, bedingt durch die kinetische Energie, seine Bewegungsrichtung nicht derart stark verändern wie ein langsames, einparken-

¹ Ein bekanntes Beispiel ist der „Gorillas in unserer Mitte“ Versuch von Simons und Chabris, bei dem der Betrachter eines Videoclips angehalten wird die Anzahl der Pässe in einem Ballspiel zu zählen und dabei nicht bemerkt, dass ein Gorilla das Bild durchquert [SC99].

des Auto. Die maximale Gierrate ω_{\max} wurde abhängig von der Geschwindigkeit $v(t)$ als empirische Funktion definiert. Zur Ermittlung wurden mit dem FZI-Versuchsträger CoCar Kreise mit unterschiedlichen Radien gefahren und dabei Geschwindigkeiten und Gierraten aufgezeichnet. Die dabei erreichten

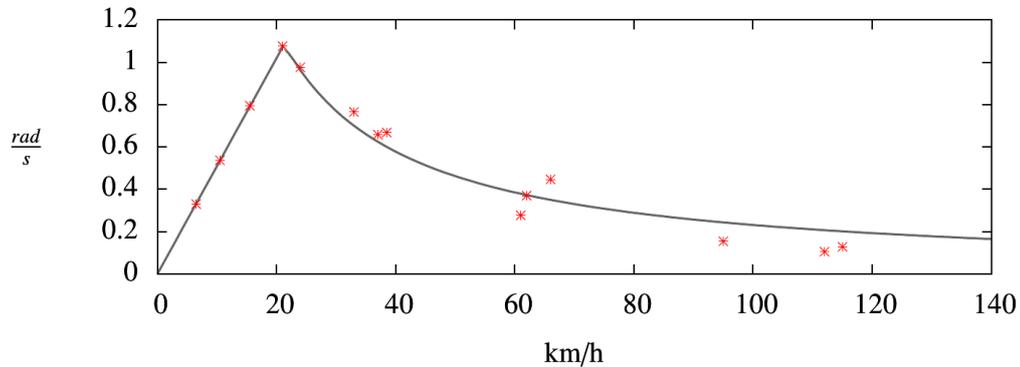


Abb. 3.22.: Die maximale Richtungsänderung ω_{\max} in einem Zeitschritt ist Abhängigkeit von der momentanen Geschwindigkeit des Fahrzeugs. Über Messversuche lässt sich die Gierrate ω_{\max} abhängig von der Geschwindigkeit $v(t)$ als empirische Funktion definieren.

Geschwindigkeiten und Gierraten sind in Abbildung 3.22 als rote Punkte eingezeichnet. Anhand der gewonnenen Messwerte wurde mit der Methode der kleinsten Quadrate eine Näherungsfunktion bestimmt.

$$\omega_{\max}(v(t)) := \begin{cases} 0,18 \cdot v(t) & , v(t) < 6 \text{ m/s} \\ 6,41 \cdot \frac{1}{v(t)} & , \text{sonst} \end{cases}$$

Der Effekt der Unaufmerksamkeitsblindheit bzw. der Veränderungsblindheit lässt sich mit der in dieser Arbeit gewählten Sensorik nicht modellieren. Nach heutigem Stand ist auch kein Sensor bekannt, mit dem dieses Phänomen nicht-intrusiv messbar wäre. Eine Modellierung der Unaufmerksamkeitsblindheit bzw. der Veränderungsblindheit würde einen messbaren Einblick in die menschlichen Gedankengänge erfordern.

Mit der Verringerung der Aufmerksamkeitskonfidenz über die Zeit β und die Dynamik des gesehenen Objekts γ , lässt sich eine Aufmerksamkeitskonfidenz $a_i(t_k)$ für ein Objekt oder einen Sichtbereich definieren:

$$a_i(t_k) = \begin{cases} 1 & , t_k = \tau \\ & \text{(Objekt wird gerade angesehen)} \\ \max(0, 1 - \beta_i(t_k) - \gamma_i(t_k)) & , \tau < t_k \\ & \text{(Objekt wird gerade nicht angesehen)} \end{cases} \quad [3.19]$$

3.4.3. Bildung von prädikatenlogischen Aussagen

Neben der Abbildung des Blicks auf geometrische Regionen oder die dynamischen Objekte der Umwelt, sind für nachgeschaltete Inferenzsysteme Aussagen über das aktuelle Blickverhalten des Fahrers von großem Nutzen. Eine Aussage wird in diesem Zusammenhang als logische Aussage verstanden, welche

entweder *wahr* oder *falsch* sein kann.

In *Markov Logik Netzen*, welche in dieser Arbeit zur Situationsinterpretation verwendet werden (siehe Abschnitt 4.1, Seite 83), werden Relationen zwischen logischen Aussagen definiert, um damit Wahrscheinlichkeiten über den Wahrheitsgehalt nicht beobachtbarer Aussagen (zum Beispiel der Fahrstilklasse) oder zukünftige Ereignisse (zum Beispiel einer Aussage über die Überholwahrscheinlichkeit) abzuleiten. Mit einem Schwellwert, den die Aufmerksamkeitskonfidenz a übersteigen muss (siehe Formel 3.19), lässt sich für jedes Sichtfeld und Objekt aus Abschnitt eine logische Aussage erstellen. Außerdem lassen sich Aussagen definieren, ob der Fahrer den rückwärtigen Verkehr beobachtet, die vorherrschende Geschwindigkeitsbegrenzung oder das gefährlichste Objekt des Straßenverkehrs (*Most-Relevant-Target*) gesehen hat.

Fahrer überwacht den rückwärtigen Verkehr: Alterniert der Blick des Fahrers zwischen dem Innen- und linkem Außenspiegel, überwacht er den rückwärtigen Verkehr. Dieses Blickmuster ist vermehrt vor einem beabsichtigtem Überholvorgang auf der Autobahn zu beobachten.

Geschwindigkeitsbegrenzung gesehen: In Kombination mit der Umfeldsensorik und der Bestimmung der in der momentanen Fahrszene relevanten Verkehrszeichen [Nie14], lässt sich ableiten ob der Fahrer das momentan vorherrschende Geschwindigkeitssymbol gesehen hat.

Ein Geschwindigkeitssymbol kann dabei eine Ampel, ein Stoppschild oder eine Geschwindigkeitsbegrenzung sein.

Fahrer sieht relevantestes Fremdobjekt: Mit der Identifizierung des für die Fahrszene relevantesten Fremdobjekts lässt sich eine Aussage erstellen, ob der Fahrer das relevanteste Fremdobjekt gesehen hat. Eine Identifizierung des relevantesten Fremdobjekts liefert beispielsweise Schamm in [Sch14].

3.5. Zusammenfassung

Die dynamische Umgebung Automobil stellt höchste Anforderungen an ein Blickrichtungserkennungssystem. Weiträumige und schnelle Kopfbewegungen des Fahrers beim Einparken müssen ebenso erkannt werden wie Spiegelblicke, die sich oftmals nur durch eine Veränderung der Pupillen abzeichnen. Dazu erschweren ständig wechselnden Lichtverhältnisse oder die zeitweisen Verdeckungen von Teilen des Gesichts die stabile Erkennung der Kopfpose. Insgesamt wurden in dieser Arbeit 10 Anforderungen an eine Blickrichtungserkennung im Automobilbereich definiert:

- Sicherheit des Fahrers
- Hohe Genauigkeit
- Fähigkeit des Verfolgens schneller Kopfbewegungen
- Echtzeitfähigkeit
- Weiträumiger Erkennungsbereich
- Reproduzierbarkeit der Ergebnisse
- Kompensation wechselnder Lichtverhältnisse
- Kompensation von Verdeckungen
- Wirtschaftliche und technische Möglichkeit der Hardwareintegration
- Mehrbenutzerfunktionalität

Als Sensorsysteme mit denen derartige Ansprüche wirtschaftlich umsetzbar wären, wurden CCD-Kameras und RGB-D-Kameras identifiziert. Der Einsatz kompakter RGB-D-Kameras, wie der Microsoft[®] Kinect[™]-Kamera, ist im Automobilbereich weitestgehend unerforscht. In Abschnitt 3.2.1 wurden deshalb speziell die Vor- und Nachteile von RGB-D-Kameras im Fahrzeug analysiert.

Die über RGB-D-Kamera umgesetzte Blickrichtungserkennung wurde speziell auf die Domäne „Automobil“ ausgerichtet und hebt sich durch folgende Eigenschaften von anderweitig publizierten Forschungsarbeiten ab:

- Die Kopfpose des Fahrers wird über einen *Multi-Template-ICP-Algorithmus* aus dem Tiefenbild der RGB-D-Kamera bestimmt. Die Verwendung mehrerer Gesichtsmasken erhöht den Erkennungsbe- reich der Kopfpose. Die Korrespondenzsuche über die intrinsischen Kameraparameter des Tiefen- bilds ermöglichen eine echtzeitfähige Kopfposenschätzung. Das Umschalten der Fehlerfunktion je nach Iterationsfortschritt ermöglicht eine schnelle und akkurate Konvergenz des ICP-Algorithmus.
- Nachteile der RGB-D-Kameras, wie die Störanfälligkeit des Tiefenbilds bei starkem Gegenlicht, werden über das RGB-Farbbild der Kamera kompensiert. Während des störungsfreien Betriebs wird ansichtsbasiert die Kopfstellung des Fahrers über *Random Forests* gelernt. Dazu wird nebenläufig eine Trainingsdatenbank aufgebaut. Bei fehlerhaftem Tiefenbild wird die Kopfpose aus dem RGB- Farbbild der RGB-D-Kamera über die nebenläufig trainierten *Random Forests* geschätzt.
- Durch eine Identifikation des Fahrers über die Silhouette des Gesichts werden personalisierte Gesichtsmasken des Fahrers geladen.
- Die Augenstellung des Fahrers wird durch eine orthogonale Projektion des RGB-Farbbilds auf eine planar zur Kamera stehende Ebene über ein Augenmodell geschätzt. Aus der Augenstellung wird die Blickrichtung in Form eines Geradenvektors bestimmt.

Neben der Schätzung der Blickrichtung über eine RGB-D-Kamera wurde eine Erkennung über eine CCD-Kamera implementiert, welche über das Erkennen von Gesichtsmerkmalen aus einem hochaufge- lösten RGB-Farbbild die Kopfpose über den *POSIT-Algorithmus* bestimmt. Als stabile Gesichtsmarkale wurden die Mundwinkel, die Augenwinkel, die Ohrmuscheln und die Nasenlöcher identifiziert und ma- schinell erkannt. Die Berechnung der Augenstellung wird ein dreidimensionales Augenmodell verwendet.

Zur wertvollen Information wird die Blickrichtung für nachgeschaltete FAS vor allem, wenn eine Be- zugnahme zu den Geschehnissen im nahen Fahrzeugumfeld hergestellt wird. Ein flüchtiger Blick auf ein Objekt bedeutet nicht, dass der Fahrer sich des Objekts bewusst ist. Um einen flüchtigen Blick von einer bewussten Wahrnehmung zu unterscheiden, wird in dieser Arbeit eine Aufmerksamkeitskonfidenz definiert. Die Aufmerksamkeitskonfidenz wird für die Entitäten des Fahrzeugumfelds gebildet und basiert auf folgenden Überlegungen:

- Eine attentive Augenfixation ist Voraussetzung einer bewussten Wahrnehmung
- Information verliert an Aktualität. Wird die Entität nicht mehr angesehen, muss die Aufmerksam- keitskonfidenz hinsichtlich dieser Entität mit zunehmender Zeit abnehmen.
- Die Umwelt ist dynamisch. Ändert eine Entität ihren Zustand, muss die Aufmerksamkeitskonfidenz hinsichtlich dieser Entität abnehmen.
- Menschen haben ein begrenztes Kurzzeitgedächtnis.

- Die Blickrichtungserkennung unterliegt einem gewissen Messrauschen
- Unaufmerksamkeitsblindheit und Veränderungsblindheit beschreiben den Effekt, dass ein Mensch eine Entität anblickt jedoch trotzdem nicht wahrnimmt. Diese Aspekte wurden in der Bestimmung der Aufmerksamkeitskonfidenz nicht berücksichtigt.

Für alle Objekte des Fahrzeugumfelds sowie den Fahrspuren und definierten fahrzeugstationären Sichtfeldern wird diese Aufmerksamkeitskonfidenz abgeleitet. Außerdem werden aussagenlogische Ausdrücke, wie „Fahrer überwacht rückwärtigen Verkehr“ oder „Fahrer hat das relevanteste Objekt wahrgenommen“, gebildet und über die Aufmerksamkeitskonfidenz bewertet.

4. Analyse des Fahrerhaltens

Eine benutzeradaptive Assistenz fordert ein aussagekräftiges *Fahrermodell*, welches den Fahrer in seinen Eigenschaften und seinem Verhalten im Straßenverkehr beschreibt und damit nachgeschalteten Assistenzsystemen die Möglichkeit eröffnet, Algorithmen und Modelle individuell auf den Fahrer anzupassen.

Definition 4.1. Die Begriffe *Persönlichkeit*, *Eigenschaften des Fahrers* und *Fahrstil* sind in dieser Arbeit wie folgt belegt:

In der Philosophie oder der Psychologie ist der Begriff *Persönlichkeit* oft als die Individualität des Menschen, hinsichtlich derer sich Menschen auf Grund Ihrer einzigartigen psychischen Eigenschaften von anderen unterscheiden, beschrieben [Wik14c]. In dieser Arbeit wird die *Persönlichkeit* vereinfacht als Menge von menschlichen Eigenschaften definiert. Als eine wichtige Untermenge der *Persönlichkeit* wird für diese Arbeit die Untermenge \mathfrak{P} , als die *für das Fahren eines Kraftfahrzeugs relevanten Eigenschaften des Fahrers* definiert.

Im Vergleich zu Persönlichkeitsdefinitionen aus der Philosophie oder der Psychologie ist der Begriff „*Persönlichkeit*“ hier bewusst stark vereinfacht. So wie sich die *Persönlichkeit* des Menschen ändern kann, können sich auch die *Eigenschaften des Fahrers* ändern. Der Fahrer kann neue *Eigenschaften* erwerben, vorhandene ändern oder verlieren. *Persönlichkeitseigenschaften* wie *sportlich* oder *ängstlich* sind im Kontext dieser Arbeit immer auf die *Fahrweise* bezogen.

Definition 4.2. Als *Fahrverhalten* wird in dieser Arbeit eine zeitlich sequentielle Folge von Aktionen verstanden, die ein Fahrer in Interaktion mit seiner Umwelt aktiv oder reaktiv ausführt um eine Fahraufgabe zu bewältigen. Das *Fahrverhalten* des Fahrers ist von seiner *Persönlichkeit* (Definition 4.1) geprägt.

Folgende Informationsarten wurden für das erklärte Ziel der fahrerindividuellen Parametrisierung von FAS als essenziell identifiziert:

Fahrdynamische Kennwerte (Komfortparameter): Fahrdynamische Kennwerte sind individuell und beschreiben das *Fahrverhalten* quantitativ in Form von Intervallgrenzen, Verteilungen oder physikalischen Kennwerten. Der Begriff *Komfortparameter* prägt sich durch die Denkweise, dass die Parameterwerte die Intervallgrenzen spezifizieren, in denen sich der Fahrer sicher und wohl fühlt. Typische *Komfortparameter* sind der üblicherweise eingehaltene Sicherheitsabstand, das übliche Brems- oder Beschleunigungsverhalten oder üblicherweise gefahrene Geschwindigkeiten. Als Beispiel eines nachgeschalteten Assistenzsystems wird in Abschnitt 5.3 der Evaluation gezeigt, wie durch die Schätzung von fahrerindividuellen Geschwindigkeitsprofilen sowie der Brems- und Beschleunigungscharakteristik die Reichweitenprognose verbessert wird.

Fahrstil- oder Risikoklasse: Mit dem Fahrstil oder der Risikobereitschaft wird das Fahrverhalten des Fahrers in einer Fahrstiloberklasse beschrieben. Die Oberklasse soll das Verhalten des Fahrers unabhängig vom Verkehrskontext klassifizieren. Im Schätzprozess wird der Verkehrskontext jedoch durchaus berücksichtigt. Eine typische Kategorisierung des Fahrverhaltens in einer Oberklasse wäre die Klassifizierung des Fahrstils als zum Beispiel defensiv, energieeffizient oder sportlich. In [Bär+11] wurde anhand der Parametrisierung des Energieeffizienzsystems ANESA mit dem Fahrstil gezeigt, dass mit Hilfe der geschätzten Fahrstilklasse Hinweise an den Fahrer effektiver vermittelt werden können.

Aktueller Zustand: Im Hinblick auf die momentane Handlung des Fahrers und dessen Fahrverhalten kann der Begriff des aktuellen Zustands weiträumig interpretiert werden. Diese Arbeit unterscheidet Zustände, welche der Fahrer (als Mensch) annehmen kann sowie Verhaltenszustände, welche ein modelliertes Fahrverhalten annehmen kann. Für den Zustand des Fahrers sind vor allem der *Aufmerksamkeitszustand*, der *Vitalitätszustand*, die momentane *Körperstellung* und der *Gemütszustand* von Bedeutung für nachgeschaltete FAS.

Hinsichtlich des Fahrerverhaltens ergeben sich die Zustände aus dem Rubikon-Modell [HG87], einem Modell aus der Psychologie zur Erklärung des menschlichen Handelns. Typisch für eine aktuelles Manöver wäre [*Überholen*]. Für einen aktuellen Aktionszustand wäre [*Handeln*] typisch.

Hypothesen über die zukünftige Situationsentwicklung: Prognosen über die Situationsentwicklung bieten FAS die Möglichkeit, Gefahren frühzeitig zu erkennen und den Fahrer diesbezüglich zu warnen. Eine Situationsprognose kann einem nachgeschalteten FAS über verschiedene Schnittstellen bereitgestellt werden. Typisch wäre in Form einer Wahrscheinlichkeit für ein zukünftiges Ereignis. Z.B. die Wahrscheinlichkeit, dass der Fahrer in den nächsten 3 Sekunden überholen wird. Eine Prognose über die Situationsentwicklung kann aber auch in Form einer Trajektorie oder einer Folge von zukünftig erwarteten Aktionen bereitgestellt werden.

Um vom beobachteten Fahrverhalten auf die vom Fahrerassistenzsystem geforderten Informationen zu schließen, wurden folgende Methoden zur situativen Analyse des Fahrverhaltens umgesetzt:

Fahrstilerkennung: Durch eine situative Analyse des Fahrerverhaltens im jeweiligen Verkehrskontext wird der Fahrstil geschätzt. Abschnitt 4.2 beschreibt das Vorgehen zur Fahrstilerkennung.

Schätzung von Komfortparametern: Durch statistische Analyse lassen sich fahrerindividuelle Geschwindigkeitsprofile oder übliche Brems- und Beschleunigungscharakteristiken bestimmen. Abschnitt 4.3.1 zeigt das Vorgehen zur Schätzung dieser Komfortparameter.

Objektspezifische Aufmerksamkeitsbestimmung: Über das Lernen eines individuellen Blickverhaltensmodells lässt sich die Aufmerksamkeit des Fahrers in Bezug auf die in seiner Umgebung befindlichen Objekte bestimmen. Abschnitt 4.4 beschreibt die Modellierung des Blickverhaltens in Bezug auf das lokale Fahrzeugumfeld und die Bestimmung einer objektspezifischen Aufmerksamkeit.

Prognose des Fahrverhaltens: Über das Erkennen von wiederkehrenden Mustern im Fahrerverhalten

können Aktionen und Manöver des Fahrers prognostiziert werden. Abschnitt 4.5 beschreibt die Erkennung des aktuellen Handlungszustands des Fahrers und die Prognose zukünftiger Handlungszustände.

Die in dieser Arbeit entwickelten Komponenten werden zu einem *Fahrermodell* kombiniert. Dem *Fahrermodell* unterliegt als Inferenzsystem die Grundstruktur eines *Markov Logik Netzes (MLN)*. Abschnitt 4.1 führt *MLNs* ein und beleuchtet die Vorteile der Netze bezüglich der Modellierung des Fahrerhaltens.

4.1. Markov Logik Netze für die Modellierung des Fahrerhaltens

Markov Logik Netze bieten die Möglichkeit, Wissen in Form von prädikatenlogischen Formeln erster Stufe zu beschreiben und gleichzeitig Unsicherheiten im Wissen über den Formeln zugehörigen Gewichten zu lernen.

Für die Analyse des Fahrerhaltens erweisen sich die folgenden Eigenschaften der *MLNs* als vorteilhaft:

- Menschliches Verhalten ist zu komplex um es direkt mathematisch abbilden zu können. Die Aktionen des Fahrers sind im allgemeinen nicht deterministisch und variieren stark in ihrer Ausführung. *MLNs* erlauben es, Korrelationen, Implikationen und Relationen im Fahrerhalten anhand von Trainingsdaten in probabilistischer Form zu lernen.
- Die Wahrnehmung des Fahrers und dessen Umwelt basiert auf Sensoren, deren Messungen grundsätzlich mit Unsicherheiten behaftet sind. Im *MLN* lassen sich diese Unsicherheiten modellieren und im Inferenzprozess berücksichtigen.
- In *MLNs* lassen sich temporale Zusammenhänge modellieren. Dies erlaubt das Generieren von Hypothesen über die zukünftige Situationsentwicklung.
- Mit der Formulierung prädikatenlogischer Terme lassen sich Relationen und Kausalitäten mathematisch fundiert modellieren.

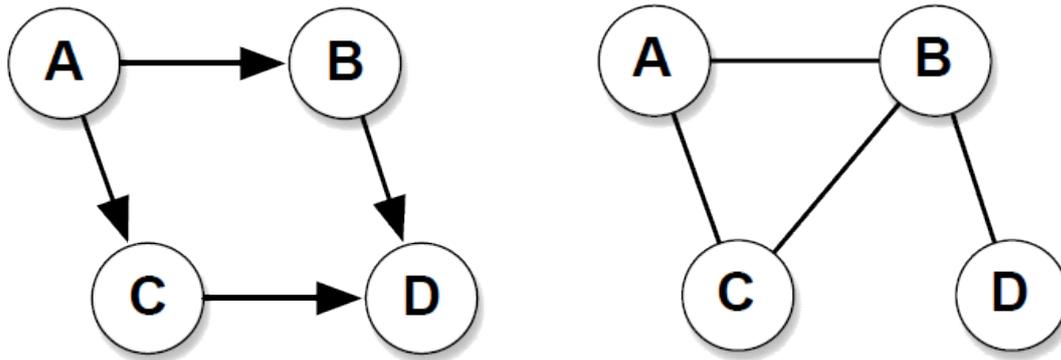
MLNs eignen sich besonders zur Modellierung von Korrelationen, d. h. wie wahrscheinlich zwei Ereignisse A und B gemeinsam auftreten. Stellt man Wissensbeschreibung über die prädikatenlogischen Formeln als Graphen dar, ergibt sich ein ungerichteter Graph.

Dies unterscheidet sie von anderen probabilistischen Inferenzmodellen wie den *Bayesschen Netzen (BNs)*. Diese eignen sich besonders zur Modellierung von Kausalitäten, d. h. wie wahrscheinlich aus einer Aktion A eine Reaktion B folgt. Durch die eindeutige Inferenzrichtung von A nach B kann ihr Wissen als azyklischer, gerichteter Graph dargestellt werden.

Abbildung 4.1 verdeutlicht den Unterschied der beiden Netzstrukturen.

Richardson und Domingos definieren ein *MLN* L als eine Menge von Paaren (F_i, w_i) mit F als eine prädikatenlogische Formel erster Stufe und $w \in \mathbb{R}$ einem zugehörigen Gewicht als reelle Zahl [RD06].

$$L := \{(F_i, w_i)\} \quad i \in \mathbb{N} \quad [4.1]$$



- (a) *Bayessche Netze (BNs)* eignen sich für die Modellierung von Kausalitäten. Sie lernen die Wahrscheinlichkeit, dass aus Aktion A die Reaktion B erfolgt. Ihr Wissen kann als azyklischer, gerichteter Graph dargestellt werden.
- (b) *Markov Logik Netze* eignen sich für die Modellierung von Korrelationen. Sie lernen inwiefern die Ereignisse A und B gemeinsam auftreten. Ihr Wissen kann als ungerichteter Graph dargestellt werden.

Abb. 4.1.: *Markov Logik Netze* eignen sich für die Modellierung von Korrelationen. *Bayesschen Netze (BNs)* eignen sich hingegen für die Modellierung von Kausalitäten. Stellt man das modellierte Wissen der Inferenzmethoden als Graphen dar, entsteht für *MLNs* ein ungerichteter Graph. Für *BNs* ergibt sich ein azyklischer, gerichteter Graph.

Prädikatenlogische Formeln erster Stufe setzen sich aus Objekten (Konstanten und Variablen) und Relationen zusammen. Tabelle 4.1 gibt einen Überblick über die in der Prädikatenlogik erster Stufe eingesetzten Sprachsymbole. Eine besondere Bedeutung kommt bei der Inferenz mittels *MLNs* den *Grundatomen* zu. Sie repräsentieren einen möglichen Weltzustand des Netzes, indem sie entweder den Zustand *wahr* oder *falsch* annehmen. Im Inferenzprozess dienen die Grundatome als Eingabe und Ausgabe zugleich:

Per Inferenz werden Wahrscheinlichkeiten für Weltzustände ermittelt. Grundatome, deren Zustand zum Beispiel durch die Beobachtung der Welt bekannt sind, werden im Inferenzprozess auf *wahr* bzw. *falsch* gesetzt. Für alle anderen (unbekannten) Grundatome wird die Wahrscheinlichkeit über ihr Auftreten in den Weltzuständen (und unter Berücksichtigung deren Wahrscheinlichkeit) ermittelt.

Zur Inferenz kann in *Markov Logik Netzen* der *MC-SAT* Algorithmus benutzt [RD06] werden. Als Initialisierung wählt dieser Algorithmus einen Netzzustand, in dem alle Formeln mit unendlichem Gewicht erfüllt sind. Anschließend werden n neue Netzzustände generiert. Dazu wird, basierend auf dem letzten Netzzustand, eine Teilmenge von Formeln gemäß ihrer Gewichtung gezogen, welche im nächsten Zustand erfüllt sein müssen. Aus den generierten Zuständen wird nach dem Zufallsprinzip über den *SampleSAT*-Algorithmus ein neuer Netzzustand bestimmt. Weiterführende Details zur Inferenz in *MLNs* gibt unter anderem [Nie14], bezogen auf die Relevanzbestimmung von Verkehrsschildern und Ampeln. Mit Hilfe der Grundatome kann das dem *MLN* unterliegende *Markov-Netz (MN)* als ungerichteter Graph dargestellt werden. Jedes Grundatom wird durch einen Knoten repräsentiert. Beeinflussen sich zwei Grundatome, d.h. sie treten in mindestens einer Formel zusammen auf, werden sie durch eine Kante verbunden.

Objekte		
Konstanten	Konstante Objekte	A, B, C
Variablen	Objekt, welches verschiedene Werte annehmen kann	a, b, c
Relationen		
Funktionen	Abbildung von Objekt-tupel auf ein Objekt	f
Prädikate	Abbildung nach wahr oder falsch	$y = f(x)$
Operatoren (spezielle Relationen)		
Quantoren	„Für alle“ bzw. „Es existiert“ Operator	\forall, \exists
Logische Verknüpfungen	„Und“, „Oder“, „Daraus folgt“, ...	$\wedge, \vee, \neg, \Rightarrow, \Leftrightarrow$
Terme und Formeln		
Terme	Repräsentation eines Objekts	$A, a, f(x)$
Grundterm	Variablenfreier Term	$f(A), A$
Atomare Formel (Atom)	Prädikat mit Objekt-tupel	$x > y$
Grundatom	Variablenfreie atomare Formel	$A > B, A = f(B)$
Formel	Verbindung von atomaren Formeln mit Quantoren und Logischen Symbolen	$\forall a \exists x : a > x$

Tab. 4.1.: Überblick über die in den *Markov Logik Netzen* eingesetzten Sprachelemente der Prädikatenlogik erster Stufe. Eine ausführliche Syntaxbeschreibung der Prädikatenlogik erster Stufe geben Russell und Norvig in [RN03, Seite 303ff].

4.2. Fahrstilerkennung

Die Erkennung des Fahrstils zielt darauf ab, das Fahrverhalten des Fahrers in einer Oberklasse (nämlich dem Fahrstil) zu beschreiben.

Definition 4.3. Unter dem Begriff *Fahrstil* wird eine von der Verkehrssituation und deren Umwelteinflüssen losgelöste Beschreibung des Fahrverhaltens (Definition 4.2) verstanden.

Mit \mathcal{D} als der Menge aller Fahrstile und \mathcal{B} als Menge der Fahrverhalten (siehe Definition 4.2), kann die Fahrstilerkennung F als ein Klassifikationsproblem formalisiert werden:

$$F : \mathcal{B} \rightarrow \mathcal{D} \quad [4.2]$$

Es ist zu beobachten, dass der Fahrstil oft als Parametrisierung für FAS verwendet wird. Die maschinelle und objektive Erkennung des Fahrstils unterbleibt jedoch häufig. Der Fahrer kann (muss) seinen Fahrstil selbst wählen (die Evaluation wird in Abschnitt 5.2, Seite 120 zeigen, dass Fahrer bei der Beurteilung ihres Fahrstils subjektiv sind).

Beispiel 4.1. Der Fahrer kann in einem automatikbetriebenen Fahrzeug das Gangwechselverhalten des Fahrzeugs auf „sportlich“ schalten. Der Fahrer gibt seinen Fahrstil somit selbst ein.

In dieser Arbeit wird das Fahrverhalten in Bezug zur vorherrschenden Verkehrsszene bewertet und hebt sich so von einer vereinfachten Fahrstilklassifikation ab, welche ausschließlich auf den Aktionen des Fahrers basiert.

Beispiel 4.2. Fährt ein Fahrer Tempo 80 km/h in der Innenstadt ist dies anders zu bewerten als wenn er 80 km/h auf der Autobahn fährt. Auch starkes Abbremsen oder ruckartiges Lenken in einer Gefahrensituation weisen nicht unbedingt auf einen sportlichen oder aggressiven Fahrer hin.

Definition 4.4. Eine *Verkehrssituation*, *Verkehrsszene*, *Fahrsituation* oder *Fahrszene* ist eine Menge von Entitäten (konkrete und abstrakte Gegenstände, Eigenschaften, Sachverhalte, Ereignisse, Prozesse) innerhalb eines zeitlich und räumlich abgeschlossenen Bereichs [Nie14, Definition 1.2].

Anmerkung 1: Mit \mathfrak{E} als die Menge aller während einer Fahrt möglichen Entitäten, kann eine *Verkehrssituation* als eine räumlich und zeitlich eingeschränkte Untermenge $S \subset \mathfrak{E}$ verstanden werden.

Um das Fahrverhalten im Kontext der Verkehrssituation zu bewerten, wird die Klassifikation des Fahrverhaltens auf Teilprobleme, nämlich die Klassifikation des Fahrverhaltens in einer gegebenen Verkehrssituation, abgebildet.

$$F_S : \mathfrak{B}_S \rightarrow \mathfrak{D}_S \quad [4.3]$$

Eine nachgelagerte Fusion der letzten k für die jeweiligen Verkehrssituationen geschätzten Fahrstile D_S führt zu einer globalen Schätzung des Fahrstils D .

$$D := D_{S,t} \oplus D_{S,t+1} \oplus \dots \oplus D_{S,t+k} \quad D \in \mathfrak{D}, D_S \in \mathfrak{D}_S \quad [4.4]$$

Wird die Klassifikation des Fahrverhaltens auf die gegenwärtige Verkehrssituation beschränkt, ergeben sich folgende Vorteile:

- Die Zielmenge $\mathfrak{D}_S \subseteq \mathfrak{D}$ der Klassifikation F_S muss nicht zwingend für alle Verkehrssituationen gleich sein. Das Stoppverhalten an einer Ampel ist beispielsweise sehr aussagekräftig in Bezug auf einen energieeffizienten Fahrstil. Betrachtet man als Verkehrssituation einen 5 km langen Autobahnabschnitt ist diese Szene in Bezug auf das Energieverhalten nur wenig aussagekräftig.
- Die Klassifikation ist nicht an ein einzelnes Klassifikationsverfahren gebunden. Für jede Fahrsituation S kann individuell das beste Klassifikationsverfahren F_S gewählt werden.
- Sind die Verkehrssituationen wohldefiniert (es ist beispielsweise klar an welcher Position ein Verkehrsschild steht und welche Geschwindigkeitsbeschränkung es trägt) kann das Fahrverhalten unterschiedlicher Fahrer bei gleichen Gegebenheiten verglichen werden. Gleichzeitig kann durch die Zunahme dieses Kontextwissens Fehlinterpretationen (siehe Beispiel 4.2) vermieden werden und das Fahrverhalten durch Expertenwissen annotiert werden.
- FAS können anhand der einzelnen Teilergebnisse D_S szenenspezifisch parametrisiert werden

- Die Ableitung eines allgemeinen Fahrstils D durch die Zusammenführung der Fahrstilklassifikationen pro Szene F_S wird aussagekräftiger, da die Teilergebnisse jeweils aussagekräftiger sind.
- Der benötigte Speicherbedarf sinkt, da nur Daten während einer Verkehrssituation gesammelt werden müssen und nach der Klassifikation verworfen werden können.

Ein Nachteil der situationsspezifischen Klassifikation ist die implizite Annahme, dass die vorherrschende Verkehrsszene bekannt oder maschinell erkennbar ist. Mit einer automatischen Erkennung der Verkehrssituation beschäftigen sich beispielsweise [Sch09; Kle14]. In dieser Arbeit wird die Verkehrssituation einer digitalen Karte entnommen.

Komponenten der Fahrstilerkennung

Für die Fahrstilerkennung werden zwei Verfahren, welche sich in der Art und Weise des angewandten Expertenwissens unterscheiden, untersucht.

Im ersten Verfahren wird der Fahrstil über ein *Fuzzy-Inferenzsystem* für eine gegebene Szene ermittelt. Sowohl die Fuzzymengen als auch die Inferenzregeln des *Fuzzy-Inferenzsystem* werden vom Experten vorgegeben. Abschnitt 4.2.1 beschreibt diese Vorgehensweise näher.

Im zweiten Verfahren wird ein *Markov Logik Netz* (siehe Abschnitt 4.1) über Prädikatenlogik vorgegeben und dessen Formelgewichtung über annotierte Messdaten gelernt. Durch die prädikatenlogischen Terme wird beispielsweise vorgegeben, dass die Geschwindigkeit einen Einfluss auf den Fahrstil hat. Ob dieser Zusammenhang richtig oder falsch ist, ob er immer, nur selten oder niemals zutrifft, wird über annotierte Messdaten gelernt. Abschnitt 4.2.2 beschreibt die Fahrstilerkennung über das Markov-Logik-Netzwerk. In beiden Verfahren sind folgende Komponenten zu finden, welche prinzipiell individuell kombinierbar sind:

Datenaufzeichnung und -vorverarbeitung: Am Ende einer Verkehrssituation werden die in der Szene aufgenommenen Fahrzeugdaten vorverarbeitet. Dazu gehört beispielsweise die Berechnung der maximalen Geschwindigkeit, des maximalen Bremsdrucks oder der Durchschnittsgeschwindigkeit. Die Datenvorverarbeitung findet in beiden Verfahren nach den gleichen Prinzipien statt.

Die Beschränkung der Datenaufzeichnung auf eine Verkehrssituation folgt der impliziten Annahme, dass Start und Ende der Verkehrsszene bekannt sind. Messdaten werden in dieser Arbeit einerseits über einen Fahr Simulator und andererseits über den FZI-Versuchsträger CoCar gewonnen. Im Fahr Simulator sind Start und Ende aus der modellierten Fahrstrecke ersichtlich. In den Realfahrten ist die *Fahrstrecke (Route)* in Form einer Folge von geographischen Koordinaten (Wegpunkten) fest vorgegeben. Der Start und das Ende der Verkehrsszene sowie Verkehrsschilder, Ampeln oder die Anzahl der Fahrspuren lassen sich bezogen auf diese geographischen Koordinaten annotieren. Durch das *GPS* des Fahrzeugs lässt sich über einen *Map Matching Algorithmus* die aktuelle Geoposition (nächster Wegpunkt bei sehr feiner Auflösung der Route) auf der Fahrstrecke bestimmen.

Diskretisierung: Im Diskretisierungsschritt werden die vorverarbeiteten Fahrzeugdaten auf verschiedene Attributswerte abgebildet. Dies kann mit Expertenwissen unscharf über eine *Fuzzifikation*

umgesetzt werden. In diesem Fall sind die Attributklassen Fuzzymengen von linguistischen Zuständen. Abschnitt 4.2.1 beschreibt dieses Vorgehen.

Eine Diskretisierung kann auch mit weniger Expertenwissen, beispielsweise über ein unüberwachtes Clusterverfahren (siehe Abschnitt 4.2.2) erfolgen.

Schätzung des Fahrstils der gegenwärtigen Szene: Sind die Attributklassen ermittelt, können diese als Eingabeparameter für ein *Fuzzy-Inferenzsystem* oder ein *MLN* dienen. Für jede Szene wird eine Klassenzugehörigkeit zu den verschiedenen Fahrstilen inferiert.

Schätzung des Fahrstils: Um eine globale, situationsabhängige Aussage über den Fahrstil des Fahrers zu erlangen werden die letzten Fahrstilbewertungen fusioniert. Die Fusion kann über eine *gewichtete Summe*, *Fuzzylogik* oder ebenfalls *MLNs* realisiert werden.

Für beide Verfahren sind die Fahrstilklassen sowie die Verkehrsszenen gleich definiert.

Definierte Fahrstile

Die Fahrstilklassen dieser Arbeit definieren sich über die Risikobereitschaft und das Energiebewusstsein des Fahrers:

Aggressiv: Aggressive Fahrer zeichnen sich durch geringe Rücksichtnahme gegenüber anderen Verkehrsteilnehmern aus. Ihre Fahrweise typisiert sich durch hohe Geschwindigkeiten, starke Beschleunigungen und abrupte Bremsmanöver. Typisch für aggressive Fahrer sind kleine Abstände zu den vorausfahrenden Fahrzeugen. Eine Missachtung der Verkehrsregeln, wie beispielsweise das Überfahren von Stoppschildern oder das Nichtbefolgen der Vorfahrtsregeln, ist ein weiteres Merkmal aggressiver Fahrer.

Sportlich: Sportliche Fahrer nutzen die volle Fahrzeugdynamik. Ihr Fahrverhalten zeichnet sich durch höhere Geschwindigkeiten, starke Beschleunigungen und mittelstarke Bremsungen aus. Besonders in Kurven ist eine Autoführung entlang der Ideallinie erkennbar. Im Gegensatz zu den aggressiven Fahrern zeichnet sich ihre Fahrweise durch Rücksichtnahme auf andere Verkehrsteilnehmer aus. Bei sportlichen Fahrern ist ein angemessener Abstand zu anderen Fahrzeugen zu beobachten.

Moderat (Defensiv, Normal): Moderate Fahrer können ihre Fähigkeiten einschätzen und kennen die Dynamik ihres Fahrzeugs. Typische Merkmale sind Geschwindigkeiten, welche nicht stark von der erlaubten Grenze abweichen. Typisch sind leichte Bremsungen und häufiges Ausrollen, geringe Beschleunigungen und ein ausreichender Sicherheitsabstand.

Unsicher: Unsichere Fahrer können die Fahrzeugdynamik nicht einschätzen. Kennzeichnend sind langsame Geschwindigkeiten aber gleichzeitig auch fehlender Kontrolle, die teilweise in starken Brems- und Beschleunigungsmanövern resultiert. Typisch sind häufige Lenkkorrekturen in der Kurve und kein oder nur ein zögerliches Überholen anderer Verkehrsteilnehmer.

Energiesparend: Energiesparende Fahrer lassen ihr Fahrzeug auf kommende Geschwindigkeitsbeschrän-

kungen ausrollen und versuchen Bremsen zu vermeiden. Typischerweise halten sie die Geschwindigkeit konstant. Beim Anfahren können starke Beschleunigungen beobachtet werden um das Fahrzeug möglichst effektiv auf die gewünschte Reisegeschwindigkeit zu beschleunigen.

Merkmale für Verkehrssituationen

Als Verkehrssituationen wurden alltägliche, häufig vorkommende Landstraßen- und Autobahnscenen gewählt. Die Szenen [Ortseinfahrt], [Scharfe Kurve] und [Stoppschild] wurden in der Simulationssoftware des FZI-Smart-Fahrsimulators modelliert, was ein exaktes Platzieren von Fahrzeugen und Geschwindigkeitsbegrenzungen ermöglicht. Die Szenen [Autobahn] und [Landstraße] wurden sowohl in der Simulationssoftware des FZI-Smart-Fahrsimulators, als auch im realen Verkehr untersucht.

Das Konzept lässt beliebige Fahrscenen zu. Für die experimentelle Validierung des Konzepts wurden fünf Fahrscenen ausgewählt. Die Szenen definieren sich wie folgt:

Ortseinfahrt: Die Verkehrsszene umfasst das Anfahren an ein Ortsschild auf einer Landstraße. In Deutschland ist eine Ortseinfahrt, ausgehend von einer Landstraße, typischerweise durch den Geschwindigkeitsübergang von 100 km/h auf 70 km/h (circa 200 m vor der Ortseinfahrt) und weiter auf 50 km/h (Ortsschild) gekennzeichnet.

Landstraßenfahrt: Kurzer Fahrabschnitt auf einer Landstraße unter der Voraussetzung einer durch andere Verkehrsteilnehmer unbehinderten Fahrt.

Autobahnfahrt: Kurzer Fahrabschnitt auf einer normal befahrenen Autobahn (kein Stau oder stockender Verkehr).

Scharfe Kurve: Durchfahrt einer scharfen Kurve auf einer Landstraße außerhalb einer Ortschaft mit einer durch die Landstraße impliziten Geschwindigkeitsbegrenzung von 100 km/h.

Anfahrt auf ein Stoppschild: Anfahrt auf eine Kreuzung mit Stoppschild ohne vorausfahrenden Verkehrsteilnehmer innerhalb eines Wohngebiets (50 km/h).

4.2.1. Fahrstilerkennung über ein *Fuzzy-Inferenzsystem*

Fuzzylogik ist eine Verallgemeinerung der zweiwertigen *booleschen Logik* und erlaubt aus Angaben wie „schnell“, „sehr schnell“, „stark“, oder „mäßig“ über logische Operatoren zu schließen [Zad65]. Für die Fahrstilerkennung bedeutet dies, dass ein Zusammenhang wie „eine hohe Geschwindigkeit ist ein Indiz für einen sportlichen Fahrstil“ in einer logischen Formel modelliert werden kann.

Fuzzylogik hat im Vergleich zur klassischen Logik nicht nur einen binären Wahrheitswert, sondern einen Zugehörigkeitsgrad zu einem gewissen Zustand. Die Berechnung des Zugehörigkeitsgrads zu einem Zustand wird *Fuzzifizierung* genannt.

Eine *Fuzzymenge* A definiert sich über die Zugehörigkeitsfunktion μ_A als geordnete Menge von Paaren

(Zugehörigkeitsgrad, Zustand) über einem Wertebereich X wie folgt:

$$A := \{(\mu_A(x), x) \mid x \in X\} \quad [4.5]$$

Beispiel 4.3. Eine auf der Autobahn gemessene Geschwindigkeit von 140 km/h wird über Fuzzifizierung auf die Zustände *schnell*, *moderat* oder *langsam* abgebildet, indem ein Zugehörigkeitsgrad zwischen 0 und 1 für jeden Zustand angegeben wird.

Der Wertebereich X hat in diesem Falle die Elemente *schnell*, *moderat* und *langsam*.

$$X = \{ \text{schnell, moderat, langsam} \}$$

Es wird festgelegt, dass die Geschwindigkeit von 140 km/h mit einem Zugehörigkeitsgrad von 0,6 dem Zustand *schnell*, mit 0,4 dem Zustand *moderat* und nicht dem Zustand *langsam* zuzuordnen ist (Fuzzifizierung). Die Fuzzymenge A ergibt sich somit zu:

$$A = \{ (0,6, \text{schnell}), (0,4, \text{moderat}), (0,0, \text{langsam}) \}$$

Anmerkung: Der Zugehörigkeitsgrad modelliert dabei Unschärfe, keine Unsicherheit: Die gefahrene Geschwindigkeit ist bekannt.

Die Fuzzifizierung der Eingabegrößen in die Zielmenge $[0, 1] \subset \mathbb{R}$ kann über Dreiecks-, Rechtecks-, Trapez-, Sigmoid-, Glockenkurven stattfinden. In dieser Arbeit wurde die Trapezfunktion gewählt, welche sich durch 4 Parameter ($a_0 \leq a_1 \leq a_2 \leq a_3$) definieren lässt:

$$\mu(x) := \begin{cases} 0 & , x < a_0 \\ \frac{x-a_0}{a_1-a_0} & , a_0 \leq x < a_1 \\ 1 & , a_1 \leq x < a_2 \\ \frac{a_3-x}{a_3-a_2} & , a_2 \leq x < a_3 \\ 0 & , a_3 \leq x \end{cases} \quad [4.6]$$

Sind die Eingabegrößen auf Fuzzymengen abgebildet, lassen sich über logische Operationen Regeln für die Fahrstilerkennung bilden.

Zur Definition der logischen Operationen gibt es eine Vielzahl unterschiedlicher Möglichkeiten. Tabelle 4.2 gibt die für die Fahrstilerkennung verwendeten Operatorendefinitionen wieder. Die Tabellen 4.3 und 4.4

\vee	Oder	$\mu_{A \cup B}(x)$	$:= \max(\mu_A(x), \mu_B(x))$
\wedge	Und	$\mu_{A \cap B}(x)$	$:= \min(\mu_A(x), \mu_B(x))$
\Rightarrow	Wenn ... Dann ...	$\mu_{A \Rightarrow B}$	$:= \min(\mu_A(x), \mu_B(y))$

Tab. 4.2.: Die in dieser Arbeit gewählten Fuzzy-Operatoren zur Inferenz des Fahrstils.

zeigen die gebildeten Regelsätze für die Landstraße- bzw. die Ortseinfahrtszene.

Die Auswertung der gebildeten Regeln auf den Fuzzymengen wird als *Inferenz* bezeichnet. Wird als

SPEED	AGGRESSIVE	ANXIOUS	ECONOMICAL	KEEN	SEDATE
BELOW	NL	VL	HL	HL	L
AS_ADVISED	L	HL	VL	VL	VL
HIGH	VL	NL	HL	VL	HL
TOO_HIGH	VL	NL	NL	L	NL

Tab. 4.3.: Fuzzy-Regelsatz für die Fahrstilerkennung im Landstraßenszenario mit den Zuständen NL (not likely), HL (hardly likely), L (likely) and VL (very likely).

SPEED	AGGRESSIVE	ANXIOUS	ECONOMICAL	KEEN	SEDATE
BELOW	NL	VL	HL	HL	L
AS_ADVISED	L	HL	VL	VL	VL
HIGH	VL	NL	HL	VL	HL
TOO_HIGH	VL	NL	NL	L	NL

FREEWH_DISTANCE	AGGRESSIVE	ANXIOUS	ECONOMICAL	KEEN	SEDATE
VERY_SMALL	VL	NL	NL	L	HL
SMALL	VL	L	NL	VL	L
REGULAR	HL	HL	L	VL	VL
HIGH	HL	HL	VL	L	L
VERY_HIGH	NL	NL	VL	HL	L

BRAKE_FORCE	AGGRESSIVE	ANXIOUS	ECONOMICAL	KEEN	SEDATE
SMALL	NL	L	VL	L	L
REGULAR	L	L	HL	L	VL
HIGH	VL	NL	NL	VL	NL

Tab. 4.4.: Fuzzy-Regelsatz für die Fahrstilerkennung im Ortseinfahrtsszenario mit den in Tabelle 4.3 eingeführten Zuständen.

Zielmenge eine Fuzzymenge über die Fahrstile D definiert, ergibt sich aus der Inferenz mit $\mu_F(x)$ direkt der Zugehörigkeitsgrad des beobachteten Fahrverhaltens in einer gegebenen Verkehrssituation zu einem spezifischen Fahrstil x .

$$F := \{(\mu_F(x), x) \mid x \in D\} \quad [4.7]$$

Mit dieser Zuordnung kann nach jedem Durchfahren einer Szene der Gesamtfahrstil des Fahrers aus den letzten Szenenbewertungen gebildet werden. Wird der Gesamtfahrstil aus den letzten k Fahrscenen gebildet, gilt:

$$\mu'(x) = \frac{1}{k} \sum_{i=0}^k \mu_i(x) \quad [4.8]$$

Da sich die Fahrstile über die Risikobereitschaft des Fahrers definieren, kann über *Defuzzifizierung* die Risikobereitschaft y des Fahrers als ein Zahlenwert abgeleitet werden. Eine gebräuchliche Methode der Defuzzifizierung ist die Schwerpunktmethode:

$$y = \frac{\int x \cdot \mu(x) dx}{\int \mu(x) dx} \quad [4.9]$$

In dieser definieren die durch Inferenz ermittelten Zugehörigkeitsgrade $\mu_F(x)$ auf den Fuzzymengen eine Fläche, deren Schwerpunkt auf die Achse des Wertebereichs projiziert werden kann.

4.2.2. Fahrstilerkennung über *Markov Logik Netze*

Für die Fahrstilerkennung erweist sich der Einsatz von *MLNs* (siehe Abschnitt 4.1) als Vorteil, da die grundsätzlichen Zusammenhänge zwischen Fahrverhalten und Fahrstil vorgegeben werden können, während die konkreten Ausprägungen aus Trainingsdaten gelernt werden.

Beispiel 4.4. Hohe Geschwindigkeiten sind gemäß der Definition aus Abschnitt 4.2 auf sportliche oder aggressive Fahrer zurückzuführen. Mit den Variablen und deren möglichen Belegungen

$$\begin{aligned} \text{Fahrer} &= \{ \text{„Franz“}, \text{„Hans“}, \text{„Fritz“}, \dots \} \\ \text{Geschwindigkeitswert} &= \{ \text{„hoch“}, \text{„moderat“}, \text{„gering“}, \dots \} \\ \text{Fahrstil} &= \{ \text{„aggressiv“}, \text{„sportlich“}, \text{„unsicher“}, \dots \} \end{aligned}$$

den Prädikaten

$$\begin{aligned} \text{Hat_Geschwindigkeit}(\text{Fahrer}, \text{Geschwindigkeitswert}) \\ \text{Ist_Fahrstil}(\text{Fahrer}, \text{Fahrstil}) \end{aligned}$$

und der Implikation \Rightarrow kann daher wird folgender Zusammenhang definiert werden:

$$\begin{aligned} \text{Hat_Geschwindigkeit}(\text{„Franz“}, \text{„hoch“}) &\Rightarrow \text{Ist_Fahrstil}(\text{„Franz“}, \text{„aggressiv“}) \\ \text{Hat_Geschwindigkeit}(\text{„Franz“}, \text{„hoch“}) &\Rightarrow \text{Ist_Fahrstil}(\text{„Franz“}, \text{„sportlich“}) \end{aligned}$$

und umgekehrt

$$\begin{aligned} \text{Hat_Geschwindigkeit}(\text{„Franz“}, \text{„hoch“}) &\Rightarrow \neg \text{Ist_Fahrstil}(\text{„Franz“}, \text{„unsicher“}) \\ \text{Hat_Geschwindigkeit}(\text{„Franz“}, \text{„hoch“}) &\Rightarrow \neg \text{Ist_Fahrstil}(\text{„Franz“}, \text{„moderat“}) \end{aligned}$$

Wie zutreffend nun der Zusammenhang zwischen hoher Geschwindigkeit und einem bestimmten Fahrstil ist, kann über Trainingsdaten gelernt werden. Gelernt wird eine Gewichtung der Formeln

entsprechend ihres Auftretens in den Trainingsdaten.

Die Möglichkeit zyklischer Abhängigkeiten erweist sich als vorteilhaft, wenn über die Fahrstilerkennung zukünftige Ereignisse prognostiziert werden sollen.

Beispiel 4.5. Der in Beispiel 4.4 definierte Zusammenhang zwischen hohen Geschwindigkeiten und einem sportlichen Fahrstil lässt sich auch in die andere Richtung abbilden. Bei einem Fahrer mit sportlichem Fahrstil lassen sich mit Sicherheit hohe Geschwindigkeiten beobachten. Folglich gilt auch der Zusammenhang:

$$\text{Ist_Fahrstil}(„Franz“ , „sportlich“) \Rightarrow \text{Hat_Geschwindigkeit}(„Franz“ , „hoch“)$$

und somit

$$\text{Ist_Fahrstil}(„Franz“ , „sportlich“) \Leftrightarrow \text{Hat_Geschwindigkeit}(„Franz“ , „hoch“)$$

Ist der Fahrstil des Fahrers über die Beobachtung in Szene A bekannt, kann über das MLN inferiert werden, wie wahrscheinlich der Fahrer in Szene B eine hohe Geschwindigkeit fahren wird. Über das MLN kann so das Fahrverhalten prognostiziert werden.

Das Bilden von Grundatomen Um über *MLNs* vom Fahrverhalten in einer Verkehrssituation auf den Fahrstil zu schließen (siehe Formel 4.4), muss das Fahrverhalten in der jeweiligen Szene über Grundatome (Evidenzen) beschrieben werden. Die als Skalare vorliegenden Datenmerkmale müssen dazu auf eine endliche Menge von diskreten Beobachtungen abgebildet werden. Diese Abbildung kann, ähnlich der Fuzzifizierung in Abschnitt 4.2.1, über Expertenwissen definiert werden.

Eine weitere Möglichkeit ist, mit weniger Expertenwissen eine Zuordnungsmetrik über (unüberwachte) Clusterverfahren zu bestimmen. Der *k-Means-Algorithmus* (auch k-Mittel-Clustering) startet mit k zufällig gewählten Normalverteilungen über die vorliegenden Datenmerkmale und schätzt abwechselnd die Zuordnung der Daten zu den einzelnen Normalverteilungen (Expectation-Schritt) und die Parameter der Normalverteilungen (Mittelwert und die Varianz) auf Basis der letzten Zuordnung (Maximization-Schritt) [RN04].

In der überwachten Form wird die Anzahl der Normalverteilungen k a-priori vorgegeben. In dieser Arbeit wird die Anzahl der Cluster k unüberwacht über das *Bayesian Information Criterium (BIC)* bestimmt [Dau12].

Wählt man bei der Clusteranalyse zu viele Cluster führt dies zu großen Zustandsräumen mit negativen Effekten auf die Generalisierung (*Overfitting*). Wählt man hingegen zu wenige Cluster geht durch die zu grobe Diskretisierung möglicherweise schon zu viel Information verloren. Das *BIC* quantifiziert diesen Kompromiss über einen Belohnungs- und einen Strafterm. Als Belohnungsterm gilt die Summe der Distanzen der Messdaten zu ihren dedizierten Clusterzentren, welche mit jedem neu hinzugefügten

Cluster geringer wird. Als Strafterm wird die Anzahl der Cluster (k) mit der logarithmischen Anzahl der Messdaten multipliziert. Das *BIC* bildet sich somit zu:

$$BIC = \frac{1}{N} \sum_{i=1}^N \|x_i - \hat{x}_i\| + k \cdot \log(N) \quad [4.10]$$

Die optimale Anzahl an Clustern wird mit der Minimierung der BIC-Funktion gefunden.

Mit k Normalverteilungen $\mathcal{N}_i(\mu, \sigma)$ definiert sich die Zuordnungsfunktion $f : \mathbb{R} \rightarrow \mathbb{N}^k$ eines Datenpunkts $x \in \mathbb{R}$ zu einem Cluster i als die Suche nach dem zum Datenpunkt x am nächsten liegenden Clusterzentrum (repräsentiert durch den Erwartungswert μ_i).

$$f(x) := \operatorname{argmin}_i (|\mu_i - x|)$$

Nach der Durchfahren einer Verkehrssituation S lässt sich für alle k Cluster eine logische Aussage

$$v_S : \mathbb{N}^k \rightarrow \mathbb{B}$$

bestimmen, ob die Beobachtung $x \in \mathbb{R}$ dem Cluster i zugeordnet werden kann.

$$v_{S,x}(i) := \begin{cases} \text{wahr} & , f(x) = i \\ \text{falsch} & , \text{sonst} \end{cases} \quad [4.11]$$

Für eine feste Beobachtung $x \in \mathbb{R}$ bilden die Terme $v_{S,x}(0) \dots v_{S,x}(k-1)$ die als Eingabe verwendeten Grundatome des *MLNs*.

Beispiel 4.6. MLN-Syntax: Mit 5 Clustern für die Durchschnittsgeschwindigkeit in der Verkehrssituation „Autobahnfahrt“ können die folgenden 5 Grundatome, welche jeweils wahr oder falsch sein können, gebildet werden.

```
Durchschnittsgeschwindigkeit( "CLUSTER_0" )
Durchschnittsgeschwindigkeit( "CLUSTER_1" )
...
Durchschnittsgeschwindigkeit( "CLUSTER_4" )
```

Beschreibung des Netzes über prädikatenlogische Terme Der Zusammenhang, dass eine Beobachtung $v_S(i)$ auf einen Fahrstil $d_S(j)$ schließen lässt, wird für eine Verkehrssituation in prädikatenlogischer Form wie folgt beschrieben:

$$\forall i \in \mathbb{N}^k \exists j \in \mathbb{N}^{|D|} : v_S(i) \Rightarrow d_S(j) \quad [4.12]$$

Formel 4.12 kann sprachlich als „Für alle Beobachtungen $v_S(i)$ existiert ein Fahrstil $d_S(j)$ für den gilt, dass wenn $v(i)$ beobachtet wurde auch der Fahrstil $d_S(j)$ gefahren wurde.“ ausgedrückt werden. Der Index $j \in [0, |D| - 1]$ nummeriert die Fahrstile.

Die definierte Regel bildet dabei von einer Beobachtung auf einen Fahrstil ab. Im *MLN* lassen sich durchaus auch komplexe Regeln definieren, welche mehrere Beobachtungen kombinieren. Die benötigte Rechenleistung des Netzes steigt bei komplexen Regeln jedoch stark. Um die Tauglichkeit des Ansatzes zu zeigen wurden möglichst einfache Regeln gewählt.

Genauso kann definiert werden, dass wenn Fahrstil $d_S(j)$ gefahren wurde auch eine Beobachtung $v_S(i)$ existiert (siehe Beispiel 4.5).

$$\forall j \in \mathbb{N}^{|D|} \exists i \in \mathbb{N}^k : v_S(i) \Leftarrow d_S(j) \quad [4.13]$$

Mit den Formeln 4.12 und 4.13 ist die Fahrstilerkennung für eine Szene in prädikatenlogischer Form für das *MLN* definiert. Eine Verallgemeinerung zu \Leftrightarrow kann übrigens nicht stattfinden, da zum Beispiel mehrere Beobachtungen auf den gleichen Fahrstil hinweisen könnten.

Ohne die Existenz- und Allquantoren (Skolemisierung) kann Formel 4.12 als *Konjunktive Normalform (KNF)* dargestellt werden (auf die Indizierung der Verkehrsszene wird zur besseren Lesbarkeit verzichtet):

$$\begin{aligned} \forall i \in \mathbb{N}^k \exists j \in \mathbb{N}^{|D|} : v(i) \Rightarrow d_S(j) \\ \Leftrightarrow \\ (v(0) \Rightarrow d_S(0) \vee v(0) \Rightarrow d_S(1) \vee \dots \vee v(0) \Rightarrow d_S(|D|-1)) \\ \wedge (v(1) \Rightarrow d_S(0) \vee v(1) \Rightarrow d_S(1) \vee \dots \vee v(1) \Rightarrow d_S(|D|-1)) \\ \vdots \\ \wedge (v(k) \Rightarrow d_S(0) \vee v(k) \Rightarrow d_S(1) \vee \dots \vee v(k) \Rightarrow d_S(|D|-1)) \end{aligned} \quad [4.14]$$

In jedem der \wedge -verbundenen Terme ist für genau einen der durch \vee -verbundenen Terme die Relation „ \Rightarrow “ wahr. Das definierte *MLN* zur Fahrstilerkennung enthält alle Relationen. Ob die Relation zutrifft oder nicht, wird in Form ihrer Gewichtung über die Trainingsdaten gelernt.

Beispiel 4.7. MLN-Syntax: Aus Formel 4.14 lässt sich für die Autobahnscene folgendes Netz definieren:

```

1 // Funktionsdefinitionen
2 Geschwindigkeit_Autobahn( geschwindigkeit! , id )
3 Fahrstil_Autobahn( fahrstil! , id )
4
5 // Relationen
6 Geschwindigkeit_Autobahn( +v , i ) => Fahrstil_Autobahn( +s , i )

```

Symbolik: Jede Durchfahrt der Szene erhält in den Messdaten eine eindeutige Nummer *id*. Das + Symbol bewirkt, dass im *MLN* eine Formel für jeden möglichen Wert der Variable entsteht. Über das ! Symbol wird dargestellt, dass diese Beziehung für genau eine Variable zutrifft. In diesem Fall gibt es pro *id* genau eine beobachtete Geschwindigkeit und genau einen beobachteten Fahrstil.

Inferenz des Gesamtfahrstils aus den einzelnen Szenen Um von den szenenspezifischen Fahrstilen auf den Gesamtfahrstil des Fahrers zu schließen, kann in prädikatenlogischer Form folgender Zusammenhang definiert werden:

$$\forall i \in \mathbb{N}^{|D|} : d_S(i) \Rightarrow d(i) \quad [4.15]$$

Da es sich um einen definierten Zusammenhang handelt, kann dieser Formel das Gewicht ∞ zugeordnet werden.

Beispiel 4.8. MLN-Syntax: Ein MLN, mit dem aus den szenenspezifischen Beobachtungen der Gesamtfahrstil des Fahrers inferiert werden kann, definiert sich über den MLN-Syntax wie folgt:

```

1 // Funktionsdefinitionen
2 Person( name , id )
3 Fahrstil( fahrstil , name )
4
5 // Relationen (fest)
6 Person( p , i ) ^ Fahrstil_Autobahn( f , i ) => Fahrstil( f , p ) .
7 Person( p , i ) ^ Fahrstil_Landstrasse( f , i ) => Fahrstil( f , p ) .
8 ...
9 Person( p , i ) ^ Fahrstil_Stoppschild( f , i ) => Fahrstil( f , p ) .

```

Ein Punkt am Ende der Formel bewirkt, dass dieser Zusammenhang als zweifelsfrei (Gewicht ∞) angenommen wird. An das Netz kann die Anfrage `Fahrstil("Moderat" , "Klaus")` gestellt werden, was die Wahrscheinlichkeit liefert, dass der Fahrer „Klaus“ „moderat“ fährt.

Auch die Anfrage `Fahrstil` ist möglich, was für jede Fahrer-Fahrstil Kombination die Wahrscheinlichkeit ausgibt.

Eine feste Zuordnung von Fahrer zu entsprechendem Fahrstil kann über die *Maximum Likelihood Methode* erreicht werden.

4.2.3. Vergleich der Fahrstilerkennung über *Fuzzylogik* und *Markov Logik Netze*

Über Fuzzymengen, definiert auf nominalen Wertebereichen, und über Regeln, die an der menschlichen, logischen Denkweise orientiert sind, kann ein transparentes und leicht nachvollziehbares System zur Schätzung des Fahrstils entworfen werden. Eine Erweiterung der Fahrstilerkennung auf neue Szenen oder neue Fahrstile ist über die Definition neuer Regeln und Fuzzymengen problemlos möglich.

Der Entwurf des *Fuzzy-Inferenzsystems* zur Fahrstilerkennung setzt allerdings viel Expertenwissen voraus: Die Fuzzymengen und damit die Zugehörigkeitsfunktionen μ zu den jeweiligen Zuständen müssen festgelegt werden. Bildet man beispielsweise eine Fuzzymenge für die Geschwindigkeit über eine Trapezfunktion (Formel 4.6) in vier linguistische Zustände ab, fordert dies eine Parametrisierung von $4 \times 4 = 16$ Werten. Die hier beschriebene Fahrstilerkennung fordert die Festlegung von $67 \times 4 = 268$ Parameter. Weiter müssen die Fuzzyregeln (in dieser Arbeit waren es 70) definiert werden.

Einfache Verkehrsszenen lassen sich durch Experten parametrisieren. Werden die Szenen komplexer

kann die Parametrisierung nur automatisiert über *Lernen* erfolgen.

Bei der Fahrstilerkennung über *MLNs* dient das Expertenwissen nur dazu generelle Zusammenhänge zu definieren. Beispielsweise wird definiert, dass eine Beobachtung auf einen Fahrstil schließen lässt (Formel 4.12) und wie viele Fahrstilklassen vorhanden sind. Eine Gewichtung der *MLN-Formeln* und die Diskretisierung der Wertebereiche der Eingabesignale findet unüberwacht statt. Die Fahrstilerkennung über *MLNs* bietet in der Modellierung folgende Vorteile:

- Intuitive Modellierung über Prädikatenlogik erster Stufe.
- Wissen über bekannte Zusammenhänge kann in mathematischen Formeln beschrieben werden. Unsicherheiten im Wissen kann über Gewichte modelliert und aus Trainingsdaten gelernt werden.
- Unüberwachte Diskretisierung der Eingabewerte über Bestimmung des *BIC* (Formel 4.10).
- Die Beschreibung in prädikatenlogischen Formeln erlaubt eine einfache Erweiterung der Fahrstilerkennung in Bezug auf weitere Szenen und weitere Fahrstile.

Außerdem bieten *MLNs* noch folgende Vorteile, welche in dieser Arbeit aber nicht weiter verfolgt wurden:

- *MLNs* erlauben die Definition komplexer Regeln mit der sich Beobachtungen kombinieren lassen.
- Über Strukturlernen können selbst die Regeln aus den annotierten Messdaten gelernt werden. Voraussetzung ist allerdings eine umfangreiche und annotierte Messdatenbank.

4.3. Statistische Modellierung des Fahrverhaltens

Über statistische Auswertungen lassen sich fahrerindividuelle Komfortparameter bestimmen, die das Fahrverhalten in quantitativer Art und Weise beschreiben. Abschnitt 4.3.1 zeigt das Vorgehen zur Schätzung dieser Komfortparameter.

4.3.1. Schätzung von Komfortparametern

Ähnlich dem Fahrstil generalisieren *Komfortparameter* das Fahrverhalten des Fahrers. Während die Fahrstilschätzung dem Fahrverhalten eine Oberklasse zuordnet, zielt die Bestimmung der *Komfortparameter* darauf ab, die physikalischen Kennwerte und Intervallgrenzen, in denen sich der Fahrer sicher und wohl fühlt, quantitativ zu modellieren.

Typische *Komfortparameter* sind der übliche Sicherheitsabstand zum Vorausfahrenden oder die üblicherweise gewählten (Kurven-) Geschwindigkeiten, Beschleunigungen und Bremsverzögerungen.

Mit Definition 4.2 des Fahrverhaltens als sequentielle Folge von Aktionen, welche der Fahrer in Interaktion mit seiner Umwelt ausführt, können *Komfortparameter* als die *Parametrisierung* dieser Aktionen verstanden werden.

Beispiel 4.9. Die Anfahrt an ein Stoppschild ist üblicherweise eine Abfolge der Aktionen [*Geschwindigkeit halten*], [*Ausrollen*] und [*Bremsen*].

Aufgrund individueller Unterschiede zwischen Fahrern ergibt sich die Parametrisierung der Aktionsfolge durch die *Komfortparameter*. Bei sportlichen Fahrern sind höhere Bremskräfte und Anfangsgeschwindigkeiten zu beobachten (die *Komfortparameter* „typische Geschwindigkeit“ und „typische Bremskraft“ sind nominal höher). Energiesparende Fahrer werden im Gegensatz dazu versuchen, die Ausrollphase durch vorausschauende Fahrweise so früh wie möglich zu beginnen. Tabelle 4.5 zeigt die longitudinalen Aktionen des Fahrers und ihre Parametrisierung über die *Komfortparameter*.

Aktion	Parametrisierung
Bremsen	typischer Bremsdruck, typische Geschwindigkeit
Beschleunigen	typische Beschleunigung, typische Geschwindigkeit
Ausrollen	typische Geschwindigkeit
Geschwindigkeit halten	typische Geschwindigkeit

Tab. 4.5.: Aktionen des Fahrers und ihre Parametrisierung über die *Komfortparameter*.

Für diese Arbeit wurden die typische Geschwindigkeitswahl sowie die typische Brems- und Beschleunigungscharakteristik als die wichtigsten *Komfortparameter* identifiziert. Sie wirken sich wesentlich auf den Energieverbrauch des Fahrers aus. Das Kapitel „Evaluation und Ergebnisse“ wird in Abschnitt 5.3 Seite 130 ff. zeigen, dass sich die Prognose des Energieverbrauchs durch die Schätzung dieser *Komfortparameter* um bis zu circa 17% Prozent verbessern lässt.

Analog zum Fahrstil lassen sich auch *Komfortparameter* nur kontextsensitiv bestimmen.

Beispiel 4.10. Soll die typische Geschwindigkeit des Fahrers auf der Landstraße (100 km/h) bestimmt werden, so ist zu beachten, ob der Fahrer von anderen Verkehrsteilnehmern beeinflusst wird oder ob der Fahrer durch die Streckenführung (Kurven, Baustellen, etc.) gezwungen ist, geringere Geschwindigkeiten als üblich zu fahren.

Der *Komfortparameter* Geschwindigkeit wird abhängig von der vorherrschenden Geschwindigkeitsbeschränkung ermittelt. Mathematisch wird sie durch eine Normalverteilung über die gemessenen Geschwindigkeiten modelliert.

Eine Geschwindigkeitsmessung fließt jedoch nur in die Kalkulation ein, wenn der Fahrer ungestört von anderen Verkehrsteilnehmern oder unbeeinflusst von Geschwindigkeitsbegrenzungen der Infrastruktur fährt. Zusammenfassend fließt eine gemessene Geschwindigkeit in die Berechnung des *Komfortparameter* Geschwindigkeit ein, wenn das Fahrzeug sich weder in einer Beschleunigungs- noch Verzögerungsphase befindet, der Fahrer nicht von vorausfahrenden Fahrzeugen zu niedrigeren Geschwindigkeiten gezwungen wird, das Fahrzeug sich nicht in einer Kurvenfahrt befindet und das Fahrzeug eine Mindestgeschwindigkeit nicht unterschreitet. Um zu bestimmen, ob der Fahrer frei oder eingeschränkt fährt, müssen verschiedene Informationsquellen fusioniert werden. Über den Lenkwinkel und die Geschwindigkeit des Fahrzeugs, erfasst durch die Fahrzeugsensorik, wird der Fahrschlauch des Ego-Fahrzeugs ermittelt.

Befindet sich ein Objekt im Fahrschlauch, welches langsamer oder gleich der eigenen Geschwindigkeit fährt, ist anzunehmen, dass der Fahrer durch das vorausfahrende Objekt in seiner Geschwindigkeitswahl beschränkt ist.

Die aktuell vorherrschende Geschwindigkeitsbegrenzung ergibt sich aus der Lokalisierung des Fahrzeugs auf der Karte *OpenStreetMap* [HW08b] und der in der Karte annotierten Geschwindigkeitsbeschränkungen. Alternativ kann die Geschwindigkeitsbegrenzung während der Fahrt über das Kamerabild bestimmt werden (zum Beispiel über die Arbeiten von [Nie14]). Ob sich das Fahrzeug in einer Kurvenfahrt befindet, wird über den Lenkradwinkel bestimmt. Alternativ kann diese Information über eine kamerabasierte Fahrspurerkennung abgeleitet werden.

Die typischen Brems- und Beschleunigungscharakteristiken werden analog zur typischen Geschwindigkeit über eine Normalverteilung modelliert. Eine Filterung, ob der Fahrer frei oder eingeschränkt fährt, wird für die Beschleunigungen nicht angewandt, da Beschleunigungen und Verzögerungen gerade durch die Interaktion mit anderen Verkehrsteilnehmern entstehen.

4.4. Analyse des Blickrichtungsverhaltens

Da der Fahrer 90% aller Informationen visuell aufnimmt führen übersehene Objekte im Straßenverkehr unmittelbar zu erhöhtem Gefahrenpotential. Gleichzeitig stellt nicht *jedes* übersehene Objekt eine Gefahr dar. Eine Bewertung, welche Objekte des Straßenverkehrs für den vorherrschenden Verkehrskontext in Bezug auf eine Kollision relevant oder weniger relevant sind, findet sich in [Sch14]. In Kombination mit der Blickrichtungserkennung und der Projektion des Blicks auf die Objekte des lokalen Fahrzeugumfelds (siehe Abschnitt 3.4, Seite 69) können bei der Mensch-Maschine-Fahrzeugübergabe übersehene, gefährliche Objekte visuell hervorgehoben werden oder objektspezifische Warnungen angezeigt werden.

Ein weiterer Aspekt ist die Modellierung des Blickverhaltens. Wissen über das übliche Blickverhalten des Fahrers erlaubt, Abweichungen zum üblichen Blickverhalten zu detektieren und so Rückschlüsse auf die Aufmerksamkeit oder die Konzentrationsfähigkeit des Fahrers zu ziehen. Die Dynamik des Blickverhaltens ist jedoch sehr individuell und zu komplex um es mathematisch abbilden zu können. Es empfiehlt sich, das Blickverhalten des Fahrers zu *lernen*.

Um zu lernen, welche Objekte des Straßenverkehrs die visuelle Aufmerksamkeit des Fahrers auf sich ziehen, werden in dieser Arbeit Objekteigenschaften in einem Merkmalsvektor aggregiert. Zur Bildung des Merkmalsvektors werden die Informationen aus Fahrzeugsensoren, Laserscanner- und Kameradaten fusioniert. Abbildung 4.2 zeigt die Projektion der zu einem Objekt gehörigen 3-D-Punktmenge in das Kamerabild. Über das umschließende Polygon werden die Farbeigenschaften des Objekts in Form von Farbe, Sättigung, Helligkeit und Kontrast bestimmt. Weitere Objekteigenschaften werden von der *Laserscanner-ECU* über die Analyse der Punktwolken der Laserscanner bestimmt. Für jedes Objekt wird ein Merkmalsvektor bestimmt, welcher sich aus folgenden Attributen zusammensetzt:

- Distanz
- Typ
- Geschwindigkeit
- Gierwinkel
- Beschleunigung
- Farbe
- Sättigung
- Helligkeit
- Kontrast
- Fahrzeuggeschwindigkeit



Abb. 4.2.: Neben den Objektmerkmalen gegeben durch die Laserscanner-ECU (Geschwindigkeit, Distanz, etc.) werden für jedes Objekt die Farbeigenschaften in Form von Farbe, Sättigung, Helligkeit und Kontrast bestimmt. Über einen *C4.5 Entscheidungsbaum* wird anhand dieser Merkmale gelernt, wie visuell ansprechend (*[salient]* oder *[nicht salient]*) ein bestimmtes Objekt für den Fahrer ist .

Über die in Abschnitt 3.4 bestimmte Zuordnung der Blickrichtung zu einem konkreten Objekt kann für jedes Objekt ein Label *[salient]* oder *[nicht salient]* gebildet werden. Der mit dem Label versehene Merkmalsvektor wird über einen *Entscheidungsbaum (C4.5)* gelernt. Abbildung 4.3a verdeutlicht den Informationsfluss der Lernphase.

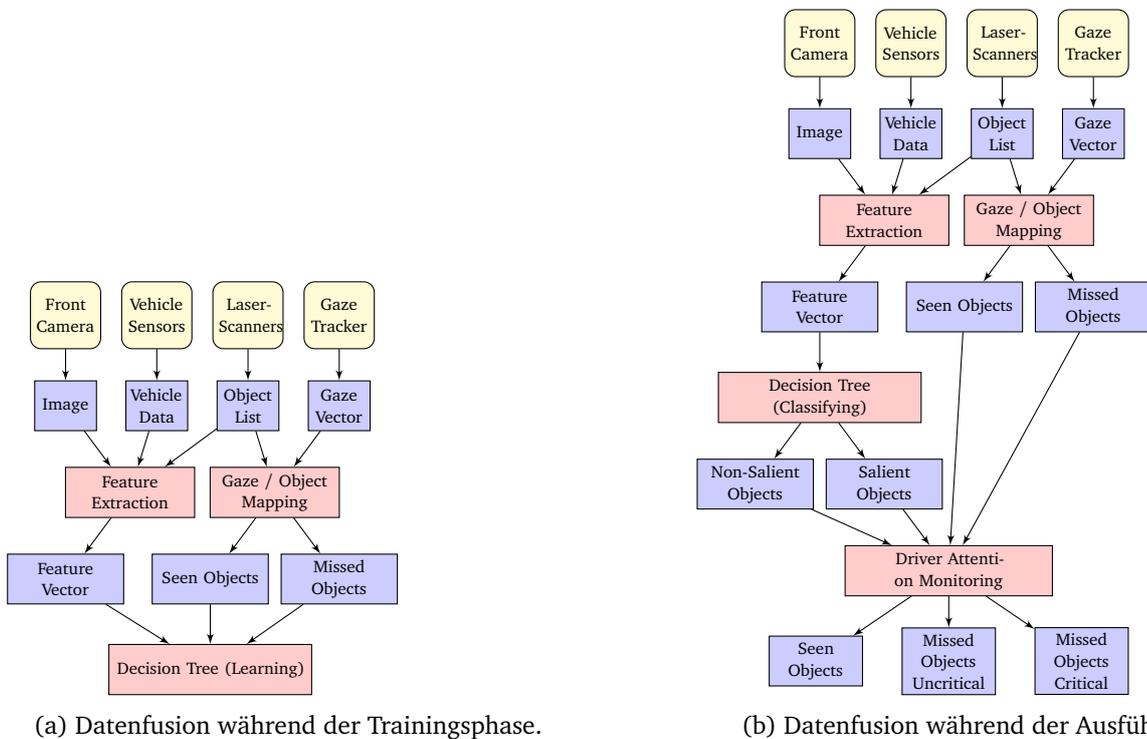


Abb. 4.3.: Während der Trainings- sowie der Ausführungsphase werden die Daten der Laserscanner, der Szenekamera und der Fahrzeugsensoren fusioniert und ein Merkmalsvektor wird gebildet.

In der Trainingsphase (Abbildung 4.3a) wird dieser Merkmalsvektor mit einem *[gesehen]* bzw. *[ungesehen]* Label annotiert, welches über die Blickrichtungserkennung abgeleitet wird. Über die Merkmale und das Label wird über einen Entscheidungsbaum gelernt, auf welche Objekte der Fahrer üblicherweise blickt.

Während der Fahrt (Abbildung 4.3b) wird für jedes Objekt das gelernte Blickverhalten mit dem momentan beobachteten Blickverhalten verglichen, um festzustellen, ob das Blickverhalten des Fahrers vom üblichen Verhalten abweicht. Die Objekte der Umgebung werden so als *[gesehen]*, *[ungesehen]* oder *[ungesehen salient]* eingestuft.

Während der Fahrt wird das gelernte Blickverhalten mit dem momentan beobachteten Blickverhalten

objektspezifisch verglichen. Weicht das Blickrichtungsverhalten zur Fahrzeit vom gelernten Blickverhalten ab, wird das Objekt als *[ungesehen - salient]* eingestuft. Insgesamt werden so die Objekte der Umgebung als *[gesehen]*, *[ungesehen]* oder *[ungesehen salient]* eingestuft. Abbildung 4.3b zeigt wie das Blickrichtungsverhalten zur Fahrzeit mit dem gelernten Blickverhalten verglichen wird.

4.5. Prognose zukünftiger Manöver und Aktionen

Die bisher beschriebene Analyse des Fahrerverhaltens zielte darauf ab, den Fahrer in seinem Verhalten möglichst genau zu beschreiben. In diesem Abschnitt sollen die bestimmten Kennwerte und Modelle genutzt werden, um zusammen mit den Geschehnissen im lokalen Fahrzeugumfeld Aussagen über die zukünftigen Manöver und Aktionen des Fahrers zu treffen.

Definition 4.5. Ein *(Fahr-)Manöver* wird in dieser Arbeit als eine zeitlich begrenzte Sequenz von *Aktionen* verstanden, über die der Fahrer seine Fahraufgabe bewältigt. Ein *(Fahr-)Manöver* ist dabei eine *bewusste, zielgerichtete* Handlung des Fahrers. Beispiele für Manöver sind *[Rechts Abbiegen]* oder *[Überholen]*.

Vergleichbare Definitionen bzw. Auffassungen des Begriffs „Manöver“ finden sich zum Beispiel in [\[Mic85; DBB03; TT11\]](#).

Definition 4.6. Eine *Aktion* wird vom Fahrer bewusst oder unbewusst ausgeführt. Im Gegensatz zu einem Manöver (Definition 4.5) lässt sich eine Aktion nicht weiter unterteilen. In dieser Arbeit wird vorausgesetzt, dass eine Aktion eine messbares Ereignis ist.

Mit der Definition des Manövers als eine bewusste, zielgerichtete Handlung, lässt sich das Manöver nach dem Rubikon-Modell, einem Modell aus der Psychologie, in vier Handlungsphasen aufteilen [\[HG87\]](#):

Abwägen: In der Abwägungsphase wägt der Fahrer Handlungsmöglichkeiten ab, die in der momentanen Verkehrssituation in Betracht kommen. Ein Algorithmus, der die Handlungsmöglichkeiten des Fahrers abschätzt und die Handlung auswählt, welche mit hoher Wahrscheinlichkeit auch der Fahrer wählen würde, wird als *Intentionserkennung* bezeichnet.

Planen: Der Fahrer plant die Realisierung der Handlung und bereitet sich auf das Handeln vor. Es geht nicht mehr darum *was* der Fahrer erreichen will, sondern *wie*.

Handeln: In der Handlungsphase (aktionale Phase) führt der Fahrer Aktionen aus und setzt das geplante Manöver in die Tat um.

Bewerten: Das Manöver wird abgeschlossen und reflektiert.

Abbildung 4.4 zeigt das Rubikon-Modell und die zeitlichen Zusammenhänge der Handlungsphasen. Eine Manöverprädiktion zielt darauf ab, die aktionale Handlungsphase (*den Zustand Handeln*) möglichst früh zu erkennen.

Die Handlungsphasen des Fahrers sind jedoch nicht direkt beobachtbar bzw. messbar. Man spricht in diesem Zusammenhang auch von *verborgenen Zuständen (Hidden States)*. Beobachtbar sind jedoch *Evidenzen*,

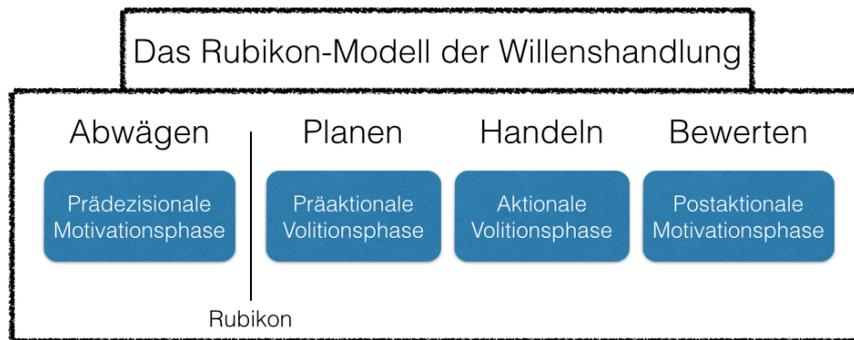


Abb. 4.4.: Das Rubikon-Modell teilt eine bewusste Handlung des Menschen in vier Phasen auf: *Abwägen, Planen, Handeln* und *Bewerten*.
Quelle: [Ker14]

welche in probabilistischer Form auf eine Handlungsphase hinweisen. Führt der Fahrer auf der Autobahn beispielsweise eine laterale Fahrzeugbewegung nach links aus (messbares Ereignis) ist es wahrscheinlich, dass er sich in der aktionalen Phase eines Überholvorgangs befindet. Eine hundertprozentige Sicherheit für diesen Zusammenhang gibt es jedoch nicht (zum Beispiel könnte er eingeschlafen sein und das Fahrzeug unabsichtlich nach links steuern).

Mit dem Blick auf die beschriebenen Handlungsphasen ergibt sich für die Assistenz des Fahrers ein Dilemma: Ziel eines Fahrerassistenzsystems ist es, in der Abwägungs- oder der Planungsphase zu assistieren. Maschinell erkennbare Evidenzen treten in der Interaktion mit dem Fahrzeug jedoch frühestens in der Handlungsphase auf, da der Fahrer erst zu diesem Zeitpunkt messbare Fahraktionen ausführt.

Für ein frühzeitiges Erkennen von Manövern ist es demnach unabdingbar, die Geschehnisse im lokalen Fahrzeugumfeld sowie die Aktionen des Fahrers mit einzubeziehen, über die sich der Fahrer in der Abwägungsphase Informationen beschafft und die nicht primär der Steuerung des Fahrzeugs dienen. Eine dieser Aktionen wäre beispielsweise der Schulterblick zur Beschaffung der Belegtheitsinformation der (Ziel-)Fahrspur.

Mathematisch lässt sich dieser Zusammenhang von verborgenen, nicht messbaren Zuständen und beobachtbaren Evidenzen als Indikatoren für diese Zustände über ein *Hidden Markov Modell (HMM)* modellieren (siehe Abbildung 4.5). Das *HMM* bildet diesen doppelt-stochastischen Prozess ab, indem es die Wahrscheinlichkeiten der Zustandsübergänge ($p(X_{t+1}|X_t)$) und die Wahrscheinlichkeiten von Emissionen gegeben des Zustands ($p(y_t|X_t)$) in Beziehung setzt [Rab89]. Mit der Modellierung als *HMM* lässt sich die aktuelle Handlungsphase bei einer bestimmten Sequenz von Evidenzen über den *Viterbi-Algorithmus* bestimmen. Eine Hypothese über die zukünftige Entwicklung der Zustände lässt sich über den *Forward-Algorithmus* bestimmen. Die Modellparameter (Zustandsübergangswahrscheinlichkeiten und Emissionenswahrscheinlichkeiten) eines *HMMs* können über den *Baum-Welch-Algorithmus* gelernt werden.

Ein *HMM* lässt sich über die in dieser Arbeit verwendeten *Markov Logik Netze* über wenige prädikatenlogische Formeln abbilden. Sumner und Domingos zeigt in [SD10] die Überführung eines *HMMs* in ein

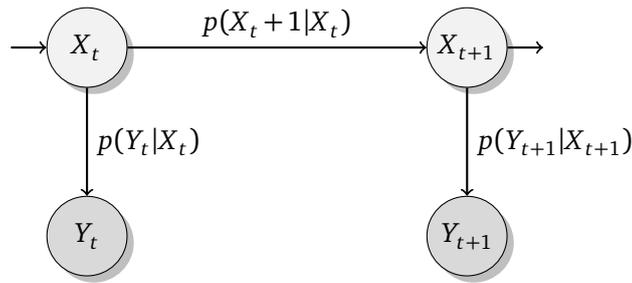


Abb. 4.5.: Ein *HMM* modelliert einen doppelt-stochastischen Prozess. Als Knoten des Diagramms sind die versteckten, nicht beobachtbaren Zustände X und die beobachtbaren Emissionen Y dargestellt. Die Wahrscheinlichkeit $p(X_{t+1}|X_t)$ eines Zustandswechsels von X nach X_{t+1} ist als waagrechte Kante dargestellt. Die Wahrscheinlichkeit $p(Y_t|X_t)$ einer Emission Y_t im Systemzustand X_t ist als senkrechte Kante dargestellt. Die Wahrscheinlichkeit $p(X_t = x|O)$, dass sich das System zum Zeitpunkt t bei einer Beobachtungssequenz von $O = [y_0, \dots, y_t]$ im Zustand x befindet, kann über den *Forward-Algorithmus* bestimmt werden.

MLN über die Formeln:

$$\forall t \exists! s : \text{State}(s, t) \quad [4.16]$$

$$\forall t \exists! o : \text{Observation}(o, t) \quad [4.17]$$

$$\forall t \forall s_1 \exists s_2 : \text{State}(s_1, t) \Rightarrow \text{State}(s_2, t+1) \quad [4.18]$$

$$\forall t \forall o \exists s : \text{Observation}(o, t) \Rightarrow \text{State}(s, t) \quad [4.19]$$

$$\exists! s : \text{State}(s, 0) \quad [4.20]$$

Die Formeln 4.16 bzw. 4.17 beschreiben, dass sich das System zu jedem Zeitpunkt in genau einem Zustand befindet bzw. dass für jeden Zeitpunkt genau eine Observation existiert. Formel 4.18 besagt, dass jeder Zustand in einen Nachfolgezustand übergeht und Formel 4.19 formuliert, dass jede Beobachtung auf einen Zustand hinweist. Die Tatsache, dass es einen Startzustand gibt wird über Formel 4.20 ausgedrückt.

Die Überführung in ein *MLN* bringt (neben der einfachen und leserlichen Formulierung) zwei wesentliche Vorteile für die Manövererkennung und -prädiktion mit:

Softe Evidenzen: In einem *HMM* wird vorausgesetzt, dass eine Beobachtung zweifelsfrei ist. Mit einer Beobachtung der Umwelt über Sensoren ist eine Beobachtung jedoch immer wahrscheinlichkeitsbehaftet. *MLNs* bieten über softe Evidenzen die Möglichkeit, Beobachtungen selbst schon probabilistisch zu gewichten.

Fehlende Observation: In *HMMs* wird vorausgesetzt, dass zu jedem Zeitpunkt eine Beobachtung existiert. Inferiert man mittels *MLN*, bieten diese die Möglichkeit, fehlende Beobachtungen zu kompensieren. Beispielsweise kann bei der Inferenz die Wahrscheinlichkeit des Zustands zum Zeitschritt 5 über die Beobachtungssequenz o_0, o_1, o_5 bestimmt werden. Die Beobachtungen o_2, o_3, o_4 würden in diesem Fall während der Inferenz mit geschätzt werden.

Weitere Vorteile, welche sich zum Teil auch in *HMMs* oder *DBNs* (als Verallgemeinerung der *HMMs*) realisieren lassen, sind:

Faktenwissen: *MLNs* bieten die Möglichkeit Faktenwissen transparent abzubilden. Beispielsweise kann die Tatsache, dass aus der Planungsphase nicht die Reflektionsphase folgen darf, wird über eine einzige Formel beschrieben werden:

$$\forall t : \text{State}(\text{Planen}, t) \Rightarrow \neg \text{State}(\text{Reflektion}, t + 1)$$

In *HMMs* würde dies durch eine Übergangswahrscheinlichkeit von Null erreicht werden.

Zerlegung der Observationen: Das Grundatom der Observation kann in spezifische Grundatome für die Observationen zerlegt werden, beispielsweise in eine Observation für die Blickrichtung und eine Observation für die Belegtheit der Ego-Fahrspur. Zusammenhänge können so spezifischer formuliert werden.

Leichtes Einbinden von Kontextinformationen: Die Formulierung der Zusammenhänge in prädikatenlogischen Formeln macht es möglich Kontextinformationen leicht einfließen zu lassen. Beispielsweise kann ein und dieselbe Beobachtung unterschiedlich eingeordnet werden, je nachdem ob sie auf der Autobahn oder auf der Landstraße beobachtet wird.

Prädiktionsfähigkeit: Soll die Situationsentwicklung prognostiziert werden, werden im Inferenzprozess die Observationen der Zeitpunkte t_0, \dots, t_k angegeben und die Wahrscheinlichkeit eines Zustands zum Zeitpunkt t_{k+n} angefragt (siehe fehlende Observation).

Ein Nachteil ist sicherlich die erhöhte Rechenzeit des Netzes bei der Inferenz. Im Inferenzprozess werden über den *MC-SAT* Algorithmus immer n neue Netzzustände (ausgehend von den definierten Formeln und ihren Gewichtungen) generiert, was für komplexe Netze großen Rechenaufwand darstellt.

4.6. Zusammenfassung

Die Analyse des Fahrerverhaltens zielt darauf ab, den Fahrer in seiner Interaktion mit dem Fahrzeug und dem lokalen Umfeld möglichst genau zu beschreiben und so *Wissen* über den Fahrer abzuleiten. Über das abgeleitete Wissen sind Assistenzsysteme in der Lage, ihre Algorithmen und Modelle speziell an den individuellen Fahrer anzupassen.

In der vorliegenden Arbeit wird das Verhalten des Fahrers in Form folgender Komponenten analysiert:

Fahrstilerkennung: Mit der Fahrstilerkennung wird das Fahrverhalten des Fahrers in einer Oberklasse, wie [*sportlich*] oder [*energiesparend*], klassifiziert. Der Fahrstil des Fahrers wird in dieser Arbeit in Bezug zur vorherrschenden Verkehrsszene geschätzt und hebt sich damit von einer vereinfachten Fahrstilklassifikation, welche ausschließlich die Aktionen des Fahrers betrachtet, ab. Zur Fahrstilerkennung wurden zwei Konzepte umgesetzt, welche sich stark in der Verwendung des vorhandenen Expertenwissens unterscheiden:

- In der Fahrstilerkennung über das *Fuzzy-Inferenzsystem* werden die Zugehörigkeitsfunktionen der Fuzzymengen sowie die Fuzzyregeln zur Inferenz vom Experten definiert. Durch die

- Definition von linguistischen Fuzzyregeln, die sich an der menschlich logischen Denkweise orientieren, entsteht ein transparentes und leicht erweiterbares System zur Fahrstilerkennung.
- In der über *Markov Logik Netze* umgesetzten Fahrstilerkennung wird weit weniger Expertenwissen benötigt. Eingabesignale werden unüberwacht, basierend auf dem *Bayesian Information Criterium (BIC)*, diskretisiert. Über Prädikatenlogik wird allgemein definiert, dass sich eine Beobachtung einem Fahrstil zuordnen lässt, wobei die Gewichtungen der prädikatenlogischen Formeln aus den Trainingsdaten gelernt werden.

Objektspezifische Aufmerksamkeitsbestimmung: Über eine objektspezifische Aufmerksamkeitsbestimmung werden alle Objekte des lokalen Fahrzeugumfelds als *[gesehen]*, *[ungesehen]* oder *[ungesehen salient]* eingestuft. Um zu lernen, welche Objekte die visuelle Aufmerksamkeit des Fahrers auf sich ziehen, werden die Daten der Szenekamera, der Laserscanner, des Fahrzeugs und der Blickrichtungserkennung fusioniert und so Objektmerkmale generiert. Auf Basis der Objektmerkmale wird ein gelerntes, übliches Blickverhalten mit dem momentan beobachteten Blickverhalten verglichen und Anomalien werden festgestellt.

Schätzung von Komfortparametern: Komfortparameter beschreiben den Fahrer quantitativ in Form von Parameter- und Intervallgrenzen in denen sich der Fahrer sicher und wohl (komfortabel) fühlt. Als Komfortparameter werden in dieser Arbeit der übliche Bremsdruck, die üblichen Geschwindigkeiten und Beschleunigungen in Form von statistischen Verteilungen modelliert. Die Schätzung setzt voraus, dass der Fahrer frei von Beeinflussungen anderer Verkehrsteilnehmer oder der Infrastruktur fährt, was über die Datenfusion aus Szenekamera-, Laserscanner- und GPS-Lokalisierungsdaten abgeleitet wird.

Manöverprädiktion: Nach dem psychologischen *Rubikon-Modell* durchläuft ein Fahrer für jede Manöverausführung die Verhaltenszustände *[Abwägen]*, *[Planen]*, *[Handeln]* und *[Bewerten]*. Die Manöverprädiktion zielt darauf ab, die aktionale Handlungsphase (den Zustand *[Handeln]*) des Fahrers möglichst früh zu erkennen. Die Verhaltenszustände sind nicht direkt beobachtbar oder messbar. Es sind lediglich Evidenzen beobachtbar, welche probabilistisch den Zuständen zuzuordnen sind. Dieser doppelt-stochastische Prozess wird in dieser Arbeit in einem *MLN* modelliert. Über das *MLN* ist es möglich, den aktuellen Zustand des Fahrers zu bestimmen und ein Eintreten der aktionalen Handlungsphase zu prognostizieren.

Mehrere der zur Analyse des Fahrerverhaltens realisierten Komponenten unterliegen der Grundstruktur eines *Markov Logik Netzes*. Eine Einführung ausgerichtet auf die Analyse des Fahrerverhaltens gab Abschnitt 4.1, indem folgende Vorteile eines *MLNs* identifiziert wurden:

- *MLNs* bieten die Möglichkeit, grobe Zusammenhänge in prädikatenlogischen Termen zu formulieren und die genauen Zusammenhänge über eine Gewichtung der Formeln zu lernen.
- In einem *MLN* können Unsicherheiten, bedingt durch die partielle Observation der Umwelt, in probabilistischer Art und Weise berücksichtigt werden.
- Die Modellierung temporaler Zusammenhänge ist möglich, was eine Prognose über zukünftiger Ereignisse erlaubt.
- Zusammenhänge lassen sich in prädikatenlogischen Termen formulieren.

5. Evaluation und Ergebnisse

Die Evaluation der in dieser Arbeit realisierten Komponenten ist strukturell an den Aufbau der Konzeptkapitel „Blickrichtungserkennung im Automobil“ (Kapitel 3) und „Analyse des Fahrerverhaltens“ (Kapitel 4) angelehnt.

Abschnitt 5.1 evaluiert die auf der Grundlage der CCD-Kamera und RGB-D-Kamera entwickelten Blickrichtungserkennungssysteme und geht dabei speziell auf die in Abschnitt 3.1 (Seite 3.1) formulierten Anforderungen im Fahrzeug ein.

Die realisierte Fahrstilerkennung wird in Abschnitt 5.2 anhand von Experimenten im FZI-Smart-Fahr-simulator evaluiert. Die Evaluation der Fahrstilklassifikation im Fahrsimulator hat zum Vorteil, dass alle Probanden die gleiche Teststrecke unter exakt gleichen Umständen fahren. Die Messungen sind so direkt vergleichbar. Die Fahrstilklassen richten sich nach der Risikobereitschaft des Fahrers, aber auch nach dem Energieverhalten des Fahrers. Letzteres hebt sie von anderen, in der Literatur veröffentlichten Verfahren zur Fahrstilschätzung, ab. Weiterhin wird speziell auf die Unterschiede einer Klassifikation über das *Fuzzy-Inferenzsystem* und die *MLNs* eingegangen.

Die abgeleiteten Komfortparameter werden in Abschnitt 5.3 anhand des Anwendungsfalls der Energieprognose evaluiert. Dazu wird eine durch die Komfortparameter parametrisierte Verbrauchsprognose mit der einer Standardprognose verglichen. Darüber hinaus werden für den Autobahnabschnitt von Karlsruhe nach Rastatt während der Hinfahrt im FZI-Versuchsträger CoCar die Komfortparameter des Fahrers gebildet und anhand dieser der Energieverbrauch des Rückwegs prognostiziert.

Abschnitt 5.4 widmet sich der Evaluation der über ein *MLN* realisierten Zustands- und Manövererkennung. Als Anwendungsfall dient hier eine Autobahnfahrt, auf der die aktuelle Handlungsphase des Fahrers und die bevorstehenden Überholvorgänge vorhergesagt werden.

Eine Zusammenfassung und Bewertung der erzielten Ergebnisse gibt abschließend Abschnitt 5.5.

5.1. Blickrichtungserkennung hinsichtlich den Ansprüchen der Automobilindustrie

Zur Blickrichtungserkennung wurden in dieser Arbeit zwei Systeme umgesetzt. Eines auf der Sensorgrundlage der CCD-Kamera und das andere auf der Sensorgrundlage einer RGB-D-Kamera. In ihrer Evaluation beziehen sich die Abschnitte 5.1.1 bis 5.1.4 auf die Blickrichtungserkennung, realisiert über das RGB-D-Kamerasystem. Abschnitt 5.1.5 bewertet das System, welches die Blickrichtungserkennung über eine CCD-Kamera schätzt.

Zentrale Kennzahlen für die Tauglichkeit einer Blickrichtungserkennung sind ihre Genauigkeit und ihr Erkennungsbereich. Diese Größen werden in Abschnitt 5.1.1 anhand einer von der ETH-Zürich bereitge-

stellten Datenbank evaluiert und mit vergleichbaren Arbeiten aktueller Veröffentlichungen verglichen. Die Datenbank der ETH-Zürich bietet 24 Aufnahmesequenzen mit weiträumigen Kopfbewegungen auf mehr als 15.000 Bildern. Die Kopfstellung ist auf dieser Datenbank mit einer Genauigkeit von 1° annotiert.

Durch die Initialisierung der Kopfposenschätzung mit der letzten erkannten Kopfpose und der Verschlusszeit der Kamera entsteht eine Limitierung hinsichtlich der von der Kopfposenschätzung verarbeitbaren Bewegungsschnelligkeit des Kopfes. Abschnitt 5.1.2 zeigt experimentell, dass die entworfene Methode für schnelle und weiträumige Kopffrotationen, wie sie beispielsweise beim Einparken getätigt werden, tauglich ist.

Die eigentliche Blickrichtung wird unter Berücksichtigung der Augenstellung aus dem RGB-Farbbild der RGB-D-Kamera geschätzt. Abschnitt 5.1.3 zeigt, dass unter Hinzunahme der Augenstellung der Blick des Fahrers auf Sichtfelder innerhalb und außerhalb des Fahrzeugs abgebildet werden kann. Gleichzeitig wird gezeigt, dass alleine die Kopfstellung für diese Zuordnung nicht ausreicht.

Eine Blickrichtungserkennung muss im Automobilbetrieb vor allem unter wechselnden Lichtverhältnissen funktionsfähig sein. Ist das Tiefenbild der RGB-D-Kamera durch starkes Gegenlicht oder Überblendungen fehlerhaft, schaltet der Algorithmus auf die Kopfposenerkennung über das Farbbild um (siehe Abschnitt 3.2.4, Seite 58). Abschnitt 5.1.4 evaluiert die Genauigkeitseinbußen, welche durch den Ausfall des Tiefenbilds zu erwarten sind.

In Abschnitt 5.1.5 wird die Genauigkeit der Kopfposenbestimmung über die CCD-Kamera untersucht und dabei speziell die Lokalisationsgenauigkeit der in Abschnitt 3.3.1 (Seite 66) aufgeführten Gesichtsmarkerevaluierung evaluiert. Die Limitierungen des Systems werden qualitativ anhand einer Fahrt unter realen Bedingungen bewertet.

5.1.1. Genauigkeit und Erkennungsbereich der RGB-D-Kopfposenschätzung

Um die Genauigkeit der Kopfposenschätzung zu untersuchen, wird auf die *Biwi Kinect Head Pose Database* der ETH-Zürich von Fanelli et al. zurückgegriffen, die im Zusammenhang mit der im Stand der Technik vorgestellten Veröffentlichung [Fan+11] entstand. Die Datenbank beinhaltet 24 Videosequenzen 20 unterschiedlicher Testpersonen (6 Frauen, 14 Männer), welche in einer Laborumgebung in einem Abstand von circa einem Meter zu einer RGB-D-Kamera weiträumige Kopfbewegungen (Nicken $\pm 75^\circ$, Wanken $\pm 60^\circ$, Gieren $\pm 50^\circ$) ausführen. Die Datenbank besteht aus insgesamt 15.287 Einzelbildern, welche jeweils mit einer Kopfpose relativ zum Kamerasystem annotiert sind. Die Annotation der Einzelbilder wird mit einer Rotationswinkelabweichung von weniger als einem Grad und einer Positionsabweichung von unter einem Millimeter angegeben. Abbildung 5.1 zeigt typische Bilder der Datenbank.

In Abbildung 5.2 wird die Genauigkeit der Kopfposenbestimmung mit aktuellen Veröffentlichungen verglichen. Die Genauigkeit der ermittelten Kopfstellung ist auf der y -Achse und die tolerierte Abweichung um die Gierachse ist auf der x -Achse aufgetragen. Mit einer tolerierten Abweichung von 4° lässt sich die Kopfpose in 82% aller Fälle richtig bestimmen.

Verglichen zum Verfahren von Fanelli et al., welches auf *Random Forests* basiert und im Stand der Technik in Abschnitt 2.1.3 vorgestellt wurde, ermittelt das in dieser Arbeit entwickelte *Multi-Template-ICP-Verfahren* die Kopfpose je nach Toleranz um 13% bis 6% präziser. Das nach dieser Arbeit entstandene



Abb. 5.1.: Beispielbilder aus der *Biwi Kinect Head Pose Database* der ETH-Zürich [Fan+11]. Die Versuchspersonen sitzen in einer Laborumgebung circa einen Meter vor einer RGB-D-Kamera und bewegen den Kopf in weiträumigen Bewegungen. Die Datenbank besteht aus insgesamt mehr als 15-tausend Einzelbildern, welche mit einer Genauigkeitsabweichung von unter einem Grad in Kopfrotation und unter einem Millimeter in Kopfposition annotiert sind.

Quelle: Datenbank [Fan+11]

Verfahren von Martin basiert wie diese Arbeit auf dem ICP-Verfahren. Es erreicht auf den im Labor aufgenommenen Messdaten von Fanelli et al., welche keine wechselnden Lichtverhältnisse beinhalten, eine drei bis fünf Prozent höhere Erkennungsrate.

Die erzielte Genauigkeit hängt jedoch stark mit der relativen Kopfstellung zur Kamera ab. Die Abbildungen in 5.3 zeigen die erzielten Genauigkeiten, aufgetragen über Gier- und Nickwinkel der annotierten Kopfstellung. Die Genauigkeit der Kopfposenerkennung ist als Prozentangabe in den einzelnen Feldern zu sehen. Die Anzahl der für diese Kopfstellung berücksichtigten Einzelbilder ist farblich visualisiert.

Die Diagramme zeigen, dass der entworfene Algorithmus ab dem 5° Toleranzbereich selbst für starke Rotationen eine hohe Genauigkeit aufweist. Ist das Gesicht direkt zur Kamera gewandt, wird die Kopfpose mit einer Gierratenabweichung von 5° in 91% der Fälle korrekt erkannt.

5.1.2. Schnelle Kopfbewegungen

Schnelle Kopfbewegungen führen sowohl im RGB-Farbbild als auch im Tiefenbild zu Bewegungsunschärfe. Im RGB-Farbbild verwischen Farben und Kontraste; im Tiefenbild entstehen Löcher und Ungenauigkeiten. Für die Initialisierung des ICP-Schritts bedeutet eine schnelle Kopfbewegung außerdem, dass die zuletzt geschätzte Kopfpose deutlich von der momentanen Kopfpose abweicht. Dies führt unter Umständen zu einer Divergenz des ICP-Schritts. Ein negativer Einfluss dieser Effekte ist bis zu einer Rotationsgeschwindigkeit des Kopfs von bis zu $600^\circ/s$ nicht zu erkennen.

Grossman et al. messen in einer Studie zur menschlichen Bewegung eine maximale Kopfrotationengeschwindigkeit von $780^\circ/s$ [Gro+88]. Rotationsgeschwindigkeiten während der Fahrt fallen erheblich geringer aus. In der FZI-Messdatenbank, welche Stadtfahrten, Einparkmanöver oder Rechts-Vor-Links-Situationen beinhalten, wurden maximale Rotationsgeschwindigkeiten von $500^\circ/s$ gemessen.

Abbildung 5.4 zeigt die Rotationsgeschwindigkeit des Kopfs bei sehr schnellen Kopfbewegungen um die Gierachse. Die Messung zeigt einen Fahrer, der seinen Kopf von links nach rechts bewegt. Zunächst sind die Bewegungen langsam und werden dann immer schneller. Am steigenden Nickwinkel als auch an der steigenden X- und Z-Position des Kopfs relativ zur Kamera ist zu erkennen, dass der Fahrer mit zunehmender Schnelligkeit der Kopfbewegung eine aufrechtere Sitzposition einnimmt.

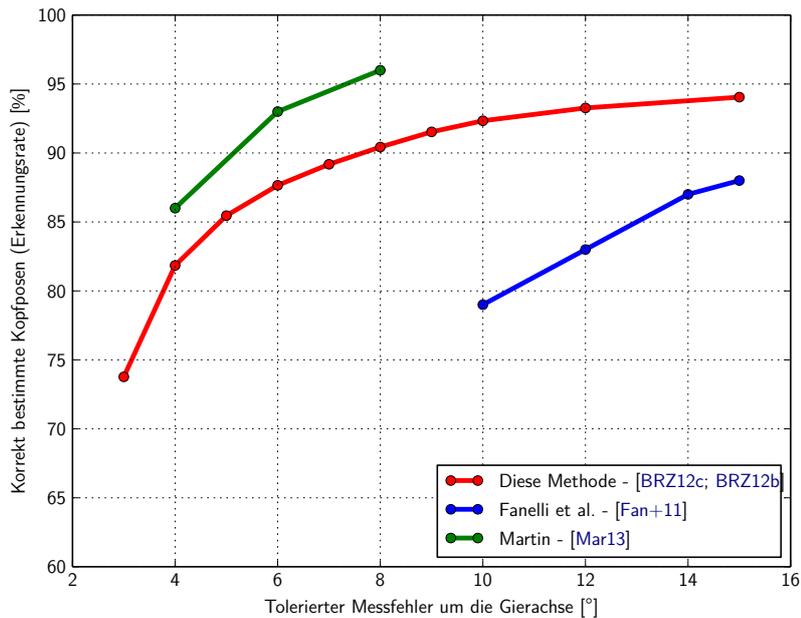


Abb. 5.2.: Korrektheit der Kopfposenbestimmung im Vergleich zu den Arbeiten [Mar13] und [Fan+11]. Die Korrektheit, als prozentualer Anteil der richtig erkannten Kopfposen, ist über den tolerierten Messfehler um die Gierachse aufgetragen. Die Kopfpose wird in 82% aller Messdaten richtig bestimmt, sofern eine Abweichung um die Gierachse von 4° toleriert wird.

Das von dieser Arbeit inspirierte Verfahren von Martin, welches ebenfalls auf dem ICP-Verfahren basiert, erreicht auf den im Labor aufgenommenen Messdaten von Fanelli et al. eine drei bis fünf Prozent höhere Erkennungsrate. Die für den Automobilbetrieb wichtige Kompensation von Fremdlicht sowie die Berechnung der Blickrichtung durch die Ermittlung der Pupillenstellung wird weder in der Arbeit von Martin noch im Verfahren von Fanelli et al. berücksichtigt.

5.1.3. Bestimmung des Aufmerksamkeitsfelds im Fahrzeuginnenraum

In einem weiteren Experiment wurde untersucht, inwiefern über die geschätzte Blickrichtung differenziert werden kann, auf welchen Bereich des Fahrzeugs der Fahrer seine Aufmerksamkeit richtet. In einer Messreihe wurden Fahrer instruiert, typische Bereiche des Fahrzeugs anzuschauen: [Tachometer], [linker Spiegel], [rechter Spiegel], [Innenspiegel] und [Windschutzscheibe]. Abbildung 5.5a zeigt die Stellung des Kopfs beim Betrachten der jeweiligen Sichtfelder, aufgetragen über den Gier- und den Nickwinkel. Aus der Abbildung wird ersichtlich, dass die Kopfstellung alleine nicht ausreicht, um den vom Fahrer anvisierten Bereich zu bestimmen. Die Kopfstellungen beim Betrachten des Innenspiegels (rot) und des rechten Außenspiegels (blau) überlappen sich stark und können nicht eindeutig voneinander abgegrenzt werden.

Unter Berücksichtigung der Augenstellung lässt sich der Blick in den *Innenspiegel* eindeutig vom Blick in den *rechten Außenspiegel* unterscheiden. Abbildung 5.5b zeigt die Blickrichtung, diesmal ermittelt aus Kopf und Augenstellung, beim Betrachten der jeweiligen Felder. Über die Modellierung der Aufmerksamkeitsfelder als Dreiecksnetze (siehe Abschnitt 3.4.1 Seite 72) lassen sich die Sichtbereiche nahezu verwechslungsfrei differenzieren.

Die Abbildungen 5.5c und 5.5d zeigen die *Konfusionsmatrizen* (oft auch *Wahrheitsmatrizen*) für die einzelnen Sichtfelder mit und ohne der Berücksichtigung der Augenstellung.

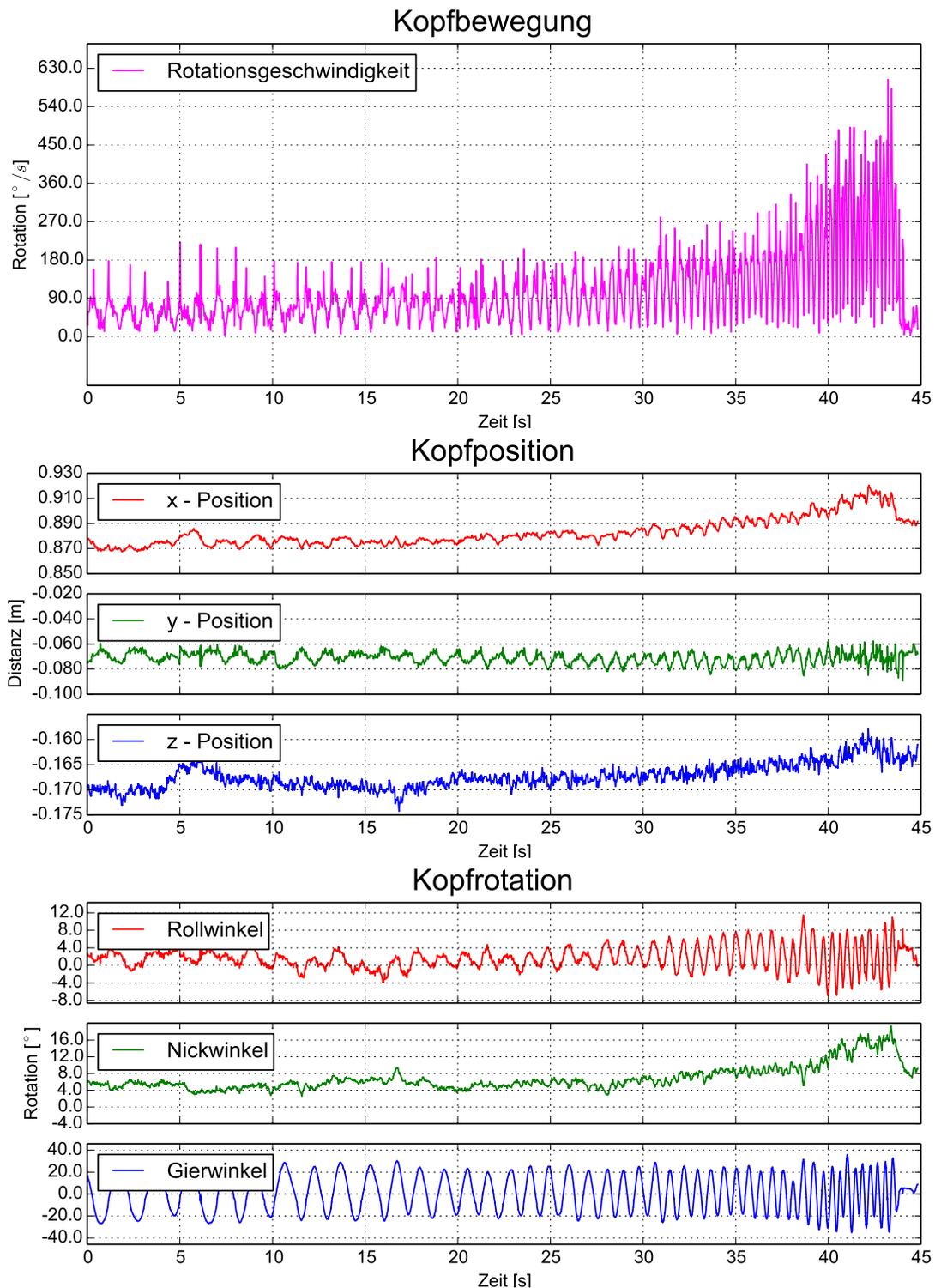


Abb. 5.4.: Zur Evaluation der vom Blickrichtungserkennungssystem maximal zu gewährleistenden Rotationsgeschwindigkeit wurde ein Fahrer instruiert, den Kopf mit steigender Geschwindigkeit um die Gierachse von links nach rechts zu schwenken. Die Kopfpose konnte bei Rotationsgeschwindigkeiten bis zu $600^\circ/\text{s}$ um die Gierachse zuverlässig bestimmt werden.

Definition 5.1. Eine Konfusionsmatrix stellt das Klassifikationsergebnis eines Klassifikators der tatsächlichen Klassenzugehörigkeit gegenüber. Die tatsächliche Klassenzugehörigkeit (englisch als *Ground Truth* bezeichnet) wird in dieser Arbeit immer in den Zeilen dargestellt. Das Klassifikationsergebnis wird in den Spalten aufgetragen. Die Matrix enthält auf der Hauptdiagonalen richtig klassifizierte Ergebnisse und in den anderen Feldern die auftretenden Verwechslungen. Zur einfacheren Vergleichbarkeit beinhalten die Zellen nicht die absoluten Werte sondern Prozentangaben. Gleichzeitig sind die Zellen entsprechend ihrem Zahlenwert eingefärbt.

Beispiel 5.1. In Abbildung 5.5d wird der Blick in den Innenspiegel in 98,2% aller Fälle als Blick in den Innenspiegel klassifiziert. 1,1% aller Blicke in den Innenspiegel werden fälschlicherweise als Blick auf die Windschutzscheibe klassifiziert. 3,0% aller Blicke in den rechten Spiegel werden fälschlicherweise als Blick in den Innenspiegel klassifiziert.

Vor allem der Blick in den linken Außenspiegel profitiert bei der Erkennung stark von der Berücksichtigung der Augenstellung.

Vergleicht man die Wertebereiche in den Abbildungen 5.5a und 5.5b zeigt sich, dass sich durch die Betrachtung der Augenstellung ein fast doppelt so hoher Winkelbereich für die Blickrichtung ergibt.

5.1.4. Kompensation der Überbelichtung durch Kopfposenschätzung aus dem RGB-Farbbild

Ist das Tiefenbild fehlerhaft, wird die Kopfpose über *Random Regression Forests* bestimmt (siehe Abschnitt 3.2.4 auf Seite 58).

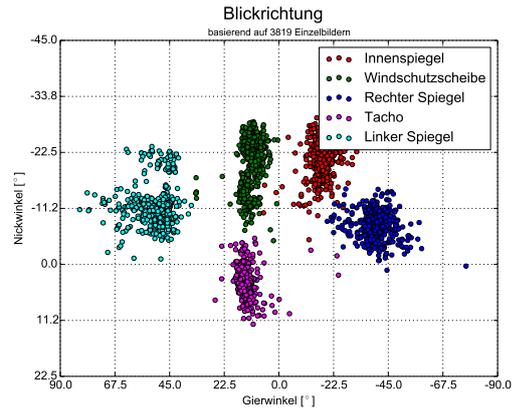
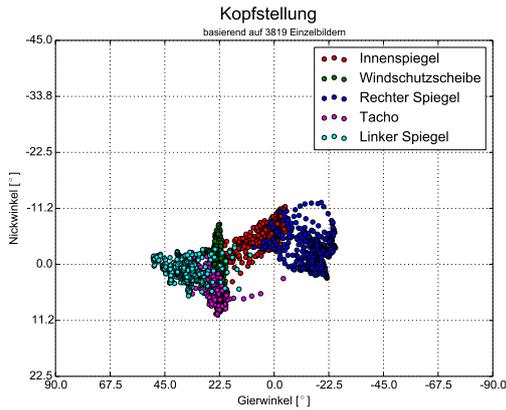
Abbildung 5.6 zeigt das typische Systemverhalten beim Wechsel zwischen der Kopfposenbestimmung über das Tiefenbild und über das RGB-Farbbild. Bei der Bestimmung der Kopfpose über *RFs* haben zwei Parameter entscheidenden Einfluss auf die Laufzeit und die Genauigkeit der Kopfposenbestimmung: Die Rasterdichte von Patches und die Anzahl von verwendeten Bäumen.

Als Eingabe der *RFs* dienen dem RGB-Farbbild entnommene Patches, welche aus der Gesichtsregion des RGB-Farbbilds in einem äquidistanten Raster extrahiert werden. Die Dichte der Rasterung ist dabei ein Kompromiss aus der Schnelligkeit und der Genauigkeit der Kopfposenerkennung. Eine zu dichte Rasterung erzeugt viele Patches und steigert die Genauigkeit, wohingegen eine grobe Rasterung die Geschwindigkeit steigert. Abbildung 5.7 (Seite 116) zeigt die Evaluation der Rasterdichte hinsichtlich der erreichten Genauigkeit und der erreichten Schnelligkeit der Erkennung. In der Praxis ist eine Rasterdichte von 3 Pixeln eine gute Wahl. Die Erkennungsleistung sinkt nach dieser Rasterdichte rapide. Gleichzeitig bringt eine Erhöhung der Rasterdichte keinen großen Laufzeitgewinn.

Weiter wirkt sich die Anzahl der verwendeten Bäume auf die Laufzeit und die Genauigkeit aus. Abbildung 5.7b zeigt die Geschwindigkeit und die Laufzeit, aufgetragen über der Anzahl der verwendeten Bäume. Mit einer Anzahl von 6 Bäumen wird auf einem 4-Kern Intel® Core™ i7-2700K CPU @ 3.50GHz Prozessor eine Erkennungsrate von 94,7% bei einer Laufzeit von 23,66 ms pro Bild (etwa 40 Bilder pro Sekunde) erreicht. Abbildung 5.8 zeigt die Evaluation der *RFs* zur Kopfposenschätzung, aufgetragen über Winkelbereiche von jeweils 10×10 Grad. Zur Evaluation wurden zwei Messreihen untersucht:

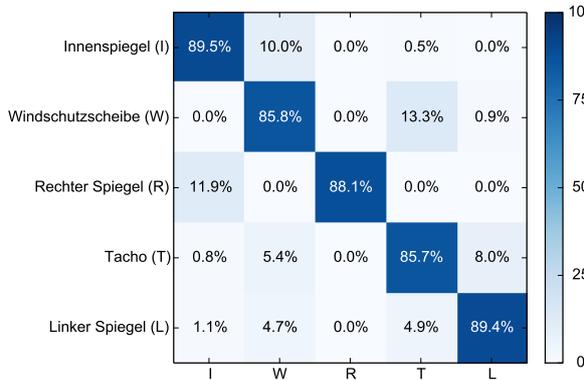
Eine erste Messreihe umfasst eine Autobahnfahrt, bei der die Kopfposen des Fahrers nur gering variieren.

5. Evaluation und Ergebnisse

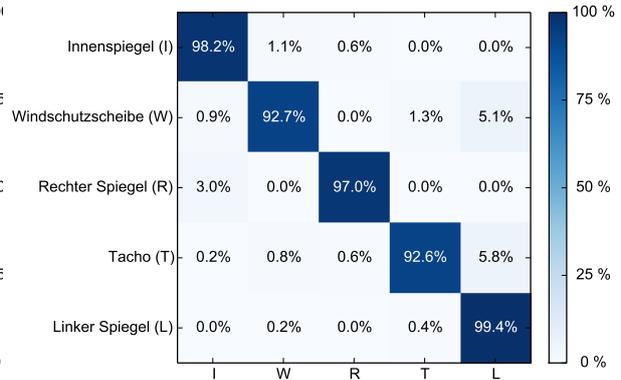


(a) Die Kopfstellung des Fahrers beim Betrachten verschiedener Aufmerksamkeitsfelder. Alleine durch die Kopfstellung lässt sich nicht fehlerfrei unterscheiden, ob der Fahrer in den rechten Außenspiegel oder den Innenspiegel blickt. Auch die Kopfposen für einen Tachometerblick, den Blick nach vorne und den Blick in den linken Außenspiegel überlappen sich stark.

(b) Über die Blickrichtung, gewonnen aus Kopf- und Augenstellung, lässt sich der Blick des Fahrers eindeutig den Sichtfeldern zuordnen. Der durch die Augen aufgespannte Winkelbereich ist mehr als doppelt so groß als der Winkelbereich, der alleine durch die Kopfstellung aufgespannt wird.



(c) Konfusionsmatrix der Sichtfeldbestimmung ohne Berücksichtigung der Augenstellung.



(d) Konfusionsmatrix der Sichtfeldbestimmung unter Berücksichtigung der Augenstellung.



(e) Die Kopfstellung für den Blick zum Tachometer und dem Blick nach vorne ist sehr ähnlich. Ebenso wenig unterscheidet sich die Kopfstellung vom Blick in den Innenspiegel vom Blick in den rechten Außenspiegel. Durch die Betrachtung der Augenstellung lassen sich die Aufmerksamkeitsfelder eindeutig bestimmen.

Abb. 5.5.: Über die entworfene Blickrichtungsschätzung lässt sich die Blickrichtung des Fahrers nahezu verwechslungsfrei Sichtfeldern im Fahrzeuginnenraum zuordnen. Selbst Spiegelblicke, welche in der Regel ohne eine Änderung der Kopfpose und nur durch eine kurze Bewegung der Pupillen getätigt werden, lassen sich erkennen.

Die in den Abbildungen 5.5c und 5.5d stellen die Klassifikationsergebnisse in Form von Konfusionsmatrizen dar. In den Matrizen sind auf der Hauptdiagonalen die richtig klassifizierte Ergebnisse dargestellt. Die anderen Feldern zeigen auftretenden Verwechslungen (siehe auch Definition 5.1 und Beispiel 5.1).

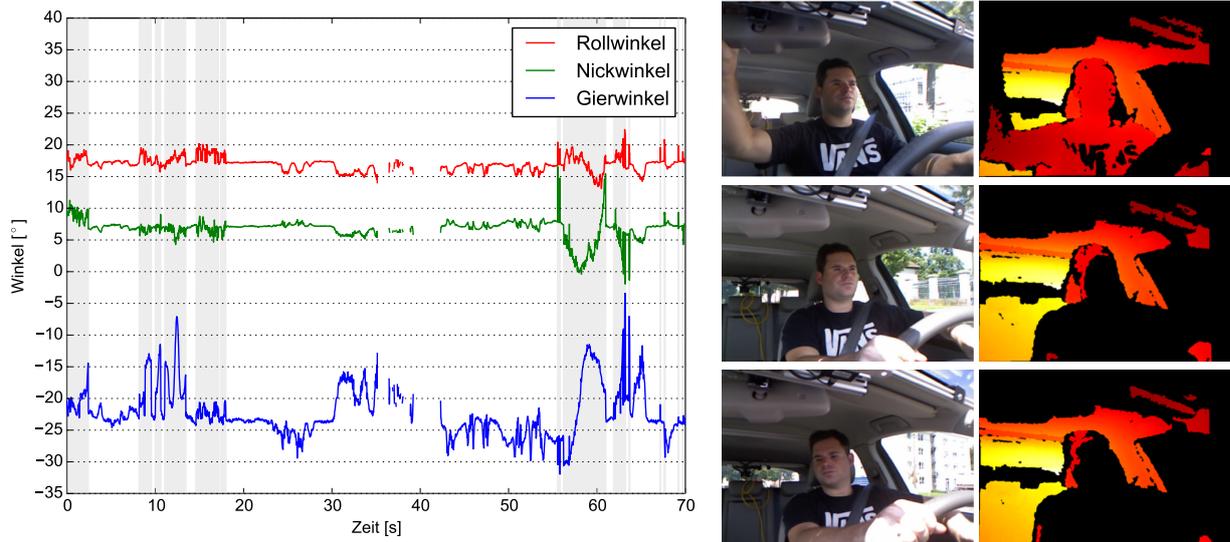


Abb. 5.6.: Die *RFs* werden über eine zur Laufzeit generierte Datenbank trainiert. Die Kopfpose wird dabei über die ICP-Methode geschätzt (wie zum Beispiel im oberen Bildpaar). Im weiß unterlegten Bereich des Graphen ist das Tiefenbild fehlerhaft und die Kopfpose wird über *RFs* aus dem RGB-Farbbild bestimmt (siehe mittleres Bildpaar als Beispiel). Starke Kopffrotationen sind in dieser Datenbank zunächst unterrepräsentiert. Demzufolge scheitert die Schätzung der Kopfpose über *RFs* bei sehr starken Kopffrotationen (wie beispielsweise im unteren Bildpaar).

Kopfposen mit starken Rotationen sind in dieser Messreihe stark unterrepräsentiert. Abbildung 5.8a zeigt links die Positionsabweichung der lokalisierten Nasenposition in Relation zur Größe des Gesichts von circa 110×140 Pixel. Die mittlere und die rechte Abbildungen in 5.8a zeigen die durchschnittlichen Abweichungen des Gier- bzw. Nickwinkels für die jeweiligen Winkelbereiche. In Winkelbereichen, in denen genügend Messdaten vorhanden sind (dunkelblau), erreicht die Kopfposenschätzung präzise Ergebnisse. Die Kopfpose kann in diesem Bereich bei einer Autobahnfahrt mit circa 2° Abweichung bestimmt werden.

In einer zweiten Messreihe wurden absichtlich starke Kopffrotationen des Fahrers aufgezeichnet. Der zentrale Bereich beinhaltet dennoch mehr Messdaten als die Randbereiche, da der Fahrer den Kopf immer wieder durch die Mitte bewegt. Die Abbildungen in 5.8b zeigen die Erkennungsraten für die Nasenposition und die durchschnittlichen Abweichungen des Gier- bzw. Nickwinkels. Es ist zu erkennen, dass die Genauigkeit mit steigender Kopffrotation abnimmt. Gleichzeitig lässt sich erkennen, dass die Kopfposenschätzung im zentralen Bereich präzise Ergebnisse liefert. Die durchschnittliche Abweichung bei Nick- und Gierwinkel liegt dort unter 6° im Mittel.

5.1.5. Schätzung der Kopfpose über CCD-Kamerabilder

Um die Genauigkeit der Lokalisation der Gesichtsmerkmale zu prüfen, wurden in zwei Videosequenzen von je 300 Einzelbilder per Hand die Pixel der Gesichtsmerkmale annotiert und die resultierende Kopfposition über den *POSIT-Algorithmus* bestimmt. Aus mehreren untersuchten Kamerapositionen wurde als beste Position die im Bereich der Instrumentenanzeige identifiziert, da aus dieser das Gesicht des Fahrers nahezu frontal aufgenommen wird. Die Nasenlöcher, die Mundwinkel und die Augenwinkel können aus dieser Position sehr gut erkannt werden. Allerdings verdecken die Hände des Fahrers bei einem starken

Lenkeinschlag das Kamerabild.

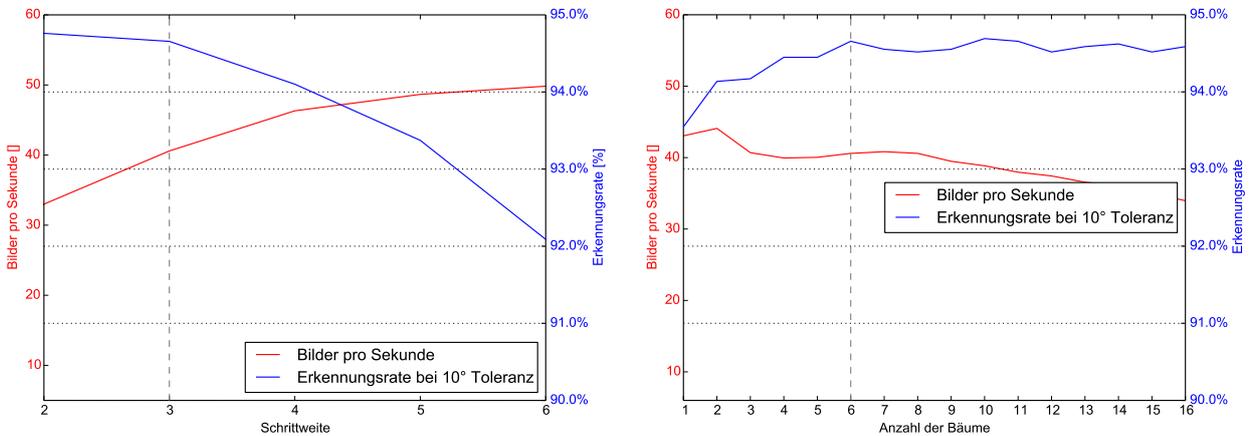
In den aufgenommenen Videosequenzen richtet der Fahrer seinen Blick hauptsächlich nach vorne. Blicke in den Rück- und Seitenspiegel führt der Fahrer mit einer Veränderung der Kopfpose aus.

Abbildung 5.9 stellt die Abweichung der vom Experten markierten Position gegenüber der vom Blickrichtungserkennungssystem erkannten Position mit entsprechender Varianz dar. Eine Abweichung von 30 Pixeln entspricht ungefähr dem Durchmesser der Iris ($\approx 14\text{mm}$). Es ist zu beobachten, dass die Abweichung durch die Rotation des Kopfs bei einem Blick in den Spiegel zunimmt (siehe Abbildungen 5.9a und 5.9b). Besonders stark ist die Abweichung des rechten Ohrs (O1). Dies liegt daran, dass die Ohren je nach Kamerastellung nicht erkannt werden können. In diesem Fall wird die Position des Ohrs anhand der Anordnung der anderen Merkmale geschätzt. Die Erkennung der Position des linken Ohrs ist deshalb besser, weil die Kopfpose des Fahrers sich beim Fahren meist auf den rechten Fahrbahnrand richtet.

Die Auswirkungen der Abweichungen der Gesichtsmerkmale auf die Kopfpose zeigt Abbildung 5.10.

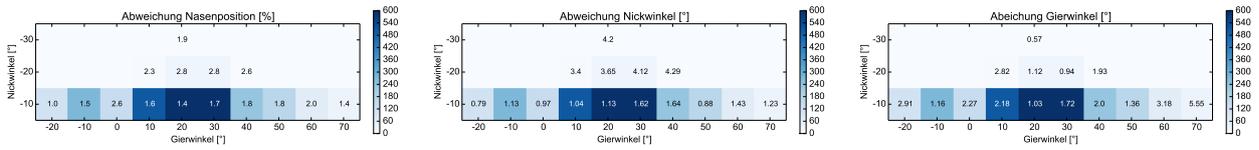
Abbildung 5.11 zeigt die Abweichungen der maschinell erkannten Pupillenmittelpunkte. In horizontaler Richtung (X-Richtung) ist die Abgrenzung der Iris von der Lederhaut durch eine klare Kante erkennbar, während in vertikaler Richtung (Y-Richtung) die Iris vom Augenlid mit einer unscharfen Kante abgegrenzt wird. Deshalb sind die Abweichungen in Y-Richtung größer als die Abweichungen in X-Richtung.

Abbildung 5.12 zeigt qualitativ das Systemverhalten während einer Landstraßenfahrt. Vorwiegend wird der Sichtbereich vor dem Fahrzeug und im Innenspiegel kontrolliert. In Bild 150 ist die Rotation des Kopfs zu stark. Die Gesichtsmerkmale können nicht mehr zuverlässig erkannt werden und die Blickrichtungsbestimmung fällt kurzfristig aus. Auch bei der Durchquerung eines Tunnels (Bild 1580 bis 2150) fällt die Blickrichtungsschätzung über das RGB-Farbbild aus.

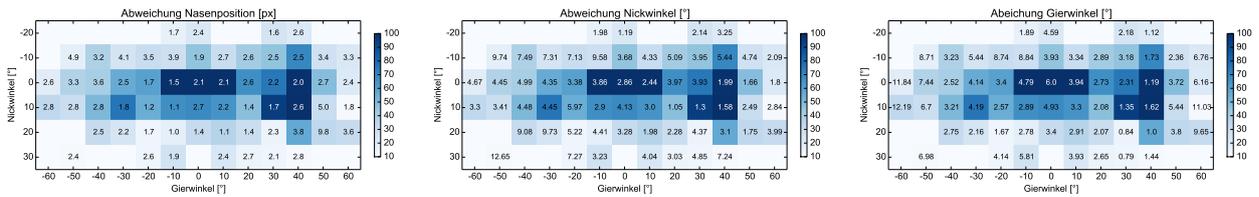


- (a) Beim Erzeugen der Patches ist die Rasterdichte wählbar. Eine große Rasterdichte wirkt sich positiv auf die Laufzeit der Erkennung aus. Ist die Rasterdichte allerdings zu groß gewählt, werden Gesichtsmerkmale übersprungen und die Erkennungsrate sinkt. Für die Auswertung wurde eine Anzahl von 6 Bäumen gewählt.
- (b) Auswirkungen der Anzahl der verwendeten Bäume auf die Laufzeit des Algorithmus. Mit einer zunehmenden Anzahl der Bäume steigt die Auswertungszeit pro Bild. Die Erkennungsrate steigt nach 6 Bäumen nur noch minimal. Für die Auswertung wurde eine Rasterdichte von 3 Pixel gewählt.

Abb. 5.7.: Laufzeit und Fehlerrate der Kopfposenerkennung über RFs, aufgetragen über verschiedene Rasterdichten und die Anzahl an trainierten Bäumen. Als Fehler ist eine Abweichung größer als 10° von der eigentlichen Kopfstellung definiert. Mit einer Rasterdichte von 3 Pixeln und einer Anzahl von 6 Bäumen wird eine Erkennungsrate von 94,7% bei einer Laufzeit von 40 Bildern pro Sekunde erreicht.

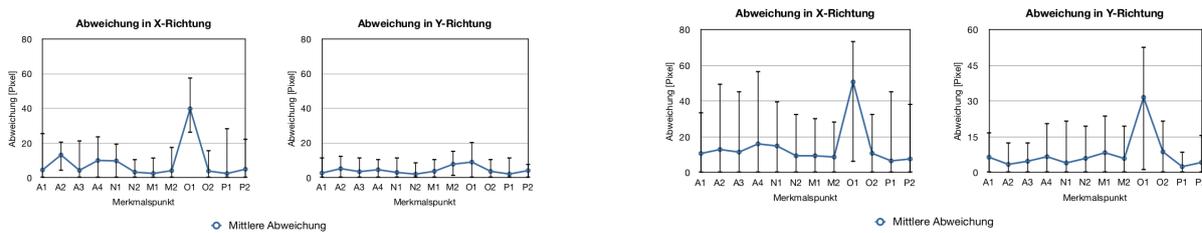


(a) Durchschnittliche Abweichung der Nasenposition, des Gier- und des Nickwinkels während einer Autobahnfahrt. Die Nasenposition ist prozentual zur Gesichtsgröße (circa 110×140 Pixel) angegeben. In Winkelbereichen in denen viele Trainingsbeispiele in der Lerndatenbank vorhanden sind (dunkelblau), kann die Nasenposition mit unter 3 Pixel Abweichung und die Kopfpose mit unter 2° Abweichung bestimmt werden.



(b) Durchschnittliche Abweichung der Nasenposition sowie des Nick- und Gierwinkels in einer Messreihe mit starken Kopftotationen. Die durchschnittliche Nick- und Gierwinkelabweichung beträgt $4,97^\circ$ bzw. $3,92^\circ$. Die Nasenposition kann auf durchschnittlich 2,44% genau bestimmt werden (gemessen an der Gesichtsgröße im Bild).

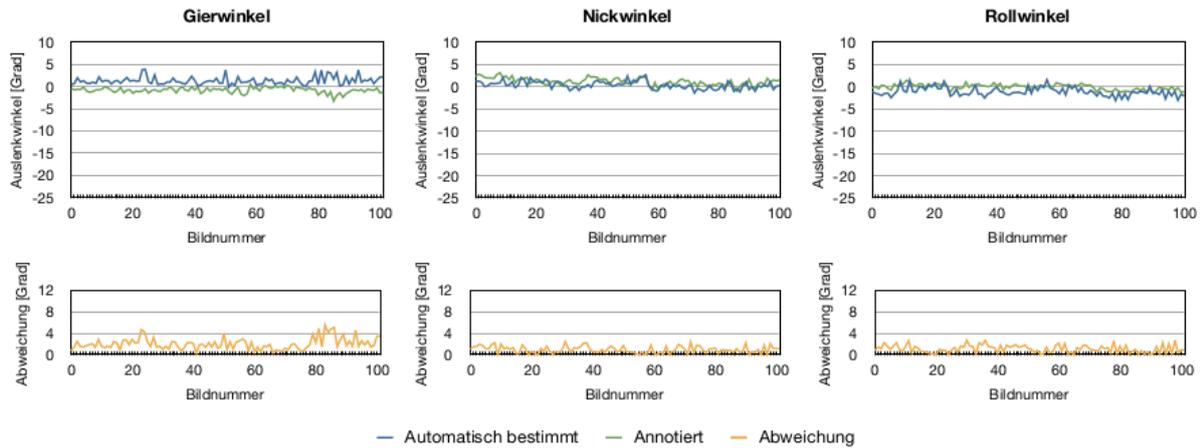
Abb. 5.8.: Positionsabweichungen der erkannten Nasenposition und Abweichung der Nick- und Gierwinkelschätzung der Kopfpose, aufgetragen über den Nick- und Gierwinkel der realen Kopfposition. Im Zentralbereich werden Winkelabweichungen von circa 2° gemessen. Bei stärkeren Kopftotationen verschlechtert sich die Abweichung auf mehr als 9° zur realen Kopfstellung.



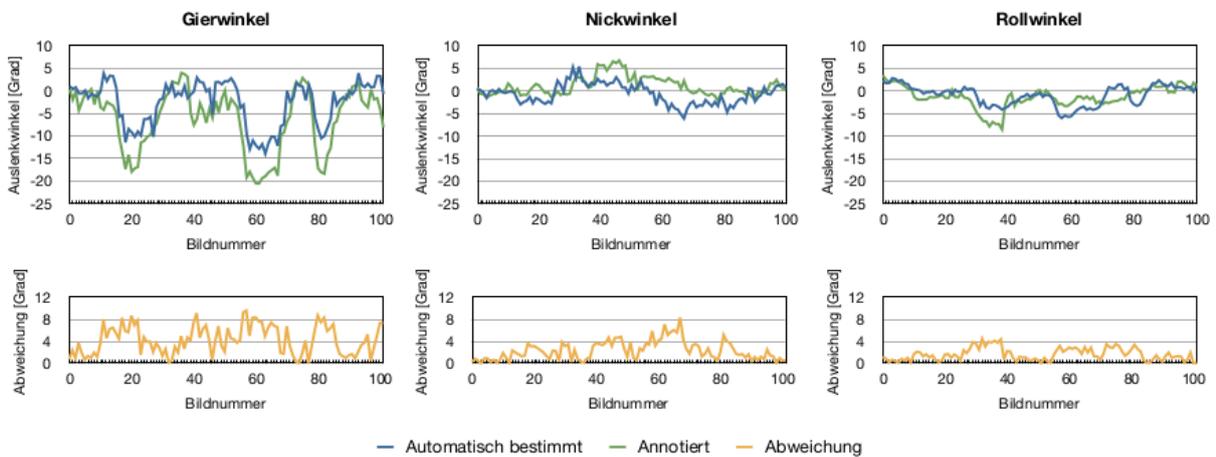
(a) Durchschnittliche Abweichung und Varianz der erkannten Gesichtsmerkmale bei nach vorne gerichtetem Blick.

(b) Durchschnittliche Abweichung und Varianz der erkannten Gesichtsmerkmale bei rotierter Kopfposition, verursacht durch einen Blick in den Spiegel.

Abb. 5.9.: Abweichung der erkannten Gesichtsmerkmale in Pixel. 30 Pixel entsprechen ungefähr dem Durchmesser der Iris. Die Position des rechten Ohrs (O1) ist nicht auf allen Kamerabildern erkennbar. Kann es nicht lokalisiert werden, wird es auf Basis der anderen Gesichtsmerkmale geschätzt. Dies verursacht eine hohe Varianz und Positionsabweichung.



(a) Die Abweichung der Kopfposesbestimmung bei nach vorne gerichtetem Blick beträgt im Durchschnitt weniger als 3° .



(b) Die Abweichung der Kopfposesbestimmung steigt mit einer von der Kamera abgewandten Kopfpose. Beim Blick in den Spiegel beträgt die Abweichung der Kopfposeschätzung im Durchschnitt 4° .

Abb. 5.10.: Die Winkelabweichung der Kopfposesbestimmung, aufgetragen über 100 Testbilder. Mit zunehmendem Rotationswinkel des Kopfs steigt die Abweichung von der annotierten Kopfpose.

Wechselnde Lichtverhältnisse stören die zuverlässige Erkennung der Gesichtsmerkmale. Werden diese jedoch erkannt, kann der Pupillenmittelpunkt im hochauflösten RGB-Farbbild mit einer durchschnittlichen Abweichung von 1,1 Pixel in horizontaler und 2,4 Pixel in vertikaler Richtung erkannt werden. Mit dem beschriebenen Augenmodell lässt damit die Blickrichtung mit einer Winkelabweichung von 2° bzw. $4,3^\circ$ bestimmen.

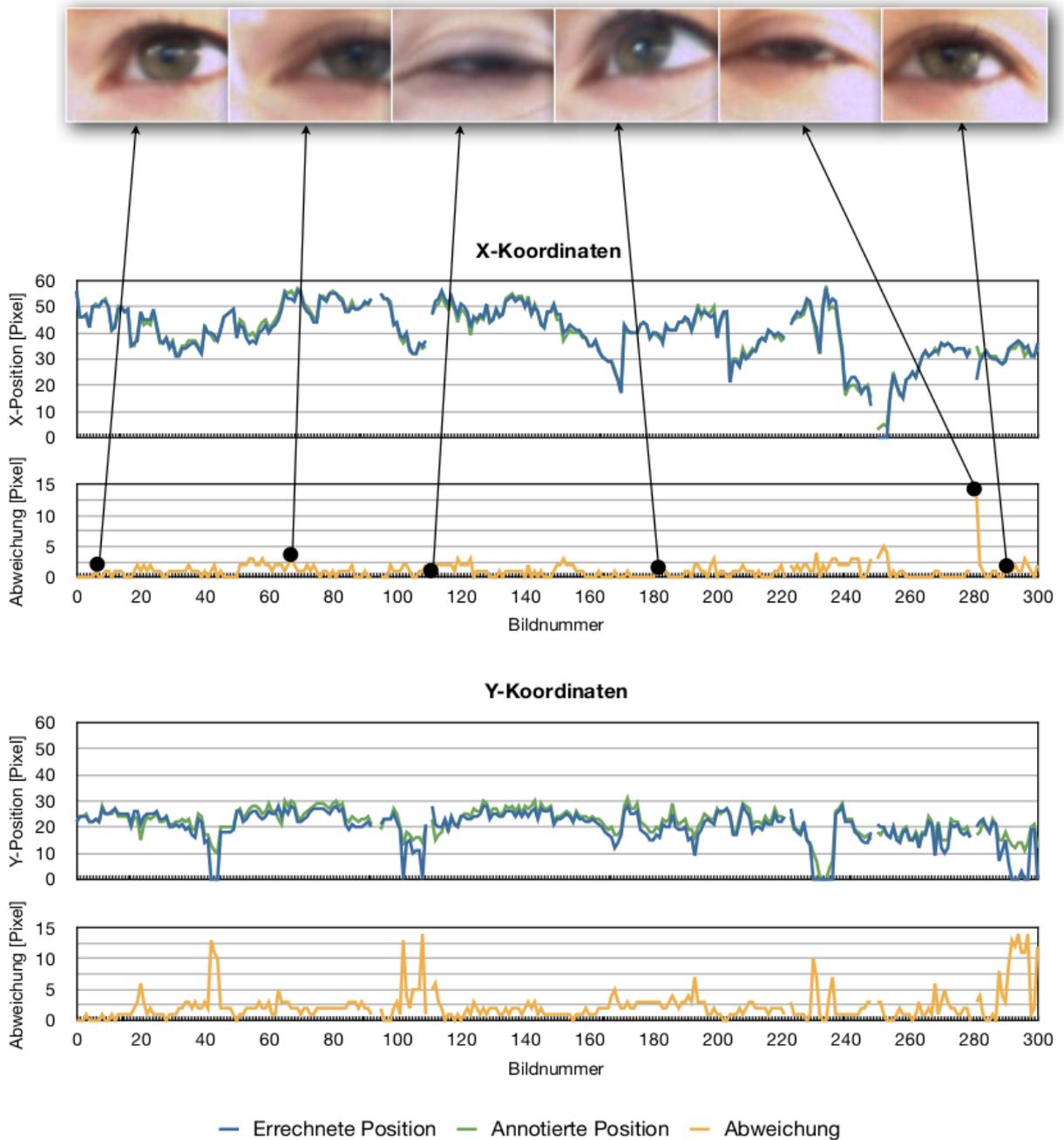


Abb. 5.11.: Abweichung der erkannten Pupillenmittelpunkte in vertikaler und horizontaler Richtung. Für die horizontale Richtung ergibt sich eine durchschnittliche Abweichung von 1,1 Pixel, während die Abweichung in vertikaler Richtung 2,4 Pixel beträgt. Mit einem Augendurchmesser von 24 mm im Augenmodell ergibt sich dadurch eine Winkelabweichung von 2° bzw. $4,3^\circ$ bei der Bestimmung der Blickrichtung.

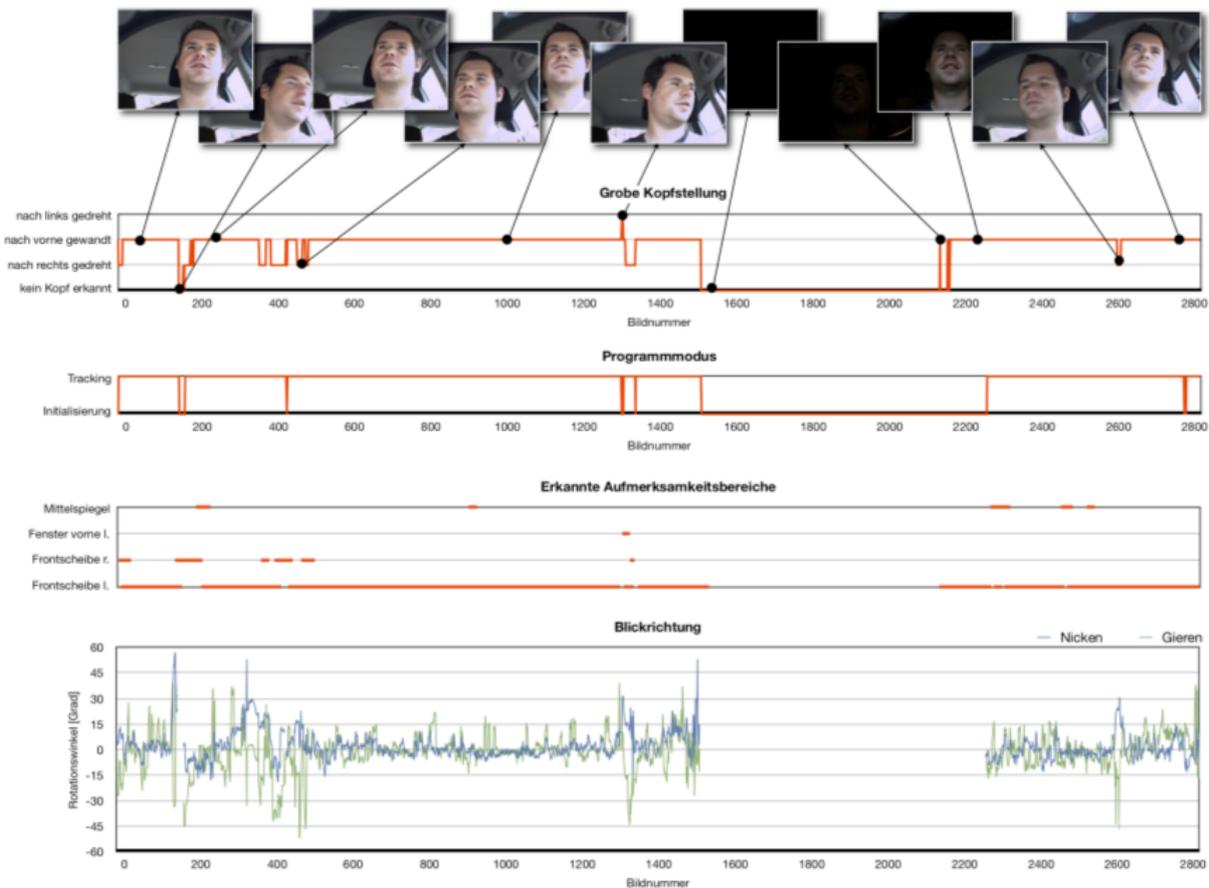


Abb. 5.12.: Qualitative Betrachtung des Systemverhaltens während einer realen Testfahrt. Sowohl die Durchfahrt eines Tunnels als auch zu starke Kopfrotationen führen zu einem Ausfall der CCD-Kamera-Blickrichtungserkennung. Ist das Gesicht des Fahrers gut im Bild erkennbar, können die Kopfpose und die Augenstellung des Fahrers akkurat erkannt und das Aufmerksamkeitsfeld des Fahrers bestimmt werden.

5.2. Fahrstil

Zur Evaluation der Fahrstilerkennung wurde im FZI-Smart-Fahrsimulator eine circa 10 km lange Teststrecke mit Landstraßencharakter modelliert. Die Teststrecke beinhaltet zwei Ortsdurchfahrten und die sechs zur Analyse stehenden Verkehrsszenen (siehe Abschnitt 4.2 Seite 89). Auf dieser Teststrecke wurde durch Fahrten von Studenten, Besuchern und Mitarbeitern eine Messdatenbank mit unterschiedlichen Fahrertypen und unterschiedlicher Erfahrung im Straßenverkehr erstellt. Da alle Probanden die gleiche Teststrecke unter exakt gleichen Umständen fahren, sind die Messungen direkt vergleichbar.

Negativ wirken sich jedoch fehlende Fliehkräfte und fehlendes haptisches Feedback aus. Generell wurde beobachtet, dass Probanden durch fehlende Fliehkräfte und eingeschränktes haptisches Feedback im FZI-Smart-Fahrsimulator schneller fahren als im realen Straßenverkehr. Der Fahrer neigt zu einer höheren Risikobereitschaft, da Unfälle ohne Folgen bleiben.

Zur Auswertung wurde die komplette Fahrt eines Probanden (nicht die Einzelszenen) von einem Experten in eine der Fahrstilklassen *[unsicher]*, *[moderat]*, *[sportlich]* und *[aggressiv]* unterteilt. Der Experte schätzte die Fahrstilzugehörigkeit als Beifahrer und unbeeinflusst von Diagrammen wie Geschwindigkeits-, Beschleunigungs- oder Verzögerungsprofilen. Die Unterteilung zwischen *[aggressiv]* und *[sportlich]* war

selbst für den Experten nicht immer eindeutig. Grundsätzlich wurden Messungen, in denen der Fahrer schnell aber ohne Gefährdung anderer Verkehrsteilnehmer fuhr, als sportlich annotiert. Messungen, in denen durch zu schnelles Fahren eine Gefährdung anderer Verkehrsteilnehmer in Kauf genommen wurde (zum Beispiel starkes Schneiden der Kurve), wurden als aggressiv gekennzeichnet.

5.2.1. Fahrstilerkennung über Fuzzyregeln

Die Abbildung 5.13 zeigt links die Geschwindigkeitsprofile beim Durchfahren der jeweiligen Verkehrssituationen neben den zur Fahrstilklassifikation extrahierten Merkmalen in der Mitte und der Konfusionsmatrix der Fahrstilerkennung rechts. Die im mittleren Graphen dargestellten Merkmale dienen der Fuzzyklassifikation als Eingabeparameter. Abbildung 5.14 führt exemplarisch den Fuzzyregelsatz für die Kurvenszene auf.

Die Konfusionsmatrizen zeigen wie verwechslungsfrei die Merkmale den jeweiligen Fahrstilen zugeordnet werden können. Konfusionsmatrizen wurden in Definition 5.1 definiert. Die ausgeprägte Diagonale in den Konfusionsmatrizen zeigt, dass die Fahrstile über die gewählten Merkmale gut differenziert werden können. In den Testfahrten im Fahrsimulator konnte folgendes Verhalten beobachtet werden:

Geschwindigkeitsmessung: Niedrig gewählte Geschwindigkeiten deuten auf einen unsichere bzw. moderat fahrende Fahrer hin. Hohe Geschwindigkeiten werden sportlichen oder aggressiven Fahrern zugeordnet. Über den aus diesen Beobachtungen gewonnenen Fuzzyregelsatz lassen sich die Messungen zu 83,9% den vom Experten gewählten Fahrstilklassen zuordnen. Die Konfusionsmatrix (Abbildung 5.13a, rechts) zeigt vor allem Verwechslungen zu den jeweiligen Nachbarklassen.

Kurvenszene: Aggressive und sportliche Fahrer zeichnet eine hohe Anfahrtsgeschwindigkeit an Kurven aus, was ein (starkes) Bremsen vor der Kurve erfordert. Nach dem Scheitelpunkt der Kurve beschleunigen aggressive und sportliche Fahrer wieder stark.

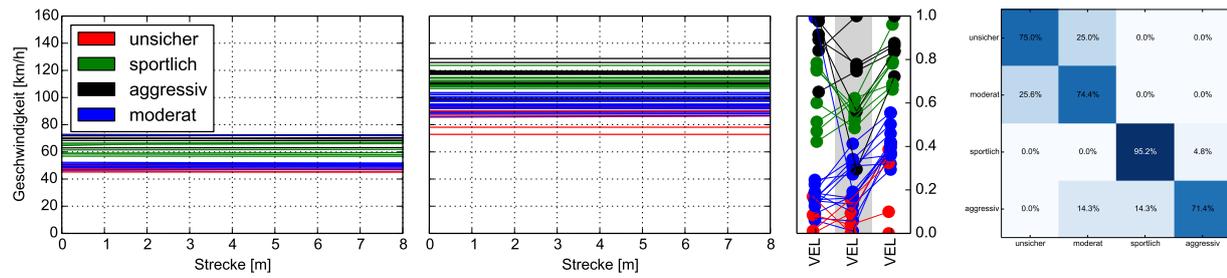
Unsichere Fahrer durchfahren die Szene mit einer niedrigeren Geschwindigkeit. Oft bremsen sie vor der Kurve stark ab, um die Fliehkräfte beim Durchfahren der Kurve gering zu halten. Nach dem Scheitelpunkt der Kurve ist ein leichtes Beschleunigen zu beobachten.

Moderate Fahrer (und zum Teil auch sportliche Fahrer) wählen ihre Geschwindigkeit in der Kurve vorausschauend. Sie müssen fast nicht abbremesen und rollen im Idealfall auf die Kurve aus. Mit einem aus diesen Beobachtungen abgeleiteten Fuzzyregelsatz (siehe Abbildung 5.14 auf Seite 123) konnten 72,6% aller Messungen richtig zugeordnet werden. Die Konfusionsmatrix in Abbildung 5.13b zeigt, dass durch die vorausschauende Geschwindigkeitswahl vor allem moderate Fahrer verwechslungsfrei zugeordnet werden können.

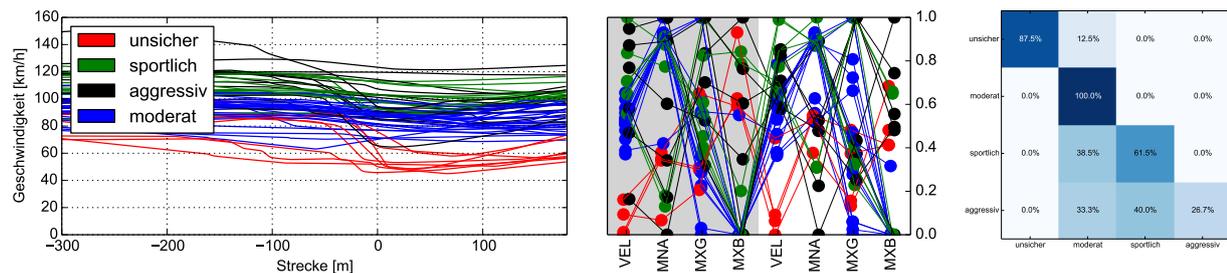
Stoppschildszene: Bei unsicheren Fahrern kann ein sehr frühes Bremsen vor dem Stoppschild beobachtet werden. Aggressive und sportliche Fahrer bremsen dagegen sehr spät. Oft wird das Auto auch nicht komplett zum Stillstand gebracht, wie es die Straßenverkehrsordnung vorschreibt (zumindest nicht im Fahrsimulator, indem keine Gefahren und keine Bußgelder zu erwarten sind). Moderate Fahrer typisiert eine ausgeprägte Ausrollphase vor dem Stoppschild.

Ein unter diesen Prämissen gewählter Fuzzyregelsatz kann 67,7% aller Fahrten richtig zuordnen.

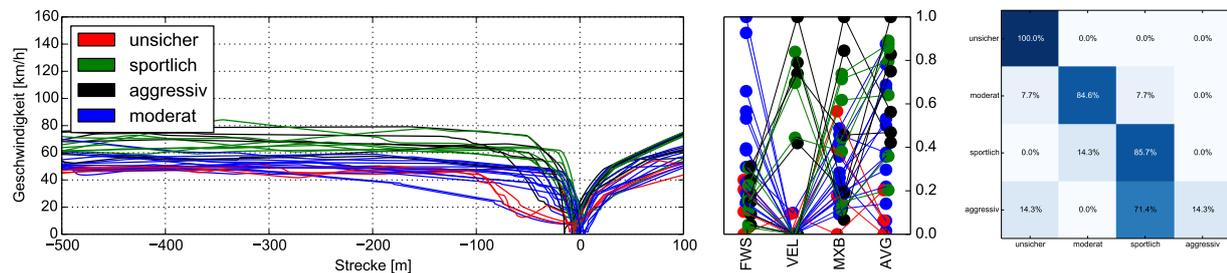
5. Evaluation und Ergebnisse



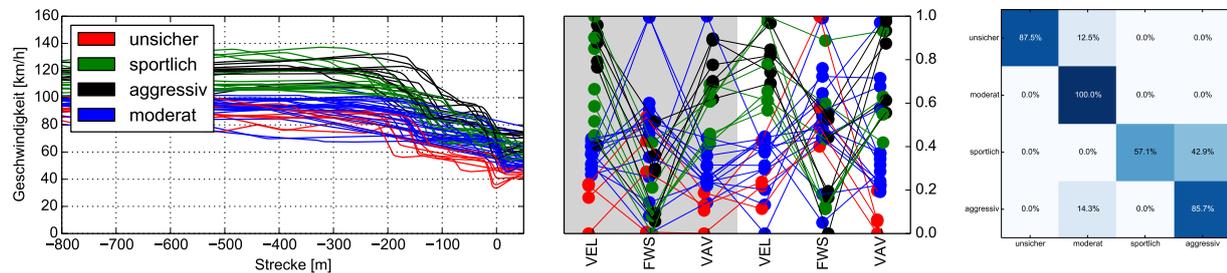
(a) **Geschwindigkeitsmessung** bei einer Geschwindigkeitsbeschränkung von 50 km/h (eine Messung) bzw. einer Geschwindigkeitsbeschränkung von 100 km/h (zwei Messungen). Niedrig gewählte Geschwindigkeiten deuten auf einen unsicheren bzw. moderat fahrenden Fahrer hin. Hohe Geschwindigkeiten werden sportlichen oder aggressiven Fahrern zugeordnet. Über Fuzzyregeln lassen sich 83,9% aller Messungen richtig zuordnen.



(b) Geschwindigkeitsprofil beim Durchfahren einer **Kurvenszene** (zwei Messungen). Der Scheitelpunkt der Kurve befindet sich am Streckenabschnitt 0. Moderate Fahrer schätzen die Kurve richtig ein und Bremsen kaum. 72,6% aller Messungen werden über die Fuzzyregeln richtig zugeordnet.



(c) Geschwindigkeitsprofil der **Stopschildszene** bei einer vorgeschriebenen Geschwindigkeit von 50 km/h. Moderate Fahrer rollen auf das Stoppschild aus und bremsen dezent. Bei aggressiven, sportlichen und unsicheren Fahrern sind stärkere Bremsungen beobachtbar. Nach dem Stoppschild fahren sportliche und aggressive Fahrer rasanter an als moderate und unsichere Fahrer. Es konnte eine Erkennungsrate von 67,7% erzielt werden.



(d) Geschwindigkeitsprofil der **Ortsanfahrtsszene** mit dem typischen Geschwindigkeitsübergang von 100 km/h auf 70 km/h auf 50 km/h. Bei sportlichen und aggressiven Fahrern sind beim Anfahren auf eine Ortschaft höhere Geschwindigkeiten zu beobachten. Moderate Fahrer rollen frühzeitig aus und bremsen kaum. Der aufgestellte Fuzzyregelsatz kann 85,5% aller Messungen richtig klassifizieren.

Abb. 5.13.: Geschwindigkeitsprofile (links) beim Durchfahren der jeweiligen Verkehrsszene neben den für die Fahrstilklassifikation verwendeten Merkmalen in der Mitte. Die Konfusionsmatrix rechts zeigt, wie eindeutig der Fahrstil für die jeweilige Szene bestimmt werden kann. Die Merkmale im mittleren Graphen, wie beispielsweise die gefahrenen Geschwindigkeiten in Abbildung 5.13a, sind auf einen Wertebereich von 0 bis 1 normiert. In der gleichen Fahrt auftretende Merkmale sind mit einer Linie verbunden. Das Geschwindigkeitsprofil und auch die Merkmale sind mit dem annotierten Fahrstil eingefärbt.

Die Konfusionsmatrix für die Stoppschildszene (Abbildung 5.13c) zeigt vor allem Verwechslungen zu den jeweiligen Nachbarklassen.

Ortsanfahrtsszene: Auch beim Anfahren an eine Ortschaft zeichnet einen moderaten Fahrer vor allem die lange Ausrollphase aus. Sie durchfahren die Szene fast ohne zu bremsen. Sportliche und aggressive Fahrer interpretieren die ausgeschriebene Geschwindigkeit weniger streng. Bei der Einfahrt in die Ortschaft sind bei ihnen erhöhte Geschwindigkeiten von bis zu 80 km/h zu beobachten. Unsichere Fahrer orientieren sich hingegen streng an den ausgeschriebenen Geschwindigkeitsbegrenzungen und bremsen zum Teil sehr stark, um diese auch möglichst genau einzuhalten. Ein Fuzzyregelsatz, welcher diese Beobachtungen umsetzt, klassifiziert die Messungen zu 85,5% richtig. Moderate Fahrer können über ihre vorausschauend gewählte Geschwindigkeit sehr gut klassifiziert werden, was die Konfusionsmatrix in Abbildung 5.13d zeigt.

```

if VAS is HIGH and FWS is LONG and VAV is HIGH then DRIVING_STYLE is KEEN
if VAS is HIGH and FWS is SHORT and VAV is HIGH then DRIVING_STYLE is KEEN
if VAS is NORMAL and FWS is LONG and VAV is HIGH then DRIVING_STYLE is SEDATE
if VAS is NORMAL and FWS is LONG and VAV is NORMAL then DRIVING_STYLE is SEDATE
if VAS is NORMAL and FWS is LONG and VAV is SLOW then DRIVING_STYLE is ANXIOUS
if VAS is NORMAL and FWS is LONG and VAV is TOO_HIGH then DRIVING_STYLE is SEDATE
if VAS is NORMAL and FWS is SHORT and VAV is NORMAL then DRIVING_STYLE is SEDATE
if VAS is NORMAL and FWS is SHORT and VAV is SLOW then DRIVING_STYLE is SEDATE
if VAS is NORMAL and FWS is SHORT and VAV is TOO_HIGH then DRIVING_STYLE is AGGRESSIVE
if VAS is SLOW and FWS is LONG and VAV is NORMAL then DRIVING_STYLE is SEDATE
if VAS is SLOW and FWS is LONG and VAV is SLOW then DRIVING_STYLE is ANXIOUS
if VAS is SLOW and FWS is LONG and VAV is TOO_HIGH then DRIVING_STYLE is SEDATE
if VAS is SLOW and FWS is SHORT and VAV is SLOW then DRIVING_STYLE is ANXIOUS
if VAS is TOO_HIGH and FWS is LONG and VAV is HIGH then DRIVING_STYLE is AGGRESSIVE
if VAS is TOO_HIGH and FWS is LONG and VAV is NORMAL then DRIVING_STYLE is KEEN
if VAS is TOO_HIGH and FWS is LONG and VAV is TOO_HIGH then DRIVING_STYLE is AGGRESSIVE
if VAS is TOO_HIGH and FWS is SHORT and VAV is HIGH then DRIVING_STYLE is AGGRESSIVE
if VAS is TOO_HIGH and FWS is SHORT and VAV is TOO_HIGH then DRIVING_STYLE is AGGRESSIVE

```

Abb. 5.14.: Fuzzyregelsatz für die Kurvenszene. VAS steht für die Geschwindigkeit vor der Kurve, FWS für die Länge der Ausrollphase und VAV für die Geschwindigkeit am Scheitelpunkt der Kurve.

Die erste Fuzzyregel formuliert den Zusammenhang: Ist die *Geschwindigkeit vor der Kurve hoch*, die *Ausrollphase lang* und die *Geschwindigkeit am Scheitelpunkt der Kurve hoch*, so ist dies ein Merkmal für einen *sportlichen Fahrstil*.

Neben der Betrachtung des Fahrstils angesichts der Aggressivität bzw. Risikobereitschaft kann das Fahrverhalten auch hinsichtlich des Energieverbrauchs klassifiziert werden. Die Annotation der Messdaten ergibt sich in diesem Fall direkt aus dem Energieverbrauch der Messfahrt und unterliegt nicht der subjektiven Einschätzung eines Experten. Anhand des Energieverbrauchs auf der gesamten Fahrt wurden die Fahrstilklassen [*verschwendend*], [*normal*] und [*ökonomisch*] definiert. Erwartungsgemäß verbrauchen vor allem Fahrer der moderaten Fahrstilklasse weniger Energie, während sportliche oder aggressive Fahrer durch starkes Beschleunigen und starkes Bremsen mehr Energie verbrauchen. Abbildung 5.15a zeigt die Korrelation zwischen den Fahrstilklassen der Risikobereitschaft und den Fahrstilklassen des Energieverbrauchs. Abbildung 5.15b zeigt den Energieverbrauch der Fahrer in den jeweiligen Verkehrssi-

tuationen. Je nach Verkehrssituation ist der Energieverbrauch auf der Einzelszene nicht repräsentativ für den Verbrauch auf der gesamten Fahrstrecke. Überfährt ein aggressiver Fahrer beispielsweise ein Stoppschild mit niedriger Geschwindigkeit verbraucht er weniger Energie als ein moderater Fahrer, der sein Auto zum Stillstand bringt. Über Fuzzyregelsätze, welche den Fahrstil hinsichtlich des Energieverbrauchs klassifizieren können die Messdaten zu 93,5% richtig zugeordnet werden. Abbildung 5.15c zeigt die Konfusionsmatrix, die lediglich geringfügige Verwechslungen zu den jeweiligen Nachbarklassen aufweist. Zwar fällt die Richtig-Klassifikationsrate der Einzelszenen leicht geringer aus (Stoppschildszene 80,6%, Ortsanfahrtsszene 87,1% und Kurvenszene 80,7%), dennoch lässt sich durch die Fusion der Ergebnisse aus den Einzelszenen (Mehrheitsentscheidung) eine stabile Gesamtklassifikation erstellen. Abbildung

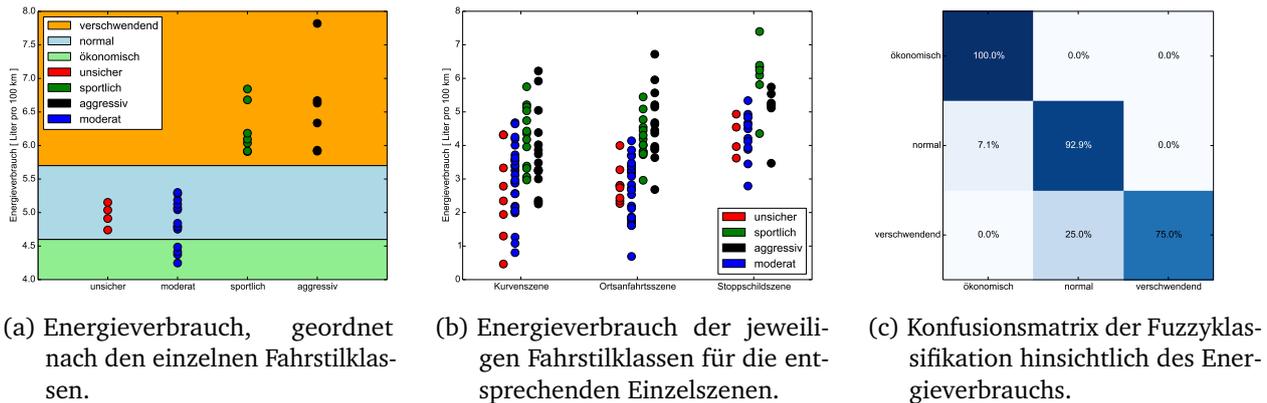


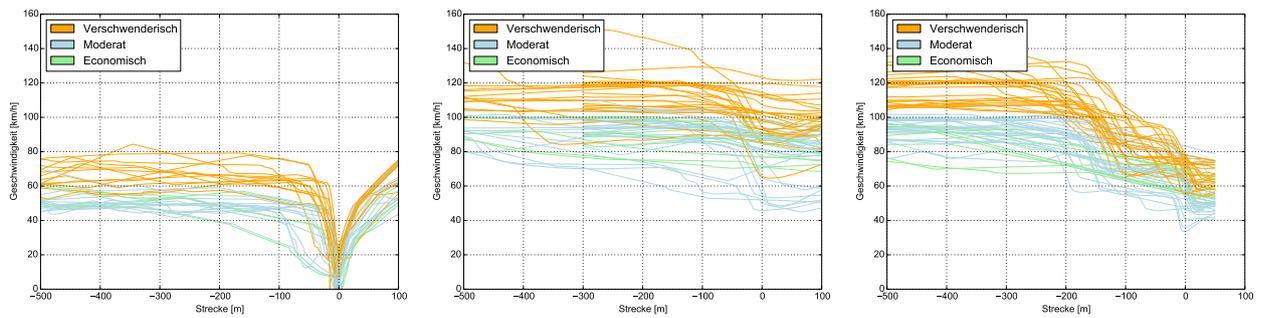
Abb. 5.15.: Über Fuzzyregelsätze, welche den Fahrstil hinsichtlich des Energieverbrauchs klassifizieren, können die Messdaten zu 93,5% richtig zugeordnet werden. Geringe Verwechslungen gibt es für die jeweiligen Energie-Nachbarklassen. Ein sportlicher bzw. aggressiver Fahrstil fordert tendenziell einen erhöhten Energieverbrauch.

5.16 stellt für die Einzelszenen die Konfusionsmatrizen und die Geschwindigkeitsprofile der jeweiligen Energieklassen dar.

5.2.2. Fahrstilerkennung über MLNs

Die Klassifikation des Fahrstils über *Markov Logik Netze* basiert auf den gleichen Merkmalen wie die Fahrstilklassifikation über Fuzzyregeln. Im Vergleich zur Fahrstilklassifikation über Fuzzyregeln wird bei der Klassifikation über *MLNs* jedoch weit weniger Expertenwissen eingesetzt. Lediglich die Fahrstilklassen (*[unsicher]*, *[moderat]*, *[sportlich]* und *[aggressiv]* für die Klassifizierung hinsichtlich des Risikoverhaltens und *[verschwendend]*, *[normal]* und *[ökonomisch]* hinsichtlich des Energieverbrauchs werden vorgegeben.

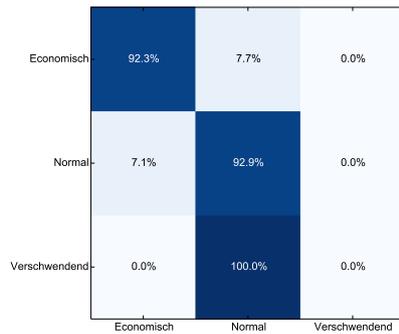
Unüberwachtes Clusterverfahren Zur Klassifizierung werden zunächst die Merkmale über ein unüberwachtes Clusterverfahren, basierend auf dem *k-Means Algorithmus*, diskretisiert. Dazu wird a-priori die Anzahl der Cluster *k* über das *Bayesian Information Criterium (BIC)* bestimmt (siehe Abschnitt 4.2.2). Abbildung 5.17 zeigt das *BIC* aufgetragen über die Anzahl an Clustern für das Merkmal Geschwindigkeit in der Szene „Geschwindigkeitsmessung bei einer Beschränkung von 50 km/h“. Die optimale Anzahl an Clustern wird im abgebildeten Beispiel mit dem *BIC* als 7 bestimmt.



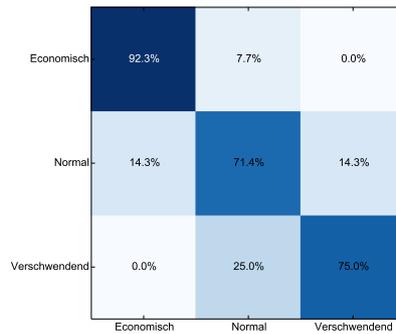
(a) Geschwindigkeitsprofile Stoppschildsszene.

(b) Geschwindigkeitsprofile der Kurvenszene.

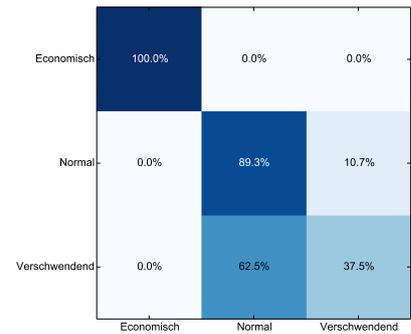
(c) Geschwindigkeitsprofile der Ortsanfahrtsszene.



(d) Konfusionsmatrix der Stoppschildsszene. Richtig-Klassifikationsrate: 80,6%.



(e) Konfusionsmatrix der Kurvenszene. Richtig-Klassifikationsrate: 80,6%.



(f) Konfusionsmatrix der Ortsanfahrtsszene. Richtig-Klassifikationsrate: 87,1%.

Abb. 5.16.: Geschwindigkeitsprofile der Einzelszenen (oben) gefärbt mit der jeweiligen Verbrauchsklasse. Konfusionsmatrizen der Fuzzyregelsätze für die Einzelszenen (unten).

Definition des Markov Logik Netzes Im *MLN* wird definiert, dass sich jedes Merkmal auf den Fahrstil bzw. das Energieverhalten des Fahrers auswirkt. Dies kann automatisiert erfolgen. Abbildung 5.18 zeigt das automatisch generiertes *MLN* für die Ortsanfahrtsszene. Als Grundatome des Netzes werden die Merkmale als Eingabe und die Fahrstilklasse als Ausgabe definiert. Als Relationen wird definiert, dass jedes Merkmal eine Auswirkung auf den Fahrstil des Fahrers hat.

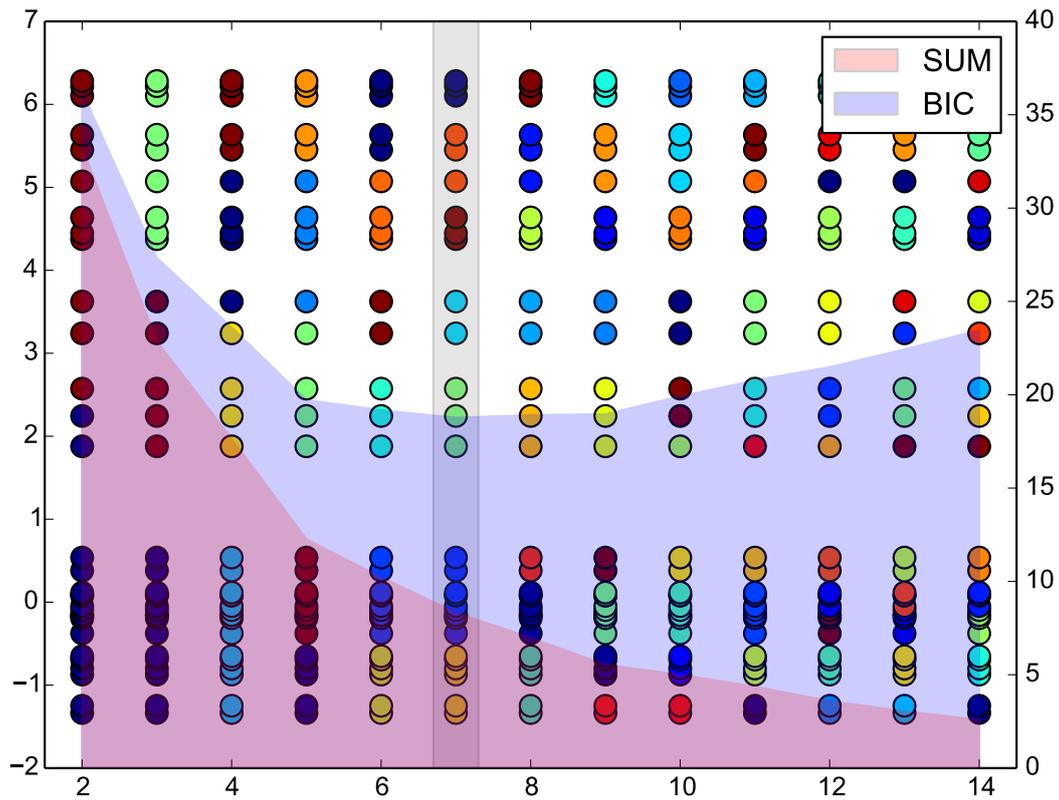


Abb. 5.17.: Die optimale Anzahl für das *k-Means Clustering* wird für jedes Merkmal über das *Bayesian Information Criterion (BIC)* bestimmt. Für das hier dargestellte Merkmal (Geschwindigkeit in der Szene „Geschwindigkeitsmessung bei einer Beschränkung von 50 km/h“) sind 7 Cluster ideal, da der *BIC* Wert für 7 Cluster ein Minimum hat.

Der *SUM* Wert summiert die Differenzwerte der Messwerte zu ihren zugeordneten Clusterzentren auf (siehe Formel 4.10, Seite 94).

```

1 // Grundatome - Merkmale (Eingabe)
2 village_freewheeling_start( cluster_nr , id )
3 village_approaching_speed( cluster_nr , id )
4 village_avg_brake_pressure( cluster_nr , id )
5 village_max_brake( cluster_nr , id )
6 village_vel_at_village( cluster_nr , id )
7
8 // Grundatome - Fahrstil (Ausgabe)
9 style_village( style! , id )
10
11 // Relationen (Merkmal wirkt sich auf Fahrstil aus)
12 village_freewheeling_start( +v , id ) => style_village( +s , id )
13 village_approaching_speed( +v , id ) => style_village( +s , id )
14 village_avg_brake_pressure( +v , id ) => style_village( +s , id )
15 village_max_brake( +v , id ) => style_village( +s , id )
16 village_vel_at_village( +v , id ) => style_village( +s , id )

```

Abb. 5.18.: Generiertes *MLN* für die Ortsanfahrtsszene. Die *id* ordnet eine Messung den Merkmalen eindeutig zu. Findet sich in der Formel ein Ausrufezeichen, bedeutet dies Eindeutigkeit. Im Beispiel gibt es für jede *id* genau einen Fahrstil. Ist einer Variablen ein + vorgestellt, bedeutet dies, dass die Relation für alle möglichen Werte der Variablen definiert wird. Im Beispiel steht +v für alle möglichen Zustände des Merkmals (alle Cluster) und +s für alle möglichen Fahrstile.

Lernen der Gewichte Ob die über die Netzstruktur definierten Relationen konsistent mit der beobachteten Welt sind, wird im *MLN* über Gewichte spezifiziert. In der umgesetzten Fahrstilerkennung werden diese Gewichte aus den Messdaten (Trainingsdaten) gelernt. Der Textauszug 5.19 zeigt zwei Trainingsbeispiele aus denen die Gewichte des *MLN* bestimmt werden.

```

1 style_village( "sedate" , 2028 )
2 village_approaching_speed( "CLUSTER3" , 2028 )
3 village_freewheeling_start( "CLUSTER5" , 2028 )
4 village_vel_at_village( "CLUSTER3" , 2028 )
5 village_avg_brake_pressure( "CLUSTER2" , 2028 )
6 village_max_brake( "CLUSTER5" , 2028 )
7
8 style_village( "anxious" , 1009 )
9 village_approaching_speed( "CLUSTER1" , 1009 )
10 village_freewheeling_start( "CLUSTER0" , 1009 )
11 village_vel_at_village( "CLUSTER0" , 1009 )
12 village_avg_brake_pressure( "CLUSTER3" , 1009 )
13 village_max_brake( "CLUSTER8" , 1009 )

```

Abb. 5.19.: Zwei Trainingsbeispiele aus denen die Gewichte des *MLNs* gelernt werden.

Das Ergebnis des Lernens sind die den Relationen zugeordneten Gewichte, welche die Konsistenz der Relation mit der Trainingsmenge angeben.

Abbildung 5.20 zeigt die ersten und die letzten vier Zeilen der gelernten Gewichte für das Merkmal „Geschwindigkeit bei der Ortseinfahrt“ der Ortsanfahrtszene.

Es wurde gelernt, dass eine sehr niedrige Geschwindigkeit (Cluster 0) auf einen ängstlichen Fahrstil hinweist. Hohe Geschwindigkeiten (Cluster 8 bzw. Cluster 7) deuten hingegen auf einen aggressiven Fahrstil hin. Eine negative Gewichtung der Formeln signalisiert ein „Nichtzutreffen“ der Relation. Geringe Geschwindigkeiten (Cluster 1 und Cluster 2) werten die Fahrstilklassen [*aggressiv*] und [*sportliche*] ab. Laut der letzten Zeile sprechen sehr niedrige Geschwindigkeiten (Cluster 0) auch nicht für moderate Fahrer.

```

1 2.0541 village_vel_at_village("CLUSTER0",i) => style_village("anxious",i)
2 1.73421 village_vel_at_village("CLUSTER8",i) => style_village("aggressive",i)
3 1.52599 village_vel_at_village("CLUSTER7",i) => style_village("aggressive",i)
4 1.41681 village_vel_at_village("CLUSTER2",i) => style_village("sedate",i)
5 ...
6 -0.0655919 village_vel_at_village("CLUSTER1",i) => style_village("keen",i)
7 -0.0840615 village_vel_at_village("CLUSTER2",i) => style_village("aggressive",i)
8 -0.199847 village_vel_at_village("CLUSTER0",i) => style_village("sedate",i)

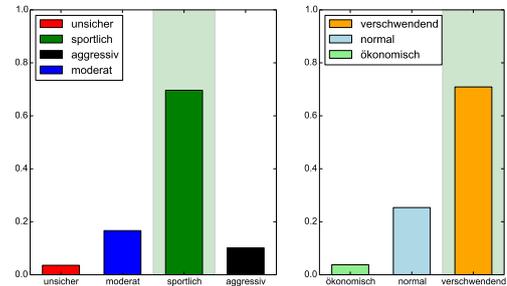
```

Abb. 5.20.: Auszug aus den gelernten Gewichten für das Merkmal „Geschwindigkeit Ortseinfahrt“ der Ortsanfahrtszene.

Inferenz Mit definierten Relationen und der über die Trainingsdaten gelernten Gewichtung der Relationen, lässt sich für die Testdaten eine Zugehörigkeitswahrscheinlichkeit für jede Fahrstilkategorie ableiten.

Abbildung 5.21 zeigt ein Eingabebeispiel für das *MLN* und die daraus resultierende Wahrscheinlichkeitsverteilung für die Fahrstil- bzw. die Energieklassen. Eine Klassifikation wird anschließend über die *Maximum-Likelihood-Hypothese* (wahrscheinlichste Klasse) getroffen. Sie ist in Abbildung 5.21b grün hinterlegt.

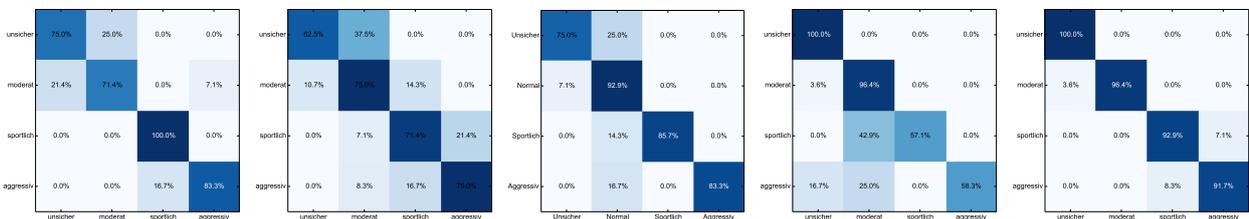
```
tight_curve_v_in_curve( "CLUSTER10" , 1016 )
tight_curve_max_acc_after_curve( "CLUSTER0" , 1016 )
tight_curve_min_a( "CLUSTER3" , 1016 )
tight_curve_max_gas_after_curve( "CLUSTER1" , 1016 )
tight_curve_max_brake_before_curve( "CLUSTER1" , 1016 )
tight_curve_fw_start( "CLUSTER6" , 1016 )
```



(a) Testbeispiel (Eingabeparameter) für das *MLN* zur Erkennung des Fahrstils hinsichtlich Risikobereitschaft und Energieverhalten. (b) Über das *MLN* bestimmte Wahrscheinlichkeiten für den Fahrstil hinsichtlich Risikobereitschaft und Energieverhalten.

Abb. 5.21.: Mit den Merkmalen als Eingabeparameter wird über das *MLN* eine Wahrscheinlichkeit für jeden Fahrstil und jede Energieklasse abgeleitet. Die endgültige Risiko- bzw. die Energieklasse wird über die *Maximum-Likelihood-Hypothese* (in der Abbildung grün hinterlegt) klassifiziert.

Die Abbildungen 5.22 und 5.23 zeigen die Konfusionsmatrizen der Fahrstilerkennung über *MLNs* für die einzelnen Fahrscenen. Außerdem zeigen sie die erzielten Erkennungsraten auf den gewählten Fahrscenen.

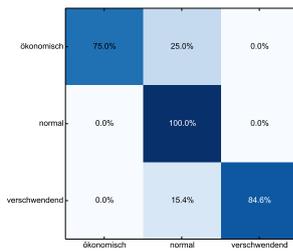


(a) 50 km/h Szene (80,6%) (b) 100 km/h Szene (72,6%) (c) Stoppschild (87,1%) (d) Scharfe Kurve (80,5%) (e) Ortsanfahrt (95,2%)

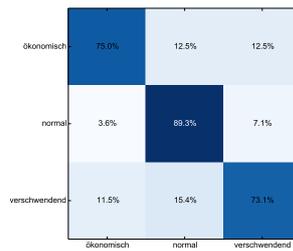
Abb. 5.22.: Konfusionsmatrizen der einzelnen Szenen für die Fahrstilerkennung über *MLNs*. Starke Verwechslungen treten lediglich zu den Nachbarklassen auf.

5.2.3. Vergleich zwischen der Fahrstilklassifikation über *MLNs* und *Fuzzylogik*

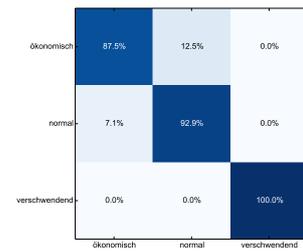
Abbildung 5.24 stellt die Erkennungsleistung der Fahrstilerkennung über Fuzzyregeln der Fahrstilerkennung über *MLNs* gegenüber. Für komplexe Szenen fällt die Erkennungsleistung der *MLNs* besser aus als die Erkennungsleistung über die Fuzzyregeln. Dies ist darauf zurückzuführen, dass in Szenen mit vielen Eingabeparametern es dem Experten schwer fällt, einen konsistenten Fuzzyregelsatz aufzustellen. Ein maschinelles Lernen der Zusammenhänge generalisiert komplexe Szenen besser als eine Parametrisierung durch den Experten.



(a) Konfusionsmatrix für die Stoppschildszene.
(Erkennungsrate: 90,3%).



(b) Konfusionsmatrix für die Kurvenszene.
(Erkennungsrate: 80,6%).



(c) Konfusionsmatrix für die Ortsanfahrtsszene.
(Erkennungsrate: 95,2%).

Abb. 5.23.: Konfusionsmatrizen der einzelnen Szenen für die Erkennung des Energieverhaltens über *MLNs*. Fusioniert man die Klassifikationsergebnisse der Einzelszenen über einen Mehrheitsentscheid zu einer Klasse für die gesamte Fahrt, wird über *MLNs* eine Erkennungsleistung von 96,8% erzielt.

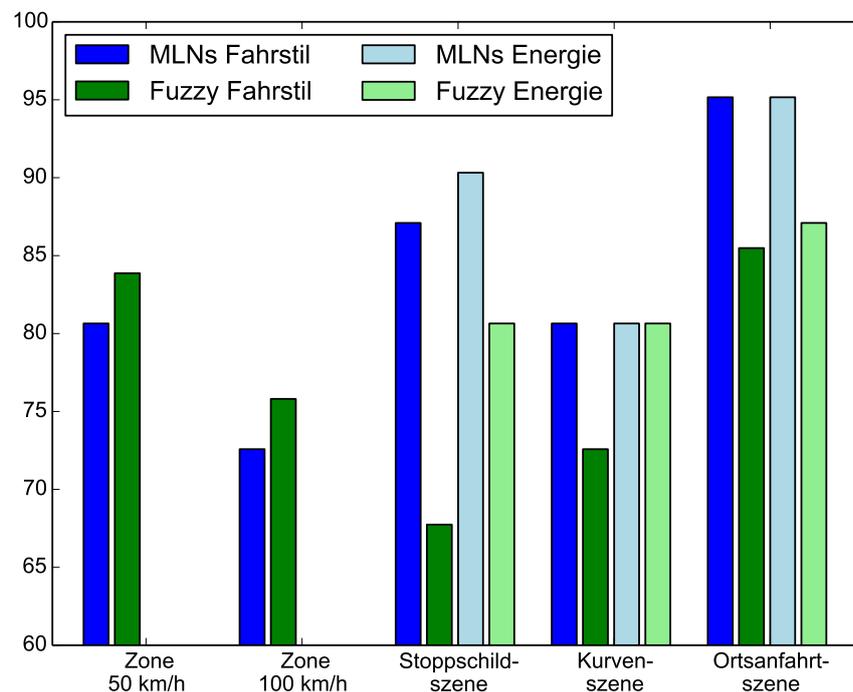


Abb. 5.24.: Vergleich der Erkennungsraten über Fuzzyregeln und *MLNs*. Vor allem für komplexe Szenen fällt die unüberwachte Klassifikation des Fahrstils über *MLNs* besser aus als die Klassifikation über die Fuzzyregeln.

5.3. Verbesserung der Reichweitenprognose über fahrerindividuelle Komfortparameter

In Abschnitt 4.3.1 werden Komfortparameter bestimmt, welche das Fahrerverhalten quantitativ beschreiben. Unter anderem wurde ein fahrerindividuelles Geschwindigkeitsprofil sowie die Brems- und Beschleunigungscharakteristik des Fahrers unter Betrachtung des situativen Kontexts berechnet. In diesem Abschnitt wird demonstriert, wie anhand dieser Werte die Reichweitenprognose von Fahrzeugen verbessert werden kann.

Der Energieverbrauch des Fahrzeugs wird in dieser Arbeit über das *FZI-Routen-Analyse-Tool* geschätzt, welches den Verbrauch des Fahrzeugs über folgende Daten und Modelle ermittelt:

- **Echtzeit Verkehrsinformationen**, wie Staus, Baustellen oder stockender Verkehr extrahiert aus den *Google Traffic Services* [Inc07] (Details siehe [Web+11])
- **Geschwindigkeitsbeschränkungen** und **Kurvenradien** abgeleitet aus dem freien Kartenmaterial der *OpenStreetMap* [HW08b]
- **Höhendaten** aus *SRTM-Daten* [Jar+08] bzw. aus Aufzeichnungen über das Inertialsystem des FZI-Versuchsträgers CoCar
- **Verbrauchsmodell des Fahrzeugs** in Form einer empirisch für den FZI-Versuchsträger CoCar ermittelten Verbrauchstabelle
- **Komfortparameter** des Fahrers, quantifiziert über Geschwindigkeitsprofile und der Brems- und Beschleunigungscharakteristik des Fahrers (siehe Abschnitt 4.3.1)

Betrachtet man die zur Prognose benutzten Komponenten, so stellt man fest, dass die Individualität des Fahrers einzig durch die Komfortparameter zur Geltung kommt. Alle anderen Komponenten sind fahrerunabhängig.

Zur Prognose des Verbrauchs wird im *FZI-Routen-Analyse-Tool* ein Geschwindigkeitsverlauf für die Strecke berechnet, der sich aus den Geschwindigkeitsbegrenzungen, den Kurvenradien und den Echtzeit-Verkehrsinformationen der Strecke ableitet. Mit diesem berechneten Geschwindigkeitsverlauf wird über eine Mikrosimulation in einem zweiten Schritt eine Fahrt mit den fahrerindividuellen Komfortparametern simuliert. Dies wird durch eine sukzessive Simulation der longitudinalen Aktionen des Fahrers aus dem Aktionskatalog [*bremsen*], [*beschleunigen*], [*Geschwindigkeit halten*] und [*ausrollen*] umgesetzt. Die Aktionen sind durch die Komfortparameter individuell auf den Fahrer parametrisiert (siehe Tabelle 4.5, Seite 98). Die in einem Simulationsschritt gewählte Aktion bestimmt sich aus

- der Geschwindigkeit im aktuellen Simulationsschritt,
- einer Zielgeschwindigkeit, bestimmt durch den berechneten Geschwindigkeitsverlauf und
- dem individuellen Geschwindigkeitsprofil des Fahrers, bestimmt durch die Komfortparameter.

Damit berechnet sich für die Route ein fahrerindividuelles Geschwindigkeitsprofil, aus dem über das Verbrauchsmodell des Fahrzeugs der Energieverbrauch abgeleitet werden kann. Die Vorgehensweise zur Berechnung des Energieverbrauchs wurde in Teilen in [Nie+12; Koh+11] veröffentlicht.

Die Verbesserung der Verbrauchsschätzungen durch die Berücksichtigung des individuellen Fahrerverhaltens wird auf zwei Szenarien gezeigt: In Abschnitt 5.3.1 wird dargestellt, wie sich für eine 18 km lange Autobahnfahrt von Karlsruhe nach Rastatt die Verbrauchsprognose während der Fahrt sukzessive

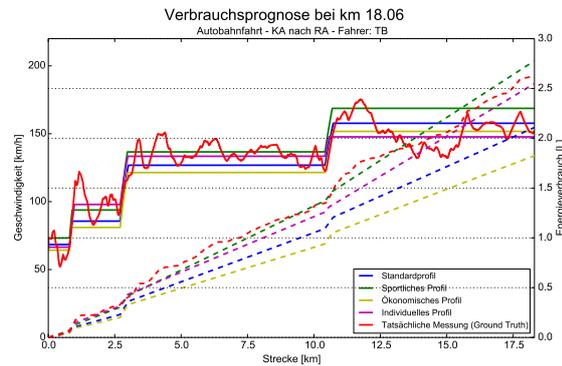
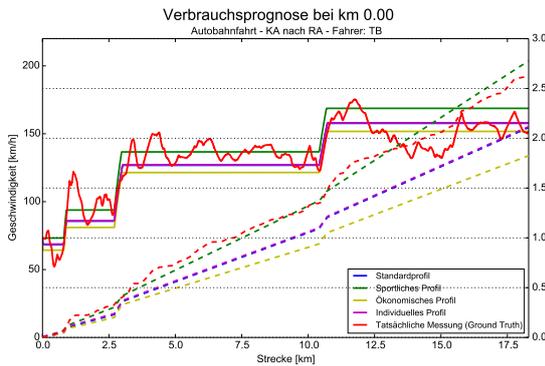
verbessert und sich durch die berechneten Komfortparameter eine genauere Verbrauchsschätzung für die Rückfahrt berechnen lässt. Dies zeigt einerseits die direkte Verbesserung der Verbrauchsschätzung und gleichzeitig die Übertragbarkeit auf unbekannte Routen mit anderen Geschwindigkeitsbeschränkungen und Verkehrsgegebenheiten.

Abschnitt 5.3.2 zeigt, dass das Fahrerverhalten stets individuell ist. Auf einer 25 km langen Teststrecke (Turmbergstrecke) werden für zwei verschiedene Fahrer die Energieverbräuche basierend auf dem eigenen Fahrerprofil sowie dem Fahrerprofil des anderen Fahrers berechnet und mit realen Messungen, aufgezeichnet über den FZI-Versuchsträger CoCar, verglichen.

5.3.1. Fahrerindividuelle Verbrauchsschätzung auf einer Autobahnfahrt

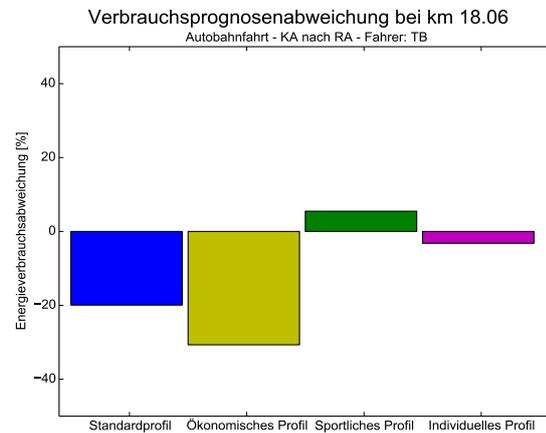
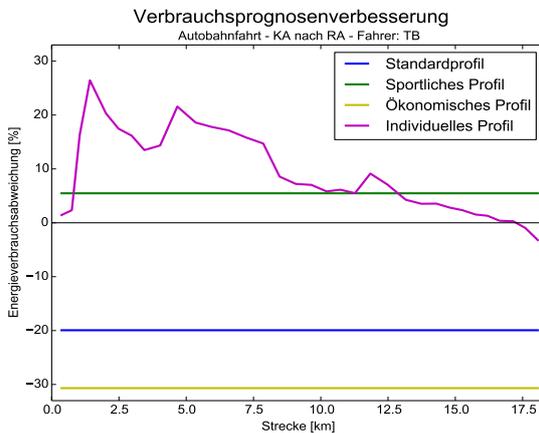
Als Teststrecke für die Evaluation wurde eine 18 km lange Autobahnfahrt von Karlsruhe nach Rastatt gewählt. Die Fahrt beginnt mit dem Autobahnzubringer (mit Geschwindigkeitsbegrenzungen von 60 und 80 km/h) und endet mit der Fahrt auf der Autobahn. Die anfängliche Geschwindigkeitsbegrenzung von 120 km/h und einer unbegrenzten Geschwindigkeitsbegrenzung ab Streckenkilometer 11. Zur Verbrauchsschätzung stehen im *FZI-Routen-Analyse-Tool* jeweils drei vorparametrisierte Profile zur Verfügung. Diese sind [*sportlich*], [*ökonomisch*] und [*standard (default)*]. Abbildung 5.25a zeigt die initialen Verbrauchsschätzungen ohne Kenntnisse des individuellen Fahrverhaltens. In der initialen Schätzung wird der Fahrer als *Standardfahrer* eingestuft. Im dargestellten Fall würde die sportliche Verbrauchsschätzung am ehesten der Realität entsprechen. Abbildung 5.25b zeigt das gleiche Diagramm am Ende der Autobahnfahrt (Streckenkilometer 18,06). Im Gegensatz zur initialen Verbrauchsschätzung von 2,11 Liter, welche 0,52 Liter (19,93%) vom echten Benzinverbrauch (2,63 Liter) abweicht, besteht die Abweichung der Verbrauchsschätzung am Ende der Fahrt aus nur noch 0,09 Liter. Dies entspricht einer Abweichung von 3,2%. Abbildung 5.25d zeigt die prozentualen Abweichungen der Verbrauchsschätzungen für die drei vorparametrisierten Profile und die individuelle Schätzung. Die kontinuierliche Verbesserung der Schätzung zeigt Abbildung 5.25c. Durch die anfangs sehr hohe Geschwindigkeit von 120 km/h in der 80 km/h Zone bei Streckenkilometer 2 ist zunächst eine starke Abweichung der Verbrauchsschätzung zu sehen, welche jedoch mit zunehmender Fahrt verschwindet.

Abbildung 5.26 zeigt die initialen Verbrauchs- und Geschwindigkeitsprognosen für die Rückfahrt. Obwohl sich die Geschwindigkeitsbeschränkungen von denen der Hinfahrt unterscheiden, fällt die Verbrauchsprognose für die Rückfahrt genauer aus als die Prognose über die vorparametrisierten Profile. Für die vorparametrisierten Profile liegt für die Rückfahrt ebenfalls die sportliche Schätzung mit einer Abweichung von 15,3% bei geschätzten 3,16 Liter Verbrauch der Realität von 2,70 Litern am nächsten. Die individuelle Verbrauchsprognose schätzt den Verbrauch mit einer Abweichung von 3,8% (geschätzte 2,8 Liter) am Besten.



(a) Initiale Verbrauchsprognose. Die Komfortparameter sind noch nicht individuell auf den Fahrer angepasst und gleichen dem Standard. Von den drei vorparametrisierten Profilen prognostiziert die sportliche Schätzung den Benzinverbrauch am besten.

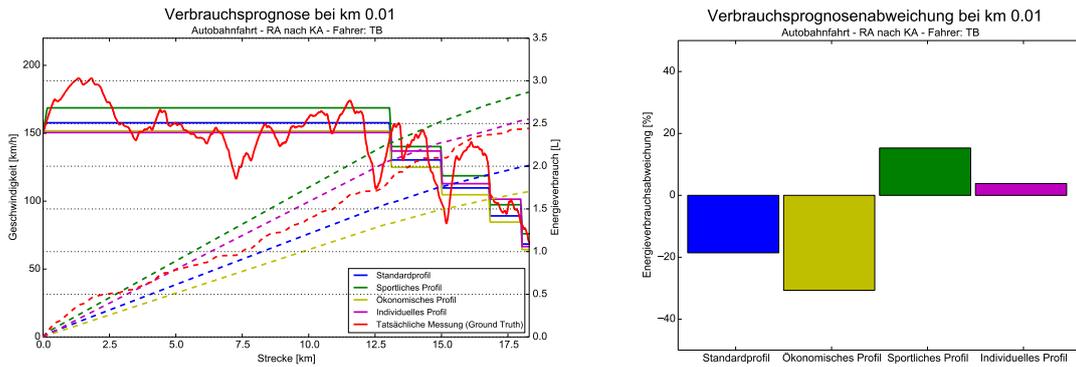
(b) Verbrauchsprognose mit initialisierten Komfortparametern. Am Ende der Autobahnfahrt weicht die individuelle Schätzung des Benzinverbrauchs nur noch geringfügig (0,09 Liter) vom echten Verbrauch ab.



(c) Während der Fahrt wird die Verbrauchsschätzung sukzessive verbessert. Die Schätzung weicht anfangs bis zu 20% vom echten Benzinverbrauch ab. Gegen Ende der Fahrt beträgt die Abweichung nur noch 3,2%. Durch die hohe Geschwindigkeit von 120 km/h auf der Autobahnzufahrt (beschildert mit 80 km/h bei Streckenkilometer 2), wird anfangs ein sehr hoher Energieverbrauch angenommen.

(d) Gegenüberstellung der prozentualen Abweichungen der Verbrauchsprognosen der drei vorparametrisierten Profile und der individuellen Verbrauchsschätzung, welche das Fahrverhalten des Fahrers über die Komfortparameter berücksichtigt.

Abb. 5.25.: Verbrauchsprognose einer 18 km lange Autobahnfahrt von Karlsruhe nach Rastatt. Während der Fahrt verbessert sich die Verbrauchsschätzung durch die berechneten Komfortparameter sukzessive. Während die Schätzung anfangs bis zu 20% vom echten Benzinverbrauch abweicht, beträgt die Abweichung gegen Ende der Fahrt nur noch 3,2%. In den Abbildungen 5.25a und 5.25b zeigen die gestrichelten Linien den prognostizierten Energieverbrauch und die durchgezogenen Linien den berechneten Geschwindigkeitsverlauf. Die rote Linie zeigt den am Versuchsträger gemessenen Geschwindigkeitsverlauf und den gemessenen Kraftstoffverbrauch.



(a) Geschätzter Geschwindigkeitsverlauf und Verbrauch vor Antritt der Rückfahrt.

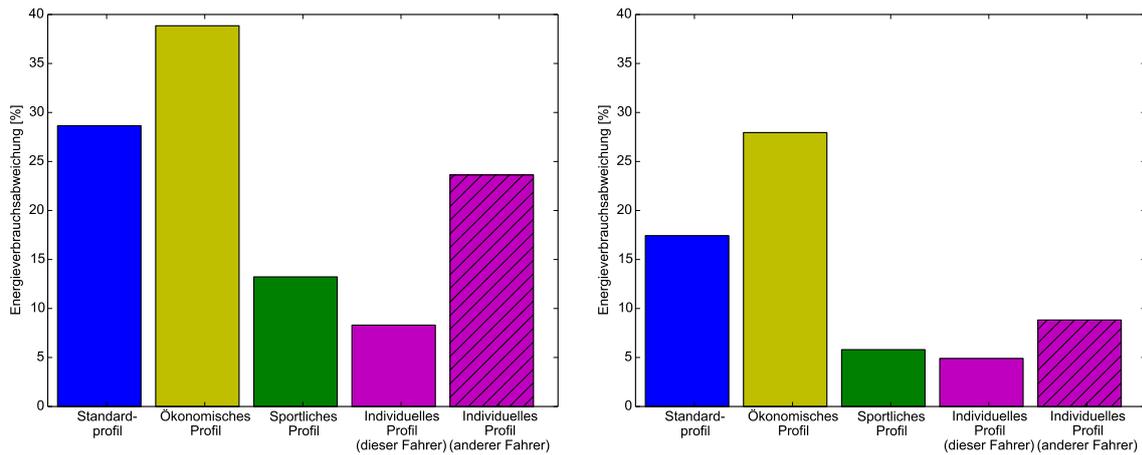
(b) Vergleich der Verbrauchsschätzungen für die Rückfahrt.

Abb. 5.26.: Verbrauchsprognose für die Rückfahrt von Rastatt nach Karlsruhe. Obwohl auf der Rückfahrt andere Geschwindigkeitsbeschränkungen und Verkehrsgegebenheiten existieren, schneidet die individuelle Verbrauchsschätzung besser ab als die drei vorparametrisierten Profile.

5.3.2. Verbrauchsprognose für unterschiedliche Fahrer

Durch unterschiedliches Fahrverhalten haben verschiedene Fahrer auf gleicher Strecke unterschiedliche Energieverbräuche. Über den Vergleich von Verbrauchsprognosen zweier unterschiedlicher Fahrer auf der gleichen Strecke soll gezeigt werden, dass nur die eigenen Komfortparameter zur besseren Verbrauchsprognose führen. Eine Verbrauchsprognose basierend auf fahrerfremden Komfortparametern kann durchaus zu schlechteren Ergebnissen führen.

Die Abbildungen 5.27a und 5.27b zeigen sowohl die prozentualen Abweichungen der Verbrauchsschätzungen für die vorparametrisierten Fahrerprofile als auch die mit den eigenen und den fahrerfremden Komfortparametern. Während für die Turmbergstrecke der Verbrauch von Fahrer TB mit den eigenen Komfortparametern auf 0,31 Liter genau bestimmt werden kann (Abweichung von 8%), liefern die fahrerfremden Komfortparameter eine Abweichung von 0,88 Liter (23%). Auch für Fahrer DN liefern die eigenen Komfortparameter eine genauere Schätzung als die fahrerfremden. Die Abweichung liegt bei 0,11 Liter (4,9%) für die eigenen Parameter und 0,21 Liter (8,8%) für die fahrerfremden Parameter.



- (a) Abweichungen des geschätzten Verbrauchs für den Fahrer *TB*. Parametrisiert mit den eigenen Komfortparametern weicht die Verbrauchsprognose um 8% vom realen Verbrauch ab. Mit den Komfortparametern eines anderen Fahrers (*DN*) parametrisiert, weicht die Schätzung um circa 24% ab.
- (b) Abweichungen des geschätzten Verbrauchs für den Fahrer *DN*. Parametrisiert mit den eigenen Komfortparametern weicht die Verbrauchsprognose um 5% vom realen Verbrauch ab. Mit den Komfortparametern eines anderen Fahrers (*TB*) parametrisiert, weicht die Schätzung um circa 8% ab.

Abb. 5.27.: Verbrauchsprognosen zweier unterschiedlicher Fahrer (*DN* und *TB*) auf nahezu identischer Fahrstrecke. Die Verbrauchsprognosen basieren jeweils auf den eigenen sowie den Komfortparametern des anderen Fahrers. Die Abbildung zeigt, dass nur die eigenen Komfortparameter die Verbrauchsprognose tatsächlich verbessern.

5.4. Zustandserkennung und Prognose des Fahrverhaltens

In Abschnitt 4.5 werden über das Rubikon-Modell die vier Handlungsphasen beschrieben, welche einer bewussten, zielgerichteten Handlung zu Grunde liegen: *[Abwägen]*, *[Planen]*, *[Handeln]*, *[Reflektieren]*. Die Handlungsphasen sind dabei nicht direkt beobachtbar. Es gibt lediglich Evidenzen, die Hinweise für die entsprechenden Handlungsphasen sind. Sowohl die Aussage, dass sich ein Fahrer in einer bestimmten Handlungsphase befindet als auch die Tatsache, dass eine Beobachtung einer Handlungsphase zuzuordnen ist, sind wahrscheinlichsbehaftet. Dies führt zu einem doppelt-stochastischen Prozess der sich, wie in Abschnitt 4.5 beschrieben, über prädikatenlogische Formeln durch ein *Markov Logik Netz* modellieren lässt.

Als Tauglichkeitsnachweis werden für diese Modellierung anhand dem Szenario „Überholvorgang auf der Autobahn“ zwei wesentliche Fragestellungen untersucht:

- Wie gut kann die aktuelle Handlungsphase des Fahrers erkannt werden (als Manövererkennung bezeichnet)?
- Wie gut können zukünftige Handlungen prognostiziert werden und über welchen Zeithorizont (als Manöverprädiktion bezeichnet)?

Abbildung 5.28 zeigt eine sechsminütige Autobahnfahrt mit neun getätigten Überholmanövern. Die oberste Zeile zeigt die laterale Fahrzeugposition relativ zum rechten Fahrbahnrand der dreispurigen Autobahn.

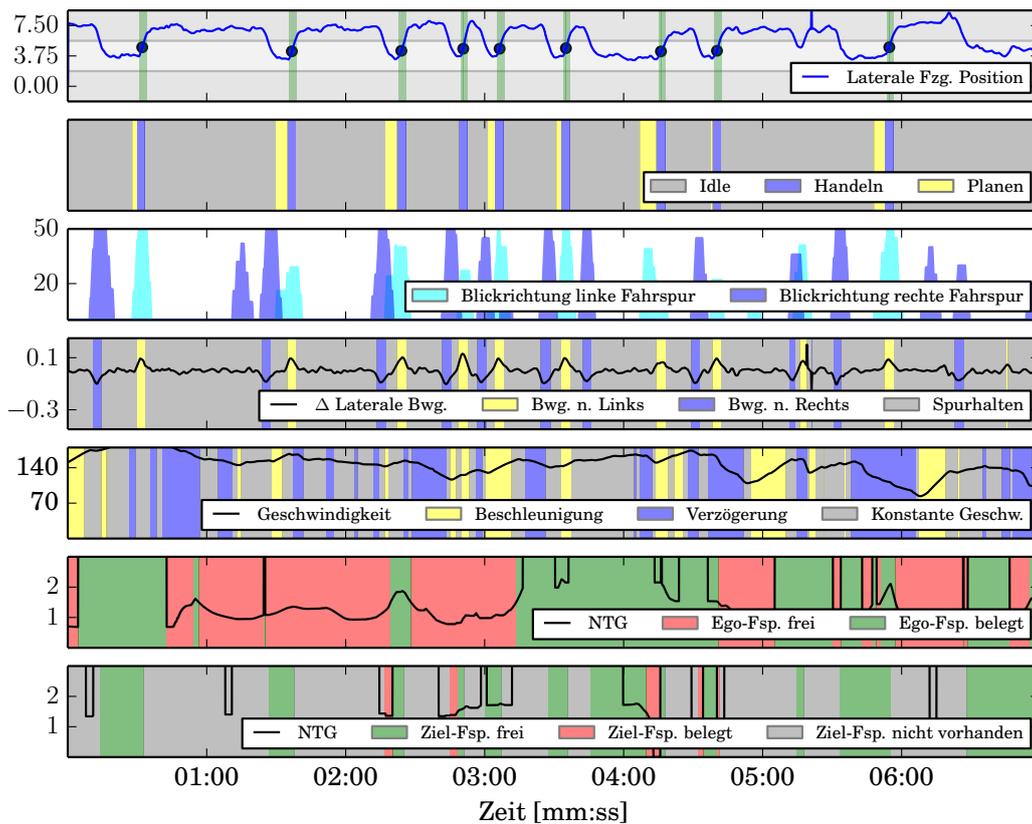


Abb. 5.28.: Sechsminütige Autobahnfahrt mit neun annotierten Spurwechseln. Die erste Zeile zeigt die laterale Position des Fahrzeugs relativ zum rechten Fahrbahnrand. In der zweiten Zeile sind die annotierten Handlungsphasen farblich dargestellt. Die dritte Zeile zeigt die Anzahl an Messungen, auf denen in den letzten fünf Sekunden ein Blick nach links bzw. nach rechts detektiert wurde. Zeile vier zeigt die laterale Bewegung des Fahrzeugs und die Zeilen fünf und sechs die Belegtheit der Ego- bzw. Zielfahrspur.

Fährt der FZI-Versuchsträger CoCar (Spurbreite 1,6 m) mittig auf einer Autobahn mit einer typischen Spurbreite von 3,75 m besteht zur linken und zur rechten Fahrspur ein Abstand von 1,075 m. Das *Ereignis Spurwechsel* ist definiert als das Überschreiten von 90% dieses Abstands mit anschließender kontinuierlicher Lateralbewegung in Richtung der Zielspur. In Abbildung 5.28 sind die *Ereignisse Spurwechsel* durch blaue Markierungen dargestellt. In den Messdaten wurde das Zeitfenster 1 Sekunde vor und 2 Sekunden nach dem *Ereignis Spurwechsel* als *Spurwechsel* markiert. Die grün markierten Bereiche zeigen die Zeitfenster des *Spurwechsels* nach links, welche das Überholmanöver einleiteten.

Wird vom Erkennungsalgorithmus in diesem Bereich ein Spurwechsel erkannt, gilt das Ergebnis als *Richtig-Positiv*. Wird in diesem Bereich vom Erkennungsalgorithmus hingegen kein Spurwechsel erkannt, gilt das Ergebnis als *Falsch-Negativ*. Wird außerhalb dieses Zeitfensters ein Spurwechsel erkannt bzw. nicht erkannt, handelt es sich um ein *Falsch-Positives* bzw. *Richtig-Negatives* Ergebnis.

Die Annotation der Handlungsphase wird in Abbildung 5.28 in der zweiten Zeile farblich dargestellt. Als *Planungsphase* wurde das Zeitfenster vor einem Spurwechsel markiert, in dem der Fahrer durch mehrfache Blickbewegungen in den Rückspiegel und den Innenspiegel den rückwärtigen Verkehr beobachtet. Die Blickbewegungen des Fahrers sind in der dritten Zeile der Abbildung dargestellt. Aufgetragen ist die Anzahl der Bilder im Zeitfenster der letzten 5 Sekunden, in denen eine Blickbewegung zur linken bzw.

zur rechten Fahrspur detektiert wurde.

Als *Handlungsphase* wurde das Zeitfenster annotiert, in dem der Fahrer das Fahrzeug lateral Richtung Zielspur bewegt. Die vierte Zeile des Plots zeigt die Ableitung der lateralen Bewegung des Fahrzeugs. Wurde vom Experten weder eine *Planungs-* noch eine *Handlungsphase* beobachtet, wurde der Zeitschritt mit dem Label „Idle“ markiert.

Auf die Annotation der *Reflektionsphase* und der *Abwägephase* wurde verzichtet. Die *Reflektionsphase* findet nach dem Spurwechsel statt und bringt für die Prädiktion keinen Mehrwert. In der *Abwägephase* beschäftigt sich der Fahrer rein gedanklich mit dem Spurwechsel, was keinerlei Evidenz beobachten lässt.

Die Zeilen 5–7 des Diagramms 5.28 zeigen die longitudinale Bewegung des Fahrzeugs (Zeile 5) sowie die Belegtheit der Ego-Fahrspur (Zeile 6) bzw. der Zielfahrspur (Zeile 7). Die Belegtheit einer Fahrspur definiert sich über den *Net-Time-Gap* — der Zeitspanne die vergeht, bis das Ego-Fahrzeug das vorausfahrende Fahrzeug bei gleichbleibender Geschwindigkeit beider Fahrzeuge erreicht [Cos+03, Definition 1, Seite 183].

Das Blickverhalten, die Belegtheit der Fahrspuren, die laterale und die longitudinale Bewegung des Fahrzeugs stehen dem *MLN* als diskrete Beobachtungen zur Schätzung der verborgenen Handlungszustände zur Verfügung.

Abschnitt 5.4.1 beschreibt die Konzeption des *MLNs* zur Erkennung der aktuellen Handlungsphase des Fahrers. In Abschnitt 5.4.2 werden Ergebnisse der Trainingsphase (dem Lernen der Gewichte) dargestellt. Die Abschnitte 5.4.3 und 5.4.4 beschreiben, wie das konzipierte und über Trainingsdaten gelernte Netz zur Erkennung der aktuellen Handlungsphase sowie zur Manöverprädiktion genutzt wird (Inferenz).

5.4.1. Konzeption des *MLNs*

Textauszug 5.29 definiert das zur Erkennung der Handlungsphase verwendete *MLN*. Es setzt die im Konzeptkapitel (Abschnitt 4.5) formulierten, prädikatenlogischen Formeln 4.16 bis 4.20 (Seite 103) zur Manövererkennung um.

In Zeile 2 wird die Handlungsphase des Fahrers als Funktion *State* definiert. Die Variable *idx* steht für einen eindeutigen Index eines Test- oder Trainingsbeispiels. *time* steht für einen diskreten Zeitpunkt ($t = 0, 1, 2, 3, \dots$), und *state* steht für die Handlungsphase ($s = [Start, Idle, Planen, Handeln]$). Das Ausrufezeichen bei der Deklaration der *state* Variable gibt an, dass die Handlungsphase über der Zeit und dem Index eindeutig ist. Der Fahrer kann sich beispielsweise zu einem Zeitpunkt nicht gleichzeitig in der *[Planen]* und der *[Handeln]* Phase befinden. In Zeile 5 wird in ähnlicher Weise die Beobachtung (als Funktion *Observation*) definiert. Die Beobachtung umfasst den Fahrer in Form seiner Blickrichtung über die Variable *gaze*, das lokale Fahrzeugumfelds in Form der Belegtheit der eigenen und der Zielfahrspur über die Variablen *ego_lane* bzw. *tgt_lane* und die Fahrzeugbewegung in Form der lateralen Bewegung des Autos über die Variable *movement*. Wieder sind die Beobachtungen gegenüber der Zeit und dem Index eindeutig. In den Zeilen 8 bis 11 wird diese Eindeutigkeit als Formel beschrieben.

Die Zeilen 14 bis 18 beinhalten vorgegebenes Expertenwissen. Es wird definiert, dass sich das System zum Zeitpunkt $t = 0$ im Zustand *[Start]* befindet (Zeile 14). Die Zeilen 17 und 18 formulieren, dass das

```

1 // Query (Open World)
2 State( idx , time , state! )
3
4 // Input (Closed World)
5 Observation( idx , time , gaze! , egolane! , tgtlane! , movement! )
6
7 // An observation is unique
8 !Observation( i , t , g1 , e , r , m ) v !Observation( i , t , g2 , e , r , m ) v [ contains( g1 , g2 ) ].
9 !Observation( i , t , g , e1 , r , m ) v !Observation( i , t , g , e2 , r , m ) v [ contains( e1 , e2 ) ].
10 !Observation( i , t , g , e , r1 , m ) v !Observation( i , t , g , e , r2 , m ) v [ contains( r1 , r2 ) ].
11 !Observation( i , t , g , e , r , m1 ) v !Observation( i , t , g , e , r , m2 ) v [ contains( m1 , m2 ) ].
12
13 // Sure things - System starts at "Start"
14 State( i , t , Start ) v [ t != 0 ].
15
16 // Neither a state nor an observation leads to the "Start" state
17 State( i , t , s ) , [ t != 0 ] => !State( i , t , Start ).
18 State( i , t1 , s ) , Observation( i , t1 , g , e , r , m ) , [ t1 + 1 = t2 ] => !State( i , t2 , Start ).
19
20 // The things to learn
21 1.0 State( i , t1 , s1+ ) , Observation( i , t1 , g+ , e+ , t+ , m+ ) , [ t1 + 1 = t2 ] => State( i , t2 , s2+ )

```

Abb. 5.29.: *Markov Logik Netz* zur Erkennung der aktuellen Handlungsphase des Fahrers.

System zu keinem Zeitpunkt wieder in den *Start* Zustand übergeht und es keine Beobachtung und keinen Zustand enthält, aus dem es wieder in den *Start* Zustand überführt wird. Der an die Formel angehängte Punkt signalisiert, dass dieser Zusammenhang mit absoluter Sicherheit gilt. Das Gewicht der Formel wird über den Punkt auf unendlich gesetzt. In Zeile 21 wird der zu lernende Zusammenhang formuliert: Das System befindet sich zu einem Zeitpunkt t in einem Zustand und wird mit einer Beobachtung in einen Folgezustand ($t \rightarrow t + 1$) überführt. Das Plus Zeichen hinter den Variablen bewirkt, dass im *MLN* beim Lernen eine Formel für jede Variablenbelegung entsteht. Die Formel aus Zeile 21 wird mit den möglichen Variablenbelegungen

```

states = [ "Action", "Planning" , "Idle" , "Start" ]
gaze   = [ "GazeLeft" , "GazeStraight" ]
ego    = [ "Occupied" , "Free" ]
target = [ "Occupied" , "Free" , "Invalid" ]
move   = [ "Left" , "NotLeft" ]

```

zu $4 \cdot 2 \cdot 2 \cdot 3 \cdot 2 \cdot 4 = 384$ Formeln „ausgerollt“. Diese große Anzahl von Formeln führt beim Lernen und der Inferenz im *MLN* zu erhöhtem Rechenaufwand. In der Praxis wurden deshalb nur die Kombinationen gebildet, die in der Test- und Trainingsmenge tatsächlich vorkommen.

Das der Formel vorangestellte Gewicht 1.0 wird in der Trainingsphase durch ein aus Trainingsbeispielen gelerntes Gewicht ersetzt.

5.4.2. Trainieren des Netzes

Als *Trainieren* des *MLNs* wird das Bestimmen der den Formeln assoziierten Gewichte durch Trainingsbeispiele bezeichnet. Üblich ist eine $2/3$ zu $1/3$ Gewichtung von Trainings- und Testdaten. Beim Trainieren

werden die Beobachtungen und die annotierten Zustände zeitlich diskretisiert und dem Netz als Trainingsbeispiele zur Verfügung gestellt. Zur zeitlichen Diskretisierung wurde eine Diskretisierungsschrittweite von einer Sekunde gewählt. Ein Trainingsbeispiel umfasst die letzten fünf Sekunden aus annotierten Zuständen und Beobachtungen. Im Textauszug 5.30 sind zwei Trainingsbeispiele gezeigt.

```

State( 26 , 0 , Start )
Observation( 26 , 0 ,
GazeStraight , Occupied , Free , NotLeft )
State( 26 , 1 , Idle )
Observation( 26 , 1 , GazeStraight , Occupied , Free , NotLeft )
State( 26 , 2 , Idle )
Observation( 26 , 2 , GazeLeft , Occupied , Free , NotLeft )
State( 26 , 3 , Planning )
Observation( 26 , 3 , GazeLeft , Occupied , Free , NotLeft )
State( 26 , 4 , Planning )
Observation( 26 , 4 , GazeLeft , Occupied , Free , Left )
State( 26 , 5 , Action )

State( 134 , 0 , Start )
Observation( 134 , 0 , GazeStraight , Free , Invalid , NotLeft )
State( 134 , 1 , Idle )
Observation( 134 , 1 , GazeStraight , Free , Invalid , NotLeft )
State( 134 , 2 , Idle )
Observation( 134 , 2 , GazeStraight , Free , Occupied , NotLeft )
State( 134 , 3 , Idle )
Observation( 134 , 3 , GazeLeft , Free , Occupied , NotLeft )
State( 134 , 4 , Planning )
Observation( 134 , 4 , GazeLeft , Free , Occupied , NotLeft )
State( 134 , 5 , Planning )

```

Abb. 5.30.: Über Trainingsbeispiele werden die den Formeln assoziierten Gewichte des *MLNs* gelernt.

Textauszug 5.31 zeigt Beispiele der gelernten Zusammenhänge. Als unwahrscheinlich (kleines Gewicht) gilt der Zusammenhang, dass auf die Handlungsphase *Idle* mit der Beobachtung

`Observation(v0, v1, "GazeLeft", "Occupied", "Free", "Left")`

die Handlungsphase *Action* folgt (Zeile 1). Sehr wahrscheinlich (großes Gewicht) ist, dass das System in der Handlungsphase *Idle* bleibt, wenn die Beobachtung

`Observation(v0, v1, "GazeStraight", "Free", "Invalid", "NotLeft")`

vorliegt (siehe Zeile 3).

```

1  0.0195 !State(v0, v1, "Handeln") v !Observation(v0, v1, "GazeLeft", "Occupied", "Free", "Left") v State(v0,
   ↪ v2, "Idle") v [NOT ((v1 + 1) = v2)]
2  ...
3  18.1342 !State(v0, v1, "Idle") v !Observation(v0, v1, "GazeStraight", "Free", "Invalid", "NotLeft") v
   ↪ State(v0, v2, "Idle") v [NOT ((v1 + 1) = v2)]

```

Abb. 5.31.: Auszug aus dem über die Trainingsdaten gelernten *MLN*.

5.4.3. Bestimmung der aktuellen Handlungsphase - Inferenz

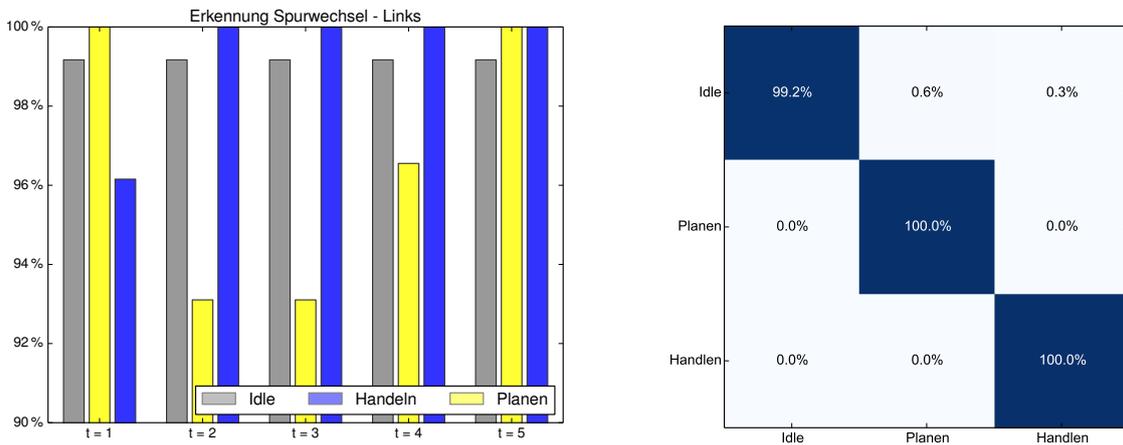
Die Bestimmung der aktuellen Handlungsphase (Inferenz) basiert auf den letzten 5 Beobachtungen, was mit einer Beobachtung pro Sekunde einem Zeitfenster der letzten 5 Sekunden entspricht. Im

Inferenzprozess wird für jeden Zeitpunkt ($t = 1, \dots, 5$) eine Handlungsphase (Zustand) geschätzt.

- Der Zustandserkennung zum Zeitpunkt $t = 1$ liegt die erste Beobachtung ($t = 0$) zu Grunde.
- Der Zustandserkennung zum Zeitpunkt $t = 2$ liegt die Zustandserkennung zum Zeitpunkt $t = 1$ und die Beobachtung zum Zeitpunkt $t = 1$ zu Grunde.
- ...
- Der Zustandserkennung zum Zeitpunkt $t = 5$ liegt die Zustandserkennung zum Zeitpunkt $t = 4$ und die Beobachtung zum Zeitpunkt $t = 4$ zu Grunde.

Abbildung 5.32a zeigt die Erkennungsleistung des Netzes bezüglich der Zustandserkennungen zu den einzelnen Zeitpunkten. Die aktuelle Handlungsphase ($t = 5$) kann mit sehr hoher Sicherheit richtig erkannt werden. Lediglich in 0,8 Prozent der Fälle wird die *Idle* Phase mit der *Handlungsphase* oder der *Planungsphase* verwechselt. Für die Zustände *Handeln* und *Planen* konnte eine 100% Erkennungsrate erzielt werden, der Zustand *Idle* wird mit 99% Wahrscheinlichkeit richtig geschätzt. Eine Konfusionsmatrix zeigt Abbildung 5.32b.

Abbildung 5.33 zeigt ein Testbeispiel, wie es dem Netz zur Inferenz vorliegt.



(a) Erkennungsleistung der aktuellen Handlungsphase des MLN. (b) Die Zustände [Idle], [Planen] und [Handeln] können nahezu verwechslungsfrei unterschieden werden.

Abb. 5.32.: Die aktuelle Handlungsphase kann verwechslungsfrei erkannt werden. Es werden lediglich 0,8 Prozent der als *Idle* annotierten Zeitschritte mit der *Handlung* oder der *Planung* verwechselt. Für die Zustände [Handeln] und [Planen] konnte eine Erkennungsrate von 100% erzielt werden.

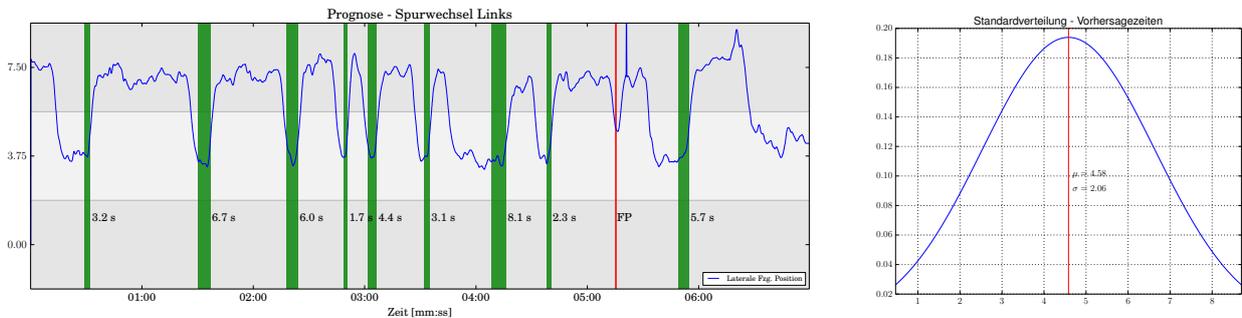
```

State( 413 , 0 , Start )
Observation( 413 , 0 , GazeStraight , Free , Free , NotLeft )
Observation( 413 , 1 , GazeStraight , Free , Free , NotLeft )
Observation( 413 , 2 , GazeStraight , Occupied , Free , NotLeft )
Observation( 413 , 3 , GazeStraight , Occupied , Free , NotLeft )
Observation( 413 , 4 , GazeStraight , Free , Free , NotLeft )
    
```

Abb. 5.33.: Testbeispiel (Eingabe) für den Inferenzprozess.

5.4.4. Prognose des Fahrspurwechsels

Mit der Annahme, dass auf eine Planung eine aktive Handlung folgt, kann ein Spurwechsel mit dem Erkennen der Planungsphase prognostiziert werden. Abbildung 5.34a zeigt über das Erkennen der Planungsphase prognostizierte Spurwechsel nach links mit den erzielten Vorhersagezeiten. Eine Prognose der Aktion des Fahrers über das Erkennen der Planungsphase muss aus folgenden Gründen kritisch betrachtet werden:



(a) Prädiktion des Spurwechsels auf Basis einer erkannten Planungsphase. Im Durchschnitt konnte ein Spurwechsel 4,93 Sekunden vor dem Eintreten erkannt werden. Als Vorhersagezeiten wurden Zeiten zwischen 2 und 8 Sekunden erzielt (Standardabweichung $\sigma = 2,06$).

(b) Normalverteilung über die erzielten Vorhersagezeiten.

Abb. 5.34.: Wird über das *MLN* eine Planungsphase erkannt, kann davon ausgegangen werden, dass dieser eine aktive Handlungsphase folgt. Der Eintrittszeitpunkt des Spurwechsels ist bei dieser Vorgehensweise nicht eindeutig bestimmbar.

Führt der Fahrer nach der Planungsphase *keinen* Spurwechsel durch (d.h. der Fahrer handelt entgegen der oben formulierten Annahme) wird fälschlicherweise ein Spurwechsel erkannt. Die Wahrscheinlichkeit, dass ein Spurwechsel geplant war aber abgebrochen wird, lässt sich über die Messdaten quantifizieren. Gleichzeitig ist die Erkennung der Planungsphase ist mit einer gewissen *Fehlerwahrscheinlichkeit* behaftet, welche sich über die Testdaten bestimmen lässt.

Der Eintrittszeitpunkt des Spurwechsels ist nicht eindeutig bestimmbar. Die Vorhersagezeiten des Spurwechsels schwanken zwischen 2 und 8 Sekunden (Durchschnitt = 4,93 s, Standardabweichung = 1,90 s). Je nach Anwendung ist eine wohldefinierte Vorhersagezeit erforderlich.

Im Wesentlichen terminiert der Fahrer den Zeitpunkt des Fahrspurwechsels nach seiner Einschätzung, wann die Zielfahrspur frei wird oder wann die eigene Fahrspur durch einsicherende Fahrzeuge belegt sein wird. Um die Vorhersagezeit zu präzisieren ist eine Prädiktion der aktuellen Fahrsituation nötig. Die Prädiktion der aktuellen Fahrsituation, welche zur Ermittlung genauerer Vorhersagezeiten führen könnte, ist Gegenstand aktueller Forschungsarbeiten.

Im Durchschnitt konnte ein Spurwechsel 4,93 Sekunden vor dem Eintreten erkannt werden. Die Standardabweichung beträgt dabei 1,9 s und die Fehlerquote 10%. Abbildung 5.34b zeigt die erreichten Vorhersagezeiten als Normalverteilung.

5.5. Zusammenfassung

Bei der Evaluation der Blickrichtungserkennung wurde speziell auf die Gegebenheiten der Domäne „Automobil“ eingegangen. Die Eignung der entstandenen Systeme zur Blickrichtungserkennung wurden nach den in Abschnitt 3.1 (Seite 3.1) formulierten Anforderungen bewertet.

Die Forderung der **Hardware-Integrationsmöglichkeit** und **Sicherheit** wurde durch die Wahl von nicht-intrusiver, im Fahrzeug kompakt integrierbarer Sensorik erfüllt. Zum **Mehrbenutzersystem** wird die Blickrichtungserkennung durch die Identifikation des Fahrers und die Möglichkeit fahrerindividuelle Gesichtsmasken zu laden. Die **Kompensation von Verdeckungen** wurde qualitativ in [RBZ12] untersucht. Die geforderte **Reproduzierbarkeit** ist durch die Wahl der Algorithmen gegeben. In diesem Kapitel wurden speziell folgende Anforderungen evaluiert:

Genauigkeit: Mit einem Toleranzbereich von 4° um die Gierachse lässt sich die Kopfpose mit einer Erkennungsrate von 82% bestimmen. Die Genauigkeit der Kopfpose wurde relativ zur annotierten Kopfstellung evaluiert. Ist das Gesicht der Kamera nicht zu sehr abgewandt steigt die Erkennungsrate bei einem Toleranzbereich von 5° auf bis zu 91%.

Unter Berücksichtigung der Augenstellung ist die entworfene Blickrichtungserkennung genau genug, um den Blick auf Sichtfelder im Fahrzeuginnenraum und -außenraum abzubilden (siehe Abbildung 5.5). Selbst Blicke in den Innenspiegel, welche oft nur über eine Änderung der Augenstellung getätigt werden, konnten zu 98,2% richtig zugeordnet werden.

Schnelligkeit der Kopfbewegung: Die Kopfpose konnte selbst bei Rotationsgeschwindigkeiten von bis zu $600^\circ/s$ um die Gierachse bestimmt werden. In der kompletten FZI-Messdatenbank, welche eine Vielzahl von Verkehrsszenen beinhaltet, wurden maximale Rotationsgeschwindigkeiten von $500^\circ/s$ gemessen.

Echtzeitfähigkeit: Die Echtzeitfähigkeit ist bedingt durch die unterliegende Hardware. Auf einem 4-Kern Intel® Core™ i7-2700K CPU @ 3.50GHz Prozessor beträgt die durchschnittliche Verarbeitungszeit der ICP-Schätzung bei möglicher Transformation 13,30 ms (Varianz: 0,86 ms).

Erkennungsbereich: Durch die Trennung der Gesichtsmaske in eine rechte und eine linke Hälfte kann ein Erkennungsbereich von $\pm 90^\circ$ abgedeckt werden. Die Genauigkeit nimmt im Randbereich allerdings ab.

Kompensation wechselnder Lichtverhältnisse: Ist das Tiefenbild gestört, lässt sich die Kopfpose über *Random (Regression) Forests (RFs)* aus dem RGB-Farbbild bestimmen. Im zentralen Bereich kann dies mit einer Abweichung unter 6° in Nick- und Gierwinkel erfolgen (siehe Abschnitt 5.1.4 Seite 113). Bei stärkeren Kopffrotationen beträgt die Abweichung circa 9° zur realen Kopfstellung.

Für die Blickrichtungserkennung kann festgestellt werden, dass die umgesetzte Blickrichtungserkennung die zentralen Anforderungen der Automobilindustrie erfüllt.

Die Fahrstilerkennung wurde im FZI-Smart-Fahrsimulator anhand einer 10 km langen Teststrecke mit Landstraßencharakter evaluiert. Ein Experte fungiert als Beifahrer und unterteilt die Fahrten nach

der Risikobereitschaft der Probanden in die Klassen [*unsicher*], [*moderat*], [*sportlich*] und [*aggressiv*]. Hinsichtlich des Energieverhaltens werden die Fahrten nach dem Energieverbrauch der Messfahrt in [*verschwendend*], [*moderat*] und [*ökonomisch*] aufgeteilt.

In der Evaluation werden zwei Varianten der Fahrstilerkennung verglichen. Über *MLNs* könnten leicht höhere Erkennungsraten erzielt werden. Während mit der Fahrstilklassifikation über *Fuzzylogik* hinsichtlich des Energieverbrauchs 93,5% der Messdaten richtig zugeordnet werden, waren es über *MLNs* 96,8%. Die Risikoklasse wurde über *Fuzzylogik* zu 85,5% richtig geschätzt. Vor allem für komplexe Szenen, wie der Ortsanfahrtsszene, fällt die Erkennungsleistung der *MLNs* besser aus als die des *Fuzzy-Inferenzsystem*, da es dem Experten schwer fällt, einen konsistenten Fuzzyregelsatz aufzustellen. Ein maschinelles Lernen der Zusammenhänge, wie über die *MLNs* realisiert, führt in komplexen Szenen zu besseren Ergebnissen.

Die abgeleiteten Komfortparameter wurden anhand des Anwendungsfalls der Energieprognose auf ihre Tauglichkeit untersucht. Dazu wurde auf einer realen Autobahnfahrt von Karlsruhe nach Rastatt eine a-priori Energiebedarfsschätzung *ohne* fahrerindividuelle Komfortparameter mit einer a-posteriori Schätzung *mit* fahrerindividuellen Komfortparametern verglichen. Während die initiale Verbrauchsschätzung 20% vom tatsächlich gemessenen Wert abweicht, weicht die Schätzung am Ende der Messfahrt nur noch 3,2% vom tatsächlich gemessenen Energieverbrauch ab. Für die Rückfahrt, auf der andere Verkehrsaufkommen und Geschwindigkeitsbeschränkungen existieren, kann der Energieverbrauch mit einer Abweichung von 3,82% geschätzt werden. Die Schätzung ohne fahrerindividuelle Komfortparameter weicht hingegen 15,29% vom tatsächlich gemessenen Verbrauch ab.

Um die Individualität der Komfortparameter zu zeigen werden zwei unterschiedliche Fahrer auf nahezu identischer Strecke verglichen. Der Energiebedarf wurde dabei für beide Fahrer jeweils mit den eigenen, als auch mit den Komfortparametern des anderen Fahrers prognostiziert. In beiden Fällen schnitt die Verbrauchsprognose parametrisiert über die eigenen Komfortparameter deutlich besser ab (8% zu 23% und 5% zu 9%).

Auch die *MLN-Modellierung* zur Erkennung der aktuellen Handlungsphase wird anhand des Szenarios „Überholvorgang auf der Autobahn“ untersucht. Die Evaluierung beschränkt sich dabei auf den Nachweis der Machbarkeit (*Proof of Concept*). Eine Evaluierung über mehrere Stunden Autobahnfahrt mit einer Vielzahl an verschiedenartig durchgeführten Überholvorgängen von unterschiedlichen Fahrern konnte an dieser Stelle nicht geleistet werden.

Die Handlungsphasen [*Idle*], [*Planen*] und [*Handeln*] werden im *MLN* nahezu verwechslungsfrei erkannt (Idle: 99%, Planen: 100%, Handeln: 100%). Die aktuelle Handlungsphase wird über das *MLN* probabilistisch aus der Beobachtungshistorie der letzten fünf Sekunden inferiert. Als Beobachtungen fließen Daten über den Fahrer (Blickrichtung), das Umfeld (Belegtheit der eigenen Fahrspur und der Zielfahrspur) und der lateralen Bewegung des Autos ein.

Folgt man aus einer erkannten Planung eine nachfolgende aktive Handlung kann darüber ein bevorstehender Spurwechsel des Fahrers prognostiziert werden. Im Durchschnitt kann so ein Spurwechsel 4,93 Sekunden vor dem Eintreten erkannt werden (Standardabweichung 1,9 s, Fehlerquote 10%).

6. Schlussbetrachtung

6.1. Zusammenfassung

Zukünftige Generationen von Fahrerassistenzsystemen dienen dem Fahrer als *kooperativer Begleiter*; Fahrer und Fahrzeug interagieren Hand in Hand und lösen die Fahraufgabe gemeinsam. Diese Kooperation fordert intuitive Übergabestrategien und damit eine maschinelle Einschätzung des Fahrers, dessen Situationsbewusstseins und dessen situativen Verhaltens. Diese Arbeit leistet einen Beitrag zur Steigerung der Mensch-Technik-Interaktionsfähigkeit intelligenter Automobile durch eine situative Betrachtung der sich wechselseitig beeinflussenden und nur partiell beobachtbaren Komponenten *Fahrer*, *Fahrzeug* und *Fahrzeugumfeld* (Situationsanalyse).

Über das durchgängige Konzept, welches Fahrerassistenzsystemen erstmals Wissen auf *allen* drei Ebenen zur Einschätzung des Situationsbewusstseins (*Wahrnehmen*, *Verstehen* und *Projizieren* [End95]) bereitstellt, ist die Adaption von Assistenzsystemen an die Eigenschaften des individuellen Fahrers möglich. Beiträge zur Verbesserung von Sicherheit und Effizienz werden geleistet, indem eine speziell auf die Ansprüche der Automobil-Domäne optimierte Blickrichtungserkennung entwickelt wurde, über die das Blickverhalten des Fahrers in Bezug zum situativen Verkehrskontext analysiert wird.

Die Blickrichtungserkennung (die Schätzung der Kopfpose und der Pupillenstellung) wird über optische Sensoren realisiert. Für den Automobilbetrieb besonders tauglich erwies sich die Schätzung basierend auf RGB-D-Kameras. Im dem präsentierten Verfahren wird die Kopfpose über einen *Multi-Template-ICP-Algorithmus* aus dem Tiefenbild berechnet. Eine optimierte Korrespondenzsuche über die intrinsischen Kameraparameter sowie das Umschalten des Fehlermaßes je nach Iterationsfortschritt ermöglicht eine schnelle und gleichzeitig akkurate Konvergenz der Kopfposenschätzung. Der im Auto erforderliche weiträumige Erkennungsbereich wird über die Verwendung mehrerer Gesichtsmasken erreicht. Die Pupillenstellung wird im RGB-Farbbild über ein 3-D-Augenmodell berechnet. Störungen im Tiefenbild, hervorgerufen durch starke Sonneneinstrahlung und wechselnde Lichtverhältnisse, werden anhand einer Regressionsschätzung, basierend auf zur Laufzeit gelernten Gesichtsmerkmalen im RGB-Farbbild, kompensiert.

Die situative Analyse des Fahrverhaltens wird über die *Fahrstilerkennung*, die *Komfortparameterschätzung*, die Ableitung einer *objektspezifischen Aufmerksamkeit*, und die *Manöverprädiktion* umgesetzt. Erstmals wurde eine Fahrstilerkennung situationsspezifisch konzipiert und bezieht die Merkmale der Verkehrssituation, wie etwa die Verkehrszeichen oder die Streckenführung, in die Verhaltensanalyse mit ein. Durch die Umsetzung der vorgeschlagenen Konzepte wird nachgeschalteten FAS eine maschinelle Wissensbeschreibung des Fahrerverhaltens ermöglicht: Komfortparameter stellen Intervallgrenzen und Schwellwerte bereit, die eine einfache Adaption an den Fahrer erlauben. Der Fahrstil beschreibt das

Fahrverhalten in einer generalisierenden Verhaltensoberklasse. Aus dynamischen Verhaltensmodellen wird eine objektspezifische Aufmerksamkeit als auch der nächste Aktionszustand des Fahrers abgeleitet.

Eine Validierung des Konzeptes zur situativen Analyse des Fahrerverhaltens wurde über Experimente im realen Straßenverkehr über den FZI-Versuchsträger CoCar, im FZI-Smart-Fahrsimulator als auch über externe Messdatenbanken durchgeführt. Für die Blickrichtungsschätzung wird der Nachweis der Anforderungsentsprechung entsprechend den im Konzept formulierten Ansprüchen für die Einsatz im Fahrzeug geführt. Die Eignung der Fahrstilerkennung wird über Experimente im FZI-Smart-Fahrsimulator gezeigt, da dieser die Möglichkeit bietet verschiedene Fahrerverhalten unter exakt gleichen Bedingungen zu messen. Am Anwendungsfall der Energieprognose wird demonstriert, wie anhand von fahrerindividuellen Komfortparametern die Prognosegenauigkeit von Assistenzsystemen gesteigert werden kann. Die Prognosefähigkeit des zur Erkennung von Handlungsphasen konzipierten *Markov Logik Netz* wird über eine reale Autobahnfahrt mit mehreren Überholvorgängen gezeigt.

6.2. Diskussion

In der vorgelegten Dissertation wird das Verhalten des Fahrers abhängig vom situativen Verkehrskontext analysiert und über maschinelle Lernverfahren Wissen über den Fahrer und dessen Verhalten abgeleitet. Erstmals werden für die Informationszusammenführung der sich wechselseitig beeinflussenden Komponenten *Fahrer*, *Fahrzeug* und *lokales Fahrzeugumfeld Markov Logik Netze* genutzt, welche in der Lage sind die Unsicherheiten der Perzeption zu modellieren und mit sicherem Faktenwissen zu kombinieren. Als Grundmotiv unterliegt den konzipierten Modellen die Eigenschaft der Lernfähigkeit und der damit verbundenen Möglichkeit, die Ableitung von neuem Wissen und Verhaltensbeschreibungen auf den individuellen Fahrer zu adaptieren.

Ausgangsbasis für die Analyse des situativen Fahrerverhaltens sind robuste und für die Domäne Automobil konzipierte Perzeptionsverfahren. Neuartig an der entwickelten Blickrichtungserkennung sind insbesondere die Optimierungen hinsichtlich eines weiten Erkennungsbereichs, eine Kompensation von wechselnden und starken Fremdlichteinwirkungen und die Berechnung der Augenstellung zur Erkennung von nur durch eine Augenbewegung stattfindender Blickbewegungen. Die Abbildung des Blickverhaltens auf die Entitäten des Straßenverkehrs berücksichtigt unter anderem die Aktualität und die dynamischen Zustandsänderungen anderer Verkehrsteilnehmer.

Die Ansätze im Automobilbetrieb wurde durch umfangreiche Experimente im realen Straßenverkehr, im Fahrsimulator und anhand externer Datenbanken validiert. Die Experimente zielten vor allem darauf ab, das Potential und die Nutzbarkeit der Ansätze zu zeigen. Bewusst wurden dafür zum Teil speziell Szenen gewählt, an denen das Potential der Ansätze gut verdeutlicht werden kann. Vor der Übertragung der vorgestellten Modelle und Konzepte in Fahrzeugprototypen und Serienfahrzeuge, müssen aufbauende Arbeiten die Evaluierungen auf einer breiteren Datenbasis fortführen.

Die eingangs formulierte Forschungsfrage „*Wie kann das Verhalten des Fahrers beobachtet und analysiert werden, so dass Fahrerassistenzsysteme der Zukunft individuell auf die Eigenschaften des Fahrers abgestimmt werden können?*“ kann damit wie folgt beantwortet werden:

Das in dieser Arbeit vorgelegte durchgängige Konzept, welches das Situationsbewusstsein des Fahrers in den drei Ebenen *Wahrnehmen*, *Verstehen* und *Projizieren* analysiert und dabei den situativen Kontext der in Beziehung stehenden Komponenten Fahrer, Fahrzeug und lokales Umfeld berücksichtigt, zeigt, dass eine individuell auf die Eigenschaften des Fahrers abgestimmte Assistenz möglich ist. Grundlage dafür ist die maschinelle Beschreibung von Wissen hinsichtlich des Fahrerverhaltens. Das in Form von dynamischen Verhaltensmodellen, quantitativen Intervallen und Schwellwerten, die Klassifizierung des Verhaltens sowie die Prognose zukünftiger Fahrerhandlungen repräsentierte Wissen ermöglicht einen intuitiven, kooperativen Umgang mit Fahrerassistenzsystemen der nächsten Generation. Es steigert gleichzeitig deren Effizienz und Sicherheit und ist damit Grundlage einer breiten Kundenakzeptanz.

6.3. Ausblick

Am umgesetzten Konzept zur situativen Analyse des Fahrerverhaltens sowie den gewonnenen Erkenntnissen und Ergebnissen können eine Vielzahl aktueller Forschungsfelder profitieren. Zukünftige Generationen *intelligenter Transportsysteme* können ihre Funktionalität personalisieren um neue Möglichkeiten erweitern.

Die fortschreitende Vernetzung des Fahrzeugs mit anderen Fahrzeugen als auch mit der Infrastruktur bietet beispielsweise die Möglichkeit, den Fahrer schon frühzeitig auf kritische und gefährliche Verkehrssituationen hinzuweisen. Eine kritische Verkehrssituation ist wesentlich von der Tatsache geprägt, inwiefern Verkehrszeichen, Fußgänger oder andere Fahrzeuge *gesehen* wurden. Wird die (objektspezifische) Aufmerksamkeit des Fahrers an andere Verkehrsteilnehmer übermittelt, können vernetzte Fahrzeuge ihre *Situationsanalyse* präzisieren und damit effizienter sicherheitskritische Verkehrssituationen prognostizieren.

Die klassische Automobilarchitektur, mit einem verantwortlichen Fahrer hinter dem Lenkrad, einem Beifahrer und den weiteren Insassen auf der Rückbank, verliert durch die rasanten Entwicklungen im *Autonomen Fahren* ihre zwingende Notwendigkeit. Futuristische Konzepte stellen das Fahrzeug der Zukunft als Erlebnisraum dar, in dem gleichgestellte Teilnehmer mit den Kontroll- und Informationssystemen des Fahrzeugs interagieren. Durch das selbstfahrende Auto wird die Fahraufgabe zu einer zu überwachenden Nebenaufgabe, welche nicht zwingend an *eine* Person gebunden sein muss.

Die entwickelten Perceptions- wie Situationsanalysekomponenten können verwendet werden um Interaktionen in der Fahrgastzelle zu modellieren. In der vorliegenden Arbeit ist die Situationsanalyse wie die Perzeption auf *einen* Fahrer fokussiert. Anknüpfende Arbeiten könnten die Konzepte auf mehrere Fahrer, die mit verschiedenen Eigenschaften und Fähigkeiten miteinander und mit dem Fahrzeug interagieren, erweitern.

A. Quellennachweise

A.1. Literatur

- [AB14] Smart Eye AB. *SmartEye | If you want reality*. 2014. URL: <http://www.smarteye.se> (besucht am 31.05.2014).
- [And+12] Michael Riis Andersen, Thomas Jensen, Pavel Lisouski, Anders Krogh Mortensen, Mikkel Kragh Hansen, Torben Gregersen und Peter Ahrendt. *Kinect depth sensor evaluation for computer vision applications*. Techn. Ber. Arhus Universitet, 2012.
- [Asu14] Asus. *Asus Xtion Pro - Specifications*. 2014. URL: <http://www.asus.com/Multimedia/Xtion/specifications/> (besucht am 21.08.2014).
- [Aut14] Autonomes-Fahren.de. *Autonomes Fahren ab 2025 | VW, BMW, Daimler, Continental & Google*. 2014. URL: <http://www.autonomes-fahren.de/autonomes-fahren-ab-2025-vw-bmw-daimler-continental-google/> (besucht am 26.06.2015).
- [Avi14] Aviva Insurance plc. *Car Insurance Discounts for Safer Driver - AVIVIA App*. 2014. URL: <http://www.aviva.co.uk/drive/> (besucht am 22.06.2015).
- [Bak+04] Simon Baker, Iain Matthews, Jing Xiao, Ralph Gross, Takeo Kanade und Takahiro Ishikawa. „Real-Time Non-Rigid Driver Head Tracking for Driver Mental State Estimation“. In: *11th World Congress on Intelligent Transportation Systems* (2004), S. 1–12.
- [Bär+11] Tobias Bär, Dennis Nienhüser, Ralf Kohlhaas und J Marius Zöllner. „Probabilistic driving style determination by means of a situation based analysis of the vehicle data“. In: *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. IEEE, Okt. 2011, S. 1698–1703. ISBN: 978-1-4577-2197-7.
- [BB07] Shan Bao und Linda Ng Boyle. „Visual Search Strategies of Older Drivers at Rural Expressway Intersections“. en. In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 51.24 (Okt. 2007), S. 1560–1564. ISSN: 1071-1813.
- [BB15] Wilhelm Burger und Mark James Burge. *Digitale Bildverarbeitung*. Bd. 1. Springer Berlin Heidelberg, 2015, S. 1689–1699. ISBN: 9788578110796. arXiv: [arXiv:1011.1669v3](https://arxiv.org/abs/1011.1669v3).
- [BD09] Holger Berndt und Klaus Dietmayer. „Driver intention inference with vehicle onboard sensors“. In: *2009 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*. IEEE, 2009, S. 102–107. ISBN: 978-1-4244-5442-6.
- [BD13] Katherina Barby und Barbara Deml. „Detection of Driver Visual Field Narrowing“. In: *IEEE International Conference on Intelligent Transportation Systems*. The Hague, The Netherlands: IEEE, 2013.

- [Beg13] Shahina Begum. „Intelligent Driver Monitoring Systems Based on Physiological Sensor Signals: A Review“. In: *16th International IEEE Annual Conference on Intelligent Transportation Systems*. Okt. 2013.
- [Ben75] Jon Louis Bentley. „Multidimensional binary search trees used for associative searching“. In: *Communications of the ACM* 18.9 (1975), S. 509–517. ISSN: 00010782.
- [Bey94] David Beymer. „Face recognition under varying pose“. In: *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on* (1994). ISSN: 1063-6919.
- [BGR10] Colin Bauer, Dr. Mark Gonter und Prof. Raul Rojas. „Fahrerspezifische Analyse des Fahrverhaltens zur Parametrierung aktiver Sicherheitssysteme“. In: *4. Tagung Sicherheit durch Fahrerassistenz*. München: TÜV-Süd München, 2010, S. 11.
- [Bis95] Christopher Michael Bishop. *Neural Networks for Pattern Recognition*. Bd. 92. CLARENDON-PRESS OXFORD, 1995, S. 482. ISBN: 0198538642.
- [BL95] Gerard Blais und Martin Levine. „Registering multiview range data to create 3D computer objects“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17.8 (1995), S. 820–824. ISSN: 01628828.
- [BM92] Paul J Besl und Neil D McKay. „A method for registration of 3-D shapes“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. TSA '09 14.2 (1992), S. 239–256. ISSN: 01628828.
- [BNZ14] Tobias Bär, Dennis Nienhüser und J.Marius Zöllner. „Driver Individual Energy Consumption Forecast“. In: *3rd International Conference on Energy Efficient Vehicles (ICEEV 2014)*. Dresden, Germany, 2014, S. 6.
- [BPH05] Lawrence Barr, Stephen Popkin und Heidi Howarth. „A review and evaluation of emerging driver fatigue detection measures and technologies“. In: *National Transportation Systems Center, Cambridge. US Department of Transportation, Washington. Di* (2005).
- [Bre01] Leo Breiman. „Random Forests“. In: *Statistics* 45.1 (2001), S. 1–33. ISSN: 08856125.
- [BRZ12a] Tobias Bär, Jan-Felix Reuter und J. Marius Zöllner. „Blickrichtungserkennung des Fahrers mittels ICP auf Farb- und Tiefenbilddaten“. In: *Forum Bildverarbeitung 2012*. Hrsg. von F. Puente Leon und M. Heizmann. 1. Aufl. Karlsruhe: KIT, Scientific Publishing, 2012. Kap. 5, S. 257–265.
- [BRZ12b] Tobias Bär, Jan-Felix Reuter und J. Marius Zöllner. „Blickrichtungserkennung des Fahrers über ICP, basierend auf 2-D und Tiefenbilddaten“. In: *Forum Bildverarbeitung 2012*. Hrsg. von F. Puente Leon und M. Heizmann. 1. Aufl. Karlsruhe: KIT Scientific Publishing, 2012. Kap. 5, S. 356.
- [BRZ12c] Tobias Bär, Jan-Felix Reuter und J. Marius Zöllner. „Driver Head Pose and Gaze Estimation based on Multi-Template ICP 3-D Point Cloud Alignment“. In: *2012 15th International IEEE Conference on Intelligent Transportation Systems ITSC*. Anchorage, AK: IEEE, 2012, S. 1797–1802.

- [BS99] Raouf Benjemaa und Francis Schmitt. „Fast global registration of 3D sampled surfaces using a multi-z-buffer technique“. In: *Image and Vision Computing* 17.2 (1999), S. 113–120. ISSN: 02628856.
- [BZS14] Philipp Bender, Julius Ziegler und Christoph Stiller. „Lanelets: Efficient map representation for autonomous driving“. In: *Intelligent Vehicles Symposium Proceedings, 2014 IEEE*. Juni 2014, S. 420–425.
- [CFE13a] German Castignani, Raphael Frank und Thomas Engel. „An evaluation study of driver profiling fuzzy algorithms using smartphones“. In: *2013 21st IEEE International Conference on Network Protocols (ICNP)*. IEEE, Okt. 2013, S. 1–6. ISBN: 978-1-4799-1270-4.
- [CFE13b] German Castignani, Raphael Frank und Thomas Engel. „Driver behavior profiling using smartphones“. In: *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, Okt. 2013, S. 552–557. ISBN: 978-1-4799-2914-6.
- [CGH00] Ira Cohen, Ashutosh Garg und Thomas S Huang. „Emotion recognition from facial expressions using multilevel HMM“. In: *Science And Technology* (2000), S. 85.
- [Cha+09] Bhavin Chamadiya, Stefan Heuer, Ulrich G Hofmann und Manfred Wagner. „Towards a capacitively coupled electrocardiography system for car seat integration“. In: *4th European Conference of the International Federation for Medical and Biological Engineering. IFMBE Proceedings 22* (2009). Hrsg. von Ratko Magjarevic, Jos Sloten, Pascal Verdonck, Marc Nyssen und Jens Haueisen, S. 1217–1221.
- [Clk] ClkerFreeVectorImages. *Eye Diagram Human Body Parts*. URL: <https://pixabay.com/en/eye-diagram-biology-sight-31145/> (besucht am 06.08.2015).
- [CM91] Yang Chen und Gerard Medioni. „Object modeling by registration of multiple range images“. In: *Proceedings. 1991 IEEE International Conference on Robotics and Automation*. IEEE Comput. Soc. Press, 1991, S. 2724–2729. ISBN: 0-8186-2163-X.
- [CMS85] Myung-Soon Chang, Carroll J Messer und Alberto J Santiago. „TIMING TRAFFIC SIGNAL CHANGE INTERVALS BASED ON DRIVER BEHAVIOR“. In: *Transportation Research Record* 1027 (1985). ISSN: 0361-1981.
- [CMV10] Zoran Constantinescu, Cristian Marinoiu und Monica Vladioiu. „Driving style analysis using data mining techniques“. In: *International Journal Of Computers Communications and Control* 5.5 (2010), S. 654–663.
- [Cos+03] Massimo Cossentino, Piermarco Burrafato, Saverio Lombardo und Luca Sabatucci. „Agent Technologies, Infrastructures, Tools, and Applications for E-Services“. In: *Agent Technologies Infrastructures Tools and Applications for EServices*. Lecture Notes in Computer Science 2592.11 (2003). Hrsg. von Jaime G Carbonell, Jörg Siekmann, Ryszard Kowalczyk, Jörg P Müller, Huaglorry Tianfield und Rainer Unland, S. 107–120. ISSN: 00985589.
- [CPT07] Shinko Yuanhsien Cheng, Sangho Park und Mohan Manubhai Trivedi. „Multi-spectral and multi-perspective video arrays for driver body tracking and activity analysis“. In: *Computer Vision and Image Understanding* 106.2-3 (2007), S. 245–257.

- [CT06] Shinko Yuanhsien Cheng und Mohan Manubhai Trivedi. „Turn-Intent Analysis Using Body Pose for Intelligent Driver Assistance“. In: *IEEE Pervasive Computing* 5.4 (2006). ISSN: 1536-1268.
- [CT10] Shinko Yuanhsien Cheng und Mohan Manubhai Trivedi. „Vision-Based Infotainment User Determination by Hand Recognition for Driver Assistance“. In: *IEEE Transactions on Intelligent Transportation Systems* 11.3 (2010). ISSN: 1524-9050.
- [CT11] Tran Cuong und Mohan Manubhai Trivedi. *Chapter 30 Vision for Driver Assistance: Looking at People in a Vehicle*. 2011. URL: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.228.4584>.
- [Dan+12] Matthias Dantone, Juergen Gall, Gabriele Fanelli und Luc Van Gool. „Real-time Facial Feature Detection using Conditional Regression Forests“. In: *CVPR*. 2012.
- [Dau12] Hal Daumé III. „A Course in Machine Learning“. In: *chapter 5* (2012), S. 69.
- [DBB03] Ismail Dagli, Michael Brost und Gabi Breuel. „Action Recognition And Prediction For Driver Assistance Systems Using Dynamic Belief Networks“. In: *Technology* 2592.2592 (2003), S. 179–194.
- [DD95] Daniel F. DeMenthon und Larry S. Davis. „Model-Based Object Pose in 25 Lines of Code“. In: *International Journal of Computer Vision* 15 (1995), S. 123–141.
- [DR02] Ismail Dagli und Dirk Reichardt. „Motivation-based approach to behavior prediction“. In: *Intelligent Vehicle Symposium, 2002. IEEE*. Bd. 1. IEEE, 2002, S. 227–233. ISBN: 0-7803-7346-4.
- [DSK08] Ankur Datta, Yaser Sheikh und Takeo Kanade. „Linear motion estimation for systems of articulated planes“. In: *2008 IEEE Conference on Computer Vision and Pattern Recognition* (2008). ISSN: 1063-6919.
- [DT09] Anup Doshi und Mohan Manubhai Trivedi. „On the roles of eye gaze and head dynamics in predicting driver’s intent to change lanes“. In: *IEEE Transactions on Intelligent Transportation Systems* 10.3 (2009), S. 453–462.
- [DT10] Anup Doshi und Mohan Manubhai Trivedi. „Attention estimation by simultaneous observation of viewer and view“. In: *Computer Vision and Pattern Recognition Workshops CVPRW 2010 IEEE Computer Society Conference on*. IEEE, 2010, S. 1–7. ISBN: 9781424470297.
- [Eic+13] Andreas von Eichhorn, Moritz Werling, Peter Zahn und Dieter Schramm. „Maneuver prediction at intersections using cost-to-go gradients“. In: *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, Okt. 2013, S. 112–117. ISBN: 978-1-4799-2914-6.
- [End95] Mica R. Endsley. „Toward a Theory of Situation Awareness in Dynamic Systems“. In: *Human Factors: The Journal of the Human Factors and Ergonomics Society* 37.1 (1995), S. 32–64.
- [Fan+11] Gabriele Fanelli, Thibaut Weise, Juergen Gall und Luc Van Gool. „Real Time Head Pose Estimation from Consumer Depth Cameras“. In: *visioneeethz.ch. Lecture Notes in Computer Science 33rd Annua*. 6835 (2011). Hrsg. von Rudolf Mester und Michael Felsberg, S. 101–110.

- [FGG11] Gabriele Fanelli, Juergen Gall und Luc Van Gool. „Real Time Head Pose Estimation with Random Regression Forests“. In: *Training*. Lecture Notes in Computer Science 7098 (2011). Hrsg. von Zhenan Sun, Jianhuang Lai, Xilin Chen und Tieniu Tan, S. 66–73. ISSN: 10636919.
- [Fit03] Andrew W. Fitzgibbon. „Robust registration of 2D and 3D point sets“. In: *Image and Vision Computing*. Bd. 21. 13-14. 2003, S. 1145–1153.
- [Fle+03] Luke Fletcher, Nick Apostoloff, Lars Petersson und Alexander Zelinsky. „Vision in and out of vehicles“. In: *IEEE Intelligent Systems*. Bd. 18. 3. 2003, S. 12–17.
- [Fle+05a] Luke Fletcher, Gareth Loy, Nick Barnes und Alexander Zelinsky. „Correlating driver gaze with the road scene for driver assistance systems“. In: *Robotics and Autonomous Systems* 52.1 (2005), S. 71–84. ISSN: 09218890.
- [Fle+05b] Luke Fletcher, Lars Petersson, Nick Barnes, David Austin und Alexander Zelinsky. „A Sign Reading Driver Assistance System Using Eye Gaze“. In: *Robotics*. Ieee, 2005, S. 4655–4660. ISBN: 078038914X.
- [Fle08] Luke Fletcher. „An Automated Co-driver for Advanced Driver Assistance Systems: The next step in road safety.“ Diss. Canberra, Australia: Research School of Information Sciences und Engineering, Australian National University, 2008.
- [FZ09] Luke Fletcher und Alexander Zelinsky. „Driver Inattention Detection based on Eye Gaze–Road Event Correlation“. In: *The International Journal of Robotics Research* 28.6 (2009), S. 774–801. ISSN: 02783649.
- [GRB94] Guy Godin, Marc Rioux und Rejean Baribeau. „Three-dimensional registration using range and intensity information“. In: *Photonics for ...* 2350.1994 (1994), S. 279–290. ISSN: 0277786X.
- [Gre15] GreenRoad Technology Ltd. *Green Road*. 2015. URL: <http://greenroad.com/> (besucht am 22. 06. 2015).
- [Gro+88] G. E. Grossman, Richard Jhon Leigh, Larry A. Abel, D. J. Lanska und S. E. Thurston. „Frequency and velocity of rotational head perturbations during locomotion“. In: *Experimental Brain Research* 70.3 (1988), S. 470–476.
- [HA13] Bryan Higgs und Montasir Abbas. „A two-step segmentation algorithm for behavioral clustering of naturalistic driving styles“. In: *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, Okt. 2013, S. 857–862. ISBN: 978-1-4799-2914-6.
- [Har03] Volker Hargutt. „Das Lidschlussverhalten als Indikator für Aufmerksamkeits- und Müdigkeitsprozesse bei Arbeitshandlungen“. Phd. Thesis. Julius-Maximilians Universität Würzburg, 2003, S. 255.
- [HG87] Heinz Heckhausen und Peter M. Gollwitzer. „Thought contents and cognitive functioning in motivational versus volitional states of mind“. In: *Motivation and Emotion* 11.2 (1987), S. 101–120.

- [HNL06] Craig Hennessey, Borna Nouredin und Peter Lawrence. „A single camera eye-gaze tracking system with free head motion“. In: *Measurement*. ETRA '06 1.March (2006), S. 27–29.
- [Hof10] Marek Hoffmann. *Kinect splitterfasernackt: Das steckt in Microsofts neuer Bewegungssteuerung*. 2010. URL: <https://www.basichthinking.de/blog/2010/11/05/kinect-splitterfasernackt-das-steckt-in-microsofts-neuer-bewegungssteuerung/> (besucht am 23.06.2015).
- [HP00] Jennifer Healey und Rosalind W Picard. „SmartCar: detecting driver stress“. In: *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000* 4 (2000). ISSN: 1051-4651.
- [HP05] Jennifer Healey und Rosalind W Picard. „Detecting stress during real-world driving tasks using physiological sensors“. In: *IEEE Transactions on Intelligent Transportation Systems* 6.2 (2005). ISSN: 1524-9050.
- [HPS11] Un Hu, Nikos Pitsianis und Xiaobai Sun. „Motion Saliency Map Generations for Video Data Analysis: Spatio-temporal Signatures in the Array Operations“. In: *Fifteenth Annual Workshop of High Performance Extreme Computing Conference*. 2011, S. 3.
- [HSW98] Jeffrey Huang, Xue Shao und Harry Wechsler. „Face pose discrimination using support vector machines (SVM)“. In: *Proceedings. Fourteenth International Conference on Pattern Recognition (Cat. No.98EX170)* 1 (1998).
- [HW08a] Mordechai (Muki) Haklay und Patrick Weber. „OpenStreetMap: User-Generated Street Maps“. In: *IEEE Pervasive Computing* 7.4 (Okt. 2008), S. 12–18. ISSN: 1536-1268.
- [HW08b] Mordechai Haklay und Patrick Weber. „OpenStreet map: User-generated street maps“. In: *IEEE Pervasive Computing* 7.4 (2008), S. 12–18.
- [Inc07] Google Inc. *Google Traffic*. 2007. URL: <http://maps.google.com>.
- [Ish+04] Takahiro Ishikawa, Simon Baker, Iain Matthews und Takeo Kanade. „Passive Driver Gaze Tracking with Active Appearance Models“. In: *Robotics* (2004), S. 1–18.
- [Jar+08] A. Jarvis, H. I. Reuter, A. Nelson und E. Guevara. *Hole-filled seamless SRTM data V4*. Guevara, 2008. URL: <http://srtm.csi.cgiar.org> (besucht am 28.05.2014).
- [Jim11] Pedro Jiménez Molina. *Face pose estimation with automatic 3D model creation for a driver inattention monitoring application*. eng. Alcalá, 2011. URL: <http://dspace.uah.es/dspace/handle/10017/9743>.
- [JT11] Derick A. Johnson und Mohan Manubhai Trivedi. „Driving style recognition using a smartphone as a sensor platform“. In: *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. IEEE, Okt. 2011, S. 1609–1615. ISBN: 978-1-4577-2197-7.
- [JV03] Michael Jeffrey Jones und Paul Viola. „Fast Multi-view Face Detection“. In: *Mitsubishi Electric Research Lab TR2000396* July (2003).
- [KA01] Shunji Katahara und Masayoshi Aoki. „Motion estimation of driver’s head from nostrils detection“. In: *Faculty of Engineering, SEIKEI University* (2001).
- [KE12] Kourosh Khoshelham und Sander Oude Elberink. „Accuracy and resolution of Kinect depth data for indoor mapping applications.“ en. In: *Sensors (Basel, Switzerland)* 12.2 (Jan. 2012), S. 1437–54. ISSN: 1424-8220.

- [Ker14] Jan-Benedikt Kersting. *Lerne zu Handeln*. 2014. URL: <http://www.bildungsadler.de/handlung-und-verhalten/> (besucht am 29.06.2015).
- [Kle14] Jan Kleindieck. „Klassifikation von Verkehrssituationen für das hochautomatisierte Fahren“. Master Thesis. KIT - Karlsruhe Institute of Technology - ITIV, 2014.
- [Koh+11] Ralf Kohlhaas, Thomas Schamm, Dennis Nienhuser und J. Marius Zollner. „Anticipatory energy saving assistant for approaching slower vehicles“. In: *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. IEEE, Okt. 2011, S. 1966–1971. ISBN: 978-1-4577-2197-7.
- [Kol+11] Abhiram Kolli, Alireza Fasih, Fadi Al Machot und Kyandoghene Kyamakya. „Non-intrusive car driver’s emotion recognition using thermal camera“. In: *Proceedings of the Joint INDS’11 & ISTET’11* (2011), S. 1–5.
- [Kum+03] Toru Kumagai, Yasuo Sakaguchi, Masayuki Okuwa und Motoyuki Akamatsu. „Prediction of Driving Behavior through Probabilistic Inference“. In: *Engineering Applications of Neural Networks, Eighth International Conference on September* (2003), S. 8–10.
- [Lan06] Michael F Land. „Eye movements and the control of actions in everyday life“. In: *Progress in Retinal and Eye Research*. Lecture Notes in Computer Science 25.3 (2006). Hrsg. von Jesús López, Emilio Benfenati und Werner Dubitzky, S. 296–324.
- [Lan92] Michael F Land. „Predictable eye-head coordination during driving.“ In: *Nature* 359.6393 (1992), S. 318–320. ISSN: 0028-0836.
- [Le +06] Olivier Le Meur, Patrick Le Callet, Dominique Barba und Dominique Thoreau. *A coherent computational approach to model bottom-up visual attention*. 2006. URL: <http://www.ncbi.nlm.nih.gov/pubmed/16640265>.
- [LFS12] Martin Liebner, Klanner Felix und Christoph Stiller. „Der Fahrer im Mittelpunkt – Eye-Tracking als Schlüssel zum mitdenkenden Fahrzeug?“ In: *8. Workshop Fahrerassistenzsysteme FAS 2012*. Hrsg. von Universität Ulm, Institut für Mess- und Regel und Mikrotechnik. Walting, Germany, 2012, S. 6.
- [Li+04] Yongmin Li, Shaogang Gong, Jamie Sherrah und Heather Liddell. „Support vector machine based multi-view face detection and recognition“. In: *Image and Vision Computing* 22.5 (2004), S. 413–427. ISSN: 02628856.
- [Lie+12] Martin Liebner, Michael Baumann, Felix Klanner und Christoph Stiller. „Driver intent inference at urban intersections using the intelligent driver model“. In: *2012 IEEE Intelligent Vehicles Symposium*. IEEE, Juni 2012, S. 1162–1167. ISBN: 978-1-4673-2118-1.
- [Lie+13a] Martin Liebner, Felix Klanner, Michael Baumann, Christian Ruhhammer und Christoph Stiller. „Velocity-Based Driver Intent Inference at Urban Intersections in the Presence of Preceding Vehicles“. In: *IEEE Intelligent Transportation Systems Magazine* 5.2 (2013), S. 10–21. ISSN: 1939-1390.
- [Lie+13b] Martin Liebner, Christian Ruhhammer, Felix Klanner und Christoph Stiller. „Generic driver intent inference based on parametric models“. In: *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, Okt. 2013, S. 268–275. ISBN: 978-1-4799-2914-6.

- [Lit12] Krystof Litomisky. *Consumer RGB - D Cameras and their Applications*. 2012. URL: <http://alumni.cs.ucr.edu/%7B~%7Dklitomis/files/RGBD-intro.pdf>.
- [LL94] Michael F. Land und David Lee. „Where we look when we steer“. In: *Nature* 369.6483 (1994), S. 742–744. ISSN: 00280836.
- [Low04] Kok-Lim Low. „Linear Least-squares Optimization for Point-to-plane ICP Surface Registration“. In: *Chapel Hill, University of North Carolina* February (2004), S. 2–4.
- [Mar13] Manuel Martin. „Simultaneous Head Pose Detection and Head Model Generation on Consumer Depth Cameras“. Diploma Thesis. KIT - Karlsruher Institut für Technologie, 2013, S. 61.
- [McC+07] Joel C. McCall, David P. Wipf, Mohan Manubhai Trivedi und Bhaskar D. Rao. „Lane Change Intent Analysis Using Robust Operators and Sparse Bayesian Learning“. In: *IEEE Transactions on Intelligent Transportation Systems*. Bd. 3. 3. Ieee, 2007, S. 431–440.
- [Met09] Barbara Metz. „Worauf achtet der Fahrer? Steuerung der Aufmerksamkeit beim Fahren mit visuellen Nebenaufgaben“. Diss. Universität Würzburg, Sep. 2009.
- [Mic85] John A. Michon. „A critical view of driver behaviour models: What do we know, what should we do?“ In: 1985.
- [Mir+13] Peter Mirwaldt, Arne Bartels, Than-Binh To, Hagen Malberg, Sebastian Zaunseder und Karsten Lemmer. „Evaluation von Sensoren zur kontaktlosen Messung der Herzrate im Fahrzeug“. In: *Proceedings der 7. VDI-Tagung Der Fahrer im 21. Jahrhundert*. 2013.
- [Mor+12] Masataka Mori, Chiyomi Miyajima, Pongtep Angkititrakul, Takatsugu Hirayama, Yiyang Li, Norihide Kitaoka und Kazuya Takeda. „Measuring driver awareness based on correlation between gaze behavior and risks of surrounding vehicles“. In: *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*. Anchorage, AK: IEEE, 2012, S. 644–647.
- [MR70] Ronald R. Mourant und Thomas H. Rockwell. „Mapping eye-movement patterns to the visual scene in driving: an exploratory study.“ In: *Human factors* 12.1 (1970), S. 81–87. ISSN: 0018-7208.
- [MT07] Joel C. McCall und Mohan Manubhai Trivedi. „Driver Behavior and Situation Aware Brake Assistance for Intelligent Vehicles“. In: *Proceedings of the IEEE* 95.2 (Feb. 2007), S. 374–387. ISSN: 0018-9219.
- [MT08] Erik Murphy-Chutorian und Mohan Manubhai Trivedi. „HyHOPE: Hybrid Head Orientation and Position Estimation for vision-based driver head tracking“. In: *IEEE Intelligent Vehicles Symposium, Proceedings*. 2008, S. 512–517.
- [MT09] Erik Murphy-Chutorian und Mohan Manubhai Trivedi. „Head Pose Estimation in Computer Vision: A Survey“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Bd. 31. Google Inc., Mountain View, CA 94043, USA. erikmchut@gmail.com. IEEE Computer Society, 2009, S. 607–26.

- [MT10] Erik Murphy-Chutorian und Mohan Manubhai Trivedi. „Head pose estimation and augmented reality tracking: An integrated system and evaluation for monitoring driver awareness“. In: *IEEE Transactions on Intelligent Transportation Systems* 11.2 (2010), S. 300–311.
- [Mur07] Erik Murphy-Chutorian. „Head pose estimation for driver assistance systems: A robust algorithm and experimental evaluation“. In: *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE. 2007*. ISBN: 1424413966.
- [MvS14] Martin Manuel, Florian van de Camp und Rainer Stiefelhagen. „Real Time Head Model Creation and Head Pose Estimation on Consumer Depth Cameras“. In: *2014 2nd International Conference on 3D Vision*. Bd. 1. Dez. 2014, S. 641–648.
- [NF96] S. Niyogi und W.T. Freeman. „Example-based head tracking“. In: *Proceedings of the Second International Conference on Automatic Face and Gesture Recognition* (1996).
- [NG02] Jeffrey Ng und Shaogang Gong. „Composite support vector machines for detection of faces across views and pose estimation“. In: *Image and Vision Computing*. Bd. 20. 5-6. 2002, S. 359–368.
- [NG99] Jeffrey Ng und Shaogang Gong. „Multi-View Face Detection and Pose Estimation Using a Composite Support Vector Machine across the View Sphere“. In: *RATFG-RTS* (1999).
- [Nie+12] Dennis Nienhüser, Tobias Bär, Ralf Kohlhaas, Thomas Schamm, Jochen Zimmermann, Thomas Gump, Marcus Strand, Oliver Bringmann und J Marius Zöllner. „Energy Efficient Driving and Operation Strategies Based on Situation Awareness and Reasoning“. In: *Information Technology* 54.1 (2012), S. 5–16. ISSN: 16112776.
- [Nie14] Dennis Nienhüser. „Kontextsensitive Erkennung und Interpretation fahrrelevanter statischer Verkehrselemente“. Diss. KIT - Karlsruher Institut für Technologie, 2014, S. 200.
- [Ohn+14] Eshed Ohn-Bar, Sujitha Martin, Ashish Tawari und Mohan Manubhai Trivedi. „Head, Eye, and Hand Patterns for Driver Activity Recognition“. In: *International Conference on Pattern Recognition (ICPR2014)*. Hrsg. von IEEE. Stockholm: IEEE Computer Society, 2014.
- [OMY02] Takehiko Ohno, Naoki Mukawa und Atsushi Yoshikawa. „FreeGaze : A Gaze Tracking System for Everyday Gaze Interaction“. In: *Symposium on Eye Tracking Research & Applications (ETRA'02)*. 2002, S. 125–132. ISBN: 1581134673.
- [OP00a] Nuria Oliver und Alex P Pentland. „Driver behavior recognition and prediction in a Smart-Car“. In: *Proceedings of SPIE* 4023 (2000). Hrsg. von Jacques G Verly, S. 280–290. ISSN: 0277786X.
- [OP00b] Nuria Oliver und a.P. Pentland. „Graphical models for driver behavior recognition in a smartcar“. In: *IEEE Intelligent Vehicles Symposium*. Bd. IV. Mi. IEEE. Ieee, 2000, S. 7–12. ISBN: 0780363639.
- [PAZ03] Lars Petersson, Nicholas Apostoloff und Alexander Zelinsky. „Driver Assistance: An Integration of Vehicle Monitoring and Control“. In: *Proceedings of the 2003 IEEE International Conference on Robotics and Automation* (2003), pages.

- [PB98] R. Pappu und P. A. Beardsley. „A qualitative approach to classifying gaze direction“. In: *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition* (1998).
- [PBZ10] Martin Pfeffer, Tobias Bär und J. Marius Zöllner. „Analyse der Aufmerksamkeit des Fahrers mittels Blickrichtung im Fahrzeuginnenraum“. Diploma Thesis. KIT - Karlsruher Institut für Technologie, 2010, S. 103.
- [Pér+03] A Pérez, M L Córdoba, A García, R Méndez, M L Muñoz, J L Pedraza und F Sánchez. „A Precise Eye-Gaze Detection and Tracking System“. In: *Journal of WSCG* (2003), S. 105–108.
- [Pet+04] Lars Petersson, Luke Fletcher, Nick Barnes und Alexander Zelinsky. „An interactive driver assistance system monitoring the scene in and out of the vehicle“. In: *Robotics and Automation 2004 Proceedings ICRA 04 2004 IEEE International Conference on*. Bd. 4. 2004, 3475–3481 Vol.4.
- [Pet+06] Lars Petersson, Luke Fletcher, Alexander Zelinsky, Nick Barnes und Fredrik Arnell. „Towards Safer Roads by Integration of Road Scene Monitoring and Vehicle Control“. In: *The International Journal of Robotics Research* 25.1 (2006), S. 53–72. ISSN: 02783649.
- [PFZ05] Lars Petersson, Luke Fletcher und Alexander Zelinsky. „A framework for driver-in-the-loop driver assistance systems“. In: *Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE*. IEEE Computer Society, 2005, S. 771–776. ISBN: 0780392159.
- [PG07] W Pedrycz und F Gomide. „Fuzzy Systems Engineering (Toward Human-Centric Computing)“. In: (2007).
- [PMP10] Ming-Zher Poh, Daniel J. McDuff und Rosalind W. Picard. „Non-contact, automated cardiac pulse measurements using video imaging and blind source separation.“ In: *Optics express* 18.10 (2010), S. 10762–10774. ISSN: 1094-4087.
- [PR00] Maja Pantic und LJM Rothkrantz. „Automatic analysis of facial expressions: The state of the art“. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.12 (2000), S. 1424–1445. ISSN: 0162-8828. arXiv: [1203.6722](https://arxiv.org/abs/1203.6722).
- [Pul99] Kari Pulli. „Multiview registration for large data sets“. English. In: *Second International Conference on 3-D Digital Imaging and Modeling (Cat. No.PR00062)*. IEEE Comput. Soc, 1999, S. 160–168. ISBN: 0-7695-0062-5.
- [Qui86] J. R. Quinlan. „Induction of decision trees“. In: *Machine Learning* 1.1 (1986), S. 81–106.
- [Qui93] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Hrsg. von Morgan Kaufmann. Bd. 1. Morgan Kaufmann series in {M}achine {L}earning 3. Morgan Kaufmann, 1993, S. 302.
- [Quo+07] Zhibo Quo, Huajun Liu, Qiong Wang und Jingyu Yang. „A fast algorithm face detection and head pose estimation for driver assistant system“. In: *International Conference on Signal Processing Proceedings, ICSP*. Bd. 3. 2007.
- [Rab89] Lawrence R. Rabiner. „A tutorial on hidden Markov models and selected applications in speech recognition“. In: *Proceedings of the IEEE* 77.2 (1989), S. 257–286. ISSN: 00189219.
- [RB12] Avi Rosenfeld und Zevi Bareket. „Learning Driver’s Behavior to Improve Adaptive Cruise Control“. In: *IAAI*. Toronto, Canada, Juli 2012, S. 8.

- [RBH07] Constantin A. Rothkopf, Dana H. Ballard und Mary M. Hayhoe. „Task and context determine where you look.“ In: *Journal of vision* 7.14 (2007), S. 16.1–20. ISSN: 15347362.
- [RBK98] H. A. Rowley, S. Baluja und T. Kanade. „Rotation invariant neural network-based face detection“. In: *Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No.98CB36231)* (1998). ISSN: 1063-6919.
- [RBZ12] Jan-Felix Reuter, Tobias Bär und J. Marius Zöllner. „Bestimmung der Blickrichtung eines Autofahrers mittels Video- und 3D-Tiefendaten“. Diploma Thesis. KIT - Karlsruher Institut für Technologie, 2012, S. 69.
- [RD06] Matthew Richardson und Pedro Domingos. „Markov logic networks“. In: *Machine Learning* 62.1-2 (Jan. 2006), S. 107–136. ISSN: 0885-6125.
- [RL01] Szymon Rusinkiewicz und Marc Levoy. „Efficient variants of the ICP algorithm“. In: *3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference on. IEEE* (2001), S. 145–152. ISSN: 08876185.
- [RN03] Stuart Jonathan Russell und Peter Norvig. *Künstliche Intelligenz - Ein moderner Ansatz (2. Auflage)*. Pearson Studium, 2003.
- [RN04] Stuart Russell und Peter Norvig. „Künstliche Intelligenz“. In: *Ein moderner Ansatz, München* (2004).
- [RPR14] Ali Rakhshan, Hossein Pishro-Nik und Evan Ray. *Real-time estimation of the distribution of brake response times for an individual driver using Vehicular Ad Hoc Network*. 2014.
- [Sal06] Dario D. Salvucci. „Modeling driver behavior in a cognitive architecture.“ In: *Human Factors* 48.2 (2006), S. 362–380.
- [SC99] Daniel J. Simons und Christopher F. Chabris. „Gorillas in our midst: Sustained inattention blindness for dynamic events“. In: *Perception* 28.9 (1999), S. 1059–1074.
- [Sch09] Jörg Henning Schneider. „Modellierung und Erkennung von Fahrsituationen und Fahrmanövern für sicherheitsrelevante Fahrerassistenzsysteme“. In: (Nov. 2009).
- [Sch14] Thomas Schamm. „Modellbasierter Ansatz zur probabilistischen Interpretation von Fahrsituationen“. Diss. Karlsruher Institut für Technologie (KIT), 2014, S. 181.
- [SD10] Marc Sumner und Pedro Domingos. *The Alchemy Tutorial*. Techn. Ber. Washington: University of Washington, 2010, S. 18.
- [SFS13] Malin Sundbom, Paolo Falcone und Jonas Sjöberg. „Online driver behavior classification using probabilistic ARX models“. In: *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, Okt. 2013, S. 1107–1112. ISBN: 978-1-4799-2914-6.
- [SGO01] J. Sherrah, S. Gong und E. J. Ong. „Face distributions in similarity space under varying head pose“. In: *Image and Vision Computing* 19.12 (2001), S. 807–819.
- [Shi08] David Shinar. „Looks are (almost) everything: where drivers look to get information.“ In: *Human factors* 50.3 (2008), S. 380–384. ISSN: 0018-7208.

- [SSd00] Paul Smith, Mubarak Shah und Niels da Vitoria Lobo. „Monitoring head/eye motion for driver alertness with one camera“. In: *Pattern Recognition, International Conference on*. Bd. 4. IEEE Computer Society. 2000, S. 4636.
- [SSM12] Arun Sahayadhas, Kenneth Sundaraj und Murugappan Murugappan. „Detecting driver drowsiness based on sensors: a review.“ In: *Sensors (Basel, Switzerland)* 12.12 (Jan. 2012), S. 16937–53. ISSN: 1424-8220.
- [Sta15] State Farm Mutual Automobile Insurance Company. *Drive Smarter. State Farm Driver Feedback Scores Your Driving and Gives You Tips to Help You Drive Even Better*. 2015. URL: <https://www.statefarm.com/customer-care/download-mobile-apps/driver-feedback>.
- [SYW96] Rainer Stiefelhagen, Jie Yang und Alex Waibel. „A model-based gaze tracking system“. In: *International Journal on Artificial Intelligence Tools* 6.02 (1996), S. 193–209.
- [THH00] Martin Treiber, Ansgar Hennecke und Dirk Helbing. „Congested traffic states in empirical observations and microscopic simulations“. In: *Physical Review E* 62.2 (Aug. 2000), S. 1805–1824. ISSN: 1063-651X.
- [Tra+10] Toralf Trautmann, Burkhard Müller, Tino Staffetius, Jürgen Bönninger und Jörg van Calker. „Fahrerindividuelle Erkennung von fahrdynamischen Grenzwerten“. In: *TÜV-Süd Tagung - Sicherheit durch Fahrerassistenz*. Hrsg. von TÜV-Süd Tagung. Bd. 4. München: TÜV-Süd Tagung, 2010, S. 9.
- [Tri+04] Mohan Manubhai Trivedi, Shinko Yuanhsien Cheng, Edwin Malcolm Clayton Childers und Stephen Justin Krotosky. „Occupant posture analysis with stereo and thermal infrared video: algorithms and experimental evaluation“. In: *IEEE Transactions on Vehicular Technology* 53.6 (2004). ISSN: 0018-9545.
- [TT09a] Cuong Tran und Mohan Manubhai Trivedi. „Driver assistance for “Keeping Hands on the Wheel and Eyes on the Road”“. In: *2009 IEEE International Conference on Vehicular Electronics and Safety (ICVES)* (2009), S. 97–101.
- [TT09b] Cuong Tran und Mohan Manubhai Trivedi. „Introducing XMOB: Extremity Movement Observation Framework for Upper Body Pose Tracking in 3D“. In: *2009 11th IEEE International Symposium on Multimedia* (2009).
- [TT10] Ashish Tawari und Mohan Manubhai Trivedi. „Speech based emotion classification framework for driver assistance system“. In: *Intelligent Vehicles Symposium (IV), 2010 IEEE* (2010). ISSN: 1931-0587.
- [TT11] Cuong Tran und Mohan Manubhai Trivedi. „Vision for Driver Assistance: Looking at people in a vehicle“. In: (2011). Hrsg. von Thomas B. Moeslund, Adrian Hilton, Volker Krüger und Leonid Sigal, S. 597–614.
- [Web+11] Andreas Weber, Dennis Nienhüser, J Marius Zöllner und Rüdiger Dillmann. „Semiautomatische OpenStreetMap-Kartierung mit einem Segway Personentransporter“. Bachelorarbeit. Karlsruher Institut für Technologie, 2011.
- [Wik14a] Wikipedia. *Kopfimpulstest* — *Wikipedia, Die freie Enzyklopädie*. 2014. URL: <http://de.wikipedia.org/w/index.php?title=Kopfimpulstest>.

- [Wik14b] Wikipedia. *Peripheres Sehen* — Wikipedia, Die freie Enzyklopädie. 2014. URL: http://de.wikipedia.org/wiki/Peripheres%7B%5C_%7DSehen (besucht am 11.03.2014).
- [Wik14c] Wikipedia. *Persönlichkeit* — Wikipedia, Die freie Enzyklopädie. 2014. URL: <http://de.wikipedia.org/w/index.php?title=Pers%EF%BF%BDlichkeit>.
- [WT10] Junwen Wu und Mohan Manubhai Trivedi. „An eye localization, tracking and blink pattern recognition system“. In: *ACM Transactions on Multimedia Computing Communications and Applications* 6.2 (2010), S. 1–23. ISSN: 15516857.
- [Ygu08] Manuel Yguel. „Rigid Transformation Estimation in Point Registration Procedures: Study of the Use of Mahalanobis Distances“. In: (2008).
- [Zad65] Lotfi A. Zadeh. „Fuzzy Sets“. In: *Information and Control* 8 (1965), S. 338–353. ISSN: 00199958.
- [ZB07] Xiao Zhang und Ghulam H. Bham. „Estimation of driver reaction time from detailed vehicle trajectory data“. In: (Mai 2007), S. 574–579.
- [Zen+09] Zhihong Zeng, Maja Pantic, Glenn I Roisman und Thomas S Huang. „A survey of affect recognition methods: audio, visual, and spontaneous expressions.“ In: *IEEE transactions on pattern analysis and machine intelligence* 31.1 (2009), S. 39–58.
- [Zha+07] Zhenqiu Zhang, Yuxiao Hu, Ming Liu und Thomas Huang. „Head Pose Estimation in Seminar Room Using Multi View Face Detectors“. In: *Proceedings of the 1st International Evaluation Conference on Classification of Events, Activities and Relationships*. CLEAR'06. Berlin, Heidelberg: Springer-Verlag, 2007, S. 299–304. ISBN: 978-3-540-69567-7.
- [Zha+12a] C. H. Zhao, B. L. Zhang, J. He und J. Lian. „Recognition of driving postures by contourlet transform and random forests“. In: *IET Intelligent Transport Systems* 6.2 (2012), S. 161–168. ISSN: 1751-956X.
- [Zha+12b] Chihang Zhao, Yongsheng Gao, Jie He und Jie Lian. „Recognition of driving postures by multiwavelet transform and multilayer perceptron classifier“. In: *Engineering Applications of Artificial Intelligence* 25.8 (2012), S. 1677–1686.
- [Zha92] Zhengyou Zhang. *Iterative point matching for registration of free-form curves*. Research Report RR-1658. 1992, S. 42.
- [Zha94] Zhengyou Zhang. „Iterative point matching for registration of free-form curves and surfaces“. In: *International Journal of Computer Vision* 13.2 (1994), S. 119–152.
- [ZOZ04] Yilu Zhang Yilu Zhang, Y. Owechko und Jing Zhang Jing Zhang. „Driver cognitive workload estimation: a data-driven perspective“. In: *Proceedings. The 7th International IEEE Conference on Intelligent Transportation Systems (IEEE Cat. No.04TH8749)* (2004).