

Berichtsblatt

1. ISBN oder ISSN geplant	2. Berichtsart Schlußbericht
3a. Titel des Berichts Kontextabhängige Bedienassistentz und neue Interaktionstechnologien im Fahrzeug, Schlussbericht zum MTI-Leitprojekt-Vorhaben EMBASSI	
3b. Titel der Publikation Kontextabhängige Bedienassistentz und neue Interaktionstechnologien im Fahrzeug	
4a. Autoren des Berichts (Name, Vorname(n)) Baumann, Axel; Fänger, Jens; Mecke, Sven; Kußmann, Holger; Kynast Andreas	5. Abschlußdatum des Vorhabens 30.06.2003
	6. Veröffentlichungsdatum
4b. Autoren der Publikation (Name, Vorname(n)) Baumann, Axel; Fänger, Jens; Mecke, Sven; Kußmann, Holger; Kynast Andreas	7. Form der Publikation
	9. Ber.Nr. Durchführende Insitution
8. Durchführende Institution(en) (Name, Adresse) Robert Bosch GmbH Abteilung FV/SLM Postfach 77 77 77 D-31132 Hildesheim	10. Förderkennzeichen *) 01 IL 904 N 4
	11a. Seitenzahl Bericht 59
	11b. Seitenzahl Publikation
	12. Literaturangaben 35
13. Fördernde Institution (Name, Adresse) Bundesministerium für Bildung, Wissenschaft, Forschung und Technologie (BMBF) 53170 Bonn	14. Tabellen 4
	15. Abbildungen 55
	16. Zusätzliche Angaben
17. Vorgelegt bei (Titel, Ort, Datum)	
18. Kurzfassung Zukünftige Fahrerinformations- und Unterhaltungssysteme im Fahrzeug werden sich nur unwesentlich in ihrer Funktionalität unterscheiden. Ihre Bedienbarkeit und ihre optimale Integration in das Fahrzeug-gesamtkonzept unter Berücksichtigung von Usability-Aspekten wird ein wesentliches, wenn nicht gar das entscheidende Kaufkriterium darstellen. Im Rahmen dieses Projektes wurden daher einerseits notwendige Basistechnologien entwickelt und andererseits neue Bedienkonzepte entworfen, die eine Nutzung der Infotainment-Systeme im Fahrzeug auch während der Fahrt erlauben. Um die Interaktion zwischen Fahrer und Fahrzeug zu verbessern, wurden neue Interaktionskanäle wie die Blickrichtung untersucht. Bestehende Technologien und deren Eignung für die Fahrzeugumgebung wurden evaluiert und neue Lösungen entwickelt. Für typische Applikationen wie die Navigation oder die Entertainment-Einheit (MP3) wurden Bedienkonzepte entwickelt und umgesetzt, die sich neben der Verwendung von selbstlernenden Benutzer-Profilen und Metadaten wie MPEG 7 auch durch kontextabhängige (Fahrsituation, Fahrerbelastung) Bedienunterstützungsfunktionen auszeichnen. Diese Konzepte wurden dann mit Hilfe von Experten- und Nutzertests evaluiert. Darüber hinaus wurde eine szenarienübergreifende (Home, Kfz, Terminal) Architektur entwickelt, die auf einer definierten EMBASSI-Ontologie basiert und typische SW-Agentenkommunikationsmechanismen nutzt. Die Ergebnisse wurden in Form mehrerer Demonstratoren integriert und sehr erfolgreich sowohl intern als auch auf internationalen Ausstellungen wie z.B. der IFA2001 oder der Internationalen Statustagung MTI 2003 dargestellt.	
19. Schlagwörter Fahrsituations- und Fahrerzustandsabhängige Interaktion, HMI, Bedienassistentz im Kfz, XML, KQML, Software-Agenten, Multiagenten-Systeme, Ontologie, SW-Architektur, Navigation, MP3, MPEG7, adaptive Nutzerprofile, Blickrichtungserkennung, Head-Tracking, Eye-Tracking	
20. Verlag	21. Preis

*) Auf das Förderkennzeichen des BMBF soll auch in der Veröffentlichung hingewiesen werden.

Document Control Sheet

1. ISBN or ISSN planned	2. Type of Report Final report
3a. Report Title Kontextabhängige Bedienassistentz und neue Interaktionstechnologien im Fahrzeug, Schlussbericht zum MTI-Leitprojekt-Vorhaben EMBASSI	
3b. Title of Publication Kontextabhängige Bedienassistentz und neue Interaktionstechnologien im Fahrzeug,	
4a. Author(s) of the Report (Family Name, First Name(s)) Baumann, Axel; Fänger, Jens; Mecke, Sven; Kußmann, Holger; Kynast Andreas	5. End of Project 30.06.2003
4b. Author(s) of the Publication (Family Name, First Name(s)) Baumann, Axel; Fänger, Jens; Mecke, Sven; Kußmann, Holger; Kynast Andreas	6. Publication Date
8. Performing Organization(s) (Name, Address) Robert Bosch GmbH Abteilung FV/SLM Postfach 77 77 77 D-31132 Hildesheim	7. Form of Publication Brochure
13. Sponsoring Agency (Name, Address) Bundesministerium für Bildung und Forschung (BMBF) 53170 Bonn	9. Originator's Report No.
	10. Reference No. 01 IL 904 N 4
	11a. No. of Pages Report 59
	11b. No. of Pages Publication
	12. No. of References 35
	14. No. of Tables 4
	15. No. of Figures 55
16. Supplementary Notes	
17. Presented at (Title, Place, Date)	
18. Abstract Future driver infotainment systems will be distinguished not only by their functionalities and features. One of the main criterion of buying a system will be the usability and the optimum integration in the whole car interaction concept. Therefore, within this project on the one hand the basic technologies and on the other hand new user control concepts were developed, which allows the use of infotainment applications during the driving task considering aspects as distraction and safety. In order to increase the usability and improve the interaction between driver and system, new interaction channels as for example the driver gaze were investigated. Therefore existing technologies and their suitability for an automotive use were considered and new solutions developed. User interface concepts for some typical applications like navigation and entertainment (mp3) were developed and implemented taking into account technologies like the use of meta-data (MPEG7), sw-agents, multi-agent-systems and adaptive user profiles as well as context-dependent (driving situation and driver state) user assistance. These concepts were implemented and evaluated using expert- and user tests. In addition to that a sw-architecture was developed covering the requirements of home entertainment, automotive infotainment and mobile terminal systems. This EMBASSI-architecture is based on a special ontologie, XML and typical sw-agent communication mechanisms. The results were integrated in the form of several demonstrators. They have been successfully demonstrated internally, as well as on international shows and fairs, e.g. at the IFA 2001 or the Statustagung MTI 2003.	
19. Keywords driving situation and driver state recognition, driver assistance, HMI, in-car user interaction, XML, KQML, software-agents, multi-agent-system, ontologie, sw-architectur, navigation, MP3, MPEG7, adaptive user profile, gaze detection, head-tracking, eye-tracking	
20. Publisher	21. Price

Teilvorhaben: 01 IL 904 N 4
Zuwendungsempfänger: Robert Bosch GmbH

Schlussbericht zum MTI-Leitprojekt-Vorhaben EMBASSI
Berichtszeitraum: 01.07.99 - 30.06.2003
Ausgangsplan vom: 19.05.1999

Oktober, 2003

Autoren: Baumann, A; Fänger, J.; Mecke, S; Kußmann H.; Kynast, A.

Projektleiter: Dipl.-Ing. Holger Kußmann
Robert Bosch GmbH
Abt.: FV/SLM Tel.: +49 5121 49 5393
Postfach 77 77 77 Fax.: +49 5121 49175393
31132 Hildesheim Email: holger.kussmann@de.bosch.com

Inhaltsverzeichnis:

1	Ziele und Aufgabenstellung des Vorhabens.....	3
1.1	Gesamtziele.....	3
1.2	Wissenschaftliche und/oder technische Aufgabenstellungen des Vorhabens	3
2	Voraussetzungen, unter denen das Vorhaben durchgeführt wurde	4
3	Planung und Ablauf des Vorhabens	4
4	Wissenschaftlicher und technischer Stand	5
5	Zusammenarbeit mit anderen Stellen	5
6	Darstellung der Ergebnisse	6
6.1	Architektur	6
6.1.1	Abbildung der Generischen Embassi Architektur auf das KFZ- Szenario	6
6.1.2	Kommunikation.....	8
6.1.3	GUI.....	9
6.2	Applikationsspezifische adaptive Bedienassistentz.....	16
6.2.1	Navigation.....	16
6.2.2	Entertainment	20
6.2.3	Fahrer-Warn- und Assistenzsystem.....	24
6.3	Fahrsituationserkennung und Ableitung der Fahrerbelastung.....	25
6.4	Demonstrator.....	28
6.4.1	Demonstrationsträger (Sitzkisten).....	28
6.4.2	Integration der externen Beiträge zum Kfz-Demonstrator	31
6.5	Durchführung und Betreuung von Evaluationen.....	37
6.5.1	Evaluation des Navigations-Systems	37
6.5.2	Expertenevaluierung der Bedienoberfläche	39
6.5.3	Evaluation der Störungswarnkomponente.....	40
6.6	Technologien.....	42
6.6.1	SW-Agenten	42
6.6.2	Blickrichtungserkennung	44
7	Veröffentlichungen / Vorträge	57
8.	Literatur.....	58

1 Ziele und Aufgabenstellung des Vorhabens

1.1 Gesamtziele

Die Robert Bosch GmbH beteiligt sich nach gemeinsamer Spezifikation der Teil- und Gesamtarchitekturen an der Entwicklung und Anpassung von Multiagentensystemen zur intelligenten, adaptiven Nutzerassistenz und der Entwicklung und Optimierung von neuartigen videobasierten MMI Technologien.

Ein Teilziel der Aktivitäten ist die Nutzbarmachung von menschengerechten, intuitiven Benutzerschnittstellen für mobile Anwendungen im Kraftfahrzeug mit erweiterter Funktionalität im Bereich der Nutzerassistenz. Bei der konzeptionellen Erarbeitung der Nutzerassistenz werden im Entwicklungsprozess konsequent Ergonomie- und Human-Factor-Aspekte berücksichtigt. Die entwickelten Konzepte zur Bedienassistenz sollen in einem Demonstrationsträger realisiert und der Öffentlichkeit präsentiert werden. Darüber hinaus werden die Ergebnisse mit Hilfe von Nutzertests evaluiert.

Weiterhin soll die Verwendung neuartiger MMI Technologien wie Blickrichtungsverfolgung zur Nutzerassistenz bzw. Unterstützung bei der Bedienung von Geräten untersucht werden. Ziel der Aktivitäten für dieses Arbeitsfeld ist die Erweiterung und Optimierung der Verfahren, um sie für die Bedienung von Geräten im Alltag nutzbar zu machen. Insbesondere der Einsatz innerhalb eines KFZ mit den daraus resultierenden Anforderungen ist hier von Interesse. Mit Hilfe eines solchen Verfahrens zur Blickrichtungsverfolgung soll die Interaktion des Nutzers mit den Funktionseinheiten des zu bedienenden Systems verbessert werden. Diese Verfahren können einerseits zu einer intuitiveren Interaktion des Benutzers mit dem Gerät beitragen und andererseits auch Aufschlüsse über den aktuellen Zustand des Benutzers an die funktionsunterstützenden Einheiten liefern.

1.2 Wissenschaftliche und/oder technische Aufgabenstellungen des Vorhabens

Zur Erreichung der Gesamtziele können somit sechs Arbeitsbereiche identifiziert werden:

1. Gesamtarchitektur, um die unterschiedlichen Nutzungsbereiche (Home, Kfz, Terminal) zusammenzuführen.
2. Menschgerechte, adaptive Nutzerassistenz für mobile Anwendungen
3. Fahrsituationserkennung und Ableitung der Fahrerbelastung zur Realisierung der Adaptivität
4. Realisierung in einem Demonstrator
5. Evaluierung der Nutzerassistenz
6. Evaluation neuer MMI Technologien wie Blickrichtungserkennung und -Verfolgung

Zu 1.: Für die Systemarchitektur sind zunächst Anwendungsszenarien im Fahrzeug zu entwickeln und zu analysieren. Diese sind mit den Anwendungsszenarien in den anderen Nutzungsbereichen abzustimmen und gegebenenfalls zu koppeln. Ziel ist es für gemeinsame Anforderungen innerhalb des Projektes gleiche Strategien und Module zu entwickeln und einzusetzen. Für die unterschiedlichen Anforderungen muss im Projekt eine Vorgehensmethodik im Sinne des Nutzers entwickelt werden, damit es - auch bei unterschiedlicher Auslegung der Bedienassistenz in den verschiedenen Szenarien - nicht zu Irritationen beim Nutzer kommt. Im Weiteren werden dazu geeignete Architekturen (Komponentenstrukturen und Schnittstellen sowie Protokolle) analysiert und ausgewählt. Existierende und geplante Standards sowie die Modularität und Erweiterbarkeit der zu erarbeitenden Lösungen sollen dabei im Vordergrund stehen.

Zu 2.: Für eine menschengerechte, adaptive Nutzerassistenz bieten sich Anwendungen im Kfz mit hohem Interaktionsbedarf durch den Benutzer an. Hierbei sollen Applikationen wie z.B. die Navigation oder Infotainment-Komponenten betrachtet werden. Nach Analyse der Applikation müssen unterschiedliche Stufen der Assistenz im Projekt definiert und eine Auswahl und Kombination dieser Stufen erfolgen. Ebenso sollen systematische Ansätze zur Berücksichtigung von Nutzerverhalten erarbeitet sowie Änderungen des Nutzerverhaltens durch lernende Systemkomponenten erfaßt werden. Dazu können z.B. Nutzerpräferenzen im System berücksichtigt werden. Für die Realisierung der Bedienassistenz werden unterschiedliche Technologien wie die SW-Agententechnologie oder Methoden der KI untersucht und gegebenenfalls ausgewählt.

Zu 3.: Die Adaptivität kann prinzipiell durch kontextabhängiges Systemverhalten erreicht werden. Dabei spielen die Nutzerpräferenzen, die Fahrsituation im Fahrzeug und die Fahrerbelastung eine wichtige Rolle. Für die Applikationen sollen entsprechend Strategien für das kontextabhängige Verhalten unter Berücksichtigung der

Assistenzstufen entwickelt werden. Ebenso werden Ansätze zur Fahrsituationserkennung erarbeitet und erste Ableitungen für die Fahrerbelastung getroffen.

Zu 4.: Nachdem die Anwendungsszenarien erstellt und analysiert wurden und alle Anforderungen aus den weiteren Aktivitäten (siehe Punkt 1.-3.) vorliegen, erfolgt die Umsetzung der Architektur, der Funktionalität der Demonstrationsapplikationen, der Bedienassistentz und der Kontextmodule (Nutzerpräferenzen, Fahrsituation und Fahrerzustand). Bei der Realisierung sind vor allem auch spezielle Anforderungen für die durchzuführenden Nutzertests zu berücksichtigen und umzusetzen. Parallel dazu wird ein Cockpit eines Kfz (Mock up Aufbau) als Demonstrationsträger nachgebildet und an eine virtuelle Fahrsimulation angebunden. Die realisierten Software-Komponenten werden in dem Aufbau zu einem Gesamtsystem integriert.

Zu 5.: Mit dem unter 4. genannten Aufbau und den realisierten Software-Komponenten werden die Usability-Tests durchgeführt. Hier steht die Effektivität, die Effizienz und die Akzeptanz der Assistenzsysteme im Vordergrund. Zusätzlich werden Usability-Aspekte für die reine Bedienoberfläche untersucht.

Zu 6.: Die Detektion und Verfolgung der Blickrichtung eines Benutzers soll in zwei Stufen erfolgen. Zunächst soll unter Verwendung einer einfachen Kamera zur Bildaufnahme die grobe Blickrichtung ermittelt werden. Hierzu sind als Vorstufe die Kopfposition zu bestimmen und Bewegungen und Drehungen des Kopfes zu verfolgen. Basierend auf der Lokalisierung und Verfolgung des Kopfes soll eine Augendetektion mit Blickrichtungsanalyse erfolgen. Ziel ist es, eine Aussage über die grobe Blickrichtung des Benutzers zu erhalten. Hieraus läßt sich ableiten, wie die Aufmerksamkeit gerichtet ist. Innerhalb des KFZ-Szenarios sind Aussagen interessant wie: Der Fahrer schaut auf die Straße, in den Rückspiegel oder zum Beifahrer etc. In dieser ersten Entwicklungsstufe sollen die Möglichkeiten und Grenzen der Bildanalyse mit einer einfachen Kamera aufgezeigt werden.

In der zweiten Stufe sollen die Möglichkeiten zur Feinanalyse der Blickrichtung aufgezeigt werden. Mit der Feinanalyse soll die exakte Position des Blickes z.B. auf einem Monitor oder auf dem Armaturenbrett eines Fahrzeugs ermittelt werden.

Bei diesem Arbeitspaket ist ebenfalls die Einbeziehung der kraftfahrzeugspezifischen Randbedingungen von Bedeutung. So muss z.B. die Zuverlässigkeit der Systeme gewährleistet sein und Aspekte wie stark variable Lichtverhältnisse und die Fahrerposition müssen beachtet werden.

2 Voraussetzungen, unter denen das Vorhaben durchgeführt wurde

Die Robert Bosch GmbH verfügt durch vorausgegangene Projekte über umfangreiche Erfahrungen und Kenntnisse in den Arbeitsgebieten des Vorhabens. Insbesondere im Bereich der fahrzeugspezifischen Anforderungen für Fahr- und Fahrerassistenz sowie der menschengerechten Bedienung von In-Car-Systemen existieren umfangreiche Expertisen.

Der Bereich der Bildsequenzanalyse wird durch langjährige Erfahrungen sowohl bei der Erfassung des Fahrzeug-Innenraums als auch des Fahrzeug-Außenbereichs unterstützt.

Im Themenumfeld SW-Agenten besteht durch die Mitarbeit in der FIPA (Foundation for Intelligent Physical Agents) ein fundiertes Wissen. Darüber hinaus wurde bereits im Projekt Motiv-PTA durch Bosch die Entwicklung und der Einsatz von Agenten auf der Infrastrukturseite zur Dienstenutzung untersucht. Das vorhandene Know-How war für dieses Vorhaben von außerordentlichem Nutzen.

3 Planung und Ablauf des Vorhabens

Zur effizienten Bearbeitung wurde das Vorhaben Bosch-intern nach Themengebieten geordnet und in verschiedene Arbeitspakete unterteilt. Diese werden im folgenden kurz beschrieben.

API: Systemarchitektur (Schnittstellen und Komponentenstruktur)

In diesem Arbeitspaket sollte die Gesamtprojekt-Architektur erarbeitet werden. Dazu sollten zunächst die funktionalen und nicht-funktionalen Anforderungen in den unterschiedlichen Anwendungsbereichen (Home, Kfz, Terminal) erarbeitet werden. In den unterschiedlichen Anwendungsbereichen wurden Szenarien erarbeitet und Gemeinsamkeiten und Unterschiede analysiert. Auf dieser Basis und unter Berücksichtigung bestehender Standards und Technologien sollte eine geeignet Funktions- und Kommunikations-Struktur entwickelt werden, die allen Anwendungsbereichen gerecht wird und bereichsübergreifend gleiche Strategien und Module zur Verfügung stellt.

AP2: Neue MMI-Technologien und Assistenzsysteme

Dieses Arbeitspaket umfaßte alle technologischen Untersuchungen und Entwicklungen. Dazu gehörte die Evaluation und Weiterentwicklung von Methoden zur groben und genauen Blickrichtungserkennung.

Darüber hinaus wurden in dieser AP die SW-Agententechnologie sowie weitere Verfahren der künstlichen Intelligenz, die zur Umsetzung der Assistenzfunktionalität notwendig sind, untersucht.

In diesem Arbeitspaket erfolgte auch die Analyse der ausgewählten Applikationen und die Festlegung und Ausgestaltung der einzelnen Assistenzgrade. Dabei standen insbesondere auch adaptive, selbst-lernende Nutzerpräferenzen im Vordergrund sowie die Nutzung von Inhalte-Beschreibungssprachen wie MPEG 7.

AP3: Demonstrator und Sitzkiste

Um die Überlegungen zur einheitlichen Architektur und die Bedienassistenten-Konzepte zu verifizieren und um die Ergebnisse des Vorhabens evaluieren und präsentieren zu können, sollte ein Demonstrator aufgebaut werden. Dieser sollte in einer Sitzkiste (Mock up) realisiert werden und mit einer virtuellen Fahrsimulation gekoppelt sein, um möglichst realistisch die Bedienung dieser neuen Assistenz prüfen und erfahren zu können.

AP4: Nutzerevaluationen

Die Nutzerevaluationen selbst sollten weitgehend bei der Humboldt Universität stattfinden, was aus Kapazitäts- und Komplexitätsgründen für den Demonstratoraufbau nicht immer möglich war. Daher wurden einige Evaluationen mit Hilfe von Studenten bei Bosch durchgeführt und von der Humboldt Universität betreut.

Die Nutzerevaluationen sollten technisch begleitet und unterstützt werden.

AP5: Implementierung, Integration und Test

Dieses Arbeitspaket umfaßte die Implementierung der einzelnen SW-Module und die Integration in die Sitzkiste. Weiterhin sollte hier die Einbindung aller Bosch-externen Module in den Kfz-Gesamtdemonstrator erfolgen.

Ebenso war hier die Umsetzung der Kontexterfassung (Fahrsituation und Ableitung des Fahrerzustandes) sowie die Module der Nutzerpräferenzen eingeordnet.

Die Bearbeitung der einzelnen Arbeitspunkte erfolgte zum Teil in intensiver Zusammenarbeit mit den anderen Projektpartnern. Innerhalb des Gesamtprojektes gab es eine Struktur mit unterschiedlichen Arbeitsgruppen, die sich mit der internen Arbeitsgruppenstruktur deckt und in die jeweils Beteiligte aus den internen Arbeitspaketen entsandt wurden. So gab es im Gesamtprojekt eine Arbeitsgruppe W-Tech, darunter mit einer Gruppe T-Vision, in der die Blickrichtungserkennung bearbeitet wurde und einer Gruppe T-Assist für die Assistenzrealisierung. Weiterhin gab es eine Arbeitsgruppe W-Konz, in der die Architekturkonzepte behandelt wurden, eine Arbeitsgruppe W-Eva für die Nutzerevaluationen und eine Arbeitsgruppe W-Imp, in der die Implementierungen für die Demonstratoren thematisiert wurden. Die Gesamtkoordination für den Kfz-Demonstrator erfolgte durch Bosch in dem übergeordneten Anwendungsbereich A-Kfz.

Das Projekt war insgesamt in zwei Phasen unterteilt worden, wobei die erste Phase bis Ende Februar 2001 andauerte (Abschluss mit der MC 2001). Dadurch konnte zur Hälfte der Projektlaufzeit ein erster Demonstrator für Nutzerevaluationen zur Verfügung gestellt werden und eine erste öffentliche Präsentation der Ergebnisse stattfinden.

4 Wissenschaftlicher und technischer Stand

Der wissenschaftliche und technische Stand, auf dem die Arbeiten des Vorhabens aufgesetzt haben, wurde zu Beginn des Projektes für jedes Themenfeld und alle Technologien, die den Projektpartnern bekannt waren und von ihnen recherchiert werden konnten, zusammengeführt. Das Ergebnis wurde in dem EMBASSI-State-of-the-Art-Report dokumentiert. Dieses Dokument umfasst 330 Seiten und beinhaltet darüber hinaus für jedes Themenfeld die wichtigsten internationalen Literaturverweise und Akteure auf dem jeweiligen Gebiet.

Für die Recherche des Stand der Technik wurde das Internet, Fachliteratur sowie einschlägige Datenbanken wie INSPEC oder ZDEE verwendet. Darüber hinaus konnten zusätzlich firmeninterne Wissensspeicher (umfangreiche Bosch-interne Bibliotheken und Wissensdatenbanken) verwendet werden.

5 Zusammenarbeit mit anderen Stellen

Innerhalb des Anwendungsszenarios Kraftfahrzeug arbeitete die Robert Bosch GmbH auf dem Gebiet der Spracherkennung und es Dialogmanagements insbesondere mit der Siemens AG und FORWISS (Bayerisches Forschungszentrum für Wissensbasierte Systeme), auf dem Gebiet der Blickrichtungserkennung mit dem Fraunhofer-IIS, HumanScan und der Fraunhofer IGD, auf dem Gebiet der Fahrsimulation mit dem

ZGDV(Zentrum für Graphische Datenverarbeitung) und auf dem Gebiet der Nutzerevaluationen mit der Humboldt Universität zu Berlin zusammen. Die Architektur wurde im Rahmen des Gesamtprojektes mit allen Partnern erarbeitet.

6 Darstellung der Ergebnisse

6.1 Architektur

Die Architektur wurde unter Mithilfe aller Partner entwickelt und auf die einzelnen Anwendungsbereiche Home, Kfz und Terminal abgebildet. Für allgemeinere Informationen siehe [Hildebrand 2000], [Herfet et al. 2000] und [Herfet et al. 2001]. Für den Kfz-Bereich, für den Bosch zuständig war, ist diese Abbildung im Folgenden dargestellt:

6.1.1 Abbildung der Generischen Embassi Architektur auf das KFZ- Szenario

Im Rahmen des Arbeitspaketes 1 wurde zunächst eine umfangreiche Anforderungsanalyse und Szenarien-erarbeitung im Gesamtprojekt durchgeführt. Die Ergebnisse bildeten die Basis für die im folgenden erstellten Konzepte, Umsetzungen und Untersuchungen und wurden im Rahmen der Spezifikationen dokumentiert. Eine Darstellung der Architektur ist in Abbildung 1 abgebildet. Sie zeigt sowohl die logische Trennung der wesentlichsten Komponenten der Architektur als auch das Deployment der Realisierung.

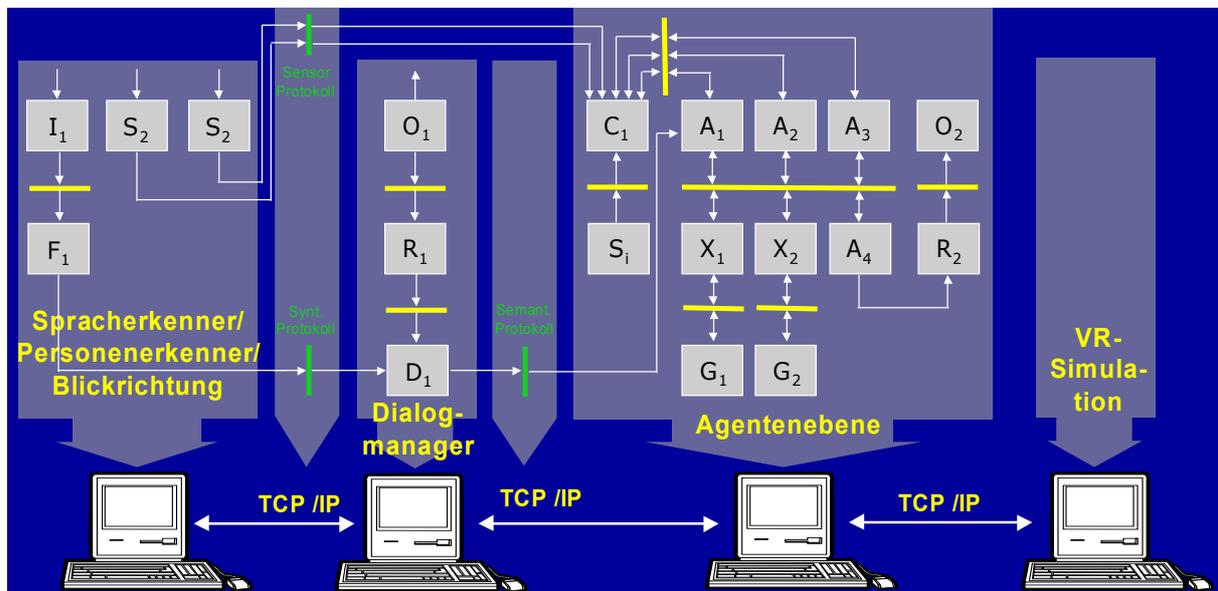


Abbildung 1: Abbildung der generischen EMBASSI-Architektur auf den Kfz-Bereich

Die einzelnen Komponenten dieser Architektur (siehe Abbildung 1) werden in der Tabelle 1 aufgelistet und erläutert:

Abkürzung	Bedeutung in EMBASSI	Realisierung im KFZ- Szenario	Bereitstellender Partner
I ₁	Unimodale Eingabegeräte	Mikrophon	Bosch
F ₁	Filterkomponente	Spracherkennung	Siemens
S ₂	Sensor	Personenerkennung	BioID
D ₁	Dialogmanager	Dialogmanager	Forwiss
R ₁	Renderer	TTS (Entertainment)	Forwiss
O ₁	Unimodale Ausgabegeräte	Lautsprecher	Bosch
C ₁	Contextmanager	Contextmanager (Java RMI)	Bosch
S _i	Sensor	Sensordaten der Simulation	Bosch / IGD
A ₁	Agent	Entertainment - Agent	Bosch
A ₂	Agent	Navi- Agent	Bosch

A ₃	Agent	Warn- Agent	Bosch
A ₄	Agent	Fahrerzustands- Agent	Bosch
R ₂	Renderer	TTS (Navi) und Touch GUI	Bosch
O ₂	Unimodale Ausgabegeräte	Lautsprecher / Bildschirm	Bosch
X ₁ G ₁	Ausführungskomponente Gerät	Off- Board Navi	Bosch
X ₂ G ₂	Ausführungskomponente Gerät	Multimedia Player	Bosch

Tabelle 1: Einzelkomponenten und deren Realisierung im KFZ-Szenario

Definition der Ontologie KFZ- Szenario

Wie in Kapitel 6.6.1 beschrieben wird besteht eine der Hauptaufgaben bei der Realisierung eines Systems, welches als Agenten System realisiert ist, darin, die Ontologie zu definieren. Diese Ontologie ist im allgemeinen domänenspezifisch, da eine Ontologie, die versucht, zu viele Details zu beschreiben, u.a. an ihrer Größe, Komplexität und Dauer der Zusammenstellung scheitert. Aus diesem Grund wurde in EMBASSI der pragmatische Weg der Domänenspezifischen Ontologie gewählt. Für den KFZ-Bereich wurde der Audio-Entertainment- Bereich für die Sprachbedienung ausgewählt. Die Aufgabe bestand darin, für diese spezielle Domäne eine Ontologie zu definieren und diese in die Gesamt-EMBASSI- Ontologie (die alle EMBASSI Szenarien beinhaltet) zu integrieren.

Als Quelle der Definition dient die Domäne selbst bzw. die Entitäten, die in ihr vorkommen bzw. die Funktionalität, welche die zu bedienende Applikation bietet. Bei der Erstellung der Ontologie wurde die Ontologie zuerst von BOSCH als Klassen-Modell vorgegeben und dann von FORWISS in die Ontologie Darstellung in Form einer Document Type Declaration (DTD)(siehe auch <http://www.w3.org/TR/REC-xml#dt-doctype>) umgewandelt. Im weiteren wurden die verschiedenen Ontologien aller EMBASSI Szenarien durch FORWISS zu einer Gesamtontologie zusammengefügt.

Abbildung 2 zeigt Multimedia- bzw. KFZ- Entertainment-spezifische Teile der Gesamt-Ontologie als XML-Schema Darstellung:

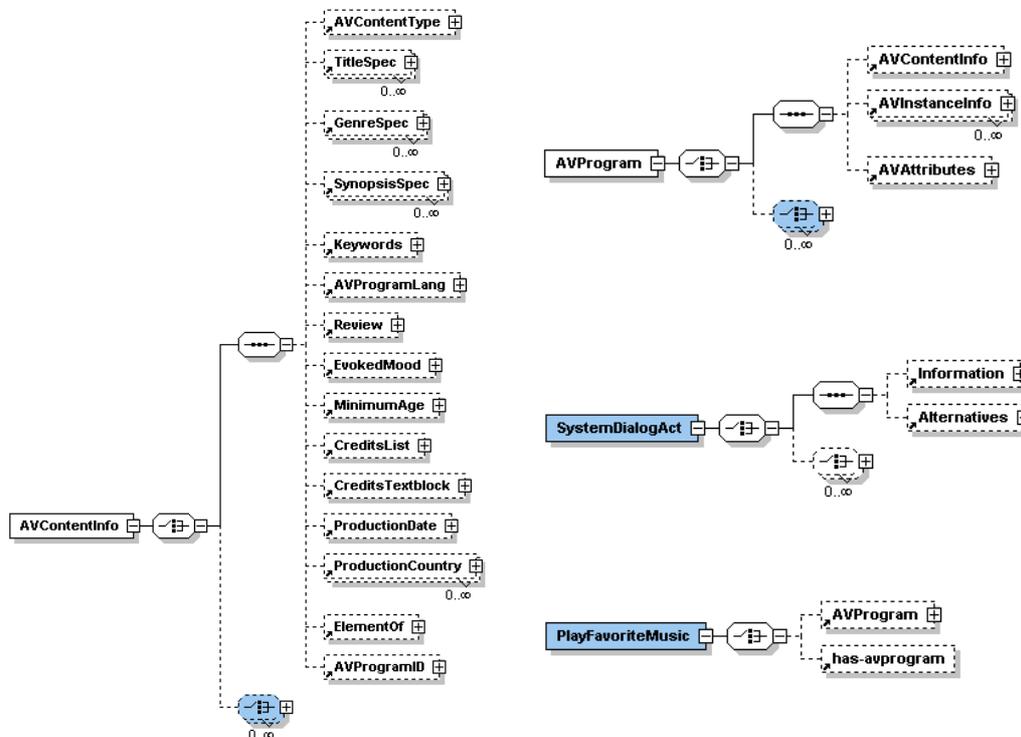


Abbildung 2: Zwei beispielhafte Kfz-spezifische Ontologie-Teile in XML-Schema-Darstellung

Die entsprechenden DTD's ergeben sich wie folgt:

EntertainmentCDControls.dtd:

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT EntertainmentControls ((StopTrack | SkipForwardTrack | SkipBackwardTrack | SkipForwardCD |
SkipBackwardCD | PlayTrack))>
<!ELEMENT StopTrack (#PCDATA)>
<!ELEMENT PlayTrack (#PCDATA)>
<!ELEMENT SkipForwardTrack (#PCDATA)>
<!ELEMENT SkipBackwardTrack (#PCDATA)>
<!ELEMENT SkipForwardCD (#PCDATA)>
<!ELEMENT SkipBackwardCD (#PCDATA)>
```

EntertainmentListControls.dtd:

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT EntertainmentControls ((StepElementsForward | StepElementsBackward | StepListBackward)) >
<!ELEMENT StepElementsForward (#PCDATA)>
<!ELEMENT StepElementsBackward (#PCDATA)>
<!ELEMENT StepListBackward (#PCDATA)>
```

6.1.2 Kommunikation

Die Kommunikation der einzelnen Komponenten in EMBASSI ist weitgehend als Agentenkommunikation (siehe Kapitel 6.6.1) realisiert. Ausnahmen bilden hierbei aus Aufwandsgründen konkret realisierte HW-Sensoren wie z.B. die Pedalsensorik.

Bei dieser Art der Kommunikation werden zwischen den einzelnen Komponenten Textnachrichten ausgetauscht. Unterscheidet man dabei 3 verschiedene Ebenen (jede Ebene bildet dabei eine logische Einheit bezüglich der Funktion bzw. der Inhalte der Kommunikation) so basiert die Kommunikation auf folgenden Technologien:

- TCP/IP Socket Connection
- KQML Message Routing (über EMBASSI Server) [Genesereth 1994],[Labrou & Finin 1997]
- XML Content Language [W3C XML 2000]

Um eine Nachricht von einer Komponente zur anderen versenden zu können, muss diese Komponente:

- Zugang zum gemeinsamen Netzwerk haben.
- sich beim EMBASSIServer anmelden.
- eine KQML Nachricht generieren (mit u.a. Angabe des Senders, Empfängers, Typ der Nachricht (Performative z.B.: achieve, inform o.ä.) etc.).
- den XML-Inhalt erzeugen (was soll gemacht werden) und in die KQML Nachricht einhängen.
- die Nachricht über den EMBASSI Server an die Komponente senden.

Das Parsing-Konzept

Hat eine Komponente eine Nachricht erhalten, erfordert das Konzept ein Parsing. Dabei wird zunächst der KQML Anteil ausgewertet, um zu ermittelt, von wem die Nachricht kommt und von welchem Typ (Performative z.B.: achieve, inform o.ä.) die Nachricht ist. Als nächster Schritt erfolgt dann das Parsing des Inhalts (in XML-Format) der KQML Nachricht. Diese Inhalte müssen dabei in der Sprache, die durch die Ontologie vorgeschrieben ist, in XML beschrieben sein. Dann werden die Anweisung auf ausführbaren Programmiercode umgesetzt und ausgeführt. Reichen die Informationen zur Durchführung einer Aktion nicht aus, so kann der Empfänger mit einer definierten Nachricht Rückfragen an den Sender stellen.

Bei der Verwendung von XML als Inhalts-Sprache der Agenten ist ein Konzept erforderlich, mit dem ein effizientes Parsing von XML erfolgen kann. Dabei ist es aus Gründen der einfachen Verarbeitung bzw. Erweiterbarkeit anzustreben, für die XML Repräsentation eine geeignete Abbildung auf Objekte der verwendeten Programmiersprache (hier Java) durchzuführen. Für diese Umwandlung von XML in Java bzw.

umgekehrt gibt es bereits einige Frameworks, die im einzelnen auf ihre Leistungsfähigkeit untersucht wurden. Die betrachteten Frameworks Quick [Quick 2000], Suns Data Binding [Sun Data Binding 2003] und Castor [Castor 2003] basieren alle auf der Voraussetzung, dass vorhandene Java Klassen in XML umgewandelt werden sollen. Die Umsetzung erfolgt dabei nach dem folgenden Prinzip:

Java:	<- conversion ->	XML :
Klasse_1	<- conversion ->	<Tag_1>
MemberKlasse_1	<- conversion ->	<Attribut_1>
o. SimpleMemberType_1		DATA_1
		</Attribut_1>
.		
.		
MemberKlasse_n	<- conversion ->	<Attribut_n>
o. SimpleMemberType_n		DATA_n
		</Attribut_n>
		</Tag_1>
.		
.		
Klasse_n	<- conversion ->	<Tag_n>
usw.		

Das bedeutet, dass für jedes weitere Unterelement immer jeweils eine weitere Klasse benötigt wird. Betrachtet man jedoch umfangreichere XML Definitionen wie sie z.B. bei MPEG7 [MPEG 2003] verwendet werden, so fällt auf, dass diese sehr viel mehr Unterelemente besitzen als die zuvor betrachtete Variante. Das gilt ebenso für die EMBASSI-Ontologie (siehe folgendes Beispiel):

```
<PlayTrack>
  <AVProgram>
    <AVContentInfo>
      <GenreSpec>
        <GenreGeneric>
          <GenreGenericValue>HipHop</GenreGenericValue>
        </GenreGeneric>
      </GenreSpec>
    </AVContentInfo>
  </AVProgram>
</PlayTrack>
```

Zur Abbildung dieser XML Struktur in Java werden auf der Java Seite bereits 5 in einander verschachtelte Java Klassen und ein Klassenattribut benötigt. Eine Abbildung auf eine Klasse mit einem Klassenattribut würde jedoch ausreichen und die Implementierung wesentlich vereinfachen und damit auch im Ablauf beschleunigen. Aus diesem Grund wurde im KFZ Entertainment Bereich von EMBASSI ein anderes Verfahren zur Java -> XML bzw. XML-> Java Umwandlung angewendet. Anstatt der Verwendung der Frameworks Quick, Suns Data Binding und Castor kommt hier das XML Tool XSLT und ein einfacher Java-> XML XML-> Java Converter von SUN zum Einsatz. Dieser Konverter wirft eine Art Java XML heraus, welches nahe an der Klassendarstellung von Java liegt. Mit XSLT lassen sich prinzipiell zwei beliebige XML Dokumente ineinander transferieren. Die Umwandlung oder Übersetzung der beiden XML Dialekte erfolgt mit Hilfe eines XLS Dokumentes, welches die Konvertierungsvorschriften enthält.

6.1.3 GUI

6.1.3.1 Konzept

Im Rahmen dieses Arbeitspaketes wurde ein Konzept für ein Bediensystem entworfen, das alle Kfz-Applikationen unter einer einheitlichen Bedienoberfläche zusammenfasst. Ziel dieser Oberfläche ist eine konsistente Bedienung aller Assistenzkomponenten, wie es übliche Fahrerinformationssysteme (FIS) gewährleisten [Ross 1997]. Dabei sollte die Bedienung die Kriterien für eine fahrzeugtaugliche Umgebung

erfüllen. Hierzu zählt, dass die Blickabwendungen von der Straße während der Bedienung so kurz wie möglich gehalten werden.

Zur Berücksichtigung der ergonomischen Aspekte und Randbedingungen im Fahrzeug erfolgte eine enge Zusammenarbeit mit Psychologen und Ergonomern im Projektteam und intern bei Bosch.

Unter Berücksichtigung der Anforderungen entstand das nachfolgend beschriebene Konzept für die Bedienoberfläche (siehe Abbildung 3).

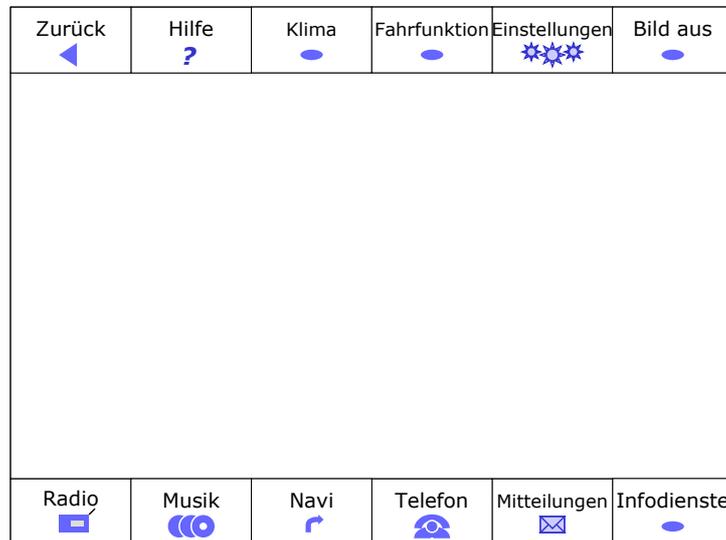


Abbildung 3 Schematischer Aufbau des Fahrerinformationssystems

Das FIS unterteilt sich in zwei Bereiche. Am oberen und unteren Rand des Bildschirms befinden sich jeweils 6 Funktionstasten, mit denen die wichtigsten Funktionen des FIS im direkten Zugriff des Fahrers stehen. Durch einen einzigen Tastendruck kann der Nutzer eine Applikation (Navigation, Entertainment, etc.) auswählen. Lange Bedienschritte zur Selektion werden somit vermieden. Die gewünschte Applikation wird daraufhin im Hauptbereich auf dem Bildschirm dargestellt. Abzüglich des von den 12 Funktionstasten eingenommenen Platzes besitzt dieser Bereich das im Multimediabereich übliche Bildschirmverhältnis 16:9. Das FIS wird mit Hilfe eines Touchscreens gesteuert, da diese Lösung für ein prototypisches System die flexibelste Lösung darstellt.

Dies hatte Auswirkungen auf die ergonomische Gestaltung der Bedienoberfläche. So ist es z.B. notwendig, dass alle auf dem Bildschirm dargestellten Bedienelemente (z.B. Tasten) eine Mindestgröße von 1,5 x 1,5 cm besitzen. Nur so kann gewährleistet werden, dass der Fahrer auch die von ihm beabsichtigte Taste drückt. Bei kleineren Ausmaßen besteht die Gefahr, dass eine danebenliegende Taste berührt und somit unbeabsichtigte Funktionen ausgeführt werden. Außerdem erfordern kleinere Bedienelemente eine höhere Feinmotorik des Benutzers, die in einem sich bewegenden Fahrzeug kaum zu gewährleisten ist.

Die vom Fahrer durch Tastendruck aktivierte Applikation wird exklusiv auf dem Bildschirm dargestellt und kann abzüglich der Funktionstasten den gesamten freien Raum auf dem Bildschirm für ihre Ausgaben verwenden. Es gibt keine Splitscreens oder andere Techniken, um die Ausgaben mehrerer Applikationen auf dem Bildschirm zu vermischen. Durch die zu erwartende Informationsüberflutung des Fahrers wurden solche Verfahren nicht weiter verfolgt.

Das FIS unterstützt die personenbezogene Einstellung der Applikationen und der Bedienoberfläche. Somit kann sich jeder Fahrer die Umgebung nach seinen Vorlieben einrichten. Von der Personalisierung sind verschiedene Bereiche betroffen. Hierzu zählt z.B. das dynamische Wechseln der Landessprache zwischen Deutsch und Englisch. Außerdem integriert das FIS einen Mechanismus zum Wechsel des Oberflächendesigns. Dieses Konzept erlaubt dem Fahrer, zwischen verschiedenen Tag- und Nachtdesigns auszuwählen. Genau wie bei der Sprachwahl ist dieser Wechsel jederzeit zur Laufzeit des Systems möglich. Es ist also nicht notwendig, das FIS neu zu starten.

Neben diesen oberflächenspezifischen Einstellungsmöglichkeiten umfasst die Personalisierung auch applikationsabhängige Einstellungen. So berücksichtigt beispielsweise die Entertainmentapplikation die individuellen Musikvorlieben des Fahrers. Beim Wechsel des Fahrers erhält der neue Fahrer seine persönlichen Lieblingslieder und nicht die des letzten Fahrers. Weiterhin können Assistenzgrade für jeden Benutzer eingestellt

werden, um die Erfahrung im Umgang mit dem System besser berücksichtigen zu können. So ist es auch möglich, dass erfahrene Benutzer Assistenzen individuell verringern bzw. deaktivieren.

Ein besonderer Schwerpunkt bei der Konzeption lag darauf, die Navigation und Orientierung innerhalb des FIS zu verbessern. Hierbei spielten zwei Aspekte eine Rolle. Zum einen sollen im System nur Informationen angezeigt werden, die für den Fahrer eine hohe Relevanz besitzen. Alle anderen werden zu Gunsten einer schnelleren Wahrnehmung des Bildschirminhalts unterdrückt. Aus diesem Grund wurden bewusst Statusinformationen ausgelassen, die in einem FIS üblicherweise permanent in einer Statuszeile dargestellt werden, z.B. Anzahl empfangener Satelliten, die Feldstärke des Telefons oder die derzeitige Senderkennung.

Außerdem wurde das Problem aufgegriffen, dass ein Fahrer einen Dialog mit dem System durchführt und dann aufgrund der Verkehrsbedingungen für längere Zeit auf die Straße schauen muss. Beim Blick zurück auf den Bildschirm muss er erneut herausfinden, um welchen Dialog es sich handelt und welche Eingaben aktuell erwartet werden. Um diese Orientierung zu unterstützen, ist in jeder Applikation ein fester Bereich vorgesehen, in dem der Name bzw. der Zweck des aktuellen Dialogs eingeblendet wird (siehe Abbildung 4). Ohne diese Navigationszeile müsste der Fahrer anhand des spezifischen Seitenaufbaus den Zweck des Dialogs identifizieren. Bei ähnlichen Seiten kann diese Aufgabe zeitaufwendig sein und dadurch die Blickabwendung von der Straße erhöhen.



Abbildung 4: Die Navigationszeile unterstützt den Fahrer bei der Orientierung

Alle Applikationen, die nicht aktiv sind, d.h. nicht auf dem Bildschirm dargestellt werden, speichern ihren aktuellen Zustand. Werden sie erneut dargestellt, findet der Fahrer sie genau in dem Zustand vor, wie er sie das letzte Mal verlassen hatte. Dadurch kann der Fahrer jederzeit zwischen den Applikationen wechseln, auch wenn er sich gerade in einem längeren Dialog mit einer Applikation befindet. So kann er z.B. während der Zieleingabe der Navigation zum Entertainmentssystem wechseln und ein neues Lied auswählen. Er wird nicht gezwungen, zuvor den laufenden Dialog abzuschließen. Diese Freiheit bei der Bedienung ist besonders während der Fahrt wichtig.

Zur Vereinfachung der Bedienung im System gibt es außerdem eine permanente Zurück-Funktion (siehe Abbildung 5). Diese erlaubt es, jederzeit die letzten Bedienschritte innerhalb einer Applikation rückgängig zu machen. Diese Zurück-Funktion ist unbegrenzt, d.h. der Fahrer kann durch wiederholtes Drücken der Taste beliebig viele Schritte zurückkehren. Von der daraus resultierenden erhöhten Fehlertoleranz des Systems profitiert die Touchscreen-Bedienung, da versehentliches Berühren des Bildschirms oder das Verfehlen der gewünschten Taste leicht korrigiert werden können.

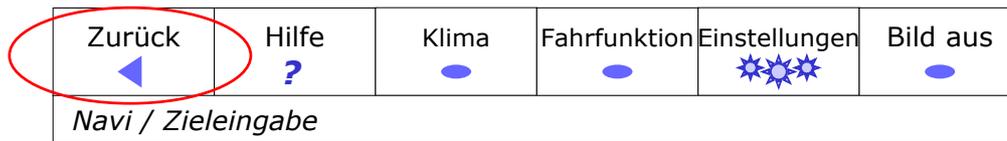


Abbildung 5: Die Zurück-Taste zur schnellen Korrektur von Bedienfehlern

6.1.3.2 Realisierung

Im Arbeitspaket 5 wurden die zuvor beschriebenen Konzepte in eine Systemarchitektur umgesetzt und implementiert.

FIS-Rahmen

Eine für den Fahrer wichtige Komponente ist das FIS, da erst hierdurch die Interaktion mit dem Gesamtsystem ermöglicht wird. Das realisierte FIS setzt sich aus zwei Teilen zusammen, den Applikationen und einem Rahmen, mit dem die Applikationen gesteuert werden können. Letzteres ist Aufgabe einer Rahmenkomponente. Dabei handelt es sich um ein grafisches Gerüst für das Fahrerinformationssystem. Dieses integriert die Infotainment-Applikationen und steuert bzw. verwaltet deren Ausgaben. Diese Rahmen GUI belegt den kompletten Bildschirm der Mittelkonsole und ist in zwei Bereiche aufgeteilt, den Applikations- und den Funktionsbereich. Der Funktionsbereich zeigt Tasten, mit denen die zugehörigen Applikationen aktiviert werden können, der Applikationsbereich wiederum stellt die aktivierte Applikation dar.

Neben diesen Basisfunktionen bietet die Rahmen GUI eine Reihe von Möglichkeiten, die über die Fähigkeiten herkömmlicher FIS hinausgehen. Eine für den Fahrer sehr offensichtliche Erweiterung ist der Austausch der Oberfläche. Das heißt, dass das Oberflächendesign nicht fest ist, sondern jederzeit ausgetauscht werden kann. Dieses sogenannte "Skinning" erlaubt es, beispielsweise das FIS konservativ, modern oder auch jugendlich darzustellen (siehe Abbildung 6).



Abbildung 6: Verschiedene, frei wählbare Designs der Oberfläche (während der System-Laufzeit umschaltbar)

Die Architektur des FIS ist so ausgelegt, dass der Wechsel des Designs jederzeit möglich ist, es ist kein Neustart bzw. Reinitialisierung notwendig. Hierzu wurde die flexible Look-and-Feel-Architektur von Java als Basis genutzt und um die Spezifika des FIS erweitert. So können beim Designwechsel neben den Grafiken auch viele andere Oberflächeneigenschaften geändert werden, beispielsweise Farben, Schriftarten oder auch die Breite von Trennlinien.

Der Wechsel des Designs erfolgt für die Applikationen vollkommen transparent. Somit können sich die Entwickler der Applikationen vollkommen auf die Funktion konzentrieren, das Oberflächendesign kann davon getrennt gestaltet werden (siehe Abbildung 7). Dies ist besonders im Hinblick auf die Wünsche der Fahrzeughersteller bezüglich eines Corporate Designs von Vorteil.

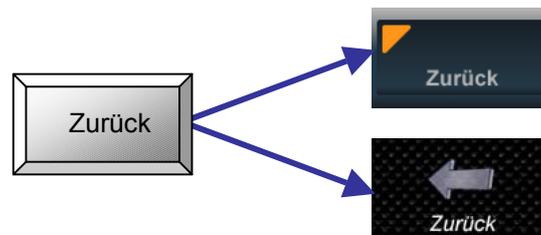


Abbildung 7: Bedienelemente ändern durch den Austausch der Grafiken nur ihre Gestaltung (Skinning), nicht ihre Funktion

Eng verbunden mit dem Designwechsel ist die Fähigkeit des FIS die Sprache zu wechseln. Das System ist konsequent mehrsprachig ausgelegt und kann an verschiedenste Landessprachen und sogar Sprachvarianten angepasst werden, ohne dass die Software geändert werden muss. Der Wechsel der Sprache ist wie auch der Designwechsel komplett dynamisch zur Laufzeit möglich.



Abbildung 8: Die Sprache der Oberfläche kann dynamisch geändert werden

Um dies zu ermöglichen, muss die Beschriftung der Oberflächenelemente abstrahiert werden. Hierzu werden alle GUI-Elemente mit logischen Namen belegt, anstatt sie in einer konkreten Sprache zu beschriften. Für die vom FIS unterstützten Sprachen existiert eine Übersetzungstabelle, die für alle logischen Beschriftungen den übersetzten Text beinhaltet (siehe Abbildung 9). Bei der Konzeption des GUI-Systems wurde darauf geachtet, dass die Neubeschriftung der GUI-Elemente beim Sprachwechsel ebenfalls transparent für die Applikationen erfolgt. Diese müssen somit keine Softwareunterstützung für den Sprachwechsel beinhalten, weswegen die Entwicklung neuer mehrsprachiger Applikationen für das FIS wesentlich vereinfacht wurde.

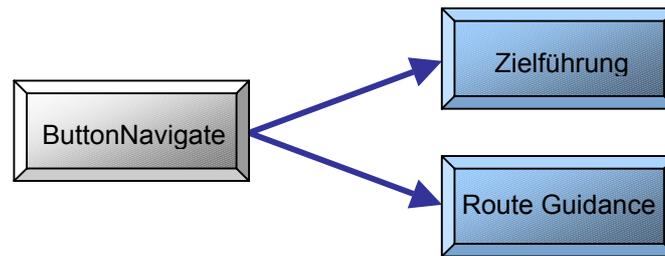


Abbildung 9: Oberflächenelemente werden je nach Sprache anders beschriftet

Durch seine Mehrsprachigkeit kann das FIS auch weitere länderspezifische Eigenheiten berücksichtigen. Beispielsweise ist das Oberflächendesign ebenfalls landesabhängig ausgelegt. Somit können beispielsweise Symbole für deutsche und englische Oberflächen verschiedenartig gestaltet sein. Dies ist besonders dann von Vorteil, wenn keine weltweit eindeutigen Symbole genutzt werden können. So können die Symbole optimal an die kulturellen Gegebenheiten eines Landes angepasst und missverständliche Symbole mit negativem Charakter vermieden werden.

Applikationsinterface

Die Rahmen GUI wurde so konzipiert, dass Applikationen flexibel in das FIS integriert werden können. Die Applikationen folgen in ihrem Aufbau dem Designpattern Model-View-Controller, das eine Entkopplung der Applikationsdaten und –funktionen von deren Darstellung vorsieht [GoF 1994]. Hierdurch ist es möglich, die Bedienerführung im FIS anzupassen, ohne dass die Funktion der Applikation an sich geändert werden muss.

Basis der Einbindung von Applikationen ist ein Set von Schnittstellen, die wichtigste hierfür ist die Schnittstelle mit dem Namen IApp. Jede Applikation, die diese Schnittstelle erfüllt, kann in das System eingebunden werden. Sie definiert, wie Applikationen ihre Inhalte im FIS darstellen, ihren aktuellen Inhalt durch einen aussagekräftigen Titel beschreiben, auf das Drücken der Zurück-Taste oder etwa auf einen Wechsel des Benutzerprofils reagieren.

Applikationen sind für die Darstellung ihres Inhalts selbst verantwortlich. Hierzu definiert die Schnittstelle IApp Funktionen, mit denen der Inhalt des Hauptfensters und der Navigationszeile erfragt werden kann. Diese Inhalte werden bei Aktivierung der Applikation in die Bedienoberfläche des FIS eingeblendet. Durch dieses Konzept haben Applikationen bei der Gestaltung ihres Inhaltes weitreichende Möglichkeiten.

HMI Framework

Die Freiheit der Applikationen zu beliebigen Ausgaben kann dazu führen, dass Darstellung und Bedienung der Applikationen sehr verschieden sind. Um dem entgegenzuwirken und ein einheitliches Aussehen sowie die fahrzeugtaugliche Bedienung zu gewährleisten, wurden der Rahmen GUI umfangreiche Funktionen zur Seite gestellt. Das daraus resultierende HMI Framework unterstützt den Entwickler bei der Realisierung der Applikationen. So sind oft benötigte Funktionalitäten zentral verfügbar. Dies vereinfacht die Entwicklung und stellt die konsistente Bedienung sicher. Häufig benötigte Bedienelemente sind im Framework abrufbar und können von allen Applikationen eingebunden werden.

Das HMI Framework besitzt hierfür zentrale Schnittstellen zur Generierung der GUI-Elemente, die mit Hilfe des Factory-Design Patterns [GoF 1994] umgesetzt wurden. So existieren Factories zur Erzeugung von einfachen Bedienelementen wie Tasten oder Listenfeldern, aber auch für komplexe Komponenten wie z.B. eine Bildschirmtastatur (siehe Abbildung 10). Jede Applikation, die Texteingaben vom Fahrer benötigt, kann auf diese zurückgreifen und von Funktionalitäten wie z.B. automatische Anpassung des Tastenlayouts an die Landessprache oder Komfortfunktionen wie eine Tastenwiederholung bei längerem Druck profitieren. Um solche Funktionen zu realisieren, müssten die Applikationsentwickler ansonsten erhebliche Entwicklungsarbeit leisten. Außerdem würden sich applikationsspezifische Versionen solcher Komponenten in der Bedienung gegebenenfalls erheblich unterscheiden.



Abbildung 10: Das HMI Framework enthält komplexe Bedienelemente wie eine Bildschirmtastatur

Neben solchen GUI-Elementen beinhaltet das Framework auch Funktionen zur Datenverwaltung. Beispielsweise stellt sie den Applikationen ein API zur Verwaltung der Bedienhistorie zur Verfügung. Eingebunden in Applikationen erlaubt dieses API eine Navigation die mit einem Web-Browser vergleichbar ist und dem Benutzer eine Vor- und Rückwärtsbewegung innerhalb der Bedienhistorie einer Applikation erlaubt.

Weitere Teile des Frameworks unterstützen die Personalisierungsfunktionen der Rahmen GUI. Sie erlauben beispielsweise die softwaregesteuerte Änderung des GUI-Designs oder der Landessprache. Durch Bereitstellung von Listener-Schnittstellen können außerdem Applikationen auf die Änderungen solcher Personalisierungsoptionen reagieren und somit ihre Ausgaben eng auf die Einstellungen abstimmen.

Kombiinstrument

Neben dem Fahrerinformationssystem wurde auch ein Kombiinstrument realisiert (siehe Abbildung 11). Dieses zeigt auf einem separaten Bildschirm die Betriebsdaten an, die von einem Fahrsimulator gesendet werden. Dazu stellt ein Tachometer die Geschwindigkeit und den Kilometerstand des virtuellen Fahrzeugs dar. Der Zweck des Instrumentes ist neben der Herstellung eines größeren Fahrzeugbezugs des Kfz-Demonstrators eine Unterstützungsfunktion bei der Durchführung von Evaluierungen. Das Kombiinstrument erlaubt es zum Beispiel, dass Versuchspersonen im Rahmen einer Nebenaufgabe vorgegebene Geschwindigkeiten einhalten können.

Eine weitere Aufgabe des Kombiinstrumentes ist die Ausgabe von wichtigen Meldungen mit Bezug zur Fahraufgabe. In einem separaten Bereich des Bildschirms können dazu Warnmeldungen eingeblendet werden. Diese Fähigkeit wird von dem Störungswarnassistenten genutzt, um den Fahrer auf gefährliche Situationen hinzuweisen. Durch die Einblendung in das Kombiinstrument werden lange Blickabwendungszeiten zum FIS der Mittelkonsole in solchen Situationen vermieden. Wie das FIS unterstützt auch diese Komponente die "Skinning-Funktionalität", bei der der gesamte Hintergrund ausgetauscht werden kann.



Abbildung 11: Das Kombiinstrument mit eingeblendeter Störungswarnung

Text-to-Speech

Neben der grafischen Bedienoberfläche werden zur Bedienerführung auch akustische Sprachhinweise im FIS genutzt. Für diesen Zweck wurde eine Text-to-Speech-Komponente zentral in die Rahmen-GUI integriert und allen Applikationen bereitgestellt. Dazu wurde eine kommerzielle Text-to-Speech-Komponente mit Hilfe des Java Native Interfaces (JNI) in das Java-System eingebunden. Die Schnittstelle wurde dabei so konzipiert, dass sie auf dem standardisierten Java Speech API (JSAPI) basiert und somit bei Bedarf ein Austausch der Sprachsoftware gegen eine anderes Modell möglich ist. Die Landessprache des Text-to-Speech-Systems kann durch die realisierte Schnittstelle dynamisch geändert werden und unterstützt somit die Mehrsprachigkeit des FIS.

6.2 Applikationsspezifische adaptive Bedienassistentz

6.2.1 Navigation

Ein Komponente, an der die adaptive Bedienassistentz im Fahrzeug demonstriert wurde, ist das Navigationssystem. Verwirklicht wurden eine Reihe von Funktionalitäten, die verschiedene Aspekte der Adaption zeigen [Console 2002]. Zum einen handelt es sich dabei um die direkt vom Benutzer gesteuerte Adaption der Bedienoberfläche unter Nutzung eines Benutzerprofils. Zum anderen wurde eine automatische Adaption aufgrund situativer Faktoren realisiert. Dabei passt sich die Navigationsassistentz in ihrer Funktionalität an den Fahrerzustand und die Fahrsituation an.

Notwendige Voraussetzung zur Implementierung der adaptiven Assistentz war es, ein bestehendes Navigationssystem in das EMBASSI-System zu integrieren. Zur Demonstration der situativen Assistentz ist es erforderlich, dass der Fahrer das Navigationssystem real verwenden kann und Zielführungshinweise erhält. Da der Demonstrator nicht in ein Kfz integriert wurde, musste auf eine Fahrsimulation zurückgegriffen werden. Diese erlaubt virtuelle Fahrten bei gleichzeitiger Verwendung eines Navigationssystems. Die Realisierung der adaptiven Fahrzeug-Navigation gliederte sich in folgende Arbeiten:

- Anbindung des Navigationssystems an einen Fahrsimulator
- Integration des Navigationssystems in das EMBASSI-System
- Erstellung einer Bedienoberfläche für das Navigationssystem und Integration in das FIS
- Realisierung der adaptiven Bedienassistentz

Anbindung des Navigationssystems an einen Fahrsimulator

Zur Durchführung von Fahrten mit dem Navigationssystem wurde eine vom ZGDV erstellte Fahrsimulation eingesetzt. Diese generiert die Fahrumgebung dynamisch anhand des digital erfassten Straßennetzes und erlaubt

uneingeschränkte Fahrten in dieser Landschaft. Der Fahrer kann demzufolge beliebige Routen abfahren und an Kreuzungen frei entscheiden, in welche Richtung er weiterfährt. Diese Funktionalität ist die Voraussetzung für die Kopplung mit dem Navigationssystem. Dieses kann in Kombination mit der Fahrsimulation reale Zielführungshinweise generieren, die den Fahrer in der virtuellen Welt zu einem vorgegeben Ziel führen. Genau wie bei einer realen Autofahrt besteht die Möglichkeit, dass der Fahrer die Zielführungshinweise nicht korrekt befolgt und somit falsch fährt. In diesem Fall kann das Navigationssystem dynamisch eine neue Route berechnen und den Fahrer trotz der Falschfahrt zum Ziel leiten.

Die Anbindung des Navigationssystems basiert darauf, dass die Fahrsimulation nicht nur eine virtuelle Fahrumgebung anhand realer Kartendaten darstellt, sondern dabei auch Informationen über Position und Richtung des virtuellen Fahrzeugs über eine Netzwerkschnittstelle nach außen sendet. Das angeschlossene Navigationssystem kann diese Informationen zur Ortung des Fahrzeugs in der virtuellen Welt nutzen. Damit übernimmt das Simulationssystem die Rolle eines GPS-Empfängers (Global Positioning System), der üblicherweise das Navigationssystem mit Positionsinformationen versorgt.

Die von der Fahrsimulation gesendeten Informationen können allerdings nicht direkt in das Navigationssystem eingespeist werden, da die verwendeten Datenformate und Schnittstellen nicht kompatibel sind. So erwartet das Navigationssystem alle Daten im standardisierten NMEA-Format [NMEA 2003], wie es von GPS-Empfängern genutzt wird. Die Simulation nutzt dagegen ein eigenständiges Format.

Aus diesem Grund war es notwendig, einen Datenkonverter zu erstellen, der sowohl Verbindung mit der Fahrsimulation als auch mit dem Navigationssystem aufnimmt (siehe Abbildung 12). Er vermittelt zwischen beiden Systemen, indem er gesendete Daten in Echtzeit von einem Format in das vom jeweilig anderen Partner benötigte umwandelt und anschließend weitersendet. Durch diese Verfahrensweise gestattet er dem Navigationssystem transparenten Zugriff auf die Ortungsdaten der Simulation.

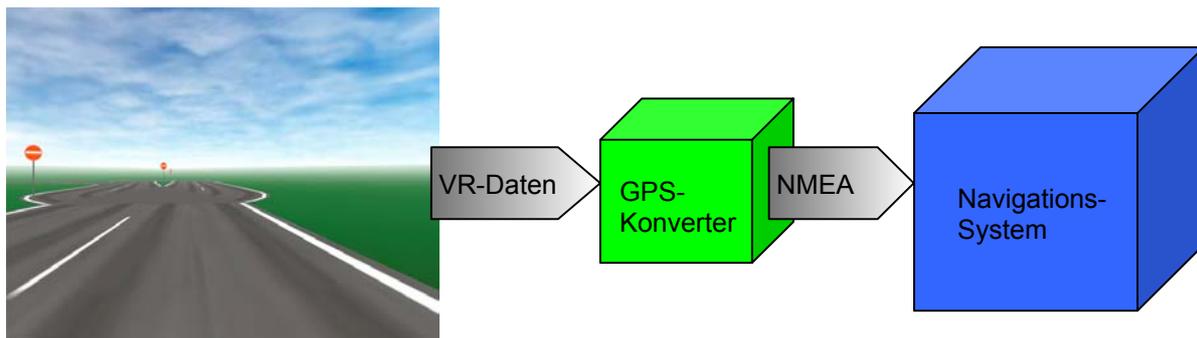


Abbildung 12: Die Positionsdaten der Simulation werden durch einen Konverter in das GPS Format gewandelt und in das Navigationssystem eingespeist. Somit stellt die Simulation eine Art GPS-Empfänger für das Navigationssystem dar.

Integration des Navigationssystems in das EMBASSI-System

Abgesehen von der Anbindung der Simulation an das Ortungsmodul war es ebenfalls notwendig, eine Verbindung zwischen dem Navigationssystem und dem EMBASSI-System zu realisieren. Beim eingesetzten Navigationssystem handelt es sich um ein separat entwickeltes Produkt von Bosch, das alle Komponenten zur Zielführung bereitstellt. In diesem Navigationskern befinden sich Module zur Routenerzeugung, Ortung, Map-Matching und zur Generierung der grafischen und akustischen Zielführungshinweise. Um auf diese Funktionen zurückgreifen zu können, war es notwendig, eine Schnittstelle zwischen dem Navigationssystem und der Java-Plattform zu schaffen. Hierzu wurde das Java Native Interface eingesetzt [JNI 2003]. Mit diesem wurden alle wesentlichen Schnittstellen des Navigationskerns in Java nachgebildet, so dass eine Nutzung der Zielführungsfunktionen durch beliebige Java-Applikationen möglich ist.

Erstellung einer adaptiven Bedienoberfläche für das Navigationssystem und Integration in das FIS

Das eingesetzte Navigationssystem stellt einen "Rechenkern" ohne Bedienoberfläche dar. Durch die Java-Anbindung ist es möglich, Fahrziele in das Navigationssystem einzuspeisen sowie Fahrhinweise vom Navigationssystem entgegenzunehmen. Auf Basis dieser Schnittstellen wurde eine Bedienoberfläche entwickelt und in das Fahrerinformationssystem integriert (siehe Abbildung 13).

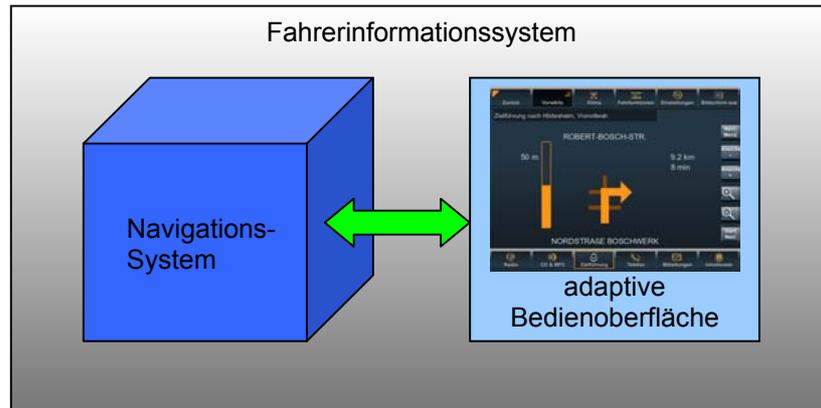


Abbildung 13: Ein bestehendes Navigationssystem wurde in das FIS eingebunden und um eine adaptive Bedienoberfläche erweitert

Hierzu wurde die in Kapitel 6.1.3.2 (unter Applikationsinterface) beschriebene Schnittstelle IApp des FIS implementiert und somit die Ein- und Ausgabe der Navigationsapplikation realisiert. Neben dieser funktionalen Einbindung in die Architektur war es zudem notwendig, alle für die Bedienung der Navigation notwendigen Dialoge nachzubilden. Hierzu zählt vor allem die Eingabe von Fahrzielen (siehe Abbildung 14). Die zugehörigen Dialoge orientieren sich in ihrer Gestaltung an der vom FIS vorgegebenen Gestaltungs- und Bedienrichtlinie und wurden mit Hilfe von Expertenevaluierungen durch die Humboldt Universität zu Berlin mehrfach optimiert (siehe Kapitel 6.5.2)

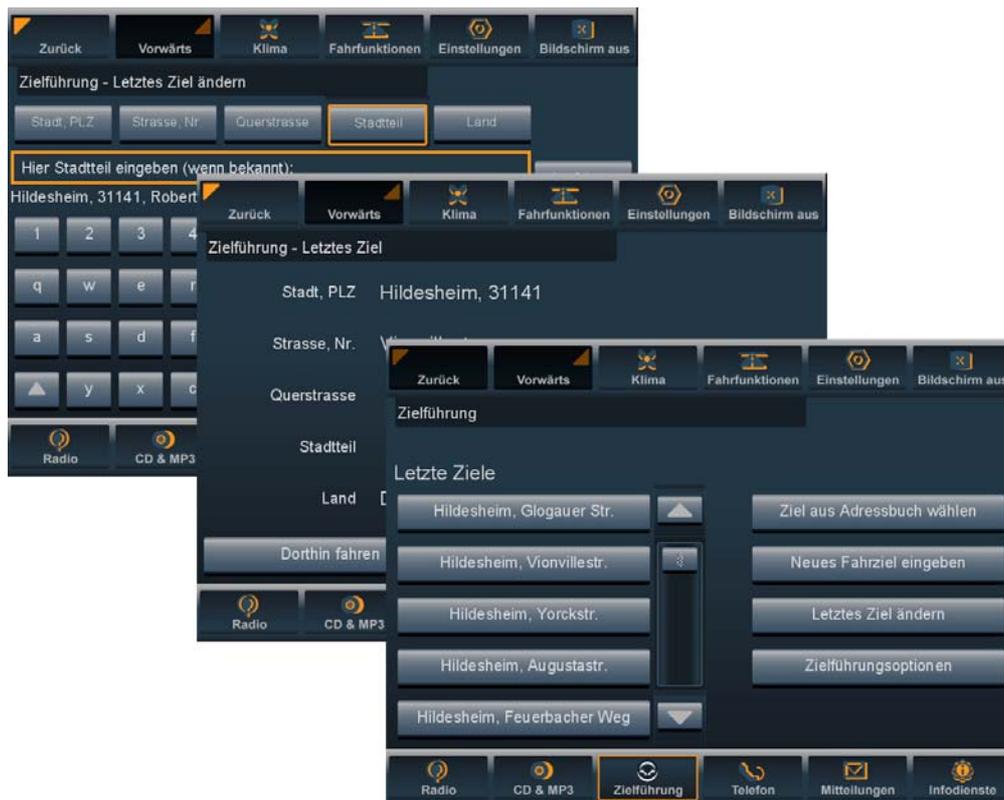


Abbildung 14: Das Navigationssystems wird über verschiedene Eingabedialoge bedient

Realisierung der adaptiven Bedienassistentz

Die Navigationsapplikation zeigt exemplarisch die Adaptivität an den Benutzer. Hierzu ist sie sehr eng in das FIS eingebunden und nutzt intensiv dessen Möglichkeiten. So wird die Navigation über zentrale Änderungen am Benutzerprofil informiert und passt etwa beim Designwechsel die grafischen Zielführungshinweise an die

aktuelle Farbgebung an. Beim Wechsel der Systemsprache werden auch die Sprachhinweise umgestellt, außerdem werden die Maßeinheiten an das Land angepasst und z.B. von Meter auf Yard umgestellt (siehe Abbildung 15).

Abgesehen von der benutzergesteuerten Adaption demonstriert die Navigation die situationspezifische Anpassung der Fahrempfehlungen. Um diese Assistenzfunktion realisieren zu können, mussten entsprechende Schnittstellen zwischen Navigationskern und Bedienoberfläche geschaffen werden. Die adaptive Ausgabe der Fahrhinweise macht es notwendig, umfangreiche Informationen an die Assistenzkomponente weiterzureichen. Diese Informationen erlauben es der Bedienoberfläche, die Visualisierung bzw. sprachlichen Fahrempfehlungen eigenverantwortlich zu generieren und damit flexibel an äußere Bedingungen anzupassen. Die ebenfalls vorhandenen Ausgabefunktionen des Navigationssystems wurden im Gegenzug nicht mehr benötigt und daher stillgelegt.

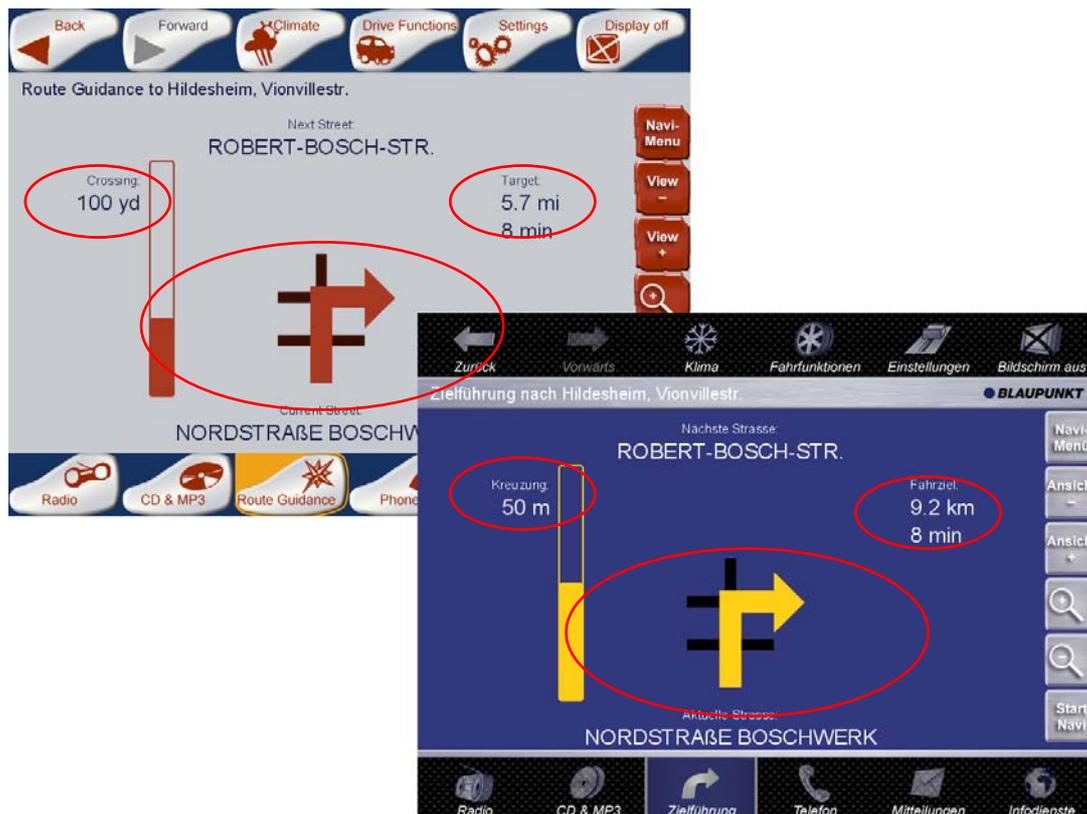


Abbildung 15: Die Navigationsapplikation adaptiert sich an Benutzervorgaben und variiert z.B. die Maßeinheiten und Pfeildarstellung je nach Ländereinstellung und Design

In Situationen, die eine hohe Aufmerksamkeit des Fahrers für die Fahraufgabe erfordern, muss die Ablenkung durch das FIS möglichst gering sein, damit sich der Fahrer verstärkt auf das Verkehrsgeschehen konzentrieren kann [Green 2000], [Färber 1998]. Deshalb war eine Anforderung an das Navigationssystem, dass die Blickabwendungszeiten infolge von Zielführungshinweisen kurz bleiben. Ein Mittel hierfür ist die Reduktion der Komplexität von Bildschirmdarstellungen in belastenden Situationen, d.h. die Form von Ausgaben wird vereinfacht. Der Inhalt bleibt aber grundsätzlich erhalten.

Um Aussagen über die Fahrsituation zu erhalten, greift die Navigationsapplikation auf eine Komponente zur Fahrsituationserkennung zurück (siehe Kapitel 6.3). Sie identifiziert anhand der Sensordaten des Fahrzeugs eine Verkehrssituation und leitet daraus eine Prognose über die zu erwartende Belastung des Fahrers ab. Diese Prognose wird von der Navigation zur Adaption genutzt und der Informationsgehalt der Fahrhinweise in Abhängigkeit der Situation gewählt.

Dazu werden alle Navigationsausgaben von speziell entwickelten Render-Komponenten generiert. Diese sind in der Lage, die semantische Beschreibung der Zielführung in grafische und akustische Meldungen umzuwandeln und anschließend im FIS auszugeben. Die Navigation setzt u.a. Renderer für die Pfeildarstellung, eine 3D-Kartendarstellung und die verbale Beschreibung ein. Diese Renderer sind in der Lage, die Ausgestaltung der Ausgabe entsprechend externer Vorgaben zu modifizieren. Durch dieses Rendererkonzept war es möglich,

verschiedene Varianten der Informationsanpassung zu testen, ohne umfangreiche Änderungen an der Navigationsapplikation vornehmen zu müssen.

Beispielsweise stellt der Pfeilrenderer den Zielführungshinweis in unbelastenden Situationen ausführlich dar, mit zunehmender Fahrerbelastung entfernt er verschiedene Informationen aus dem Bild (siehe Abbildung 16). Zu Beginn wird nur der Zielführungstext aus der grafischen Darstellung ausgeblendet. Steigt die Belastung weiter, werden alle Erklärungs- bzw. Hilfstexte unterdrückt. Zuletzt werden die Zeit- und Entfernungsangaben entfernt, so dass nur noch der Richtungspfeil als wesentliche Information zu sehen ist.

Neben dieser Informationsreduktion innerhalb eines Zielführungshinweises sind auch Varianten realisiert worden, die in Abhängigkeit der Belastung zwischen verschiedenen Darstellungsarten wechseln. Dabei wurde automatisch zwischen einer komplexen 3D-Visualisierung, einer Kartendarstellung und der Pfeildarstellung umgeschaltet. Diese harte Umschaltung ist für den Fahrer allerdings offensichtlicher als die schrittweise Veränderung einer Darstellungsart, weswegen Verständnis- und Akzeptanzprobleme beim Fahrer nicht ausgeschlossen sind.



Abbildung 16: Mit steigender Belastung des Fahrers wird schrittweise die Komplexität des Zielführungshinweises reduziert

6.2.2 Entertainment

Eine weitere Applikation, die im Rahmen der adaptiven Bedienassistentz betrachtet wurde, war Entertainment. Dabei ist hier unter Entertainment im Fahrzeug zunächst eine Musikauswahl und -abspieľfunktion zu verstehen. Diese Verallgemeinerung wird vor dem Hintergrund betrachtet, dass in einem vernetzen System im Kfz mit CD, DVD, MP3-Player, Radio, usw. die Bedienung dieser Geräte sinnvoll über ein Bedienkonzept erfolgen sollte. Die Anzahl der verfügbaren Geräte im Kfz steigt, wobei es den Benutzer meist nicht interessiert, woher die gewünschten Inhalte kommen und welche Spezifika die einzelnen Geräte haben. Wichtiger ist, dass der Benutzer Zugriff auf die Inhalte erhält. Dabei sollte er möglichst nicht von der eigentlichen Fahraufgabe abgelenkt werden. Das im Folgenden beschriebene Konzept adressiert im wesentlichen 2 Aspekte:

1. Es trägt der Entwicklung Rechnung, dass die Anzahl der Inhalte stark ansteigt und damit die Auswahl der gewünschten Inhalte für den Nutzer mit den bestehenden Konzepten im Fahrzeug kaum mehr möglich ist. Der Anstieg der Anzahl der Inhalte resultiert dabei aus den Entwicklungen im Bereich der Speicher- (DVD,

große Festplatten, usw.) und der Komprimierungstechnologien (mp3, AAC, usw.) sowie der Erhöhung des Programmumfangs durch die Digitalisierung des Rundfunks (DAB, DVB, DRM). Damit sind heute schon einige Tausend Musikstücke im Fahrzeug verfügbar. Ein Ziel war es, den Zugriff auf eine solche Datenmenge für den Fahrer zu ermöglichen und gleichzeitig die Ablenkung von der Fahraufgabe zu minimieren.

2. Bei der Entwicklung des Bedienkonzeptes sollten Assistenzfunktionen berücksichtigt werden, die den Fahrer bei der Auswahl der Inhalte unterstützen (im Sinne von [Timpe 1998]) und, wenn sinnvoll, adaptiv durch den Fahrerzustand aktiviert werden können.

Das Konzept wurde für mp3-Musikdaten erstellt, wobei es auch z.B. auf Radioprogramme übertragen werden kann. Die Grundlage zur Berücksichtigung der oben genannten Aspekte bildet die Verwendung von Benutzerpräferenzen und weitergehender Information zu den Inhalten, die in Form von standardisierten Metadaten verwendet wurde.

Bedienkonzept unter Verwendung von Metadaten und Benutzerpräferenzen

Anders als bei der Navigations-Applikation musste für die Entertainment Applikation ein komplett neues Bedienkonzept erstellt werden, welches den schnellen Zugriff auf Daten erlaubt, da diese Problematik bei verfügbaren Serienfahrzeugen bisher nicht betrachtet wurde.

Das Erarbeitete Konzept soll nun kurz dargestellt werden:

Um den Zugriff auf die großen Datenmengen zu ermöglichen, wurde zunächst eine Sortierung oder Kategorisierung vorgenommen. Dabei bieten sich Metadaten an. Die zusätzliche Information über die Inhalte wird genutzt, um Kategorien zu bilden. Diese Kategorien sollten dem üblichen Nutzungsmodell des Anwenders entsprechen, d.h. in diesem Fall: Üblicherweise werden heute Musikstücke als "Paket" in Alben verkauft, sie klassifizieren sich weiterhin im wesentlichen über ihren Interpreten oder gehören einem bestimmten Genre an.

Im vorliegenden Konzept wurde das Genre, der Künstler-Name, der Album-Name und der Lied-Name als Kategorie verwendet.

Weiterhin vereinfacht die Auswahl, dass die Kategorien Abhängigkeiten besitzen und eine sehr unterschiedliche Anzahl von Elementen. Die Abhängigkeiten ergeben sich hierbei dadurch, dass jeder Musiktitel einen Namen hat, einem Album angehört, eindeutig einem Künstler zugeordnet werden kann und einem Genre angehört. Es ist auch offensichtlich, dass im Normalfall (bei üblichem Hörgewohnheiten) relativ wenige unterschiedliche Genre auftreten werden und die Anzahl der Lieder größer ist als die der Alben und die ist wiederum höher als die Anzahl der Interpreten. Somit kann beginnend über die Kategorie mit den wenigsten Elementen eine Auswahl erfolgen, ohne dass man durch Tausende von Einträgen scrollen muss.

Die Metadaten bzw. Kategorien wurden in diesem Fall so gewählt, dass das System bereits heute über die ID3-Tags der MP3-Dateien genutzt werden kann. Zukünftig kann man sich mit Hilfe von Metadaten, die z.B. über den digitalen Rundfunk ausgestrahlt oder zu Hause beliebig vergeben und in das Fahrzeug übertragen werden können, unzählige Varianten vorstellen.

Da die Vorlieben für Musik sich individuell sehr stark unterscheiden, liegt es nahe, ein Benutzerprofil zu berücksichtigen. Dieses Benutzerprofil wurde konzeptionell so aufgebaut, dass es mit dem Home-Bereich austauschbar ist, d.h. die Beschreibung erfolgt in XML und die verwendeten Metadaten sind Bestandteil der Ontologie. Die Herausforderung bestand darin, eine geeignete Form des Profils zu entwickeln, die sich auf die Kategorisierung abbilden lässt. Zusätzlich sollte sich das zeitlich geänderte Hörverhalten im Profil widerspiegeln. Dabei wird die Systemänderung auf Initiative des Systems (Auto-Adaptivität) [Eberleh et al. 1994] durchgeführt. Bei der Auto-Adaptivität beobachtet ein System die Aktionen des Nutzers oder folgt ihnen und verändert sein Verhalten hierzu passend.

Die Abbildung des Profils auf die Kategorien erfolgt so, dass prinzipiell für jede Kategorie eine Liste erstellt wird, in der die einzelnen Elemente nach einer Priorität sortiert sind. Darüber hinaus werden die Hörgewohnheiten des Benutzers aufgezeichnet und automatisch bewertet. Dieses wird z.B. dadurch erreicht, dass bei jeder kompletten Auswahl bis zum Musikstück, die bis dahin gewählten Elemente in den Listen aufgewertet werden. Das bedeutet, dass nach kurzer Zeit die bevorzugten Kategorie-Elemente bzw. Genre, Künstler Album und Songs in der Liste mit hoher Priorität versehen sind und somit für den Benutzer leichter erreichbar sind, weil diese Elemente bei Anwahl einer Kategorie zuerst angezeigt werden.

Die bewerteten Listen werden ebenso bei unterschiedlichen Assistenzfunktionen berücksichtigt.

Erstellung einer adaptiven Bedienoberfläche

Hierzu wurde die in Kapitel 6.1.3.2 (unter Applikationsinterface) beschriebene Schnittstelle IApp des FIS implementiert und somit die Ein- und Ausgabe der Entertainment-Applikation realisiert. Neben dieser funktionalen Einbindung in die Architektur war es zudem notwendig, alle für die Bedienung der Applikation notwendigen Dialoge zu entwickeln. Die zugehörigen Dialoge orientieren sich in ihrer Gestaltung an der vom FIS vorgegebenen Gestaltungs- und Bedienrichtlinie und wurden mit Hilfe von Expertenevaluierungen durch die Humboldt Universität zu Berlin mehrfach optimiert (siehe Kapitel 6.5.2).

Das Konzept der Kategorisierung und der dazugehörigen Listen ist in Abbildung 17 dargestellt.

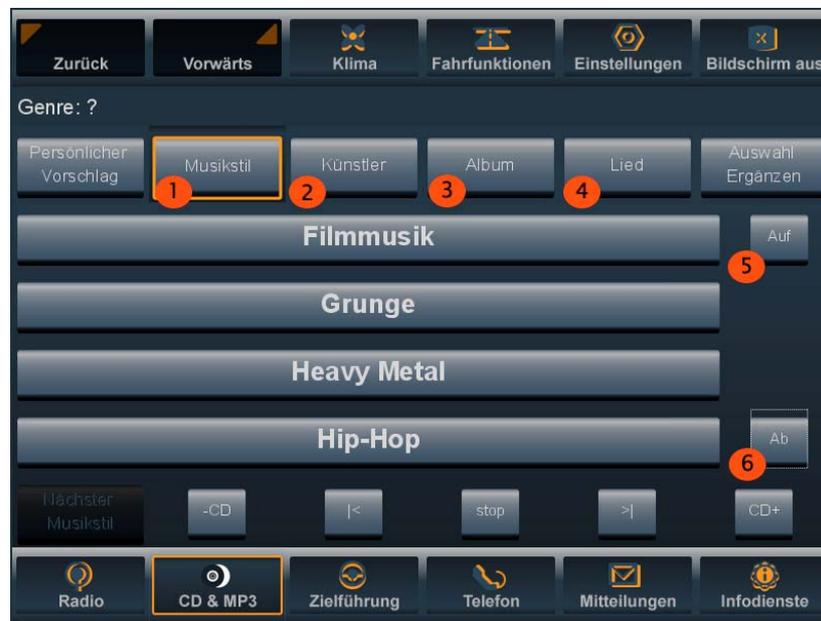


Abbildung 17: Bedienoberfläche der Entertainment-Applikation

Drückt der Benutzer auf einen der Buttons (1-4) so erscheinen die ersten 4 Elemente der Kategorie. Durch Druck auf die Buttons "auf" (5) und "ab" (6) kann die Liste durchlaufen werden. Es werden bewusst nur jeweils 4 Elemente dargestellt, um den Informationsgehalt zu reduzieren.

Durch die Auswahl über die verschiedenen Kategorien kann deutlich einfacher auf die Inhalte zugegriffen werden, da man sich dem gewünschten Musikstück sozusagen schrittweise annähert. In der Praxis kann der Benutzer von der generellen (Musikstil z.B.: Rock) zur spezifischeren Kategorie (Album z.B.: ACDC) bis letztendlich zum speziellen Musikstück (z.B.: Come And Get It) auswählen. Dieser Zugriff ist deutlich effektiver, als das Heraussuchen eines Songs aus einer einzigen langen Liste. Dieser Ablauf wird durch die personalisierten Listen noch wesentlich unterstützt.

Zudem ist dem Benutzer freigestellt mit welcher Kategorie er die Auswahl beginnt. Die bereits ausgewählten Kategorie Elemente werden dem Benutzer in einer extra Statuszeile angezeigt, um die Orientierung zu vereinfachen oder bei einer Unterbrechung die Wiederaufnahme der Bedienung wesentlich zu beschleunigen (siehe auch Kapitel 6.1.3.1).

Eine Besonderheit war bei der Ansteuerung der Bedienoberfläche noch zu berücksichtigen, da die Entertainment-Applikation ebenfalls über Sprachbedienung gesteuert werden konnte. Diese multimodale Ansteuerung konnte über die komponentenbasierte Architektur sehr gut nachträglich integriert werden, da alle Steuerungsdaten abstrahiert werden. Die zusätzliche Verwendung des Model-View-Control-Ansatzes [GoF 1994] bei der Realisierung erlaubt eine einfache Abstimmung der beiden "Views" (Sprache und Haptik), da beide auf einem Model basieren.

Realisierung adaptive Bedienassistentz

Zunächst wurden Szenarien erarbeitet, die aufzeigen, welche Möglichkeiten der Unterstützung bei dieser Anwendung denkbar sind. Dabei sollte ein hoher und ein mittlerer Unterstützungsgrad berücksichtigt werden. Für beide Assistenzgrade wurden Funktionen ausgewählt, die zunächst direkt vom Benutzer angewählt werden können, indem sie die Assistenz per Knopfdruck aktivieren können (siehe Abbildung 18 und Abbildung 19).

Für das Konzept der Entertainment-Bedienung wurde als mittlere Unterstützung die Verkürzung des Auswahlprozesses ausgewählt, da es durch die Kategorisierung möglich ist, dass man zwar einen konkreten

Künstler auswählen will, aber nicht an einem konkreten Musikstück von diesem Künstler interessiert ist. In diesem Fall wählt die Entertainment Applikation automatisch, die bis dahin noch nicht gewählten Kategorien aus. In der Praxis könnte der Benutzer z.B.: Den Musikstil „Rock“ und die Band „ACDC“ wählen und dann die automatische Ergänzung anwählen, indem er den Button (7) „Auswahl ergänzen“ (siehe Abbildung 18) drückt. Es wird dann automatisch das höchst priorisierte Musikstück aus dem bevorzugten Album gespielt.

Die Taste (8) "nächstes Album" (siehe Abbildung 19) ist immer mit der Kategorie belegt, ab der die Automatik angewählt wurde. Das wird auch über die Beschriftung deutlich, die jeweils angepasst wird.



Abbildung 18: Mittlerer Assistenzgrad

Die hohe Unterstützung erfordert nur einen Knopfdruck zur Musikauswahl. Das unterstützt besonders die Fahrer, die einfach nur Musik hören wollen, aber keine konkreten Vorstellungen dazu haben. Sie möchten aber dennoch Musikstücke, die ihnen prinzipiell gefallen. Wählt der Benutzer diesen Button (9) (siehe Abbildung 19), so erhält er einen Musik-Vorschlag ohne Vorauswahl. In diesem Fall besteht ebenfalls die Möglichkeit über den Button (8) in eine neue Auswahl anzustarten.

Die Entertainment- Applikation bewertet darüber hinaus die "Einstiegs-kategorie", mit welcher der Benutzer am häufigsten seine Auswahl beginnt. Die Statistik, die damit erstellt wird, verwendet diese Funktionalität, um in der Kategorie bzw. der ihr zugeordneten Liste mit der Auswahl zu beginnen, die der Benutzer am häufigsten beim Beginn seiner Auswahl verwendet. Dies geschieht um den Hörgewohnheiten des Benutzers möglichst nahe zu kommen. Steigt der Benutzer z.B.: am häufigsten über die Genre ein, so wechselt er wahrscheinlich oft den Musikstil. In diesem Fall bietet die Entertainment-Applikation öfter Musikstücke aus verschiedenen Musikstilen an.



Abbildung 19: Hoher Assistenzgrad

Die Aktivierung dieser Assistenzgrade durch den Belastungsgrad wurde als nicht sinnvoll bewertet, da zum einen der dann folgende automatisierte Ablauf für den Benutzer nicht transparent ist und zum anderen eine Bevormundung des Nutzers erfolgt, da dieser zunächst den Vorgang der Auswahl initiieren muss. Es ist generell fragwürdig, ob ein Eingabevorgang des Benutzers adaptiv verändert werden sollte. Zu diesem Themenbereich sind weitere Untersuchungen notwendig.

Sprachbedienung

Alle beschriebene Funktionalitäten der Entertainment Applikation lassen sich im EMBASSI Szenario auch per Sprache bedienen. Darüber hinaus kann per Sprachbedienung, ohne den Umweg über die Listen, direkt ein Listen- Element ausgewählt, werden (z.B.: Ich möchte Nena hören.). Dies funktioniert auch in Kombination mit der „Auswahl ergänzen“ Funktionalität (z.B.: Ich möchte irgendwas von Nena hören).

An dieser Stelle sei darauf hingewiesen, dass eine Sprachbedienung der dargestellten Komponente derzeit in einem Produkt noch nicht sinnvoll ist, da der Stand der Technik und auch die Entwicklungen in EMBASSI im Bereich der Spracherkennung noch Defizite bei der Mehrsprachigkeit und durch den großen Wortschatz bei der Robustheit aufweisen.

6.2.3 Fahrer-Warn- und Assistenzsystem

Aufgabe dieses Arbeitspakets war die Entwicklung eines Störungswarnassistenten, der Autofahrer auf unerwartete Situationen auf der Strasse hinweist. Ziel dieses Assistenten ist die Erhöhung der Fahrsicherheit. Dazu wird der Fahrer vor einem Stauende oder einer Störungsstelle informiert bzw. gewarnt und Unfälle verhindert. Dabei setzt das System unterschiedliche visuelle und auditive Warnstrategien ein, die durch die Wünsche des Fahrers und die derzeitige Fahrsituation beeinflusst werden [Pffaffert 1992].

Es wurde ein Assistenzsystem realisiert, das die genannten Anforderungen umsetzt. Der Prototyp wurde nicht in ein Kfz integriert, sondern ist auf die Nutzung einer Fahrsimulation ausgerichtet. Die Fahrsimulation stellt eine virtuelle Umgebung mit einer Reihe von Störungen dar, vor denen der Störungswarnassistent warnt. Der Fahrsimulator stellt dem Assistenzsystem dazu über eine Netzwerkverbindung die Betriebsdaten des Fahrzeugs, u.a. Geschwindigkeit, Position und Richtung zur Verfügung. Außerdem liefert er Hinweise zur Fahrsituation, etwa Nebel. Diese Informationen werden genutzt, um die Warnstrategie zu beeinflussen. Im Nebel unterscheiden sich die Warnstrategien von den üblichen, um die geringere Sichtweite des Fahrers zu berücksichtigen.

Neben den Fahrzeugbetriebsdaten besitzt der Störungswarnassistent Informationen über die Position und Ausdehnung von Störungen. Beides wird kombiniert, um die Entfernung des Fahrzeugs zu einer potentiellen Gefahrenstelle zu ermitteln. Stellt der Störungswarnassistent fest, dass sich das Fahrzeug in der Nähe einer Störung befindet, so verfolgt es die Warnstrategie für den eingestellten Assistenzgrad. Diese Warnstrategie wird beibehalten, bis das Fahrzeug die Störungsstelle passiert hat. Zur Umsetzung der Warnung kann das System akustische Ausgaben nutzen und auf 2 Bildschirme zurückgreifen, die das Kombiinstrument bzw. den Monitor in der Mittelkonsole eines Fahrzeugs repräsentieren (siehe Abbildung 20).



Abbildung 20: Das Assistenzsystem warnt den Fahrer vor Störungen auf der Strasse

Die gewünschte Warnstrategie wird vom Fahrer über den Assistenzgrad eingestellt. Folgende Assistenzgrade stehen dabei zur Auswahl:

- keine Assistenz,
- Information,
- Information mit Handlungsanweisung,
- Information mit Handlungsvorschlag und Ausführung,
- Information und automatischer Eingriff in das Fahrzeug

Die Assistenzstufen und deren Funktionsweise können frei mit Hilfe von Regeln beschrieben werden. Sie werden von einem zu diesem Zweck entwickelten Interpreter ausgewertet, der die definierten Strategien umsetzt. Diese Regeln erlauben es, getrennt für jede Störungsart festzulegen, wie sich das Warnsystem verhalten soll. Dazu können für Kombinationen von Abständen zur Störung und Geschwindigkeiten festgelegte Reaktionen ausgelöst werden. Hierzu zählen die Ausgabe von Texten auf dem Bildschirm der Mittelkonsole oder über synthetische Sprache. Außerdem können Geräusche abgespielt, Symbole im Kombiinstrument und auf dem Bildschirm der Mittelkonsole eingeblendet werden. Wird ein automatischer Eingriff in das Fahrzeug gewünscht, so kann festgelegt werden, wie stark das Fahrzeug gebremst werden soll. Die Bremswirkung kann sich dabei am verbleibenden Abstand zur Störung und der Fahrzeuggeschwindigkeit orientieren. Falls es gewünscht wird, kann das Fahrzeug dadurch punktgenau vor einer Störung gebremst werden.

Kritisch für die Wirksamkeit der Störungswarnassistenten ist die Detektion einer Störung. Das System empfängt eine Liste von Störungen und deren Position. Daraus errechnet es über den Abstand zum Fahrzeug, welche Störungen relevant sind.

Die Bestimmung der Entfernung zu Störungsstellen kann allerdings nicht durch einfache Abstandskriterien wie die Luftlinie bestimmt werden, da sonst in kurvigen Strecken Falschinformationen erfolgen. Außerdem muss die Straßenseite, auf der die Störung liegt, berücksichtigt werden, damit nur Störungen in Fahrtrichtung gemeldet werden.

Um diese Anforderungen realisieren zu können, besitzt der Störungsassistent eine Datenbank, in der die digitale Karte der Umgebung gespeichert ist. Vergleichbar zu einem Navigationssystem projiziert der Störungsassistent die Störungen und das Fahrzeug auf diese Karte. Da die Straßen in der Karte keine seitliche Ausdehnung besitzen, wird ein Map-Matching durchgeführt. Dieses gewährleistet, dass für Positionen, die nicht direkt auf dem Straßennetz liegen, ein zugehöriges Straßennetzstück gefunden wird.

Nachdem die Position des Fahrzeugs und der Störungen dem Straßennetz überlagert wurden, wird im nächsten Schritt die Entfernung bestimmt. Dabei werden nur die Störungen berücksichtigt, die sich auf der richtigen Straßenseite befinden. Von diesen Störungen ausgehend wird das Straßennetz rückwärts in Richtung Fahrzeug verfolgt und die Länge des Weges ermittelt. Dieses Verfahren garantiert, dass die Meldung von Entfernungen korrekt ist, etwa „Unfall in 2000 m“.

Bei der Rückverfolgung des Weges wird ein Abbruchkriterium genutzt, damit nur die Störungen berücksichtigt werden, die sich in der Nähe des Fahrzeugs befinden. Die anderen zu weit entfernten werden nicht weiterverfolgt. Dadurch wird eine hohe Performance des Systems sichergestellt.

6.3 Fahrsituationserkennung und Ableitung der Fahrerbelastung

Adaptive Systeme sind in der Lage, sich selbsttätig an den Unterstützungsbedarf des Nutzers, der in Abhängigkeit verschiedener Faktoren variieren kann, anzupassen. Bisher verhalten sich Fahrzeuge jedoch wenig adaptiv in Bezug auf den Zustand des Fahrers. Dies liegt unter anderem daran, dass die Messung des physischen und mentalen Fahrerzustands technisch sehr aufwendig bzw. nicht verlässlich realisierbar ist (siehe auch [Hering 1999], [Healey & Picard 2000]). Unter Einschränkungen ist es jedoch möglich, diese aufwendigen Messungen zu umgehen. Hierbei wird die Annahme getroffen, dass durch die Bewertung der gegenwärtigen Fahrsituation Aussagen zur Belastung des Fahrers möglich sind. Die Fahrsituation wird durch zahlreiche Faktoren geprägt, etwa Umweltbedingungen wie Witterungs- und Lichtverhältnisse, Verkehrsdichte, Straßenzustand, Art des Verkehrswegs usw..

Anhand der im Fahrzeug verfügbaren Sensoren können Rückschlüsse auf die Fahrsituation gezogen werden [Sonnerat 2002].

Aufgabe der für diesen Zweck entwickelten Fahrsituationserkennung ist, die aktuelle Fahrsituation ausgehend von Sensor- und Fahrzeugbetriebsdaten zu klassifizieren (z.B. Nebel vs. klare Sicht). Im Fahrzeug sind z.B. Sensordaten für die Geschwindigkeit, Beleuchtung, Außentemperatur, Scheibenwischerstufe und ABS verfügbar.

Daneben wurden auch Informationen über die Längs- und Querverführung des Fahrzeugs berücksichtigt. Diese resultieren aus der Betätigung von Lenkrad und Pedalarie des Kfz und vermitteln ein Bild über die Interaktion des Fahrers mit dem Fahrzeug. Diese Daten werden auf verschiedene Kriterien hin analysiert. Zum einen ist dies die Standardabweichung der Bewegungsintensität, d.h. mit welcher Stärke bewegt der Fahrer Lenkrad oder

Pedale. Zum anderen wird die Frequenz der Richtungswechsel dazu verwendet, Aussagen über Korrekturbewegungen der Spurlage zu erhalten. Beide zusammen sind Indikatoren dafür, wie viel Aufwendung der Fahrer in die Fahraufgabe investiert [deWaard 1996].

Aus den gemessenen Daten können Informationen zur Fahrsituation abgeleitet werden. So weist eine hohe Wischerstufe auf starken Regen bzw. schlechte Sichtverhältnisse hin, eingeschaltetes Licht deutet auf Dunkelheit oder auch schlechte Sichtverhältnisse hin. Das eingesetzte Verfahren lehnt sich an die Situationserkennung von XFCD (eXtended Floating Car Data) [Lädke & Huber 1998] an und erfasst z.B. Regen, Schnee oder Nebel.

Aufgrund der aktuellen Fahrsituation sind Rückschlüsse auf den Fahrerzustand möglich. Hierzu wird die Situation durch eine Komponente zur Fahrerzustandserkennung bewertet (siehe Abbildung 21). Beispielsweise führen eine hohe Verkehrsdichte verbunden mit schlechtem Wetter zu einer hohen Belastung des Fahrers.

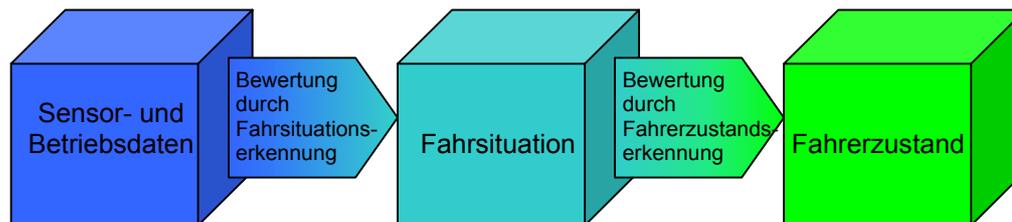


Abbildung 21: Der Fahrerzustand wird in einem mehrstufigen Prozess aus Sensordaten abgeleitet

Die Bewertung wurde mit Hilfe eines Interpreters realisiert, der in Echtzeit Bewertungstabellen mit den Sensordaten verknüpfen kann. Die Bewertungstabellen werden mit separater Software wie Excel bearbeitet und können jederzeit zur Laufzeit des Systems ausgetauscht werden (siehe Abbildung 22). Diese Trennung hat den Vorteil, dass Verkehrsexperten die Situationsbewertung unabhängig von der eigentlichen Fahrerzustandskomponente mit ihnen vertrauten Hilfsmitteln durchführen können. Durch die Möglichkeit zum dynamischen Neuladen von Tabellen können die Bewertungsregeln im Rahmen von Untersuchungen flexibel optimiert bzw. auch vollständig ausgetauscht werden.

	A	B	C	D	E	F
1	ASR-Sensor	Nebelscheinwerfer	Nebelschlußleuchte	Wischfrequenz	Regen und Nebel in %	
2	system.kb.item	system.kb.item.fog	system.kb.item.rear	system.kb.item.v	system.kb.item.fog_plus_rain	
3				system.kb.item.w	100	
4					0,00	
5	25,00				25,00	
6		50,00			50,00	
7	25,00	30,00			55,00	
8			65,00		65,00	
9	25,00		45,00		70,00	
10		30,00	35,00		65,00	
11	15,00	25,00	30,00		70,00	
12				80,00	80,00	
13	15,00			65,00	80,00	
14		50,00		35,00	85,00	
15	15,00	35,00		35,00	85,00	
16			30,00	50,00	80,00	
17	15,00		25,00	40,00	80,00	
18		20,00	20,00	45,00	85,00	
19	15,00	20,00	20,00	30,00	85,00	

Abbildung 22: Situationen werden in Form von Tabellen bewertet, die mit separaten Programmen (z.B. Excel) bearbeitet werden können. Geänderte Bewertungstabellen können dynamisch zur Laufzeit in die Erkennungskomponente geladen werden.

Die Reaktion der Fahrer auf Belastungen, d.h. die Beanspruchung, ist jedoch sehr individuell. Allerdings existieren gruppenspezifische Tendenzen. So werden ältere Menschen durch schlechte Sichtverhältnisse in der Regel stärker beansprucht als junge. Die Schätzung der Beanspruchung wurde letztlich aber nicht realisiert, da

hierfür kein ausreichend sicheres Modell zur Verfügung stand. Somit wurde nur die objektiv zu wertende Belastung aufgrund einer Situation berücksichtigt.

Die von der Fahrerzustandserkennung ermittelten Daten zur Fahrerbelastung werden den Applikationen im Fahrerinformationssystem zur Verfügung gestellt. Sie nutzen diese beispielsweise zur automatischen Assistenzgradumschaltung oder zur Anpassung an die Situation. So reduziert das Navigationssystem in belastenden Situationen die Komplexität ihrer Ausgabe, damit die Blickabwendungszeiten von der Strasse möglichst kurz sind.

Da keine Integration des Kfz-Demonstrators in ein Fahrzeug erfolgte, wurde zur Präsentation und Evaluierung der entwickelten Fahrsituations- und Fahrerzustandserkennung eine Sensorplatte aufgebaut. Diese erlaubt es, durch Taster die Sensorenzustände eines virtuellen Fahrzeugs zu simulieren. Beispielsweise können durch diese Sensorplatte Geschwindigkeit, Frequenz der Scheibenwischer oder etwa die Fahrzeugbeleuchtung beeinflusst werden (siehe Abbildung 23). Diese Sensordaten werden von der Fahrsituations- und Fahrerzustandserkennung unmittelbar berücksichtigt und eine Aussage zur erwartenden Belastung des Fahrers getroffen.



Abbildung 23: Mit einer Sensorplatte können die Betriebsdaten des Fahrzeugs simuliert werden

Neben dieser Sensorplatte wurde im Hinblick auf Untersuchungen ein Demonstrator in Form einer Cockpitstudie aufgesetzt. Dieser bildet einen Teil der im realen Fahrzeug vorhandenen Schalter nach, z.B. für die komplette Lichtanlage und Scheibenwischer. Diese Schalter wurden ebenfalls an die Erkennungsalgorithmen angebunden und erlauben im Zusammenspiel mit Fahrversuchen Rückschlüsse auf die Fahrerbelastung während der Fahrt.

Zur anschaulichen Demonstration der Funktionsweise von Fahrsituations- und Fahrerzustandserkennung wurde eine separate Komponente entwickelt und in das EMBASSI-System integriert. Diese visualisiert die Sensorinformationen und die durch Erkennungsalgorithmen ermittelten Daten (siehe Abbildung 24).

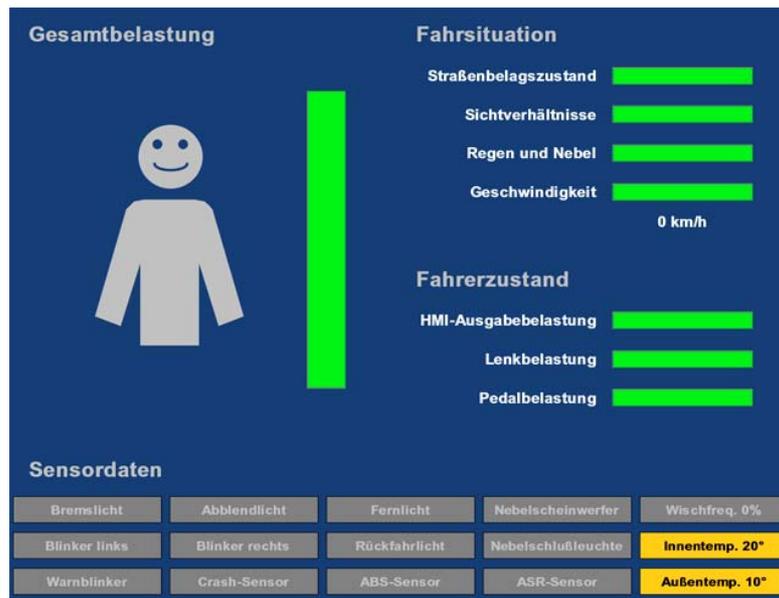


Abbildung 24: Die grafische Darstellung der Fahrerzustandsberechnung

6.4 Demonstrator

6.4.1 Demonstrationsträger (Sitzkisten)

Der Demonstrator, bestehend aus Hardware und Software, stellt eine Art Referenzimplementierung des definierten und spezifizierten Systems dar. Der Demonstrator hat folgende Aufgaben:

- Test der Architektur und des Systemverhaltens,
- Erprobung des Systemverhaltens (Evaluierung),
- Beweis der technischen Lösbarkeit der Aufgabe,
- Demonstration der Projekt-Konzepte vor Kunden.

An dieser Stelle wird näher auf den Demonstrationsträger eingegangen, der im Kfz-Umfeld auch Sitzkiste oder Cockpit-Studie genannt wird. Das Ziel bestand darin, eine modulare Cockpit-Studie für Fahrzeuge (PKW) zu konzipieren und zu gestalten, mit der die Schnittstelle zwischen dem Menschen und den im Fahrzeug integrierten Geräten präsentiert werden soll. Die besonderen ergonomischen Aspekte im Fahrzeug müssen dabei berücksichtigt werden. Die Cockpit-Studie soll modular aufgebaut sein, um den Transport zu verschiedenen Präsentationsorten zu erleichtern. Es ergeben sich somit die folgenden Randbedingungen für den Bau der Cockpit-Studie:

- Modularer und zerlegbarer Aufbau, leichte Bauweise und "ansprechendes, futuristisches Design".
- Zwei Sitzmöglichkeiten für Fahrer und Beifahrer mit Umsetzung des Fahrverhaltens.
- Lenkrad, Gas- und Bremspedal (Automatik) zur Durchführung der Fahraufgabe.
- Integration einer kompletten Lichtanlage mit den dazugehörigen Bedienelementen.
- Anbringung von zwei Außenspiegeln (Realisierung durch 6,5 " Displays).
- Halterung für den Beamer zur Projektion der Fahrsimulation (min. Distanz zur Leinwand ca. 1 Meter).
- Vorrichtung zur Unterbringung portabler Endgeräte wie z.B. PDAs oder Handys.
- Display für Kombiinstrument im Format 16:9 zur Anzeige von Geschw., Temp., Tankzustand usw.
- Display mit Touchscreen für Fahrerinformationssystem (10,4 ").
- Eingabelemente zur Bedienung von Grundfunktionen wie z.B. Lautstärke, Klima usw.
- Einbau von vier Lautsprechern zur räumlichen Soundgenerierung.
- Berücksichtigung von Kabelkanälen (auch für spätere Systemerweiterungen).



Abbildung 25: Ansicht der gesamten Sitzkiste mit Leinwand für Fahrsimulation



Abbildung 26: Ansicht des Cockpits mit Lenkrad, Kombiinstrument und Display in Mittelkonsole

In Abbildung 25 und Abbildung 26 ist die realisierte Sitzkiste zu sehen, die den aufgestellten Randbedingungen genügt. Dabei ist der Touchscreen in der Mittelkonsole zur Bedienung des Fahrerinformationssystem gut zu erkennen. Die Tasten am Lenkrad dienen während der Versuchsreihen als Eingabemöglichkeit für Nebenaufgaben. Außerhalb der Versuchsreihen konnten damit Systemfunktionen bedient werden (z.B. Lautstärke).

Der Demonstrator wurde auf der Statustagung "Mensch-Technik-Interaktion" in Berlin (03. - 04. Juni 2003) gezeigt, der durch das Bundesministerium für Bildung und Forschung folgendermaßen angekündigt wurde: „Seit 1999 werden von BMBF und BMWA sechs Leitprojekte zum Themenfeld "Mensch-Technik-Interaktion" gefördert. Im Rahmen der abschließenden Statustagung am 03.-04. Juni 2003 im Berliner Congress Center

werden der internationalen Fachwelt die wesentlichen Projektergebnisse vorgestellt. Präsentationen erfolgen durch Vorträge, Poster, computergestützte Demonstrationen, Prototypen und weitere Exponate.“

In

Abbildung 27, Abbildung 28 und Abbildung 29 sind einige Bilder von der Abschlusspräsentation des Leitprojektes EMBASSI zu sehen.



Abbildung 27: Präsentationsaufbau mit Darstellung des Fahrerzustandes und Sensortafel



Abbildung 28: Präsentationsaufbau mit Sitzkiste auf MTI-Statustagung in Berlin



Abbildung 29: Präsentation der Projektergebnisse für Prof. W. Wahlster vom Leitprojekt SmartKom

6.4.2 Integration der externen Beiträge zum Kfz-Demonstrator

6.4.2.1 Integration Fahrsimulation

Die verschiedenen Telematikkomponenten und Assistenzsysteme des Kfz-Demonstrators erfordern es, dass ein Benutzer mit einem Kfz fährt. Nur durch die damit verbundene Dynamik kann ein realistischer Eindruck der Funktionsweise vermittelt werden. Da jedoch eine Integration des Demonstrators in ein Kfz zu aufwendig und schlecht präsentierbar gewesen wäre, wurde auf eine Fahrsimulation zurückgegriffen. Diese wurde vom Projektpartner ZGDV unter Berücksichtigung der speziellen Anforderungen des EMBASSI Kfz-Demonstrators entwickelt.

Die erste Version des Fahrsimulators stellte einen Rundkurs beliebiger Form dar. Dieser wurde manuell in einem Vorverarbeitungsschritt konstruiert. Daneben enthielt er eine Verkehrssimulation mit verschiedenen computergesteuerten Fahrzeugen. Diese besitzen ein eigenständiges Fahrverhalten und können z.B. andere Fahrzeuge überholen oder vor Gefahrenstellen anhalten (siehe Abbildung 30). Der Fahrer kann mit Hilfe von Lenkrad und Pedalarie die Geschwindigkeit und Spurlage regeln. Alle Fahrzeugbetriebsdaten werden permanent über eine Netzwerkschnittstelle an eine Gegenstelle übertragen. Dazu zählen die Geschwindigkeit, Spurlage, Fahrzeugposition, Abstände zu anderen Fahrzeugen oder die Wettersituation.

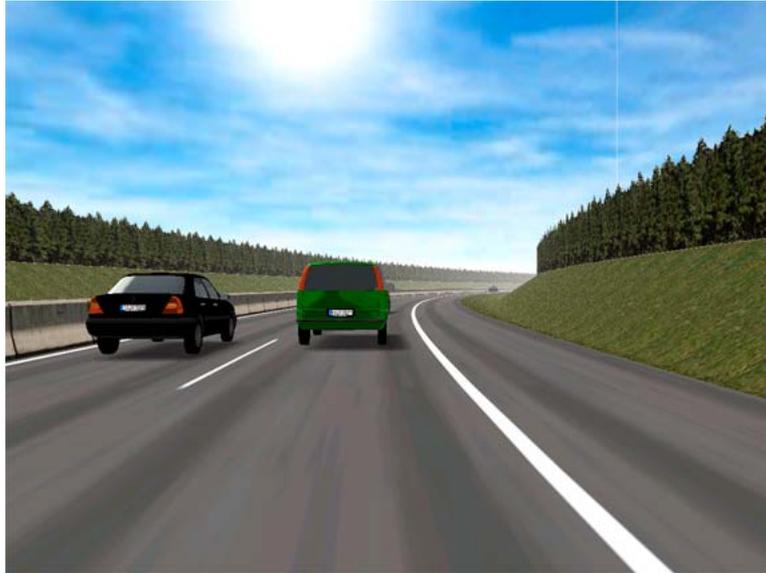


Abbildung 30: Die erste Version der Fahrsimulation mit computergesteuerten Fahrzeugen

Im Laufe des Projektes änderten sich die Anforderungen an die Fahrsimulation, so dass im Rahmen der zweiten Projektphase eine weitreichende Umgestaltung notwendig wurde. Hauptgrund hierfür war die Nutzung eines Navigationssystems. Dieses kann Zielführungshinweise nur für reale Strecken generieren. Für künstlich erzeugte Parcours wie dem der ersten Fahrsimulation konnte sie keine Zielführung gewährleisten, da sie keine entsprechenden Kartendaten besitzt. Außerdem erforderte die Zielführung eine Fahrstrecke mit Kreuzungen und der Möglichkeit zur freien Streckenwahl. Beides konnte die Simulation mit ihrem Rundkurs nicht bieten.

Der neue Fahrsimulator stellt die simulierte Welt mit Hilfe realer, digitalisierter Kartendaten dar. Die dazu verwendeten Kartendaten werden in anderer Form auch vom Navigationssystem eingesetzt. Die Änderungen an der Fahrsimulation betrafen deshalb vor allem das der Straßenbeschreibung zugrunde liegende Datenformat und die damit verbundene Generierung der Fahrscene. In der alten Version wurde die virtuelle Welt komplett in einem Offline-Prozess generiert und während des Fahrens unter Berücksichtigung der Fahrzeugposition dargestellt. Das bedeutet, dass beispielsweise Form und Breite der Straße sowie die zugehörige Texturierung bereits vor Fahrtantritt festgelegt waren. In der neuen Version ist dies jedoch nicht mehr möglich. Die Kartendaten können sehr große Gebiete umfassen, etwa ganz Deutschland.

Aus Speicherplatz- und Geschwindigkeitsgründen können umfangreiche Gebiete nicht während eines Vorverarbeitungsschritts zur Darstellung vorbereitet werden. Daher wird während der Fahrt mit dem Simulator nur ein kleiner Teil der Kartendaten in der Umgebung des Fahrzeugs berücksichtigt und in Echtzeit eine Landschaft mit den Fahrspuren konstruiert. Hierzu wurde zwischen dem ZGDV und Bosch eine Schnittstelle definiert, die dem Fahrsimulator Zugriff auf eine digitale Karte von Bosch erlaubt. Diese Schnittstelle musste dabei performance-kritische Gesichtspunkte berücksichtigen, um die verzögerungsfreie Echtzeitdarstellung der Simulationswelt zu gewährleisten.

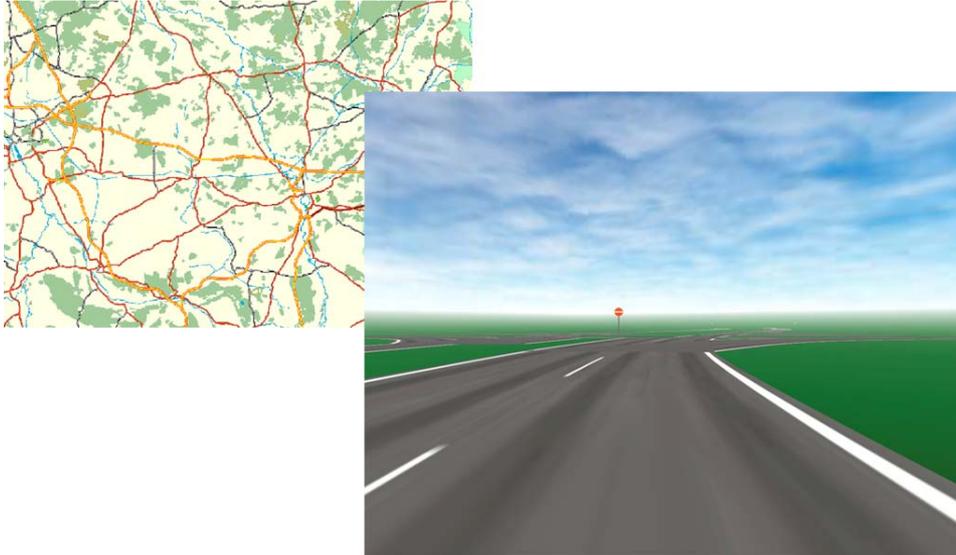


Abbildung 31: Die zweite Fahrsimulation generiert die Fahrumgebung anhand einer digitalisierten Straßenkarte

Neben dieser Schnittstelle wurde ein weiteres Interface dazu verwendet, Assistenzsystemen im FIS die Kommunikation mit der Fahrsimulation zu ermöglichen. Dazu wurde eine Netzwerkschnittstelle realisiert, über die verschiedene Betriebsdaten bzw. Situationsinformationen empfangen und umgekehrt die Simulationsumgebung beeinflusst werden können. Durch letzteres haben beispielsweise Assistenzsysteme die Möglichkeit, das Fahrzeug automatisch zu bremsen. Außerdem lässt sich die Wettersituation ändern, indem unterschiedlich dichter Nebel hinzugeschaltet wird. Der Aufbau des zugehörigen Verbindungsprotokolls konnte über die gesamte Projektlaufzeit beibehalten werden. Zur Berücksichtigung neuer Anforderungen genügte es, die neuen spezifische Inhalte zu integrieren.

6.4.2.2 Integration Personenerkennung und Nutzerprofilverwaltung

Innerhalb dieses Arbeitsbereiches wurde die Personenerkennung an den EMBASSI KFZ- Demonstrator bzw. an die dafür zuständige Nutzerprofilverwaltung angebunden. Die Kommunikation erfolgt hierbei über TCP/IP Socket-Verbindungen. Das Protokoll ist in XML definiert und beinhaltet sowohl das Initiieren des Erkennungsvorgangs durch den Benutzer bzw. die Nutzerprofilverwaltung als auch die Rückgabe der erkannten Person oder die Meldung einer Fehlererkennung.

Darüber hinaus wurden in diesem Arbeitspaket aus den gesammelten Anforderungen an die Nutzerprofilverwaltung ein Implementierungskonzept erstellt. Das berücksichtigt auch die Anbindung der Personenerkennung. Den verschiedenen Aufgaben des Profilmanagements (Personenerkennungs-Anbindung und Profilmanagement) wurde durch die Kapselung der Funktionalität in die beiden Interfaces IRecognizer und IProfilServices Rechnung getragen. Abbildung 32 zeigt den Ablauf eines Erkennungsprozesses als Sequenzdiagramms.

Darüber hinaus wurde eine Oberfläche für die Profilauswahl entworfen und umgesetzt (siehe Abbildung 33). Auf dieser Oberfläche ist nun sowohl die manuelle Auswahl eines Profils als auch die automatische Auswahl per Video-Personen-erkennung möglich.

Wie bei allen anderen GUI Komponenten wurde auch hier die Erzeugung der abhängigen Komponenten ausschließlich über Factories abgewickelt und die Designs und sprachenabhängigen Beschriftungen in Bundles ausgelagert.

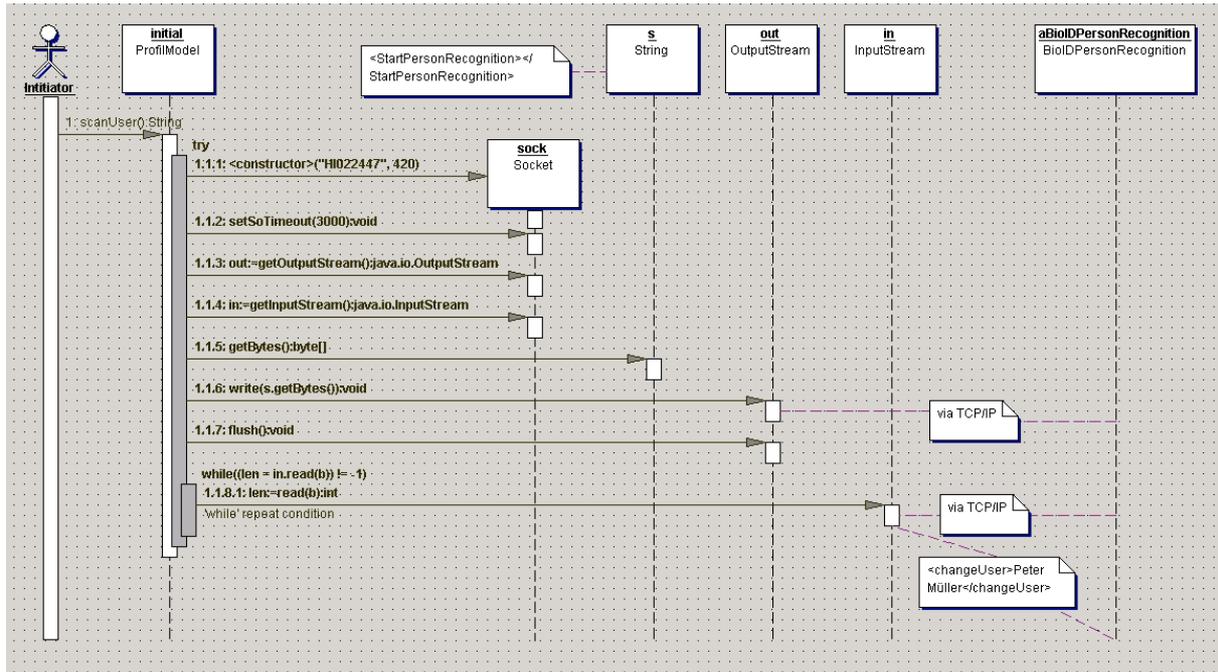


Abbildung 32: Ablauf eines Erkennungsprozesses als Sequenzdiagramm

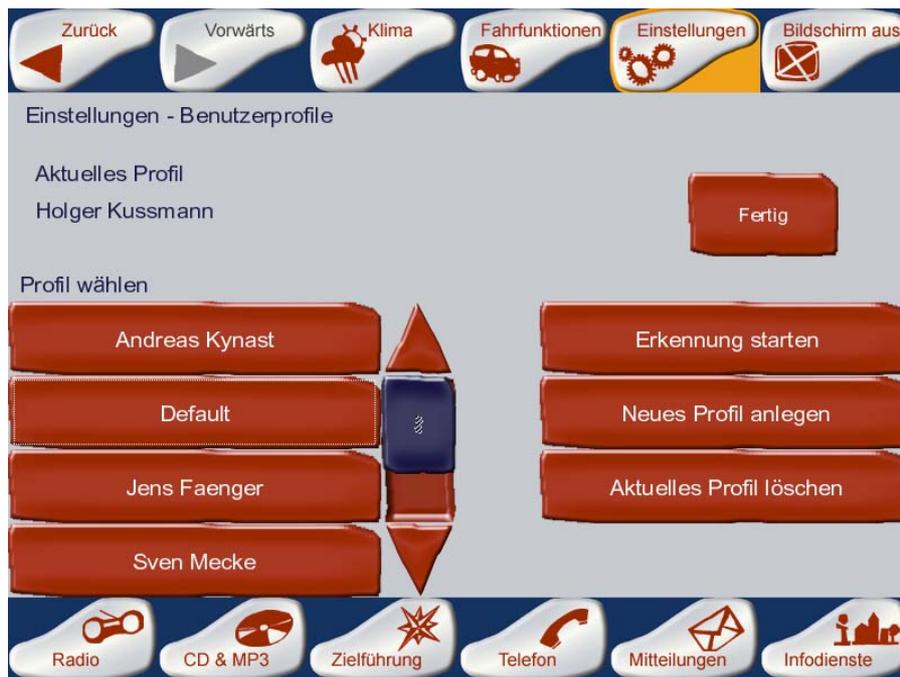


Abbildung 33: GUI für die Profilauswahl

6.4.2.3 Integration eines Kopffindungsalgorithmus in die Blickrichtungserkennung

Mit dem Ziel die Robustheit der Blickrichtungserkennung zu erhöhen, wurde dem in Kapitel 6.6.2 beschriebenen Verfahren der Gesichtsfinder des Fraunhofer ISS vorgeschaltet. Mit Hilfe seines Detektionsergebnisses wird der Suchbereich für das primäre Template-Matching beim Gradient-Orientations-Matching für jeden Frame neu festgelegt. Dadurch wird der Rechenaufwand bei der Merkmalsextraktion erheblich reduziert. Insgesamt ließ sich jedoch keine Verbesserung der Erkennungsrate feststellen, da zum Teil Gesichter von dem Gesichtsfinder nicht detektiert werden, die über die Merkmalsextraktion noch gefunden werden. Dies ist z.B. in den Grenzbereichen der Kopfdrehung oder auch bei sehr dunklen Aufnahmebedingungen der Fall. Nichtsdestotrotz

konnte die Robustheit in einigen Fällen leicht erhöht werden, weil der Gesichtsfinder durch seinen globaleren Ansatz bei schlechter Ausprägung oder Sichtbarkeit einzelner Gesichtsmarkmale etwas toleranter ist. Zusammenfassend konnte also keine wesentliche grundsätzliche Leistungssteigerung festgestellt werden. Für bestimmte Anwendungsgebiete (siehe oben) könnte eine Kombination beider Verfahren jedoch von Vorteil sein.

6.4.2.4 Integration Spracherkennung / Dialogmanager

Im Bereich dieses Arbeitsbereichs wurden Konzepte für die sprachliche Steuerung der GUI erstellt. Dafür wurden von BOSCH Rahmensätze (Grammatiken) zur sprachlichen Steuerung der GUI entworfen und es wurde ein Workshop zur Wortmodellfindung vorbereitet, durchgeführt und ausgewertet. Innerhalb dieses Workshops wurde den Probanden die Entertainmentkomponente vorgestellt. Anschließend wurden einzelne Funktionen der GUI betrachtet und die Probanden hatten die Aufgabe, sich zu überlegen, wie sie die vorgestellte Funktion per Sprache aufrufen würden. In der Auswertungsphase wurden die Sätze normiert. Tabelle 2 zeigt einen Auszug der Ergebnisse:

Funktion 14. Auto Auswahl

Normalisiert

Spiel	Was Schönes	(hören)
Mach	Meine Lieblingsmusik	(an)(hören)
Ich möchte	Irgendetwas	(an)(hören)
	Was mir gefällt	(hören)
	Irgendwelche Musik	(an)(hören)
	Einen Vorschlag	(hören)
Automatische(r)	Vorschlag	(bitte)
	Auswahl	

Schlag Musik vor

Tabelle 2: Auszug aus „Ergebnisse Workshop Sprachbedienung“

Realisierung der Umsetzung eines sprachlichen Befehls vom Erkennen über den Dialogmanager bis hin zum Funktionsaufruf im Entertainmentagenten

Aus den Ergebnissen des „Workshop Sprachbedienung“ erzeugte Siemens, das eigentliche Wortmodell des Spracherkenners. Mit diesem Wortmodell ist der Erkennen in der Lage, gesprochene Worte auf im Modell gespeicherte Parameter abzubilden. Das Ergebnis, die erkannten Sätze bzw. die wahrscheinlichsten Erkennungsergebnisse, werden dann an den Dialogmanager weitergeleitet.

Die Tabelle 2 beinhaltet ausser des hier sichtbaren Teils der normalisierten Sätze, einen weiteren Teil, der die Abbildung auf die KQML/ XML Funktionsaufrufe beschreibt.

Dieser Teil wurde von FORWISS benutzt, um innerhalb des Dialogmanagers die Abbildung eines erkannten Satzes auf einen KQML/ XML Funktionsaufruf für den Entertainment-Agenten zu realisieren.

Automatische Modenumschaltung

Im weiteren wurde in diesem Arbeitsbereich ein Konzept erstellt, mit dem eine automatische Modenumschaltung von Sprache auf GUI während eines Dialoges ermöglicht wird. Durch die Aufteilung der Entertainment Applikation nach dem erweiterten ModelViewController Pattern (ModelViewSpeechController) [GoF 1994] konnte diese Funktionalität auf einfach Weise in das System integriert werden. Die Bedienung der Entertainment-Applikation kann nun während eines Dialoges von Sprache auf GUI wechseln. Dabei wird bei der Sprachbedienung das Ergebnis immer per TTS ausgegeben und gleichzeitig auf der GUI dargestellt. Wechselt der Benutzer zur GUI Bedienung, indem er ein Element per Knopfdruck auswählt, wird das Ergebnis nur noch auf der GUI ausgegeben. Abbildung 34 zeigt das Sequenzdiagramm für einen solchen gemischten Dialog:

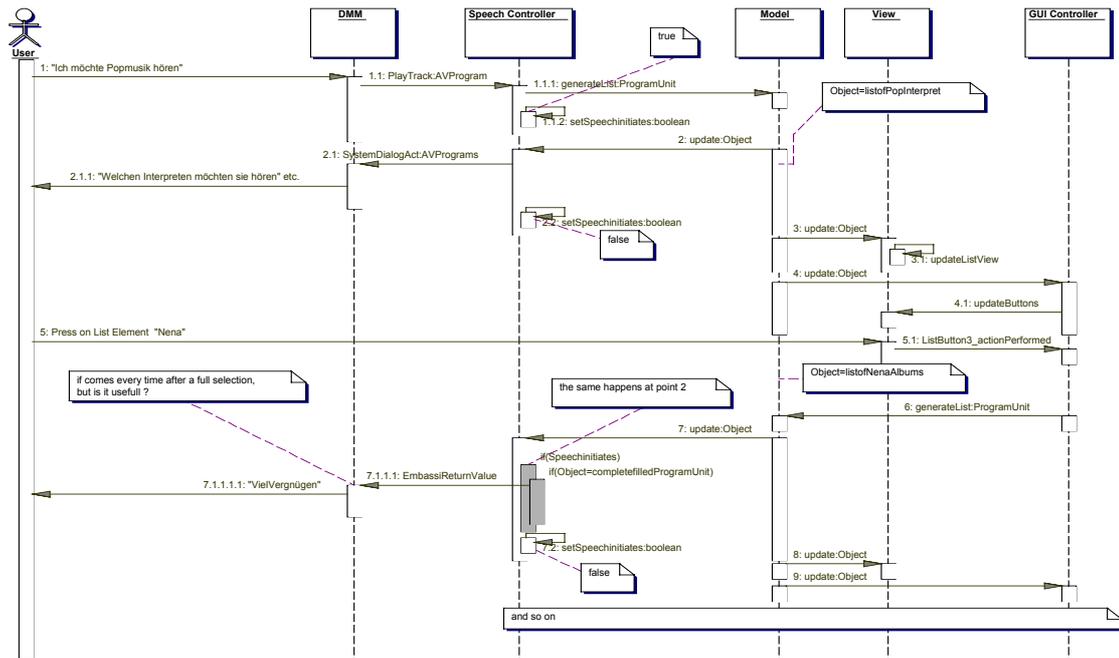


Abbildung 34: Sequenzdiagramm für einen gemischten Sprach-GUI-Dialog

Kommunikationsprotokoll Dialogmanager - Entertainment-Agent

Die Kommunikation besteht zum einen aus einem statischen Teil, nämlich der Sprachinhalte und deren Bedeutung (definiert durch die Ontologie) und zum anderen aus dem dynamischen Teil, der die Reihenfolge der Nachrichten beschreibt, die ausgetauscht werden müssen, um eine bestimmte Aktion durchzuführen. Abbildung 35 zeigt beispielhaft ein Teil des Protokolls, das zur Auswahl eines Musiktitels benötigt wird.

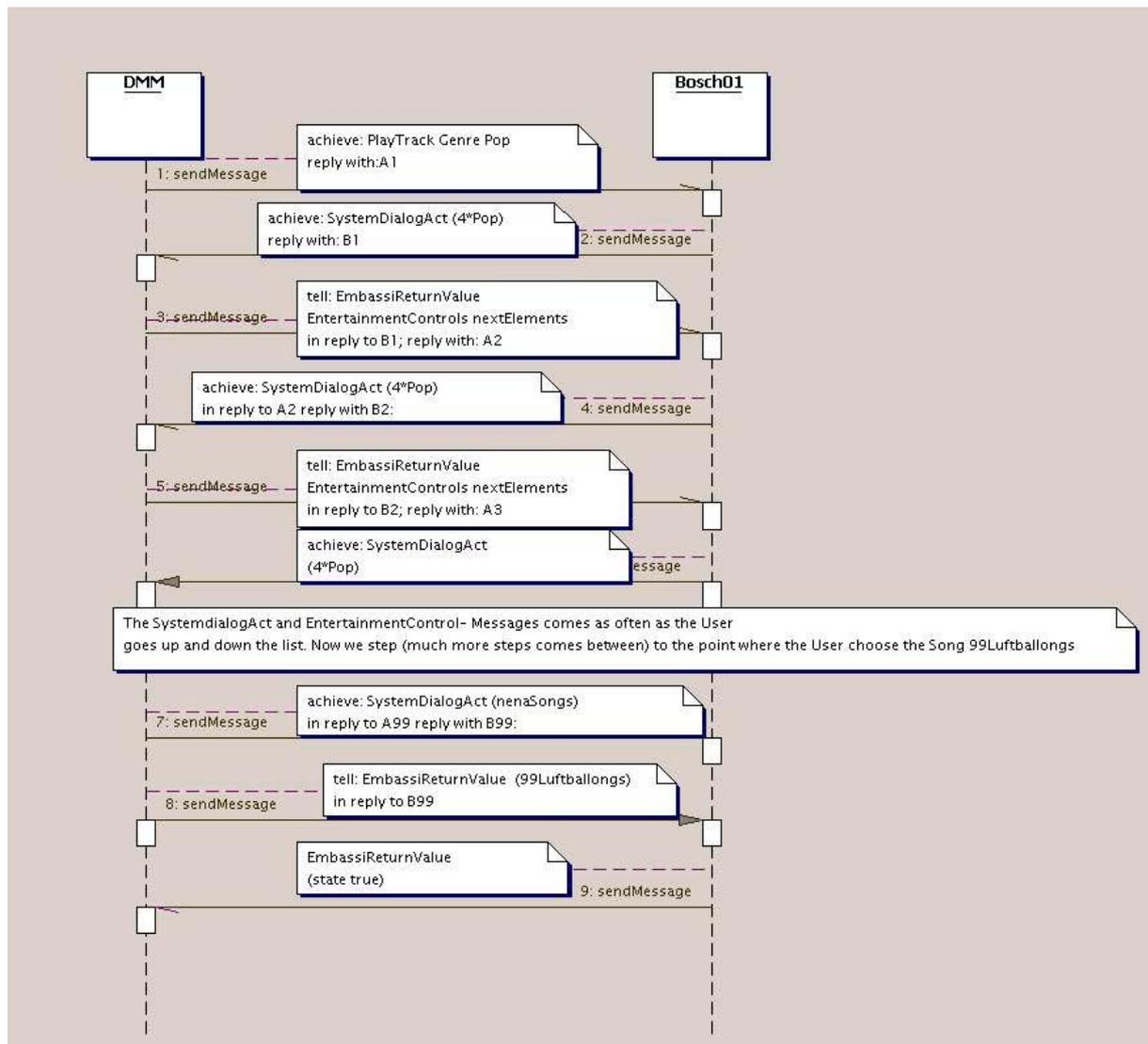


Abbildung 35: Auszug aus dem Kommunikationsprotokoll Dialogmanager - Entertainment-Agent

6.5 Durchführung und Betreuung von Evaluationen

6.5.1 Evaluation des Navigations-Systems

6.5.1.5 Literaturrecherche

Vorbereitend zur Evaluierung des Navigationssystems wurde in Zusammenarbeit mit der HUB im Rahmen eines Praktikums eine Literaturrecherche durchgeführt. Hierbei wurde untersucht, wie die Darstellung des Navigationssystems in Abhängigkeit der Situation variiert werden kann. Die Fragestellung: „Welche Form der Darstellung ist in welcher Situation am sinnvollsten?“ stand hierbei im Fokus. Die Ergebnisse dieser Recherche wurden bei der Gestaltung des Navigationssystems berücksichtigt und im Rahmen der Evaluierung untersucht.

6.5.1.6 Evaluierung

In Kooperation mit der HUB erfolgte im Rahmen einer Diplomarbeit eine Evaluierung des adaptiven Navigationssystems. Durch die Evaluierung wurden die Auswirkungen der Adaptivität im Vergleich mit nicht-adaptiven Varianten auf die Verkehrssicherheit und die Beanspruchung von Fahrern untersucht. Kriterien der Untersuchung waren dabei Effektivität, Effizienz und Zufriedenheit der Fahrer. Das Ziel der Untersuchung war es, herauszufinden, wie Benutzer auf die automatische Umschaltung reagieren bzw. welche Darstellungsvariante für welche Situation am geeignetsten ist.

Grundlage für die Umsetzung der adaptiven Funktionalität war die Annahme, dass durch eine visuelle Ausgabe von Navigationshinweisen eine erhebliche Ablenkungsgefahr vom Verkehrsgeschehen besteht, die bei einer Sprachausgabe geringer ist. Im Zusammenhang mit verschiedenen beanspruchenden Situationen sollte die daraus resultierende Ablenkung des Fahrers von der eigentlichen Fahraufgabe untersucht werden. In der adaptiven Variante wurde in nicht beanspruchenden Situationen sowohl eine Pfeildarstellung als auch die Sprachausgabe zur Zielführung eingesetzt. In beanspruchenden Situationen hingegen wurde nur eine Sprachausgabe eingesetzt, um Ablenkungseffekte infolge einer grafischen Ausgabe zu vermeiden.

Die adaptive Variante wurde mit zwei nicht-adaptiven verglichen. Dabei handelt es sich um eine kombinierte Zielführungsvariante mit Sprachausgabe und einem Richtungspfeil sowie um eine ausschließlich auditive Variante mit Sprachführung.

Während des Versuchs führen die Probanden mit einem Fahrsimulator auf verschiedenen Strecken, die alle aus dem Raum Hildesheim stammten und für diesen Zweck digitalisiert wurden. Auf diesen Strecken befanden sich unterschiedlich beanspruchende Situationen und Nebenaufgaben. Als Versuchsträger für die Untersuchung diente eine Cockpitstudie mit Lenkrad, Pedalarie und zwei Monitoren (siehe Kapitel 6.4.1). Diese stellten den Tachometer bzw. die Ausgaben des Navigationssystems dar. Die virtuelle Fahrumgebung wurde über einen Beamer auf eine Wand vor den Versuchsträger projiziert. Außerdem konnte in diese Fahrscene eine Nebenaufgabe eingeblendet werden (siehe Abbildung 36).

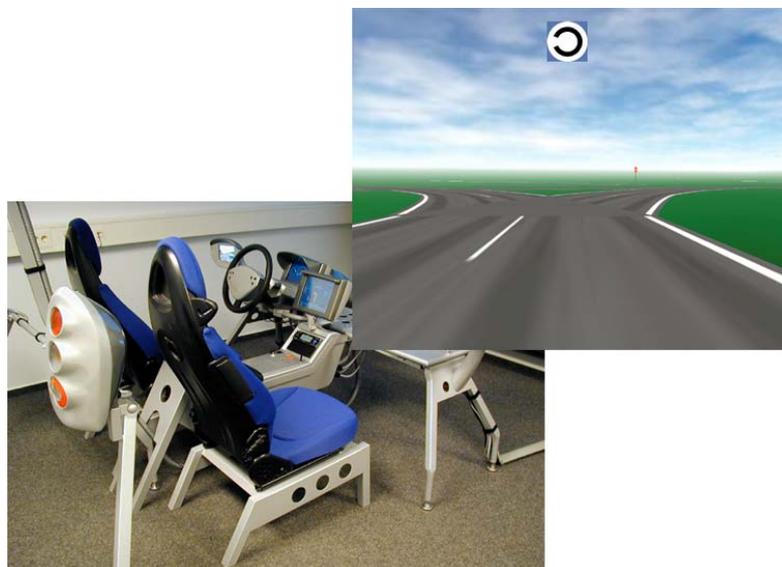


Abbildung 36: Der Versuchsträger und die Fahrumgebung mit eingeblendeter Nebenaufgabe

Aus den beiden nicht-adaptiven und der adaptiven Zielführungsvariante resultierten drei Versuchsgruppen. 63 Personen nahmen an der Untersuchung teil, allerdings konnten nur Daten von 30 Personen (10 je Versuchsgruppe) in der Auswertung berücksichtigt werden. Ursache hierfür waren anfängliche technische Probleme bei der Versuchsprotokollierung und Abbrüche aufgrund von Übelkeit (Simulator-Krankheit).

Ergebnisse

Effektivität:

Der Beanspruchungsgrad der Situation hat einen signifikanten Einfluss auf die Richtigkeit des Abbiegeverhalten, so dass die Anzahl der Falschfahrten mit dem Beanspruchungsgrad zunimmt. In der Versuchsgruppe „Standard“, welche die Navigationsinformationen durchgängig akustisch und visuell erhielt, traten tendenziell weniger Falschfahrten auf als in den anderen beiden Versuchsgruppen, die entweder temporär oder permanent ohne visuelle Informationsquelle auskommen mussten.

Effizienz:

In gering beanspruchenden Situationen wurden in der Gesamtstichprobe mehr Blicke zum Display getätigt, was zum größten Teil dadurch bedingt sein dürfte, dass die Versuchsgruppe „Adaptiv“ in dieser Bedingung die Informationen auch visuell dargeboten bekommen hat und diese Informationsquelle auch nutzte. Die Nutzung dieser Informationsquelle bestätigt auch einen Ablenkungseffekt infolge visueller Informationsdarbietung.

Zufriedenheit:

Ein Vergleich der Akzeptanzurteile nach der Fahrt zeigt zwar weder für die Dimension „Nützlichkeit“ noch für die Dimension „Zufriedenheit“ einen signifikanten Gruppenunterschied, jedoch ist eine Tendenz dahingehend erkennbar, dass die adaptive Systemvariante sowohl in Bezug auf die wahrgenommene Nützlichkeit als auch die Zufriedenheit die schlechteste Bewertung erhielt, die aber trotzdem noch positiv ist.

Am besten wird die Standardvariante mit akustischer und visueller Ausgabe bewertet, die auf beiden Dimensionen die besten Bewertungen erhält. Die Systemvariante „Sprache“ nimmt eine Mittelstellung ein – sie wird als ebenso nützlich bewertet wie die Variante „Standard“, die Nutzer sind jedoch damit ähnlich unzufrieden wie mit einer adaptiven Variante.

Ein Vergleich der Akzeptanzurteile vor und nach der Fahrt zeigt, dass sich in der Gruppe „Adaptiv“ die Bewertung in Bezug auf die wahrgenommene Nützlichkeit des Systems signifikant verschlechtert, wohingegen auf der Dimension „Zufriedenheit“ eine signifikant schlechtere Bewertung nach der Fahrt in der Gruppe „Sprache“ erkennbar ist.

Anhand der Untersuchungsergebnisse lässt sich zusammenfassend festhalten, dass das adaptive Entfernen des visuellen Ausgabekanals in Situationen mit einer hohen Beanspruchung zwar Ablenkungseffekte weg vom Verkehrsgeschehen vermeidet. Diese Abwendungen vom Display lägen mit ihrer kurzen Dauer jedoch im akzeptablen Bereich und können somit als relativ wenig gefährdend für die Verkehrssicherheit angesehen werden. Dieser vermeintliche Gewinn in Bezug auf die „Kosten“ des Nutzers in der Interaktion mit dem System wird durch schlechte Systembewertungen mehr als ausgeglichen. Nicht nur die Nützlichkeit eines solch adaptiven Systems wird in Frage gestellt, sondern auch die Zufriedenheit der Nutzer mit einem System, welches seine Ausgabe in Abhängigkeit von der Situation verändert, ist gering. Möglicherweise sind anders gestaltete adaptive Navigationssysteme nutzbringender als die in der Untersuchung begutachtete adaptive Systemausprägung. Bei der Bewertung der Akzeptanzurteile sollte jedoch berücksichtigt werden, dass möglicherweise die Untersuchungsmethode „Fahrsimulation“ in die Urteile mit einfließt und die Akzeptanzurteile in negativer Richtung verfälscht.

6.5.2 Expertenevaluierung der Bedienoberfläche

Für die Bedienung des Demonstrators wurde eine einheitliche Bedienoberfläche entwickelt, die alle Applikationen unter einem gemeinsamen Rahmen zusammenfasst. Ziel bei der Entwicklung dieser Oberfläche waren die Kfz-Tauglichkeit und eine einfache Bedienbarkeit. Hierzu gab es eine enge Kooperation mit den Psychologen der HUB, die im Rahmen von Expertenevaluierungen unterschiedliche Konzepte und Entwürfe bewerteten. Das in der finalen Form vorliegende Fahrerinformationssystem ist das Ergebnis verschiedener Iterationszyklen von Entwurf und Bewertung.

Im Rahmen dieser Evaluierungen wurden verschiedene Aspekte des Systems untersucht, u.a. der Bildschirmaufbau, die Beschriftung von Bildelementen und die Bedienlogik. Der Beschriftung der Bedienelemente wurde besondere Aufmerksamkeit gewidmet, um eine hohe Verständlichkeit durch Laien zu erzielen. Dazu wurden die Beschriftungen aller Bedienelemente im System gesammelt und in einer Liste zusammengefasst. Zu jedem Begriff wurden verschiedene Alternativen aufgestellt, die zusammen mit der HUB bewertet wurden (siehe Tabelle 3). Dabei stand als Bewertungskriterium im Vordergrund, ob ein Begriff die damit verbundene Funktionalität für den Nutzer verständlich beschreibt.

Funktion	Beschreibung	Begriffsalternativen
...
Zieleingabe	Zeigt die Bildschirmseite, in der das Fahrziel eingegeben werden kann	Zieleingabe Ziel wählen Ziel eingeben Neues Fahrziel eingeben
Zielspeicher	Liste mit den letzten angefahrenen Zielen. Die Einträge werden automatisch hinzugefügt, wenn der Benutzer ein neues Fahrziel eingegeben hat.	Zielspeicher Letzte Ziele
Persönliche Ziele	Liste mit Zielen, die explizit vom Fahrer gespeichert wurden, zum Beispiel „Nach Hause“	Persönliche Ziele Meine Fahrziele Lieblingsziele

		Bekannte Ziele Gespeicherte Ziele Adressbuch Favoriten
Einstellungen	Einstellungen, die die Berechnung des Weges beeinflussen (z.B. schnellste Route, keine Tunnel)	Navigationsoptionen Routenoptionen
...

Tabelle 3: Die Beschriftung von Bedienelementen ist das Ergebnis eines umfangreichen Auswahlprozesses. Dargestellt ist ein Ausschnitt der Liste mit Begriffsalternativen.

Neben der Beschriftung wurden auch eine Reihe von Bedienschritten optimiert. Das dahinter stehende Ziel war, dass der Fahrer mit möglichst wenigen Interaktionsschritten sein gewünschtes Ziel erreicht und somit die Blickabwendungszeiten von der Straße gering gehalten werden.

Ein Beispiel für die Optimierung ist die Bedienoberfläche des Navigationssystems: Die Gestaltung von Hauptmenü und Zieleingabe durchliefen bis zur finalen Variante verschiedene Zyklen. So enthielt beispielsweise das Hauptmenü in der ersten Version nur eine Liste von 6 Bedienelementen, die wiederum zu neuen Dialogseiten führten. In dieser übersichtlichen Variante lagen allerdings für den Fahrer oft benötigte Funktionen in tieferen Menüebenen versteckt. Das Hauptmenü wurde deshalb schrittweise erweitert, so dass es in der endgültigen Version auch eine Liste von bevorzugten Fahrzielen zeigt. Dies ermöglicht die Schnellwahl eines Ziels mit nur einem Bedienschritt. Außerdem wurden die ursprünglichen 6 Funktionen zu 4 Funktionen gebündelt (siehe Abbildung 37).

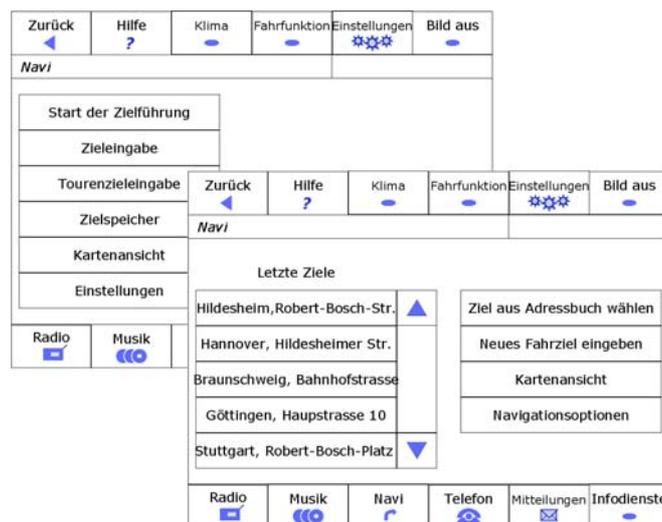


Abbildung 37 Das Hauptmenü des Navigationssystems durchlief eine mehrfache Optimierung

Wie bei der Navigations Applikation wurde auch die Entertainment Applikation Oberfläche von Experten evaluiert. Beispiele für Änderungsvorschläge zeigt die folgendende Liste für Bezeichnungsänderungen:

- **AutoVorschlag** - Lieblingsmusik starten ist nicht korrekt, da eventuell ein Vorschlag generiert wird, der nicht zur Lieblingsmusik gehört. Begriffsvorschlag: „Persönlicher Vorschlag“
- **AutoErgänzen** - Bevorzugter Begriff: „Auswahl ergänzen“
- **Next Button** - Begriffsvorschlag: „Neuer Vorschlag“ oder „Nächster Vorschlag“
- **Musik** (Neuer Begriff für Liste) - Unterschied zum Radio-Knopf ist nicht klar. Begriffsvorschlag: „CD / MP3“, „Gespeicherte Musik“ oder „Jukebox“

6.5.3 Evaluation der Störungswarnkomponente

Die Störungswarnassistentz wurde in Kooperation mit der HUB evaluiert. Dabei wurden verschiedene Assistenzgrade hinsichtlich der Kriterien Effektivität, Effizienz und Akzeptanz untersucht:

- Warnung über die Störung
- Warnung und Handlungsvorschlag
- Warnung mit Handlungsvorschlag und automatische Systemreaktion nach Zeitintervall
- Automatische Systemreaktion

Das Ziel der Untersuchung war es, herauszufinden, welcher dieser Assistenzgrade den größten Gewinn für die Verkehrssicherheit darstellt und dabei noch vom Fahrer akzeptiert wird.

Szenariountersuchung

Im Vorfeld der eigentlichen Untersuchung wurde unter Nutzung der Szenariotechnik der Unterstützungsbedarf durch die Nutzer ermittelt [Reiser et al. 1978] [Schreiber et al. 1990]. Dazu wurde jeder Versuchsperson eine Beschreibung der Funktionsweise des Assistenzsystems gegeben. Anhand dieser Erläuterung sollten die Probanden in einem Fragebogen entscheiden, in welcher Art die Warnkomponente den Fahrer in bestimmten Situationen unterstützen soll. Hierbei wurden 30 Versuchspersonen befragt. Die Situation (Beanspruchung: emotional hoch vs. niedrig, kognitiv hoch vs. niedrig, Landstraße vs. Autobahn) und der Automatisierungsgrad wurden variiert.

Das Ergebnis der Befragung ist, dass die Automatisierungsgrade „Information“ und „Information und Handlungsvorschlag“ deutlich bevorzugt wurden. Alle anderen Automatisierungsgrade erzielten geringere Erwünschtheitswerte, mit teilweise eher negativer Bewertung. Der Einfluss der unterschiedlichen Situationen fällt dabei sehr gering aus..

Die Automatisierungsgrade unterscheiden sich teilweise deutlich in ihrer Bewertung durch Versuchspersonen, wobei die Akzeptanz mit zunehmendem Automatisierungsgrad geringer wird. Möglicherweise stellt die automatische Fahrassistenz einen Eingriff in den Kompetenzbereich und das Kontrollbedürfnis des Fahrers dar, der negativ bewertet wird. Eine weitere Ursache für die geringe Akzeptanz der hoch automatisierten Assistenzgrade könnte in ihrer Neuartigkeit liegen und in einer generellen, menschlichen Tendenz zur Höherbewertung von Gewohntem und der Ablehnung von Neuem begründet sein. Allerdings wurden die automatischen Assistenzgrade in stärker beanspruchenden Situationen besser bewertet.

Untersuchung

Die Evaluierung des Warnsystems wurde mit Hilfe eines Fahrsimulators durchgeführt. Zwei Arten von Störungen wurden in dieser Simulation realisiert, ein Unfall mit einem quer stehenden LKW und eine Ölspur auf der linken Fahrspur. Diese Simulation wurde per Video-Beamer an eine Wand projiziert. Der Versuchsaufbau bestand weiterhin aus einem Lenkrad, zwei Fußpedalen zum Gasgeben und Bremsen sowie zwei Minibildschirmen für die Tachometeranzeige bzw. Darstellung der Störungswarnung. Auf einem separaten Monitor wurde außerdem eine Nebenaufgabe präsentiert

48 Versuchspersonen nahmen an der Untersuchung teil. Jede Versuchsperson absolvierte insgesamt 5 Fahrten, jeweils eine mit jedem der implementierten Automatisierungsgrade in einer festen Situationskombination. Nach jeder Fahrt füllten die Versuchspersonen einen Fragebogen zur Bewertung der gerade erlebten Systemvariante aus.

Die Fahrsituation wurde zwischen den Versuchspersonen mit den Faktoren Sicht (Nebel, kein Nebel) und Verkehrsaufkommen (viel Verkehr, wenig Verkehr) variiert. Aus deren Kombination ergaben sich vier Versuchsgruppen. Innerhalb der Versuchspersonen variierte der Automatisierungsgrad.

Ergebnisse:

Effektivität:

Zeitige Warnungen vor Störungen führten bei keiner Assistenzart zu bedeutenden Reaktionen des Fahrers. Hingegen zeigten Warnungen ab frühestens 1000 m vor der Störungsstelle signifikante Unterschiede bei den Durchschnittsgeschwindigkeit bezogen auf die Assistenzart (siehe Abbildung 38).

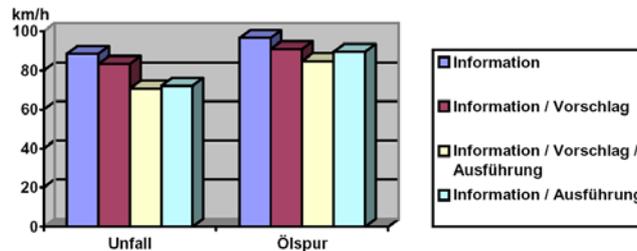


Abbildung 38 Durchschnittsgeschwindigkeiten vor den Störungsstellen

Effizienz:

Die Systeme arbeiten effizient und führen zu keiner erhöhten Beanspruchung des Fahrers.

Akzeptanz:

Beim Störungsmeldungssystem werden die Assistenzvarianten „Warnung“ und „Warnung mit Handlungsvorschlag“ signifikant hinsichtlich der erwarteten Konsequenzen besser bewertet als die Assistenzvarianten „Warnung, Handlungsvorschlag mit automatischer Systemreaktion“ und „Warnung mit automatischer Systemreaktion“. Bezüglich der Angenehmheit der Unterstützung erhalten nur die Varianten „Warnung“ und „Warnung mit Handlungsvorschlag“ eine positive Bewertung. Die am stärksten automatisierte Variante „Warnung mit automatischer Systemreaktion“ wurde als einzige bezüglich einer eventuell vorhandenen Kaufabsicht negativ bewertet. Am schlechtesten wurde eine Variante „automatische Systemreaktion ohne Information“ bewertet, die allerdings nur in einer auf die Simulationsuntersuchung folgenden Befragung vorgestellt wurde. Fast die Hälfte der Nutzer (46%) hätten sich im Anschluss an die Untersuchung, als alle Assistenzvarianten vorgestellt waren, für die Variante „Warnung, Handlungsvorschlag mit automatischer Systemreaktion“ entschieden, welche sich durch eine hohe Unterstützung bei gleichzeitiger Kontrollierbarkeit auszeichnet.

Situationsfaktoren hatten insgesamt nur wenig Einfluss auf die Akzeptanzmaße. Die als positiv eingestuften Varianten „Warnung“ und „Warnung mit Handlungsvorschlag“ wurden bei schlechter Sicht besser bewertet als bei guter. Bei größerem Verkehrsaufkommen würden die Versuchspersonen die Variante „Warnung“ häufiger benutzen. Die Variante „Warnung mit Handlungsvorschlag“ würde bei schlechter Sicht eher genutzt werden.

6.6 Technologien

6.6.1 SW-Agenten

In EMBASSI wurde die Kommunikation der meisten Komponenten als Agentenkommunikation realisiert. Damit sind diese Komponenten prinzipiell als Softwareagenten in einem Multi-Agentensystem [FIPA 1997] zu verstehen. Problematisch ist, dass der Begriff Agent (bzw. Multi-Agenten-System) in der Literatur nicht eindeutig definiert ist [FIPA 1997], [Bradshaw & Jeffrey 1997] oder [Jennings et al. 1997]. Alle bekannten bzw. in der Literatur beschriebenen Agenten-Systeme haben jedoch Gemeinsamkeiten, die auch für die EMBASSI Agenten zutreffen. Diese sollen im Folgenden kurz dargestellt werden:

Message Broker:

Agenten kommunizieren untereinander über einen Message Broker, der alle Agenten kennt und die Nachrichten von einem zum anderen Agenten transportiert bzw. zustellt.

Spezielle Agentensprache:

Die Kommunikation von Software Agenten erfolgt u.a. über sog. Agentensprachen. Ein de facto Standard für eine Agentenkommunikationssprache (ACL) ist mit KQML (Knowledge Query and Manipulation Language) und KIF (Knowledge Interchange Format) der Knowledge Sharing Initiative des ARPA gegeben [Genesereth 1994],[Labrou & Finin 1997].

KQML ist ein sprachensbasiertes Austauschprotokoll, mit dem Agenten Wissen in ihrer eigenen Repräsentation verschicken können. Die Agentensprachen enthalten u.a. sogenannte Performatives, welche die Art des Kommunikationsaktes beschreiben. Diese Performatives sind zum Teil den menschlichen Sprachaktionen nachempfunden. So gibt es z.B.: Performatives zur Beauftragung (achive) und Information (inform) von anderen Agenten. Die verwendeten Agentensprachen fungieren bei der Kommunikation quasi als Briefumschlag für den eigentlich zu kommunizierenden Inhalt (Content).

Ontologie:

Um die Zusammenarbeit von Agenten zu gewährleisten, bedarf es über die Agentensprache hinaus eine Beschreibung der Semantiken, welche die Agenten kennen und verstehen. Diese Beschreibung wird für die einzelnen Arbeitsbereiche der Agenten definiert. Die Arbeitsbereiche nennt man Domänen und umfassen zum Beispiel eine Applikation oder allgemein den Kontext, in dem die Agenten agieren. Für jede spezielle Domäne wird definiert werden, was ein kommunizierter Begriff semantisch bedeutet. Um sicher zu stellen, dass die Begriffe immer gleich interpretiert werden, werden die Beschreibungen der Domänen in einer Ontologie zusammengefasst. Alle Agenten, die basierend auf einer Ontologie realisiert werden, sind auch in der Lage miteinander zu kommunizieren und sich gegenseitig zu verstehen.

In EMBASSI eingesetzte Elemente:

- MessageBroker -> EMBASSIServer (eigens für EMBASSI entwickelt)
- Agenten Sprache -> KQML- Kontent Query and Manipulating Language
- Content Sprache -> sem.dtd EMBASSI eigene Ontologie auf xml bzw. dtd Basis

Beispiel für eine KQML / XML- Anfrage des Dialogmanager Agenten (GDM) an den Entertainment Agenten (BOS01):

Dargestellte Farben:

KQML

XML (sem.dtd als Basis)

```
(  
  (achieve  
    :language XML  
    :reply-with GDM_-5219750376599444875_1  
    :receiver BOS01  
    :ontology sem.dtd  
    :content  
  (  
    <?xml version="1.0" encoding="UTF-8"?>  
    <!-- edited by Sven Mecke (Robert Bosch GmbH) -->  
    <!DOCTYPE sem SYSTEM sem.dtd">  
    <PlayTrack>  
      <AVProgram>  
        <AVContentInfo>  
          <GenreSpec>  
            <GenreGeneric>  
              <GenreGenericValue>HipHop</GenreGenericValue>  
            </GenreGeneric>  
          </GenreSpec>  
        </AVContentInfo>  
      </AVProgram>  
    </PlayTrack>  
  )  
  :sender GDM)
```

Eigenschaften bzw. Vorteile dieser Technologie:

- Sehr lose Kopplung der einzelnen Komponenten (bzw. Agenten), dadurch einfaches Hinzufügen neuer Agenten möglich
- Schnittstellen basieren (zumeist) auf Text-Nachrichten
Kommunikation mit jedem anderen Agenten ist prinzipiell möglich
- sogenannte YellowPage Agenten können Agenten zur Durchführung eines Auftrags finden.
Die Agenten untereinander sind weitgehend unabhängig von einander.

Nachteile dieser Technologie:

- Es wird eine Ontologie benötigt (hoher Absprache-/Definitions-Aufwand):
Diese Definition der Ontologie ist eine der Hauptaufgaben bei der Erstellung von Agenten Systemen und muss für jede Problem-Domäne neu durchgeführt werden. Ausserdem ist eine Erweiterung der Funktionalität meist nur durch eine Erweiterung der Gesamt- Ontologie zu bewältigen.
- Für Text-Nachrichten muss ein Parsing durchgeführt werden.
Jede Nachricht von Agenten muss aufwendig überprüft und kontrolliert werden. Dieser Vorgang erfordert Rechenleistung und kostet Zeit, so dass bei Systeme mit hohen zeitlichen Anforderungen (z.B. Echtzeitsysteme) Performance-Probleme auftreten können. Aus diesem Grund wurden im Kfz-Senario Sensoranbindung auch nicht über diese Form der Kommunikation angebunden.

6.6.2 Blickrichtungserkennung

Die Verfahren zur Ermittlung der Blickrichtung lassen sich in zwei Gruppen unterteilen: in eine grobe und eine genaue Erkennung. Die genaue Blickrichtung (Genauigkeit $< 1^\circ$) wurde in der Vergangenheit im Wesentlichen für drei Anwendungen verwendet:

- Ermittlung der Blickrichtung für die Evaluationen im Bereich der Werbung oder der Eignung von Bedienoberflächen mit Probanden.
- Die Ermittlung der Blickrichtung von Schwerstbehinderten, die nur über Augenbewegungen am PC kommunizieren können.
- Die Ermittlung der Blickrichtung im Umfeld medizinischer Untersuchungen.

Die beiden letzten Bereiche treiben vor allem die IR-gestützten Verfahren zur Auswertung verschiedener Reflexionen im Auge. Die Anwendungen erlauben allerdings nur eine Beschränkung auf sehr stark fixierte Kopfpositionen (Kinnauflage, Kopfarretierung).

Der Nachteil der genauen Verfahren ist neben der Kalibrierung eine starke Begrenzung des Sichtbereichs, der erfasst werden kann. Für bestimmte Anwendungen (z.B. die Ermittlung der Aufmerksamkeit) ist die hohe Genauigkeit nicht notwendig. Allerdings muss hier der Sichtbereich sehr groß sein, um z.B. zu erfassen, dass der Fahrer in den rechten Außenspiegel schaut. Für diese Anwendungen sollte daher eine grobe Blickrichtungserkennung verfolgt werden. In diesem Fall wird die Blickrichtung aus der Kopforientierung abgeleitet.

Parallel zu diesen Arbeiten wurde ein Verfahren entwickelt, das robust gegenüber Lichtschwankungen sein und ohne Initialisierung funktionieren sollte. Daraufhin entstand das Gradient-Orientations-Matching.

Um eine Aussage über die Anforderungen bezüglich des Skalierungsbereichs und des Betrachtungswinkels treffen zu können, wurde zunächst eine Studie durchgeführt, in der potentielle Kamerapositionen im Kfz untersucht worden sind.

6.6.2.7 Positionierung von Kameras im Kfz zur Beobachtung des Fahrers

Wenn eine Kamera im Kfz integriert werden soll, stellt sich zunächst die Frage: Wo soll die Kamera platziert werden? Die Antwort auf diese Frage hängt von den verschiedensten Faktoren ab und sollte daher anhand einer Studie näher untersucht werden.

Mit der Kamera sollten Testsequenzen zur Untersuchung von Analyse-Algorithmen aufgenommen werden. Diese Algorithmen müssen auf die fahrzeugspezifischen Umgebungsbedingungen wie z.B. Lichtverhältnisse angepasst werden. Im Idealfall sollte das Gesicht des Fahrers immer vollständig zu sehen sein, um charakteristische Merkmale eines Fahrers einwandfrei erfassen zu können. Das ist allerdings teilweise auf Grund der Bewegungen des Fahrers unmöglich.

Zur Evaluierung wurden daher für verschiedene Kamerapositionen Standbild-Aufnahmen vom Fahrer in verschiedenen typischen Situationen gemacht. Zur Beurteilung der Sichtbarkeit des Gesichts wurden die Aufnahmen zusätzlich mit Personen verschiedener Körpergröße wiederholt, die neben der normalen auch die extremen Sitzpositionen einnehmen.

6.6.2.7.1 Versuchsbeschreibung

Kamerapositionen

Die Kameraposition entscheidet darüber, wie gut das Gesicht des Fahrers sichtbar ist und welche Bildinhalte zu erwarten sind. Dieses hat fundamentalen Einfluß auf die Algorithmen der Bildanalyse. Sind die Charakteristiken eines Gesichtes gut sichtbar, so kann ein z.B. Eye-Tracking System die Gesichter einfacher und damit zuverlässiger finden und verfolgen.

Darüber hinaus ist aber auch wichtig, was neben dem Fahrer sonst noch zu sehen ist. Ist der Hintergrund eher statisch oder verändert er sich während der Fahrt – sind zum Beispiel Fenster im Bild oder eher nicht. Gerade

Fenster beeinflussen stark die Lichtverhältnisse bzw. führen zu starken Schwankungen der Helligkeit, welche die Ergebnisse der Bildanalyse wesentlich verschlechtern.

Ein wichtiger Aspekt, der den Einbau einer Kamera erschwert, ist der Aufbau eines Automobils. Eine Installation der Kamera direkt vor dem Gesicht des Fahrers ist unmöglich, da dieser freie Sicht auf das Verkehrsgeschehen haben muss. Dabei liefert diese Position natürlich die beste Sicht auf das Gesicht des Fahrers. Betrachten wir nun verschiedene Anbringungsmöglichkeiten und diskutieren diese etwas genauer. Eine eingehende Beurteilung erfolgt bei der Auswertung der Messdaten und Bilder. In Abbildung 39 sind die geeignetsten Positionen eingezeichnet.

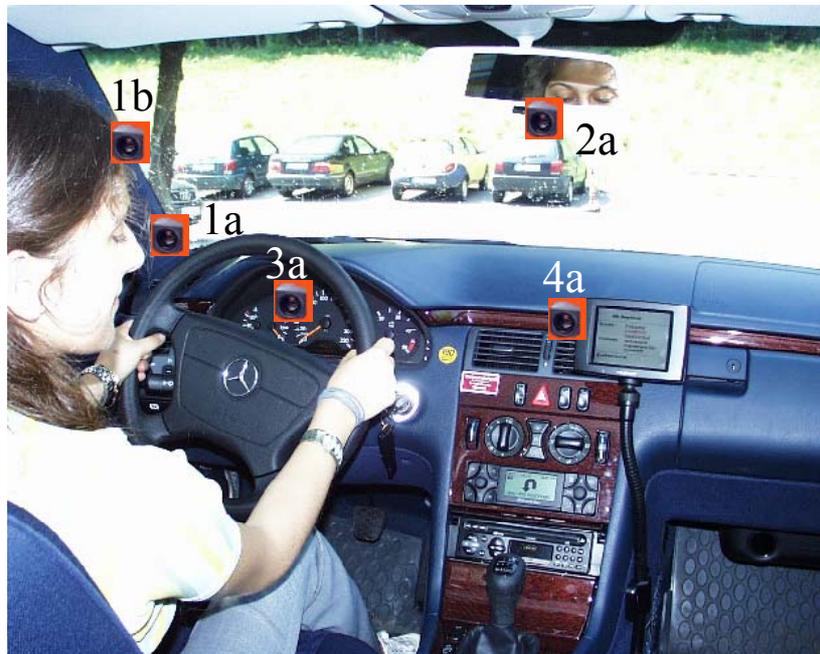


Abbildung 39: Untersuchte Kamerapositionen im Testfahrzeug

Kameraposition 1a	Kameraposition 1b	Kameraposition 2	Kameraposition 3	Kameraposition 4

Abbildung 40: Aufnahmen aus den verschiedenen Kamerapositionen

Testpersonen

Da die Sitzposition stark von der Testperson abhängt und damit auch der Sichtbarkeitsbereich der Kamera sehr unterschiedlich ist, wurden Testpersonen verschiedener Körpergröße verwendet, die jeweils auch die Sitzeinstellungen und Körperhaltungen variierten.

Es wurden drei verschiedene Testpersonen beispielhaft „vermessen“:

- | | | |
|-----------------------|--------|--------|
| 1. kleine Person | Größe: | 165 cm |
| 2. mittelgroße Person | Größe: | 180 cm |
| 3. große Person | Größe: | 196 cm |

Da hier zunächst nur die Größe eine Bedeutung zur Erfassung der Sichtbarkeitsbereiche der Kamera hatte, wurden keine anderen Anforderungen an die Testpersonen gestellt.

Sitzeinstellungen

Die folgenden Sitzeinstellungen wurden für die unterschiedlichen Messungen eingestellt:

1. Optimale Sitzposition
2. Sitz extrem weit vorne
3. Sitz extrem weit hinten

Bewertungskriterien

Zur Bewertung der verschiedenen Kamerapositionen ist in erster Linie den Ansprüchen des Head-/Eyetracking-Systems Rechnung zu tragen, d.h. den Analysealgorithmen der Bildverarbeitung. Dabei spielt sowohl der Abstand des Kopfes der Testperson eine Rolle als auch der Blickwinkel, mit dem die Kamera die Testperson sieht. Im Idealfall sind beide Augen im Kamerabild gut sichtbar und der Kamerachip ist annähernd parallel zum Gesicht des Fahrers. Im Allgemeinen sollte möglichst das gesamte Gesicht sichtbar sein.

Des Weiteren ist der Bildhintergrund von Bedeutung, der im Idealfall keine Konturen, Bewegungen und Helligkeitsschwankungen aufweist. Je statischer der Hintergrund desto einfacher ist z.B. die Lokalisation und Verfolgung des Fahrers. Das bedeutet z.B., dass Kamerapositionen, die als Hintergrund ein Fenster zeigen, weniger geeignet sind als jene, die als Hintergrund das Fahrzeugdach aufweisen.

6.6.2.7.2 Durchführung der Messungen

Messungen zur groben Ermittlung der geeignetsten Kameraposition

Für alle oben genannten Kamerapositionen wurden die folgenden Messungen durchgeführt:

1. **Eine mittelgroße Person (180 cm)** stellt den Sitz in die optimale Position ein. Nun werden Testbilder von typischen Blickrichtungen aufgenommen. Diese sind:
 - Blick geradeaus
 - Blick durch das linke Fenster
 - Blick durch das rechte Fenster
 - Blick in den linken Seitenspiegel
 - Blick in den rechten Seitenspiegel
 - Blick in den Innenspiegel
2. Bei der optimalen Sitzposition wird der Abstand vom Kopf des Fahrers zur Kamera in 3 verschiedenen Haltungen gemessen, die bei einer realistischen Fahrt vorkommen:
 - normale, bequeme Haltung (normaler Abstand)
 - Kopf angelehnt an die Kopfstütze - bequeme Sitzhaltung (maximaler Abstand)
 - Oberkörper etwas nach vorne gebeugt (minimaler Abstand)
3. Die extremen Sitzpositionen vorne und hinten werden eingestellt und Testbilder aufgenommen, wobei der Fahrer einmal mittig im Sitz, einmal etwas nach rechts und einmal etwas nach links gelehnt ist.
4. In allen Haltungen aus 3. wurde der Abstand zwischen Kopf und Kamera gemessen.

Messungen mit Testpersonen unterschiedlicher Körpergröße

Aus den unter 3.1 beschriebenen Messungen wurden 2 Kamerapositionen ermittelt, die mit Blick auf die Sichtbarkeitsbereiche und unter Berücksichtigung der Realisierbarkeit am geeignetsten erscheinen (siehe Auswertung in Kapitel 4). Für diese Positionen wurden nun Testbilder mit verschiedenen Testpersonen aufgenommen. Diese sollten nun im einzelnen zeigen, welche Probleme unter Umständen bei großen und kleinen Personen auftreten.

Dazu wurden erneut die unter 3.1 genannten Messungen für die 3 Testpersonen und die 2 Kamerapositionen durchgeführt.

6.6.2.7.3 Meßergebnisse und Auswertung

Abstandswerte:

Die folgenden Graphiken zeigen jeweils bei optimaler Sitzposition den ungefähren Bewegungsbereich des Fahrers. Dabei wurden die Abstände ermittelt.

Der Bewegungsbereich variiert zwischen 22 cm (Position 2a) und 35 cm (Position 3a). Für die ausgewählten Positionen 1a und 4 beträgt er 28 cm und 23 cm. Ganz grob kann man festhalten, daß der Fahrer meistens einen Abstand von 50 cm bis 86 cm zur Kamera hat (für die in Frage kommenden Positionen). Für die Position 4 liegen die Abstände bei 62 cm und 85 cm, wobei der Bewegungsbereich am kleinsten ist (siehe Abbildung 41).

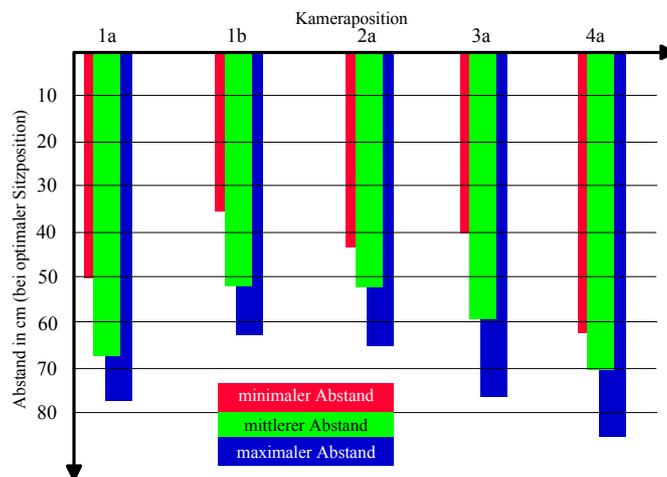


Abbildung 41: Abstände zwischen Kopf und Kamera bei unterschiedlichen Kamerapositionen und Körperhaltungen (normaler Bewegungsbereich bei optimaler Sitzposition)

Bei extremen Sitzpositionen vergrößert sich natürlich der Bewegungsbereich. Ein System, welches auch dann noch einwandfrei funktionieren soll, muss einen Bereich von bis zu 45 cm abdecken können. In Abbildung 42 sind die minimalen und maximalen Abstände dargestellt.

Auch hier ist der Bewegungsbereich für Position 4 mit etwa 31 cm relativ gering. Die Abstände liegen dann zwischen 55 cm und 90 cm. Bei Position 1a sind es 40 cm und 85 cm.

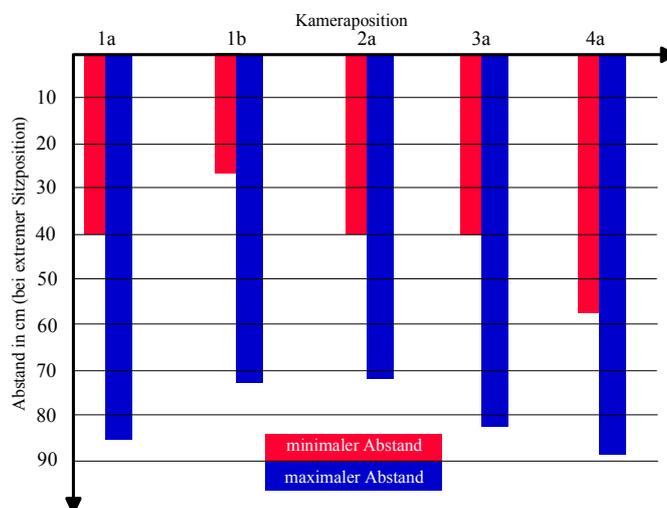


Abbildung 42: : Abstände zwischen Kopf und Kamera bei unterschiedlichen Kamerapositionen und Körperhaltungen (normaler Bewegungsbereich bei extremer Sitzposition)

Abstandsdaten für unterschiedliche Testpersonen

Auch für verschieden große Testpersonen wurden die Abstände zwischen Kopf und Kamera ermittelt, um zu sehen, ob hier große Unterschiede zu erwarten sind. Dabei wurden die Kamerapositionen 1a und 4 berücksichtigt, da diese am geeignetsten erscheinen.

Für die erste Position (1a) wird ersichtlich (Abbildung 43), dass der Bewegungsbereich von der Größe der Testperson abhängt. Je größer die Person, um so größer der Bewegungsbereich. Das gilt natürlich auch für die minimalen und maximalen Abstände.

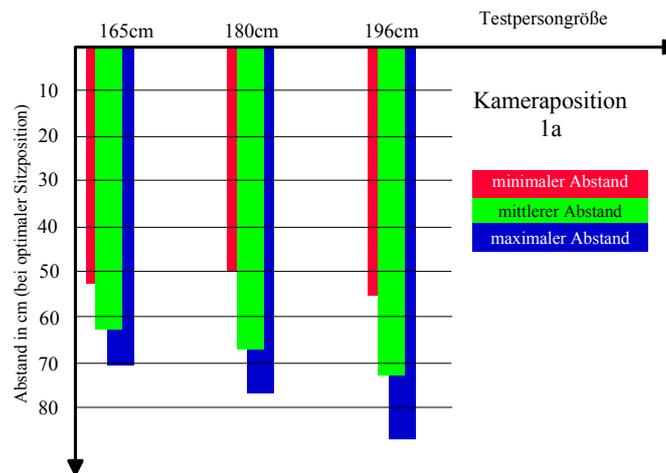


Abbildung 43: Abstand zw. Kopf und Kamera bei optimaler Sitzposition in Kameraposition 1a

Der mittlere Abstand, der einer bequemen, optimalen Sitzposition entspricht, liegt somit zwischen 60 cm und 70 cm bei Position 1a. In Abbildung 44 wird deutlich, daß dieser bei Kameraposition 4 größer ist und etwa zwischen 70 cm und fast 90 cm liegt. Hier zeigt sich aber auch, daß die Abstandsmessung nur grobe Anhaltspunkte liefern kann, da zum einen jeder Fahrer anders sitzt und zum anderen gerade bei diesen Messungen keine exakt reproduzierbaren Werte ermittelt werden, da kein objektives Verfahren verwendet wurde.

Der maximale Abstand der kleinen Person ist genauso groß, wie der der mittleren Person. Der mittlere Abstand der kleinen Person ist sogar größer als der der mittleren Person.

Das liegt daran, daß jeder Fahrer individuell entscheidet, welche Position er noch für praktisch relevant hält. Außerdem spielt hier die Anatomie der Testfahrer eine Rolle (Armlänge, Beinlänge, usw.), die aber nicht weiter betrachtet werden soll. Um hier statistisch exakte Werte zu erhalten, müßten wesentlich umfangreichere Messungen durchgeführt werden. Hier sollte lediglich ein Richtwert ermittelt werden. Der wiederum ist in vielen Fällen durchaus ausreichend, da für das System hauptsächlich eine grobe Abschätzung des mittleren Abstandes, des Bewegungsbereiches und des minimalen und maximalen Abstandes von Interesse ist.

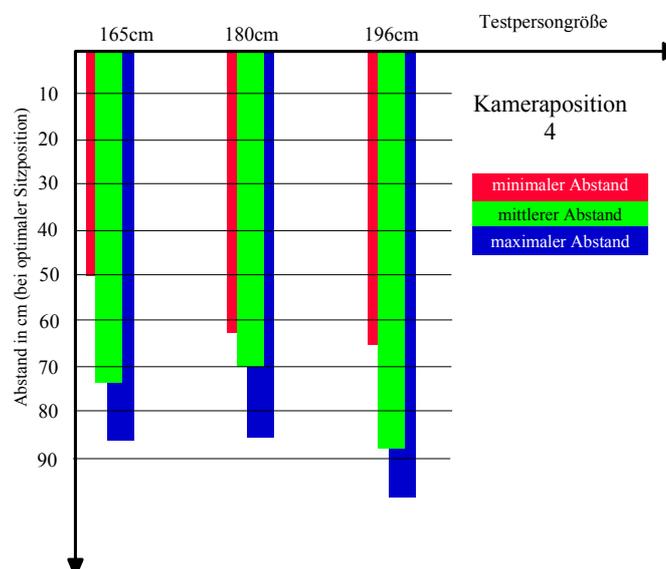


Abbildung 44: Abstand zw. Kopf und Kamera bei optimaler Sitzposition in Kameraposition 4

Fazit

Aufgrund der Tatsache, dass Kameraposition 3 quasi der Kameraposition im stationären Laborbetrieb des Eyetracking-Systems entspricht, würde man diese als am besten geeignet befinden. Nimmt man aber die Probleme, die durch die zeitweise Verdeckung der Sicht durch das Lenkrad entstehen, hinzu, erscheint diese Kameraposition als nicht ohne weiteres realisierbar.

Die Kamerapositionen 1b und 2 scheiden aufgrund ihres geringen Abstandes zum Fahrer aber auch wegen ihrer schrägen Blickrichtung auf den Fahrer aus. In diesen Fällen lassen andere Objektive bessere Ergebnisse erwarten.

Die besten Ergebnisse liefern die Kamerapositionen 1a und 4, die beide wegen ihres geeigneten Bildausschnitts und nicht zu starker Schrägansicht einen sinnvollen Kompromiß darstellen.

Bei Position 4 ist nur das Fenster ein Störfaktor, dessen Auswirkungen in Zukunft untersucht werden müssen.

Für die Abstände zwischen Kopf und Kamera läßt sich zusammenfassend folgendes festhalten:

Es ergibt sich ein mittlerer Abstand von etwa 65 cm für Position 1a und 80 cm für Position 4. Der Bewegungsbereich wird für beide Positionen etwa 30 cm bis maximale 40 cm sein. Der minimale Abstand liegt ungefähr bei 50 cm und der maximale Abstand kann auf fast 100 cm ansteigen. Bei Position 1a ist er mit 90 cm etwas geringer.

Bemerkung: Diese Werte gelten natürlich zunächst nur für das verwendete Fahrzeug (Mercedes E-Klasse Bj.1999).

6.6.2.8 Ermittlung der Kopforientierung mittels Template-Matching

Zunächst entstand ein Programm, das die Drehrichtung des Kopfes um die Hochachse (d.h. die seitliche Orientierung des Kopfes) durch einen Bildmustervergleich erkennt und grob klassifiziert.

Es ist anfänglich eine individuelle Initialisierung erforderlich, bei der der Anwender eine Kopfdrehung von links nach rechts durchzuführen hat. Dabei werden nacheinander (optional) fünf Phasen der Kopfdrehung aufgenommen und in Form von Referenz-Bildblöcken gespeichert.

Beim Tracking werden die aktuellen Kamerabilder (nicht-transformierte Luminanzwerte) durch ein Block-Matching-Verfahren nach den Referenzmustern durchsucht [Brunelli & Poggio 1993]. Ausgewählt wird das Referenzmuster mit der relativ größten Ähnlichkeit zu einem entsprechenden Bildblock im aktuellen Kamerabild. Auf diese Weise können die in Abbildung 45 gezeigten fünf Phasen der Kopfdrehung zuverlässig durch das Modul klassifiziert werden.



Abbildung 45: Schematische Darstellung der vom Standardmodul klassifizierten fünf Phasen der Kopforientierung

Definition eines Orientierungsvektors

In der Initialisierungsphase wird der Anwender aufgefordert, nacheinander fünf Raumpunkte $R_1 \dots R_5$ mit bekannten Raumkoordinaten durch eine entsprechende Kopfdrehung anzuschauen. Bei diesem Vorgang werden fünf Referenzmuster $P_1 \dots P_5$ des Kopfes aufgenommen.

In der anschließenden Trackingphase werden die aktuellen Kamerabilder durch ein Block-Matching-Verfahren nach den gespeicherten Referenzmustern durchsucht. Zur Bestimmung der Orientierung des Kopfes wird dabei das (zum Raumpunkt R_i gehörige) Referenzmuster P_i mit der relativ größten Ähnlichkeit zu einem entsprechenden Bildblock B_{aktuell} im aktuellen Kamerabild ausgewählt.

Die Position des zum Mittelpunkt von B_{aktuell} korrespondierenden Objektpunktes $O(x,y,z)$ wird als Ursprungsort des Orientierungsvektors definiert. Die x,y,z -Koordinaten von O können bei bekanntem Abstand des Kopfes von der Kamera (z -Koordinate von O) und bekannten internen Kameraparametern wie Brennweite und Chipgröße aus den Mittelpunktskoordinatenwerten x_c, y_c von B_{aktuell} berechnet werden (Reprojektion des Bildpunktes $B(x_c, y_c)$ in den Raum). Der Ursprung des Koordinatensystems liegt im Zentrum des Kamerachips. Der von $O(x,y,z)$ auf den Raumpunkt R_i zeigende Vektor definiert die Richtung des Orientierungsvektors.

Interpolation von Zwischenwerten der Kopforientierung

Wenn der Anwender seinen Kopf in der Trackingphase nicht exakt in der gleichen Weise ausrichtet wie während der Initialisierung, weichen die im Kamerabild gefundenen ähnlichsten Bildblöcke B_{aktuell} mehr oder weniger stark von den Referenzmustern $P_1 \dots P_5$ ab. Es wurde untersucht, ob auf Grundlage eines Differenzmaßes eine Interpolation von Zwischenwerten der Kopforientierung (ggf. mit Winkelangabe) möglich ist.

Dazu wurden jeweils drei Messungen bei vorgegebener Kopforientierung durchgeführt. Die Messergebnisse sind im folgenden tabellarisch und (auszugsweise) graphisch dargestellt. Die Zahlenwerte kennzeichnen die Differenz der Intensitätswerte zwischen den fünf gespeicherten Referenzmustern $P_1 \dots P_5$ einer Testperson und dem bei der vorgegebenen Orientierung gemessenen ähnlichsten Bildblock B_{aktuell} .

Orientierung	P ₁	P ₂	P ₃	P ₄	P ₅
max. links	754	1788	3033	4024	2148
max. links	813	1438	2896	3447	1957
max. links	790	1332	2979	3464	1881
links	1027	1006	2857	3182	1661
links	1168	754	2552	3199	1665
links	1695	825	1445	3114	1653
mittig	2213	1559	969	2987	2022
mittig	2276	1513	922	2659	2264
mittig	2272	1908	1519	1584	2270
rechts	2289	1967	1623	1383	2309
rechts	2276	1910	1811	934	2187
rechts	2255	2056	2326	1241	2181
max. rechts	2297	2172	3166	2462	1283
max. rechts	2012	1932	3440	2960	924
max. rechts	1867	1648	3725	3326	852

Tabelle 4: Messungen zur Interpolation von Zwischenwerten der Kopforientierung

Wie man sieht, variieren die Differenzwerte $B_{\text{aktuell}} - P_1$ bei wiederholten Messungen relativ stark, so dass eine Interpolation im Allgemeinen keine zuverlässigen Aussagen über Zwischenwerte der Kopforientierung ermöglichen würde. Während der in Abbildung 46 als Beispiel aufgetragene Kurvenverlauf für eine Interpolation geeignet wäre, zeigen die Kurvenverläufe in Abbildung 47 starke Streuungen bei Messwiederholungen. Hinzu kommt das Problem, dass die Intensitätsdifferenzen in Abbildung 46 allein keine Aussage über die korrekte Richtung der Interpolation ermöglichen. Beispielsweise tauchen bestimmte interpolierte Differenzwerte sowohl links als auch rechts von P_2 auf.

Abbildung 48 zeigt, dass bereits geringfügige, kaum wahrnehmbare Veränderungen der Raumbelichtung zu starken Veränderungen der Intensitätsdifferenzen führen.

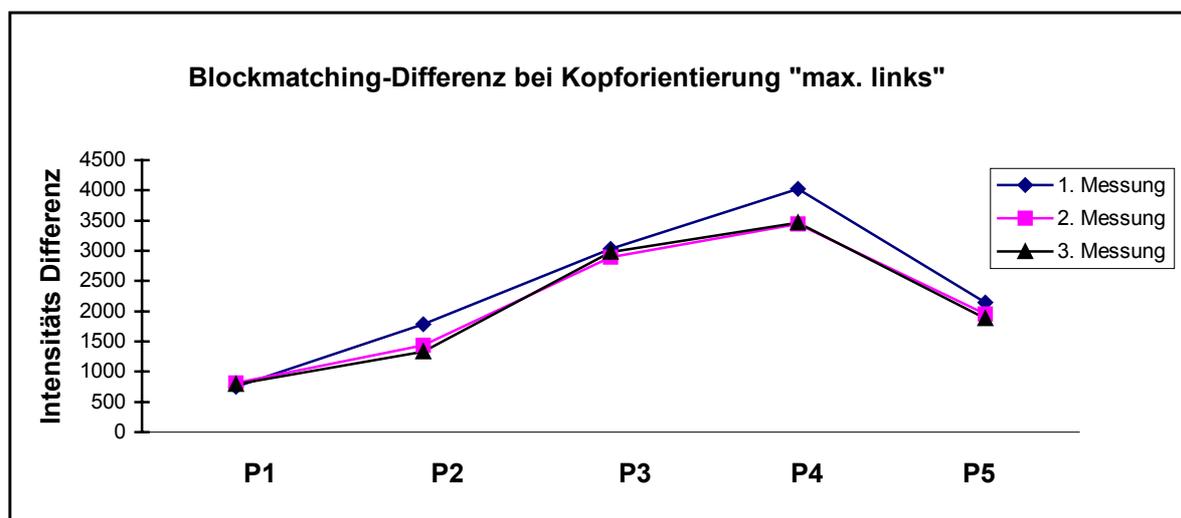


Abbildung 46: Diese Kurven zeigen einen Verlauf, der für eine zuverlässige Interpolation geeignet wäre ("max. links" = -20°, P₁ ... P₅ = Referenzmuster der Testperson für die fünf Raumpunkte "max. links", "links", "Mitte", "rechts", "max. rechts").

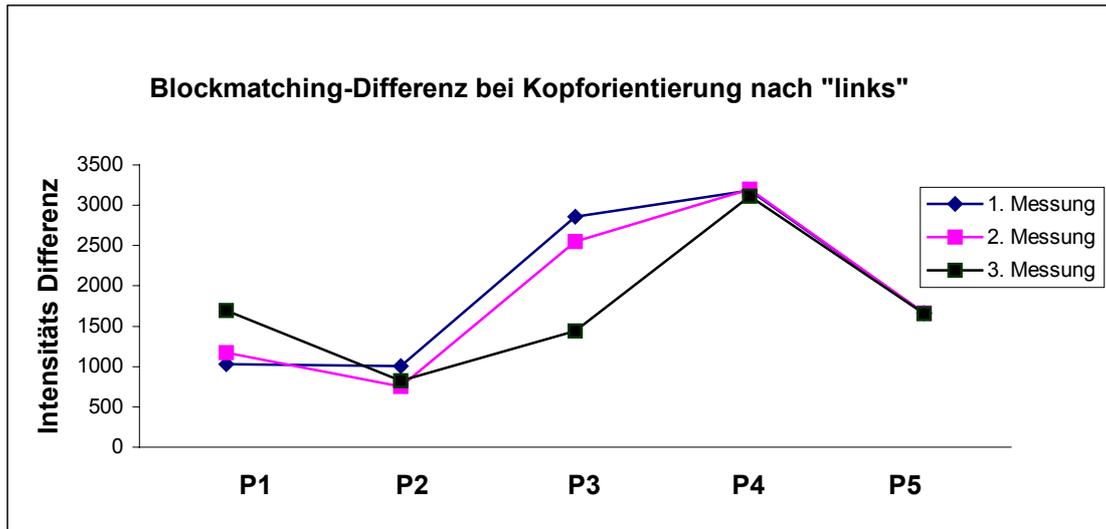


Abbildung 47: Beispiel für eine relativ starke Streuung der Intensitätsdifferenzen bei Messwiederholungen ("links" entspricht einem Drehwinkel des Kopfes von -10°).

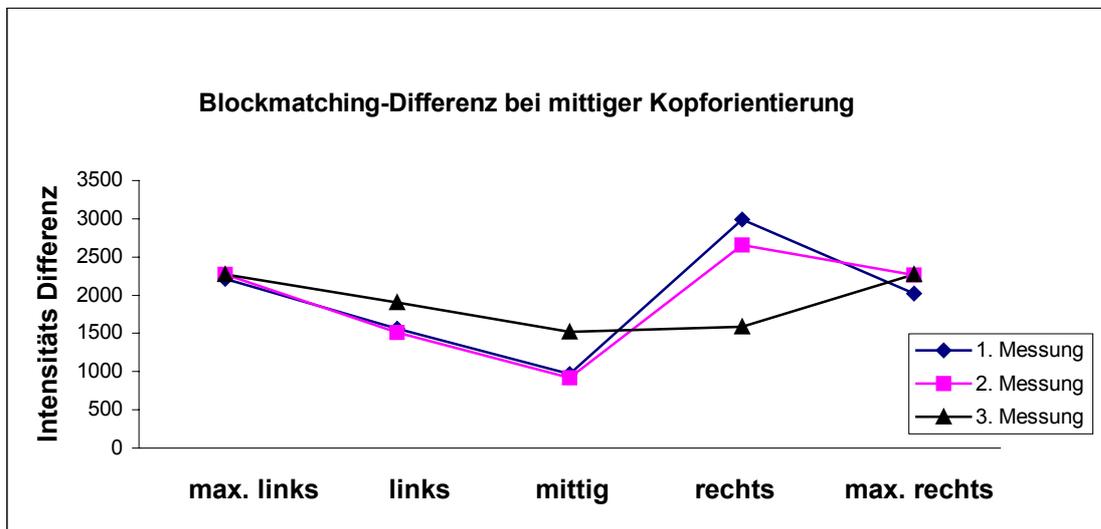


Abbildung 48: Abhängigkeit der Blockmatching-Differenz von der Raumbeleuchtung. Vor der Aufnahme der dritten Messkurve wurde die Beleuchtung geringfügig geändert.

6.6.2.9 Ermittlung der Blickrichtung mittels Gradient-Orientations-Matching

Es wurde ein Verfahren entwickelt, das anhand von monochromen Videobildern die Blickrichtung einer Person schätzt.

Der Ablauf gliedert sich in mehrere Stufen (siehe Abbildung 49):

Zunächst wird das Gesicht innerhalb des Videobildes mittels eines kantenbasierten Matching-Verfahrens lokalisiert [Baumann 2001]. Dann findet eine verfeinerte Merkmalsextraktion statt, die dazu dient, Bezugspunkte für die anschließende Kopfposen-Schätzung zu präzisieren. Aus der ermittelten Kopfpose wird in einem abschließenden Schritt die Blickrichtung abgeleitet.

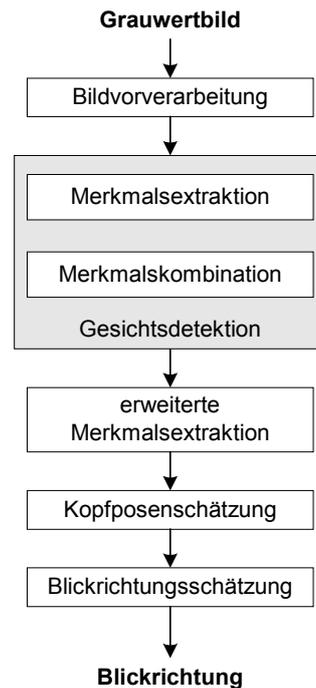


Abbildung 49: Stufen der Blickrichtungsschätzung mittels Gradient-Orientations-Matchings

Im Folgenden soll nun auf die einzelnen Stufen des Verfahrens näher eingegangen werden:

6.6.2.9.1 Bildvorverarbeitung

Anpassung des Videobildes

Der Algorithmus ist für monochrome Videobilder mit 256 Graustufen und einer Auflösung von 384x288 Bildpunkten ausgelegt. Es findet eine handelsübliche CCD-Kamera Anwendung, deren Signal mit einem Framegrabber digitalisiert wird. Die Luminanzwerte werden dann in dem benötigten Format für die weitere Verarbeitung bereitgestellt.

Generierung des Kanalwert-Bildes

Das Verfahren beruht auf der Auswertung der lokalen Orientierung des Bildinhaltes. Dazu wird für jeden Bildpunkt die Richtung der lokalen Intensitätsveränderung bestimmt.

Die genaue Bestimmung der lokalen Orientierung geschieht durch Bildung des Gradienten:

$$\nabla I(x, y) = \begin{pmatrix} \frac{\partial}{\partial x} I(x, y) \\ \frac{\partial}{\partial y} I(x, y) \end{pmatrix}$$

Dabei ist $I(x, y)$ der Intensitätswert I an der Position (x, y) des Grauwertbildes. Der Gradient der Intensität ist also ein Vektor im 2D-Bildraum, der den Änderungsgrad von dunkel nach hell angibt.

Es werden zunächst separat die horizontalen und die vertikalen Kantenlagen durch Verwendung des Sobel-Operators ermittelt, der folgende Anordnung besitzt:

$$\Delta_x f(x, y) = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad \text{bzw.} \quad \Delta_y f(x, y) = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Die Differenzierung erfolgt mit einem symmetrischen Bezugspunkt über die direkten Bildpunkt-Nachbarn. Die mit dem Sobel-Operator gefilterten Bilder sind im Folgenden mit dem Index „sobel“ versehen.

Sie ergeben sich aus:

$$I_{sobel,x}(x, y) = \Delta_x f(x, y) * I(x, y) \quad I_{sobel,y}(x, y) = \Delta_y f(x, y) * I(x, y)$$

Aus diesen Gradientenbildern lassen sich nun das Gradientenrichtungsbild ableiten:

$$\text{dir}(x, y) = \arctan\left(\frac{I_{sobel,y}(x, y)}{I_{sobel,x}(x, y)}\right)$$

Dabei ist $\text{dir}(x,y)$ die Richtung der Änderung von dunkel nach hell in dem Punkt mit den Koordinaten (x, y) . Die Richtungswerte werden in 16 Richtungskanäle quantisiert. Jedem Bildpunkt kann dann ein Richtungskanal zugeordnet werden, der seine lokale Orientierung repräsentiert. Das so entstandene Bild wird im Folgenden Kanalwertbild genannt.

6.6.2.9.2 Merkmalsextraktion

Zur Merkmalsextraktion wird für die verschiedenen gesuchten Merkmale ein Template-Matching auf dem Kanalwertbild durchgeführt. Die verwendeten Gesichtsmerkmale sind Augen, Augenbrauen und Mund, da ihre Kanalwertmuster relativ personenunspezifisch sind. Um die Rechenzeit möglichst gering zu halten, wird die Merkmalsextraktion auf zwei Matching-Durchgänge reduziert, indem sowohl für das linke und das rechte Auge als auch für Mund und Augenbrauen jeweils ein gemeinsames Template verwendet wird.

Die Generierung der Templates erfolgte über von Hand extrahierte Trainingsdaten aus Bildern mit unterschiedlichen Personen und Aufnahmebedingungen.

Bei dem verwendeten Template-Matching-Verfahren wird das jeweilige Template über das Kanalwertbild bewegt und für jede Position ein Korrelationsmaß berechnet, das sich aus dem Grad der Übereinstimmung der Kanalwerte der einzelnen Pixel zusammensetzt. Je größer die Ähnlichkeit zwischen Template und Bild an einer Position ist desto höher ist der Korrelationswert. Aus den so erhaltenen Ergebnissen werden durch Ermittlung der lokalen Maxima Merkmalskandidaten für Augen und für Mund bzw. Augenbrauen extrahiert.

6.6.2.9.3 Merkmalskombination / Gesichtsdetektion

Die bei der Merkmalsextraktion ermittelten Kandidaten werden in ein gemeinsames Kandidatenbild übertragen, indem an ihre jeweiligen Positionen ein entsprechender Index gesetzt wird.

Auf diesem Kandidatenbild findet nun die eigentliche Gesichtssuche statt. Dazu wird eine Gesichtsmaske wie beim Template-Matching über das Kandidatenbild bewegt, welche die geometrischen Zwänge zwischen den Koordinaten von Augen, Mund und Augenbrauen berücksichtigt. Sie gibt vor, in welchem Rahmen die relativen Positionen zueinander variieren dürfen, um noch als Gesicht identifiziert zu werden.

Abbildung 50 zeigt die Auslegung der Maske für den angestrebten Anwendungsfall im Kraftfahrzeug. Die gegebenen Voraussetzungen sind hier ein Abstand zwischen Kopf und Kamera von 60 bis 100 Zentimetern und ein Kameraöffnungswinkel von zirka 35 Grad. In dieser Auslegung finden die unterschiedlichen Skalierungen durch die von Fahrer zu Fahrer variierenden Sitzpositionen Berücksichtigung. Darüber hinaus wird auch eine leichte Seitenansicht, bedingt durch die Kameraposition oder eine Kopfdrehung des Fahrers, berücksichtigt.



Abbildung 50: links: Auslegung der Gesichtsmaske (Bemaßung in Bildpunkten),
rechts: Ergebnis der Merkmalskombination

Jeder Merkmalskombination ist ein bestimmter Gütenwert zugeordnet, der repräsentativ für die Wahrscheinlichkeit ist, mit der es sich bei der jeweiligen Merkmalsanordnung um ein Gesicht handelt.

An jeder Maskenposition wird der zugehörige Gütenwert ermittelt und mit der zugehörigen Position festgehalten. Als Gesichtshypothese wird diejenige Maskenposition bestimmt, die unter den besten Kombinationen ist und zusätzlich die besten Matching-Güten der einzelnen Merkmale aufweist.

6.6.2.9.4 Verfeinerte Merkmalsextraktion

Ausgehend von der ermittelten Gesichtspose werden nun in einer verfeinerten Suche die Bezugsmerkmale für die Blickrichtungsschätzung extrahiert. Dies sind die Mundwinkel, die Augenwinkel, die Iris und die Nasenspitze.

Ausgehend von der ermittelten Gesichtspose werden nun in einer verfeinerten Suche die Bezugsmerkmale für die Blickrichtungsschätzung extrahiert. Dies sind die Mundwinkel, die Augenwinkel, die Iris und die Nasenspitze. Dazu wird mit Hilfe der Bildpositionen der Augen und des Mundes zunächst eine Lageprädiktion für die Mundwinkel durchgeführt. Das Ergebnis dient dann zur Eingrenzung des Suchbereichs bei der Segmentierung des Mundes, die auch durch das beschriebene Matching-Verfahren geschieht. Ist der Mund segmentiert werden durch integrale Projektion die Bildkoordinaten der Mundwinkel ermittelt [Stiefelhagen et al. 1997]. Ausgehend von den Augenpositionen und den Mundwinkeln wird nun der Suchbereich für die Nasenspitze festgelegt und auch dort ein Matching durchgeführt. Abschließend werden noch die Augenwinkel und die Pupille lokalisiert, wozu die Lage der Augenkandidaten zueinander dazu genutzt wird, die Suchbereiche zu platzieren.

6.6.2.9.5 Kopfposenschätzung

Die Berechnung der Kopfpose erfolgt mit Hilfe eines dreidimensionalen Kopfmodells, das aus der Anordnung der verwendeten Bezugsmerkmale besteht. Dies sind die vier Augenwinkel, die Mundwinkel und die Nasenspitze.

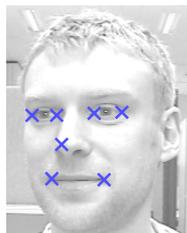


Abbildung 51: Bezugsmerkmale

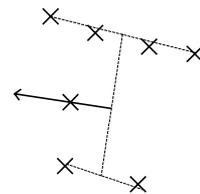


Abbildung 52: 3D-Kopfmodell

Ein spezielles Verfahren, der sogenannte POSIT-Algorithmus ([DeMenthon et al. 1996], [Gee & Cipolla 1994]) ermittelt diejenige Position und Ausrichtung im Raum des Modells, bei der die Projektion der Modell-Merkmale die beste Übereinstimmung mit den Bild-Merkmalen besteht.

6.6.2.9.6 Blickrichtungsschätzung

Grundlegend für die Blickrichtungsschätzung ist die Kopforientierung. Hierbei wird die Richtung des Normalenvektors des Gesichts als grobe Blickrichtung angenommen. Eine genauere Schätzung wird durch das Einbeziehen der Ausrichtung der Augen erlangt. Dazu wird die Lage der Iris innerhalb der Augenumrandung ermittelt und daraus auf einen Winkelversatz geschlossen, welcher mit der Kopforientierung verrechnet wird (siehe Abbildung 53).

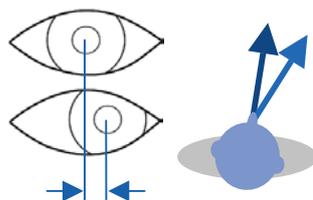


Abbildung 53: Augenausrichtung und Winkeloffset

6.6.2.9.7 Ergebnisse

Abbildung 54 zeigt für den Ausschnitt einer Videosequenz den Verlauf der horizontalen Kopforientierung und der horizontalen Blickrichtung. Der Proband lässt hierbei den Blick von 45° links zunächst gerade nach vorn (0°) und anschließend weiter auf 45° rechts schweifen.

Hierbei ist gut das typische Verhalten beim Anvisieren eines Objektes zu beobachten, dass die Augen voreilen und sich der Kopf erst mit einer Verzögerung in die Richtung dreht (Frame 200ff).

Deutlich werden auch die Grenzen des Systems bezüglich des unterstützten Winkelbereichs. Zwischen Frame 160 und 207, was in etwa einem Winkelbereich von $\pm 30^\circ$ entspricht, werden Ergebnisse erzielt, die zumindest qualitativ verwertbar sind.

Abbildung 55 zeigt für den Ausschnitt einer Videosequenz den Verlauf der Kopforientierung für die drei Freiheitsgrade. Der Proband wendet hierbei seinen Blick von vorne um 30° nach oben. Der Winkelverlauf um die x-Achse zeigt, dass auch diese Art von Kopfbewegungen erfasst werden.

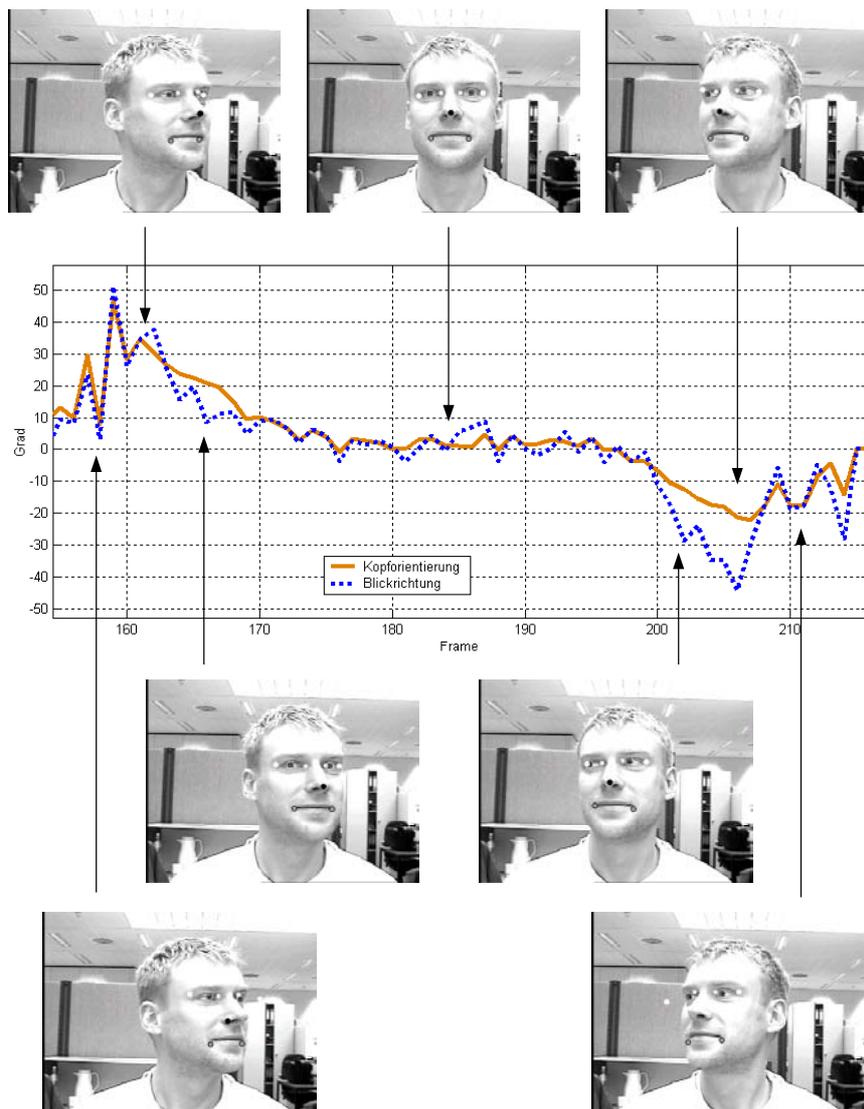
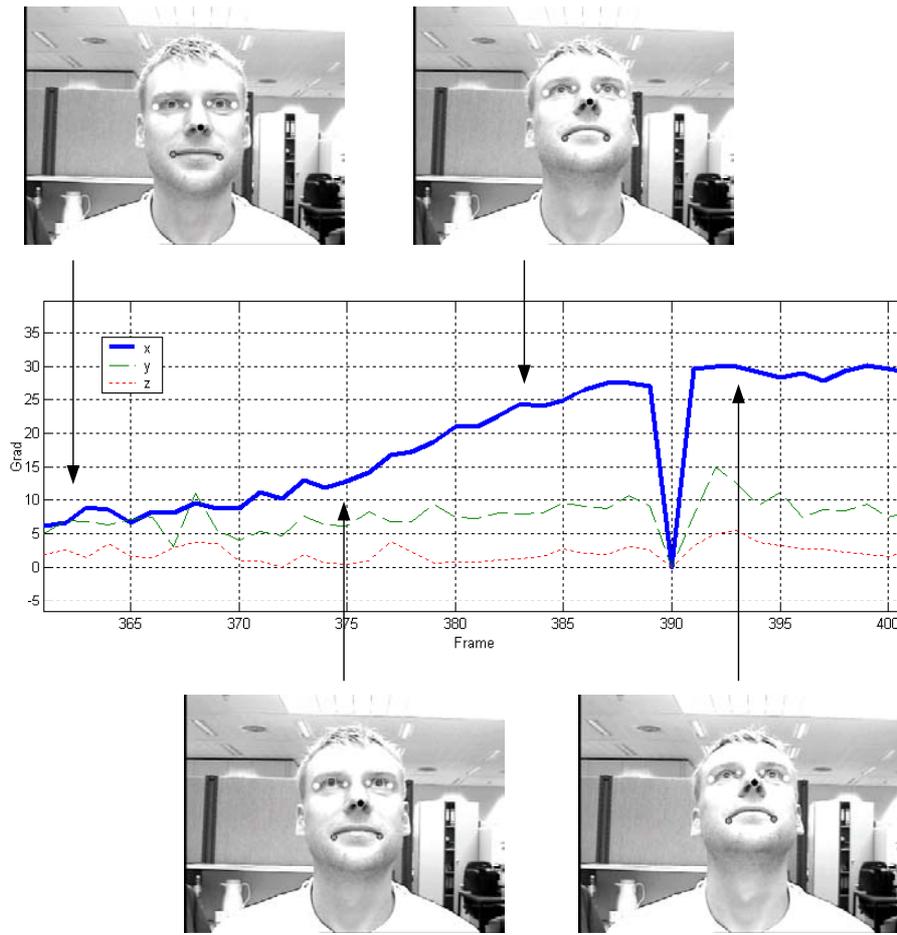


Abbildung 54: Verlauf der horizontalen Kopforientierung und der horizontalen Blickrichtung. Der Proband lässt den Blick von 45° links zunächst gerade nach vorn (0°) und anschließend weiter auf 45° rechts schweifen.



**Abbildung 55: Verlauf der Kopforientierung für die drei Freiheitsgrade.
Der Proband wendet seinen Blick von vorne um 30° nach oben.**

Einflussfaktoren auf das Ergebnis

Die erzielte Genauigkeit bei der Blickrichtungsschätzung wird von verschiedenen Faktoren beeinflusst, die im Folgenden aufgeführt werden.

Einflüsse der Merkmalsextraktion:

Besonders starken Einfluss hat die Qualität der Merkmalsextraktion, welche die Grundlage für alle weiteren Schritte darstellt. Wie oben beschrieben werden zur Bestimmung der Kopforientierung die ermittelten Bildkoordinaten von sieben Merkmalen verwendet. Oft werden einzelne Merkmale von Frame zu Frame nicht konsistent lokalisiert, sondern springen etwas in der Position. Dies wirkt sich dann mehr oder weniger stark auf das Ergebnis des POSIT-Algorithmus aus. Eine zentrale Bedeutung kommt hierbei der Nasenspitze zu, da sie als einziges Merkmal deutlich aus der Ebene, welche die übrigen Merkmale (nahezu) bilden, heraus ragt. Dadurch beeinflusst ihre Position die berechnete Kopforientierung am stärksten. Hinzu kommt, dass die Lokalisierung der Nasenspitze sehr schwierig ist, sobald der Betrachtungswinkel deutlich von der Frontalansicht abweicht. Zum Einen variiert die Gestalt der Nase von Person zu Person sehr stark, was bei einem seitlichen Betrachtungswinkel stärker zum Tragen kommt. Zum Anderen ändert sich ihr Aussehen deutlich bei wechselnder Beleuchtung. Außerdem ragt die Nasenspitze ab einem gewissen Drehwinkel des Kopfes in den Bildhintergrund hinein, wodurch weitere Störungen auftreten können.

Die Lokalisierung der Iris ist aufgrund der geringen Bildauflösung ebenfalls sehr störungsbehaftet. Fehler wirken sich über den auf ihr basierenden Winkeloffset direkt auf die berechnete Blickrichtung aus.

Die beschriebenen Faktoren legen, wie auch in Abbildung 54 zu erkennen ist, die Grenzen des Systems fest. Darüber hinaus führen sie dazu, dass die berechneten Winkel eine gewisse Inkonsistenz aufweisen. Die Schwankungen belaufen sich auf bis zu $\pm 3^\circ$ bei der Kopforientierung und bei bis zu $\pm 12^\circ$ bei der Blickrichtung.

Einflüsse des 3D-Kopfmodells:

Die Genauigkeit der berechneten Kopforientierung hängt insbesondere davon ab, wie genau das verwendete 3D-Kopfmodell mit den Abmessungen des Kopfes und der Anordnung der Merkmale übereinstimmt. Je nach Art der Abweichung kann dies das Ergebnis mehr oder weniger stark verfälschen. In der Regel belaufen sich die Abweichungen auf wenige Grad.

Einflüsse durch die Wahl der Parameter:

Das System weist zahlreiche Parameter auf, die zur Optimierung des Ergebnisses angepasst werden können. Dabei ist die Genauigkeit mit der Prozessgeschwindigkeit abzuwägen. Zum Beispiel kann für die jeweiligen Template-Matching-Durchgänge ein Raster gewählt werden. Je feiner die Rasterung desto genauer das Ergebnis. Ist das Raster grob gewählt, kann dies auch dazu führen, dass dadurch Merkmale nicht detektiert werden. Dies ist beispielsweise in Abbildung 55 bei Frame 390 zu beobachten. Durch den nach oben gerichteten Kopf erscheinen die Augen schmaler, wodurch sich für das Matching ein geringerer Spielraum ergibt. Hier tritt der ungünstige Fall ein, dass beide Augen quasi durch das Raster fallen und das Gesicht nicht detektiert wird.

Eine Rasterung wird lediglich bei der Lokalisierung der Primär-Kandidaten angewandt. Das gewählte Raster nimmt jedoch Einfluss auf die Positionen der angepassten Suchbereiche für das sekundäre Matching, da diese auf den Ergebnissen des primären Matchings beruhen. Ein Springen der Position eines Suchbereichs kann zu Inkonsistenzen bei der Lokalisierung der Bezugsmerkmale führen. Diese Einflussnahme wird durch die Anwendung eines Kalman-Filters auf die Koordinaten der Primär-Kandidaten abgeschwächt.

Auswertung

Es zeigt sich, dass das beschriebene Verfahren prinzipiell in der Lage ist, die Blickrichtung einer Person mit einer Genauigkeit von wenigen Grad zu ermitteln. In der Praxis stellt sich jedoch heraus, dass sich durch oben genannte Einflussfaktoren die Blickrichtung nur sehr vage ermitteln lässt. Besonders in den Grenzbereichen (ab ca. $\pm 25^\circ$) häufen sich die Schwankungen.

Während die Kopfposenschätzung in dem Winkelbereich von ± 25 Grad mit einer Genauigkeit von wenigen Grad verhältnismäßig verlässliche Werte liefert, werden die Ergebnisse bei stärkeren Kopfdrehungen zunehmend unzuverlässiger. Ist der Kopf weiter als 30 Grad zur Seite gedreht, sinkt die Detektionsrate, so dass nur noch vereinzelt das Gesicht richtig gefunden wird. Die Folge sind nicht mehr auswertbare Ergebnisse.

Die Prozessgeschwindigkeit beträgt bei einem Pentium III mit 1,4 GHz etwa 15 Bilder pro Sekunde. Diese hängt jedoch sehr grundlegend von den gewählten Parametern (Raster, Schwellwerte, etc.) ab.

6.6.2.10 Fazit

Die pragmatischste Lösung für eine qualitative Abschätzung der Kopforientierung stellt das in Abschnitt 6.6.2.8 beschriebene Blockmatching-Verfahren dar. Unter Laborbedingungen ist das System in der Lage, die horizontale Kopforientierung einer Person in fünf Stufen zu klassifizieren. Eine Alltagstauglichkeit ist aufgrund der nötigen Initialisierung und der mangelnden Robustheit bei Beleuchtungsschwankungen nicht gegeben. Darüber hinaus ist durch das geringe Auflösungsvermögen des Verfahrens das Anwendungsfeld sehr begrenzt.

Der in Abschnitt 6.6.2.9 beschriebene Ansatz auf Basis des Gradient-Orientations-Matchings bietet ein höheres Auflösungsvermögen. Hervorzuheben ist hier der geringe materielle Installationsaufwand und dass keine Initialisierung erforderlich ist. Die theoretisch erreichbare Genauigkeit des Ansatzes wird aufgrund der oben beschriebenen Einflussfaktoren nicht erreicht.

Aufgrund der Störanfälligkeit bei zum Beispiel teilweise verdecktem Gesicht ist auch dieses Verfahren nach dem momentanen Stand der Forschung nicht für einen Einsatz im Kfz geeignet.

7 Veröffentlichungen / Vorträge

Zusammen mit der Humboldt Universität wurden 3 Veröffentlichungen erstellt. Darüber hinaus gab es diverse Präsentationen des Kfz-Demonstrators für die Öffentlichkeit.

1. K. Polkehn, C. Preuschhof, H. Kussmann, „Assistenz und Automatisierung im Kfz - Simulationsstudien zur Unterstützung der Abstandshaltung“, DGLR, Fachausschuss T5.4 Anthropotechnik, 44. Fachausschusssitzung Anthropotechnik, 22./23. Oktober 2002, Langen, Deutschland
2. H. Kussmann, K. Polkehn, C. Preuschhof, „Assistance for car driver: prototyping via scenario and simulation“, HFES europe chapter, human factors in the age of virtual reality, 23.-25. Oktober 2002, Dortmund, Germany

3. K. Polkehn, C. Preuschhof, H. Kussmann, „Simulationsuntersuchungen zur Evaluation von Assistenzsystemen im KFZ“, 43. Kongreß der Deutschen Gesellschaft für Psychologie 2002, 22.-26.09.2002, Berlin, Deutschland
4. Demonstration des Kfz-Szenarios auf der Mensch und Computer vom 05. bis 08. März 2001 in Bad Honnef
5. Demonstration des Kfz-Szenarios auf der International Status Conference der MTI (Mensch Technik Interaktion) Leitprojekte vom BmBf am 26./27. Oktober 2001 in Saarbrücken
6. Demonstration des Kfz-Szenarios auf der Internationalen Funkausstellung vom 25. August bis zum 02. September 2001 in Berlin
7. Demonstration des Kfz-Szenarios auf der Internationalen Statustagung "Mensch-Technik-Interaktion" vom 03. bis 04. Juni 2003 in Berlin

8. Literatur

[Baumann 2001] A. Baumann, "Videobasierte Merkmalsextraktion und –kombination zur Gesichtslokalisierung", Diplomarbeit, Robert Bosch GmbH, 2001.

[Bradshaw & Jeffrey 1997] Bradshaw, Jeffrey M.: An Introduction to Software Agents, Veröffentlichtes Paper, 1997.

[Brunelli & Poggio 1993] R. Brunelli, T. Poggio, "Face recognition: Features versus templates", IEEE Trans. PAMI, 15(10):1042--1062, 1993.

[Castor 2003] <http://castor.exolab.org/>

[Console 2002] L. Console, S. Gioria, I. Lombardi, V. Surano, I. Torre, Adaptation and personalization on board cars: a framework and its application to tourist services. Department of Computer Sciences – University of Torino, 2002.

[DeMenthon et al. 1996] D. Oberkampff, D. DeMenthon, L. Davis, "Iterative Pose Estimation Using Coplanar Feature Points", in Comp. Vision and Image Understanding, vol.63, no.3, 1996.

[deWaard 1996] Dick de Waard, The Measurement of Drivers' Mental Workload. PhD Thesis, Groningen, Netherlands, 1996.

[Eberleh 1994] Eberleh, E.; Oberquelle, H.; Oppermann, R.: Einführung in die Software-Ergonomie – Gestaltung graphisch-interaktiver Systeme: Prinzipien, Werkzeuge, Lösungen, Walter de Gruyter, Berlin New York 1994.

[Färber 1998] Brigitte Färber & Berthold Färber, Expert evaluation of driver assistance systems with respect to traffic safety. 9th International Conference Road Safety in Europe, Bergisch Gladbach, Germany, Sept. 21st -23rd, 1998.

[FIPA 1997] Standardisierungswerk der Foundation for intelligent, physical Agents, FIPA, 1997

[Gee & Cipolla 1994] A. Gee, R. Cipolla, "Determining the gaze of faces in images", In Image and Vision Computing Volume 12 Number 10, Butterworth-Heinemann Ltd., pp. 639—647, 1994.

[Genesereth et al. 1994] Michael R. Genesereth, et.al., „Software Agents“, in Intelligent Agents, Communications of the ACM, July 1994, Vol. 37, No.7.

[GoF 1994] Gamma, Helm, Johnson, Vlissides: Design Patterns - Elements of Reusable Object Oriented Software. Addison-Wesley, 1994.

[Green 2000] Paul Green, Crashes Induced by Driver Information Systems and What Can Be Done to Reduce Them. Proceedings of Convergence 2000, Detroit, 16.- 18. October, 2000.

[Healey & Picard 2000] Jennifer Healey and Rosalind Picard, SmartCar: Detecting Driver Stress. Proceedings of ICPR'00, Barcelona, Spain, 2000.

[Herfet et al. 2000] Herfet, T., Tödtmann, T. (2000). Distributed Multimedia Applications based on EMBASSI, 19. Jahrestagung der FK TG, Braunschweig, Mai 2000.

[Herfet et al. 2001] Herfet, T., Kirste, T., Schnaider, M. (2001). EMBASSI – Multimodal Assistance for Infotainment & Service Infrastrukturen. C&G (Computer & Graphics) Vol. 25, 4/2001.

[Hering 1999] K. Hering, S. Thiel, E. Stephan und W. Hussy, Das Kölner Verfahren zur vergleichenden Erfassung der kognitiven Beanspruchung im Straßenverkehr (K-VEBIS). In: Informations- und Assistenzsysteme im Auto benutzergerecht gestalten, Berichte der Bundesanstalt für Straßenwesen, Mensch und Sicherheit Heft M116, S. 16-24, Bergisch Gladbach, 1999.

[Hildebrand 2000] Hildebrand, A.(2000). EMBASSI: Electronic Multimedia and Service Assistance. Intelligent Interactive Assistance and Mobile Computing, IMC 2000, Rostock, Germany

[Jennings et al. 1997] Jennings, Nicholas R., Wooldridge, Michael J.: Agent Technology. 1. Auflage, Springer Verlag 1997.

[JNI 2003] Java Native Interface, <http://java.sun.com/products/jdk/1.2/docs/guide/jni/>

[Labrou & Finin 1997] Yannis Labrou and Tim Finin, „A Proposal for a new KQML Specification“, Technical Report TR CS-97-03, February 3, 1997.

[Lädke & Huber 1998] M. Lädke und W. Huber, Erweiterte mobile Datenerfassung im Kfz zur Fahrersofortwarnung. In: Elektronik im Kraftfahrzeug, VDI Berichte 1415, S. 857-868, Baden-Baden, 1998.

[MPEG 2003] <http://www.chiariglione.org/mpeg/index.htm>

[NMEA 2003] NMEA 0183 Specification, <http://www.nmea.org/pub/index.html>

[Pfafferott 1992] I. Pfafferott, Risikoanpassung - werden technische Sicherheitspotentiale ausreichend genutzt? In: Faktor Mensch im Straßenverkehr, Berichte der Bundesanstalt für Straßenwesen, Mensch und Sicherheit Heft M9, S. 14-19, Bergisch Gladbach, 1992.

[Quick 2000] <http://jxquick.sourceforge.net/quick3/>

[Reiser et al. 1978] B.J. Reiser, S.M. Kosslyn, T.M. Ball, Visual images preserve metric spatial information: Evidence from studies of image scanning. Journal of Experimental Psychology: Human Perception & Performance, Vol. 4 (1), pp. 47-60, 1978.

[Ross 1997] T. Ross, G. Burnett, R. Graham, A. May, M. Ashby, State-of-the-Art Review: Human Machine Interface Integration for Driver Systems. INTEGRATE Project, Deliverable No. 1, December 1997.

[Schreiber et al. 1990] K. Schreiber, G. Kebeck, S. Figge, The relationship between physical and imagined distances: An experiment on the foundations of mental training. Gestalt Theory, Vol 12(2), pp. 96-115, Jun 1990, 1990.

[Sonnerat 2002] D. Sonnerat, N. Tricot, JC. Popieul, Driver's environment identification using automatic classification methods. Active Safety Application. IV'2002 Proceedings, IEEE Intelligent Vehicle Symposium, Versailles, France, 2002.

[Stiefelhagen et al. 1997] R. Stiefelhagen, J. Yang, A. Waibel, "A model-based gaze tracking system", Int. Journal of Artificial Intelligence Tools, Vol. 6, No. 2, February, pp. 193 - 209, 1997

[Sun Data Binding 2003] <http://java.sun.com/xml/jaxp/dist/1.0.1/docs/binding/DataBinding.html>

[Timpe 1998] Timpe, K.-P. (1998). Unterstützungssysteme als interdisziplinäre Herausforderung - Einführung in die Tagung „Wohin führen Unterstützungssysteme?“. In H.-P. Willumeit & H. Kolrep (Hrsg.), Wohin führen Unterstützungssysteme? Entscheidungshilfe und Assistenz in Mensch-Maschine-Systemen, 2. Berliner Werkstatt Mensch-Maschine-Systeme, Berlin 7. bis 9. Oktober 1997 (S. 1-20). Sinzheim: Pro Universitate Verlag.

[W3C XML 2000] <http://www.w3.org/XML/>