TECHNISCHE UNIVERSITÄT MÜNCHEN
Lehrstuhl für Mensch-Maschine-Kommunikation

Multimodale Mensch-Roboter-Interaktion für Ambient Assisted Living

Tobias F. Rehrl

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzende: Univ.-Prof. Dr.-Ing. Sandra Hirche
Prüfer der Dissertation: 1. Univ.-Prof. Dr.-Ing. habil. Gerhard Rigoll
2. Univ.-Prof. Dr.-Ing. Horst-Michael Groß
(Technische Universität Ilmenau)

Die Dissertation wurde am 17. April 2013 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 8. Oktober 2013 angenommen.

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

ISBN 978-3-8439-1372-0

Sternstr. 18, 80538 München
Tel.: 089/66060798
www.dr.hut-verlag.de

1. Auflage 2013

Zusammenfassung

Der demografische Wandel verändert die deutsche sowie die europäische Gesellschaft, daher gewinnen besonders Aspekte der Mensch-Maschine-Kommunikation für die Entwicklung, Gestaltung und Umsetzung von Informations- und Kommunikationstechnologien immer mehr an Bedeutung, damit auch ältere Menschen diese Technologien einfach und intuitiv bedienen können. Unter dem Begriff *Ambient Assisted Living* können verschiedene Bemühungen zusammengefasst werden, die darauf abzielen, durch den Einsatz von modernen Informations- und Kommunikationstechnologien die Auswirkungen des demografischen Wandels abzumildern.

Die nonverbalen Komponenten in der Mensch-Maschine-Interaktion bilden den ersten Ansatzpunkt in dieser Arbeit, um die Kommunikation zwischen Mensch und Maschine einfach und intuitiv zu gestalten. Hierbei werden sowohl Gesten als auch der Ausdruck von Emotionen mittels Mimik betrachtet. Bildbasierte Verfolgungsmethoden bilden einen zentralen Bestandteil von videobasierten Systemen, die vermehrten Einsatz in Alltagssituationen finden. Auch im Bereich der Gesten- und Mimikerkennung werden bildbasierte Verfolgungsmethoden eingesetzt, um echtzeitfähige Verarbeitungen zu verwirklichen. In dieser Arbeit werden die bildbasierten Verfolgungsmethoden mithilfe von Graphischen Modellen dargestellt sowie erweitert. Mithilfe eines Graphischen Modells wird eine bildbasierte Verfolgungsmethode mit einer Gestenklassifikation kombiniert, wodurch es möglich ist, die Bewegung einer Hand zu verfolgen und gleichzeitig die dargestellte Geste zu klassifizieren. Die Inferenzverfahren der Graphischen Modelle werden für die Anpassung der bildbasierten Verfolgungsmethoden genutzt. Auf diese Weise wird sowohl das Bewegungsmodell als auch die Partikel-Bewertungsfunktion adaptiv an die gegenwärtige Beobachtungssequenz angepasst.

Abschließend wird in dieser Arbeit ein Spieleszenario als eine beispielhafte Anwendung für *Ambient Assisted Living* vorgestellt. In diesem Spieleszenario werden die entwickelten Methoden zur Gesten- und Mimikerkennung aufgegriffen und kommen auf einer Roboterplattform zum Einsatz. Diese Roboterplattform ist aufgrund ihrer technischen Ausstattung zu einer multimodalen Mensch-Roboter-Interaktion befähigt. Somit stehen für das Spieleszenario mehrere Modalitäten zur Eingabe sowie zur Ausgabe zur Verfügung.

Abstract

The demographic change affects the German as well as the European society, therefore, aspects of human-machine communication are becoming increasingly important for the development, design and implementation of information and communication technologies, so that elderly people can use these technologies in a simple and intuitive way. Under the term *Ambient Assisted Living* various efforts can be summarized, which attempt to mitigate the impact of the demographic change by applying modern information and communication technologies.

The non-verbal components in the human-machine interaction are the first points of departure in this thesis to make the communication between humans and machines easy and intuitive, therefore both gestures and the expression of emotions via facial expressions are considered. Image-based tracking methods are a central component of video-based systems, which are applied increasingly in everyday situations. In the field of gesture and facial expression recognition image-based tracking methods are used to achieve real-time processing. In this thesis graphical models are applied to realize and augment image-based tracking methods. An image-based tracking method is combined with a classification step using a graphical model, thus it is possible to track the movement of a hand, while classifying the shown hand gesture. The inference process of graphical models can be used for the adjustment of image-based tracking methods, thus it is possible to adapt the motion model as well as the particle evaluation function according to the current observation sequence.

Finally, in this thesis a game scenario is presented as an example application for *Ambient Assisted Living*. In this scenario, the developed methods for the gesture and facial expression recognition are picked up and are applied on a robotic platform. The technical equipment of the robotic platform provides for a multimodal human-robot interaction, thus for playing the game several modalities for input and output are available.

Inhaltsverzeichnis

Kapitel 1

Einleitung

1.1 Motivation

Der demografische Wandel verändert die Zusammensetzung sowie die Altersstruktur der Bevölkerung. Dieser Prozess der Veränderung ist sowohl für die europäische [1, 2] als auch – insbesondere – für die deutsche Gesellschaft [3, 4, 5] von großer Bedeutung. Insgesamt lassen sich drei wesentliche Einflussfaktoren (Fertilitätsrate, Lebenserwartung, Wanderungssaldo aus Zu- und Fortzügen) bestimmen. Die ausgelöste Veränderung betrifft in direkter Art und Weise die Bevölkerungszahl, den Altersaufbau sowie das Durchschnittsalter. Die Gesamtbevölkerung wird laut [2] für Deutschland, basierend auf den Zahlen von 2008 im Jahre 2050, um 9,4 % abgenommen haben, während es für die 27 Länder der Europäischen Union eine Zunahme von 4,0 % geben wird. Hierbei ist aber zu beachten, dass ab dem Jahr 2040 auch die Bevölkerung dieser 27 Länder wieder abnehmen wird [6]. Der Anteil der über 65-Jährigen wird sich für Deutschland von 20,6 % im Jahr 2010 auf 32,3 % im Jahr 2050 erhöhen, für die 27 Länder der Europäischen Union gibt es für die gleichen Jahre einen Anstieg von 17,4 % (2010) auf 23,7 % (2050) [6]. Das Durchschnittsalter wird sich für Deutschland laut [7] von 43,8 Jahre für das Jahr 2009 auf 49,9 Jahre für das Jahr 2060 erhöhen. Für die 27 Länder der Europäischen Union steigt das Durchschnittsalter laut [1] von 40,4 Jahre für das Jahr 2008 auf 47,9 Jahre für das Jahr 2060.

Aus diesen direkten Veränderungen, basierend auf dem demografischen Wandel, ergeben sich indirekte Auswirkungen, die soziale sowie ökonomische Komponenten aufweisen und die alternde Gesellschaft vor Probleme und Herausforderungen stellen werden. Beispielsweise ergeben sich aufgrund der Abnahme der erwerbstätigen Bevölkerung Probleme am Arbeitsmarkt. Der erhöhte Anteil von alten Menschen an der Gesamtbevölkerung bringt stärkere Belastungen für den Sozialstaat, insbesondere für das Gesundheitssystem.

Um diesen Veränderungen und Herausforderungen, die sich für eine alternde Ge-

sellschaft ergeben, zu begegnen, bedarf es laut [8] Bemühungen sowohl in technologischer als auch sozioökonomischer Hinsicht. Diese Bemühungen werden unter dem Begriff *Ambient Assisted Living* gebündelt. *Ambient Assisted Living* setzt unter anderem auf den Einsatz von Informations- und Kommunikationstechnologien, die eine intelligente unterstützende Umgebung schaffen sollen, um älteren und behinderten Menschen in Alltagssituationen helfen zu können [9].

Generell können Informations- und Kommunikationstechnologien in vielfältiger Art und Weise eingesetzt werden, um die Auswirkungen des demografischen Wandels abzumildern. Im Folgenden werden Aspekte der Mensch-Maschine-Kommunikation beim Einsatz von Informations- und Kommunikationstechnologien dargestellt, wobei nonverbale Formen der Mensch-Maschine-Interaktion (Gesten, Mimik), zugrunde liegende notwendige Techniken (bildbasierte Verfolgungsmethoden) sowie eine exemplarische Anwendung auf einer Roboterplattform Beachtung finden.

1.2 Gliederung dieser Arbeit

Das Thema dieser Arbeit ist es, unterschiedliche Aspekte der Mensch-Maschine-Kommunikation für *Ambient Assisted Living* zu untersuchen. Hierbei werden unterschiedliche Verfahren entwickelt, realisiert und evaluiert. In Kapitel 2 und Kapitel 3 werden sowohl die wesentlichen Grundlagen von *Ambient Assisted Living* als auch Aspekte der Mustererkennung, die in mehreren Kapiteln dieser Arbeit Anwendung finden, vorgestellt. Die anschließenden zwei Kapitel beschreiben Verfahren zur nonverbalen Mensch-Maschine-Interaktion, wobei sowohl Gesten- als auch Mimikerkennungssysteme betrachtet werden. Bildbasierte Verfolgungsmethoden, die anhand von Graphischen Modellen realisiert werden, werden in den Kapitel 6 betrachtet. Abschließend wird ein Spieleszenario als eine exemplarische Anwendung für *Ambient Assisted Living* vorgestellt. In diesem Spieleszenario werden die entwickelten Methoden zur Gesten- und Mimikerkennung aufgegriffen und kommen auf einer Roboterplattform zum Einsatz.

In **Kapitel 2** wird zunächst der Begriff *Ambient Assisted Living* erläutert, anschließend wird die wesentliche Ursache – der demografische Wandel – für den Einsatz von unterschiedlichen *Ambient Assisted Living*-Technologien genauer beleuchtet. Um bedarfsgerechte *Ambient Assisted Living*-Anwendungen realisieren zu können, werden zuerst die Anforderungen und Bedürfnisse der potentiellen *Ambient Assisted Living*-Nutzer bestimmt, anschließend werden unterschiedliche Technologien (Smart Home, gesundheitsbezogene Anwendungen, Robotik) für *Ambient Assisted Living*-Lösungen vorgestellt. Das Kapitel endet mit einer kurzen Vorstellung des ALIAS-Projektes, in dem einzelne Teile dieser Arbeit entstanden sind.

In **Kapitel 3** werden zunächst grundlegende Eigenschaften von Graphischen Modellen erläutert, hierbei werden kurz die unterschiedlichen Arten von Graphischen Mo-

dellen – Bayes'sche Netze, Hidden-Markov-Modelle, *Markov Random Fields* – vorgestellt. Anschließend werden Verfahren zur Inferenz sowie für das Lernen der Belegung der Zufallsvariablen innerhalb von Graphischen Modellen beschrieben. Des Weiteren werden Grundlagen zur Durchführung und Bewertung von Experimenten erläutert, die in den Kapiteln 4, 5 und 6 zum Einsatz kommen.

In **Kapitel 4** werden drei Verfahren zur Gestenerkennung vorgestellt, denn mit ihnen lassen sich sowohl statische und dynamische Handgesten als auch Kopfgesten erkennen. Alle Verfahren sind in eine einheitliche Architektur eingebunden, die auf zwei unterschiedlichen Robotersystemen zum Einsatz gekommen ist. Aufgrund der Einbindung in diese Architektur ist es sowohl möglich die Gesten echtzeitfähig zu erkennen als auch das Problem des Auffindens des Beginns einer Geste zu behandeln. Die erzielten Klassifikationsergebnisse können über die Architektur einem Dialogsystem, das die Interaktion zwischen Mensch und Maschine steuert, bereitgestellt werden.

In **Kapitel 5** werden zwei Ansätze zur Erkennung der sechs universellen Gesichtsausdrücke zur Emotionsdarstellung vorgestellt. Diese Ansätze werden mithilfe von Graphischen Modellen realisiert. Aufgrund der Tatsache, dass die zeitliche Komponente der Mimik einen unterschiedlichen Beitrag zur Erkennungsleistung hat, werden anhand von Merkmalsselektionsverfahren die relevanten Merkmale bestimmt. Hierbei kommen sequentielle Verfahren zum Einsatz sowie zwei Verfahren, die die Merkmale, basierend auf der Kullback-Leibler-Divergenz auswählen.

In **Kapitel 6** werden bildbasierte Verfolgungsmethoden vorgestellt, die Graphische Modelle zur Realisierung der Methoden verwenden. Ausgehend von einer Erweiterung des CONDENSATION-Algorithmus um einen Klassifikationsschritt, werden sowohl das Bewegungsmodell als auch die Gewichtungsfunktion für die Partikel mithilfe der Inferenz von Graphischen Modellen erweitert, um die zukünftigen Bearbeitungsschritte adaptiv an die gegenwärtige Beobachtung anzupassen. In einem ganzheitlichen Graphischen Modell werden die vorgestellten Verfahren für das adaptive Bewegungsmodell sowie für die adaptive Gewichtungsfunktion miteinander kombiniert.

In **Kapitel 7** wird ein exemplarisches Spieleszenario als eine mögliche *Ambient Assisted Living*-Anwendung vorgestellt, hierbei werden die entwickelten Methoden zur Gesten- und Mimikerkennung aufgegriffen. Das Spieleszenario ist auf einer mobilen Roboterplattform, die große Ähnlichkeiten mit der Plattform des ALIAS-Projektes aufweist, integriert. Die Interaktion im Spieleszenario ist von multimodaler Natur, da unterschiedliche Modalitäten sowohl zur Eingabe als auch zur Ausgabe verwendet werden können.

Kapitel 2

Ambient Assisted Living

Inhaltsangabe

In diesem Kapitel soll zunächst ein Überblick über die Begrifflichkeit von Ambient Assisted Living (AAL) gegeben werden. Danach werden kurz die Ursachen erläutert, warum es in Zukunft eines verstärkten Einsatzes von AAL-Anwendungen bedarf. Diese Anwendungen beziehen sich auf die AAL-Nutzer, die bestimmte Anforderungen und Bedürfnisse an die verwendeten Technologien haben. Es sind unterschiedliche AAL-Technologien denkbar, um die AAL-Nutzer in ihrem Alltag zu unterstützen. Exemplarisch werden Lösungen für Smart Home, gesundheitsbezogene Anwendungen und Robotik beschrieben. Das Kapitel schließt mit der Vorstellung des ALIAS-Projektes, bei dem Teile dieser Arbeit entstanden sind.

2.1 Einführung

Ein breites Spektrum an Themen, ausgehend von den Ursachen (Alterung und Schrumpfung der Gesellschaft, steigende Pflegekosten etc.) sowie Lösungen (z. B. Smart Home, gesundheitsbezogene Anwendungen, Robotik etc.), werden anhand von *Ambient Assisted Living* (AAL) angesprochen. AAL wird manchmal als eine spezielle nutzerorientierte Form von *Ambient Intelligence* (AmI) [8] angesehen. Unten werden unterschiedliche Aspekte von AAL vorgestellt, weitere Informationen zu AAL können in [8, 10, 9, 11, 12] gefunden werden.

2.1.1 Begriffsklärung

Das Konzept von AAL bezieht sich auf den Einsatz von Produkten und Dienstleistungen aus der Informations- und Kommunikationstechnologie (IKT), um eine intelligente unterstützende Umgebung zu schaffen, die älteren und behinderten Menschen in ihrem Alltag assistiert [9]. Ein besonderes Augenmerk liegt hierbei auf der häuslichen Umgebung, da diese im besonderen Maße bedeutend für das Wohlbefinden von älteren und behinderten Menschen ist [8]. Die verwendeten IKT-Systeme sind in der Lage, in einer personalisierten Art und Weise mit dem AAL-Nutzer[1] zu interagieren, wobei die Systeme den Nutzer in seinen alltäglichen Handlungen unterstützen sollen, um somit die Lebensqualität des Nutzers zu verbessern. Diese IKT-Systeme sollen sich dabei an die Bedürfnisse und Anforderungen des Nutzers anpassen und nicht umgekehrt. Generell sollen die AAL-Produkte und -Dienstleistungen ein breites Spektrum an Funktionalitäten abdecken, die die Bereiche Gesundheit, Sicherheit, Unabhängigkeit, Mobilität und Soziale Einbindung umfassen [8]. Die Ziele von AAL können gemäß [10, 8] folgendermaßen zusammengefasst werden: die Zeit zu verlängern, die ältere Menschen alleine zu Hause leben, Steigerung von Autonomie und Selbstbewusstsein, Unterstützung bei Alltagsaufgaben, Förderung eines gesunden Alterns, Unterstützung von Menschen mit Behinderungen, soziale Einbindung von älteren und behinderten Menschen, Unterstützung von Angehörigen und pflegenden Personen mittels neuer Produkte und Dienstleistungen, Förderung von Sicherheit und Optimierung der Ressourcennutzung in einer alternden Gesellschaft. Zur Verwirklichung dieser Ziele gibt es unterschiedliche Initiativen wie beispielsweise auf nationaler Ebene durch das Bundesministerium für Bildung und Forschung [13] sowie auf europäischer Ebene durch das *Ambient Assisted Living Joint Programme* [14].

[1] Anmerkung: Im folgenden Text wird lediglich aus Gründen der Vereinfachung die männliche Form verwendet, obwohl der Bezug auf beide Geschlechter gemeint ist.

2.1.2 Ursache für den Einsatz von AAL-Technologien

Dem Einsatz von AAL-Technologien liegen zwei wesentliche Zielsetzungen zugrunde [8]: Zum einen soll die Lebensqualität von älteren und behinderten Menschen mithilfe von Produkten und Dienstleistungen aus der IKT gesteigert werden, damit diese längere Zeit zu Hause leben können, zum anderen, verbunden mit der Tatsache eines längeren Aufenthalts von AAL-Nutzern in den eigenen vier Wänden, sollen die Kosten für den Sozialstaat reduziert werden. Diese Zielsetzungen liegen eigentlich im demografischen Wandel begründet, da dieser eine wesentliche Ursache für den Einsatz von AAL-Technologien für ältere Menschen ist.

Der demografische Wandel, die Veränderung des Aufbaus der Bevölkerung, wird im Wesentlichen von drei Faktoren (Fertilitätsrate, Lebenserwartung, Wanderungssaldo aus Zu- und Fortzügen) bestimmt. Annahmen für diese drei Faktoren sind bei Vorausberechnungen der Bevölkerungsentwicklung zu berücksichtigen.

Im Zuge der Untersuchungen von Eurostat [6] im Jahre 2008 für die demografische Entwicklung auf europäischer Ebene – Eurostat Bevölkerungsprognosen 2008 (engl. *Eurostat Population Projections* 2008, EUROPOP2008) – sind für die 27 Länder der Europäischen Union (EU)[1] und Deutschland folgende Entwicklungen (siehe Tabelle 2.1) für die Fertilitätsrate sowie Lebenserwartung für die Jahre 2008 und 2050 vorausberechnet worden [2].

		2008	2050
Fertilitätsrate	Deutschland	1,34	1,49
	EU-27	1,54	1,65
Lebenserwartung Frauen	Deutschland	82,6 Jahre	88,0 Jahre
	EU-27	82,1 Jahre	87,9 Jahre
Lebenserwartung Männer	Deutschland	77,3 Jahre	83,6 Jahre
	EU-27	76,1 Jahre	83,2 Jahre

Tabelle 2.1: Die Fertilitätsrate und Lebenserwartung (Frauen und Männer) für Deutschland und die EU-27, basierend auf den Daten von EUROPOP2008

Es zeigt sich, dass sich die Lebenserwartung in Zukunft weiter erhöhen wird, des Weiteren ist die Fertilitätsrate trotz eines leichten Anstiegs von der bestandserhaltenden Fertilitätsrate von 2,1 [8] weiterhin entfernt.

Eine wichtige Kennzahl, die die Veränderung der Altersstruktur der Bevölkerung beschreibt, ist der sogenannte Altenquotient [5], der den Anteil der Bevölkerung im Rentenalter (65 Jahre und älter) dem Anteil der erwerbstätigen Bevölkerung (20 – 64 Jahre) gegenüberstellt. Daneben kann der prozentuale Anteil unterschiedlicher Alters-

[1] Im Folgenden werden die 27 Länder der EU mit EU-27 abgekürzt. Die EU-27 umfasst die EU-Staaten, die ab 2007 bis einschließlich Juni 2013 Mitglieder sind.

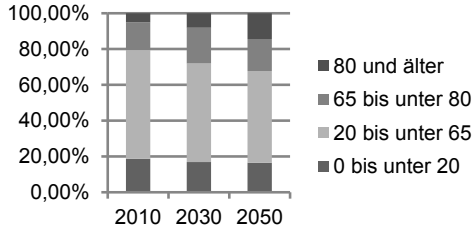
gruppen an der Gesamtbevölkerung Aufschluss über die Entwicklung der Altersstruktur geben.

Basierend auf den Daten von Eurostat [6] im Jahre 2010 für die demografische Entwicklung auf europäischer Ebene – Eurostat Bevölkerungsprognosen 2010 (engl. *Eurostat Population Projections* 2010, EUROPOP2010) – ergibt sich für die Jahre 2010, 2030 und 2050 folgender Altenquotient[1] bzw. folgende Altersstruktur der Bevölkerung, siehe unten stehende Tabellen (Tabelle 2.2 und Tabelle 2.3).

		2010	2030	2050
Altenquotient	Deutschland	0,34	0,51	0,63
	EU-27	0,28	0,42	0,55

Tabelle 2.2: Der Altenquotient für Deutschland und die EU-27 für die Jahre 2010, 2030 und 2050, basierend auf den Daten von EUROPOP2010

(a) Altersaufbau Deutschland



(b) Altersaufbau EU-27

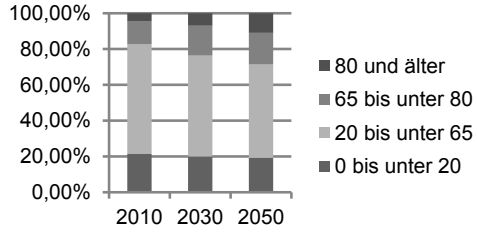


Tabelle 2.3: Altersaufbau für Deutschland (links) und die EU-27 (rechts) für die Jahre 2010, 2030 und 2050, basierend auf den Daten von EUROPOP2010

Neben dem Anstieg des Altenquotienten und den damit verbundenen Belastungen für den Sozialstaat ist auch der Anstieg des Anteils der sehr alten Bevölkerung (80 Jahre und älter) zu erwähnen. Diese Bevölkerungsgruppe hat in der Regel mit einer erhöhten Anzahl an Erkrankungen sowie chronischen Erkrankungen zu kämpfen (Multimorbidität), was sich in einer erhöhten Pflegequote für diese Altersgruppe niederschlägt [15].

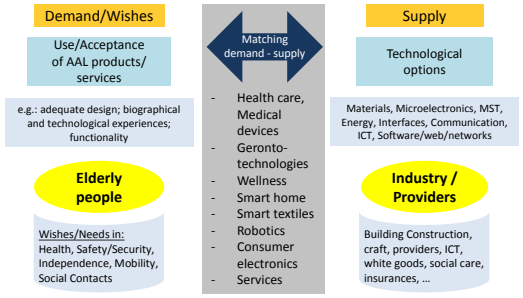
2.2 AAL-Anwendungen

Der Bedarf an AAL-Technologien ist im Grunde dem demografischen Wandel geschuldet, daher wird im Folgenden ein besonderer Fokus auf die älteren AAL-Nutzer gelegt.

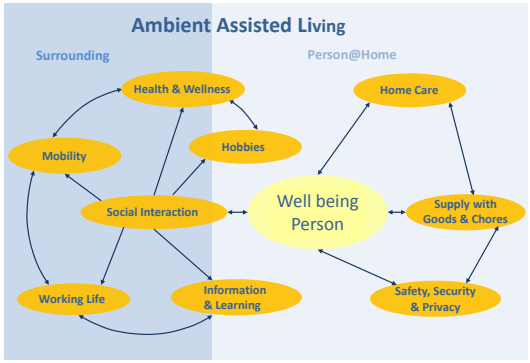
[1] Eine Berücksichtigung der Anhebung des Renteneintrittsalters auf 67 Jahre in Deutschland beim Altenquotient war bei den hier präsentierten Zahlen aufgrund der Gestaltung der Daten nicht möglich.

Generell gibt es ein breites Spektrum an technischen Systemen, z. B. Assistenzsystemen, Assistiven Technologien (definiert in [16]), die für den Einsatz in AAL denkbar sind. Zur Bestimmung der richtigen technischen Systeme müssen zunächst die Bedürfnisse und Anforderungen von älteren Menschen für die Unterstützung in ihrem Alltag erfasst bzw. beschrieben werden.

Im Zuge der AAL-Forschung sind zwei Modelle vorgestellt worden (siehe Abbildung 2.1), die die Bedürfnisse und Anforderungen von älteren Menschen an AAL-Lösungen beschreiben sollen: *AAL innovation model* [8], *AAL multi-factor model* [12].



(a) *AAL innovation model* nach [10]



(b) *AAL multi-factor model* nach [12]

Abbildung 2.1: Entwickelte AAL-Modelle zur Beschreibung der Bedürfnisse und Anforderungen von älteren Menschen an AAL-Lösungen

Beide Modelle beschreiben prinzipiell den Bedarf von älteren Menschen an AAL-Technologien, dennoch bedienen sich die Modelle unterschiedlicher Ansatzpunkte zur Problembeschreibung. Das *AAL innovation model* bedient sich einer Beschreibung mittels des Konzeptes von Angebot und Nachfrage und versucht sowohl die Nutzerseite mit ihren Bedürfnissen als auch die zugehörigen technischen Lösungen zu beschreiben. Das *AAL multi-factor model* hingegen beschreibt die Umgebung der Anwendung (häuslich, soziales Umfeld) und setzt dabei den Schwerpunkt auf die Nutzerseite und lässt die technische Umsetzung außen vor.

Die Bedürfnisse von älteren Personen zur Unterstützung in ihrem Alltag sowie die dazugehörigen technischen Hilfsmittel umfassen ein weites Spektrum, weshalb nachfolgend nur wesentliche Punkte vorgestellt werden.

2.2.1 Anforderungen und Bedürfnisse von AAL-Nutzern

Die Bedürfnisse von AAL-Nutzern und deren Anforderungen an technische Systeme folgen keinem standardisierten Schema. Die Gruppe potentieller AAL-Nutzer ist von heterogener Natur, somit müssen auch die vielseitigen AAL-Lösungen unterschiedlichen Bedürfnissen Rechnung tragen [8].

Mithilfe der zwei vorgestellten Modelle können folgende wichtige Bedürfnisbereiche[1] für ältere Menschen identifiziert werden: Gesundheit, Sicherheit, Mobilität, Unabhängigkeit und Kommunikation. Daneben zählen auch Hobbys sowie Zugang zu Informationen und Bildung zu wichtigen Aktivitäten von älteren Menschen. Die Grenzen zwischen den einzelnen Bereichen sind nicht klar, es gibt fließende Übergänge beispielsweise im Bereich von Gesundheit und Sicherheit, Alarmsysteme (passiv oder aktiv) können im Fall des Auftretens gesundheitlicher Probleme sowohl dem Bereich Sicherheit als auch Gesundheit zugeordnet werden.

Gesundheit: Die Weltgesundheitsorganisation (engl. *World Health Organization*, WHO) definiert Gesundheit als „einen Zustand des vollständigen körperlichen, geistigen und sozialen Wohlbefindens und nicht nur die Abwesenheit von Krankheit oder Gebrechen“ [17]. Die gesundheitlichen Beeinträchtigungen nehmen mit dem Alter zu, dabei steigt sowohl die Anzahl der Erkrankten als auch die Problematik der Erkrankung an sich [18]. Die Entwicklung des Gesundheitszustandes nach Altersgruppen für Deutschland ist in Tabelle 2.4 dargestellt.

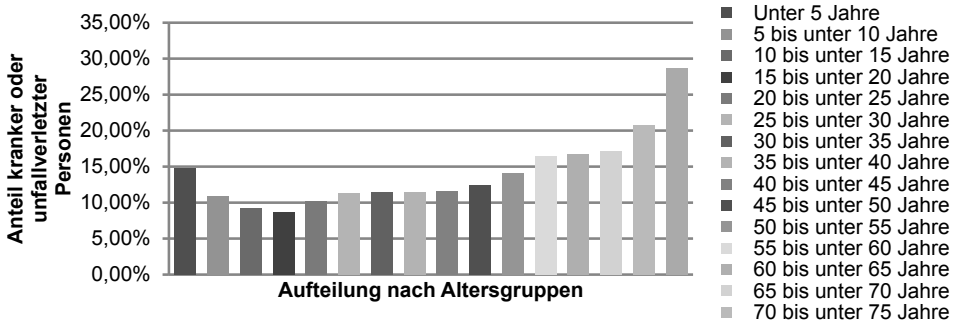


Tabelle 2.4: Anteil der kranken oder unfallverletzten Personen der deutschen Bevölkerung nach Altersgruppen aufgeschlüsselt (entnommen aus [19, 11]). Daten basieren auf dem Mikrozensus – Fragen zur Gesundheit, Statistisches Bundesamt, Zweigstelle Bonn

Der Tabelle 2.4 ist ein Anstieg der erkrankten bzw. unfallverletzten Personen insbesondere bei älteren Personen zu entnehmen, somit wird ersichtlich, dass Gesundheit für ältere Menschen eine wesentliche Rolle spielt.

Sicherheit: Der Bereich der Sicherheit ist eng mit der Gesundheit wie auch mit der häuslichen Umgebung verbunden. Die häusliche Umgebung spielt eine entscheidende Rolle für ältere Menschen aufgrund der Tatsache, dass ältere Menschen im-

[1] Die Einteilung in die Bedürfnisbereiche ist angelehnt an [11], somit werden auch Quellen von [11] für die folgende Argumentation aufgegriffen.

mer mehr Zeit zuhause verbringen [20].[1] Basierend auf [21], können folgende IKT-Anwendungen im Bereich von Gesundheit identifiziert werden, die für ältere Menschen von Bedeutung sind: aktive/ passive Alarmsysteme, Fernzugriff für Verwandte und Pflegepersonal, weiterentwickelte Videotelefoniekonzepte. Die Bereiche, in denen ältere Menschen ein Sicherheitsbedürfnis in den eigenen vier Wänden haben, umfassen unter anderem: Einbrecheralarm, Rauchmelder, automatische Nachtbeleuchtung, automatisches Ausschalten von Herden [8].

Mobilität: Die Mobilität älterer Menschen setzt sich aus zwei Komponenten zusammen: physische Mobilität und Verkehrsmobilität [11]. Die Verkehrsmobilität setzt natürlich ein gewisses Maß an physischer Mobilität voraus. Die physische Mobilität nimmt im Alter ab, dabei sind vor allem Frauen stärker betroffen als Männer [19]. Die Tendenz der Mobilitätseinschränkung mit steigendem Alter ist aus Tabelle 2.5 gut erkenntlich.

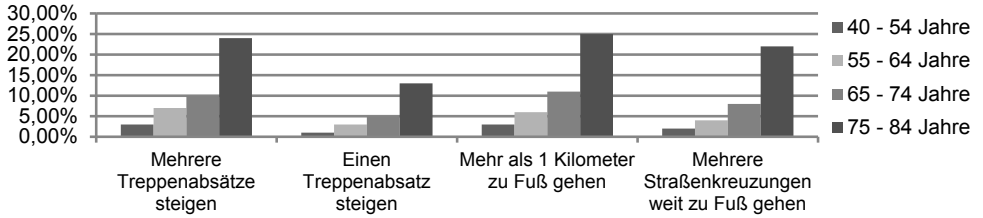


Tabelle 2.5: Mobilitätseinschränkungen im Alter entnommen aus [19, 11], basierend auf den Daten des Alterssurvey 2002

Generell spielt Mobilität für ältere Menschen eine wichtige Rolle [11], daher können im Bereich von Verkehrsmobilität sowohl IKT-Systeme helfen als auch sogenannte „assistive devices“ wie Rollatoren im Bereich von physischer Mobilität.

Unabhängigkeit: Der Begriff *Unabhängigkeit* kann einfach definiert werden, indem man *Unabhängigkeit* als die Abwesenheit von *Abhängigkeit* definiert, dabei geht es insbesondere um die Vermeidung von anderen Personen abhängig zu sein. Es hat sich gezeigt, dass diese Definition nicht ausreichend ist, daher ist in [22] ein zweidimensionales Modell von Unabhängigkeit definiert worden, das die Begriffe *Unabhängigkeit* und *Abhängigkeit* in Beziehung setzt. Dieses Modell ist in der Lage, unterschiedliche kulturelle Aspekte abzubilden. Beispielsweise spielt im amerikanischen und britischen Kulturraum die Selbstständigkeit (Unabhängigkeit) für ältere Menschen eine größere Rolle, weshalb es Abhängigkeit von anderen Personen möglichst zu vermeiden gilt, auch wenn das Einschränkungen für die älteren Menschen bedeutet.

[1] In der Quelle wird von außerhalb der Wohnung verbrachter Zeit gesprochen. Dies lässt aber natürlich Rückschlüsse auf die in der Wohnung verbrachte Zeit zu.

Kommunikation und Information: Die älteren Menschen nutzen Kommunikationsmittel, um Beziehungen und Kontakte zu pflegen, wobei das Telefon die wichtigste Rolle spielt [11]. Generell gibt es eine Vielzahl von modernen IKT-Systemen (z. B. Mobiltelefone, Smartphones etc.), die eine starke Verbreitung finden. Die älteren Menschen greifen diese modernen Technologien nicht in dem Umfang auf wie die Durchschnittsbevölkerung, da das Interesse an neuen Technologien abnimmt, dennoch gibt es in Bezug auf Akzeptanz keine Unterschiede aufgrund des Alters [8]. Des Weiteren trägt auch die Heterogenität der Gruppe von älteren Menschen dazu bei, dass die Nutzung von neuen IKT-Technologien unterschiedlich ausfällt [8].

Die Bedürfnisse von älteren Menschen bzw. potentiellen Nutzern von AAL-Technologien umfassen ein breites Spektrum, bei dem auch die Nutzerakzeptanz eine wichtige Rolle spielt, sprich welche Anforderungen und Wünsche haben ältere Menschen an die Technologie selbst, damit sie diese in ihrem Alltag einsetzen. Ein interdisziplinäres Forschungsteam untersuchte im Forschungsprojekt *Seniorengerechte Technik im häuslichen Alltag* (sentha) [23] die Technikakzeptanz von älteren Menschen, Ausgangsgrundlage war eine bundesweite Befragung von 1417 älteren Menschen ab dem 55. Lebensjahr. Eine Auswertung der gewonnenen Daten aus den Befragungen ergab, dass sich zwei Dimensionen bei der Technikakzeptanz unterscheiden lassen, wobei eine Dimension eher rationale Aspekte abbildet, während die zweite Dimension emotional-affektive Bewertungen abbildet. Somit ergeben sich vier unterschiedliche Typen der Technikakzeptanz: Befürworter, Rationalisierer, Skeptiker und Kritiker [11].

Generell haben die Ergebnisse der Untersuchungen im Rahmen des sentha-Projektes gezeigt, dass die älteren Menschen unterschiedliche Einstellungen zu neuen technischen Lösungen haben – hierbei lassen sich weder absolute Technikfeindlichkeit noch unkritische Technikbegeisterung erkennen. Die Technikakzeptanz von älteren Menschen hängt davon ab, inwieweit die neue Technologie einen Mehrwert für die ältere Person verspricht, daneben spielen auch Faktoren wie Geschlecht, Bildungsniveau und Einkommen eine Rolle [11].

2.2.2 Technologien für AAL-Lösungen

Generell sollen, laut AAL-Definition (siehe Abschnitt 2.1.1), IKTs verwendet werden, um älteren und behinderten Menschen zu helfen. Unter dem Schlagwort IKTs wird allgemein die Fähigkeit beschrieben, dass Menschen Informationen untereinander mittels technischer Hilfsmittel austauschen können, dabei werden die Bereiche wie Telefonie, Rundfunkmedien, alle Arten von Verarbeitungen und Übermittlungen von Audio- und Videodaten erfasst.

Ein breites Spektrum an Technologien für unterschiedliche AAL-Lösungen sind

denkbar, in [8] sind folgende wesentliche Bereiche für AAL-Anwendungen und -Produkte identifiziert worden: Gesundheitswesen, medizinische Geräte, Gerontotechnik, Wellness, Dienstleistungen, Smart Home, intelligente Textilien, Robotik, Unterhaltungselektronik etc. Im Folgenden werden nur Anwendungen betrachtet, die im AAL-Bereich von besonderer Bedeutung sind und eine Überschneidung mit Teilen dieser Arbeit aufweisen. Ein wichtiger Aspekt ist die Tatsache, dass ältere Menschen immer mehr Zeit zuhause verbringen [20], weshalb die Ausstattung des Eigenheims mit IKTs ein sinnvoller Ansatz ist. Des Weiteren spielt die Gesundheit eine wesentliche Rolle für ältere Menschen (siehe Tabelle 2.4). Abschließend wird der Einsatz von Robotik für AAL-Anwendungen beleuchtet, da auch im Projekt *Adaptable Ambient LIving ASsistant* (ALIAS), das im Abschnitt 2.3 vorgestellt wird, eine Roboterplattform als ein IKT-System in verschiedenen AAL-Anwendungen dienen soll.

Smart Home: Unter dem Begriff *Smart Home* können unterschiedliche Projekte zusammengefasst werden, die sich mit der Vernetzung des Eigenheims befassen. Eines der ersten Projekte mit *Smart Home* war das *TRON Intelligent House*, das 1989 fertiggestellt worden war [24]. Daneben gibt es noch weitere Arbeiten zu diesem Forschungsgebiet, eine Übersicht findet sich in [25]. Weitere Projekte, die sich mit *Smart Home*-Technologien im Bereich von AAL auseinandersetzen, sind in [26] zu finden. Nachfolgend werden einige Beispiele kurz vorgestellt: *Assisted Living in Kaiserslautern* [27], hierbei wurde an vier Standorten in Rheinland-Pfalz an seniorengerechten Wohnungen geforscht. Im Projekt *SmartHome in Paderborn* [28] sollen durch eine universelle Fernbedienung Funktionen des Hauses einfach bedient werden können. *Assisted Living in Karlsruhe* ist ein Projekt, das seit 2009 läuft: Hierbei ist eine Wohnung älterer Menschen mit zusätzlichen Sensoren ausgestattet worden, mit denen Stürze erkannt werden sollen. Neben Projekten in Deutschland gibt es auch aktuelle *Smart Home*-Einrichtungen in anderen Ländern, eine Auflistung findet sich in [26].

Gesundheitsbezogene Anwendungen: Im Alter nehmen sowohl die somatischen Erkrankungen als auch die psychischen Erkrankungen (z. B. Depressionen) zu [11]. Des Weiteren gibt es auch einen Anstieg von Demenz (siehe [29]), hierbei wird mit *MCI* (engl. *mild cognitive impairment*) eine milde kognitive Beeinträchtigung beschrieben, die über den normalen Alterungsprozess hinausgeht. Es gibt eine Vielzahl an technischen Lösungen, die für ältere Menschen von Bedeutung sein können. Im Folgenden werden einige Beispiele vorgestellt, wichtig ist dabei, dass diese Systeme die besonderen Einschränkungen von älteren Menschen wie reduzierte kognitive Fähigkeiten, Verlust der Sehkraft, Hörverlust und die Unerfahrenheit mit interaktiven Systemen berücksichtigen sollten. In [30] wird das *HEARTRONIC*-Projekt vorgestellt, bei dem mittels eines tragbaren Systems der Zustand des Herzens von Patienten, die Risikofaktoren für kardiovaskuläre Erkrankungen haben, kontinuierlich überwacht und

analysiert werden soll. Ein *Remote Assistance System* wird in [31] vorgestellt, mit dem drahtlos unterschiedliche Sensordaten (z. B. Pulsmesser, Temperatursensor, Lagesensor etc.) erfasst werden können, um automatisch die Aktivitäten sowie die Gesundheit von älteren Menschen oder Patienten zu überwachen und gegebenenfalls nach Hilfe zu rufen. *MobiSense* [32] ist ein weiteres mobiles Gesundheits-Monitoring-System, das in der Lage ist, Haltungen von Menschen (z. B. Sitzen, Stehen etc.) mittels eines regelbasierten heuristischen Klassifikationsschemas zu erkennen. In [33, 34, 35, 36] werden verschiedene Forschungs- und Entwicklungsarbeiten auf dem Gebiet von tragbaren Systemen zur Gesundheitsüberwachung vorgestellt.

Robotik: Das Gebiet der Robotik deckt eine breite Palette von unterschiedlichen Arten von Robotern ab, diese kommen im Bereich von industriellen Anwendungen in Fabriken vor (z. B. Einsatz in der Automobilzulieferer-Industrie [37]), sowie in der häuslichen Umgebung (z. B. als Staubsaugerroboter [38]). In der Regel ist eine exakte Definition sowie eine Kategorisierung der Roboter schwer vorzunehmen. Dennoch lässt sich zwischen Industrie- und Servicerobotik unterscheiden [39], wobei auch hier die Grenzen immer mehr verschwimmen und Serviceroboter in einem industriellen Umfeld eingesetzt werden sollen [40]. Ein aktueller Überblick über den Einsatz von Servicerobotik im Bereich von AAL wird in [41] gegeben. Unterschiedliche Bereiche können von Servicerobotik in AAL angesprochen werden, dazu zählen beispielsweise: Haushaltstätigkeiten, Unterhaltung, Gesundheit und Pflege. Natürlich gibt es aber Robotersysteme, die mehreren Bereichen zugeordnet werden können, beispielsweise kann die Roboterplattform Nao [42] sowohl zur Unterhaltung dienen, während sie im Projekt *ROBO M.D.* [43] eher gesundheitsbezogen agiert. In diesem Projekt überwacht und erkennt die Roboterplattform kritische Situationen bei Menschen mit Herzkrankheiten und soll bei Bedarf, schnell Angehörige oder medizinisches Personal verständigen. Anhand dieses Beispiels lässt sich zeigen, dass eine einfache Kategorisierung der Robotersysteme nach unterschiedlichen Bereichen in der Regel nicht möglich ist, da die technische Grundausstattung diverse Einsatzgebiete ermöglicht. Unten werden kurz einige Beispiele vorgestellt, um einen Eindruck zu vermitteln, welche unterschiedliche Aktivitäten im Bereich der Servicerobotik geschehen.

In den Bereich von Haushaltstätigkeiten können sowohl einfache Systeme fallen, die sich um verschiedene Reinigungstätigkeiten (z. B. Böden wischen [44], Staub saugen [38] und Rasen mähen [45]) kümmern, als auch komplexe Systeme, die als eine Art Butler (z. B. Care-O-bot® 3 [46], ASTROMOBILE [47]) eine Vielzahl an unterschiedlichen Tätigkeiten erledigen sollen. Da die häusliche Umgebung für Menschen gemacht ist, ist der Einsatz von sogenannten humanoiden Robotern, die in ihrem Erscheinungsbild dem Mensch ähneln, sinnvoll. Es gibt zahlreiche Projekte, die humanoide Roboter entwickelt haben, hier werden nur kurz einige bekannte Beispiele aufgezählt: Asimo von der Firma Honda [48], Justin vom Deutschen Zentrum für Luft-

und Raumfahrt [49], CHARLI [50] von der VirginiaTech Universität, daneben finden sich in [51] weitere Informationen und Beispiele zu humanoiden Robotern.

Ein bekanntes Beispiel für den Einsatz im Unterhaltungsbereich ist beispielsweise der von Sony bis 2006 produzierte Roboterhund AIBO [52]. Ein anderer Roboter für den Unterhaltungsbereich ist PaPeRo [53]. Dieser von der Firma NEC entwickelte Roboter kann in vielfältiger Art und Weise (Spracherkennung, Sprachsynthese, Gesichtserkennung etc.) mit Menschen interagieren. Ein weiteres bekanntes Beispiel für einen Roboter aus dem Unterhaltungsbereich ist der bereits oben erwähnte Nao vom Roboterhersteller *Aldebaran Robotics* [42], dieser Roboter hat eine humanoide Form.

Gesundheit und Pflege sind für AAL-Robotikanwendungen von großer Bedeutung. Zum einen können die Robotersysteme helfen, die Gesundheit von älteren Menschen zu überwachen, dabei können diese Systeme kritische Situationen erkennen und Hilfe verständigen, wie im Projekt *ROBO M.D.* [43]. In der Pflege lassen sich unterschiedliche Bereiche für den Einsatz von Roboter identifizieren, sie können sowohl das Pflegepersonal entlasten, z. B. im WimiCare-Projekt [54] bei der Erledigung einfacher Tätigkeiten (Botendienste, Getränkeverteilung), als auch Unterstützung beim Heben von Patienten aus Rollstühlen oder Betten (siehe z. B. [55]) geben. Bekanntestes Beispiel für den Einsatz von Robotern in der Pflege dürfte die Roboterrobbe Paro [56] sein. Paro wird bei Demenzpatienten [57] sowie zur Linderung von Einsamkeit eingesetzt. Im AAL-Projekt DOMEO [58] wurden zwei Robotersysteme entwickelt: *RobuWalker* soll physische Assistenz leisten, während *RobuMate* den Nutzer sozial einbinden soll (Verbindung zu Angehörigen oder medizinischer Zentrale).

2.3 Das ALIAS-Projekt

Einige Teile dieser Arbeit sind während des Projektes *Adaptable Ambient LIving ASsistant* (ALIAS) entstanden. Im Folgenden werden kurz die Zielsetzung sowie die Umsetzung des Projektes beschrieben, weitere Informationen zum Projekt bzw. zu der Roboterplattform finden sich in [59, 60].

2.3.1 Zielsetzung

Das Projekt ALIAS verfolgt das Ziel, eine Roboterplattform (siehe Abbildung 2.2) zu entwickeln, auf der moderne IKTs integriert sind. Diese IKTs sollen die soziale Einbindung und Teilhabe des Nutzers fördern, um somit dessen Lebensqualität zu verbessern. Die Roboterplattform soll dabei nicht die Mensch-Mensch-Kontakte ersetzen, sondern vielmehr diese fördern und verbessern. Zur Interaktion mit dem Menschen ist die Roboterplattform mit einem Touchscreen, einer Spracherkennung und einem *Brain Computer Interface* (BCI) ausgestattet. Das ALIAS-System soll in der Lage sein, natürlich gesprochene Sprache zu verstehen und somit auf einfache Art und Weise mit dem

Menschen interagieren zu können. Die Roboterplattform kann aufgrund ihrer Mobilität von sich aus mit dem Nutzer interagieren (proaktives Verhalten), des Weiteren kann die Mobilität der Roboterplattform hilfreich sein, wenn externe Personen (Verwandte, Vertraute) mittels Telepräsenz mit dem ALIAS-Nutzer in Verbindung treten wollen.

Neben der Entwicklung einer mit IKTs ausgestatteten Roboterplattform liegt ein weiterer Schwerpunkt des Projektes auf der Untersuchung der Akzeptanz von sozialen Robotersystemen, die mit Menschen in einer häuslichen Umgebung interagieren sollen. Um dieser Untersuchung Rechnung zu tragen, spielt die Einbindung von potentiellen Nutzern im Entwicklungsprozess eine entscheidende Rolle.

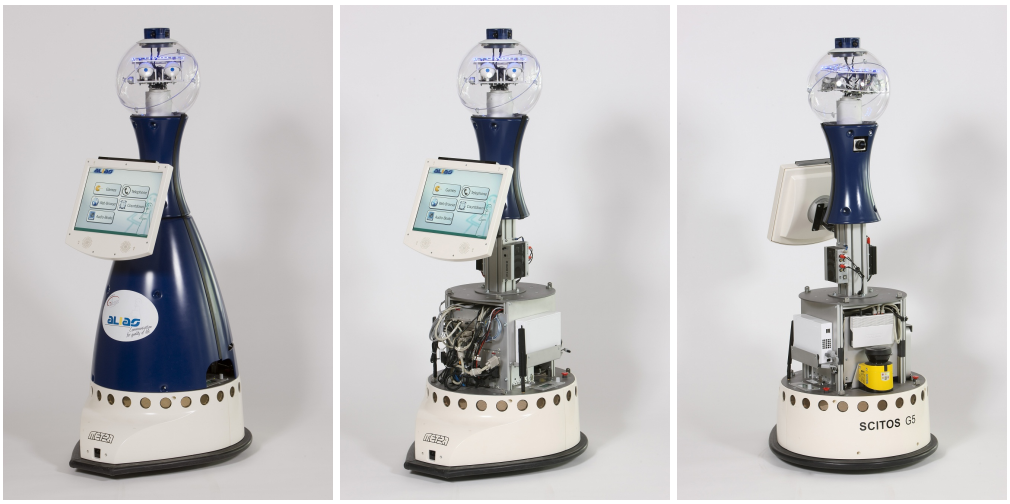


Abbildung 2.2: Roboterplattform des ALIAS-Projektes (Fotos ©CoTeSys/Kurt Fuchs)

2.3.2 Umsetzung

Der ALIAS-Entwicklungsprozess ist in insgesamt vier Phasen eingeteilt. In Phase 1 werden, ausgehend von der aktuellen Roboterplattform, die Wünsche und Bedürfnisse von älteren Menschen an eine mobile Roboterplattform erfasst. Ausgehend von der bestehenden Roboterplattform und den auf der Roboterplattform umsetzbaren Wünschen und Bedürfnissen wird ein erstes lauffähiges System entwickelt und in einem Feldversuch getestet (Phase 2). Basierend auf den gewonnenen Erkenntnissen des Feldversuchs aus Phase 2 wird ein verbessertes System entwickelt. Zudem werden Wünsche und Bedürfnisse, die nicht in Phase 2 umgesetzt worden sind, dem neuen System hinzugefügt. Dieses neue System wird abermals mittels Feldversuch getestet (Phase 3), die gewonnenen Erkenntnisse vom zweiten Feldversuch werden in das finale Prototypensystem integriert (Phase 4). Generell spielt die Einbeziehung der Nutzer im ALI-

AS-Entwicklungsprozess eine wesentliche Rolle, deswegen wird auch neben den Feldversuchen die Meinung und Rückmeldung von älteren Menschen eingeholt.

Die Einteilung der entwickelten Funktionen im ALIAS-Projekt kann in drei Kategorien erfolgen: *Kommunikation*, *Unterhaltung* und *Sicherheit*. Abbildung 2.3 zeigt die graphische Umsetzung der Benutzeroberfläche [1].

(a) Hauptmenü (b) Kommunikationsmenü

(c) Unterhaltungsmenü (d) Sicherheitsmenü

Abbildung 2.3: Nutzerschnittstelle des ALIAS-Projektes

Die Kategorie *Kommunikation* umfasst eine Skype-basierte Telefonfunktionalität, Internet Browser sowie Veranstaltungen. Im Bereich der Veranstaltungen ist es möglich, sich Informationen und Ausführungen über zukünftige und vergangene Ereignisse (Konzert, Aufführungen etc.) anzeigen zu lassen.

Die Kategorie *Unterhaltung* bietet die Möglichkeit des Fernsehens, dafür ist die Roboterplattform mit einem zusätzlichen DVB-T-Empfangsmodul ausgestattet worden. Als weitere Unterhaltungsmöglichkeiten können Hörbücher wiedergeben sowie Spiele gespielt werden. Neben einer kleinen Anzahl von Spielen, die direkt in die

[1] Das Fraunhofer-Institut für Digitale Medientechnologie war im ALIAS-Projekt verantwortlich für die Gestaltung der grafischen Benutzeroberfläche.

ALIAS-Benutzerschnittstelle integriert worden sind, können auch mit der Nintendo Wii-Spielekonsole unterschiedliche Spiele gespielt werden.

Die Kategorie *Sicherheit* bittet drei verschiedene Funktionalitäten. Mit der Notruffunktion kann nach einem Hilferuf des Nutzers eine bestimmte Skype-Verbindung aufgebaut werden. Im Punkt Navigation lässt sich der Roboter auf eine bestimmte vordefinierte Position fahren. Die Video-Demo-Funktion bittet die Möglichkeit, ein kurzes Video mit der Beschreibung der wesentlichen ALIAS-Funktionalitäten abzuspielen.

Für die Entwicklung des ALIAS-Systems sind unterschiedliche Szenarien, die verschiedene Funktionalitäten und Aspekte der Roboterplattform ansprechen sollen, entwickelt worden. Die Bestimmung der Szenarien erfolgte in Abstimmung zwischen den technischen Partnern sowie den Partnern der Nutzereinbindung. Am Projektende gab es fünf unterschiedliche Szenarien: Veranstaltungsszenario (engl. *event scenario*), Führungsszenario (engl. *guiding scenario*), Steuerung durch einen externen Nutzer (engl. *remote control by secondary user*), BCI-Szenario (engl. *BCI scenario*) und ein Selbsterleben-Szenario (engl. *self-experience scenario*). Im Folgenden werden die fünf Szenarien kurz vorgestellt.

- Veranstaltungsszenario: In diesem Szenario soll der Nutzer sich sowohl über vergangene als auch zukünftige Ereignisse, Veranstaltungen etc. mithilfe des ALIAS-Systems informieren können.

- Führungsszenario: Dieses Szenario beschreibt die Möglichkeit, dass der Roboter seinen Nutzer an ein bestimmten Ort führen kann. Für den Einsatz in dunklen Situationen (z. B. in der Nacht) hat der Roboter eine zusätzliche Beleuchtungseinheit für den Boden.

- Steuerung durch externen Nutzer: Die Navigation des ALIAS-Roboters lässt sich von extern steuern, somit ist es möglich mit dem Roboter die Wohnung seines älteren Nutzers zu überprüfen bzw. mit dem Nutzer in Kontakt zu treten.

- BCI-Szenario: Dieses Szenario handelt von der Steuerung verschiedener Funktionalitäten des ALIAS-Systems mithilfe des BCI-Systems, somit ist es beispielsweise möglich die Vorlesefunktion für Hörbücher über das BCI zu steuern.

- Selbsterleben-Szenario: In diesem Szenario soll der Nutzer die verschiedenen Funktionalitäten des ALIAS-Systems in einer spielerischen Art und Weise ausprobieren bzw. testen, somit umfasst dieses Szenario die Funktionalitäten (z. B. Spiele), für die es kein eigenes Szenario gibt.

Die Definition von konkreten Szenarien hat sich bei der Entwicklung des ALIAS-Systems als hilfreich erwiesen, da anhand von konkreten Problembeschreibungen und Zielsetzungen schnell und einfach verschiedene Herausforderungen im Entwicklungsprozess identifiziert werden konnten.

Kapitel 3

Grundlagen

Inhaltsangabe

In diesem Kapitel werden wesentliche Grundlagen vorgestellt, die sich in mehreren Kapiteln dieser Arbeit wiederfinden. Der Großteil dieses Kapitels widmet sich der Beschreibung von Graphischen Modellen (GMs). Diese GMs finden Einsatz in den Klassifizierungsaufgaben von Kapitel 4 und Kapitel 5. Des Weiteren wird im Kapitel 6 der Inferenzprozess von GMs für bildbasierte Verfolgungsmethoden verwendet. Neben den GMs werden auch kurz Grundlagen für die Experimente, die in dieser Arbeit durchgeführt worden sind, vorgestellt. Hierbei geht es um die Kreuzvalidierung sowie den Vergleich von Erkennungssystemen anhand von statistischer Signifikanz.

3.1 Graphische Modelle

Graphische Modelle (GMs) repräsentieren in einer einfachen Art und Weise die Zufallsvariablen (ZVs) eines Systems oder eines Prozesses und deren wechselseitige Abhängigkeiten, daher finden GMs Verwendung in vielen verschiedenen Forschungsbereichen wie Bild-, Sprachverarbeitung, medizinischen Anwendungen, Verkehrsanalyse, Roboterlokalisierung etc. Zur Darstellung eines Systems oder eines Prozesses kombinieren GMs Wahrscheinlichkeitstheorie mit Graphentheorie, um Abhängigkeiten zwischen unterschiedlichen ZVs anhand eines Graphen darzustellen. Im Folgenden werden grundlegende Eigenschaften von GMs vorgestellt, da sie in mehreren Kapiteln dieser Arbeit verwendet werden. Grundlagen zu Wahrscheinlichkeitstheorie und Graphentheorie finden sich in Anhang A bzw. in Anhang B, ausführlichere Beschreibungen von GMs können in [61, 62, 63, 64, 65, 66] gefunden werden.

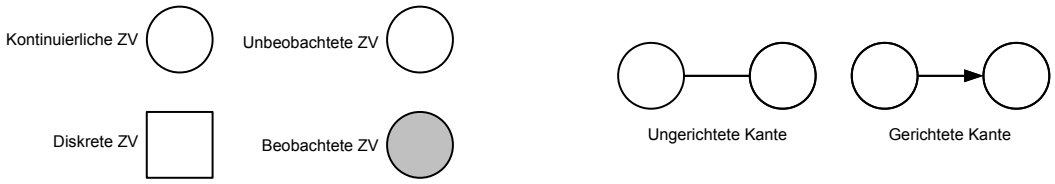
3.1.1 Einführung

Einige zentrale Aspekte von GMs reichen weit in die Vergangenheit zurück: Graphentheorie mit Königsberger Brückenproblem von Leonhard Euler 1736 [67], Bayes Theorem 1763 [68], Axiomatisierung der Wahrscheinlichkeitstheorie 1933 von Andrei Kolmogorow [69]. Dennoch ist der Bedarf zur Darstellung von Problemen mittels GMs von der Künstlichen Intelligenz (KI)-Forschung bei der Gestaltung von sogenannten Expertensystemen (z. B. MYCIN-System [70]) geweckt worden, dabei spielt Pearl mit [71] eine wichtige Rolle. Der Umgang mit Unsicherheiten (engl. *uncertainties*) bescherte den Expertensystemen trotz Erweiterungen einige Probleme, und mittels des Bayes Theorems lassen sich Unsicherheiten gut in die Wissensrevision (engl. *Belief Revision*) einbringen. Laut [66] beruht der Erfolg von den GMs auf zwei Tatsachen: erstens die Arbeiten [72, 71] legen die theoretischen Grundlagen für die Nachrichtenübertragung in GMs, zweitens, die erfolgreichen Umsetzungen dieser Forschungsergebnisse in Expertensysteme wie z. B. das *Pathfinder* Expertensystem [73].

Aufgrund ihrer flexiblen Anwendungsmöglichkeiten finden GMs vermehrten Einsatz, in [65] werden GMs in unterschiedlichen Bereichen der Mensch-Maschine-Kommunikation eingesetzt. In [74] werden GMs im Bereich der Sprachverarbeitung angewandt, daneben werden in [75] GMs im Bereich der Bildverarbeitung zur Segmentierung verwendet. Des Weiteren gibt es medizinische Anwendungen [73, 76, 77], bei denen unterschiedliche GM-Systeme für die Erstellung von Diagnosen entwickelt worden sind. Daneben finden sind auch noch Beispiele aus der Verkehrsanalyse [78] sowie der Genetik [79], somit zeigt sich, dass GMs das Potential haben Beschreibungen und Lösungen für unterschiedliche Problemstellungen von verschiedenen Forschungsbereichen zu liefern.

In der Struktur von GMs können wichtige Eigenschaften in einer einfachen Art und Weise repräsentiert werden. ZVs werden als Knoten im Graphen dargestellt, daneben

kann man anhand der Form und Erscheinung des Knotens folgende Informationen direkt aus dem Graphen ableiten: diskrete oder kontinuierliche Verteilung der ZV, beobachtete oder unbeobachtete ZV. In der Regel wird eine kontinuierliche Verteilung der ZV mit einem Kreis dargestellt, während eine diskrete Verteilung durch eine quadratische Knotenform repräsentiert wird. Neben der Verteilung ist auch ein Rückschluss anhand der Darstellung im Graphen möglich, ob die ZV beobachtet wird oder nicht. Beobachtete ZVs werden in der Regel mit einem grauschattierten Knoten dargestellt, unbeobachtete ZVs bleiben dagegen unschattiert. Des Weiteren enthält auch die Gestaltung der Kanten Informationen über die ZVs. Die Kanten beschreiben die Beziehungen unter den ZVs, dabei gibt es gerichtete Kanten, die eine kausale Abhängigkeit ausdrücken, während eine ungerichtete Kante eine wechselseitige Abhängigkeit beschreibt. Die unterschiedlichen Darstellungen von Knoten (siehe Abbildung 3.1(a)) und Kanten (siehe Abbildung 3.1(b)) werden in der unten stehenden Abbildung 3.1 kurz zusammenfassend vorgestellt.



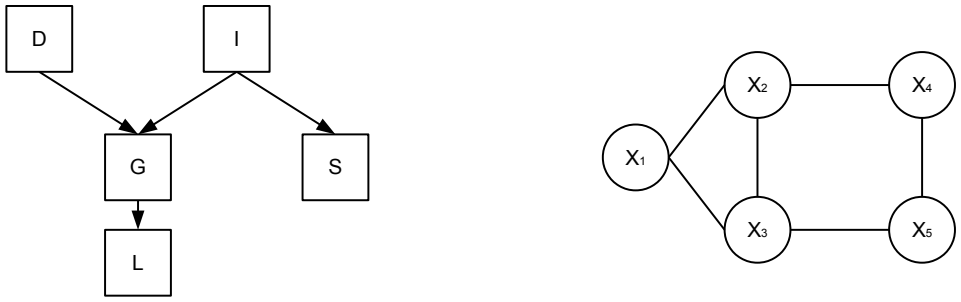
(a) Knoten (b) Kanten

Abbildung 3.1: Unterschiedliche Darstellungen von Knoten und Kanten in einem GM

Besteht ein Graph nur aus gerichteten Kanten und ist der Graph dabei noch zyklenfrei (siehe Abschnitt B.2), so spricht man von einem Bayes'schen Netz (BN). Besteht der Graph dagegen nur aus ungerichteten Kanten, spricht man von einem Markov Random Field (MRF). In der Abbildung 3.2 wird sowohl ein Beispiel für ein BN gezeigt (siehe Abbildung 3.2(a)), das auf dem Beispiel von [66] (*student example*) basiert, als auch ein Beispiel für ein MRF, das auf dem Beispiel von [80] (siehe Abbildung 3.2(b)) aufbaut.

In den nachfolgenden Abschnitten werden einige grundlegende Typen und Eigenschaften (BN, HMM, MRF, Inferenz und Lernen) von GMs kurz vorgestellt. Dennoch können die gesamten Einzelheiten der Thematik hier nicht in vollem Umfang wiedergegeben werden, hier sei auf [66] verwiesen, wo eine umfangreiche Einführung in GMs gegeben wird. Zur Realisierung von GMs stehen unterschiedliche sogenannte *Toolkits* zur Verfügung, in dieser Arbeit sind *The Bayes Net Toolbox for Matlab* [81], *The Graphical Models Toolkit* [82] und das *Hidden Markov Model Toolkit* [83] zum Einsatz gekommen.

Bevor BNs und MRFs genauer betrachtet werden, wird eine kurze allgemeine Beschreibung von GMs vorgestellt, die sowohl auf BNs als auch MRFs zutrifft.



(a) Beispiel für BN nach [66] (b) Beispiel für MRF nach [80]

Abbildung 3.2: Beispiele für ein BN (links) und ein MRF (rechts)

Sei $\mathcal{G}(\mathcal{V},\mathcal{E})$ ein Graph der Menge $\mathcal{V}$ Knoten und mit der Menge $\mathcal{E}$ Kanten. Die N Knoten $V_i \in \mathcal{V}$ repräsentieren die ZVs $\mathcal{X} = \{X_1,\ldots,X_N\}$ eines Systems bzw. Prozesses, die Abhängigkeiten zwischen den ZVs werden durch die M Kanten $E_j \in \mathcal{E}$ dargestellt.

3.1.2 Bayes'sche Netze

Abhängigkeiten, die von kausaler Natur sind, werden bei GMs mittels gerichteter (bzw. direkter) Kanten wiedergegeben. Hat ein GM nur direkte Kanten und des Weiteren einen zyklenfreien Graphen (siehe Abschnitt B.2), so erhält man ein BN. Die direkte Kante $E_i = (X_j, X_k)$ zwischen zwei Knoten, die die ZVs X_j und X_k darstellen, beschreibt die Abhängigkeit von X_k von X_j ($X_j \rightarrow X_k$). Die Bedeutung hinter dieser Darstellung ist, dass jede ZV X_i nur von den ZVs abhängig ist, die auf die ZV X_i zeigen. Die Menge der ZVs, von denen eine andere ZV X_i abhängt, wird als *Eltern* $pa(X_i)$ von X_i bezeichnet, somit ergibt sich folgende Definition:

$$pa(X_i) = \{X_j | (X_j, X_i) \in \mathcal{E}\}. \tag{3.1}$$

Es gilt für eine ZV X_i, dass diese nur von *Eltern* ZVs $pa(X_i)$ abhängig ist, alle anderen ZVs des BN haben keinen Einfluss auf diese ZV (bedingte Unabhängigkeit, siehe [66] für weitere Details).

Dieser Zusammenhang spielt eine entscheidende Rolle bei der Berechnung der Verbundwahrscheinlichkeit in einem BN, denn im Allgemeinen kann die Verbundwahrscheinlichkeit aller ZVs $\mathcal{X} = \{X_1,\ldots,X_N\}$ in einem BN folgendermaßen berechnet werden (Kettenregel für BNs [66]):

$$p(\mathcal{X}) = \prod_{i=1}^{N} p(X_i | pa(X_i)), \tag{3.2}$$

somit reduziert sich der Rechenaufwand für die Bestimmung der Verbundwahrscheinlichkeit.

Abbildung 3.2(a) zeigt ein Beispiel für ein BN, die Verbundwahrscheinlichkeit dieses BN vereinfacht sich unter Berücksichtigung nur der Abhängigkeit von den Elternknoten von folgender Faktorisierung

$$p(D,I,G,S,L) = p(D)P(I|D)p(G|D,I)p(S|D,I,G)p(L|D,I,G,S) \tag{3.3}$$

zu der vereinfachten Verbundwahrscheinlichkeit

$$p(D,I,G,S,L) = p(D)P(I)p(G|D,I)p(S|I)p(L|G). \tag{3.4}$$

Die vorgestellten BNs können nur die Konfigurationen bzw. Abhängigkeiten unterschiedlicher ZVs sowie deren Realisierungen modellieren. Die Erfassung von Veränderungen über die Zeit in den Prozessen bzw. in den Systemen bleibt außen vor.

Spielt die zeitliche Veränderung bzw. Entwicklung der Prozesse oder Systeme eine bedeutende Rolle, werden sogenannte Dynamische Bayes'sche Netze (DBNs) verwendet. Mit einem DBN kann man die Veränderung bzw. Entwicklung der ZVs eines BN über die Zeit darstellen, hierbei müssen neben den Abhängigkeiten für die ZVs innerhalb eines Zeitschrittes auch die Abhängigkeiten der ZVs zwischen den Zeitschritten modelliert werden.

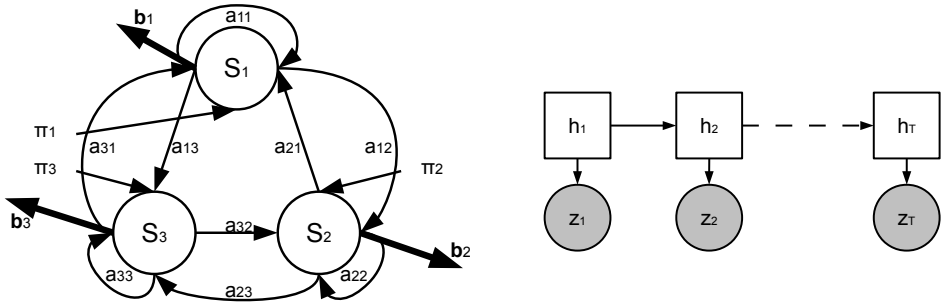
Die bekanntesten Vertreter von DBNs sind Hidden-Markov-Modelle, die in dem folgenden Abschnitt vorgestellt werden.

3.1.3 Hidden-Markov-Modelle

Hidden-Markov-Modelle (HMMs) sind eine einfache Form von DBNs und sind insbesondere durch ihren Einsatz in der Sprachverarbeitung bekannt (siehe [84]), daneben finden HMMs in vielen anderen Forschungsbereichen Verwendung, beispielsweise werden in Kapitel 4 Ansätze vorgestellt, die unterschiedliche Gesten anhand von HMMs unterscheiden können. Kapitel 5 stellt Ansätze vor, die DBNs im Bereich der Mimikerkennung zur Emotionsklassifikation einsetzen.

Zunächst werden die wichtigsten Eigenschaften von HMMs, angelehnt an die Notation von Rabiner [84], eingeführt, abschließend erfolgt eine Überleitung auf die GM-Notation, die in dieser Arbeit verwendet wird. Beide Notation sind in Abbildung 3.3 dargestellt.

HMMs modellieren zwei stochastische Prozesse, die in einer Beziehung stehen. Die zugrunde liegenden Eigenschaften des betrachteten Systems werden mit dem ersten Prozess beschrieben. Dieser Prozess wird als eine Markov-Kette erster Ordnung dargestellt, somit sind die Zustände nur von ihrem unmittelbaren zeitlichen Vorgänger abhängig. Der Ablauf der Zustandsfolge im ersten Prozess modelliert den zeitlichen Verlauf für das System, dabei kann aber nicht direkt auf die Zustände geschlossen werden, weswegen man von verborgenen (engl. *hidden*) Zuständen spricht. Ein zweiter Prozess modelliert, wie aus dem ersten Prozess Beobachtungen entstehen.



(a) HMM in Notation von [84] (b) HMM in GM-Notation

Abbildung 3.3: Unterschiedliche Darstellungsformen für HMMs

Die Menge der verborgenen Zustände für die Beschreibung der Markov-Kette umfasst N Elemente $\mathcal{S} = \{S_1, \ldots, S_N\}$, die zeitliche Realisierung eines Zustandes für den Zeitpunkt t wird mit der Wahrscheinlichkeit $p(q_t = S_j)$ beschrieben, somit ergibt sich die Menge für die zeitlichen Realisierungen $\mathcal{Q}_T = \{q_1, \ldots, q_T\}$. Eine Besonderheit bildet der erste zeitliche Zustand $q_{t=1}$, hierbei wird die Einsprungswahrscheinlichkeit für einen bestimmten Zustand S_j mit

$$\pi_j = p(q_1 = S_j), \qquad 1 \leq j \leq N \tag{3.5}$$

beschrieben. Aufgrund der ersten Ordnung der Markov-Kette reicht für die Gestaltung des zeitlichen Übergangs von einem Zustand q_t zu dem nächsten Zustand q_{t+1} folgende Zustandsübergangswahrscheinlichkeit:

$$a_{ij} = p(q_{t+1} = S_j | q_t = S_i), \qquad 1 \leq i, j \leq N \tag{3.6}$$

Die Zustandsübergangswahrscheinlichkeiten a_{ij} können in einer sogenannten $N \times N$ Zustandsmatrix A zusammengefasst werden. Je nach Ausgestaltung der Zustandsübergangswahrscheinlichkeiten a_{ij} können verschiedene HMM-Typen unterschieden werden. Bei sogenannten Links-Rechts-HMMs hat die Zustandsmatrix A eine obere Dreiecksform

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ 0 & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & a_{NN} \end{pmatrix} \tag{3.7}$$

somit kann von einem Zustand S_i nur in den gleichen Zustand S_i bzw. in einen Zustand S_j mit $j > i$ gewechselt werden. Bei sogenannten ergodischen HMMs sind $\forall i, j$ die Zustandsübergangswahrscheinlichkeiten $a_{ij} > 0$.

Die tatsächliche Beobachtungssequenz $\mathcal{O} = \{o_1, \ldots, o_T\}$, die durch den zweiten stochastischen Prozess emittiert wird, kann sowohl von diskreter als auch kontinuierlicher Natur sein. Für diskrete Verteilungen steht eine Menge von M-Elementen $\mathcal{V} = \{v_1, \ldots, v_M\}$ zur Emission zur Verfügung, bei kontinuierlichen Verteilungen werden häufig mehrdimensionale Gaußsche (Misch-)Verteilungen für die Modellierung der Beobachtung verwendet. Somit ergibt sich die Wahrscheinlichkeit für das Auftreten der Beobachtung o_t für einen bestimmten Zustand S_j folgendermaßen:

$$b_j(o_t) = p(o_t|q_t = S_j). \tag{3.8}$$

Ähnlich wie bei der Zustandsmatrix A können auch für alle N-Zustände die Wahrscheinlichkeitsverteilungen der Beobachtungen in einer sogenannten Beobachtungsmatrix B zusammengefasst werden. Ein HMM kann nun mit dem Parametersatz $\lambda = (A, B, \pi)$ vollständig beschrieben werden.

Die Verbundwahrscheinlichkeit für ein HMM mit dem Parametersatz λ kann für eine Sequenzlänge von T wie folgt geschrieben werden:

$$p(\mathcal{Q}_T, \mathcal{O}_T|\lambda) = \pi_{q_1} \prod_{t=2}^{T} a_{q_{t-1},q_t} \prod_{t=1}^{T} b_{q_t}(o_t). \tag{3.9}$$

Betrachtet man die Verbundwahrscheinlichkeit für das HMM in GM-Notation (siehe Abbildung 3.3(b)), ergibt sich folgende Faktorisierung:

$$p(\mathcal{H}_T, \mathcal{Z}_T) = p(h_1) \prod_{t=2}^{T} p(h_t|h_{t-1}) \prod_{t=1}^{T} p(z_t|h_t). \tag{3.10}$$

Vergleicht man Gleichung 3.9 und Gleichung 3.10 miteinander, kann man folgende Zuordnungen erkennen:

Einsprungswahrscheinlichkeit:	π_{q_1}	$\widehat{=}$ $p(h_1)$	(3.11)
Zustandsübergangswahrscheinlichkeit:	a_{q_{t-1},q_t}	$\widehat{=}$ $p(h_t\|h_{t-1})$	(3.12)
Beobachtungswahrscheinlichkeit:	$b_{q_t}(o_t)$	$\widehat{=}$ $p(z_t\|h_t)$	(3.13)

Aufgrund dieser Zuordnung ist es leicht möglich, ein HMM in GM-Notation darzustellen. Dieser Zusammenhang von HMMs und GMs beschränkt sich nicht nur auf die Darstellung, sondern auch der *Baum-Welch*-Algorithmus [85, 84], der für das Lernen von HMMs zum Einsatz kommt, ist bei genauer Betrachtung eine spezielle Realisierung des *Expectation-Maximization* (EM)-Algorithmus [86, 87], der beim Lernen von GM-Parametern verwendet wird. Darüber hinaus gibt es Überschneidungen für die Inferenz zwischen HMMs und GMs, bei beiden kommt der *Viterbi*-Algorithmus [88] zum Einsatz, um den wahrscheinlichsten Pfad in einem Modell bestimmen zu können. Daneben gibt es Ähnlichkeiten zwischen dem *Vorwärts-Rückwärts*-Algorithmus [84] und der Nachrichtenpropagierung nach Pearl [71] für GMs.

3.1.4 Markov Random Fields

In den Fällen, in denen die Beziehungen zwischen den verschiedenen ZVs nicht von kausaler Natur sind, sondern wechselseitige Abhängigkeiten beschreiben, sind gerichtete Kanten nicht geeignet. In diesen Situationen werden ungerichtete Kanten verwendet – besteht ein Graph nur aus ungerichteten Kanten, spricht man von einem MRF.

Ein MRF wird von einem Graph $\mathcal{G}'(\mathcal{V},\mathcal{E}')$ erzeugt, hierbei besteht die Menge $\mathcal{E}'$ nur aus ungerichteten Kanten, während $\mathcal{V}$ wieder die Menge von N Knoten $V_i \in \mathcal{V}$ ist, die die ZVs $\mathcal{X} = \{X_1,\ldots,X_N\}$ beschreiben. Während Schleifen bei der Modellierung von kausalen Zusammenhängen in BNs nicht zulässig sind, ist die Verwendung von Schleifen in MRFs zulässig und oft auch notwendig.

Die Kante zwischen zwei Knoten $E_k = \{X_i, X_j\}$ in einem MRF ist ungerichtet und ist daher besser selbst als Menge beschrieben, als ein geordnetes Tupel (wie im Falle eines BN). Die ungerichtete Kante kann beschrieben werden als eine Kante, die in beide Richtungen zeigt, somit werden die ZVs, die durch eine ungerichtete Kante miteinander verbunden sind, wechselseitig beeinflusst.

Die Definition der Verbundwahrscheinlichkeit in MRFs erfolgt über sogenannte *Cliquen*. Eine Clique ist eine vollständige, zusammenhängende Teilmenge von Knoten $\mathcal{V}$ (siehe Anhang B), bei der jeder Knoten eine Kante mit jedem anderen Knoten aus der Teilmenge hat. Intuitiv beschreibt somit jede Clique eine Teilmenge des Graphen, wobei die ZVs wechselseitige Abhängigkeiten besitzen. Jeder dieser Cliquen C_i aus der gesamten Cliquenmenge $\mathcal{C}$ kann eine nicht-negative Funktion zugewiesen werden, diese Funktion wird als Cliquenpotential ${}^{C_i}\psi$ bezeichnet. Anhand dieser Cliquenpotentiale kann nun die Verbundwahrscheinlichkeit der ZVs $\mathcal{X}$ beschrieben werden als

$$p(\mathcal{X}) = \frac{1}{Z} \prod_{C_i \in \mathcal{C}} {}^{C_i}\psi, \tag{3.14}$$

um die Wahrscheinlichkeit zu garantieren, bedarf es eines Normalisierungsfaktors, der folgendermaßen definiert ist

$$Z = \sum_{\mathcal{X}} \prod_{C_i \in \mathcal{C}} {}^{C_i}\psi. \tag{3.15}$$

Abbildung 3.2(b) zeigt ein Beispiel für ein MRF, die zugehörigen Cliquen und Cliquenpotentiale sind in der Abbildung 3.4 dargestellt.[1]

Anhand dieser Cliquenpotentiale kann die Verbundwahrscheinlichkeit dieses Beispiels folgendermaßen berechnet werden:

$$p(X_1,X_2,X_3,X_4,X_5) = \frac{1}{Z} {}^{a}\psi, {}^{b}\psi, {}^{c}\psi, {}^{d}\psi, \tag{3.16}$$

[1] Der Vollständigkeit halber sind in den jeweiligen Cliquenpotentialen noch die in der Clique involvierten ZVs gezeigt. In Zukunft wird auf die Erwähnung der ZVs verzichtet.

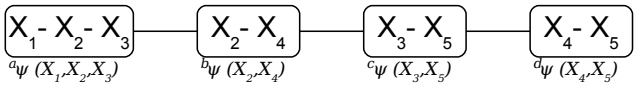


Abbildung 3.4: Cliquen und dazugehörige Potentiale für das MRF von Abbildung 3.2(b)

mit dem Normalisierungsfaktor

$$Z = \sum_{X_1} \sum_{X_2} \sum_{X_3} \sum_{X_4} \sum_{X_5} {}^{a}\psi, {}^{b}\psi, {}^{c}\psi, {}^{d}\psi.$$

3.1.5 Inferenz in Graphischen Modellen

Inferenz, die Schlussfolgerung über gewisse Zustände und Belegungen von ZVs eines Systems, ist eine schwierige Aufgabe, da die Verbundwahrscheinlichkeit aller ZVs sehr schnell sowohl schwierig repräsentierbar als auch rechnerisch beherrschbar wird. GMs nutzen die Informationen über Unabhängigkeiten der ZVs, um sowohl die Darstellung des Systems als auch die Berechnung zu vereinfachen, dabei können exakte und inexakte Inferenzalgorithmen verwendet werden. Bei der exakten Inferenz werden die Ergebnisse (Belegungen der ZVs) exakt bestimmt, dennoch gibt es auch hier unterschiedliche Verfahren, die sich um eine Optimierung der Berechnungsvorschrift kümmern. Jedoch hängt die Berechnungszeit für die Inferenz stark von der Ausgestaltung des GM ab. Bei der inexakten Inferenz kann auf die Berechnungszeit mehr Einfluss genommen werden, hierbei können die Wahrscheinlichkeitsdichtefunktionen (WDFs) durch Partikel-Filter oder iterierende Funktionen approximiert werden. Anhand der Freiheitsgrade, die diese Methoden bieten (Partikel-Filter: Anzahl der Partikel, iterierende Funktion: Anzahl der Iterationen), ist es möglich, Genauigkeit und Berechnungsgeschwindigkeit den jeweiligen Anforderungen anzupassen.

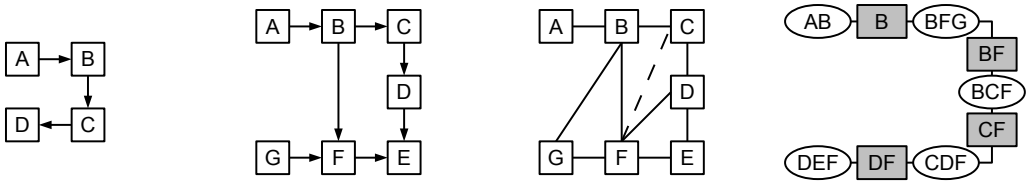
Die Ausgestaltung der Abhängigkeit in BNs und MRFs ist von verschiedener Natur. In BNs besteht eine kausale Abhängigkeit zwischen den einzelnen ZVs, während die Abhängigkeit in MRFs mehr eine wechselseitige Abhängigkeit darstellt, gleichwohl kann die Inferenz in beiden auf die gleiche Art und Weise durchgeführt werden. Um die gleichen Inferenzmethoden anwenden zu können, werden die BNs zu MRFs umgewandelt.

Im Folgenden werden kurz die wesentlichen Punkte exakter und inexakter Inferenzmethoden für GMs vorgestellt, weitere Details finden sich in [89, 64, 66].

3.1.5.1 Exakte Inferenz

Es gibt mehrere Verfahren, um eine exakte Inferenz in einem GM durchzuführen, generell lassen sich dabei zwei grundlegende Methoden unterscheiden: Variableneliminaton (engl. *variable elimination*) und Nachrichtenpropagierung (engl. *message passing*).

Beide Verfahren werden kurz anhand von zwei Beispielen erklärt, die betrachteten GMs sind in Abbildung 3.5(a) bzw. in Abbildung 3.5(b) dargestellt. Zusätzlich dazu finden sich in Abbildung 3.5(c) und in Abbildung 3.5(d) die weiteren Schritte, die für die Inferenz mittels einer Nachrichtenpropagierung notwendig sind.



(a) GM: Beispiel für Variablenelimination

(b) GM: Beispiel für Nachrichtenpropagierung

(c) GM: Beispiel für Moralgraph mit Triangulation (gestrichelt)

(d) GM: Beispiel für Verbundbaum

Abbildung 3.5: GMs für die Inferenzbeispiele: Variablenelimination (a) und Nachrichtenpropagierung (b-d)

Die Variablenelimination bildet die einfache und intuitive Methode, um exakte Inferenz durchzuführen, es ist dabei möglich sowohl mit Wahrscheinlichkeitsfunktionen (WFs)[1] bzw. WDFs[2] von BNs als auch mit Cliquenpotentialen von MRFs zu arbeiten. Das Grundkonzept von Variablenelimination besteht darin, im Rahmen einer Verbundwahrscheinlichkeit über alle nicht gesuchten ZVs zu summieren bzw. zu integrieren, um anschließend die gesuchte ZV zu erhalten. Hierbei können auch Beobachtungen einfließen, indem dann die zugehörige Summierung bzw. Integration entfällt. Eine Möglichkeit, dieses Verfahren zu beschleunigen, ist, die Summierungen (im diskreten Fall) über die ZVs soweit wie möglich nach rechts in der Gleichung zu schieben. Folgendes Beispiel soll dies verdeutlichen.

Wir nehmen das BN von Abbildung 3.5(a) und versuchen die Verteilung $p(d)$ zu bestimmen, des Weiteren haben alle ZVs A,B,C,D nur zwei Zustände $\{0,1\}$. Ausgehend davon hat die Verbundwahrscheinlichkeit insgesamt nur acht Belegungen, somit ist klar, dass zur Bestimmung von $p(d)$ nur über die Einträge von A,B,C summiert werden muss. Somit ergibt sich für die $p(d = 1)$, bei Berücksichtigung die Summierungen möglichst weit nach rechts zu schieben, folgende Summierung:

[1] Diskrete Verteilung der ZVs.

[2] Kontinuierliche Verteilung der ZVs.

$$p(d=1) = \sum_{a,b,c} p(a,b,c,d=1) \tag{3.17}$$

$$= \sum_{a,b,c} p(d=1|c)p(c|b)p(b|a)p(a) \tag{3.18}$$

$$= \sum_{c} p(d=1|c) \sum_{b} p(c|b) \sum_{a} p(b|a)p(a) \tag{3.19}$$

Aus dem Produkt $\xi_1 = p(b|a)p(a)$ summiert man über A und erhält die Funktion $f_a(b) = \sum_A \xi_1$, und erhält nun folgende Summierung:

$$p(d=1) = \sum_{c} p(d=1|c) \sum_{b} p(c|b) f_a(b). \tag{3.20}$$

Hierbei wird wiederum aus dem Produkt $\xi_2 = p(c|b)f_a(b)$ über B summiert, und man erhält $f_b(c) = \sum_B \xi_2$, somit ergibt sich für die letzte Summierung über C:

$$p(d=1) = \sum_{c} p(d=1|c) f_b(c). \tag{3.21}$$

Bei Anwendung dieser Technik ist die Anzahl von Summierungen auf sechs reduziert worden, was zwar keine große Einsparung gegenüber acht Summierungen bedeutet, aber dennoch ist Tendenz erkennbar und insbesondere bei größeren Graphen von Vorteil. Ein wesentlicher Nachteil dieser Inferenzmethode liegt darin, dass immer für eine bestimmte gesuchte ZV – im Beispiel $p(d=1)$ – die Berechnung durchgeführt werden muss. Inferenzverfahren, basierend auf der Nachrichtenpropagierung, sind in diesem Zusammenhang effizienter, denn hierbei muss nicht für jede ZV eines GM die gesamte Faktorisierung wiederholt werden.

Im Folgenden wird kurz die Nachrichtenpropagierung nach dem HUGIN-Verfahren[1] [90] (beschrieben in [65]) vorgestellt. Dieses Verfahren betreibt die Nachrichtenpropagierung auf dem sogenannten Verbundbaum (engl. *junction tree*), hierbei müssen etwaige gerichtete GMs noch in ungerichtete GMs überführt werden (siehe Abschnitt B). Für die Aufstellung des Verbundbaumes eines gerichteten GM werden folgende Schritte durchgeführt. Erstens, das gerichtete GM wird moralisiert, hierbei werden alle Elternknoten jedes Knotens mit einer ungerichteten Kante verbunden, anschließend werden die restlichen gerichteten Kanten in ungerichtete überführt. Dieses Vorgehen ist exemplarisch in Abbildung 3.5(c) dargestellt, hierbei ist das gerichtete GM von Abbildung 3.5(b) moralisiert worden (nur durchgezogene Linien). Zweitens, der entstandene sogenannte Moralgraph wird trianguliert, das heißt für jeden Zyklus (ein Pfad mit dem gleich Start- und Endknoten) der eine Länge von ≥ 4 besitzt und der nicht von weiteren Kanten durchkreuzt wird, werden zusätzliche Kanten hinzugefügt,

[1] HUGIN engl. Handling Uncertainty in General Inference Network

damit alle minimalen Zyklen maximal eine Länge von drei haben. Dieses Vorgehen ist exemplarisch in Abbildung 3.5(c) dargestellt, hierbei ist die zusätzliche Kante zur Triangulation (gestrichelt dargestellt) in den Graph eingefügt worden. Das Ergebnis der Triangulation ist nun ein MRF, aus dessen Cliquen wird nun der Verbundbaum konstruiert. Die Cliquen des MRF bilden dabei als sogenannte *Cluster* die Knoten des Verbundbaumes. Zwischen zwei *Cluster*-Knoten sitzt jeweils ein *Seperator*-Knoten, der die Schnittmenge der ZVs aus beiden *Cluster*-Knoten enthält.

Den Potentialfunktionen, für *Cluster* ψ und für *Seperatoren* ϕ, wird initial der Wert 1 zugewiesen. Im nächsten Schritt wird die bedingte Wahrscheinlichkeit jedes Knoten aus dem ursprünglichen gerichteten GM zu demjenigen Clusterpotential ψ hinzumultipliziert, das den jeweiligen Knoten und all seine Elternknoten enthält. Somit wird sichergestellt, dass die Verbundwahrscheinlichkeit des gerichteten GM dem Verbundbaum entspricht. Für die Herstellung von lokaler Konsistenz, das heißt bei Marginalisierung aus verschiedenen *Cluster* muss sich für die gleiche ZV auch immer derselbe Wert einstellen, daher muss eine initiale Nachrichtenpropagierung durchgeführt werden. Dabei werden, ausgehend von einem beliebigen *Cluster* als Wurzelknoten, die *Separatoren* an die vorangegangenen *Cluster* angepasst durch folgende Summierung:

$$ {}^{S_1}\phi^* := \sum_{\mathcal{X} \in C_1 \setminus S_1} {}^{C_1}\psi. \tag{3.22} $$

Die nachfolgenden *Cluster* werden an die aktualisierten *Separatoren* mit folgender Multiplikation angepasst:

$$ {}^{C_2}\psi^* = \frac{{}^{S_1}\phi^*}{{}^{S_1}\phi} {}^{C_2}\psi. \tag{3.23} $$

Nachdem diese Information in beide Richtungen – zum Wurzelknoten hin und wieder zurück – propagiert worden ist, ist der Verbundbaum nun lokal konsistent. Somit lässt sich die Verbundwahrscheinlichkeit im Verbundbaum anhand der $\mathcal{C}$ *Clusterpotentiale* ψ und der $\mathcal{S}$ *Seperatorpotentiale* ϕ folgendermaßen definieren:

$$ p(\mathcal{X}) = \frac{\prod_{C \in \mathcal{C}} {}^{C}\psi}{\prod_{S \in \mathcal{S}} {}^{S}\phi}. \tag{3.24} $$

3.1.5.2 Inexakte Inferenz

Die Möglichkeiten, die Inferenz-Geschwindigkeit bei Verfahren der exakten Inferenz zu erhöhen, sind begrenzt, daher gibt es weitere Möglichkeiten – inexakte Inferenz –, um die Geschwindigkeit zu steigern. Hierbei wird versucht mit unterschiedlichen Methoden die WFs bzw. die WDFs der ZVs des GM zu approximieren. In der Regel basieren die Methoden der inexakten Inferenz auf Monte-Carlo-Methoden für statische GMs bzw. Sequentielle Monte-Carlo-Methoden (auch Partikel-Filter genannt)

für dynamische GMs. Daneben gibt es auch die Möglichkeit des Einsatzes von sogenannten Variationsalgorithmen (engl. *variational algorithms*), die die Aufgabe der Approximation der Wahrscheinlichkeitsfunktionen der ZVs in ein Optimierungsproblem überführen [89].

Grundlagen zu Monte-Carlo-Methoden und Sequentielle Monte-Carlo-Methoden (bzw. Partikel-Filter-Verfahren), wie *Perfect Monte Carlo Sampling*, *Importance Sampling*, *Sequential Importance Sampling* etc., können in [91, 92, 93] gefunden werden. Im Folgenden werden kurz wesentliche Eigenschaften zu Monte-Carlo-Methoden anhand des *Perfect Monte Carlo Sampling* vorgestellt, weitere Ausführungen für Partikel-Filter-Verfahren sowie *Importance Sampling* finden sich im Abschnitt 6.2.4, in dem Partikel-Filter-Verfahren für bildbasierte Verfolgungsprozesse vorgestellt werden.

Die Grundidee von Monte-Carlo-Methoden zielt darauf ab, von einer hochdimensionalen WDF $p(\mathbf{x})$ eine Reihe von N unabhängig und identisch verteilten (engl. *independent and identically distributed*, i. i. d.) Stichproben $\mathbf{s}^{(i)}$ (auch Partikel genannt) zu nehmen, die die WDF $p(\mathbf{x})$ bestmöglich beschreiben sollen. Für das *Perfect Monte Carlo Sampling* wird die Dirac-Funktion für die Abtastung verwendet, somit ergibt sich folgende Approximation der ursprünglichen WDF $p(\mathbf{x})$:

$$P_N(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\delta(\mathbf{x},\mathbf{s}^{(i)}), \tag{3.25}$$

wobei die Dirac-Funktion folgendermaßen definiert ist:

$$\delta(\mathbf{x},\mathbf{s}^{(i)}) = \begin{cases} 1 & \text{falls } \mathbf{x} = \mathbf{s}^{(i)} \\ 0 & \text{ansonsten.} \end{cases}$$

Abbildung 3.6 zeigt exemplarisch die Approximation von der WDF $p(\mathbf{x})$ mittels *Perfect Monte Carlo Sampling*, womit sich die neue $P_N(\mathbf{x})$ ergibt.

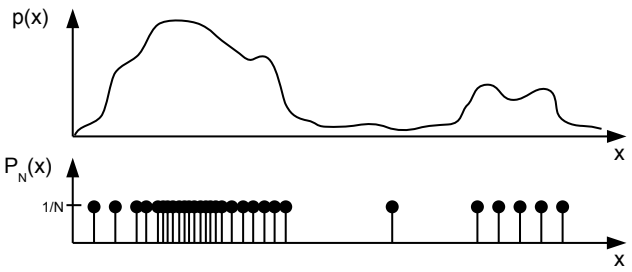


Abbildung 3.6: Eine WDF und ihre zugehörige Approximation mittels N-Partikel

Daneben gibt es noch weitere Verfahren wie beispielsweise *Gibbs Sampling*, den *Metropolis-Hastings*-Algorithmus, die bei GMs für die Approximation der WDFs der ZVs verwendet werden können [89].

3.1.6 Lernen

Neben der Inferenz spielt auch das Lernen eine entscheidende Rolle für GMs. Beim Lernen können mehrere Fälle unterschieden werden, diese Unterscheidungen beziehen sich auf die Art des GM (BN oder MRF), die Struktur des Netzes (bekannt oder unbekannt) und die Beobachtbarkeit der Knoten (komplett oder teilweise). Eine Aufstellung für die unterschiedlichen Fälle findet sich in [94, 63]. Im Folgenden werden nur kurz die Fälle betrachtet, in denen die Struktur des GM gegeben ist, da in dieser Arbeit nur solche Fälle betrachtet werden. Ein bekanntes Lernverfahren ist das Maximum Likelihood-Lernen (engl. *maximum-likelihood estimation*, MLE) [95], hierbei wird versucht die Datenwahrscheinlichkeit von N-Trainingsbeispielen $\vec{x}_i$ für einen gegebenen Parametersatz θ zu maximieren.

Das MLE lässt sich folgendermaßen für die Gesamtdatenwahrscheinlichkeit beschreiben

$$\mathcal{L}(\theta|\boldsymbol{\mathcal{X}}) = \log \prod_{i=1}^{N} p(\vec{x}_i|\theta) = \sum_{i=1}^{N} \log p(\vec{x}_i|\theta), \tag{3.26}$$

wobei $\vec{x}_i$ ein Trainingsbeispiel für das gesamte GM ist, alle Belegungen der k Knoten bzw. ZVs $\mathcal{X} = \{X_1, \ldots, X_k\}$ sind bekannt.

Ziel ist es die Gesamtdatenwahrscheinlichkeit zu maximieren:

$$\hat{\theta} = \arg\max_{\theta} \mathcal{L}(\theta|\boldsymbol{\mathcal{X}}). \tag{3.27}$$

Für den Fall, dass alle Knoten beobachtet werden, kann das frequenzbasierte Lernen zum Einsatz kommen. Exemplarisch soll dieses Lernverfahren für die Wahrscheinlichkeit $p(G|D,I)$ aus Abbildung 3.2(a) gezeigt werden. Es wird angenommen, dass die ZVs D und I nur zwei Zustände haben und die ZV G drei Zustände (1,2,3) hat, somit ergibt sich für die geschätzte Wahrscheinlichkeit für die Belegung von $p(G=2|D=1,I=1)$ folgende Abschätzung

$$p_{ML}(G=2|D=1,I=1) = \frac{\#(G=2,D=1,I=1)}{\#(D=1,I=1)}, \tag{3.28}$$

$\#(G=2,D=1,I=1)$ ist die Anzahl an Trainingsbeispielen, wobei die ZV G den Zustand 2 hat, die ZVs D und I jeweils den Zustand 1 haben. Somit lässt sich die WF für jeden Knoten des GM abschätzen.

Falls nicht alle Knoten des GM beobachtet sind, kommt eine andere Abschätzung als MLE zum Einsatz, da die WF bzw. WDF des GM nicht mehr direkt berechnet werden kann. Es kann entweder ein Gradientenverfahren, wie beispielsweise in [63], verwendet werden oder der sogenannte EM-Algorithmus [86, 87] kommt zum Einsatz. Die Menge der Knoten $\mathcal{X} = \{X_1, \ldots, X_k\}$ eines Trainingsbeispiels $\vec{x}_i$ teilt sich nun in

eine beobachtete Menge $\vec{o}_i$ und in eine unbeobachtete Menge $\vec{h}_j$ (von H-möglichen Zuständen) auf, woraus sich folgende Abschätzung für die Wahrscheinlichkeit ergibt:

$$\mathcal{L}(\theta|\boldsymbol{\mathcal{X}}) = \log \prod_{i=1}^{N} \sum_{j=1}^{H} p(\vec{o}_i, \vec{h}_j|\theta). \tag{3.29}$$

Aufgrund der Summe innerhalb des Logarithmus ist keine direkte Berechnung der WF bzw. der WDF des GM möglich.

Der EM-Algorithmus beruht auf zwei Schritten: einen *Erwartungswert*-Schritt (engl. *expectation step*) und einen *Maximierung*-Schritt (engl. *maximization step*). Beide Schritte beruhen darauf, dass zunächst ein beliebiger Parametersatz θ' gewählt worden ist. Ausgehend von einer Hilfsfunktion $Q(\theta, \theta')$ wird dann zunächst ein Erwartungswert für die unbeobachteten Variablen $\mathcal{H}$ für den gegebenen Parametersatz θ' sowie die gegebene Beobachtung $\mathcal{O}$ berechnet (*Erwartungswert*-Schritt)

$$Q(\theta, \theta') = \mathbb{E}[\log p(\mathcal{O}, \mathcal{H}|\theta)|\mathcal{O}, \theta']. \tag{3.30}$$

Anschließend wird von diesem Erwartungswert über den Parametersatz θ maximiert (*Maximierung*-Schritt)

$$\hat{\theta} = \arg\max_{\theta} Q(\theta, \theta'). \tag{3.31}$$

Der *Erwartungswert*-Schritt und der *Maximierung*-Schritt werden iterativ wiederholt, bis der Algorithmus konvergiert, in der Regel findet sich ein lokales Maximum.

3.2 Experimente

Im Rahmen dieser Arbeit wurden mehrere Experimente durchgeführt, um unterschiedliche Ansätze miteinander zu vergleichen. Dabei treten in der Regel zwei Problemstellungen auf: erstens, wie lässt sich anhand einer bestimmten, meist zu kleinen, Datenmenge bestmöglich ein Erkennungssystem erstellen bzw. trainieren. Zweitens, wie vergleicht man unterschiedliche Ansätze miteinander, um eine Aussage über die Qualität der Systeme zu machen. Dem zuerst genannten Problem wird in der Regel mit dem Prinzip der Kreuzvalidierung begegnet, für das zuletzt genannte Problem wird versucht anhand von statistischer Signifikanz eine Aussage über die Qualität der Erkennungssysteme zu geben.

3.2.1 Kreuzvalidierung

Kreuzvalidierung wird verwendet, wenn die Anzahl an Datensätzen nicht ausreicht, um eine Einteilung der Daten in Test-, Validierung- und Trainingsmenge vorzuneh-

men [96]. Es lassen sich zwei unterschiedliche Arten von Kreuzvalidierung unterscheiden: k-fache Kreuzvalidierung und *Leave-One-Out*-Kreuzvalidierung. Bei der k-fachen Kreuzvalidierung wird die Menge von N-Daten in k unterschiedliche, nichtüberlappende Teilmengen eingeteilt, daraufhin werden $k-1$ der Teilmengen für das Training des Erkennungssystems verwendet, um anschließend auf der verbliebenen Teilmenge das Erkennungssystem zu testen. Diese Prozedur wird k-mal wiederholt, bis auf jeder Teilmenge ein Test durchgeführt worden ist. Die *Leave-One-Out*-Kreuzvalidierung basiert darauf, dass die Anzahl von k gleich der Anzahl der N-Datensätze ist.

Bei den hier durchgeführten Experimenten wurde immer personenunabhängig getestet, das heißt die Datensätze von einer bestimmten Person befanden sich immer ausschließlich in der Trainings- bzw. Testmenge, aber nie in beiden. Eine Sonderform dieser personunabhängigen Kreuzvalidierung ist die *Leave-One-Person-Out*-Kreuzvalidierung, hierbei werden die Daten immer nur von einer Person zum Testen verwendet, die übrigen Personen bilden die Trainingsmenge.

3.2.2 Statistische Signifikanz

Ein Vergleich zweier Erkennungssysteme E_1 und E_2 kann beispielsweise anhand ihrer Erkennungsrate (R) erfolgen, dabei reicht es in der Regel nicht, einfach die jeweiligen Erkennungsraten R_1 und R_2 miteinander zu vergleichen, sondern es bedarf einer statischen Analyse. Hierbei wird von der Nullhypothese H_0 ausgegangen, der zufolge die Erkennungssysteme E_1 und E_2 gleich leistungsfähig sind. In dieser Arbeit ist der gepaarte t-Test verwendet worden [97], der eine neue ZV $X(t)$ einführt, die die Leistung der beiden Erkennungssysteme E_1 und E_2 folgendermaßen vergleicht:

$$X(t) = \begin{cases} 1 & E_1 \text{ korrekt und } E_2 \text{ falsch} \\ 0 & E_1 \text{ und } E_2 \text{ beide korrekt bzw. falsch} \\ -1 & E_1 \text{ falsch und } E_2 \text{ korrekt} \end{cases}$$

Für eine genügend hohe Anzahl an Stichproben N_{test} und bei gleichzeitigem Zutreffen der Nullhypothese kann eine Testgröße

$$Z = \frac{\mu_x}{\sqrt{\frac{\sigma_x^2}{N_{test}}}}, \tag{3.32}$$

eingeführt werden, wobei μ_x der Mittelwert und σ_x^2 die Varianz der ZV $X(t)$ sind.

Diese neue ZV Z liefert eine Wahrscheinlichkeit p_n für die Gültigkeit der Nullhypothese zurück. Über $p_r = 1 - p_n$ lässt sich das Signifikanzniveau definieren. Für $p_r \geq 0{,}95$ hat man einen statistisch signifikanten Unterschied zwischen beiden Erkennungssystemen E_1 und E_2, ab $p_r \geq 0{,}99$ spricht man von einem statistisch hochsignifikanten Unterschied.

Kapitel 4

Gestenerkennung

Inhaltsangabe

In diesem Kapitel werden drei Verfahren zur Gestenerkennung vorgestellt, dabei beziehen sich die ersten beiden Verfahren auf die Erkennung von Handgesten (statisch, dynamisch), während mit dem dritten Verfahren Kopfgesten erkannt werden können. Alle Verfahren sind in eine einheitliche Architektur eingebunden, um sowohl eine echtzeitfähige Verarbeitung zu ermöglichen als auch einem Dialogsystem eine einheitliche Schnittstelle für die Gestenerkennung zu liefern.

4.1 Einführung

4.1.1 Motivation

Die Menschen kommunizieren durch Sprache, Gesten, Blick, Mimik und Körperhaltung, um ihre Emotionen, Stimmungen und Zuwendungen auszudrücken, dabei kommt eine Mischung von audio-visuellen Signalen zum Einsatz [98, 99, 100]. Für die Kommunikation stehen dem Menschen unterschiedliche Modalitäten zur Verfügung, die sich nach [101] in zwei Kategorien einteilen lassen. Modalitäten, mit denen die Umgebung wahrgenommen werden kann (*Sensing Modality*), und Modalitäten, mit denen der Mensch mit der Umgebung interagieren kann (*Action Modality*). Die Abbildung 4.1 enthält eine Aufstellung der wahrnehmenden Modalitäten (*Sensing Modality*) und agierenden Modalitäten (*Action Modality*) des Menschen nach [101].

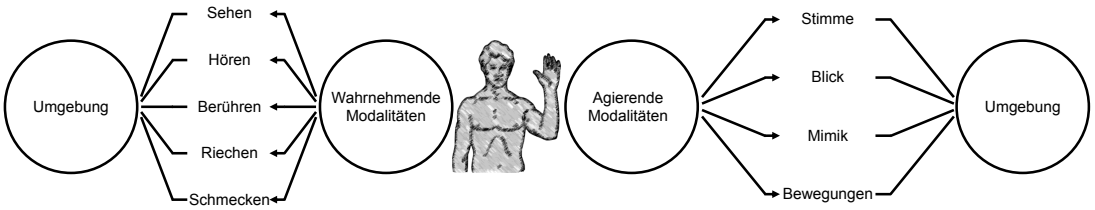


Abbildung 4.1: Definition für die Modalitäten des Menschen nach [101]

Bei der Mensch-Maschine-Interaktion (MMI) ist es ein wesentliches Ziel, diese Kommunikation der Mensch-Mensch-Interaktion ähnlicher zu machen [101], daher soll eine Maschine (z.B. ein Roboter, ein Computersystem) auch in der Lage sein, mit dem Menschen in einer multimodalen Art und Weise zu kommunizieren. Für eine Maschine lassen sich unterschiedliche Modalitäten definieren, diese umfassen auch Aspekte menschlicher Modalitäten, etwa visuell mithilfe von Kameras oder auditiv mithilfe von Mikrofonen. Darüber hinaus gibt es Sensoren, die es Maschinen ermöglichen ihre Umgebung sowohl haptisch als auch olfaktorisch zu erfassen.

Ein zentraler Bestandteil von multimodaler Kommunikation ist die Tatsache, dass sich die unterschiedlichen Modalitäten meist komplementär verhalten, und nicht, wie meist angenommen, redundant [102]. Daher sollte eine Maschine mit dem Mensch multimodal interagieren, um die Gesamtheit der Interaktion erfassen zu können. Laut [103, 104] wird unter natürlicher Interaktion zwischen Mensch und Maschine der Einsatz von Sprache und Gesten verstanden, somit spielen sowohl verbale als auch nonverbale Kommunikationsaspekte eine Rolle. Eine natürliche und intuitive Kommunikation zwischen Mensch und Maschine kann insbesondere für ältere und behinderte Menschen hilfreich sein, damit neue technische Systeme leicht und einfach bedient bzw. benutzt werden können.

Gesten bilden einen zentralen Bestandteil der nonverbalen Kommunikation, sie

können sowohl unbewusst (Gestikulation) als auch bewusst (Zeichensprache) in der Kommunikation eingesetzt werden und umfassen einfache und komplexe Bewegungen. Gesten finden auch vermehrt Einsatz bei der Bedienung von technischen Geräten, beispielsweise lassen sich heutzutage Fernseher (z.B. *Samsung Smart TV* [105]) mithilfe von Gesten steuern. Im Bereich von AAL-Anwendungen lassen sich unterschiedliche Ansätze für den Einsatz von Gesten finden: In [106] wird der Einsatz von Gesten zur Steuerung von unterschiedlichen Geräten in einem Smart Home diskutiert, z.B. soll sich durch das Deuten auf eine Lampe das Licht einschalten lassen. In [107] wird ein System zur Gestensteuerung für ein Smart Home vorgestellt, das den Wii-Controller verwendet, um unterschiedliche Gegenstände wie z.B. den Fernseher zu bedienen.

In diesem Kapitel werden drei verschiedene Ansätze im Bereich der Gestenerkennung vorgestellt. Im ersten und zweiten Ansatz werden statische bzw. dynamische Handgesten betrachtet, während mit dem dritten Ansatz Kopfgesten erkannt werden können. Die vorgestellten Ansätze sind teilweise in [108, 109, 110, 111, 112] veröffentlicht worden, überdies sind die Ansätze, aufgrund der Einbettung in einer einheitlichen Architektur, leicht portierbar, weshalb sie sowohl auf der Roboterplattform von [113] als auch auf der Roboterplattform ELIAS (siehe Abschnitt 7.2) zum Einsatz gekommen sind.

4.1.2 Stand der Technik

Die Definition einer Geste ist nicht exakt, es gibt je nach Forschungsbereich unterschiedliche Definitionen, generell haben Gesten die Eigenschaften, dass sie sowohl Mehrdeutigkeiten erlauben als sich auch nur unvollständig beschreiben lassen [114]. Im Folgenden, siehe Abbildung 4.2, wird die Definition nach [115] verwendet, wobei zunächst bei Bewegungen (in [115] Arm- und Handbewegungen) zwischen unbeabsichtigten Bewegungen und Gesten unterschieden wird, bei den Gesten kann wiederum zwischen manipulativen und kommunikativen Gesten unterschieden werden. Diese Definition ist an [116] angelehnt.

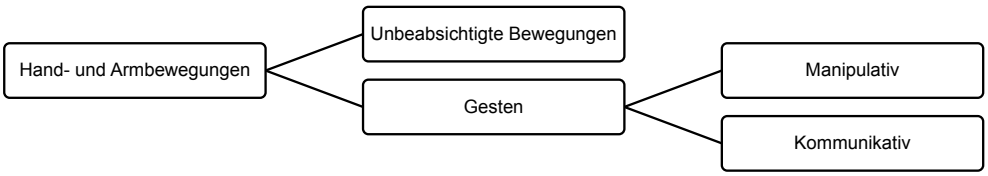


Abbildung 4.2: Definition für Gesten nach [115]

Die Unterscheidung zwischen kommunikativen und manipulativen Gesten ist hier ausreichend, da nur kommunikative Gesten in dieser Arbeit betrachtet werden. Ferner ist es möglich, die kommunikativen Gesten noch weiter zu unterteilen (siehe

[116, 115]). Der Einsatz von Gesten in der MMI bildet den Forschungsgegenstand zahlreicher Arbeiten, man kann in diesem Kontext sowohl nach der Art der untersuchten Gesten unterscheiden, als auch nach dem Sensortyp, der zur Erfassung der Geste dient. Die Darstellung von Gesten kann von statischer oder dynamischer Natur sein, zudem gibt es beispielsweise im Bereich von Zeichensprache auch eine Kombination von statischen und dynamischen Gesten. Eine Geste kann von unterschiedlichen Körperbewegungen gebildet werden, dabei können Finger, Hände, Arme, der Kopf, das Gesicht oder auch der ganze Körper bewegt werden, um eine Geste zu formen [114].

Für die Erfassung von Gesten gibt es eine Vielzahl an unterschiedlichen Sensoren: elektrisch, optisch, akustisch, magnetisch und mechanisch [117]. Die Art des verwendeten Sensors hat auch Einfluss auf Faktoren wie Genauigkeit, Auflösung, Latenz, Bereich für die Gestenerfassung etc., eine Betrachtung unterschiedlicher Gestenerkennungssysteme nach Sensortypen findet sich in [117].

Des Weiteren kann man zwischen Systemen unterscheiden, die vom Nutzer getragen werden müssen (z. B. ein Datenhandschuh [118]), oder Systemen, die nicht invasiv sind, wie z. B. bildbasierte Systeme. Bei den bildbasierten Systemen gibt es auch markerbasierte Ansätze, beispielsweise wird in [119] eine Gestenerkennung durch Infrarottracking vorgestellt.

Trotz der unterschiedlichen Eigenschaften besteht ein System zur Gestenerkennung aus folgenden wesentlichen Komponenten: 1) die Geste, die erkannt werden soll, 2) ein Sensor, der verwendet wird, um die Geste zu erfassen, 3) ein Vorverarbeitungsschritt, der je nach eingesetztem Sensortyp unterschiedlich ist (z. B. wird im Bereich der Bildverarbeitung versucht anhand einer Segmentierung zunächst den relevanten Bildteil für die Gestenerkennung zu bestimmen), 4) eine Merkmalsextraktion, hierbei werden Merkmale bestimmt, anhand derer sich eine Geste erkennen lässt und schließlich 5) eine Klassifikation, die eine Geste aus einem definierten Gestenalphabet bestimmt bzw. bei mangelnder Klassifikationsgüte keine Geste auswählt. Die Übergabe des Klassifikationsergebnisses an eine höhere Verarbeitungsebene (z. B. Dialogsystem) kann eine weitere mögliche Komponente eines Gestenerkennungssystems sein. Je nach verwendetem Sensor und eingesetzter Geste gibt es Variationen dieser Komponenten – es kann beispielsweise nach der Merkmalsextraktion eine Merkmalsselektion mit anschließender Merkmalsreduktion erfolgen. Ein Überblick über den prinzipiellen Aufbau eines Gestenerkennungssystems ist in Abbildung 4.3 dargestellt.

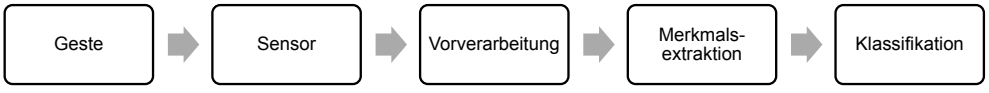


Abbildung 4.3: Prinzipieller Aufbau eines Gestenerkennungssystems

In diesem Kapitel werden drei bildbasierte Ansätze vorgestellt, dennoch gibt es

auch bei dieser Art von Ansatz viele Variationsmöglichkeiten: zweidimensionale (2-D) und dreidimensionale (3-D) Ansätze, die Anzahl der verwendeten Kameras, die ausgewählten Merkmale (*low-level*, *high-level*), die Art der Repräsentation (modellbasiert, erscheinungsbasiert), Farbräume etc. Aufgrund der Tatsache, dass es viele unterschiedliche Aktivitäten für die Erkennung von Gesten in der MMI gibt, wird für eine ausführliche Aufstellung und Bewertung von unterschiedlichen Ansätzen auf [115, 120, 114, 121, 104, 117] verwiesen. Im Folgenden werden kurz einige Ansätze, die Überschneidungen mit den hier präsentierten Ansätzen in Bezug auf Einsatzgebiet, Merkmalsextraktion und Klassifikation aufweisen, vorgestellt.

Der Ansatz zur Erkennung von statischen Handgesten bedient sich des Microsoft Kinect-Sensors [122] zur Datenerfassung, ein ähnlicher Ansatz findet sich in [123], wobei ein *Template Matching*-Ansatz verwendet wird, um die statischen Gesten der Hand zu erkennen. In [124] wird der Kinect-Sensor verwendet, um eine 3-D-Verfolgung der Hand zu ermöglichen, hierbei wird mit einem 3-D-Handmodell die Hand in beinahe Echtzeit (15 Hz) verfolgt. Die Erkennung von fingerbasiertem Buchstabieren wird in [125] auch mithilfe des Kinect-Sensors realisiert, generell bietet dieser Sensor vielseitige Einsatzmöglichkeiten, und es werden in Zukunft noch weitere Arbeiten diesen Sensor bzw. dessen Nachfolger verwenden. Der Ansatz zur statischen Handgestenerkennung verwendet unter anderem Merkmale, die auf dem sogenannten Histogramm orientierter Gradienten (engl. *Histogram of Oriented Gradients*, HOG) [126] basieren. Ein ähnlicher Ansatz, der HOG-Merkmale für die Erkennung von britischer Zeichensprache verwendet, ist der von [127], in seinem Fall wird ähnlich wie beim Ansatz zur dynamischen Handgestenerkennung eine Vorverarbeitung, basierend auf Farbinformationen zur Detektion der Hand, verwendet. Für die Erkennung von dynamischen Gesten gibt es viele unterschiedliche Ansätze, die auf HMMs beruhen, beginnend mit der Arbeit von [128], in der unterschiedliche Gesten beim Tennisspiel erkannt werden, bis hin zu weiteren Ansätzen, insbesondere aus dem Anwendungsbereich der Zeichensprache [129, 130]. In [131] werden Kopfgesten mittels einer HMM-Klassifikation erkannt. Kopfgesten bieten die Möglichkeit schnell und einfach Zustimmung bzw. Ablehnung zu signalisieren, daneben können sie wie z. B. in [132] verwendet werden, um durch die Dokumente zu navigieren.

Der Einsatz von Gesten ist unter anderem im Bereich der Robotik interessant, denn somit kann man einerseits mit einem mobilen Robotersystem interagieren, falls sich dieses gerade in Bewegung befindet, anderseits können auch Gesten genutzt werden, um schnell und intuitiv unterschiedliche Greifprozesse unmittelbar an einem Roboterarm einzuprogrammieren. Generell kann man laut [133] folgende Gründe für den Einsatz von Gesten im Bereich der Robotik finden: erstens, durch Gesten kann eine redundante Kommunikation zwischen Mensch und Roboter verwirklicht werden, sodass es für bestimmte Sprachkommandos (z. B. Stopp) auch eine entsprechende Geste gibt, zweitens, kann man dem Roboter auf einfache Art und Weise die Position verschiedener Objekte angeben (z. B. mit einer Zeigegeste), drittens, eine intuitive Interaktion, da

unterschiedliche Modalitäten des Menschen gleichzeitig interpretiert werden können (so lässt sich beispielsweise die Aufgabe für den Roboter, ein bestimmtes Objekt an einem bestimmten Ort abzulegen, für den Menschen am einfachsten mit einer Zeigegeste und einem kurz Sprachkommando befehlen). In [134] wird ein HMM-basiertes Gestenerkennungssystem, mit dem man eine Roboterplattform steuern kann, vorgestellt. Weitere Gestenerkennungssysteme, die Anwendungen im Bereich der Robotik haben, sind unter anderem die Arbeiten [135, 136, 137], darüber hinaus findet sich in [104] eine Aufstellung unterschiedlicher Gestenerkennungsansätze im Bereich der Robotik.

4.2 Architektur

In diesem Kapitel werden drei verschiedene Ansätze zur Erkennung von statischen bzw. dynamischen Handgesten und Kopfgesten vorgestellt. Die Prozesse zur Vorverarbeitung, Merkmalsextraktion und Klassifikation sind für die jeweiligen Ansätze unterschiedlich, dennoch sind die einzelnen Aspekte der Verarbeitung in einer einheitlichen Software-Architektur eingebettet. Diese Software-Architektur dient als *Middleware* und sorgt somit für eine reibungslose Kommunikation zwischen den einzelnen Software-Modulen der verschiedenen involvierten Systeme. Prinzipiell stehen für diese Art von Anforderungen unterschiedliche Systeme als *Middleware* zur Verfügung, eine Aufstellung über unterschiedliche *Middlewares* für den Einsatz im Bereich der Robotik findet sich in [138].

Zur Bereitstellung einer einheitlichen Architektur für die Gestenerkennung ist die Realzeitdatenbasis (engl. *real-time database*, RTDB) [139, 140] verwendet worden, um sowohl die Datenverarbeitung auf *low-level*-Ebene (z. B. Sensordaten) als auch auf *high-level*-Ebene (z. B. Klassifikationsergebnisse) in eine einheitliche Kommunikationsstruktur einzubetten. Es gab drei wesentliche Veranlassungen für den Einsatz der RTDB: Erstens soll für die Gestenerkennung eine echtzeitfähige Verarbeitung möglich sein, damit eine natürliche Interaktion zwischen Nutzer und Maschine verwirklicht werden kann. Zweitens soll mit einer einheitlichen Struktur für die Verarbeitung von unterschiedlichen Gesten dafür gesorgt werden, dass das Dialogsystem, das für die Interaktion zwischen Nutzer und Maschine zuständig ist, auf eine einheitliche Art und Weise Informationen über die aktuellen und vergangenen Ereignisse erhalten kann. Drittens kann aufgrund des Ringpuffers der RTDB für eine bestimmte Zeit auf vergangene Daten, extrahierte Merkmale und Entscheidungen zurückgegriffen werden. Dieser Sachverhalt erwies sich für das Dialogsystem bei der Entscheidungsfindung als nützlich.

Nachfolgend werden einige Aspekte der RTDB kurz vorgestellt, weitere Details zur RTDB finden sich in [139, 140].

Ursprünglich ist die RTDB in kognitiven autonomen Fahrzeugen verwendet wor-

den [139], dennoch wird diese Software-Architektur auch in anderen Bereichen wie beispielsweise in der Mensch-Roboter-Kooperation im industriellen Umfeld eingesetzt [141]. Die RTDB ist in der Lage mit Datenströmen zurechtzukommen, die unterschiedliche Eigenschaften bezüglich Datenrate und Paketverlust haben. Die RTDB bietet die Möglichkeit eines *Shared-Memory*-Zugriffs auf Daten, die für eine definierte Zeitperiode T in einem Ringpuffer vorrätig gehalten werden, zudem ist es möglich, Daten in eine Datei zu speichern. Zwei wesentliche Merkmale der RTDB, die für den Einsatz zur Gestenerkennung eine Rolle gespielt haben, sind: erstens, unterschiedliche Software-Module können gleichzeitig, ohne jegliche Blockierungen, auf die Daten aus einer Quelle zugreifen, z. B. ist es möglich, dass ein Videokamerabild von zwei unterschiedlichen Prozessen sowohl für die Erkennung von dynamischen Handgesten als auch für die Erkennung von Kopfgesten verwendet wird, zweitens, Daten, die aus verschiedenen Quellen mit unterschiedlichen Aktualisierungsraten stammen, können in einer quasi-synchronen Weise verarbeitet werden, da der interne Zeitstempel der RTDB für ein einheitliches Zeitformat der Daten sorgt. Der Datenverarbeitung mittels der RTDB liegen zwei wesentliche Konzepte zugrunde: Mit sogenannten *RTDB-Writern* werden Daten in die RTDB geschrieben, während diese Daten mit sogenannten *RTDB-Readern* wieder ausgelesen werden können. Für jeden Datentyp (z. B. Videokamerabild, Kinect-Tiefenbild) muss ein Datenbehälter definiert sein, mit dem dann die jeweiligen Daten in die RTDB geschrieben werden. Das Konzept der *RTDB-Writer* und *RTDB-Reader* kann sowohl auf *low-level*-Datenebene als auch auf *high-level*-Datenebene verwendet werden, somit ist mit der RTDB eine Partitionierung z. B. eines Gestenerkennungssystems in einzelne Software-Module möglich. Des Weiteren können auch Verarbeitungsergebnisse aus einem Software-Modul (z. B. der Gesichtsdetektion) von unterschiedlichen Ansätzen zur Gestenerkennung genutzt werden, wodurch sich Ressourcen einsparen lassen, da die gleichen Verarbeitungsschritte für unterschiedliche Ansätze nicht wiederholt werden müssen.

Die einzelnen Komponenten für die Gestenerkennung werden in die RTDB-Architektur eingebunden. Die Ausgangssensordaten, entweder die Daten des Microsoft Kinect-Sensors oder das Bild von einer Videokamera, werden als Erstes in der RTDB erfasst. Ausgehend von diesen *low-level*-Daten, werden die Ergebnisse der einzelnen Verarbeitungsschritte wieder in die RTDB geschrieben, bis zum Schluss ein Klassifikationsergebnis vorliegt. Die Steuerung der einzelnen Software-Module obliegt dem Dialogsystem, dieses kann, je nach Bedarf, einzelne Anwendungen starten oder beenden (weitere Details zum Dialogsystem bzw. Dialogmanager finden sich in [142]). Die Realisierung eines Gestenerkennungssystem in der RTDB ist in Abbildung 4.4 dargestellt.

Ein Problem bei der Gestenverarbeitung besteht darin, dass der exakte Zeitpunkt der Beginn einer Geste nicht feststeht (*gesture spotting*) [104]. Dieses Problem kann mit der RTDB angegangen werden, da aufgrund des Ringpuffers für eine bestimmte Dauer T einzelne Ergebnisse von den Verarbeitungsschritten zur Verfügung stehen.

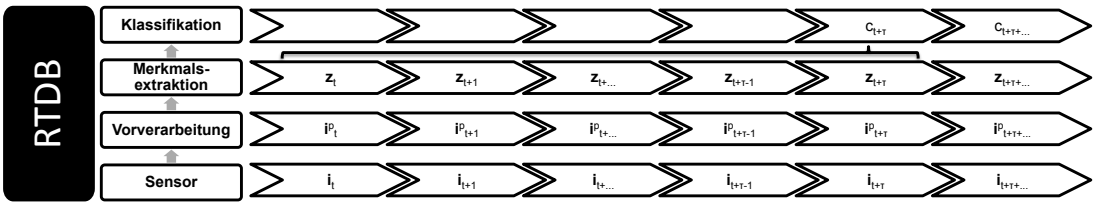


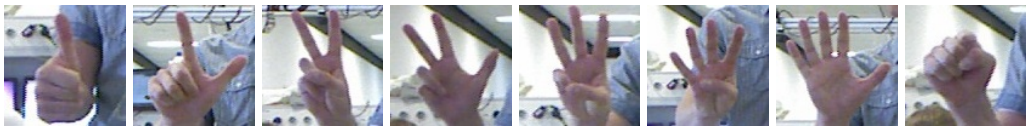
Abbildung 4.4: Einbettung eines Gestenerkennungssystems in die RTDB

Es ist also möglich mit einem *Sliding Window*-Verfahren Klassifikationsergebnisse für die Gesten der Länge τ zu erhalten und diese in die RTDB zu schreiben. Somit kann je nach Länge des Ringpuffers und der damit verbundenen Dauer T auch auf Klassifikationsergebnisse aus der Vergangenheit zugegriffen werden. Aufgrund dieses Sachverhalts wird das Problem des Auffindens des Startpunktes einer Geste angegangen, zusätzlich kann das Dialogsystem auf Gestenereignisse der Vergangenheit zurückgreifen. Überdies wird der Ringpuffer dazu verwendet, über eine bestimmte Zeitlänge $\Delta t < \tau$ die Klassifikationsergebnisse zu mitteln, wodurch die Robustheit für die Erkennungsleistung erhöht wird.

4.3 Statische Handgestenerkennung im Tiefenbild

Im Folgenden wird ein Ansatz beschrieben, der aus einem Tiefenbild statische Handgesten erkennt, hierzu ist der Microsoft Kinect-Sensor [122] verwendet worden. Der Kinect-Sensor verwendet einen strukturierten Licht-Ansatz [143], um ein Tiefenbild zu erstellen.

Insgesamt werden acht verschiedene Handgesten unterschieden, dabei geht es um die Darstellung von Zahlen mit den Fingern. Neben den fünf Gesten für die Zahlen *Eins* bis $Fünf$ wurden auch noch die Faust sowie zwei alternative Darstellungsformen für die Zahlen *Zwei* und *Drei* in das Gestenalphabet aufgenommen. Das gesamte Alphabet ist in Abbildung 4.5 dargestellt.



(a) *Eins* (b) *Zwei* (c) $Zwei_{alt}$ (d) *Drei* (e) $Drei_{alt}$ (f) *Vier* (g) $Fünf$ (h) *Faust*

Abbildung 4.5: von links nach rechts: Das Gestenalphabet für die statischen Handgesten (Zahlen *Eins* – $Fünf$, *Faust*), Abbildung 4.5(c) und Abbildung 4.5(e) zeigen alternative Darstellungsformen für *Zwei* bzw. *Drei*

4.3.1 Realisierung

Insgesamt umfasst der Ansatz zur statischen Handgestenerkennung fünf wesentliche Schritte: Bilderfassung, Handkonturextraktion, Rotationskompensation, Merkmalsextraktion und eine abschließende Klassifikation. Eine Übersicht über die einzelnen Schritte liefert Abbildung 4.6.

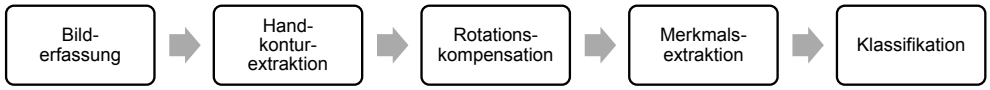


Abbildung 4.6: Ablaufdiagramm für die statische Handgestenerkennung im Tiefenbild

Die einzelnen Schritte werden für jedes Videokamerabild durchlaufen, daher ist es wichtig bei der Ausgestaltung der einzelnen Schritte dafür zu sorgen, dass eine echtzeitfähige Verarbeitung der statischen Handgeste möglich ist, damit eine natürliche Interaktion zwischen Nutzer und Maschine umgesetzt werden kann.

4.3.1.1 Handkonturextraktion

Nach der Microsoft Kinect-basierten Erfassung des Tiefenbildes ist der erste wichtige Schritt für die statische Handgestenerkennung die Extraktion der Handkontur aus dem Tiefenbild. Der Handmittelpunkt $\mathbf{p}_h = (x_h, y_h, z_h)^T$ wird mithilfe der OpenNI-Software [144] bestimmt, somit kann dieser Punkt als eine erste Abschätzung dienen, um anschließend die Handkontur zu extrahieren. Aufgrund der Tatsache, dass die z-Komponente des bestimmten Handmittelpunktes nicht immer tatsächlich auf der Hand liegt (eine Verschiebung in der z-Achse kann auftreten, wenn der Handmittelpunkt zwischen zwei geöffnete Finger gesetzt wird), und es somit zu falschen z-Werten kommen kann, bedarf es einer Hilfestellung. Diese Hilfestellung verwendet eine 2-D-Gauß-Verteilung mit folgendem Mittelwert μ und folgender Kovarianzmatrix Σ:

$$\mu = \begin{pmatrix} x_h \\ y_h \end{pmatrix}, \qquad \Sigma = \begin{pmatrix} \frac{k}{d} & 0 \\ 0 & \frac{k}{d} \end{pmatrix}.$$

k ist ein konstanter Faktor und d gibt den Abstand zwischen dem Nutzer und der Kamera an. Anhand dieser 2-D-Gauß-Verteilung werden in der x,y-Ebene um den Handmittelpunkt 31 Punkte ausgewählt. Eine anschließende Median-Berechnung bestimmt nun die neue z_h^s-Komponente für den geglätteten Handmittelpunkt $\mathbf{p}_h^s = (x_h, y_h, z_h^s)^T$. Die Handkontur wird um $\mathbf{p}_h^s$ gemäß der Entfernung zur Kamera folgendermaßen extrahiert. In näheren Bereichen (1 m–2 m) wird eine Variation von 5 % der z-Werte erlaubt, um die Handfläche zu bestimmen, während mit zunehmendem Abstand zur Kamera ($\geq$ 2 m) die Variation sukzessive bis maximal 11 % erhöht wird. Der Anstieg der Variation der z-Werte ist durch die Tatsache bestimmt, dass sich mit zunehmendem Abstand

zwischen dem Nutzer und der Kamera das Bildrauschen erhöht. Das so erhaltene Bild der Handfläche wird zu einer festen Größe von 64×64 Pixel (px) skaliert. Morphologische Operationen [145] werden zur Rauschreduzierung eingesetzt, des Weiteren dienen diese dazu, ein gutes Abstraktionsniveau für das Handkonturbild $\mathbf{I}_c(x,y)$ zu erhalten.

4.3.1.2 Rotationskompensation

Die Kompensation der Rotation hat sich als ein wichtiger Bestandteil in der Verarbeitung erwiesen, da Personen dazu neigen ihre statischen Handgesten mit unterschiedlichen Drehungen zu zeigen. Zudem war einer der beiden Merkmalsextraktionsansätze, der auf den HOG-Merkmalen basiert, nicht rotationsinvariant. Es wurden zwei unterschiedliche Methoden angewendet, um die Rotation der gezeigten statischen Handgeste zu kompensieren. Die erste Methode berechnete die Momente zweiter Ordnung, um die Neigungsunterschiede in den Gesten zwischen den verschiedenen Personen auszugleichen. Die zweite Methode basierte auf einem Fingermodell, um die Rotationsunterschiede zu kompensieren.

Momente zweiter Ordnung
Die erste Methode zur Rotationskompensation dreht das Handkonturbild $\mathbf{I}_c(x,y)$ um seinen Schwerpunkt $\mathbf{p}_{cg}$, indem, anhand der Momente zweiter Ordnung, der Rotationswinkel α folgendermaßen berechnet wird [146]:

$$\alpha = \frac{1}{2} \arctan\left(\frac{2\mu_{11}}{\mu_{20} - \mu_{02}}\right). \tag{4.1}$$

Dabei gilt für die Momente μ_{pq}:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q \mathbf{I}_c(x,y) \tag{4.2}$$

und den Schwerpunkt $\mathbf{p}_{cg}$:

$$\mathbf{p}_{cg} = (\bar{x}, \bar{y})^T \qquad \bar{x} = \frac{\sum_x \sum_y x \cdot \mathbf{I}_c(x,y)}{\sum_x \sum_y \mathbf{I}_c(x,y)} \qquad \bar{y} = \frac{\sum_x \sum_y y \cdot \mathbf{I}_c(x,y)}{\sum_x \sum_y \mathbf{I}_c(x,y)}. \tag{4.3}$$

Diese Methode erzielt prinzipiell gute und schnelle Ergebnisse, dennoch haben sowohl Artefakte im Bild als auch die Form des Handgelenks einen Einfluss auf die Bestimmung des Schwerpunktes sowie der Momente. Diese Änderungen ziehen somit auch Auswirkungen auf den Rotationswinkel nach sich.

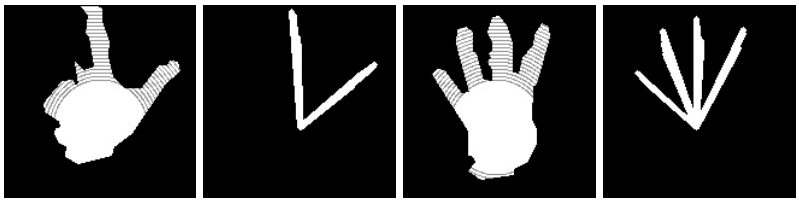
Fingermodell

Generell ist es sehr wichtig für die Merkmalsextraktion sowie für die anschließende Klassifikation, dass die Finger in geeigneter Art und Weise ausgerichtet sind. Deswegen ist eine Methode zur Rotationskompensation entwickelt worden, die sich von den Störfaktoren (Artefakte im Bild, Form des Handgelenks) nicht beeinflussen lässt. Die neue Methode verwendet Informationen über die ausgestreckten Finger, um den Rotationswinkel α zu bestimmen.

Das skalierte Handkonturbild $\mathbf{I}_c(x,y)$ ist der Ausgangspunkt für das Fingermodell, hierbei werden Kreise mit unterschiedlichen Radien um den Schwerpunkt $\mathbf{p}_{cg}(x,y)$ erzeugt, um so die einzelnen Finger zu detektieren und auf dieser Grundlage einen Rotationswinkel α zu bestimmen. Der Radius r der Kreise beginnt mit 40 px und wird schrittweise um je 3 px erhöht, bis der erzeugte Kreis komplett das Handkonturbild $\mathbf{I}_c(x,y)$ verlässt. Der Schnittpunkt zwischen der Kreislinie und dem Bild wird untersucht, und dabei werden folgende zwei Regeln für die Untersuchung angewandt. Nur wenn beide Aussagen wahr sind, wird angenommen, dass es sich um einen Finger handelt.

- 1. Regel: Der Abstand d_i zwischen dem Ausgangspunkt einer Überschneidung in einem Kreisdurchlauf und dem Endpunkt dieser Überschneidung muss innerhalb von einer Länge von $2px \leq d_i \leq 8px$ liegen.
- 2. Regel: Mindestens vier Überschneidungen d_i von unterschiedlichen Kreisradien müssen übereinander liegen.

Anhand der erhaltenen Finger wird nun der Rotationswinkel α bestimmt, dabei werden nur die erkannten Fingerstrukturen berücksichtigt. Für diese Strukturen wird wiederum der Rotationswinkel α anhand der Momente zweiter Ordnung bestimmt. In Abbildung 4.7 wird für die Gesten *Zwei* und *Vier* das entsprechende Fingermodell gezeigt. Die Kreise zur Schnittbildung werden in Schwarz dargestellt, die entsprechenden Fingermodelle sind in Abbildung 4.7(b) bzw. in Abbildung 4.7(d) zu sehen.



(a) Geste *Zwei* (b) Fingermodell für die Handgeste *Zwei* (c) Geste *Vier* (d) Fingermodell für die Handgeste *Vier*

Abbildung 4.7: Schnittbildung und erhaltenes Fingermodell für die Zahlen *Zwei* und *Vier*

4.3.1.3 Merkmalsextraktion

Im Folgenden werden zwei unterschiedliche Ansätze zur Merkmalsextraktion betrachtet: Zernike-Momente und HOG-Merkmale.

Zernike-Momente
Zernike-Momente [147] (siehe Gleichung 4.4) werden anhand der sogenannten Zernike-Polynome $V_{mn}(\rho,\theta)$ berechnet, diese Polynome sind in [148] vorgestellt worden. Die Zernike-Momente A_{mn} einer Bildfunktion $\mathbf{I}(x,y)$ können auf folgende Art und Weise bestimmt werden:

$$A_{mn} = \frac{m+1}{\pi} \sum_x \sum_y \mathbf{I}(x,y) V^*_{mn}(\rho,\theta)\, dx\, dy, \tag{4.4}$$

mit $x^2 + y^2 \leq 1$, $m - |n| = gerade$ und $|n| \leq m$.

Der Faktor m definiert die Ordnung der Zernike-Momente, die invariant zu Translation, Skalierung und Rotation sind.

HOG-Merkmale
HOG-Merkmale sind in [126] für die Detektion von Fußgängern vorgestellt worden, sie können als semi-lokale Merkmalsdeskriptoren beschrieben werden, die die Gradienten über sogenannte *Zellen* berechnen. Eine *Zelle* ist eine kleine zusammenhängende Menge von Pixeln (engl. *pixel blobs*). Mehrere dieser Zellen werden zu einer größeren Einheit, dem sogenannten *Block*, zusammengefasst. Mithilfe der *Blöcke* wird eine Kontrast-Normalisierung des Bildes in Bezug auf Störungen (z. B. Lichtwechsel, Schatten etc.) durchgeführt. Für die Berechnung der Gradienten in x- bzw. y-Richtung können beispielsweise Sobel-Filter [80] verwendet werden, die berechneten Gradienten werden anschließend auf die Anzahl der gewählten *Bins* (diskrete Intervalle in einem Histogramm) verteilt. In [126] waren unterschiedliche Verfahren zur *Block*-Normalisierung vorgestellt worden, für den vorgestellten Ansatz zur Erkennung von statischen Handgesten erzielte eine *Maximum*-Normalisierung (verwendet das Maximum der absoluten Werte) die besten Ergebnisse. Der Ablauf zur Bestimmung der HOG-Merkmale ist in Abbildung 4.8 gezeigt.

Für die HOG-Merkmale ist es manchmal sinnvoll, je nach Dimensionierung der *Zellen* und *Blöcke* bzw. der Anzahl der verwendeten *Bins*, die Merkmale zu reduzieren. Prinzipiell gibt es dafür mehrere geeignete Verfahren, beispielsweise die Hauptkomponentenanalyse (engl. *Principal Component Analysis*, PCA) oder die Lineare Diskriminanzanalyse (engl. *Linear Discriminant Analysis*, LDA) [95]. Bei Vorversuchen zeigte sich durch eine Merkmalsreduktion mittels einer PCA eine signifikante Verbesserung der Ergebnisse, bei den hier vorgestellten Versuchen zog eine Merkmalsreduktion keine signifikante Verbesserung nach sich.

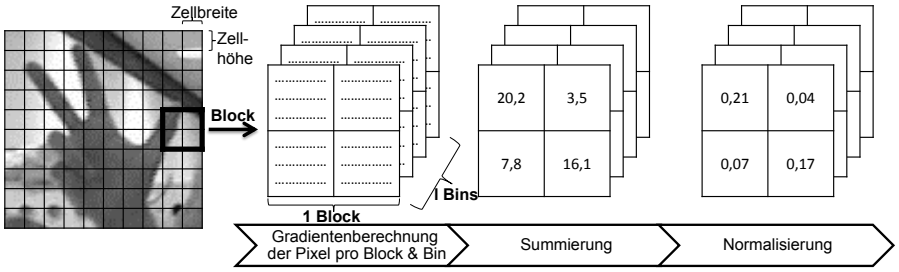


Abbildung 4.8: Ablaufdiagramm für die Bestimmung von HOG-Merkmalen

4.3.1.4 Klassifikation

Für die Klassifikation der acht statischen Handgesten wurden zwei unterschiedliche Klassifikatoren verwendet: erstens Klassifikation anhand von k-Nächste-Nachbarn (engl. *k-nearest neighbor*, k-NN) [95], diese bilden eine einfache und schnelle Klasse von Klassifikatoren, und zweitens Stützvektormethoden (engl. *Support Vector Machines*, SVMs) [149, 150], die ursprünglich eingeführt worden sind, um zwei unterschiedliche Muster bestmöglich zu trennen. Die Trennebene wird insbesondere von den Vektoren beeinflusst, die sich in der Nähe der Trennebene befinden. Die Distanzklassifikation bildet die Grundlage für beide Methoden, wobei bei der k-NN-Klassifikation die Distanz zu den Referenzvektoren betrachtet wird und bei SVMs die Distanz zur Trennebene. Im Verlauf der Untersuchungen zeigte sich, dass sich beim HOG-Ansatz mit der k-NN-basierten Klassifikation ab 64 Referenzvektoren keine signifikanten Verbesserungen ergaben, der Ansatz mit den Zernike-Momenten verzeichnete ab 32 Referenzvektoren keine signifikanten Verbesserungen mehr. Bei der SVM-basierten Klassifikation hatten die Erweiterungen mit Kernelfunktionen keine signifikanten Verbesserungen gebracht, deswegen wurden nur lineare SVMs eingesetzt.

4.3.2 Experimente

4.3.2.1 Datensatz

Zum Testen des Ansatzes zur Erkennung von statischen Handgesten aus einem Tiefenbild wurde ein Datensatz von insgesamt zehn Personen (acht Männer, zwei Frauen) aufgenommen. Es wurden die acht Gesten – *Eins*, *Zwei*, $Zwei_{alt}$, *Drei*, $Drei_{alt}$, *Vier*, *Fünf*, *Faust* – sowohl für die rechte als auch die linke Hand erfasst. Unten werden nur die Ergebnisse für die rechte Hand betrachtet, da diese immer etwas besser waren, als die Ergebnisse der linken Hand. Eine mögliche Ursache dafür könnte sein, dass der Datensatz zu 90 % aus Rechtshändern bestand. Das System wurde mit dem Verfahren *Leave-One-Person-Out* (siehe Abschnitt 3.2.1) getestet, hierbei wurde immer auf den Datensatz einer Person getestet, die übrigen neun Personen bildeten die Trainingsmenge. Insgesamt wurden 40 Beispiele pro Geste pro Versuchsperson genommen, somit

umfasste ein Durchlauf für die Kreuzvalidierung insgesamt 2880 Trainingsbeispiele bei 320 Testbeispielen.

4.3.2.2 Ergebnisse

Im Folgenden werden die Ergebnisse der beiden Merkmalsansätze – Zernike-Momente, HOG-Merkmale – vorgestellt. Zunächst wird der Einfluss der Kompensation der Rotation betrachtet. Bei dem Referenz-Ansatz, der auf den Zernike-Momenten basiert, ist zu beachten, dass die Zernike-Momente rotationsinvariant sind, daher sollte eine Kompensation der Rotation also hier theoretisch keinen wesentlichen Beitrag zur Verbesserung der Ergebnisse liefern. Demgegenüber kann bei den HOG-Merkmalen, die nicht rotationsinvariant sind, diese Vorverarbeitungsstufe einen Einfluss auf die Ergebnisse nehmen. Die vorgestellten Resultate umfassen nur diese, die jeweils die besten Ergebnisse geliefert haben, denn insbesondere bei den HOG-Merkmalen gibt es aufgrund der unterschiedlichen Gestaltungen von *Zellen*, *Blöcke* und *Bins* viele Konfigurationsmöglichkeiten.

Zunächst werden die Ergebnisse, die mit den Zernike-Momenten erzielt worden sind, vorgestellt. Tabelle 4.1 zeigt die erzielten Ergebnisse für die Erkennungsrate. Bei den Zernike-Momenten kann man mit m die Ordnung variieren, die besten Ergebnisse wurden mit einer Ordnung von $m = 10$ gefunden.

	Klassifikation	
Rotationskompensation	k-NN 32	SVM
Ohne	64,68 %	67,84 %
Momemente 2. Ordnung	69,69 %	72,22 %
Fingermodell	71,69 %	77,78 %

Tabelle 4.1: Übersicht über die erzielten Ergebnisse für den Ansatz, basierend auf den Zernike-Momenten. Das System wurde mit zwei Strategien zur Kompensierung der Rotation getestet.

Im Gegensatz zu den Zernike-Momenten gibt es bei dem auf HOG-Merkmalen basierenden Ansatz mehrere verschiedene Parameter, die variiert werden können, dazu zählen beispielsweise die Eigenschaften wie Höhe und Breite von *Zellen* und *Blöcken* sowie die Anzahl von *Bins*, überdies kann auch die *Block*-Normalisierung Einfluss auf die Ergebnisse ausüben. In einer Voruntersuchung wurden viele unterschiedliche Parameterkonfigurationen getestet, es zeigten sich folgende Erkenntnisse: Aufgrund der binären Darstellung der Handkontur aus dem Tiefenbild ist eine Anzahl von vier *Bins* ausreichend. Des Weiteren hat sich bei der Gestaltung des Verhältnisses von *Block*-Höhe und -Breite gezeigt, dass in der Regel die besten Ergebnisse erzielt werden, wenn dieses Verhältnis dem des untersuchten Bildes entspricht.

Tabelle 4.2 zeigt die erzielten Ergebnisse der Erkennungsrate des Ansatzes mit den HOG-Merkmalen. Im Gegensatz zu dem auf den Zernike-Momenten fußenden

Ansatz braucht es 64 Referenzvektoren für die k-NN-Klassifikation, damit die besten Ergebnisse erzielt werden.

	Klassifikation	
Rotationskompensation	k-NN 64	SVM
Ohne	72,13 %	78,01 %
Momente 2. Ordnung	76,29 %	80,90 %
Fingermodell	79,75 %	88,69 %

Tabelle 4.2: Übersicht über die erzielten Ergebnisse für den Ansatz, basierend auf den HOG-Merkmalen. Das System wurde mit zwei Strategien zur Kompensierung der Rotation getestet.

Bis auf eine Ausnahme (k-NN-Klassifikation mit Fingermodell) erzielte stets die HOG-Parameter-Konfiguration von Tabelle 4.3 die besten Ergebnisse.

HOG-Parameter-Konfiguration	
Block-Höhe	8 px
Block-Breite	8 px
Zellenanzahl in x-Richtung	2
Zellenanzahl in y-Richtung	2
Bin-Anzahl	4
Block-Normalisierung	*Maximum*

Tabelle 4.3: Übersicht über die verwendete HOG-Parameter-Konfiguration

Im Hinblick auf die Ausnahme gab es nur eine Veränderung bei *Block*-Breite von 8 px auf 16 px, ansonsten blieben die Parameterwerte gleich.

Diskussion: Der Ansatz mit den HOG-Merkmalen lieferte durchweg bessere Ergebnisse als der Zernike-Momente-Ansatz. Darüber hinaus zeigt sich eine Überlegenheit der SVM-basierten Klassifikation gegenüber der k-NN-Klassifikation. Während bei den HOG-Merkmalen durchaus damit zu rechnen war, dass eine Kompensation der Rotation sich auf die Klassifikation positiv auswirkt, konnte man bei dem auf den Zernike-Momenten basierenden Ansatz diesen Effekt aufgrund ihrer Rotationsinvarianz eigentlich nicht erwarten. Eine mögliche Erklärung, dass die Rotationskompensation beim Zernike-Momente-Ansatz einen positiven Effekt aufweist, könnte vielleicht sein, dass die Lernverfahren für die Klassifikatoren positiv beeinflusst worden sind.

4.4 Dynamische Handgestenerkennung

Die dynamischen Handgesten bestehen aus zwei Komponenten: Zunächst können verschiedene Bewegungsrichtungen – links, rechts, oben, unten – unterschieden werden,

des Weiteren wird auch die Form der Hand – Faust, geschlossene Hand – betrachtet. Eine Beispielsequenz für eine Geste ist in Abbildung 4.9 dargestellt.

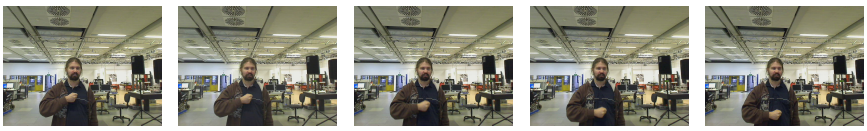


Abbildung 4.9: Beispielsequenz für dynamische Handgesten, hier wird eine Handbewegung nach unten dargestellt

Im Gegensatz zum vorherigen Ansatz wurde für die Erkennung von dynamischen Handgesten ein gewöhnliches 2-D-Videokamerabild verwendet. Die Ursache hierfür ist die Tatsache, dass der Ansatz zur Erkennung von dynamischen Handgesten in enger Abstimmung bzw. in Überschneidung zum Ansatz zur Erkennung von Kopfgesten (siehe Abschnitt 4.5) entwickelt worden ist. Daher teilen sich beide Ansätze den Vorverarbeitungsschritt zur Gesichtsdetektion mithilfe des Ansatzes von Viola und Jones [151, 152]. Zudem war es aufgrund der Gesicht-Modellanpassung (engl. *face model fitting*) des CANDIDE-3-Modells [153] möglich, ein gutes und robustes Hautfarbenmodell zu erstellen, um damit die Hand zu finden.

4.4.1 Realisierung

Insgesamt umfasst der Ansatz fünf wesentliche Schritte: Bilderfassung, Gesichtsdetektion, Erstellung eines adaptiven Hautfarbenmodells, Merkmalsextraktion und eine abschließende Klassifikation. Ein Übersicht ist in Abbildung 4.10 dargestellt.

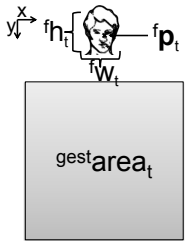
Abbildung 4.10: Ablaufdiagramm für die dynamische Handgestenerkennung im Videokamerabild

4.4.1.1 Gesichtsdetektion und Erstellung eines Hautfarbenmodells

Bevor die Klassifizierung der dynamischen Handgesten erfolgt, muss zunächst die Hand gefunden werden, dafür wird für die Erstellung eines Hautfarbenmodells zunächst das Gesicht gesucht. Der Ansatz von Viola und Jones [151, 152] zur Gesichtsdetektion erfreut sich in der Bildverarbeitung großer Beliebtheit, da er sowohl schnelle als auch zuverlässige Ergebnisse liefert, bei mehreren gefundenen Gesichtern wird bei diesem Ansatz zur Gestenerkennung immer das größte Gesicht gewählt.

Basierend auf dem Farbhistogramm des detektierten Gesichts (hierfür werden anhand des CANDIDE-3-Modells bestimmte Regionen für das Hautfarbenmodell ausgewählt), wird ein Farbfilter entwickelt, das die Hand finden soll. Die Hand wird in einem Bild gesucht, in dem alle Gesichtshypothesen, die die Gesichtsdetektion geliefert hat, herausgefiltert worden sind, somit wird nicht irrtümlicherweise ein Gesicht als Hand verwendet. Nach dieser Filteroperation werden morphologische Operationen [145] zur Reduzierung des Rauschens eingesetzt. Ähnlich wie in [154], wobei anhand der Kopfposition ein Aktionsradius bestimmt wird, werden bei diesem Ansatz, basierend auf der Größe ${}^{f}area_t = {}^{f}w_t \times {}^{f}h_t$ und Position ${}^{f}\mathbf{p}_t = ({}^{f}x_t, {}^{f}y_t)^T$ des detektieren Gesichtes, Restriktionen für die Größe ${}^{h}area_t$ sowie die Position ${}^{h}\mathbf{p}_t = ({}^{h}x_t, {}^{h}y_t)^T \in {}^{gest}area_t$ der Hand abgeleitet:

$$
\begin{aligned}
0,3 \cdot {}^{f}area_t \leq {}^{h}area_t &\leq 1,2 \cdot {}^{f}area_t \\
{}^{f}x_t - 2 \cdot {}^{f}w_t \leq \quad {}^{h}x_t \quad &\leq {}^{f}x_t + 2 \cdot {}^{f}w_t \\
{}^{f}y_t + 1 \cdot {}^{f}w_t \leq \quad {}^{h}y_t \quad &\leq {}^{f}y_t + 5 \cdot {}^{f}w_t
\end{aligned}
$$



Wiederum wird nur die größte Kontur als Hand gewählt, die sich innerhalb der oben genannten Grenzen befindet, somit erhält man das Handkonturbild $\mathbf{I}_c(x,y,t)$ für den Zeitpunkt t.

4.4.1.2 Merkmalsextraktion

Für die Klassifizierung der dynamischen Handgesten werden mehrere Merkmale verwendet, dabei gibt es Merkmale, die sich auf die Position des Handkonturbilds $\mathbf{I}_c(x,y,t)$ beziehen, sowie Merkmale, die die Form des Handkonturbilds $\mathbf{I}_c(x,y,t)$ beschreiben.

Die Position des Handkonturbilds $\mathbf{I}_c(x,y,t)$ wird anhand des Schwerpunktes $\mathbf{x}_t^m = (\bar{x}_t, \bar{y}_t, t)^T$ (siehe Gleichung 4.3) beschrieben. Da es sich um dynamische Gesten handelt, ist nicht die absolute Position interessant, sondern die zeitliche Veränderung der Position, daher wird für die Klassifikation die Differenz zwischen zwei Zeitpunkten betrachtet, diese wird folgendermaßen beschrieben:

$$\Delta\mathbf{x}_\tau^m = \mathbf{x}_\tau^m - \mathbf{x}_{\tau-1}^m \quad \forall\, \tau \subseteq \{1,\ldots,t,\ldots,T\}^1, \quad \mathbf{x}_{\tau=0}^m = \mathbf{0}. \tag{4.5}$$

Diese Differenzbildung kann auf einfache Art und Weise mit der RTDB-Architektur realisiert werden.

[1] t beschreibt einen bestimmten Zeitpunkt, während τ als eine Laufvariable für die Zeit dient.

Für die Beschreibung der Form der Handkontur werden die sogenannten Hu-Momente [155] verwendet. Diese Merkmale, insgesamt aus sieben Momenten bestehend, sind invariant gegenüber Skalierung, Translation und Rotation. Basierend auf den Momenten μ_{pq} der Gleichung 4.2 können die sieben Hu-Momente folgendermaßen berechnet werden:

$$hu_1 = \mu_{20} + \mu_{02} \tag{4.6}$$

$$hu_2 = (\mu_{20} - \mu_{02})^2 + (2\mu_{11})^2 \tag{4.7}$$

$$hu_3 = (\mu_{30} - 3\mu_{12})^2 + (3\mu_{21} - \mu_{03})^2 \tag{4.8}$$

$$hu_4 = (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2 \tag{4.9}$$

$$\begin{aligned} hu_5 = &(\mu_{30} - 3\mu_{12})(\mu_{30} + \mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] \\ &+ (3\mu_{21} - \mu_{03})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \end{aligned} \tag{4.10}$$

$$\begin{aligned} hu_6 = &(\mu_{20} - \mu_{02})[(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \\ &+ 4\mu_{11}(\mu_{30} + \mu_{12})(\mu_{21} + \mu_{03}) \end{aligned} \tag{4.11}$$

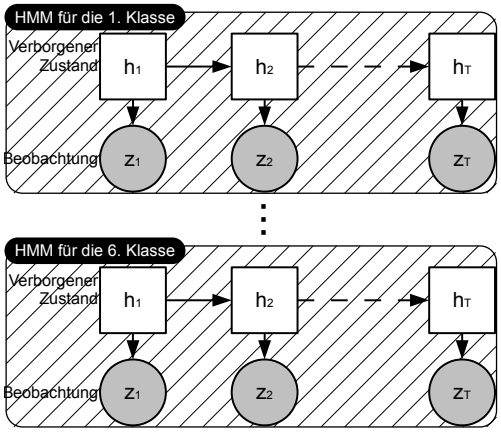
$$\begin{aligned} hu_7 = &(3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] \\ &+ (\mu_{30} - 3\mu_{12})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \end{aligned} \tag{4.12}$$

Der Vektor $\mathbf{x}_t^s = (hu_{1_t}, hu_{2_t}, hu_{3_t}, hu_{4_t}, hu_{5_t}, hu_{6_t}, hu_{7_t})^T$ beschreibt die formbezogenen Merkmale, während die positionsbezogenen Merkmale mit dem Vektor $\Delta\mathbf{x}_t^m$ beschrieben werden. Beide Vektoren – $\mathbf{x}_t^s, \Delta\mathbf{x}_t^m$ – werden im Vektor $\mathbf{z}_t$ zusammengefasst und beschreiben die gesamte Menge an Beobachtungen, die zur Verfügung steht und für die Klassifikation genutzt wird.

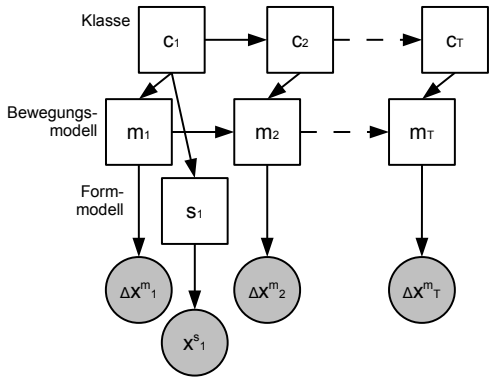
4.4.1.3 Klassifikation

Im Gegensatz zum Ansatz von Abschnitt 4.3 sollen nun dynamische Gesten erkannt werden, daher werden Klassifikatoren verwendet, die besser geeignet sind, um solche Gesten zu erkennen. Es kommen zwei Arten von Klassifikatoren zum Einsatz: HMMs und ein GM. Der erste Ansatz nutzt HMMs, hierbei wird für jede Gestenklasse ein HMM trainiert, das die Bewegung und Form einer Klasse modelliert. Unter Berücksichtigung der Abläufe für die Gesten wurde dieser HMM-Ansatz modifiziert und resultierte in einem GM-Ansatz. Die Klassifikatoren (HMM-Ansatz, GM-Ansatz) sind in Abbildung 4.11 dargestellt.

Der Vektor $\mathbf{z}_\tau = \{\mathbf{x}_\tau^s, \Delta\mathbf{x}_\tau^m\}$ bildet die Ausgangsgrundlage für die Klassifikation, dennoch werden die Merkmale unterschiedlich bei beiden Klassifikatoren eingesetzt. Der HMM-Ansatz nutzt den Vektor $\mathbf{z}_\tau$ zur Klassifikation und macht keine Unterscheidung zwischen den formbezogenen Merkmalen $\mathbf{x}_\tau^s$ bzw. positionsbezogenen Merkma-



(a) HMMs für die dynamische Handgestenerkennung



(b) GM für die dynamische Handgestenerkennung

Abbildung 4.11: Übersicht über die verwendeten Klassifikatoren zur Erkennung der dynamischen Handgesten

len $\Delta\mathbf{x}_\tau^m$. Eine Unterscheidung zwischen den formbezogenen und positionsbezogenen Merkmalen wird jedoch beim GM-Ansatz vorgenommen.

HMM-Ansatz

Der HMM-Ansatz bildet den Referenzansatz, hierbei werden alle beobachteten Merkmale $\widehat{\mathcal{Z}}_T = \{\widehat{\mathcal{X}}_T^s, \Delta\widehat{\mathcal{X}}_T^m\}$ zur Klassifikation herangezogen. Jede der sechs unterschiedlichen Gesten wird mit einem HMM modelliert, während der Bewegungsverlauf der jeweiligen Geste über die verborgene Zustandssequenz $h_1, h_2, \ldots, h_T$ abgebildet wird. Das Gestenalphabet $\mathcal{C} = \{{}^1c, {}^2c, \ldots, {}^6c\}$ umfasst insgesamt sechs Klassen. Anhand des *Baum-Welch*-Algorithmus [85, 84] wird für jede Klasse ${}^kc$ ein HMM-Parametersatz ${}^k\lambda$ bestimmt. Für jede der sechs Gestenklassen ${}^kc \in \mathcal{C}$ kann nun die Produktionswahrscheinlichkeit für den jeweiligen HMM-Parametersatz ${}^k\lambda$ ermittelt werden, um somit die wahrscheinlichste Klasse $\hat{c}$ zu bestimmen. Für die erfasste Beobachtungssequenz $\widehat{\mathcal{Z}}_T = \{\widehat{\mathbf{z}}_1, \ldots, \widehat{\mathbf{z}}_T\}$ kann nun die wahrscheinlichste Klasse $\hat{c} \mathrel{\hat{=}} \hat{\lambda}$ wie folgt bestimmt werden:

$$\hat{\lambda} = \arg\max_k p(\widehat{\mathcal{Z}}_T | {}^k\lambda) \tag{4.13}$$

$$= \arg\max_k \sum\nolimits_{\forall h \in \mathcal{H}_T} p(\widehat{\mathcal{Z}}_T, \mathcal{H}_T | {}^k\lambda). \tag{4.14}$$

Gleichung 4.14 lässt sich anhand des sogenannten Vorwärtsalgorithmus (siehe [84]) effizienter berechnen. Die Produktionswahrscheinlichkeit für ein HMM mit dem Parametersatz ${}^k\lambda$ kann nun folgendermaßen für den Zeitpunkt T bestimmt werden:

$$p(\widehat{\mathcal{Z}}_T|^k\lambda) = \sum_{i=1}^{N} \alpha_T(i) \tag{4.15}$$

Hierbei wird mit einem rekursiven Schritt die sogenannte Vorwärtswahrscheinlichkeit $\alpha_\tau(i)$ bestimmt, die sich auf folgende Art und Weise berechnen lässt:

$$\alpha_1(i) = \pi_i b_i(o_1) \qquad 1 \le i \le N \tag{4.16}$$

$$\alpha_{\tau+1}(j) = \sum_{i=1}^{N} \alpha_\tau(i) a_{ij} b_j(o_{\tau+1}) \qquad \begin{array}{l} 1 \le \tau \le T-1, \\ 1 \le j \le N \end{array} \tag{4.17}$$

Generell zeigt sich, dass dieser Ansatz, eine gute Vorverarbeitung vorausgesetzt, sehr gute Ergebnisse liefert, obgleich es manchmal zur Verwechslung der Form zwischen Hand und Faust kommt, weshalb die HMMs modifiziert werden, um diese Fehlerquelle zu minimieren. Die resultierende Anpassung des HMM-Ansatzes wird in dem folgenden GM-Ansatz verwirklicht.

GM-Ansatz

Der GM-Ansatz ist dem HMM-Ansatz recht ähnlich, dennoch kommt es zu zwei wesentlichen Änderungen: Erstens werden die Merkmale für die Bewegung und Form getrennt behandelt, zweitens wird die Form nur im ersten Bild bestimmt. Anhand der zwei verborgenen Zustände m_τ und s_1 soll die Trennung der Bewegung und Form realisiert werden. Die ZV m_τ modelliert die Bewegung der Geste, s_1 liefert eine Beschreibung der Form. Die Form der Hand wird nur im sogenannten *Prolog* ($t = 1$) bestimmt, für die restlichen Bilder aus der Gestensequenz, dem sogenannten *Chunk*, wird nur die Bewegung ausgewertet, da durch die Bewegungsunschärfe die Bestimmung der Form erschwert wird. Des Weiteren gibt es jetzt nicht für jede Klasse ein eigenes HMM mit einem zugehörigen Parametersatz λ, sondern alle Informationen werden in einem einzigen GM erfasst. Die wahrscheinlichste Klasse $\hat{c}$ für die erfasste Beobachtungssequenz $\widehat{\mathcal{Z}}_T$ kann wie folgt aus der Verbundwahrscheinlichkeit des GM bestimmt werden:

$$\hat{c_T} = \arg\max_{\mathcal{C}_T} \sum_{\mathcal{M}_T, s_1} p(\mathcal{C}_T, \mathcal{M}_T, S_1, \widehat{\mathcal{Z}}_T) \tag{4.18}$$

mit $\widehat{\mathcal{Z}}_T = \{\widehat{\mathcal{X}}_T^s, \Delta\widehat{\mathcal{X}}_T^m\}$

$$= \arg\max_{\mathcal{C}_T} \sum_{\mathcal{M}_T, s_1} p(\mathcal{C}_T, \mathcal{M}_T, S_1, \widehat{\mathcal{X}}_T^s, \Delta\widehat{\mathcal{X}}_T^m) \tag{4.19}$$

$$= \arg\max_{\mathcal{C}_T} \sum_{\mathcal{M}_T, s_1} p(c_1)p(s_1|c_1)p(\widehat{\mathbf{x}}_1^s|s_1)p(m_1|c_1)p(\Delta\widehat{\mathbf{x}}_1^m|m_1) \prod_{\tau=2}^{T} p(c_\tau|c_{\tau-1})p(m_\tau|c_\tau, m_{\tau-1})p(\Delta\widehat{\mathbf{x}}_\tau^m|m_\tau) \tag{4.20}$$

Die Klasse c_τ mit $\tau \in [1, \ldots, T]$ einer Handgeste bleibt während einer ganzen Beobachtungssequenz $\widehat{\mathcal{Z}}_T$ konstant, somit wird für jeden Zeitschritt τ $c_\tau = c$ gesetzt. Aufgrund dessen gibt es für bedingte Wahrscheinlichkeit $p(c_\tau|c_{\tau-1})$ nur zwei Zustände: Entweder beide Klassen sind gleich, und somit wird die $p(c_\tau|c_{\tau-1}) = 1$, oder man erhält $p(c_\tau|c_{\tau-1}) = 0$. Die bedingte WF $p(c_\tau|c_{\tau-1})$ kann nun, wie folgt, vereinfacht werden:

$$p(c_\tau|c_{\tau-1}) = \delta(c_\tau, c_{\tau-1}), \tag{4.21}$$

mit dem Kronecker Delta

$$\delta(c_\tau, c_{\tau-1}) = \begin{cases} 1 & \text{falls } c_\tau = c_{\tau-1} \\ 0 & \text{sonst.} \end{cases} \tag{4.22}$$

Mit dieser Vereinfachung ergibt sich für Gleichung 4.20 folgende neue Form:

$$\hat{c} = \arg\max_{\mathcal{C}_T} \sum_{\mathcal{M}_T, s_1} p(c_1)p(s_1|c_1)p(\widehat{\mathbf{x}}_1^s|s_1)p(m_1|c_1)p(\Delta\widehat{\mathbf{x}}_1^m|m_1) \prod_{\tau=2}^{T} \delta(c_\tau|c_{\tau-1})p(m_\tau|c_\tau, m_{\tau-1})p(\Delta\widehat{\mathbf{x}}_\tau^m|m_\tau) \tag{4.23}$$

$$= \arg\max_{c} \sum_{\mathcal{M}_T, s_1} p(c)p(s_1|c)p(\widehat{\mathbf{x}}_1^s|s_1)p(m_1|c)p(\Delta\widehat{\mathbf{x}}_1^m|m_1) \prod_{\tau=2}^{T} p(m_\tau|c, m_{\tau-1})p(\Delta\widehat{\mathbf{x}}_\tau^m|m_\tau) \tag{4.24}$$

4.4.2 Experimente

4.4.2.1 Datensatz

Zum Testen des Ansatzes zur Erkennung von dynamischen Handgesten aus einem Videokamerabild wurde ein Datensatz von insgesamt zehn Personen aufgenommen, dabei wurden pro Person zwei Durchgänge aufgenommen. Insgesamt gab es sechs unterschiedliche Gesten, die sich jeweils auf die rechte Hand bezogen, hierbei wurde

entweder die Faust oder die Hand bewegt (links, rechts, oben, unten). Zur Evaluierung wurde eine fünffache Kreuzvalidierung angewendet, dabei wurden immer die Daten von acht Personen zum Training eingesetzt. Auf den verbleibenden Datensätzen der anderen beiden Personen wurde anschließend ein Test durchgeführt. Dieser Vorgang wurde fünfmal wiederholt.

4.4.2.2 Ergebnisse

Die Ergebnisse sowohl für den HMM-Ansatz als auch den GM-Ansatz sind in Tabelle 4.4 zu sehen. Generell liefern beide Ansätze sehr gute Ergebnisse, dies hängt aber im Wesentlichen davon ab, inwieweit die Vorverarbeitung – Detektion des Gesichtes, Erstellung des Hautfarbenmodells, Bestimmung der Hand – funktioniert hat.

	Klassifikation
HMM	94,17 %
GM	99,17 %

Tabelle 4.4: Übersicht über die erzielten Ergebnisse für den HMM-Ansatz und GM-Ansatz bei der Erkennung von dynamischen Handgesten

Die Verwendung des GM-Ansatzes brachte eine Erhöhung der Erkennungsrate auf 99,17 % von 94,17 %, welche mit dem HMM-Ansatz erreicht wurden. Basierend auf dem Signifikanztest, der im Abschnitt 3.2.2 vorgestellt worden ist, liefert der GM-Ansatz eine signifikante Verbesserung gegenüber dem HMM-Ansatz.

4.5 Kopfgestenerkennung

Der Ansatz zur Erkennung von Kopfgesten beruht, genauso wie der Ansatz zur Erkennung von dynamischen Handgesten, auf einem gewöhnlichen 2-D-Videokamerabild. Die Erkennung von Kopfgesten beschränkt sich auf drei verschiedene Klassen: Kopfnicken, Kopfschütteln sowie eine neutrale Kopfhaltung. Mit diesen drei Klassen soll auf nonverbale Art und Weise die Zustimmung bzw. Ablehnung in einer Dialogsituation erkannt werden. Eine Beispielsequenz für ein Kopfschütteln ist in Abbildung 4.12 dargestellt.

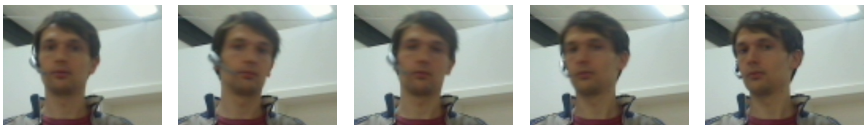


Abbildung 4.12: Beispielsequenz für Kopfgesten

4.5.1 Realisierung

Insgesamt besteht der Ansatz zur Kopfgestenerkennung aus fünf wesentlichen Schritten: Bilderfassung, Gesichtsdetektion, Anpassen eines Gesichtsmodells, Merkmalsextraktion und eine abschließende Klassifikation. Ein Übersicht ist in Abbildung 4.13 dargestellt.

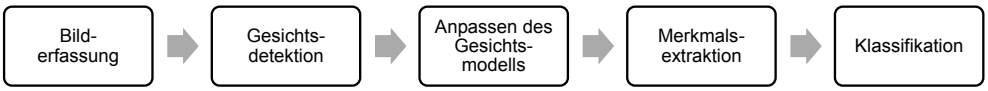


Abbildung 4.13: Ablaufdiagramm für die dynamische Kopfgestenerkennung im Videokamerabild

4.5.1.1 Gesichtsdetektion und Anpassen des Gesichtsmodells

Ausgangsgrundlage für die Klassifikation von Kopfgesten ist, wie im Ansatz für die dynamischen Handgesten (siehe Abschnitt 4.4), der Ansatz von Viola und Jones [151, 152] zur Gesichtsdetektion. Falls die Gesichtsdetektion mehrere Hypothesen zurückliefert, dann wird, genauso wie bei den dynamischen Handgesten, das größte Gesicht für die weitere Verarbeitung ausgewählt. Die gefundene Position des detektierten Gesichtes dient als Ausgangsgrundlage für eine Gesicht-Modellanpassung (engl. *face model fitting*), dabei wird ein CANDIDE-3-Modell [153] (siehe Abschnitt 5.2.2 für weitere Details) verwendet, um eine Abstraktion für das Gesicht zu erhalten (siehe Abbildung 4.14). Diese Abstraktion lässt sich nutzen, um die wesentlichen Merkmale eines Gesichtes mit einem einfachen Parametervektor $\mathbf{p}_t$ zu beschreiben.

4.5.1.2 Merkmalsextraktion

Der Parametervektor $\mathbf{p}_t$ umfasst auch Informationen, mit dem sich die Mimik des Gesichtes beschreiben lässt (siehe Abschnitt 5.2.2), diese Informationen sind aber für die Bestimmung von Kopfgesten nicht notwendig, daher wird aus dem Parametervektor $\mathbf{p}_t$ nur eine bestimmte Teilmenge θ_t verwendet. Dieser neue Vektor θ_t umfasst die Translationen (px_t, py_t, pz_t) sowie die Rotationen $(\alpha_t, \beta_t, \gamma_t)$ des Gesichtes, somit gilt: $\theta_t = (px_t, py_t, pz_t, \alpha_t, \beta_t, \gamma_t)^T$. Das Koordinatensystem, das den Zusammenhang zwischen den Translationen und Rotationen beschreibt, ist in Abbildung 4.14 dargestellt, zusätzlich findet sich eine Darstellung des CANDIDE-3-Modells [153] darin.

Da es sich bei den Kopfgesten um dynamische Bewegungen handelt, sind für die Klassifikation nur die zeitlichen Veränderungen zwischen zwei Vektoren θ_t, θ_{t-1} interessant, daher werden nur die Differenzen betrachtet:

$$\Delta\theta_\tau = \theta_\tau - \theta_{\tau-1} \quad \forall\, \tau \subseteq \{1, \ldots, t, \ldots, T\}, \quad \theta_{t=0} = \mathbf{0}. \tag{4.25}$$

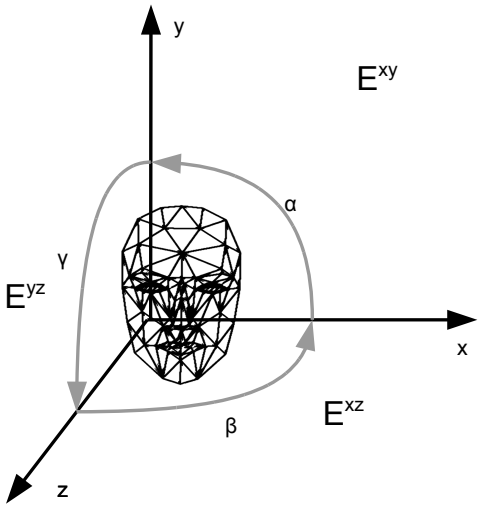


Abbildung 4.14: CANDIDE-3-Modell [153] im dazugehörigen Koordinatensystem

Diese Differenzbildung kann wieder, wie beim Ansatz für die dynamischen Handgesten, einfach mit der RTDB-Architektur realisiert werden.

4.5.1.3 Klassifikation

Die Klassifikation beruht, ähnlich wie bei dem Ansatz zur Klassifikation der dynamischen Handgesten, sowohl auf einem HMM-Ansatz als auch auf einem GM-Ansatz.

HMM-Ansatz
Der HMM-Ansatz bildet wieder den Referenzansatz, hierbei werden alle Merkmale $\Delta\widehat{\Theta_T} = \{\Delta\widehat{\theta}_1, \ldots, \Delta\widehat{\theta}_T\}$ zur Klassifikation herangezogen. Wenn man $\mathbf{z}_\tau$ von Abbildung 4.11(a) durch $\Delta\theta_\tau$ ersetzt, erhält man die HMM-Darstellung für die Kopfgesten (daher wird auf eine Abbildung verzichtet). Die Berechnung der wahrscheinlichsten Klasse $\hat{c}$ der drei Kopfgesten erfolgt wieder anhand der Bestimmung der Produktionswahrscheinlichkeit für die drei unterschiedlichen HMMs. Diese HMMs werden erneut durch ihren jeweiligen, mittels *Baum-Welch*-Algorithmus [85, 84] bestimmten HMM-Parametersatz ${}^k\lambda$ beschrieben. Die Produktionswahrscheinlichkeit für ein HMM mit dem Parametersatz ${}^k\lambda$ kann wieder anhand des Vorwärtsalgorithmus [84] folgendermaßen für die Beobachtungssequenz $\Delta\widehat{\Theta}_T = \{\Delta\widehat{\theta}_1, \ldots, \Delta\widehat{\theta}_T\}$ bestimmt werden:

$$p(\Delta\widehat{\Theta}_T|{}^k\lambda) = \sum_{i=1}^{N} \alpha_T(i) \tag{4.26}$$

Die rekursive Bestimmung der Vorwärtswahrscheinlichkeit $\alpha_\tau(i)$ erfolgt wieder mit Gleichung 4.16 sowie Gleichung 4.17.

Für die erfasste Beobachtungssequenz $\Delta\widehat{\Theta}_t = \{\Delta\widehat{\theta}_1, \ldots, \Delta\widehat{\theta}_T\}$ kann nun die wahrscheinlichste Klasse $\hat{c} \triangleq \hat{\lambda}$ wie folgt bestimmt werden:

$$\hat{\lambda} = \arg\max_k p(\Delta\widehat{\Theta}_T|^k\lambda). \tag{4.27}$$

GM-Ansatz

Beim GM-Ansatz werden diesmal auch alle Merkmale über die gesamte Sequenzlänge T genutzt, aber es gibt neben dem Klassenzustand noch drei weitere verborgene Zustände, die die Bewegungen für die unterschiedlichen Bewegungsebenen E^{xz}, E^{yz}, E^{xy} (siehe Abbildung 4.14) modellieren. Somit besteht das GM für die Kopfgestenerkennung insgesamt aus vier diskreten verborgenen Knoten (siehe Abbildung 4.15) und sechs Beobachtungen Δpx_τ,Δpy_τ, Δpz_τ, $\Delta\alpha_\tau$[1], $\Delta\beta_\tau$, $\Delta\gamma_\tau$.

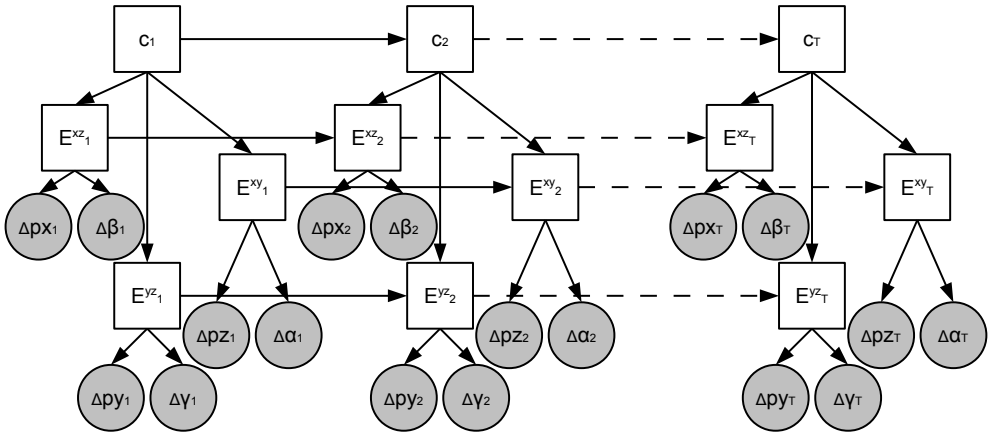


Abbildung 4.15: GM für die Erkennung von Kopfgesten

Die Motivation hinter dieser Gruppierung der ZVs speist sich aus dem Umstand, dass die Beobachtungen, die für gleiche Bewegungen zutreffen, in einem verborgenen Zustand zusammengefasst werden, daher ist immer eine Translationsänderung mit einer Rotationsänderung kombiniert worden. Beispielsweise ändert sich beim Kopfschütteln primär der Rotationswinkel β und die Position des Kopfes in x-Richtung. Ähnlich wie beim GM-Ansatz für die dynamischen Handgesten bleibt die Klasse c_τ mit $\tau \in [1, \ldots, T]$ für eine Kopfgeste während einer ganzen Beobachtungssequenz $\Delta\widehat{\Theta}_T$ konstant, somit wird wiederum für jeden Zeitschritt τ $c_\tau = c$ gesetzt. Des Weiteren gilt der Zusammenhang von Gleichung 4.22 (Kronecker Delta, $\delta(c_\tau, c_{\tau-1})$).

[1] α_τ beschreibt die Rotation um die z-Achse. Für die verwendeten Kopfgesten gibt es in der Regel keine Änderungen für diesen Winkel. Da die Winkeländerung $\Delta\alpha_\tau$ weder einen positiven noch einen negativen Einfluss auf die Klassifikation ausübte, wurde dieses Merkmal der Vollständigkeit halber im GM berücksichtigt. Auch das Merkmal Δpz_τ nahm auf die verwendeten Kopfgesten nur gering Einfluss, dennoch ist es für die Klassifikation der Vollständigkeit halber verwendet worden.

Diese Vereinfachung wird bei der Bestimmung der wahrscheinlichsten Klasse $\hat{c}$ für die erfasste Beobachtungssequenz $\Delta\widehat{\Theta}_T = \{\Delta\widehat{\mathcal{PX}}_T, \Delta\widehat{\mathcal{PY}}_T, \Delta\widehat{\mathcal{PZ}}_T, \Delta\widehat{\mathcal{A}}_T, \Delta\widehat{\mathcal{B}}_T, \Delta\widehat{\Gamma}_T\}$ aus der Verbundwahrscheinlichkeit des GM berücksichtigt. Somit ergibt sich für die Bestimmung der wahrscheinlichsten Klasse $\hat{c}$ folgender Zusammenhang:

$$\hat{c_T} = \arg\max_{\mathcal{C}_T} \sum_{\mathcal{E}_T^{xz}, \mathcal{E}_T^{xy}, \mathcal{E}_T^{yz}} p(\mathcal{C}_T, \mathcal{E}_T^{xz}, \mathcal{E}_T^{xy}, \mathcal{E}_T^{yz}, \Delta\widehat{\mathcal{PX}}_T, \Delta\widehat{\mathcal{PY}}_T, \Delta\widehat{\mathcal{PZ}}_T, \Delta\widehat{\mathcal{A}}_T, \Delta\widehat{\mathcal{B}}_T, \Delta\widehat{\Gamma}_T) \tag{4.28}$$

$$\begin{aligned} = \arg\max_{\mathcal{C}_T} \sum_{\mathcal{E}_T^{xz}, \mathcal{E}_T^{xy}, \mathcal{E}_T^{yz}} & p(c_1) p(E_1^{xz}|c_1) p(E_1^{xy}|c_1) p(E_1^{yz}|c_1) \\ & p(\Delta\widehat{px}_1, \Delta\widehat{\beta}_1|E_1^{xz}) p(\Delta\widehat{pz}_1, \Delta\widehat{\alpha}_1|E_1^{xy}) p(\Delta\widehat{py}_1, \Delta\widehat{\gamma}_1|E_1^{yz}) \\ & \prod_{\tau=2}^{T} p(c_\tau, c_{\tau-1}) p(E_\tau^{xz}|c_\tau, E_{\tau-1}^{xz}) p(E_\tau^{xy}|c_\tau, E_{\tau-1}^{xy}) p(E_\tau^{yz}|c_\tau, E_{\tau-1}^{yz}) \\ & \quad p(\Delta\widehat{px}_\tau, \Delta\widehat{\beta}_\tau|E_\tau^{xz}) p(\Delta\widehat{pz}_\tau, \Delta\widehat{\alpha}_\tau|E_\tau^{xy}) p(\Delta\widehat{py}_\tau, \Delta\widehat{\gamma}_\tau|E_\tau^{yz}) \end{aligned} \tag{4.29}$$

Bei Berücksichtigung der Vereinfachung mit dem Kronecker Delta (siehe Gleichung 4.22) ergibt sich:

$$\begin{aligned} \hat{c} = \arg\max_{\mathcal{C}_T} \sum_{\mathcal{E}_T^{xz}, \mathcal{E}_T^{xy}, \mathcal{E}_T^{yz}} & p(c_1) p(E_1^{xz}|c_1) p(E_1^{xy}|c_1) p(E_1^{yz}|c_1) \\ & p(\Delta\widehat{px}_1, \Delta\widehat{\beta}_1|E_1^{xz}) p(\Delta\widehat{pz}_1, \Delta\widehat{\alpha}_1|E_1^{xy}) p(\Delta\widehat{py}_1, \Delta\widehat{\gamma}_1|E_1^{yz}) \\ & \prod_{\tau=2}^{T} \delta(c_\tau, c_{\tau-1}) p(E_\tau^{xz}|c_\tau, E_{\tau-1}^{xz}) p(E_\tau^{xy}|c_\tau, E_{\tau-1}^{xy}) p(E_\tau^{yz}|c_\tau, E_{\tau-1}^{yz}) \\ & \quad p(\Delta\widehat{px}_\tau, \Delta\widehat{\beta}_\tau|E_\tau^{xz}) p(\Delta\widehat{pz}_\tau, \Delta\widehat{\alpha}_\tau|E_\tau^{xy}) p(\Delta\widehat{py}_\tau, \Delta\widehat{\gamma}_\tau|E_\tau^{yz}) \end{aligned} \tag{4.30}$$

$$\begin{aligned} = \arg\max_{c} \sum_{\mathcal{E}_T^{xz}, \mathcal{E}_T^{xy}, \mathcal{E}_T^{yz}} & p(c) p(E_1^{xz}|c) p(E_1^{xy}|c) p(E_1^{yz}|c) \\ & p(\Delta\widehat{px}_1, \Delta\widehat{\beta}_1|E_1^{xz}) p(\Delta\widehat{pz}_1, \Delta\widehat{\alpha}_1|E_1^{xy}) p(\Delta\widehat{py}_1, \Delta\widehat{\gamma}_1|E_1^{yz}) \\ & \prod_{\tau=2}^{T} p(E_\tau^{xz}|c, E_{\tau-1}^{xz}) p(E_\tau^{xy}|c, E_{\tau-1}^{xy}) p(E_\tau^{yz}|c, E_{\tau-1}^{yz}) \\ & \quad p(\Delta\widehat{px}_\tau, \Delta\widehat{\beta}_\tau|E_\tau^{xz}) p(\Delta\widehat{pz}_\tau, \Delta\widehat{\alpha}_\tau|E_\tau^{xy}) p(\Delta\widehat{py}_\tau, \Delta\widehat{\gamma}_\tau|E_\tau^{yz}) \end{aligned} \tag{4.31}$$

4.5.2 Experimente

4.5.2.1 Datensatz

Zum Testen des Ansatzes zur Erkennung von Kopfgesten aus einem Videokamerabild wurde ein Datensatz von insgesamt zehn Personen aufgenommen, pro Person wurden zwei Durchgänge aufgenommen. Insgesamt gab es drei unterschiedliche Gesten. Bei der Aufnahme der Gesten ist darauf geachtet worden, dass die Darstellung der Gesten einer natürlichen Dialogsituation entsprechen und die Gesten nicht zu übertrieben dargestellt werden. Zur Evaluierung wurde eine Fünffach-Kreuzvalidierung angewendet. Hierbei wurden immer die Daten von acht Personen zum Training eingesetzt, auf den verbleibenden zwei Datensätzen wurde anschließend ein Test durchgeführt. Der Vorgang wurde fünfmal wiederholt.

4.5.2.2 Ergebnisse

Die Ergebnisse sowohl für den HMM-Ansatz als auch den GM-Ansatz sind in Tabelle 4.5 zu sehen. Generell liefern beide Ansätze gute Ergebnisse, dennoch spielt, wie schon beim Ansatz für dynamische Handgesten, eine robuste Vorverarbeitung eine entscheidende Rolle.

	Klassifikation
HMM	81,67 %
GM	90,00 %

Tabelle 4.5: Übersicht über die erzielten Ergebnisse für den HMM-Ansatz und GM-Ansatz bei der Erkennung von Kopfgesten

Die Verwendung des GM-Ansatzes brachte eine Erhöhung der Erkennungsrate auf 90,00 % von 81,67 % für den HMM-Ansatz. Basierend auf dem Signifikanztest, der im Abschnitt 3.2.2 vorgestellt worden ist, liefert der GM-Ansatz eine signifikante Verbesserung gegenüber dem HMM-Ansatz.

4.6 Diskussion

In diesem Kapitel wurden drei Ansätze zur Gestenerkennung vorgestellt. Alle Ansätze waren aufgrund der Einbindung in eine einheitliche Architektur in der Lage, echtzeitfähig für ein Dialogsystem Klassifikationsergebnisse zu liefern. Die Gestenerkennungssysteme kamen auf zwei unterschiedlichen Roboterplattformen zum Einsatz und bildeten neben Sprache eine zusätzliche Modalität zur Interaktion.

Die RTDB hat sich beim Problem des *gesture spotting* als hilfreich erwiesen und kann aufgrund ihres Konzepts der *Writer* und *Reader* helfen, eine modulare Umset-

zung der Gestenerkennung zu ermöglichen. In dieser Architektur ist es möglich unterschiedliche Gesten zu bearbeiten, deswegen hilft die RTDB bei der Gestaltung einer multimodalen und somit natürlichen und intuitiven Interaktion zwischen Mensch und Maschine. Darüber hinaus bietet diese Architektur die Möglichkeit, neben einer Fusion der Modalitäten auf Entscheidungsebene, wie es in den verwendeten Dialogsystemen zum Einsatz gekommen ist, eine Fusion der unterschiedlichen Modalitäten auf Daten- bzw. Merkmalsebene durchzuführen.

Ein anderer Punkt, der sich bei der Gestenverarbeitung gezeigt hat, ist, dass die Vorverarbeitung einen entscheidenden Einfluss für die spätere Klassifikation hat. Eine gute Segmentierung der Hand bzw. des Kopfes war eine Notwendigkeit für das Erzielen von guten Klassifikationsergebnissen.

Für die Klassifikation hat sich wiederum gezeigt, dass sich die Klassifikationsergebnisse für die Hand- und Kopfgesten durch den Einsatz von komplexeren Klassifikatoren (SVM, GM) verbessern ließ, auch wenn diese Maßnahmen nicht den gleichen Einfluss wie die Vorverarbeitung bei der Gestenerkennung ausüben.

Abschließend lässt sich feststellen, dass das Feld der Gestenerkennung ein breites Spektrum an Aufgaben bietet, an denen auch in Zukunft noch ausgiebig geforscht werden wird, um mit der Erkennung von Gesten einen entscheidenden Beitrag für die multimodale Interaktion zwischen Mensch und Maschine zu liefern.

Kapitel 5

Mimikerkennung

Inhaltsangabe

In diesem Kapitel werden zwei GM-Verfahren zur Mimikerkennung vorgestellt. Mit diesen Verfahren können Mimiksequenzen, die einen der sechs universellen Emotionsgesichtsausdrücke (Freude, Traurigkeit, Wut, Ekel, Angst, Überraschung) darstellen, klassifiziert werden. Zur Klassifikation werden drei unterschiedliche Merkmalskonfigurationen verwendet, die Merkmale werden anhand eines Gesichtsmodells gewonnen. Eine Mimiksequenz besteht aus verschiedenen zeitlichen Phasen, woraus sich ergibt, dass diese Phasen einen unterschiedlichen Beitrag zur Erkennung der Mimikemotionen liefern. Infolgedessen wird eine Merkmalsselektion als Vorverarbeitungsschritt eingeführt. Zunächst wird ein Standard-Verfahren, die Sequentielle Merkmalsselektion

(Sequential Feature Selection), verwendet, anschließend werden zwei Verfahren beschrieben, die die Merkmale mithilfe der Kullback-Leibler-Divergenz auswählen.

5.1 Einführung

5.1.1 Motivation

Zur Kommunikation stehen Menschen unterschiedliche Modalitäten zur Verfügung (siehe Abschnitt 4.1.1). Die gestenbasierte Interaktion stand im Fokus von Kapitel 4, der Ausdruck von Emotionen mithilfe von Mimik ist Gegenstand dieses Kapitels. Eine besondere Eigenschaft der mimikbasierten Emotionserkennung liegt darin, dass sich, im Gegensatz zu den generell nicht eindeutigen Gesten [114], sechs universelle Gesichtsausdrücke finden lassen [156]. Diese Gesichtsausdrücke beschreiben unabhängig von Alter, Geschlecht und Kulturkreis die emotionalen Zustände: Freude (engl. *happiness*), Traurigkeit (engl. *sadness*), Wut (engl. *anger*), Ekel (engl. *disgust*), Angst (engl. *fear*), Überraschung (engl. *surprise*) [156]. Die sechs universellen Emotionsgesichtsausdrücke sind in Abbildung 5.1 dargestellt, die gezeigten Bilder stammen aus der MMI-Datenbank [157, 158].

Der Ausdruck von Emotionen mittels Mimik ist nach [159] bei älteren Menschen im gleichen Umfang möglich wie bei jüngeren Menschen, falls bestimmte Emotionen (z. B. Erlebnisse aus der Vergangenheit) wieder durchlebt werden sollen. Unterschiede zwischen älteren und jüngeren Menschen lassen sich bei [159] finden, wenn die Menschen instruiert werden, bestimmte Mimikausdrücke zu bilden, wobei die Gesichtsausdrücke der älteren Menschen schwächer ausgeprägt sind. Generell spielt die Mimik insbesondere dann eine wichtige Rolle, wenn Gefühle und Einstellungen kommuniziert werden sollen, die Mimik trägt laut [160] mehr als die Hälfte zur Kommunikation bei. Im Bereich von AAL wird Mimik in [161] eingesetzt, um anhand von fotorealistischen Animationen mittels eines sogenannten *Ambient Facial Interface* dem AAL-Nutzer Rückmeldungen geben zu können. Dieses System ist in [162] auf eine Roboterplattform integriert worden, um den AAL-Nutzer bei der Einhaltung der Medikamenteneinnahme zu unterstützen. Neben AAL gibt es weitere Anwendungen für Mimik in der MMI. In [163] wird ein System vorgestellt, das die erkannte Mimik direkt spiegeln kann. Dieses System kann seine Mimikspiegelung auf den Roboterplattformen Aibo [52] und Robovie [164] wiedergeben. Eine ähnliche Vorgehensweise zur spiegelnden Darstellung von Mimik auf einer Roboterplattform findet sich in [165].

In diesem Kapitel werden zwei GM-Ansätze zur mimikbasierten Emotionserkennung vorgestellt, des Weiteren wird eine Merkmalsselektion betrachtet. Hierbei kommen sequentielle Verfahren zum Einsatz, anschließend werden zwei Verfahren beschrieben, die die Merkmale mithilfe der Kullback-Leibler-Divergenz auswählen.

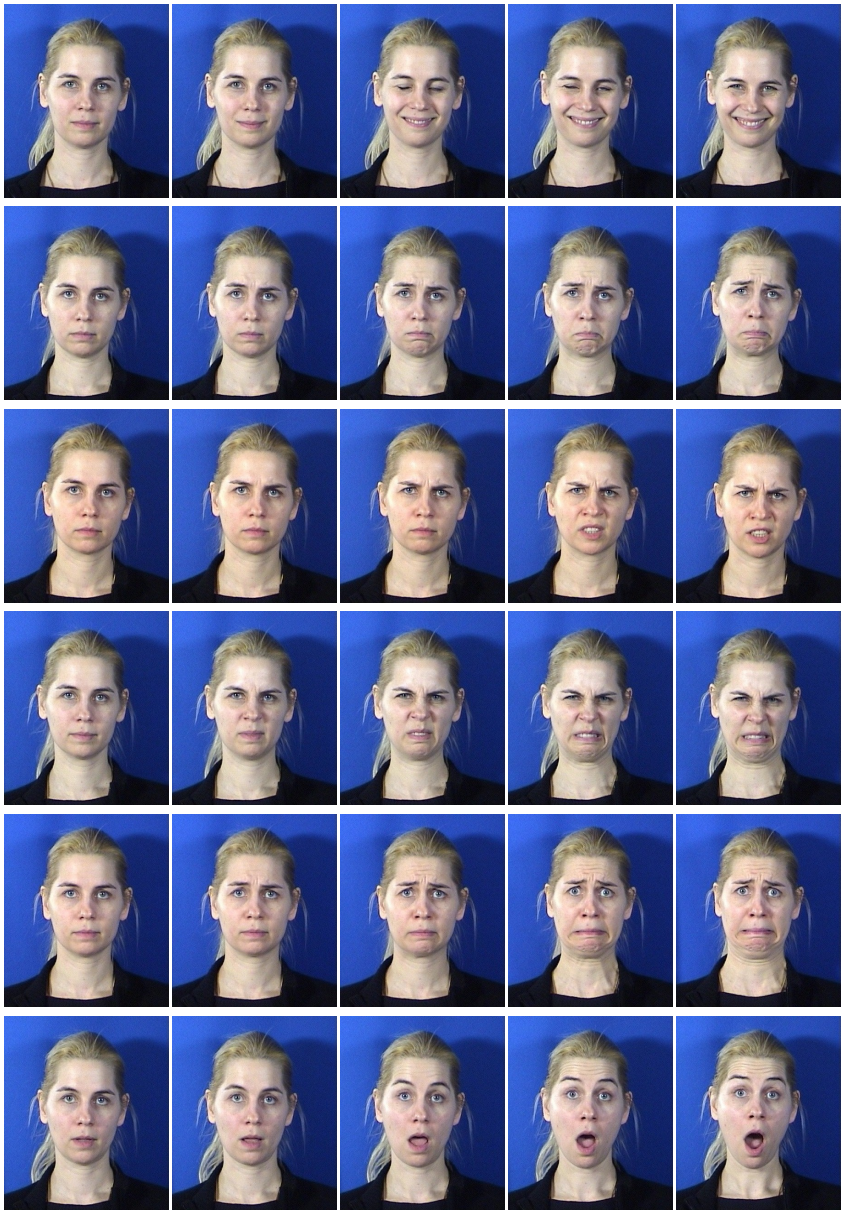


Abbildung 5.1: Überblick über die sechs universellen Emotionsgesichtsausdrücke. Die Bildsequenzen stammen aus der MMI-Datenbank [157, 158] und zeigen die sechs emotionalen Zustände: Freude, Traurigkeit, Wut, Ekel, Angst und Überraschung.

5.1.2 Stand der Technik

Ekman lieferte mit seinen Arbeiten aus den 1970er Jahren einen wesentlichen Beitrag zur Mimikanalyse, in [156] wurden sechs universelle Gesichtsausdrücke zur Emotionsdarstellung identifiziert. Zur Unterteilung wurde von Ekman und Friesen das so-

genannte Gesichtsbewegungs-Kodierungssystem (engl. *Facial Action Coding System*, FACS) entwickelt [166], um unterschiedliche Gesichtsausdrücke beschreiben (bzw. codieren) zu können, eine Aktualisierung dieses Systems wurde im Jahre 2002 durchgeführt [167]. Diese Arbeiten haben einen wesentlichen Einfluss auf die Erstellung eines Mimikerkennungssystems, dessen prinzipieller Aufbau (ähnlich wie der Aufbau eines Gestenerkennungssystems von Abschnitt 4.1.2) in Abbildung 5.2 dargestellt ist. Ähnlich wie bei dem System zur Gestenerkennung können für ein Mimikerkennungssystem folgende wesentliche Komponenten bestimmt werden: 1) die Mimik, die erkannt werden soll, 2) ein Sensor, der verwendet wird, um die Mimik zu erkennen, 3) eine Gesichtsdetektion, die einen zentralen Vorverarbeitungsschritt darstellt, 4) eine Merkmalsextraktion, die Merkmale bestimmt, anhand derer sich eine Mimik erkennen lässt und 5) eine Klassifikation, die die Mimik einer bestimmten Mimik aus einem spezifizierten Alphabet zuordnet. Die Weiterleitung des Klassifikationsergebnisses an eine höhere Verarbeitungsebene (z. B. Dialogsystem) stellt einen weiteren möglichen Schritt dar.

Der Aufbau des Mimikerkennungssystems (siehe Abbildung 5.2) ist angelehnt an [168], wo drei wesentliche Aufgaben für die automatische Mimikanalyse beschrieben werden: Gesichtsdetektion (engl. *face detection*), Mimikdatenextraktion (engl. *facial expression data extraction*) und Mimikklassifikation (engl. *facial expression classification*).

Abbildung 5.2: Prinzipieller Aufbau eines Mimikerkennungssystems

Neben den sechs universellen Gesichtsausdrücken zur Emotionsdarstellung [156], die unabhängig von Alter, Geschlecht und Kulturkreis die emotionalen Zustände – Freude, Traurigkeit, Wut, Ekel, Angst, Überraschung – abbilden, kann das menschliche Gesicht laut [169, 170] 136 emotionale Zustände mittels Mimik ausdrücken.

Der zeitliche Verlauf einer Mimiksequenz besteht im Wesentlichen aus drei unterschiedlichen Phasen [171], wobei man zunächst von einer neutralen Mimik ausgeht. Die erste Phase, die sogenannte Aktivierung (engl. *onset*), beschreibt die Zunahme der Ausdrucksstärke der Mimik bis zur zweiten Phase, dem sogenannten Halten (engl. *apex*), während die Abnahme der Ausdrucksstärke der Mimik, das sogenannte Entspannen (engl. *offset*), die dritte Phase umfasst. Ein erster Überblick über Mimikerkennung und die notwendigen Schritte, die vorher zu lösen sind, findet sich in [172]. Die Arbeiten aus den 1990er Jahren auf diesem Gebiet sind in [168, 173] gut beschrieben, ein aktueller Überblick findet sich in [170], weitere Beschreibungen zur Mimikerkennung finden sich in [174, 171]. Daneben gibt es noch weitere Recherchen, die unterschiedliche Schwerpunkte setzen, aber auch Überschneidungen mit der Mimi-

kerkennung aufweisen. Beispielsweise wird in [175] ein Überblick über multimodale Affekterkennung (engl. *affect recognition*) gegeben, auch Mimik spielt in diesem Zusammenhang eine wichtige Rolle, um die Gemütserregung eines Menschen zu erkennen. In [176] werden Aspekte des menschlichen Verhaltens beschrieben, die relevant sind, um eine intuitive MMI zu realisieren, hierbei ist unter anderem Mimik bedeutend, um das Verhalten des Menschen besser zu verstehen.

Die Systeme, die zur Erkennung von Mimik auf dem Prinzip des FACS beruhen, können für zwei unterschiedliche Arten der Mimikanalyse verwendet werden: erstens die Erkennung von Aktivitäten einzelner Bewegungseinheiten (engl. *action units*) von unterschiedlichen Gesichtspartien (engl. *Facial Action Recognition*), beispielsweise das Heben einer Augenbraue, und zweitens Die Erkennung von ganzen Mimiksequenzen (engl. *Facial Expression Recognition*), die eine bestimmte semantische Bedeutung besitzen, beispielsweise die Darstellung von Freude mittels Mimik.[1] Nachfolgend werden nur Systeme betrachtet, die sich mit der Analyse einer ganzen Mimiksequenz beschäftigen – eine Aufstellung von Arbeiten, die sich mit der Erkennung von Aktivitäten einzelner Bewegungseinheiten auseinandersetzen, findet sich beispielsweise in [177]. Die Ansätze zur Analyse von Mimiksequenzen können anhand von drei wesentlichen Merkmalen unterschieden werden: erstens die Datenbank, die verwendet wird, um ein Mimikerkennungssystem zu trainieren sowie zu testen, zweitens die Merkmale, die verwendet werden, um wichtige Informationen aus den Mimiksequenzen zu extrahieren, und drittens das verwendete Klassifikationsverfahren, das benutzt wird, um verschiedene Gesichtsausdrücke zu unterscheiden. Hierbei ist auch zu beachten, ob nur ein Bild einer Mimiksequenz verwendet wird oder ob die ganze Sequenz zur Klassifikation herangezogen wird. In [170] findet sich sowohl eine Vorstellung der aktuellen Datenbanken zur Mimikerkennung als auch eine Beschreibung von 19 ausgewählten Arbeiten auf dem Gebiet der Mimikerkennung. In Bezug auf die ausführliche Diskussion der verschiedenen Ansätze zur Mimikerkennung sei auf die bereits oben erwähnten Arbeiten [172, 168, 173, 170, 177] verwiesen. Im Folgenden wird die Aufmerksamkeit auf Arbeiten gelenkt, die Überschneidungen hinsichtlich Datenbank, Merkmalen sowie – insbesondere – Klassifikation mit den vorgestellten GM-Ansätzen aufweisen. Hierbei wird ein besonderes Augenmerk auf Verfahren gelegt, die HMMs verwenden, da diese eine einfache Form von DBNs darstellen.

Eine erste Arbeit, die HMMs im Bereich von Mimikerekennung verwendet hat, ist [178], zur Merkmalsextraktion ist die Wavelet-Transformation verwendet worden, und diskrete HMMs haben zur Klassifikation gedient. Weitere anfängliche Arbeiten, die auf HMMs zur Klassifikation von Mimiksequenzen beruhen, sind die Arbeiten [179, 180, 181]. In [179] werden Links-Rechts-HMMs mit kontinuierlichen Beobach-

[1] Die Trennung der Begrifflichkeiten *Facial Action Recognition* und *Facial Expression Recognition* erfolgt nicht strikt. Es finden sich auch Arbeiten, die die Analyse unterschiedlicher Gesichtspartien als *Facial Expression Recognition* bezeichnen.

tungen verwendet, die Merkmale werden von der Bildregion des rechten Auges und der Bildregion des Mundes anhand einer Wavelet-Transformation gewonnen. Für die Evaluierung wurden Mimiksequenzen von drei Personen aufgezeichnet, sie konnten in einer personenspezifischen Evaluierung eine Erkennungsrate von 98 % erreichen. Für personenunabhängige Erkennung fiel die Erkennungsrate auf 84 %, wobei aber nicht mehr sechs, sondern nur vier universelle Gesichtsausdrücke zur Emotionsdarstellung verwendet worden waren. In [180] ist das bestehende System aus [179] erweitert worden, um für den Einsatz für unterschiedliche Personen besser geeignet zu sein. Des Weiteren ist die Merkmalsextraktion verändert worden, diese basiert nun auf einem, durch Optischen Fluss ermittelnden, Geschwindigkeitsvektor zwischen zwei aufeinanderfolgenden Kamerabildern. Anhand einer Fourier-Transformation werden niederfrequente Anteile aus dem Geschwindigkeitsvektor extrahiert. Die Klassifikation beruht weiterhin auf Links-Rechts-HMMs. HMMs werden in [181] verwendet, um die Bewegungen im Ablauf der Mimik (entspannt, anspannend, halten, entspannend, entspannt) zu unterscheiden. In [182] ist ein System entwickelt worden, dass unterschiedliche Aktivitäten von einzelnen Bewegungseinheiten des FACS mithilfe von HMMs erkennen kann, dabei werden aber nur Aktivitäten der oberen Gesichtshälfte untersucht. In den Arbeiten [183, 184] werden vier Gesichtsausdrücke zur Emotionsdarstellung mittels HMMs klassifiziert, die Merkmale werden von invarianten Momenten (ähnlich dem Ansatz von Abschnitt 4.4) gewonnen. Die Merkmale werden in sieben rechteckige Regionen extrahiert: linkes/ rechtes Auge, linke/ rechte Augenbraue, obere/ untere Mundpartie und der Region zwischen den Augen. In [185] sind die sechs Sequenzen der universellen Mimikemotionen mit einem zusätzlichen sogenannten Unterhaltungsmodell (engl. *talk*) erweitert worden, um zwischen Gesprächssituationen und dem Ausdruck einer der sechs Mimikemotionen zu unterscheiden. Die Merkmale basieren hierbei auf den Gesichtsanimationsparametern (engl. *facial animation parameters*, FAPs) (siehe Abschnitt 5.2.2).

Bis jetzt sind nur einfache HMM-Topologien verwendet worden, eine Erweiterung der HMM-Struktur findet sich in [186]. In dieser Arbeit wird ein sogenanntes *Multi-Level-HMM* zur Klassifikation der Mimik eingesetzt. Diese neue HMM-Struktur kombiniert die Segmentierung und die Klassifikation der Mimik und erhöht dabei auch die Erkennungsrate. Bei personenabhängigen Klassifikationsexperimenten steigert sich die Erkennungsrate von 78,49 % auf 82,46 %, bei personenunabhängigen Klassifikationsexperimenten steigt die Erkennungsrate von 55 % auf 58 %. Für die Experimente sind die Sequenzen der sechs universellen Mimikemotionen von fünf Personen verwendet worden, diese Daten stammen aus der Cohn-Kanade-Datenbank [187, 188] (siehe Abschnitt 5.2.1). Ein anderer Ansatz zur Erweiterung der HMM-Struktur findet sich in [189], hier werden sogenannte Pseudo-3-D-HMMs verwendet, um vier Mimiksequenzen von sechs Personen zu erkennen. Auch in [190] ist die HMM-Struktur erweitert worden, hier wird ein *Multi-Stream-HMM* verwendet, um die sechs universellen Mimikemotionen aus der Cohn-Kanade-Datenbank zu klassifizieren, als Merk-

male haben die FAPs gedient. In dieser HMM-Struktur gibt es zwei unterschiedliche *Streams*, wobei einer Informationen der Augenbrauen-FAPs enthält, während der zweite die Informationen der äußeren Lippen-FAPs benutzt. Die besten Ergebnisse (93,66 %) werden erzielt, wenn die *Streams* anhand der erzielten Einzelklassifikationsleistung gewichtet werden.

Daneben finden sich auch neuere Arbeiten, die HMMs zur Klassifikation von Mimikemotionen benutzen. In [191] werden unterschiedliche Merkmalsvorverarbeitungsmethoden (*principal component analysis (PCA)*, *independent component analysis (ICA)*, *enhanced ICA (EICA)* etc.) verwendet, um die HMM-basierte Klassifikation der sechs universellen Mimikemotionen aus der Cohn-Kanade-Datenbank zu verbessern. Die besten Ergebnisse (92,85 %) lieferte das in der Arbeit vorgestellte Verfahren zur Merkmalsselektion, die sogenannte *fisher independent component analysis* (hierbei wird *ICA* durch *Fishersche Diskriminanzfunktion* erweitert). Ähnlich zu [191] werden auch in [192] verschiedene Verabeitungsmethoden verwendet, um die HMM-basierte Klassifikation der sechs universellen Mimikemotionen aus der Cohn-Kanade-Datenbank zu verbessern, hierbei zielt der Einsatz der unterschiedlichen Verarbeitungsmethoden auf die Merkmale selbst ab. Insgesamt werden drei unterschiedliche Merkmale (*PCA*, *orientation histograms*, *optical flow estimation*) verwendet, um in vier Gesichtsregionen Merkmale zu extrahieren. Die besten Ergebnisse (86,11 %) wurden mit einem Fusionierungsansatz erzielt. Ergodische HMMs werden in [193] eingesetzt, um nicht-frontale Mimikemotionen klassifizieren zu können. Ein anderer neuer Ansatz, der auch auf der Mimikanalyse beruht, wird in [194] vorgestellt, hierbei sollen anhand der Mimik unterschiedliche Personen mittels HMM-Klassifikation identifiziert werden.

Es zeigt sich, dass es viele unterschiedliche Arbeiten für die Mimikanalyse gibt. Diese Arbeiten können anhand von verwendeten Datenbanken, extrahierten Merkmalen und Klassifikatoren unterschieden werden. Die hier präsentierten Ansätze werden mit der Arbeit von Mayer [177] verglichen, da es sowohl Überschneidungen mit den extrahierten Merkmalen als auch den verwendeten Datenbanken gibt.

5.2 GM-Ansätze

Im Folgenden werden zwei unterschiedliche GM-Ansätze vorgestellt, die dynamische Sequenzen der sechs universellen Mimikemotionen klassifizieren können. Bevor die Ansätze zur Klassifikation detaillierter beschrieben werden, werden aber noch relevante Sachverhalte wie Datensätze und Merkmale erörtert.

5.2.1 Datensätze

Es gibt unterschiedliche Datensätze, die für die Mimikanalyse zur Verfügung stehen. Trotz der gleichen Zielstellung, der Bereitstellung eines Datensatzes für die Mimikanalyse, unterscheiden sich die Datensätze, da verschiedene Fragestellungen bzw. Schwerpunkte bei der Aufzeichnung gesetzt worden sind. Eine Beschreibung unterschiedlicher Datensätze – *Cohn-Kanade database* [187, 188], *MMI database* [157, 158], *authentic facial expression database* [195, 196], *AR Face Database* [197], *CMU Pose, Illumination, and Expression Database* [198, 199], *Japanese Female Facial Expression (JAFFE) Database* [200] – findet sich in [170]. Daneben gibt es weitere Datensätze – *FGNet Facial Emotions and Expressions Database* [201], *Facial Expressions Database of the MPI for Biological Cybernetics* [202], *Belfast Naturalistic Emotional Database* [203] – bzw. es sind für die Arbeiten eigene Datensätze erstellt worden (siehe z. B. [179, 189]).

Für die Evaluierung der GM-Ansätze sowie der Merkmalsselektion ist die Cohn-Kanade-Datenbank [187, 188] verwendet worden, einige Beispielbilder aus dieser Datenbank finden sich in Abbildung 5.3. Diese Datenbank umfasst insgesamt 2015 Videosequenzen, in denen unterschiedliche Mimikaktivitäten, darunter auch die sechs universellen Mimikemotionen, von 182 Personen beiderlei Geschlechts frontal erfasst wurden. Das Emotionale Gesichtsbewegungs-Kodierungssystem (engl. *Emotional Facial Action Coding System*, EmFACS) [204] wurde angewandt, um die Aufnahmen zu annotieren. Die Gründe zur Verwendung dieser Datenbank lagen sowohl im Umfang an Datensätzen – dieser Sachverhalt spielt insbesondere bei der Merkmalsselektion eine Rolle – als auch darin, dass diese Datenbank in vielen unterschiedlichen Arbeiten zum Einsatz gelangte. Neben ihren Vorteilen hat die Datenbank auch Nachteile, zunächst wird die Mimik nur bis zur zweiten Phase, dem sogenannten Halten (engl. *apex*), gezeigt, zudem zeichnen sich die Mimiksequenzen durch eine starke Ausdrucksstärke der jeweiligen Mimik aus.

Abbildung 5.3: Beispielbilder aus der Cohn-Kanade-Datenbank [187, 188]

Für die Evaluierung der GM-Ansätze wurde neben der Cohn-Kanade-Datenbank auch die MMI-Datenbank [157, 158] verwendet. Diese Datenbank enthält Mimiksequenzen, die im Vergleich zur Cohn-Kanade-Datenbank weniger ausdrucksstark dargestellt werden und alle drei Phasen (Aktivierung, Halten, Entspannen) umfassen. Die MMI-Datenbank enthält, zusätzlich zur Frontalperspektive, auch eine Profilperspek-

tive. Eine erste Version der Datenbank wurde im Jahr 2005 vorgestellt [157], eine Erweiterung der Datenbank wurde im Jahr 2010 präsentiert [158].

5.2.2 Merkmale

Die Merkmale, die für die Erkennung von Gesichtsausdrücken verwendet werden können, lassen sich in zwei unterschiedliche Kategorien einteilen: Die erste Kategorie umfasst bild- bzw. erscheinungsbasierte Ansätze, während die zweite Kategorie modellbasierte Ansätze enthält. Die erste Kategorie versucht direkt relevante Merkmale aus dem Gesicht zu erhalten, beispielsweise werden in [183, 184] in sieben rechteckigen Regionen invariante Momente als Merkmale berechnet. Auch die Arbeiten [179, 180, 181] verwenden bild- bzw. erscheinungsbasierte Ansätze, hierbei kommen Verfahren wie Optischer Fluss, Fourier-Transformation sowie Wavelet-Transformation zum Einsatz, um in bestimmten Bildregionen (Augen, Mund) Merkmale zu berechnen.

Die zweite Kategorie versucht anhand eines Modells das menschliche Gesicht und relevante Merkmale davon (Position im 3-D-Raum, Form, Textur etc.) zu repräsentieren. Diese Art der Merkmalsgewinnung wird auch für die GM-Ansätze verwendet und wird nun näher beschrieben.

Prinzipiell kann man zwei Typen von Modellen für die Beschreibung eines Gesichtes oder eines Objektes im Allgemeinen finden: Der erste Typ beschreibt die Form des Gesichtes (Objektes), der zweite Typ beschreibt zusätzlich zur Form noch die Textur. Ein Aktives-Form-Modell (engl. *Active Shape Model*, ASM) [205, 206] kann die Lage, Skalierung und Form eines Objektes beschreiben, dafür werden, ausgehend von mehreren Trainingsbeispielen, Stützpunkte (engl. *landmarks*) des gesuchten Objektes manuell annotiert. Über eine PCA lassen sich die Durchschnittsform und die wesentlichen Verformungsrichtungen (anhand der Eigenvektoren der PCA) bestimmen. Das Aktive-Erscheinungs-Modell (engl. *Active Appearance Model*, AAM) [207] ist eine Erweiterung von ASM, hier wird nun auch die Information über die Textur in das Modell integriert. Diese Modelle enthalten aber nur 2-D-Informationen, 3-D-Informationen liefern andere Modelle, beispielsweise in [208] das sogenannte *Morphable 3D Face Model* oder auch das in dieser Arbeit zur Merkmalsextraktion verwendete CANDIDE-3-Modell [153].

Das CANDIDE-3-Modell erfreut sich aufgrund seiner Einfachheit großer Beliebtheit und wird in unterschiedlichen Arbeiten verwendet [209, 210, 211]. Das Modell besteht aus 113 3-D-Knoten und bildet insgesamt 168 Oberflächen, des Weiteren besitzt das Modell zwei unterschiedliche Parameterarten, Formparameter (engl. *Shape Parameters*) und Bewegungsparameter (engl. *Action Parameters*) [153]. Die Formparameter beschreiben die Anpassung des Modells an ein bestimmtes individuelles Gesicht, während die Bewegungsparameter verwendet werden, um über die Zeit unterschiedliche Gesichtsanimationen abzubilden [153]. Eine Darstellung des Modells findet sich in Abbildung 5.4.

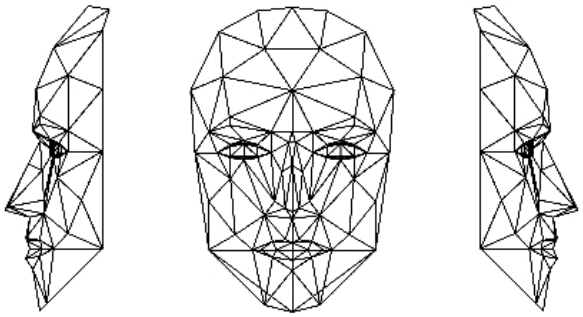


Abbildung 5.4: CANDIDE-3-Modell entnommen aus [153]

Das CANDIDE-3-Modell hat in seiner Beschreibung Korrespondenzen zu etablierten Systemen der Mimikbeschreibung. Das eine System sind die, bereits oben erwähnten, FACS, das zweite System beschreibt die FAPs, die im MPEG-4-Standard definiert worden sind [212]. Somit können die Bewegungsparameter des CANDIDE-3-Modells bestimmten Parametern des FACS zugeordnet werden bzw. haben Überschneidungen mit den FAPs. Die Extraktion der Merkmale zur Erkennung der sechs universellen Mimikemotionen ist durch die Arbeit von Mayer [177] erfolgt, daher wird hinsichtlich weiterer Details, wie z. B. Modellanpassung (engl. *Model Fitting*), direkt auf diese Arbeit verwiesen. Im Folgenden werden die gewonnenen Parameter aus dem CANDIDE-3-Modell vorgestellt, die als Merkmale für die GM-Ansätze dienen.

Insgesamt umfasst der aus dem CANDIDE-3-Modell gewonnene Merkmalsvektor elf Einträge, die in Tabelle 5.1 dargestellt sind. Hier findet sich eine Beschreibung des Parameters, die korrespondierende Bewegungseinheit (engl. *action unit*) des aktualisierten FACS sowie der Modellparametervektor (engl. *action unit vector*) des CANDIDE-3-Modells [153].

5.2.3 Ansätze

Im Folgenden werden zwei Ansätze betrachtet, die anhand von GMs in Form von DBNs die sechs universellen Mimikemotionen klassifizieren sollen. Zunächst werden aber noch kurz die Merkmalssätze vorgestellt, die bei den GM-Ansätzen für die Klassifikationsaufgabe zur Verwendung kommen.

5.2.3.1 Merkmalssätze

Die vorgestellten zwei GM-Ansätze verwenden die elf Merkmale aus dem CANDIDE-3-Modell aus Tabelle 5.1. Dieser ursprüngliche Merkmalssatz für einen Zeitpunkt t wird mit dem Vektor $\mathbf{f}_t^o$ beschrieben und enthält die absoluten Werte für die Merkmalsvektoren. Es handelt sich beim Ausdruck der sechs universellen Mimikemotionen um

Merkmal	Beschreibung	Bewegungseinheit Aciton Unit (AU)	Modellparametervektor Action Unit Vector (AUV)
1	Oberer-Lippen-Heber engl. *upper lip raiser*	AU10	AUV0
2	Kiefer-Senker engl. *jaw drop*	AU26/27	AUV11
3	Lippen-Strecker engl. *lip stretcher*	AU20	AUV2
4	Augenbrauen-Senker engl. *brow lowerer*	AU4	AUV3
5	Lippen-Ecken-Drücker engl. *lip corner depressor*	AU13/15	AUV14
6	Äußerer-Augenbrauen-Heber engl. *outer brow raiser*	AU2	AUV5
7	Augen-Geschlossen engl. *eyes closed*	AU42/43/44/45	AUV6
8	Lid-Verenger engl. *lid tightener*	AU7	AUV7
9	Nasen-Knitterer engl. *nose wrinkler*	AU9	AUV8
10	Lippen-Presser engl. *lip presser*	AU23/24	AUV9
11	Oberer-Lid-Heber engl. *upper lid raiser*	AU5	AUV10

Tabelle 5.1: Beschreibung der elf Merkmale für die Erkennung der sechs universellen Mimikemotionen. Daneben finden sich auch die korrespondierenden Beschreibungen der Bewegungseinheiten (*action units*) des FACS und zusätzlich die Modellparametervektoren (engl. *action unit vectors*) des CANDIDE-3-Modells [153].

einen dynamischen Prozess, daher ist die Betrachtung von ausschließlich zeitlichen Veränderungen der Merkmalsvektoren sinnvoll. Als Basisreferenz für die Differenzbestimmung dient für die jeweilige Person die neutrale Mimik der Person, diese wird am Anfang einer Sequenz bestimmt. Dieser neue differentielle Merkmalssatz für einen Zeitpunkt t wird mit dem Vektor $\mathbf{f}_t^d$ beschrieben. Zusätzlich können sowohl absolute als auch differentielle Merkmalsvektoren kombiniert werden, diese Zusammenfassung beschreibt den kompletten Merkmalssatz $\mathbf{f}_t^c$.

5.2.3.2 1. GM-Ansatz: Einfache DBN-Struktur

Die erste verwendete GM-Topologie (siehe Abbildung 5.5) ist eine einfache DBN-Struktur, hierbei werden die sechs unterschiedlichen universellen Gesichtsausdrücke zur Emotionsdarstellung (Freude, Traurigkeit, Wut, Ekel, Angst, Überraschung) mit einem einfachen Klassenknoten c_t modelliert. Im Gegensatz zu einem Ansatz, der

HMMs zur Klassifikation verwendet, wird bei diesem DBN-Ansatz nur ein Modell gelernt, während bei einem HMM-Ansatz für jede Klasse ein Modell trainiert werden würde. Zur Bestimmung der Parameter des DBN-Modells wird der EM-Algorithmus [86, 87] verwendet. Die Abbildung 5.5 zeigt die GM-Topologie des einfachen DBN-Ansatzes.

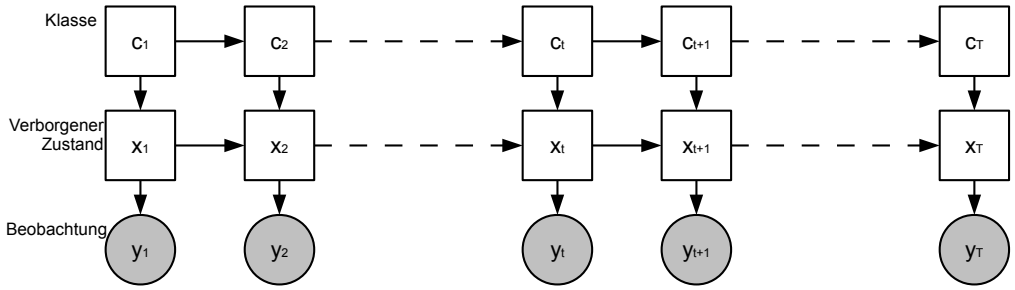


Abbildung 5.5: 1. GM-Ansatz: Einfache DBN-Struktur zur Erkennung der sechs universellen Gesichtsausdrücke zur Emotionsdarstellung

Die drei ZVs der Abbildung 5.5 beschreiben folgende Eigenschaften des Modells:

- Emotionsklasse c_t:
  Die ZV c_t beschreibt eine Klasse eines beobachteten Musters, in diesem Fall einen der sechs universellen Gesichtsausdrücke zur Emotionsdarstellung. Der zeitliche Übergang zwischen den Zuständen von c_{t-1} und c_t ist von deterministischer Natur, da angenommen wird, dass ein emotionaler Gesichtsausdruck eine komplette Sequenz umfasst, somit bleibt die Emotionsklasse während einer gesamten Sequenz unverändert.

- Verborgener Zustand x_t:
  Der zeitliche Ablauf der Mimik wird mithilfe eines verborgenen Knotens x_t modelliert, die einzelnen Phasen der Mimik werden mit unterschiedlichen Zuständen gestaltet.

- Beobachtungsvektor $\mathbf{y}_t$:
  Als Beobachtung $\mathbf{y}_t$ können die drei unterschiedlichen Merkmalssätze $\mathbf{f}_t^o$, $\mathbf{f}_t^d$ und $\mathbf{f}_t^c$, die anhand des CANDIDE-3-Modells bestimmt werden, verwendet werden. Eine Beschreibung der Merkmalssätze – $\mathbf{f}_t^o$, $\mathbf{f}_t^d$, $\mathbf{f}_t^c$ – findet sich in Abschnitt 5.2.3.1.

Die wahrscheinlichste Klasse $\hat{c}_T$ für die erfasste Beobachtungssequenz $\widehat{\mathcal{Y}}_T$ wird aus der Verbundwahrscheinlichkeit des ersten GM-Ansatzes wie folgt bestimmt:

$$\hat{c_T} = \arg\max_{C_T} \sum_{\mathcal{X}_T} p(C_T, \mathcal{X}_T, \widehat{\mathcal{Y}}_T) \tag{5.1}$$

$$\arg\max_{C_T} \sum_{\mathcal{X}_T} p(c_1) p(x_1|c_1) p(\widehat{\mathbf{y}}_1|x_1) \prod_{\tau=2}^{T} p(c_\tau|c_{\tau-1}) p(x_\tau|c_\tau, x_{\tau-1}) p(\widehat{\mathbf{y}}_\tau|x_\tau) \tag{5.2}$$

Bei Berücksichtigung der Vereinfachung der Emotionsklasse c_T (ähnlich wie Gleichung 4.22) ergibt sich:

$$\hat{c} = \arg\max_{C_T} \sum_{\mathcal{X}_T} p(c_1) p(x_1|c_1) p(\widehat{\mathbf{y}}_1|x_1) \prod_{\tau=2}^{T} \delta(c_\tau|c_{\tau-1}) p(x_\tau|c_\tau, x_{\tau-1}) p(\widehat{\mathbf{y}}_\tau|x_\tau) \tag{5.3}$$

$$= \arg\max_{c} \sum_{\mathcal{X}_T} p(c) p(x_1|c) p(\widehat{\mathbf{y}}_1|x_1) \prod_{\tau=2}^{T} p(x_\tau|c, x_{\tau-1}) p(\widehat{\mathbf{y}}_\tau|x_\tau) \tag{5.4}$$

5.2.3.3 2. GM-Ansatz: Doppel-Stream-DBN

Die zweite verwendete GM-Topologie (siehe Abbildung 5.6) besteht wiederum aus einem Klassenknoten, der die sechs universellen Gesichtsausdrücke unterscheidet. Das Modell besitzt aber nun zwei verborgene Zustände und ordnet die Merkmale aus dem CANDIDE-3-Modell den verborgenen Zuständen so zu, dass mit dem einen verborgenen Zustand die Abläufe der Mimik in der Augenregion beschrieben werden. Der zweite verborgene Zustand modelliert die Aktivitäten in der Mundregion. Ein ähnlicher Ansatz zu diesem ist der von [190], hier wird ein *Multi-Stream-HMM* verwendet, um die Merkmale der FAPs zwei unterschiedlichen *Streams* zuzuordnen. Ein *Stream* beschreibt die Augenbrauenregion (*eyebrow*), der zweite *Stream* die äußere Lippenregion (*outer-lip*). Es wird auch eine gewichtete *Multi-Stream-HMM*-Variante vorgestellt, hierbei hat die äußere Lippenregion mehr Einfluss auf die Klassifikation und erhöht damit auch die Erkennungsrate.

Die fünf ZVs der Abbildung 5.6 beschreiben folgende Eigenschaften des Modells:

- Emotionsklasse c_t:
  Die ZV c_t beschreibt, wie bei der einfachen DBN-Struktur, die Klasse und somit einen der sechs universellen Gesichtsausdrücke zur Emotionsdarstellung. Der

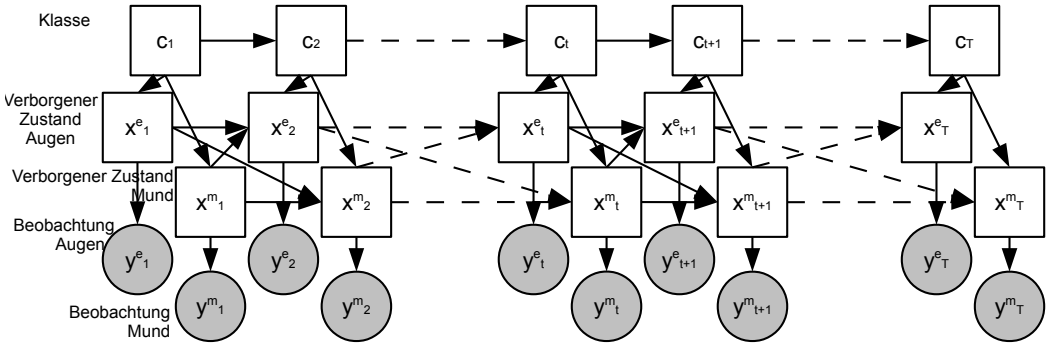


Abbildung 5.6: 2. GM-Ansatz: Doppel-Stream-DBN zur Erkennung der sechs universellen Gesichtsausdrücke zur Emotionsdarstellung

zeitliche Übergang zwischen den Zuständen von c_{t-1} und c_t ist auch wieder von deterministischer Natur.

- Verborgener Zustand Augenregion x_t^e:
  Der verborgene Knoten x_t^e modelliert den zeitlichen Verlauf der unterschiedlichen Mimikphasen in der Augenregion.

- Verborgener Zustand Mundregion x_t^m:
  Der verborgene Knoten x_t^m modelliert den zeitlichen Verlauf der unterschiedlichen Mimikphasen in der Mundregion.

- Beobachtungsvektor $\mathbf{y}_t^e$:
  Der Beobachtungsvektor $\mathbf{y}_t^e$ enthält alle Merkmale des CANDIDE-3-Modells, die in Wechselbeziehung mit der Augenregion stehen, somit umfasst der Vektor die folgenden fünf Merkmale von Tablelle 5.1: $\{4,6,7,8,11\}$. Für diese Merkmale können auch die drei unterschiedlichen Merkmalssätze $\mathbf{f}_t^o$, $\mathbf{f}_t^d$ und $\mathbf{f}_t^c$ verwendet werden (siehe Abschnitt 5.2.3.1 für eine Beschreibung der Merkmalssätze).

- Beobachtungsvektor $\mathbf{y}_t^m$:
  Der Beobachtungsvektor $\mathbf{y}_t^m$ enthält alle Merkmale des CANDIDE-3-Modells, die in Wechselbeziehung mit der Mundregion stehen, somit umfasst der Vektor die folgenden sechs Merkmale von Tablelle 5.1: $\{1,2,3,5,9,10\}$. Für diese Merkmale können auch die drei unterschiedlichen Merkmalssätze $\mathbf{f}_t^o$, $\mathbf{f}_t^d$ und $\mathbf{f}_t^c$ verwendet werden (siehe Abschnitt 5.2.3.1 für eine Beschreibung der Merkmalssätze).

Die wahrscheinlichste Klasse $\hat{c}_T$ für die erfassten Beobachtungssequenzen $\widehat{\mathcal{Y}}_T^e$ bzw. $\widehat{\mathcal{Y}}_T^m$ wird wie folgt aus der Verbundwahrscheinlichkeit des zweiten GM-Ansatzes bestimmt:

$$\hat{c_T} = \arg\max_{\mathcal{C}_T} \sum_{\mathcal{X}_T^e, \mathcal{X}_T^m} p(\mathcal{C}_T, \mathcal{X}_T^e, \mathcal{X}_T^m, \widehat{\mathcal{Y}}_T^e, \widehat{\mathcal{Y}}_T^m) \tag{5.5}$$

$$= \arg\max_{\mathcal{C}_T} \sum_{\mathcal{X}_T^e, \mathcal{X}_T^m} p(c_1) p(x_1^e|c_1) p(x_1^m|c_1) p(\widehat{\mathbf{y}}_1^e|x_1^e) p(\widehat{\mathbf{y}}_1^m|x_1^m)$$
$$\prod_{\tau=2}^{T} p(c_\tau|c_{\tau-1}) p(x_\tau^e|c_\tau, x_{\tau-1}^e) p(x_\tau^m|c_\tau, x_{\tau-1}^m) p(\widehat{\mathbf{y}}_\tau^e|x_\tau^e) p(\widehat{\mathbf{y}}_\tau^m|x_\tau^m) \tag{5.6}$$

Bei Berücksichtigung der Vereinfachung der Emotionsklasse c_T (ähnlich wie Gleichung 4.22) ergibt sich:

$$\hat{c} = \arg\max_{\mathcal{C}_T} \sum_{\mathcal{X}_T^e, \mathcal{X}_T^m} p(c_1) p(x_1^e|c_1) p(x_1^m|c_1) p(\widehat{\mathbf{y}}_1^e|x_1^e) p(\widehat{\mathbf{y}}_1^m|x_1^m)$$
$$\prod_{\tau=2}^{T} \delta(c_\tau|c_{\tau-1}) p(x_\tau^e|c_\tau, x_{\tau-1}^e) p(x_\tau^m|c_\tau, x_{\tau-1}^m) p(\widehat{\mathbf{y}}_\tau^e|x_\tau^e) p(\widehat{\mathbf{y}}_\tau^m|x_\tau^m) \tag{5.7}$$

$$= \arg\max_{c} \sum_{\mathcal{X}_T^e, \mathcal{X}_T^m} p(c) p(x_1^e|c) p(x_1^m|c) p(\widehat{\mathbf{y}}_1^e|x_1^e) p(\widehat{\mathbf{y}}_1^m|x_1^m)$$
$$\prod_{\tau=2}^{T} p(x_\tau^e|c, x_{\tau-1}^e) p(x_\tau^m|c, x_{\tau-1}^m) p(\widehat{\mathbf{y}}_\tau^e|x_\tau^e) p(\widehat{\mathbf{y}}_\tau^m|x_\tau^m) \tag{5.8}$$

Neben den zwei vorgestellten GM-Varianten wurden noch weitere GM-Topologieformen erstellt und evaluiert. Beispielsweise wurde im *Doppel-Stream-DBN*-Ansatz auf die Verbindung zwischen den verborgenen Knoten der Augen- bzw. Mundregion verzichtet, um eine unabhängige Modellierung der Mimik zu bilden. Ein anderes GM hat die Merkmale von zwei Zeitpunkten $(t, t+1)$ in einem verborgenen Knoten kombiniert, um die Veränderung der Mimik im zeitlichen Fortschritt besser zu erfassen. Diese Ansätze lieferten aber keine besseren Ergebnisse, daher werden nur die Ergebnisse der beiden vorgestellten GM-Ansätze betrachtet.

5.2.4 Ergebnisse

Als Datenbanken zur Erkennung der sechs universellen Mimikemotionen für die beiden GM-Ansätze wurden sowohl die Cohn-Kanade-Datenbank [187, 188] als auch die MMI-Datenbank [157, 158] verwendet. Bei der Cohn-Kanade-Datenbank sind für die Experimente insgesamt 331 Mimiksequenzen verwendet worden, davon zeigen 35 Sequenzen Wut, 36 Ekel, 57 Angst, 74 Freude, 59 Traurigkeit und 70 Überraschung. Aus der MMI-Datenbank sind insgesamt 109 Mimiksequenzen zum Einsatz gekommen, zwölf Sequenzen zeigen Wut, elf Ekel, 21 Angst, 25 Freude, 20 Traurigkeit und

20 Überraschung. Zur Evaluierung wurde eine Fünffach-Kreuzvalidierung verwendet, dabei wurde darauf geachtet, dass eine personenunabhängige Einteilung (die gleiche Person war nicht zur selben Zeit in der Trainings- und Testmenge) erfolgte. Es wurden alle drei unterschiedlichen Merkmalssätze $\mathbf{f}_t^o$, $\mathbf{f}_t^d$ und $\mathbf{f}_t^c$ in den Experimenten verwendet.

Tabelle 5.2 zeigt alle Ergebnisse für die zwei GM-Ansätze und die drei unterschiedlichen Merkmalssätze, die mit dem Datensatz der Cohn-Kanade-Datenbank erzielt worden sind.

	Merkmalssatz		
	$\mathbf{f}_t^o$	$\mathbf{f}_t^d$	$\mathbf{f}_t^c$
1. GM-Ansatz	66,47 %	81,27 %	75,23 %
2. GM-Ansatz	62,84 %	78,25 %	72,21 %

Tabelle 5.2: Ergebnisse zur Erkennung der sechs universellen Mimikemotionen mit den Daten aus der Cohn-Kanade-Datenbank. Es werden zwei GM-Ansätze vorgestellt, die auf drei unterschiedlichen Merkmalssätzen trainiert worden sind.

Für beide GM-Ansätze gilt, dass der differentielle Merkmalssatz $\mathbf{f}_t^d$ die besten Ergebnisse liefert und der Merkmalssatz mit den ursprünglichen Merkmalen $\mathbf{f}_t^o$ die schlechtesten. Die Erkennungsleistung der beiden GM-Ansätze war für alle drei Merkmalssätze mit dem Test zur statistischen Signifikanz von Abschnitt 3.2.2 verglichen worden, hierbei zeigte sich aber keinerlei statisch signifikanter Unterschied zwischen beiden GM-Ansätzen.

Die Ergebnisse, die mit der MMI-Datenbank erzielt worden sind, sind in Tabelle 5.3 dargestellt, für diese Ergebnisse ergibt sich ein anderes Bild als bei der Cohn-Kanade-Datenbank.

	Merkmalssatz		
	$\mathbf{f}_t^o$	$\mathbf{f}_t^d$	$\mathbf{f}_t^c$
1. GM-Ansatz	62,39 %	60,55 %	61,47 %
2. GM-Ansatz	63,30 %	65,14 %	59,63 %

Tabelle 5.3: Ergebnisse zur Erkennung der sechs universellen Mimikemotionen mit den Daten aus der MMI-Datenbank. Es werden zwei GM-Ansätze vorgestellt, die auf drei unterschiedlichen Merkmalssätzen trainiert worden sind.

Alle Ergebnisse aus der MMI-Datenbank fallen, im Vergleich zur Cohn-Kanade-Datenbank, deutlich schlechter aus, was zum einen daran liegt, dass weniger Datensätze zum Training zur Verfügung gestanden haben, und zum anderen, dass die Ausdrucksstärke der Mimiksequenzen in den Datenbanken verschiedenartig ist. Die unterschiedlichen Merkmalssätze liefern auf den zwei GM-Ansätzen unterschiedliche Resultate, es ergibt sich kein einheitliches Bild mehr wie bei der Cohn-Kanade-Datenbank. Die Erkennungsleistung der beiden GM-Ansätze war wiederum mit dem

Test zur statistischen Signifikanz von Abschnitt 3.2.2 verglichen worden, hierbei zeigte sich aber keinerlei statisch signifikanter Unterschied zwischen beiden GM-Ansätzen für die drei Merkmalssätze.

Die erzielten Ergebnisse mit den vorgestellten GM-Ansätzen für die Cohn-Kanade-Datenbank bzw. die MMI-Datenbank können mit einem Ansatz von Mayer [177] verglichen werden, der zur Klassifikation SVMs verwendet hat. Dieser Vergleich eignet sich besonders, da die Merkmale, die für die Klassifikation verwendet werden, von der gleichen Umsetzung des CANDIDE-3-Modells gewonnen worden sind. Die erzielten Ergebnisse der SVM-basierten Klassifikation sind in Tabelle 5.4 abgebildet. Bei diesen Ergebnissen ist zu beachten, dass, gleich den vorgestellten GM-Ansätzen, alle Merkmale vom ersten bis zum letzten Bild der Mimiksequenzen sowohl zum Training als auch zum Testen verwendet worden sind.

	Merkmalssatz		
	$\mathbf{f}_t^o$	$\mathbf{f}_t^d$	$\mathbf{f}_t^c$
Cohn-Kanade-Datenbank	61,4 %	78,9 %	76,2 %
MMI-Datenbank	60,9 %	61,3 %	61,6 %

Tabelle 5.4: Ergebnisse aus [177] für Mimikemotionserkennung auf Cohn-Kanade-Datenbank bzw. der MMI-Datenbank. Es werden die Ergebnisse für die drei unterschiedlichen Merkmalssätze vorgestellt.

Ein zentraler Ansatz der Arbeit von Mayer [177] war, die Mimikklassifikation auf unterschiedlichen Datenbanken zu untersuchen, zur Realisierung dieser Experimente waren drei unterschiedliche Datenbanken (Cohn-Kanade-Datenbank, MMI-Datenbank, *FGNet Facial Emotions and Expressions Database*) verwendet worden. In Tabelle 5.5 sind die Ergebnisse der Datenbank-Kreuzvalidierung (Training des Klassifikators auf einer Datenbank, Testen auf einer anderen Datenbank) für die Cohn-Kanade-Datenbank und MMI-Datenbank dargestellt. Es zeigt sich, dass, trotz der Annahme einer universellen Darstellung der sechs Mimikemotionen, die Ergebnisse für die Datenbank-Kreuzvalidierung erheblich schlechter sind, als wenn nur eine Datenbank zum Training sowie zum Testen verwendet wird.

	Test-Datenbank					
	Cohn-Kanade			MMI		
Training-Datenbank	$\mathbf{f}_t^o$	$\mathbf{f}_t^d$	$\mathbf{f}_t^c$	$\mathbf{f}_t^o$	$\mathbf{f}_t^d$	$\mathbf{f}_t^c$
Cohn-Kanade	61,4 %	78,9 %	76,2 %	45,2 %	53,2 %	53,2 %
MMI	41,3 %	60,8 %	49,3 %	60,9 %	61,3 %	61,6 %

Tabelle 5.5: Ergebnisse der Datenbank-Kreuzvalidierung aus [177] für Mimikemotionserkennung auf Cohn-Kanade-Datenbank bzw. der MMI-Datenbank. Es werden die Ergebnisse für die drei unterschiedlichen Merkmalssätze vorgestellt.

Im Hinblick auf die vorgestellten GM-Ansätze ist die Datenbank-Kreuzvalidierung für die drei unterschiedlichen Merkmalssätze wiederholt worden, hierbei zeigt sich, dass auch die Ergebnisse der Datenbank-Kreuzvalidierung deutlich schlechter ausfallen. Diese bleiben vor allem, wenn die MMI-Datenbank als Trainingsgrundlage dient, auch hinter den Ergebnissen der SVM-basierten Klassifikation zurück. Die erzielten Ergebnisse für die Datenbank-Kreuzvalidierung sind in Tabelle 5.6 dargestellt.

GM-Ansatz	Training-Datenbank	Test-Datenbank: Cohn-Kanade $\mathbf{f}_t^o$	Cohn-Kanade $\mathbf{f}_t^d$	Cohn-Kanade $\mathbf{f}_t^c$	MMI $\mathbf{f}_t^o$	MMI $\mathbf{f}_t^d$	MMI $\mathbf{f}_t^c$
1. GM-Ansatz	Cohn-Kanade	66,47 %	81,27 %	75,23 %	35,78 %	57,80 %	49,54 %
	MMI	27,19 %	43,81 %	38,07 %	62,39 %	60,55 %	61,47 %
2. GM-Ansatz	Cohn-Kanade	62,84 %	78,25 %	72,21 %	36,70 %	47,71 %	33,03 %
	MMI	25,68 %	43,81 %	30,82 %	63,30 %	65,14 %	59,63 %

Tabelle 5.6: Ergebnisse der Kreuzvalidierung der beiden GM-Ansätze für Mimikemotionserkennung auf Cohn-Kanade-Datenbank bzw. der MMI-Datenbank. Es werden die Ergebnisse für die drei unterschiedlichen Merkmalssätze vorgestellt.

Die Ergebnisse aus der Arbeit von Mayer [177] für das Trainieren und Testen auf der gleichen Datenbank (siehe Tabelle 5.4) sind denen der vorgestellten GM-Ansätze im Großen und Ganzen recht ähnlich. Die Ergebnisse der SVM-basierten Klassifikation von [177] können aber deutlich verbessert werden, wenn nicht die ganze Mimiksequenz, bestehend aus den Merkmalen vom ersten bis zum letzten Bild, sondern nur die Bilder aus der zweiten Phase, dem sogenannten Halten (engl. *apex*), zum Trainieren und Testen verwendet werden. Mit diesem Vorgehen lässt sich bei Mayer [177] eine Erkennungsrate von 84,1 % auf der Cohn-Kanade-Datenbank erzielen, für die MMI-Datenbank wird eine Erkennungsrate von 79,8 % erreicht. Anhand dieser Ergebnisse wird ersichtlich, dass die Merkmale aus den verschiedenen Mimikphasen unterschiedliche Beiträge zur Klassifikation liefern. Deshalb scheint es sinnvoll zu sein, eine Merkmalsselektion als einen Vorverabeitungsschritt vor der Klassifikation einzuführen.

5.3 Merkmalsselektion

Die Merkmalsselektion verfolgt das Ziel aus einer gegebenen vollständigen Merkmalsmenge $\vec{f}^K = \{m_1, \ldots, m_K\}$ mit der Kardinalität $\mathcal{K}$ eine Teilmenge $\vec{f}^S = \{\hat{m}_1, \ldots, \hat{m}_S\}$ mit der Kardinalität $\mathcal{S}$ anhand eines Evaluierungskriteriums $J(\vec{f}^S)$ auszuwählen, dabei können laut [213] drei unterschiedliche Kriterien zur Auswahl der Merkmale herangezogen werden: 1) $J_1(\vec{f}^S)$: Die ausgewählte Teilmenge der Merkmale mit einem bestimmten Umfang maximiert das Evaluierungskriterium. 2) $J_2(\vec{f}^S)$: Eine ausgewählte kleinere Teilmenge der Merkmale erfüllt eine definierte Einschränkung in Bezug auf

das Evaluierungskriterium. 3) $J_3(\vec{f}^S)$: Allgemeiner Fall, die ausgewählte Teilmenge stellt einen Ausgleich zwischen dem, einerseits, zu maximierenden Evaluierungskriterium und der, andererseits, zu minimierenden Merkmalsteilmenge her.

Prinzipiell können aus einer Menge, die aus N-Elementen besteht, 2^N-Teilmengen gebildet werden [214, 213], daher zeigt sich, dass das Problem der Merkmalsselektion schnell rechenintensiv wird, um eine optimale Lösung unter Berücksichtigung aller Möglichkeiten zu finden. Für die Merkmalsselektion können die Verfahren nach unterschiedlichen Kriterien unterschieden werden (siehe Abbildung 5.7), laut [215] können zwei wesentliche Kategorien festgelegt werden: Verfahren, die künstliche neuronale Netze verwenden, und Verfahren, die auf statistischer Mustererkennung beruhen. Bei den statistischen Mustererkennungsverfahren lässt sich zwischen optimalen Verfahren, die alle möglichen Teilmengen der vollständigen Merkmalsmenge durchtesten, und suboptimalen Verfahren, die nicht alle möglichen Teilmengen durchtesten, unterscheiden [215]. Die suboptimalen Verfahren unterteilen sich in deterministische und stochastische Ansätze, die nur eine Lösungen finden, und deterministische und stochastische Ansätze, die mehrere Lösungen erzeugen [215].

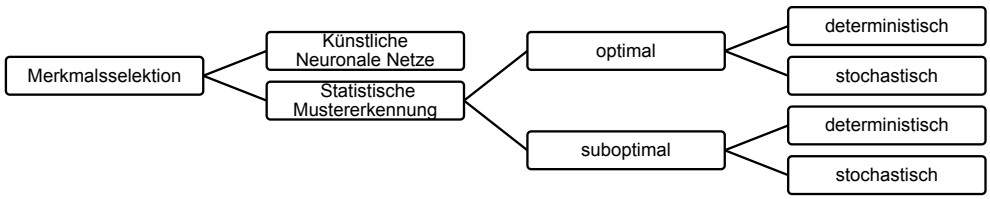


Abbildung 5.7: Unterteilung der Merkmalsselektion-Verfahren nach [215]

Basierend auf den Ergebnissen von [177] – wobei sich gezeigt hat, dass sich auf den Bildern der zweiten Phasen, dem sogenannten Halten (engl. *apex*), die besten Ergebnisse erzielt worden sind – ist es sinnvoll auch die zeitliche Komponente in die Merkmalsselektion mit einfließen zu lassen. Daher wird für die vorgestellten suboptimalen sowie deterministischen Verfahren der Merkmalsselektion auch die zeitliche Phase der Mimik berücksichtigt.

5.3.1 Sequentielle Merkmalsselektion

Sequentielle Verfahren zur Merkmalsselektion beruhen darauf, dass die Merkmale aus der ursprünglichen vollständigen Merkmalsmenge $\vec{f}^K$ der Reihe nach ausgewählt werden, um eine neue Teilmenge von Merkmalen $\vec{f}^S$ zu erzeugen, dabei kann man zwei grundlegend verschiedene Verfahren unterscheiden: Sequentielle Vorwärts-Auswahl (engl. *Sequential Forward Selection*, SFS) [214] und Sequentielle Rückwärts-Auswahl (engl. *Sequential Backward Selection*, SBS) [216]. Der wesentliche Unterschied zwischen SFS und SBS liegt darin, dass die Ausgangsmenge für den Merkmalssatz verschieden ist. Die SFS beginnt mit einer leeren Menge für den Merkmalssatz und fügt

in weiteren Iterationen Merkmale hinzu. Die SBS beginnt im Gegensatz zur SFS mit dem vollständigen Merkmalssatz und entfernt in weiteren Iterationen Merkmale aus dieser Menge.

Die SFS ist in [214] vorgestellt worden, dieses Verfahren ist vorteilhaft, wenn der zu selektierende Merkmalssatz $\vec{f}^S$ eine kleine Teilmenge des vollständigen Merkmalssatzes $\vec{f}^K$ ist. Als nachteilig bei diesem Verfahren erweist sich die Tatsache, dass, nachdem ein Merkmal einmal in den Merkmalssatz $\vec{f}^S$ aufgenommen worden ist, der Beitrag dieses Merkmals nicht mehr erneut in weiteren Iterationsschritten des SFS-Algorithmus untersucht wird. Deshalb verbleibt ein ausgewähltes Merkmal auch in dem Merkmalssatz $\vec{f}^S$, obwohl dessen Beitrag möglicherweise in einem weiteren Iterationsschritt obsolet geworden ist [217]. Algorithmus 1 zeigt die notwendigen Schritte, um die SFS durchzuführen.

Algorithmus 1 SFS

Ausgehend von einem leeren Merkmalssatz $\vec{f}_0^S = \{\varnothing\}$ und $i = 0$
while Evaluierungskriterium $J_j(\vec{f}_i^S)$ $j \in [1,2,3]$ nicht erfüllt: **do**

1. Wähle das nächstbeste Merkmal $m \in \vec{f}^K$, das das Evaluierungskriterium $J_j(\vec{f}_i^S + m)$ maximiert
   $$\hat{m}_i = \arg\max_{m \notin \vec{f}_i^S} \mathcal{F}_j(\vec{f}_i^S + m)$$
2. Erzeuge den neuen Merkmalssatz $\vec{f}_{i+1}^S = \vec{f}_i^S + \hat{m}_i$
3. $i = i + 1$

end while
$\vec{f}_i^S$ bildet den finalen Merkmalssatz der SFS.

Die Auswahl der relevanten Merkmale anhand der SFS erfolgt für die Mimiksequenzen unter Berücksichtigung der zeitlichen Phase im Mimikverlauf, daher werden für jeden Zeitschritt die relevanten Merkmale bestimmt. Die Experimente wurden auf der Cohn-Kanade-Datenbank durchgeführt, dabei kamen unterschiedliche Funktionen $\mathcal{F}$ – LDA, Quadratische Diskriminanzanalyse (engl. *Quadratic Discriminant Analysis*, QDA), Mahalanobis-Distanz (MD) [218] – als Evaluierungskriterium $J_3(\vec{f}_i^S)$ zum Einsatz. Die Funktionen LDA, QDA und MD dienen als Diskriminanzfunktionen für die Klassifikation zwischen den sechs Emotionsklassen. Tabelle 5.7 zeigt die erzielten Ergebnisse mit einer Merkmalsselektion, die auf der SFS basieren. Die Experimente sind sowohl auf den beiden GM-Ansätzen als auch auf den drei unterschiedlichen Merkmalssätzen ($\mathbf{f}_t^o$, $\mathbf{f}_t^d$, $\mathbf{f}_t^c$) durchgeführt worden.

Alle Ergebnisse sind besser als die ursprünglichen Ergebnisse ohne Merkmalsselektion (siehe Tabelle 5.2), es ergeben sich mit dem Test zur statistischen Signifikanz von Abschnitt 3.2.2 sogar signifikante (angemerkt mittels $^+$) bzw. hochsignifikante (angemerkt mittels *) Verbesserungen.

	Merkmalssatz		
	$\mathbf{f}_t^o$	$\mathbf{f}_t^d$	$\mathbf{f}_t^c$
1. GM-Ansatz	${}_{MD}69{,}49\,\%$	${}_{QDA}83{,}38\,\%$	${}_{LDA}79{,}15\,\%$
2. GM-Ansatz	${}_{LDA}67{,}67\,\%$	${}_{LDA}{}^{*}83{,}38\,\%$	${}_{LDA}{}^{+}77{,}64\,\%$

Tabelle 5.7: Ergebnisse zur Erkennung der sechs universellen Mimikemotionen mit den Daten aus der Cohn-Kanade-Datenbank. Ein Vorverarbeitungsschritt, der auf die SFS aufbaut, wählt relevante Merkmale aus. Es werden beide GM-Ansätze sowie die drei Merkmalssätze betrachtet.

Ein ähnlicher Ansatz wie die SFS zur Merkmalsselektion ist die SBS, hierbei wird aber von dem kompletten Merkmalssatz $\vec{f}^K$ ausgegangen, und iterativ ein Merkmal aus dem Merkmalssatz entfernt. In Algorithmus 2 wird die systematische Vorgehensweise beschrieben. Die SBS, vorgestellt in [216], eignet sich für den Fall, wenn die zu selektierende Merkmalsmenge einen großen Umfang der ursprünglichen Merkmalsmenge $\vec{f}^K$ abdecken soll, nachteilig erweist sich die Tatsache, dass, wenn ein Merkmal erst einmal aussortiert worden ist, es nicht mehr in einem weiteren Iterationsschritt ausgewählt werden kann.

Algorithmus 2 SBS

Ausgehend von einem vollen Merkmalssatz $\vec{f}_0^S = \vec{f}^K$ und $i = 0$
while Evaluierungskriterium $J_j(\vec{f}_i^S)$ $j \in [1,2,3]$ noch erfüllt: **do**

1. Entferne das nächste Merkmal $m \in \vec{f}^K$, das das Evaluierungskriterium $J_j(\vec{f}_i^S - m)$ maximiert
   $$\hat{m}_i = \arg\max_{m \in \vec{f}_i^S} \mathcal{F}_j(\vec{f}_i^S - m)$$
2. Erzeuge den neuen Merkmalssatz $\vec{f}_{i+1}^S = \vec{f}_i^S - \hat{m}_i$
3. $i = i + 1$

end while
$\vec{f}_{i-1}^S$ bildet den finalen Merkmalssatz der SBS.

Die Ergebnisse der Merkmalsselektion, die auf dem SBS-Verfahren basiert, sind in Tabelle 5.8 dargestellt. Die Experimente sind sowohl auf den beiden GM-Ansätzen als auch auf den drei unterschiedlichen Merkmalssätzen ($\mathbf{f}_t^o$, $\mathbf{f}_t^d$, $\mathbf{f}_t^c$) durchgeführt worden. Genau wie bei dem SFS-Verfahren kamen wieder die unterschiedlichen Funktionen $\mathcal{F}$ – LDA, QDA, MD [218] – als Evaluierungskriterium $J_3(\vec{f}_i^S)$ zum Einsatz.

Die Ergebnisse an sich stellen eine leichte Verbesserung gegenüber den Ergebnissen ohne Merkmalsselektion von Tabelle 5.2 dar, dennoch bleiben sie hinter den Ergebnissen der Merkmalsselektion, die auf dem SFS-Verfahren basiert, zurück. Des Weiteren sind die Verbesserungen auch nicht statistisch signifikant.

	Merkmalssatz		
	$\mathbf{f}_t^o$	$\mathbf{f}_t^d$	$\mathbf{f}_t^c$
1. GM-Ansatz	$_{LDA}$67,07 %	$_{MD}$82,78 %	$_{QDA}$78,25 %
2. GM-Ansatz	$_{LDA}$65,56 %	$_{LDA}$78,55 %	$_{QDA}$74,92 %

Tabelle 5.8: Ergebnisse zur Erkennung der sechs universellen Mimikemotionen mit den Daten aus der Cohn-Kanade-Datenbank. Ein Vorverarbeitungsschritt, der auf dem SBS-Verfahren basiert, wählt relevante Merkmale aus. Es werden beide GM-Ansätze sowie die drei Merkmalssätze betrachtet.

5.3.2 Kullback-Leibler-Divergenz-Ansatz

Die Merkmalsselektion übte durchweg einen positiven Einfluss auf die Ergebnisse aus. Im Folgenden wird ein Kullback-Leibler-Divergenz-Ansatz vorgestellt, um Merkmale auszuwählen. Es kommen zwei verschiedene Verfahren beim Kullback-Leibler-Divergenz-Ansatz zum Einsatz, das erste Verfahren wählt eine bestimmte Menge an Merkmalen aus, das zweite Verfahren verwendet einen Schwellenwert, um die Merkmale zu selektieren.

Der Ansatz zur Merkmalsselektion der beiden Verfahren beruht auf der Kullback-Leibler-Divergenz, diese ist in [219] vorgestellt worden. Die Kullback-Leibler-Divergenz ist ein Maß, um die Ungleichheit zwischen zwei WFs bzw. WDFs zu bestimmen. Der Unterschied zwischen zwei diskreten WFs $p(x)$ und $q(x)$ wird mit der Kullback-Leibler-Divergenz $D(p||q)$ wie folgt bestimmt:

$$D(p||q) = \sum_{x \in X} p(x) \log \frac{p(x)}{q(x)}. \tag{5.9}$$

Die Kullback-Leibler-Divergenz besitzt folgende Eigenschaften:

- $D(p||q) \geq 0$
- $D(p||q) = 0$, wenn und nur wenn $p(x) = q(x)$
- Die Kullback-Leibler-Divergenz ist nicht symmetrisch, daher gilt $D(p||q) \neq D(q||p)$.

Die Kullback-Leibler-Divergenz wird in unterschiedlichen Anwendungen zur Merkmalsselektion verwendet, beispielsweise wird sie in [220] zur Merkmalsselektion im Bereich der Textklassifikation eingesetzt, [221] nutzt die Kullback-Leibler-Divergenz zur Merkmalsbewertung im Bereich der Textkategorisierung.

Das erste der beiden Verfahren bestimmt anhand des Evaluierungskriteriums $J_1()$ eine feste Anzahl $\mathcal{S}$ an Merkmalen. Hierbei wird die Kullback-Leibler-Divergenz zwischen der WF eines Merkmals ${}^{c_i}p(m_j)$ einer bestimmten Klasse c_i und der WF des

gleichen Merkmals ${}^{\mathcal{C}}p(m_j)$ für die gesamte Klassenmenge bestimmt. Dieses Verfahren bestimmt nun für jeden Zeitschritt $\tau \in [1,\ldots,t,\ldots,T]$ die $\mathcal{S}$-relevanten Merkmale, die exakte Vorgehensweise zur Auswahl der Merkmale für einen Zeitschritt t wird in Algorithmus 3 beschrieben.

Algorithmus 3 Erstes Verfahren: Feste Anzahl an Merkmalen

${}^{c_l}p(m_j)$ ist die WF des j-ten Merkmals der l-ten Klasse c.
${}^{\mathcal{C}}p(m_j)$ ist die WF des j-ten Merkmals über alle Klassen $\mathcal{C}$.
$\mathcal{S}$ ist die Anzahl, der zu selektierenden Merkmale.
Ausgehend von einem leeren Merkmalssatz $\vec{f}^S = \{\varnothing\}$.
for $i = 1 \rightarrow \mathcal{S}$ **do**

1. Wähle das Merkmal $m_j \in \vec{f}^K$, das folgendes Evaluierungskriterium $J_1()$ maximiert:
$$\hat{m}_i = \arg\max_{m_j \notin \vec{f}^S} \sum_{l=1}^{\mathcal{C}} D\Big({}^{c_l}p(m_j) \| {}^{\mathcal{C}}p(m_j)\Big)$$
2. Erzeuge den neuen Merkmalssatz $\vec{f}^S = \vec{f}^S + \hat{m}_i$

end for
$\vec{f}^S$ bildet den finalen Merkmalssatz.

Die Anzahl der zu selektierenden Merkmale $\mathcal{S}$ wurde für alle drei Merkmalssätze $(\mathbf{f}_t^o, \mathbf{f}_t^d, \mathbf{f}_t^c)$ innerhalb der Grenzen $\mathcal{S}^{\mathbf{f}_t^o} \in \{1,\ldots 10\}$, $\mathcal{S}^{\mathbf{f}_t^d} \in \{1,\ldots 10\}$ sowie $\mathcal{S}^{\mathbf{f}_t^c} \in \{1,\ldots 21\}$ variiert, um die Klassifikationsergebnisse zu erhalten.

Die besten Klassifikationsergebnisse der Merkmalsselektion, basierend auf dem ersten Kullback-Leibler-Divergenz-Verfahren, sind in Tabelle 5.9 dargestellt. Die Anzahl der ausgewählten Merkmale $\mathcal{S}$ findet sich links unten. Der Vorverarbeitungsschritt wurde auf allen drei unterschiedlichen Merkmalssätzen $(\mathbf{f}_t^o, \mathbf{f}_t^d, \mathbf{f}_t^c)$ durchgeführt.

	Merkmalssatz		
	$\mathbf{f}_t^o$	$\mathbf{f}_t^d$	$\mathbf{f}_t^c$
1. GM-Ansatz	${}_{10}68{,}58\,\%$	${}_{9}82{,}78\,\%$	${}^{*}_{12}83{,}38\,\%$
2. GM-Ansatz	${}_{5}64{,}95\,\%$	${}^{+}_{9}82{,}18\,\%$	${}^{+}_{12}77{,}95\,\%$

Tabelle 5.9: Ergebnisse zur Erkennung der sechs universellen Mimikemotionen mit den Daten aus der Cohn-Kanade-Datenbank. Ein Vorverarbeitungsschritt basierend auf der Kullback-Leibler-Divergenz wählt eine feste Anzahl an relevanten Merkmalen aus (siehe Zahl links unten). Es werden beide GM-Ansätze sowie die drei Merkmalssätze betrachtet.

Alle Ergebnisse sind besser als die ursprünglichen Ergebnisse (siehe Tabelle 5.2) ohne Merkmalsselektion, für den ersten GM-Ansatz und den kompletten Merkmalssatz $\mathbf{f}_t^c$ ergibt sich eine hochsignifikante Verbesserung (angemerkt mittels *). Für den zweiten GM-Ansatz findet sich für den differentiellen Merkmalssatz $\mathbf{f}_t^d$ und den kompletten Merkmalssatz $\mathbf{f}_t^c$ eine signifikante Verbesserung (angemerkt mittels $^+$).

Das erste Kullback-Leibler-Divergenz-Verfahren bestimmt eine feste Anzahl von Merkmalen für jeden Zeitschritt, und obwohl mit diesem Ansatz teilweise eine signifikante bzw. hochsignifikante Verbesserung erzielt worden ist, berücksichtigt dieses Verfahren nicht die Tatsache, dass die verschiedenen Phasen der Mimik unterschiedliche Beiträge zur Erkennungsleistung haben. Diesem Sachverhalt wird mit dem zweiten Kullback-Leibler-Divergenz-Verfahren Rechnung getragen, hierbei wird mittels eines Schwellenwerts eine bestimmte Menge an Merkmalen für einen Zeitschritt t ausgewählt. Die genaue Vorgehensweise für das zweite Kullback-Leibler-Divergenz-Verfahren ist in Algorithmus 4 beschrieben.

Algorithmus 4 Zweites Verfahren: Auswahl, basierend auf einem Schwellenwert

${}^{c_l}p(m_i)$ ist die WF des i-ten Merkmals der l-ten Klasse c.
${}^{\mathcal{C}}p(m_i)$ ist die WF des i-ten Merkmals über alle Klassen $\mathcal{C}$.
k ist ein beliebiger positiver Schwellenwert.
Ausgehend von einem leeren Merkmalssatz $\vec{f}^S = \{\varnothing\}$.
for $i = 1 \rightarrow \mathcal{K}$ **do**
  Wähle das Merkmal $m_i \in \vec{f}^K$, das folgendes Evaluierungskriterium $J_3()$ maximiert:
    $KB = \arg\max_l D\left({}^{c_l}p(m_i) \| {}^{\mathcal{C}}p(m_i)\right)$
  if $KB \geq k$ **then**
    Erzeuge den neuen Merkmalssatz $\vec{f}^S = \vec{f}^S + m_i$
  else
    $\vec{f}^S = \vec{f}^S$
  end if
end for
$\vec{f}^S$ bildet den finalen Merkmalssatz.

Ähnlich wie bei der Anzahl der zu selektierenden Merkmale $\mathcal{S}$ kann auch der Schwellenwert k variiert werden, um die Anzahl der ausgewählten Merkmale und somit die Klassifikationsergebnisse zu beeinflussen.

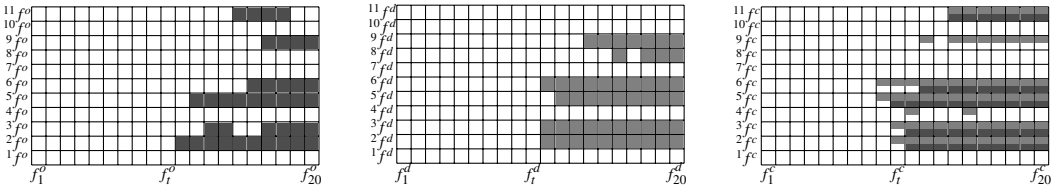
Die besten Klassifikationsergebnisse der Merkmalsselektion, basierend auf dem zweiten Kullback-Leibler-Divergenz-Verfahren, sind in Tabelle 5.10 dargestellt. Der Vorverarbeitungsschritt wurde auf allen drei unterschiedlichen Merkmalssätzen ($\mathbf{f}_t^o$, $\mathbf{f}_t^d$, $\mathbf{f}_t^c$) durchgeführt.

Alle Ergebnisse sind besser als die ursprünglichen Ergebnisse (siehe Tabelle 5.2), es ergeben sich durchweg signifikante (dargestellt durch $^+$) bzw. hochsignifikante (dargestellt durch *) Verbesserungen für alle drei Merkmalssätze. Generell zeigt sich, dass die besten Ergebnisse erreicht werden, wenn nur eine geringe Anzahl an Merkmalen selektiert wird. Abbildung 5.8 zeigt die Verteilung der ausgewählten Merkmale in einer sogenannten Merkmalskarte für den ersten GM-Ansatz und die drei Merkmalssätze.

Es werden insbesondere die Merkmale aus der zweiten Hälfte der Mimiksequenz ausgewählt, hierbei ist zu beachten, dass die Cohn-Kanade-Datenbank die Mimik nur

	Merkmalssatz		
	$\mathbf{f}_t^o$	$\mathbf{f}_t^d$	$\mathbf{f}_t^c$
1. GM-Ansatz	$_{0,17}{}^{*}$72,81 %	$_{0,23}{}^{+}$84,89 %	$_{0,25}{}^{*}$86,71 %
2. GM-Ansatz	$_{0,32}{}^{*}$70,69 %	$_{0,20}{}^{*}$85,80 %	$_{0,15}{}^{*}$82,18 %

Tabelle 5.10: Ergebnisse zur Erkennung der sechs universellen Mimikemotionen mit den Daten aus der Cohn-Kanade-Datenbank. Ein Vorverarbeitungsschritt basierend auf der Kullback-Leibler-Divergenz wählt Merkmale aus. Hierbei werden über einen Schwellenwert die relevanten Merkmale bestimmt. Die relative Anzahl der selektierten Merkmale findet sich links unten. Es werden beide GM-Ansätze sowie die drei Merkmalssätze betrachtet.



(a) 1. GM-Ansatz: $\mathbf{f}_t^o$ (b) 1. GM-Ansatz: $\mathbf{f}_t^d$ (c) 1. GM-Ansatz: $\mathbf{f}_t^c$

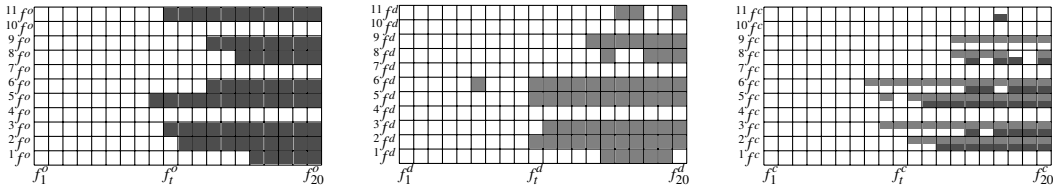
Abbildung 5.8: Merkmalskarten[a] für die drei Merkmalssätze – $\mathbf{f}_t^o$ (ausgewählte Merkmale sind dunkelgrau dargestellt), $\mathbf{f}_t^d$ (ausgewählte Merkmale sind hellgrau dargestellt), $\mathbf{f}_t^c$ (ausgewählte Merkmale sind dunkelgrau und hellgrau dargestellt) – des 1. GM-Ansatzes. Die Abszisse gibt den zeitlichen Verlauf an, die Ordinate beschreibt die Merkmale (siehe Tabelle 5.1).

[a] Die dargestellten Merkmalskarten zeigen nur eine Auswahl für einen Durchlauf aus der Fünffach-Kreuzvalidierung, zwischen den einzelnen Durchläufen kann es zu Unterschieden kommen.

bis zur zweiten Phase, dem sogenannten Halten (engl. *apex*), zeigt. Die ausgewählten Merkmale sind für alle drei Merkmalssätze ähnlich. Für alle drei Merkmalssätze werden folgende Merkmale (siehe Tabelle 5.1) ausgewählt: zwei (*Kiefer-Senker*), drei (*Lippen-Strecker*), fünf (*Lippen-Ecken-Drücker*), sechs (*Äußerer-Augenbrauen-Heber*), neun (*Nasen-Knitterer*) und elf (*Oberer-Lid-Heber*).

Ein ähnliches Bild zur Merkmalsselektion findet sich auch für den zweiten GM-Ansatz (siehe Abbildung 5.9). Hier werden auch die gleichen Merkmale (2,3,5,6,9) für alle drei Merkmalssätze ausgewählt, zusätzlich findet sich noch in allen drei Merkmalssätzen das achte Merkmal (*Lid-Verenger*). Des Weiteren werden, wie beim ersten GM-Ansatz, insbesondere die Merkmale aus der zweiten Hälfte der Mimiksequenz ausgewählt.

Es hat sich gezeigt, dass alle Verfahren zur Merkmalsselektion zu einer Verbesserung der Erkennungsrate beitragen, einige dieser Ansätze erreichen sogar signifikante bzw. hochsignifikante Verbesserungen. Die Besten der erzielten Ergebnisse liegen im



(a) 2. GM-Ansatz: $\mathbf{f}_t^o$ (b) 2. GM-Ansatz: $\mathbf{f}_t^d$ (c) 2. GM-Ansatz: $\mathbf{f}_t^c$

Abbildung 5.9: Merkmalskarten[a] für die drei Merkmalssätze – $\mathbf{f}_t^o$ (ausgewählte Merkmale sind dunkelgrau dargestellt), $\mathbf{f}_t^d$ (ausgewählte Merkmale sind hellgrau dargestellt), $\mathbf{f}_t^c$ (ausgewählte Merkmale sind dunkelgrau und hellgrau dargestellt) – des 2. GM-Ansatzes. Die Abszisse gibt den zeitlichen Verlauf an, die Ordinate beschreibt die Merkmale (siehe Tabelle 5.1).

[a]Die dargestellten Merkmalskarten zeigen nur eine Auswahl für einen Durchlauf aus der Fünffach-Kreuzvalidierung, zwischen den einzelnen Durchläufen kann es zu Unterschieden kommen.

Bereich der Arbeit von Mayer [177] (84,1 % Erkennungsrate), wo nur Bilder des ausgeprägtesten Mimikausdruckes zur Klassifikation verwendet worden sind.

5.4 Diskussion

In diesem Kapitel wurden zwei GM-Ansätze für die Erkennung der sechs universellen Emotionsgesichtsausdrücke (Freude, Traurigkeit, Wut, Ekel, Angst, Überraschung) vorgestellt. Diese Gesichtsausdrücke sind unabhängig von Alter, Geschlecht und Kulturkreis und können herangezogen werden, um in der MMI den emotionalen Zustand eines Menschen abzuschätzen. Basierend auf einem CANDIDE-3-Modell sind drei unterschiedliche Merkmalssätze generiert worden, um die Mimiksequenzen zu klassifizieren. Die erzielten Ergebnisse mit den GM-Ansätzen liegen im gleichen Bereich wie die SVM-basierten Ergebnisse der Arbeit von Mayer [177], falls alle Merkmale einer Sequenz berücksichtigt werden. Werden bei der SVM-basierten Klassifikation hingegen nur die Merkmale des ausgeprägtesten Mimikausdruckes zum Trainieren und Testen verwendet, fallen die vorgestellten GM-Ansätze ab. Aufgrund dieser Tatsache sind unterschiedliche Ansätze zur Merkmalsselektion verwendet worden, um die relevanten Merkmale zu bestimmen. Der erste Ansatz hat zwei Verfahren der sequentiellen Merkmalsselektion (SFS, SBS) verwendet, um die relevanten Merkmale zu bestimmen. Der zweite Ansatz basierte auf der Kullback-Leibler-Divergenz und wandte dieses Maß in zwei unterschiedlichen Verfahren an, um die relevanten Merkmale zu bestimmen. Mit den Verfahren zur Merkmalsselektion können die Ergebnisse verbessert werden und liegen im Bereich der Arbeit von Mayer [177], wenn nur die Bilder des ausgeprägtesten Mimikausdruckes zur Klassifikation verwendet worden sind.

Die Ergebnisse der Datenbank-Kreuzvalidierung haben gezeigt, dass es, trotz der Annahme von sechs universellen Emotionsgesichtsausdrücken, noch keine einheitlichen Mimikerkennungssysteme gibt. Es wird daher auch in Zukunft noch weitere Aktivitäten im Bereich der Mimikerkennung geben, um die Erkennungsleistung über mehrere Datenbanken hinweg auf einem ähnlichen Niveau zu halten, und somit anhand einer Erkennung unterschiedlicher Mimiksequenzen einen weiteren Beitrag für eine multimodale MMI zu liefern.

Kapitel 6

Graphische Modelle für bildbasierte Verfolgungsmethoden

Inhaltsangabe

In diesem Kapitel werden bildbasierte Verfolgungsmethoden vorgestellt, die mithilfe von GMs realisiert worden sind. Zunächst wird der CONDENSATION*-Algorithmus als GM dargestellt, ausgehend davon werden unterschiedliche Erweiterungen in den bildbasierten Verfolgungsprozess integriert. Im ersten Schritt wird die bildbasierte Verfolgung mit einem Klassifizierungsschritt kombiniert. Von diesem Verfahren ausgehend werden sowohl das Bewegungsmodell als auch die Gewichtungsfunktion der Partikel mithilfe der GM-Inferenz erweitert, um die zukünftigen Bearbeitungsschritte der gegenwärtigen Beobachtung anzupassen. In einem ganzheitlichen GM-Ansatz werden zuletzt die vorgestellten Verfahren für das adaptive Bewegungsmodell und für die adaptive Gewichtungsfunktion miteinander kombiniert. Die vorgestellten Verfahren werden auf einem verrauschten Gestendatensatz evaluiert.*

6.1 Einführung

6.1.1 Motivation

Die Verfolgung (engl. *Tracking*) von Personen bzw. Objekten ist von großer Bedeutung in der Bildverarbeitung, da es nach einem erfolgreichen Detektionsschritt rechnerisch meist effizienter ist, gesuchte Personen bzw. Objekte in den folgenden Videokamerabildern zu verfolgen, als für jedes neue Videokamerabild den Detektionsschritt zu wiederholen. Eine zuverlässige Verfolgung ist eine Grundvoraussetzung für viele verschiedene Bildverarbeitungsanwendungen, die von Personen- und Objektverfolgung bis zu Szenenüberwachung und -verständnis reichen, da durch den Einsatz von bildbasierten Verfolgungsmethoden Aspekte wie Echtzeitfähigkeit und Performancesteigerungen ermöglicht werden können. Dementsprechend gibt es viele Forschungsbemühungen, die sich mit der Verbesserung von bildbasierten Verfolgungsmethoden auseinandersetzen.

Im Bereich der AAL-Anwendungen spielt unter anderem die Verfolgung von Personen ein Rolle, da sich die älteren Menschen immer mehr zuhause aufhalten [20]. Hierbei kann die bildbasierte Verfolgung dazu beitragen, kritische und ungewöhnliche Situationen (z. B. Stürze) zu erkennen und gegebenenfalls entsprechende Maßnahmen einzuleiten, wie z. B. in [222], indem eine SMS verschickt wird. Neben bildbasierten Ansätzen zur Personenverfolgungen, siehe [223], können auch andere Sensortypen, wie z. B. Beschleunigungssensoren [224], genutzt werden, um die ältere Person zu verfolgen bzw. kritische Situationen zu erkennen.

Die Ansätze, die in diesem Kapitel vorgestellt werden, beschreiben bildbasierte Verfolgungsmethoden, die mittels GMs dargestellt und realisiert werden. Ein erster Schritt zur Kombination von GMs und der probabilistischen bildbasierten Verfolgung wurde in [225] eingeführt. Die Darstellung des Verfolgungsprozesses innerhalb einer

GM-Topologie bietet die Möglichkeit, dass die probabilistische bildbasierte Verfolgung mit Aspekten von inexakter und exakter Inferenz von GMs erweitert werden kann. Die vorgestellten Änderungen zielen auf die Einbeziehung eines Klassifizierungsschrittes innerhalb des Verfolgungsprozesses ab, sowie die Anpassung des Bewegungsmodells und der Partikel-Gewichtungsfunktion des Verfolgungsprozesses basierend auf dem GM-Inferenzprozess. Einige der vorgestellten Ansätze wurden teilweise in [226, 227] veröffentlicht.

6.1.2 Stand der Technik

Die Grundlage für probabilistische bildbasierte Verfolgungsalgorithmen stellt der *Conditional Density Propagation* (CONDENSATION)-Algorithmus, der in [228, 229, 230] veröffentlicht worden ist. Der Vorteil des CONDENSATION-Algorithmus gegenüber der probabilistischen bildbasierten Verfolgung, die auf einem Kalman-Filter [231] basiert, ist, dass auch nichtlineare, nicht gaußsche Prozesse modelliert werden können [229]. Auch Erweiterungen des Kalman-Filters – *Extended Kalman Filter* [232, 233], *Unscented Kalman Filter* [234, 235] – zielen auf einen besseren Umgang mit der Zustandsschätzung in nichtlinearen dynamischen Systemen ab.

Mehrere Möglichkeiten zur Verbesserung sowohl des Detektionsschrittes als auch des bildbasierten Verfolgungsprozesses von Personen bzw. Objekten in der Bildverarbeitung wurden präsentiert. Drei Kategorien (Partikel-Filter-Verfahren, Beobachtungsmodell, Bewegungsmodell) werden unten verwendet, um verschiedene Ansätze zur Verbesserung des probabilistischen bildbasierten Verfolgungsprozesses vorzustellen.

Unterschiedliche Arbeiten [236, 237, 238, 239, 240, 228, 241, 242] haben sich mit der Zustandsschätzung in nichtlinearen dynamischen Systemen beschäftigt, daher gibt es unterschiedliche Bezeichnungen für eine ähnliche Problemstellung: *Bootstrap Filter* [238], *Survival of the Fittest* [239], *Monte Carlo Filter* [240], CONDENSATION [228], *Particle Filter* [242]. Im Folgenden wird von *Partikel-Filter-Verfahren* gesprochen, wobei der CONDENSATION-Algorithmus eine spezielle Realisierung für die bildbasierte Verfolgung ist.

Grundlegende Details zu Partikel-Filter-Verfahren, wie *Perfect Monte Carlo Sampling*, *Importance Sampling*, *Sequential Importance Sampling* etc., können in [91, 92] gefunden werden. Zur Verbesserung des Partikel-Filter-Verfahrens wurden unterschiedliche Strategien entwickelt. Der sogenannte ICONDENSATION-Algorithmus [243] ist eine Erweiterung des CONDENSATION-Algorithmus um eine zusätzliche Informationsquelle, die auf globalen systemnahen (engl. *low-level*) Merkmalen basiert. Das *Factored Sampling* [244] des CONDENSATION-Algorithmus wird mit einem *Importance Sampling*-Verfahren [93] erweitert, das die globalen systemnahen Merkmale nutzt, um die Partikel des bildbasierten Verfolgungsprozesses besser zu verteilen. In [245, 246] wird der sogenannte *Auxiliary Particle Filter* vorgestellt. Bei diesem Filter wird eine zusätzliche Variable (engl. *auxiliary variable*) eingefügt, um in den

Resampling-Schritt die zukünftige Beobachtung miteinzubeziehen; dadurch soll die Effizienz der eingesetzten Partikel gesteigert werden. Für den Einsatz von Partikel-Filtern in hochdimensionalen Anwendungen wurde in [247] der *Annealed Particle Filter* vorgestellt. Dieser Filter wird eingesetzt, damit sich die verwendeten Partikel verstärkt auf das globale Maximum in einer approximierten WDF konzentrieren. Der *Rao-Blackwellised*-Partikel-Filter [248] benutzt die als *Rao-Blackwellisation* [249] bekannte Technik zur Effizienzsteigerung; dabei wird der Zustandsvektor des Objektes in zwei Teile aufgeteilt. Ein Teil umfasst die linearen (z. B. mit einem Kalman-Filter) beschreibbaren Aspekte des Zustandsvektors, während der zweite Teil die nichtlinearen Aspekte mithilfe eines Partikel-Filters beschreibt.

Trotz der Tatsache, dass die Verbesserung der Verteilung der Partikel einen positiven Effekt auf den bildbasierten Verfolgungsprozess von Objekten bzw. Personen haben kann, ist diese Verbesserung noch keine ideale Lösung, da die Dynamik und die Gestalt der Objekte bzw. Personen im Verfolgungsprozess noch kaum erfasst werden.

Im Gegensatz zu den Partikel-Filter-Verfahren sind im Bereich des Beobachtungsmodells verschiedene Verfahren vorgeschlagen worden, um die Detektion dem gesuchten Objekt anzupassen. In [250] wird die Verfolgung von mehreren Personen im Videokamerabild verwirklicht, indem ein Wechselspiel zwischen verschiedenen Arten von Detektoren durchgeführt wird: Es wird ein Fußgänger-Detektor für die allgemeine Personenerkennung verwendet, und ein personenspezifischer Detektor, der online trainiert wird. Mehr Robustheit im Verfolgungsprozess wird in [251] erreicht, indem Elemente aus dem Detektionsschritt und dem Verfolgungsprozess kombiniert werden. In [252] wird der Partikel-Filter um einen Energiefeld-Ansatz erweitert. Dieser nutzt die Bewegungsinformation im Bild, damit die Verteilung der Partikel angepasst werden kann. Ein adaptives Beobachtungsmodell, basierend auf einem *appearance model*, wird in [253] verwendet, um den bildbasierten Verfolgungsprozess zu verbessern. In diesem Ansatz wird auch die Anzahl der verwendeten Partikel variabel gestaltet, daneben kann auch das Bewegungsmodell adaptiert werden.

Das Bewegungsmodell ist ein entscheidender Punkt für den probabilistischen Verfolgungsprozess. Für die Verbesserung des Bewegungsmodells wurden verschiedene Ansätze vorgestellt: Verschiedene Zustände des Bewegungsmodells für den Verfolgungsprozess wurden in [254] verwendet. Hier wurde ein CONDENSATION-Algorithmus implementiert, der mehrere Gaußsche autoregressive Prozesse als Bewegungsmodell nutzt. Die Entscheidung, welcher Zustand für die Erzeugung der Partikel zur Verwendung kommt, wird mittels eines endlichen Automaten realisiert. Ein Partikel-Filter mit Speicherfunktion wurde in [255] vorgestellt. Dieser erweiterte Partikel-Filter nutzt die vorangegangenen geschätzten Zustände von Pose und Aussehen der verfolgten Objekte, um eine Wahrscheinlichkeitsverteilung für diese Zustände zu berechnen. Ausgehend von den vorangegangenen Zuständen wird eine Wahrscheinlichkeit bestimmt, die beschreibt, inwieweit vergangene Zustände wieder auftreten.

6.2 Probabilistische bildbasierte Verfolgung

Die Grundlagen für die probabilistische bildbasierte Verfolgung von Objekten bzw. Personen[1] wurden in der Arbeit von Isard und Blake [228, 229] mittels des CONDENSATION-Algorithmus gelegt. Besonders bei störenden Faktoren wie Rauschen, Lichtänderungen etc. ist der CONDENSATION-Algorithmus eine robuste Methode, um Objekte in Videosequenzen verfolgen zu können. Der CONDENSATION-Algorithmus lässt sich in zwei Schritte einteilen: Im ersten Schritt, wenn das Objekt im aktuellen Videokamerabild gesucht werden soll, versucht der Algorithmus die Bewegung des Objektes vorherzusagen. Basierend auf den geschätzten Positionen im aktuellen Bild aus dem ersten Schritt versucht der Algorithmus im zweiten Schritt das gesuchte Objekt zu finden. Der Algorithmus wiederholt beide Schritte nacheinander, um somit das Objekt über die gesamte Videosequenz zu verfolgen.

In der Praxis sind jedoch weder die Vorhersage der Bewegung (erster Schritt) noch die Suche des Objektes (zweiter Schritt) vollkommen richtig, daher erhält man nur eine WDF darüber, ob das Objekt sich an einer bestimmten Position befindet oder nicht. Um diese probabilistische Beschreibung der Objektposition auf ein berechenbares Maß zu reduzieren, verwendet der CONDENSATION-Algorithmus eine Partikel-Filter-Methode. Hierbei beschreiben sogenannte Partikel einzelne Proben der approximierten WDF der Objektposition. Somit stellt der CONDENSATION-Algorithmus eine bestimmte Form eines Partikel-Filters dar.

Im Folgenden werden kurz basierend auf [228, 229] die wesentlichen Schritte für die probabilistische bildbasierte Verfolgung vorgestellt.

6.2.1 Beobachtungsmodell

In den meisten Fällen liefert eine Videokamera innerhalb von diskreten Zeitschritten $\tau \in [1,\ldots,T]$ eine Sequenz von 2-D-Bildern $\mathcal{I}_T = \{\mathbf{I}_1,\ldots,\mathbf{I}_T\}$, die das zu verfolgende Objekt enthalten. Die aus dem Videokamerabild gewonnene Beobachtung wird meist mittels eines Merkmalsvektors $\mathbf{z}_\tau$ beschrieben und nicht direkt durch das Bild $\mathbf{I}_\tau$ selbst. Das Problem liegt darin, dass die Beobachtung auf einer Messung beruht und damit der Objektzustand geschätzt werden muss. Das gesuchte Objekt wird mittels eines Zustandsvektors $\mathbf{x}_\tau$ beschrieben. Dieser Vektor umfasst Informationen wie Objektposition sowie Parameter, die das Objekt beschreiben (Histogramm, Form etc.). Die beobachtete Merkmalsvektorsequenz $\mathcal{Z}_T = \{\mathbf{z}_1,\ldots,\mathbf{z}_T\}$ steht in einem Bezug zur Zustandsvektorsequenz $\mathcal{X}_T = \{\mathbf{x}_1,\ldots,\mathbf{x}_T\}$, aber beschreibt diese, unter anderem aufgrund des Messcharakters, nicht direkt.

Die Beziehung zwischen dem Merkmalsvektor $\mathbf{z}_\tau$ und dem Zustandsvektor $\mathbf{x}_\tau$ gilt

[1] Im Folgenden wird der Einfachheit halber nur noch von Objekten gesprochen, hierbei kann es sich aber auch um Personen handeln.

nur für das jeweilige Bild $\mathbf{I}_\tau$, $\tau \in [1,\ldots,T]$. Es wird angenommen, dass die Beobachtungen untereinander unabhängig sind und nicht von vorangegangenen Zustandsvektoren abhängen. Dieser Sachverhalt lässt sich folgendermaßen zusammenfassen:

$$p(\mathcal{Z}_t|\mathcal{X}_t) = \prod_{\tau=1}^{t} p(\mathbf{z}_\tau|\mathbf{x}_\tau). \tag{6.1}$$

6.2.2 Bewegungsmodell

Das gesuchte Objekt bewegt sich durch die Bildsequenz $\mathcal{I}_T$, diese Bewegung kann mithilfe eines Bewegungsmodells beschrieben werden. Die spezifische Information über das Bewegungsmodell ist in der Zustandsübergangswahrscheinlichkeit $p(\mathbf{x}_t|\mathcal{X}_{t-1})$ enthalten, in der die Zustandsvektorsequenz $\mathcal{X}_{t-1}$ aus den vorherigen Bildern $\mathcal{I}_{t-1}$ zum nächsten Zustandsvektor $\mathbf{x}_t$ für das Bild $\mathbf{I}_t$ übergeht.

Im CONDENSATION-Algorithmus wird die Annahme getroffen, dass der aktuelle Zustand des Objektes $\mathbf{x}_t$ nur von seinem unmittelbaren zeitlichen Vorgänger $\mathbf{x}_{t-1}$ abhängt und nicht von weiter vorangegangenen Zuständen $\mathcal{X}_{t-2} = \{\mathbf{x}_1,\ldots,\mathbf{x}_{t-2}\}$ (Markov-Kette erster Ordnung):

$$p(\mathbf{x}_t|\mathcal{X}_{t-1}) = p(\mathbf{x}_t|\mathbf{x}_{t-1}). \tag{6.2}$$

Diese Annahme vereinfacht die Beschreibung des Bewegungsmodells, dennoch gibt es viele Ansätze für die Beschreibung unterschiedlicher Bewegungsmodelle. Ein einfaches Bewegungsmodell kann mittels eines stochastischen Zufallsprozesses realisiert werden, daneben gibt es auch Möglichkeiten, wie beim CONDENSATION-Algorithmus von Isard und Blake, einen autoregressiven Prozess als Modell zu verwenden.

6.2.3 Realisierung der probabilistischen bildbasierten Verfolgung

Das Ziel der probabilistischen bildbasierten Verfolgung von Objekten besteht darin, in jedem Videokamerabild $\mathbf{I}_t$ den aktuellen Zustandsvektor $\mathbf{x}_t$ des gesuchten Objektes zu finden. Die Informationen, die bis zum gegenwärtigen Zeitpunkt t zur Verfügung stehen, sind die Beobachtungen $\mathcal{Z}_t$. Die bedingte WDF $p(\mathbf{x}_t|\mathbf{z}_t)$ beschreibt, wie man aus der Information des aktuellen Videokamerabilds versucht, das Objekt zu finden. Dieser Ansatz kann mittels Detektionsverfahren verwirklicht werden, dennoch werden hierbei die vorangegangenen Informationen $\mathcal{Z}_{t-1}$ außer Acht gelassen. Im probabilistischen Verfolgungsprozess versucht man diese Informationen zu berücksichtigen, und wertet daher die bedingte WDF $p(\mathbf{x}_t|\mathcal{Z}_t)$ zur Bestimmung des gesuchten Objektes aus. Im Folgenden werden die zwei rekursiven Schritte (*Messen* und *Prädiktion*) vorgestellt, die bei der Berechnung der bedingten WDF $p(\mathbf{x}_t|\mathcal{Z}_t)$ helfen.

6.2.3.1 Messschritt

Der Messschritt beinhaltet zwei Annahmen: Die Beobachtungen $\mathcal{Z}_t$ sind voneinander unabhängig sowie unabhängig vom Bewegungsmodell, das durch die bedingte Wahrscheinlichkeit $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ beschrieben wird. Beide Annahmen werden in Gleichung 6.1 zusammengefasst. Wenn die bedingte Wahrscheinlichkeit $p(\mathbf{x}_{t-1}|\mathcal{Z}_{t-1})$ für das Videokamerabild $\mathbf{I}_{t-1}$ bestimmt worden ist, besteht der nächste Schritt für das Videokamerabild $\mathbf{I}_t$ darin, die bedingte Wahrscheinlichkeit $p(\mathbf{x}_t|\mathcal{Z}_t)$ zu berechnen. Die gegenwärtige bedingte Wahrscheinlichkeit $p(\mathbf{x}_t|\mathcal{Z}_t)$ kann wie folgt als

$$
\begin{aligned}
p(\mathbf{x}_t|\mathcal{Z}_t) &= p(\mathbf{x}_t|\mathbf{z}_t, \mathcal{Z}_{t-1}) && (6.3)\\
&= \frac{p(\mathbf{x}_t, \mathbf{z}_t|\mathcal{Z}_{t-1})}{p(\mathbf{z}_t|\mathcal{Z}_{t-1})} && (6.4)\\
&= \frac{p(\mathbf{z}_t|\mathbf{x}_t, \mathcal{Z}_{t-1})p(\mathbf{x}_t|\mathcal{Z}_{t-1})}{p(\mathbf{z}_t|\mathcal{Z}_{t-1})} && (6.5)\\
&= k_t p(\mathbf{z}_t|\mathbf{x}_t, \mathcal{Z}_{t-1})p(\mathbf{x}_t|\mathcal{Z}_{t-1}) && (6.6)
\end{aligned}
$$

beschrieben werden, wobei $k_t = \frac{1}{p(\mathbf{z}_t|\mathcal{Z}_{t-1})}$ ein Faktor ist, der unabhängig vom Zustandsvektor $\mathbf{x}_t$ ist.

Unter Berücksichtigung der gegenseitigen Unabhängigkeit zwischen den Beobachtungen (siehe Gleichung 6.1) kann die Gleichung 6.6 folgendermaßen als

$$
p(\mathbf{x}_t|\mathcal{Z}_t) = k_t p(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathcal{Z}_{t-1}) \tag{6.7}
$$

ausgedrückt werden.

Diese Gleichung 6.7 kann in zwei WDFs aufgeteilt werden. Die bedingte WDF $p(\mathbf{x}_t|\mathcal{Z}_{t-1})$ prädiziert den Zustandsvektor $\mathbf{x}_t$ im gegenwärtigen Videokamerabild $\mathbf{I}_t$ unter Berücksichtigung der Informationen der vorangegangenen Beobachtungen $\mathcal{Z}_{t-1}$, die aus den Bildern $\mathcal{I}_{t-1}$ stammen. Die bedingte WDF misst $p(\mathbf{z}_t|\mathbf{x}_t)$ die Validität der Prädiktion, indem sie die gegenwärtige Beobachtung $\mathbf{z}_t$ berücksichtigt.

6.2.3.2 Prädiktionsschritt

Die Prädiktion der bedingten WDF $p(\mathbf{x}_t|\mathcal{Z}_{t-1})$ kann mittels Marginalisierung von allen vorangegangenen Zustandsvektoren $\mathcal{X}_{t-1}$ folgendermaßen ausgedrückt werden:

$$
\begin{aligned}
p(\mathbf{x}_t|\mathcal{Z}_{t-1}) &= \int_{\mathcal{X}_{t-1}} p(\mathbf{x}_t, \mathcal{X}_{t-1}|\mathcal{Z}_{t-1})\, d\mathcal{X}_{t-1} && (6.8)\\
&= \int_{\mathcal{X}_{t-1}} p(\mathbf{x}_t|\mathcal{X}_{t-1}, \mathcal{Z}_{t-1})p(\mathcal{X}_{t-1}|\mathcal{Z}_{t-1})\, d\mathcal{X}_{t-1} && (6.9)
\end{aligned}
$$

Berücksichtigt man nun die Markov-Eigenschaft von Gleichung 6.2 und die Tatsache, dass die Beobachtung bei gegebenem Zustandsvektor keine zusätzliche Information liefert, dann kann die Gleichung 6.9 folgendermaßen formuliert werden:

$$p(\mathbf{x}_t|\mathcal{Z}_{t-1}) = \int_{\mathcal{X}_{t-1}} p(\mathbf{x}_t|\mathbf{x}_{t-1}) p(\mathcal{X}_{t-1}|\mathcal{Z}_{t-1})\, d\mathcal{X}_{t-1} \tag{6.10}$$

$$= \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t|\mathbf{x}_{t-1}) p(\mathbf{x}_{t-1}|\mathcal{Z}_{t-1})\, d\mathbf{x}_{t-1}. \tag{6.11}$$

6.2.3.3 Rekursive Verfolgung (Propagation)

Die bedingte WDF $p(\mathbf{x}_t|\mathcal{Z}_t)$ des Zustandsvektors kann rekursiv von der vorangegangenen WDF $p(\mathbf{x}_{t-1}|\mathcal{Z}_{t-1})$ mittels der zwei Schritte *Prädiktion* und *Messung* bestimmt werden.

Prädiktionsschritt: In Gleichung 6.11 wird eine Prädiktion für den gegenwärtigen Zustandsvektor $\mathbf{x}_t$ mit Berücksichtigung der vorangegangenen Beobachtungen $\mathcal{Z}_{t-1}$, aber ohne die aktuelle Beobachtung $\mathbf{z}_t$, durchgeführt. Die bedingte WDF $p(\mathbf{x}_t|\mathcal{Z}_{t-1})$ wird daher A-priori-WDF genannt.

Messschritt: In Gleichung 6.7 wird die Qualität der vorherigen Prädiktion bewertet, indem man die aktuelle Beobachtung $\mathbf{z}_t$ berücksichtigt. Aufgrund dieser neuen Information wird die bedingte WDF $p(\mathbf{x}_t|\mathcal{Z}_t)$ A-posteriori-WDF genannt.

Prädiktions- und Messschritt beschreiben, wie in einer Bildersequenz ein Objekt rekursiv verfolgt werden kann. Die folgende Gleichung 6.12 beschreibt diesen Sachverhalt:

$$p(\mathbf{x}_t|\mathcal{Z}_t) = k_t p(\mathbf{z}_t|\mathbf{x}_t) \int_{\mathbf{x}_{t-1}} p(\mathbf{x}_t|\mathbf{x}_{t-1}) p(\mathbf{x}_{t-1}|\mathcal{Z}_{t-1})\, d\mathbf{x}_{t-1}. \tag{6.12}$$

In dieser Gleichung wird die WDF des Zustandsvektors über die Zeit verfolgt. Diese WDF kann von den WDFs der vorangegangenen Zustände mittels zweier Funktionen – $p(\mathbf{x}_t|\mathbf{x}_{t-1})$, $p(\mathbf{z}_t|\mathbf{x}_t)$ – bestimmt werden.

Die bedingte WDF $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ drückt die Abhängigkeit des aktuellen Zustandsvektors von seinem vorhergehenden Zustandsvektor aus, das heißt die Änderung des Zustandsvektors über die Zeit. Aus diesem Grund kann diese bedingte WDF als Bewegungsmodell des Objektes interpretiert werden.

Die bedingte WDF $p(\mathbf{z}_t|\mathbf{x}_t)$ beschreibt die Wahrscheinlichkeit, dass das gesuchte Objekt $\mathbf{x}_t$ in der Beobachtung $\mathbf{z}_t$ gefunden werden kann. Mittels dieser bedingten WDF $p(\mathbf{z}_t|\mathbf{x}_t)$ wird auch die vorangegangene Prädiktion bewertet.

6.2.4 Partikel-Filter

Das Integral in Gleichung 6.11 – Integration über alle möglichen Zustände des vorangegangenen Zustandsvektors $\mathbf{x}_{t-1}$ – behindert die rechnerische Durchführbarkeit des

probabilistischen bildbasierten Verfolgungsprozesses. Daher bedarf es einer Approximation der bedingten WDF $p(\mathbf{x}_t|\mathcal{Z}_{t-1})$ durch eine Summe über eine endliche Menge von sogenannten diskreten Partikeln $\mathbf{s}_t^{(n)}$ anstelle eines Integrals.

Dieses Verfahren wird als Sequentielle Monte-Carlo-Methode oder auch Partikel-Filter bezeichnet [91, 93]. Mehrere verschiedene Techniken stehen zur Realisierung von Partikel-Filtern zur Verfügung. Der CONDENSATION-Algorithmus verwendet das sogenannte *Factored Sampling* [244] für die Verwaltung der Partikel. Es werden dabei folgende Schritte umgesetzt:

Die A-posteriori-WDF $p(\mathbf{x}_{t-1}|\mathcal{Z}_{t-1})$ vom vorangegangenen Zeitschritt $t-1$ wird mittels einer ungewichteten Partikelmenge $\mathcal{S}_{t-1}^N$ mit N-Partikeln $\{\mathbf{s}_{t-1}^{(1)},\ldots,\mathbf{s}_{t-1}^{(N)}\}$ als folgende Funktion $P_N(\mathbf{x}_{t-1}|\mathcal{Z}_{t-1})$ approximiert. $P_N(\mathbf{x}_t|\mathcal{Z}_{t-1})$ approximiert für den Zeitschritt t die aktuelle A-priori-WDF $p(\mathbf{x}_t|\mathcal{Z}_{t-1})$, nach Gleichung 6.11 auch in Form von ungewichteten Partikeln $\mathcal{S}'^N_t$. Der Unterschied zwischen der Partikelmenge $\mathcal{S}_{t-1}^N$ und der Partikelmenge $\mathcal{S}'^N_t$ liegt in einem Prädiktionsschritt, der durch den ersten Term von Gleichung 6.11 gebildet wird; hierbei werden die Partikel aufgrund eines Bewegungsmodells an neue Positionen gesetzt.

Im Messschritt von Gleichung 6.7 werden die ungewichteten Partikel mit $p(\mathbf{z}_t|\mathbf{x}_t)$ multipliziert, was einer Gewichtung (siehe *importance sampling* [91]) mit

$$\begin{aligned}
\tilde{P}_N(\mathbf{x}_t|\mathcal{Z}_t) &= k_t p(\mathbf{z}_t|\mathbf{x}_t) P_N(\mathbf{x}_t|\mathcal{Z}_{t-1}) && (6.13)\\
&= \sum_{n=1}^{N} (k_t p(\mathbf{z}_t|\mathbf{x}_t)\frac{1}{N})\delta(\mathbf{x}_t, \mathbf{s'}_t^{(n)}) && (6.14)\\
&= \sum_{n=1}^{N} \pi_t^{(n)} \delta(\mathbf{x}_t, \mathbf{s'}_t^{(n)}), && (6.15)
\end{aligned}$$

entspricht, wobei die Gewichtungsfunktion definiert ist als

$$\begin{aligned}
\pi_t^{(n)} &:= \tilde{w}_t(\mathbf{s'}_t^{(n)}) = k_t p(\mathbf{z}_t|\mathbf{s'}_t^{(n)})\tfrac{1}{N}\\
&\propto p(\mathbf{z}_t|\mathbf{s'}_t^{(n)}).
\end{aligned}$$

Die A-posteriori Verteilung $P_N(\mathbf{x}_{t-1}|\mathcal{Z}_{t-1})$ aus dem Zeitschritt $t-1$ ist mit ungewichteten Partikeln repräsentiert, während hingegen die gegenwärtige A-posteriori Verteilung $\tilde{P}_N(\mathbf{x}_t|\mathcal{Z}_t)$ für Zeitschritt t aus gewichteten Partikeln besteht. Um für den aktuellen Zeitschritt t auch ungewichtete Partikel zu erhalten, werden im letzten Schritt des *Factored Sampling*-Verfahrens die Partikel neu verteilt (*resampling*). Zwischen jedem Messschritt aus dem vorangegangenen Zeitschritt $t-1$ und dem Prädiktionsschritt im Zeitschritt t werden die Partikel unter Berücksichtigung ihres Gewichts $\pi_t^{(n)}$ neu verteilt. Das heißt, die neuen Partikel $\mathbf{s}_t^{(n)} \in \mathcal{S}_t^N$ werden *mit Zurücklegen* aus der gewichteten Partikelmenge $\mathcal{S}'^N_t$ gezogen, wobei jeder Partikel $\mathbf{s'}_t^{(n)}$ mit seiner Wahrscheinlichkeit $\pi_t^{(n)}$ gezogen werden kann.

6.2.5 Der CONDENSATION-Algorithmus

Innerhalb des CONDENSATION-Algorithmus werden die Schritte – Prädiktion, Messen und *resampling* – nacheinander für jedes Videokamerabild $\mathbf{I}_t$ wiederholt, bis das letzte Bild $\mathbf{I}_T$ erreicht ist.

Der rekursive Prozess benötigt für die Bestimmung der aktuellen WDF $p(\mathbf{x}_t|\mathcal{Z}_t)$ die vorangegangene A-posteriori-WDF $p(\mathbf{x}_{t-1}|\mathcal{Z}_{t-1})$, die mit einer ungewichteten Partikelmenge $\mathbf{s}_{t-1}^{(n)} \in \mathcal{S}_{t-1}^N$ approximiert ist.

Für die Initialisierung, das heißt das erste Bild $\mathbf{I}_{t=1}$, kann die WDF $p(\mathbf{x}_{t=1}|z_{t=1})$ mittels eines Detektionsschrittes bestimmt werden. Nach diesem Schritt folgt eine entsprechende Verteilung der anfänglichen Partikelmenge $\mathbf{s}_{t=1}^{(n)} \in \mathcal{S}_{t=1}^N$.

In einem Rekursionsschritt t mit einer bestimmten Partikelmenge $\mathcal{S}_{t-1}^N$ vom vorangegangenen Zeitschritt $t-1$ werden die folgenden drei Schritte ausgeführt:

1. Prädiktion: Für den Prädiktionsschritt von Gleichung 6.11 ist eine Integration über den vorangegangenen Zustandsvektor $\mathbf{x}_{t-1}$ notwendig. Dieses Integral wird mittels einer Partikelmenge $\mathcal{S}_{t-1}^N$ approximiert. Die bedingte WDF $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ kann als Bewegungsmodell interpretiert werden. Dieses Modell wird auf jeden Partikel $\mathbf{s}_{t-1}^{(n)}$ mittels zwei separater Bewegungen (Drift, Diffusion) realisiert:

   (a) Drift ist die deterministische Komponente des Bewegungsmodells, hierbei wird jeder Partikel in eine bestimmte Richtung bewegt.

   (b) Diffusion ist die nichtdeterministische Komponente des Bewegungsmodells, hierbei erhält jeder Partikel noch eine zusätzliche zufällige Bewegung.

   Das Ergebnis nach dem Prädiktionsschritt ist eine Partikelmenge $\mathbf{s'}_t^{(n)}$.

2. Messen: Beim Messschritt wird die vorangegangene A-priori-WDF mit der bedingten WDF $p(\mathbf{z}_t|\mathbf{x}_t)$ multipliziert, die angibt, inwieweit die Beobachtung dem Zustandsvektor ähnelt.

   In der praktischen Umsetzung wird dies durch eine Gewichtung realisiert, wobei jeder Partikel $\mathbf{s'}_t^{(n)}$ mit einem Faktor $\pi_t^{(n)}$ gewichtet wird. Dieser Faktor wird durch eine Messung der Beobachtung bestimmt.

3. *Resampling*: Die Partikel $\mathbf{s'}_t^{(n)}$ repräsentieren, in ihrer ungewichteten Form, die bedingte WDF $p(\mathbf{x}_t|\mathcal{Z}_{t-1})$. Die bedingte A-posteriori-WDF $p(\mathbf{x}_t|\mathcal{Z}_t)$ wird approximiert, indem man die zugehörigen Gewichte $\pi_t^{(n)}$ berücksichtigt, was sich einer gewichteten Partikelmenge niederschlägt. Um eine ungewichtete Partikelmenge $\mathcal{S}_t$ für den nächsten Iterationsschritt zu erhalten, werden aus der gewichteten Partikelmenge $\mathcal{S}'^N_t$ anhand der Gewichtung der Partikel die neuen ungewichteten Partikel $\mathbf{s}_t^{(n)} \in \mathcal{S}_t^N$ *mit Zurücklegen* gezogen.

6.3 GM-basierte probabilistische bildbasierte Verfolgung

Im Folgenden werden Ansätze für die Anpassung sowie Verbesserung des klassischen CONDENSATION-Algorithmus vorgestellt, diese Ansätze beruhen auf der Verwendung von GMs. Die Änderungen zielen auf folgende Gebiete ab: Integration eines Klassifizierungsschrittes in den bildbasierten Verfolgungsprozess, Anpassung des Bewegungsmodells sowie Anpassung der Gewichtungsfunktion der Partikel. Die Anpassungen werden mittels exakter und inexakter Inferenz in den GM-Topologien verwirklicht, um das Bewegungsmodell sowie die Gewichtungsfunktion im Hinblick auf die aktuelle Beobachtungssequenz $\mathcal{Z}_t$ anzupassen.

6.3.1 Der CONDENSATION-Algorithmus als GM

Der CONDENSATION-Algorithmus kann als GM dargestellt werden, siehe Abbildung 6.1. Diese Darstellung bildet die Grundlage für die Änderungen und Verbesserungen, die in den folgenden Abschnitten vorgestellt werden. Diese Abbildung ist sehr allgemein und enthält noch keine weiteren Einzelheiten über die Ausgestaltung des Bewegungsmodells $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ sowie der Gewichtungsfunktion der Partikel $\pi_t^i \sim p(\mathbf{z}_t|s_t^{(i)})$.

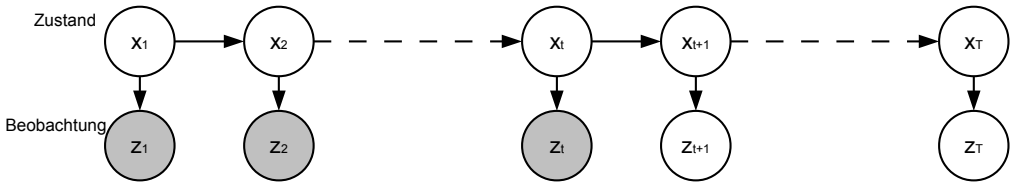


Abbildung 6.1: GM-Darstellung des CONDENSATION-Algorithmus

6.3.2 Kombinierte Klassifikation und bildbasierte Verfolgung

Ein erster Ansatz zur Erweiterung des bildbasierten Verfolgungsprozesses mit GM-Inferenz ist der Einbau eines Klassifikationsschrittes, der unterschiedliche Gesten erkennen kann. Für die Realisierung dieser Kombination wird der Verfolgungsprozess mit einem Klassifikationsschritt, basierend auf einem DBN, in einem GM fusioniert.

Für die Kombination des bildbasierten Verfolgungsprozesses mit einem Klassifikationsschritt in einer GM-Topologie wird davon ausgegangen, dass die Bewegung – oder im Allgemeinen – das Muster in der Beobachtungssequenz $\mathcal{Z}_T = \{z_1, \ldots, z_T\}$ einem bestimmten Muster der Klasse c aus einer möglichen Klassenmenge $\mathcal{C}$ angehört. T ist die Anzahl der diskreten Zeitschritte, die die zeitliche Dauer eines Musters beschreibt. Das

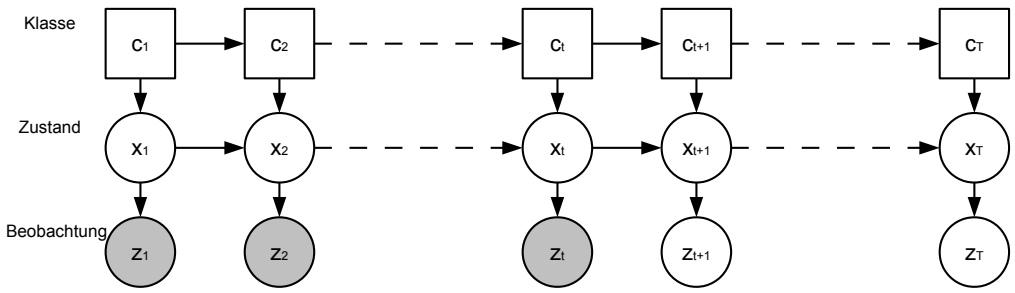


Abbildung 6.2: GM-Darstellung des Ansatzes zur Kombination des Verfolgungsprozesses mit einer DBN-basierten Klassifikation

Ziel der GM-basierten Fusion – die Kombination des bildbasierten Verfolgungsprozesses mit einem Klassifikationsschritt – ist die Bestimmung der wahrscheinlichsten Klasse $\hat{c}$ für die Beobachtungssequenz $\mathcal{Z}_\tau$, $\tau \in [2,\ldots,T]$.

Das gesuchte Muster, beschrieben durch die aktuelle Beobachtungsequenz $\mathcal{Z}_t$, gehört zur Klasse mit der höchsten inferierten Wahrscheinlichkeit:

$$\hat{c} = \arg\max_c p(c|\mathcal{Z}_t). \tag{6.16}$$

Das Beobachtungsmodell, das im CONDENSATION-Algorithmus verwendet wird, nimmt an, dass die gegenwärtige Beobachtungssequenz $\mathcal{Z}_t$ von einer verborgenen Zustandsvektorsequenz $\mathcal{X}_t$ erzeugt wird (generativer Ansatz [256]). Daher beschreibt die Zustandsvektorsequenz $\mathcal{X}_t$ alle relevanten Eigenschaften des Musters, in diesem Fall die Bewegung. Eine Marginalisierung in der Gleichung 6.16 über der Menge aller möglichen Zustände $\mathcal{X}_t$ ergibt

$$\begin{aligned} p(c|\mathcal{Z}_t) &= \int_{\mathcal{X}_t} p(c,\mathcal{X}_t|\mathcal{Z}_t)\,d\mathcal{X}_t && (6.17)\\ &= \int_{\mathcal{X}_t} p(c|\mathcal{X}_t,\mathcal{Z}_t)p(\mathcal{X}_t|\mathcal{Z}_t)\,d\mathcal{X}_t && (6.18)\\ &= \int_{\mathcal{X}_t} p(c|\mathcal{X}_t)p(\mathcal{X}_t|\mathcal{Z}_t)\,d\mathcal{X}_t, && (6.19) \end{aligned}$$

da nach dem Beobachtungsmodell $\mathcal{Z}_t$ keine zusätzlichen Informationen enthält, wenn $\mathcal{X}_t$ gegeben ist.

Daher ist die gesuchte Klasse $\hat{c}$, die die beobachtete Bewegung am wahrscheinlichsten beschreibt, jene Klasse c, die obigen Term von Gleichung 6.19 maximiert:

$$\hat{c} = \arg\max_c \int_{\mathcal{X}_t} p(c|\mathcal{X}_t)p(\mathcal{X}_t|\mathcal{Z}_t)\,d\mathcal{X}_t. \tag{6.20}$$

Zur Umsetzung eines bildbasierten Verfolgungsprozesses des beobachteten Musters (hier eine Geste) kann die bedingte WDF des Zustandsvektors $p(\mathcal{X}_t|\mathcal{Z}_t)$ in einer rekursiven Art und Weise für jedes Bild $\mathbf{I}_t$ ausgedrückt werden. Das Konzept des

CONDENSATION-Algorithmus (siehe Abschnitt 6.2.3) zur Verfolgung des Musters mit zwei Schritten – Prädiktionsschritt, Messschritt – bleibt erhalten: Zunächst wird die A-priori-WDF $p(\mathcal{X}_t|\mathcal{Z}_{t-1})$ von den vorangegangenen Bildern $\mathcal{I}_{t-1}$ geschätzt. Dann wird die geschätzte WDF mit der gegenwärtigen Beobachtung $\mathbf{z}_t$ bewertet, um eine A-posteriori-WDF $p(\mathcal{X}_t|\mathcal{Z}_t)$ zu erhalten.

Der wichtige Unterschied zwischen dem klassischen CONDENSATION-Algorithmus und dem hier vorgestellten Ansatz ist der Verlust der Markov-Eigenschaft innerhalb der Zustandsvektorsequenz $\mathcal{X}_t$, da für die Klassifizierung eines Bewegungsmusters (bzw. einer Geste) die gesamte Zustandsvektorsequenz notwendig ist. Dennoch kann das Bewegungsmodell weiterhin mit einem Modell (siehe Abschnitt 6.2.2) beschrieben werden, das die Bewegung mit einer Markov-Kette erster Ordnung modelliert.

Unter Berücksichtigung von Gleichung 6.12 kann das Objekt weiterhin rekursiv in einer Bildersequenz $\mathcal{I}_T$ verfolgt werden. Dabei bleibt die Markov-Kette erster Ordnung von Gleichung 6.2 erhalten, somit lässt sich der Klassifikationsschritt für den gegenwärtigen Zeitpunkt t folgendermaßen beschreiben:

$$\hat{c} = \arg\max_{c} \; k_t \int_{\mathcal{X}_t} p(c|\mathcal{X}_t)\, p(\mathbf{z}_t|\mathbf{x}_t)\, p(\mathcal{X}_{t-1}|\mathcal{Z}_{t-1})\, p(\mathbf{x}_t|\mathbf{x}_{t-1})\, d\mathcal{X}_t. \tag{6.21}$$

Gleichung 6.21 beschreibt die Fusionierung eines bildbasierten Verfolgungsprozesses, der mittels des CONDENSATION-Algorithmus realisiert wird, mit einer DBN-basierten Klassifikation in einer einheitlichen GM-Darstellung. Die wahrscheinlichste Klasse $\hat{c}$ kann für jeden Zeitschritt t bestimmt werden.

Das Interessante an Gleichung 6.21 ist, dass die Informationen des GM-Inferenzprozesses genutzt werden können, um Anpassungen im Verarbeitungsprozess für das zukünftige Bild $\mathbf{I}_{t+1}$ zu machen. Mehrere Möglichkeiten sind denkbar, um den Verarbeitungsprozess anzupassen.

In Abschnitt 6.3.3 wird das Bewegungsmodell, repräsentiert in der bedingten WDF $p(\mathbf{x}_{t+1}|\mathbf{x}_t)$, mittels eines GM-Inferenzprozesses aufgrund der gegenwärtigen Beobachtungssequenz $\mathcal{Z}_t$ angepasst (adaptives Bewegungsmodell). In Abschnitt 6.3.4 wird die Gewichtungsfunktion der Partikel, repräsentiert in der bedingten WDF $\pi^i_{t+1} \sim p(\mathbf{z}_{t+1}|s^{(i)}_{t+1})$, mittels eines GM-Inferenzprozesses aufgrund der gegenwärtigen Beobachtungssequenz $\mathcal{Z}_t$ angepasst (adaptive Gewichtungsfunktion). Abschließend werden beide Ansätze – adaptives Bewegungsmodell, adaptive Gewichtungsfunktion – im Abschnitt 6.3.5 in einer ganzheitlichen GM-Darstellung fusioniert.

6.3.3 Adaptives Bewegungsmodell

Die bedingte WDF $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ dient als Schätzung für die Objektbewegung, diese Bewegung kann mittels eines stochastischen Zufallsprozesses realisiert werden; daneben

gibt es auch Möglichkeiten wie beim CONDENSATION-Algorithmus einen autoregressiven Prozess als Modell zu verwenden. Die Informationen über die Bewegungsmuster von den verschiedenen Klassen $c \in \mathcal{C}$, die im GM zur Erkennung der jeweiligen Klasse abgespeichert sind, können genutzt werden, um für den zukünftigen Zeitschritt $t+1$ die Platzierung und Verteilung der Partikel $\mathbf{s}_{t+1}^{(n)}$ der gegenwärtigen Beobachtungssequenz $\mathcal{Z}_t$ anzupassen. Der in diesem Abschnitt vorgestellte Algorithmus hat folgenden grundlegenden Arbeitsablauf: In jedem Zeitschritt t wird für jeden Partikel $\mathbf{s}_t^{(n)}$ die Wahrscheinlichkeitsfunktion für die Bewegungsklassen $\mathcal{C}$ bestimmt. Basierend darauf wird, mittels GM-Inferenz, eine Schätzung der zukünftigen Bewegung gegeben, um somit die Partikel zu verteilen. Die anderen Schritte (Messen, *Sampling* und *Resampling*) beruhen weiterhin auf dem CONDENSATION-Algorithmus.

6.3.3.1 Das Graphische Modell für das adaptive Bewegungsmodell

Der Prozess zur Erzeugung der beobachteten Merkmalsvektorsequenz $\mathcal{Z}_t$ einer Bewegung kann durch das GM von Abbildung 6.3 dargestellt werden. Ein wesentlicher Unterschied zwischen diesem GM und dem GM von Abschnitt 6.3.2 ist die Tatsache, dass hierbei in der Zustandsvektorsequenz $\mathcal{X}_t^m$ nur die Bewegungsanteile betrachtet werden, des Weiteren werden mit der neuen diskreten ZV m_t die Bewegungsmuster der einzelnen Klassen modelliert. Die ZVs der Abbildung 6.3 werden im Folgenden näher erläutert.

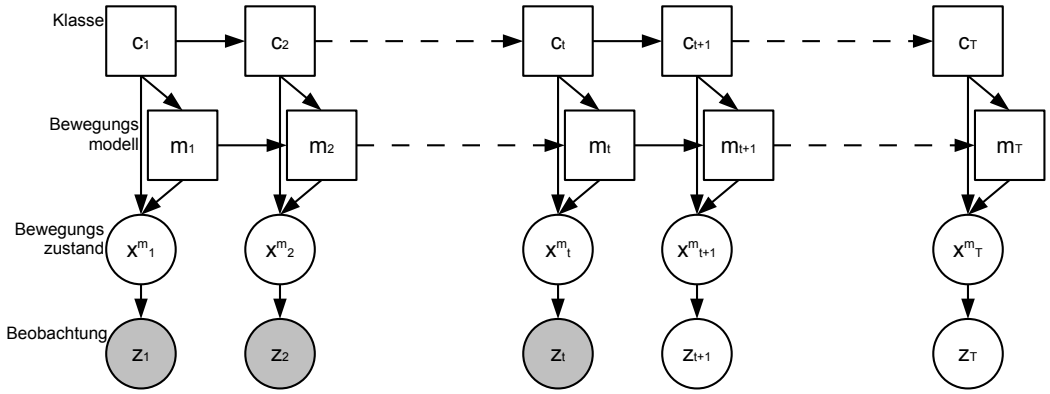


Abbildung 6.3: GM-Darstellung des Ansatzes für das adaptive Bewegungsmodell

- Bewegungsklasse c_t:
  Die ZV c_t ist die Klasse eines beobachteten Musters, in diesem Fall beschreibt ihr Zustand die dynamische Bewegung eines Objektes, z. B. eine dynamische Handgeste. Der zeitliche Übergang zwischen den Zuständen von c_{t-1} und c_t ist von deterministischer Natur, da angenommen wird, dass eine Beobachtungssequenz eine komplette dynamische Bewegung (bzw. Geste) von Anfang bis Ende

beschreibt. Somit bleibt die Bewegungsklasse während einer gesamten Sequenz unverändert.

- Bewegungsmodellzustand m_t:
  Die ZV m_t ähnelt der Zustandsgröße in einem HMM, dabei stellt m_t den zeitlichen Verlauf der Bewegung dar, wobei ein Zustand einen diskreten Zeitschritt repräsentiert. Insgesamt gibt es acht Zustände für acht unterschiedliche Bewegungsrichtungen, somit wird ein Bewegungsmodell für eine Klasse als Abfolge von diskreten Zuständen m_t modelliert.

- Bewegungszustand $\mathbf{x}_t^m$:
  Die vektorielle ZV $\mathbf{x}_t^m$ zeigt die Komponenten des Zustandsvektors $\mathbf{x}_t$ an, die Informationen über die Position des verfolgten Objektes enthalten. Für die Bestimmung des Bewegungsmodells sind diese Informationen ausreichend, Informationen in Bezug auf Größe, Form und Gestalt sind dabei nicht notwendig.

- Beobachtungsvektor $\mathbf{z}_t$:
  Die, aus dem Bild $\mathbf{I}_t$ gewonnenen, Merkmale werden mit der vektoriellen ZV $\mathbf{z}_t$ beschrieben. Generell gibt es viele Möglichkeiten aus einem Videokamerabild Informationen bzw. Merkmale zu erhalten.

6.3.3.2 Der Inferenzprozess

Die Prädiktion des Bewegungsmodells wird durch die ZVs $\mathbf{x}_t^m, m_t$ und c_t geformt, daher können folgende Vereinfachungen gemacht werden. Die Generierung des Beobachtungsvektor $\mathbf{z}_t$ aus dem Zustandsvektor $\mathbf{x}_t$ sowie der *Sampling*-Prozess, der ausschließlich auf der Ebene des Partikel-Filters stattfindet, können für die Prädiktion des Bewegungsmodells vernachlässigt werden, somit ergibt sich folgende Vereinfachung, siehe Abbildung 6.4.

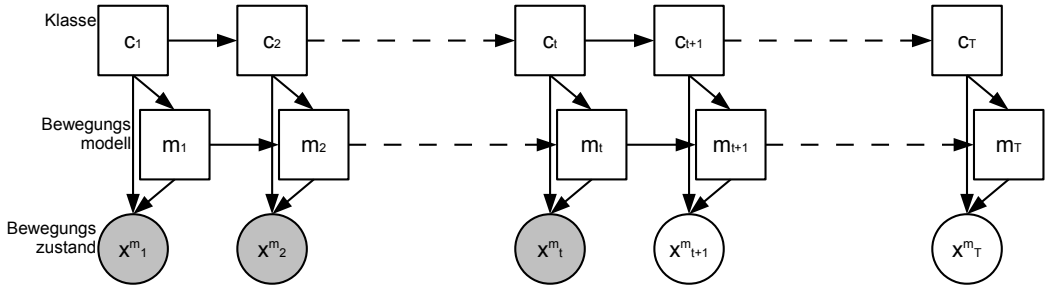


Abbildung 6.4: GM-Darstellung des Ansatzes für das adaptive Bewegungsmodell, wobei die Beobachtungssequenz vernachlässigt wird

Der Inferenzprozess muss für die Bestimmung bzw. Adaptierung des Bewegungsmodells die zukünftige unbekannte ZV $\mathbf{x}_{t+1}^m$ bestimmen, dabei kann die Information

aus der aktuellen Sequenz $\mathcal{X}_t^m$ genutzt werden, somit ist es ausreichend, Inferenz innerhalb des GM von Abbildung 6.4 zu betreiben. Dieses Modell ist vergleichbar mit dem GM von Abbildung 6.3, außer dass die Beobachtungen $\mathbf{z}_\tau, \tau \in [1,\ldots,T]$ vernachlässigt werden und sich deren Beobachtungszustand (grauschattierter Kreis in den Abbildungen) nun auf den Bewegungszustand $\mathbf{x}_\tau^m, \tau \in [1,\ldots,t]$ überträgt.

Der Prädiktionsterm $p(\mathbf{x}_{t+1}^m|\mathcal{X}_t^m)$ wird durch Inferenz im GM von Abbildung 6.4 bestimmt. Um die Nachrichtenpropagierung durchzuführen, müssen zunächst die Schritte aus Abschnitt 3.1.5.1 durchlaufen werden, um eine exakte Inferenz im Verbundbaum ausführen zu können. Zunächst muss der Moralgraph erstellt werden, anschließend wird dieser trianguliert, dabei gibt es mehrere Möglichkeiten, da das Triangulieren eines Graphens nicht eindeutig ist. Basierend auf dem triangulierten Moralgraphen wird nun der Verbundbaum aufgestellt, auf dem anschließend die Nachrichtenpropagierung anhand der Cliquenpotentiale stattfindet.

Der trianguläre Moralgraph und der davon abgeleitete Verbundbaum sind in Abbildung 6.5 bzw. in Abbildung 6.6 dargestellt.

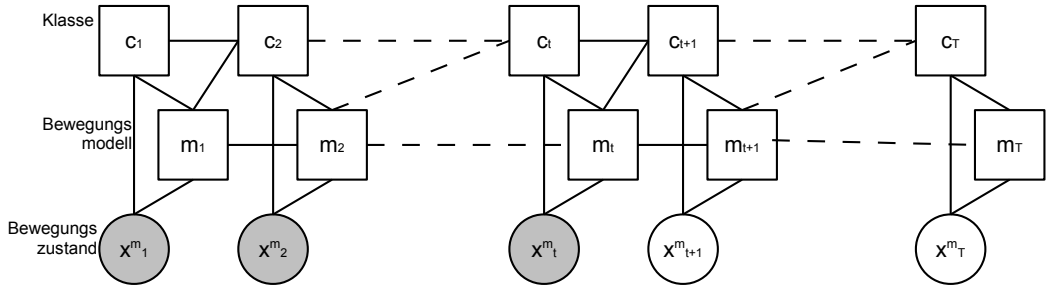


Abbildung 6.5: Der triangulierte Moralgraph des GM von Abbildung 6.4

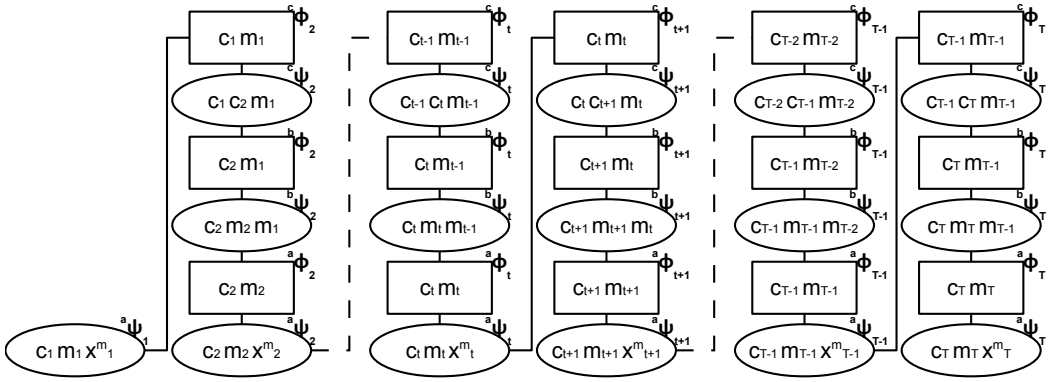


Abbildung 6.6: Der Verbundbaum des GM von Abbildung 6.4. Die Cluster werden als Kreise dargestellt, die Separatoren mithilfe von Rechtecken

Die Verbundwahrscheinlichkeit des Ansatzes für das adaptive Bewegungsmodell kann anhand der Cluster ψ und Separatoren ϕ folgendermaßen bestimmt werden:

$$p(\mathcal{C}_T, \mathcal{M}_T, \mathcal{X}_T^m) = \frac{{}^a\psi_1 \prod_{\tau=2}^{T} {}^a\psi_\tau, {}^b\psi_\tau, {}^c\psi_\tau}{\prod_{\tau=2}^{T} {}^a\phi_\tau, {}^b\phi_\tau, {}^c\phi_\tau}. \tag{6.22}$$

Die entsprechenden Potentiale der Cluster ψ und Separatoren ϕ werden für drei Zeitabschnitte – Prolog, Chunk, Epilog – definiert. Hierbei umfassen Prolog und Epilog jeweils den ersten $\tau = 1$ bzw. den letzten $\tau = T$ Zeitschritt. Der Chunk umfasst alle Zeitschritte dazwischen $[2, \ldots, T-1]$, meist hat hierbei der Epilog die gleiche Beschaffenheit wie der Chunk.

Prolog, $\tau = 1$:

$${}^a\psi_{\tau=1} = p(\mathbf{x}_1^m | m_1, c_1) p(m_1 | c_1) p(c_1).$$

Chunk und Epilog, $\tau \in [2, \ldots, T]$:

$${}^a\psi_\tau = p(\mathbf{x}_\tau^m | m_\tau, c_\tau) \tag{6.23}$$

$${}^b\psi_\tau = p(m_\tau | m_{\tau-1}, c_\tau) \tag{6.24}$$

$${}^c\psi_\tau = p(c_\tau | c_{\tau-1}), \tag{6.25}$$

wobei die Separatoren folgendermaßen für den Zeitschritt $\tau \in [2, \ldots, T]$ intialisiert sind:

$${}^a\phi_\tau = 1 \tag{6.26}$$

$${}^b\phi_\tau = 1 \tag{6.27}$$

$${}^c\phi_\tau = 1. \tag{6.28}$$

Um die bedingte Wahrscheinlichkeit $p(\mathbf{x}_{t+1}^m | \mathcal{X}_t^m)$, die die Prädiktion für das Bewegungsmodell enthält, zu bestimmen, müssen die Cluster ψ und Separatoren ϕ anhand einer Nachrichtenpropagierung aktualisiert werden. Für die Nachrichtenpropagierung wird die Clique ${}^a\psi_{t+1}$ als Wurzelknoten ausgewählt, wobei bis zum Zeitpunkt t Beobachtungen vorliegen, was sich folgendermaßen für die Beobachtungssequenz ausdrückt $\widehat{\mathcal{Z}}_t$: $\tau \in [1, \ldots, t]$, $\{p(\widehat{\mathbf{z}_1} | \mathbf{x}_1^m), \ldots, p(\widehat{\mathbf{z}_t} | \mathbf{x}_t^m)\} \equiv \delta(\widehat{\mathcal{Z}}_t)$.

Alle Cliquen müssen eine Nachricht zum Wurzelknoten ${}^a\psi_{t+1}$ schicken, dabei kommen die Nachrichten aus zwei Richtungen. Aus der ersten Richtung macht das Cluster ${}^a\psi_{\tau=1}$ den Anfang. Dieses Cluster aktualisiert das Potential des Separators ${}^c\phi_{\tau=2}$. Die zweite Richtung für die Nachrichtenpropagierung macht das Cluster ${}^a\psi_T$,

das den Separator ${}^{a}\phi_T$ aktualisiert. Die Nachrichten laufen aus beiden Richtungen bis der Wurzelknoten erreicht ist. Die Aktualisierungen der Cluster- und Separatorenpotentiale für die Zeitabschnitte Prolog, Chunk und Epilog sind wie folgt gegeben:

Prolog $\tau = 1$:Hier werden keine Potentiale aktualisiert.

Chunk $\tau \in [2,\ldots,t]$:

$$ {}^{c}\phi^{*}_{\tau} = \sum_{\widetilde{\mathbf{x}}^{m}_{\tau-1}} {}^{a}\psi_{\tau-1} = {}^{a}\psi_{\tau-1} \tag{6.29} $$

$$ {}^{c}\psi^{*}_{\tau} = \frac{{}^{c}\phi^{*}_{\tau}}{{}^{c}\phi_{\tau}} {}^{c}\psi_{\tau} \tag{6.30} $$

$$ {}^{b}\phi^{*}_{\tau} = \sum_{c_{\tau-1}} {}^{c}\psi^{*}_{\tau} \tag{6.31} $$

$$ {}^{b}\psi^{*}_{\tau} = \frac{{}^{b}\phi^{*}_{\tau}}{{}^{b}\phi_{\tau}} {}^{b}\psi_{\tau} \tag{6.32} $$

$$ {}^{a}\phi^{*}_{\tau} = \sum_{m_{\tau-1}} {}^{b}\psi^{*}_{\tau} \tag{6.33} $$

$$ {}^{a}\psi^{*}_{\tau} = \frac{{}^{a}\phi^{*}_{\tau}}{{}^{a}\phi_{\tau}} {}^{a}\psi_{\tau} \tag{6.34} $$

Chunk $\tau \in [t+2,\ldots,T-1]$:

$$ {}^{a}\psi^{*}_{\tau} = \frac{{}^{c}\phi^{*}_{\tau+1}}{{}^{c}\phi_{\tau+1}} {}^{a}\psi_{\tau} = {}^{a}\psi_{\tau} \tag{6.35} $$

$$ {}^{a}\phi^{*}_{\tau} = \sum_{\mathbf{x}^{m}_{\tau}} {}^{a}\psi_{\tau} = \sum_{\mathbf{x}^{m}_{\tau}} p(\mathbf{x}^{m}_{\tau} | m_{\tau}, c_{\tau}) = 1 \tag{6.36} $$

$$ {}^{b}\psi^{*}_{\tau} = \frac{{}^{a}\phi^{*}_{\tau}}{{}^{a}\phi_{\tau}} {}^{b}\psi_{\tau} = {}^{b}\psi_{\tau} \tag{6.37} $$

$$ {}^{b}\phi^{*}_{\tau} = \sum_{m_{\tau}} {}^{b}\psi_{\tau} = \sum_{m_{\tau}} p(m_{\tau} | m_{\tau-1}, c_{\tau}) = 1 \tag{6.38} $$

$$ {}^{c}\psi^{*}_{\tau} = \frac{{}^{b}\phi^{*}_{\tau}}{{}^{b}\phi_{\tau}} {}^{c}\psi_{\tau} = {}^{c}\psi_{\tau} \tag{6.39} $$

$$ {}^{c}\phi^{*}_{\tau} = \sum_{c_{\tau}} {}^{c}\psi_{\tau} = \sum_{c_{\tau}} p(c_{\tau} | c_{\tau-1}) = 1 \tag{6.40} $$

Epilog $\tau = T$: Die gleichen Gleichungen wie im Chunk $\tau \in [t+2,\ldots,T-1]$ mit Ausnahme der Gleichung 6.35

Chunk $t+1$, Aktualisierung des Wurzelknotens:

$$ {}^{c}\phi^{*}_{t+1} = \sum_{\widehat{\mathbf{x}}^{m}_{t}} {}^{a}\psi_{t} = {}^{a}\psi_{t} \tag{6.41} $$

$$ {}^{c}\psi^{*}_{t+1} = \frac{{}^{c}\phi^{*}_{t+1}}{{}^{c}\phi_{t+1}} {}^{c}\psi_{t+1} \tag{6.42} $$

$$ {}^{b}\phi^{*}_{t+1} = \sum_{c_t} {}^{c}\psi^{*}_{t+1} \tag{6.43} $$

$$ {}^{b}\psi^{*}_{t+1} = \frac{{}^{b}\phi^{*}_{t+1}}{{}^{b}\phi_{t+1}} {}^{b}\psi_{t+1} \tag{6.44} $$

$$ {}^{a}\phi^{*}_{t+1} = \sum_{m_t} {}^{b}\psi^{*}_{t+1} \tag{6.45} $$

$$ {}^{a}\psi^{*}_{t+1} = \frac{{}^{c}\phi^{*}_{t+2}}{{}^{c}\phi_{t+2}} \frac{{}^{a}\phi^{*}_{t+1}}{{}^{a}\phi_{t+1}} {}^{a}\psi_{t+1} \tag{6.46} $$

$$ \begin{aligned} &= p(\mathbf{x}^{m}_{t+1}|m_{t+1},c_{t+1}) \sum_{m_t} p(m_{t+1}|m_t,c_{t+1}) \sum_{c_t} p(c_{t+1}|c_t) p(\mathbf{x}^{m}_{t}|m_t,c_t) \dots \\ &\quad p(\mathbf{x}^{m}_{2}|m_t,c_2) \sum_{m_1} p(m_2|m_1,c_2) \sum_{c_1} p(c_2|c_1) p(\mathbf{x}^{m}_{1}|m_1,c_1) p(m_1|c_1) p(c_1) \end{aligned} \tag{6.47} $$

Von Gleichung 6.46 kann der Prädiktionsterm $p(\mathbf{x}^{m}_{t+1}|\mathcal{X}^{m}_{t})$ für das adaptive Bewegungsmodell folgendermaßen bestimmt werden:

$$ p(\mathbf{x}^{m}_{t+1}|\mathcal{X}^{m}_{t}) = \frac{\sum_{m_{t+1}} \sum_{c_{t+1}} {}^{a}\psi^{*}_{t+1}}{\sum_{x_{t+1}} \sum_{m_{t+1}} \sum_{c_{t+1}} {}^{a}\psi^{*}_{t+1}} \tag{6.48} $$

$$ = \frac{\sum_{\mathcal{M}_{t+1}}, \sum_{\mathcal{C}_{t+1}} p(\mathcal{C}_{t+1}, \mathcal{M}_{t+1}, \mathcal{X}^{m}_{t+1})}{\sum_{\mathcal{M}_{t+1}}, \sum_{\mathcal{C}_{t+1}}, \sum_{\mathbf{x}^{m}_{t+1}} p(\mathcal{C}_{t+1}, \mathcal{M}_{t+1}, \mathcal{X}^{m}_{t+1})} \tag{6.49} $$

$$ = \frac{\sum_{\mathcal{M}_{t+1}} \sum_{\mathcal{C}_{t+1}} p(\mathcal{C}_{t+1}, \mathcal{M}_{t+1}, \mathbf{x}^{m}_{t+1}|\mathcal{X}^{m}_{t}) p(\mathcal{X}^{m}_{t})}{\sum_{\mathcal{M}_{t+1}} \sum_{\mathcal{C}_{t+1}}, \sum_{\mathbf{x}^{m}_{t+1}} p(\mathcal{C}_{t+1}, \mathcal{M}_{t+1}, \mathbf{x}^{m}_{t+1}|\mathcal{X}^{m}_{t}) p(\mathcal{X}^{m}_{t})} \tag{6.50} $$

$$ = \sum_{\mathcal{M}_{t+1}} \sum_{\mathcal{C}_{t+1}} p(\mathcal{C}_{t+1}, \mathcal{M}_{t+1}, \mathbf{x}^{m}_{t+1}|\mathcal{X}^{m}_{t}). \tag{6.51} $$

Von der Nachrichtenpropagierung kann auch gesehen werden, dass die Aktualisierung der Potentiale für die Zeitschritte $\tau \geq t+2$ keinen Einfluss auf das Cluster ${}^{a}\psi^{*}_{t+1}$, das den Wurzelknoten bildet, hat.

Unter Berücksichtigung der gegenseitigen Unabhängigkeiten der ZVs im GM von Abbildung 6.4 kann die Gleichung 6.51, unter Berücksichtigung des Prädiktionsschrittes für $t+1$ und der gegenwärtigen Beobachtung $\tau \in [1,\dots,t]$ folgendermaßen geschrieben werden:

$$p(\mathbf{x}_{t+1}^m|\mathcal{X}_t^m) = \sum_{\mathcal{C}_t}\sum_{\mathcal{M}_t} p(\mathcal{C}_t,\mathcal{M}_t|\mathcal{X}_t^m) \sum_{c_{t+1}}\sum_{m_{t+1}} p(c_{t+1}|c_t)p(m_{t+1}|m_t,c_{t+1})p(\mathbf{x}_{t+1}^m|m_{t+1},c_{t+1}). \tag{6.52}$$

Die Bewegungsklasse c_τ bleibt während einer ganzen Beobachtungssequenz konstant, somit wird für jeden Zeitschritt τ $c_\tau = c$ gesetzt. Daher gibt es für die bedingte Wahrscheinlichkeit $p(c_\tau|c_{\tau-1})$ nun zwei Zustände: Entweder beide Klassen sind gleich, somit wird die $p(c_\tau|c_{\tau-1}) = 1$, oder man erhält $p(c_\tau|c_{\tau-1}) = 0$. Die Wahrscheinlichkeitsfunktion kann nun, wie folgt, vereinfacht werden:

$$p(c_\tau|c_{\tau-1}) = \delta(c_\tau, c_{\tau-1}), \tag{6.53}$$

mit dem Kronecker Delta

$$\delta(c_\tau, c_{\tau-1}) = \begin{cases} 1 & \text{falls } c_\tau = c_{\tau-1} \\ 0 & \text{sonst.} \end{cases} \tag{6.54}$$

Wird dieser Zusammenhang in Gleichung 6.52 eingesetzt, erhält man:

$$\begin{aligned} p(\mathbf{x}_{t+1}^m|\mathcal{X}_t^m) &= \sum_{\mathcal{C}_t}\sum_{\mathcal{M}_t} p(\mathcal{C}_t,\mathcal{M}_t|\mathcal{X}_t^m) \sum_{c_{t+1}}\sum_{m_{t+1}} \delta(c_{t+1}|c_t)p(m_{t+1}|m_t,c_{t+1})p(\mathbf{x}_{t+1}^m|m_{t+1},c_{t+1}) \\ &= \sum_{c}\sum_{\mathcal{M}_t} p(c,\mathcal{M}_t|\mathcal{X}_t^m) \sum_{m_{t+1}} p(m_{t+1}|m_t,c)p(\mathbf{x}_{t+1}^m|m_{t+1},c) \end{aligned} \tag{6.55}$$

Eine kurze Beschreibung von Gleichung 6.55 sieht folgendermaßen aus: Für die gegenwärtige Beobachtung wird die Verbundwahrscheinlichkeit der Bewegungsklasse und dem Bewegungsmodellzustand $p(c,\mathcal{M}_t|\mathcal{X}_t^m)$ bestimmt. Diese Wahrscheinlichkeitsfunktion enthält alle Informationen, die von der Beobachtungssequenz gewonnen werden konnten, um die Wahrscheinlichkeiten für die Bewegungsklasse bzw. Bewegungsmodellzustände zu bestimmen. Ausgehend von $p(c,\mathcal{M}_t|\mathcal{X}_t^m)$ kann die Übergangswahrscheinlichkeit $p(m_{t+1}|m_t,c)$ für jeden zukünftigen Bewegungsmodellzustand m_{t+1} von seinem Vorgänger m_t unter Berücksichtigung der Bewegungsklasse c berechnet werden. Abschließend bestimmt die bedingte Wahrscheinlichkeit $p(\mathbf{x}_{t+1}^m|m_{t+1},c)$ den zukünftigen Bewegungszustand $\mathbf{x}_{t+1}^m$ als eine zusammengesetzte Verteilung von Gauß-Verteilungen, deren Parameter im Lernprozess des GM mittels EM-Algorithmus [86, 87] festgesetzt worden sind. Von dieser gewonnenen Verteilung werden dann die Partikel generiert, um die Bewegung zu verfolgen.

6.3.3.3 Der Algorithmus für das adaptive Bewegungsmodell

Nachfolgend wird der Algorithmus für das adaptive Bewegungsmodell beschrieben. Hierbei ist die Neuerung, dass das Bewegungsmodell, das für die Verteilung der Partikel zuständig ist, durch Inferenz in einem GM bestimmt wird.

Initialisierung
Im ersten Schritt wird eine initiale Schätzung für die Partikelmenge $S_1 = \left\{\mathbf{s}_1^{(1)} \dots \mathbf{s}_1^{(N)}\right\}$ bestimmt, wobei N die Anzahl der verwendeten Partikel ist.

Prinzipiell gibt es mehrere Arten, um eine initiale Schätzung zu erhalten: Die Partikel können zum Beispiel anhand einer Gleichverteilung oder gemäß den Werten einer Gewichtungsfunktion verteilt werden. Im Allgemeinen soll die erste Schätzung den wahren Zustandsvektor $\mathbf{x}_1$ so gut wie möglich reflektieren, daher ist es üblich, die Kenntnis über den Zustandsvektor im ersten Bild zu übernehmen, das heißt $\forall n : \mathbf{s}_1^{(n)} = \mathbf{x}_1$. Der zuletzt beschriebene Ansatz wurde auch in dieser Arbeit gewählt.

In jedem Bild $\mathbf{I}_\tau$ berechnet der Algorithmus die bedingte Wahrscheinlichkeit $p(\mathcal{C}_\tau, \mathcal{M}_\tau | \mathcal{X}_\tau^m)$, daher wird aufgrund der initialen Schätzung für jeden Partikel die bedingte Wahrscheinlichkeit $p(c_1, m_1 | \mathbf{s}_1^{(n)})$ bestimmt. Je nach Art des Zustandsvektors kann diese Abschätzung sinnvoll sein oder auch nicht. In Fällen, in denen absolute Werte zur Zustandsvektorabschätzung dienen, kann $p(c_1, m_1 | \mathbf{s}_1^{(n)})$ bestimmt werden, bei differentiellen Werten kann man sich mit einer Gleichverteilungsapproximation behelfen. Besagter Ansatz wurde auch in dieser Arbeit gewählt.

Haupt-Iteration
Nach dem Ende der Initialisierungsphase werden die folgenden Schritte iterativ für jedes Bild $\mathbf{I}_\tau$, $\tau \in [2, \dots, T]$ wiederholt, bis das Ende der Sequenz $\tau = T$ erreicht ist. Jeder Schritt wird für jeden der ungewichteten N-Partikel $\mathbf{s}_\tau^{(n)}, 1 \leq n \leq N$ wiederholt, dabei schätzt jeder Partikel $\mathbf{s}_\tau^{(n)}$ den Bewegungszustand $\mathbf{x}_\tau$ für das Videokamerabild $\mathbf{I}_\tau$, wobei $\tau \in [1, \dots, T]$.

Variablen-Prädiktion: Der erste Schritt erfolgt nach dem *Resampling*-Schritt aus der vergangenen Iteration.

Zunächst werden die Bewegungsklasse[1] c_τ und der Bewegungsmodellzustand m_τ des aktuellen Bildes $\mathbf{I}_\tau$ anhand der ungewichteten Partikelmenge $S_\tau^{(n)}$ bestimmt. Das entspricht im Prinzip dem ersten Term von Gleichung 6.55, wobei beachtet werden muss, dass dieser Term die gesamte Historie der Bewegungsmodellzustände berücksichtigt. Des Weiteren wird diese Auswertung für jeden der N-Partikel wiederholt. Nun kann man für einen Partikel n sowohl die aktuelle Klasse $\hat{c}_\tau^{(n)}$ als

[1] Aus Gründen der Einheitlichkeit wird für die Bewegungsklasse der Zeitindex τ beibehalten.

auch den Bewegungsmodellzustand $\hat{m}_{\tau}^{(n)}$ bestimmen. Mit dieser Belegung $\{\hat{c}_{\tau}^{(n)}, \hat{m}_{\tau}^{(n)}\}$ kann die bedingte Wahrscheinlichkeit $p(m_{\tau+1}^{(n)}|\hat{c}_{\tau}^{(n)}, \hat{m}_{\tau}^{(n)})$ ausgewertet werden, um eine Prädiktion für den zukünftigen Bewegungsmodellzustand $\hat{m}_{\tau+1}^{(n)}$ des Partikels n abzugeben. Das entspricht dem zweiten Term von Gleichung 6.55.

Partikel-Prädiktion: Als nächster Schritt wird der Bewegungszustand $\mathbf{x}_{t+1}$ anhand der Belegungen der Klasse und des Bewegungsmodellzustands bestimmt. Die Menge $\{c_{t+1}, m_{t+1}\}$ definiert einen Parametersatz $\theta(c,m)$, der im Lernprozess des GM bestimmt worden ist, um die Verteilung für $\mathbf{x}_{\tau+1}$ anhand von zusammengesetzten Gauß-Verteilungen festzulegen. Hierbei werden die Parameter anhand der bestimmten Belegungen von $\hat{c}_{\tau+1}^{(n)}$ und $\hat{m}_{\tau+1}^{(n)}$ ausgewählt, und man erhält für den n-ten Partikelsatz $\mathbf{s'}_{\tau+1}^{(n)}$.

Die Prädiktion der Partikel, basierend auf GM-Inferenz, ist nun abgeschlossen. Man hat nun N-Partikel erhalten, für die, je nach Wahrscheinlichkeit der Klassen und der dazugehörigen Bewegungsmodellzustände, aus den zusammengesetzten Gauß-Verteilungen eine Bewegung ausgewählt worden ist. Die Güte der Partikel wird anschließend in dem Messschritt validiert.

Messschritt: Zur Validierung der Güte der Prädiktion der Partikel muss der Term $p(\mathbf{z}_{\tau+1}|\mathbf{x}_{\tau+1})$ von Gleichung 6.7 ausgewertet werden. Eine analytische Berechnung ist meist nicht einfach oder möglich, dennoch reicht es aus, eine Gewichtungsfunktion zu finden, die die Wahrscheinlichkeit beschreibt, wie beide Parameter – $\mathbf{z}_{\tau+1}, \mathbf{x}_{\tau+1}$ – zusammenpassen. Diese Gewichtungsfunktion muss folgendes Prinzip umsetzen: Sie muss einen größeren Wert für ein Parameterpaar, das eine höhere Auftrittswahrscheinlichkeit hat, zurückliefern und einen geringeren Wert für ein Parameterpaar, das eine geringere Auftrittswahrscheinlichkeit hat. In der praktischen Umsetzung bedeutet dies, dass an der Position, die durch das adaptive Bewegungsmodell bestimmt worden ist, eine Funktion überprüft, ob sich das zu verfolgende Objekt an dieser Position aufhält. Für diese Funktion gibt es mehrere Umsetzungen, beispielsweise kann man Farbhistogramme verwenden oder die Kontur beurteilen. In diesem Beispiel ist das gesuchte Objekt mittels der Hu-Momente [155] beschrieben und anhand einer Distanzfunktion d_m die Ähnlichkeit zwischen einer Referenzobjektbeschreibung ${}^{ref}\mathbf{f}$ und der Hu-Momentbeschreibung ${}^{\mathbf{hu}}\mathbf{s'}_{\tau+1}^{(n)}$ des gegenwärtigen Partikels $\mathbf{s'}_{\tau+1}^{(n)}$ ermittelt worden. Somit berechnet sich das Gewicht $\pi_{\tau+1}^{(n)}$ des Partikels folgendermaßen:

$$\pi_{\tau+1}^{(n)} = \frac{1}{d_m({}^{\mathbf{hu}}\mathbf{s'}_{\tau+1}^{(n)}, {}^{ref}\mathbf{f}) + \varepsilon}, \tag{6.56}$$

wobei d_m die Mahalanobis-Distanz [218] ist, und ε ein konstanter Faktor, der Divisionen durch Null verhindert.

Somit erhält man für jeden Partikel $\mathbf{s'}_{\tau+1}^{(n)}$ ein zugehöriges Gewicht $\pi_{\tau+1}^{(n)}$ von dieser Gewichtungsfunktion.

***Resampling*:** Für den nächsten Zeitschritt werden die bis jetzt gewichteten Partikel $\mathcal{S'}_{\tau+1}^{(n)}$ neu angeordnet, um ungewichtete Partikel $\mathcal{S}_{\tau+1}^{(n)}$ zu erhalten. Dabei werden die zugehörigen Partikel-Gewichte $\pi_{\tau+1}^{(n)}$ zunächst wie folgt normalisiert:

$$\tilde{\pi}_{\tau+1}^{(n)} = \frac{\pi_{\tau+1}^{(n)}}{\sum_{i=1}^{N} \pi_{\tau+1}^{(i)}} \tag{6.57}$$

Somit ähneln die neuen Partikel-Gewichte $\tilde{\pi}_{\tau+1}^{(n)}$ einer Wahrscheinlichkeitsfunktion, aus der eine kumulative Wahrscheinlichkeitsfunktion, repräsentiert durch eine Reihe von Skalaren $k_{\tau+1}^{(n)}$, erstellt wird:

$$k_{\tau+1}^{(n)} = k_{\tau+1}^{(n-1)} + \tilde{\pi}_{\tau+1}^{(n)} \tag{6.58}$$

mit $k_{t+1}^{(N)} = 1$ und $k_{t+1}^{(n)} \in [0;1]$.

Nun werden N ZVs $\hat{k}_{\tau+1}^{(n)}$ aus dem Intervall $[0;1]$ gleichermaßen gezogen. Dem ungewichteten Partikel $\mathbf{s}_{\tau+1}^{(n)}$ wird der Wert des gewichteten Partikels $\mathbf{s'}_{\tau+1}^{(l)}, 1 \leq l \leq N$ zugewiesen, wenn für $\hat{k}_{\tau+1}^{(n)}$ Folgendes gilt:

$$k_{\tau+1}^{(l-1)} < \hat{k}_{\tau+1}^{(n)} \leq k_{\tau+1}^{(l)}. \tag{6.59}$$

6.3.4 Adaptive Gewichtungsfunktion

Der Term $p(\mathbf{z}_t|\mathbf{s}_t^{(n)})$ dient als eine Art Abschätzung der Qualität, wie gut ein Partikel $\mathbf{s}_t^{(n)}$ die gegenwärtige Beobachtung $\mathbf{z}_t$ beschreibt. Dabei wird oft eine Referenzbeschreibung ${}^{ref}\mathbf{f}$ verwendet, um die Qualität des Partikels $\mathbf{s}_t^{(n)}$ anhand seiner dazugehörigen Merkmalsbeschreibung $\mathbf{f}_t$ zu beurteilen. Für diese Beurteilung wird meist eine Gewichtungsfunktion π_t^n verwendet, um eine Qualitätsabschätzung eines Partikels $\mathbf{s}_t^{(n)}$ zu liefern. Sowohl unterschiedliche Merkmale als auch unterschiedliche Funktionen können verwendet werden, um eine Beschreibung für die Ähnlichkeit von der Referenzbeschreibung ${}^{ref}\mathbf{f}$ und der Merkmalsbeschreibung $\mathbf{f}_t$ zu finden.

Die Erweiterung der *adaptiven Gewichtungsfunktion* liegt darin, dass aufgrund der Inferenz in einem GM die zukünftige Konfiguration der Gewichtungsfunktion $\pi_{t+1}^n \sim p(\mathbf{z}_{t+1}|\mathbf{s}_{t+1}^{(i)})$ unter Berücksichtigung der gegenwärtigen Beobachtungssequenz $\mathcal{Z}_t$ erstellt wird. Daher wird die Gestaltung der Gewichtungsfunktion in das GM integriert, dies lässt sich durch einen zusätzlichen Knoten im GM realisieren. Dieser neue Knoten bildet eine ZV w_t, die die Referenzbeschreibung ${}^{ref}\mathbf{f}_t^w$ modelliert.

6.3.4.1 Das Graphische Modell für die adaptive Gewichtungsfunktion

Das GM für die adaptive Gewichtungsfunktion ist in Abbildung 6.7 dargestellt, gegenüber dem GM von Abschnitt 6.3.2 gibt es folgende Änderungen: Bei der Zustandsvektorsequenz werden nur die Anteile betrachtet, die für die Form verantwortlich sind, somit erhält man nun folgende Sequenz $\mathcal{X}_t^s$. Des Weiteren wird mit der neuen diskreten ZV w_t die Form des Objektes modelliert, diese Modellierung wird für die neue adaptive Gewichtungsfunktion verwendet. Die ZVs der Abbildung 6.7 werden im Folgenden näher erläutert.

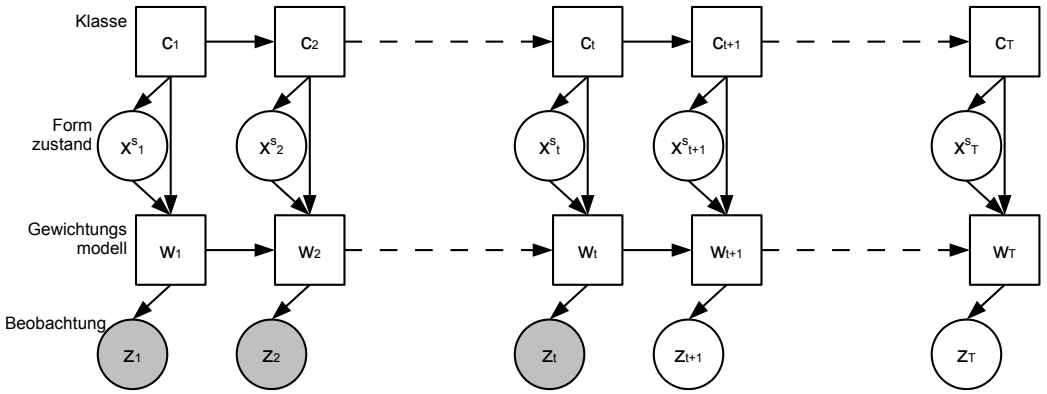


Abbildung 6.7: GM-Darstellung des Ansatzes für die adaptive Gewichtungsfunktion

- Bewegungsklasse c_t:
  Die ZV c_t ist die Klasse eines beobachteten Musters, in diesem Fall beschreibt ihr Zustand die dynamische Bewegung eines Objektes, z. B. eine dynamische Handgeste. Der zeitliche Übergang zwischen den Zuständen von c_{t-1} und c_t ist von deterministischer Natur, da angenommen wird, dass eine Beobachtungssequenz eine komplette dynamische Bewegung (bzw. Geste) von Anfang bis Ende beschreibt. Somit bleibt die Bewegungsklasse während einer gesamten Sequenz unverändert.

- Formzustand $\mathbf{x}_t^s$:
  Die vektorielle ZV $\mathbf{x}_t^s$ zeigt die Komponenten des Zustandsvektors $\mathbf{x}_t$ an, die Informationen über die Form des verfolgten Objektes enthalten. Für die Bestimmung der adaptiven Gewichtungsfunktion sind diese Informationen ausreichend, Informationen in Bezug auf die Position sowie Bewegungsrichtung sind dabei nicht notwendig.

- Formmodellzustand w_t:
  Die ZV w_t modelliert die Parameter, in diesem Fall die Referenzbeschreibung

${}^{ref}\mathbf{f}_t^w$, für die Gestaltung der Gewichtungsfunktion π (Gewichtungsmodell). Je nach Klasse und zeitlichem Fortschritt der Bewegung können unterschiedliche Referenzkonfigurationen mit dieser ZV gebildet werden, somit kann die Bewertung der Partikel $\mathcal{S}_{t+1}^{(N)}$ an die gegenwärtige Beobachtungssequenz $\mathcal{Z}_t$ angepasst werden.

- Beobachtungsvektor $\mathbf{z}_t$:
  Die aus dem Bild $\mathbf{I}$ gewonnenen Merkmale werden mit der vektoriellen ZV $\mathbf{z}_t$ beschrieben. Generell gibt es viele Möglichkeiten aus einem Videokamerabild Informationen bzw. Merkmale zu erhalten.

6.3.4.2 Der Inferenzprozess

Der Inferenzprozess für die adaptive Gewichtungsfunktion soll die zukünftige Konfiguration der ZV w_{t+1} bei gegebener Beobachtungssequenz $\widehat{\mathcal{Z}}_t$ prädizieren, somit können die Partikel $S_{t+1}^{(N)}$ bewertet werden. Die ZV w_{t+1} wird verwendet, um die Referenzbeschreibung $\mathbf{f}_{t+1}^{ref}$ der Gewichtungsfunktion π_{t+1}^n zu gestalten.

Der Prädiktionsterm $p(w_{t+1}|\widehat{\mathcal{Z}}_t)$ wird durch Inferenz im GM von Abbildung 6.7 bestimmt. Anhand der Schritte aus Abschnitt 3.1.5.1 wird die Nachrichtenpropagierung durchgeführt, um eine exakte Inferenz im Verbundbaum ausführen zu können. Zunächst muss der Moralgraph erstellt werden, anschließend wird dieser trianguliert, dabei gibt es mehrere Möglichkeiten, da das Triangulieren eines Graphens nicht eindeutig ist. Basierend auf dem triangulierten Moralgraphen wird nun der Verbundbaum aufgestellt, auf dem anschließend die Nachrichtenpropagierung anhand der Cliquenpotentiale stattfindet.

Der trianguläre Moralgraph und der davon abgeleitete Verbundbaum sind in Abbildung 6.8 bzw. in Abbildung 6.9 dargestellt.

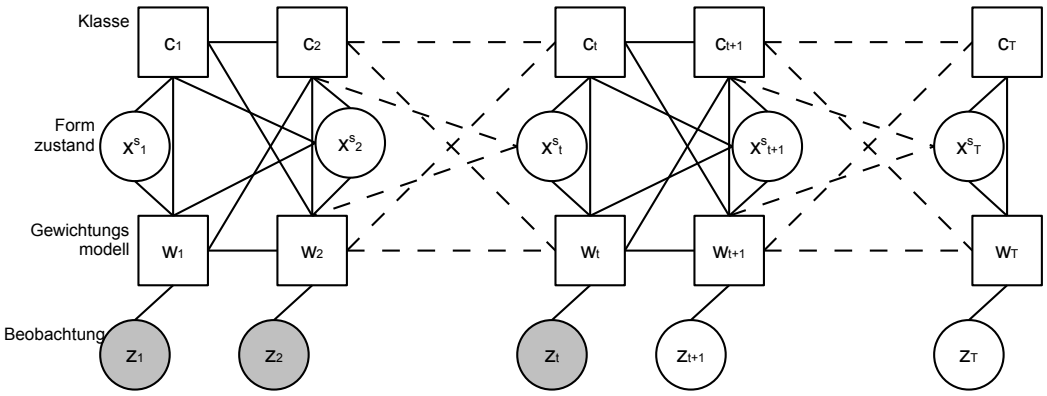


Abbildung 6.8: Der triangulierte Moralgraph des GM von Abbildung 6.7

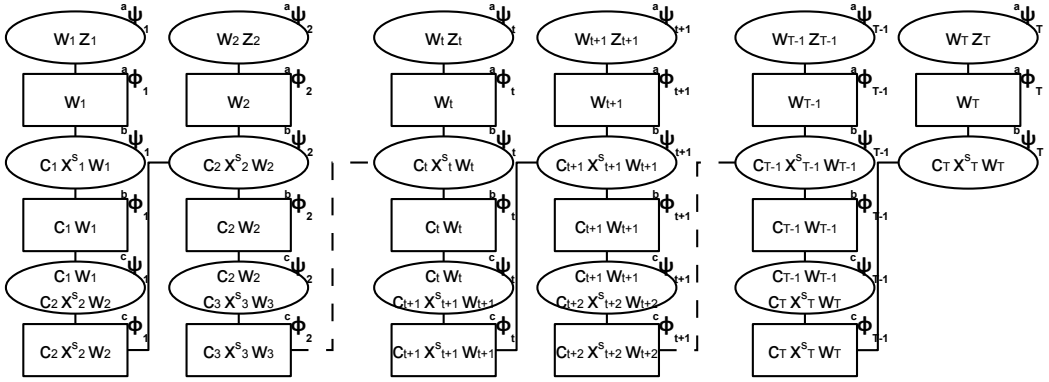


Abbildung 6.9: Der Verbundbaum des GM von Abbildung 6.7. Die Cluster werden als Kreise dargestellt, die Separatoren mithilfe von Rechtecken

Die Verbundwahrscheinlichkeit des Ansatzes für die adaptive Gewichtungsfunktion kann anhand der Cluster ψ und Separatoren ϕ folgendermaßen bestimmt werden:

$$p(\mathcal{C}_T, \mathcal{X}_T^S, \mathcal{W}_T, \mathcal{Z}_T) = \frac{\prod_{\tau=1}^{T-1}({}^a\psi_\tau, {}^b\psi_\tau, {}^c\psi_\tau), {}^a\psi_T, {}^b\psi_T}{\prod_{\tau=1}^{T-1}({}^a\phi_\tau, {}^b\phi_\tau, {}^c\phi_\tau), {}^a\phi_T}. \tag{6.60}$$

Die entsprechenden Potentiale der Cluster ψ und Separatoren ϕ werden, wie beim adaptiven Bewegungsmodell, für drei Zeitabschnitte – Prolog, Chunk, Epilog – definiert. Hierbei umfassen Prolog und Epilog jeweils den ersten $\tau = 1$- bzw. den letzten $\tau = T$-Zeitschritt. Der Chunk umfasst alle Zeitschritte dazwischen $[2, \ldots, T-1]$, meist hat hierbei der Epilog die gleiche Beschaffenheit wie der Chunk.

Prolog, $\tau = 1$:

$$\begin{aligned}
{}^a\psi_{\tau=1} &= p(\mathbf{z}_1|w_1), && (6.61)\\
{}^b\psi_{\tau=1} &= p(\mathbf{x}_1^s|c_1)p(c_1)p(w_1|\mathbf{x}_1^s, c_1), && (6.62)\\
{}^c\psi_{\tau=1} &= p(c_2|c_1)p(w_2|w_1, \mathbf{x}_2^s, c_2). && (6.63)
\end{aligned}$$

Chunk $\tau \in [2, \ldots, T-1]$:

$$\begin{aligned}
{}^a\psi_\tau &= p(\mathbf{z}_\tau|w_\tau), && (6.64)\\
{}^b\psi_\tau &= p(\mathbf{x}_\tau^s|c_\tau), && (6.65)\\
{}^c\psi_\tau &= p(c_{\tau+1}|c_\tau)p(w_{\tau+1}|w_\tau, \mathbf{x}_{\tau+1}^s, c_{\tau+1}). && (6.66)
\end{aligned}$$

Epilog $\tau = T$

$$^{a}\psi_{\tau=T} = p(\mathbf{z}_T|w_T), \tag{6.67}$$
$$^{b}\psi_{\tau} = p(\mathbf{x}_T^s|c_T), \tag{6.68}$$

wobei die Separatoren folgendermaßen initialisiert werden:

$$^{a}\phi_{\tau} = 1, \qquad \tau = [1,\ldots,T] \tag{6.69}$$
$$^{b}\phi_{\tau} = 1, \qquad \tau = [1,\ldots,T-1] \tag{6.70}$$
$$^{c}\phi_{\tau} = 1, \qquad \tau = [1,\ldots,T-1]. \tag{6.71}$$

Für die Prädiktion der adaptiven Gewichtungsfunktion, beschrieben durch die bedingte Wahrscheinlichkeit $p(w_{t+1}|\widehat{\mathcal{Z}}_t)$, müssen die Cluster ψ und Separatoren ϕ anhand einer Nachrichtenpropagierung aktualisiert werden. Für die Nachrichtenpropagierung wird die Clique $^{a}\psi_{t+1}$ als Wurzelknoten ausgewählt, wobei bis zum Zeitpunkt t Beobachtungen vorliegen, was sich folgendermaßen im Hinblick auf die Beobachtungssequenz ausdrückt $\widehat{\mathcal{Z}}_t$: $\tau \in [1,\ldots,t]$, $\{p(\widehat{z_1}|w_1),\ldots,p(\widehat{z_t}|w_t)\} \equiv \delta(\widehat{\mathcal{Z}}_t)$.

Genau wie beim adaptiven Bewegungsmodell müssen alle Cliquen eine Nachricht zum Wurzelknoten $^{a}\psi_{t+1}$ schicken, dabei kommen die Nachrichten aus zwei Richtungen. Aus der ersten Richtung macht das Cluster $^{a}\psi_{\tau=1}$ den Anfang. Dieses Cluster aktualisiert das Potential des Separators $^{a}\phi_{\tau=1}$. Die zweite Richtung für die Nachrichtenpropagierung macht das Cluster $^{a}\psi_T$, das den Separator $^{a}\phi_T$ aktualisiert. Die Nachrichten laufen aus beiden Richtungen, bis der Wurzelknoten erreicht ist. Die Aktualisierungen der Cluster- und Separatorenpotentiale für die Zeitabschnitte Prolog, Chunk und Epilog sind folgendermaßen gegeben:

Prolog $\tau = 1$:

$$^{a}\phi_1^* = \sum_{\widehat{z}_1} {}^{a}\psi_1 = {}^{a}\psi_1 = p(\widehat{z}_1|w_1), \tag{6.72}$$
$$^{b}\psi_1^* = \frac{^{a}\phi_1^*}{^{a}\phi_1} {}^{b}\psi_1 \tag{6.73}$$
$$^{b}\phi_1^* = \sum_{\mathbf{x}_1^s} {}^{b}\psi_1^* \tag{6.74}$$
$$^{c}\psi_1^* = \frac{^{b}\phi_1^*}{^{b}\phi_1} {}^{c}\psi_1 \tag{6.75}$$
$$^{c}\phi_1^* = \sum_{c_1,w_1} {}^{c}\psi_1^* \tag{6.76}$$

Chunk $\tau \in [2, \ldots, t]$:

$$ {}^{a}\phi_{\tau}^{*} = \sum_{\widehat{z}_{\tau}} {}^{a}\psi_{\tau} = {}^{a}\psi_{\tau} = p(\widehat{z}_{\tau}|w_{\tau}), \tag{6.77} $$

$$ {}^{b}\psi_{\tau}^{*} = \frac{{}^{a}\phi_{\tau}^{*}}{{}^{a}\phi_{\tau}} \frac{{}^{c}\phi_{\tau-1}^{*}}{{}^{c}\phi_{\tau-1}} {}^{b}\psi_{\tau} \tag{6.78} $$

$$ {}^{b}\phi_{\tau}^{*} = \sum_{\mathbf{x}_{\tau}^{s}} {}^{b}\psi_{\tau}^{*} \tag{6.79} $$

$$ {}^{c}\psi_{\tau}^{*} = \frac{{}^{b}\phi_{\tau}^{*}}{{}^{b}\phi_{\tau}} {}^{c}\psi_{\tau} \tag{6.80} $$

$$ {}^{c}\phi_{\tau}^{*} = \sum_{c_{\tau}, w_{\tau}} {}^{c}\psi_{\tau}^{*} \tag{6.81} $$

Chunk $\tau \in [t+2, \ldots, T-1]$:

$$ {}^{a}\phi_{\tau}^{*} = \sum_{z_{\tau}} {}^{a}\psi_{\tau} = \sum_{z_{\tau}} p(z_{\tau}|w_{\tau}) = 1, \tag{6.82} $$

$$ {}^{c}\phi_{\tau}^{*} = \sum_{\mathbf{x}_{\tau+1}^{s}} {}^{b}\psi_{\tau+1}^{*}, \tag{6.83} $$

$$ {}^{c}\psi_{\tau}^{*} = \frac{{}^{c}\phi_{\tau}^{*}}{{}^{c}\phi_{\tau}} {}^{c}\psi_{\tau} \tag{6.84} $$

$$ {}^{b}\phi_{\tau}^{*} = \sum_{c_{\tau+1}, w_{\tau+1}} {}^{c}\psi_{\tau}^{*} \tag{6.85} $$

$$ {}^{b}\psi_{\tau}^{*} = \frac{{}^{a}\phi_{\tau}^{*}}{{}^{a}\phi_{\tau}} \frac{{}^{b}\phi_{\tau}^{*}}{{}^{b}\phi_{\tau}} {}^{b}\psi_{\tau} \tag{6.86} $$

Epilog $\tau = T$:

$$ {}^{a}\phi_{T}^{*} = \sum_{z_{T}} {}^{a}\psi_{T} = \sum_{z_{T}} p(z_{T}|w_{T}) = 1, \tag{6.87} $$

$$ {}^{b}\psi_{\tau}^{*} = \frac{{}^{a}\phi_{T}^{*}}{{}^{a}\phi_{T}} {}^{b}\psi_{T} \tag{6.88} $$

Chunk $t+1$, Aktualisierung des Wurzelknotens:

$$
\begin{aligned}
{}^{c}\phi^{*}_{t+1} &= \sum_{\mathbf{x}^{s}_{t+2}} {}^{b}\psi^{*}_{t+2} && (6.89)\\
{}^{c}\psi^{*}_{t+1} &= \frac{{}^{c}\phi^{*}_{t+1}}{{}^{c}\phi_{t+1}} {}^{c}\psi_{t+1} && (6.90)\\
{}^{b}\phi^{*}_{t+1} &= \sum_{c_{t+2},w_{t+2}} {}^{c}\psi^{*}_{t+1} && (6.91)\\
{}^{b}\psi^{*}_{t+1} &= \frac{{}^{c}\phi^{*}_{t}}{{}^{c}\phi_{t}} \frac{{}^{b}\phi^{*}_{t+1}}{{}^{b}\phi_{t+1}} {}^{b}\psi_{t+1} && (6.92)\\
{}^{a}\phi^{*}_{t+1} &= \sum_{c_{t+1},\mathbf{x}^{s}_{t+1}} {}^{b}\psi^{*}_{t+1} && (6.93)\\
{}^{a}\psi^{*}_{t+1} &= \frac{{}^{a}\phi^{*}_{t+1}}{{}^{a}\phi_{t+1}} {}^{a}\psi_{t+1} && (6.94)\\
&= p(\mathbf{z}_{t+1}|w_{t+1}) \sum_{c_{t+1},\mathbf{x}^{s}_{t+1}} p(\mathbf{x}^{s}_{t+1}|c_{t+1}) \\
&\quad \sum_{c_t,w_t} p(c_{t+1}|c_t) p(w_{t+1}|w_t,x_{t+1},c_{t+1}) \sum_{\mathbf{x}^{s}_{t}} p(\mathbf{x}^{s}_{t}|c_t) p(\widehat{z}_t|w_t) \\
&\quad \dots \\
&\quad \sum_{c_1,w_1} p(c_2|c_1) p(w_2|w_1,x_2,c_2) \sum_{\mathbf{x}^{s}_{1}} p(\mathbf{x}^{s}_{1}|c_1) p(c_1) p(w_1|x_1,c_1) p(\widehat{z}_1|w_1) && (6.95)
\end{aligned}
$$

Von Gleichung 6.95 kann der Prädiktionsterm $p(w_{t+1}|\widehat{\mathcal{Z}}_t)$ für die adaptive Gewichtungsfunktion folgendermaßen bestimmt werden:

$$
\begin{aligned}
p(w_{t+1}|\widehat{\mathcal{Z}}_t) &= \frac{\sum_{\mathbf{x}^{s}_{t+1}} \sum_{c_{t+1}} {}^{a}\psi^{*}_{t+1}}{\sum_{w_{t+1}} \sum_{\mathbf{x}^{s}_{t+1}} \sum_{c_{t+1}} {}^{a}\psi^{*}_{t+1}} && (6.96)\\
&= \frac{\sum_{\mathcal{C}_{t+1}} \sum_{\mathcal{X}^{s}_{t+1}} \sum_{\mathcal{W}_t} \sum_{\mathbf{z}_{t+1}} p(\mathcal{C}_{t+1},\mathcal{X}^{s}_{t+1},\mathcal{W}_{t+1},\widehat{\mathcal{Z}}_t,\mathbf{z}_{t+1})}{\sum_{\mathcal{C}_{t+1}} \sum_{\mathcal{X}^{s}_{t+1}} \sum_{\mathcal{W}_{t+1}} \sum_{\mathbf{z}_{t+1}} p(\mathcal{C}_{t+1},\mathcal{X}^{s}_{t+1},\mathcal{W}_{t+1},\widehat{\mathcal{Z}}_t,\mathbf{z}_{t+1})} && (6.97)\\
&= \frac{\sum_{\mathcal{C}_{t+1}} \sum_{\mathcal{X}^{s}_{t+1}} \sum_{\mathcal{W}_t} \sum_{\mathbf{z}_{t+1}} p(\mathcal{C}_{t+1},\mathcal{X}^{s}_{t+1},\mathcal{W}_{t+1},\mathbf{z}_{t+1}|\widehat{\mathcal{Z}}_t) p(\widehat{\mathcal{Z}}_t)}{\sum_{\mathcal{C}_{t+1}} \sum_{\mathcal{X}^{s}_{t+1}} \sum_{\mathcal{W}_{t+1}} \sum_{\mathbf{z}_{t+1}} p(\mathcal{C}_{t+1},\mathcal{X}^{s}_{t+1},\mathcal{W}_{t+1},\mathbf{z}_{t+1}|\widehat{\mathcal{Z}}_t) p(\widehat{\mathcal{Z}}_t)} && (6.98)\\
&= \sum_{\mathcal{C}_{t+1}} \sum_{\mathcal{X}^{s}_{t+1}} \sum_{\mathcal{W}_t} \sum_{\mathbf{z}_{t+1}} p(\mathcal{C}_{t+1},\mathcal{X}^{s}_{t+1},\mathcal{W}_{t+1},\mathbf{z}_{t+1}|\widehat{\mathcal{Z}}_t). && (6.99)
\end{aligned}
$$

Aus der Nachrichtenpropagierung kann auch gesehen werden, dass die Aktualisierung der Potentiale, genau wie beim adaptiven Bewegungsmodell, für die Zeitschritte $\tau \geq t+2$ keinen Einfluss auf das Cluster ${}^{a}\psi^{*}_{t+1}$ ausübt.

Bei Berücksichtigung der gegenseitigen Unabhängigkeiten der ZVs im GM von Abbildung 6.7 kann die Gleichung 6.99, unter Berücksichtigung des Prädiktionsschritt für $t+1$ und der gegenwärtigen Beobachtung $\tau \in [1,\ldots,t]$, folgendermaßen dargestellt werden:

$$\begin{aligned} p(w_{t+1}|\widehat{\mathcal{Z}}_t) = & \sum_{\mathcal{C}_t}\sum_{\mathcal{X}_t^s}\sum_{\mathcal{W}_t} p(\mathcal{C}_t,\mathcal{X}_t^s,\mathcal{W}_t|\widehat{\mathcal{Z}}_t) \\ & \sum_{c_{t+1}}\sum_{\mathbf{x}_{t+1}^s} p(c_{t+1}|c_t)p(\mathbf{x}_{t+1}^s|c_{t+1})p(w_{t+1}|w_t,c_{t+1}\mathbf{x}_{t+1}^s) \\ & \sum_{\mathbf{z}_{t+1}} p(\mathbf{z}_{t+1}|w_{t+1}). \end{aligned} \tag{6.100}$$

Die Bewegungsklasse c_τ bleibt während einer ganzen Beobachtungssequenz konstant, somit wird für jeden Zeitschritt τ $c_\tau = c$ gesetzt. Daher gibt es für die bedingte Wahrscheinlichkeit $p(c_\tau|c_{\tau-1})$ nun zwei Zustände: Entweder beide Klassen sind gleich, und somit wird die $p(c_\tau|c_{\tau-1}) = 1$, falls nicht erhält man $p(c_\tau|c_{\tau-1}) = 0$.

$$p(c_\tau|c_{\tau-1}) = \delta(c_\tau, c_{\tau-1}), \tag{6.101}$$

mit dem Kronecker Delta

$$\delta(c_\tau, c_{\tau-1}) = \begin{cases} 1 & \text{falls } c_\tau = c_{\tau-1} \\ 0 & \text{sonst.} \end{cases} \tag{6.102}$$

Des Weiteren kann die letzte Zeile von Gleichung 6.100 vereinfacht werden, da die Summierung über die unbeobachtete bedingte WF

$$\sum_{\mathbf{z}_{t+1}} p(\mathbf{z}_{t+1}|w_{t+1}) = 1 \tag{6.103}$$

ist, somit kann die Wahrscheinlichkeitsfunktion nun, wie folgt, vereinfacht werden:

$$\begin{aligned} p(w_{t+1}|\widehat{\mathcal{Z}}_t) = & \sum_{\mathcal{C}_t}\sum_{\mathcal{X}_t^s}\sum_{\mathcal{W}_t} p(\mathcal{C}_t,\mathcal{X}_t^s,\mathcal{W}_t|\widehat{\mathcal{Z}}_t) \\ & \sum_{c_{t+1}}\sum_{\mathbf{x}_{t+1}^s} \delta(c_{t+1}|c_t)p(\mathbf{x}_{t+1}^s|c_{t+1})p(w_{t+1}|w_t,c_{t+1}\mathbf{x}_{t+1}^s) \\ & \sum_{\mathbf{z}_{t+1}} p(\mathbf{z}_{t+1}|w_{t+1}) \\ = & \sum_{c}\sum_{\mathcal{X}_t^s}\sum_{\mathcal{W}_t} p(c,\mathcal{X}_t^s,\mathcal{W}_t|\widehat{\mathcal{Z}}_t)\sum_{\mathbf{x}_{t+1}^s} p(\mathbf{x}_{t+1}^s|c)p(w_{t+1}|w_t,c,\mathbf{x}_{t+1}^s). \end{aligned} \tag{6.104}$$

Eine Interpretation von Gleichung 6.104 sieht folgendermaßen aus: Für die gegenwärtige Beobachtung $\widehat{\mathcal{Z}}_t$ wird die Verbundwahrscheinlichkeit von der Bewegungsklasse, Formzustand und Formmodellzustand $p(c,\mathcal{X}_t^s,\mathcal{W}_t|\widehat{\mathcal{Z}}_t)$ bestimmt, basierend darauf versucht $p(\mathbf{x}_{t+1}^s|c)$ die Entwicklung der Form des verfolgten Objektes zu prädizieren. Mit der Abschätzung der zukünftigen Form des Objektes,

der Bewegungsklasse und des gegenwärtigen Formmodellzustands schätzt die WDF $p(w_{t+1}|w_t, c, \mathbf{x}^s_{t+1})$ die zukünftige Konfiguration der Referenzbeschreibung ${}^{ref}\mathbf{f}^w_{t+1}$ ab. Diese Referenzbeschreibung wird bei der Bewertung mit der Gewichtungsfunktion π_{t+1} der ungewichteten Partikel verwendet.

6.3.4.3 Der Algorithmus für die adaptive Gewichtungsfunktion

Im Folgenden wird der Algorithmus für die adaptive Gewichtungsfunktion beschrieben. Die Besonderheit bei diesem Algorithmus liegt darin, dass die Bewertungsfunktion für die Partikel, aufgrund der vorangegangenen Beobachtungen, adaptiv angepasst werden kann. Die Inferenz in einem GM bestimmt über eine ZV $w_{\tau+1}$ die zukünftige Konfiguration eines Referenzvektors ${}^{ref}\mathbf{f}^w_{\tau+1}$ für die Partikelbewertungsfunktion $\pi_{\tau+1}$.

Initialisierung
Im ersten Schritt wird eine initiale Schätzung für die Partikelmenge $S_1 = \left\{\mathbf{s}_1^{(1)} \dots \mathbf{s}_1^{(N)}\right\}$ bestimmt, wobei N die Anzahl der verwendeten Partikel ist. Jeder Partikel $\mathbf{s}_\tau^{(n)}$ schätzt die Position des gesuchten Objektes $\mathbf{x}_\tau$ für das Videokamerabild $\mathbf{I}_\tau$ ab, wobei $\tau \in [1, \dots, T]$ gilt. Genau wie beim adaptiven Bewegungsmodell wird auch initial für das erste Videokamerabild $\mathbf{I}_1$ die Kenntnis über den Zustandsvektor übernommen, das heißt $\forall n : \mathbf{s}_1^{(n)} = \mathbf{x}_1$. Des Weiteren wird eine Gleichverteilungsapproximation verwendet, um die initiale Verteilung der ZVs von $p(c_1, \mathbf{x}^s_1, w_1|\mathbf{s}_1^{(n)})$ zu bestimmen.

Haupt-Iteration
Nach dem Ende der Initialisierungsphase werden die folgenden Schritte iterativ für jedes Bild $\mathbf{I}_\tau$, $\tau \in [2, \dots, T]$ wiederholt, bis das Ende der Sequenz $\tau = T$ erreicht ist. Jeder Schritt wird für jeden der ungewichteten N-Partikel $\mathbf{s}_\tau^{(n)}, 1 \leq n \leq N$ wiederholt.

Variablen-Prädiktion: Der erste Schritt erfolgt nach dem *Resampling*-Schritt aus der vergangenen Iteration.

Ausgehend von der ungewichteten Partikelmenge $S_\tau^{(n)}$ werden die Belegungen für die ZVs c_τ, $\mathbf{x}^s_\tau$ und w_τ für jeden der N-Partikel bestimmt. Dieses Vorgehen entspricht dem Auswerten des ersten Terms von Gleichung 6.104, bei dieser Auswertung fließt die gesamte Historie von $\{\mathcal{C}_{\tau-1}, \mathcal{X}^s_{\tau-1}, \mathcal{W}_{\tau-1}\}$ ein. Nun kann für einen Partikel n sowohl die aktuelle Klasse $\hat{c}_\tau^{(n)}$ als auch der Formmodellzustand $\hat{w}_\tau^{(n)}$ sowie der Formzustand $\hat{\mathbf{x}}_\tau^{s,(n)}$ bestimmt werden.

Partikel-Prädiktion: Zur Abschätzung des Bewegungsmodells werden die Schritte – Drift, Diffusion – aus dem CONDENSATION-Algorithmus verwendet, hier-

bei kommt ein stochastischer Zufallsprozess zum Einsatz, somit werden die Positionen der N-Partikel zufällig bestimmt und man erhält den Partikelsatz $S'^{(n)}_{\tau+1}$.

Messschritt: Zur Validierung der Güte der Prädiktion der Partikel wird nun der Term $p(\mathbf{z}_{\tau+1}|w_{\tau+1})$ evaluiert. Der Formstand $\mathbf{x}^s_{\tau+1}$ des gesuchten Objektes wird in diesem Fall auch mit Hu-Momenten [155] beschrieben. Die Belegung des Formmodellzustands $w_{\tau+1}$ wird durch Inferenz im GM bestimmt, somit ergibt sich die Belegung des letzten Terms – $p(w^{(n)}_{\tau+1}|\hat{w}^{(n)}_{\tau}, c^{(n)}_{\tau+1}, \mathbf{x}^{s,(n)}_{\tau+1})$ – von Gleichung 6.104. Dieser Formmodellzustand wird verwendet, um die gegenwärtige Gewichtungsfunktion anzupassen, denn aufgrund der vorangegangenen Beobachtungen wird die Referenzobjektbeschreibung des Referenzvektors ${}^{ref}\mathbf{f}^{w}_{\tau+1}$ anhand der ZV $w^{(n)}_{\tau+1}$ angepasst. Somit ergibt sich anhand der adaptiven Gewichtungsfunktion für den n-ten Partikel folgendes Gewicht

$$\pi^{(n)}_{\tau+1} = \frac{1}{d_m({}^{\mathbf{hu}}\mathbf{s'}^{(n)}_{\tau+1}, {}^{ref}\mathbf{f}^{w}_{\tau+1}) + \varepsilon}, \tag{6.105}$$

wobei d_m die Mahalanobis-Distanz [218] ist, ${}^{\mathbf{hu}}\mathbf{s'}^{(n)}_{\tau+1}$ die Hu-Momentbeschreibung des gegenwärtigen Partikels $\mathbf{s'}^{(n)}_{\tau+1}$ und ε ein konstanter Faktor, der Divisionen durch Null verhindert.

Somit erhält man für jeden Partikel $\mathbf{s'}^{(n)}_{\tau+1}$ ein zugehöriges Gewicht $\pi^{(n)}_{\tau+1}$ von dieser Gewichtungsfunktion.

***Resampling*:** Für den nächsten Zeitschritt werden die bis jetzt gewichteten Partikel $\mathcal{S}'^{(n)}_{\tau+1}$ neu angeordnet, um ungewichtete Partikel $\mathcal{S}^{(n)}_{\tau+1}$ zu erhalten, dabei werden die zugehörigen Partikel-Gewichte $\pi^{(n)}_{\tau+1}$ benutzt. Die Schritte Normalisierung (siehe Gleichung 6.57) und Ziehen (siehe Gleichung 6.59) der ungewichteten Partikel aus den gewichteten Partikeln erfolgen gemäß dem gleichen Vorgehen wie beim adaptiven Bewegungsmodell.

6.3.5 Ganzheitlicher GM-Ansatz

Die beiden Ansätze – adaptives Bewegungsmodell (siehe Abschnitt 6.3.3), adaptive Gewichtungsfunktion (siehe Abschnitt 6.3.4) – können aufgrund ihrer GM-Struktur in ein einheitliches GM zusammengefasst werden. Dieses ganzheitliche GM kombiniert die Vorteile des adaptiven Bewegungsmodells mit denen der adaptiven Gewichtungsfunktion, die entsprechende GM-Topologie ist in Abbildung 6.10 dargestellt.

Die ZVs des adaptives Bewegungsmodells und der adaptiven Gewichtungsfunktion werden im ganzheitlichen GM-Ansatz kombiniert, hierbei werden die Eigenschaften (Bewegung, Form) des zu verfolgenden Objektes $\mathbf{x}_t$ getrennt und mit den zwei ZVs $\mathbf{x}^m_t$ und $\mathbf{x}^s_t$ modelliert.

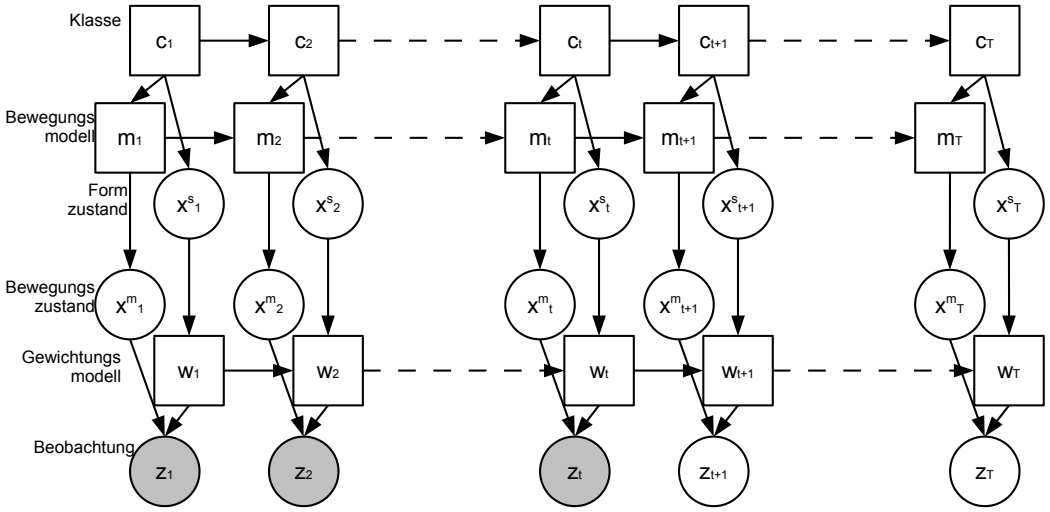


Abbildung 6.10: GM-Darstellung des ganzheitlichen GM-Ansatzes, der adaptives Bewegungsmodell und adaptive Gewichtungsfunktion in einer GM-Struktur kombiniert

Der Inferenzprozess für den ganzheitlichen GM-Ansatz bestimmt sowohl die Verteilung – $p(\mathbf{x}^m_{t+1}|\widehat{\mathcal{Z}}_t)$[1] – der zukünftigen Partikel $\mathbf{s}^{(N)}_{t+1}$ als auch deren Bewertung – $p(w_{t+1}|\widehat{\mathcal{Z}}_t)$ – anhand der gegebenen Beobachtungssequenz $\widehat{\mathcal{Z}}_t$. Zur Durchführung der Inferenz wird zunächst der Moralgraph erstellt und anschließend trianguliert. Basierend auf der gewählten Triangulation kann anschließend der Verbundbaum aufgestellt werden und die Nachrichtenpropagierung anhand der Cliquenpotentiale durchgeführt werden.

Der triangulierte Moralgraph des ganzheitlichen GM-Ansatzes und der davon abgeleitete Verbundbaum sind in Abbildung 6.11 bzw. in Abbildung 6.12 dargestellt.

Die Verbundwahrscheinlichkeit des ganzheitlichen GM-Ansatzes kann anhand der Cluster ψ und Separatoren ϕ folgendermaßen bestimmt werden:

$$p(\mathcal{C}_T, \mathcal{M}_T, \mathcal{X}^S_T, \mathcal{X}^m_T, \mathcal{W}_T, \mathcal{Z}_T) = \frac{\prod_{\tau=1}^{T-1}({}^a\psi_\tau, {}^b\psi_\tau, {}^c\psi_\tau), {}^a\psi_T, {}^b\psi_T}{\prod_{\tau=1}^{T-1}({}^a\phi_\tau, {}^b\phi_\tau, {}^c\phi_\tau), {}^a\phi_T}. \tag{6.106}$$

Die entsprechenden Potentiale der Cluster ψ und Separatoren ϕ werden, wie bei den vorangegangenen Modellen, für drei Zeitabschnitte – Prolog, Chunk, Epilog – definiert. Hierbei umfassen Prolog und Epilog jeweils den ersten $\tau = 1$- bzw. den letzten $\tau = T$-Zeitschritt. Der Chunk umfasst alle Zeitschritte dazwischen $[2, \ldots, T-1]$, meist hat hierbei der Epilog die gleiche Beschaffenheit wie der Chunk.

[1] Die eingeführte Vereinfachung beim adaptiven Bewegungsmodell wird hier nicht angewendet, daher beschreibt $p(\mathbf{x}^m_{t+1}|\widehat{\mathcal{Z}}_t)$ die Verteilung der Partikel und nicht $p(\mathbf{x}^m_{t+1}|\mathcal{X}^m_t)$.

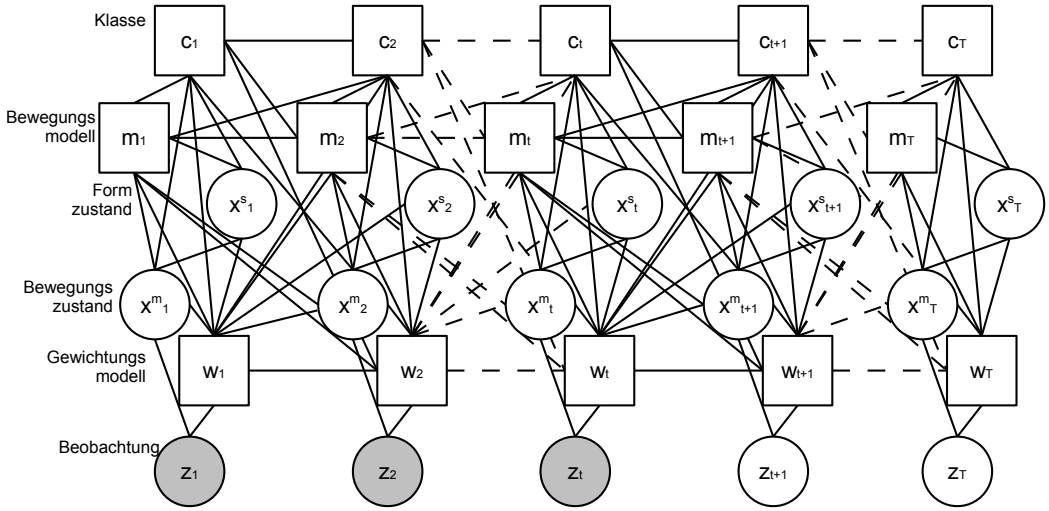


Abbildung 6.11: Der triangulierte Moralgraph des GM von Abbildung 6.10

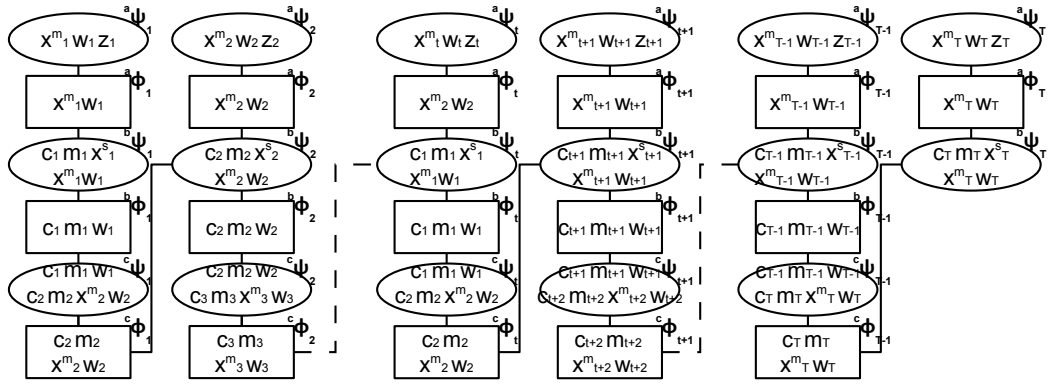


Abbildung 6.12: Der Verbundbaum des GM von Abbildung 6.10. Die Cluster werden als Kreise dargestellt, die Separatoren mithilfe von Rechtecken

Prolog, $\tau = 1$:

$$^{a}\psi_{\tau=1} = p(\mathbf{z}_1|\mathbf{x}_1^m, w_1), \tag{6.107}$$

$$^{b}\psi_{\tau=1} = p(\mathbf{x}_1^s|c_1)p(\mathbf{x}_1^m|m_1), \tag{6.108}$$

$$^{c}\psi_{\tau=1} = p(c_1)p(m_1|c_1)p(w_1|\mathbf{x}_1^s). \tag{6.109}$$

Chunk $\tau \in [2,\ldots,T-1]$:

$$^{a}\psi_{\tau} = p(\mathbf{z}_{\tau}|\mathbf{x}^{m}_{\tau}, w_{\tau}), \tag{6.110}$$

$$^{b}\psi_{\tau} = p(\mathbf{x}^{s}_{\tau}|c_{\tau})\,p(\mathbf{x}^{m}_{\tau}|m_{\tau}), \tag{6.111}$$

$$^{c}\psi_{\tau} = p(c_{\tau}|c_{\tau-1})\,p(m^{m}_{\tau}|m^{m}_{\tau-1}, c_{\tau})\,p(w_{\tau}|w_{\tau-1}, \mathbf{x}^{s}_{\tau}). \tag{6.112}$$

Epilog $\tau = T$

$$^{a}\psi_{\tau=T} = p(\mathbf{z}_{T}|\mathbf{x}^{m}_{T}, w_{T}), \tag{6.113}$$

$$^{b}\psi_{\tau} = p(\mathbf{x}^{s}_{T}|c_{T})\,p(\mathbf{x}^{m}_{T}|m_{T}), \tag{6.114}$$

wobei die Separatoren folgendermaßen intialisiert werden:

$$^{a}\phi_{\tau} = 1, \qquad \tau = [1,\ldots,T] \tag{6.115}$$

$$^{b}\phi_{\tau} = 1, \qquad \tau = [1,\ldots,T-1] \tag{6.116}$$

$$^{c}\phi_{\tau} = 1, \qquad \tau = [1,\ldots,T-1]. \tag{6.117}$$

Zur Bestimmung der Parameter des adaptiven Bewegungsmodells – $p(\mathbf{x}^{m}_{t+1}|\widehat{\mathcal{Z}}_{t})$ – und der adaptiven Gewichtungsfunktion – $p(w_{t+1}|\widehat{\mathcal{Z}}_{t})$ – müssen zunächst die Cluster ψ und Separatoren ϕ anhand einer Nachrichtenpropagierung aktualisiert werden. Die Clique $^{a}\psi_{t+1}$ wird als Wurzelknoten gewählt, somit müssen alle Cliquen eine Nachricht zu diesem Wurzelknoten schicken. Genauso wie bei den vorangegangenen Modellen fließen die Beobachtungen $\widehat{\mathcal{Z}}_{t}$ in die Nachrichtenpropagierung[1] mit ein.

6.4 Experimente

6.4.1 Datensatz

Für die Evaluierung der unterschiedlichen Ansätze zur bildbasierten Verfolgung wurden unterschiedliche dynamische Handgesten verwendet. Dafür wurden Sequenzen von fünf unterschiedlichen Personen aufgezeichnet, die mit der *Leave-One-Person-Out*-Kreuzvalidierung (siehe Abschnitt 3.2.1) evaluiert wurden. Hierbei wurde der Testdatensatz immer von einer Person gestellt, während die vier verbliebenen Personen den Trainingsdatensatz bildeten. Insgesamt wurden fünf unterschiedliche Klassen aufgenommen, wobei es für jede Geste zwölf Wiederholungen gab. Diese fünf Klassen sind in Abbildung 6.13 dargestellt. Neben den Gesten sind auch die korrespondierenden Hautfarbenbilder dargestellt, die für die Evaluierung der unterschiedlichen Ansätze zur bildbasierten Verfolgung verwendet wurden.

[1] Auf eine explizite Darstellung der Nachrichtenpropagierung und der algorithmischen Beschreibung des ganzheitlichen GM-Ansatzes wird hier verzichtet, da sich keine wesentlichen Unterschiede zum adaptiven Bewegungsmodell bzw. zur adaptiven Gewichtungsfunktion ergeben.

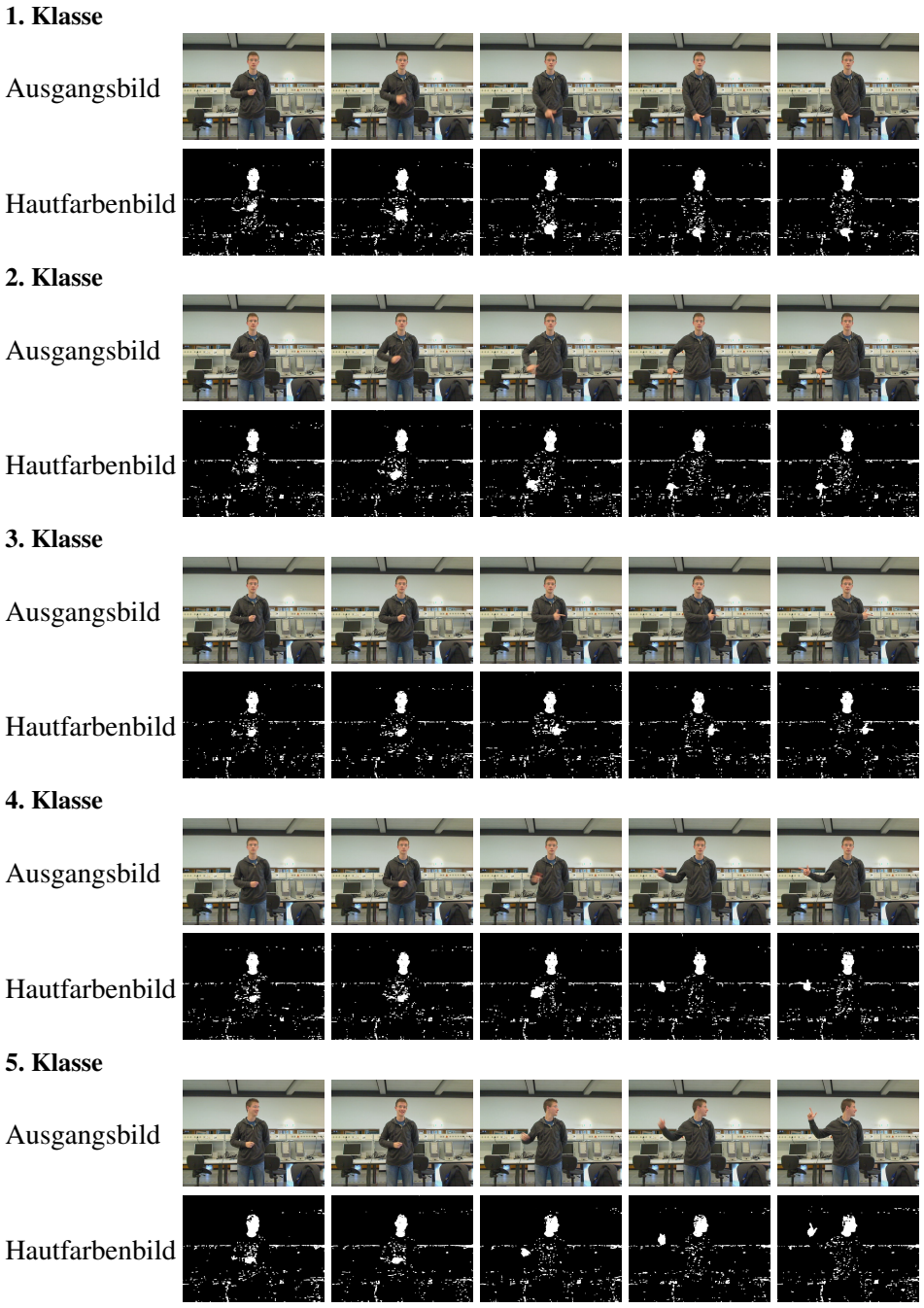


Abbildung 6.13: Überblick über die aufgenommenen Gestenklassen

Neben den aufgenommenen fünf Klassen wurden zusätzlich noch vier weitere erzeugt, indem die Klassen (2, 3, 4, 5) gespiegelt wurden. Somit wurden für die Tests neun unterschiedliche Klassen verwendet, die mit vier verschiedenen Ansätzen getes-

tet wurden: Ein CONDENSATION-Algorithmus als Referenz mit einem stochastischen Zufallsprozess als Bewegungsmodell, der Ansatz mit dem adaptiven Bewegungsmodell von Abschnitt 6.3.3, der Ansatz mit der adaptiven Gewichtungsfunktion von Abschnitt 6.3.4 und der ganzheitliche GM-Ansatz von Abschnitt 6.3.5.

6.4.2 Evaluierungsmaße

Für die Beurteilung der Leistungsfähigkeit der Algorithmen im Vergleich untereinander werden folgende Evaluierungsmaße verwendet: *Object Tracking Error* (OTE) und *Tracker Detection Rate* (TRDR), beide Maße wurden in [257] vorgestellt. Für jedes Videokamerabild wird der Schwerpunkt des verfolgten Objektes anhand der N-Partikel bestimmt (in diesem Fall die Hand, mit welcher die Geste ausgeführt wird) und mit der *Ground Truth* verglichen. Der durchschnittlich kürzeste kartesische Abstand zwischen dem ermittelten Schwerpunkt $\mathbf{p}({}^{cg}x_\tau, {}^{cg}y_\tau)$ und der *Ground Truth* $\mathbf{p}({}^{gt}x_\tau, {}^{gt}y_\tau)$ für die gesamte Zeitsequenz T bildet den OTE folgendermaßen:

$$\mathrm{OTE} = \frac{1}{T}\sum_{\tau=1}^{T}\sqrt{({}^{gt}x_\tau - {}^{cg}x_\tau)^2 + ({}^{gt}y_\tau - {}^{cg}y_\tau)^2}, \tag{6.118}$$

wobei T die Länge der zeitlichen Sequenz beschreibt.

Die TRDR definiert eine Distanzschwelle, mit der überprüft wird, ob der Schwerpunkt der ermittelten Position des Objektes einen gewissen Abstand zur *Ground Truth*-Position des gesuchten Objektes einhält bzw. unterschreitet. In der bildbasierten Verfolgung von Handgesten ist die Hand das gesuchte Objekt und hat einen Durchmesser von ca. $d = 140$ px. Die Distanzschwelle wird auf $d/2 = 70$ px gesetzt, somit wird für jedes Videokamerabild, in dem der Abstand geringer als $d/2$ ist, eine positive Erkennung gezählt, und der Zähler n_P inkrementiert. Die TRDR ist dann definiert als

$$\mathrm{TRDR} = \frac{n_P}{T}, \tag{6.119}$$

wobei T die Länge der zeitlichen Sequenz ist.

6.4.3 Ergebnisse

Für die Evaluierung der drei entwickelten Ansätze zur bildbasierten Verfolgung wurde ein Hautfarbenbild der dynamischen Handgesten erstellt. Im Gegensatz zum Hautfarbenmodell von Abschnitt 4.4.1.1, in dem ein adaptives, auf dem Farbhistogramm des Gesichtes beruhendes Hautfarbenmodell erstellt wurde, verwendete man für das Hautfarbenbild der dynamischen Handgesten ein statisches Hautfarbenmodell. Dieses statische Hautfarbenmodell hat zur Folge, dass das Rauschen im Bild größer ist. Dieser Effekt war aber gewünscht, um die Aufgabe für die entwickelten Ansätze schwieriger zu gestalten, deswegen wurde auch auf morphologische Operationen [145] zur

Rauschreduzierung verzichtet. Die Ergebnisse der drei entwickelten Ansätze sind in Tabelle 6.1 dargestellt, zusätzlich finden sich noch die Ergebnisse, die mit einem CONDENSATION-Algorithmus erhalten worden sind, dessen Bewegungsmodell auf einem stochastischen Zufallsprozess beruht.

	CONDENSATION-Algorithmus	Adaptives Bewegungsmodell	Adaptive Gewichtungs-funktion	Ganzheitlicher GM-Ansatz
OTE	134 px	125 px	96 px	81 px
TRDR	44,76 %	47,09 %	59,74 %	66,00 %

Tabelle 6.1: Überblick über OTE und TRDR für die drei entwickelten Ansätze (adaptives Bewegungsmodell, adaptive Gewichtungsfunktion und ganzheitlicher GM-Ansatz) und den CONDENSATION-Algorithmus als Vergleich

Die Ergebnisse zeigen, dass der Einsatz von GM-Inferenz für die bildbasierte Verfolgung von Handgesten vorteilhaft sein kann, insbesondere bei Rauschen. Der Referenz CONDENSATION-Algorithmus wird bei der gegebenen Aufgabenstellung mit starkem Rauschen von allen drei Ansätzen übertroffen, bei geringem Rauschen nähern sich die verschiedenen Verfahren wieder an. Der Effekt des adaptiven Bewegungsmodells fällt gegenüber dem CONDENSATION-Algorithmus nicht so stark ins Gewicht, somit ergeben sich nur geringfügige Verbesserungen. Die adaptive Gewichtungsfunktion sowie die Kombination von beiden Verfahren im ganzheitlichen GM-Ansatz bringen die besten Ergebnisse.

6.5 Diskussion

In diesem Kapitel wurde anhand von GMs der klassische CONDENSATION-Algorithmus erweitert, um sowohl eine aktuelle Beobachtungssequenz unmittelbar klassifizieren zu können als auch aufgrund von der GM-Inferenz das Bewegungsmodell bzw. die Gewichtungsfunktion, basierend auf der gegenwärtigen Beobachtung, adaptiv für den nächsten Zeitschritt anzupassen. Beide Verfahren, adaptives Bewegungsmodell und adaptive Gewichtungsfunktion, wurden noch in einer ganzheitlichen GM-Topologie fusioniert, um die Vorteile beider Verfahren zu kombinieren. Bei allen Verfahren kamen sowohl die exakte als auch inexakte Inferenz zum Einsatz, wobei die exakte Inferenz verwendet wurde, um die Belegung der Knoten (Klasse, Bewegungsmodell, Gewichtungsmodell) zu bestimmen, die die Darstellung des Modells beschreiben. Die inexakte Inferenz wurde verwendet, um die Beobachtung mit vertretbarem Aufwand zu modellieren. Das adaptive Bewegungsmodell kann für die Prädiktion der Position der nächsten Partikel auf eine gelernte Menge von Bewegungsklassen zugreifen. Diese beschreiben anhand ihres Bewegungsmodells eine WDF, aus der die

Bewegungsparameter der Partikel bestimmt werden. Im Gegensatz zum adaptiven Bewegungsmodell wird bei der adaptiven Gewichtungsfunktion die GM-Inferenz verwendet, um die Gewichtungsfunktion für die Partikel anzupassen, somit kann man Veränderungen in der Gestalt des zu verfolgenden Objektes berücksichtigen. Beide Verfahren wurden in einer ganzheitlichen GM-Topologie kombiniert. Generell zeigen sich die Vorteile der vorgestellten Verfahren insbesondere bei starkem Rauschen im Hautfarbenbild. Hierbei kann die Information, die anhand der Inferenz bereitgestellt wird, hilfreich sein, um den bildbasierten Verfolgungsprozess auf der richtigen Spur zu halten.

Abschließend lässt sich feststellen, dass der Einsatz von GMs im Bereich der bildbasierten Verfolgungsprozesse eine Vielzahl an Verwendungsmöglichkeiten bietet, da sich die GMs sowohl dazu eignen, die dynamischen Prozesse zu modellieren, als auch dazu, eine probabilistische Beschreibung der Eigenschaften des Objektes zu liefern. Der Einsatz der GM-Inferenz kann hilfreich sein, um alle beteiligten Komponenten eines bildbasierten Verfolgungsprozesses in Beziehung zu setzen und somit Abhängigkeiten sowie wechselseitige Einflussmöglichkeiten zu modellieren.

Kapitel 7

Spieleanwendung auf einer multimodalen Roboterplattform als Beispielanwendung für AAL

Inhaltsangabe

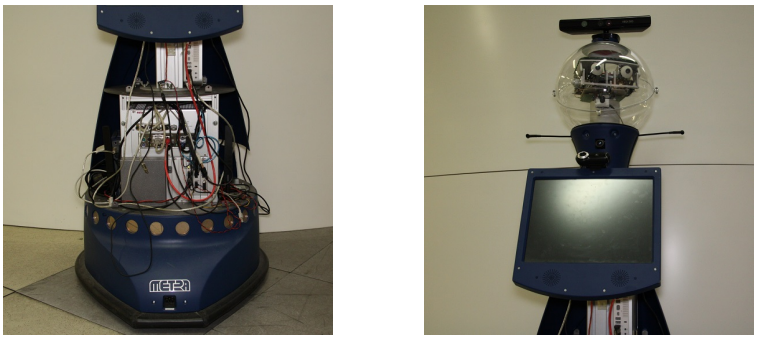
In diesem Kapitel wird ein exemplarisches Spieleszenario auf einer Roboterplattform als eine mögliche Ambient Assisted Living-Anwendung vorgestellt. Die entwickelten Methoden zur Gestenerkennung von Kapitel 4 sowie zur Mimikerkennung von Kapitel 5 werden in diesem Spieleszenario aufgegriffen. Zunächst werden die Roboterplattform und ihre Komponenten kurz vorgestellt, dem schließt sich eine Betrachtung der Roboterplattform unter dem Aspekt wahrnehmender bzw. agierender Modalitäten an. Abschließend erfolgt eine Beschreibung des Spieleszenarios, in der auf die Ausgestaltung der multimodalen Interaktionsmöglichkeiten im Spiel eingegangen wird.

7.1 Einführung

Die Forschungsroboterplattform *Enhanced Living Assistant* (ELIAS), auf der das Spieleszenario integriert worden ist, ist im Prinzip bis auf wenige Modifikationen baugleich mit der Roboterplattform vom ALIAS-Projekt (siehe Abschnitt 2.3). ELIAS ist eine rein auf Kommunikationsaspekte reduzierte Forschungsroboterplattform und soll unterschiedliche Aspekte der Mensch-Roboter-Interaktion zu untersuchen helfen. Der Fokus bei der Forschung mit der Roboterplattform ELIAS liegt darauf, die Interaktion zwischen Mensch und Roboter natürlich und intuitiv zu gestalten. Aufgrund ihrer technischen Ausstattung ist die Roboterplattform ELIAS in der Lage, multimodal mit einem Menschen zu interagieren, wodurch für die Interaktion im Spieleszenario mehrere Modalitäten zur Eingabe sowie zur Ausgabe zur Verfügung stehen. Das vorgestellte System für die Spieleanwendung auf der Roboterplattform wurde teilweise in [258, 112] veröffentlicht.

7.2 Die Roboterplattform ELIAS

Die Roboterplattform basiert auf der von der Firma Metralabs entwickelten Roboterfamilie Scitos [259]. Die ELIAS-Plattform hat eine Höhe von 1,55 m und wiegt ca. 75 kg, ihr äußeres Erscheinungsbild ist an eine Mensch-ärgere-dich-nicht-Figur angelehnt, des Weiteren ist die Plattform vom TÜV Thüringen für den Einsatz in der Nähe von Menschen zertifiziert worden. Die Plattform besteht aus zwei unterschiedlichen Einheiten: einer Antriebseinheit (siehe Abbildung 7.1(a)) sowie einer Interaktionseinheit (siehe Abbildung 7.1(b)). Die wesentlichen Eigenschaften der Plattform werden nun kurz vorgestellt, weitere Details finden sich in [259].



(a) ELIAS-Antriebseinheit (b) ELIAS-Interaktionseinheit

Abbildung 7.1: Die zwei Einheiten der ELIAS-Plattform

Die Mobilität der ELIAS-Antriebseinheit wird mit einem Differentialantrieb realisiert, dabei gibt es zwei angetriebene Räder und ein zusätzliches Stützrad. Die Platt-

form kann eine Geschwindigkeit von bis zu 1,4m/s (5km/h) erreichen. Die Batterien für die Stromversorgung der Plattform sowie die Motoren befinden sich im unteren Teil der Antriebseinheit, somit hat die Plattform einen sehr tiefen Schwerpunkt, wodurch sie eine hohe Stabilität aufweist und nur sehr schwer kippbar ist. Um eine kollisionsfreie Navigation zu verwirklichen, besitzt die Plattform sowohl einen nach vorne ausgerichteten Laser-Scanner als auch einen Ring von 24 Sonarsensoren, die die Umgebung in einer Reichweite von 20 bis zu 300 cm rund um die Plattform überwachen. Zur Handhabung der Plattform befinden sich ein Industrie Personal Computer (PC), der ein Linux basiertes Betriebssystem hat, sowie eine Steuerungselektronik an Bord der Antriebseinheit. Ein 126×64 px großes blau-grünes Display informiert über den Ladezustand des Roboters, des Weiteren lassen sich mit diesem Display und einem Dreh-Drück-Steller einzelne Komponenten der Antriebseinheit (z. B. der Industrie-PC starten) steuern. Neben den Komponenten der Antriebseinheit sind auch einige Komponenten der Interaktionseinheit (z. B. der Roboterkopf) mit dem Industrie-PC der Antriebseinheit verbunden und lassen sich durch diesen steuern. Über eine Netzwerkbrücke ist der Industrie-PC der Antriebseinheit mit dem Computer der Interaktionseinheit verbunden.

Die ELIAS-Interaktionseinheit wird von einem Mac Mini betrieben, der ein Windows Betriebssystem besitzt. Neben dem Mac Mini bilden der Touchscreen, der Roboterkopf, Mikrofone und Lautsprecher die wesentlichen Komponenten der Interaktionseinheit. Der resistive Touchscreen hat eine Auflösung von 1024×768 px und ist ein *single touch display*, somit können nicht mehrere Berührungen gleichzeitig erkannt werden. Der Roboterkopf der Interaktionseinheit ist mit dem Industrie-PC der Antriebseinheit verbunden. Dieser Kopf besitzt einen Ring aus 32 LEDs, die unterschiedliche Effekte wie Blinken, laufende Lichter etc. zeigen können. Neben dem Leuchtring hat der Roboterkopf noch zusätzliche Augen, die bei der ELIAS-Plattform eine spezielle Modifikation erfahren haben (weitere Details hierzu finden sich in [260]). Die Mikrofone der Interaktionseinheit befinden sich im Gehäuse des Touchscreens, in diesem Gehäuse sind auch die Lautsprecher installiert. Die Lautsprecher, die Mikrofone und der Touchscreen sind mit dem Mac Mini der Interaktionseinheit verbunden, auf dem auch das Dialogsystem des ELIAS-Roboters läuft.

Daneben gibt es weitere Ausrüstungen mit technischen Geräten (z. B. Microsoft Kinect, zusätzliche Webcams etc.), die je nach Anwendungsfall entweder mit dem Computer der Interaktions- bzw. Antriebseinheit verbunden werden.

Das Konzept von wahrnehmenden und agierenden Modalitäten ist in Abschnitt 4.1.1 für den Menschen vorgestellt worden, es ist möglich dieses Konzept auch auf die Roboterplattform zu übertragen. Prinzipiell gibt es Überschneidungen zwischen den Modalitäten von Mensch und Roboter, beispielsweise kann man bei den wahrnehmenden Modalitäten folgende Ähnlichkeiten finden. Die ELIAS-Plattform kann ihre Umgebung visuell mithilfe von Kameras und dem Laser-Scanner erfassen sowie auditiv durch Mikrofone und Sonarsensoren. Zusätzlich können über den

Touchscreen Berührungen wahrgenommen werden. Für die agierenden Modalitäten gibt es auch Ähnlichkeiten zwischen dem Mensch und der Roboterplattform. Zunächst kann die ELIAS-Plattform Blickverhalten darstellen, dafür stehen auf der Plattform zwei Kamerasysteme zur Verfügung. Diese Kamerasysteme sollen menschliche Augen nachbilden, hierbei kommen Piezo-Aktuatoren für die Bewegung der Kamerasysteme zum Einsatz. Mit diesen Aktuatoren ist es möglich, Eigenschaften des menschlichen Auges in Hinsicht auf Geschwindigkeit und Beschleunigung zu übertreffen, somit können auch schnelle Sakkadenbewegungen darstellt werden (weitere Details finden sich in [260]). Sprach- bzw. Geräuschsynthese sowie autonome Navigation bilden zwei weitere agierende Modalitäten der ELIAS-Plattform, die Ähnlichkeiten mit menschlichen Modalitäten haben.

Neben den Gemeinsamkeiten für die Modalitäten zwischen dem Mensch und der ELIAS-Plattform gibt es auch Modalitäten, die nur der ELIAS-Plattform zur Verfügung stehen. Zur Interaktion kann die ELIAS-Plattform Inhalte auf dem Bildschirm darstellen, zusätzlich können verschiedene Leuchtmuster mithilfe des LED-Rings dargestellt werden.

Aufgrund der Vielzahl an verschiedenen Modalitäten ist es möglich, die Kommunikation zwischen Mensch und ELIAS-Plattform sowohl redundant als auch komplementär zu gestalten, um für den Menschen eine natürliche und intuitive Interaktion zu realisieren. Von dem Konzept der Modalitäten ist das ähnliche Konzept der Kanäle abzugrenzen. Bei der Kommunikation zwischen Mensch und Maschine können verschiedene Kanäle unterschieden werden, hierbei können zwei Kommunikationskanäle aber auf der gleichen Modalität beruhen. Beispielsweise kann die menschliche Stimme sowohl mit Sprache als auch mit dem Tonfall Informationen übermitteln.

7.3 Multimodale Interaktion in einem Spieleszenario

7.3.1 Motivation

Der Einsatz von Robotern hat in AAL vielseitige Anwendungsmöglichkeiten, ein kurzer Überblick darüber ist in Abschnitt 2.2.2 gegeben worden, auch Spielen stellt eine Tätigkeit dar, für die Roboter verwendet werden können. In [261] wird eine Roboterplattform namens *Maggie* vorgestellt, die in der Lage ist, sozial mit Menschen zu interagieren, dabei können auch Spiele von Bedeutung sein. Einer der bekanntesten Roboter zum Spielen ist der von Sony produzierte Roboterhund AIBO [52]. Diese Plattform wird in [262] bei der Therapie bzw. als Freizeitspiel für Personen mit Demenz eingesetzt. Ein weiteres bekanntes Beispiel aus dem Bereich der Robotik für die Therapie von Demenzpatienten ist die Roboterrobbe Paro [57]. Insbesondere für die Gesundheit und das Wohlbefinden von älteren Menschen können Spiele eine Rol-

le spielen. Diesem Sachverhalt wird mit sogenannten *Exergames*[1] Rechnung getragen, hierbei werden physische Übungen mit einem Videospiel kombiniert, da zur Steuerung des Spiels unterschiedliche körperliche Aktivitäten gemacht werden müssen. Weitere Informationen zu *Exergames* für ältere Menschen können in [263] gefunden werden.

Im Gegensatz zu gewöhnlichen Computersystemen (PC, Laptop, Tablet etc.), die im Heimbereich Anwendung finden, sind mobile Roboterplattformen in der Lage, proaktiv mit Menschen zu interagieren. Überdies können aufgrund der meist vielseitigen technischen Ausstattung Roboter unterschiedliche Interaktionsformen anbieten.

Im Folgenden wird ein Ansatz vorgestellt, bei dem man mit der Roboterplattform ELIAS das Spiel *Akinator* [264, 265] spielen kann. Beim *Akinator*-Spiel soll der Nutzer an eine bekannte Person denken, und das System versucht, die Person mittels gezielter Fragen zu erraten. Das Spiel wird hierbei nicht, wie üblich, mittels Tastatur oder Maus bedient, sondern es gibt eine multimodale Interaktion zwischen Mensch und Roboter.

7.3.2 Realisierung

Die Realisierung des *Akinator*-Spiels auf der ELIAS-Plattform erfolgt durch das Dialogsystem, das den Spielablauf steuert. Hierbei werden insbesondere die unterschiedlichen Eingabe- bzw. Ausgabemodalitäten herausgestellt. Als erstes erfolgt eine kurze Beschreibung des *Akinator*-Spiels, danach werden die verwendeten Komponenten sowie die Ablaufsteuerung beschrieben. Abschließend wird der gesamte Spielablauf einer *Akinator*-Spielrunde betrachtet.

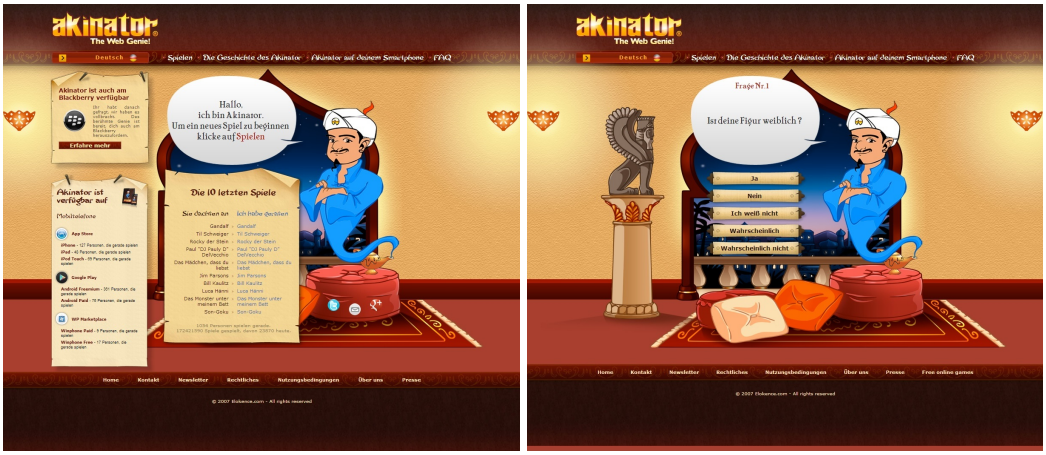
7.3.2.1 Das Spiel Akinator

Das Ziel des *Akinator*-Spiels ist es, dass eine bekannte Person erraten wird, an die der Nutzer denkt. Das System stellt dabei eine Reihe an geschickten Fragen, die mit fünf unterschiedlichen Möglichkeiten beantwortet werden können: *Ja*, *Nein*, *Ich weiß nicht*, *Wahrscheinlich* und *Wahrscheinlich nicht*. Daneben gibt es am Ende noch die Möglichkeit, falls sich die gesuchte Person nicht in der Datenbank befindet, diese hinzuzufügen [264]. Die ersten Schritte des *Akinator*-Spiels, wie man sie von der Internetseite [264] kennt, sind in Abbildung 7.2 dargestellt.

7.3.2.2 Verwendete Komponenten

Das *Akinator*-Spiel bietet die Möglichkeit zur multimodalen Interaktion. Dafür können unterschiedliche Möglichkeiten der ELIAS-Plattform zur Eingabe sowie zur Ausgabe verwendet werden, hierbei kommen kommerzielle Lösungen zum Einsatz sowie werden die entwickelten Methoden zur Gesten- und Mimikerkennung aufgegriffen. Das

[1] Ein Kunstwort aus *exercise* (Übung) und *games* (Spiele).



(a) Startseite (b) Erste Frage

Abbildung 7.2: Übersicht über das Spiel *Akinator* (Bildschirmaufnahmen von der Internetseite [264])

Konzept der RTDB von Abschnitt 4.2 wird verwendet, um die Gestenerkennung (statisch, dynamisch) sowie die Mimikerkennung in das System einzubinden. Im Folgenden werden kurz die wesentlichen Eigenschaften der beteiligten Interaktionskomponenten vorgestellt.

- Spracherkennung: Die Spracherkennung wird mit einem kommerziellen System realisiert (siehe [266] für weitere Details). Die Grammatik des Spracherkenners kann je nach Spielsituation angepasst werden, um somit die Robustheit in der Spracherkennung zu erhöhen.

- Statische Handgesten: Das System zur Erkennung von statischen Handgesten von Abschnitt 4.3 wird verwendet, um eine der fünf Antwortmöglichkeiten des *Akinator*-Spiels auswählen zu können. Für die Auswahl per statische Handgeste werden die fünf Antwortmöglichkeit mit den Zahlen *Eins* bis *Fünf* hinterlegt, siehe Abbildung 7.4(a).

- Dynamische Gesten: In Anlehnung an die *Exergames* wurde das System zur Erkennung von dynamischen Handgesten (siehe Abschnitt 4.4) abgeändert. Der Mensch hat vor Spielbeginn die optionale Möglichkeit, dem System fünf unterschiedliche dynamische Gesten der oberen Körperhälfte als Antwortmöglichkeiten beizubringen. Für die Merkmalsextraktion wird der Microsoft Kinect-Sensor verwendet. Die Realisierung des Gestenerkenners basiert auf HMMs, weitere Einzelheiten zu diesem System finden sich in [258].

- Kopfgesten: Anhand der Kopfgesten können zwei Antwortmöglichkeiten (*Ja*, *Nein*) bei der Spielaufforderung ausgewählt werden, hierfür wird das System von Abschnitt 4.5 verwendet.
- Sprachsynthese: Ähnlich wie bei der Spracherkennung wurde auch für die Sprachsynthese ein kommerzielles System verwendet (siehe [266] für weitere Details).
- Mimikerkennung: Ein Abwandlung des Systems zur Mimikerkennung von Kapitel 5 wurde in die RTDB-Architektur integriert. Die Ergebnisse einer HMM-Klassifikation können genutzt werden, um eine der sechs Mimikemotionen des Nutzers zu erkennen.
- LED-Ring: Der LED-Ring wird verwendet, um dem Nutzer anzuzeigen, ob die Spracherkennung aktiviert ist. Dies geschieht anhand eines laufenden LED-Rings. Des Weiteren wird der LED-Ring beim Trainieren der dynamischen Handgesten verwendet, hierbei dient der LED-Ring dazu, den Start bzw. das Ende eines Gestentrainings anzuzeigen.

7.3.2.3 Ablaufsteuerung

Das Dialogsystem ist für die Realisierung der Kommunikation zwischen Mensch und Roboter zuständig. Es gibt viele unterschiedliche Arten von Dialogsystemen, es lassen sich aber drei wesentliche Komponenten identifizieren: Eingangskanäle, Ausgangskanäle und der Dialogmanager. Die Eingangskanäle umfassen alle Arten von Daten, die in das Dialogsystem eingegeben werden können. Die Ausgangskanäle beschreiben die Möglichkeiten, mit denen das System mit dem Menschen kommuniziert. Der Dialogmanager sorgt für die konkrete Ablaufsteuerung der Dialogsituation, für das Spiel kommt ein Expertensystem zum Einsatz, das Fakten und Regeln benutzt, um durch unterschiedliche Dialogsituationen zu navigieren. Weitere Einzelheiten zum verwendeten Dialogsystem finden sich in [142].

Eingangskanäle für das *Akinator*-Spiel
Zur Bedienung des *Akinator*-Spiels auf der ELIAS-Plattform kann der Mensch auf zwei verschiedene Modalitäten zurückgreifen: die menschliche Stimme und Bewegungen. Die erste Modalität zur Eingabe bildet die Stimme. Ein Spracherkennungssystem, das auf der ELIAS-Plattform zum Einsatz kommt, kann die fünf unterschiedlichen Antwortmöglichkeiten erkennen. Daneben sind noch alternative Antwortmöglichkeiten im Vokabular hinterlegt worden, beispielsweise gibt es für die Möglichkeit *Ich weiß nicht*, noch Sätze wie *Ich habe keine Ahnung*, *Da weiß ich nicht Bescheid* etc. Der Mensch verfügt außerdem über die Möglichkeit sich Fragen des *Akinator*-Spiels erneut vorlesen zu lassen.

Mit der zweiten menschlichen Modalität zur Interaktion, den Bewegungen, können zwei unterschiedliche Modalitäten des ELIAS-Systems angesprochen werden. Zunächst kann auf dem Touchscreen eine Antwort ausgewählt werden, zudem können Gesten verwendet werden, um die Antwort darzustellen. Die Gesten werden anhand der installierten Kameras (Webcam, Microsoft Kinect) wahrgenommen. Das erste Gestenkonzept verwendet dynamische Gesten der oberen Körperhälfte, diese Gesten werden mithilfe des Kinect-Sensors erfasst. Die dynamischen Gesten können dem System vor dem Spielbeginn beigebracht werden. Das zweite Gestenkonzept verwendet die fünf statischen Handgesten von Abschnitt 4.3, um eine der fünf Antwortmöglichkeiten auszuwählen. Anhand der Kopfgestenerkennung können die Antwortmöglichkeiten *Ja* bzw. *Nein* gegeben werden.

Ausgangskanäle für das *Akinator*-Spiel
Das ELIAS-System kann drei verschiedene Modalitäten nutzen, um mit dem Menschen in der Spielsituation zu interagieren. Die einfachste Form zur Interaktion ist die Darstellung der Frage auf dem Bildschirm. Neben dieser einfachen Form der Darstellung wird die Frage auch mithilfe der Sprachsynthese dem Menschen vorgelesen. Des Weiteren kann der LED-Ring genutzt werden, um unterschiedliche Zustände des ELIAS-Systems zu signalisieren, beispielsweise wird die Aktivierung des Spracherkennungssystems dem Menschen über einen laufenden LED-Ring signalisiert. Im optionalen Fall, in dem dem ELIAS-Sytem dynamische Gesten beigebracht werden, wird der LED-Ring im Ablauf der Trainingsprozedur verwendet.

Ein Überblick über die beteiligten menschlichen Modalitäten und die verwendeten Systeme der ELIAS-Plattform zur Erkennung bzw. Interaktion ist in Abbildung 7.3 dargestellt.

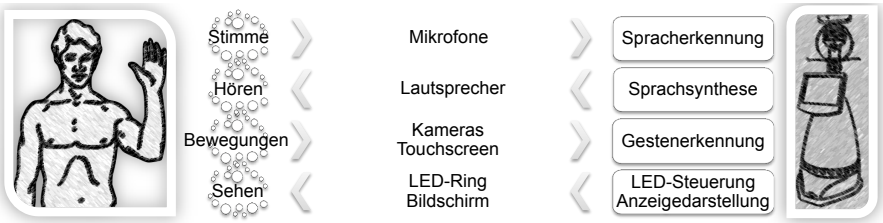


Abbildung 7.3: Überblick über die Interaktion zwischen Mensch und der ELIAS-Plattform im *Akinator*-Spiel

7.3.2.4 Spielablauf

Im Folgenden wird kurz der Ablauf einer *Akinator*-Runde mit der ELIAS-Plattform beschrieben. Bei diesem Ablauf gibt es den optionalen Schritt für das Lernen von dynamischen Gesten.

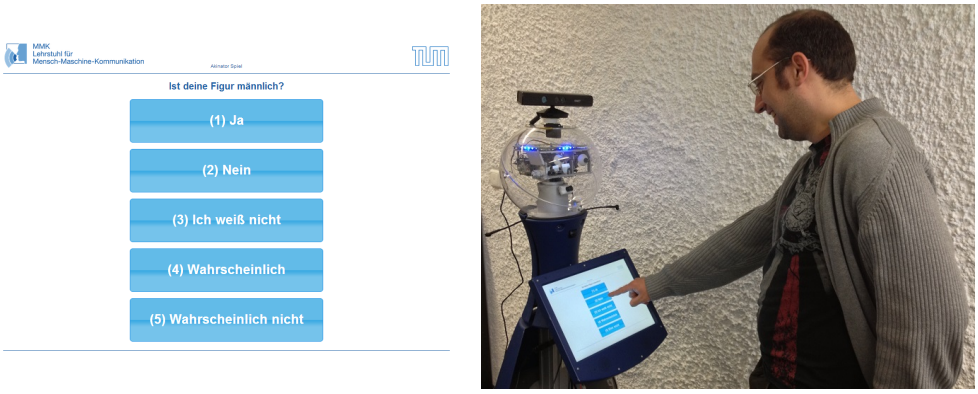
Einführung
Am Beginn erkennt der Roboter die Anwesenheit einer Person über die installierten Sonarsensoren. Anschließend fordert der Roboter die Person, die sich der Roboterplattform angenähert hat, auf, eine Runde *Akinator* zu spielen. In dieser Phase adaptiert das Dialogsystem das Vokabular des Spracherkenners auf Äußerungen zur Bestätigung (z. B. *Ja*, *Okay* etc.) sowie zur Ablehnung (z. B. *Nein*, *Nicht* etc.). Diese Adaption des Vokabulars wird vorgenommen, um die Robustheit der Spracherkennung zu erhöhen. Des Weiteren kann die Person anhand von Kopfgesten ihre Zustimmung bzw. Ablehnung für die Spielteilnahme signalisieren.

Optionales Training von dynamischen Gesten
In dieser optionalen Phase kann der Nutzer dem System fünf unterschiedliche dynamische Gesten beibringen, um die Beantwortung der Fragen mit einem leichten physischen Training zu kombinieren. Das ELIAS-System nutzt neben der Sprachausgabe auch den LED-Ring zur Steuerung des Trainingsablaufs.

Spielen
Nach den Vorbereitungen wird der Nutzer nun via Sprachausgabe aufgefordert an eine bestimmte bekannte Person zu denken. Anschließend werden per Sprachausgabe sowie per Bildschirmdarstellung unterschiedliche Fragen ausgegeben, um die gesuchte Person zu erraten, dabei kann der Nutzer per Sprache, Touchscreen sowie mit Gesten antworten. Die Antworten für die Spracherkennung werden um zusätzliche Aussagen erweitert, um die Interaktion noch natürlicher zu machen. Zusätzlich besteht die Möglichkeit, eine Frage erneut vom System vorlesen zu lassen. Daneben gibt das System auch Rückmeldung, falls eine Aussage des Nutzers nicht verstanden worden ist.

Abschluss
Nachdem das *Akinator*-System aufgrund der erhaltenen Antworten ein gewisses Maß an Konfidenz in Bezug auf die gesuchte Person erreicht hat, wird diese Vermutung dem Nutzer via Sprachausgabe mitgeteilt. Nun gibt es zwei Möglichkeiten: Entweder die Vermutung ist richtig und das *Akinator*-System war in der Lage, die richtige Person zu erraten. Dann kann der Nutzer das Spiel beenden oder eine weitere Runde spielen. Falls jedoch die Vermutung des *Akinator*-Systems falsch war, kann das Spiel weiter fortgesetzt werden.

Anhand des einfachen *Akinator*-Spieleszenarios kann gezeigt werden, wie eine multimodale Interaktion zwischen einem Menschen und einem Roboter umgesetzt werden kann. Abbildung 7.4 zeigt sowohl die grafische Benutzeroberfläche zum Spielen des *Akinator*-Spiels als auch eine Interaktion zwischen Mensch und ELIAS-Plattform.



(a) Grafische Benutzeroberfläche (b) Spiele-Interaktion

Abbildung 7.4: Interaktion mit der grafischen Benutzeroberfläche im *Akinator*-Spiel

7.4 Diskussion

In diesem Kapitel ist zunächst die Roboterplattform ELIAS beschrieben worden, welche viele Gemeinsamkeiten mit der Roboterplattform des ALIAS-Projektes hat. Prinzipiell ist die Roboterplattform ELIAS aufgrund der vielfältigen Ausstattung in der Lage mit dem Menschen in einer multimodalen Art und Weise zu interagieren, daher ist eine Betrachtung der Roboterplattform unter dem Aspekt von Modalitäten vorgenommen worden. Anschließend wurde eine exemplarische AAL-Anwendung beschrieben, die auf der ELIAS-Plattform entwickelt worden war. Diese Anwendung beschreibt, wie das *Akinator*-Spiel in einer multimodalen Art und Weise mit der ELIAS-Plattform gespielt werden kann. Aufgrund der Tatsache, dass die Spieleinteraktion im *Akinator*-Spiel eigentlich von kurzer Dauer ist, hat die Komponente zur Mimikerkennung nur einen geringen Beitrag bei der Ausgestaltung der multimodalen Mensch-Roboter-Interaktion, da das Auftreten einer der sechs Mimikemotionen eher unwahrscheinlich ist. Anhand dieser exemplarischen Spieleanwendung kann man sehen, dass die Roboterplattform aufgrund ihrer vielfältigen Ausstattung die Möglichkeit bietet, die Interaktion zwischen Mensch und Maschine multimodal zu gestalten. Neben der Spieleanwendung ist auch noch ein Gesundheitsüberwachungssystem (für weitere Details siehe [267]) auf der Roboterplattform integriert worden, somit zeigt sich, dass auf einer Roboterplattform prinzipiell viele unterschiedliche AAL-Anwendungen integriert und auf multimodale Art und Weise bedient werden können.

Kapitel 8

Zusammenfassung

Die vorliegende Arbeit befasste sich mit Aspekten der Mensch-Maschine-Kommunikation für *Ambient Assisted Living*. Der Fokus der Arbeit lag auf dem Bereich der nonverbalen Interaktion (Gesten, Mimik) sowie bildbasierten Verfolgungsmethoden. Exemplarisch wurde eine *Ambient Assisted Living*-Anwendung auf einer Roboterplattform realisiert, hierbei wurde unter anderem die Fähigkeit des Robotersystems zur multimodalen Mensch-Roboter-Interaktion betrachtet.

8.1 Beiträge und Ergebnisse

Nach einer Vorstellung des Begriffs *Ambient Assisted Living* und einer kurzen Beschreibung der wesentlichen Ursache (demografischer Wandel) für den Bedarf an *Ambient Assisted Living* wurden unter Berücksichtigung der Anforderungen von älteren Menschen an mögliche Technologien, *Ambient Assisted Living*-Anwendungen vorgestellt. Die Ausführungen zu *Ambient Assisted Living* enden mit einer kurzen Beschreibung des ALIAS-Projektes, in dem Teile dieser Arbeit entwickelt worden sind. Anschließend wurden kurz wesentliche Grundlagen zu Themen, die sich in mehreren Kapiteln dieser Arbeit wiederfinden, erläutert, hierbei wurden Graphische Modelle sowie die Durchführung und Bewertung von Experimenten betrachtet.

Nach der Einführung von *Ambient Assisted Living* sowie den Grundlagen wurden drei Ansätze zur Gestenerkennung vorgestellt. Mit diesen Ansätzen ließen sich statische und dynamische Handgesten sowie Kopfgesten erkennen. Die Gestenerkennungssysteme sind in eine einheitliche Software-Architektur eingebunden worden, um eine echtzeitfähige Verarbeitung zu ermöglichen. Das Dialogsystem war somit in der Lage, auf eine einheitliche Art und Weise auf die aktuellen und vergangenen Ergebnisse der Gestenerkennung zuzugreifen. Die Software-Architektur ermöglichte eine Partitionierung der Gestenerkennungssysteme, somit konnten einzelne Verarbeitungsschritte

für unterschiedliche Erkennungssysteme genutzt werden, dieses Vorgehen half bei der optimalen Ausnutzung der Ressourcen. Für die Erkennungssysteme hat sich gezeigt, dass eine gute Vorverarbeitung ein entscheidender Faktor für das Erzielen von guten Klassifikationsergebnissen ist. Weiterhin zeigte sich bei den Experimenten, dass sich durch den Einsatz von komplexeren Klassifikatoren die Ergebnisse verbessern ließen.

Im Gegensatz zu Gesten, die sich im Grunde genommen immer nur unvollständig beschreiben lassen, können mit der Mimik sechs universelle Emotionen ausgedrückt werden. Die Erstellung eines Mimikerkennungssystem beruhte auf Merkmalen, die anhand eines Gesichtsmodells gewonnen wurden, sowie Graphischen Modellen zur Klassifikation. Aufgrund des dynamischen Verlaufs für eine Mimikemotionen ist es sinnvoll diese Tatsache in das Erkennungssystem zu integrieren, daher sind unterschiedliche Methoden zur Merkmalsselektion verwendet worden, um die relevanten Merkmale zu erhalten. Es wurden sequentielle Verfahren zur Merkmalsselektion verwendet, sowie zwei Verfahren, die die Kullback-Leibler-Divergenz zur Bestimmung der relevanten Merkmale benutzten.

Bildbasierte Verfolgungsmethoden bildeten ein weiteres Einsatzgebiet für Graphische Modelle in dieser Arbeit. Zunächst wurde der CONDENSATION-Algorithmus als ein Graphisches Modell dargestellt, basierend auf dieser Arbeit wurden unterschiedliche Erweiterungen für den bildbasierten Verfolgungsprozess entwickelt. Die erste Erweiterung bezog sich auf die Kombination des bildbasierten Verfolgungsprozesses mit einem Klassifikationsschritt in einem Graphischen Modell. Die beiden anschließenden Erweiterungen haben den Inferenzprozess von Graphischen Modellen genutzt, um das Bewegungsmodell sowie die Partikel-Gewichtungsfunktion adaptiv an die gegenwärtige Beobachtung anzupassen. Beide Verfahren wurden in einem einheitlichen Graphischen Modell kombiniert.

Zum Abschluss der Arbeit wurde eine beispielhafte Anwendung für *Ambient Assisted Living* auf einer Roboterplattform vorgestellt. Die Vorstellung der Roboterplattform beschrieb, neben den technischen Komponenten, auch den Aspekt von Modalitäten. Eine multimodale Mensch-Roboter-Interaktion in einem Spieleszenario bildete die exemplarische *Ambient Assisted Living*-Anwendung.

8.2 Ausblick

Die hier vorgestellten Ansätze haben gezeigt, dass Aspekte der Mensch-Maschine-Kommunikation bei der Gestaltung von Anwendungen für *Ambient Assisted Living* von entscheidender Bedeutung sein können. Denn wenn mithilfe von modernen technischen Lösungen die Interaktion zwischen Mensch und Maschine natürlich und intuitiv gestaltet werden kann, fällt es auch älteren Menschen leichter neue Anwendungen aus dem Bereich der Informations- und Kommunikationstechnologien anzunehmen und zu verwenden. Für die Verwirklichung einer natürlichen und intuitiven Mensch-

Maschine-Interaktion können nonverbale Kommunikationsmittel (Gesten, Mimik) dazu beitragen, dass technische Systeme zum einen besser die Intentionen des Nutzers nachvollziehen können und zum anderen auch auf mehreren Wegen die Interaktion mit dem Menschen ermöglichen. Die Bedienung mithilfe von Gesten findet heute vermehrt Einsatz bei technischen Geräten, dennoch sind es meist sehr einfache Gesten und können nicht je nach Kontext verschieden sein. Auch die Auswertung von Mimik hat Einzug in technische Geräte gefunden, beispielsweise können heutige Kameras lächelnde Personen erkennen, dennoch haben die Ergebnisse aus der Datenbank-Kreuzvalidierung gezeigt, dass es in Zukunft auch noch weiterer Bemühungen bedarf, um robuste Mimikerkennungssysteme für Alltagssituationen zu erschaffen. Der Einsatz von Graphischen Modellen im Bereich von bildbasierten Verfolgungsmethoden hat sich als eine vielversprechende Darstellungsform für Bildverarbeitungsprobleme erwiesen, dennoch können insbesondere im Bereich der Graphischen Modelle Verbesserungen gemacht werden. Für Graphische Modelle gibt es die grundsätzlichen Probleme, da die Anzahl der Parameter sehr schnell ansteigen kann sowie die Tatsache, dass es für dynamische Systeme keine Möglichkeit gibt, die Struktur eines Graphen zu lernen, da *Brute-Force-Methoden*, die bei statischen Netzen zum Einsatz kommen können, für dynamische nicht mehr geeignet sind. Die Robotik bietet prinzipiell ein breites Spektrum an verschiedenen Anwendungsmöglichkeiten, in dieser Arbeit ist die Servicerobotik betrachtet worden. Mit Robotersystemen kann man die soziale Interaktion zwischen Mensch und Maschine auf eine multimodale Art und Weise realisieren. Dabei ist aber darauf zu achten, die unterschiedlichen Interaktionen so zu gestalten, dass der Mensch einen natürlichen und intuitiven Zugang zum Robotersystem erhalten. Erste Schritte zur Verwirklichung einer multimodalen Mensch-Roboter-Interaktion sind gemacht worden, dennoch sind noch zahlreiche Herausforderungen zu lösen, bis komplexe multimodale Robotersysteme als eine Art Haushaltsroboter Einzug in die häusliche Umgebung des Menschen finden.

Anhang A

Wahrscheinlichkeitstheorie

A.1 Einführung

Die Wahrscheinlichkeitstheorie ist ein Gebiet der Mathematik, das sich mit Zufallsexperimenten beschäftigt. Das Ziel der Wahrscheinlichkeitstheorie ist es, die Zufallsexperimente mit mathematischen Modellen sowie Methoden zu beschreiben. Eng verwandt mit dem Gebiet der Wahrscheinlichkeitstheorie ist das Gebiet der Statistik, dessen Ziel es ist, Daten aus experimentellen Befunden mit entsprechenden Modellen beschreiben zu können. Diese gefundenen Modelle sollen helfen, Entscheidungen in unsicheren Situationen zu treffen.

A.2 Definition

Im Folgenden werden kurz die wichtigsten Begriffe aus der Wahrscheinlichkeitstheorie basierend auf [268, 66] eingeführt.

Definition 1 (Experiment)
Ein Experiment $\mathcal{E}$ hat bestimmte Eigenschaften: Es ist wiederholbar (theoretisch unendlich oft). Verschiedene Wiederholungen (Versuche) des Experiments liefern unterschiedliche Ergebnisse.

Definition 2 (Ergebnis)
Der Ausgang eines Experiments $\mathcal{E}$ liefert ein Ergebnis ω (Elementarereignis).

Definition 3 (Ereignis)
Ein Menge an Ergebnissen (Elementarereignissen) $\omega_1, \ldots, \omega_n$ kann als ein Ereignis $\mathcal{A}$ zusammengefasst werden.

Definition 4 (Ergebnismenge)
Die Ergebnismenge Ω *eines Zufallsexperiments* $\mathcal{E}$ *ist die nicht-leere Menge oder Sammlung einer zählbaren Menge von Ergebnissen* $\{\omega_1, \ldots, \omega_n\}$.

Die grundlegenden Konzepte – Experiment, Ergebnis, Ereignis, Ergebnismenge – der Wahrscheinlichkeitstheorie sind eingeführt worden, dennoch fehlt ein wichtiger Punkt: das Zuweisen von Wahrscheinlichkeiten auf die Ergebnisse bzw. Ereignisse. Dafür wird eine bestimmte Menge $\mathcal{S}$ aus der gesamten Ergebnismenge Ω ausgewählt, dieser Menge $\mathcal{S}$ von Ereignissen kann eine Wahrscheinlichkeit zugewiesen werden.

Trotz der Tatsache, dass das Auftreten der Wahrscheinlichkeit und ihre Verwendung als eine Maßeinheit im Alltag durchaus üblich ist, gibt es zwei verschiedene Arten von Definitionen: den klassischen (auch *frequentistischen*) Wahrscheinlichkeitsbegriff als auch den *Bayes'schen* Wahrscheinlichkeitsbegriff.

Nach der klassischen Auffassung bieten Wahrscheinlichkeiten auf der einen Seite eine Beschreibung für Häufigkeiten von wiederholbaren zufälligen Ereignissen (z. B. das Werfen einer Münze), während auf der anderen Seite die Bayes'sche Auffassung eine Beschreibung für die Quantifizierung von Unsicherheiten (z. B. Wahrscheinlichkeit dafür, dass morgen die Sonne scheint) liefert.

Die Zuordnung von Wahrscheinlichkeiten auf Ergebnisse oder Ereignisse hat drei axiomatischen Definitionen zu entsprechen, die Andrei Kolmogorow in [69] 1933 vorstellte. Im Folgenden werden kurz die drei Axiome (Nichtnegativität, Normalisierung, σ-Additivität) für einen Wahrscheinlichkeitsraum vorgestellt, weitere Details finden sich in [268].

Definition 5 (Wahrscheinlichkeitsraum) *Eine Funktion* Pr *über die Menge* $(\Omega, \mathcal{S})$ *beschreibt die Zuweisungen von realen Werten von* $\mathcal{S}$*, die folgenden drei Axiomen gehorcht.*

Axiom 1 (Nichtnegativität)
$\Pr\{\mathcal{A}\} \geq 0,\ \mathcal{A} \subseteq \mathcal{S}$.

Axiom 2 (Normalisierung)
$\Pr\{\Omega\} = 1$.

Axiom 3 (σ-Additivität)
Falls die Ereignisse $\mathcal{A}, \mathcal{B} \in \mathcal{S}$ *und* $\mathcal{A} \cap \mathcal{B} = \varnothing$*, dann gilt* $\Pr(\mathcal{A} \cup \mathcal{B}) = \Pr(\mathcal{A}) + \Pr(\mathcal{B})$.

A.3 Unabhängigkeit

A.3.1 Statistische Unabhängigkeit

Der Begriff der statistischen Unabhängigkeit bezieht sich auf Ereignisse bzw. Ergebnisse, bei denen der jeweilige Ausgang eines Experiments keinen Einfluss auf den

Ausgang des anderen Experiments hat. Für zwei Ereignisse $\mathcal{A}$ und $\mathcal{B}$ ergibt sich somit für die bedingte Wahrscheinlichkeit von Ereignis $\mathcal{A}$, wenn das Ereignis $\mathcal{B}$ gegeben ist:

$$\Pr(\mathcal{A}|\mathcal{B}) = \Pr(\mathcal{A}), \tag{A.1}$$

unter Berücksichtigung der Definition von bedingter Wahrscheinlichkeit

$$\Pr(\mathcal{A}|\mathcal{B}) = \frac{\Pr(\mathcal{A},\mathcal{B})}{\Pr(\mathcal{B})}, \tag{A.2}$$

ergibt sich dieser Zusammenhang für die Verbundwahrscheinlichkeit von $\mathcal{A}$ und $\mathcal{B}$:

$$\Pr(\mathcal{A},\mathcal{B}) = \Pr(\mathcal{A})\Pr(\mathcal{B}). \tag{A.3}$$

Beispielsweise wird beim zweimaligen Werfen eines fairen Würfels (alle sechs Seiten sind gleich wahrscheinlich) der Ausgang des zweiten Wurfs (bzw. Experiments) nicht vom Ergebnis des ersten Wurfs (bzw. Experiments) beeinflusst. Diese Definition führt zu der statistischen Unabhängigkeit von Ereignissen bzw. Ergebnissen.

A.3.2 Bedingte Unabhängigkeit

Obwohl der oben definierte Begriff von statistischer Unabhängigkeit nützlich ist, tritt dieser Zusammenhang leider nicht sehr oft ein [66]. Dennoch gibt es noch die bedingte Unabhängigkeit für zwei Ereignisse bzw. Ergebnisse $\mathcal{A}$ und $\mathcal{B}$, wenn ein drittes Ereignis bzw. Ergebnis $\mathcal{C}$ beobachtet wird. Dieser Zusammenhang lässt sich folgendermaßen ausdrücken:

$$\Pr(\mathcal{A}|\mathcal{B},\mathcal{C}) = \Pr(\mathcal{A}|\mathcal{C}), \tag{A.4}$$

eine weitere Notation für bedingte Unabhängigkeit ist gegeben mit dem Symbol $\perp\!\!\!\perp$:

$$\mathcal{A} \perp\!\!\!\perp \mathcal{B} \mid \mathcal{C}. \tag{A.5}$$

Daneben gibt es folgende Zusammenhänge (siehe [66]), die für bedingte Unabhängigkeit gelten, dabei wird aber nicht mehr von Ereignissen bzw. Ergebnissen ausgegangen, sondern von ZVs, die im Abschnitt A.4 kurz vorgestellt werden:

- Symmetrie:

$$(X \perp\!\!\!\perp Y \mid Z) \Rightarrow (Y \perp\!\!\!\perp X \mid Z) \tag{A.6}$$

- Zerlegung:

$$(X \perp\!\!\!\perp Y,W \mid Z) \Rightarrow (X \perp\!\!\!\perp Y \mid Z) \tag{A.7}$$

- Schwache Vereinigung:

$$(X \perp\!\!\!\perp Y, W \mid Z) \Rightarrow (X \perp\!\!\!\perp Y \mid Z, W) \tag{A.8}$$

- Kontraktion:

$$(X \perp\!\!\!\perp W \mid Z, Y) \;\&\; (X \perp\!\!\!\perp Y \mid Z) \Rightarrow (X \perp\!\!\!\perp Y, W \mid Z) \tag{A.9}$$

- Schnitt:

$$(X \perp\!\!\!\perp Y \mid Z, W) \;\&\; (X \perp\!\!\!\perp W \mid Z, Y) \Rightarrow (X \perp\!\!\!\perp Y, W \mid Z) \tag{A.10}$$

Gleichung A.10 gilt, wenn alle Verteilungen positiv sind, siehe [66] für weitere Details.

A.4 Zufallsvariablen

Bei vielen Experimenten ist man nicht direkt an dem Ergebnis ω oder dem Ereignis $\mathcal{A}$, sondern vielmehr an Eigenschaften bzw. einer Funktion $X(\omega)$ interessiert, dabei hilft das Konzept der Zufallsvariablen (RVs). Beispielsweise lassen sich die sechs Ergebnisse $(\omega_1, \ldots, \omega_6)$ eines Würfelwurfs in einer ZV zusammenfassen. Formal gesehen ist eine ZV eine Funktion, die die Ergebnisse $\omega \in \Omega$ auf einen Wert abbildet.

Definition 6 (Zufallsvariable)
Funktionen $X : \Omega \to \mathcal{X}$, die Ergebnisse der Ergebnismenge Ω in einer anderen Menge $\mathcal{X}$ abbilden, heißen Zufallsvariablen.

Im Folgenden beschreiben Großbuchstaben eine ZV, während Kleinbuchstaben eine konkrete Realisierung des Experiments für $\omega \in \Omega$ beschreiben, somit gilt $X(\omega) = x$.

Basierend auf diesem Konzept kann eine Notation für die Wahrscheinlichkeitsverteilung eingeführt werden. Unten wird nur die Definition für diskrete ZVs betrachtet, die auch Wahrscheinlichkeitsfunktion genannt wird, während man bei kontinuierlichen ZVs von einer Wahrscheinlichkeitsdichtefunktion spricht.

Definition 7 (Wahrscheinlichkeitsfunktion)
$p(x_i) = P(X = x_i) = \Pr\{\omega \in \Omega : X(w) = x_i\}$

Weitere ausführliche Beschreibungen zur Wahrscheinlichkeitstheorie und Zufallsvariablen finden sich in [268, 66].

Anhang B

Graphentheorie

B.1 Einführung

Viele Probleme aus unterschiedlichen Forschungsbereichen (Regelungstechnik, Sozialwissenschaften, Nachrichtentechnik etc.) können mit bildhaften Beschreibungen verständlicher dargestellt werden als nur anhand von schriftlichen Beschreibungen. Graphen können nicht nur eine exakte, sondern auch kompakte Problembeschreibung liefern, wobei die Abhängigkeiten zwischen den einzelnen Bestandteilen des Graphen leicht erfasst werden können. Wesentliche Konzepte der Graphentheorie für GMs werden hier basierend auf [269] kurz erläutert, weitere Details finden sich in [61, 269, 65].

B.2 Grundlagen

Definition 8 (Graph) *Ein Graph* $\mathcal{G} = (V, E)$ *ist ein Paar von einer endlichen Menge von Knoten* V *mit einer endlichen Menge von Kanten* E*, dabei bilden die Kanten die Teilmenge aller geordneten Paare* $E \subseteq V \times V$.

Die Menge V und die Menge E eines $\mathcal{G}$ wird mit folgenden Symbolen beschrieben: Die N-Knoten mit $v_1, v_2, ..., v_n$ somit ist der $\mathcal{G}$ von N-ter Ordnung. Die M-Kanten werden mit $e_1, e_2, ..., e_m$ beschrieben. Die Verbindung $e_k = (v_i, v_j)$ mittels einer Kante e_k zwischen Knoten v_i und v_j kann entweder gerichtet oder ungerichtet sein.

Es gibt verschiedene Arten von Graphen: Ein *Multigraph* kann Selbst-Schleifen ($e_k = (v_i, v_i)$) und bzw. oder parallele Kanten (mehr als eine Kante zwischen den Knoten v_i und v_j) aufweisen. Ein *schlichter Graph* hat dagegen weder Selbst-Schleifen noch parallele Kanten. *Vollständige Graphen* bestehen aus Knoten, wobei jedes Knotenpaar miteinander verbunden ist. Wenn eine Teilmenge von Knoten *vollständig* miteinander verbunden ist, dann bilden diese Knoten einen *vollständigen Teilgraphen*. Wenn eine

beliebige Kante zu den *vollständigen Teilgraphen* hinzugefügt wird und dieser nicht mehr vollständig ist, ist der Teilgraph maximal und man spricht von einer *Clique*.

Eine gerichtete Verbindung zwischen den Knoten v_i und v_j, enthält die Kante $e_k = (v_i, v_j)$, aber nicht die Kante $e_l = (v_j, v_i)$. Die Richtung der Kante wird durch die Anordnung der Knoten in der Kante $e_k = (v_i, v_j)$ gegeben, wobei v_i der Ursprungsknoten ist und v_j der Zielknoten. Diese direkte Verbindung wird mit $v_i \to v_j$ dargestellt. Andererseits ist für ungerichtete Kanten sowohl die Kante $e_k = (v_i, v_j)$ als auch $e_l = (v_j, v_i)$ in der Menge der Kanten E enthalten. Die ungerichtete Verbindung wird mit $v_i \sim v_j$ dargestellt. In diesem Fall sind v_i und v_j *Nachbarn*. Die Menge der Nachbarn von einem Knoten v_i wird mit $ne(v_i)$ zusammengefasst. Für gerichtete Kanten kann anhand der Kante zwischen $v_i \to v_j$ zwischen dem Elternteil v_i und dem Kind v_j unterschieden werden. Für einen Knoten v_i wird die Menge seiner Eltern bzw. die Menge seiner Kinder folgendermaßen zusammengefasst $pa(v_i)$ bzw. $ch(v_i)$. Der *Rand* $bd(v_i)$ eines Knotens v_i ist die Menge der Eltern $pa(v_i)$ und Nachbarn $ne(v_i)$. Die *geschlossene Hülle* $cl(v_i)$ eines Knotens v_i ist der *Rand* des Knoten und der Knoten selbst.

Für eine allgemeine Definition dieser Zusammenhänge kann anstatt des Knotens v_i, auch eine Knotenmenge $v^s \in V$ betrachtet werden, somit ergibt sich:

$$ne(v^s) = \bigcup_{v_i \in v^s} ne(v_i) \backslash v^s \tag{B.1}$$

$$pa(v^s) = \bigcup_{v_i \in v^s} pa(v_i) \backslash v^s \tag{B.2}$$

$$ch(v^s) = \bigcup_{v_i \in v^s} ch(v_i) \backslash v^s \tag{B.3}$$

$$bd(v^s) = pa(v^s) \cup ne(v^s) \tag{B.4}$$

$$cl(v^s) = bd(v^s) \cup v^s = pa(v^s) \cup ne(v^s) \cup v^s. \tag{B.5}$$

Die *Markov-Decke* $bl(v_i)$ eines Knotens v_i besteht aus seinen Eltern $pa(v_i)$, seinen Kindern $ch(v_i)$ und den anderen Eltern der Kinder $ch(v_i)$ (aber ohne v_i), somit lässt sich die *Markov-Decke* folgendermaßen beschreiben:

$$bl(v_i) = pa(v_i) \cup ch(v_i) \cup \{v_j : ch(v_j) \cap ch(v_i) \neq \varnothing\}. \tag{B.6}$$

Ein *gerichteter Graph* $\mathcal{G}^{\rightarrow}$besteht nur aus gerichteten Kanten, während andererseits ein *ungerichteter Graph* nur aus ungerichteten Kanten besteht.

Zwischen zwei Knoten v_{start} und v_{end} gibt es einen *Pfad* der Länge n, falls die Sequenz $v_{start} = v_0, \ldots, v_n = v_{end}$ folgendermaßen an den Kanten $(v_{i-1}, v_i) \in E, \forall i \in 1, \ldots, n$ verläuft. Falls eine der Kanten gerichtet ist, spricht man von einem *gerichteten Pfad*.

Zwei Knoten v_i und v_j sind miteinander *verbunden*, dargestellt mit $v_i \rightleftharpoons v_j$, falls es sowohl den Pfad von v_i nach v_j gibt (dargestellt durch $v_i \mapsto v_j$), als auch den Pfad von v_j nach v_i. Besitzt ein ungerichteter Graph für jedes Knotenpaar einen Pfad, so spricht

man von einem *verbundenen Graphen*. Die Knoten V können in disjunkte Mengen eingeteilt werden, welche man als *verbundene Komponenten* bezeichnet. Eine *verbundene Komponente* beschreibt eine Knotenmenge, in welcher es einen Pfad für jeden Knoten zu jedem anderen Knoten der Teilmenge gibt. Im Gegensatz zu einem Pfad beschreibt ein *Weg* der Länge n zwischen den Knoten v_{start} und v_{end} eine Sequenz $v_{start} = v_0, \ldots, v_n = v_{end}$, wobei es die Kanten $(v_{i-1}, v_i), (v_i, v_{i-1}) \forall i \in 1, \ldots, n$ gibt.

Falls ein Pfad der Länge n denselben Knoten als Start- und Endpunkt hat, so ist der Pfad ein *n-Zyklus*. Eine *Sehne* dieses Zyklus ist ein Paar von nicht benachbarten Knoten (v_i, v_j), welche eine ungerichtete Kante $v_i \sim v_j$ haben. Ein ungerichteter Graph $\mathcal{G}$ ist *trianguliert*, falls jeder Zyklus in $\mathcal{G}$ mit einer Länger von $n \geq 4$ eine Sehne hat.

Eine *gerichteter n-Zyklus* wird durch einen gerichteten Pfad bestimmt. Im Gegensatz zu zyklischen Pfaden besitzt ein *azyklischer Pfad* keine Zyklen. Ein *azyklischer gerichteter Graph* $\mathcal{G}^D$ ist ein gerichteter Graph, der keinerlei Zyklen besitzt.

Ein Knoten v_i hat *Vorfahren* $an(v_i)$, falls ein gerichteter Pfad von jedem Vorfahren v_j zum Knoten v_i führt, aber kein gerichteter Pfad von v_i zu irgendeinem Vorfahren v_j. Neben den Vorfahren hat ein Knoten v_i *Nachfahren* $de(v_i)$, falls es einen gerichteten Pfad vom Knoten v_i zu jedem Nachfahren v_j gibt, aber kein gerichteter Pfad von irgendeinem v_j zum Knoten v_i existiert. Die *Nicht-Nachfahren* von einem Knoten v_i $nd(v_i)$ sind alle Knoten des Graphen $\mathcal{G}$, aber ohne die Menge der Nachfahren $de(v_i)$ und den Knoten v_i selbst. Für eine allgemeine Definition kann, wie vorhin, eine Knotenmenge $v^s \in V$ betrachtet werden:

$$an(v^s) = \{v_i : v^s \mapsto v_i, v^i \not\mapsto v^s\} \tag{B.7}$$

$$de(v^s) = \{v_i : v^s \not\mapsto v_i, v^i \mapsto v^s\} \tag{B.8}$$

$$nd(v^s) = V \backslash (de(v^s) \cup v^s). \tag{B.9}$$

Eine Teilmenge von Knoten $v^c \subseteq V$ ist ein *Separator* von zwei Knotenteilmengen $v^a \subseteq V$ und $v^b \subseteq V$, falls es keine Wege von irgendeinem Knoten $v_a \in v^a$ zu irgendeinem Knoten $v_b \in v^b$ gibt, die nicht über zumindest einen Knoten $v_c \in v^c$ führen. Eine Separatormenge v^c ist *minimal*, falls die reduzierte Teilmenge v^{c_r}, welche man erhält, wenn die Teilmenge der ursprünglichen Separatorknoten v^c um einen beliebigen Knoten v_c reduziert wird, keine gültige Separatormenge für die zwei Teilmengen v^a und v^b ist.

B.3 Der Verbundbaum

Ein Baum ist eine besondere Art von Graphen $\mathcal{G}$, welche bestimmte Bedingungen erfüllen: Der Graph eines Baumes ist ungerichtet, besitzt keine Zyklen, und der Graph ist verbunden, das heißt es existiert ein Pfad von jedem Knoten $v_i \in V$ zu jedem anderen Knoten $v_j \in V, i \neq j$. Abhängig von der Struktur des Ausgangsgraphen, auf welchen der *Verbundbaum* aufgestellt werden soll, müssen folgende Schritte zuerst erledigt werden: Moralisierung und Triangulation.

Die *Moralisierung* des Graphen $\mathcal{G}$ beginnt damit, dass die Eltern $pa(v_i)$ eines Knotens v_i mit einer ungerichteten Kante miteinander verbunden werden, falls es noch keine Verbindung zwischen ihnen gibt. Nachdem die Eltern miteinander verbunden worden sind, werden alle gerichteten Kanten in ungerichtete Kanten umgewandelt.

Die *Triangulation* des Graphen $\mathcal{G}$ ist notwendig, falls es nicht für jeden Zyklus mit $n \geq 4$ eine Sehne gibt. Durch die Hinzugabe von zusätzlichen Kanten zu den Graphen kann die Struktur des Graphen in einen *chordalen* oder auch *triangulierten Graphen* überführt werden. Dieser Prozess heißt Triangulation und hat in der Regel kein exaktes Resultat, da es möglich ist, die zusätzlichen Kanten auch zwischen anderen Knoten einzusetzen. Die Triangulation kann mittels *maximum cardinality search algorithm* [270] durchgeführt werden, meistens fügt dieser Algorithmus aber mehrere zusätzliche Kanten als nötig ein. Zusätzlich erfüllt die Triangulation des Graphen die sogenannte *running intersection property*: Die Schnittmenge zweier Cliquen $\mathcal{C}_1$ und $\mathcal{C}_2$ ist enthalten in jeder Clique $\mathcal{C}_i$, die sich auf dem Pfad zwischen $\mathcal{C}_1$ und $\mathcal{C}_2$ befindet. Ein Graph, der diese Eigenschaft erfüllt, heißt *Verbundbaum*. Alle Cliquen, die nach dem Prozess der Moralisierung und Triangulation entstanden sind, formen nun den *Verbundbaum*, diese Cliquen werden auch Cluster genannt. Zwischen zwei Cluster wird ein sogenannter Separator eingefügt, dieser Separator enthält die Knoten, welche sich in der Schnittmenge von beiden Clustern befinden.

Der Verbundbaum kann folgendermaßen aufgestellt werden [269]:

Algorithmus 5 Verbundbaum

Bei den Cliquen $(\mathcal{C}_1, ..., \mathcal{C}_n)$ des geordneten und triangulierten Graphen $\mathcal{G}$ gilt die *running intersection property*:

1. Assoziiere einen Knoten des Verbundbaumes mit jeder Clique $\mathcal{C}_i$.
2. Für $i = 2, ..., n$, füge eine Kante zwischen die Knoten $\mathcal{C}_i$ und $\mathcal{C}_j$ ein, wobei für j gilt: $j \in \{1, ..., i-1\}$; des Weiteren gilt:

$$\mathcal{C}_i \cap (\mathcal{C}_1 \cup ... \cup \mathcal{C}_{i-1}) \subseteq \mathcal{C}_j.$$

Abkürzungsverzeichnis

2-D	zweidimensional
3-D	dreidimensional
AAL	Ambient Assisted Living
AAM	Aktives-Erscheinungs-Modell engl. Active Appearance Model
AIBO	Artificial Intelligence roBOt
ALIAS	Adaptable Ambient LIving ASsistant
AmI	Ambient Intelligence
Asimo	Advanced Step in Innovative Mobility
ASM	Aktives-Form-Modell engl. Active Shape Model
ASTROMOBILE	Assistive SmarT RObotic platform for indoor environments: MOBILity and intEraction
BCI	Brain Computer Interface
BN	Bayes'sches Netz
CHARLI	Cognitive Humanoid Autonomous Robot with Learning Intelligence
DBN	Dynamisches Bayes'sches Netz
DOMEO	domestic robot for elderly assistance
ELIAS	Enhanced Living Assistant
EM	Expectation-Maximization

EmFACS	Emotionales Gesichtsbewegungs-Kodierungssystem engl. Emotional Facial Action Coding System
EU	Europäische Union
EUROPOP2008	Eurostat Bevölkerungsprognosen 2008 engl. Eurostat Population Projections 2008
EUROPOP2010	Eurostat Bevölkerungsprognosen 2010 engl. Eurostat Population Projections 2010
FACS	Gesichtsbewegungs-Kodierungssystem engl. Facial Action Coding System
FAPs	Gesichtsanimationsparameter engl. facial animation parameters
GM	Graphisches Modell
HMM	Hidden-Markov-Modell
HOG	Histogramm orientierter Gradienten engl. Histogram of Oriented Gradients
i. i. d.	unabhängig und identisch verteilt engl. independent and identically distributed
IKT	Informations- und Kommunikationstechnologie
k-NN	k-Nächste-Nachbarn engl. k-nearest neighbor
KI	Künstliche Intelligenz
LDA	Lineare Diskriminanzanalyse engl. Linear Discriminant Analysis
MD	Mahalanobis-Distanz
MLE	Maximum Likelihood-Lernen engl. maximum-likelihood estimation
MMI	Mensch-Maschine-Interaktion
MRF	Markov Random Field
OTE	Object Tracking Error
PaPeRo	Partner-type-Personal-Robot
PC	Personal Computer

PCA	Hauptkomponentenanalyse engl. Principal Component Analysis
px	Pixel
QDA	Quadratische Diskriminanzanalyse engl. Quadratic Discriminant Analysis
R	Erkennungsrate
RTDB	Realzeitdatenbasis engl. real-time database
SBS	Sequentielle Rückwärts-Auswahl engl. Sequential Backward Selection
sentha	Seniorengerechte Technik im häuslichen Alltag
SFS	Sequentielle Vorwärts-Auswahl engl. Sequential Forward Selection
SVM	Stützvektormethode engl. Support Vector Machine
TRDR	Tracker Detection Rate
WDF	Wahrscheinlichkeitsdichtefunktion
WF	Wahrscheinlichkeitsfunktion
WHO	Weltgesundheitsorganisation engl. World Health Organziation
ZV	Zufallsvariable

Literaturverzeichnis

[1] K. GIANNAKOURIS. **Ageing characterises the demographic perspectives of the European societies**, 2008.

[2] EUROSTAT. **Work session on demographic projections**, 2010.

[3] M. EISENMENGER, O. PÖTZSCH, UND B. SOMMER. **11. koordinierte Bevölkerungs-Vorausberechnung**, 2006.

[4] STATISTISCHES BUNDESAMT. **12. koordinierte Bevölkerungs-Vorausberechnung**, 2009.

[5] STATISTISCHE ÄMTER DES BUNDES UND DER LÄNDER, Herausgeber. **Demografischer Wandel in Deutschland: Heft 1: Bevölkerungs- und Haushaltsentwicklung im Bund und in den Ländern**. Statistisches Bundesamt, Wiesbaden, 2011.

[6] ONLINE. **Eurostat**. http://epp.eurostat.ec.europa.eu, 2012. besucht im Januar 2012.

[7] ONLINE. **Das Demographie Netzwerk**. http://demographie-netzwerk.de, 2013. besucht im März 2013.

[8] H. STEG, H. STRESE, C. LOROFF, J. HULL, UND S. SCHMIDT. **Europe Is Facing a Demographic Challenge Ambient Assisted Living Offers Solutions**. Report, 2006.

[9] S. JÄHNICHEN. **VDE-Hintergrundpapier Intelligente Assistenz-Systeme im Dienst für eine reife Gesellschaft**, 2008.

[10] ONLINE. **AAL-Deutschland**. http://www.aal-deutschland.de, 2012. besucht im Januar 2012.

[11] S. MEYER UND BMBF-VDE-INNOVATIONSPARTNERSCHAFT AAL. **AAL in der alternden Gesellschaft: Anforderungen, Akzeptanz und Perspektiven; Analyse und Planungshilfe**. VDE Verlag GmbH, 2010.

[12] K. GASSNER UND M. CONRAD. **ICT enabled independent living for the elderly: a status-quo analysis on products and the research landscape in the field of ambient assisted living (AAL) in EU-27**. Institute for Innovation and Technology, 2010.

[13] ONLINE. **Bundesministerium für Bildung und Forschung**. http://www.bmbf.de, 2012. besucht im Januar 2012.

[14] ONLINE. **AAL Joint Programme**. http://www.aal-europe.eu, 2012. besucht im Januar 2012.

[15] E. HOFFMANN UND J. NACHTMANN. **Report Altersdaten: Alter und Pflege**. Technischer Bericht 3, Deutsches Zentrum für Altersfragen, Berlin, 2007.

[16] D. COWAN UND A. TURNER-SMITH. **The Role of Assistive Technology in Alternative Models of Care for Older People**. *Royal Commission on Long Term Care*, **2**:325–346, 1999.

[17] WORLD HEALTH ORGANIZATION. **Constitution of the World Health Organization**. WHO, Geneva, 1946.

[18] KARIN BÖHM, STATISTISCHES BUNDESAMT, CLEMENS TESCH-RÖMER, DEUTSCHES ZENTRUM FÜR ALTERSFRAGEN, UND THOMAS ZIESE, ROBERT KOCH-INSTITUT, Herausgeber. **Beiträge zur Gesundheitsberichterstattung des Bundes: Gesundheit und Krankheit im Alter**. Robert Koch-Institut, Berlin, 2009.

[19] S. MENNING. **Report Altersdaten: Gesundheitszustand und gesundheitsrelevantes Verhalten Älterer**. Technischer Bericht 2, Deutsches Zentrum für Altersfragen, Berlin, 2006.

[20] FORUM DER BUNDESSTATISTIK, Herausgeber. **Alltag in Deutschland – Analysen zur Zeitverwendung**. Statistisches Bundesamt, 2004.

[21] SENIORWATCH. **European SeniorWatch Observatory and Inventory: Older People and Information Society Technology**. Technischer Bericht, 2002.

[22] J. SECKER, R. HILL, L. VILLENEAU, UND S. PARKMAN. **Promoting independence: but promoting what and how?** *Ageing & Society*, **23**(03):375–391, 2003.

[23] ONLINE. **sentha – Seniorengerechte Technik im häuslichen Alltag**. http://www.sentha.tu-berlin.de, 2013. besucht im März 2013.

[24] ONLINE. **TRON Intelligent House**. http://tronweb.super-nova.co.jp, 2012. besucht im Juli 2012.

[25] M. R. ALAM, M. B. I. REAZ, UND M. A. M. ALI. **A Review of Smart Homes – Past, Present, and Future**. *IEEE Transactions on Systems, Man, and Cybernetics*, 2012.

[26] ONLINE. **Living Labs**. http://www.aal-deutschland.de, 2012. besucht im Juli 2012.

[27] ONLINE. **Assisted Living – Selbstbestimmtes Wohnen im Alter mit moderner Technik**. http://www.assistedliving.de, 2012. besucht im Juli 2012.

[28] ONLINE. **SmartHome Paderborn**. http://www.smarthomepaderborn.de, 2012. besucht im Juli 2012.

[29] S. MENNING. **Report Altersdaten: Lebenserwartung, Mortalität und Morbidität im Alter**. Technischer Bericht 1, Deutsches Zentrum für Altersfragen, Berlin, 2006.

[30] V. ROCHA, P. BORZA, J. CORREIA, G. GONCALVES, A. PUSCAS, R. SEROMENHO, A. MASCIOLETTI, A. PICANO, S. COCORADA, UND M. CARP. **Wearable computing for patients with coronary diseases: Gathering efforts by comparing methods**. In *Proceedings of the 2010 IEEE International Conference on Automation, Quality and Testing, Robotics*, **2**, Seiten 1–9, 2010.

[31] M. MUFTI, D. AGOURIDIS, S. U. DIN, UND A. MUKHTAR. **Ubiquitous wireless infrastructure for elderly care**. In *Proceedings of the 2nd International Conference on PErvasive Technologies Related to Assistive Environments*, Seiten 22:1–22:5, 2009.

[32] A. B. WALUYO, W.-S. YEOH, I. PEK, Y. YONG, UND X. CHEN. **MobiSense: Mobile body sensor network for ambulatory monitoring**. *ACM Transactions on Embedded Computing Systems*, **10**(1):13:1–13:30, 2010.

[33] C. PATTICHIS, E. KYRIACOU, S. VOSKARIDES, M. PATTICHIS, R. ISTEPANIAN, UND C. SCHIZAS. **Wireless telemedicine systems: an overview**. *IEEE Antennas and Propagation Magazine*, **44**(2):143–153, 2002.

[34] A. PANTELOPOULOS UND N. G. BOURBAKIS. **A survey on wearable sensor-based systems for health monitoring and prognosis**. *IEEE Transactions on Systems, Man, and Cybernetics*, **40**(1):1–12, 2010.

[35] L. GATZOULIS UND I. IAKOVIDIS. **Wearable and Portable eHealth Systems**. *IEEE Engineering in Medicine and Biology Magazine*, **26**(5):51–56, 2007.

[36] A. LYMBERIS UND A. DITTMAR. **Advanced Wearable Health Systems and Applications - Research and Development Efforts in the European Union**. *IEEE Engineering in Medicine and Biology Magazine*, **26**(3):29 –33, 2007.

[37] ONLINE. **KUKA**. http://www.kuka-robotics.com/de, 2012. besucht im Juli 2012.

[38] ONLINE. **VACUUM ROBOTS**. http://www.robotmatrix.org, 2012. besucht im Juli 2012.

[39] S. MOON UND G. VIRK. **Survey on ISO standards for industrial and service robots**. In *ICROS-SICE International Joint Conference 2009*, Seiten 1878–1881, 2009.

[40] M. HÄGELE, N. BLÜMLEIN, UND O. KLEINE. **Wirtschaftlichkeitsanalysen neuartiger Servicerobotik-Anwendungen und ihre Bedeutung für die Robotik-Entwicklung**. Technischer Bericht, Fraunhofer-Institute IPA und ISI, 2010.

[41] S. MEYER. **Mein Freund der Roboter. Servicerobotik für ältere Menschen – eine Antwort auf den demographischen Wandel?** VDE Verlag GmbH, 2010.

[42] ONLINE. **Nao**. http://www.aldebaran-robotics.com, 2012. besucht im August 2012.

[43] A. A. J. VAN DE VEN, A.-M. A. G. SPONSELEE, UND B. A. M. SCHOUTEN. **Robo M.D.: a home care robot for monitoring and detection of critical situations**. In *Proceedings of the 28th Annual European Conference on Cognitive Ergonomics*, Seiten 375–376, 2010.

[44] ONLINE. **iRobot**. http://www.irobot.com/de, 2012. besucht im Juli 2012.

[45] ONLINE. **Automover**. http://www.husqvarna.com/de, 2012. besucht im Juli 2012.

[46] B. GRAF, U. REISER, M. HÄGELE, K. MAUZ, UND P. KLEIN. **Robotic home assistant Care-O-bot® 3 – Product Vision and Innovation Platform**. In *2009 IEEE Workshop on Advanced Robotics and its Social Impacts*, Seiten 139–144, 2009.

[47] ONLINE. **ASTROMOBILE**. http://simon-listens.org, 2012. besucht im August 2012.

[48] ONLINE. **Asimo**. http://asimo.honda.com, 2012. besucht im August 2012.

[49] ONLINE. **Justin**. http://www.dlr.de, 2012. besucht im August 2012.

[50] ONLINE. **CHARLI**. http://www.unirel.vt.edu, 2012. besucht im August 2012.

[51] C. C. KEMP, P. M. FITZPATRICK, H. HIRUKAWA, K. YOKOI, K. HARADA, UND Y. MATSUMOTO. **Humanoids**. In *Springer Handbook of Robotics*, Seiten 1307–1333. Springer, 2008.

[52] M. FUJITA. **On activating human communications with pet-type robot AIBO**. *Proceedings of the IEEE*, **92**(11):1804–1813, 2004.

[53] ONLINE. **PaPeRo**. http://www.nec.co.jp, 2012. besucht im August 2012.

[54] ONLINE. **WiMi-Care**. http://www.wimi-care.de, 2012. besucht im August 2012.

[55] ONLINE. **RIBA**. http://rtc.nagoya.riken.jp, 2012. besucht im August 2012.

[56] ONLINE. **Paro**. http://www.parorobots.com, 2012. besucht im August 2012.

[57] K. WADA, T. SHIBATA, T. MUSHA, UND S. KIMURA. **Robot therapy for elders affected by dementia**. *IEEE Engineering in Medicine and Biology Magazine*, **27**(4):53 –60, 2008.

[58] ONLINE. **DOMEO**. http://www.aal-domeo.org, 2012. besucht im August 2012.

[59] T. REHRL, J. BLUME, J. GEIGER, A. BANNAT, F. WALLHOFF, S. IHSEN, Y. JEANRENAUD, M. MERTEN, B. SCHÖNEBECK, S. GLENDE, UND C. NEDOPIL. **ALIAS: Der anpassungsfähige Ambient Living Assistent**. In *Tagungsband des 4. Deutschen Ambient Assisted Living (AAL 2011) Kongresses, Berlin*, 2011.

[60] T. REHRL, R. TRONCY, A. BLEY, S. IHSEN, K. SCHEIBL, W. SCHNEIDER, S. GLENDE, S. GOETZE, J. KESSLER, C. HINTERMUELLER, UND F. WALLHOFF. **The Ambient Adaptable Living Assistant is Meeting its Users**. In *Proceedings of the AAL Forum 2012*, 2012.

[61] S. L. LAURITZEN. **Graphical Models**. Oxford University Press, 1996.

[62] M. JORDAN, Herausgeber. **Learning in Graphical Models**. MIT Press, 1998.

[63] K. MURPHY. **Dynamic Bayesian Networks: Representation, Inference and Learning**. PhD thesis, UC Berkeley, 2002.

[64] M. JORDAN. **Graphical Models**. *Statistical Science*, **19**(1):140–155.

[65] M. AL-HAMES. **Graphische Modelle in der Mustererkennung**. Dissertation, Technische Universität München, München, 2008.

[66] D. KOLLER UND N. FRIEDMAN. **Probabilistic Graphical Models: Principles and Techniques**. MIT Press, 2009.

[67] T. HARJU. **GRAPH THEORY**. Lecture Notes, University of Turku, FIN-20014 Turku, Finland, 2011.

[68] T. BAYES. **An essay towards solving a problem in the doctrine of chances**. *Phil. Trans. of the Royal Soc. of London*, **53**:370–418, 1763.

[69] A. N. KOLMOGOROV. **Grundbegriffe der Wahrscheinlichkeitsrechnung**. Springer, 1933.

[70] E. H. SHORTLIFFE. **Computer-based Medical Consultations**. Elsevier, 1976.

[71] J. PEARL. **Probabilistic reasoning in intelligent systems: networks of plausible inference**. Morgan Kaufmann, 1988.

[72] J. PEARL UND T. VERMA. **The logic of representing dependencies by directed graphs**. In *Proceedings of the sixth National conference on Artificial intelligence*, **1**, Seiten 374–379, 1987.

[73] D. E. HECKERMAN UND B. N. NATHWANI. **An evaluation of the diagnostic accuracy of Pathfinder**. In *Computers and Biomedical Research*, Seiten 25–56, 1992.

[74] J. BILMES UND C. BARTELS. **Graphical Model Architectures for Speech Recognition**. *IEEE Signal Processing Magazine*, **22**(5):89–100, 2005.

[75] R. HUANG, V. PAVLOVIC, UND D. METAXAS. **A Graphical Model Framework for Image Segmentation**. In *Applied Graph Theory in Computer Vision and Pattern Recognition*. Springer.

[76] M. PRADHAN, G. PROVAN, B. MIDDLETON, UND M. HENRION. **Knowledge Engineering for Large Belief Networks**. In *Proceedings of the 10th Conference Annual Conference on Uncertainty in Artificial Intelligence*, Seiten 484–490, 1994.

[77] E. GULTEPE, H. NGUYEN, T. ALBERTSON, UND I. TAGKOPOULOS. **A Bayesian network for early diagnosis of sepsis patients: a basis for a clinical decision support system**. In *2012 IEEE 2nd International Conference on Computational Advances in Bio and Medical Sciences*, Seiten 1–5, 2012.

[78] E. HORVITZ, J. APACIBLE, R. SARIN, UND L. LIAO. **Prediction, Expectation, and Surprise: Methods, Designs, and Study of a Deployed Traffic Forecasting Service**. In *Proceedings of the 21th Annual Conference on Uncertainty in Artificial Intelligence*, Seiten 275–283, 2005.

[79] K. SACHS, O. PEREZ, D. PE'ER, D. LAUFFENBURGER, UND G. NOLAN. **Causal protein-signaling networks derived from multiparameter single-cell data**. *Science*, **308**(5721):523, 2005.

[80] S. PRINCE. **Computer Vision: Models Learning and Inference**. Cambridge University Press, 2012.

[81] K. MURPHY. **The Bayes Net Toolbox for MATLAB**. *Computing Science and Statistics*, **33**:2001, 2001.

[82] J. BILMES UND G. ZWEIG. **The graphical models toolkit: An open source software system for speech and time-series processing**. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, Seiten 3916–3919, 2002.

[83] S. J. YOUNG, G. EVERMANN, M. J. F. GALES, T. HAIN, D. KERSHAW, G. MOORE, J. ODELL, D. OLLASON, D. POVEY, V. VALTCHEV, UND P. C. WOODLAND. **The HTK Book, version 3.4**. Cambridge University Engineering Department, Cambridge, UK, 2006.

[84] L. R. RABINER. **A tutorial on hidden Markov models and selected applications in speech recognition**. *Proceedings of the IEEE*, **77**(2):257–286, 1989.

[85] L. E. BAUM UND T. PETRIE. **Statistical Inference for Probabilistic Functions of Finite State Markov Chains**. *The Annals of Mathematical Statistics*, **37**(6):1554–1563, 1966.

[86] A. P. DEMPSTER, N. M. LAIRD, UND D. B. RUBIN. **Maximum Likelihood from Incomplete Data via the EM Algorithm**. *Journal of the Royal Statistical Society. Series B (Methodological)*, **39**(1):1–38, 1977.

[87] J. BILMES. **A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models**. Technischer Bericht, Department of Electrical Engineering and Computer Science, U.C. Berkeley, 1998.

[88] A. VITERBI. **Error bounds for convolutional codes and an asymptotically optimum decoding algorithm**. *IEEE Transactions on Information Theory*, **13**(2):260–269, 1967.

[89] M. JORDAN UND Y. WEISS. **Probabilistic inference in graphical models**. *Handbook of neural networks and brain theory*, 2002.

[90] F. V. JENSEN, S. L. LAURITZEN, UND K. G. OLESEN. **Bayesian updating in causal probabilistic networks by local computations**. *Computational Statistics Quaterly*, **4**:269–282, 1990.

[91] A. DOUCET, N. DEFREITAS, UND N. GORDON. **An Introduction to Sequential Monte Carlo Methods**. Springer, 2001.

[92] C. ANDRIEU, N. DE FREITAS, A. DOUCET, UND M. I. JORDAN. **An Introduction to MCMC for Machine Learning**. *Machine Learning*, **50**(1):5–43, 2003.

[93] A. DOUCET UND A. JOHANSEN. **A Tutorial on Particle Filtering and Smoothing: Fifteen years later**. *Handbook of Nonlinear Filtering*, Seiten 656–704, 2009.

[94] K. MURPHY. **An introduction to graphical models**, 2001.

[95] G. RIGOLL. **Pattern Recognition**. Lecture Notes, Technische Universität München, München, 2012.

[96] C. M. BISHOP. **Pattern Recognition and Machine Learning (Information Science and Statistics)**. Springer, 2006.

[97] S. GÜNTER. **Vergleich von Erkennungsmethoden**. Technischer Bericht IAM-04-001, Institut für Informatik und angewandte Mathematik – Universität Bern, 2004.

[98] A. JAIMES UND N. SEBE. **Multimodal Human Computer Interaction: A Survey**. In *Workshop on HCI, 2005 IEEE International Conference on Computer Vision*, **3766**, Seiten 1–15, 2005.

[99] A. JAIMES UND N. SEBE. **Multimodal human computer interaction: A survey**. *Computer Vision and Image Understanding*, **108**(1-2):116–134, 2007.

[100] D. MCNEILL. **Hand and Mind: What Gestures Reveal about Thought**. University of Chicago Press, 1992.

[101] R. SHARMA, V. PAVLOVIC, UND T. HUANG. **Toward multimodal human-computer interface**. *Proceedings of the IEEE*, **86**(5):853–869, 1998.

[102] S. OVIATT. **Ten myths of multimodal interaction**. *Communications of the ACM*, **42**(11):74–81, 1999.

[103] M. KRUEGER. **Artificial reality II**. Addison-Wesley, 1991.

[104] J. P. WACHS, M. KÖLSCH, H. STERN, UND Y. EDAN. **Vision-based hand-gesture applications**. *ACM*, **54**(2):60–71, 2011.

[105] ONLINE. **Samsung Smart TV**. http://smart-tv.samsung.de, 2013. besucht im Februar 2013.

[106] D. ANASTASIOU. **Gestures in assisted living environments**. In *The 9th International Gesture Workshop, Gesture in Embodied Communication and Human-Computer Interaction*, Seiten 25–27, 2011.

[107] R. NESSELRATH, C. LU, C. H. SCHULZ, J. FREY, UND J. ALEXANDERSSON. **A Gesture Based System for Context-Sensitive Interaction with Smart Homes**. In *Ambient Assisted Living, 4. AAL-Kongress 2011*, Seiten 209–219, 2011.

[108] J. GAST, A. BANNAT, T. REHRL, G. RIGOLL, F. WALLHOFF, C. MAYER, UND B. RADIG. **Did I Get It Right: Head Gestures Analysis for Human-Machine Interactions**. In *Proceedings of the 13th International Conference on Human-Computer Interaction. Part II: Novel Interaction Methods and Techniques*, **LNCS 5611**, Seiten 170–177, 2009.

[109] T. REHRL, A. BANNAT, J. GAST, F. WALLHOFF, G. RIGOLL, C. MAYER, Z. RIAZ, B. RADIG, S. SOSNOWSKI, UND K. KÜHNLENZ. **Multiple Parallel Vision-Based Recognition in a Real-Time Framework for Human-Robot-Interaction Scenarios**. In *Proceedings of the 3nd International Conference on Advancements Computer-Human Interaction*, 2010.

[110] T. REHRL, N. THEISSING, A. BANNAT, J. GAST, A. ARSIC, W. WALLHOFF, UND G. RIGOLL. **Graphical Models for Real-Time Capable Gesture Recognition**. In *Proceedings of the 2010 IEEE International Conference on Image Processing*, Seiten 2445–2448, 2010.

[111] F. WALLHOFF, T. REHRL, C. MAYER, UND B. RADIG. **Real-time face and gesture analysis for human-robot interaction**. In N. KEHTARNAVAZ UND M. CARLSOHN, Herausgeber, *Proceedings of SPIE, Society of Photo-Optical Instrumentation Engineers Conference*, **7724**, Seiten 0A–1 – 0A–10, 2010.

[112] J. BLUME, T. REHRL, UND G. RIGOLL. **Multimodale Interaktion auf einer sozialen Roboterplattform**. *at – Automatisierungstechnik*, **61**(11):737–748, 2013.

[113] D. BRSCI, M. EGGERS, F. ROHRMÜLLER, O. KOURAKOS, S. SOSNOWSKI, D. ALTHOFF, M. LAWITZKY, A. MÖRTL, M. RAMBOW, V. KOROPOULI, J. HERNÁNDEZ, X. ZANG, W. WANG, D. WOLLHERR, K. KÜHNLENZ, C. MAYER, T. KRUSE, A. KIRSCH, J. BLUME, A. BANNAT, T. REHRL, F. WALLHOFF, T. LORENZ, P. BASILI, C. LENZ, T. RÖDER, G. PANIN, W. MAIER, S. HIRCHE, M. BUSS, M. BEETZ, B. RADIG, A. SCHUBÖ, S. GLASAUER, A. KNOLL, UND E. STEINBACH. **Multi Joint Action in CoTeSys - Setup and Challenges**. Technischer Bericht CoTeSys-TR-10-01, CoTeSys Cluster of Excelence: Technische Universität München & Ludwig-Maximilians-Universität München, 2010.

[114] S. MITRA UND T. ACHARYA. **Gesture Recognition: A Survey**. *IEEE Transactions on Systems, Man, and Cybernetics*, **37**(3):311–324, 2007.

[115] V. PAVLOVIC, R. SHARMA, UND T. HUANG. **Visual interpretation of hand gestures for human-computer interaction: a review**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**(7):677–695, 1997.

[116] F. K. H. QUEK. **Eyes in the interface**. *Image and Vision Computing*, Seiten 511–525, 1995.

[117] S. BERMAN UND H. STERN. **Sensors for Gesture Recognition Systems**. *IEEE Transactions on Systems, Man, and Cybernetics*, **42**(3):277–290, 2012.

[118] C. LEE UND Y. XU. **Online, Interactive Learning of Gestures for Human/Robot Interfaces**. In *Proceedings of the 1996 IEEE International Conference on Robotics and Automation*, Seiten 2982–2987, 1996.

[119] S. REIFINGER. **Multimodale Interaktion in Augmented Reality Umgebungen am Beispiel der Spieledomäne**. Dissertation, Technische Universität München, München, 2008.

[120] D. GAVRILA. **The Visual Analysis of Human Movement: A Survey**. *Computer Vision and Image Understanding*, **73**(1):82–98, 1999.

[121] P. GARG, N. AGGARWAL, UND S. SOFAT. **Vision based hand gesture recognition**. *World Academy of Science, Engineering and Technology*, **49**:972–977, 2009.

[122] ONLINE. **Microsoft Kinect Sensor**. http://www.microsoft.com, 2013. besucht im Februar 2013.

[123] Z. REN, J. MENG, J. YUAN, UND Z. ZHANG. **Robust hand gesture recognition with kinect sensor**. In *Proceedings of the 19th ACM international conference on Multimedia*, Seiten 759–760, 2011.

[124] I. OIKONOMIDIS, N. KYRIAZIS, UND A. ARGYROS. **Efficient model-based 3D tracking of hand articulations using Kinect**. In *Proceedings of the 2011 British Machine Vision Conference*, Seiten 101.1–101.11, 2011.

[125] N. PUGEAULT UND R. BOWDEN. **Spelling it out: Real-time ASL fingerspelling recognition**. In *Proceedings of the 2011 IEEE International Conference on Computer Vision*, Seiten 1114–1119, 2011.

[126] N. DALAL UND B. TRIGGS. **Histograms of oriented gradients for human detection**. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, **1**, Seiten 886–893, 2005.

[127] S. LIWICKI UND M. EVERINGHAM. **Automatic recognition of fingerspelled words in British sign language**. In *Proceedings of the 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Seiten 50–57, 2009.

[128] J. YAMATO, J. OHYA, UND K. ISHII. **Recognizing human action in time-sequential images using hidden Markov model**. In *Proceedings of the 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Seiten 379–385, 1992.

[129] T. STARNER UND A. PENTLAND. **Real-time American Sign Language recognition from video using hidden Markov models**. In *Proceedings of the 1995 IEEE International Symposium on Computer Vision*, Seiten 265–270, 1995.

[130] C. VOGLER UND D. METAXAS. **ASL Recognition Based on a Coupling Between HMMs and 3D Motion Analysis**. In *Proceedings of the 6th International Conference on Computer Vision*, Seiten 363–369, 1998.

[131] C. MORIMOTO, Y. YACOOB, UND L. DAVIS. **Recognition of head gestures using hidden Markov models**. In *Proceedings of the 13th International Conference on Pattern Recognition*, **3**, Seiten 461–465, 1996.

[132] L.-P. MORENCY UND T. DARRELL. **Head gesture recognition in intelligent interfaces: the role of context in improving recognition**. In *Proceedings of the 11th international conference on Intelligent user interfaces*, Seiten 32–38, 2006.

[133] D. KORTENKAMP, E. HUBER, R. P. BONASSO, UND M. INC. **Recognizing and Interpreting Gestures on a Mobile Robot**. In *Proceedings of the 13th National Conference on Artifical Intelligence*, Seiten 915–921, 1996.

[134] A. RAMAMOORTHY, N. VASWANI, S. CHAUDHURY, UND S. BANERJEE. **Recognition of dynamic hand gestures.** *Pattern Recognition*, **36**(9):2069–2081, 2003.

[135] M. HASANUZZAMAN, V. AMPORNARAMVETH, T. ZHANG, M. BHUIYAN, Y. SHIRAI, UND H. UENO. **Real-time Vision-based Gesture Recognition for Human Robot Interaction**. In *Proceedings of the 2004 IEEE International Conference on Robotics and Biomimetics*, Seiten 413–418, 2004.

[136] M. HASANUZZAMAN, T. ZHANG, V. AMPORNARAMVETH, H. GOTODA, Y. SHIRAI, UND H. UENO. **Adaptive visual gesture recognition for human-robot interaction using a knowledge-based software platform**. *Robotics and Autonomous Systems*, **55**(8):643–657, 2007.

[137] K. NICKEL UND R. STIEFELHAGEN. **Visual recognition of pointing gestures for human-robot interaction**. *Image and Vision Computing*, **25**(12):1875–1884, 2007.

[138] N. MOHAMED, J. AL-JAROODI, UND I. JAWHAR. **Middleware for Robotics: A Survey**. In *Proceedings of the 2008 IEEE Conference on Robotics, Automation and Mechatronics*, Seiten 736–742, 2008.

[139] M. GOEBL UND G. FÄRBER. **A Real-Time-capable Hard- and Software Architecture for Joint Image and Knowledge Processing in Cognitive Automobiles**. In *Proceedings of the 2007 IEEE Intelligent Vehicles Symposium*, 2007.

[140] M. GOEBL. **Eine realzeitfähige Architektur zur Integration kognitiver Funktionen**. Dissertation, Technische Universität München, München, 2009.

[141] F. WALLHOFF, J. BLUME, A. BANNAT, W. RÖSEL, C. LENZ, UND A. KNOLL. **A skill-based approach towards hybrid assembly**. *Advanced Engineering Informatics*, **24**(3):329–339, 2010.

[142] A. BANNAT, J. BLUME, J. T. GEIGER, T. REHRL, F. WALLHOFF, C. MAYER, B. RADIG, S. SOSNOWSKI, UND K. KÜHNLENZ. **A Multimodal Human-Robot-Dialog Applying Emotional Feedbacks.** In *Proceedings of the Second International Conference on Social Robotics*, **6414**, Seiten 1–10, 2010.

[143] B. FREEDMAN, A. SHPUNT, M. MACHLINE, UND Y. ARIELI. **DEPTH MAPPING USING PROJECTED PATTERNS**, 2008. Patent.

[144] ONLINE. **OpenNI**. http://www.openni.org, 2012. besucht im August 2012.

[145] W. PRATT. **Digital image processing**. John Wiley & Sons, Inc., New York, NY, USA, 2007.

[146] S. O. BELKASIM, M. SHRIDHAR, UND M. AHMADI. **Pattern recognition with moment invariants: A comparative study and new results**. *Pattern Recognition*, **24**(12):1117–1138, 1991.

[147] M. R. TEAGUE. **Image analysis via the general theory of moments**. *Journal of the Optical Society of America (1917-1983)*, **70**:920–930, 1980.

[148] F. ZERNIKE. **Diffraction theory of the cut procedure and its improved form, the phase contrast method**. *Physica*, **1**:689–704, 1934.

[149] V. N. VAPNIK. **The nature of statistical learning theory**. Springer, 1995.

[150] C. J. C. BURGES. **A Tutorial on Support Vector Machines for Pattern Recognition**. *Data Mining and Knowledge Discovery*, **2**(2):121–167, 1998.

[151] P. VIOLA UND M. J. JONES. **Rapid object detection using a boosted cascade of simple features**. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, **1**, Seiten 511–518, 2001.

[152] P. VIOLA UND M. J. JONES. **Robust Real-Time Face Detection**. *International Journal of Computer Vision*, **57**:137–154, 2004.

[153] J. AHLBERG. **CANDIDE-3 – an updated parameterized face**. Technischer Bericht, Linköping University, Sweden, 2001.

[154] S. SCHREIBER. **Personenverfolgung und Gestenerkennung in Videodaten**. Dissertation, Technische Universität München, München, 2009.

[155] M. HU. **Visual Pattern Recognition by Moment Invariants**. *IRE Transaction on Information Theory*, Seiten 179–187, 1963.

[156] P. EKMAN. **Universals and Cultural Differences in Facial Expressions of Emotion**. University of Nebraska Press, 1971.

[157] M. PANTIC, M. VALSTAR, R. RADEMAKER, UND L. MAAT. **Web-based database for facial expression analysis**. In *Proceedings of the 2005 IEEE International Conference on Multimedia and Expo*, Seiten 317–321, 2005.

[158] M. F. VALSTAR UND M. PANTIC. **Induced Disgust, Happiness and Surprise: an Addition to the MMI Facial Expression Database**. In *Proceedings of the International Language Resources and Evaluation Conference*, Seiten 65–70, 2010.

[159] R. W. LEVENSON, L. L. CARSTENSEN, W. V. FRIESEN, UND P. EKMAN. **Emotion, physiology, and expression in old age.** *Psychology and aging*, **6**(1):28–35, 1991.

[160] A. MEHRABIAN. **Communication without words**. In *Psychology Today*, Seiten 51–52. 1968.

[161] B. TAKÁCS UND D. HANÁK. **A mobile system for assisted living with ambient facial interfaces**. *International Journal on Computer Science and Information System*, **2**:33–50, 2007.

[162] B. TAKÁCS UND D. HANAK. **A prototype home robot with an ambient facial interface to improve drug compliance**. *Journal of Telemedicine and Telecare*, **14**(7):393–395, 2008.

[163] M. S. BARTLETT, G. LITTLEWORT, I. FASEL, UND J. R. MOVELLAN. **Real Time Face Detection and Facial Expression Recognition: Development and Applications to Human Computer Interaction**. In *Conference on Computer Vision and Pattern Recognition Workshop*, 2003.

[164] H. ISHIGURO, T. ONO, M. IMAI, T. MAEDA, T. KANDA, UND R. NAKATSU. **Robovie: an interactive humanoid robot**. *Industrial Robot: An International Journal*, Seiten 498–504, 2001.

[165] S. SOSNOWSKI, C. MAYER, K. KÜHNLENZ, UND B. RADIG. **Mirror my emotions! Combining facial expression analysis and synthesis on a robot**. In *The Thirty Sixth Annual Convention of the Society for the Study of Artificial Intelligence and Simulation of Behaviour*, 2010.

[166] P. EKMAN UND W. FRIESEN. **Facial Action Coding System: A Technique for the Measurement of Facial Movement.** Consulting Psychologists Press, 1978.

[167] P. EKMAN, W. V. FRIESEN, UND J. C. HAGER. **Facial Action Coding System: The Manual**. CD ROM, 2002.

[168] M. PANTIC UND L. ROTHKRANTZ. **Automatic analysis of facial expressions: the state of the art**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(12):1424–1445, 2000.

[169] W. PARROTT. **Emotions in social psychology: Essential readings.** Psychology Press, 2001.

[170] V. BETTADAPURA. **Face Expression Recognition and Analysis: The State of the Art**. *CoRR*, **abs/1203.6722**, 2012.

[171] M. PANTIC UND M. BARTLETT. **Machine Analysis of Facial Expressions**. In K. DELAC UND M. GRGIC, Herausgeber, *Face Recognition*, Seiten 377–416. I-Tech Education and Publishing, 2007.

[172] A. SAMAL UND P. A. IYENGAR. **Automatic recognition and analysis of human faces and facial expressions: a survey**. *Pattern Recognition*, **25**(1):65–77, 1992.

[173] B. FASEL UND J. LUETTIN. **Automatic facial expression analysis: a survey**. *Pattern Recognition*, **36**(1):259–275, 2003.

[174] Y. TIAN, T. KANADE, UND J. COHN. **Facial expression analysis**. *Handbook of face recognition*, Seiten 247–275, 2005.

[175] Z. ZENG, M. PANTIC, G. ROISMAN, UND T. HUANG. **A Survey of Affect Recognition Methods: Audio, Visual, and Spontaneous Expressions**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **31**(1):39–58, 2009.

[176] M. PANTIC, A. PENTLAND, A. NIJHOLT, UND T. HUANG. **Human computing and machine understanding of human behavior: a survey**. In *Proceedings of the 8th international conference on Multimodal interfaces*, Seiten 239–248, 2006.

[177] C. MAYER. **Facial Expression Recognition With A Three-Dimensional Face Model**. Dissertation, Technische Universität München, München, 2012.

[178] T. SAKAGUCHI, S. MORISHIMA, UND F. KISHINO. **Facial Expression Recognition from Image Sequences Using Hidden Markov Model,**. VLBV95, A-5, 1995.

[179] T. OTSUKA UND J. OHYA. **Recognition of facial expressions using HMM with continuous output probabilities**. In *5th IEEE International Workshop on Robot and Human Communication, 1996*, Seiten 323–328, 1996.

[180] T. OTSUKA UND J. OHYA. **Recognizing Multiple Persons' Facial Expressions Using HMM Based on Automatic Extraction of Significant Frames from Image Sequences**. In *Proceedings of the 1997 IEEE International Conference on Image Processing*, Seiten 546–549, 1997.

[181] T. OTSUKA UND J. OHYA. **Spotting segments displaying facial expression from image sequences using HMM**. In *Proceedings of the Third IEEE International Conference on Automatic Face and Gesture Recognition*, Seiten 442–447, 1998.

[182] J. LIEN, T. KANADE, J. COHN, UND C.-C. LI. **Automated facial expression recognition based on FACS action units**. In *Proceedings of the Thrid IEEE International Conference on Automatic Face and Gesture Recognition*, Seiten 390–395, 1998.

[183] Y. ZHU, L. DE SILVA, UND C. KO. **Using moment invariants and HMM in facial expression recognition**. In *Proceedings of the 4th IEEE Southwest Symposium Image Analysis and Interpretation*, Seiten 305–309, 2000.

[184] Y. ZHU, L. C. DE SILVA, UND C. C. KO. **Using moment invariants and HMM in facial expression recognition**. *Pattern Recognition Letters*, **23**(1-3):83–91, 2002.

[185] M. PARDAS UND A. BONAFONTE. **Facial animation parameters extraction and expression recognition using Hidden Markov Models**. *Signal Processing: Image Communication*, **17**(9):675 – 688, 2002.

[186] I. COHEN, A. GARG, UND T. S. HUANG. **Emotion Recognition from Facial Expressions using Multilevel HMM**. In *Neural Information Processing Systems*, 2000.

[187] T. KANADE, J. F. COHN, UND Y. TIAN. **Comprehensive Database for Facial Expression Analysis**. In *Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, Seiten 46–53, 2000.

[188] P. LUCEY, J. COHN, T. KANADE, J. SARAGIH, Z. AMBADAR, UND I. MATTHEWS. **The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression**. In *Proceedings of the Third International Workshop on CVPR for Human Communicative Behavior Analysis*, Seiten 94–101, 2010.

[189] S. MÜLLER, F. WALLHOFF, F. HÜLSKEN, UND G. RIGOLL. **Facial Expression Recognition Using Pseudo 3-D Hidden Markov Models**. In *Proceedings of 16th International Conference on Pattern Recognition*, **2**, Seiten 32–35, 2002.

[190] P. ALEKSIC UND A. KATSAGGELOS. **Automatic facial expression recognition using facial animation parameters and multistream HMMs**. *IEEE Transactions on Information Forensics and Security*, **1**(1):3–11, 2006.

[191] J. J. LEE, M. ZIA UDDIN, UND T.-S. KIM. **Spatiotemporal Human Facial Expression Recognition Using Fisher Independent Component Analysis and Hidden Markov Model**. In *30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Seiten 2546–2549, 2008.

[192] M. SCHMIDT, M. SCHELS, UND F. SCHWENKER. **A Hidden Markov Model Based Approach for Facial Expression Recognition in Image Sequences**. In *Artificial Neural Networks in Pattern Recognition*, **5998**, Seiten 149–160. Springer, 2010.

[193] H. TANG, M. HASEGAWA-JOHNSON, UND T. HUANG. **Non-frontal view facial expression recognition based on ergodic hidden Markov model supervectors**. In *Proceedings of the 2010 IEEE International Conference on Multimedia and Expo*, Seiten 1202–1207, 2010.

[194] A. GAWEDA UND E. PATTERSON. **Individual identification based on facial dynamics during expressions using active-appearance-based Hidden Markov Models**. In *Proceedings of the Ninth IEEE International Conference on Automatic Face and Gesture Recognition*, Seiten 797–802, 2011.

[195] N. SEBE, M. LEW, I. COHEN, Y. SUN, T. GEVERS, UND T. HUANG. **Authentic facial expression analysis**. In *Proceedings of the Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, Seiten 517–522, 2004.

[196] N. SEBE, M. S. LEW, Y. SUN, I. COHEN, T. GEVERS, UND T. S. HUANG. **Authentic facial expression analysis**. *Image and Vision Computing*, **25**(12):1856–1863, 2007.

[197] A. M. MARTINEZ UND R. BENAVENTE. **The AR Face Database**. Technischer Bericht, 1998.

[198] T. SIM, S. BAKER, UND M. BSAT. **The CMU Pose, Illumination, and Expression (PIE) database**. In *Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition*, Seiten 46–51, 2002.

[199] T. SIM, S. BAKER, UND M. BSAT. **The CMU Pose, Illumination, and Expression Database**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **25**(12):1615–1618, 2003.

[200] M. LYONS, S. AKAMATSU, M. KAMACHI, UND J. GYOBA. **Coding facial expressions with Gabor wavelets**. In *Proceedings of the Third IEEE International Conference on Automatic Face and Gesture Recognition*, Seiten 200–205, 1998.

[201] ONLINE. **Frank Wallhoff: Facial Expressions and Emotion Database**. http://www.mmk.ei.tum.de/~waf/fgnet/feedtum.html, Technische Universität München 2006, 2006. besucht im Dezember 2012.

[202] K. KAULARD, D. W. CUNNINGHAM, H. H. BÜLTHOFF, UND C. WALLRAVEN. **The MPI Facial Expression Database – A Validated Database of Emotional and Conversational Facial Expressions**. *PLoS ONE*, **7**(3):e32321, 2012.

[203] E. DOUGLAS-COWIE, R. COWIE, UND M. SCHRÖDER. **A New Emotion Database: Considerations, Sources and Scope**. In *Proceedings of the ISCA Workshop on Speech and Emotion*, Seiten 39–44, 2000.

[204] W. V. FRIESEN UND P. EKMAN. **EMFACS-7: Emotional Facial Action Coding System**. 1983. Unpublished manual.

[205] T. F. COOTES UND C. J. TAYLOR. **Active Shape Models – Smart Snakes**. In *Proceedings of the 1992 British Machine Vision Conference*, Seiten 266–275, 1992.

[206] T. F. COOTES, C. J. TAYLOR, D. H. COOPER, UND J. GRAHAM. **Active Shape Models – Their Training and Application**. *Computer Vision and Image Understanding*, **61**(1):38–59, 1995.

[207] T. F. COOTES, G. J. EDWARDS, UND C. J. TAYLOR. **Active Appearance Models**. In *Proceedings of the 5th European Conference on Computer Vision*, **2**, Seiten 484–498, 1998.

[208] V. BLANZ UND T. VETTER. **A morphable model for the synthesis of 3D faces**. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, Seiten 187–194, 1999.

[209] Y. CHEN UND F. DAVOINE. **Simultaneous tracking of rigid head motion and non-rigid facial animation by analyzing local features statistically**. In *Proceedings of the 2006 British Machine Vision Conference*, Seite II:609, 2006.

[210] I. KOTSIA UND I. PITAS. **Facial Expression Recognition in Image Sequences Using Geometric Deformation Features and Support Vector Machines**. *IEEE Transactions On Image Processing*, **16**(1):172–187, January 2007.

[211] F. DORNAIKA UND F. DAVOINE. **Simultaneous Facial Action Tracking and Expression Recognition in the Presence of Head Motion**. *International Journal of Computer Vision*, **76**:257–281, 2008.

[212] I. S. PANDZIC UND R. FORCHHEIMER, Herausgeber. **MPEG-4 Facial Animation: The Standard, Implementation and Applications**. John Wiley & Sons, Inc., 2003.

[213] L. MOLINA, L. BELANCHE, UND A. NEBOT. **Feature selection algorithms: a survey and experimental evaluation**. In *Proceedings of the 2002 IEEE International Conference on Data Mining*, Seiten 306–313, 2002.

[214] A. WHITNEY. **A Direct Method of Nonparametric Measurement Selection**. *IEEE Transactions on Computers*, **C-20**(9):1100–1103, 1971.

[215] A. JAIN UND D. ZONGKER. **Feature selection: Evaluation, application, and small sample performance**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**:153–158, 1997.

[216] T. MARILL UND D. GREEN. **On the effectiveness of receptors in recognition systems**. *IEEE Transactions on Information Theory*, **9**(1):11–17, 1963.

[217] H. LIU UND H. MOTODA, Herausgeber. **Computational Methods of Feature Selection**. Chapman & Hall/CRC, 2008.

[218] S. THEODORIDIS UND K. KOUTROUMBAS. **Pattern Recognition**. Academic Press, 2008.

[219] S. KULLBACK UND R. A. LEIBLER. **On information and sufficiency**. *Annals of Mathematical Statistics*, **22**:49–86, 1951.

[220] K.-M. SCHNEIDER. **A new feature selection score for multinomial naive Bayes text classification based on KL-divergence**. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*. Association for Computational Linguistics, 2004.

[221] Z. ZHEN, X. ZENG, H. WANG, UND L. HAN. **A global evaluation criterion for feature selection in text categorization using Kullback-Leibler divergence**. In *International Conference of Soft Computing and Pattern Recognition*, Seiten 440–445, 2011.

[222] P. MARTÍN, M. SÁNCHEZ, L. ÁLVAREZ, V. ALONSO, UND J. BAJO. **Multi-Agent System for Detecting Elderly People Falls through Mobile Devices**. *Ambient Intelligence-Software and Applications*, Seiten 93–99, 2011.

[223] M. VOLKHARDT, C. WEINRICH, C. SCHROETER, UND H.-M. GROSS. **A Concept for Detection and Tracking of People in Smart Home Environments with a Mobile Robot**. In *2nd CompanionAble Workshop co-located with the 3rd European Conference on Ambient Intelligence*, 2009.

[224] J.-I. PAN, C.-J. YUNG, C.-C. LIANG, UND L.-F. LAI. **An Intelligent Homecare Emergency Service System for Elder Falling**. In R. MAGJAREVIC, J. H. NAGEL, UND R. MAGJAREVIC, Herausgeber, *World Congress on Medical Physics and Biomedical Engineering 2006*, **14**, Seiten 424–428. Springer, 2007.

[225] M. ISARD. **PAMPAS: Real-Valued Graphical Models for Computer Vision**. In *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Seiten 613–620, 2003.

[226] T. REHRL, J. GAST, N. THEISSING, A. BANNAT, D. ARSIC, F. WALLHOFF, G. RIGOLL, C. MAYER, UND B. RADIG. **A Graphical Model for Unifying Tracking and Classification within a Multimodal Human-Robot Interaction Scenario**. In *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Workshop on Computer Vision for Human-Robot Interaction*, 2010.

[227] T. REHRL, N. THEISSING, A. BANNAT, J. GAST, A. ARSIC, W. WALLHOFF, UND G. RIGOLL. **Tracking using Bayesian Inference with a Two-Layer Graphical Model**. In *Proceedings of the 2010 IEEE International Conference on Image Processing*, Seiten 3961–3964, 2010.

[228] M. ISARD UND A. BLAKE. **Contour tracking by stochastic propagation of conditional density**. In *Proceedings of the 4th European Conference on Computer Vision*, Seiten 343–356, 1996.

[229] M. ISARD UND A. BLAKE. **CONDENSATION – Conditional Density Propagation for Visual Tracking**. *International Journal of Computer Vision*, **29**:5–28, 1998.

[230] M. ISARD. **Visual Motion Analysis by Probabilistic Propagation of Conditional Density**. PhD thesis, University of Oxford, 1998.

[231] R. E. KALMAN. **A New Approach to Linear Filtering and Prediction Problems**. *Journal Of Basic Engineering*, **82**(Series D):35–45, 1960.

[232] G. SMITH, S. SCHMIDT, UND MCGEE. **Application of statistical filter theory to the optimal estimation of position and velocity on board a circumlunar vehicle**. Technischer Bericht, National Aeronautics and Space Administration, 1962.

[233] G. WELCH UND G. BISHOP. **An Introduction to the Kalman Filter**. Technischer Bericht, University of North Carolina at Chapel Hill, 1995.

[234] S. JULIER UND J. UHLMANN. **A New Extension of the Kalman Filter to Nonlinear Systems**. *Proceedings of SPIE*, **3**(3):182–193, 1997.

[235] S. J. JULIER UND J. K. UHLMANN. **Unscented Filtering and Nonlinear Estimation**. *Proceedings of the IEEE*, **92**(3):401–422, 2004.

[236] G. KITAGAWA. **Non-Gaussian State-Space Modeling of Nonstationary Time Series**. *Journal of the American Statistical Association*, **82**(400):1032–1041, 1987.

[237] B. CARLIN, N. POLSON, UND D. STOFFER. **A Monte Carlo Approach to Nonnormal and Nonlinear State-Space Modeling**. *Journal of the American Statistical Association*, **87**(418):493–500, 1992.

[238] N. GORDON, D. SALMOND, UND A. SMITH. **Novel approach to nonlinear/non-Gaussian Bayesian state estimation**. *IEE Proceedings F – Radar and Signal Processing*, **140**(2):107–113, 1993.

[239] K. KANAZAWA, D. KOLLER, UND S. RUSSELL. **Stochastic simulation algorithms for dynamic probabilistic networks**. In *Proceedings of the 11th Annual Conference on Uncertainty in Artificial Intelligence*, Seiten 346–351, 1995.

[240] G. KITAGAWA. **Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models**. *Journal of Computational and Graphical Statistics*, **5**(1):1–25, 1996.

[241] C. BERZUINI, N. BEST, W. GILKS, UND C. LARIZZA. **Dynamic Conditional Independence Models and Markov Chain Monte Carlo Methods**. *Journal of the American Statistical Association*, **92**(440):1403–1412, 1997.

[242] J. CARPENTER, P. CLIFFORD, UND P. FEARNHEAD. **Improved particle filter for nonlinear problems**. *IEE Proceedings – Radar, Sonar and Navigation*, **146**(1):2–7, 1999.

[243] M. ISARD UND A. BLAKE. **ICONDENSATION: Unifying low-level and high-level tracking in a stochastic framework**. In *Proceedings of the 5th European Conference on Computer Vision*, Seiten 893–908, 1998.

[244] U. GRENANDER, Y. CHOW, UND D. M. KEENAN. **Hands: A Pattern Theoretic Study of Biological Shapes**. Springer, 1991.

[245] M. K. PITT UND N. SHEPHARD. **Filtering via Simulation: Auxiliary Particle Filters**. *Journal of the American Statistical Association*, **94**(446):590–599, 1999.

[246] M. PITT UND N. SHEPHARD. **Auxiliary variable based particle filters**, Seiten 273–293. Springer, 2001.

[247] J. DEUTSCHER, A. BLAKE, UND I. REID. **Articulated Body Motion Capture by Annealed Particle Filtering**. In *Proceedings of the 2000 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Seiten 126–133, 2002.

[248] A. DOUCET, N. FREITAS, K. MURPHY, UND S. RUSSELL. **Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks**. In *Proceedings of the 16th Annual Conference on Uncertainty in Artificial Intelligence*, Seiten 176–183, 2000.

[249] G. CASELLA UND C. P. ROBERT. **Rao-Blackwellisation of sampling schemes**. *Biometrika*, **83**(1):81–94, 1996.

[250] M. BREITENSTEIN, F. REICHLIN, B. LEIBE, E. KOLLER-MEIER, UND L. VAN GOOL. **Robust tracking-by-detection using a detector confidence particle filter**. In *Proceedings of the 12th IEEE International Conference on Computer Vision*, Seiten 1515–1522, 2009.

[251] C.-C. HUANG UND S.-J. WANG. **A cascaded hierarchical framework for moving object detection and tracking**. In *Proceedings of the 2010 IEEE International Conference on Image Processing*, 2010.

[252] X. SUN, H. YAO, S. ZHANG, UND S. LIU. **Adaptive particle filter based on energy field for robust object tracking in complex scenes**. In *Proceedings of the 11th Pacific Rim conference on Advances in multimedia information processing: Part I*, Seiten 437–448, 2010.

[253] S. ZHOU, R. CHELLAPPA, UND B. MOGHADDAM. **Visual Tracking and Recognition Using Appearance-Adaptive Models in Particle Filters**. *IEEE Transactions on Image Processing*, **13**:1434–1456, 2004.

[254] B. NORTH, A. BLAKE, M. ISARD, UND J. RITTSCHER. **Learning and Classification of Complex Dynamics**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(9):1016–1034, 2000.

[255] D. MIKAMI, K. OTSUKA, UND J. YAMATO. **Memory-Based Particle Filter for Tracking Objects with Large Variation in Pose and Appearance**. In *Proceedings of the 11th European Conference on Computer Vision*, Seiten 215–228, 2010.

[256] D. BARBER. **Machine Learning: A Probabilistic Approach**, 2006.

[257] F. BASHIR UND F. PORIKLI. **Performance Evaluation of Object Detection and Tracking Systems**. In *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, 2006.

[258] T. REHRL, J. BLUME, A. BANNAT, G. RIGOLL, UND F. WALLHOFF. **On-line Learning of Dynamic Gestures for Human-Robot Interaction**. In *35th German Conference on Artificial Intelligence, KI 2012*, 2012.

[259] ONLINE. **MetraLabs**. http://metralabs.com, 2013. besucht im Februar 2013.

[260] S. KOHLBECHER, E. WIESE, K. BARTL, J. BLUME, A. BANNAT, UND E. SCHNEIDER. **Studying Gaze-based Human Robot Interaction: An Experimental Platform**. In *7th ACM/IEEE International Conference on Human-Robot Interaction*, 2012.

[261] V. GONZALEZ-PACHECO, A. RAMEY, F. ALONSO-MARTIN, A. CASTRO-GONZALEZ, UND M. SALICHS. **Maggie: A Social Robot as a Gaming Platform**. *International Journal of Social Robotics*, **3**:371–381, 2011.

[262] T. HAMADA, H. OKUBO, K. INOUE, J. MARUYAMA, H. ONARI, Y. KAGAWA, UND T. HASHIMOTO. **Robot therapy as for recreation for elderly people with dementia – Game recreation using a pet-type robot –**. In *The 17th IEEE International Symposium on Robot and Human Interactive Communication*, Seiten 174–179, 2008.

[263] E. BROX, L. LUQUE, G. EVERTSEN, UND J. HERNANDEZ. **Exergames for elderly: Social exergames to persuade seniors to increase physical activity**. In *5th International Conference on Pervasive Computing Technologies for Healthcare*, Seiten 546–549, 2011.

[264] ONLINE. **Akinator**. http://de.akinator.com, 2013. besucht im Februar 2013.

[265] ONLINE. **Elokence**. http://elokence.com/fr, 2013. besucht im Februar 2013.

[266] ONLINE. **CLT Sprachtechnologie**. http://www.clt-st.de, 2013. besucht im März 2013.

[267] T. REHRL, J. GEIGER, M. GOLCAR, S. GENTSCH, J. KNOBLOCH, G. RIGOLL, K. SCHEIBL, W. SCHNEIDER, S. IHSEN, UND F. WALLHOFF. **The Robot ALIAS as a Database for Health Monitoring for Elderly People**. In *Tagungsband des 6. Deutschen Ambient Assisted Living (AAL 2013) Kongresses*, 2013.

[268] A. PAPOULIS. **Probability, Random Variables, and Stochastic Processes**. Mc-Graw Hill, 1991.

[269] R. G. COWELL, A. P. DAWID, S. L. LAURITZEN, UND D. J. SPIEGELHALTER. **Probabilistic Networks and Expert Systems**. Springer, 1999.

[270] R. E. TARJAN UND M. YANNAKAKIS. **Simple linear-time algorithms to test chordality of graphs, test acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs**. *SIAM Journal on Compuing*, **13**(3):566–579, 1984.