

Jörg Moldenhauer



# Automatische Erkennung von Zuständen in Anthropomatiksystemen



universitätsverlag karlsruhe



Jörg Moldenhauer

**Automatische Erkennung von Zuständen in  
Anthropomatiksystemen**



# **Automatische Erkennung von Zuständen in Anthropomatik- systemen**

von  
Jörg Moldenhauer



---

universitätsverlag karlsruhe

Dissertation, Universität Karlsruhe (TH)  
Fakultät für Informatik, 2005

## Impressum

Universitätsverlag Karlsruhe  
c/o Universitätsbibliothek  
Straße am Forum 2  
D-76131 Karlsruhe  
www.uvka.de



Dieses Werk ist unter folgender Creative Commons-Lizenz  
lizenziert: <http://creativecommons.org/licenses/by-nc-nd/2.0/de/>

Universitätsverlag Karlsruhe 2006  
Print on Demand

ISBN-13: 978-3-86644-068-5  
ISBN-10: 3-86644-068-5





# Automatische Erkennung von Zuständen in Anthropomatiksystemen

Zur Erlangung des akademischen Grades eines  
Doktors der Ingenieurwissenschaften

der Fakultät für Informatik  
der Universität Fridericiana zu Karlsruhe (TH)

**genehmigte**

**Dissertation**

von

**Jörg Moldenhauer**

aus Karlsruhe

Tag der mündlichen Prüfung: 20. Dezember 2005  
Erster Gutachter: Prof. Dr. Rüdiger Dillmann  
Zweiter Gutachter: Prof. Dr. Jürgen Beyerer



# Danksagung

An dieser Stelle möchte ich mich bei all denjenigen bedanken, die zum Gelingen meiner Arbeit beigetragen haben. Mein erster Dank gilt meinem Doktorvater Professor Dr. Thomas Beth, der mir am IAKS die Arbeit an spannenden Projekten ermöglicht hat und der mir in gemeinsamen Gesprächen und Seminaren neue Sichtweisen zur Lösung von wissenschaftlichen Aufgabenstellungen gegeben hat. Leider ist Professor Beth zu früh verstorben, um meine Arbeit bis zum Ende zu begleiten. Mein besonderer Dank richtet sich deshalb an Professor Dr. Rüdiger Dillmann, der nach dem Tode von Professor Beth die Betreuung meiner Promotion übernommen hat und der durch die gemeinsamen Projekte ebenfalls großen Einfluss auf meine Arbeit hatte. Mein weiterer Dank gilt Professor Dr. Jürgen Beyerer für die spontane Übernahme des Korreferats zu meiner Dissertation.

Mein Dank geht auch an die Kollegen am IAKS für die vielen anregenden Diskussionen, die wir nicht nur beim gemeinsamen Kuchenessen hatten. Besonders zu erwähnen sind Matthias Hahn, der mit mir die Stellung der „Mediziner“ gehalten hat, und Ingo Boesnach, mit dessen eifriger Zusammenarbeit VAMOS zum Laufen gebracht werden konnte. Herzlich möchte ich mich auch bei meinem ehemaligen Kollegen Dr. Martin Haimerl dafür bedanken, dass er mir durch die enge Zusammenarbeit den Weg zur Anthropomatik geebnet hat. Neben den vielen Kollegen, mit denen ich in den zahlreichen Projekten der vergangenen Jahre zusammenarbeiten konnte, bedanke ich mich bei Andreas Fischer und Thorsten Stein vom IfSS und Regine Becher vom ITEC für die freundschaftliche Zusammenarbeit im SFB 588. Das gilt auch für Professor Dr. Veit Wank, der mir das wichtigste Know-How zur Biomechanik und Bewegungsmessung vermittelt hat. Aus den Medizinprojekten bedanke ich mich bei PD Dr. Uwe Spetzger von der Neurochirurgie im Städtischen Klinikum Karlsruhe für die Durchführung und Erläuterung der Bohrversuche und bei Professor Dr. Ulrich Mende von der Radiologischen Universitätsklinik Heidelberg für die Bereitstellung und die umfassende Erklärung der radiologischen Ultraschallbilder.

Bei allen meinen Freunden bedanke ich mich dafür, dass sie in den vergangenen Jahren zu mir gestanden haben und für die notwendige Ablenkung von der Arbeit gesorgt haben. Besonders bedanke ich mich bei Michael Feger, der mir durch das Korrekturlesen geholfen hat, und bei Dr. Dirk Heuzeroth, der mir jederzeit ein offenes Ohr für Probleme geschenkt hat. Meinen größten Dank richte ich abschließend an meine liebe Freundin Anja Dörich, die mir besonders in der nervenaufreibenden letzten Phase die notwendige Unterstützung und Aufmunterung für meine Arbeit gegeben hat, und an meine Eltern, die mir während des Studiums und der Promotion immer den erforderlichen Rückhalt gegeben haben.



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Gliederung der Arbeit . . . . .	2
<b>2</b>	<b>Analyse von Anthropomatikdaten</b>	<b>5</b>
2.1	Modellierung von Zeitreihen . . . . .	5
2.2	Markov-Modelle . . . . .	7
2.3	Hidden-Markov-Modelle . . . . .	9
2.3.1	Diskrete Hidden-Markov-Modelle . . . . .	10
2.3.2	Kontinuierliche Hidden-Markov-Modelle . . . . .	10
2.3.3	Grundlegende Aufgaben und Algorithmen . . . . .	11
2.3.4	Numerische Probleme beim Einsatz von HMMs . . . . .	20
2.4	Analyse von Bohrgeräuschen . . . . .	21
2.4.1	Problemstellung und Forschungsstand . . . . .	21
2.4.2	Datenakquisition . . . . .	24
2.4.3	Generierung von Audiomerkmale . . . . .	24
2.4.4	Automatische Analyse von Bohrgeräuschen . . . . .	27
2.4.5	Einsatz von HMMs . . . . .	30
2.5	Zusammenfassung . . . . .	35
<b>3</b>	<b>Zustandserkennung mit Hidden-Markov-Modellen</b>	<b>39</b>
3.1	Ableitung von Prozessinformationen . . . . .	39
3.2	HMM-Strukturen . . . . .	39
3.3	Bestimmung von versteckten Zuständen . . . . .	42

3.4	Analyse medizinischer Ultraschallbilder . . . . .	43
3.4.1	Physik des Schalls . . . . .	45
3.4.2	Medizinische Ultraschallbilder . . . . .	49
3.4.3	Stand der Forschung . . . . .	53
3.4.4	Modellierung mit Hidden-Markov-Modellen . . . . .	54
3.4.5	Klassifikation von Bildspalten . . . . .	57
3.4.6	Rekonstruktion von Gewebeinformation aus Zuständen . . . . .	59
3.4.7	Stabilisierung der Zustände durch Merkmalsaufbereitung . . . . .	61
3.5	Zusammenfassung . . . . .	62
<b>4</b>	<b>Merkmalsgenerierung und Modellkonstruktion</b>	<b>65</b>
4.1	Problemstellung . . . . .	65
4.2	Stand der Forschung . . . . .	66
4.3	Methoden der Datenakquisition . . . . .	68
4.3.1	Videobasierte Bewegungserfassung . . . . .	69
4.3.2	Magnetisches Tracking . . . . .	71
4.3.3	Datenhandschuh . . . . .	74
4.3.4	Weitere Verfahren . . . . .	75
4.4	Merkmale zur Beschreibung von Bewegungen . . . . .	76
4.4.1	Darstellung von Bewegungen durch Winkeltrajektorien . . . . .	77
4.4.2	Erweiterte Bewegungsmerkmale durch Wavelets . . . . .	78
4.4.3	Analyse und Bewertung von Merkmalstrajektorien . . . . .	79
4.4.4	Bedeutung für Sport und Medizin . . . . .	84
4.5	Modelle zur Beschreibung von Bewegungen . . . . .	86
4.5.1	Bewegungsdaten . . . . .	86
4.5.2	Erkennung von Personen . . . . .	89
4.5.3	Länge der Beobachtungssequenzen . . . . .	91
4.5.4	Erkennung von Rahmenbedingungen . . . . .	92
4.5.5	Vergleich mit Elman-Netzen . . . . .	93
4.5.6	Erkennung von Bewegungsphasen mittels Klassifikation . . . . .	95
4.6	Geeignete Strukturen und Parameter für HMMs . . . . .	101
4.7	Komposition von HMMs . . . . .	105
4.7.1	Der einfache Kompositionsalgorithmus . . . . .	107

---

4.7.2	Erweiterungen für alternative Zustandsabfolgen . . . . .	109
4.7.3	Bewertung der Analyse mit zusammengesetzten HMMs . . . . .	111
4.8	Zusammenfassung . . . . .	116
<b>5</b>	<b>Zustandserkennung in der Praxis</b>	<b>119</b>
5.1	Eigenschaften von VAMOS . . . . .	119
5.2	Aufbau und Architektur von VAMOS . . . . .	121
5.3	Analyse von Bewegungen in VAMOS . . . . .	123
5.3.1	Bereitstellung von Daten und Merkmalen . . . . .	124
5.3.2	Erkennung von Bewegungsphasen . . . . .	126
5.3.3	Einfache Synthese von Bewegungen . . . . .	127
5.4	Zusammenfassung . . . . .	128
<b>6</b>	<b>Zusammenfassung und Ausblick</b>	<b>131</b>
<b>A</b>	<b>Wavelet-Transformation</b>	<b>135</b>
A.1	Multiskalenanalyse . . . . .	135
A.2	Dekomposition und Rekonstruktion . . . . .	137
A.3	Wavelet-Basen . . . . .	139
A.3.1	Daubechies-Wavelets . . . . .	139
A.3.2	Haar-Wavelets . . . . .	141
A.4	Wavelets als Differentialoperator . . . . .	141
<b>B</b>	<b>Neuronale Netze</b>	<b>143</b>
B.1	Grundlegende Komponenten . . . . .	143
B.2	Transferfunktionen . . . . .	144
B.3	Netzstrukturen und Klassifikation . . . . .	146
B.3.1	Ein- und Mehrschichtige Netze . . . . .	146
B.3.2	Einlernen der Netz-Parameter . . . . .	149
B.3.3	Rückgekoppelte Neuronale Netze . . . . .	151
B.3.4	Netze mit radialen Basisfunktionen . . . . .	152

<b>C Support-Vector-Machines</b>	<b>155</b>
C.1 Grundlagen . . . . .	155
C.2 Verwendung von Kernen für SVMs . . . . .	155
C.3 Bestimmung der Parameter zur Klassifikation . . . . .	157
<b>Abbildungsverzeichnis</b>	<b>159</b>
<b>Tabellenverzeichnis</b>	<b>163</b>
<b>Symbolverzeichnis</b>	<b>165</b>
<b>Abkürzungsverzeichnis</b>	<b>173</b>
<b>Literaturverzeichnis</b>	<b>175</b>
<b>Eigene Arbeiten</b>	<b>191</b>
<b>Index</b>	<b>195</b>

# Kapitel 1

---

## Einleitung

---

### 1.1 Motivation

Die Anthropomatik ist eine junge Forschungsrichtung, deren Zielsetzung es ist, die komplexe Funktionsweise des Menschen zu verstehen und eine optimale Integration von Menschen und Prozessen in ihrer Umwelt zu schaffen. Hierbei sollen sich nicht die Menschen an die Prozesse anpassen, sondern es soll vielmehr eine Abstimmung der Prozesse hinsichtlich der Bedürfnisse des Menschen erfolgen. Neben vielen anderen Wissenschaften wie Psychologie, Soziologie, Bewegungs- und Sportwissenschaft, Neurowissenschaften, Biologie, Medizin, Mathematik, Kybernetik und Physik liefert vor allem die Informatik wesentliche Beiträge, um die Ziele der Anthropomatik zu erreichen. Die wichtigste Rolle spielen dabei nicht nur die Informationstheorie, sondern insbesondere auch die Bereiche wie Kognition, Kommunikation oder Robotik. Nicht umsonst lässt sich der Kunstbegriff Anthropomatik aus dem griechischen Wort *anthropos* für Mensch und der Wissenschaftsbezeichnung *Informatik* ableiten.

Zur Analyse eines anthropomatischen Systems, bei dem also der Mensch mit den übrigen Systemkomponenten interagiert, ist von zentraler Bedeutung, wie sich Zustände in diesem System beschreiben lassen und wie diese auch automatisch mit dem Rechner erkannt werden können. Hierzu müssen die komplexen Prozesse in diesem System modelliert werden. Die Probleme hierbei sind, dass keine genauen Beschreibungen der Prozesse und Messdaten existieren, und somit keine geeigneten Modellierungen gefunden werden können, dass direkte Prozessgrößen nicht direkt messbar sind und stattdessen mit Hilfe von indirekten Beobachtungen Rückschlüsse auf die Systemzustände gezogen werden müssen oder dass herkömmliche Modellierungsverfahren, wie z. B. Differentialgleichungen, aus Effizienzgründen nicht für diese Zwecke herangezogen werden können.

In dieser Arbeit werden stattdessen probabilistische Modelle untersucht, die eine Beschreibung komplexer dynamischer Systeme ermöglichen. Den Modellen ist gemein, dass sie ein Signal von direkt messbaren Größen erfassen und damit Rückschlüsse auf den zugrunde liegenden Prozess mit Hilfe stochastischer Methoden ermöglichen. Zu den mächtigsten Vertretern für diese Art der Modellierung gehören Hidden-Markov-Modelle. Das bekann-

teste Anwendungsgebiet für diese Verfahren ist die Spracherkennung. Jedoch haben die Verfahren bereits auch in anderen Gebieten Verbreitung gefunden, z. B. bei der Analyse von EEG-Signalen oder der Aufwertung von Genomdaten. In dieser Arbeit werden die Methoden zur Analyse von Anthropomatiksystemen eingesetzt. Dabei werden explizit die Zustände untersucht, die die Hidden-Markov-Modelle bei der Klassifikation einer anliegenden Beobachtungssequenz einnehmen. Neben der reinen Modellierung, d. h. der Identifikation geeigneter Modellstrukturen und Definition passender Modellparameter, ist auch die Auswahl und Generierung aussagekräftiger Merkmale entscheidend für die Leistungsfähigkeit der im Rahmen dieser Arbeit entstandenen Modelle. Als Beispiele für Anthropomatiksysteme dienen die Bewertung von Bohrgeräuschen, wie sie der Chirurg bei Bohrvorgängen bei Eingriffen an der Wirbelsäule wahrnimmt und zur Beurteilung des Bohrfortschrittes intuitiv nutzt, die visuelle Befundung von medizinischen Ultraschallbildern, die von Radiologen nur über langjährige Erfahrung und einem fundierten Wissen über die Anatomie und den Aufnahmeprozess durchgeführt werden kann, sowie die Analyse von menschlichen Bewegungen, die insbesondere dann benötigt wird, wenn humanoide Roboter dem Menschen auch bei der Durchführung von Alltagstätigkeiten sinnvoll zur Hand gehen sollen.

## 1.2 Gliederung der Arbeit

Im einführenden Kapitel 2 werden zunächst die Grundlagen für die automatische Erkennung von Zuständen in Anthropomatiksystemen geschaffen. Dabei handelt es sich um die Beschreibung der allgemeinen Vorgehensweise bei der Modellierung und Analyse von komplexen Systemen, die Nutzung und Eigenschaften von Klassifikationsverfahren, sowie die Rolle geeigneter Merkmale für Erkennungsaufgaben. Im Detail wird auf Hidden-Markov-Modelle und deren Einsatz bei der Klassifikation von Zeitserien eingegangen. Am Beispiel der Analyse von Bohrgeräuschen in der Wirbelsäulenchirurgie wird die besondere Eignung von Hidden-Markov-Modellen für die Bewertung von Veränderungen in Zuständen von Anthropomatikdaten im Vergleich zu den verbreiteten Klassifikationsverfahren mit Neuronalen Netzen und Support-Vector-Machines gezeigt.

Darauf aufbauend werden in Kapitel 3 die in dieser Arbeit entwickelten Methoden zur Analyse von Anthropomatiksystemen durch die Rekonstruktion und Untersuchung der versteckten Zustände in Hidden-Markov-Modellen vorgestellt. Die Vorgehensweise wird am Beispiel der Interpretation von medizinischen Ultraschall-Bildern mit Maßstäben und Zielsetzungen erläutert, die der Radiologe bei seiner Befundung einsetzt.

In Kapitel 4 werden die entwickelten Analysemethoden ausführlich an mehreren Beispielen zur Erkennung und Auswertung von menschlichen Bewegungen getestet und evaluiert. Insbesondere werden dabei auch neue Methoden zur Generierung geeigneter Merkmale für Beobachtungssequenzen bei Hidden-Markov-Modellen geschaffen und die Wahl von Strukturen und Parametrisierungen der Modelle diskutiert. Aufbauend auf den Erkenntnissen wird ein Algorithmus zur automatischen Komposition von Hidden-Markov-Modelle entwickelt, mit dem sich komplexe Modelle auf Basis von elementaren Modellen für Schlüssel-

bewegungen erzeugen lassen. Im gleichen Zusammenhang wird die Effizienz entwickelter Modelle mit herkömmlichen Klassifikationsmethoden verglichen.

Im anschließenden Kapitel 5 wird auf praktische Umsetzung der besprochenen Methoden und deren Anwendung in einer realen Applikation eingegangen. Die Implementierung erfolgt durch Integration von Software-Modulen in das Visualisierungs-, Analyse- und Modellierungs-Framework VAMOS, das zu einem großen Teil im Rahmen dieser Arbeit entstanden ist.

Abschließend werden in Kapitel 6 die wichtigsten Aspekte und Ergebnisse der Arbeit zusammengefasst und ein Ausblick auf zukünftige Erweiterungen und Anwendungen der entwickelten Methoden zur Erkennung von Zuständen in Anthropomatiksystemen gegeben.

In den Anhängen A bis C werden zusätzlich Grundlagen zu Wavelets, Neuronalen Netzen und Support-Vector-Machines angegeben, die nicht direkter Bestandteil dieser Arbeit sind, aber zur Entwicklung und Bewertung der Methoden benötigt werden.



# Kapitel 2

---

## Analyse von Anthropomatikdaten

---

### 2.1 Modellierung von Zeitreihen

Die Analyse von Messreihen, bei denen der zeitliche Aspekt im Vordergrund steht, die so genannte Zeitreihenanalyse, spielt eine wichtige Rolle in den unterschiedlichsten wissenschaftlichen Bereichen, wie der Physik, der Biologie, der Medizin oder auch der Ökonomie, um nur einige Beispiele aufzuführen (siehe z. B. [57], [146], [160]). Die dort entwickelten Methoden lassen sich auch auf die in dieser Arbeit behandelten Anthropomatikdaten anwenden. So können beispielsweise in einem ersten Schritt grundlegende Charakteristika der betrachteten Daten mit Hilfe der beschreibenden Statistik und der sich aus ihr entwickelten explorativen Datenanalyse (EDA) gewonnen werden [120]. Darüber hinaus werden Modelle für die Systeme und Prozesse entwickelt, aus denen die Daten hervorgehen. Die Modellierung kann auf vielfältige Weise geschehen. In der Physik werden zu diesem Zweck häufig Differentialgleichungen (DGL) eingesetzt. Bei dieser Form der Modellierung sind die betrachteten Größen und die zu verwendenden physikalischen Gesetzmäßigkeiten bekannt. Die Modellierung physikalischer und technischer Prozesse ist Gegenstand langjähriger Forschung und daher meist sehr gut untersucht. Problem bei dieser Form der Modellierung ist aber, dass die Modellierung aufgrund der Komplexität der betrachteten Systeme oft nur schwer beherrschbar ist und dass die Lösung der DGLs nicht immer auf analytischem Wege gefunden werden können. Stattdessen werden in der Regel iterative Verfahren zur numerischen Lösung der Gleichungssysteme verwendet. Die Verfahren sind zumeist sehr rechenaufwendig und daher nur einsetzbar, wenn die Anforderungen an die Rechenzeit nicht zu hoch sind. Um genügend Rechenkapazität zur Verfügung zu stellen, werden zunehmend auch Methoden der Parallelverarbeitung genutzt. Dennoch scheitert die Modellierung mit DGLs allzu oft an den genannten Problemen [69]. Ein gutes Beispiel für den hohen Rechenaufwand bei der Analyse von Systemen, die mit DGLs modelliert sind, stellt die im späteren Abschnitt 3.4.3 beschriebene Simulation der Wellenausbreitung bei der Ultraschallbildgebung dar.

Eine Alternative stellen stochastische Modellierungen dar, die auch aus der digitalen Signalverarbeitung und dem maschinellen Lernen bekannt sind. Bei ihnen werden anhand

von gemessenen Beobachtungen bzw. Trainingsdaten die stochastischen Eigenschaften der zugrunde liegenden Prozesse ermittelt und Modellparameter bestimmt. Zielsetzung ist dabei, Funktionen zu finden, mit denen die Zustände auf der Basis vorliegender Beobachtungen berechnet werden können. Neben den bereits genannten Literaturstellen zur Analyse von Zeitreihen liefern z. B. auch [134], [66] oder [32] weitere Details hierzu. Ein Beispiel für diese Art der Modellierung bilden *lineare Modelle*, bei denen ein linearer Zusammenhang zwischen dem aktuellen Modellzustand und den bisherigen Zuständen bzw. Beobachtungen besteht. In einem einfachen Fall können mit diesen Modellen so genannte *autoregressive Prozesse (AR-Prozesse)* nachgebildet werden. Der Zustand  $q_t$  zum aktuellen Zeitpunkt  $t$  berechnet sich für einen AR-Prozess der Ordnung  $D$  aus den  $D$  vergangenen Zuständen wie folgt:

$$q_t = C + \sum_{\tau=1}^D w_{\tau} q_{t-\tau} + g_t, \quad (2.1)$$

wobei die Gewichtungen  $w_{\tau}$  die Modellparameter bilden. Der Term  $g_t$  stellt einen weißen Rauschprozess dar. Die Konstante  $C$  ist ein zusätzlicher Offset, der in der Regel aber in der Literatur zur Vereinfachung vernachlässigt wird. Voraussetzung für die AR-Modellierung ist, dass dem System ein *stationärer* Prozess zugrunde liegt.<sup>1</sup> Eine Verallgemeinerung zu den AR-Modellen sind *ARMA-Modelle* der Ordnung  $(D_1, D_2)$ , bei denen neben den vergangenen  $D_1$  Zuständen im AR-Anteil auch die vergangenen  $D_2$  Rauschterme durch einen *MA-Anteil* (engl. *moving average*) mit den jeweiligen Gewichtungen  $w_1$  und  $w_2$  berücksichtigt werden:

$$q_t = \sum_{\tau=1}^{D_1} w_{1,\tau} q_{t-\tau} + \sum_{\tau=1}^{D_2} w_{2,\tau} g_{\tau}. \quad (2.2)$$

Auf der Basis von ARMA-Modellen wurden bereits viele erweiterte Modellierungsansätze entwickelt. Beispielsweise erlauben so genannte *ARIMA-Modelle*, einen Trend in nicht-stationären Zeitreihen auszugleichen. Mehr Details hierzu und weitere aus den beschriebenen Modellen abgeleitete Formen finden sich z. B. in [146], [41], [93].

Im Allgemeinen muss zusätzlich zur zeitlichen Entwicklung der Zustandssequenz auch die Entwicklung der gemessenen Beobachtungssequenz betrachtet werden. Ein entsprechendes allgemeineres Modell, das eine Abhängigkeit zwischen Beobachtungen und Zuständen beinhaltet, wird mit Hilfe der folgenden Gleichungen ausgedrückt:

$$q_{t+1} = f(q_t) + w_t g_{1,t+1}, \quad (2.3)$$

$$o_t = h(q_t) + g_{2,t}. \quad (2.4)$$

Durch die *Zustandsgleichung* (2.3) wird der Übergang des Systems von Zustand  $q_t$  zum Zustand  $q_{t+1}$  mit Hilfe der Funktion  $f$  berechnet. Zusätzlich wird ein Rauschterm  $g_{1,t}$  zum vergangenen Zeitpunkt über die Gewichtungen  $w_t$  mit einbezogen. Die *Beobachtungsgleichung* (2.4) repräsentiert den Messprozess und berechnet mit der Funktion  $h$  aus dem

---

<sup>1</sup>Der Begriff *Stationarität* wird im folgenden Abschnitt bei der Betrachtung von Markov-Modellen noch genauer erklärt.

Zustand  $q_t$  sowie einem weiteren Rauschterm  $g_{2,t}$  die Beobachtung  $o_t$ . Sind die Funktionen  $h$  und  $f$  in den Gleichungen (2.3) und (2.4) linear, kann das System mit einem so genannten *Kalman-Filter* modelliert werden. Kalman-Filter werden heute bevorzugt zur Berechnung und Prädiktion von Trajektorien, z. B. bei der Positionsbestimmung und Navigation im Flugzeug- und Automobilbereich oder bei der Verfolgung von Objekten bzw. Markerpunkten in Videoaufnahmen, verwendet. Mehr Details zu Kalman-Filter und deren Erweiterungen sind u. a. in [30] beschrieben.

Zur Modellierung nicht zwangsläufig linearer Zusammenhänge müssen andere Methoden der Modellierung benutzt werden. In dieser Arbeit stehen Hidden-Markov-Modelle (HMM) im Vordergrund, die aber auch mit Alternativen wie Neuronalen Netzen oder Support-Vector-Machines verglichen werden. Mittelpunkt der Modellierung mit HMMs bilden wie bei den bereits beschriebenen Modellierungsmethoden erneut die zeitlichen Zustände des Systems. Sie ergeben sich aus der Folge  $Q = q_1, q_2, \dots, q_T$  von Zufallsvariablen  $q_t \in S$  zu den Zeitpunkten  $t = 1, \dots, T$ . Generell kann die Ereignismenge  $S$  sowohl diskret als auch kontinuierlich sein. Da in dieser Arbeit jedoch nur diskrete Zustände bei der Analyse der Systeme betrachtet werden sollen, beschränkt sich  $S$  im Folgenden auf diskrete Mengen. Die Menge  $S$  bestehe aus den  $N_S := |S|$  Elementen  $s_1, \dots, s_{N_S}$ . Zur Vereinfachung der Notation wird auch lediglich der Index  $1 \leq i \leq N_S$  eines Elementes  $s_i$  zur Benennung des Zustandes verwendet. Bevor auf die Kopplung von Beobachtungen und Zuständen bei HMMs eingegangen wird, soll zunächst im folgenden Abschnitt auf das zugrunde liegende Zustandsmodell eingegangen werden.

## 2.2 Markov-Modelle

Im Allgemeinen hängt bei stochastischen Prozessen die zur Zufallsvariable  $q_t$  im Zeitpunkt  $t$  gehörende Verteilungsfunktion vom Zeitpunkt  $t$  selbst und den Zufallsvariablen  $\dots, q_{t-2}, q_{t-1}, q_{t+1}, q_{t+2}, \dots$  zu anderen Zeitpunkten ab. Um die Prozesse mit dem Rechner beherrschen zu können, werden diese allgemeinen Abhängigkeiten deutlich eingeschränkt. Es werden *stationäre kausale* stochastische Prozesse verwendet. Ein stochastischer Prozess heißt (*streng*) *stationär*, genau dann, wenn die stochastischen Eigenschaften des Prozesses invariant gegenüber Verschiebungen der Zeit sind [65], d. h. die Zufallsvariablen  $q_1, \dots, q_t$  und  $q_{1+\tau}, \dots, q_{t+\tau}$  für beliebige Zeitpunkte  $t$  und Verschiebungen  $\tau$  die gleichen Verbundverteilungen besitzen [41]. Ein stochastischer Prozess heißt *kausal*, genau dann, wenn die Zufallsvariable  $q_t$  nur von den Zufallsvariablen  $q_{t-1}, q_{t-2}, \dots$  zu den Zeitpunkten vor  $t$  abhängen [41].

Schränkt man die Kausalität des Prozesses noch weiter ein, indem man nur eine endliche Anzahl von vergangenen Zufallsvariablen für die Berechnung der aktuellen Zufallsvariablen zulässt, erhält man einen so genannten *Markov-Prozess*. Die Reduzierung der Abhängigkeit wird durch die *Markov-Eigenschaft* beschrieben: Ein Markov-Prozess, bei dem nur  $D$  vorhergehende Zufallsvariablen berücksichtigt werden, besitzt für die Zufallsvariable  $q_t$

zum Zeitpunkt  $t$  die Verteilungsfunktion

$$P(q_t = s_{i_t} | q_{t-1} = s_{i_{t-1}}, q_{t-2} = s_{i_{t-2}}, \dots, q_1 = s_{i_1}) = P(q_t = s_{i_t} | q_{t-1} = s_{i_{t-1}}, q_{t-2} = s_{i_{t-2}}, \dots, q_{t-D} = s_{i_{t-D}}) \quad \text{mit } s_{i_1 \dots t} \in S. \quad (2.5)$$

Dieser Markov-Prozess besitzt die *Ordnung*  $D$ . Insbesondere bei Markov-Prozessen der Ordnung  $D = 1$  spricht man hinsichtlich der Markov-Eigenschaft auch von der *Gedächtnislosigkeit* des Markov-Prozesses.

Ein Markov-Prozess der Ordnung  $D = 1$  wird als *einfacher* Markov-Prozess bezeichnet. Die zugehörige zeitliche Folge von Zufallsvariablen bildet eine *Markov-Kette*. Aus der Markov-Eigenschaft in Gleichung (2.5) folgt für die Markov-Kette die *Chapman-Kolmogorov-Gleichung*

$$P(q_{t_2} = s_{i_{t_2}} | q_{t_1} = s_{i_{t_1}}) = \sum_{s \in S} P(q_t = s | q_{t_1} = s_{i_{t_1}}) P(q_{t_2} = s_{i_{t_2}} | q_t = s) \quad (2.6)$$

für  $s_{t_1}, s_{t_2} \in S$  und  $t_1 < t < t_2$ , mit der sich die Wahrscheinlichkeit auch für den indirekten Übergang des zugrunde liegenden Prozesses von einem Zustand zu Zeitpunkt  $t_1$  in einen Zustand zu Zeitpunkt  $t_2$  über die dazwischen liegenden Zustände berechnen lässt.

Da der betrachtete Markov-Prozess stationär ist, lassen sich die Übergangswahrscheinlichkeiten zwischen  $N_S$  gegebenen Zuständen in Form einer *semistochastischen* Matrix  $\mathbf{A} = (a_{i,j}) \in \mathbb{R}^{N_S \times N_S}$  mit  $a_{i,j} = P(q_t = s_j | q_{t-1} = s_i)$  angeben. Für die Einträge einer semistochastischen Matrix gilt:

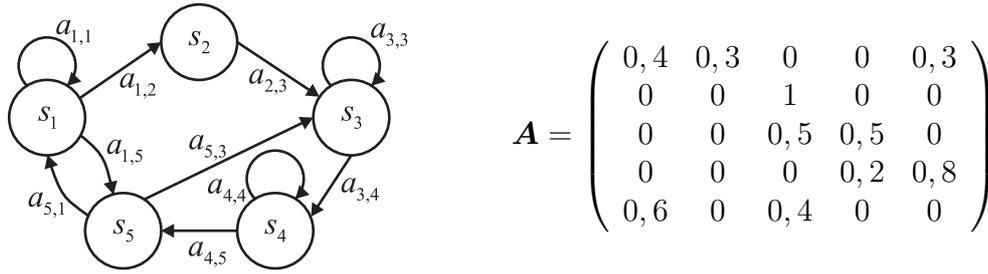
$$a_{i,j} \geq 0 \quad \text{und} \quad \sum_{i=1}^{N_S} a_{i,j} = 1. \quad (2.7)$$

Der zweite Teil von Gleichung (2.7) wird auch als *Zeilensummenkriterium* für semistochastische Matrizen bezeichnet. Matrix  $\mathbf{A}$  stellt eine probabilistische Adjazenzmatrix für den Graphen dar, der aus der Markov-Kette gebildet wird. In Abb. 2.1 wird ein Beispiel für eine Markov-Kette mit  $N_S = 5$  Zuständen gezeigt und die zugehörige Übergangsmatrix  $\mathbf{A}$  angegeben. Man beachte, dass Übergängen mit Wahrscheinlichkeit  $a_{i,j} = 0$  nicht in den Graphen eingezeichnet werden.

Markov-Ketten bilden Modelle für Systeme, bei denen Beobachtungen, z. B. physikalische Ereignisse, mit den Zuständen des Systems übereinstimmen und das Verhalten des Systems, d. h. die Zustandsübergänge des Systems, sich stochastisch beschreiben lässt. In Anlehnung an den zugrunde liegenden Markov-Prozess werden diese Modelle als *Markov-Modelle* bezeichnet. Ein Markov-Modell wird mit dem Tupel  $\kappa := (N_S, S, \mathbf{A}, \boldsymbol{\pi})$  definiert, wobei  $N_S$  die Anzahl der Zustände,  $S$  die Menge der Zustände und  $\mathbf{A}$  die Zustandsübergangsmatrix angeben. Zusätzlich wird eine *initiale Zustandsverteilung*

$$\boldsymbol{\pi} \in \mathbb{R}^{N_S} \quad \text{mit} \quad \pi_i = P(q_1 = s_i) \quad \text{und} \quad \sum_{i=1, \dots, N_S} \pi_i = 1 \quad (2.8)$$

für ein Modell vorgegeben, welche die Aufenthaltswahrscheinlichkeiten des Modells zu Beginn einer Beobachtungssequenz beschreibt.



**Abbildung 2.1:** Beispiel für den Graphen einer Markov-Kette mit  $N_S = 5$  Zuständen und die zugehörige Übergangsmatrix  $\mathbf{A}$ .

Mit diesen Modellen lassen sich bereits wesentliche Fragestellungen beantworten, wie z. B. die Bestimmung der Wahrscheinlichkeit, mit der ein System eine bestimmte Folge von Zuständen einnimmt, d. h. die Zustandsfolge  $Q = s_{i_1} s_{i_2} \dots s_{i_T}$  mit dem Modell zu beobachten ist. Diese Wahrscheinlichkeit  $P_\kappa(Q)$  berechnet sich wie folgt:

$$\begin{aligned} P_\kappa(Q) &:= P(q_1 = s_{i_1}, q_2 = s_{i_2}, \dots, q_T = s_{i_T}) \\ &= P(q_1 = s_{i_1}) \prod_{t=2, \dots, T} P(q_t = s_{i_t} | q_{t-1} = s_{i_{t-1}}) \\ &= \pi_{i_1} \prod_{t=2, \dots, T} a_{i_{t-1}, i_t}. \end{aligned} \quad (2.9)$$

Weiterhin lässt sich die Frage lösen, wie groß die Wahrscheinlichkeit ist, dass das System für  $T$  Zeitschritte im Zustand  $s_i$  *verweilt*, in dem es sich gerade befindet:

$$P_\kappa(q_1 = s_i, \dots, q_T = s_i, q_{T+1} \neq s_i) = (a_{i,i})^{T-1} (1 - a_{i,i}) =: P_i(T). \quad (2.10)$$

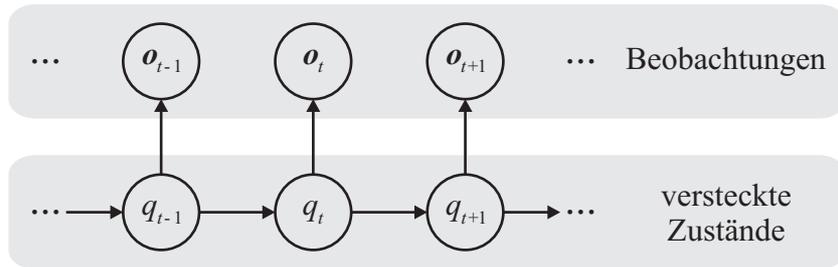
Mit  $P_i(T)$  wird eine Dichtefunktion für die *Verweildauer* in einem Zustand  $s_i$  definiert. Hierüber lässt sich als abschließendes Beispiel die Wahrscheinlichkeit für die mittlere Verweildauer in einem bestimmten Zustand berechnen:

$$\bar{T}_i = \sum_{T=1}^{\infty} T P_i(T) = \sum_{T=1}^{\infty} T (a_{i,i})^{T-1} (1 - a_{i,i}) = \frac{1}{1 - a_{i,i}}. \quad (2.11)$$

Zu Markov-Ketten und deren Anwendung gibt es bereits umfassend Literatur. Eine gute Übersicht zu diesem Thema ist in [8] zu finden.

## 2.3 Hidden-Markov-Modelle

Eine Erweiterung zu den im vorigen Abschnitt besprochenen Markov-Modellen bilden die so genannten *Hidden-Markov-Modelle (HMMs)*. Im Gegensatz zu Markov-Modellen



**Abbildung 2.2:** Abhängigkeit der Zustandsübergänge und Beobachtungen eines HMMs ausgerollt über mehrere Zeitschritte.

besitzen sie bedingte Wahrscheinlichkeitsverteilungen zur Berechnung der Beobachtungen in Abhängigkeit der jeweiligen Zustände des Modells. Sie verfügen konzeptionell über zwei miteinander verknüpfte stochastische Prozesse. Auf unterer Ebene liegt ein Markov-Modell vor, dessen Zustände jedoch nicht sichtbar, also *versteckt* (engl. *hidden*) sind. Mit Hilfe des darüber liegenden Ausgabeprozesses kann jedoch auf die versteckten Zustände rückgeschlossen werden. Da Beobachtungssequenzen  $O = o_1 o_2 \dots o_T$  und Zustandsequenzen  $Q = q_1 q_2 \dots q_T$  unterschieden werden, unterliegen HMMs nicht der Einschränkung von Markov-Ketten, dass die Zustände des Modells direkt messbaren Beobachtungen entsprechen, und eignen sich somit insbesondere für die Analyse von Anthropomatiksystemen. Abbildung 2.2 skizziert die Abhängigkeiten von Zustandsübergängen und Beobachtungen eines HMMs ausgerollt über mehrere Zeitschritte.

### 2.3.1 Diskrete Hidden-Markov-Modelle

Als Grundlage für die Definition von HMMs dienen Markov-Modelle  $\kappa := (N_S, S, \mathbf{A}, \boldsymbol{\pi})$ . Die Modelle  $\kappa$  werden um zusätzliche *Beobachtungswahrscheinlichkeiten* ergänzt, die zunächst für eine diskrete Menge von Beobachtungen in einer semistochastischen Matrix  $\mathbf{B} = (b_{i,k}) \in \mathbb{R}^{N_S \times N_V}$  gespeichert werden. Der Eintrag  $b_{i,k} := P(o_t = v_k | q_t = s_i)$  gibt die Wahrscheinlichkeit für die Beobachtung eines Symbols  $v_k$  aus der Menge aller  $N_V$  Beobachtungen  $V = \{v_1, v_2, \dots, v_{N_V}\}$  in Zustand  $s_i$  an. In der praktischen Realisierung handelt es sich bei der Matrix  $\mathbf{B}$  um eine *Lookup-Table*. Da die Beobachtungen aus einer diskreten Menge von Symbolen stammen, wird ein HMM mit Beobachtungswahrscheinlichkeiten  $\mathbf{B}$  als *diskretes HMM* bezeichnet und über das Tupel  $\lambda := (N_S, S, V, \boldsymbol{\pi}, \mathbf{A}, \mathbf{B})$  definiert.

### 2.3.2 Kontinuierliche Hidden-Markov-Modelle

Um Beobachtungen nicht nur aus einer endlichen Symbolmenge  $V$ , sondern auch kontinuierliche Beobachtungen  $\mathbf{o}$  aus einem Merkmalsraum  $\mathbf{V}$  zu erfassen, lassen sich diskrete HMMs so erweitern, dass die Wahrscheinlichkeit für die Beobachtung von  $\mathbf{o}$  in Zustand  $s_i$  direkt über eine Dichtefunktion  $b_i : \mathbf{V} \rightarrow [0, 1]$  berechnet wird. Im Zusammenhang

mit HMMs werden üblicherweise Dichten der Normalverteilungen und davon abgeleiteten *Mischverteilungen* (engl. *mixtures*) verwendet. Generell lassen sich jedoch beliebige Verteilungen mit logarithmisch konkaven oder ellipsensymmetrische Dichtefunktionen dazu heranziehen. Dieses Kriterium muss erfüllt sein, damit die Parameterabschätzung in Abschnitt 2.3.3.3 durchgeführt werden kann (siehe [129]). Zur Erfassung skalarer Beobachtungen  $o \in \mathbb{R}$  eignet sich die Dichtefunktion der Gauß-Verteilung

$$b_i(o) = \mathcal{N}(o|\mu_i, \sigma_i^2) = \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{-\frac{(o-\mu_i)^2}{2\sigma_i^2}} \quad (2.12)$$

mit spezifischem Erwartungswert  $\mu_i$  und Varianz  $\sigma_i^2$  für den Zustand  $s_i$ . Für vektorwertige Beobachtungen  $\mathbf{o} \in \mathbb{R}^D$ ,  $D \in \mathbb{N}$  kann entsprechend die mehrdimensionale Normalverteilung mit Erwartungsvektor  $\boldsymbol{\mu}_i \in \mathbb{R}^D$  und Kovarianzmatrix  $\mathbf{U}_i \in GL_D(\mathbb{R})$  verwendet werden:

$$b_i(\mathbf{o}) = \mathcal{N}(\mathbf{o}|\boldsymbol{\mu}_i, \mathbf{U}_i) = \frac{1}{\sqrt{2\pi|\mathbf{U}_i|}} e^{-\frac{1}{2}(\mathbf{o}-\boldsymbol{\mu}_i)^\top \mathbf{U}_i^{-1}(\mathbf{o}-\boldsymbol{\mu}_i)}. \quad (2.13)$$

Mit Hilfe von Mischverteilungen, deren Dichtefunktion sich aus der Überlagerung von  $N_M$  gewichteten Gaußschen Komponentendichten aus Gleichung (2.13) ergeben, lassen sich beliebige Dichtefunktionen approximieren (siehe dazu [43]). Die resultierenden Dichtefunktionen lauten

$$b_i(\mathbf{o}) = \sum_{m=1}^{N_M} w_{m,i} \mathcal{N}(\mathbf{o}|\boldsymbol{\mu}_{m,i}, \mathbf{U}_{m,i}) \quad (2.14)$$

mit Gewichtungen  $w_{i,m}$ , die erneut für Zustand  $s_i$  spezifisch sind und die Bedingung  $\sum_{m=1}^{N_M} w_{m,i} = 1$  erfüllen müssen. Durch diese Bedingung wird die Dichtefunktion  $b_i(\mathbf{o})$  normalisiert:

$$\int_{-\infty}^{\infty} b_i(\mathbf{o}) d\mathbf{o} = 1 \quad \text{für alle } i = 1, \dots, N_S. \quad (2.15)$$

Zusammengefasst wird damit ein *kontinuierliches HMM* über ein Tupel  $\lambda := (N_S, S, \boldsymbol{\pi}, \mathbf{A}, \mathbf{w}, \boldsymbol{\mu}, \mathbf{U})$  vollständig definiert. Da die Parameter für die Zustände und Beobachtung implizit durch die übrigen Modellparameter gegeben sind, wird stattdessen das HMM nur in der Kurzschreibweise  $\lambda = (\boldsymbol{\pi}, \mathbf{A}, \mathbf{w}, \boldsymbol{\mu}, \mathbf{U})$  definiert.

### 2.3.3 Grundlegende Aufgaben und Algorithmen

Da bei HMMs die Zustände des Modells nicht direkt mit den Beobachtungen assoziiert werden, wie dies bei Markov-Modellen der Fall ist, wurden einige grundlegende Algorithmen und Verfahren für die wichtigsten Fragestellungen bei der Datenanalyse mit Hilfe von HMMs in bestehenden Arbeiten entwickelt. Folgenden Problemstellungen sind für diese Arbeit von wesentlichem Interesse:

1. Bestimmung der Wahrscheinlichkeit  $P_\lambda(O)$  für die Beobachtung einer Observablensequenz  $O = \mathbf{o}_1, \dots, \mathbf{o}_2$  mit einem HMM  $\lambda$ ,

2. Klassifikation einer Observablensequenz  $O$  durch Bestimmung des Modells  $\lambda_{\max}$  aus einer Menge von Modellen  $\{\lambda_1, \dots, \lambda_{N_\lambda}\}$ , bei dem mit der höchsten Wahrscheinlichkeit  $O$  beobachtet werden kann,
3. Bestimmung der Zustandsfolge  $Q$ , die ein Modell bei der Beobachtung der Sequenz  $O$  am wahrscheinlichsten einnimmt,
4. Einlernen bzw. Anpassung der Modellparameter  $\boldsymbol{\pi}$ ,  $\mathbf{A}$  und  $\mathbf{B}$  bzw.  $\boldsymbol{w}, \boldsymbol{\mu}, \mathbf{U}$  des Modells  $\lambda$ , so dass die Wahrscheinlichkeit für eine Menge von Beobachtungssequenzen  $\{O_1, O_2, \dots\}$  maximiert wird.

Die aufgeführten Probleme 1, 2 und 4 werden in den folgenden Abschnitten ausführlicher erläutert und die zugehörigen Lösungsmethoden und Algorithmen beschrieben. Die Bestimmung der Zustandsfolge  $Q$  in Problem 3 spielt eine zentrale Rolle bei der in dieser Arbeit vorgestellten Analyse von Anthropomatiksystemen und wird daher gesondert in Kapitel 3 im Detail behandelt.

### 2.3.3.1 Erkennung von Beobachtungssequenzen

Grundlage für die Erkennung von Beobachtungssequenzen mit Hilfe von HMMs ist die Berechnung der Wahrscheinlichkeit  $P_\lambda(O)$  für die Beobachtung einer Observablensequenz  $O = \boldsymbol{o}_1 \boldsymbol{o}_2 \dots \boldsymbol{o}_T$  mit dem Modell  $\lambda$ . Diese wird im Folgenden auch *Beobachtungswahrscheinlichkeit des Modells* genannt. Unter der Annahme, dass  $\lambda$  bei der Produktion von  $O$  eine bestimmte Zustandsfolge  $Q = q_1 \dots q_T$  mit  $q_t \in S$  einnimmt, ergibt sich die Beobachtungswahrscheinlichkeit bezüglich dieser Zustandsfolge

$$P_\lambda(O|Q) = \prod_{t=1}^T P_\lambda(\boldsymbol{o}_t|q_t) = \prod_{t=1}^T b_{q_t}(\boldsymbol{o}_t), \quad (2.16)$$

wobei die Beobachtungswahrscheinlichkeit  $b_{q_t}(\boldsymbol{o}_t)$  eines kontinuierlichen HMMs angenommen wurde, die analog durch die Beobachtungswahrscheinlichkeit eines diskreten HMMs ersetzt werden kann. Die Wahrscheinlichkeit für die Einnahme der Zustandsfolge  $Q$  berechnet sich zu

$$P_\lambda(Q) = \pi_{q_1} \prod_{t=2}^T a_{q_{t-1}, q_t}, \quad (2.17)$$

wobei  $\pi$  und  $\mathbf{A}$  die initiale Zustandswahrscheinlichkeit bzw. die Zustandsübergangswahrscheinlichkeit von  $\lambda$  bezeichnen. Fasst man die Gleichung (2.16) und (2.17) zusammen, erhält man die Wahrscheinlichkeit für die Beobachtung von  $O$  für die einzelne Zustandssequenz in Gleichung (2.18). Zur Vereinfachung der Notation ist hierbei  $a_{0,i} := \pi_i$  und  $q_0 := 0$  gesetzt.

$$P_\lambda(O, Q) = P_\lambda(O|Q)P_\lambda(Q) = \prod_{t=1}^T a_{q_{t-1}, q_t} b_{q_t}(\boldsymbol{o}_t) \quad (2.18)$$

Die Wahrscheinlichkeit  $P_\lambda(O)$  ergibt sich nun durch Summation über alle Zustandsfolgen, die das Modell bei der Beobachtung von  $O$  einnehmen kann:

$$P_\lambda(O) = \sum_{Q \in \{q_1 \dots q_T | q_t \in S\}} P_\lambda(O, Q) = \sum_{Q \in \{q_1 \dots q_T | q_t \in S\}} \prod_{t=1}^T a_{q_{t-1}, q_t} b_{q_t}(\mathbf{o}_t). \quad (2.19)$$

Eine Berechnung von  $P_\lambda(O)$ , wie sie in Gleichung (2.19) angegeben ist, beruht darauf, alle Zustandsfolgen  $Q$  aufzuzählen. Der Aufwand für diese *Brute-Force*-Methode ist mit  $\mathcal{O}(TN_S^T)$  bei  $N_S$  Zuständen und Beobachtungssequenzen der Länge  $T$  exponentiell in  $N_S$  und daher nicht in der Praxis einsetzbar. Berücksichtigt man die Markov-Eigenschaft des HMMs, so lässt sich die Berechnung von  $P_\lambda(O)$  wesentlich effizienter durchführen, da die Aufenthaltswahrscheinlichkeit in einem Zustand  $q_t$  nur vom zeitlich direkten Vorgänger  $q_{t-1}$  abhängig ist. Der Weg, über den dieser Zustand erreicht wurde, ist unerheblich. Die Beobachtung führt zum *Vorwärts-Algorithmus*. Er bedient sich der *Vorwärtsvariablen*

$$\alpha_t(i) := P_\lambda(q_t = s_i | \mathbf{o}_1 \dots \mathbf{o}_t), \quad (2.20)$$

welche die Aufenthaltswahrscheinlichkeit des Modells in einem bestimmten Zustand  $q_t = s_i$  zu einem gegebenen Zeitpunkt  $t$  angibt, wenn bereits der Teil  $\mathbf{o}_1, \dots, \mathbf{o}_t$  der Observablenfolge beobachtet wurde. Die Variablen  $\alpha_t(i)$  lassen sich rekursiv berechnen. Dabei werden die Aufenthaltswahrscheinlichkeiten für die Vorgängerzustände  $\alpha_{t-1}(j), j = 1, \dots, N_S$  und die Wahrscheinlichkeit für die Beobachtung der Observablen  $\mathbf{o}_t$  in Zustand  $q_t$  betrachtet:

$$\alpha_t(i) := \sum_{j=1}^{N_S} \alpha_{t-1}(j) b_i(\mathbf{o}_t). \quad (2.21)$$

Die initiale Wahrscheinlichkeit für  $\alpha_1(i)$  ergibt sich aus der initialen Zustandsverteilung und der Beobachtungswahrscheinlichkeit für die erste Beobachtung  $\mathbf{o}_1$ :

$$\alpha_1(i) := \pi_i b_i(\mathbf{o}_1). \quad (2.22)$$

Die Beobachtungswahrscheinlichkeit  $P_\lambda(O)$  für das Modell erhält man mit Hilfe der Wahrscheinlichkeitsverteilung der Endzustände  $q_T$  nach Betrachtung aller  $T$  Zeitschritte der Beobachtungssequenz, d. h. die Rekursion wird mit der Berechnung von

$$P_\lambda(O) = \sum_{j=1}^{N_S} \alpha_T(j) \quad (2.23)$$

abgeschlossen. Der vollständige Vorwärts-Algorithmus ist in Algorithmus 2.3.1 angegeben. Im Gegensatz zur Brute-Force-Methode wird hierbei  $P_\lambda(O)$  mit einem Aufwand von  $\mathcal{O}(TN_S^2)$  berechnet.

Analog zu den Vorwärtsvariablen lassen sich auch *Rückwärtsvariablen*

$$\beta_t(i) := P_\lambda(q_t = s_i | \mathbf{o}_{t+1} \dots \mathbf{o}_T) \quad (2.24)$$

---

**Algorithmus 2.3.1** Der Vorwärts-Algorithmus.
 

---

```

// Parameter:
//  $O = \mathbf{o}_1 \dots \mathbf{o}_T$ : Observablensequenz
//  $\lambda$ : HMM zur Beobachtung von  $O$ 
// Rückgabe:
//  $P_\lambda(O)$ : Wahrscheinlichkeit für die Beobachtung von  $O$  mit  $\lambda$ 

// Initialisierung
for( $i := 1, \dots, N_S$ )
     $\alpha_1(i) := \pi_i b_i(\mathbf{o}_1)$ ;

// Rekursion
for( $t := 1, \dots, T - 1$ )
    for( $i := 1, \dots, N_S$ )
         $\alpha_{t+1}(i) := \sum_{j=1}^{N_S} \alpha_t(j) b_i(\mathbf{o}_{t+1})$ ;

return  $P_\lambda(O) = \sum_{j=1}^{N_S} \alpha_T(j)$ ;

```

---

berechnen. Diese geben die Wahrscheinlichkeit für den Aufenthalt in Zustand  $s_i$  zum Zeitpunkt  $t$  an, unter der Annahme, dass das Modell die Teilsequenz  $\mathbf{o}_{t+1} \dots \mathbf{o}_T$  als Observablen produzieren wird. Bei der rekursiven Berechnung wird der initiale Wert

$$\beta_T(i) := 1 \quad \text{für } i = 1, \dots, N_S \quad (2.25)$$

gesetzt. Bei den folgenden Rekursionsschritten werden für  $t = T - 1, \dots, 1$  die weiteren Werte

$$\beta_t(i) := \sum_{j=1}^{N_S} a_{i,j} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j) \quad (2.26)$$

ermittelt. Aufgrund der Ähnlichkeit der Berechnung wird die Bestimmung der Vorwärts- und Rückwärtsvariablen im so genannten *Vorwärts-Rückwärts-Algorithmus* zusammengefasst. Verwendung finden die Rückwärtsvariablen u. a. beim Baum-Welch-Algorithmus in Abschnitt 2.3.3.3.

### 2.3.3.2 Klassifikation über das Majoritätsprinzip

Die Wahrscheinlichkeit  $P_\lambda(O)$ , die mit Hilfe des Vorwärts-Algorithmus berechnet wird, kann als Maß interpretiert werden, das angibt, wie gut ein Modell  $\lambda$  zu einer Obser-

vablensequenz  $O$  passt. Sind mehrere Modelle  $\{\lambda_1, \dots, \lambda_{N_\lambda}\}$  gegeben, so kann für eine Sequenz  $O$  das passendste Modell mit der maximalen Wahrscheinlichkeit gesucht werden:

$$\lambda_{\max} = \arg \max_{\lambda \in \{\lambda_1, \dots, \lambda_{N_\lambda}\}} (P_\lambda(O)). \quad (2.27)$$

Spiegeln die Modelle verschiedene Klassen wider, so ist damit auch eine Klassifikation möglich, die mit Hilfe des *Majoritätsprinzips* (engl. *winner-takes-all*) realisiert wird, wie es in Algorithmus 2.3.2 angegeben ist.

---

**Algorithmus 2.3.2** Das Majoritätsprinzip zur Klassifikation.

---

```

// Parameter:
// O: zu klassifizierende Observablensequenz
//  $\lambda_{1, \dots, N_C}$ : Modelle für  $N_C$  zu unterscheidenden Klassen
// Rückgabe:
// c: wahrscheinlichste Klasse für O
 $P_{\max} := 0$ ;
 $c := 1$ ;
for( $l := 1, \dots, N_C$ ) {
     $P := P_{\lambda_l}(O)$ ; // Berechnung mit Vorwärts-Algorithmus
    if( $P_{\max} < P$ ) {
         $P_{\max} := P$ ;
         $c := l$ ;
    }
}

return c;

```

---

HMMs mit nur einem Zustand bilden einen Sonderfall bei der Klassifikation über das Majoritätsprinzip. Bei diesen HMMs hängen die Wahrscheinlichkeiten  $P_\lambda(O)$  für die Beobachtungssequenzen ausschließlich von den Beobachtungswahrscheinlichkeiten ab. Bei kontinuierlichen HMMs sind dies die Wahrscheinlichkeiten der Mischverteilungen. Die entsprechenden HMMs sind damit äquivalent zu so genannten *Mischverteilungsklassifikatoren*.

### 2.3.3.3 Training von Hidden-Markov-Modellen

Die in den vorigen Abschnitten vorgestellten Algorithmen dienen ausschließlich der Auswertung von Beobachtungssequenzen mit Hilfe von HMMs. Dabei ist noch nicht geklärt, wie geeignete Parameter der HMMs bestimmt werden können, so dass die Beobachtungssequenz und das zugrunde liegende erzeugende System modelliert wird. Das Modell soll

hinsichtlich der Produktion von Beobachtungssequenzen aus einer Klasse von Trainingsdaten spezialisiert werden. Im Folgenden wird dazu das Standardverfahren zum Training von HMMs dargestellt. Es handelt sich um den Baum-Welch-Algorithmus, der iterativ die Parameter des Modells so anpasst, dass die Beobachtungswahrscheinlichkeit  $P_\lambda(O)$  für die Trainingsdaten lokal maximiert wird. Für ein trainiertes Modell  $\hat{\lambda}$  gilt gegenüber einem untrainierten Modell  $\lambda$  nach der Parameteroptimierung  $P_{\hat{\lambda}}(O) \geq P_\lambda(O)$ . Das Verfahren bedient sich dabei der aus der Statistik bekannten *Expectation-Maximization (EM)* bzw. der äquivalenten Methode der *Maximum-Likelihood (ML)*. Beide Darstellungen lassen sich auf ein gradientenbasiertes Optimierungsverfahren zurückführen [128]. Eine analytische Methode zur Bestimmung der optimalen Parametrisierung ist nicht bekannt [43].

Zur Erklärung der iterativen Parameterberechnung wird zunächst eine weitere Wahrscheinlichkeitsdichte  $\xi_t(i, j)$  definiert. Sie gibt die Wahrscheinlichkeit an, dass sich das HMM bei gegebener Observablensequenz zum Zeitpunkt  $t$  in Zustand  $s_i$  und zum darauffolgenden Zeitpunkt  $t + 1$  in Zustand  $s_j$  befindet, d. h.

$$\xi_t(i, j) := P_\lambda(q_t = s_i, q_{t+1} = s_j | O). \quad (2.28)$$

Diese Wahrscheinlichkeit lässt sich unmittelbar mit den Vorwärts- und Rückwärtsvariablen aus Abschnitt 2.3.3.1 berechnen:

$$\xi_t(i, j) = \frac{\alpha_t(i) a_{i,j} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j)}{P_\lambda(O)} = \frac{\alpha_t(i) a_{i,j} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N_S} \sum_{j=1}^{N_S} \alpha_t(i) a_{i,j} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j)}. \quad (2.29)$$

Des Weiteren ergibt sich aus  $\xi_t(i, j)$  die Wahrscheinlichkeit  $\gamma_t(i)$ , dass sich das Modell zum Zeitpunkt  $t$  bei gegebener Beobachtungssequenz in Zustand  $s_i$  befindet. Sie berechnet sich aus

$$\gamma_t(i) := \sum_{j=1}^{N_S} \xi_t(i, j). \quad (2.30)$$

Aus der Summation über die Zeitpunkte  $t = 1, \dots, T - 1$  erhält man als Erwartungswerte  $\sum_{t=1}^{T-1} \xi_t(i, j)$  für die Anzahl von Übergängen von Zustand  $s_i$  zu  $s_j$  und  $\sum_{t=1}^{T-1} \gamma_t(i)$  für die Anzahl von Übergängen aus Zustand  $s_i$ . Aus dem Verhältnis der erwarteten Anzahl von Übergängen von  $s_i$  nach  $s_j$  und der erwarteten Anzahl von Übergängen aus Zustand  $s_i$  lässt sich für ein diskretes HMM eine Neuabschätzung  $\tilde{a}_{i,j}$  der Zustandsübergangswahrscheinlichkeit erzielen:

$$\tilde{a}_{i,j} := \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}. \quad (2.31)$$

Ebenso lässt sich die Beobachtungswahrscheinlichkeit für ein Symbol  $v_k \in V$  mit

$$\tilde{b}_{j,k} := \frac{\sum_{t=1}^T \gamma_t(i, j)}{\sum_{t=1}^T \gamma_t(i)} \quad (2.32)$$

aus dem Verhältnis der erwarteten Anzahl von Zeitpunkten in Zustand  $s_j$  und gleichzeitiger Beobachtung von  $v_k$  zur gesamten Anzahl von Zeitpunkten in Zustand  $s_j$  neu abschätzen. Eine Neuabschätzung für die initiale Zustandsverteilung ergibt sich aus der Häufigkeit, mit der sich das Modell in Zustand  $s_i$  zum Zeitpunkt  $t = 1$  befindet:

$$\tilde{\pi}_i := \gamma_1(i). \quad (2.33)$$

Die beschriebenen Neuabschätzungen können mehrmals wiederholt werden. Bei jedem Iterationsschritt wird dabei durch die nachweisbare Eigenschaft  $P_{\tilde{\lambda}}(O) \geq P_{\lambda}(O)$  eine Verbesserung des Modells erzielt oder ein Fixpunkt erreicht, bei dem das neue Modell als optimale Lösung für das Training genommen werden kann (siehe [129]). In der Praxis wird häufig zusätzlich ein Abbruchkriterium in Form einer maximalen Anzahl von Iterationen  $N_{\max\text{Iter}}$  angegeben, bei dem aus Erfahrung weitere Iterationsschritte keine wesentliche Verbesserung der Modellparameter mehr bewirken. Der vollständige Baum-Welch-Algorithmus für diskrete HMMs ist in Algorithmus 2.3.3 angegeben.

Im Falle von kontinuierlichen HMMs lassen sich die Modellparameter nach dem gleichen Schema trainieren. Die Neuabschätzung der initialen Zustandsverteilung  $\boldsymbol{\pi}$  und der Übergangswahrscheinlichkeiten  $\mathbf{A}$  werden übernommen und die Parameter für die Mischverteilungen zur Bestimmung der Beobachtungswahrscheinlichkeiten werden wie folgt durchgeführt:

$$\tilde{c}_{j,m} := \frac{\sum_{t=1}^T \gamma_{m,t}(j)}{\sum_{t=1}^T \sum_{k=1}^{N_M} \gamma_{k,t}(j)}, \quad (2.34)$$

$$\tilde{\boldsymbol{\mu}}_{j,m} := \frac{\sum_{t=1}^T \gamma_{m,t}(j) \mathbf{o}_t}{\sum_{t=1}^T \gamma_{m,t}(j)}, \quad (2.35)$$

$$\tilde{\mathbf{U}}_{j,m} := \frac{\sum_{t=1}^T \gamma_{m,t}(j) (\mathbf{o}_t - \boldsymbol{\mu}_{j,m})(\mathbf{o}_t - \boldsymbol{\mu}_{j,m})^\top}{\sum_{t=1}^T \gamma_{m,t}(j)}, \quad (2.36)$$

wobei  $\gamma_{m,t}$  die Unterscheidung von  $\gamma_t$  bezüglich der verwendeten Mischverteilungskomponenten darstellt, d. h. die Wahrscheinlichkeit, dass sich das Modell zum Zeitpunkt  $t$  über Komponente  $m$  der Mischverteilung in Zustand  $s_j$  befindet. Sie berechnet sich zu

$$\gamma_{m,t}(j) = \left( \frac{\alpha_t(j) \beta_t(j)}{\sum_{i=1}^{N_S} \alpha_t(i) \beta_t(i)} \right) \left( \frac{w_{j,m} \mathcal{N}(\mathbf{o}_t, \boldsymbol{\mu}_{j,m}, \mathbf{U}_{j,m})}{\sum_{k=1}^{N_M} w_{j,k} \mathcal{N}(\mathbf{o}_t, \boldsymbol{\mu}_{j,k}, \mathbf{U}_{j,k})} \right). \quad (2.37)$$

---

**Algorithmus 2.3.3** Der Baum-Welch-Algorithmus.
 

---

```

// Parameter:
//  $O = o_1 \dots o_T$ : Observablensequenz
//  $\tilde{\lambda} = (\tilde{\pi}, \tilde{\mathbf{A}}, \tilde{\mathbf{B}})$ : Startmodell mit geeigneten initialen Parametern
//  $N_{\max\text{Iter}}$ : maximale Anzahl von Iterationen
// Rückgabe:
//  $\lambda = (\pi, \mathbf{A}, \mathbf{B})$ : bezüglich  $O$  spezialisiertes HMM

n := 0;
repeat{
   $\lambda := \tilde{\lambda}$ ;
  for( $t := 1, \dots, T - 1$ ) {
     $\xi_t(i, j) := \frac{\alpha_t(i)a_{i,j}b_j(o_{t+1})\beta_{t+1}(j)}{P_{\lambda}(O)} = \frac{\alpha_t(i)a_{i,j}b_j(o_{t+1})\beta_{t+1}(j)}{\sum_{i=1}^{N_S} \sum_{j=1}^{N_S} \alpha_t(i)a_{i,j}b_j(o_{t+1})\beta_{t+1}(j)}$ ;
     $\gamma_t(i) := \sum_{j=1}^{N_S} \xi_t(i, j)$ ;
  }

   $\tilde{\pi}_i := \gamma_1(i)$ ;
   $\tilde{a}_{i,j} := \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$ ;
   $\tilde{b}_{j,k} := \frac{\sum_{t=1}^T \gamma_t(i, j)}{\sum_{t=1}^T \gamma_t(i)}$ ;

  n := n + 1;
} until ( $n = N_{\max\text{Iter}} \vee (P_{\tilde{\lambda}}(O) - P_{\lambda}(O)) < \varepsilon$ );

return  $\lambda := \tilde{\lambda}$ ;

```

---

Erneut kann gezeigt werden, dass das Iterationsverfahren konvergiert und Parametrisierung mit maximaler Beobachtungswahrscheinlichkeit bezüglich der Trainingsdaten erreicht wird (siehe [129]).

Ein wesentlicher Aspekt beim Training von HMMs mit dem Baum-Welch-Verfahren ist, eine geeignete Parametrisierung als Startwert für das Iterationsverfahren zu wählen. Insbesondere bleiben Null-Einträge in der Zustandsübergangsmatrix erhalten, so dass die grundlegende Struktur des Modells erhalten bleibt, wie sie initial angegeben wird, und durch das Training nicht modifiziert werden kann. Da aber die Struktur des Modells stark von der jeweiligen Anwendung abhängig ist, kann an dieser Stelle keine pauschale Lösung gegeben werden. In dieser Arbeit wird stattdessen im Rahmen der jeweiligen Anwendungsbeispiele gezielt die Wahl der initialen Parametrisierungen erläutert.

Bei der beschriebenen Trainingsmethode wird davon ausgegangen, dass die Parameter des HMMs lediglich anhand einer Observablensequenz eintrainiert wird. In der Praxis liegen jedoch viele Observablensequenzen vor, um eine Überanpassung der Modelle zu verhindern. Sei  $\mathbf{O} = \{O_1, O_2, \dots, O_{N_D}\}$  die Menge der Beobachtungssequenzen mit voneinander unabhängigen Sequenzen  $O_d = \mathbf{o}_{d,1}, \dots, \mathbf{o}_{d,T_d}$ ,  $d = 1, \dots, N_D$ , dann ist das Ziel des Trainings die Beobachtungswahrscheinlichkeit  $P_\lambda(\mathbf{O}) = \prod_{d=1}^{N_D} P_\lambda(O_d)$  bezüglich aller Trainingssequenzen zu maximieren. Da die Neuabschätzungen in den Gleichung (2.31) bis (2.36) auf Auftrittswahrscheinlichkeiten ausgezeichneter Ereignisse basieren, lassen sich die Neuabschätzungsformeln für mehrere Observablensequenzen durch Addition der Häufigkeiten für die einzelnen Beobachtungssequenzen erweitern. Die Erweiterungen für die Parameter diskreter HMMs lauten damit:

$$\tilde{\pi}_i := \frac{1}{N_D} \sum_{d=1}^{N_D} \frac{1}{P_\lambda(O_d)} \left( \sum_{t=1}^{T_d-1} \alpha_{d,t}(i) \beta_{d,t}(i) \right), \quad (2.38)$$

$$\tilde{a}_{i,j} := \frac{\sum_{d=1}^{N_D} \frac{1}{P_\lambda(O_d)} \left( \sum_{t=1}^{T_d-1} \alpha_{d,t}(i) a_{i,j} b_j(\mathbf{o}_{d,t+1}) \beta_{d,t+1}(j) \right)}{\sum_{d=1}^{N_D} \frac{1}{P_\lambda(O_d)} \left( \sum_{t=1}^{T_d-1} \alpha_{d,t}(i) \beta_{d,t}(i) \right)} \quad (2.39)$$

und

$$\tilde{b}_j(k) := \frac{\sum_{d=1}^{N_D} \frac{1}{P_\lambda(O_d)} \left( \sum_{\substack{t=1 \\ \wedge \mathbf{o}_{d,t}=v_k}}^{T_d} \alpha_{d,t}(j) \beta_{d,t}(j) \right)}{\sum_{d=1}^{N_D} \frac{1}{P_\lambda(O_d)} \left( \sum_{t=1}^{T_d} \alpha_{d,t}(j) \beta_{d,t}(j) \right)}. \quad (2.40)$$

Analog wird mit den Parametern für die Mischverteilungen in kontinuierlichen HMMs verfahren:

$$\tilde{w}_{j,m} := \frac{\sum_{d=1}^{N_D} \sum_{t=1}^{T_d} \gamma_{d,m,t}(j)}{\sum_{d=1}^{N_D} \sum_{t=1}^{T_d} \sum_{k=1}^{N_M} \gamma_{d,k,t}(j)}, \quad (2.41)$$

$$\tilde{\boldsymbol{\mu}}_{j,m} := \frac{\sum_{d=1}^{N_D} \sum_{t=1}^{T_d} \gamma_{d,m,t}(j) \mathbf{o}_{d,t}}{\sum_{d=1}^{N_D} \sum_{t=1}^{T_d} \gamma_{d,m,t}(j)} \quad (2.42)$$

und

$$\tilde{\mathbf{U}}_{j,m} := \frac{\sum_{d=1}^{N_D} \sum_{t=1}^{T_d} \gamma_{d,m,t}(j) (\mathbf{o}_{d,t} - \boldsymbol{\mu}_{j,m})(\mathbf{o}_{d,t} - \boldsymbol{\mu}_{j,m})^\top}{\sum_{d=1}^{N_D} \sum_{t=1}^{T_d} \gamma_{d,m,t}(j)}. \quad (2.43)$$

## 2.3.4 Numerische Probleme beim Einsatz von HMMs

### 2.3.4.1 Skalierung

Bei der Berechnung der Vorwärts- und Rückwärtsvariablen  $\alpha_t(i)$  und  $\beta_t(i)$  entstehen Produktterme mit sehr vielen Faktoren aus dem Intervall  $[0, 1]$ , so dass die Werte der Variablen mit zunehmender Zeit  $t$  exponentiell gegen 0 gehen und als Gleitkommazahl in der Maschindarstellung nicht mehr angegeben werden können. Um dies zu verhindern, wird, wie in [129], beschrieben eine Skalierung der Werte der Vorwärtsvariablen  $\check{\alpha}_t(i) := c_t \alpha'_t(i)$  unabhängig vom Zustand für jeden Zeitpunkt vorgenommen. Die Werte  $\alpha'_t(i)$  werden dabei wie in Gleichung (2.21) rekursiv über

$$\alpha'_t(i) = \sum_{j=1}^{N_S} \check{\alpha}_{t-1}(j) b_i(\mathbf{o}_t) \quad (2.44)$$

mit bereits skalierten Vorgängern  $\check{\alpha}_{t-1}$  berechnet. Der Skalierungsfaktor in jedem Zeitschritt berechnet sich durch

$$c_t = \left( \sum_{i=1}^{N_S} \alpha_t(i) \right)^{-1}. \quad (2.45)$$

Durch Induktion kann gezeigt werden, dass eine vereinfachte Berechnung

$$\check{\alpha}_t(i) = \left( \prod_{\tau=1}^t c_\tau \right) \alpha_t(i) \quad (2.46)$$

direkt mit Hilfe der Folge von Skalierungsfaktoren berechnet werden kann. Analog zu den Vorwärtsvariablen lassen sich auch die Rückwärtsvariablen mit denselben Faktoren  $c_t$  skalieren. Für diese gilt

$$\check{\beta}_{t+1}(i) = \left( \prod_{\tau=t+1}^T c_\tau \right) \beta_t(i). \quad (2.47)$$

Die Skalierung der Vorwärtsvariablen bewirkt eine Normierung

$$\sum_{i=1}^{N_S} \check{\alpha}_t(i) = c_t \sum_{i=1}^{N_S} \alpha'_t(i) = \frac{1}{\sum_{i=1}^{N_S} \alpha'_t(i)} \sum_{i=1}^{N_S} \alpha'_t(i) = 1, \quad (2.48)$$

aus der sich auch die Beobachtungswahrscheinlichkeit des Modells über

$$1 = \sum_{i=1}^{N_S} \alpha_t(i) = \left( \prod_{\tau=1}^T c_t \right) \sum_{i=1}^{N_S} a_T(i) = \left( \prod_{\tau=1}^T c_t \right) P_\lambda(O) =: \tilde{c}_T P_\lambda(O) \quad (2.49)$$

berechnen lässt. Vereinfacht kann nun  $P_\lambda(O) = 1/\tilde{c}_T$  geschrieben werden. Dieser Ausdruck lässt sich in eine logarithmische Darstellung übertragen, in der eine Darstellung von Wahrscheinlichkeiten nahe 0 immer noch möglich ist und somit resultierende numerische Probleme nicht mehr auftreten:

$$\log P_\lambda(O) = \log \tilde{c}_T^{-1} = \log \left( \prod_{\tau=1}^T c_t \right)^{-1} = - \sum_{\tau=1}^T \log c_t. \quad (2.50)$$

### 2.3.4.2 Erhaltung des Zeilensummenkriteriums

Ein Problem beim praktischen Einsatz des Baum-Welch-Algorithmus zum Training der HMM-Parameter stellt die Erhaltung des Zeilensummenkriteriums in Gleichung (2.7) dar. Bei der Neuabschätzung der Zustandsübergangswahrscheinlichkeiten in Matrix  $\mathbf{A}$  und bei Beobachtungswahrscheinlichkeiten diskreter HMMs in Matrix  $\mathbf{B}$  können in numerischen Ausnahmefällen auch Null-Einträge produziert werden, die dazu führen, dass alle folgenden Vorwärtswahrscheinlichkeiten sowie die Beobachtungswahrscheinlichkeit des Modells zum Wert 0 kollabieren, wenn die übrigen Einträge in der Matrixzeile ebenfalls Null sind.

Die Neuabschätzung der Wahrscheinlichkeiten mit dem Baum-Welch-Algorithmus wurde in dieser Arbeit zur Vermeidung des Problems um die Fallunterscheidung

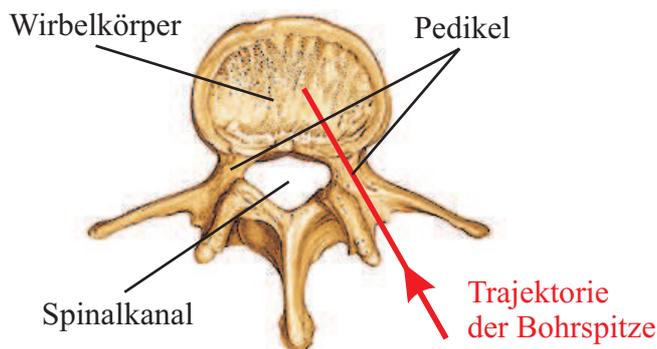
$$\check{a}_{i,j} := \begin{cases} \check{a}_{i,j} & \text{wenn } \check{a}_{i,j} > \varepsilon \wedge a_{i,j} > 0, \\ \varepsilon & \text{sonst} \end{cases} \quad (2.51)$$

ergänzt. Betrachtet werden dabei Matrixeinträge  $\check{a}_{i,j}$ , die bereits wie im vorigen Abschnitt skaliert als Zwischenergebnis im Baum-Welch-Algorithmus vorliegen. Die Fallunterscheidung verhindert, dass die Matrixeinträge unter einen vorgegebenen Wert  $\varepsilon > 0$  absinken. Ausnahme bilden die Einträge, die in der unskalierten Matrix  $\mathbf{A}$  bereits 0 sind und somit nicht verändert werden dürfen, da sonst die Struktur des HMMs verändert wird. Abschließend bewirkt eine erneute Skalierung nach der Fallunterscheidung, dass das Zeilensummenkriterium schließlich erhalten bleibt. Die Erhaltung des Kriteriums kann analog durch Korrektur der Einträge in der Matrix  $\mathbf{B}$  durchgeführt werden.

## 2.4 Analyse von Bohrgeräuschen

### 2.4.1 Problemstellung und Forschungsstand

Von großer Bedeutung für die Neurochirurgie ist die hohe Genauigkeit, mit der Operationen durchgeführt werden müssen, um bei einem Eingriff keine schwerwiegenden Folgeschäden beim Patienten zu verursachen. Dies betrifft nicht nur die präoperative Planung,

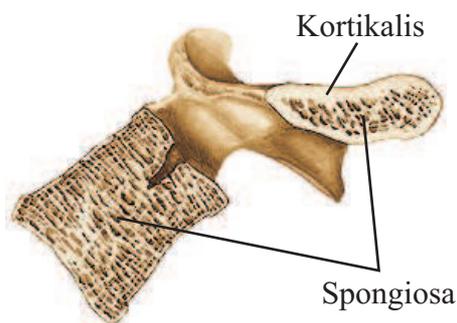


**Abbildung 2.3:** Weg des Bohrers durch den Pedikel in den Wirbelkörper.

sondern auch den intraoperativen Einsatz von Geräten. Besonders hohe Anforderungen an die Genauigkeit der durchgeführten Operation ergeben sich bei Bohrvorgängen an der Wirbelsäule. Diese sind erforderlich, wenn ein Bereich der Wirbelsäule, z. B. nach einem Bandscheibenvorfall, einer Wirbelfraktur oder einer anderen degenerativen Veränderungen des Wirbelkörpers mit Hilfe von Platten und Gittern aus Titan versteift werden sollen [59]. Mehrere Wirbel werden in diesen Fällen über die Metallstrukturen fest miteinander verbunden. Die Fixierung erfolgt mit Hilfe von Schrauben, die in den *Pedikel* eingedreht werden. Für diese *Pedikelschrauben* müssen zunächst *Kernlöcher* in den Knochen mit einer Präzision von 1 mm vorgefräst werden [79], [147]. Dabei dürfen wichtige Blutgefäße und das Rückenmark im Spinalkanal nicht verletzt werden. Abbildung 2.3 veranschaulicht den Weg des Bohrers durch den Pedikel in den Wirbelkörper.

Vor der Operation werden tomographische Volumendatensätze des Patienten aufgenommen, mit denen sich der Chirurg einen räumlichen Eindruck von dem betroffenen Wirbelsäulensegment machen kann. Häufig werden die Volumendaten auch in ein Operationsplanungssystem geladen, um mit Hilfe des Rechners einen geeigneten Zugang und eine optimale Trajektorie zum Führen des Bohrers zu bestimmen. Während der Operation werden die zu verschraubenden Wirbel durch Ablösen, Verschieben und Entfernen von Fett- und Muskelgewebe freigelegt und die Kernlöcher für die Pedikelschrauben manuell gebohrt. Dabei stützt sich der Chirurg auf sein anatomisches Wissen und die Informationen aus den präoperativ akquirierten Volumendatensätzen. Heute erfährt der Chirurg zunehmend Unterstützung durch intraoperative Röntgenbilder mit dem C-Bogen [131], [49] oder dem mobilen CT [179] und durch Navigationssysteme, die jedoch häufig Schwierigkeiten bei der Registrierung von prä- und intraoperativen Daten haben [76], [119], [168], [167], [171], [63]. Letztendlich ist der Chirurg in hohem Maß auf sein räumliches Vorstellungsvermögen und sein sensomotorisches Feingefühl angewiesen. Zusätzlich erschwerend ist, dass die Sicht des Chirurgen während des Eingriffes durch chirurgische Instrumente, Blut und den engen Operationszugang beeinträchtigt wird und dass intraoperative Wirbelverschiebungen bis zu 1 cm auftreten können [48].

Eine nicht geringe Zahl von Fällen mit fehlplatzierten Pedikelschrauben und daraus resultierenden Komplikationen für den Patienten belegen die Schwierigkeiten beim Setzen der



**Abbildung 2.4:** In einem Schnitt durch den Wirbel sind die Bereiche mit unterschiedlicher Dichte des Knochenmaterials zu erkennen. (Bildmaterial aus [126])

Pedikelschrauben [152], [144]. Eine Verbesserung der Erfolgsquoten bei den Operationen durch den Einsatz von Chirurgierobotern (siehe [97], [121], [133], [122]) ist nicht absehbar. Die Robotersysteme zum Setzen von Pedikelschrauben sind noch in der Entwicklung und der Einsatz von Robotern für andere chirurgische Eingriffe ist sehr umstritten [45]. Alternative Verfahren zur Unterstützung des Chirurgen bei der Operation sind demnach erforderlich.

Neben den optischen Informationen erhält der Chirurg während der Eingriffe an der Wirbelsäule wertvolle haptische und akustische Informationen über den Bohrer. Die Interpretation dieser Informationen erfordert lange praktische Erfahrung und wird bisher noch nicht in rechnergestützten Chirurgesystemen eingesetzt. Insbesondere geben die Bohrgeräusche Auskunft über die Beschaffenheit des Gewebes, in dem der Bohrer gerade Material abträgt. Daraus leitet der Chirurg die Position der Bohrspitze im Knochen ab. Die feste Knochenwand (*Kortikalis*) besitzt eine wesentlich höhere Dichte als das weiche Knocheninnere (*Spongiosa*). Übergänge zwischen Bereichen unterschiedlicher Knochendichte stehen in hohem Zusammenhang mit Veränderungen der Bohrgeräusche. Abbildung 2.4 zeigt die Bereiche verschiedener Dichte im Wirbelknochen.

Die Veränderungen der Bohrgeräusche sowie die daraus abgeleiteten Informationen über Material und Position der Bohrspitze im Wirbel sollten sich auch durch den Rechner auswerten lassen, um den Chirurgen z. B. durch Warnung vor Erreichen eines Risikobereichs zu unterstützen. Dieses Szenario ist ein ausgezeichnetes Beispiel für ein Anthropomatiksystem. Der Mensch, in diesem Falle der Chirurg, interagiert mit seiner Umwelt und nimmt dabei Informationen wahr, die durch den Prozess des Bohrvorgangs erzeugt werden. Die Informationen über das Fortschreiten des Bohrvorganges und das gerade abgetragene Material sind jedoch nicht direkt mit technischen Mitteln messbar. In dieser Arbeit sollen dennoch die Informationen mit Hilfe rechnerbasierter Klassifikation bestimmt werden. Das Hauptaugenmerk gilt dabei der Anwendung von HMMs. Eine Darstellung der Methoden und Ergebnisse ist auch in [20] zu finden.



**Abbildung 2.5:** Links: Raummikrofon CCM 2 der Fa. Schoeps. Rechts: Oberflächenmikrofon C 562 BL der Fa. AKG.

## 2.4.2 Datenakquisition

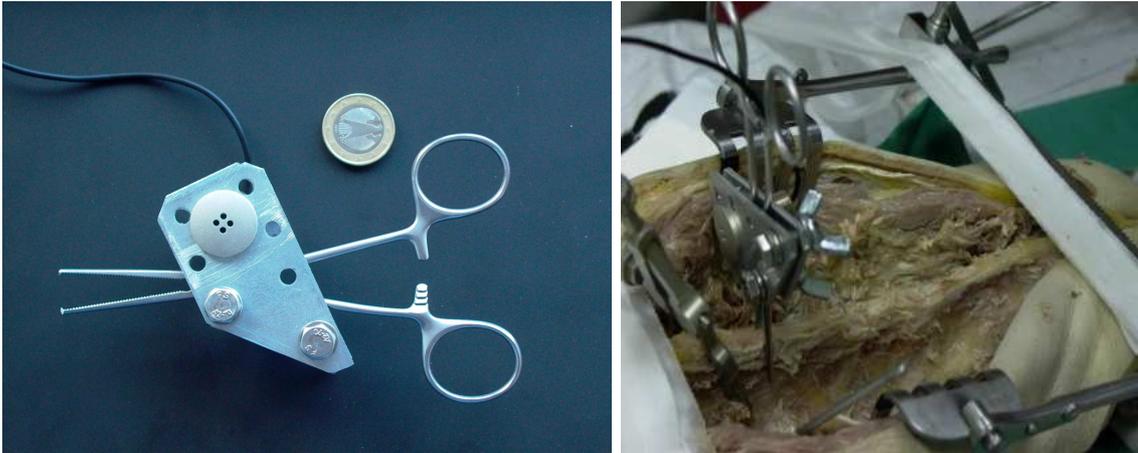
Um Datenmaterial für die Entwicklung und Bewertung der Verfahren in dieser Arbeit zu gewinnen, wurden Geräuschaufnahmen während mehrerer Bohrungen mit einem *chirurgischen Handbohrer* (TPS U<sub>2</sub> Drill der Fa. Stryker) an der Wirbelsäule eines menschlichen Kadaverpräparates in der Neurochirurgie Karlsruhe gemacht. Dabei kamen zwei verschiedene Typen von Mikrofonen zum Einsatz. Es wurde sowohl ein *Raummikrofon* (CCM 2 der Fa. Schoeps) frei hängend über dem Operationstisch an der Operationslampe angebracht als auch ein *Oberflächenmikrofon* (C 562 BL der Fa. AKG) direkt am Wirbel fixiert. Beide Mikrofone sind in Abb. 2.5 dargestellt.

Das Raummikrofon erfasst die Geräusche, die auch der Chirurg während der Operation wahrnimmt. Das Oberflächenmikrofon liefert hingegen zusätzliche Information über die Geräusche, die direkt im Knochen durch den Bohrer verursacht werden. Die Fixierung des Oberflächenmikrofons am Wirbel erfolgt mit Hilfe einer *Operationsklemme*, an der eine Halterung mit einer Aufnahme für das Oberflächenmikrofon fest verschraubt wird. Die im Knochen auftretenden Schallwellen können so direkt auf das Oberflächenmikrofon übertragen werden. Abbildung 2.6 zeigt die Operationsklemme mit Mikrofon und die Fixierung am Wirbel während des chirurgischen Eingriffes. Beim Anbringen beider Mikrofone wird darauf geachtet, dass der Chirurg möglichst wenig durch die zusätzlichen Geräte behindert wird.

Die Übertragung der aufgenommenen Bohrgeräusche auf den Rechner erfolgt mit einer Audiokarte (Duo USB der Fa. M-Audio), die an einen handelsüblichen PC angeschlossen wird. Die Audiokarte gestattet die synchrone Aufnahme der Bohrgeräusche von beiden Mikrofonen mit einer Sampling-Rate von 96 kHz und einer Auflösung von 24 bit pro Sample. Insgesamt wurden mit dieser Konfiguration Datensätze von 14 Bohrungen an 8 Wirbeln eines menschlichen Kadaverpräparates aufgenommen.

## 2.4.3 Generierung von Audiomeerkmalen

In der Audiodatenverarbeitung haben sich einige häufig verwendete Merkmale als Standard etabliert [34], die daher auch zur Beschreibung der Bohrgeräusche in dieser Arbeit



**Abbildung 2.6:** Links: Halterung für das Oberflächenmikrofon an einer Operationsklemme. Rechts: Anbringung des Oberflächenmikrofon mittels Operationsklemme und Halterung direkt am Wirbel.

verwendet werden. Für ihre Bestimmung wird zunächst eine Frequenzanalyse auf den Audiorohdaten mit Hilfe einer gefensterten Fourier-Transformation (*short-time fourier-transform*, *STFT*) durchgeführt. Diese ist definiert als

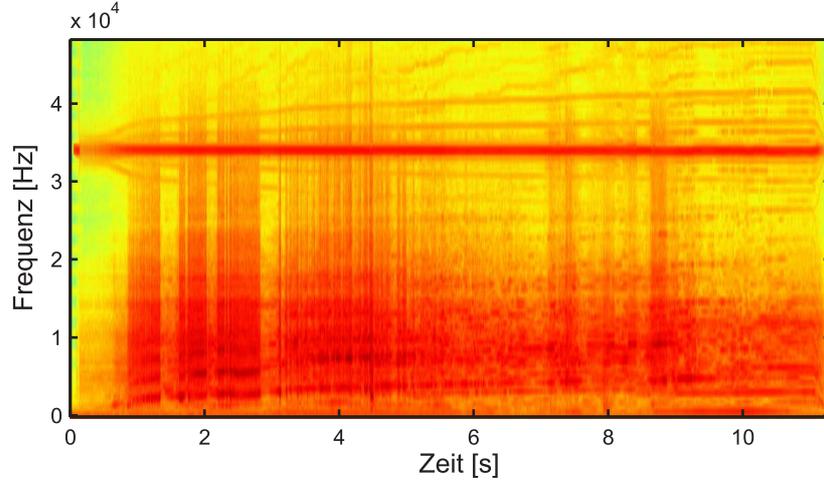
$$\hat{f}_w(t, \nu) := [\mathcal{F}_w f](\tau) = \int f(\tau)w(\tau - t)e^{-2\pi i\nu\tau} d\tau \quad (2.52)$$

und liefert für das Signal  $f$  und einem festen Fenster  $w$  den Signalanteil mit Frequenz  $\nu$  zum Zeitpunkt  $t$  (siehe auch [10]). Das daraus resultierende *Spektrogramm*  $|\hat{f}_w(t, \nu)|^2$  gibt zu den jeweiligen Zeitpunkten  $t$  die Energieverteilung in Abhängigkeit der Frequenzen  $\nu$  an.

Anhand des Spektrogramms lassen sich bereits deutliche Veränderungen der Geräusche im zeitlichen Verlauf eines Bohrvorganges erkennen. Abbildung 2.7 zeigt dazu das Spektrogramm zu einem exemplarischen Bohrvorgang. Berechnet wurde das Spektrogramm mit einem *Blackman-Fenster*  $w_B$  für die Fensterfunktion  $w$ . Das Blackman-Fenster dient in der digitalen Signalverarbeitung häufig einer linearen Filterung als diskreter Faltungskern endlicher Länge (*FIR-Filter*, engl.: *finite impulse response filter*). Es eignet sich besonders gut zur Dämpfung hoher Frequenzen und wird deshalb sowohl zur Fensterung als auch zur Tiefpassfilterung eingesetzt [77]. Die Koeffizienten des Blackman-Fensters mit Länge  $T_{w_B}$  sind wie folgt definiert:

$$w_B(t+1) = \begin{cases} 0,42 - 0,5 \cos\left(\frac{2\pi t}{T_{w_B}-1}\right) + 0,08 \cos\left(\frac{4\pi t}{T_{w_B}-1}\right) & \text{für } t = 0, \dots, T_{w_B}-1, \\ 0 & \text{sonst.} \end{cases} \quad (2.53)$$

Zur Berechnung des Spektrogramms in Abb. 2.7 wurde ein Blackman-Fenster der Länge  $T_{w_B} = 1024$  Samples gewählt. Dies entspricht bei der Sample-Rate der Rohdaten einer Dauer von 10,6 ms. Die Dauer des gesamten Bohrvorganges beträgt im Vergleich 11,3 s.



**Abbildung 2.7:** Spektrogramm einer ausgewählten Bohrung. Die Energien sind durch einen Farbverlauf von grün (niedrig) nach rot (hoch) dargestellt.

Mit Hilfe der Frequenzdaten werden zu jedem Fenster 16 Standard-Merkmale generiert. Insgesamt ergeben sich somit für beide Mikrofone zusammengenommen Merkmalsvektoren  $\mathbf{o} \in \mathbb{R}^{32}$ . Bei den einzelnen Merkmalen handelt es sich zunächst um das *Volumen*

$$\text{Vol}(t) = \frac{\int |f(\tau)w(\tau - t)|^2 d\tau}{\int w(\tau) d\tau} = \frac{\int |\hat{f}_w(t, \nu)|^2 d\nu}{\int w(\tau) d\tau}, \quad (2.54)$$

die *Bandenergie* in 10 Teilbändern des Spektrums

$$\text{BE}_{\nu_i, \nu_{i+1}}(t) = \frac{\int_{\nu_i}^{\nu_{i+1}} |\hat{f}_w(t, \nu)|^2 d\nu}{\int w(\tau) d\tau} \quad \text{mit } i = 0, \dots, 9, \quad (2.55)$$

das *Bandenergieverhältnis*

$$\text{BER}_{\nu_i, \nu_{i+1}}(t) = \frac{\text{BE}_{\nu_i, \nu_{i+1}}(t)}{\text{Vol}(t)} \quad \text{mit } i = 0, \dots, 9, \quad (2.56)$$

die *mittlere Frequenz*

$$\bar{\nu}(t) = \frac{\int \nu |\hat{f}_w(t, \nu)|^2 d\nu}{\int |\hat{f}_w(t, \nu)|^2 d\nu} \quad (2.57)$$

und die *Bandbreite*

$$\text{BW}(t) = \frac{\int (\nu - \bar{\nu}(t))^2 |\hat{f}_w(t, \nu)|^2 d\nu}{\int |\hat{f}_w(t, \nu)|^2 d\nu}. \quad (2.58)$$

Bei den restlichen Merkmalen handelt es sich um die ersten drei Koeffizienten aus dem so genannten *Cepstrogramm*, das die inverse Fourier-Transformierte der logarithmierten Spektrogramm-Funktion darstellt:

$$\hat{f}_w(t, \tau) = \frac{1}{2} \int \log \left( |\hat{f}_w(t, \nu)|^2 \right) e^{2\pi i \nu \tau} d\nu. \quad (2.59)$$

Die Cepstral-Koeffizienten werden wie die anderen Merkmale häufig als Standardmerkmale in der Audiodatenverarbeitung genutzt. Sie zeichnen sich insbesondere dadurch aus, dass sie die nichtlinearen Eigenschaften des menschlichen Gehörs widerspiegeln. Eine ausführliche Beschreibung der Cepstral-Koeffizienten ist in [130] zu finden. Die Berechnung aller hier beschriebenen und verwendeten Merkmale erfolgt in dieser Arbeit mit Hilfe der Matlab-Programmbibliothek VOICEBOX [25].

## 2.4.4 Automatische Analyse von Bohrgeräuschen

### 2.4.4.1 Herkömmliche Verfahren

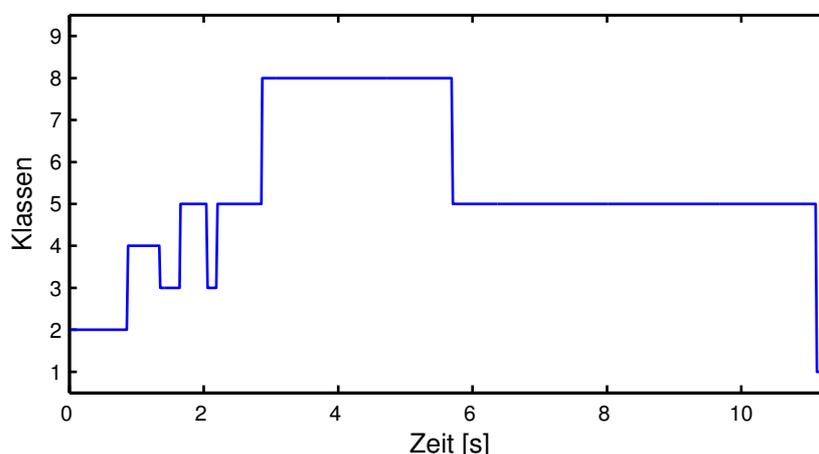
Für die automatische Analyse der Bohrgeräusche werden zunächst Neuronale Netze (NNs) und Support-Vector-Machines (SVMs) zur Analyse der Bohrgeräusche herangezogen. Beide Verfahren haben sich in vielen Bereichen für Klassifikationsaufgaben, z. B. der Spracherkennung, der Musteranalyse oder der Objektidentifikation bereits etabliert, so dass für die Untersuchungen in dieser Arbeit auf bestehende ausgereifte Programmbibliotheken zurückgegriffen werden konnte. Die wesentlichen Grundlagen, Datenstrukturen und Algorithmen zu den Verfahren sind in den Anhängen B und C zusammengefasst.

Zur Erzeugung von Trainingsdaten für NNs und SVMs wurden die Merkmalsvektoren aus allen akquirierten Datensätzen wie im vorigen Abschnitt beschrieben anhand eines Blackman-Filters mit Fensterlänge  $T_{w_B} = 1024$  Samples berechnet. Um nur einen Teil der akquirierten Daten für Trainingszwecke zu verwenden, wurde das Filter nicht bei jedem Sample, sondern mit einer Fensterverschiebung von  $T_{\text{Shift}} = 2048$  Samples angewendet. Zusätzlich wurden die einzelnen Komponenten der Merkmalsvektoren durch lineare Abbildungen so normalisiert, dass das Maximum in den Trainingsdaten auf den Wert 1 und das Minimum auf den Wert 0 abgebildet werden. Für die einzelnen Merkmalsvektoren gilt nach dieser Normalisierung  $\mathbf{o} \in [0, 1]^{32}$ . Zur Erzeugung von Testdaten wurden Sequenzen von Merkmalsvektoren mit gleicher Fensterverschiebung und Normalisierung generiert, jedoch wurde bei der Fensterung mit dem Sample bei  $T_{\text{Offset}} = 1025$  begonnen. Aufgrund der verwendeten Fensterlänge  $T_{w_B}$ , der Fensterverschiebung  $T_{\text{Shift}}$  und des Offsets  $T_{\text{Offset}}$  wird erreicht, dass keine Überlappung zwischen den Trainings- und Testdaten besteht.

Die akquirierten Audiodaten wurden manuell in 9 Klassen eingeteilt, wie sie in Tab. 2.1 aufgeführt sind. Bei den Klassen handelt es sich um eine grobe Unterscheidung der Geräusche, die für den Chirurgen zur Bewertung des Bohrfortschritts während der Operation wichtig sind. Neben verschiedenen Klassen für das Bohren im Knochen und für das Eindringen des Bohrers in den Spinalkanal wurden Klassen für Bohrpausen und die unbeabsichtigte Berührung des Mikrofons definiert. Ausnahmefälle, in denen keine eindeutige Zuordnung des Bohrgeräusches zu einer der definierten Klassen erfolgen konnte, wurden in einer eigenen Klasse ebenfalls berücksichtigt. Hinsichtlich der automatischen Klassifikation dient die manuell erfolgte Klasseneinteilung der Bereitstellung von Zielwerten für Trainingszwecke und der Überprüfung der Klassifikationsergebnisse bei den durchgeführten Tests. Abbildung 2.8 zeigt die manuelle Klassifikation der Bohrgeräusche des ausgewählten Datensatzes aus dem vorigen Abschnitt.

Klasse	Beschreibung
1	undefiniert
2	Leerlauf
3	Unterbrechung
4	Bohren im Knochen, Kortikalis
5	Bohren im Knochen, Spongiosa
6	Bohren im Knochen, leichtes Kratzen, Verkanten des Bohrers
7	Bohren im Knochen, starkes Kratzen
8	Perforation des Wirbelkanals
9	Mikrofon berührt

**Tabelle 2.1:** Die Klassen 1–9 von Bohrgeräuschen für die manuelle Klassifikation.



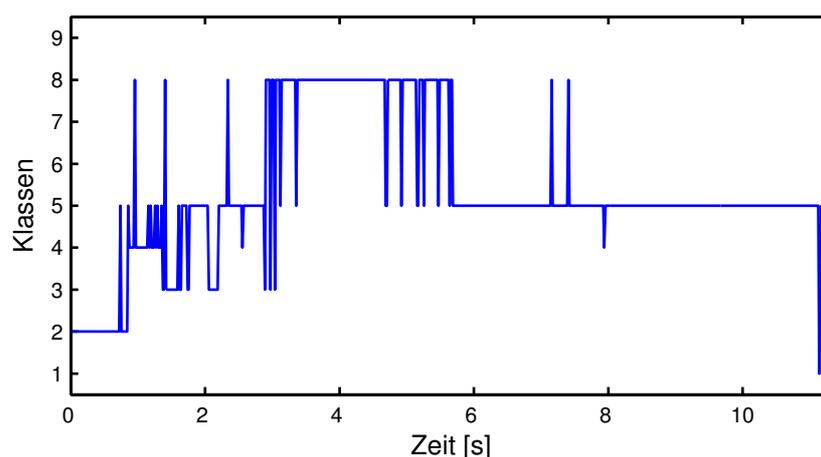
**Abbildung 2.8:** Manuelle Klassifikation der Bohrgeräusche in einem ausgewählten Datensatz.

Zur Klassifikation der Bohrgeräusche mit NNs wurden etliche Tests mit mehrschichtigen Feed-Forward-Netzen durchgeführt. Als Programm-bibliothek diente die Neural Networks Toolbox für Matlab [35]. Die 32 Audiomerkmale aus Abschnitt 2.4.3 bilden die Eingaben für die Netze. Die Zielausgabe stellt ein 1 aus 9 Binärvektor dar, der in derjenigen Komponente einen von Null verschiedenen Eintrag besitzt, die dem manuell klassifizierten Bohrgeräuschtyp aus Tab. 2.1 entspricht. Als Transferfunktion für die Netzknoten werden durchgehend Fermi-Funktionen (siehe Gleichung (B.5)) verwendet. In allen Testfällen erfolgt das Training mit Hilfe des Back-Propagation-Algorithmus (siehe Abschnitt B.3.2).

Als Beispiel für die Erkennungsleistung von NNs soll ein 2-schichtiges Netz mit 50 Neuronen in der ersten Schicht und 9 Neuronen in der zweiten Schicht dienen. Es lieferte in den Tests eine mittlere Erkennungsrate von 77,9% über alle Testdatensätze und war damit eines der besten Netze in der Studie. Im Vergleich dazu liefert ein Netz mit 20 Neuronen in

$P(i, j)$	$j = 1$	2	3	4	5	6	7	8	9
$i = 1$	0,577	0,027	0,010	0,207	0,150	0,013	0,013	0,003	0,000
2	0,136	0,437	0,024	0,194	0,170	0,034	0,005	0,000	0,000
3	0,041	0,110	0,137	0,137	0,411	0,123	0,000	0,041	0,000
4	0,056	0,019	0,007	0,760	0,108	0,022	0,008	0,019	0,000
5	0,010	0,001	0,000	0,027	0,933	0,024	0,001	0,004	0,000
6	0,002	0,002	0,002	0,007	0,229	0,702	0,037	0,015	0,004
7	0,004	0,017	0,004	0,017	0,172	0,086	0,631	0,069	0,000
8	0,000	0,000	0,004	0,028	0,218	0,133	0,098	0,519	0,000
9	0,000	0,000	0,000	0,035	0,000	0,000	0,035	0,000	0,930

**Tabelle 2.2:** Erkennungsraten  $P(i, j)$  des 2-schichtigen NNs bei ausgegebener Klasse  $j$  und vorgegebener Klasse  $i$  über alle Datensätze und Geräuschklassen  $i, j \in \{1, \dots, 9\}$ .



**Abbildung 2.9:** Ausgabe des NNs für den ausgewählten Datensatz.

der ersten Schicht und 9 Neuronen in der zweiten Schicht eine Erkennungsrate von 74,8%. Da Experimente mit 3-schichtigen Netzen zu keiner Verbesserung der Erkennungsleistung geführt haben, sind im weiteren Verlauf nur Netze mit den ausreichenden zwei Schichten betrachtet worden. Tabelle 2.2 zeigt die Wahrscheinlichkeiten  $P(i, j)$  für die Erkennung eines Merkmalsvektors als Klasse  $j$  bei vorgegebener manueller Klassifikation  $i$  mit dem erstgenannten Netz. Man beachte, dass die Klasse 3 „Pause“ und die Klasse 5 „Bohren im Knochen, Spongiosa“ nur schwer mit dem gegebenen Modellen zu differenzieren sind. Jedoch ist eine derartige Unterscheidung hinsichtlich der Identifikation von Risikobereichen anhand der Bohrgeräusche nicht von Relevanz. Die Klassifikation des ausgewählten Datensatzes aus Abschnitt 2.4.4.1 ergibt eine Erkennungsrate von 72,07%. Das Ergebnis der Klassifikation ist in Abb. 2.9 graphisch dargestellt.

$P(i, j)$	$j = 1$	2	3	4	5	6	7	8	9
$i = 1$	0,713	0,073	0,010	0,107	0,090	0,000	0,000	0,000	0,007
2	0,058	0,723	0,024	0,150	0,029	0,015	0,000	0,000	0,000
3	0,000	0,137	0,630	0,027	0,151	0,014	0,027	0,014	0,000
4	0,033	0,031	0,007	0,850	0,051	0,015	0,004	0,006	0,003
5	0,009	0,007	0,002	0,024	0,930	0,019	0,002	0,006	0,000
6	0,011	0,011	0,007	0,031	0,109	0,789	0,007	0,035	0,002
7	0,000	0,009	0,009	0,009	0,086	0,052	0,760	0,077	0,000
8	0,004	0,000	0,025	0,049	0,084	0,056	0,067	0,716	0,000
9	0,000	0,000	0,000	0,012	0,000	0,000	0,000	0,000	0,988

**Tabelle 2.3:** Erkennungsraten  $P(i, j)$  der mit 32 Merkmalen trainierten SVM bei ausgegebener Klasse  $j$  und vorgegebener Klasse  $i$  über alle Datensätze und Geräuschklassen  $i, j \in \{1, \dots, 9\}$ .

Als alternatives Klassifikationsverfahren wurden die Bohrgeräusche mit Hilfe von SVMs analysiert. Zur Abbildung der Merkmale auf einen hochdimensionalen Merkmalsraum und zur Berechnung der trennenden Hyperebenen wurden Kerne aus Gaußschen RBFs

$$k(\mathbf{x}_{d_1}, \mathbf{x}_{d_2}) = e^{-C\|\mathbf{x}_{d_1} - \mathbf{x}_{d_2}\|^2} \quad (2.60)$$

mit  $C > 0$  verwendet (vgl. Gleichung (C.10)). Als Programmbibliothek wurde die Matlab-Toolbox LIBSVM [104] verwendet, mit der der Parameter  $C = 1,95$  auf der Basis einer Gittersuchmethode (siehe [67]) für die Analyse der Bohrgeräusche bestimmt werden kann. Die Programmbibliothek bietet zusätzlich eine Methode zur Klassifikation von mehr als 2 Klassen durch einen paarweisen Vergleich an, so dass alle Bohrgeräuschtypen aus Tab. 2.1 gleichzeitig unterschieden werden können.

Es wurden eine SVM unter Berücksichtigung aller 32 Merkmalswerte beider Mikrofone und zwei SVMs mit jeweils 16 Merkmalswerte der getrennten Mikrofone trainiert. Mit Hilfe der kleineren SVMs wurden maximale Klassifikationsraten von 79,5% für die Merkmale des Oberflächenmikrofons und 72,3% für die des Raummikrofons erzielt. Im Gegensatz dazu erzielte die SVM mit 32 Merkmalen eine Klassifikationsrate von 85,2% für alle Daten. Die Erkennungsraten für die einzelnen Klassen sind analog zu Tab. 2.2 in Tab. 2.3 angegeben. Die Verwendung beider Mikrofone führte zu besseren Ergebnissen als die ledigliche Verwendung eines einzelnen Mikrofons. Für den ausgewählten Datensatz wurde eine korrekte Klassifikation bei 88,8% der Samples erzielt. Das Klassifikationsergebnis ist in Abb. 2.10 dargestellt.

## 2.4.5 Einsatz von HMMs

Neuronale Netze und SVMs betrachten lediglich einzelne Merkmalsvektoren  $\mathbf{o}_t$  zur Klassifikation. Im Gegensatz zu diesen Klassifikationsmethoden beruht die Klassifikation mit



Zur Generierung von Trainingsbeispielen werden, wie einleitend beschrieben, Zeitfenster über die Geräuschmerkmale geschoben und diese zusätzlich anhand der manuellen Klassifikation einer von 9 Trainingsmengen zugeordnet. Jede Trainingsmenge dient als Eingabe für den Baum-Welch-Algorithmus aus Abschnitt 2.3.3.3, mit dem die Spezialisierung des Modells auf die bestimmte Klasse von Beobachtungen erfolgt.

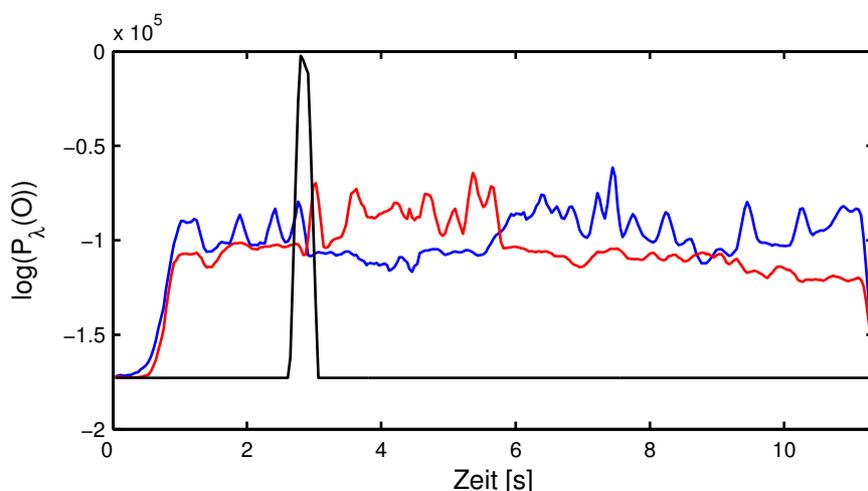
Um eine geeignete Parametrisierung der HMMs zu finden, wurden zahlreiche Tests durchgeführt. Das Hauptaugenmerk gilt bei diesen Tests der Bestimmung der erforderlichen Anzahl der Zustände  $N_S$ , der Länge  $T$  der Beobachtungssequenzen  $O$  und der Anzahl von Mischverteilungskomponenten  $N_M$ . Zunächst ist zu erwarten, dass eine Vergrößerung der jeweiligen Parameter eine Verbesserung der Klassifikationsergebnisse mit sich bringen würde. Die Tests mit  $N_S = 1, \dots, 32$ ,  $N_M = 1, \dots, 32$  und  $T = 10, \dots, 1000$  ergeben jedoch, dass ein optimales Klassifikationsergebnis bereits mit  $N_S = 1$ ,  $N_M = 1$  und  $T = 200$  (dies entspricht 2,1 ms) erzielt werden kann. Bei diesen HMMs handelt es sich demnach um reine Mischverteilungsklassifikatoren (siehe Abschnitt 2.3.3.2). Eine Betrachtung der Historie in den Beobachtungen und das gleichzeitige Durchlaufen einer Sequenz von versteckten Zuständen scheint demnach für die Klassifikation der Bohrgeräusche zunächst nicht erforderlich.

Das gravierende Ergebnis ist die erzielte geringe Erkennungsrate von 21,7% über alle Testdatensätze. In Tab. 2.4 sind die Erkennungsraten nach den einzelnen Klassen aufgeschlüsselt, in denen deutliche Fehlklassifikationen zu erkennen sind. Das Klassifikationsergebnis für den ausgewählten Datensatz aus dem vorigen Abschnitt ist in Abb. 2.11 dargestellt. Fasst man einige Geräuschklassen zusammen, so lässt sich die Erkennungsrate erheblich verbessern. Als Beispiel wurden die Klassen 2 und 3 für Leerlauf und Unterbrechung, die Klassen 4, 5 und 6 für das Bohren im Knochen sowie die Klassen 7 und 8 für starkes Kratzen bzw. Perforation des Spinalkanals zu einer Klasse zusammengefasst und neue HMMs trainiert. Die zugehörigen Erkennungsraten sind in Tab. 2.5 aufgeführt. Betrachtet man alle Testdaten, so erzielen diese HMMs eine Erkennungsrate von 56,4%. Diese liegt aber immer noch deutlich unter denen der NNs und SVMs im vorigen Abschnitt.

Um dieses Phänomen zu klären, werden die Beobachtungswahrscheinlichkeiten  $P_\lambda(O)$  betrachtet, die mit dem Vorwärts-Algorithmus für einzelne Modelle berechnet werden. Abbildung 2.12 zeigt zu dem ausgewählten Datensatz die logarithmischen Wahrscheinlichkeiten  $P_{\lambda_{4,5,6}}$  und  $P_{\lambda_8}$  der Modelle für das Bohren im Knochen und für die Perforation des Spinalkanals. Die Wahrscheinlichkeiten liegen dicht beieinander. Da jedoch nur zwei Modelle betrachtet werden, können die zwei für den Chirurgen entscheidenden Klassen von Bohrgeräuschen dennoch deutlich unterschieden werden. Bei mehr als zwei Klassen treten aufgrund der dicht beieinander liegenden Wahrscheinlichkeiten Fehlentscheidungen durch das Majoritätsprinzip auf.

Der Grund für dieses Verhalten der HMMs ist in den geringen Merkmalsveränderungen innerhalb einer Geräuschklasse zu suchen. Da die Veränderungen fehlen, können keine Übergänge in den versteckten Zuständen der HMMs durch die anliegenden Beobachtungen angestoßen werden. Dies erklärt auch, warum in den Tests mit unterschiedlichen Anzahlen von Zuständen nur unwesentliche Unterschiede bei den Erkennungsraten erzielt wurden





**Abbildung 2.12:** Unterscheidung der zusammengefassten Klassen 4, 5, 6 und der Klasse 8 für Bohren im Knochen und Perforation des Wirbelkanals anhand der Modellwahrscheinlichkeiten  $\log P_{\lambda_{4,5,6}}$  (blaue Kurve) und  $\log P_{\lambda_8}$  (rote Kurve). Zusätzlich sind die Modellwahrscheinlichkeiten  $\log P_{\lambda_{\text{trans}}}$  für ein HMM  $\lambda_{\text{trans}}$  aufgetragen, das zur Erkennung des Übergangs zwischen den zwei Klassen trainiert wurde (schwarze Kurve).

und warum HMMs, die zu Mischverteilungsklassifikatoren degeneriert sind, eine maximale Erkennungsrate liefern können.

Wie in Abschnitt 2.3 beschrieben, werden bei HMMs Veränderungen beobachteter Merkmale dazu benutzt, um Zustandsübergänge in den durch die HMMs modellierten stochastischen Prozessen zu steuern. Die Beobachtungssequenzen müssen dazu einen Verlauf besitzen, der diese Steuerung der Zustände ermöglicht. Um dies auch für Modelle zu den hier vorliegenden Audiodaten zu belegen, wurde ein HMM  $\lambda_{\text{trans}}$  zur Erkennung des Übergangs zwischen zwei unterschiedlichen Geräuschklassen trainiert. Da für die vorliegende Problemstellung aus der Wirbelsäulenchirurgie der Übergang zwischen den Geräuschklassen für Bohren im Knochen (Klassen 4, 5 und 6) und für Perforation des Spinalkanals (Klasse 8) von besonderer Relevanz sind, wurden diese für den Test ausgewählt. Zur Erzeugung geeigneter Trainingsdaten wurden in den bisherigen Trainingsdaten die Zeitpunkte lokalisiert, in denen ein Übergang zwischen den interessierenden Bohrgeräuschen auftritt. Danach wurden mit der gleichen Fensterung wie bei den vorigen Tests Beobachtungssequenzen aus den Trainingsdaten extrahiert und mit ihnen das HMM trainiert. Als initiale Parametrisierungen wurden ebenfalls dieselben wie bei den vorherigen Modellen für einzelne Geräuschklassen verwendet, wobei verschiedene Anzahlen von Zuständen  $N_S$  erprobt wurden.

Abbildung 2.12 zeigt neben den Beobachtungswahrscheinlichkeiten der Modelle  $\lambda_{4,5,6}$  und  $\lambda_8$  für Bohren im Knochen und Perforation des Spinalkanals auch die Beobachtungs-

wahrscheinlichkeiten des Modelles  $\lambda_{\text{trans}}$ , das für den Übergang zwischen diesen zwei ausgewählten Geräuschklassen trainiert wurde. Das Modell verfügt über  $N_S = 18$  Zustände, die es ihm ermöglichen, den Übergang zwischen den Geräuschklassen innerhalb der zu analysierenden Testsequenzen zu finden. Das gefährliche Eindringen des Bohrers in den Spinalkanal wird mit einem deutlichen Peak im Verlauf der Wahrscheinlichkeitskurve angezeigt, der sich an der gleichen Stelle befindet, an der sich auch die Wahrscheinlichkeitskurven der zwei Modelle  $\lambda_{4,5,6}$  und  $\lambda_8$  für einzelne Geräuschklassen schneiden. Der Peak besitzt jedoch wesentlich mehr Signifikanz als die nur schwach ausgeprägte Differenz zwischen Wahrscheinlichkeiten der zwei für Geräuschphasen trainierten HMMs  $\lambda_{4,5,6}$  und  $\lambda_8$  mit  $-43,0 \cdot 10^3 \leq |\log P_{\lambda_{4,5,6}} - \log P_{\lambda_8}| \leq 0$ . Innerhalb von 18140 Samples, dies entspricht 0,19 s, steigt er von Wahrscheinlichkeit  $\log P_{\lambda_{\text{trans}}} = -172,8 \cdot 10^{-3}$  auf  $\log P_{\lambda_{\text{trans}}} = -2,2 \cdot 10^{-3}$  an. Da die Flanke des Peaks stark ausgeprägt ist, wird eine frühzeitige Erkennung der Perforation des Wirbelkanals mit dem einzelnen HMM  $\lambda_{\text{trans}}$  ermöglicht. Beispielsweise kann infolgedessen der Chirurg durch ein Warnsignal alarmiert oder der Bohrer gestoppt werden.

Vergleicht man abschließend die Leistung der HMMs mit denen der NNs und SVMs, so zeigt sich zunächst, dass sich NNs und SVMs für Klassifikationsaufgaben eignen, bei denen einzelne Merkmalsvektoren oder Beobachtungssequenzen mit geringer Variation in den Merkmalsvektoren vorliegen. Der quantitative Vergleich zeigt, dass HMMs bei diesen Aufgaben schlechtere Erkennungsraten liefern. HMMs schneiden deutlich besser ab, wenn Veränderungen der Merkmalswerte in Beobachtungssequenzen dazu genutzt werden können, Übergänge zwischen Zuständen in den HMMs zu steuern. Insbesondere ermöglichen sie die Analyse von Beobachtungssequenzen auf Basis der zeitlichen Veränderungen, die durch NNs oder SVMs in dieser Form grundsätzlich nicht durchgeführt werden kann. In diesen Fällen liefern HMMs signifikante Ergebnisse. Dies legt nahe, die Zustände der HMMs genauer zu untersuchen und gezielt zur Klassifikation einzusetzen. Beispielsweise ist die Assoziation von Phasen des Bohrvorganges mit den Zuständen der HMMs denkbar. Eine Analyse des zeitlichen Zustandsverlauf der HMMs kann so Informationen über den Phasenverlauf liefern und folglich auch zur Klassifikation eingesetzt werden. Diese Vorgehensweise wird daher im folgenden Kapitel erneut aufgegriffen und gezielt im Detail untersucht.

## 2.5 Zusammenfassung

In neurochirurgischen Operationen werden hohe Anforderungen an die Genauigkeit bei der Durchführung der Operation gestellt, da sonst schwerwiegende Verletzungen am Patienten verursacht werden können. Bei Eingriffen an der Wirbelsäule ist das Rückenmark eine besonders gefährdete Risikostruktur. Um ein Wirbelsäulensegment z. B. nach der Schädigung eines Wirbels zu versteifen, muss der Chirurg Schrauben zur Fixierung von Titanplatten in benachbarte Wirbel eindrehen. Dazu werden Kernlöcher mit Hilfe eines chirurgischen Bohrers durch den Pedikel gefräst, ohne dabei das Rückenmark im Spinalkanal zu verletzen. Bei der Durchführung ist der Chirurg im Wesentlichen auf sein anatomo-

misches Wissen und sein sensomotorisches Feingefühl angewiesen. Die Sicht auf das enge Operationsfeld wird durch Blut und chirurgische Instrumente erheblich eingeschränkt und technische Hilfestellung wie Operationsplanungs- und Navigationssysteme sind auf Grund von Problemen bei der Kalibrierung und dem Abgleich mit präoperativ aufgenommenen Tomographiebildern nicht immer zuverlässig.

Zur zusätzlichen Orientierung nutzt der Chirurg bei der Durchführung der Bohrvorgänge auch akustische und haptische Informationen, die durch das Bohren in Knochenbereichen unterschiedlicher Dichte hervorgerufen werden. Mit diesen Informationen kann der Chirurg erkennen, ob die Bohrspitze sich gerade im weichen Knocheninneren oder in den härteren Randbereichen in der Nähe des Spinalkanals befindet. Die akustischen Informationen werden auch in dieser Arbeit verwendet. Sie werden hier jedoch mit Hilfe des Rechners analysiert und können so dem Chirurgen im Sinne der Anthropomatik auf technischem Wege zur Verfügung gestellt werden.

Zur Analyse der Bohrgeräusche werden unterschiedliche Methoden der Signalklassifikation verwendet. Audioaufnahmen von einem Raummikrofon im Operationssaal und einem Oberflächenmikrofon am Wirbel dienen als Grundlage für die Generierung von frequenzbasierten Audiomerkmale, die als Standard aus der Audiodatenverarbeitung bekannt sind. Aus den Geräuschen mehrerer Bohrungen an einem Kadaverpräparat werden Trainings- und Testdaten für die Klassifikationsverfahren erzeugt. Eine manuelle Segmentierung anhand chirurgischen Expertenwissens erlaubt die Bewertung der Klassifikationsergebnisse. Als herkömmliche Klassifikationsverfahren werden zunächst Neuronale Netze und Support-Vector-Machines eingesetzt. Bei der Erkennung der Geräuschtypen anhand einzelner Samples liefern die Verfahren Erkennungsraten von bis zu 72,1 % bzw. 85,2 %.

Im Vergleich zu den herkömmlichen Klassifikationsverfahren werden in dieser Arbeit auch kontinuierliche Hidden-Markov-Modelle für die Erkennung der Bohrgeräusche entwickelt. Der Grundgedanke bei der Anwendung der HMMs zur Klassifikation ist dabei, dass diese nicht nur einzelne Samples, sondern Zeitserien von Samples als Beobachtungen betrachten und mit ihren versteckten Zuständen die nicht direkt messbaren Gewebetypen während des Bohrvorgangs modellieren. Trainingsläufe und Tests mit gefensterten Daten und Auswertungen mit dem Majoritätsprinzip ergeben zunächst nur geringe Erkennungsraten von 21,7 %, die auch durch Zusammenfassen schlecht zu unterscheidender Geräuschklassen nur geringfügig bessere Erkennungsraten von 56,4 % liefern.

Eine genauere Analyse der Beobachtungswahrscheinlichkeiten der Hidden-Markov-Modelle ergibt, dass die Möglichkeit zur Ausbildung von Zustandssequenzen von den Modellen nicht genutzt wird. Stattdessen degenerieren diese zu Mischverteilungsklassifikatoren. Die Unterschiede zwischen den Modellwahrscheinlichkeiten sind bei einer großen Anzahl von Geräuschklassen gering. Der Grund für dieses Verhalten sind die schwachen Variationen der Merkmale während einer Geräuschphase, welche die Übergänge zwischen den versteckten Zuständen der Modelle kontrollieren könnten. Als Gegenmaßnahme wird ein Modell zur Detektion des chirurgisch relevanten Übergangs der Bohrspitze vom Knochen in den Spinalkanal generiert. Dieses Modell verfügt über 18 versteckte Zustände, es betrachtet den

Signalverlauf in dem beobachteten Zeitfenster und es erlaubt eine äußerst robuste und frühzeitige Erkennung der gefährlichen Perforation des Wirbelkanals.

Die automatische Analyse von Bohrgeräuschen stellt eine wichtige Erweiterung für chirurgische Navigationssysteme zur Assistenz des Chirurgen dar. Die in dieser Arbeit vorgestellte Untersuchung von Bohrgeräuschen wird mit exemplarischen Datensätzen vorgenommen. Eine umfassende Untersuchung auch im praktischen Einsatz ist erforderlich, um die Alltagstauglichkeit der Verfahren zu zeigen. Als langfristige Perspektive sollte die automatische Analyse von Bohrgeräuschen ein fester Bestandteil der rechnergestützten Wirbelsäulenchirurgie werden. Der Integrationsaufwand der vorgestellten Systeme in den Operationssaal ist sehr gering. Ebenso sind die anfallenden Kosten für die benötigten Geräte sehr niedrig. Dies kommt insbesondere Bestrebungen zur Kostenreduzierung im Gesundheitswesen zugute.



# Kapitel 3

---

## Zustandserkennung mit Hidden-Markov-Modellen

---

### 3.1 Ableitung von Prozessinformationen

Im Gegensatz zur Klassifikation von Beobachtungssequenzen aus Anthropomatiksystemen durch mehrere HMMs und dem Majoritätsprinzips aus Abschnitt 2.3.3.2 lassen sich auch einzelne HMMs zur Erkennung der Zustände in den Systemen einsetzen. Wie im vorigen Kapitel gezeigt, eignen sich HMMs besonders zur Analyse von Zeitreihen, in denen Informationen über die Zustandsveränderungen des Systems zu erkennen sind. Diese Eigenschaft von HMMs bildet die Basis für die in dieser Arbeit entwickelten Methoden zur automatischen Erkennung von Zuständen in Anthropomatiksystemen. Entscheidend dabei ist die Definition des Zustandsraums und die Bestimmung der Übergangswahrscheinlichkeiten für eine gute Klassifikation. In Abhängigkeit von diesen Parametern durchlaufen HMMs eine Folge von versteckten Zuständen bei der Klassifikation einer Beobachtungsfolge. In dieser Arbeit wird gezeigt, wie die Parameter eines HMMs angepasst werden müssen, damit die Folge von versteckten Zuständen im HMM die zu analysierenden, nicht direkt messbaren Phasen eines Prozesses widerspiegeln. Dazu werden zunächst in diesem Kapitel verschiedene Strukturen von HMMs sowie Algorithmen zur Bestimmung der Zustandsfolgen angegeben. Im zweiten Teil des Kapitels werden diese Parameter gezielt zur Interpretation von Gewebetypen in medizinischen Ultraschallbildern eingesetzt.

### 3.2 HMM-Strukturen

Wie ein HMM die versteckten Zustände für eine Beobachtungsfolge durchläuft, hängt vorwiegend von der Gestalt der Zustandsübergangsmatrix  $\mathbf{A}$  ab. Betrachtet man das HMM als probabilistischen Automaten, so stellt  $\mathbf{A}$  die Adjazenzmatrix des Zustandsgraphen des Automaten dar, d. h. die Struktur des HMMs. Da es sich bei einem HMM um einen probabilistischen Automaten handelt, geben die Einträge  $a_{i,j}$  der Übergangsmatrix die

Übergangswahrscheinlichkeiten  $a_{i,j} = P(q_{t+1} = s_j | q_t = s_i)$  zwischen Zuständen  $s_i$  und  $s_j$  in jedem Zeitpunkt  $t$  einer betrachteten Beobachtungssequenz an. Da immer ein Folgezustand eingenommen werden muss, ist  $\mathbf{A}$  eine semistochastische Matrix, d. h. neben  $a_{i,j} > 0$  gilt

$$\forall i : \sum_j a_{i,j} = 1 \quad (3.1)$$

für die Zeilensummen von  $\mathbf{A}$ . Man beachte, dass die Einhaltung dieses Kriteriums für das korrekte Arbeiten von HMMs entscheidend ist und, wie in Abschnitt 2.3.4.2 beschrieben, besondere Lösungen zur Behebung von numerischen Problemen erforderlich sind.

In Abb. 3.1 sind einige Grundformen für HMM-Strukturen abgebildet. Die in dieser Arbeit entwickelten Modelle ergeben sich aus diesen Grundformen mit zugehörigen Übergangsmatrizen. Sie werden jeweils an die vorliegenden Aufgabenstellungen angepasst und mittels des Baum-Welch-Algorithmus (siehe Abschnitt 2.3.3.3) auf die vorgegebenen Trainingsdaten spezialisiert. Durch das Training können daher auch einzelne Einträge  $a_{i,j} = 0$  werden. Da Nulleinträge bei den Iterationen des Baum-Welch-Algorithmus erhalten bleiben, werden hier angegebene Strukturen aber durch die Nulleinträge charakterisiert, für sonstige Einträge mit möglichen Zustandsübergängen gelte  $a_{i,j} \geq 0$ .

Die allgemeinste Form eines Zustandsgraphen stellt ein ergodischer Graph dar, d. h. von jedem Zustand ist der Übergang in jeden beliebigen anderen Zustand möglich. Die Übergangsmatrix  $\mathbf{A}$  besitzt die Einträge

$$a_{i=1\dots N_S, j=1\dots N_S} = 1/N_S. \quad (3.2)$$

Das zugehörige HMM wird als *ergodisches HMM* bezeichnet. In Abbildung 3.1 a ist der zugehörige Zustandsgraph für  $N_S = 4$  dargestellt.

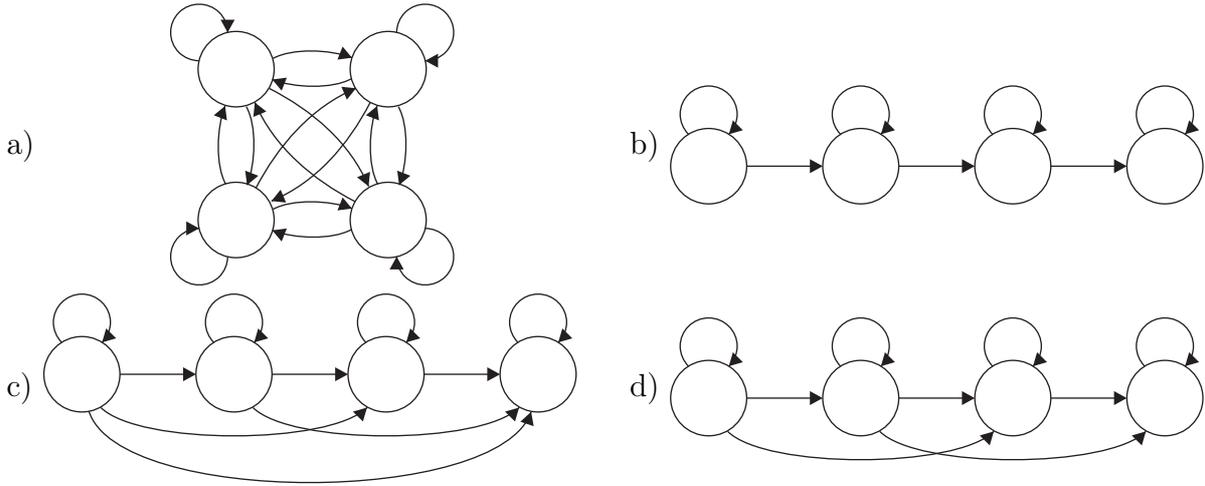
Das einfachste Modell, das sich als Spezialfall aus dem allgemeinen Graphen erzeugen lässt, ist das *lineare HMM*. Wie in Abbildung 3.1 b ersichtlich ist, sind bei diesem Modell nur Übergänge zum jeweils nächsten Zustand bzw. ein Verweilen im aktuellen Zustand durch reflexive Übergänge erlaubt. Die Übergangsmatrix  $\mathbf{A}$  für ein lineares HMMs kann somit geschrieben werden als

$$a_{i=1,\dots,N_S, j=1,\dots,N_S} = \begin{cases} 0,5 & \text{für } j = i \vee j = i + 1, \\ 0 & \text{sonst.} \end{cases} \quad (3.3)$$

Sind beliebige Übergänge in Folgezustände, aber keine Rücksprünge erlaubt, entsteht ein so genanntes *Links-Rechts-Modell* (engl. *left-right model*). Die Übergangsmatrix  $\mathbf{A}$  für dieses HMM lautet

$$a_{i=1,\dots,N_S, j=1,\dots,N_S} = \begin{cases} 1/(N_S - i + 1) & \text{für } j \geq i, \\ 0 & \text{sonst.} \end{cases} \quad (3.4)$$

Die zugehörige schematische Darstellung des Graphen ist in Abbildung 3.1 c zu sehen.



**Abbildung 3.1:** Schematische Darstellung verschiedener Zustandsübergangsgraphen für HMMs mit  $N_S = 4$  Zuständen: a) ergodisches Modell, b) lineares Modell, c) Links-Rechts-Modell, d) Bakis-Modell.

Ein Links-Rechts-Modell, das sich lediglich auf Übergänge in die nächsten zwei Folgezustände beschränkt, wird als *Bakis-Modell* bezeichnet. Abbildung 3.1 d zeigt den Graphen für ein derartiges Modell. Die zugehörige Übergangsmatrix ist

$$a_{i=1,\dots,N_S,j=1,\dots,N_S} = \begin{cases} 1/3 & \text{für } i \leq j \leq i+2, \\ 0 & \text{sonst.} \end{cases} \quad (3.5)$$

Je mehr Einträge von  $\mathbf{A}$  ungleich null sind, desto variabler sind die Pfade durch das Modell. Dies bedeutet entsprechend mehr Aufwand für das Training und die Auswertung des jeweiligen Modells. Die Wahl der initialen Zustandsübergangswahrscheinlichkeiten stellt somit einen Kompromiss zwischen der Modellflexibilität und Berechnungseffizienz dar. Bakis-Modelle haben sich als gute Kompromisslösung bewährt und werden daher häufig in der Praxis eingesetzt (siehe dazu auch [43]). Bei der zielgerichteten Konstruktion von HMMs in dieser Arbeit, wird gezeigt, dass Bakis-Modelle auch aufgrund der Erzeugung der Beobachtungssequenzen zumindest für die untersuchten Fälle geeignete initiale Übergangswahrscheinlichkeiten liefern.

Bei der Anpassung der Übergangswahrscheinlichkeiten aller Modelle kann gezielt auf die Verweildauer in einem Zustand Einfluss genommen werden. Hierzu wird eine Gewichtung der Übergangswahrscheinlichkeiten in den Diagonalelementen  $a_{i,i}$  mit einem Faktor  $0 < w_{1,\dots,N_S} < 1$  vorgenommen. Die übrigen Elemente  $a_{i,j \neq i}$  werden mit  $1 - w_i$  gewichtet. Die Matrix  $\mathbf{A}$  wird als *diagonaldominant* bezeichnet, genau dann, wenn

$$\forall i = 1 \dots N_S, j = 1 \dots N_S : |a_{i,i}| \gg |a_{i,j \neq i}| \quad (3.6)$$

gilt.

### 3.3 Bestimmung von versteckten Zuständen

Das verbreitetste Verfahren zur Bestimmung von Zustandssequenzen eines HMMs zu einer gegebenen Beobachtungssequenz stellt der *Viterbi-Algorithmus* dar. Er liefert die wahrscheinlichste Sequenz von Zuständen  $Q = q_1 \dots q_T$  für eine gegebene Beobachtungssequenz  $O = \mathbf{o}_1 \dots \mathbf{o}_T$ , d. h. es wird diejenige Zustandsfolge aus der Menge aller möglichen Zustandsfolgen  $\{q_1 \dots q_T | q_t \in S\}$  berechnet, für die die A-posteriori-Wahrscheinlichkeit bei gegebener Beobachtung  $O$  maximal ist:

$$Q_{\max} = \arg \max_{Q \in \{q_1 \dots q_T | q_t \in S\}} P_\lambda(Q|O). \quad (3.7)$$

Die bedingte Wahrscheinlichkeit  $P_\lambda(Q|O)$  für das Durchlaufen der Zustandsfolge  $Q$  bei Beobachtung von  $O$  lässt sich mit Hilfe der Bayes-Regel zu

$$P_\lambda(Q|O) = \frac{P_\lambda(Q, O)}{P_\lambda(O)} \quad (3.8)$$

umformulieren und damit die Maximum-Bestimmung vereinfachen, da die Beobachtungswahrscheinlichkeit des Modells  $P_\lambda(O)$  unabhängig von  $Q$  und damit unerheblich ist. Die Bestimmung der optimalen Zustandsfolge gestaltet sich somit einfach als

$$Q_{\max} = \arg \max_{Q \in \{q_1 \dots q_T | q_t \in S\}} P_\lambda(Q|O) = \arg \max_{Q \in \{q_1 \dots q_T | q_t \in S\}} P_\lambda(Q, O). \quad (3.9)$$

Der Viterbi-Algorithmus bedient sich der Technik des dynamischen Programmierens zur Bestimmung der Lösung von Gleichung (3.9). In einem ersten Schritt werden analog zur Berechnung der Vorwärtswahrscheinlichkeiten beim Vorwärts-Algorithmus (siehe Abschnitt 2.3.3.1) für alle Zeitpunkte der Beobachtungssequenz beginnend mit der ersten Beobachtung rekursiv die Wahrscheinlichkeiten  $\delta_t(i)$  für lokal optimale Pfade gespeichert. Es handelt sich hierbei um die maximale Wahrscheinlichkeit

$$\delta_{t+1}(j) := \max_i (\delta_{t+1}(i) a_{i,j} b_j(\mathbf{o}_{t+1})) \quad (3.10)$$

mit der von einem der Zustände zu Zeitpunkt  $t$  in Folgezustand  $j$  zu Zeitpunkt  $t + 1$  gesprungen wird. Bei dieser Berechnung werden nur direkt aufeinanderfolgende Zeitpunkte berücksichtigt. Wie beim Vorwärts-Algorithmus aus Abschnitt 2.3.3.1 kommt die Markov-Eigenschaft des HMMs zum Tragen. Im selben Schritt wird in einer Variablen

$$\psi_{t+1}(j) := \max_i (\delta_{t+1}(j) a_{i,j}) \quad (3.11)$$

der Index des Zustandes gespeichert. Initial werden

$$\delta_1(i) := \pi_i b_i(\mathbf{o}_1) \quad (3.12)$$

mit Hilfe der initialen Zustandsverteilung des HMMs bestimmt bzw. durch

$$\psi_1(i) := 0 \quad (3.13)$$

mit einem beliebigen und für die folgenden Rekursionsschritte irrelevanten Indexwert belegt.

Sind alle Zeitpunkte  $T$  mit Hilfe der Rekursion berechnet, wird im zweiten Schritt des Algorithmus ausgehend vom Finalzustand  $q_T$  mit maximaler Wahrscheinlichkeit die Folge von Zuständen anhand der gespeicherten Zustände zurückverfolgt und damit der global optimale Zustandspfad bestimmt. Der gesamte Ablauf mit beiden beschriebenen Teilen ist in Algorithmus 3.3.1 zusammengefasst.

Gegenüber einer simplen Brut-Force-Suche über alle Zustände und Zeitpunkte mit Aufwand  $\mathcal{O}(TN_S^T)$  besitzt der Viterbi-Algorithmus lediglich den Aufwand  $\mathcal{O}(N_S^2T)$ , da in jedem Zeitpunkt nur benachbarte Zustände betrachtet werden. Weitere Details zum Algorithmus finden sich u. a. in [129] oder [43]. Neben dem Viterbi-Algorithmus wurden alternative Methoden zur Bestimmung der optimalen Zustandsfolge  $Q_{\max}$  entwickelt. Als Beispiel kann hier das so genannte *Token-Passing* (siehe [177], [162]) genannt werden. Die Pfadbestimmung mittels dieser Verfahren ist äquivalent zum Viterbi-Algorithmus. Die Bestimmung der maximalen lokalen Übergangswahrscheinlichkeit erfolgt jedoch nicht durch Auswahl des Pfades anhand des Nachfolgezustandes, sondern anhand des Vorgängerzustandes. Zusätzlich können Zusatzinformationen (*Token*) bei der Konstruktion der Zustandsfolge mitgeführt werden. Die alternativen Verfahren besitzen jedoch keinen Vorteil hinsichtlich der Laufzeiteffizienz, weshalb sich diese Arbeit ausschließlich auf den Einsatz des Viterbi-Algorithmus konzentriert.

## 3.4 Analyse medizinischer Ultraschallbilder

Die Verwendung von versteckten Zuständen zur Analyse von Anthropomatiksystemen wird in diesem Kapitel am Beispiel der Interpretation von medizinischen Ultraschallbildern vorgestellt. Die Analyse der Bilddaten soll sich dabei an den Kriterien orientieren, die der Radiologe bei einer Befundung mit Hilfe der Bilddaten verwendet. Um Ultraschallbilder zu interpretieren, muss der Radiologe mental eine Rekonstruktion der aufgenommenen anatomischen Strukturen auf Basis der Bilddaten durchführen. Dazu muss er große Erfahrung mitbringen, die er sich in langjähriger Praxis angeeignet hat. Neben der Erfahrung ist ein grundlegendes Wissen über die Anatomie und die systematischen Eigenschaften des Bildgebungsprozesses erforderlich. Auch die Befundung von Ultraschallbildern kann daher als eine Aufgabenstellung für die Anthropomatik angesehen werden.

Um die Grundlagen für die in dieser Arbeit vorgestellten Verfahren zu schaffen, wird zunächst auf die typischen Eigenschaften von Ultraschallbildern und die Probleme eingegangen, die bei der Befundung mit ihnen auftreten können. Die auch als *Sonographie* bezeichnete Ultraschallbildgebung bietet gegenüber anderen Bildgebungsverfahren bedeutende Vorteile. Sie gestattet Patientenuntersuchungen praktisch ohne Nebenwirkungen, und Bewegungsabläufe können in Echtzeit betrachtet werden. Aufgrund der geringen Größe der Geräte werden diese zunehmend zur Verlaufskontrolle während chirurgischer Eingriffe eingesetzt. Zusätzlich wird der verbreitete Einsatz auch außerhalb des klinischen Bereichs

---

**Algorithmus 3.3.1** Der Viterbi-Algorithmus.
 

---

```

// Parameter:
//  $O = \mathbf{o}_1 \dots \mathbf{o}_T$ : Observablensequenz
//  $\lambda$ : HMM zur Beobachtung von  $O$ 
// Rückgabe:
//  $Q_{\max}$ : wahrscheinlichste Zustandssequenz,
// die  $\lambda$  bei der Beobachtung von  $O$  einnimmt

// Initialisierung
for( $i := 1, \dots, N_S$ ) {
     $\delta_1(i) := \pi_i b_i(\mathbf{o}_1)$ ;
     $\psi_1(i) := 0$ ;
}

// Teil 1:
// Rekursion
for( $t := 1, \dots, T - 1$ )
    for( $j := 1, \dots, N_S$ ) {
         $\delta_{t+1}(j) := \max_i (\delta_{t+1}(i) a_{i,j} b_j(\mathbf{o}_{t+1}))$ ;
         $\psi_{t+1}(j) := \max_i (\delta_{t+1}(j) a_{i,j})$ ;
    }

// Rekursionsabschluss
 $\delta_T^* := 0$ ;
for( $i := 1, \dots, N_S$ )
    if( $\delta_T^* \leq \delta_T(i)$ ) {
         $\delta_T^* := \delta_T(i)$ ;
         $q_T := i$ ;
    }

// Teil 2:
// Rückverfolgung der Zustandsfolge
for( $t := T - 1, \dots, 1$ )
     $q_t := \psi_{t+1}(q_{t+1})$ ;

return  $Q_{\max} = q_1 \dots q_T$ ;

```

---

ermöglicht auf Grund der relativ geringen Kosten für die Geräte und deren Betrieb, der einfachen Handhabung der Anwendungsteile und des Ausschlusses von gesundheitlichen Belastungen für das Bedienpersonal. Mit der Entwicklung von Ultraschall-Scannern zur Erfassung von dreidimensionalen Aufnahmen bietet sich die Sonographie immer mehr als Ergänzung oder Alternative für Verfahren wie Computertomographie (CT) oder Magnetresonanztomographie (MRT) an.

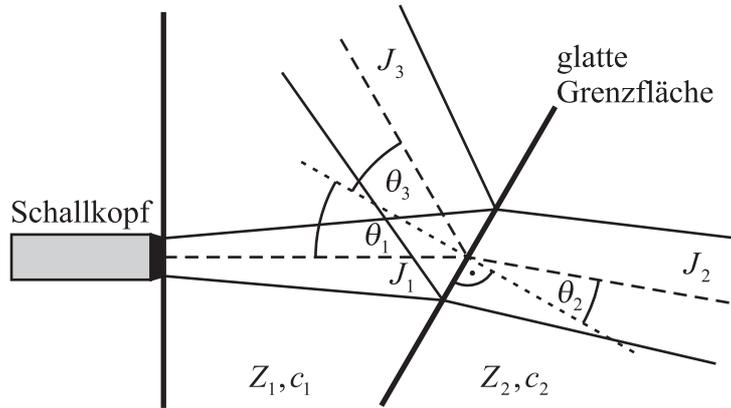
Neben den genannten Vorteilen sind aber gerade auch die Nachteile erheblich, welche die Ultraschalltechnik mit sich bringt. Die große Zahl von negativen Bildeigenschaften, wie sie in Abschnitt 3.4.2 noch aufgeführt werden, erschweren die automatische Analyse der Bilddaten erheblich gegenüber anderen bildgebenden Verfahren. Für eine zuverlässige Interpretation von medizinischen Ultraschallbildern benötigt der Radiologe langjährige Erfahrung bei der Befundung mit Ultraschallbildern. Hauptproblem ist, dass aufgrund von Rauschen und der Charakteristika des bildgebenden Prozesses Informationen über relevante Gewebestrukturen nicht direkt anhand der Grauwerte in den Bildern ablesen lassen, wie dies z. B. bei der CT oder der MRT der Fall ist.

Anstelle einer reinen grauwertbasierten Bildanalyse, betrachtet der Radiologe die Grauwertveränderungen im Bild. Diese sind hauptsächlich in der Tiefe des Bildes zu beobachten, d. h. in der Richtung, mit der in das Gewebe eingestrahlt wurde. Zusätzlich liefern Artefakte, wie z. B. Abschattungen (siehe Abschnitt 3.4.2.4) oder Speckle-Rauschen, die eigentlich für die rechnerbasierte Bildauswertung äußerst störend sind, dem Radiologen weitere wertvolle Hinweise bei der Interpretation des Bildes. Bisherige rechnerbasierte Verfahren zur Analyse von Ultraschallbildern nutzen diese Informationen nicht aus (siehe Abschnitt 3.4.3).

In dieser Arbeit werden stattdessen durch die Verwendung von HMMs probabilistische Modellierungen der Ultraschallbilddaten vorgenommen, in denen die Informationen auf statistischem Wege berücksichtigt werden können. Zur genauen Beschreibung des Verfahrens werden im Folgenden in den Abschnitten 3.4.1 und 3.4.2.1 zuerst die physikalischen Grundlagen für die Ultraschallbildgebung sowie eine Betrachtung des bildgebenden Prozesses mit den charakteristischen Merkmalen für die Gewebedarstellung beschrieben, die auch der Radiologe bei der Bildbefundung nutzt. Eine umfassende Beschreibung der Ultraschallphysik, der Gerätetechnik und der Bildgewinnung kann in [111], [37] oder in [88] nachgelesen werden. Aufbauend auf den Grundlagen zur Ultraschallphysik und -gerätetechnik werden die eigentlichen Methoden dieser Arbeit zur automatischen Analyse von Gewebe in Ultraschallbilddaten dargestellt und anhand von Beispieldatensätzen diskutiert.

### 3.4.1 Physik des Schalls

Die Sonographie beruht auf dem Aussenden und Empfangen von Schallwellen, die in Wechselwirkung mit dem durchschallten Gewebe treten. Die typischen Frequenzen der in der medizinischen Diagnostik verwendeten Schallwellen liegen in einem Bereich von 2–10 MHz. Schall ist eine an ein Medium gebundene Kompressionswelle. Im Falle der medizinischen



**Abbildung 3.2:** Transmission, Reflexion und Brechung von Schallwellen an einer glatten ebenen Grenzfläche zwischen verschiedenen Geweben.

Ultraschalluntersuchung dient das menschliche Gewebe als Medium. Die mit dem Schall verbundene Energie bildet ein *Schallfeld* aus.

Trifft eine Schallwelle auf die Grenzfläche zwischen zwei Geweben, wird die Welle, ähnlich einem Lichtstrahl in der Optik, *reflektiert* und *gebrochen*. Wie stark die auftretende *Reflexion* und *Brechung* ist, hängt vom *Einfallswinkel* der Schallwelle und von den charakteristischen *Schallimpedanzen* der Ausbreitungsmedien ab. Abbildung 3.2 zeigt den Reflexions- und Brechungseffekt eines Schallstrahls an der Grenzschicht zweier Gewebe unterschiedlicher Schallimpedanz. Als Idealisierung handelt es sich bei der Grenzschicht um eine glatte ebene Fläche.

Neben den gewebespezifischen Größen für *Dichte*  $\rho$  und *Schallgeschwindigkeit*  $c$  stellt die Schallimpedanz  $Z$  eine wesentliche Größe bei der Betrachtung der Schallausbreitung dar. Sie ergibt sich aus Dichte und Schallgeschwindigkeit:

$$Z = c\rho. \quad (3.14)$$

Die Einheit für  $Z$  ist Rayl mit  $1 \text{ Rayl} = 1 \text{ g}/(\text{cm}^2\text{s})$ . Gewebespezifische Werte für die Schallimpedanz in der medizinischen Diagnostik sind in Tab. 3.1 angegeben.

Bei gegebenem Einfallswinkel  $\theta_1$  gilt für den Reflexionswinkel  $\theta_3$  das *Reflexionsgesetz*

$$\theta_1 = -\theta_3 \quad (3.15)$$

und für den Ausfallswinkel  $\theta_2$  das *Brechungsgesetz*

$$\frac{\sin \theta_1}{\sin \theta_2} = \frac{\lambda_1}{\lambda_2} = \frac{c_1}{c_2} \quad (3.16)$$

für die Schallgeschwindigkeiten  $c_{1,2}$  bzw. Wellenlängen  $\lambda_{1,2}$  in den zugehörigen Medien.

Eine einfallende Welle mit Intensität  $J_1$  wird in eine reflektierte Welle mit Intensität

$$J_3 = J_1 \left( \frac{Z_1 \cos \theta_2 - Z_2 \cos \theta_1}{Z_1 \cos \theta_2 + Z_2 \cos \theta_1} \right)^2 \quad (3.17)$$

Substanz	$c$ [m/s]	$\rho$ [g/cm <sup>3</sup> ]	$Z = c\rho$ [g/(cm <sup>2</sup> s)=Rayl]	Dämpfung [dB/(MHz cm)]
Fett	1470	0,97	$1,42 \cdot 10^5$	0,5
Knochenmark	1700	0,97	$1,65 \cdot 10^5$	
Muskel	1568	1,04	$1,63 \cdot 10^5$	2
Leber	1540	1,055	$1,66 \cdot 10^5$	0,7
Gehirn	1530	1,02	$1,56 \cdot 10^5$	1
Knochen (kompakt)	3600	1,77	$6,12 \cdot 10^5$	4–10
Knochen (porös)			$2,5 \cdot 10^5$	30–90
Wasser (20 °C)	1492	0,9982	$1,49 \cdot 10^5$	0,002
Luft (NN)	331	0,0013	43	0,5

**Tabelle 3.1:** Gewebespezifische Werte für Schallgeschwindigkeit  $c$ , Dichte  $\rho$ , Schallimpedanz  $Z$  und Dämpfung (Quelle: [111]).

und eine transmittierte Welle mit Intensität

$$J_2 = J_1 \left( \frac{4Z_1 Z_2 \cos \theta_1 \cos \theta_2}{Z_1 \cos \theta_2 + Z_2 \cos \theta_1} \right)^2 \quad (3.18)$$

aufgespalten. Für einen senkrechten Einfall einer Schallwelle auf eine Trennfläche mit Winkel  $\theta_1 = 0^\circ$  ergibt sich damit der *Reflexionskoeffizient*

$$C_R = J_3/J_1 = \left( \frac{Z_1 - Z_2}{Z_1 + Z_2} \right)^2 \quad (3.19)$$

und der *Transmissionskoeffizient*

$$C_T = J_2/J_1 = (1 - C_R) = \frac{4Z_1 Z_2}{(Z_1 + Z_2)^2}. \quad (3.20)$$

Da gewöhnlich die Grenzflächen zwischen verschiedenen Medien nicht glatt sind, sondern eine gewisse Rauigkeit aufweisen, werden Schallwellen gestreut. Auftreffender Schall regt jeden Punkt der Grenzfläche zum Schwingen und zum Aussenden einer Kugelwelle an. Die Überlagerung aller Kugelwellen liefert eine Schallkeule. Die insgesamt rückgestreute Leistung ergibt sich zu

$$P_S = \sigma_S J_1, \quad (3.21)$$

wobei  $J_1$  die einfallende Schallintensität bezeichnet. Die Aufweitung der rückgestreuten Schallkeule hängt von der Wellenlänge und der Rauigkeit der Grenzfläche ab.

Geht Schall durch ein Medium mit statistisch verteilten kleinsten, harten Kugeln mit einem Radius deutlich kleiner als die Wellenlänge  $r \ll \lambda$ , so tritt die so genannte *Rayleigh-Streuung* mit der rückgestreuten Intensität

$$J_S(r_S, \theta_S) = \frac{16}{9} \pi^4 J_1 \frac{r^6}{\lambda^4 r_S^2} \left( 1 + \frac{3}{2} \cos \theta_S \right)^2 \quad (3.22)$$

auf, wobei  $r_S$  den Abstand vom Streuer und  $\theta_S$  den Streuwinkel bezeichnet. Die insgesamt rückgestreute Leistung ergibt sich durch Integration von Gleichung (3.22):

$$P_S = 2\pi \int J_S r_S^2 \sin \theta_S d\theta_S = \sigma_S J_1. \quad (3.23)$$

Der Streuquerschnitt errechnet sich aus

$$\sigma_S = \frac{112}{9} \pi^2 \frac{r^6}{\lambda^4}. \quad (3.24)$$

Erst durch Streuung wird es bei Ultraschallgeräten möglich, Bilder von Organoberflächen zu bekommen, die nicht senkrecht beschallt werden, da sich Sender und Empfänger an derselben Stelle befinden.

Die Intensität einer Schallwelle nimmt während ihrer Ausbreitung im Medium ab. Die kinetische Energie der schwingenden Atome wird dabei durch Reibungseffekte in Wärme umgewandelt. Die Abschwächung erfolgt exponentiell, d. h.

$$J(x) = J(0) e^{C_A x}, \quad (3.25)$$

wobei  $C_A$  den *Absorptionskoeffizienten* bezeichnet. Typische Werte für die Abschwächung in für die medizinische Diagnostik relevanten Materialien sind ebenfalls in Tab. 3.1 zu finden. Für Weichgewebe wie z. B. Leber, Niere oder Gehirn gilt als Näherung

$$10 \frac{d^2(\lg(J(x)/J(0)))}{d\nu dx} = 1 \frac{\text{dB}}{\text{MHz cm}} \quad (3.26)$$

zur Berechnung der *Absorption*.

Anhand von Gleichung (3.26) lässt sich leicht erkennen, dass ein direkter Zusammenhang zwischen der Schallabsorption und der Frequenz  $\nu$  besteht, mit der eingestrahlt wurde. Um eine möglichst hohe Ortsauflösung bei der Darstellung der untersuchten Gewebe zu erhalten, wird eine möglichst kurze Wellenlänge angestrebt. Eine Verkleinerung der Wellenlänge geht jedoch mit einer geringeren *Eindringtiefe*, d. h. der maximalen Tiefe in der ein Objekt noch untersucht werden kann, aufgrund der höheren Absorption einher. Typische Werte für Schallfrequenzen bzw. Wellenlängen und zugehörige Eindringtiefen sowie Ortsauflösungen, die in der medizinische Diagnostik zum Einsatz kommen, sind in Tab. 3.2 zu finden.

Weiterhin führt jede räumliche Begrenzung des Schallfeldes zu *Beugungseffekten*. Auch in der Ultraschalltechnik treten diese Effekte auf, da die Geräte eine begrenzte *Apertur* (Fläche, von der die Schallwellen ausgesendet werden) besitzen. Es ergibt sich ein

Frequenz [MHz]	Wellenlänge [mm]	Eindringtiefe [cm]		Ortsauflösung [mm]	
		hin	rück	lateral	axial
2	0,78	25	3	0,8	
3,5	0,44	14	1,7	0,5	
5	0,31	10	1,2	0,35	
7,5	0,21	6,7	0,8	0,25	
10	0,16	5	0,6	0,2	
15	0,1	3,3	0,4	0,15	

**Tabelle 3.2:** Typische Eindringtiefen und Auflösungen bei vorgegebenen Einstrahlfrequenzen in der Ultraschalldiagnostik (Quelle: [111]).

ungleichmäßiges Schallfeld. Durch den Einsatz von Schallimpulsen wird in der Gerätetechnik versucht, die Beugungseffekte zu eliminieren. Das Schallfeld kann in den *Nah-*, *Fern-* und *Fokusbereich* eingeteilt werden. Nur im Fokusbereich ist das Feld ausreichend homogen und besitzt einen minimalen Durchmesser. Die Lage des Fokusbereichs hängt von der gewählten Schallfrequenz ab und muss möglichst so gewählt werden, dass die zu untersuchenden Strukturen darin zu liegen kommen.

Der *Doppler-Effekt* und die Reflexion an kleinen Strukturen sind weitere physikalische Erscheinungen, die bei der Ultraschalldiagnostik, z. B. für Strömungsmessungen des Blutes, verwendet werden, aber im Rahmen dieser Arbeit nicht weiter erläutert werden sollen. Mehr Details hierzu finden sich z. B. in [51].

## 3.4.2 Medizinische Ultraschallbilder

### 3.4.2.1 Akquisition

Die Akquisition von medizinischen Ultraschallbildern erfolgt durch das Aussenden von Schallwellen in den menschlichen Körper mit einem *Schallkopf*. Reflektierte und gestreute Schallwellen werden als Echos von dem Schallkopf wieder registriert, und durch die Auswertung der Intensitäten und Laufzeiten der Echos ist eine Abbildung der durchstrahlten Objekte möglich.

Zum Aussenden der Ultraschallwellen werden elektrische Signale in Schallwellen umgewandelt. Der Empfang der rücklaufenden Echos erfolgt umgekehrt. Die für beide Fälle benötigten Wandler nutzen die *Piezo-Technik*. Mit ihr lassen sich Membranen aus zumeist keramischen Materialien durch Anlegen von Wechselspannungen oder Spannungsimpulsen in Schwingungen versetzen. Genauso lassen sich Schwingungen auch wieder in Wechselspannungen oder Spannungsimpulse rückwandeln.

Aus der Laufzeit der Echos wird die Tiefe der zugehörigen Schichten ermittelt. Außerdem wird mit der Laufzeit automatisch eine zeitabhängige *Tiefenkompensation* (engl. *depth-*

*gain-compensation, DGC*) vorgenommen, da die Echos mit zunehmender Eindringtiefe, wie im vorigen Abschnitt beschrieben, aufgrund von Absorptionseffekten an Intensität verlieren.

Um mehrdimensionale Aufnahmen zu erhalten, müssen die Schallstrahlen Flächen oder ganze Volumen überstreichen. Für zweidimensionale Aufnahmen sind heute eine Vielzahl stark miniaturisierter Wandler zu so genannten *Convex-* oder *Linear-Arrays* in einem Schallkopf zusammengefasst. Die Einzelwandler werden zeitlich versetzt für einen (*Scan*) angesteuert. Je nach Anordnung der Elemente ergeben sich fächerförmige oder parallele Bilder.

### 3.4.2.2 Darstellung

Die von den Wandlern empfangenen Signale stellen zunächst den Amplitudenverlauf der Spannung an der Empfangseinheit dar, das so genannte *A-Bild*. Erst die Zuordnung von Helligkeiten (engl. *brightness*) zu den Amplituden ergibt das *B-Bild* mit üblicherweise 128 oder 256 Grautönen. Aus dem A-Bild kann im Gegensatz zum B-Bild zum Teil anhand charakteristischer Signalverläufe noch auf Gewebetypen geschlossen werden. Ebenso sind spezielle Rauschreduktionsverfahren möglich (siehe [99]). Das A-Bild findet jedoch meist geräteinterne Verwendung und steht nicht zur Weiterverarbeitung zur Verfügung. Auf weitere Darstellungsmodi, wie z. B. das *TM-Bild*, soll hier nicht weiter eingegangen werden.

Aufgrund des Aufnahmeprozesses repräsentieren die Grauwerte in Ultraschalldaten tiefengewichtete Gradientenbetragswerte von lokal variierenden Impedanzfunktionen, die auf unterschiedlichen Schallgeschwindigkeiten beruhen (vgl. Tabelle 3.1). Durch diese Eigenschaft ist kein funktionaler Zusammenhang zwischen Gewebewerten und den Grauwerten in Ultraschallbildern zu finden [52]. Die Datensätze beinhalten keine direkt messbaren Informationen über Gewebewerte. Stattdessen werden weitestgehend die Ränder von Geweberegionen abgebildet, an denen die Schallstrahlen reflektiert werden [54]. Die Folge ist, dass Ultraschalldaten nicht mit den klassischen Verfahren der medizinischen Bildverarbeitung zur Segmentierung, Registrierung oder Visualisierung verwendet werden können, wie dies etwa bei CT- oder MRT-Aufnahmen der Fall ist [19], [38]. Insbesondere kann nicht die bei CT übliche Zuordnung von Hounsfield-Werten zu Gewebewerten vorgenommen werden (siehe [111]).

Sollen Ultraschallvolumendaten mit dem Rechner verarbeitet werden, ist es normalerweise erforderlich, dass die Volumendaten in einem *Datenquader* gespeichert vorliegen. Da die aufgenommenen Einzelschichten in der Regel gegeneinander verschoben sind oder bei Fächer-Scans eine unpassende Form besitzen, muss während der Volumenberechnung eine Einbettung der Schichtbilder vorgenommen werden. Der Grauwert für die Einbettung entspricht der Amplitude 0 und kann in den eingescannten Bereichen nicht beobachtet werden, da hier immer geringe Echos durch leichte Schwankungen der Schallimpedanz entstehen. Die gewonnenen Datenquader haben üblicherweise eine Größe von  $128 \times 128 \times 100$  bis  $256 \times 256 \times 300$  Volumenelementen (*Voxel*).

### 3.4.2.3 Speckle-Rauschen

Wie bei allen bildgebenden Verfahren tritt auch bei der Ultraschalltechnik Rauschen in den Aufnahmen auf. Für den Ultraschall typisch ist das als *Speckles* bezeichnete Rauschen. Die Theorie zur Speckle-Bildung stammt ursprünglich aus der Optik, wo das gleiche Phänomen bei der Beleuchtung von rauhen Objektoberflächen mit kohärentem Licht und der Abbildung des reflektierten Lichts auf einem Schirm als granulares Muster zu beobachten ist [176]. Die wesentlichen Erkenntnisse zu Speckles kommen aus der Laser- und Radar-Technik [31]. Speckles stellen eine zufällige Intensitätsverteilung eines Wellenfeldes dar, die durch Interferenz entsteht, wenn eine kohärente Welle an einer rauhen Oberfläche oder in einem inhomogenen Medium reflektiert bzw. gebeugt wird. Sowohl die Oberfläche als auch das Medium weisen dabei eine stochastische Struktur mit der gleichen Größenordnung wie die Wellenlänge auf. Dies gilt insbesondere für die in der Radiologie eingesetzten Schallwellen und untersuchten Gewebe.

Hinsichtlich der Analyse von Ultraschallbildern ist zu beachten, dass das durch das Speckle-Rauschen hervorgerufene granulare Muster im Bild nicht den tatsächlich vorhandenen Gewebe-Mikrostrukturen entspricht [92]. Im Gegensatz dazu bewirkt das Speckle-Rauschen sogar, dass feine Detailstrukturen im Bildmaterial nicht mehr beobachtet werden können.

Umfangreiche Untersuchungen haben ergeben, dass sich das Speckle-Rauschen durch ein signalabhängiges, nichtstationäres Rauschmodell beschreiben lässt. In [92] und [78] wird ein Rauschmodell für Ultraschallbilder im B-Mode entwickelt. Anhand von Messungen in verrauschten Ultraschallbildbereichen wurde gezeigt, dass das Verhältnis von Grauwertvarianzen und Mittelwerten in diesen Bereichen konstant ist. Dies legt nahe, dass es sich um ein signalabhängiges Rauschmodell der Form

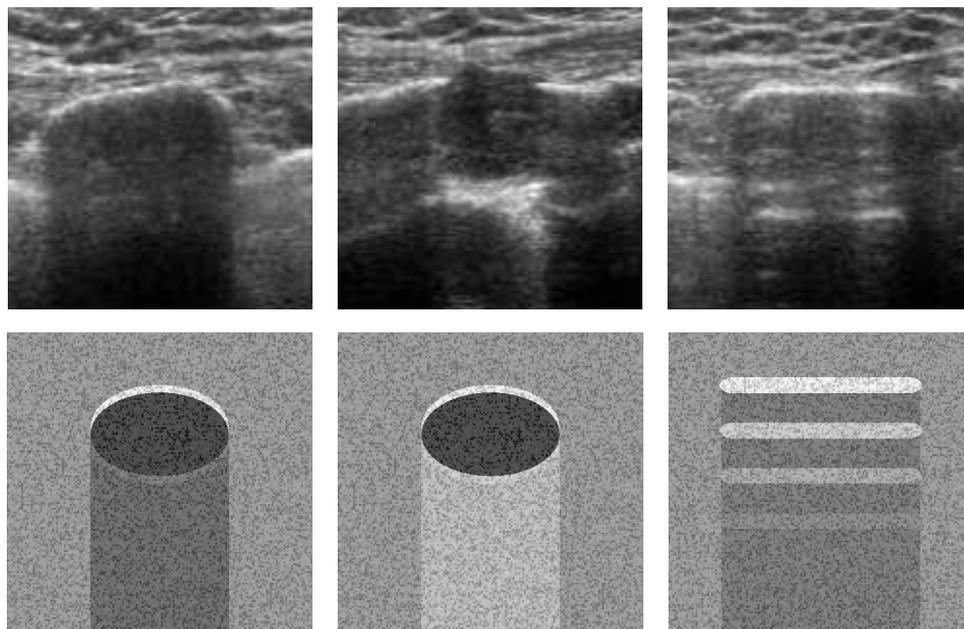
$$\tilde{f}(t) = f(t) + f(t)^\gamma \cdot g(t) \quad (3.27)$$

handelt, wobei  $\tilde{f}$  das beobachtete verrauschte Signal,  $f$  das unverrauschte Signal und  $g$  das Rauschen darstellt. Im Modell wird angenommen, dass  $g$  ein weißer Gaußscher Rauschprozess ist. Üblicherweise wird der Exponent  $\gamma = 1/2$  gesetzt. Das Modell wurde in dieser Form experimentell bestätigt.

Eine wichtige Feststellung in [92] ist, dass Signalverarbeitungsmethoden, wie z. B. Tiefenkompensation (siehe Abschnitt 3.4.2.1), zusätzlich das Speckle-Rauschen verstärken können und damit die Bildinformation zusätzlich verfälschen. Um dennoch verlässliche Bildinformationen zu erhalten wurden zahlreiche Filtermethoden für Ultraschallbilder entwickelt. Hauptsächlich kommen hierbei nichtlineare Filterungen zum Einsatz, wie z. B. adaptive gewichtete Medianfilter [92], adaptive Speckle-Suppression Filter [78], homomorphe Filter [72] oder Wavelet-basierte Signalaufbereitungsverfahren [1], [92], [78]. Eine Übersicht über die wesentlichen Filterverfahren ist in [99] zu finden.

### 3.4.2.4 Artefakte

Neben dem Speckle-Rauschen ist eine weitere Schwierigkeit bei der Ultraschallbildgebung die große Zahl an auftretenden Aufnahme Fehlern, den *Artefakten*. Sie haben unterschied-



**Abbildung 3.3:** Artefakte in der medizinischen Ultraschallbildgebung. Obere Zeile: Beispiele aus einer radiologischen Aufnahme des Brustbeins. Links: Abschattung hinter der Knochenoberfläche des Brustbeins. Mitte: Verstärkung der Bereiche hinter einem Tumor mit geringer Dämpfung. Rechts: Mehrfachechos durch die Knochenoberfläche des Brustbeins. Untere Zeile: vereinfachte graphische Darstellung der jeweiligen Artefakte.

lichste Ursachen und Erscheinungsformen (siehe auch [111]). Die am häufigsten auftretenden Artefakte sind:

- *Abschattungen*: sie entstehen besonders an starken Reflektoren. Ein Großteil der Schallenergie wird von diesen Reflektoren reflektiert und gestreut. Die tieferliegenden Bereiche erscheinen relativ zu ihrer Umgebung zu dunkel. Reflektiert ein Gewebe nicht direkt zum Sensor zurück, kann es trotzdem aufgrund der Abschattung, die es verursacht, lokalisiert werden.
- *Schallverstärkungen*: sie treten bei Strukturen auf, die hinter einem Bereich mit geringer Dämpfung liegen. Dies kann durch Tiefenkompensation noch verstärkt werden.
- *Mehrfachechos* und *Spiegelungen*: sie erscheinen bei stark reflektierenden Oberflächen.

Abbildung 3.3 zeigt Beispiele für die beschriebenen Artefakte. Neben den vorgestellten Artefakten treten etliche weitere in der medizinischen Ultraschallbildgebung. Eine ausführlichere Übersicht ist in [111] zu finden. In vielen Fällen liefern die beschriebenen Artefakte

dem erfahrenen Radiologen wichtige Zusatzinformationen, die sonst nicht direkt anhand der bloßen Grauwerte beobachtbar wären. Bisher ist es allerdings noch nicht gelungen, die Effekte durch Rechnereinsatz zu erkennen, zu nutzen oder wirksam zu eliminieren.

### 3.4.3 Stand der Forschung

Die meisten Arbeiten, die sich mit der Verarbeitung von medizinischen Ultraschallbildern befassen, haben zum Ziel, die Daten hinsichtlich einer verbesserten Darstellung aufzubereiten. In erster Linie betrifft dies Schichtbilder. Für die Aufbereitung werden unterschiedlichste Filtermethoden verwendet. Standard sind bereits klassische Filterverfahren, wie z. B. die Medianfilterung, für die auch adaptive Formen entwickelt wurden [114], [5], [139]. Mittlerweile finden aber auch Verfahren, die die Signalcharakteristika von Ultraschallaufnahmen berücksichtigen Anwendung. In [99], [100], [58] und [2] werden Wavelet-basierte Verfahren zur Reduktion des Speckle-Rauschens entwickelt. Die Rauschreduktion erfolgt in diesen Fällen durch Wavelet-Shrinkage, bei dem Wavelet-Koeffizienten unterhalb eines bestimmten Schwellwertes eliminiert werden [36]. Die Arbeiten [99] und [100] benutzen zur Schätzung geeigneter Schwellwertparameter lokale Statistiken und homomorphe Filterungen. In [2] wird eine Maximum-a-posteriori-Schätzung (MAP) mit Hilfe einer Bayes'schen Signalmodellierung vorgenommen. Zusätzlich ermöglicht eine logarithmische Darstellung der Daten die Übertragung des multiplikativen Rauschmodells auf ein besser zu handhabendes additives Rauschmodell. Ein Multiskalen-Prozess erlaubt die Analyse von Signalstatistiken in [3]. Alternativ wird in [78] eine adaptive Glättung durch ein Region-Growing-Verfahren durchgeführt. Das Region-Growing-Verfahren orientiert sich dabei ebenfalls an lokalen Statistiken.

Neben der Datenaufbereitung befassen sich in den letzten Jahren die meisten Arbeiten mit der Segmentierung und vor allem der Visualisierung von Volumendatensätzen. Jüngste Neuentwicklungen auf dem Gebiet der Wandlertechnik erweitern dieses Gebiet auch um Zeitserien von Volumendatensätzen [118]. Neben der Untersuchung innerer Organe werden die Volumendaten in der pränatalen Diagnostik eingesetzt. In [137] und [138] wird dazu eine Multiskalen-Technik entwickelt, bei der eine schrittweise Binarisierung, Tiefpassfilterung, Schwellwertbildung und Propagation der Ergebnisse auf feinere Skalierungsstufen erfolgt. Die als BLTP bezeichnete Technik ermöglicht eine geglättete Darstellung von Objektstrukturen, die in Ultraschalldaten gut zu lokalisieren sind. In [80] werden modifizierte Median-Filter anstelle von BLTP eingesetzt, die zwar eine verbesserte Wiedergabe von Detailstrukturen ermöglichen, aber dafür Konturen rauer darstellen. Die Arbeiten [106] und [109] übertragen die BLTP-Technik auf ein effizienteres Wavelet-Verfahren, in das zusätzlich eine Rauschreduktion mittels Wavelet-Shrinkage integriert werden kann.

Die beschriebenen Verfahren orientieren sich alle an den Grauwerten im Ultraschallbild. Diese stellen jedoch keine Gewebewerte dar und werden stark durch den Aufnahmeprozess beeinflusst, z. B. bei Veränderung der Einstrahlrichtung oder der Tiefenkompensation. Ansätze zu einer approximativen Rekonstruktion von Gewebewerten werden in [52] auf Basis pseudo- und mikrolokaler Bildeigenschaften entwickelt. In [172] wird eine Rekonstruktion

der Tiefengewichtung und eine anschließende Intensitätskorrektur der Grauwerte vorgenommen. Zielsetzung des Verfahrens ist die Segmentierung von Grauwertwertbereichen mit Markov-Random-Fields. Für eine echte Rekonstruktion von Gewebewerten ist immer noch eine aufwendige, physikalisch korrekte Modellierung des Bildgebungsprozesses auf Basis von Differentialgleichungen zur Beschreibung des Wellenausbreitungsprozesses notwendig [81], [28]. Deren effiziente Umsetzung und Auswertung mit dem PC sind jedoch heute noch nicht möglich. Selbst bei vereinfachten Modellen der Wellenausbreitung werden z. B. bei der Simulation des Ultraschallbildgebungsprozesses Rechenzeiten im Stundenbereich benötigt [75], [42], [156].

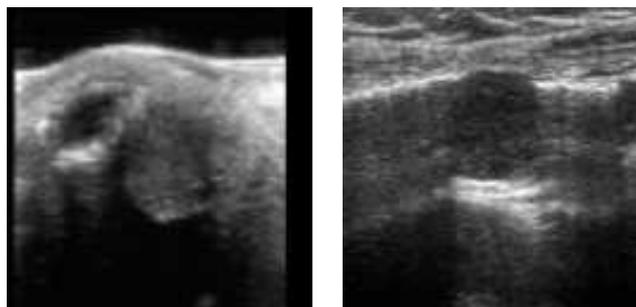
### 3.4.4 Modellierung mit Hidden-Markov-Modellen

#### 3.4.4.1 Verwendetes Datenmaterial

Als Datenmaterial für die Betrachtungen in den folgenden Abschnitten sollen zwei Beispieldatensätze dienen, die aus der klinischen Radiologie stammen. Es handelt sich bei den Daten um Schichten aus Volumendatensätzen. Die Größen der gesamten Datensätze betragen  $128 \times 128 \times 128$  bzw.  $128 \times 128 \times 256$  Voxel. In Abbildung 3.4 sind die zwei im Folgenden als *Beispiel 1* und *Beispiel 2* bezeichneten Einzelschichten dargestellt.

Das linke Bild (*Beispiel 1*) zeigt die Aufnahme eines menschlichen Auges. Der Schallkopf wurde bei der Aufnahme nicht direkt, sondern mit einem Kopplungsstück auf das Auge aufgesetzt. Somit ist auch die natürliche äußere Form des Auges im oberen Bildrand zu erkennen. Ein Tumor ist als hellere Region in der Mitte des Bildes zu erkennen. Er befindet sich in der Nähe des Ziliarkörpers, der die Iris umgibt. Die Iris selbst ist als linsenförmige Struktur in der linken oberen Bildecke deutlich zu erkennen. Die genauen Konturen sind unscharf und zeichnen sich nur gegenüber der Flüssigkeit im Glaskörper deutlich ab. Das rechte Bild (*Beispiel 2*) ist der Scan eines menschlichen Sternums, bei dem der Schallkopf direkt auf die Brust aufgesetzt wurde. Die unregelmäßigen Strukturen im oberen Bereich des Bildes stellen Haut und Fettschichten dar. Ein Tumor ist in diesem Bild als dunkle, runde Struktur in der Mitte des Bildes zu erkennen. Das Sternum selbst ist nur in Form der Knochenränder durch die hellen Kanten im Bild sichtbar. Bei den hellen Bildpunkten unterhalb des Tumors handelt es sich um Aufhellungsartefakte, die durch die geringere Gewebedichte des Tumors gegenüber dem umgebenden Knochenmaterial verursacht werden.

Aus der Sicht des Radiologen ist in beiden Fällen von besonderem Interesse, die in den Datensätzen auftretenden Regionen verschiedenen Gewebetyps zu rekonstruieren und damit eine Unterscheidung der Bereiche mit tumorösem Gewebe von jenen Bereichen ohne maligne Strukturen zu erhalten. Unter diesem Gesichtspunkt wurde eine manuelle Segmentierung der Datensätze auf der Basis des radiologischen Expertenwissens vorgenommen.



**Abbildung 3.4:** Ultraschallbilder für die Analyse mit HMMs. Links: Aufnahme des Auges mit Tumor im Bereich des Ziliarkörpers (Beispiel 1). Rechts: Aufnahme des Sternums mit Tumor im Knochen (Beispiel 2).

#### 3.4.4.2 Grundprinzip der Modellierung

Im Gegensatz zu den in Abschnitt 3.4.3 beschriebenen Verfahren zur Verarbeitung von Ultraschalldaten orientiert sich die hier vorgestellte Methode direkt am Aufnahmeprozess. Um die erforderlichen Gewebeformationen zu erhalten, wird ein probabilistisches Modell für die Ultraschalldaten entwickelt. Das Modell beschreibt einzelne Scan-Zeilen bei der Ultraschallaufnahme und versucht deren zeitliches dynamisches Verhalten nachzubilden [107].

Die einzelnen Scan-Zeilen sind als Spalten in den Volumendatensätzen zu finden. Jede Spalte repräsentiert eine Zeitserie von Echoamplituden, die am Schallwandler nach Ausenden der Schallwelle wieder registriert werden. Die Modellierung dieser Zeitserien kann erneut als Problem der Analyse von Beobachtungssequenzen durch HMMs verstanden werden. Das Grundprinzip bei der Modellierung der Scan-Zeilen mit HMMs ist, dass die unbekanntes Größen der Gewebewerte durch die Zustände des HMMs dargestellt werden und durch die direkt gemessenen Grauwerte als Beobachtungen kontrolliert werden. Aufgrund der Eigenschaften des Bildgebungsprozesses stellen die Grauwerte in den Bildern Informationen über örtliche Änderungen von Schallgeschwindigkeit und Schallimpedanz dar, über die auf Gewebewerte rückgeschlossen werden muss (siehe Abschnitt 3.4.2.2). Folglich werden die Übergänge zwischen den Zuständen durch Kanteninformationen zu den Gewebebereichen gesteuert.

Bei der Verwendung der Grauwerte als Merkmale für die Beobachtungen ist zu beachten, dass es sich bei den Werten meist um bereits vorverarbeitete Signale aus B-Bildern handelt. Beispiele für deren Vorverarbeitung sind z. B. Rauschreduktion oder Tiefenkompensation. Wünschenswert wäre für die Analyse von HMMs die direkte Verarbeitung der Echo-Amplituden, wie sie unmittelbar vom Schallwandler geliefert werden. Diese Amplituden-Information ist allerdings bei den in der Diagnostik eingesetzten Geräten nicht in digitaler Form verfügbar, so dass hier auf die bearbeiteten B-Mode-Informationen zurückgegriffen wird und durchgeführte Filterungen wieder kompensiert werden müssen.

Die vom Modell berücksichtigten Gewebetypen bilden die Zustandsmenge  $S$  und die im Datensatz auftretenden Grauwerte die Menge der beobachtbaren Symbole  $V$ . Üblicherweise handelt es sich bei den Merkmalen in  $V$  um 256 diskrete Grauwerte. Folglich können effiziente diskrete HMMs für die Modellierung verwendet werden, deren Beobachtungswahrscheinlichkeiten als semistochastische Matrix  $\mathbf{B}$  in Form einer Lookup-Table abgespeichert werden. Die Beobachtungssequenzen  $O = \mathbf{o}_1 \dots \mathbf{o}_T$  mit  $\mathbf{o}_{t=1,\dots,T} \in V$  sind die gemessenen Zeitserien von Echoamplituden, d. h. die mittels des B-Modes umgesetzten Grauwerte in den Spalten der Bilddatensätze. Für die Beispiele 1 und 2 aus Abschnitt 3.4.4.1 ergeben sich Sequenzen der Länge  $T = 128$ .

### 3.4.4.3 Modellstruktur und Parameterwahl

Allgemein ist eine geeignete Wahl der Parameter für die Zustandsübergangsmatrizen  $\mathbf{A}$ , der Beobachtungswahrscheinlichkeiten  $\mathbf{B}$  und der initialen Zustandsbelegung  $\boldsymbol{\pi}$  entscheidend für eine verlässliche Funktionsweise von HMMs. Insbesondere haben die Zustandsübergangswahrscheinlichkeiten und die Modellstruktur, die über die Adjazenzrelationen in Matrix  $\mathbf{A}$  gegeben sind, einen wesentlichen Einfluss auf die Modellierung der Bildspalten.

Zielsetzung ist, die Abfolge von Gewebestrukturen in der Tiefe des Objekts nachzubilden. Dabei soll den verschiedenen Gewebetypen jeweils genau ein Zustand zugeordnet werden. Gewebe gleichen Typs in unterschiedlichen Tiefen sollen zur besseren Unterscheidung auch unterschiedlichen Zuständen zugeordnet werden. Eine Zusammenführung zu einer Gewebeklasse ist nachträglich durchführbar. Dementsprechend wird für die HMMs ein Links-Rechts-Modell gewählt. Hiermit wird eine strikte Abfolge der Zustände bzw. Gewebetypen mit zunehmender Tiefe gewährleistet. Die Matrix  $\mathbf{A}$  hat damit die für Links-Rechts-Modelle typische Dreiecksgestalt, wie sie in Gleichung (3.4) in Abschnitt 3.2 definiert wird.

Zusätzlich zur Gewebeabfolge muss gewährleistet werden, dass das Modell so lange wie möglich in einem Zustand verweilt. Der Grund hierfür ist, dass bei häufigen Zustandsübergängen in Geweben mit ähnlicher Charakteristik oder bei Rauschen in homogenen Bereichen die Erfassung großer zusammenhängender Regionen gestört würde. Dies macht sich dadurch bemerkbar, dass lediglich eine Unterscheidung der Gewebetypen in den oberen Bereichen und nicht über die gesamte Tiefe des erfassten Objektbereichs erreicht wird. Zur Übertragung dieser Eigenschaft auf das Modell wird die Matrix  $\mathbf{A}$  diagonaldominant (siehe Definition in Abschnitt 3.2) gewählt. Unter Beibehaltung der Einschränkung an Matrix  $\mathbf{A}$  können die restlichen Einträge beliebig gewählt werden. In der vorliegenden Implementierung wird  $\mathbf{A}$  initial wie in Gleichung (3.28) gesetzt. Für die Wahrscheinlichkeiten auf der Hauptdiagonalen wird dabei ein hoher heuristischer Wert  $C_{\text{diag}} = 0,95$  gewählt. Die Wahrscheinlichkeiten für die Übergänge in Folgezustände sind gleichverteilt. Alle Wahrscheinlichkeiten müssen durch das Training für die Spezialisierung des Modells

noch angepasst werden.

$$a_{i=1,\dots,N_S,j=1,\dots,N_S} = \begin{cases} C_{\text{diag}} & \text{für } j = i \neq N_S, \\ 1 & \text{für } j = i = N_S, \\ \frac{1-C_{\text{diag}}}{N_S-i} & \text{für } j > i, \\ 0 & \text{sonst} \end{cases} \quad (3.28)$$

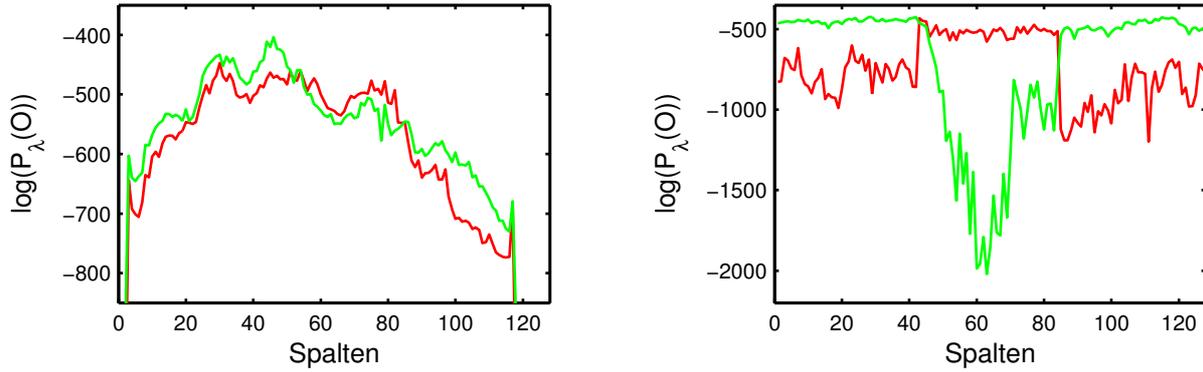
Für die Beobachtungswahrscheinlichkeiten  $\mathbf{B}$  und die initialen Zustandsverteilungen  $\boldsymbol{\pi}$  bestehen nur die Einschränkungen bezüglich stochastischer Normierung, die allgemein für HMMs vorgegeben sind (siehe Abschnitt 2.3). Die Parameter  $\boldsymbol{\pi}$  und  $\mathbf{B}$  werden daher gleichverteilt mit  $\pi_{i=1,\dots,N_S} = 1/N_S$  und  $b_{i=1,\dots,N_S,k=1,\dots,N_V} = 1/N_V$  als Anfangswerte für das anschließende Training gewählt.

Nach der Definition der Parameter wird das Modell mit Hilfe des Baum-Welch-Algorithmus aus Abschnitt 2.3.3.3 mit ausgewählten Trainingsspalten spezialisiert. Das resultierende HMM ist dasjenige, das eine allgemeine Bildspalte in einem Datensatz oder Bildspalten mit bestimmten enthaltenen Gewebetypen durch Beobachtungssequenzen  $O$  nachbildet. Zu beachten ist, dass nicht immer alle  $N_V$  Beobachtungen aus der Symbolmenge  $V$  in den Trainingsdaten enthalten sind. Die Modelle werden bezüglich dieser Beobachtungen nicht spezialisiert und der Trainingsalgorithmus setzt für die entsprechenden Symbole  $\mathbf{v}_{k_0}$  die Wahrscheinlichkeiten  $b_{i,k_0} = 0$  für alle Zustände  $i = 1, \dots, N_S$ . Bei einem Test würde eine Beobachtungssequenz  $O$  mit einem Auftreten von  $\mathbf{o}_t = \mathbf{v}_{k_0}$  die Wahrscheinlichkeit  $P_\lambda(O) = 0$  berechnen. Um dies zu verhindern, werden nach den Trainingsschritten die betreffenden Spalten in  $\mathbf{B}$  durch Kopieren der Wahrscheinlichkeiten aus der Spalte eines zulässigen benachbarten Symbols und damit eines Grauwertes mit vernachlässigbarem Helligkeitsunterschied eliminiert.

### 3.4.5 Klassifikation von Bildspalten

Zunächst sollen HMMs zur Unterscheidung von Bildspalten in einem Ultraschalldatensatz nach der Methode im vorigen Abschnitt konstruiert und getestet werden. Es soll als exemplarische Aufgabenstellung erkannt werden, welche Spalten Gewebe mit Tumorstrukturen beinhalten und welche aus Bereichen mit ausschließlich gesundem Gewebe stammen. Dazu werden zwei HMMs  $\lambda_T$  für tumoröses Gewebe und  $\lambda_B$  für gesundes Gewebe, wie im vorigen Abschnitt beschrieben, definiert und mit entsprechenden Spalten aus jeweils einer Schicht mit Trainingsdaten generiert. Mit Hilfe des Vorwärts-Algorithmus (siehe Abschnitt 2.3.3.1) können nun die jeweiligen Wahrscheinlichkeiten für die Beobachtung einer Testspalte berechnet werden. Abbildung 3.5 liefert die Wahrscheinlichkeiten  $P_{\lambda_T}(O)$  und  $P_{\lambda_B}(O)$  für alle Spalten  $O = O_{x=1,\dots,128}$  zu Beispiel 1 und Beispiel 2 aus Abschnitt 3.4.4.1. Man beachte, dass auch hier aus numerischen Gründen logarithmische Wahrscheinlichkeiten (siehe Abschnitt 2.3.4.1) angegeben sind. Die Modelle für beide Beispiele wurden individuell für das jeweilige Beispiel trainiert.

Unstetigkeiten an den Rändern der Kurven für Beispiel 1 resultieren aus der Einbettung der Ultraschalldaten nach der Akquisition. Die HMMs reagieren empfindlich auf diese



**Abbildung 3.5:** Beobachtungswahrscheinlichkeiten  $\log P_{\lambda_T}(O)$  (rote Kurve) und  $\log P_{\lambda_B}(O)$  (grüne Kurve) für die Bildspalten der Einzelschichten aus Abb. 3.4 als Testdaten. Links: Beispiel 1. Rechts: Beispiel 2.

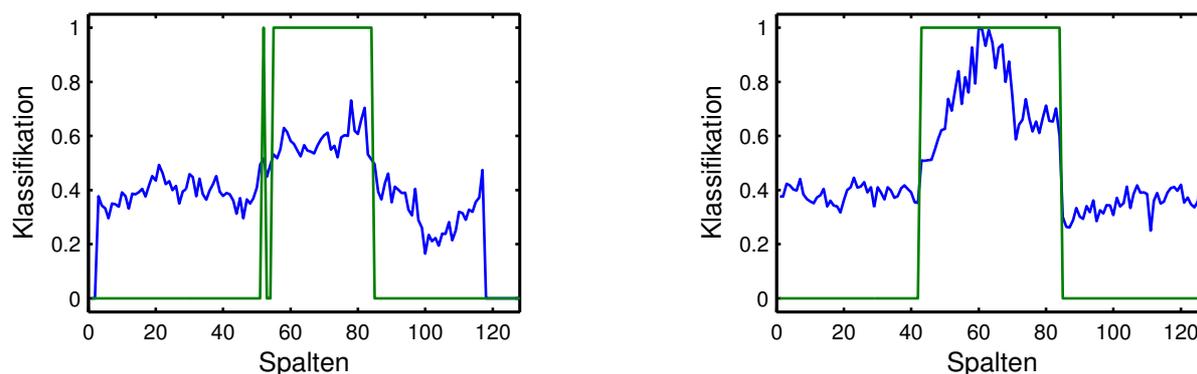
Einbettung. Grund hierfür ist, dass die Spalten der Einbettung nicht die Charakteristika der eintrainierten Spalten mit Ultraschalldaten enthalten. In diesen Fällen gehen beide Wahrscheinlichkeiten  $P_\lambda(O)$  gegen den Wert 0.

Mit Hilfe des Majoritätsprinzips aus Abschnitt 2.3.3.2 kann das HMM ausgewählt werden, das den Typ der Spalte angibt. Das Majoritätsprinzip liefert eine *harte Klassifikation* der Spalten im Datensatz. Eine präzisere Klassifikation kann durch eine *weiche Klassifikation* durchgeführt werden. Sie ergibt sich aus den Verhältnisse zwischen den betrachteten Wahrscheinlichkeiten  $\Delta P(x) := \log P_{\lambda_T}(O_x) - \log P_{\lambda_B}(O_x)$  und zusätzlicher Normalisierung auf das Intervall  $[0, 1]$  über alle Spaltenindizes  $x = 1, \dots, X = 128$ :

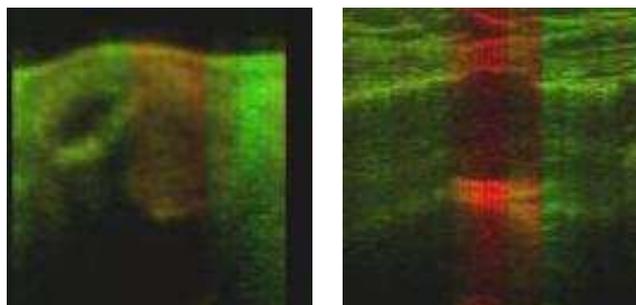
$$c_{\text{soft}}(O_x) := \frac{1}{2} \left( 1 + \frac{\Delta P(x)}{\max_{1, \dots, X} |\Delta P(x)|} \right). \quad (3.29)$$

Die mittlere Abweichung  $\sigma_{\text{soft}} = \frac{1}{X} \sum_{1, \dots, X} (c_{\text{soft}}(O_x) - 0,5)$  der Verhältnisse  $\Delta P(x)$  vom Mittelwert 0,5 gibt zusätzlich ein Maß für die Signifikanz der weichen Klassifikation an.

Abbildung 3.6 zeigt die Ergebnisse der harten und weichen Klassifikation für die Beispiele aus Abschnitt 3.4.4.1. Aus den Kurvenverläufen wird ersichtlich, dass die HMMs eine deutliche Klassifikation von Bildspalten mit und ohne Tumorgewebe sowohl durch die harte als auch durch die weiche Klassifikation vornehmen können. Aufgrund der ausgeprägteren Kanteninformationen, welche die Zustandsübergänge im Modell steuern, liefert die weiche Klassifikation im Beispiel 2 eine bessere Unterscheidung der Bildspalten, was sich auch durch die Signifikanz  $\sigma_{\text{soft}} = 0,19$  für Beispiel 2 gegenüber  $\sigma_{\text{soft}} = 0,11$  für Beispiel 1 äußert. Zur Veranschaulichung sind in Abb. 3.7 die Ergebnisse der weichen Klassifikation durch Gewichtung der Rot- und Grünkanäle für die Farbwerte in den Spalten der originalen Ultraschallbilder nochmals dargestellt. In beiden Beispielen werden die Spalten mit Tumorgewebe durch die Rotfärbung korrekt hervorgehoben. Insgesamt wurden die trainierten Modelle mit Testspalten aus jeweils drei ausgewählten Schichten der



**Abbildung 3.6:** Ergebnisse der harten Klassifikation (grüne Kurve) und der weichen Klassifikation (blaue Kurve) für die beiden Testdatensätze. Links: Beispiel 1. Rechts: Beispiel 2.

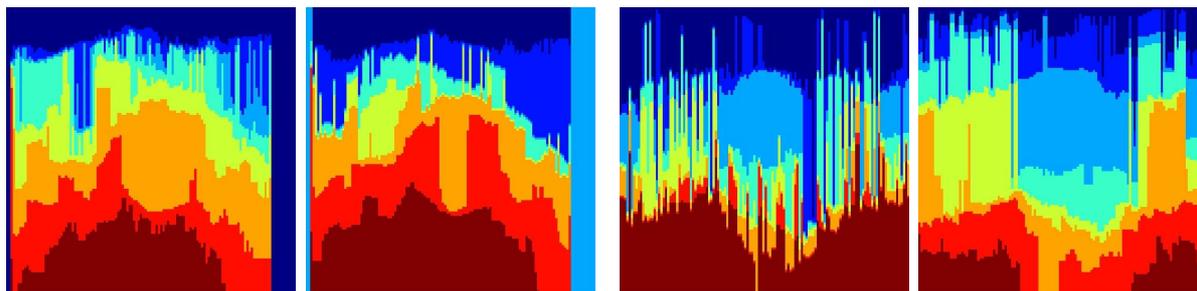


**Abbildung 3.7:** Darstellung der Ergebnisse der weichen Klassifikation durch Gewichtungen des Rot- und des Grünkanals in den originalen Ultraschallbildern. Die Spalten mit Tumorgewebe sind durch die Rotfärbung von den Spalten mit gesundem Gewebe zu unterscheiden. Links: Beispiel 1. Rechts: Beispiel 2.

Beispieldatensätze getestet. Bei den resultierenden 384 Spalten zu Beispiel 1 wird eine Erkennungsrate von 99,2% und bei denen zu Beispiel 2 eine Erkennungsrate von 97,1% erzielt.

### 3.4.6 Rekonstruktion von Gewebeinformation aus Zuständen

Die im vorigen Abschnitt beschriebene Datenklassifikation gibt lediglich Auskunft über gesamte Spalten. Jedoch ist auch eine Unterscheidung der Gewebetypen innerhalb der Spalten wünschenswert. Wie in Abschnitt 3.4.4.3 beschrieben repräsentieren die Zustände der HMMs für die Spaltenklassifikation die Regionen mit unterschiedlichen Schallgeschwindigkeiten und damit auch unterschiedliche Gewebewerte. Es ist daher naheliegend,



**Abbildung 3.8:** HMM-Zustände zur Rekonstruktion von Gewebetypen. Die Indizes der Zustände sind farblich codiert. Bilder links: Zustände der Modelle für Beispiel 1. In beiden Teilbildern ist die Oberfläche des Auges als Übergang im blauen Bereich zu erkennen. Braune Bereiche stellen das Innere des Auges dar. Im linken Teilbild für das Tumor-Modell zeichnet sich der Tumor orangen ab. Hingegen werden Strukturen in vergleichbaren Tiefen rechts und links des Tumors im rechten Teilbild vom Nicht-Tumor-Modell besser klassifiziert, z. B. die Augenlinse (gelbe Struktur links des Tumors). Bilder rechts: Zustände des Modells für Beispiel 2. Weichgewebe vor dem Brustbein wird dunkelblau dargestellt, das Gewebe in den tieferen Regionen hinter dem Brustbein in rot und braun. Im rechten Teilbild für das Nicht-Tumor-Modell ist das Brustbein an der gelben und orangenen Farbe erkennbar. Im linken Teilbild für das Tumor-Modell wird der Tumor als kreisrunde Struktur in blauer Farbe erkennbar. In allen Bildern treten kammartige Artefakte aufgrund von Klassifikationsunterschieden zwischen benachbarten Bildspalten auf.

dass die Rekonstruktion der Zustandsfolge zu einer gegebenen Beobachtungssequenz, d. h. einer zu untersuchenden Spalte aus dem Datensatz, direkt zu den Gewebeinformationen führt.

Um die Rekonstruktion von Gewebetypen zu demonstrieren, wird der Viterbi-Algorithmus aus Abschnitt 3.3 auf die HMMs in Verbindung mit den Beispieldatensätzen angewandt. Die Zielsetzung ist dabei die Lokalisierung der Tumorregionen innerhalb der Spalten und die Unterscheidung vom umgebenden gesunden Gewebe. Abbildung 3.8 zeigt die Zustände bzw. Gewebebereiche, die für die Einzelschichten aus Abschnitt 3.4.4.1 ermittelt wurden. Die unterschiedlichen Farbwerte repräsentieren die Nummern der Zustände. Als Anzahl von Zuständen wurde in diesem Experiment  $N_S = 8$  gewählt. Diese Anzahl von Zuständen ist ausreichend um die wichtigsten Gewebetypen in den Beispieldatensätzen zu erfassen.

In Abb. 3.8 lassen sich die korrespondierenden Zustände erkennen, obwohl die Farbwerte nicht unbedingt übereinstimmen müssen. Es zeigt sich auch, dass das Training der Modelle für tumoröses und gesundes Gewebe zu einer Spezialisierung für bestimmte Zustandssequenzen führt. Modell  $\lambda_T$  detektiert die Gewebetypen in den Tumorbereichen besser, während Modell  $\lambda_B$  die Gewebe in den umgebenden gesunden Regionen verlässlicher auflöst. Eine Selektion der Zustände für das passendste Modell kann allgemein nach

dem Majoritätsprinzip aus Abschnitt 2.3.3.2 auf Basis der Modellwahrscheinlichkeiten erfolgen.

### 3.4.7 Stabilisierung der Zustände durch Merkmalsaufbereitung

In den Arbeiten [53], [56] und [55] wurde belegt, dass durch die Vorverarbeitung mit so genannten *Calderón-Operatoren* eine deutliche Verbesserung von Ultraschalldaten hinsichtlich der Segmentierbarkeit von Geweben für unterschiedlichste Anwendungen erzielt werden kann. Auch im Zusammenhang mit der Rekonstruktion von Gewebeeigenschaften durch HMMs zeichnet sich diese Datenvorverarbeitungsmethode dadurch aus, dass sie geeignete Merkmale zur verlässlicheren Kontrolle der versteckten Zustände generiert. Dies wird im Folgenden anhand der Beispieldatensätze gezeigt.

Sei  $\hat{f}$  die Fourier-Transformierte einer Funktion  $f : \mathbb{R}^D \rightarrow \mathbb{R}$  mit

$$\hat{f}(\boldsymbol{\omega}) := [\mathcal{F}f](\boldsymbol{\omega}) = \int_{\mathbb{R}^D} f(\boldsymbol{x}) e^{-i\langle \boldsymbol{x}, \boldsymbol{\omega} \rangle} d\boldsymbol{x}, \quad (3.30)$$

dann lassen sich, wie in [52] beschrieben, Calderón-Operatoren  $\mathcal{C}_{\alpha, \Omega_e}$  im Fourier-Raum als

$$\left[ \mathcal{C}_{\alpha, \Omega_e} \hat{f} \right](\boldsymbol{\omega}) := \|\boldsymbol{\omega}\|^\alpha \Omega_e \left( \frac{\boldsymbol{\omega}}{\|\boldsymbol{\omega}\|} \right) \hat{f}(\boldsymbol{\omega}) \quad (3.31)$$

mit dem Parameter  $\alpha \in \mathbb{R}$  und einer Richtungsgewichtung  $\Omega_e : \mathbb{R}^D \rightarrow \mathbb{R}$  darstellen. Sie bilden eine Unterklasse der Pseudodifferentialoperatoren, die genau die linearen Operatoren  $\mathcal{C}$  enthält, die im Fourier-Raum der Abschätzung

$$\left\| \frac{\partial^\alpha}{\partial \boldsymbol{\omega}^\alpha} \hat{f}(\boldsymbol{\omega}) \right\| \leq C_\alpha \|\boldsymbol{\omega}\|^{-\|\alpha\|} \quad (3.32)$$

für beliebige  $\boldsymbol{\alpha} \in \mathbb{R}^D$  und einer Konstanten  $C_\alpha \in \mathbb{R}$  genügen (siehe [154]). Bezüglich der  $\mathcal{C}_{\alpha, \Omega_e}$  gelte hierbei vereinfachend für alle Einträge  $\alpha_1, \dots, \alpha_D = \alpha$  in Vektor  $\boldsymbol{\alpha}$ . Die zusätzliche auf der  $D$ -dimensionalen Einheitskugel definierte Gewichtungsfunktion  $\Omega_e$  bewirkt, dass der Operator eine Vorzugsrichtung  $\boldsymbol{e}$  besitzt und somit eine anisotrope Verarbeitung von Daten gestattet. Eine Realisierung für die Gewichtungsfunktion  $\Omega_e$  ist

$$\Omega_e \left( \frac{\boldsymbol{\omega}}{\|\boldsymbol{\omega}\|} \right) := \text{sgn}(\sin(\theta(\boldsymbol{\omega}, \boldsymbol{e}))) |\sin(\theta(\boldsymbol{\omega}, \boldsymbol{e}))|^w, \quad (3.33)$$

wobei  $\theta(\boldsymbol{\omega}, \boldsymbol{e}) := \arccos(\langle \boldsymbol{\omega}, \boldsymbol{e} \rangle)$  die Winkelabweichung zwischen  $\boldsymbol{e}$  und  $\boldsymbol{\omega}$  berechnet. Der Parameter  $w$  gibt eine zusätzliche Gewichtung dieser Abweichung an.

Zur Aufbereitung von Ultraschallbildern werden diese zunächst mittels einer Schnellen Fourier-Transformation (FFT) (siehe [10]) in den Fourier-Raum transformiert, dort die beschriebenen Calderón-Operatoren angewandt und danach wieder in den Ortsbereich zurücktransformiert. Wie in [52] gezeigt wird, bewirken dabei Operatoren  $\mathcal{C}_{\alpha, \Omega_e}$  mit Parametern  $\alpha < 0$  eine lokale Integration der Grauwerte in den Ultraschallbildern. Diese

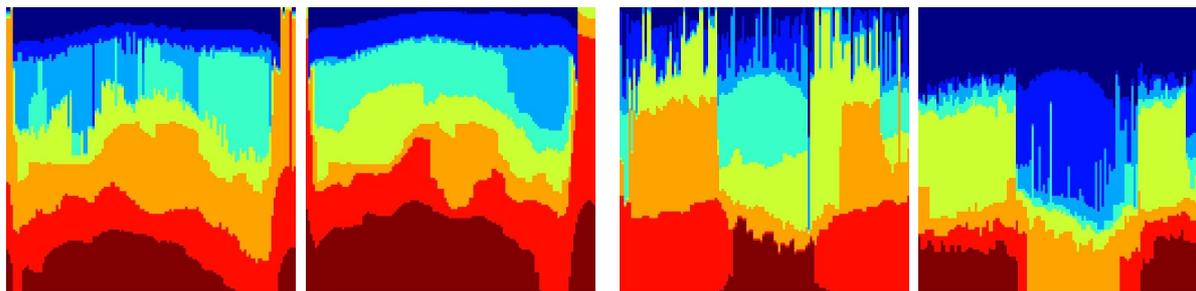
wird mittels der Richtungsgewichtung  $\Omega_e$  so eingeschränkt, dass als Vorzugsrichtung  $e$  die Einstrahlrichtung des Schalls verwendet wird. In [52] wird als geeigneter Wert für die Gewichtung der Winkelabweichung  $w = 0,1$  empfohlen. Üblicherweise handelt es sich bei den in dieser Arbeit betrachteten Parallel-Scans bei Vektor  $e$  um die Richtung der Spalten in den Datensätzen. Die Integration bewirkt eine Glättung der Daten, durch die das Rauschen in den Grauwerten deutlich reduziert wird. Relevante Kantenstrukturen bleiben hingegen erhalten. Gerade diese Eigenschaft ist wichtig für die Anwendung der hier beschriebenen HMMs, da die Zustandsübergänge, wie in Abschnitt 3.4.4.2 beschrieben, von den Kanteninformationen gesteuert werden.

Als Beispiel wurde der Operator  $\mathcal{C}_{\alpha, \Omega_e}$  mit  $\alpha = -0,2$  auf die Trainingsdaten aus dem vorigen Abschnitt angewandt, um die initiale Parametrisierung der Modelle  $\lambda_T$  und  $\lambda_B$  mit vorverarbeiteten Daten zu spezialisieren. Danach wurden Tests mit den ebenfalls vorverarbeiteten Testdaten durchgeführt. Abbildung 3.9 zeigt die resultierenden detektierten Zustände für die Ultraschallbilder nach der Calderón-Vorverarbeitung. Wie in Abbildung 3.9 zu erkennen ist, bewirkt die Vorverarbeitung eine Stabilisierung der Zustandsübergänge beim Vergleich benachbarter Spalten. Die Ausbildung von kammartigen Artefakten ist deutlich vermindert. Eine Erhöhung der Signifikanz  $\sigma_{\text{soft}}$  bei weicher Klassifikation auf bis zu 0,269 belegt ebenfalls die Wirksamkeit der Vorverarbeitung mit Calderón-Operatoren.

## 3.5 Zusammenfassung

Medizinische Ultraschallbilder unterscheiden sich auf charakteristische Weise von Aufnahmen anderer Bildgebungsmodalitäten, wie z. B. die Computertomographie oder die Magnetresonanztomographie. In der Regel sind die Bilder stark verrauscht, bieten einen schlechten Kontrast und weisen viele Artefakte auf. Der wichtigste Unterschied ist jedoch, dass die Grauwerte in den Bildern keine Informationen über Gewebewerte darstellen, sondern stattdessen örtliche Änderungen von Schallgeschwindigkeit und Schallimpedanz repräsentieren, über die auf Gewebetypen rückgeschlossen werden muss. Ein Radiologe braucht dazu langjährige Erfahrung und ein fundiertes Wissen über die Anatomie und die Eigenschaften des Bildgebungsprozesses. Mit diesem Wissen kann er sich mental ein Bild über die realen Strukturen im Körper machen. Die Befundung von Ultraschallaufnahmen stellt folglich ein gutes Beispiel aus der Anthropomatik dar.

Die genauere Betrachtung der Ultraschallbildgebung hinsichtlich der zugrunde liegenden physikalischen Modelle, der Gerätetechnik und der Erzeugung der Bilddaten aus Wandlersignalen ergeben, dass im Wesentlichen Kantenstrukturen und Inhomogenitäten der Gewebe in Bildspalten aufgetragen werden. Die Bildspalten können als Zeitserien aufgefasst werden, die mit Hilfe von HMMs modelliert und analysiert werden können. Die Grundlage hierbei ist die Verwendung der Grauwerte im Bild als Beobachtungen für die Modelle und die Nachbildung der Gewebeverläufe durch die versteckten Zustände in den HMMs. Hierzu müssen geeignete Parametrisierungen gefunden werden, die eine vollständige Erfassung der Zustände über die volle Länge einer Bildspalte erfassen.



**Abbildung 3.9:** HMM-Zustände zur Rekonstruktion von Gewebetypen mit Anwendung des Calderón-Operators im direkten Vergleich zu Abb. 3.8. In allen Bildern ist zu erkennen, dass Kammartefakte deutlich vermindert und dadurch Objektkonturen wesentlich besser zu erkennen sind. Bilder links: Die Oberfläche des Auges ist erneut als Übergang in den blauen Bereichen sichtbar. Das Innere des Auges ist rot und braun. Im linken Teilbild für das Tumor-Modell sind Tumorstrukturen an der Orangefärbung zu erkennen. Im Nicht-Tumor-Modell zeichnet sich im Gegensatz zu Abb. 3.8 die Linse (gelb) deutlicher ab und selbst ein Teil des Knochens um das Auge wird erkennbar (mittelblaue Struktur am rechten Bildrand). Bilder rechts: Anhand der Zustandsverläufe für das Tumor-Modell im linken Teilbild ist erneut der Tumor als kreisrunde Struktur in der Bildmitte erkennbar (türkis). Die Bereiche des Brustbeins rechts und links des Tumors (gelb) heben sich besser vom Gewebe vor und hinter dem Brustbein (blau bzw. rot, braun) im rechten Teilbild für das Nicht-Tumor-Modell ab.

Anhand von Beispielen aus der klinischen Radiologie wird gezeigt, wie HMMs mit Hilfe des Majoritätsprinzips eine verlässliche Klassifikation von Bildspalten durchführen. Zielsetzung ist, mit den Modellen Bereiche gesunden Gewebes von Bereichen mit Tumorstrukturen zu unterscheiden. Bei zwei ausgewählten Testbeispielen werden Erkennungsraten von 99,2 % bzw. 97,1 % erzielt. Neben der Klassifikation von Bildspalten ist ein Hauptziel der Arbeit, die Zustandsequenzen der Modelle zu bestimmen und damit auch Rückschlüsse über die in den Spalten auftretenden Gewebeverteilungen zu gewinnen. Anhand der Beispieldaten wird gezeigt, dass die verwendeten HMMs sich durch das Training auf bestimmte Gewebestrukturen spezialisieren und damit gezielt auf die Gewebeerkenntnis durch die Zustände Einfluss genommen wird. Kammartige Artefakte, die bei der Rekonstruktion der Zustandssequenzen aufgrund der spaltenweisen Betrachtung zwischen benachbarten Spalten entstehen, können durch geeignete Merkmalsaufbereitung reduziert werden. Hierbei kommen Calderón-Operatoren zum Einsatz, welche eine Glättung der Bilddaten bei gleichzeitiger Erhaltung der im Bild enthaltenen Kantenstrukturen bewirken. Diese Eigenschaft der Aufbereitungsoperatoren ist entscheidend für die Analyse mit HMMs, da die Übergänge zwischen Geweben und damit auch zwischen den versteckten Zuständen der HMMs über die Kanteninformationen definiert sind.

Aus radiologischer Sicht liefern die vorgestellten Methoden vielversprechende Ergebnisse für zukünftige Anwendungen, die sich die automatische Analyse von Ultraschalldaten mit Hilfe der HMM-Zustände zu Nutze machen können. In zukünftigen Arbeiten müssen die Modelle mit einer größeren Anzahl von Datensätzen getestet und mit Referenzdaten, z. B. aus CT-Aufnahmen, verglichen werden. Anwendungsgebiete für die entwickelten Methoden sind das Durchsuchen größerer Datenbanken, beispielsweise radiologische Bildarchive, nach bestimmten Gewebestrukturen oder die Kontrolle der Krankheitsentwicklung und des Behandlungsverlaufs bei einem Patienten.

# Kapitel 4

---

## Zielgerichtete Merkmalsgenerierung und Modellkonstruktion

---

### 4.1 Problemstellung

Nachdem in den vorigen Kapiteln eine neue Methodik zur Analyse von Anthropomatomiksystemen basierend auf der Zustandsauswertung von HMMs gezeigt wurde, soll in diesem Kapitel die Merkmalsgenerierung und Modellbildung in einer umfassenden Bewegungsstudie durchgeführt und dabei im Detail untersucht werden. Die Bewegungsanalysen im Rahmen dieser Arbeit sind auf anthropomatische Fragestellungen ausgerichtet, die sich bei der Interaktion des Menschen mit einem Roboter ergeben. Humanoide Roboter sollen zunehmend in unser Alltagsleben integriert werden. Bedarf hierfür gibt es insbesondere bei der Unterstützung von alten und behinderten Menschen. Um ein effektives Arbeiten mit dieser Zielgruppe zu erreichen und um eine große Akzeptanz des Roboters durch den Menschen zu erzielen, ist es erforderlich, dass der Roboter einerseits die Bewegungen des Benutzers richtig interpretieren kann und sich bei seiner Handlungsplanung auf die aktuellen Aktionen des Benutzers einstellt. Andererseits soll sich der Roboter auch durch sein Äußeres und somit auch durch seine Bewegungen den menschlichen Vorstellungen von einer natürlichen Bewegung annähern.

Bei den beschriebenen Aufgabenstellungen ist die Berücksichtigung der intra-individuellen Variationen ebenso wie die der personenspezifischen Charakteristika von zentraler Bedeutung. Sie sind maßgeblich für die stochastische Bewertung und Optimierung von Bewegungsdaten bzw. Bewegungssequenzen. Durch diese Größen wird die Aufbereitung und Glättung von Bewegungsdaten, die Erkennung und Prädiktion von Bewegungsverläufen sowie die Generierung bzw. Übertragung von Bewegungstrajektorien für andere anatomische Konstellationen gesteuert. Die Ermittlung und Berücksichtigung individueller Charakteristika ist vor allem dann entscheidend, wenn menschliche Bewegungen automatisch analysiert und auf einen humanoiden Roboter übertragen werden sollen, um dem Roboter menschenähnliches Verhalten beizubringen und so die Grundlage für eine zuverlässige Kooperation zwischen Mensch und Roboter zu schaffen.

Den zwei Aufgaben der Bewegungserkennung und der Bewegungsausführung ist die gemeinsame Fähigkeit zur Bewertung von Bewegungstrajektorien hinsichtlich ihrer Menschenähnlichkeit erforderlich. Hierzu gehört es insbesondere, elementare Bewegungsphasen zu identifizieren sowie individuelle Charakteristika, die in menschlichen Bewegungen enthalten sind, zu erkennen.

Um diese individuellen Charakteristika menschlicher Bewegungen zu ermitteln, muss zunächst eine gezielte Extraktion geeigneter Bewegungsmerkmale vollzogen werden. Mit Hilfe der Merkmale müssen auf der einen Seite inter-individuelle Abgrenzungen (Variationen bei Ausführung einer bestimmten Bewegung durch verschiedene Personen) durchgeführt und personenspezifische Charakteristika ermittelt werden können. Auf der anderen Seite müssen anhand der Merkmale *intra-individuelle Variationen* (d. h. Variationen bei mehrmaliger Bewegungsausführung durch ein und dieselbe Person) bewertet werden können. Gleichzeitig sollen die ermittelten personenspezifischen Charakteristika weitgehend invariant gegenüber diesen intra-individuellen Variationen sein. Zudem muss durch die Bewegungsmerkmale eine Unterscheidung von Bewegungsphasen bzw. Bewegungszuständen gegeben sein, damit eine automatische Analyse von Bewegungssequenzen u. a. mit Hilfe von automatischen Klassifikationsmethoden (siehe Abschnitt 4.5) erfolgen kann. Dies betrifft vor allem die Analyse mit HMMs, wie sie in dieser Arbeit durchgeführt wird.

## 4.2 Stand der Forschung

Die Untersuchung der charakteristischen Eigenschaften von menschlichen Bewegungen ist ursprünglich in den weitgefächerten Bereichen von Arbeitswissenschaft, Sportwissenschaft, Biomechanik, Funktionaler Anatomie, Neurophysiologie bis hin zur Psychologie angesiedelt. Aus diesen Bereichen gibt es bereits eine Vielzahl von Arbeiten, in denen Bewegungen sowohl qualitativ als auch quantitativ mit sehr hohen räumlichen und zeitlichen Auflösungen analysiert werden und bereits ausführlich dokumentiert sind [116], [170], [39], [148]. Bei diesen Arbeiten steht die Untersuchung des menschlichen Bewegungsapparates im Vordergrund. Der Rechner kommt dabei überwiegend zur Akquisition von Bewegungsdaten und deren Aufbereitung zum Einsatz [4], [98].

Für die Erfassung von Bewegungen haben sich bereits vielfältige Verfahren auf dem Markt etabliert. Die deutlichsten Unterschiede bei den Verfahren liegen in der zeitlichen und räumlichen Auflösung in einem gegebenen Raumvolumen, dem Aufwand für die Vor- und Nachbereitung einer Messung sowie im Anschaffungspreis. Oft unbeachtet bleibt die Rückwirkungsfreiheit der Messverfahren auf die Bewegungsausführung. Gerade bei der Erfassung von charakteristischen Bewegungsmerkmalen dürfen Probanden nicht durch angebrachte aktive Marker, Sensoren oder Kabel in ihrem Bewegungsraum eingeschränkt werden. Eine genauere Übersicht und eine Erläuterung der Vor- und Nachteile der in dieser Arbeit eingesetzten Akquisitionsverfahren gibt Abschnitt 4.3.

Stark zunehmend findet auch die Analyse von Bewegungen mit Hilfe des Rechners Verbreitung. Hauptanwendungsgebiete sind das maschinelle Sehen basierend auf Videostreamen

und die Generierung realitätsnaher Bewegungen von Avataren in virtuellen Umgebungen. Dementsprechend sind auch die verwendeten Methoden gezielt auf diese Aufgabenstellung zugeschnitten [165].

Grundlage für die Arbeiten zur Erkennung und Klassifikation von Personenbewegungen in Videostreamen ist eine robuste Extraktion geeigneter Bewegungsmerkmale [161]. Hierzu werden zumeist signifikante Punkte in den Videodaten verfolgt oder grobe Skelettmodelle von Probanden erstellt. Feine Strukturen, wie z. B. die einzelnen Finger, werden vernachlässigt oder zumindest nicht in der Auflösung erfasst, wie sie für die Bewegungsanalysen in dieser Arbeit benötigt werden. Bei der Extraktion von Bewegungsdaten aus erfassten Trajektorien werden in der Literatur häufig reine Positionsdaten verwendet [165], im Gegensatz zu den in dieser Arbeit eingesetzten Trajektorien mit Winkelinformationen oder Orientierungsangaben zu Markern. Der Grund hierfür ist, dass sich harte Vorgaben für die Bewegungsgenerierung, wie etwa das Berühren von Gegenständen oder die Energie-minimierung auf der Basis von Positionsangaben leichter umsetzen lassen. Alternative Ansätze, die sich mit der Extraktion von Bewegungsmerkmalen aus Winkeltrajektorien, z. B. durch Betrachtung unterschiedlicher Frequenzbänder, der Korrelation von Gelenkwinkeln und Texturierung befassen, sind in [125], [124] zu finden.

Bisherige Arbeiten für die Erzeugung von menschlichen Bewegungen sind zu einem großen Teil im Bereich der Animationstechnik publiziert worden. Die dabei angewendeten Techniken realisieren die Synthese von Bewegungen durch eine Verkettung von Teilbewegungen und der Berechnung von glatten Übergängen anhand von geeigneten Kostenfunktionen [83], [82], [164]. Häufig wird zusätzlich noch eine echtzeitnahe Verarbeitung der Daten angestrebt [86]. Dabei wird versucht, den Einfluss manueller Korrekturen durch automatische Bestimmung von Zwangsbedingung möglichst klein zu halten und die aufwendige Modellierung von Muskelkräften zu umgehen [90]. Daneben ermöglicht auch der Einsatz von direkter und inverser Kinematik eine verbesserte Interpolation zwischen Teilbewegungen [7].

Die Extraktion und Bewertung charakteristischer Merkmale von Trajektorien unterschiedlicher Bewegungsmuster wurde vor allem auf dem Gebiet der Ganganalyse zur Identifikation von Bewegungen und deren Klassifikation vorangetrieben. Im Unterschied zu den Publikationen über die Bewegungsgenerierung in der Animationstechnik werden hierbei nur eine kleine Anzahl an unterschiedlichen Bewegungen unter Aspekten wie Schrittfrequenz oder Schrittlänge betrachtet. Dafür kommen komplexere Auswertverfahren auf der Basis statistischer Analysen zur Auswertung der Trajektorien zum Einsatz [6].

In einigen Arbeiten zur Modellierung von Bewegungen werden die Trajektorien geeigneter Merkmale als Ausgaben dynamischer Systeme mit unbekanntem stochastischen Eingaben betrachtet. Die Modelle werden einerseits zur Analyse, Erkennung und Vorhersage ausgewählter Bewegungen verwendet. Forschungsschwerpunkt bilden dabei menschliche Gangbewegungen [136], [16]. Andererseits wurden mit Hilfe dieser Modelle Verfahren zur Bewegungssynthese entwickelt [89]. Beide Aufgabenstellungen haben wieder überwiegend ihren Ursprung in der Animationstechnik und finden im Bereich des so genannten Motion-Editings Anwendung [85], [23].

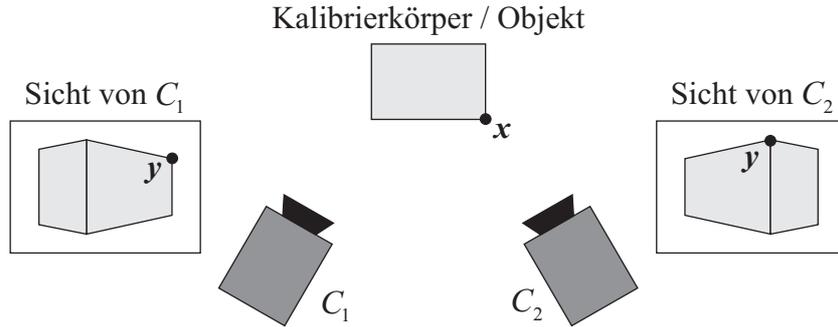
Die Arbeiten zur Generierung von Bewegungen für einen humanoiden Roboter basieren im Wesentlichen auf der Entwicklung von Skelettmodellen mit Gelenken, die sich am Körperbau des Menschen oder am Aufbau des Roboters orientieren. Auf die Freiheitsgrade in den Modellen werden dann mit unterschiedlichsten Verfahren erfasste menschliche Bewegungsdaten oder künstlich erzeugte Trajektorien abgebildet [173]. Die Möglichkeiten hierzu gehen von der Imitation vorgemachter Bewegungen [70], über das Programmieren durch Vormachen [178] bis hin zum motorischen Lernen [140]. In [155], [33] wird beschrieben, wie bei den Abbildungen der Bewegungstrajektorien auf die Freiheitsgrade des Roboters besonders Limitierungen der Gelenkwinkelbereiche, der Winkelgeschwindigkeiten und dynamische Rahmenbedingungen, wie z. B. die Einhaltung der Balance, berücksichtigt werden.

Zu der schwierigen Aufgabe der Interaktion des Menschen mit einem humanoiden Roboter existieren weit weniger Lösungsmethoden. Bei dieser Aufgabenstellung müssen die Trajektorien des Roboters an die des Menschen angepasst werden [174]. Insbesondere ist auch eine Bewertung von Bewegungstrajektorien erforderlich, mit der entschieden werden kann, wie gut eine Bewegungstrajektorie des Roboters zu einer auszuführenden Handlung passt. Hidden-Markov-Modelle haben sich als leistungsfähiges Analysewerkzeug zu diesem Zweck erwiesen [9], [71] erwiesen.

Zumeist beruhen die Analysen mit HMMs auf einer Auswertung mit dem Majoritätsprinzip (siehe Abschnitt 2.3.3.2). Erweiterungen einfacher HMMs [113], wie z. B. gekoppelte oder faktorielle bzw. parallele HMMs, werden zunehmend in der Erkennung von Gesten [162], [162] und Zeichensprache [149], [150] oder auch zum Lippenlesen [175] eingesetzt. In [17], [18] wird gezeigt, wie komplexe Aktionen mit Hilfe von HMMs und einer probabilistischen Grammatik, welche die Zusammenhänge der HMMs auf einer höheren Beschreibungsebene definiert, erkannt werden können. Die angewandten Methoden sind auch aus der Spracherkennung im Zusammenhang mit Wahrscheinlichkeiten für  $n$ -Gramme und stochastischen kontextfreien Grammatiken bekannt [153]. Die Rolle elementarer Teilm Modelle in einem komplexen HMM und die explizite Nutzung der versteckten Zustände, die wesentlicher Gegenstand dieser Arbeit sind, werden jedoch nicht berücksichtigt.

### 4.3 Methoden der Datenakquisition

Die Studie von menschlichen Bewegungen in Hinblick auf die Extraktion charakteristischer Bewegungsmerkmale und einer leistungsfähigen Analyse der Bewegungen erfordert die Akquisition kinematischer Daten von Bewegungssequenzen. Bei diesen Bewegungsmessungen werden üblicherweise Trajektorien von signifikanten Körperpunkten (z. B. geschätzte Gelenkzentren) oder Winkeln an Gelenken gemessen. Die verfügbaren Messmethoden unterscheiden sich stark in ihrer Anwendbarkeit, in der zeitlichen und räumlichen Auflösung, im Zeitaufwand für den Versuchsaufbau und dem Aufwand für die Datenaufbereitung. Nicht zu vernachlässigen ist auch die Einschränkung des Bewegungsspielraum des Probanden. Zum Beispiel ist der Bewegungsspielraum durch den Messbereich der Sensoren beschränkt oder die Bewegungen werden durch Kabel und Befestigungen behindert.



**Abbildung 4.1:** Aufbau zur Bestimmung von Objektkoordinaten  $\mathbf{x} \in \mathbb{R}^3$  aus Bildkoordinaten  $\mathbf{y} \in \mathbb{R}^2$  in verschiedenen Kameransichten zweier Kameras  $C_1$  und  $C_2$  mit Hilfe der DLT.

Im Folgenden werden unterschiedliche Verfahren der Bewegungserfassung vorgestellt, die zur Gewinnung des Datenmaterials in dieser Arbeit verwendet wurden. Es handelt sich dabei um videobasierte und magnetische Bewegungserfassungssysteme, sowie um die Erfassung der feingliedrigen Bewegung der Hand mit einem Datenhandschuh. Die jeweiligen Verfahren sind auch in [11], [12] und [13] dargestellt.

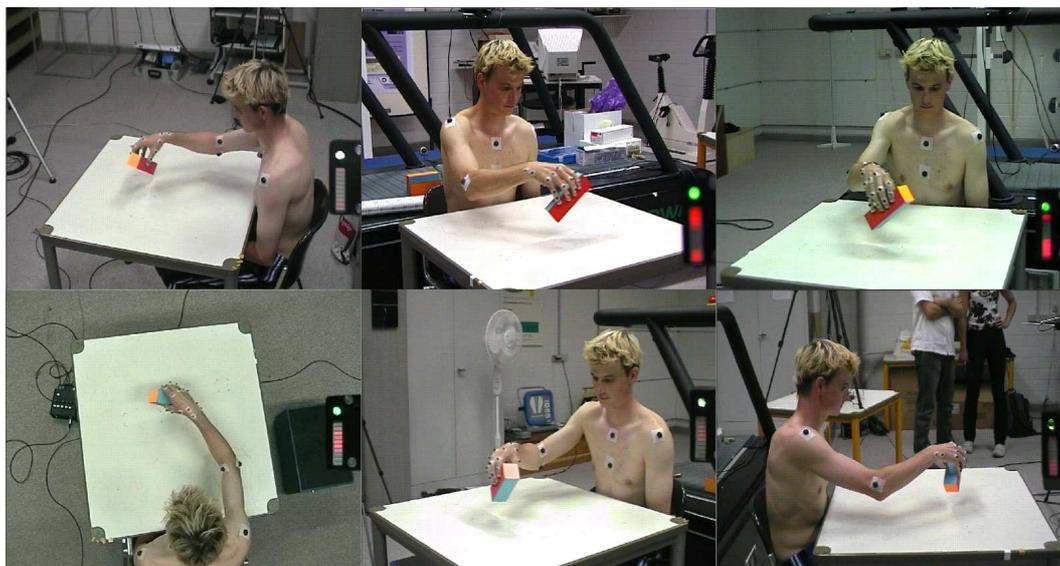
### 4.3.1 Videobasierte Bewegungserfassung

In vielen empirischen Studien werden videobasierte Bewegungserfassungssysteme zur kinematischen Analyse von komplexen Mehrkörpersystemen eingesetzt. Beispiele hierfür sind Aufzeichnung von Dummy-Bewegungen bei Crashtests [60] oder die Analyse von Bewegungsabläufen im Wettkampfsport [61]. Relevante Punkte werden dabei am Probanden mit *optischen Markern* gekennzeichnet und aus verschiedenen Kameransichten aufgenommen. Mit Hilfe der *Direct-Linear-Transformation (DLT) Technique* (siehe z. B. [4]) können aus den 2D Positionen der Markerpunkte in den unterschiedlichen Kameransichten die ursprünglichen 3D Koordinaten rekonstruiert werden. Hierzu sind mindestens zwei Kameransichten notwendig. In der Regel werden aber bis zu 8 Kameras verwendet um durch redundante Punktinformation das Rekonstruktionsergebnis zu optimieren. Abbildung 4.1 zeigt schematisch den Aufbau eines 2-Kamera-Systems, bei dem mit Hilfe der DLT die 3D Koordinaten eines Objektes bestimmt werden sollen.

Die Berechnungen von 3D Koordinaten mit Hilfe der DLT-Methode beruht auf Rekonstruktionsgleichungen der Form

$$\begin{aligned} y_1 + \delta y_1 &= \frac{L_1 x_1 + L_2 x_2 + L_3 x_3 + L_4}{L_9 x_1 + L_{10} x_2 + L_{11} x_3 + 1}, \\ y_2 + \delta y_2 &= \frac{L_5 x_1 + L_6 x_2 + L_7 x_3 + L_8}{L_9 x_1 + L_{10} x_2 + L_{11} x_3 + 1}, \end{aligned} \quad (4.1)$$

wobei der Vektor  $\mathbf{y} \in \mathbb{R}^2$  die Bildkoordinaten eines Objektpunktes  $\mathbf{x} \in \mathbb{R}^3$  für eine Kameransicht darstellt. Die Größen  $L_1, \dots, L_{11}$  beschreiben Kameraparameter, wie Position,



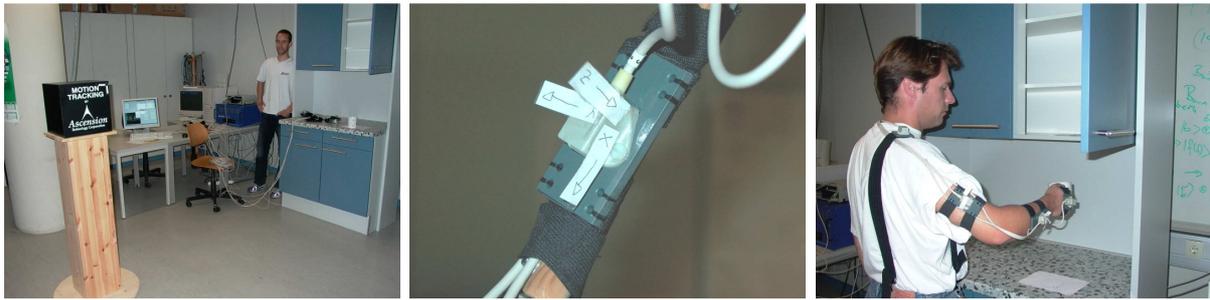
**Abbildung 4.2:** Proband bei der Manipulation eines Objektes in sechs verschiedenen Kamerasichten. Die Lauflichtleisten am jeweiligen rechten Rand der Bilder dienen der Synchronisation der Kameras.

Orientierung und optische Eigenschaften. Um die unbekannt Parameter  $L_1, \dots, L_{11}$  zu ermitteln, wird ein *Kalibrierkörper* mit bekannter Position und Bemaßung verwendet, so dass bekannte Koordinatenpaare  $(\mathbf{x}, \mathbf{y})$  in Gleichung (4.1) eingesetzt werden können. Dies wird für alle Kameras durchgeführt und das entstehende Gleichungssystem z. B. mit Hilfe einer Singulärwertzerlegung gelöst. Die Koordinatenkorrekturen  $\delta\mathbf{y}$  geben den Abbildungsfehler an, der zwischen unterschiedlichen Kamerasichten auftreten kann. Bei bekannten Parametern  $L_1, \dots, L_{11}$  können sodann die 3D Koordinaten  $\mathbf{x}$  eines Punktes aus seinen 2D Koordinaten  $\mathbf{y}$  in den Kamerasichten über Gleichung (4.1) bestimmt werden. Weiterführende Erläuterungen zur DLT-Methode sind z. B. in [4] zu finden.

Die maximale Sample-Rate der videobasierten Bewegungserfassung hängt von der zeitlichen Auflösung der Kameras ab. Diese beträgt bei handelsüblichen Video-Kameras 25–50 Hz. Da die Kameras keine Einrichtung zur elektronischen Synchronisation besitzen, muss ein optisches Signal zum Abgleich der verschiedenen Kamerasichten in das Videobild eingeblendet werden. Die Kameras für die Bewegungsaufnahmen in dieser Arbeit wurden mit Hilfe von LED-Lauflichtleisten synchronisiert. Die LEDs auf den Leisten werden mit einer Frequenz von 500 Hz weitergeschaltet, so dass die zeitliche Verzögerung zwischen den verwendeten Kameras mit einer Auflösung bis zu  $2 \cdot 10^{-3}$  s bestimmt werden kann. Abbildung 4.2 zeigt einen Probanden bei einer Objektmanipulation aus 6 verschiedenen Kamerasichten. Am rechten Rand der Kamerasichten ist jeweils die Lauflichtleiste mit dem optischen Synchronisationssignal zu sehen.

Häufigste Fehlerquelle bei der videobasierten Bewegungserfassung ist die Verdeckung von Markerpunkten. Selbst bei einer großen Anzahl von Kameras sind nicht alle Markerpunk-





**Abbildung 4.4:** Magnetisches Tracking. Von links nach rechts: Versuchsaufbau, Sensor mit Orientierung, Proband mit Markern bei der Versuchsdurchführung.

deobasierten Bewegungserfassung können mit Hilfe von magnetischen Tracking-Systemen umgangen werden. Bei diesen Systemen wird mit einem Feldgenerator durch Wechselströme oder gepulste Gleichströme ein moduliertes Magnetfeld generiert. Mit Magnetfeldsensoren, die als Marker fungieren, können aus den modulierten Information die Position und zusätzlich die Orientierung der Marker im Magnetfeld bestimmt werden. Eine vergleichbare Orientierungsinformation lässt sich mit Hilfe der videobasierten Verfahren nur schlecht gewinnen. Hierzu müssen durch zusätzliche Marker lokale Koordinatensysteme aufgespannt werden, die in Bezug zu einem globalen System gesetzt werden müssen. Eine entsprechende Platzierung der Marker ist allerdings wie im vorigen Abschnitt bereits beschrieben nur schlecht möglich und die resultierenden Ergebnisse sind wenig verwertbar.

Abbildung 4.4 zeigt ein magnetisches Bewegungserfassungssystem. Das am IAKS vorhandene System Motion-Star der Fa. Ascension erlaubt Messungen mit 30–140 Hz. Somit ist eine Erfassung von Bewegungen in Echtzeit möglich. Das System kann mit maximal 18 Sensoren betrieben werden. Die Sensoren besitzen eine Ortsauflösung von 0,76 mm bei einem Feldradius von 1,5 m und 2,54 mm bei 3,5 m Radius. Die entsprechende Winkelauflösung der Orientierung beträgt  $0,1^\circ$  bzw.  $0,2^\circ$ . Die angegebenen Feldradien geben die nach Spezifikation maximalen Messbereiche des Feldgenerators mit einfacher bzw. erweiterter Reichweite an.

Neben den genannten Vorteilen bringt die magnetische Bewegungserfassung auch einige Nachteile mit sich. Wie bereits angegeben ist die Anzahl der Marker begrenzt. Sie sind mit einer Abmessung von  $2,45 \text{ mm} \times 2,54 \text{ mm} \times 0,8 \text{ mm}$  im Vergleich zu optischen Markern relativ groß und lassen sich daher zur Erfassung von feingliedrigen Bewegungen, z. B. Fingerbewegungen, nicht verwenden. Weiterhin sind die Sensoren verkabelt, so dass durch die Kabel der Proband bei der Bewegung eingeschränkt werden kann. Der weitaus entscheidendere Nachteil ist allerdings die starke Störung der Messungen durch metallische Gegenstände im Magnetfeld. Im Hinblick auf die Erfassung von Alltagsbewegungen sind dadurch Bewegungen, wie z. B. das Beladen einer Spülmaschine oder das Ausräumen eines Kühlschranks nicht möglich.

### 4.3.2.1 Rekonstruktion von Gelenkpunkten

Zumeist ist eine Fixierung der Marker bei der magnetischen Bewegungserfassung nicht so leicht möglich wie beim videobasierten System. Der Grund dafür ist die Größe und die Masse der Sensoren. Anstelle die Marker auf der Haut aufzukleben werden diese mit Klettbandern befestigt. Dennoch eignet sich das Verfahren deutlich besser zur Erfassung der exakten Gelenkzentren als das videobasierte Verfahren. Grundlage bildet der im Folgenden beschriebene Algorithmus zur Bestimmung von Gelenkzentren, der sich die zusätzliche Information der Markerorientierung zu Nutzen macht.

Der am IAKS entwickelte Algorithmus zur Bestimmung der Gelenkzentren beruht auf [117] und geht davon aus, dass jedes Körpersegment eine fixe Position bezüglich des lokalen Koordinatensystems eines Sensors besitzt. Das bedeutet, dass die Position eines Gelenkzentrums am Ende eines Segmentes über die Zeit konstant in diesem Koordinatensystem ist. Geht man weiter davon aus, dass zwei Segmente über ein Gelenk an ihren Enden verbunden sind und diese Enden damit in globalen Koordinaten über die Zeit die identische Position haben, so lässt sich die Gleichung

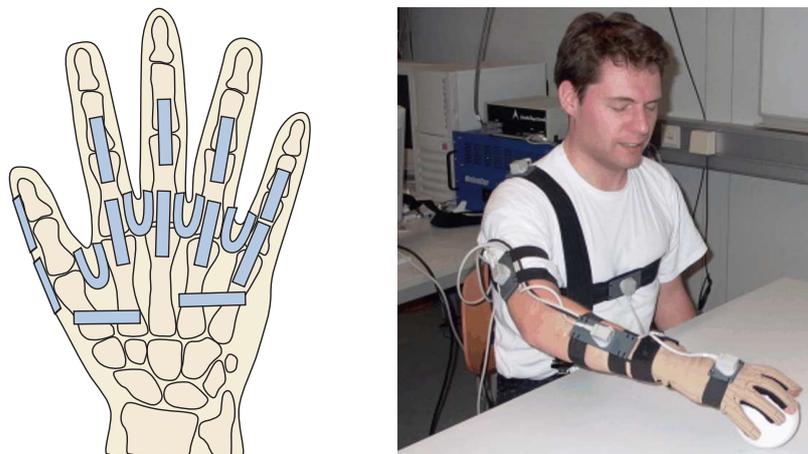
$$\mathbf{R}_{1,t}\mathbf{x}_1 + \mathbf{s}_{1,t} = \mathbf{R}_{1,t}\mathbf{x}_2 + \mathbf{s}_{2,t} \quad (4.2)$$

aufstellen, wobei  $\mathbf{R}_{1,t}, \mathbf{R}_{2,t} \in \text{SO}_3$  und  $\mathbf{s}_{1,t}, \mathbf{s}_{2,t} \in \mathbb{R}^3$  die Rotationsmatrizen und Translationsvektoren zum Zeitpunkt  $t$  sind, welche lokal konstante Koordinaten  $\mathbf{x}_1$  und  $\mathbf{x}_2$  in globale Koordinaten überführen. Durch Umformung von Gleichung (4.2) erhält man das Gleichungssystem

$$(\mathbf{R}_{1,t_1 \dots t_n} \quad \mathbf{R}_{2,t_1 \dots t_n}) \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{pmatrix} = -(\mathbf{s}_{1,t_1 \dots t_T} - \mathbf{s}_{2,t_1 \dots t_T}) \quad (4.3)$$

für  $T$  Zeitpunkte  $t_1, \dots, t_T$  mit  $3T$  linearen Gleichungen und 6 Unbekannten. Wurde die Bewegung nicht über alle Freiheitsgrade (*DOF*, engl. *degree of freedom*) der beteiligten Gelenke durchgeführt, so ist das Gleichungssystem unterbestimmt. Im Allgemeinen ist das Gleichungssystem jedoch für  $T \geq 3$  überbestimmt. Hinzu kommt, dass Messungenauigkeiten vorliegen und das verwendete Modell der Gelenkzentren im menschlichen Körper nur eine Idealisierung darstellen. Eine numerisch stabile Näherungslösung für das Gleichungssystem (4.3) mit Minimierung des mittleren quadratischen Fehlers lässt sich mit Hilfe der Singulärwertzerlegung lösen (siehe [123]). Aufgrund der Tatsache, dass die Gelenkzentren nicht direkt, sondern über dieses Optimierungsverfahren indirekt aus den Markerinformationen berechnet werden, ergibt sich, dass die rekonstruierten Gelenkzentren wesentlich geringer durch Verschiebung der Markerpunkte auf der Haut gestört werden (vgl. Abschnitt 4.3.1).

An dieser Stelle sei bemerkt, dass der beschriebene Algorithmus auch zur Bestimmung der Segmente eines Körpermodells verwendet werden kann. Diese Funktionalität wurde jedoch im Rahmen dieser Arbeit nicht verwendet, so dass an dieser Stelle lediglich auf die Beschreibung in [13] verwiesen wird.



**Abbildung 4.5:** Links: Positionen der Sensoren (hellblau) im Datenhandschuh zur Erfassung der Gelenkwinkel der Hand. Rechts: Einsatz des Datenhandschuhs in Verbindung mit Magnetsensoren zur Erfassung der Gelenkwinkel von Hand und Arm.

### 4.3.3 Datenhandschuh

Zur Messung feingliedriger Bewegungen sind die bisher beschriebenen Methoden nur bedingt geeignet. Zur Aufnahme von Fingerbewegungen werden daher andere Verfahren verwendet. In diesem Bereich haben sich Messungen mit Datenhandschuhen etabliert. Für Aufnahmen der Bewegungen der Hand wurde in dieser Arbeit ein CyberGlove der Fa. Immersion verwendet. Dieser Datenhandschuh besitzt 18 DOFs zur Messung der Gelenkwinkel bei Bewegungen der Finger und des Handgelenks. Erfasst werden Flexion und Extension der ersten zwei Gelenke der Finger und deren Abduktion. Abbildung 4.5 zeigt die Positionen der Sensoren im Datenhandschuh zur Erfassung der Gelenkwinkel der Hand. Die distalen Fingergelenke werden nicht erfasst und müssen aus der Stellung des zweiten Gelenkes geschätzt werden. In [84] und [87] wird dazu das Verhältnis zwischen den Gelenkwinkeln  $\theta_{\text{dist}}$  und  $\theta_{\text{prox}}$  empirisch ermittelt und in dieser Arbeit die Berechnung mit  $\theta_{\text{dist}} := \frac{2}{3}\theta_{\text{prox}}$  durchgeführt, wobei bei gestrecktem Finger  $\theta_{\text{dist}} = \theta_{\text{prox}} = 0$  gilt. Ebenso wie die DOFs der Finger wird auch die Flexion und Extension des Daumens, dessen Rotation und Abduktion gemessen. Zusätzliche Größen sind die zwei Freiheitsgrade des Handgelenks sowie die Krümmung des Handrückens. Die maximale Sample-Rate für alle vom Datenhandschuh erfassten 18 DOFs beträgt 150 Hz.

Die Messung der Gelenkwinkel erfolgt durch Widerstandsveränderungen an *Dehnmessstreifen*, die in den Datenhandschuh eingearbeitet sind. Sie besitzen ein lineares Übertragungsverhalten von Krümmungswinkeln zu Spannungswerten und weisen eine konstante Empfindlichkeit über die gesamte Länge des Messstreifens auf. Probleme sind jedoch starke Temperaturabhängigkeit und Luftfeuchtigkeit. Dies erfordert gelegentliches Nachkalibrieren während der Bewegungsdurchführung. Gerade die Kalibrierung stellt ein schwer-

wiegendes Problem dar, da die analogen Spannungswerte in digitale Werte mit 8 Bit mit einer Maximalen Auflösung von  $0,5^\circ$  gewandelt werden, damit nicht die Winkel widerspiegeln und erst über eine lineare Transferfunktion berechnet werden müssen. Dabei ist auch zu beachten, dass die DOFs des Handschuhs nicht genau denen der menschlichen Gelenke entsprechen. Beispielsweise beeinflusst die Flexion des Zeigefingers sowohl den Sensor zur Messung der Flexion als auch den zur Messung der Abduktion. Eine aufwendigere Kalibrierung ist daher ebenfalls notwendig. Sie ist jedoch nicht Gegenstand dieser Arbeit (für Details siehe [47]).

Da nach der Transformation der Messwerte ausschließlich Winkeldaten für die Gelenke der Hand vorliegen, ist zusätzlich eine Rekonstruktion der Gelenkpunkte über eine direkte Kinematik notwendig, wenn Raumkoordinaten für die Punkte der Hand benötigt werden. Hierzu wurde in dieser Arbeit ein kinematisches Modell der Hand entwickelt. Die Längen der Segmente und die Achsen für die Freiheitsgrade der Gelenke ergeben sich z. T. aus Vorgaben der Literatur [127] und eigenen Messungen (siehe Tab. 4.1). Um die globale Position und Orientierung der Hand zu erhalten sind zusätzlich die lokalen Koordinaten aus dem kinematischen Modell der Hand in ein übergeordnetes Modell einzubetten. Hierzu ist beim Datenhandschuh die Fixierung eines Sensors für magnetisches Tracking am Unterarm vorgesehen. Da die Sensoren für die Bewegung des Handgelenkes sehr ungenau sind, ist die Platzierung eines magnetischen Sensors auf dem Handrücken sinnvoller (siehe Abb. 4.5).

	Mittelhand- knochen	Phalanx proximalis	Phalanx media	Phalanx distalis
Daumen	50	30	—	25
Zeigefinger	75	40	30	15
Mittelfinger	80	45	30	20
Ringfinger	70	43	30	20
kleiner Finger	60	35	20	15

**Tabelle 4.1:** Das verwendete Handmodell: Längen der Fingersegmente in mm.

#### 4.3.4 Weitere Verfahren

Neben den in dieser Arbeit verwendeten Verfahren zur Erfassung von Bewegungen gibt es eine Vielzahl weiterer Methoden, die hier aber nur der Vollständigkeit wegen genannt werden. Als Alternative zur videobasierten Markererfassung bieten sich Systeme mit Infrarot-Kameras an. Die Probanden werden bei diesen Systemen mit passiven Markern versehen, die infrarotes Licht sehr gut reflektieren. Die Kameras liefern weitestgehend nur Bilder der Markerpunkte, die sich vollautomatisch und wesentlich robuster erfassen lassen. Der Nachteil der Systeme liegt im hohen Anschaffungspreis. Außerdem müssen Videos, z. B. zur Dokumentation des Versuchs, zusätzlich aufgenommen werden.

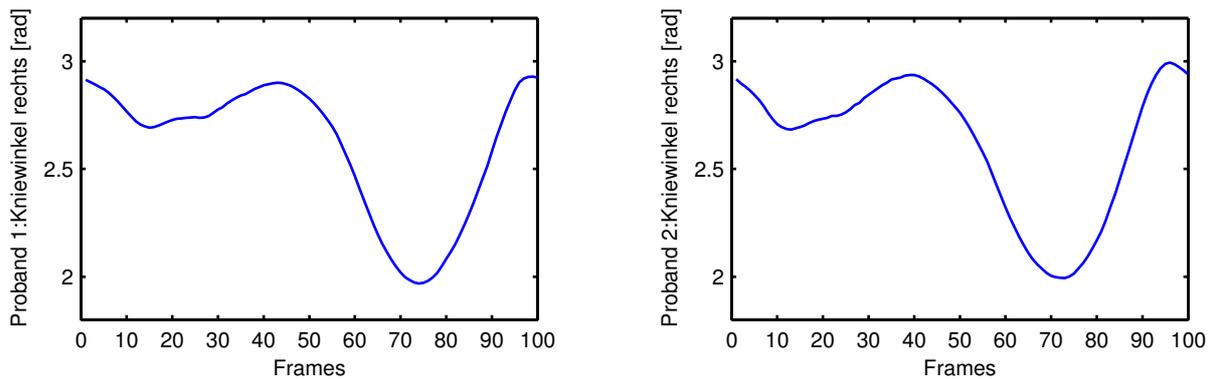
Dehnmessstreifen, wie sie beim Datenhandschuh zur Messung der Gelenkwinkel verwendet werden, können auch für andere Gelenke des Körpers eingesetzt werden. Die Verfahren sind jedoch weniger verbreitet, da sie neben den Nachteilen, die in Abschnitt 4.3.3 beschrieben sind, auch das Problem der Fixierung besitzen. Das Gleiche gilt für mechanische Winkelmesser (*Goniometer*), die den Bewegungsraum der Probanden zumeist zusätzlich einschränken.

Weitere Möglichkeiten zur Erfassung von Bewegungen bestehen in der Messung von aufgewendeten Kräften, die jedoch hauptsächlich für dynamische Bewegungsmodelle von Bedeutung sind. Auch die direkte Messung der Muskelaktivität durch Elektromyogramme (EMG) ist in der Medizin und Sportwissenschaft verbreitet. Hierbei werden die Erregungspotentiale der Muskeln erfasst. Da die Methoden in dieser Arbeit auf rein kinematischen Modellen beruhen, wurde auf eine Erweiterung der Bewegungsmerkmale durch die beschriebenen Größen jedoch verzichtet.

## 4.4 Merkmale zur Beschreibung von Bewegungen

Die Extraktion charakteristischer Bewegungsmerkmale ist eine der Schlüsselaufgaben bei der genauen Analyse und Klassifikation von menschlichen Bewegungen. Geeignete Merkmale müssen die Unterscheidung inter-individueller Charakteristika in Bewegungssignalen ermöglichen. Diese Unterscheidung ist notwendig für die Erkennung von Personen und die Unterteilung in Bewegungsphasen. Gleichzeitig müssen die Merkmale invariant d.h. unempfindlich gegenüber intra-individuellen Variationen der Bewegung sein. Ebenso ist darauf zu achten, dass die Merkmale stabil bezüglich der Position eines beobachteten Probanden oder der manipulierten Objekte bleibt und geringfügige Veränderungen der Rahmenbedingungen oder Abweichungen von der gegebenen Aufgabenstellung keinen Einfluss auf die Bewegungsanalyse haben. Die Generierung von eindeutigen und gleichzeitig robusten Korrespondenzen zwischen Merkmalswerten und Bewegungscharakteristika ist daher die wesentliche Anforderungen bei allen Fragestellungen der Bewegungsanalyse [151].

Die Darstellung von Bewegungen durch Trajektorien absoluter Positionen z. B. von Objekt- oder Gelenkkoordinaten ist nicht geeignet für die Generierung von eindeutigen Korrespondenzen zwischen Merkmalen und Bewegungscharakteristika. Räumliche Translationen würden beispielsweise die Merkmale unbrauchbar machen. Relative Positionen, z. B. die Position des Fußes relativ zur Hüfte, können hilfreich für die Bewegungsanalyse unter biomechanischen Gesichtspunkten sein. Sie hängen aber stark von den anatomischen Gegebenheiten ab. Gelenkwinkel hingegen hängen weniger stark von Veränderungen in der Anatomie ab und bilden daher einen guten Ansatz zur Analyse von Bewegungssignalen [166].



**Abbildung 4.6:** Winkelverläufe für das rechte Kniegelenk bei der Gehbewegung zweier Probanden (jeweils ein Gangzyklus, normiert auf 100 Frames).

#### 4.4.1 Darstellung von Bewegungen durch Winkeltrajektorien

Durch die Darstellung mittels Gelenkwinkelverläufen werden Manipulationen auf den Daten erheblich vereinfacht. Es sind nur wenige Nebenbedingungen, wie z. B. maximale Streck- oder Beugewinkel und Kollisionsfreiheit, bei Operationen auf den Winkeldaten einzuhalten. Bei den klassischen kinematischen Trajektorien, die aus den Winkeldaten durch direkte Kinematik zu errechnen sind, muss hingegen stets eine Reihe algebraischer Gleichungen, die das anatomische Modell widerspiegeln, erfüllt bleiben. Das macht z. B. die Glättung von Bewegungsdaten in dieser Domäne problematisch. Reine kinematische Daten werden daher vorwiegend zur Bewertung des Bewegungsverhaltens eingesetzt, da sich ergonomische Aspekte, wie z. B. Körperhaltung und Gleichgewichtsbedingungen, mit Hilfe der Winkeldaten nicht unmittelbar bestimmen lassen.

Es wird ersichtlich, dass für die gegebenen Aufgabenstellungen Erweiterungen des klassischen kinematischen Merkmalsraums erforderlich sind. Um die Bewegung umfassend zu beschreiben und analysieren zu können, müssen Trajektorien in diesem erweiterten Merkmalsraum, d. h. zeitliche Abfolgen von Merkmalsvektoren, betrachtet werden. Die Berücksichtigung von Segmentorientierungen ist eine erste derartige Erweiterung. Um individuelle Bewegungscharakteristika zu erfassen sind jedoch weder kinematische Trajektorien noch Gelenkwinkelverläufe ausreichend. Abbildung 4.6 verdeutlicht dies für den Winkelverlauf des rechten Kniegelenks bei der Gehbewegung zweier Probanden. Aus dieser Darstellung lassen sich zunächst die lokalen Extrema der Gelenkwinkel als markante individuelle Merkmale erfassen.

Wie in Abb. 4.6 zu erkennen ist, spiegeln sich individuelle Charakteristika auch in kinematischen Größen wie Winkelgeschwindigkeiten und Beschleunigungswerten wider. Die Erfassung dieser Größen erfolgt in dieser Arbeit mittels Wavelet-Transformationen, die auf den Gelenkwinkelverläufen operieren. Dabei wird eine redundante, translationsinvariante Wavelet-Darstellung verwendet, so dass für jeden Zeitpunkt ein Vektor aus Wavelet-

Koeffizienten generiert wird. Bei geeigneter Wahl der Wavelets stellen die einzelnen Koeffizienten Winkelgeschwindigkeiten auf unterschiedlichen Skalierungsstufen dar.

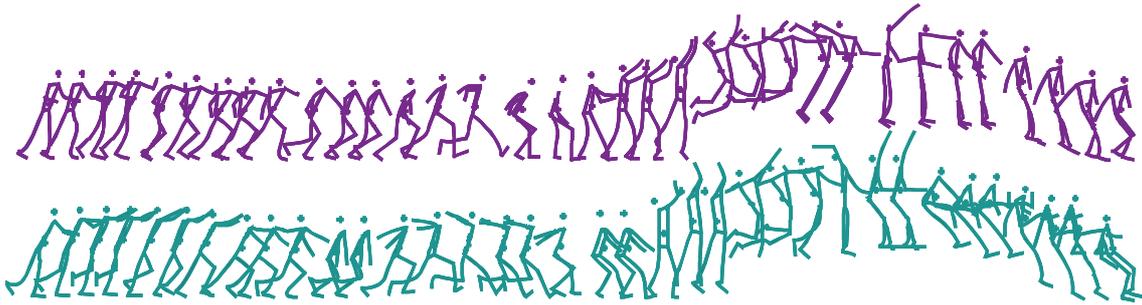
#### 4.4.2 Erweiterte Bewegungsmerkmale durch Wavelets

Die bisher zur Bewegungsanalyse genutzten Merkmale beruhen auf einer direkten Beobachtung der zeitlichen Gelenkwinkelverläufe und der Extraktion von Teilbewegungen durch geeignete Fensterung. Für weitere Analysemöglichkeiten ist es wünschenswert, die Bewegungssignale gleichzeitig mit unterschiedlichen Fensterbreiten zu untersuchen. Diese Idee deckt sich auch mit der natürlichen Herangehensweise des Menschen bei der Beobachtung und Klassifikation von Bewegungen. Der Mensch unterscheidet sowohl den groben Verlauf einer Bewegung als auch feine Variationen innerhalb kürzerer Bewegungsabschnitte.

Die Signalverarbeitung kennt einen vergleichbaren Ansatz unter dem Stichwort Multiskalen-Verfahren. Ein Signal wird auf unterschiedlichen Approximationsstufen betrachtet. Übliche Verfahren zur Erstellung dieser Approximationsstufen sind Wavelet-Transformationen (siehe Anhang A). In dieser Arbeit wird gezeigt, wie die zusätzliche Nutzung der Winkelinformation auf unterschiedlichen Skalierungsstufen zu einer Verbesserung der Merkmale für die Analyse mit HMMs genutzt werden kann.

Zur Motivation wird die Bewegung zweier Volleyballspieler beim Sprungaufschlag analysiert. Der typische Bewegungsablauf beim Anlauf, Sprung, Schlag und Landung ist für zwei Spieler in Abb. 4.7 anhand von Strichdiagrammen dargestellt. Das Hauptaugenmerk gilt bei diesem Beispiel dem zeitlichen Verlauf des rechten Ellenbogenwinkels des Schlagarmes. In Abb. 4.8 ist im obersten Diagramm der Verlauf der Winkeldaten für die zwei Spieler aufgetragen. Beide Kurven weisen einen ähnlichen Verlauf im mittleren Bereich zwischen den Frames 30–80 auf. Hierbei handelt es sich um die Sprung- und Schlagphase der Bewegung. In den Bereichen davor und danach, der Anlauf- und der Landephase, unterscheiden sich die Bewegungen deutlich. Unter den Originaldaten sind in Abb. 4.8 vier zugehörige Approximationsstufen abgebildet. Diese wurden mit Hilfe einer diskreten Wavelet-Transformation (DWT) mit Haar-Wavelets ohne Unterabtastung erzeugt. In den unterschiedlichen Skalierungen bis zur größten Darstellung bleibt der charakteristische Kurvenverlauf erhalten. Lokale Schwankungen werden mit wachsender Skalierungsstufe zunehmend eliminiert. Somit ist eine Abstraktion auf den wesentlichen Merkmalsverlauf möglich.

Betrachtet man zusätzlich die durch die Wavelet-Transformation berechneten Wavelet-Koeffizienten der unterschiedlichen Skalierungsstufen (siehe Abbildung 4.8 rechte Spalte), so zeichnet sich mit zunehmender Approximation ein deutlicher Peak bei Frame 70 ab. Dieser kennzeichnet den Zeitpunkt des Schlages. Aufgrund der differenzierenden Eigenschaften der Haar-Wavelets (siehe Abschnitt A.4) repräsentieren die Wavelet-Koeffizienten Informationen über die bei der Bewegung vorherrschende Winkelgeschwindigkeit. Gerade beim Schlag zeigt sich in dieser Größe ein deutlicher Peak, da der Spieler versucht, möglichst viel Energie auf den Ball zu übertragen. Der Peak ist für beide Spieler bei



**Abbildung 4.7:** Strichdarstellung zweier Volleyballspieler beim Sprungaufschlag.

zunehmender Approximation deutlicher ausgeprägt und hat über alle Skalierungsstufen nahezu im selben Frame sein Maximum. Die zeitliche Lokalisation des Schlages ist damit über alle Skalierungsstufen in den Wavelet-Koeffizienten invariant. Eine genaue Lokalisation des Zeitpunktes des Schlages in den reinen Winkeldaten ist selbst mit Hilfe der verschiedenen Approximationsstufen nicht mit vergleichbarer Sicherheit möglich. Offensichtlich stellt die Information in den Wavelet-Koeffizienten eine ausgezeichnete Basis für einer Erweiterung der bestehenden Bewegungsmerkmale dar.

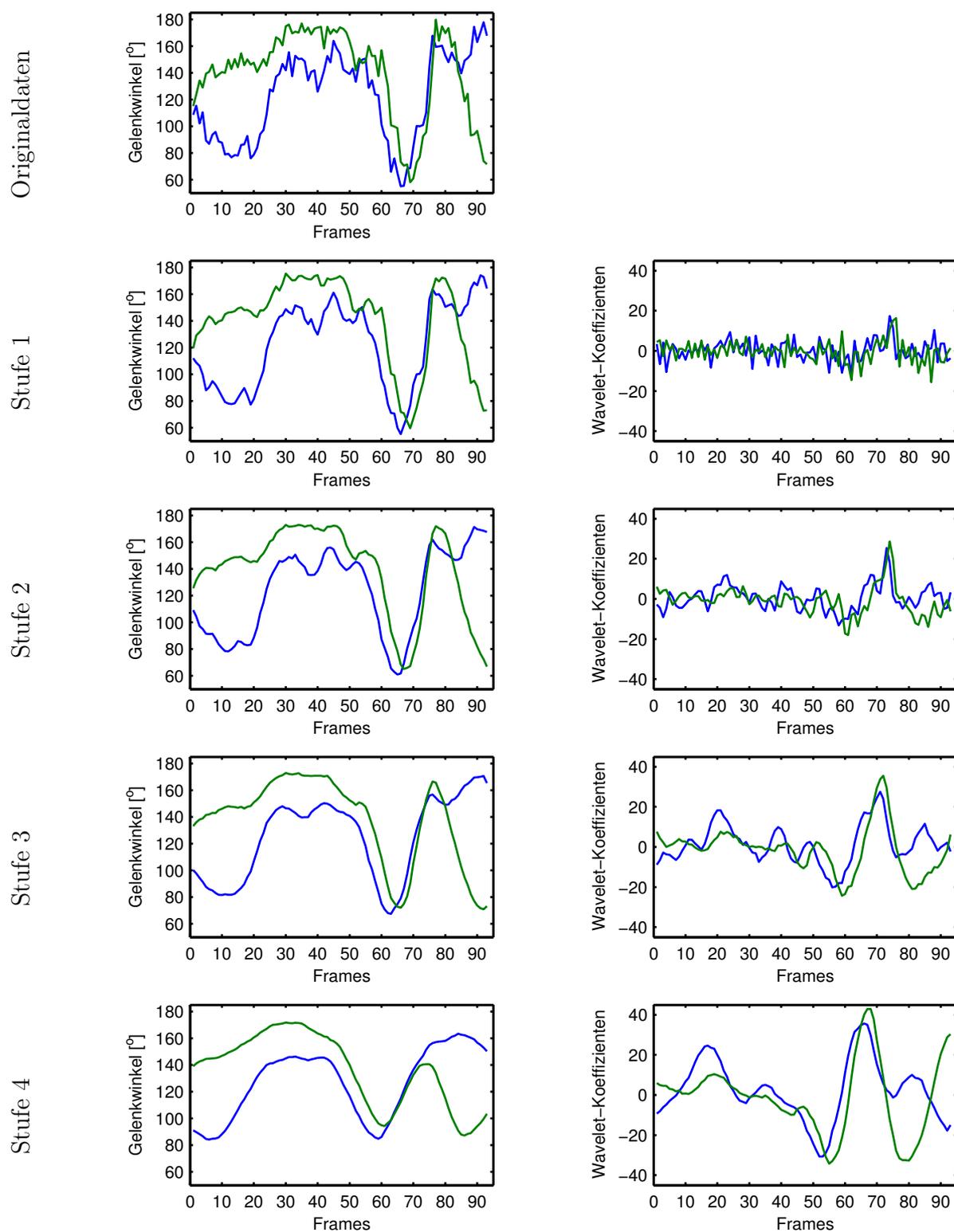
In der Motivation werden für eine geeignete Aufbereitung von Bewegungsmerkmalen Wavelets  $\psi$  verwendet, die durch Ableitung einer Skalierungsfunktion  $\varphi$  entstehen, die wiederum einen Glättungsoperator charakterisiert. Das bedeutet, dass neben den klassischen Anforderungen an Skalierungs- bzw. Wavelet-Funktionen (siehe Abschnitt A.1) die Bedingungen

$$\int_{\mathbb{R}} \varphi(t) dt = 1 \quad \text{und} \quad \psi(t) = \frac{d}{dt} \varphi(t) \quad (4.4)$$

gelten. Aus diesen Eigenschaften folgt, dass die Wavelet-Koeffizienten schrittweise mittels  $\varphi$  geglättete und skalierte Ableitungen der Winkelverlaufskurven darstellen. Als Wavelets werden in dieser Arbeit Haar-Wavelets verwendet. Sie besitzen ein verschwindendes Moment und erfüllen demnach die oben aufgeführten Anforderungen in Verbindung mit den Eigenschaften Orthogonalität und Symmetrie (siehe Abschnitte A.3.2 und A.4). Um die Glättungseigenschaften noch zu verbessern, sind auch biorthogonale Wavelets denkbar. Die Berechnung der Wavelet-Koeffizienten ist für diese Wavelets jedoch mit mehr Aufwand verbunden, so dass gerade in Hinblick auf eine schnelle Signalverarbeitung für echtzeitnahe Anwendung die Verwendung von Haar-Wavelets vorgezogen wird (siehe dazu auch [94]).

### 4.4.3 Analyse und Bewertung von Merkmalstrajektorien

Anhand der Wavelet-Koeffizienten können signifikante Merkmale der Bewegung identifiziert werden. Abbildung 4.9 zeigt die Wavelet-Darstellung (Wavelet-Koeffizienten) für den Verlauf der Kniewinkel während der Gehbewegungen aus Abb. 4.6. Es sind 4 Stufen



**Abbildung 4.8:** Unterschiedliche Skalierungsstufen des rechten Ellenbogenwinkels und zugehörige Wavelet-Koeffizienten für zwei Volleyballspieler beim Sprungaufschlag.

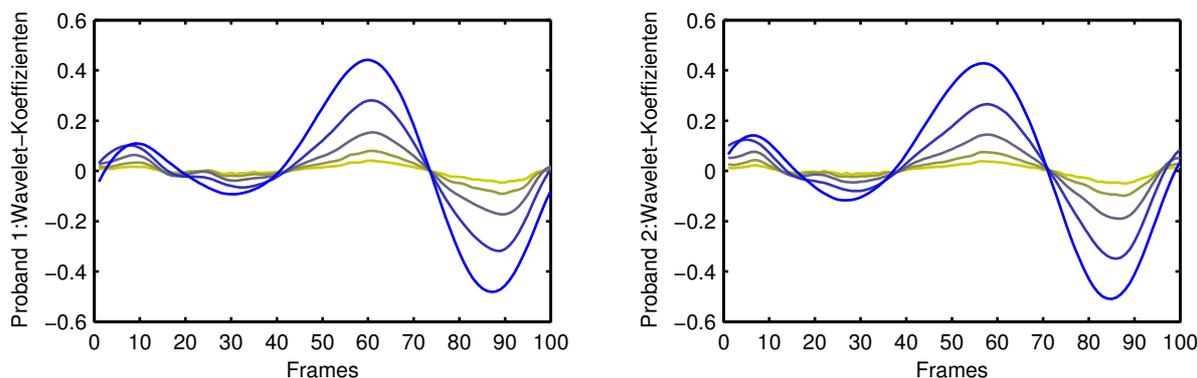
der Wavelet-Zerlegung aufgetragen. Der Gelenkwinkelverlauf selbst entspricht der Approximationsstufe 0 der Wavelet-Zerlegung. Zunächst ist festzuhalten, dass in weiten Bereichen des Bewegungsablaufs ein großer Bewegungsspielraum genutzt wird, in dem sich zwar individuelle Präferenzen bei der Bewegungsausführung widerspiegeln, der jedoch keine wesentliche Rolle bei der Bewertung der Bewegungscharakteristik besitzt. In der Wavelet-Darstellung lässt sich jedoch der Verlauf und damit auch die Charakteristik der Bewegung analysieren. Die signifikanten Merkmale gestatten die Gliederung der Bewegung in Schlüsselabschnitte, die insbesondere für die Erkennung des Bewegungszustandes erforderlich sind. Dies lässt sich anhand von Abb. 4.9 exemplarisch verdeutlichen.

Die einzelnen Bewegungsphasen sind in der Wavelet-Darstellung durch die Nulldurchgänge der Wavelet-Koeffizienten sowie durch die lokalen Minima und Maxima der Koeffizientenkurven charakterisiert. Die Nulldurchgänge repräsentieren die Zeitpunkte der Bewegung, in denen ein Richtungswechsel erfolgt. An den Minimal- bzw. Maximalstellen erfolgt eine Umkehrung des Beschleunigungsverhalten. Beide Kriterien weisen auf den Übergang in eine neue Bewegungsphase hin. Jedoch ist zu berücksichtigen, dass auf den feinen Skalierungsstufen u. a. aufgrund von Rauschen eine zuverlässige Detektion der Bewegungsphasen nicht gewährleistet ist. Hier kommt der Vorteil der Wavelet-Darstellung zum Tragen, dass die jeweils optimale Skalierungsstufe adaptiv ausgewählt werden kann. Beispielsweise sind für die Ermittlung der Nulldurchgänge die Skalierungsstufen 2 und 3 geeignet. Die Extraktion der Minima und Maxima wird stärker durch Rauschen beeinträchtigt und muss daher auf größeren Stufen durchgeführt werden. In Abb. 4.9 lassen sich anhand der Wavelet-Koeffizienten folgende Bewegungsphasen unterscheiden, die auch in der Sportwissenschaft zur Charakterisierung von Gangbewegungen definiert werden:

- *Phase 1*: vordere Stützphase,
- *Phase 2*: Doppelstützphase und hintere Stützphase (diese beiden Phasen sind verschmolzen, da sie sich anhand des Kniewinkels nicht eindeutig unterscheiden lassen),
- *Phase 3*: hintere Schwungphase,
- *Phase 4*: vordere Schwungphase.

Während in den Phasen 3 und 4 ein verhältnismäßig großer Bewegungsspielraum vorhanden ist, der sich in der großen Variabilität der Merkmalswerte widerspiegelt, lassen sich innerhalb der ergonomisch bedeutsamen Phasen 1 und 2 signifikante individuelle Merkmale verdeutlichen. Während der Proband im oberen Diagramm in Abb. 4.9 einen gleichmäßigen Verlauf der Winkelgeschwindigkeit aufweist, zeigt die Wavelet-Darstellung unten ein zusätzliches Maximum bzw. Minimum der Wavelet-Koeffizienten der Skalierungsstufe 2 und 3. Der Proband vollzieht seine Bewegung nicht gleichmäßig, sondern knickt während der Stütz- bzw. Schrittphase erkennbar ein.

Eine weitere Betrachtung der Wavelet-Koeffizienten (siehe auch [13]) ergibt, dass die Positionen der Nulldurchgänge stabil über die unterschiedlichen Skalierungsstufen sind. Daher eignen sich die Nulldurchgänge als robustes Kriterium zur Detektion von Übergängen zwischen Bewegungsphasen. Die Skalierungsstufen 3–5 ergeben den besten Kompromiss

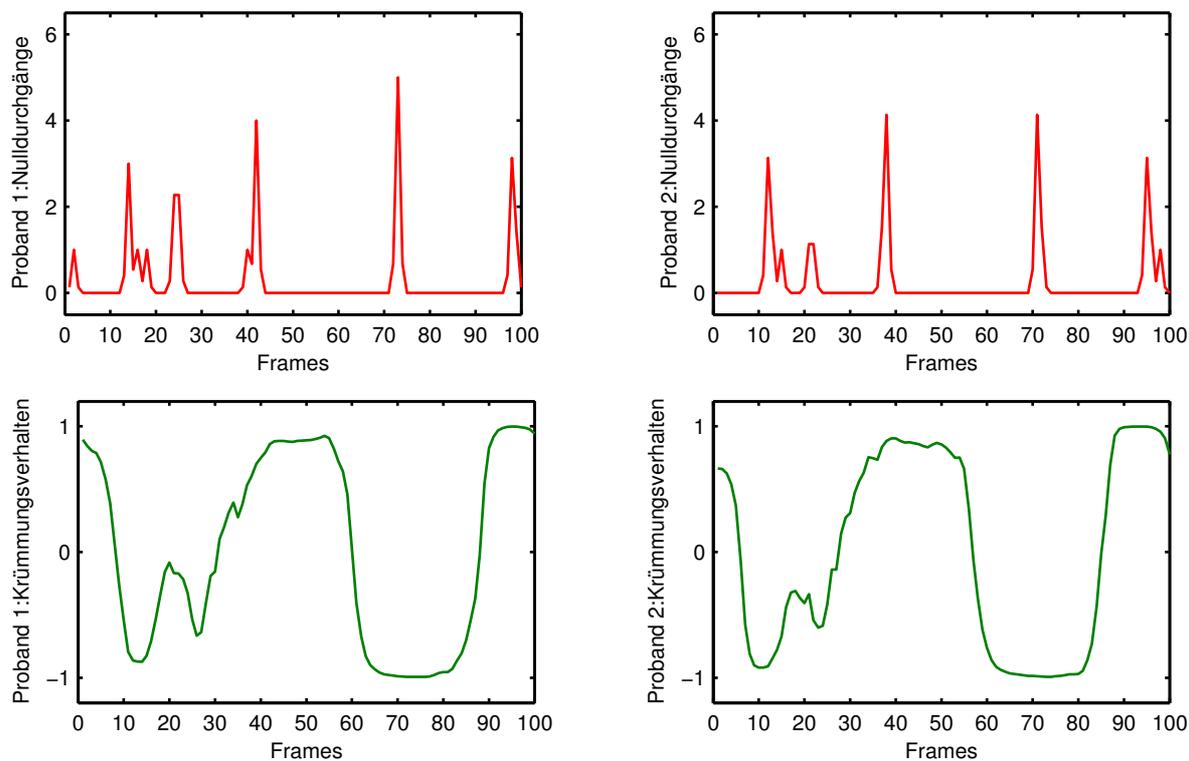


**Abbildung 4.9:** Detailinformationen (Wavelet-Koeffizienten) auf den Skalierungsstufen 1 (gelb) bis 5 (blau) zu den Winkelverläufen aus Abb. 4.6.

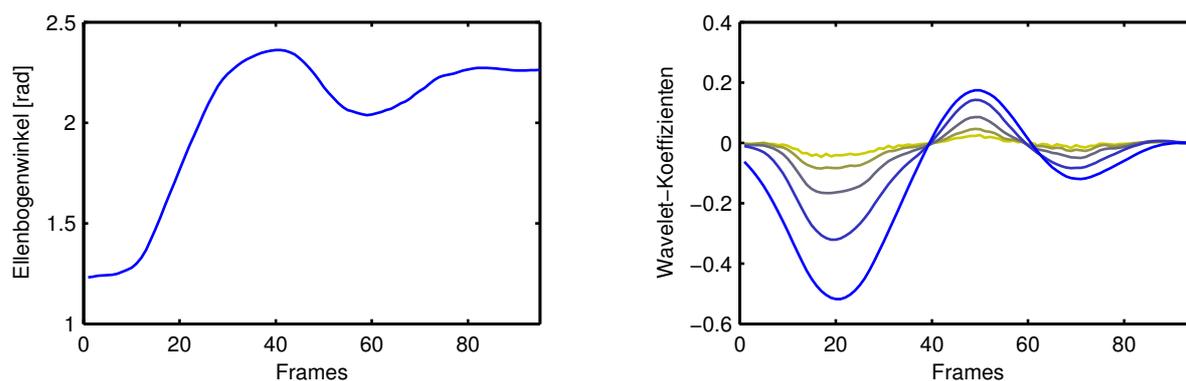
zwischen Fehlertoleranz und Genauigkeit der Positionen. Basierend auf diesen Beobachtungen können durch eine gewichtete Summation von Wavelet-Koeffizienten über alle Skalen Merkmale zur verlässlichen Detektion von Phasenübergängen generiert werden (vgl. Abb. 4.10). Zusätzlich dient das Vorzeichen des Krümmungsverhaltens der Winkelkurven als Merkmal zur verbesserten Unterscheidung von Bewegungsphasen, da benachbarte Phasen unterschiedliche Vorzeichen in der lokalen Umgebung der Phasenübergänge besitzen. Eine effiziente Berechnung der Krümmung erfolgt bei Verwendung der beschriebenen Wavelet-Funktionen direkt aus den Wavelet-Koeffizienten. Eine Berechnung der Gradienten der Wavelet-Koeffizienten realisiert diese Aufgabe. Eine Berechnung der gewichteten Summe ist auch in diesem Fall angebracht, da die Berechnung des Krümmungsverhaltens einem Ableitungsoperator zweiter Ordnung entspricht und somit stark durch Rauschen beeinträchtigt werden kann. Zusätzlich wird die sigmoidale Transferfunktion (B.6) aus Anhang B als glatte Approximation der Vorzeichenfunktion verwendet, um die Detektion von Übergängen robuster zu gestalten. Abbildung 4.10 zeigt die berechneten Merkmale zur Unterscheidung der Bewegungsphasen.

Die berechneten Merkmale dienen in erster Linie zur lokalen Unterscheidung von Bewegungsphasen. Die Lokalisation ist nicht nur bei Gangbewegungen möglich, sondern kann auch für beliebige andere Bewegungen, z. B. aus dem Alltag, eingesetzt werden. Abbildung 4.11 zeigt dazu exemplarisch den Gelenkwinkelverlauf des Ellenbogens bei einer Greif- und Verschiebewegung mit den zugehörigen Skalierungsstufen der Wavelet-Koeffizienten.

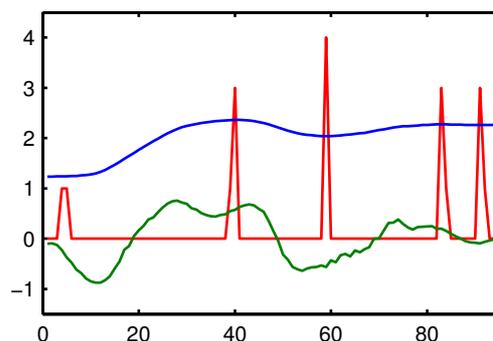
Analog zur Gangbewegung sind in Abb. 4.12 die erweiterten Merkmale aufgetragen, mit denen die Übergänge zwischen einzelnen Bewegungsphasen deutlich erkannt werden können. In Kombination mit anderen Bewegungsmerkmalen, wie z. B. Gelenkwinkeln, dienen diese Merkmale zur Spezifikation allgemeiner Charakteristika von Bewegungsphasen. Die im folgenden Abschnitt entwickelten Bewegungsmodelle berücksichtigen diese Eigenschaft, indem in den Modelle sowohl Zustände für Bewegungsphasen als auch für Übergänge



**Abbildung 4.10:** Extrahierte Merkmale zur Phasenbestimmung (obere Zeile: Information über Nulldurchgänge, untere Zeile: Krümmungsverhalten) zu den Winkelverläufen aus Abb. 4.6.



**Abbildung 4.11:** Winkelverläufe des Ellenbogens (links) bei einer Greif- und Verschiebewegung mit zugehörigen Wavelet-Koeffizienten (rechts) auf den Skalierungsstufen 1 (gelb) bis 5 (blau).



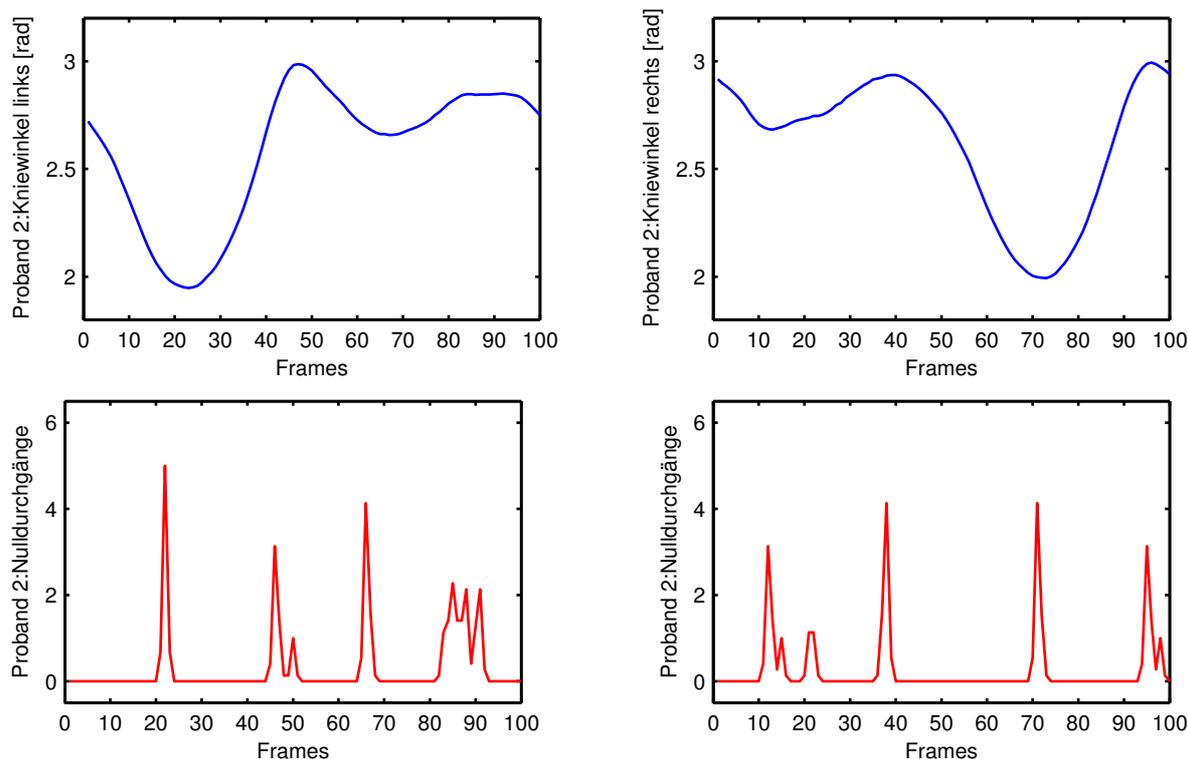
**Abbildung 4.12:** Extrahierte Merkmale in Form der Information über Nulldurchgänge (rot) und des Krümmungsverhaltens (grün) zur Bestimmung der Bewegungsphasen des Winkelverlaufs (blau) aus Abb. 4.11.

zwischen den Phasen vorhanden sind, die anhand der Gelenkwinkel und der Wavelet-Informationen identifiziert werden können.

#### 4.4.4 Bedeutung für Sport und Medizin

Aus der Sicht der Medizin ist die Bewegungsanalyse ein Werkzeug zur objektiven Bewertung von Bewegungen. Eine derartige Bewertung wird zur Erkennung und Behandlung von Erkrankungen des Bewegungsapparates benötigt. Beispiele hierfür sind die Planung von chirurgischen Eingriffen, bei der die natürliche Funktionalität des Bewegungsapparates möglichst erhalten oder wieder hergestellt werden soll z. B. bei der Implantation künstlicher Gelenke, bei der Rehabilitation zur Verlaufskontrolle der Behandlung bei motorischen Schädigungen des Patienten oder auch zur Fertigung von Prothesen, die auf die individuellen Bewegungscharakteristika eines Menschen abgestimmt werden müssen. In der Sportwissenschaft findet die automatische Analyse von Bewegungen ebenfalls zunehmend Verwendung. Wichtigstes Aufgabenfeld ist hierbei die Optimierung bestimmter Bewegungsabläufe oder das Training von Sportlern [46].

Die in dieser Arbeit vorgestellten Verfahren wurden noch nicht in klinischen oder sportwissenschaftlichen Versuchen untersucht. Jedoch konnte bereits anhand der in Abschnitt 4.6 untersuchten Gangbewegungen die Relevanz der entwickelten Verfahren zur Merkmalsgewinnung, z. B. für orthopädische Fragestellungen belegt werden. Bei der Erzeugung der charakteristischen Bewegungsmerkmale fielen bei einigen Datensätzen Anomalien auf. Als Beispiel sei hier die Gangbewegung einer Probandin vorgestellt. Abbildung 4.13 zeigt den zeitlichen Verlauf des linken und des rechten Kniewinkels. In den Winkelverläufen unterscheiden sich die Diagramme für das linke und das rechte Knie abgesehen von der Phasenverschiebung kaum. Zusätzlich sind die generierten Bewegungsmerkmale zur Erkennung der Bewegungsphasen aufgetragen. Es handelt sich dabei um die Nulldurchgänge



**Abbildung 4.13:** Erkennung einer unergonomischen Gehbewegung mit Hilfe der erweiterten Merkmale. Verglichen wird das linke Bein mit dem rechten Bein (linke und rechte Spalte der Abbildung). Oben: Das Einknicken des linken Sprunggelenks ist anhand der Kniewinkeltrajektorien nicht zu erkennen. Unten: die Häufigkeiten der Nulldurchgänge in den Kurven der Wavelet-Koeffizienten machen die Anomalie durch den fehlenden ausgeprägten Peak sichtbar.

in den Kurven der Wavelet-Koeffizienten. Deutlich ist hier ein Fehlen des charakteristischen Peaks für den Phasenübergang in die vordere Stützphase beim linken Bein zu beobachten. Diese Anomalie führte zu Beginn der Arbeiten zu einer schlechten automatischen Erkennung, woraufhin der Datensatz in der Visualisierung untersucht wurde. In der Visualisierung ist eine untypische Lateralbewegung bei der Schwungphase und ein Einknicken des Sprunggelenks beim Aufsetzen des Fußes zu beobachten. Dieses Verhalten konnte durch Rückfrage bei den beteiligten Sportwissenschaftlern auch mit Hilfe des originalen Videomaterials bestätigt werden. Als Resultat konnte die Probandin gezielt auf dieses Bewegungsproblem aufmerksam gemacht werden und beim weiteren Training diese Schwäche berücksichtigt werden. Neben der Optimierung des Laufstils ist es somit auch möglich, eventuelle Spätfolgen durch falsche Belastungen zu vermeiden.

## 4.5 Modelle zur Beschreibung von Bewegungen

Eine genaue deterministische Beschreibung der Merkmalsverläufe gestaltet sich aufgrund der vielfältigen biomechanischen Einflussgrößen des komplexen Systems der menschlichen Bewegung äußerst schwierig. Weit naheliegender ist daher eine stochastische Modellierung der Bewegung mit denselben Methoden, wie sie in den vorigen Kapiteln vorgestellt werden. Im Folgenden werden HMMs zur Analyse von Bewegungen eingesetzt. Dabei wird neben der Erkennung von personenspezifischen Charakteristika und Rahmenbedingungen [108] bei der Bewegungsausführung ausführlich auf die Detektion von Bewegungsphasen [22] und damit auf die Erfassung des Bewegungszustandes eines Benutzers eingegangen. Als Anwendungsszenario dient eine Küchenumgebung, in der Alltagsbewegungen ausgeführt werden. Diese Bewegungen bilden einen Schwerpunkt im SFB 588 „Humanoide Roboter“ [145].

### 4.5.1 Bewegungsdaten

Bei den in dieser Arbeit betrachteten Bewegungen handelt es sich um eine Folge von Teilbewegungen, die beim Tischdecken erfolgen. Die Bewegung sollte von mehreren Benutzern durchgeführt werden. In der vorgegebenen Ausgangsposition standen alle Testpersonen vor dem Tisch. Die Objekte, die bei der Bewegungsdurchführung manipuliert wurden (Tasse, Untertasse, Kaffeekanne, Zuckerdose und Löffel), befanden sich ebenfalls an vorgegebenen Positionen auf einem Tisch und einer erhöhten Ablage. Für die Gesamtbewegung wurden insgesamt 13 einzelne Bewegungsphasen spezifiziert, die von den Probanden durchgeführt werden sollten. Der Versuchsaufbau und spezifizierte Positionen von Objekten und Testpersonen sind in Abbildungen 4.14 anhand mehrerer ausgewählter Zeitpunkte in den wesentlichen Bewegungsphasen zu sehen. Bei den 13 Bewegungsphasen handelt es sich im Detail um folgende elementare Bewegungen:

1. Bewegung der Hand und des Armes aus der Startposition zur Untertasse und Greifen der Untertasse,
2. Anheben der Tasse, Ablegen auf dem Tisch und Öffnen der Hand,
3. Rückbewegung des Armes und Greifen der Tasse,
4. Anheben der Tasse und Absetzen auf der Untertasse,
5. Rückbewegung des Armes, Greifen der Kaffeekanne, Anspannung der Muskeln zum Anheben,
6. Anheben der Kaffeekanne und Führung in die Nähe der Tasse,
7. Eingießen von Flüssigkeit in die Tasse,
8. Rückstellen der Kaffeekanne auf den Tisch,

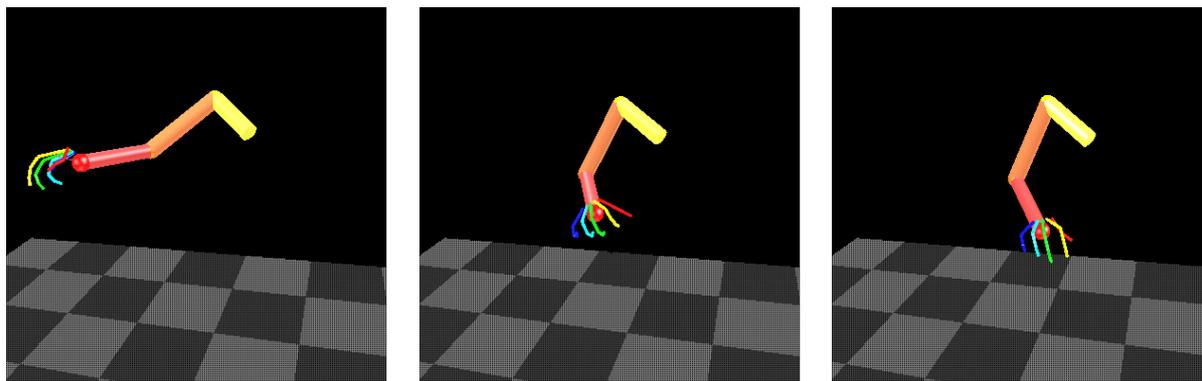


**Abbildung 4.14:** Bilder vom Versuchsaufbau und von ausgewählten Bewegungsphasen (Phase 1: Greifen, Phase 7: Eingießen, Phase 11: Umrühren) der untersuchten Alltagsbewegung.

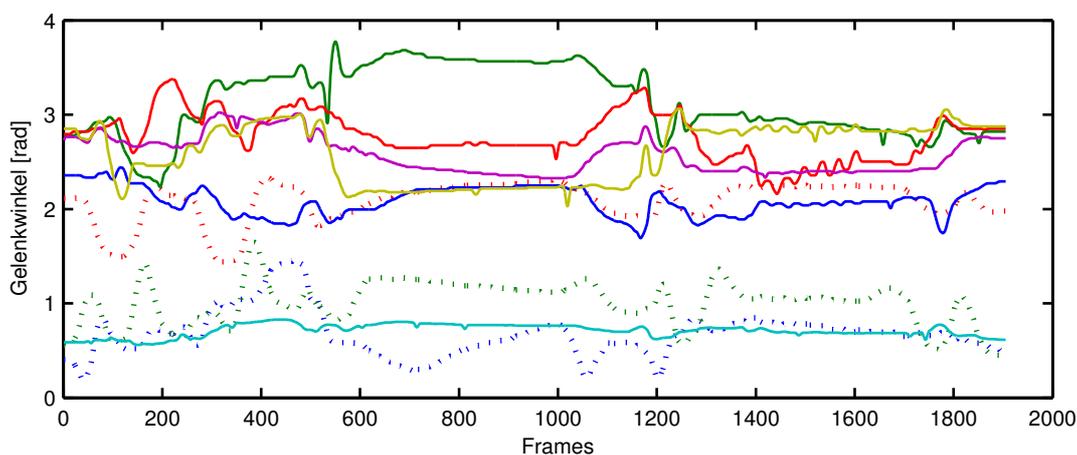
9. Bewegung zum Löffel in der Zuckerdose und Greifen,
10. Bewegung mit dem Löffel zur Tasse und Absenken,
11. Umrühren des Tasseninhalts und Anheben des Löffels,
12. Rücklegen des Löffels,
13. Einnehmen der Ausgangsposition.

Da für die Untersuchungen in dieser Arbeit eine größere Menge von Daten erforderlich ist, wurden insgesamt 7 Probanden beobachtet, wobei jeder der Probanden 10 Wiederholungen durchführte. Zusätzlich wurde durch den Füllstand der Kaffeekanne (voll und nahezu leer) Rahmenbedingungen für die Bewegungsdurchführung vorgegeben. Alle Bewegungen wurden jeweils unter den 2 Rahmenbedingungen durchgeführt, so dass insgesamt ein Bewegungskatalog von  $7 \times 2 \times 10$  Bewegungen erzeugt wurde. Die Dauer der Bewegungen lag zwischen 17,2s und 23,0s.

Die Erfassung der Bewegungen erfolgte mit dem magnetischen Tracking-System aus Abschnitt 4.3.2 in Verbindung mit dem Datenhandschuh aus Abschnitt 4.3.3. Die Daten wurden bei einer Abtastfrequenz von 86,0 Hz zusammengeführt. 6 Magnetsensoren wurden zur Erfassung des Armes verwendet. Die Marker waren dabei auf dem Sternum, der Schulter, der Mitte des Oberarms, der Mitte des Unterarms, dem Handgelenk und dem Handrücken positioniert. Winkel für Schulter, Ellenbogen und Handgelenk wurden mit dem Algorithmus zur Rekonstruktion von Gelenkzentren in Abschnitt 4.3.2.1 bestimmt. Um die feingliedrigen Bewegungen der Hand zu detektieren wurde mit einem kinematischen Modell der Hand (siehe Abschnitt 4.3.3) aus den Sensorwerten die Flexions- und Abduktionswinkel von Daumen und Zeigefinger ermittelt. Insgesamt ergeben sich damit 28 Winkelkomponenten für die Merkmalsvektoren zu jedem Zeitpunkt einer Bewegung. Visualisierungen des vollständigen kinematischen Modells für ausgewählte Bewegungs-



**Abbildung 4.15:** Visualisierung der Komponenten des Modells für die ausgewählten Bewegungsphasen in Abb. 4.14.



**Abbildung 4.16:** Ausgewählte Gelenkwinkel einer Bewegung: Winkel der Schulter (rot, gepunktet), Winkel des Ellenbogens (grün, gepunktet), Winkel des Handgelenks (blau, gepunktet), Rotation des Daumens (blau), Flexion des Daumengrundgelenks (grün), Flexion des distalen Daumengelenks (rot), Abduktion zwischen Daumen und Zeigefinger (cyan), Flexion des Zeigefingergrundgelenks (magenta), Flexion des medialen Zeigefingergelenks (gelb).

phasen sind in Abb. 4.15 zu sehen. Abbildung 4.16 zeigt ausgewählte Winkeltrajektorien in einem aufgenommenen Datensatz.

Im Gegensatz zu absoluten räumlichen Koordinaten geben die Trajektorien von Gelenkwinkeln relative Informationen, die unabhängig von der Position des Probanden und der Länge von Gliedmaßen die menschliche Bewegungen vollständig beschreiben. In dieser Arbeit werden daher ausschließlich mit Winkeltrajektorien gearbeitet. Während für detaillierte kinematische Analyse von Bewegungen weitestgehend alle 28 gemessenen Win-



	$u = 1$	$u = 2$	$u = 3$	$u = 4$	$u = 5$	$u = 6$	$u = 7$
$\lambda_1$	<b>-2,2348</b>	-5,6312	-3,2888	-4,1871	-3,5080	-3,2916	-5,3301
$\lambda_2$	-3,5637	<b>-2,0914</b>	-3,6518	-3,6666	-3,2739	-3,2114	-3,1241
$\lambda_3$	-2,9236	-3,9969	<b>-2,0385</b>	-3,4827	-3,5338	-2,9912	-4,3696
$\lambda_4$	-3,5124	-4,2642	-2,9616	<b>-2,0704</b>	-3,7080	-2,9421	-4,0808
$\lambda_5$	-2,6774	-3,1919	-2,9368	-3,1109	<b>-1,9132</b>	-2,6686	-3,5728
$\lambda_6$	-4,7377	-5,4407	-3,9250	-4,9334	-4,5901	<b>-2,6951</b>	-5,4017
$\lambda_7$	-2,6882	-2,4654	-2,5267	-2,7277	-2,8242	-2,2387	<b>-1,4187</b>

**Tabelle 4.2:** Wahrscheinlichkeiten  $\log P_{\lambda_1, \dots, 7}(O_u)$  für Beobachtungen von Bewegungen der Personen  $u = 1, \dots, 7$  mit den HMMs  $\lambda_1, \dots, \lambda_7$  zur Erkennung von Personen mit  $N_S = 20$  Zuständen (Skalierung mit Faktor  $10^{-4}$ ).<sup>1</sup>

Zur Durchführung einer Kreuzvalidierung wurden Trainings- und Testdaten erzeugt. Dabei wurden in Anlehnung an das *Leaving-one-out*-Verfahren [169] die 20 Datensätze pro Person in 18 für Trainingszwecke und 2 (volle und nahezu leere Kaffeekanne) für Testzwecke aufgespaltet. Insgesamt wurden pro Person 10 Trainings- und Testläufe mit den zwei unterschiedlichen, für den Test reservierten Datensätzen durchgeführt. Grundlage für die Selektion der Testdatensätze bildet die Permutation über die pro Person und Rahmenbedingung durchgeführten 10 Bewegungsdurchführungen. Durch diese Form der Kreuzvalidierung wird sowohl ein praktikabler Rechenaufwand für Training und Tests als auch eine Stratifizierung durch Aufteilung in gleich mächtige Trainingsmengen erzielt [169]. Insgesamt ergeben sich 140 Testfälle bei der beschriebenen Kreuzvalidierung.

Die HMM-Parameter der Übergangswahrscheinlichkeiten und der Mischverteilungskomponenten wurden mit Hilfe des Baum-Welch-Algorithmus mit den Trainingsdaten auf die jeweilige Person angepasst. Tests haben gezeigt, dass 20 Iterationen beim Training zeigen einen guten Kompromiss zwischen Trainingsaufwand und Fix-Punkt-Annäherung ergeben. Zur Ermittlung einer optimalen Anzahl von Zuständen wurden HMMs mit unterschiedlichen Anzahlen untersucht. Tabelle 4.2 gibt die mittleren Wahrscheinlichkeiten der personenspezifischen HMMs über alle 10 Bewegungen der Testpersonen einschließlich der zwei Rahmenbedingungen wieder. Alle HMMs in diesem Testfall haben  $N_S = 20$  Zustände.

Jedes HMM liefert die höchsten Wahrscheinlichkeiten für die Bewegungen der Person, für die es trainiert wurde. Die maximalen Wahrscheinlichkeiten, die über das Majoritätsprinzip die Klassifikation herbeiführen, sind in der Tabelle auch die markierten Diagonalelemente. Demzufolge erlauben die HMMs eine verlässliche Erkennung der Person. Betrachtet man alle 140 Testfälle wird eine Erkennungsrate von 100,0 % erzielt.

Erstaunlicherweise ergeben auch HMMs mit  $N_S = 1$  Zustand bereits eine gute Erken-

<sup>1</sup>Für die gemessenen logarithmischen Wahrscheinlichkeiten  $\log P_\lambda(O)$  aus dem Bereich  $[-6,0 \cdot 10^{-4}, 0]$  gilt  $(1 - P_\lambda(O)) \approx -\log P_\lambda(O)$  mit einem maximalen Fehler  $1,8 \cdot 10^{-7}$ . Zur besseren Darstellung werden daher hier die für HMMs üblichen logarithmischen Wahrscheinlichkeiten angegeben, wie sie vom Vorwärts-Algorithmus berechnet und auch zur Klassifikation mittels des Majoritätsprinzips verwendet werden.

	$u = 1$	$u = 2$	$u = 3$	$u = 4$	$u = 5$	$u = 6$	$u = 7$
$\lambda_1$	<b>-1,9455</b>	-4,6935	-2,8096	-3,1406	-2,7435	-2,5426	-4,8437
$\lambda_2$	-3,0554	<b>-1,7358</b>	-3,0795	-2,6512	-2,4883	-2,4803	-2,5053
$\lambda_3$	-2,3881	-3,1699	<b>-1,7710</b>	-2,3722	-2,6921	-2,3323	-3,7895
$\lambda_4$	-3,0315	-3,4803	-2,4815	<b>-1,8442</b>	-3,1437	-2,4601	-3,5975
$\lambda_5$	-2,1763	-2,5236	-2,4346	-2,4791	<b>-1,6977</b>	-2,1323	-3,0461
$\lambda_6$	-4,1460	-4,5814	-3,3457	-3,8873	-3,7039	<b>-2,4079</b>	-4,5250
$\lambda_7$	-2,3059	-2,0871	-2,1053	-2,0124	-2,2205	-1,7184	<b>-1,2449</b>

**Tabelle 4.3:** Wahrscheinlichkeiten  $\log P_{\lambda_1, \dots, \lambda_7}(O_u)$  für Beobachtung von Bewegungen der Personen  $u = 1, \dots, 7$  mit den HMMs  $\lambda_1, \dots, \lambda_7$  zur Erkennung von Personen mit  $N_S = 1$  Zustand (Skalierung mit Faktor  $10^{-4}$ ).<sup>2</sup>

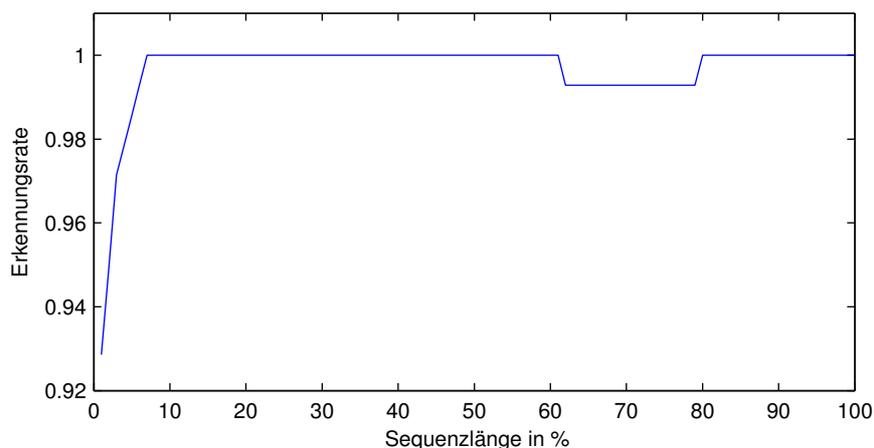
nungsrates. Bei diesen HMMs handelt es sich um Mischverteilungsklassifikatoren (siehe Abschnitt 2.3.3.2). Die Erkennungsrate beträgt 99,3%. Jedoch werden in der Regel mehr Zustände benötigt, um die maximalen Wahrscheinlichkeiten signifikanter von den übrigen Wahrscheinlichkeiten zu unterscheiden. Tabelle 4.3 gibt entsprechend die mittleren Wahrscheinlichkeiten der personenspezifischen HMMs über alle 10 Bewegungen der Testpersonen einschließlich der zwei Rahmenbedingungen an.

### 4.5.3 Länge der Beobachtungssequenzen

In einem weiteren Test wurden Bewegungen unterschiedlicher Länge untersucht. Zielsetzung war es, die Auswirkung der Länge der Beobachtungssequenzen auf die Erkennung der Personen zu analysieren. Für den Versuch wurden Testdatensätze mit Beobachtungssequenzen  $O_\tau = \mathbf{o}_1 \dots \mathbf{o}_{\lfloor \tau T \rfloor}$  unterschiedlicher Länge  $1,00\% < \tau < 100,00\%$  aus den ursprünglichen Testdatensätzen mit Sequenzen  $O = \mathbf{o}_1 \dots \mathbf{o}_T$  durch Extraktion von Frames generiert. Abbildung 4.17 zeigt die Gesamterkennungsrates in Abhängigkeit der Sequenzlängen. In diesem Versuch wurden erneut die personenspezifischen HMMs mit  $N_S = 20$  Zuständen verwendet.

Bereits ein Sequenz mit  $\tau = 7,00\%$  der Frames erlaubt die korrekte Klassifikation der Person in allen Testbewegungen. Jedoch geht die Erkennungsrate ab  $\tau = 61,00\%$  auf 99,3% zurück und erreicht 100,0% nach  $\tau = 80,00\%$ . Der Grund für dieses Verhalten ist die große intra-individuelle Varianz der Bewegungen im Vergleich zu inter-individuellen Unterschieden im Bereich  $61,00\% < \tau < 80,00\%$ . Als Folge gelangen die HMMs in eine falsche Zustandsfolge und nicht alle Bewegungen werden korrekt klassifiziert. Sobald ausreichend

<sup>2</sup>Für die gemessenen logarithmischen Wahrscheinlichkeiten  $\log P_\lambda(O)$  aus dem Bereich  $[-6,0 \cdot 10^{-4}, 0]$  gilt  $(1 - P_\lambda(O)) \approx -\log P_\lambda(O)$  mit einem maximalen Fehler  $1,8 \cdot 10^{-7}$ . Zur besseren Darstellung werden daher hier die für HMMs üblichen logarithmischen Wahrscheinlichkeiten angegeben, wie sie vom Vorwärtsalgorithmus berechnet und auch zur Klassifikation mittels des Majoritätsprinzips verwendet werden.



**Abbildung 4.17:** Erkennungsraten bei der Personenerkennung mit Beobachtungssequenzen unterschiedlicher Länge.

Informationen zur Charakterisierung der Person gegeben sind, liefern die HMMs wieder korrekte Ergebnisse.

#### 4.5.4 Erkennung von Rahmenbedingungen

In diesem Versuch wurden HMMs zur Erkennung von Rahmenbedingungen bei der Bewegungsausführung verwendet. Als Rahmenbedingung wurde der Füllstand der Kaffeekanne betrachtet. Aufgrund des unterschiedlichen Gewichtes und Ausgießverhaltens sollte der Füllstand der Kanne entscheidend Einfluss auf die Bewegungsausführung nehmen.

Zwei HMMs  $\lambda_1$  und  $\lambda_2$  wurden mit den Trainingsdaten der unterschiedlichen Rahmenbedingungen trainiert und mit Bewegungssequenzen voller Länge getestet. Tabelle 4.4 beinhaltet die Wahrscheinlichkeiten der HMMs für die Rahmenbedingungen. Es wurden  $N_S = 40$  Zustände für die HMMs verwendet, um die Ergebnisse zu produzieren.

Mit den angegebenen HMMs wurde eine mittlere Gesamterkennungsrate von 89,3% erzielt. Da die Erkennung der Rahmenbedingungen für die Bewegungsausführung wesentlich schwieriger ist als die Personenerkennung in Abschnitt 4.5.2, sind auch die Erkennungsraten deutlich schlechter. Jedoch ermöglichen die HMMs eine signifikante Klassifikation des Füllstandes der Kanne in den meisten Testfällen.

Erneut ist die Frage der Anzahl von Zuständen entscheidend für das Klassifikationsergebnis. Offensichtlich erfordert die Erkennung der feinen Unterschiede in der Bewegungsausführung mehr Zustände als bei der Personenerkennung in Abschnitt 4.5.2. Tabelle 4.5 zeigt die Erkennungsraten für alle Testdatensätze bei unterschiedlichen Anzahlen von Zuständen. Die Erkennungsrate ging deutlich zurück, wenn  $N_S < 40$  Zustände verwendet wurden. Im Gegensatz dazu wurden keine Verbesserungen mehr erzielt, wenn  $N_S > 40$  Zustände für die Klassifikation verwendet wurden.

	$c = 1$	$c = 2$
$\lambda_1$	<b>-2,5370</b>	-2,5962
$\lambda_2$	-2,4038	<b>-2,2917</b>

**Tabelle 4.4:** Wahrscheinlichkeiten  $\log P_{\lambda_{1,2}}(O_c)$  für Beobachtungen von Bewegungen unter den Rahmenbedingungen  $c = 1$  (volle Kaffeekanne) und  $c = 2$  (nahezu leere Kaffeekanne) mit den HMMs  $\lambda_1$  und  $\lambda_2$  zur Erkennung der Rahmenbedingung bei der Bewegungsausführung (Skalierung mit Faktor  $10^{-4}$ ). Die Anzahl der HMM-Zustände beträgt  $N_S = 40$ .<sup>3</sup>

$N_S = 1$	$N_S = 20$	$N_S = 40$	$N_S = 80$
80,7 %	85,7 %	89,3 %	89,3 %

**Tabelle 4.5:** Mittlere Erkennungsraten bei der Identifikation von Rahmenbedingungen mit unterschiedlichen Anzahlen von Zuständen  $N_S$  pro HMM.

#### 4.5.5 Vergleich mit Elman-Netzen

Um die Erkennungsraten der HMMs aus Abschnitt 4.5.2 mit alternativen Klassifikationsverfahren zu vergleichen, wurde die Erkennung von Personen mit Neuronalen Netzen als klassischem Lösungsansatz für Erkennungsaufgaben durchgeführt. Da die Ausgaben eines Neuronalen Netzes nur von den Eingaben zu einem bestimmten Zeitpunkt abhängig sind, im vorliegenden Fall jedoch Bewegungstrajektorien untersucht werden sollen, wurden rekurrente Neuronale Netze in Form von Elman-Netzen (siehe Abschnitt B.3.3.1) verwendet.

Für die Erkennungsaufgabe wurde für jede Person ein Elman-Netz mit den personenspezifischen Trainingsdaten eintrainiert. Analog zu den Tests mit HMMs wurden insgesamt 126 Datensätze für das Training verwendet, die restlichen 14 Datensätze wurden für Testzwecke zurückgehalten. Insgesamt wurden 10 Testläufe mit jeweils unterschiedlichen Testdaten durchgeführt. Jedes Netz besitzt in der ersten Schicht 15 Neuronen und 1 in der zweiten Schicht. Als Transferfunktion wurde für die Neuronen der ersten Schicht die Funktion (B.6) und für die Neuronen der zweiten Schicht die Funktion (B.5) verwendet.

Die Aufgabe jedes Elman-Netzes ist, zwischen den Bewegungen der eintrainierten Person und der Bewegungen der übrigen Personen zu unterscheiden. Jedes Elman-Netz für eine Person  $u$  besitzt dazu die Ausgabefunktion  $y_u : \mathbf{V}^T \times \{0, \dots, T\} \rightarrow [0, 1]$  für eine

<sup>3</sup>Für die gemessenen logarithmischen Wahrscheinlichkeiten  $\log P_\lambda(O)$  aus dem Bereich  $[-6,0 \cdot 10^{-4}, 0]$  gilt  $(1 - P_\lambda(O)) \approx -\log P_\lambda(O)$  mit einem maximalen Fehler  $1,8 \cdot 10^{-7}$ . Zur besseren Darstellung werden daher hier die für HMMs üblichen logarithmischen Wahrscheinlichkeiten angegeben, wie sie vom Vorwärts-Algorithmus berechnet und auch zur Klassifikation mittels des Majoritätsprinzips verwendet werden.

	$u = 1$	$u = 2$	$u = 3$	$u = 4$	$u = 5$	$u = 6$	$u = 7$
$EN_1$	<b>0,8789</b>	0,0009	0,0207	0,0000	0,0074	0,0247	0,0045
$EN_2$	0,0219	<b>0,9960</b>	0,0010	0,0000	0,0069	0,0010	0,0006
$EN_3$	0,0043	0,0004	<b>0,9066</b>	0,0043	0,0795	0,0333	0,0000
$EN_4$	0,0054	0,0024	0,0200	<b>0,9851</b>	0,0044	0,0259	0,0000
$EN_5$	0,0207	0,0000	0,0294	0,0021	<b>0,8094</b>	0,1014	0,0114
$EN_6$	0,0558	0,0000	0,0198	0,0085	0,0633	<b>0,8133</b>	0,0003
$EN_7$	0,0130	0,0002	0,0025	0,0000	0,0290	0,0003	<b>0,9831</b>

**Tabelle 4.6:** Erkennungsraten für Elman-Netze mit 15 Neuronen. Bei den Werten handelt es sich um die mittleren Wahrscheinlichkeiten  $P_{u=1,\dots,7}(O, t)$  über alle Rahmenbedingungen und Bewegungsdurchführungen zum Zeitpunkt  $t = T$ , wenn eine Bewegung getestet wird, die von Person  $u$  durchgeführt wurde. Eine perfekte Klassifikation ergäbe die Einheitsmatrix.

Bewegungssequenz  $O$  der Länge  $T$  mit Beobachtungen  $\mathbf{o}_{1,\dots,T} \in \mathbf{V} = \mathbb{R}^{18}$  und einem Zeitpunkt  $t = 0, \dots, T$ . Die zugehörigen Zielfunktion  $y_{d,u} : \mathbf{V}^T \times \{0, \dots, T\} \rightarrow \{0, 1\}$ , mit der die Netze trainiert wurden, liefert den Wert 1, wenn die Bewegung in Trainingsdatensatz  $O_{d=1,\dots,N_D}$  von Person  $u$  stammt und 0 sonst. Die Auswertung der Ausgaben aller Netze wird analog zu den HMMs über das Majoritätsprinzip durchgeführt, d. h. die getestete Bewegung wird mit der Person assoziiert, deren zugehöriges Netz den höchsten Ausgabewert berechnet hat. Die Wahrscheinlichkeit, mit der eine Person korrekt bis zu einem Zeitpunkt  $t$  mit Hilfe aller Netze erkannt wurde, wird über die relative Anzahl der Zeitpunkte mit korrekter Klassifikation berechnet:

$$P_u(O, t) = \frac{1}{T} \left| \left\{ 0 \leq \tau \leq t \mid u = \arg \max_{k=1,\dots,7} (y_k(O, \tau)) \right\} \right|. \quad (4.6)$$

Tabelle 4.6 gibt eine Übersicht über alle getesteten Bewegungen und den Wahrscheinlichkeiten  $P_{u=1,\dots,7}(O, T)$  am Ende der Bewegungen. Die mittlere Wahrscheinlichkeit für die korrekte Erkennung der Personen beträgt 91,5 %. Die mittleren Erkennungsraten zeigen, dass die Bewegungen einiger Testpersonen sehr leicht, andere dafür aber wesentlich schwerer erkannt werden können (siehe Tabelle 4.7). Der Füllstand der Kaffeekanne spielt dabei keine entscheidende Rolle. In Tabelle 4.7 werden zusätzlich die Auswirkungen unterschiedlicher Anzahlen  $N_N$  von Neuronen auf die Erkennungsraten dargestellt. Im Gegensatz zu den Netzen mit 15 Neuronen liefern Netze mit 20 und mehr Neuronen keine wesentlichen Verbesserungen der Erkennungsraten. Werden stattdessen 10 oder sogar nur 5 Neuronen verwendet, so sinken die Erkennungsraten deutlich ab. Offenbar scheinen für diese Aufgabe Netze mit 15 Neuronen einen guten Kompromiss zwischen Generalisierung und Erkennungsrate zu liefern.

Vergleich man die Elman-Netze mit den HMMs, so stellt sich heraus, dass beide Methoden zur Erkennung von Personen genutzt werden können, die HMMs jedoch höhere

	$N_N = 5$	$N_N = 10$	$N_N = 15$	$N_N = 20$
$u = 1$	0,8712	0,8902	0,8808	0,8870
$u = 2$	0,9984	0,9989	0,9960	0,9989
$u = 3$	0,8397	0,8312	0,9077	0,9212
$u = 4$	0,9887	0,9712	0,9853	0,9915
$u = 5$	0,6812	0,7568	0,8101	0,8199
$u = 6$	0,7229	0,8174	0,8126	0,8076
$u = 7$	0,9641	0,9700	0,9832	0,9779
Mittelwert	0,8666	0,8908	0,9108	0,9149

**Tabelle 4.7:** Wahrscheinlichkeiten für korrekte Klassifikation bei verschiedenen Anzahlen von Neuronen  $N_N$  in der ersten Schicht der Elman-Netze. Die Werte zeigen die mittleren Raten für die korrekte Erkennung der Testdatensätze.

Erkennungsraten liefern. Der Grund hierfür ist einerseits die bessere Erfassung und Bewertbarkeit der Merkmalsvektoren durch die verwendeten Mischverteilungsklassifikatoren. Andererseits scheint die Repräsentation von signifikanten Bewegungsphasen besser über die versteckten Zustände in den HMMs als über die Rückkopplung der Ausgaben in die Eingaben der Neuronen zu wirken. Grund hierfür ist insbesondere, dass die Informationen über die Historie durch die Zuführung immer neuer Information exponentiell mit der Zeit abnimmt.

Weiterhin muss bezüglich der Erkennung des Füllstandes der Kaffeekanne bei der Bewegungsausführung bemerkt werden, dass keine leistungsfähigen Elman-Netze für diese Klassifikationsaufgabe erzeugt werden konnten. Eine Unterscheidung der für diese Bewegung charakteristischen Merkmale ist offenbar trotz der zusätzlichen Rückkopplung in Elman-Netzen durch eine Separierung mittels Hyperebenen im Merkmalsraum nicht möglich. Eine Bewertung der Merkmale mit Hilfe von Mischverteilungskomponenten, deren Rolle bei der Analyse von Bewegungsphasen im späteren Abschnitt 4.5.6.3 auch im Zusammenhang mit RBFNs bzw. SVMs noch weiter erläutert wird, und die hier wichtige ausreichend große Anzahl von Zuständen in den HMMs stellen hingegen eine Lösung für die gestellte Erkennungsaufgabe dar.

#### 4.5.6 Erkennung von Bewegungsphasen mittels Klassifikation

Bei der Erkennung von Personen und Rahmenbedingungen anhand von Bewegungstrajektorien wird in den vorigen Abschnitten ersichtlich, dass die Anzahl von Zuständen in den Modellen und die damit assoziierten elementaren Bewegungsphasen neben den Eigenschaften der Bewegungsmerkmale die entscheidende Rolle bei den Erkennungsaufgaben spielen. Im Folgenden wird daher gezielt auf die Analyse dieser elementaren Bewegungsphasen eingegangen. Gerade ein Verständnis für die Phasen in einzelnen Bewegungen sind

die Grundlage für den Entwurf leistungsfähiger Modelle zur Analyse und Erkennung von Bewegungen.

#### 4.5.6.1 Manuelle Segmentierung der Bewegungsphasen

Als Datenmaterial dient in diesem Abschnitt wieder der umfassende Bewegungskatalog von Abschnitt 4.5.1. Da nicht nur die Gesamtbewegungen untersucht werden sollen, wurden die Daten zusätzlich anhand von Videoaufzeichnungen und anhand der Visualisierungen des Körpermodells in (siehe Abb. 4.15) in die vorgegebenen Bewegungsabschnitte manuell segmentiert. Da die Versuchspersonen die Bewegungen unter den zwei Rahmenbedingungen „volle Kaffeekanne“ und „nahezu leere Kaffeekanne“ durchführten, wurden auch die Bewegungsphasen, in denen die Kaffeekanne bewegt wurde, in zwei Klassen unterteilt. Die Phasen 6 bis 8 beziehen sich auf die Bewegungen mit der nahezu leeren Kaffeekanne, während die zusätzlich eingeführten Phasen 14 bis 16 diejenigen mit der vollen Kanne darstellen.

#### 4.5.6.2 Analyse von Bewegungsphasen mit HMMs

Zur Analyse der Bewegungsphasen mit HMMs werden als Initialisierung für die Modelle dieselben Parameter für Bakis-Modelle wie in Abschnitt 4.5.2 verwendet. Im Gegensatz zur Erkennung von Personen und Rahmenbedingungen, bei denen immer gesamte Bewegungssequenzen analysiert werden, sind in diesem Fall Teilbewegungen aus den Gesamtbewegungen herausgegriffen. Dazu werden über die Gesamtsequenz  $O = \mathbf{o}_1 \dots \mathbf{o}_T$  Fenster der Länge  $T_W$  zur Erzeugung von Teilbewegungen  $O_{t_0} = \mathbf{o}_{t_0} \dots \mathbf{o}_{t_0+T_W}$  mit  $t_0 < T - T_W$  geschoben. Jeder Teilsequenz wird zusätzlich eine der Bewegungsphasen  $p = 1, \dots, 13$  über das Startelement  $\mathbf{o}_{t_0}$  zugeordnet.

Mit den Teilbewegungen wurde anschließend für jede Bewegungsphase  $p$  ein HMM  $\lambda_p$  mit dem Baum-Welch-Algorithmus trainiert. Um den Aufwand für das Training niedrig zu halten, wurden nicht alle Bewegungssequenzen pro Phase für das Training herangezogen. Die Fenster zur Extraktion von Teilbewegungen wurden mit einer Schrittweite von 10 Frames (dies entspricht einem zeitlichen Versatz von 0,12s) verschoben, um noch eine ausreichende Anzahl von Trainingsdatensätzen zu haben. Der Baum-Welch-Algorithmus lief mit 10 bis 20 Iterationen als Kompromiss zwischen Trainingsaufwand und Dauer bis zum Erreichen des Fixpunktes. Pro Phase wurden wieder 10 Testläufe generiert, wobei eine Bewegung für Tests, die restlichen für das Training verwendet wurden. In Summe ergeben sich somit über alle Personen und Rahmenbedingungen erneut insgesamt 140 Testfälle.

Im Gegensatz zu dem im späteren Abschnitt 4.7 vorgestellten Verfahren, erfolgt die Erkennung der Bewegungsphase hier erneut anhand des herkömmlichen Majoritätsprinzips und den Wahrscheinlichkeiten  $P_{\lambda_p}(O)$  der HMMs  $\lambda_p$ , die über den Vorwärts-Algorithmus berechnet werden. Bei der Auswertung sind die Beobachtungssequenzen wie beim Training gefenstert. Im Unterschied zum Training wird jedoch die Schrittweite auf 1 gesetzt, so dass alle Frames abgetastet werden.

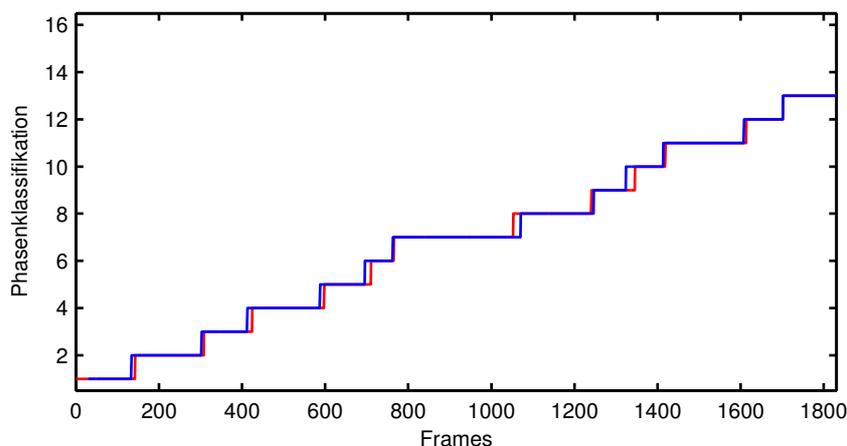
Zur Bewertung der HMMs wurden zunächst unterschiedliche Fensterlängen  $T_W$  betrachtet, um die erforderliche Größe zu ermitteln. Tabelle 4.8 gibt die mittleren Erkennungsraten über alle 140 Testdatensätze bei unterschiedlichen Fensterlängen wieder. Sind die Fenster zu breit gewählt, werden insbesondere die HMMs für die kürzeren Phasen schlecht trainiert. Der Grund hierfür ist, dass das Fenster benachbarte Phasen überlappt und somit die Charakteristika dieser Phasen mit eintrainiert werden. Beobachtungssequenzen mit 30 Frames ergaben die besten Erkennungsraten. Jedoch erlauben bereits Fenster der Länge 1, die Betrachtung von Einzel-Frames, eine gute Klassifikation der aktuellen Bewegungsphase. Dies ist ein Hinweis darauf, dass sich die Charakteristika der Bewegungsphasen bereits schon anhand geeigneter Merkmale gut ablesen lassen und besonderen Wert auf die Generierung aussagekräftiger Merkmale gelegt werden sollte.

Fenstergröße $T_W$	Zustände $N_S$	Erkennungs- rate	ohne Rahmen- bedingungen
100	1	0,7464	–
30	1	0,8478	0,8796
30	2	0,8515	0,8875
30	5	0,8653	0,8673
30	10	0,8215	0,8403
1	1	0,8414	0,8721

**Tabelle 4.8:** Erkennungsraten bei der Klassifikation von Bewegungsphasen bei HMMs mit unterschiedlichen Parametrisierungen. Die Erkennungsraten steigen, wenn zwischen den Phasen 6 und 14, 7 und 15 sowie 8 und 16 nicht unterschieden wird (ohne Rahmenbedingungen).

Ein weiteres Problem ist die Bestimmung der geeigneten Anzahl von Zuständen  $N_S$ . Hierzu wurden auch unterschiedlichen Parametrisierung der HMMs getestet. Die Ergebnisse sind in Tabelle 4.8 zusammengefasst. Die Tests zeigen, dass wie bei der Erkennung von Personen und Rahmenbedingungen eine Anzahl von  $N_S = 1$  Zuständen bereits eine gute Klassifikation erlauben. Die Verwendung von  $N_S = 2$  Zuständen ergibt nur eine geringfügige Verbesserung der Erkennungsrate. Mit größeren Anzahlen von Zustände gehen die Erkennungsraten wieder zurück. Der Grund hierfür ist, dass die Modelle mit geringeren Anzahlen von Zuständen eine bessere Anpassung an die Trainingsdaten bei den gegebenen Datenmengen erlauben. Die geringe Anzahl von Zuständen für die Erkennungsphase ist ein großer Vorteil beim Einsatz in zeitkritischen Applikationen z. B. mit Echtzeitanforderungen. Zusätzlich sind die geringen Zustände ein Indiz dafür, dass es sich bei den vorgegebenen Bewegungsphasen tatsächlich um elementare Teilbewegungen handelt. Diese Sachverhalte werden, in Abschnitt 4.7 bei der Komposition von HMMs noch ausführlicher erläutert.

In Tabelle 4.8 sind neben den Erkennungsraten für alle 16 Bewegungsphasen auch die Erkennungsraten dafür angegeben, wenn zwischen den Phasen unterschiedlicher Rahmen-

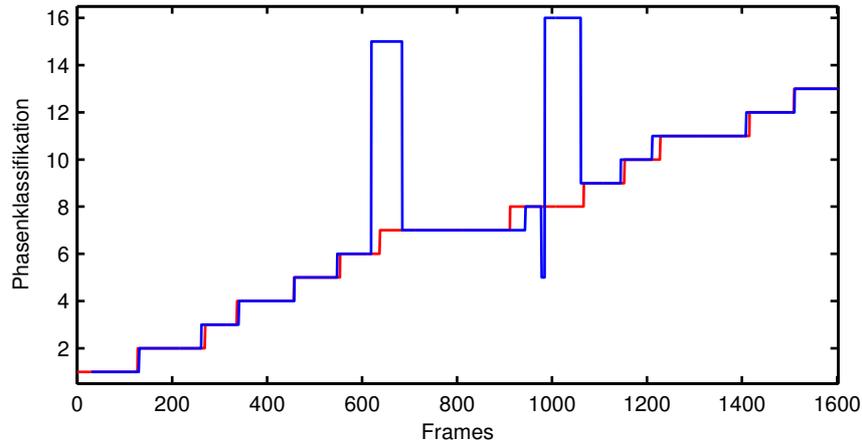


**Abbildung 4.18:** Ergebnisse der Phasenklassifikation mit HMMs ( $N_S = 2$  Zustände, Fensterlänge  $T_W = 30$ ). Die Erkennungsrate für diese Bewegung entspricht der mittleren Erkennungsrate über alle Testdatensätze. Blaue Kurve: Klassifikationsergebnis. Rote Kurve: Vorgabe aus manueller Klassifikation.

bedingungen 6–8 und 14–16 nicht unterschieden wird. Die Veränderungen in den Erkennungsraten gegenüber den unterschiedenen Bewegungsphasen ist marginal. Folglich können mit Hilfe der HMMs auch die Rahmenbedingungen bei der Bewegungsdurchführung erkannt werden. Die generell häufigste Ursache für Fehlklassifikationen der Modelle sind geringfügige Unterschiede bei den Anfängen der Bewegungsphasen gegenüber den manuell segmentierten Zielwerten. Der Verlauf der Bewegung wird jedoch in allen Fällen korrekt wiedergegeben. Die Abbildungen 4.18 und 4.19 gestatten den direkten Vergleich der automatischen Erkennung durch HMMs mit den Vorgaben der manuellen Segmentierung für ausgewählte Datensätze.

#### 4.5.6.3 Vergleich mit Neuronalen Netzen

Zum Vergleich mit anderen Methoden der Datenklassifikation werden die HMMs des vorigen Abschnittes ebenfalls mit Elman-Netzen und zusätzlich mit RBFNs verglichen. Grundlagen zu diesen Spezialformen von Neuronalen Netzen finden sich im Anhang in den Abschnitten B.3.3.1 und B.3.4. Im Gegensatz zu den Elman-Netzen sind RBFNs nicht sensitiv gegenüber dem zeitlichen Verlauf der Merkmalstrajektorien, die an den Eingang der Netze Frame-weise angelegt werden. RBFNs teilen den Merkmalsraum über die Euklidische Distanz zwischen Merkmalsvektoren auf. Bei Elman-Netzen erfolgt dies über Hyperebenen. RBFNs sind daher mit Mischverteilungsklassifikatoren verwandt und sollten, wie in Abschnitt 4.5.2 beschrieben, ebenfalls verlässliche Erkennungsraten liefern. Man beachte in diesem Zusammenhang auch, dass RBFNs eine Spezialform von SVMs sind (siehe Anhang C.2).

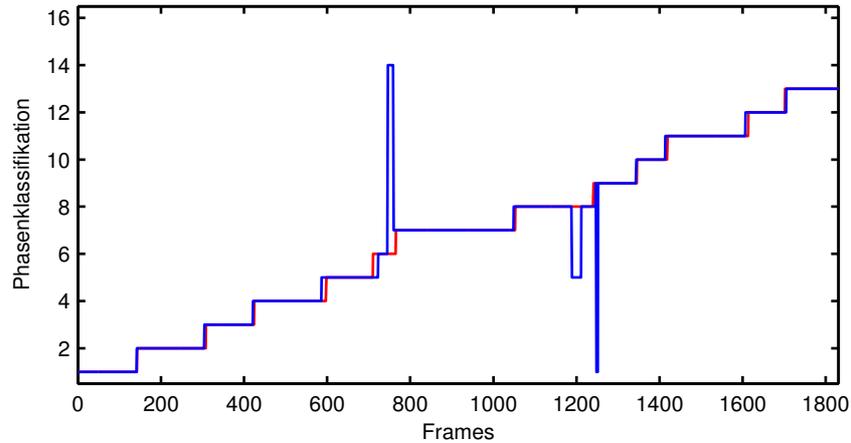


**Abbildung 4.19:** Ergebnisse der Phasenklassifikation mit HMMs ( $N_S = 2$  Zustände, Fensterlänge  $T_W = 30$ ). Die Erkennungsrate für diese Bewegung war eine der höchsten verglichen mit allen Testdatensätzen. Blaue Kurve: Klassifikationsergebnis. Rote Kurve: Vorgabe aus manueller Klassifikation.

Die Elman-Netze besitzen 20 Neuronen in der ersten Schicht und 1 Neuron in der zweiten Schicht. Da bei der Generalisierung während des Trainings die Elman-Netze diejenigen Klassen bevorzugen, die häufiger in den Trainingsdaten auftreten, sind die Bewegungsphasen bezüglich der Frame-Zahl normalisiert. Um genau zu sein, sind die Bewegungsphasen mit mehr als 50 Frames unterabgetastet, so dass die maximale Anzahl von Trainings-Samples für jede Bewegungsphase 50 Frames beträgt. Hierdurch wird gewährleistet, dass alle Bewegungsphasen beim Training gleichermaßen berücksichtigt werden.

Zum Training der RBFNs wurde zunächst die Anzahl der Trainingsamples durch Unterabtafung um den Faktor 40 verkleinert. Der Grund hierfür ist, den Hauptspeicherbedarf beim Training zu reduzieren. Um eine optimale Anzahl von Basisfunktionen zu finden, wurden die Netze mit 16, 32, 64, 125, 250, 500 und 1000 RBFs getestet. Die resultierenden Erkennungsraten sind in Tabelle 4.9 aufgelistet.

Die Ergebnisse für 1000 RBFs zeigen, dass die Klassifikation von Bewegungsphasen bei 87,09% aller Frames korrekt durchgeführt werden kann. Abbildung 4.20 zeigt das Ergebnis der Klassifikation für eine ausgewählte Bewegung im Vergleich zu den manuell segmentierten Bewegungsphasen. Die Erkennungsrate für diese Bewegung beträgt 87,95%. Die häufigsten Fehlklassifikationen treten in den Phasen 6 und 8 auf, da sie sehr kurz sind und leicht mit den Phasen 14 und 16 verwechselt werden können. Die Erkennungsrate steigt um 2%, wenn zwischen diesen Phasen nicht unterschieden wird (siehe rechte Spalte von Tabelle 4.9). Das Klassifikationsergebnis für eine Bewegung mit einer hohen Erkennungsrate von 94,27% (siehe Abb. 4.21) ergibt, dass wie bei den HMMs die Klassifikationsfehler durch geringfügige Verschiebungen an den Phasenübergängen entstehen.

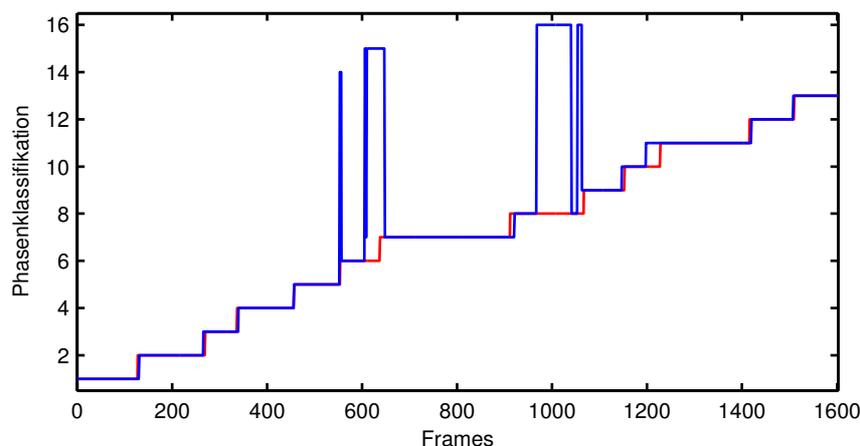


**Abbildung 4.20:** Ergebnisse der Phasenklassifikation mit RBFNs mit 1000 RBFs. Die Erkennungsrate für diese Bewegung entspricht der mittleren Erkennungsrate über alle Testdatensätze. Blaue Kurve: Klassifikationsergebnis. Rote Kurve: Vorgabe aus manueller Klassifikation.

Anzahl RBFs	Erkennungsrate	ohne Rahmenbedingungen
16	0,5533	0,6029
32	0,6186	0,6572
64	0,6789	0,7081
125	0,7476	0,7680
250	0,8078	0,8267
500	0,8558	0,8732
1000	0,8709	0,8903

**Tabelle 4.9:** Erkennungsraten von RBFNs zur Phasenklassifikation bei unterschiedlichen Anzahlen von Basisfunktionen. Wie bei HMMs steigen die Erkennungsraten, wenn zwischen den Phasen 6 und 14, 7 und 15 sowie 8 und 16 nicht unterschieden wird (ohne Rahmenbedingungen).

Tabelle 4.10 beinhaltet die Erkennungsraten der obigen RBFNs im Vergleich mit denen der Elman-Netzen sowie der HMMs aufgeschlüsselt nach den einzelnen Bewegungsphasen. Deutlich wird, dass beide Modelle Erkennungsprobleme in den Phasen 6 und 8 besitzen. Die RBFNs liefern jedoch deutlich bessere Erkennungsraten. Eine Unterscheidung der Merkmale durch RBFs ist wesentlich verlässlicher als die Separierung durch Hyperebenen bei Elman-Netzen. Die Ergebnisse der RBFNs und der HMMs sind gleich gut. Dies wird auch durch den Vergleich bei konkreten Datensätzen belegt: bei den Abbildungen 4.20 und 4.18 bzw. 4.21 und 4.19 handelt es sich jeweils um die gleichen Datensätze. Ein



**Abbildung 4.21:** Ergebnisse der Phasenklassifikation mit RBFNs mit 1000 RBFs. Die Erkennungsrate für diese Bewegung war eine der höchsten verglichen mit allen Testdatensätzen. Blaue Kurve: Klassifikationsergebnis. Rote Kurve: Vorgabe aus manueller Klassifikation.

Vergleich der Parameter der RBFs der RBFNs mit den Parametern der Mischverteilungskomponenten der HMMs zeigen klare Übereinstimmung. Dies lässt sich dadurch erklären, dass es sich bei den Mischverteilungskomponenten um eine Verallgemeinerung der RBFs handelt.

Beide Ansätze haben bei den gleichen Bewegungsphasen Schwierigkeiten mit der Erkennung. Bei den Phasen handelt es sich um diejenigen mit kurzer Länge, großer Variation bei der Bewegungsdurchführung aufgrund von Rahmenbedingungen, großer Variation zwischen verschiedenen Personen oder solchen mit hoher Ähnlichkeit zu anderen Bewegungsphasen.

Die Tatsache, dass bereits Beobachtungssequenzen der Länge 1 bzw. ein Zustand für eine verlässliche Erkennung der Bewegungsphasen ausreichend ist, ist ein weiterer Hinweis auf die Ähnlichkeit der zwei Modelltypen. Die Generierung von Merkmalen, welche die charakteristischen Informationen über die Bewegungsphase enthalten, sind Voraussetzung für dieses Verhalten. Beide Modelle lernen die Charakteristika der elementaren Bewegungsphasen auswendig. Dennoch bilden die Modelle die Grundlage für die Erkennung komplexerer Bewegungsmodelle, wie in Abschnitt 4.7 bei der Komposition von HMMs erläutert wird. Bei diesen Modellen ist ein höherer Grad der Generalisierung mit den gegebenen Trainingsdaten notwendig.

## 4.6 Geeignete Strukturen und Parameter für HMMs

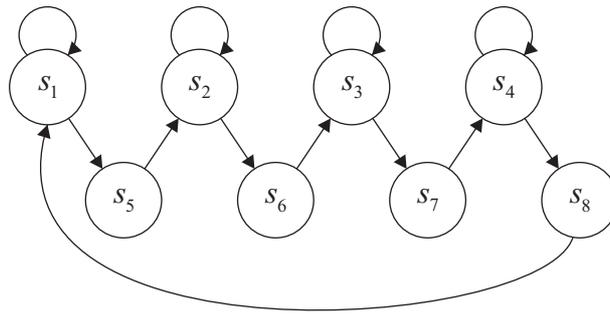
Bei den Tests mit HMMs zur Bewegungsanalyse in den vorigen Abschnitten wurden die versteckten Zustände noch nicht bei der Klassifikation berücksichtigt. Wie in Kapitel 3

Bewegungsphase	Erkennungsrate Elman-Netze	Erkennungsrate RBFNs	Erkennungsrate HMMs
1	0,4980	0,9610	0,9426
2	0,8372	0,9580	0,9477
3	0,7022	0,9406	0,9322
4	0,8591	0,9714	0,9750
5	0,6581	0,8780	0,8682
6	0,0020	0,5024	0,6375
7	0,6520	0,8240	0,8066
8	0,2658	0,7065	0,6248
9	0,5169	0,9182	0,8742
10	0,4663	0,6784	0,8100
11	0,6604	0,9253	0,9444
12	0,7101	0,8554	0,9061
13	0,6676	0,9581	0,8813
14	0,5590	0,6509	0,7607
15	0,2799	0,8141	0,8659
16	0,3440	0,7680	0,9929

**Tabelle 4.10:** Vergleich der Erkennungsraten von Elman-Netzen, RBFNs und HMMs für einzelne Bewegungsphasen.

gezeigt, können die Zustandsverläufe jedoch ausgezeichnet für die Analyse von Anthropomatikdaten herangezogen werden. Entscheidend dabei ist es, neben der bereits erklärten Auswahl von Merkmalen als Beobachtungen auch die Strukturen und Parameter der HMMs so zu wählen, dass die verborgenen Zustände die gesuchte, nicht direkt messbare Größen des Systems repräsentieren. Auf diese Problemstellung wird im Folgenden am Beispiel der Identifikation von Bewegungsphasen eingegangen.

Es werden mit Hilfe von Bewegungsstudien charakteristische Übergangswahrscheinlichkeiten zwischen Teilbewegungen ermittelt und diese Kenntnisse beim Aufbau von der Übergangsmatrix  $\mathbf{A}$  berücksichtigt. Dies betrifft sowohl die a-priori-Initialisierung von  $\mathbf{A}$  als auch die Spezialisierung beim Training. Genauso wichtig ist eine vergleichbare Anpassung der Beobachtungswahrscheinlichkeiten, da sich gezeigt hat, dass die Leistungsfähigkeit eines HMMs auch stark von diesen Parametern abhängt (siehe [129]). Neben den statistischen Erfahrungswerten können Rahmenbedingungen an die Strukturen und Parameter von HMMs aufgrund von biomechanischen Einschränkungen gegeben werden. Die detaillierte Analyse der Matrixstrukturen ermöglicht zusätzlich, wieder Rückschlüsse auf charakteristische Bewegungsübergänge zu ziehen oder auch äquivalente Übergangsbereiche zu finden, so dass die Auswertung effizienter erfolgen kann und evtl. Mehrdeutigkeiten bei der Rekonstruktion der Zustandsfolgen  $Q$  eliminiert werden können.



**Abbildung 4.22:** Zyklisches Modell für Gehbewegungen.

Wie in Abschnitt 4.4.3 beschrieben, können Bewegungen in charakteristische Bewegungsphasen unterteilt werden. Das Ziel beim Design von geeigneten HMMs für die Bewegungsanalyse ist die Repräsentation der Bewegungsphasen durch die Zustände des Modells und deren korrespondierenden zeitlichen Ablauf durch die Struktur des Automaten bzw. der Adjazenzstruktur der Übergangsmatrix  $\mathbf{A}$ . Das Wissen über die Parameter kann nur auf Basis von empirischen Daten aus detaillierten Bewegungsstudien gewonnen werden. Insbesondere hängen die Parameter auch von den extrahierten Merkmalen ab (siehe Abschnitt 4.4.3).

Am Beispiel der Gangbewegungen aus Abschnitt 4.4.3 sollen die Entwurfsprinzipien für HMMs und die Anwendung der Methoden demonstriert werden. Aufgrund der strengen Abfolge der einzelnen Bewegungsphasen sollten die Zustandsübergangsmatrizen  $\mathbf{A}$  Links-Rechts-Modelle oder Bakis-Modelle (siehe Abschnitt 3.2 mit Abb. 3.1) repräsentieren. Aufgrund der zyklischen Eigenschaft von Gehbewegungen muss zusätzlich die Schleife zwischen dem ersten Zustand und dem Finalzustand des Modells geschlossen werden. Das Modell ist so unabhängig davon, in welcher Phase es sich zu Beginn einer Beobachtung befindet. Bei azyklischen Bewegungen, wie z. B. beim Greifen, darf die Zustandskette hingegen nicht geschlossen werden.

Da nicht nur die Bewegungsphasen, sondern auch die Übergänge zwischen den Phasen wichtig für die Bewegungsanalyse sind, werden diese Charakteristika ebenfalls in das Modell integriert. Hierzu werden Hilfszustände definiert, die nur einmal betreten und im nächsten Zeitschritt spontan wieder verlassen werden. Das Modell verweilt nicht in einem der Hilfszustände. Erzielt wird dieses Verhalten über Zustandsübergangswahrscheinlichkeiten  $a_{i,i} = 0$  für einen Hilfszustand mit Index  $i$ . Andernfalls können die Merkmale, welche die Phasenübergänge anzeigen (z. B. Nulldurchgänge von Wavelet-Koeffizienten), nicht in Betracht gezogen werden. Die Einführung der Hilfszustände stellt somit nicht nur ein technisches Hilfsmittel zur Vereinfachung der Modellkonstruktion dar, sondern ist auch notwendig, wenn Zustandsübergänge bei Beobachtung ausgezeichneter Merkmale erzwungen werden sollen.

Da ein Gehzyklus in 4 Phasen unterteilt werden kann, ergibt sich nach Einführung der Hilfszustände ein Zustandsmodell mit  $N_S = 8$  Zuständen (4 Bewegungsphasen und 4 Phasenübergänge). Das endgültige Zustandsdiagramm ist in Abb. 4.22 gezeigt, wobei die



die sigmoidale Transferfunktion zu erhalten. Im Gegensatz dazu sind genaue Winkel und Krümmungswerte für die ungeraden Zustände (Phasenübergänge) eher irrelevant, so dass die zugehörigen Varianzen vergrößert werden können. Diese Anpassungen wurden hier heuristisch durchgeführt und sind in den Kovarianzmatrizen für die Beobachtungen in Zuständen der Bewegungsphasen

$$\mathbf{U}_{i,1} = \begin{pmatrix} 5 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 5 \end{pmatrix} \quad \text{mit } i \in \{1, 3, 5, 7\} \quad (4.10)$$

und für die Beobachtungen in Zuständen der Bewegungsübergänge

$$\mathbf{U}_{i,1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0,2 \end{pmatrix} \quad \text{mit } i \in \{2, 4, 6, 8\} \quad (4.11)$$

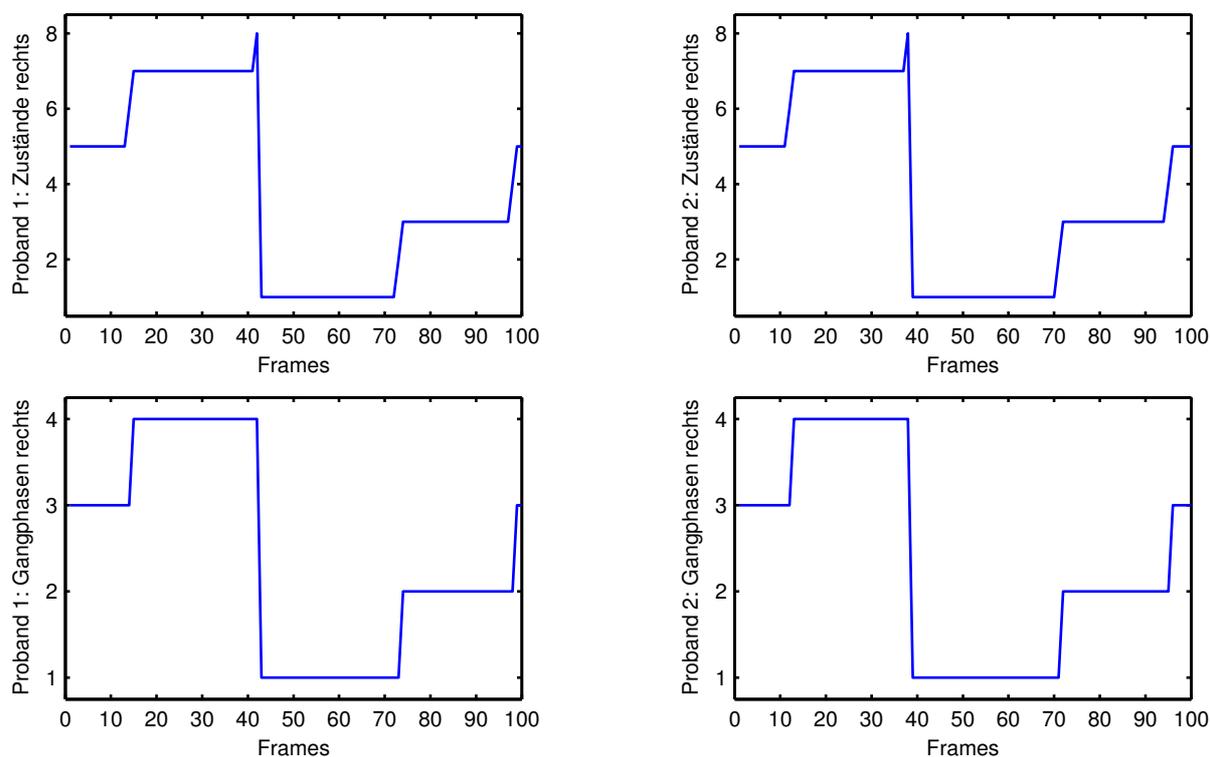
zu erkennen.

Um das beschriebene Gehmodell zu testen, wurden Testdaten mit ihm analysiert. Neben anderen Messungen enthalten diese Daten die Kniewinkeltrajektorien von 11 Probanden bei 4 Gehgeschwindigkeiten auf dem Laufband (1,6 m/s, 1,8 m/s, 2,2 m/s und 2,6 m/s) sowohl für das linke als auch das rechte Bein. Mit dem anfänglich definierten Modell wurde in 89 % der Testdatensätze eine korrekte Erkennung des Phasenverlaufs der Gangbewegung, wie sie in Abschnitt 4.4.3 definiert wird, erzielt. Eine zusätzliche Verfeinerung der Modellparameter mit Hilfe von EM-Lernen machte das Modell robuster, sodass mit dieser Modifikation bei allen Datensätzen ein korrekter Phasenverlauf erkannt und somit eine Erkennungsrate von 100 % erreicht wurde. Die hier angegebenen Erkennungsraten ergeben sich aus der qualitativen Bewertung der Phasenverläufe durch die an der Untersuchung beteiligten Sportwissenschaftler.

Das Modell ist unabhängig von der einleitenden Bewegungsphase und der Anzahl von Gangzyklen im Datenmaterial. Daher kann selbst für Teilsequenzen mit 25 % eines Gesamtzyklus eine korrekte Phasendetektion erzielt werden. Abbildung 4.23 zeigt die Zustände  $s_{i=1,\dots,8}$  des Modells bei der Erkennung der in Abschnitt 4.4.3 erzeugten Merkmalswerte. Um auf die tatsächlichen vier Gangphasen  $p_{j=1,\dots,4}$  zu schließen, wurden die zusätzlichen Hilfszustände  $s_{2,4,6,8}$  für Spontanübergänge wieder eliminiert. Dies erfolgt über die Umrechnung der Zustandsindizes  $i$  in Phasenindizes  $j = \lfloor (i+1)/2 \rfloor$ . Abbildung 4.24 zeigt im Vergleich die Ergebnisse der Phasenbestimmung für das linke Bein. Deutlich ist zu erkennen, dass sich die Bewegungsphasen ähneln, jedoch die zu erwartende Phasenverschiebung mit der halben Länge eines Gangzyklus zwischen dem rechten und dem linken Bein auftritt.

## 4.7 Komposition von HMMs

Um nicht für jede zu analysierende Bewegung ein HMM mit Hilfe des Baum-Welch-Algorithmus zu trainieren oder von Hand wie im vorigen Abschnitt zu definieren, sind

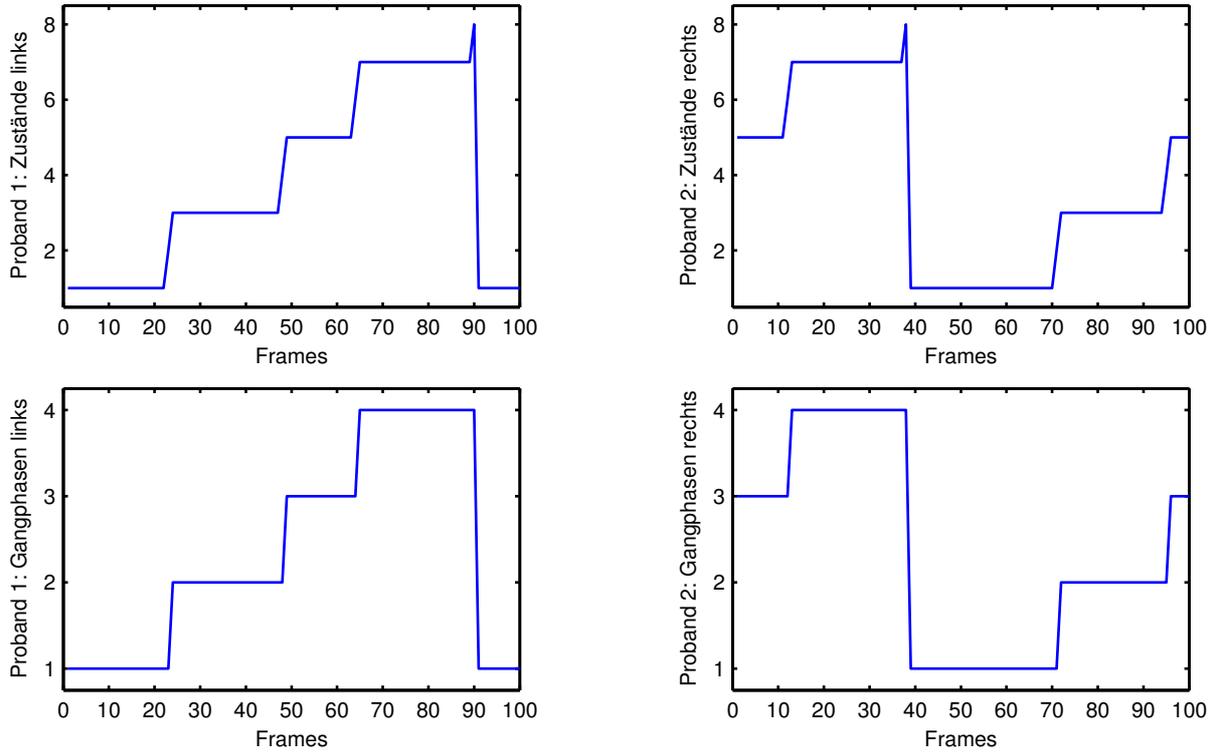


**Abbildung 4.23:** Oben: berechnete HMM-Zustände zur Bestimmung der Bewegungsphasen des rechten Beines zweier Probanden (rechts und links). Unten: Erkannte Bewegungsphasen nach Elimination der Zustände für Spontanübergänge.

automatische Verfahren zur Generierung von HMMs erforderlich. In dieser Arbeit wurde ein solches Verfahren entwickelt, das sich an den elementaren Bewegungsphasen orientiert und aus Modellen zu diesen Schlüsselbewegungen ein komplexes Modell für eine aufwendigere Bewegung produziert.

Bei der Klassifikation der Bewegungsphasen in Abschnitt 4.5.6 wurden die Ausgaben unterschiedlicher HMMs mittels des Majoritätsprinzips verglichen und damit das wahrscheinlichste Modell bzw. die zugehörige Klasse ausgewählt. Mit der neuen Erkennungsmethode in diesem Abschnitt werden explizit die versteckten Zustände der HMMs zur Zuordnung von Bewegungsphasen verwendet. Bei dem Verfahren werden die gleichen Grundprinzipien zur Auswertung von Zeitserien verwendet, wie sie in Abschnitt 3.4.6 vorgestellt werden.

Wie in Abschnitt 4.5.6 beschrieben, liefern die HMMs mit einer sehr geringen Anzahl von Zuständen bereits verlässliche Klassifikationsergebnisse für die Bewegungsphasen für die sie trainiert wurden. Assoziiert man die Zustände mit Phasen, so lassen sich die Modelle nicht weiter durch Reduktion von Zuständen vereinfachen, sie sind *elementar*. Eine komplexe Bewegung die aus den untersuchten Bewegungsphasen zusammengesetzt ist, sollte



**Abbildung 4.24:** Ergebnisse der Phasenerkennung für das linke Bein im Vergleich zu denen für das rechte Bein in Abb. 4.23.

sich auch mit einem komplexeren Modell, das aus den elementaren Modellen zusammengesetzt ist, analysieren lassen. Mit Hilfe der Parameter der elementaren HMMs werden im Folgenden die Parameter für solche komplexe HMMs konstruiert.

In Anlehnung an die bisherige Bezeichnung von HMMs werden die zu konstruierenden komplexen HMMs im Folgenden mit  $\lambda$  bezeichnet und mit den üblichen Parametern  $\boldsymbol{\pi}, \mathbf{A}, \mathbf{U}, \boldsymbol{\mu}, \mathbf{w}$  definiert. Zur Unterscheidung werden die HMMs für die elementaren Bewegungen und ihre Parameter mit den Indizes der Bewegungsphasen versehen:  $\lambda_{p=1, \dots, N_P} = (\boldsymbol{\pi}_p, \mathbf{A}_p, \mathbf{U}_p, \boldsymbol{\mu}_p, \mathbf{w}_p)$ . Die Parameter dieser Modelle werden unverändert aus den Modellen zur Erkennung der Bewegungsphasen in Abschnitt 4.5.6 nach dem Training übernommen.

### 4.7.1 Der einfache Kompositionsalgorithmus

Bei der Komposition der elementaren HMMs  $\lambda_p$  zu einem komplexen HMM  $\lambda$  werden zuerst die Zustandsübergangsmatrizen  $\mathbf{A}_p$  der elementaren HMMs zu einer Gesamtmatrix  $\mathbf{A}$  zusammengeführt. Zur Vereinfachung wird angenommen, dass die elementaren HMMs alle die gleiche Anzahl von Zuständen  $N_S$  besitzen, d. h.  $\mathbf{A}_{p=1, \dots, N_P} \in \mathbb{R}^{N_S \times N_S}$ . Um die  $\mathbf{A}_p$  als Teilblöcke aufzunehmen, wird  $\mathbf{A} \in \mathbb{R}^{N_P N_S \times N_P N_S}$  initialisiert.



Insgesamt gilt damit für die Zeilen  $i = N_S, 2N_S, \dots, N_P N_S$  von  $\mathbf{A}$

$$\sum_{j=1}^{N_P N_S} a_{i,j} = \frac{1}{T_i + 1} \left( T_i \left( \sum_{j=1}^{N_S} a_{\lfloor i/N_S \rfloor + 1; N_S, j} \right) + 1 \right) = \frac{1}{T_i + 1} (T_i + 1) = 1 \quad (4.14)$$

und für die sonstigen Zeilen

$$\sum_{j=1}^{N_P N_S} a_{i,j} = \sum_{j=1}^{N_S} a_{\lfloor i/N_S \rfloor + 1; N_S, j} = 1 \quad (4.15)$$

wonach die Normalisierungsbedingung für eine semistochastische Matrix erfüllt ist.

Die initiale Zustandsverteilung  $\boldsymbol{\pi} \in \mathbb{R}^{N_P N_S}$  sowie die Mischverteilungsparameter zur Berechnung der Beobachtungswahrscheinlichkeiten  $\mathbf{U} \in \mathbb{R}^{D \times D \times N_P N_S \times N_M}$ ,  $\boldsymbol{\mu} \in \mathbb{R}^{D \times N_P N_S \times N_M}$  und  $\mathbf{w} \in \mathbb{R}^{N_P N_S \times N_M}$  ergeben sich aus der einfachen Komposition der Parameter der elementaren HMMS für alle Zustände  $i = 1, \dots, N_P N_S$ :

$$\pi_i = \pi_{\lfloor (i-1)/N_S \rfloor + 1; ((i-1) \bmod N_S)}, \quad (4.16)$$

$$U_{1\dots D, 1\dots D, i, 1\dots N_M} = U_{\lfloor (i-1)/N_S \rfloor + 1; 1\dots D, 1\dots D, ((i-1) \bmod N_S) + 1, 1\dots N_M}, \quad (4.17)$$

$$\mu_{1\dots D, i, 1\dots N_M} = \mu_{\lfloor (i-1)/N_S \rfloor + 1; 1\dots D, ((i-1) \bmod N_S) + 1, 1\dots N_M}, \quad (4.18)$$

$$w_{i, 1\dots N_M} = w_{\lfloor (i-1)/N_S \rfloor + 1; ((i-1) \bmod N_S) + 1, 1\dots N_M}, \quad (4.19)$$

wobei  $D$  die Dimension des Merkmalsraums für die Beobachtungen und  $N_M$  die Anzahl der Mischverteilungskomponenten bezeichnet. Man beachte, dass das so konstruierte HMM nicht mehr trainiert werden muss, da bereits die elementaren Modelle eintrainiert sind. Das HMM ermöglicht die Erkennung einer Bewegung mit einer strikten Abfolge von elementaren Schlüsselbewegungen.

## 4.7.2 Erweiterungen für alternative Zustandsabfolgen

Die im vorigen Abschnitt vorgestellte Konstruktion lässt sich dahingehend erweitern, dass das resultierende Modell nicht nur eine strikte Abfolge von Bewegungsphasen erkennen kann, sondern auch alternative Teilfolgen in einer Bewegung zulässt. Dies wird u. a. dann benötigt, wenn Teilbewegungen signifikante intra-individuelle Unterschiede bei der Bewegungsausführung besitzen und diese detektiert werden sollen. Beispielsweise unterscheiden sich die Daten des Bewegungskataloges aus Abschnitt 4.5.1 in den Bewegungsphasen in denen die Kaffeekanne manipuliert wird aufgrund des Füllstandes der Kanne. Entsprechend werden künstlich zusätzliche Bewegungsphasen definiert, die alternativ zum gleichen Zeitpunkt in der Bewegungsabfolge auftreten. Allgemein kann die Abfolge von elementaren Bewegungsphasen mit Hilfe einer Adjazenzmatrix  $\check{\mathbf{A}} \in \{0, 1\}^{N_P \times N_P}$  oder aber auch z. B. mit Hilfe einer kontextfreien Grammatik spezifiziert werden.

Das Konstruktionsverfahren wird so erweitert, dass über zusätzliche Zustandsübergänge nicht nur in eine Folgephase sondern in mehrere alternative Bewegungsphasen gesprungen





**Algorithmus 4.7.1** Komposition elementarer HMMs.

---

```

// Parameter:
//  $\lambda_{1,\dots,p}$ : elementare HMMs mit  $\lambda_p = (\boldsymbol{\pi}_p, \mathbf{A}_p, w_p, \boldsymbol{\mu}_p, \mathbf{U}_p)$ 
//  $T_{1,\dots,p}$ : mittlere Längen der Bewegungsphasen
//  $\check{\mathbf{A}} \in \{0, 1\}^{N_P \times N_P}$ : Adjazenzmatrix mit Einträgen  $\check{a}_{p,q} = 1$ , wenn es einen
// Übergang von Phase  $p$  nach Phase  $q$  gibt, und  $\check{a}_{p,q} = 0$  sonst
// Rückgabe:
//  $\lambda$ : zusammengesetztes HMM
 $N_P$  := Anzahl Phasen;
 $D$  := Dimension der Beobachtungsvektoren;
 $N_S$  := Anzahl Zustände in  $\lambda_{1,\dots,p}$ ;
 $N_M$  := Anzahl Mischverteilungskomponenten in  $\lambda_{1,\dots,p}$ ;
 $\mathbf{A} := 0^{N_P N_S \times N_P N_S}$ ; // neue Zustandsübergangsmatrix
for( $p := 1, \dots, N_P$ ) {
  // Matrixblock  $\mathbf{A}_p$  in  $\mathbf{A}$  eintragen:
   $h := (p - 1)N_S$ ; // Hilfsindex
  for( $n_1 := 1, \dots, N_S$ )
    for( $n_2 := 1, \dots, N_S$ )
       $a_{h+n_1, h+n_2} := a_{n_1, n_2}$ ;
  // Übergangswahrscheinlichkeiten anpassen:
   $C_{\text{norm}} := 1/(T_p + 1)$ ; // Normalisierungsfaktor
   $a_{h+N_S, h+N_S} := T_p/C_{\text{norm}}$ ;
   $n := \sum_{q=1, \dots, N_P} \check{a}_{p,q} - 1$ ; // Anzahl alternativer Phasen
  for( $q := 1, \dots, N_P$ )
    if( $\check{a}_{p,q} = 1 \wedge p \neq q$ )
       $a_{h+N_S, (q-1)N_S+1} := 1/(nC_{\text{norm}})$ ; // entspricht  $a_p/n$ 
  // initiale Zustandsverteilung und Parameter für Mischverteilungen:
  for( $n_1 := 1, \dots, N_S$ ) {
     $i := h + n_1$ ;
     $\pi_i := \pi_{p; n_1}$ ;
    for( $m := 1, \dots, N_M$ ) {
       $w_{i,m} := w_{p; n_1, m}$ ;
      for( $d_1 := 1, \dots, D$ ) {
         $\mu_{d_1, i, m} := \mu_{p; d_1, n_1, m}$ ;
        for( $d_2 = 1, \dots, D$ )
           $U_{d_1, d_2, i, m} := U_{p; d_1, d_2, n_1, m}$ ;
      }
    }
  }
}
return  $\lambda = (\boldsymbol{\pi}, \mathbf{A}, \mathbf{w}, \boldsymbol{\mu}, \mathbf{U})$ ;

```

---

Phase $p$	$T_p$	$a_p$	Phase $p$	$T_p$	$a_p$
1	111,03	0,0089	9	85,86	0,0115
2	127,94	0,0078	10	78,89	0,0125
3	84,15	0,0117	11	185,93	0,0053
4	121,98	0,0081	12	74,91	0,0132
5	96,64	0,0102	13	111,88	0 (0,0089)
6	24,45	0,0393	14	36,09	0,0270
7	145,99	0,0068	15	88,99	0,0111
8	75,49	0,0131	16	77,34	0,0128

**Tabelle 4.11:** Mittlere Längen  $T_p$  und Übergangswahrscheinlichkeiten  $a_p = \frac{1}{T_{p+1}}$  der Phasen  $p = 1, \dots, 16$ . Phase 13 ist Finalphase und hat daher die Übergangswahrscheinlichkeit  $a_{13} = 0$ .

von HMM-Zuständen zu den gesuchten Zuständen in den Anthropomatikdaten vorgenommen werden. Die HMM-Komposition löst damit auch das Problem der Abbildung von Zuständen auf gesuchte Systemgrößen. Im Falle der Ultraschallbilder aus Abschnitt 3.4.6 ist dies die gewünschte Interpretation der Gewebetypen.

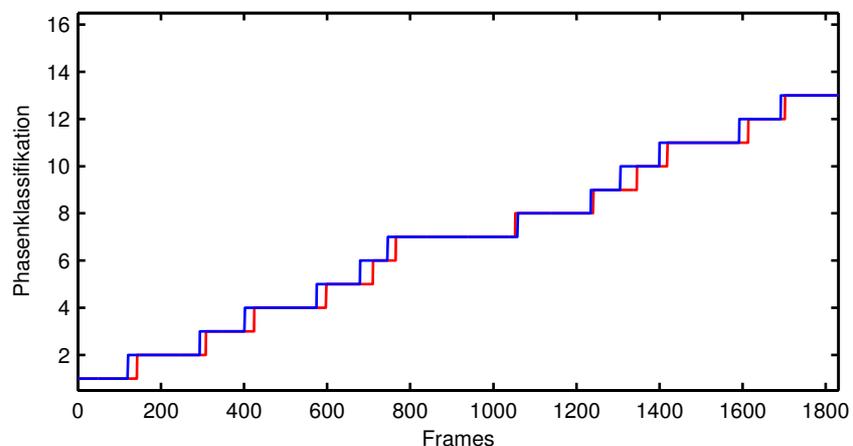
Ein weiterer Vorteil bei der Analyse von Bewegungen mit den zusammengesetzten HMMs ist, dass keine Fensterung der zu untersuchenden Beobachtungssequenzen mehr nötig ist, wie dies in Abschnitt 4.5.6 der Fall ist. Die Klassifikation der Bewegungsphasen kann mit Beobachtungssequenzen beliebiger Länge analog zur Klassifikation von Personen oder Rahmenbedingungen in den Abschnitten 4.5.2 und 4.5.4 erfolgen. Der Vorteil macht sich insbesondere im Rechenaufwand bemerkbar. Liegt eine Beobachtungssequenz der Länge  $T$  vor und sollen  $N_P$  Phasen durch  $N_P$  elementare HMMs mit jeweils  $N_S$  Zuständen unterschieden werden, so besitzt das Erkennungsverfahren mit Majoritätsprinzip einen Rechenaufwand von  $\mathcal{O}(N_P T T_W N_S^2)$ , wobei  $T_W$  die Länge der verwendeten Fenster ist. Der Aufwand ergibt sich aus dem Aufwand für die Klassifikation der Teilbewegungen in einem Fenster durch den Vorwärts-Algorithmus (siehe Abschnitt 2.3.3.1), die für alle Zeitpunkte und Phasen durchgeführt werden muss. Im Gegensatz dazu besitzt die Analyse der Bewegungsphasen mit einem zusammengesetzten HMM den Aufwand  $\mathcal{O}(T(N_P N_S)^2)$ . Dieser ergibt sich aus der einmaligen Anwendung des Viterbi-Algorithmus (siehe Abschnitt 3.3). Die Anzahl der Bewegungsphasen geht zwar quadratisch in den Aufwand ein, jedoch ist die Zahl nicht sonderlich groß (die komplexen Bewegungen in der vorliegenden Bewegungsdaten kommen mit  $N_P = 16$  Phasen aus) und die Übergangsmatrix  $\mathbf{A}$  ist dünn besetzt (sie besitzt nahezu Diagonalgestalt), so dass bei der Auswertung etliche Zustandsübergänge nicht berechnet werden müssen. Die Größe  $N_P$  hat weit weniger Einfluss als die Länge  $T_W$  bei der Fensterung. Geht man von einer linearen Abhängigkeit zwischen  $T_W$  und der wesentlichen Einflussgröße  $T$  aus, so besitzt der in dieser Arbeit entwickelte Algorithmus eine wesentlich höhere Laufzeiteffizienz als das herkömmliche Klassifikationsverfahren.

Neben den genannten Vorteilen zeichnen sich die zusammengesetzten HMMs durch eine verlässlichere Klassifikation der Beobachtungen und damit auch durch höhere Erkennungsraten im Vergleich zur Klassifikation mit dem Majoritätsprinzip aus. Tabelle 4.12 gibt die Raten für die Erkennung der Bewegungsphasen in den Bewegungsdaten aus Abschnitt 4.5.1 an. Diese sind mit den Ergebnissen aus Abschnitt 4.5.6 vergleichbar. Die Ähnlichkeit wird auch durch den Vergleich der Phasenverläufe anhand der Diagramme in den Abbildungen 4.25 und 4.26 beim Vergleich mit den Verläufen in den Abbildungen 4.18 und 4.19 ersichtlich.

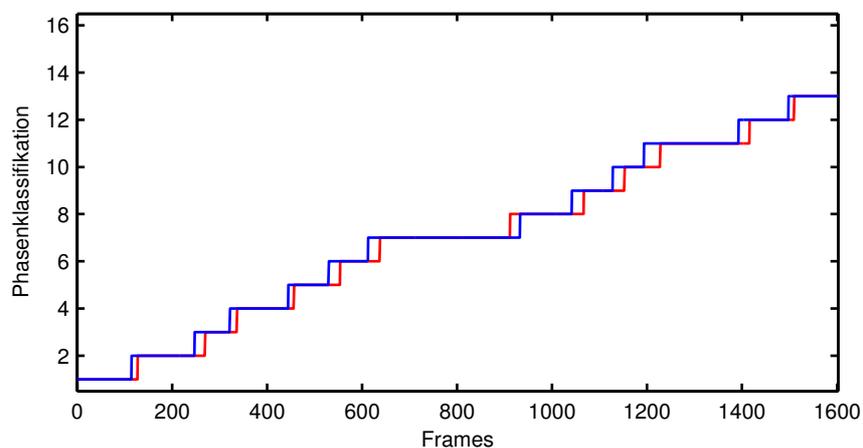
Zustände $N_S$ der elem. HMMs	Mischvert.- komponenten $N_M$	Erkennungs- rate	ohne Rahmen- bedingungen
10	1	0,28956	0,29685
10	2	0,28957	0,29686
1	1	0,82587	0,84346
1	2	0,82587	0,84346
20	1	0,82205	0,83420
2	1	0,81772	0,82830
2	2	0,81772	0,82830
3	1	0,81164	0,82446
3	2	0,80303	0,80860
4	1	0,81487	0,82718
4	2	0,80128	0,80575
5	1	0,75685	0,76716
5	2	0,75685	0,76716

**Tabelle 4.12:** Erkennungsraten bei der Klassifikation von Bewegungsphasen durch zusammengesetzte HMMs mit unterschiedlichen Parametrisierungen (Anzahl der Zustände  $N_S$  und Mischverteilungen  $N_M$  in den elementaren HMMs). Zusätzlich sind die Erkennungsraten angegeben, wenn der Füllstand der Kaffeekanne irrelevant ist, d. h. wenn zwischen den Phasen 6 und 14, 7 und 15 sowie 8 und 16 nicht unterschieden wird (ohne Rahmenbedingungen).

Eine deutliche Verbesserung der Erkennungsrate weisen die zusammengesetzten HMMs bei der Erkennung des Füllstandes der Kaffeekanne als Rahmenbedingung bei der Bewegungsausführung auf. Die verbesserten Erkennungsraten sind in Tabelle 4.13 angegeben. Im Vergleich zum Majoritätsprinzip werden die einzelnen Bewegungsphasen nicht isoliert betrachtet, sondern der Zusammenhang und die Abfolge in der gesamten Bewegungssequenz berücksichtigt. Das Modell reagiert somit wesentlich robuster gegenüber starken Variationen in der Bewegungsdurchführung bei gleichzeitig geringen Unterschieden zwischen den Phasen. Versehentliche Sprünge in eine falsche alternative Bewegungsphase durch Bestimmung der falschen maximalen Wahrscheinlichkeit, wie sie in Abbildung 4.18 zu finden sind, treten in Abbildung 4.25 nicht mehr auf.



**Abbildung 4.25:** Ergebnisse der Phasenklassifikation mit einem zusammengesetzten HMM (16·2 Zustände). Der Datensatz entspricht dem von Abb. 4.18. Blaue Kurve: Klassifikationsergebnis. Rote Kurve: Vorgabe aus manueller Klassifikation.



**Abbildung 4.26:** Ergebnisse der Phasenklassifikation mit einem zusammengesetzten HMM (16·2 Zustände). Der Datensatz entspricht dem von Abb. 4.19. Blaue Kurve: Klassifikationsergebnis. Rote Kurve: Vorgabe aus manueller Klassifikation.

Zustände $N_S$ der elem. HMMs	Mischvert.- komponenten $N_M$	Erkennungs- rate
10	1	0,67857
10	2	0,67857
1	1	0,82857
1	2	0,82857
20	1	0,94286
2	1	0,94286
2	2	0,94286
3	1	0,92857
3	2	0,96429
4	1	0,94286
4	2	0,96429
5	1	0,90000
5	2	0,90000

**Tabelle 4.13:** Erkennungsraten bei der Klassifikation des Füllstandes der Kaffeekanne durch zusammengesetzte HMMs mit unterschiedlichen Parametrisierungen (Anzahl der Zustände  $N_S$  und Mischverteilungen  $N_M$  in den elementaren HMMs).

## 4.8 Zusammenfassung

In diesem Kapitel wird gezeigt, dass eine zielgerichtete Merkmalsaufbereitung entscheidend für eine gute Klassifikation und Analyse von Bewegungsdaten ist. Aus akquirierten Bewegungsdaten werden in dieser Arbeit Winkeltrajektorien für die Freiheitsgrade der zu untersuchenden Gelenke berechnet. Mit ihnen lassen sich sowohl inter- als auch intra-individuelle Unterschiede bei der Bewegungsausführung erfassen und benötigen keine aufwendige Berücksichtigung der Rahmenbedingungen, wie dies z. B. bei Trajektorien mit Raumkoordinaten der Fall ist. Die Verwendung von Winkelverläufen bildet die Grundlage für eine robuste Merkmalsgenerierung.

Die Wavelet-Darstellung von Gelenkwinkeln ermöglicht die gleichzeitige Untersuchung von Bewegungssignalen auf unterschiedlichen Skalierungsstufen. Dies entspricht auch der Bewegungswahrnehmung des Menschen, der sowohl den Grobverlauf einer Bewegung als auch Charakteristika in Teilen der Bewegung betrachtet. Der zusätzliche Vorzug der Wavelet-Darstellung ist, dass mit den Wavelet-Koeffizienten signifikante Zusatzinformationen über den Bewegungsverlauf geliefert werden, die mit Hilfe der originalen Bewegungsmerkmale nicht erkennbar sind. Am Beispiel der Gangbewegung wird dies belegt.

Die entwickelten erweiterten Bewegungsmerkmale zeichnen sich besonders als Beobachtungswerte für HMMs aus. Erst durch sie lassen sich Schlüsselbewegungen oder elementare

Bewegungsphasen mit Hilfe der versteckten Zustände der HMMs eindeutig repräsentieren. Der Grund dafür ist, dass die manuell oder durch Training definierten Zustandsübergangs- und Beobachtungswahrscheinlichkeiten der Modelle den Verlauf der Zustandssequenz bei anliegenden Beobachtungen verlässlich steuern.

Anhand einer Bewegungsstudie mit einer komplexen Alltagsbewegung wird gezeigt, dass sich HMMs sehr gut zur Erkennung von personenspezifischen Charakteristika eignen. Ebenso können Veränderungen in den Rahmenbedingungen bei der Bewegungsausführung erkannt werden, wie dies am Beispiel des Füllstandes einer Kaffeekanne bei der Bewegungsdurchführung gezeigt wird. Mit diesen Informationen kann beispielsweise das kognitive Modell, das ein Roboter von seiner Umwelt hat, bereichert werden. Dasselbe gilt auch für die Identifikation von elementaren Bewegungsphasen in der Gesamtbewegung, die ebenfalls verlässlich mit HMMs durchgeführt werden kann und z. B. die essentiellen Aussagen über den Bewegungszustand eines Benutzers für die kooperative Handlungsplanung eines Roboters liefert.

Der Vergleich der Klassifikationsergebnisse von HMMs mit verschiedenen Ausprägungen von Neuronalen Netzen ergibt, dass HMMs bessere Erkennungsraten liefern, sobald der Verlauf einer Bewegung erfasst werden soll. Wie in den vorigen Kapiteln bereits beschrieben, ist der Grund hierfür, dass HMMs die Historie in Beobachtungssequenzen in den versteckten Zuständen berücksichtigen können. Selbst die implizite Zustandspeicherung per Rückkopplung der Ausgabe in Elman-Netzen ist dafür nicht ausreichend. So ist die Erkennung des Füllstandes der Kaffeekanne in der Bewegungsstudie dieser Arbeit nur mit HMMs möglich.

HMMs und RBFNs unterscheiden sich bei der Erkennung der Bewegungsphase in den Erkennungsraten nur unwesentlich. Die Erklärung hierfür ist, dass es sich bei der Klassifikation durch das Majoritätsprinzip mit HMMs für die elementaren Bewegungsphasen eigentlich um eine Mischverteilungsklassifikation handelt. Die Variationen in den einzelnen Bewegungsphasen sind so gering, dass die HMMs nur sehr wenige Zustände benötigen, um signifikante Beobachtungswahrscheinlichkeiten zu liefern. Die Dichtefunktionen zu den Mischverteilungen der HMMs gleichen den Basisfunktionen der RBFNs und reichen zur Bewertung eines einzelnen Zeitpunktes in der Bewegungssequenz aus. Die Bewegungsphasen und die zugehörigen HMMs sind elementar und bilden die Grundlage für das in dieser Arbeit entwickelte Verfahren zur Komposition elementarer HMMs.

Durch die Komposition von elementaren HMMs zu einem komplexen Modell für Bewegungsabläufe, die sich aus elementaren Bewegungen zusammensetzen, kann die Klassifikation mit Hilfe des Majoritätsprinzips umgangen werden. Stattdessen kommt die effizientere Auswertung durch die explizite Bestimmung und Betrachtung der Zustandsfolge zum Einsatz. Die Zustände repräsentieren direkt die gesuchten Bewegungsphasen. Eine Fensterung der Beobachtungssequenz ist nicht mehr notwendig und bei der Analyse kann der vollständige Verlauf der gesamten Bewegung erfasst werden. Neben dem verbesserten Berechnungsaufwand hat das Verfahren den Vorteil, dass das Modell zu einem konsistenten Zustandsverlauf gezwungen wird und damit deutlich weniger Fehlklassifikationen

als bei der konventionellen Methode auftreten. Insbesondere wird dies durch die robuste Erfassung des Füllstandes der Kaffeekanne als Rahmenbedingung bei der Bewegungsausführung belegt.

Hinsichtlich des Einsatzes der Methoden bei der Benutzererfassung und Bewegungsplanung für einen humanoiden Roboter zeichnet sich das entwickelte Kompositionsverfahren dadurch aus, dass sich benötigte Bewegungsmodelle nach einem Baukastenprinzip automatisch generieren lassen. Liegt ein Katalog von HMMs für elementare Bewegungsphasen vor, können diese z. B. durch Angabe von Adjazenzrelationen oder einer kontextfreien Grammatik definiert werden. Somit wird die Konstruktion von HMMs für beliebige Bewegungen auf der Basis eines vorliegenden Bewegungskatalogs ermöglicht.

# Kapitel 5

---

## Zustandserkennung in der Praxis

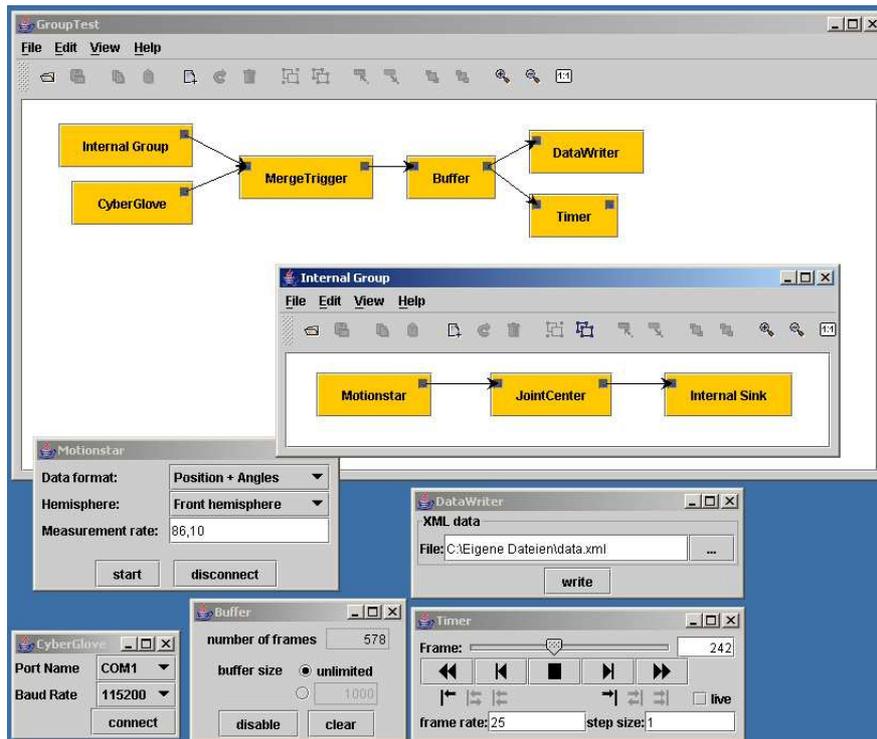
---

In diesem Kapitel wird gezeigt, wie die in dieser Arbeit entwickelten Methoden in eine Applikation integriert und damit auch in der Praxis verwendet werden können. Als Plattform zur Durchführung der Tests, zur Berechnung von Analyseergebnissen und zum Vergleich zwischen unterschiedlichen Erkennungsmethoden diente in den vorigen Kapiteln die Programmierumgebung Matlab [95] der Fa. Mathworks mit der zugehörigen Toolbox für Neuronale Netze [96] sowie weitere frei verfügbare Toolboxen für SVMs [26] und HMMs [112]. Bei den Programmbibliotheken handelt es sich um universell einsetzbare Standardlösungen, die nur begrenzt hinsichtlich bestimmter Aufgabenstellungen optimiert werden können. Um eine solche Optimierung zu gewährleisten und auch die Laufzeiteffizienz der in dieser Arbeit vorgestellten Methoden in der Praxis zu untersuchen, wurden die Verfahren zur Analyse von menschlichen Bewegungen aus Kapitel 4 in das zu einem großen Teil auch in dieser Arbeit entstandene Visualisierungs-, Analyse- und Modellierungs-Framework VAMOS integriert.

### 5.1 Eigenschaften von VAMOS

VAMOS ist ein Visualisierung-, Analyse- und Modellierungssystem, dessen Hauptanwendungsbereich die Analyse von menschlichen Bewegungen ist. Insbesondere stehen die Aufgabenstellungen des SFB 588 „Humanoide Roboter“ [145] im Mittelpunkt der Entwicklung von VAMOS. Durch den konsequenten Einsatz von abstrakten Entwurfsmustern und einer universell einsetzbaren Schnittstellendefinition ist es jedoch möglich, das Framework für allgemeinere Fragestellungen der echtzeitnahen Signalverarbeitung und Signalanalyse zu verwenden und eignet sich daher insbesondere auch zur Analyse von Anthropomatikdaten. Eine ausführliche Beschreibung zu VAMOS findet sich in [21] sowie im Internet auf der Homepage des Projekts [157].

Zentraler Bestandteil von VAMOS bilden *Module* für unterschiedlichste Aufgaben der Datenverarbeitung. In einer graphischen Benutzeroberfläche können die Module vom Benutzer bedient werden und durch Datenflusskanten miteinander verbunden werden. Die graphische Benutzeroberfläche dient ebenfalls der Darstellung der Daten durch einzelne



**Abbildung 5.1:** Verknüpfung und Gruppierung von Modulen auf der Benutzeroberfläche von VAMOS.

Module, wobei konzeptionell das Modul und seine graphische Benutzerschnittstelle getrennt sind, so dass unterschiedliche Sichten auf die gleichen Daten ermöglicht werden. Die Ablaufsteuerung des Systems sorgt zudem dafür, dass die Darstellung eines einzelnen Moduls nur dann aktualisiert wird, wenn es sich in einem definierten Zustand befindet, d. h. über aktuell konsistente Daten verfügt. Die Module lassen sich hierarchisch zu *Gruppen* zusammenfassen und abspeichern. Auf diese Weise können häufig verwendete Module gekapselt und übersichtlich in neue Projekte integriert werden. Abbildung 5.1 zeigt als Beispiel einige auf der Benutzeroberfläche von VAMOS verknüpfte Module und deren graphische Darstellung sowie eine Gruppe von Modulen.

VAMOS verfügt über eine größere Anzahl von grundlegenden Modulen. Im Folgenden ist eine Liste der wichtigsten Modulkategorien angegeben als Übersicht angegeben.

- Erfassung von Sensorwerten, z. B. von einem CyberGlove (siehe Abschnitt 4.3.3) oder einem MotionStar (siehe Abschnitt 4.3.2).
- Pufferung von Daten
- Datenflusskontrolle, z. B. durch kontrollierte Fusionierung von Frames und deren Inhalt

- Interaktion, beispielsweise durch einen *Timer*, der das Abspielen von Frames mit vorgegebenen Taktraten ermöglicht
- Laden und Speichern von Daten in verbreiteten textuellen und binären Formaten sowie in einer eigenen Darstellung mit XML
- Darstellung von Daten, z. B. in tabellarischer Form, als Kurvenplots oder als 3D Visualisierung
- Verteiltes Rechnen durch Verteilung von Modulen auf mehrere Rechner und Kommunikation zwischen diesen durch Client- und Server-Module mit einem CORBA-basierten Protokoll zum Datenaustausch.

Zur Laufzeit von VAMOS können beliebige Module importiert und ausgeführt werden. Jedes Modul läuft dabei in einem eigenen Prozess und einem geschützten Speicherbereich ab. Für die Kommunikation mit anderen Modulen kann jedes Modul ein oder mehrere Ein- oder Ausgänge anbieten. Dabei werden die eingehenden Daten in Form einer Warteschlange verwaltet und das zugehörige Modul über neue oder modifizierte anliegende Daten mittels Ereignisse benachrichtigt. Die Daten selbst werden aus Effizienzgründen per Referenz übergeben, wobei spezielle Mechanismen dafür sorgen, dass die Module dennoch parallel und asynchron ablaufen können. Insbesondere wird ermöglicht, dass Quellmodule Daten an ihrem Ausgang unabhängig vom Fortschritt der angeschlossenen Zielmodule modifizieren, ohne dass dabei benötigte alte Daten überschrieben werden oder auf Zielmodule gewartet werden muss.

Ein wesentlicher Vorteil bei der Entwicklung neuer Module für VAMOS ist, dass während der Laufzeit die Funktionalität einzelner Module modifiziert und neu übersetzt werden kann und die Module unter Beibehaltung der anliegenden Daten reinstanciiert werden können. Die Basis hierfür bietet der Import und Export von Daten und Modulen mit Hilfe von XML und ein eigener dynamischer Lader für die Klassen des Systems.

Abgesehen von den Aufgabenstellungen in dieser Arbeit zeichnet sich VAMOS dadurch aus, dass viele Entwickler durch die strenge Modularisierung unabhängig voneinander Teile einer großen Applikation erstellen und im Framework integrieren können. Diese Eigenschaft hat sich auch in Hinblick auf die Arbeit mit Studenten und (z. B. Praktika und Studienarbeiten) als sehr wertvoll erwiesen. Es kann damit auch gezeigt werden, dass die in dieser Arbeit entwickelten Methoden auch in einem größeren Projekt flexibel einsetzbar sind.

## 5.2 Aufbau und Architektur von VAMOS

VAMOS ist vollständig in Java geschrieben und nutzt einige zusätzliche Bibliotheken wie z. B. Java3D, JMF oder die Java COMM-API. Intern werden die Module von VAMOS durch Objekte vom Typ *Device* repräsentiert. Zu jedem *Device* gibt es eine unabhängige

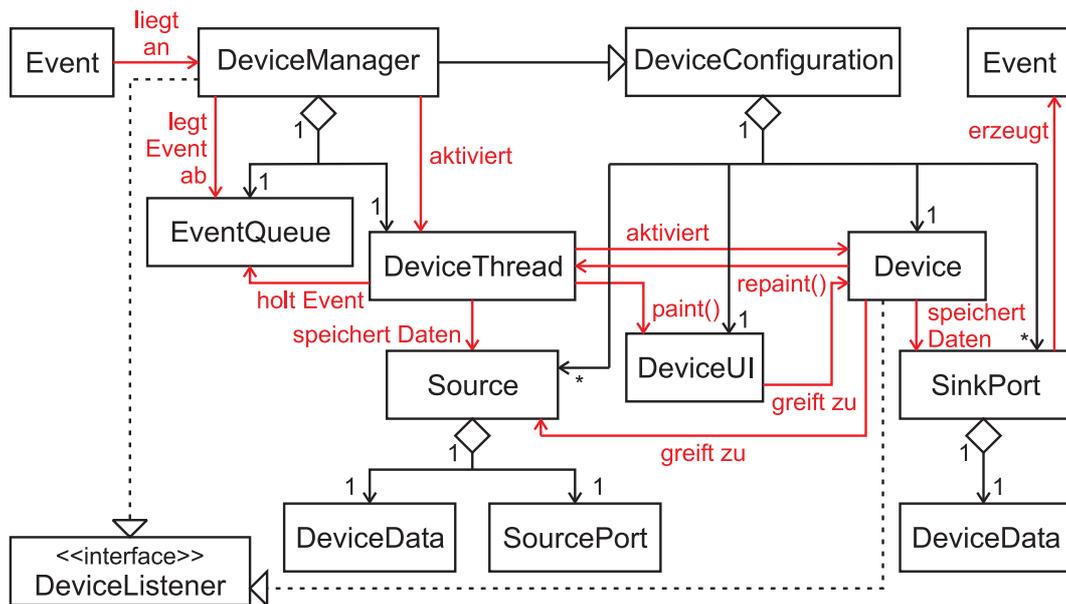


Abbildung 5.2: Schematischer Aufbau eines Moduls in VAMOS.

graphische Benutzerschnittstelle und Darstellung, die durch Instanzen der Klasse `DeviceUI` implementiert wird. Die Objekte von `Device` und `DeviceUI` sind in einem gemeinsamen Kontext zur Laufzeit eingebettet, der so genannten `DeviceConfiguration`. Die `DeviceConfiguration` wird von einem Objekt des Typs `DeviceManager` verwaltet, indem es die Abarbeitung der Ereigniswarteschlange steuert und die Verarbeitung von Daten an den Ein- und Ausgängen der Module kontrolliert. Dies ermöglicht die Kapselung der Modulfunktionalität und die effiziente Verwaltung gemeinsam verwendeter Speicherinhalte. Abbildung 5.2 verdeutlicht den Aufbau von Modulen in VAMOS mit ihrem Kontext in einer schematischen Darstellung.

Im Detail bildet die Klasse `DeviceConfiguration` den Zustand eines Moduls ab und stellt diesem eine Vielzahl an Methoden zur Interaktion mit VAMOS zur Verfügung. Zusätzlich zu den für das Modul sichtbaren Methoden bietet die Klasse `DeviceManager` Funktionen zur Verwaltung der Module an und bildet damit eine Prozesseinheit. Ein von außen ankommendes `Event` aktiviert den zugehörigen `EventListener`, der das `Event` in einer `EventQueue` ablegt und bei Bedarf den `DeviceThread` startet. Dieser verarbeitet die Einträge in der `EventQueue` sukzessive, indem er das Objekt vom Typ `DeviceData` am `SourcePort` aktualisiert und den `DeviceListener` benachrichtigt. Das `Device` verarbeitet die `DeviceData` vom `SourcePort` und legt die Ergebnisse seiner Berechnung an den gewünschten `SinkPort` an. Die `SinkPorts` generieren automatisch zu den Änderungen an den `DeviceData` passende Ereignisse. Der `DeviceThread` startet danach den Bildaufbau im `DeviceUI` über die Methode `paint()`, falls das `Device` ein Neuzeichnen durch die Methode `repaint()` angefordert hat und eine minimale Wartezeit abgelaufen ist.

Erzeugt ein Modul einen Ausgang, so wird mit diesem zugleich eine anfangs leere Liste von Frames assoziiert. Frames beinhalten Daten zu bestimmten Zeitpunkten und eignen sich z. B. damit zum Speichern von Merkmalsvektoren einer Zeitserie. Das Klassendiagramm mit der Klasse `Frame` und deren Zusammenhang mit der Klasse `DeviceData` ist in Abb. 5.3 angegeben. Komponenten eines Frames sind Instanzen der Klasse `FrameElement`, zu der es Unterklassen gibt, die beispielsweise Skalare, Vektoren, Quaternionen, Matrizen oder auch Videobilder implementieren. Welche Elemente sich in einem Frame befinden, wird durch Objekte der Klasse `FrameInfo` definiert. Neben Frames verfügen die Module in VAMOS zusätzlich über Modellwissen. Dieses Modellwissen wird in einem Objekt vom Typ `ModelData` gespeichert. Modelle werden in VAMOS für die unterschiedlichsten Zwecke verwendet. Beispiele reichen von einfachen Informationen über die Darstellung von Frame-Elementen bis hin zu aufwendigen kinematischen Modellen des menschlichen Körpers. Für eine ausführliche Beschreibung sei auf [157] verwiesen.

Jedes Mal, wenn das Modul neue Frames am Ausgang hinzufügt, modifiziert oder löscht, sendet VAMOS an jedes angeschlossene Modul ein entsprechendes `Event`, das wiederum eine Referenz auf die Liste der Frames enthält. Diese Ereignisse werden in die `EventQueue` der Zielmodule gestellt und warten dort wie bereits beschrieben auf die Abarbeitung. Die Methode des Datenaustauschs ist hinsichtlich Speicherverbrauch und Kopieraufwand sehr effizient. Sie bringt jedoch auch das Problem mit sich, dass die Daten von jedem Modul verändert werden können und diese Veränderungen sich auf alle Module auswirken. Zur Lösung dieses Problems wurde in VAMOS die Methode des *partiellen Duplizierens* entwickelt. Hierbei wird jeder Liste von Frames eine Menge von Modulen zugeordnet, die Zugriff auf diese Liste haben. Bei einem Schreibzugriff eines Moduls auf die Liste wird geprüft, ob es noch andere konkurrierende Module gibt, die ebenfalls Zugriff auf die Liste haben. Ist dies der Fall, wird die Liste teilweise kopiert, indem die vom Schreibzugriff nicht betroffenen Frames per Referenz kopiert werden, während die übrigen Frames neu erzeugt und per Wert kopiert werden. Der gesamte Mechanismus wird dabei vollständig von VAMOS abgewickelt und ist für die verwendeten Module transparent. Die Funktionalität des partiellen Duplizierens wird in einer eigenen Klasse `ShareableList` implementiert. Ein Beispiel für das partielle Duplizieren von Frames ist in Abbildung 5.4 zu sehen.

## 5.3 Analyse von Bewegungen in VAMOS

In den folgenden Abschnitten wird dargestellt, wie die Analyse von Bewegungsdaten anhand der in Kapitel 4 beschriebenen Methoden mit VAMOS in Echtzeit durchgeführt werden kann. Hierzu werden die entwickelten Module vorgestellt und anhand eines Beispielszenarios gezeigt, wie erweiterte Bewegungsmerkmale aus gemessenen Sensordaten generiert werden, daraufhin die Phasen der Bewegung automatisch erkannt werden und schließlich die Phaseninformation zur Synthese einer neuen Bewegung eingesetzt wird. Die Umsetzung des Szenarios zur Bewegungsanalyse in ein Modul-Schema und die Benutzeroberflächen zu den darin enthaltenen Modulen sind in Abb. 5.5 dargestellt.

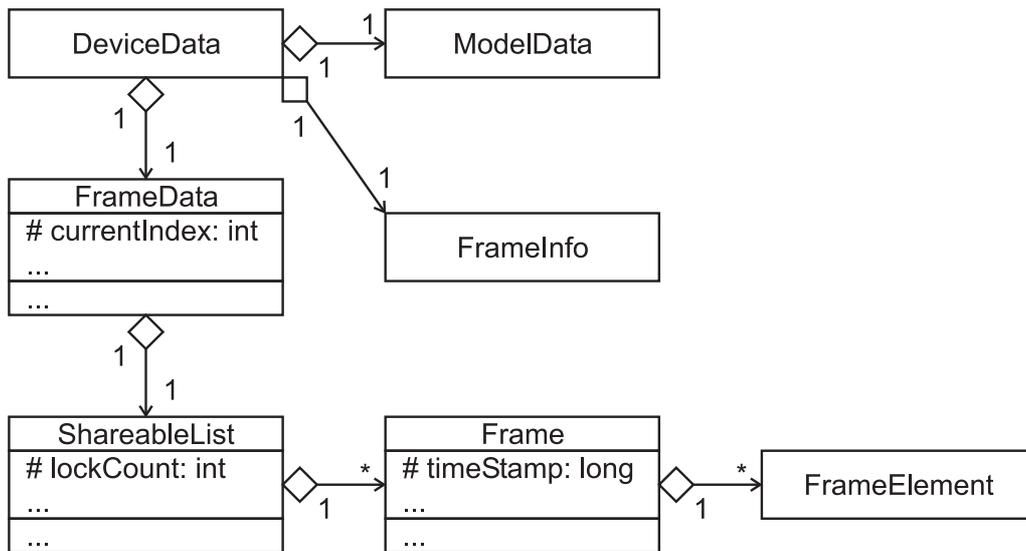


Abbildung 5.3: Aufbau und Verwaltung von Frame-Daten in VAMOS.

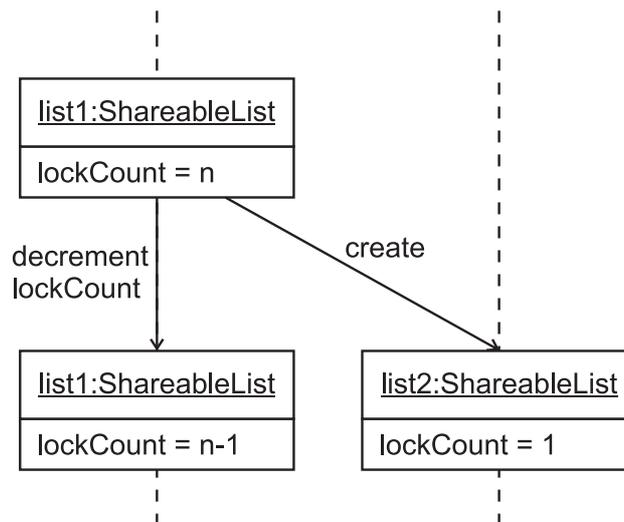
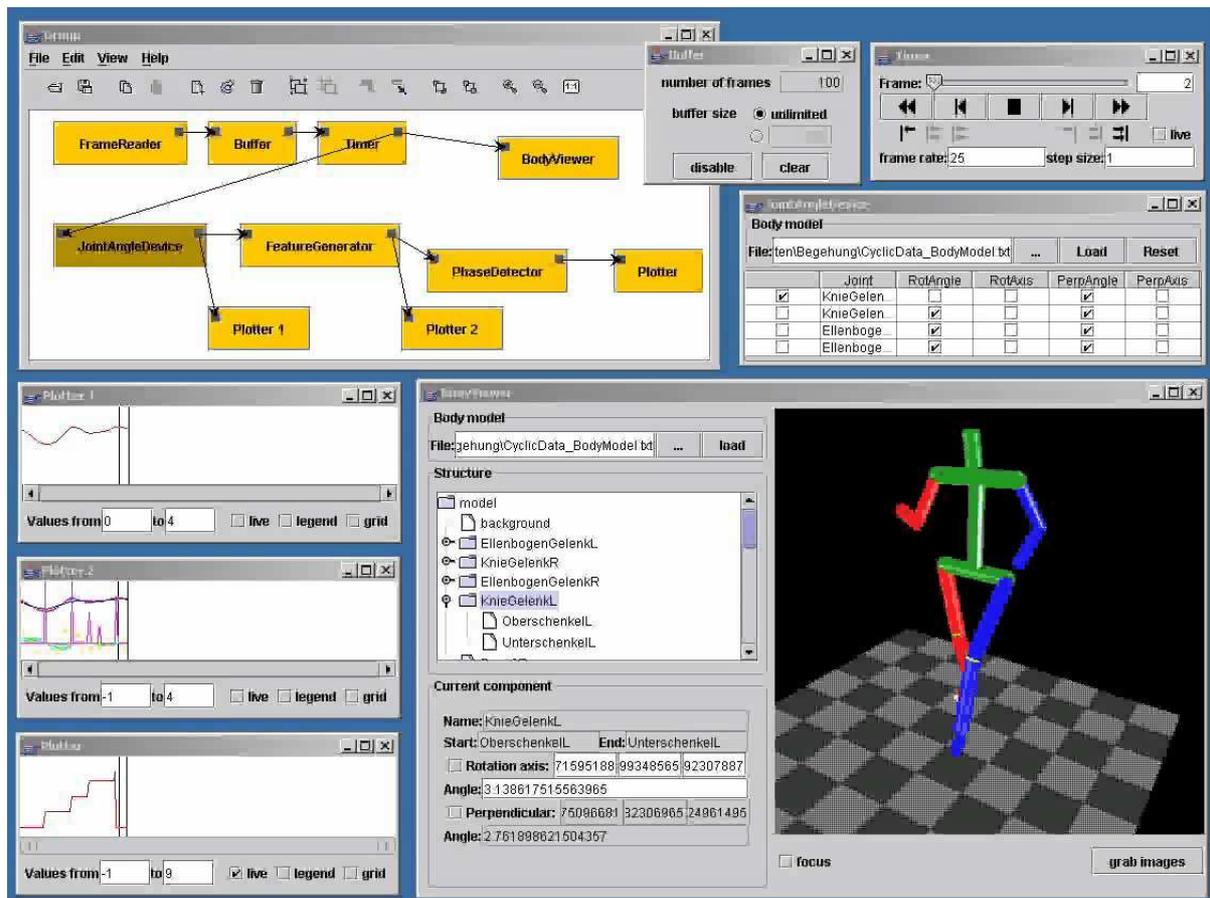


Abbildung 5.4: Ein Beispiel für die partielle Duplikation von Frames mit der Klasse ShareableList.

### 5.3.1 Bereitstellung von Daten und Merkmalen

In dem Schema werden die Winkeltrajektorien des Kniegelenks eines Probanden, wie sie in Abschnitt 4.6 beschrieben sind, mittels eines Moduls `FrameReader` von der Festplatte geladen. Die Daten können von einem beliebigen Bewegungserfassungssystem stammen, da der `FrameReader` die Daten in textueller Form von der Platte lädt und unterschiedliche Dateiformate anhand einer Format-Spezifikation mit einer eigenen Beschreibungssprache



**Abbildung 5.5:** Bildschirmdarstellung eines Schemas zur Bewegungsanalyse in VAMOS mit Benutzeroberflächen der eingesetzten Module.

gelesen werden können. Nach dem Einlesen werden die Daten in einem Modul Buffer gespeichert und stehen danach in VAMOS für die weitere Verarbeitung zur Verfügung. Mit Hilfe eines Moduls vom Typ Timer werden einzelne Frames aus den Daten in den Datenfluss der folgenden Module eingespeist. Dies erfolgt durch manuelle Benutzersteuerung oder automatisch durch Vorgabe einer bestimmten Frame-Rate, wie z. B. der Frame-Rate des ursprünglichen Tracking-Systems. Die Verwendung bereits aufgezeichneter Daten dient lediglich der Wiederholbarkeit der Versuche. Zwischen den abgespielten Daten und tatsächlich online per Bewegungserfassungssystem akquirierten Daten besteht kein Unterschied. Insbesondere können durch diesen Mechanismus auch Bewegungsdaten nach der aufwendigen Nachbereitung bei der video-basierten Bewegungserfassung erneut mit der gleichen Bildrate wiedergegeben werden, wie sie ursprünglich mit den Kameras aufgenommen wurden. Außerdem wird durch den Timer das wiederholte Abspielen von Daten ermöglicht, so dass z. B. ein einzelner Zyklus einer Gangbewegungssequenz mit mehreren Gangphasen wiedergegeben werden kann.

Im ersten Schritt der Verarbeitungskette werden aus den Gelenkwinkeltrajektorien mit Hilfe einer DWT, wie in Abschnitt 4.6 beschrieben, im Modul `FeatureGenerator` erweiterte Bewegungsmerkmale zu den Gelenkwinkeln auf verschiedenen Skalierungsstufen berechnet. Beim Eintreffen eines Frames am Eingang des Moduls `FeatureGenerator` wird dabei immer ein gleitendes Zeitfenster mit bisherigen Frames und dem aktuell hinzugekommenen Frame berücksichtigt und die entsprechenden Merkmale am Ausgangsport angepasst, so dass eine Datenstromverarbeitung ermöglicht wird. Die minimale und die maximale Länge des Zeitfensters können vom Benutzer bestimmt werden. Als Merkmale liegen am Ausgang des Moduls insgesamt die durch die DWT erzeugten Approximationsstufen der Kniewinkelverläufe sowie die Informationen über die Häufigkeit der Nulldurchgängen und das Krümmungsverhalten der Kurven zu den Wavelet-Koeffizienten an.

Die Verläufe der Kniewinkel und die der erweiterten Merkmale sind als Kurvenplots in den graphischen Benutzerschnittstellen zweier Module vom Typ `Plotter` in Abb. 5.5 aufgetragen. Die Position des aktuellen Frames in den zyklischen Gangdaten wird durch eine schwarze senkrechte Linie angezeigt.

### 5.3.2 Erkennung von Bewegungsphasen

Zur Verwendung von Hidden-Markov-Modellen in VAMOS wurde eine eigene HMM-Bibliothek entwickelt. Die Pakete dieser Bibliothek enthalten sowohl Klassen für diskrete und kontinuierliche HMMs (DHMM und MVCHMM) als auch Klassen zur Verwaltung der Beobachtungen und zur Berechnung von Mischverteilungen. Eine Schnittstelle `GeneralHMM`, die alle für VAMOS entwickelten HMMS erfüllen müssen, ermöglicht auch die Implementierung von komplexeren HMM-Typen, wie hierarchische, gekoppelte und faktorielle Modelle, die ebenfalls Bestandteil der Bibliothek sind. Mit Hilfe der Bibliothek ist es auch möglich, Modelle über XML zu importieren und zu exportieren und damit beispielsweise in Matlab erstellte Modelle in VAMOS einzuladen.

Auf der Basis der Klasse `MVCHMM` für kontinuierliche HMMs wurde ein Modul `PhaseGenerator` zur Erkennung von Bewegungsphasen geschrieben. Dieses Modul nutzt die in dieser Arbeit entwickelte Methode zur Bewegungsanalyse mit Hilfe der versteckten Zustände. Die Beobachtungen für das HMM erhält das Modul über seinen Eingang. In Abb. 5.5 ist dieser an das Modul `FeatureGenerator` angeschlossen, das die erweiterten Bewegungsmerkmale produziert. An den Ausgang des Moduls werden die mit dem Viterbi-Algorithmus berechneten Zustandssequenzen gelegt, welche die Bewegungsphasen repräsentieren. Zur Spezifikation der Beobachtungssequenzen werden wie beim Modul `FeatureGenerator` aus dem vorigen Abschnitt die Grenzen für ein Zeitfenster auf den Frames vom Benutzer vorgegeben. Die Ausgabe der Bewegungsphasen für die Gangbewegung aus dem praktischen Szenario sind in Abb. 5.5 ebenfalls als Kurvenverlauf mit einem Modul `Plotter` aufgetragen.

Die Geschwindigkeit, mit der die Berechnung der Phaseninformation in VAMOS erfolgt, hängt linear von der Anzahl der Frames pro betrachtetem Zeitfenster ab. Auf einem handelsüblichen PC (Intel Pentium 4 Prozessor mit 2,0 GHz Taktrate und 2,0 GB Hauptspei-

cher) beträgt die Rechenzeit für das Verarbeiten der Merkmale und die Bestimmung der Zustandssequenz mit dem HMM für Gangbewegungen aus Abschnitt 4.6 lediglich 8,5 ms bei einem Zeitfenster mit 100 Frames und 17,7ms bei einem Fenster mit 200 Frames. Bei einer typischen Frame-Rate von 50 Hz haben die vergleichbaren Bewegungssequenzen eine Länge von 2 bzw. 4s. Eine Analyse von Bewegungsdaten mit hinreichend langen Zeitfenstern kann demnach in VAMOS ohne Probleme in Echtzeit erfolgen.

### 5.3.3 Einfache Synthese von Bewegungen

In Abb. 5.5 ist neben den graphischen Benutzerschnittstellen der bereits beschriebenen Module auch die Darstellung eines 3D Körpermodells zu sehen, das zur Visualisierung der Gangbewegung verwendet wird. Dabei kommt das Modul **BodyViewer** zum Einsatz, das entwickelt wurde, um ein Körpermodell anhand definierter Markerpositionen in Frames zu visualisieren und zu animieren. Das Modul kann beliebige Körpermodelle anzeigen, die über eine Beschreibungssprache mit Hilfe von Segmenten und Gelenken in einer kinematischen Kette definiert werden. Die Grammatik für die Beschreibungssprachen besitzt dazu Produktionen der Form:

```
component ::=  
    segment componentID := segment ; |  
    joint componentID := joint ; |  
    ...  
  
segment ::=  
    segment( markerID , appearance )  
  
joint ::=  
    joint( componentID , componentID , appearance )
```

Liegt ein aktueller Frame am Eingang des Moduls **BodyViewer** an, so werden die darin enthaltenen Markerposition mit den Bezeichnungen *markerID* verwendet, um die Pose für das Skelett zu berechnen. Auf die Bezeichner *appearance*, welche die Farbe und Gestalt der Komponenten im Körpermodell angeben, sowie auf zusätzliche Komponenten, wie Punkte, Ebenen und Lichtquellen, soll hier nicht weiter eingegangen werden. In einer neueren Version von VAMOS kann das Körpermodell auch interaktiv vom Benutzer mit einem Editor erstellt werden und die Segmente mit einer Oberfläche aus Dreiecksnetzen für die Haut versehen werden.

Des Weiteren erlaubt ein Modul **GaitMapper** die Auswahl eines passenden Frames aus einer vollständigen Bewegungssequenz auf der Basis der aktuellen Bewegungsphase, die durch einen Phasenindex angegeben wird. Die Frames in den Bewegungssequenzen müssen dazu ebenfalls mit Phasenindizes versehen sein und liegen an einem Eingang des Moduls an.

Die Angabe der Phase erfolgt durch die Phasenindizes, die am zweiten Eingang eingelesen werden. Da die Abbildung der Phasenindizes auf die Indizes der Frames surjektiv ist, wird der Index des Frames anhand der bisherigen Verweildauer in einer Phase geschätzt. Der vollständige Frame wird daraufhin an den Ausgang des Moduls gelegt.

Für einige der Bewegungen aus Abschnitt 4.6 wurde eine Phasenbestimmung aus errechneten Winkeltrajektorien mit Hilfe von HMMs vorgenommen und die gewonnenen Phasenindizes zusätzlich in die Frames mit den ursprünglichen Positionsdaten geschrieben. Diese dienen als Datenbasis für den **GaitMapper**. Außerdem wurde der **PhaseDetector** aus dem vorigen Abschnitt an den Eingang für die Phaseninformation angeschlossen. Der Ausgang wurde direkt mit dem eingangs beschriebenen **BodyViewer** verbunden. Mit dieser Konfiguration wird lediglich die Winkeltrajektorie eines Gelenkes erfasst. In diesem Beispiel war dies der Kniewinkel der Probanden. Aus den Winkeldaten werden in VAMOS die erweiterten Bewegungsmerkmale über einen **FeatureGenerator** berechnet, die Bewegungsphasen mit Hilfe des **PhaseDetector** und der HMM-basierten Analysemethode bestimmt und schließlich eine virtuelle Darstellung des gesamten Körpers durch den **GaitMapper** und den **BodyViewer** erreicht. Die Berechnungen dieser Verarbeitungskette erfolgen ebenfalls in Echtzeit.

## 5.4 Zusammenfassung

Die Praxistauglichkeit der expliziten Nutzung von Zustandssequenzen aus HMMs wird am Beispiel der Erkennung von Phasen in Bewegungsdaten gezeigt. Hierzu wird das Framework VAMOS zur Visualisierung, Analyse und Modellierung von Anthropomatiksystemen verwendet. VAMOS ist zu einem großen Teil auch im Rahmen dieser Arbeit entstanden und erlaubt die Konstruktion komplexer Anwendungen durch Verknüpfung von einzelnen Modulen zu einem Datenflussschema. VAMOS ist durch fest vorgegebene Schnittstellen leicht erweiterbar und stellt über interne Mechanismen einen konsistenten und hinsichtlich Speicher- und Rechenzeitbedarf effizienten Austausch von Daten zwischen den Modulen zur Verfügung. In das System sind bereits wichtige Module zur Bearbeitung von Frame-Daten vorhanden, dies gilt insbesondere für die Anbindung an Bewegungserfassungssysteme.

Zur Integration der in dieser Arbeit entwickelten Analysemethoden wird VAMOS um zusätzliche Module ergänzt. Hierbei handelt es sich um ein Modul zur Generierung erweiterter Bewegungsmerkmale auf Basis der Wavelet-Analyse. Berechnet werden Winkeltrajektorien unterschiedlicher Approximationsstufen und Bewegungscharakteristika aus den Detailinformationen bei der Wavelet-Analyse. In einem weiteren Modul erfolgt die Auswertung der Bewegungsmerkmale mit Hilfe von HMMs. Die versteckten Zustände in den Modellen werden durch die beobachteten Merkmalssequenzen gesteuert und können durch den Viterbi-Algorithmus bestimmt werden. Sie dienen der Erkennung der Bewegungsphasen. Durch die Entwicklung eines geeigneten Datenflussschemas in VAMOS wird gezeigt, dass die Verarbeitungsschritte auf einem gängigen PC in Echtzeit durchgeführt werden können.

In einem abschließenden Beispiel werden die Bewegungsmodelle zur Synthese einer Ganzkörperbewegung verwendet. Die Bewegung wird als Animation eines Körpermodells in einer 3D Visualisierung wiedergegeben. Eine Beschreibungssprache definiert durch die Verkettung von Segmenten und Gelenken und die Zuordnung von Markerbezeichnern das Körpermodell. Als Eingabe für die Bewegungsgenerierung dienen lediglich die Winkeltrajektorien eines einzelnen Gelenkes. Aus ihnen wird die Phaseninformation mit HMMs bestimmt und mittels einer adaptiven Schätzung den Frames aus einer Datenbasis zugeordnet. Diese einfache Form der Bewegungssynthese erfolgt ebenfalls in Echtzeit.



# Kapitel 6

---

## Zusammenfassung und Ausblick

---

Eine sich an den individuellen Anforderungen des Menschen orientierende Anpassung von Abläufen und Prozessen in seiner Umwelt ist eine der zukunftsweisenden Herausforderungen der Informatik. Als Beitrag zu dieser Aufgabenstellung werden in dieser Arbeit neue adaptive Methoden zur Modellierung und Analyse von Anthropomatikdaten vorgestellt, die mit Hilfe von HMMs realisiert werden. Die Methoden orientieren sich an den kognitiven Fähigkeiten und der funktionellen Anatomie des Menschen und haben als Zielsetzung die automatische Erkennung von Zuständen in Anthropomatiksystemen.

Insbesondere bei den komplexen und dynamischen Prozessen der Mensch-Umwelt-Schnittstelle werden neuartige Modellierungen benötigt, die mit bisherigen Ansätzen, wie z. B. Differentialgleichungen, nicht auf den Rechner übertragbar sind. Die Aufgabenstellung wird zusätzlich erschwert, wenn der zu analysierende Prozess nicht direkt messbar ist und stattdessen nur indirekte Beobachtungen vorliegen. In dieser Arbeit wird gezeigt, wie durch HMMs diese Probleme gelöst werden können, indem neben der Klassifikation auf Basis des Vorwärts-Algorithmus auch die mit Hilfe des Viterbi-Algorithmus bestimmten versteckten Modellzustände zur Analyse verwendet werden. Das Hauptaugenmerk der Arbeit gilt der Optimierung der Zustände und der Struktur der HMMs sowie der Generierung von Merkmalen, welche die Übergänge zwischen den Modellzuständen geeignet steuern. Grundlage für alle vorgestellten Verfahren ist eine ausführliche Untersuchung des jeweils vorliegenden Messaufbaus, mit deren Hilfe die Modelle entwickelt werden.

Zunächst wird in der Arbeit die Datenklassifikation durch HMMs mit anderen Klassifikationsverfahren wie Neuronalen Netzen und Support-Vector-Machines am Beispiel der Analyse von Bohrgeräuschen während chirurgischer Eingriffe an der Wirbelsäule verglichen. Diese Eingriffe verlangen ein hohes Maß an Präzision bei der Durchführung, da gefährdete Strukturen wie das Rückenmark oder wichtige Blutgefäße leicht verletzt werden können und der Patient dadurch schwerste Schädigungen erleidet. Bisherige chirurgische Operations- und Navigationssysteme bieten dem Chirurgen jedoch nur eine unzureichende Hilfestellung, so dass er auf sein anatomisches Wissen und sein sensomotorisches Feingefühl angewiesen ist. Aufgrund seiner Erfahrung liefern ihm die Geräusche bei Bohrungen an Wirbeln Informationen über die Beschaffenheit des gerade abgetragenen Knochenmaterials und dadurch auch über die Position der Bohrspitze im Knochen. Der Chirurg kann

sich somit an den akustischen Informationen orientieren. Die in dieser Arbeit entwickelten automatischen Erkennungsmethoden bilden die Grundlage für eine entsprechende technische Umsetzung.

In der Arbeit wird gezeigt, wie mit Neuronalen Netzen, Support-Vector-Machines und HMMs im Vergleich die Phasen des Bohrvorgangs anhand geeigneter Audiomerkmale erkannt werden können. Die HMMs eignen sich dabei weniger zur Erkennung von einzelnen Bohrphasen, dafür aber wesentlich besser zur Detektion von Phasenübergängen. Entscheidend ist hierbei, dass HMMs in den versteckten Zuständen die Historie der betrachteten Zeitreihen erfassen. Die Ausgabe eines geeignet trainierten HMMs kann durch die berechnete Beobachtungswahrscheinlichkeit deutlich vor einem Eindringen des Bohrers in den Spinalkanal warnen.

Der Integrationsaufwand des entwickelten Analyseverfahrens in den Operationssaal ist gering. Ebenso sind die anfallenden Kosten für die benötigten Geräte sehr niedrig. Dies kommt insbesondere allgemeinen Bestrebungen zur Reduzierung von Kosten im Gesundheitswesen zugute. Auf lange Sicht sollte daher die automatische Analyse der Bohrgerausche ein fester Bestandteil von Assistenzsystemen in der Wirbelsäulen Chirurgie werden.

In der Arbeit wird des Weiteren am Beispiel von medizinischen Ultraschallbildern auf die Rolle der versteckten Zustände der HMMs und die Nachbildung eines nicht direkt messbaren Prozesses eingegangen. Aufgrund des Bildgebungsverfahrens spiegeln die Grauwerte in Ultraschallbildern lediglich die Grenzen zwischen unterschiedlichen Gewebereichen wider. Eine Darstellung der spezifischen Gewebecharakteristika in Bildbereichen wie bei Computer- oder der Magnetresonanztomographie wird durch den Bildgebungsprozess nicht erreicht. Ein Radiologe muss sich diese Informationen mit Hilfe von Expertenwissen ableiten, das er in langer Praxis erworben hat.

Die in der Arbeit entwickelten HMMs nutzen die beschriebenen Eigenschaften des bildgebenden Prozesses gezielt zur Kontrolle der Übergänge in den Modellzuständen aus. Durch eine Bestimmung der Zustandsverläufe in den HMMs wird eine Rekonstruktion von Gewebereichen mit einheitlichen Charakteristika ermöglicht. Anhand der Erkennung von Tumorstrukturen im Auge und im Brustbein wird dies an Beispielen aus der klinischen Radiologie belegt. Eine Aufbereitung der Bildmerkmale durch Glättung und gleichzeitige Kantenerhaltung mit Calderón-Operatoren bewirkt eine zusätzliche Stabilisierung der Zustandsübergänge und damit eine noch zuverlässigere Klassifikation. Aus radiologischer Sicht liefern die vorgestellten Methoden vielversprechende Ergebnisse für zukünftige Anwendungen, die sich die automatische Analyse von Ultraschalldaten mit Hilfe der HMM-Zustände zu Nutze machen können. Anwendungsgebiete für die entwickelten Methoden sind das Durchsuchen größerer Datenbanken, z. B. radiologische Bildarchive, nach bestimmten Gewebestrukturen oder die Kontrolle der Krankheitsentwicklung und des Behandlungsverlaufs bei einem Patienten.

Auf den Erkenntnissen über die Rolle der versteckten Zuständen der HMMs aufbauend, werden zielgerichtet Merkmale und Modelle für die Klassifikation von menschlichen Alltagsbewegungen in der Arbeit entwickelt. Eine Wavelet-basierte Darstellung von Gelenk-

winkeln ermöglicht dabei die gleichzeitige Untersuchung von Bewegungssignalen auf unterschiedlichen Skalierungsstufen. Dies entspricht auch der Bewegungswahrnehmung des Menschen, der sowohl den Grobverlauf einer Bewegung als auch Charakteristika in Teilen der Bewegung wahrnimmt. Der zusätzliche Vorzug der Wavelet-Darstellung ist, dass mit den Wavelet-Koeffizienten signifikante Zusatzinformationen über den Bewegungsverlauf geliefert werden, die mit Hilfe herkömmlicher Bewegungsmerkmale nicht erkennbar sind. Am Beispiel einer Gangbewegung wird dies belegt und zugleich die Bedeutung für Sport und Orthopädie hinsichtlich der frühzeitigen Erkennung gesundheitsschädlicher Bewegungsabläufe ersichtlich.

Die entwickelten Bewegungsmerkmale zeichnen sich besonders als Beobachtungswerte für HMMs aus, da sie die Zustandsverläufe verlässlich steuern. Am Beispiel einer umfassenden Studie mit komplexen Alltagsbewegungen wird gezeigt, dass die in der Arbeit entwickelten HMMs eine robuste Erkennung von personenspezifischen Charakteristika, von Veränderungen in den Rahmenbedingungen bei der Bewegungsausführung und von elementaren Bewegungsphasen gestatten. Ein Vergleich mit verschiedenen Ausprägungen von Neuronalen Netzen untermauert dieses Ergebnis. Mit den Informationen der HMM-basierten Analyse kann beispielsweise ein Robotersystem das kognitive Modell von seiner Umwelt und vom Bewegungszustand eines Benutzers bereichern und eine situations- und benutzergerechte Kooperation mit dem Menschen planen. Dies ist eine unerlässliche Grundvoraussetzung für die zukünftige Eingliederung von Robotern in den Alltag des Menschen, beispielsweise als Haushaltshilfe für ältere oder behinderte Menschen.

Um ein möglichst breites Spektrum von Bewegungen automatisch analysieren zu können, wird in dieser Arbeit ein Algorithmus zur Komposition von Modellen für Schlüsselbewegungen zu einem komplexen Bewegungsmodell entwickelt. Gegenüber einer herkömmlichen Klassifikation von Bewegungsphasen mit mehreren HMMs kann mit einem einzigen komplexen Modell eine effizientere Auswertung durch die in dieser Arbeit verfolgte Erkennung von Zustandssequenzen durchgeführt werden. Neben dem verbesserten Berechnungsaufwand hat das Verfahren den Vorteil, dass das Modell zu einem konsistenten Zustandsverlauf gezwungen wird und damit deutlich weniger Fehlklassifikationen als bei der konventionellen Methode auftreten. Insbesondere wird dies durch die robuste Erfassung der in der Arbeit untersuchten Rahmenbedingungen bei der Bewegungsausführung belegt. Eine Zukunftsperspektive ist, den vorgestellten Algorithmus in ein Robotersystem zu integrieren, dieses mit einem umfassenden Katalog von elementaren Bewegungsmodellen auszustatten und dann während des Betriebs benötigte Bewegungsmodelle nach einem Baukastenprinzip zusammenzustellen.

Abschließend wird die Einsatzfähigkeit der entwickelten Methoden in einer praktischen Anwendung demonstriert. Hierzu wird das Framework VAMOS zur Visualisierung, Analyse und Modellierung von Anthropomatiksystemen verwendet, das zu einem großen Teil im Rahmen dieser Arbeit entstanden ist. Mit VAMOS können komplexe Anwendungen durch Verknüpfung von einzelnen Modulen zu einem Datenflussschema erzeugt werden. Die Grundlage hierfür bildet eine speicher- und rechenzeiteffiziente Programmarchitektur zur Signalverarbeitung. Am Beispiel der Analyse von Gangbewegungen durch die in dieser Arbeit entwickelte Methodik wird die Echtzeitfähigkeit der Bestimmung von Bewegungs-

phasen aus Modellzuständen gezeigt. Die erkannten Bewegungsphasen werden in einem weiteren Beispiel zur Synthese und 3D Visualisierung einer Ganzkörperbewegung aus der Winkeltrajektorie eines einzelnen Gelenkes verwendet.

Die in der Arbeit entwickelten Methoden zur Generierung von Merkmalen und Modellen stellen neue Verfahren zur Analyse von Anthropomatikdaten dar. Vergleichbare Analysen sind mit herkömmlichen Methoden derzeit nicht oder nur schwer realisierbar. Über die vorliegenden Anwendungen hinaus liefern die vorgestellten Methoden neue Ansätze zur Analyse von Zuständen weiterer Anthropomatiksysteme, bei denen Messdaten wie gezeigt in geeignete Merkmale überführt werden können. Eine Grundlage für diese zukünftigen Arbeiten stellt das bereits prämierte Framework VAMOS dar.

# Anhang A

---

## Wavelet-Transformation

---

In den letzten Jahren haben sich Wavelets als mächtiges Werkzeug in der Signalverarbeitung etabliert. Bildanalyse, Datenkompression und Rauschreduktion sind nur eine kleine Auswahl aus der Vielzahl von Anwendungsbeispielen der Wavelet-Transformation. Motivation für die Entwicklung der Wavelet-Transformation war, eine Frequenzanalyse eines Signals wie mit Hilfe der Fourier-Transformation [10] durchzuführen. Jedoch sollte nicht nur der globale Verlauf des Signals anhand des Frequenzraumes, sondern zugleich auch der zeitliche lokale Verlauf des Signals im Ortsraum untersucht werden können. Die Wavelet-Transformation verbindet die Eigenschaften der Darstellung eines Signals im Ortsraum und im Frequenzraum.

Man erhält die Wavelet-Transformation durch Faltung des Signals mit gestauchten und verschobenen Basisfunktionen, den so genannten *Wavelets*. Bei den Basisfunktionen sind orthonormale und nicht orthonormale zu unterscheiden. Hier soll vor allem auf erstere eingegangen werden, da sie beispielsweise in Bezug auf Rauschminderung Vorteile aufweisen. Unter anderem bringt eine Transformation mit ihnen keine unnötige Redundanz in das Signal und sie ergeben Orthonormalbasen (ONB), mit denen die Invertierung der Transformation auf einfache Weise möglich wird. Weiterhin ist die Wavelet-Transformation wie die Fourier-Transformation sowohl für den kontinuierlichen als auch den diskreten Fall definiert. Aufgrund der Relevanz für praktische Implementierungen soll in diesem Rahmen lediglich der diskrete Fall behandelt werden.

Eine Übersicht zur Entwicklung und Anwendung von Wavelets wird u. a. in [68] gegeben. Weiterführende Informationen und Details zu den in diesem Anhang beschriebenen Sachverhalten sind u. a. in [91], [29] oder [94] zu finden.

### A.1 Multiskalenanalyse

Üblicherweise wird zur Beschreibung und Konstruktion von Wavelet-Orthonormalbasen die *Multiskalenanalyse (MSA)* herangezogen. Dieser Ansatz wird auch hier verfolgt. Grundlage für die Betrachtungen sei, dass es sich bei dem zu untersuchenden Signal um eine

Funktion  $f$  aus dem Raum der *quadratisch integrierbaren Funktionen*  $L^2(\mathbb{R})$  handelt, für die gilt

$$\int_{-\infty}^{+\infty} |f(t)|^2 dt < \infty.$$

Diese Annahme ist zulässig, da die im Rahmen dieser Arbeit auftretenden Signale endliche Energien besitzen.

Das zu  $L^2(\mathbb{R})$  gehörende Skalarprodukt und die Norm sind definiert als

$$\langle f, g \rangle := \int_{-\infty}^{+\infty} f(t)\bar{g}(t) dt \quad \text{und} \quad \|f\| := \sqrt{\langle f, f \rangle} = \sqrt{\int_{-\infty}^{+\infty} |f(t)|^2 dt},$$

wobei  $\bar{g}$  die komplex konjugierte Funktion zu  $g$  bezeichnet.

Eine Folge abgeschlossener Unterräume  $\mathbf{V}_n \subset L^2(\mathbb{R})$  mit  $n \in \mathbb{Z}$  bildet eine orthonormale Multiskalenanalyse, wenn folgende Bedingungen erfüllt sind:

1.  $\dots \subset \mathbf{V}_2 \subset \mathbf{V}_1 \subset \mathbf{V}_0 \subset \mathbf{V}_{-1} \subset \mathbf{V}_{-2} \subset \dots$
2.  $\bigcap_{n \in \mathbb{Z}} \mathbf{V}_n = \{0\}$  und  $\overline{\bigcup_{n \in \mathbb{Z}} \mathbf{V}_n} = L^2(\mathbb{R})$ <sup>1</sup>
3.  $f \in \mathbf{V}_n \iff f(2 \cdot) \in \mathbf{V}_{n-1}$
4.  $f \in \mathbf{V}_0 \iff f(\cdot - k) \in \mathbf{V}_0$  mit  $k \in \mathbb{Z}$
5. Es gibt eine Funktion  $\varphi \in L^2(\mathbb{R})$ , so dass  $\{\varphi_{0,k} := \varphi(\cdot - k) \mid k \in \mathbb{Z}\}$  eine ONB für  $\mathbf{V}_0$  ist.

Die Funktion  $\varphi$  heißt *Skalierungsfunktion*. Man sagt, dass  $\varphi \in L^2(\mathbb{R})$  eine MSA erzeugt, wenn die Unterräume  $\mathbf{V}_n$ , welche durch die Funktionen in der Menge  $\{\varphi_{n,k} \mid k \in \mathbb{Z}\}$  mit

$$\varphi_{n,k} := 2^{-\frac{n}{2}} \varphi(2^{-n} \cdot -k)$$

aufgespannt werden, eine MSA bilden.

Die Unterräume  $\mathbf{V}_n$  stellen  $\mathbf{V}_0$  bei verschiedenen Auflösungen  $2^{-n}$  dar und werden deshalb *Approximationsräume* genannt. Die Variable  $n$  ist der *Skalenindex*. Da  $\mathbf{V}_{n+1} \subset \mathbf{V}_n$ , existiert ein orthogonaler Vektorraum  $\mathbf{W}_{n+1}$ , für den gilt

$$\mathbf{V}_n = \mathbf{V}_{n+1} \oplus \mathbf{W}_{n+1}.$$

Dies lässt sich induktiv fortsetzen:

$$\begin{aligned} \mathbf{V}_n &= \mathbf{V}_{n+1} \oplus \mathbf{W}_{n+1} \\ &= \mathbf{V}_{n+2} \oplus \mathbf{W}_{n+2} \oplus \mathbf{W}_{n+1} \\ &= \dots \end{aligned}$$

<sup>1</sup> $\overline{M}$  bezeichne den Abschluss einer Menge  $M$  in  $L^2(\mathbb{R})$ .

Insgesamt ergibt sich

$$L^2(\mathbb{R}) = \overline{\bigoplus_{j \in \mathbb{Z}} \mathbf{W}_j}.$$

Die Unterräume  $\mathbf{W}_n$  beinhalten Detailinformationen zum Übergang eines Approximationsraumes in einen benachbarten und werden aufgrund dessen *Detailräume* bezeichnet.

Analog zu den Unterräumen  $\mathbf{V}_n$  lässt sich auch für die Unterräume  $\mathbf{W}_n$  eine Menge von Funktionen  $\{\psi_{n,k}\}$ , den *Wavelets*, angeben, welche die Eigenschaften (3) und (4) der MSA besitzen und eine ONB des Unterraumes  $\mathbf{W}_n$  bilden. Insbesondere lassen sich Funktionen  $\psi_{n,k}$  aus einer Grundfunktion  $\psi$ , dem *Mutter-Wavelet*, mittels Translation und Dilatation erzeugen. Diese stellen ebenfalls eine ONB für  $L^2(\mathbb{R})$  dar.

## A.2 Dekomposition und Rekonstruktion

Bevor auf die Berechnung der Wavelet-Transformation eingegangen wird, muss der Zusammenhang zwischen der Skalierungsfunktion  $\varphi$  und dem Mutter-Wavelet  $\psi$  hergestellt werden. Da  $\varphi \in \mathbf{V}_0 \subset \mathbf{V}_{-1}$ , lässt sich  $\varphi$  als Linearkombination aus der ONB  $\{\varphi_{-1,k}\}$  zu  $\mathbf{V}_{-1}$  darstellen:

$$\varphi(t) = \sqrt{2} \sum_{k \in \mathbb{Z}} h_k \varphi_{-1,k}(t) \quad \text{mit} \quad h_k = \langle \varphi, \varphi_{-1,k} \rangle. \quad (\text{A.1})$$

Diese Gleichung wird *Dilatations-, Verfeinerungs- oder Zweiskalengleichung* genannt. Sie entspricht einer Filterung des Signals, wobei  $h_k$  die Filterkoeffizienten eines diskreten Filters sind. In der Regel ist  $\varphi$  so konstruiert, dass es sich bei der Filterung um einen Tiefpass handelt.

Es lässt sich mit Gleichung (A.1) zeigen, dass  $\psi$  ebenfalls aus der ONB zu  $\mathbf{V}_{-1}$  berechenbar ist:

$$\psi(t) = \sqrt{2} \sum_{k \in \mathbb{Z}} (-1)^k h_{-k+1} \varphi_{-1,k}(t). \quad (\text{A.2})$$

Die Filterkoeffizienten  $g_k := (-1)^k h_{-k+1}$  ergeben sich in umgekehrter Reihenfolge der  $h_k$  mit alternierendem Vorzeichen und bewirken üblicherweise eine Hochpassfilterung.

Mit diesen Zusammenhängen lässt sich beschreiben, wie eine Funktion  $f \in L^2(\mathbb{R})$  bezüglich einer Wavelet-Basis dargestellt werden kann. Dazu seien  $\{v_{n,k} | n, k \in \mathbb{Z}\}$  die Koeffizienten der Skalierungsfunktion  $\varphi_{n,k}$ , *Skalierungs- oder Approximationskoeffizienten* genannt, und  $\{w_{n,k} | n, k \in \mathbb{Z}\}$  die *Wavelet-Koeffizienten* oder auch *Detailkoeffizienten*. Sie berechnen sich allgemein aus

$$v_{n,k} = \langle f, \varphi_{n,k} \rangle = \int f(t) \overline{\varphi_{n,k}}(t) dt \quad \text{und} \quad w_{n,k} = \langle f, \psi_{n,k} \rangle = \int f(t) \overline{\psi_{n,k}}(t) dt \quad .$$

Mit Hilfe der Gleichung (A.1) und (A.2), die auf eine beliebige Skala  $n$  übertragen werden,

erhält man schließlich die Berechnungsvorschriften:

$$\begin{aligned} v_{n,k} &= \langle f, \varphi_{n,k} \rangle \\ &= \sum_{l \in \mathbb{Z}} h_{l-2k} \langle f, \varphi_{n-1,k} \rangle \\ &= \sum_{l \in \mathbb{Z}} h_{l-2k} v_{n-1,l} \end{aligned}$$

und

$$\begin{aligned} w_{n,k} &= \langle f, \psi_{n,k} \rangle \\ &= \sum_{l \in \mathbb{Z}} g_{l-2k} \langle f, \varphi_{n-1,k} \rangle \\ &= \sum_{l \in \mathbb{Z}} g_{l-2k} v_{n-1,l}. \end{aligned}$$

Bei genauerer Betrachtung stellt sich heraus, dass nur die Hälfte der Koeffizienten zur Dekomposition benötigt wird. Man führt deshalb eine *Unterabtastung* (engl. *subsampling*) durch. Damit ist ein Algorithmus zur schnellen Wavelet-Transformation gefunden. Er besitzt einen Speicher- und Zeitaufwand von  $\mathcal{O}(m)$ , wobei  $m$  die Anzahl der von 0 verschiedenen gegebenen Skalierungskoeffizienten in  $\{v_{0,k}\}$  ist. Eine Wavelet-Transformation ohne Unterabtastung wird als *redundant* bezeichnet.

Für die Umkehrung der Transformation wird eine vergleichbare Rekonstruktionsformel benötigt. Aufgrund von  $\varphi_{n,k} \in \mathbf{V}_n = \mathbf{V}_{n+1} \oplus \mathbf{W}_{n+1}$  muss

$$\varphi_{n,k} = \sum_{l \in \mathbb{Z}} a_{2l-k} \varphi_{n+1,l} + b_{2l-k} \psi_{n+1,l}$$

gelten, wobei sich für die zunächst unbekanntenen Koeffizienten  $a_k$  und  $b_k$  aus den Gleichung (A.1) und (A.2)  $a_k = h_{-k}$  und  $b_k = (-1)^k h_{k+1} = g_{-k}$  ergeben. Insgesamt folgt daraus die Rekonstruktionsformel zu

$$\begin{aligned} v_{n,k} &= \langle f, \varphi_{n,k} \rangle \\ &= \sum_{l \in \mathbb{Z}} a_{2l-k} v_{n+1,k} + b_{2l-k} w_{n+1,k} \\ &= \sum_{l \in \mathbb{Z}} h_{k-2l} v_{n+1,k} + g_{k-2l} w_{n+1,k}. \end{aligned}$$

Der Rekonstruktionsalgorithmus besitzt ebenso wie der Zerlegungsalgorithmus einen Aufwand von  $\mathcal{O}(m)$ .

Aufgrund der Orthonormalität der Wavelet-Basen, können dieselben Wavelets sowohl für die Zerlegung als auch für die Rekonstruktion verwendet werden. Dies wird durch die Verwendung derselben Koeffizienten  $\{h_k\}$  und  $\{g_k\}$  für die diskreten Filterungsoperationen bei den Zerlegungs- und Rekonstruktionsschritten deutlich. Eine Abschwächung der strikten Orthonormalität erlaubt die Konstruktion so genannter *biorthogonaler* Wavelets. Diese Basisfunktionen werden, wie eingangs beschrieben, nicht in dieser Arbeit betrachtet. Details können der angegebenen Literatur entnommen werden.

## A.3 Wavelet-Basen

Nachdem in den vorigen Abschnitten das Prinzip der Wavelet-Transformation erklärt worden ist, geht dieser Abschnitt darauf ein, welche Funktionen sich als Skalierungs- bzw. Wavelet-Funktionen eignen und welche Eigenschaften diese besitzen. Allgemein werden an eine Familie von Wavelet-Funktionen folgende Grundforderungen gestellt:

**Orthonormalität:** Die beschriebene Konstruktion der MSA beruht auf der Orthonormalität der verwendeten Wavelet-Basis. Daher werden auch alle im Folgenden beschriebenen Wavelets über diese Eigenschaft verfügen.

**Kompakter Träger:** Für die lokalisierenden Eigenschaften der Wavelets ist ein kompakter Träger erforderlich, d. h. es darf nur eine endliche Anzahl von Filterkoeffizienten ungleich null sein.

**Symmetrie:** Für viele Anwendungen ist es wichtig, dass die Wavelets symmetrisch sind. Beispiele dafür sind Kompressionsverfahren oder Verfahren zur Detektion von Singularitäten.

Weitere wünschenswerte Eigenschaften sind Glattheit, z. B. bei Rauschreduktionsverfahren, und verschwindende Momente, die u. a. bei der Differentiation mit Wavelets (siehe Abschnitt A.4) von wesentlicher Bedeutung sind.

### A.3.1 Daubechies-Wavelets

Eine der wichtigsten Familien bilden die Wavelets von DAUBECHIES. Sie sind so konstruiert, dass sie den Forderungen nach Orthonormalität und einem kompakten Träger nachkommen. Sie besitzen den Nachteil, dass für die Skalierungs- und Detailfunktionen keine geschlossenen Darstellungen angegeben werden können.

Hier soll grob die Konstruktion angegeben werden, mit der die Daubechies-Wavelets bestimmt werden. [91] gibt eine detaillierte Darstellung. Grundlage für die Skalierungsfunktion  $\varphi$  und die Detailfunktion  $\psi$  ist ein trigonometrisches Polynom

$$q(\omega) := \sqrt{2} \sum_{k \in \mathbb{Z}} h_k e^{-ik\omega},$$

das die Fourier-Transformation des Filters der Skalierungsfunktion darstellt. Für einen endlichen Träger soll  $q$  nur endlich viele  $h_k \neq 0$  besitzen. Die Fourier-Transformierte  $\hat{\varphi}$  zur Skalierungsfunktion  $\varphi$  lässt sich durch  $q$  ausdrücken:

$$\begin{aligned} \hat{\varphi}(\omega) &= q\left(\frac{\omega}{2}\right) \hat{\varphi}\left(\frac{\omega}{2}\right) \\ &= q\left(\frac{\omega}{2}\right) q\left(\frac{\omega}{4}\right) \hat{\varphi}\left(\frac{\omega}{4}\right) \\ &= \dots \\ &= \frac{1}{\sqrt{2\pi}} \prod_{j=0}^{\infty} q(2^{-j}\omega) \end{aligned} \tag{A.3}$$

	Filterkoeffizienten $h_k$			
	$D = 1$	$D = 2$	$D = 3$	$D = 4$
$k = 0$	$\frac{1}{2}\sqrt{2}$	$\frac{1-\sqrt{3}}{4\sqrt{2}}$	0,332671	0,230378
$k = 1$	$\frac{1}{2}\sqrt{2}$	$\frac{3-\sqrt{3}}{4\sqrt{2}}$	0,806892	0,714847
$k = 2$		$\frac{3+\sqrt{3}}{4\sqrt{2}}$	0,459878	0,630881
$k = 3$		$\frac{1+\sqrt{3}}{4\sqrt{2}}$	-0,135011	-0,027984
$k = 4$			-0,085441	-0,187035
$k = 5$			0,035226	0,030841
$k = 6$				0,032883
$k = 7$				-0,010597

**Tabelle A.1:** Filterkoeffizienten für  $\varphi_D$  (Quelle: [91])

Für die Fourier-Transformierte  $\hat{\psi}$  des Wavelets  $\psi$  ergibt sich

$$\hat{\psi}(\omega) = e^{-\frac{i\omega}{2}q} \left(\frac{\omega}{2} + \pi\right) \hat{\varphi}\left(\frac{\omega}{2}\right). \quad (\text{A.4})$$

Zu den Gleichung (A.3) und (A.4) können Funktionen  $\varphi$  und  $\psi$  in Abhängigkeit der Filterkoeffizienten  $h_k$  gefunden werden. Die Lösungen haben dabei immer eine gerade Anzahl  $2D$  von Filterkoeffizienten, die von null verschieden sind. Als Trägerintervalle ergeben sich somit  $[0, 2D - 1]$  für  $\varphi$  und  $[-D + 1, D]$  für  $\psi$ . Tabelle A.1 enthält die Filterkoeffizienten der mit  $\varphi_D$  bezeichneten Skalierungsfunktionen für  $D = 1, \dots, 4$ . Die Filterkoeffizienten  $g_k$  der zugehörigen Wavelets  $\psi_D$  berechnen sich wie in Abschnitt A.2 angegeben.

Den Daubechies-Wavelets fehlt für  $D > 1$  die Eigenschaft der Symmetrie. Dafür kann gezeigt werden, dass die Glattheit der Funktionen  $\psi_D$  mit  $D$  linear zunimmt und die Wavelets  $D$  verschwindende Momente besitzen, d. h.

$$\int_{-\infty}^{+\infty} x^n \psi(t) dt = 0 \quad \text{für } n = 0, \dots, D - 1.$$

Die Skalierungsfunktion reproduziert somit Polynome bis zum Grad  $D - 1$ .

### A.3.2 Haar-Wavelets

Eine besondere Rolle spielen die Daubechies-Wavelets mit  $D = 1$ . Die so genannten Haar-Wavelets wurden von HAAR vor der Entwicklung der MSA entdeckt und im Jahre 1910 in [50] beschrieben. Sie stellen die einfachsten Vertreter für eine Wavelet-Familie dar und bilden zugleich ein gemeinsames Mitglied vieler weiterer Wavelet-Familien neben den Daubechies-Wavelets.

Im Detail sind die Approximations- und Detailfunktionen der Haar-Wavelets definiert als

$$\varphi(t) = \begin{cases} 1 & \text{für } 0 \leq t < 1, \\ 0 & \text{sonst} \end{cases}$$

und

$$\psi(t) = \begin{cases} \frac{1}{2}\sqrt{2} & \text{für } 0 \leq t < \frac{1}{2}, \\ -\frac{1}{2}\sqrt{2} & \text{für } \frac{1}{2} \leq t < 1, \\ 0 & \text{sonst.} \end{cases}$$

Die Haar-Wavelets sind folglich orthonormal, besitzen einen äußerst kompakten Träger und weisen Symmetrie auf. Es handelt sich bei ihnen um die einzige Familie, die tatsächlich alle drei Grundforderungen an Wavelet-Familien ohne Einschränkungen erfüllt. Leider geht dabei die Eigenschaft der Glattheit verloren, was eine schlechtere Frequenzauflösung mit sich bringt. Dieser Nachteil wird aber wieder wettgemacht, da sich eine DWT mit Haar-Wavelets aufgrund der Anzahl von Filterkoeffizienten sehr effizient implementieren und berechnen lässt, wodurch sie sich besonders für zeitkritische Signalverarbeitungsaufgaben eignet. Zudem besitzen Haar-Wavelets durch ihre Zugehörigkeit zu den Daubechies-Wavelets  $D = 1$  verschwindende Momente, wodurch sie wie im folgenden Abschnitt beschrieben zur Ableitung von Signalen auf verschiedenen Skalierungsstufen verwendet werden können.

## A.4 Wavelets als Differentialoperator

Die wesentlichen Informationen in einem Signal sind häufig an Singularitäten und Unstetigkeiten abzulesen. Beispielsweise kennzeichnen starke Grauwertschwankungen Kanten in einem Bild. Derartige Singularitäten können mit Hilfe der Wavelet-Analyse und der Betrachtung charakteristischer Veränderungen der Wavelet-Koeffizienten auf unterschiedlichen Skalierungsstufen detektiert werden. Grundlage hierfür bilden Wavelets  $\psi$ , die ein schnelles Abklingverhalten haben, d. h. dass es für jeden Exponenten  $\epsilon$  bei einem exponentiellen Abklingen eine Konstante  $C_\epsilon$  gibt, so dass gilt

$$\forall t \in \mathbb{R} : |\psi(t)| \leq \frac{C_\epsilon}{1 + |t|^\epsilon}. \quad (\text{A.5})$$

Verfügen diese Wavelets  $\psi$  über  $n$  verschwindenden Momente, so können sie als die  $n$ -te Ableitung einer Funktion  $\vartheta$  geschrieben werden. Die resultierende Wavelet-Transformation

kann damit als Multiskalen-Differentialoperator angesehen werden. Erklärt wird diese Eigenschaft durch den folgenden Satz.

Ein Wavelet  $\psi$  mit schnellem Abklingen hat  $n$  verschwindende Momente genau dann, wenn es eine Funktion  $\vartheta$  mit schnellem Abklingen gibt, so dass gilt

$$\psi(t) = (-1)^n \frac{d^n \vartheta(t)}{dt^n}. \quad (\text{A.6})$$

Daraus folgt auch insgesamt für die Wavelet-Transformation  $\mathscr{W}$ , dass

$$[\mathscr{W} f](t, s) = s^n \frac{d^n}{dt^n} [f * \tilde{\vartheta}_s](t) \quad (\text{A.7})$$

mit  $\tilde{\vartheta}_s(t) = \frac{1}{\sqrt{s}} \vartheta(-\frac{t}{s})$  ist. Außerdem hat  $\psi$  nicht mehr als  $n$  verschwindende Momente, genau dann wenn  $\int_{-\infty}^{+\infty} \vartheta(t) dt \neq 0$ .

Der Beweis zu diesem Satz ist in [94] vollständig angegeben. Er beruht auf der Fourier-Darstellung von  $\psi$  und macht sich die Integrations- und Ableitungseigenschaften der Fourier-Transformation sowie den Zusammenhang zwischen Abklingverhalten und Regularität einer Funktion zu nutzen. Die wichtige Aussage des Satzes ist, dass die Faltung  $[f * \tilde{\vartheta}_s](t)$  als gewichtete Mittelung von  $f$  mit einem um Faktor  $s$  dilatierten Filterkern darstellt, wenn  $\int_{-\infty}^{+\infty} \vartheta(t) dt = C \neq 0$ . Die Transformation  $[\mathscr{W} f](t, s)$  ist die Ableitung  $n$ -ter Ordnung einer Mittelung von  $f$  über einem Bereich proportional zu  $s$ .

# Anhang B

---

## Neuronale Netze

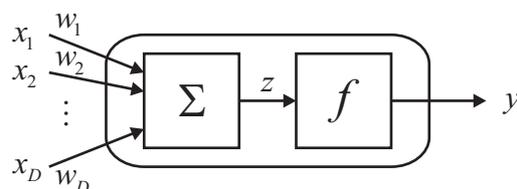
---

### B.1 Grundlegende Komponenten

Die Theorie zu künstlichen *Neuronalen Netzen* (NN) orientiert sich an den Erkenntnissen über den Aufbau des menschlichen Gehirns, die man im Bereich der Neurowissenschaften gesammelt hat. Analog zu den Nervenzellen im menschlichen Gehirn bilden *Neuronen* die elementaren Einheiten bzw. Knoten in künstlichen Neuronalen Netzen. Die einzelnen Neuronen sind über gerichtete Verknüpfungen miteinander verbunden. Genauso wie eine Nervenzelle Reize über ihr Axon weitergibt, geschieht dies bei künstlichen NNs über die Verknüpfungen.

Formal definiert ist ein Neuron ein Tupel  $\mathcal{N} = (\mathbf{x}, \mathbf{w}, f)$  mit Eingabevariablen  $\mathbf{x} \in \mathbb{R}^D$  für eingehende Reize von vorgeschalteten Neuronen, Gewichtungen  $\mathbf{w} \in \mathbb{R}^D$  für die Eingänge und einer Transferfunktion  $f : \mathbb{R} \rightarrow Y \subseteq \mathbb{R}$ , die den eigenen Aktivierungszustand  $y \in Y$  berechnet. Abbildung B.1 zeigt die schematische Darstellung eines einzelnen Neurons.

Ein *Reiz* wird genau dann von einem Neuron  $\mathcal{N}$  über seine Ausgabe an seine Nachfolger übertragen, wenn genügend Reize von seinen Vorgängern bei ihm anliegen und eine bestimmte Aktivierungsschwelle überschritten ist. Zu diesem Zweck berechnet das Neuron



**Abbildung B.1:** Schematische Darstellung eines Neurons, der elementaren Einheit eines künstlichen Neuronalen Netzes.

die gewichtete Summe seiner Eingaben

$$z = \sum_{i=1}^D w_i x_i. \quad (\text{B.1})$$

Die Variablen  $x_i$  bezeichnen hierbei die Stärke der eingehenden Aktivierung der Vorgänger, die mit den Faktoren  $w_i$  gewichtet werden. Die ausgehende Aktivierung des Neurons berechnet sich mit Hilfe einer Transferfunktion  $f$ , welche die Aktivierungsschwelle eines natürlichen Neurons nachbilden soll. Konkrete Beispiele für Realisierungen von Transferfunktionen sind in Abschnitt B.2 angegeben. Eine Verschiebung der mit  $f$  fest vorgegebenen Aktivierungsschwelle wird üblicherweise mit Hilfe eines zusätzlichen Bias-Gewichtes  $w_0$  mit einer zugehörigen konstanten Eingabe  $x_0$  in Gleichung (B.1) erzielt. Der ausgehende Aktivierungszustand eines Neurons ergibt sich damit insgesamt aus

$$a = f(z) = f\left(\sum_{i=0}^D w_i x_i\right). \quad (\text{B.2})$$

## B.2 Transferfunktionen

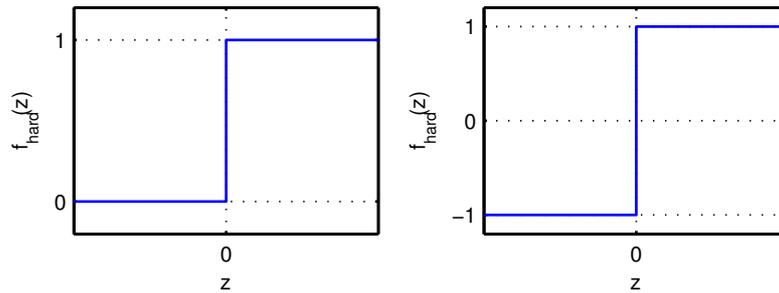
Welche Transferfunktion  $f$  bei einem NN verwendet wird, hängt stark vom Netzmodell und Anwendungsgebiet ab. Unterschiedliche Varianten werden in [24] angegeben. Typische Vertreter sind *binäre Transferfunktionen*  $f : \mathbb{R} \rightarrow Y = \{0, 1\}$ , wobei der Wert 0 den Zustand „inaktiv“ und 1 den Zustand „aktiv“ bezeichnet. Die einfache Realisierung einer solchen Funktion ist die *harte Schwellwertfunktion*

$$f_{\text{hard}} : z \mapsto \begin{cases} 1 & \text{für } z \geq 0, \\ 0 & \text{sonst.} \end{cases} \quad (\text{B.3})$$

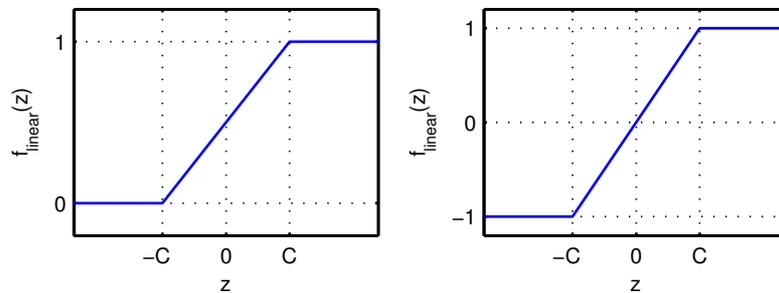
Anstelle des Wertebereichs  $Y = \{0, 1\}$  kann auch  $Y = \{-1, 1\}$  verwendet werden, so dass mit der Transformation  $z \mapsto 2z - 1$  die Transferfunktion durch die Vorzeichenfunktion  $\text{sgn}$  realisiert wird. Die beiden Varianten der binären Transferfunktion sind in Abb. B.2 graphisch dargestellt.

Oft ist es wünschenswert, dass die Aktivierung in einem gewissen Bereich proportional zur Eingabe ist. Zu diesem Zweck werden *begrenzt lineare Transferfunktionen*  $f : \mathbb{R} \rightarrow [0, 1]$  definiert. Bei diesen Transferfunktionen muss ein Schwellwert  $C_1$  überschritten werden, um eine Aktivierung hervorzurufen. Gleichzeitig steigt der Grad der Aktivierung mit größer werdenden Eingabewerten bis eine Sättigung erreicht ist, wonach sich der Ausgabewert nicht mehr ändert. Damit die Funktion symmetrisch zu  $z = 0$  wird, verwendet man üblicherweise  $C := C_1 = -C_2$ . Realisiert wird diese Transferfunktion mit der *Rampenfunktion*

$$f_{\text{linear}} : z \mapsto \begin{cases} 1 & \text{für } z \geq C, \\ (z/C + 1)/2 & \text{für } -C < z < C, \\ 0 & \text{sonst.} \end{cases} \quad (\text{B.4})$$



**Abbildung B.2:** Transferfunktionen mit hartem Schwellwert, links: Wertebereich  $\{0, 1\}$ , rechts: Wertebereich  $\{-1, 1\}$ .



**Abbildung B.3:** Begrenzt lineare Transferfunktionen, links: Wertebereich  $[0, 1]$ , rechts: Wertebereich  $[-1, 1]$

Analog zur Schwellwertfunktion in Gleichung (B.3) kann die Rampenfunktion auch bzgl. Symmetrie zum Ursprung und Wertebereich  $[-1, 1]$  angepasst werden. Beide Transferfunktionen werden in Abb. B.3 gezeigt.

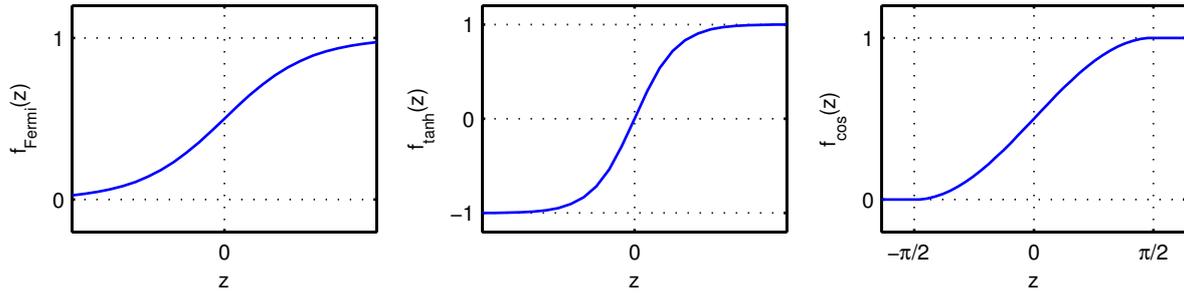
Stetige Transferfunktionen werden auch als *weiche Schwellwertfunktionen* bezeichnet. Zur Definition von Transferfunktion, bei denen nicht nur die Funktion selbst, sondern auch deren Ableitung stetig ist, haben sich *sigmoidale Transferfunktionen* zur Erzeugung von NNs etabliert. Mit Hilfe dieser Transferfunktionen lässt sich das Anstiegsverhalten der Aktivierung besser kontrollieren, obwohl an dieser Stelle vorweggenommen sei, dass sich das Verhalten der NNs nur unwesentlich verändert. Beispiele für sigmoidale Transferfunktionen sind die *Fermi-Funktion*

$$f_{\text{Fermi}} : z \mapsto (1 + e^{-Cz})^{-1}, \quad C \in \mathbb{R} \quad (\text{B.5})$$

und der *Tangens hyperbolicus* als bezüglich des Ursprungs symmetrisches Pendant

$$f_{\text{tanh}} : z \mapsto 2f_{\text{Fermi}}(2z) - 1 = \tanh(Cz), \quad C \in \mathbb{R}. \quad (\text{B.6})$$

Eine weitere weiche Schwellwertfunktion ist die *Kosinus-Quetschfunktion* (engl. *cosine*



**Abbildung B.4:** Sigmoidale Transferfunktionen, links: Fermi-Funktion, Mitte: Tangens hyperbolicus, rechts: Kosinus-Quetschfunktion

*squasher*)

$$f_{\cos} : z \mapsto \begin{cases} 1 & \text{für } z \geq \pi/2, \\ \frac{1}{2}(1 + \cos(z - \pi/2)) & \text{für } -\pi/2 < z < \pi/2, \\ 0 & \text{sonst.} \end{cases} \quad (\text{B.7})$$

Abbildung B.4 zeigt die beschriebenen sigmoidalen Transferfunktionen.

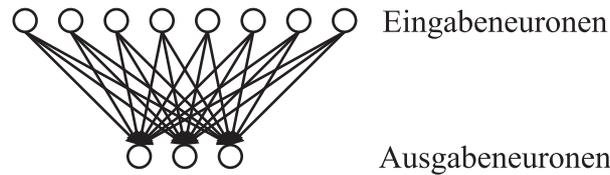
## B.3 Netzstrukturen und Klassifikation

Aufbauend auf der formalen Definition eines Neurons lässt sich ein NN als gerichteter Graph  $\mathcal{G} := (\mathcal{V}, \mathcal{E})$  mit Knotenmenge  $V := \{\mathcal{N}_1, \dots, \mathcal{N}_{N_N}\}$  aller  $N_N$  Neuronen und der Kantenmenge  $\mathcal{E} := \{(\mathcal{N}_i, \mathcal{N}_j) : \exists \text{ Verbindung } \mathcal{N}_i \rightarrow \mathcal{N}_j\}$  der Verbindungen zwischen den Neuronen definieren. Jede Kante  $\mathcal{N}_i \rightarrow \mathcal{N}_j$  ist mit der Gewichtung  $w_{i,j}$  aus Neuron  $\mathcal{N}_j$  versehen.

Eine besondere Rolle kommt den Neuronen zu, die keine Verbindung zu einem Vorgänger oder Nachfolger im Graphen besitzen. Es handelt sich hierbei einerseits um reine *Eingabeneuronen*  $\mathcal{L}_{\text{in}} := \{\mathcal{N}_j : (\mathcal{N}_{i=1, \dots, N_N}, \mathcal{N}_j) \notin \mathcal{E}\}$ , die lediglich als Datenquellen zur Eingabe von Merkmalsvektoren dienen. Ihre Eingänge besitzen keine Gewichtungen. Oft werden die Eingabeneuronen daher nicht weiter betrachtet und stattdessen lediglich von den Eingaben gesprochen. Bei den anderen Neuronen handelt es sich um *Ausgabeneuronen*  $\mathcal{L}_{\text{out}} := \{\mathcal{N}_i : (\mathcal{N}_i, \mathcal{N}_{j=1, \dots, N_N}) \notin \mathcal{E}\}$ . Sie dienen als Datensenke im Netz und liefern das berechnete Ergebnis bzw. das Klassifikationsresultat.

### B.3.1 Ein- und Mehrschichtige Netze

Üblicherweise sind NNs so strukturiert, dass paarweise disjunkte Teilmengen der Neuronen zu *Schichten* (engl. *layer*)  $\mathcal{L} \subseteq \mathcal{V}$  zusammengefasst werden. Die Menge der Eingabeneuronen bildet die Eingangsschicht  $\mathcal{L}_{\text{in}}$  und wird bei der Betrachtung der Schichten nicht



**Abbildung B.5:** Ein Perzeptron mit 10 Eingabe- und 3 Ausgabeneuronen.

mitgezählt. Die Ausgaben einer Schicht hängen ausschließlich von den Neuronen der davorgeschalteten Schicht ab, die durch die jeweiligen Gewichtungen  $\mathbf{w}$  beeinflusst werden. Die Ausgaben untereinander sind unabhängig.

Die einfachste Form von NNs bilden einschichtige Netze, die als *Perzeptron* bezeichnet werden. Abbildung B.5 zeigt ein solches Perzeptron. Betrachtet man ein einzelnes Ausgabeneuron des Perzeptrons, so ermöglicht dieses eine Klassifikation von Eingangsdaten  $\mathbf{x} \in \mathbb{R}^D$  an  $D = |\mathcal{L}_{\text{in}}|$  Eingabeneuronen durch lineare Separierung. Eine entscheidende Rolle übernehmen hierbei die Gewichtungen  $\mathbf{w}$  der Eingangsneuronen. Die Ausgabe des Neurons berechnet sich aus

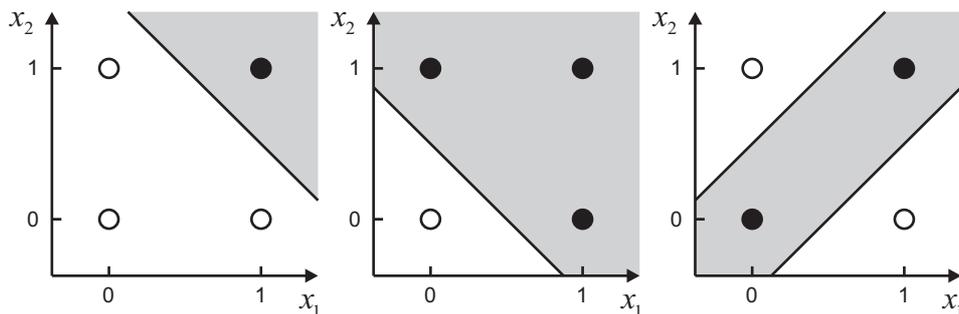
$$y = f\left(\sum_{i=0}^D w_i x_i\right) = f(\langle \mathbf{w}, \mathbf{x} \rangle). \quad (\text{B.8})$$

Es ist  $\langle \mathbf{w}, \mathbf{x} \rangle = 0$  die Hyperebene im Eingaberaum, an der die Trennung vorgenommen wird. Die Transferfunktion entscheidet, auf welcher Seite der Ebene sich der Eingabevektor  $\mathbf{x}$  befindet.

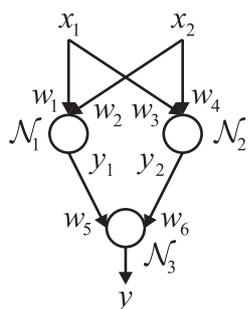
Geht man von einer Transferfunktion mit harter Schwellwertbildung  $f_{\text{hart}}$  (siehe Gleichung (B.3)) aus, so realisiert das Perzeptron eine Boolesche Funktion. Beispielsweise lassen sich die Funktionen AND und OR nicht aber die Funktion XOR realisieren (vgl. [135]). In Abbildung B.6 wird dies verdeutlicht.

Dieses Problem lässt sich durch Hinzufügen von weiteren Zwischenschichten umgehen. Man bezeichnet diese Schichten in *mehrschichtigen Neuronalen Netzen* als *verborgene Schichten* (engl. *hidden layers*). Durch die Einbeziehung zusätzlicher Schichten lassen sich Überlagerungen unterschiedlicher Transferfunktionen erzielen. Beispielsweise ermöglicht das mehrschichtige NN in Abbildung B.7 die Berechnung der Funktion XOR. Es sei angemerkt, dass die Funktion XOR auch mit einem einschichtigen Netz durch Multiplikation der Eingaben realisierbar ist, jedoch sind hierzu Neuronen mit höherer Funktionalität erforderlich.

Mit einer einzigen, ausreichend großen verborgenen Schicht und einer sigmoidalen Transferfunktion ist es möglich, jede beliebige stetige Funktion mit beliebiger Genauigkeit darzustellen (siehe [135], [64], [24]). Als Beispiel hierfür wird in Abbildung B.8 die mehrfache Überlagerung von sigmoidalen Transferfunktionen gezeigt.

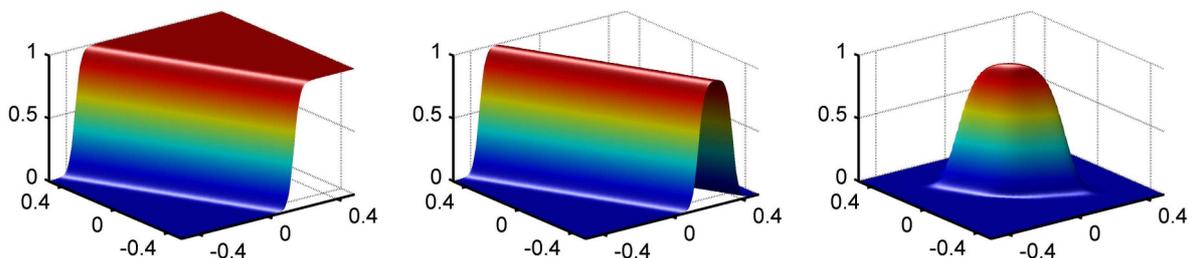


**Abbildung B.6:** Berechnung der Booleschen Funktionen  $x_1$  AND  $x_2$  (links) und  $x_1$  OR  $x_2$  (Mitte) mit einem Perzeptron durch lineare Separierung an der Hyperebene  $\langle \mathbf{w}, \mathbf{x} \rangle = 0$ . Die Berechnung von  $x_1$  XOR  $x_2$  (rechts) lässt sich auf diese Weise nicht durchführen. Das Ergebnis der Funktionen sind durch gefüllte (Wert 1) bzw. ungefüllte Kreise (Wert 0) dargestellt.



$x_1, x_2$ : Eingaben  
 $\mathcal{N}_1, \mathcal{N}_2$ : Neuronen  
 mit hartem Schwellwert  $C = 1/3$   
 $\mathcal{N}_3$ : Ausgabeneuron  
 mit hartem Schwellwert  $C = 1/3$   
 $w_1, w_4, w_5, w_6 = 1/2$   
 $w_2, w_3 = -1/2$

**Abbildung B.7:** Mehrschichtiges NN zur Berechnung von  $x_1$  XOR  $x_2$ .



**Abbildung B.8:** Links: sigmoidale Transferfunktion; Mitte: Überlagerung zweier entgegengesetzter sigmoidalen Transferfunktionen zu einer Welle; Rechts: Überlagerung zweier Wellen zu einem Stempel.

### B.3.2 Einlernen der Netz-Parameter

Das Prinzip des Parameterlernens von mehrschichtigen Netzen beruht auf der Anpassung der Gewichtungen zwischen den Neuronen, so dass der Ausgabefehler des Netzes für alle Trainingsdaten minimiert wird. Bei gegebenen Trainingsdaten  $(\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_{N_D}, \mathbf{y}_{N_D}) \in \mathbb{R}^{|\mathcal{N}_{in}(\mathcal{V})|, |\mathcal{N}_{out}(\mathcal{V})|}$  wird üblicherweise der quadratische Abstand zwischen dem vorgegebenen Zielwert  $\mathbf{y}_d$  und dem errechneten Wert  $\mathbf{y}$  als *Klassifikationsfehler*

$$E_d = \frac{1}{2} \sum_{i=1}^{|\mathcal{L}_{out}|} (y_{d,i} - y_i)^2, \quad d = 1, \dots, N_D \quad (\text{B.9})$$

berechnet.

Im Folgenden werden zur Vereinfachung zwei Annahmen gemacht. Einerseits wird für das NN eine sigmoidale Transferfunktion  $f$  verwendet, wobei die Differenzierbarkeit von  $f$  für die Herleitung und den angegebenen Algorithmus benötigt wird. Es lassen sich aber auch vergleichbare Lösungen für NNs mit nicht-differenzierbaren Transferfunktionen aus dem vorherigen Abschnitt angeben. Andererseits wird o. B. d. A. von einem zweischichtigen NN ausgegangen. Aufgrund ihrer ausreichenden Mächtigkeit finden diese die häufigste Verwendung in der praktischen Anwendung. Es lassen sich jedoch auch Erweiterungen für NNs mit mehr als zwei Schichten analog herleiten.

Im zweischichtigen NN mit Ein- und Ausgabeschichten  $\mathcal{L}_{in}$  und  $\mathcal{L}_{out}$  sowie einer verborgenen Schicht  $\mathcal{L}$  berechnet sich der Fehler pro Ausgabeneuron im Detail zu

$$E_d = \frac{1}{2} \sum_{i=1}^{|\mathcal{L}_{out}|} \left( y_{i,d} - f \left( \sum_{j=1}^{|\mathcal{L}|} w_{i,j} f \left( \sum_{k=1}^{|\mathcal{L}_{in}|} w_{j,k} x_{d,k} \right) \right) \right)^2. \quad (\text{B.10})$$

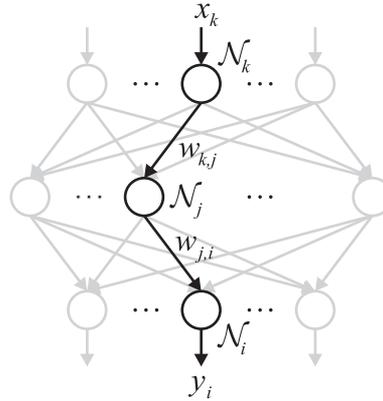
Hierbei sind die Bezeichnungen und Indizierungen für die Gewichte sowie Ein- und Ausgabewerte wie in Abbildung B.9 gewählt. Legt man die sigmoidale Transferfunktion zu Grunde, ist  $E_d$  beliebig oft bezüglich der Gewichtungen  $\mathbf{w}$  differenzierbar, so dass ein Gradienten-Abstiegsverfahren zur Minimierung der Fehler  $E_d$  durch Auswahl geeigneter Gewichte durchgeführt werden kann. Die Anpassung der Gewichte wird dabei schrittweise Schicht für Schicht durchgeführt, wobei der Fehler in der Ausgabeschicht bis zu den Gewichten in der Eingabeschicht propagiert wird. Der im Folgenden beschriebene Algorithmus wird daher als *Back-Propagation-Algorithmus* bezeichnet.

An dieser Stelle soll nur die Idee sowie der Algorithmus selbst angegeben werden. Eine genaue Beschreibung dieses Algorithmus wird in [15] gegeben. Zunächst wird die Differentiation von  $E_d$  bezüglich eines beliebigen Gewichts  $w_{j,i}$  zwischen zwei verbundenen Neuronen  $\mathcal{N}_i$  und  $\mathcal{N}_j$  untersucht:

$$\frac{\partial E_d}{\partial w_{j,i}} = \frac{\partial E_d}{\partial z_j} \frac{\partial z_j}{\partial w_{j,i}}. \quad (\text{B.11})$$

Mit Hilfe der Definition

$$\delta_j := \frac{\partial E_d}{\partial z_j} \quad (\text{B.12})$$



**Abbildung B.9:** Zweischichtiges NN mit den Bezeichnungen und Indizierungen für den Back-Propagation-Algorithmus.

und Gleichung (B.1) erhält man

$$\frac{\partial E_d}{\partial w_{j,i}} = \delta_j x_i, \quad (\text{B.13})$$

wobei  $x_i$  die Eingabe von Neuron  $\mathcal{N}_i$  sei. Für ein zusätzliches Neuron  $\mathcal{N}_k$ , das mit  $\mathcal{N}_j$  verbunden ist, erhält man mit demselben Ansatz und Umformung

$$\delta_k := \frac{\partial E_d}{\partial z_k} = \frac{\partial E_d}{\partial y_k} \frac{\partial y_k}{\partial z_k} = \frac{\partial E_d}{\partial y_k} \frac{\partial f(z_k)}{\partial z_k} = f'(z_k) \frac{\partial E_d}{\partial y_k}. \quad (\text{B.14})$$

Nun kann auch  $\delta_j$  für die vorgeschalteten Neuronen bestimmt werden:

$$\delta_j = \frac{\partial E_d}{\partial z_j} = \sum_k \frac{\partial E_d}{\partial z_k} \frac{\partial z_k}{\partial z_j} = \sum_k \delta_k \frac{\partial z_k}{\partial z_j} = f'(z_j) \sum_k w_{k,j} \delta_k. \quad (\text{B.15})$$

Hiermit lässt sich der Back-Propagation-Algorithmus für ein Trainingsdatum  $(x_d, y_d)$  wie folgt angeben:

---

**Algorithmus B.3.1** Der Back-Propagation-Algorithmus.

---

1. Berechnung der Ausgabe des NNs für einen Eingabevektor  $\mathbf{x}_d$ .
  2. Berechnung aller  $\delta_k$  für die Ausgabeneuronen mit Hilfe von Gleichung (B.14).
  3. Back-Propagation des Fehlers  $E_d$  und Berechnung der  $\delta_j$  der Eingabeneuronen mit Hilfe von Gleichung (B.15).
- 

Die Bestimmung des Gesamtfehlers für alle Trainingsdaten ergibt sich aus der Summation der Fehler  $E_1, \dots, E_{N_D}$  bei Anwendung aller Trainingsdaten. Zusätzlich wird eine *Lernrate*  $C_{LR}$  definiert, mit der gesteuert werden kann, wie stark sich das Netz an gegebene

Trainingsdaten anpassen kann:

$$C_{\text{LR}} \frac{\partial E}{\partial w_{j,i}} = C_{\text{LR}} \sum_{d=1}^{N_D} \frac{\partial E_d}{\partial w_{j,i}} =: \Delta w_{j,i}. \quad (\text{B.16})$$

Über das resultierende  $\Delta w_{j,i}$  lassen sich beim Training alle Gewichtungen  $w_{j,i}$  des NNs anpassen. Dieser Schritt wird in der Regel mehrfach durchgeführt. Die einzelnen Schritte bei dieser iterativen Parameteranpassung werden als *Epoche* bezeichnet.

Der asymptotische Berechnungsaufwand für den Back-Propagation-Algorithmus beträgt  $\mathcal{O}(|\mathcal{E}|)$  bei gegebener Anzahl  $|\mathcal{E}|$  von Verbindungskanten im Graphen bzw. Gewichtungen. Der lineare Aufwand in Abhängigkeit der Anzahl der Gewichte beruht auf der schichtweisen Berechnung der Neuronenausgaben und der Propagation des Berechnungsfehlers in entgegengesetzter Richtung. Eine Brute-Force-Berechnung der Gewichtsänderungen hätte im Gegensatz einen Aufwand von  $\mathcal{O}(|\mathcal{E}|^2)$  zur Folge. Nähere Erläuterungen zum Berechnungsaufwand des Trainings-Algorithmus werden in [15] gegeben.

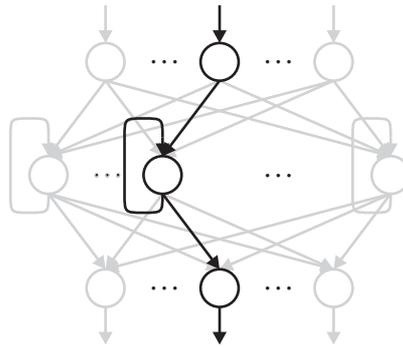
### B.3.3 Rückgekoppelte Neuronale Netze

Bei den in den vorigen Abschnitten angegebenen NNs werden die Ausgaben der Neuronen einer Schicht genau an die Neuronen der darauffolgenden Schicht weitergeleitet. Man nennt diese Netze mit zyklischem Graph  $\mathcal{G}$  daher auch *Feed-Forward-Netze*. Im Gegensatz dazu existieren Netzstrukturen, bei denen auch Rückkopplungen in Form von Rückwärtskanten in  $\mathcal{G}$  zugelassen sind. Diese werden als *rückgekoppelte Neuronale Netze* (engl. *recurrent* oder *feedback neural network*) bezeichnet.

Rückgekoppelte Netze erreichen selbst bei konstanter Eingabe nicht zwangsläufig einen konstanten Zustand. Eine Beschreibung der Dynamik des Netzes z. B. mit gekoppelten Differentialgleichungssystemen ist meist nicht durchführbar oder sehr schwierig. Die wichtigsten Ansätze zur Beherrschbarkeit insbesondere der Trainingsalgorithmen dieser Modelle entstammen der theoretischen Physik und der statistischen Mechanik in Form von *Hopfield-Netzen* und *Boltzmann-Maschinen*. Detaillierte Beschreibungen dieser Modelle sowie Erläuterungen zum Training mittels rückgekoppelter Back-Propagation finden sich in [64] und [24].

#### B.3.3.1 Elman-Netze

Als besonderen Vertreter der rückgekoppelten Netze wird an dieser Stelle auf so genannte *Elman-Netze* (ENs) (siehe [40]) eingegangen werden. In der Regel bestehen ENs aus zwei Schichten mit einer Rückkopplung von der Ausgabe der verborgenen Neuronen zu den Eingabeneuronen. Mit Hilfe dieser Rückkopplung wird das Ergebnis eines Klassifikationsschrittes beim darauffolgenden Klassifikationsschritt als Eingabe mitverwendet. Das Netz verfügt so über einen Zustandsspeicher, der insbesondere bei der Analyse von Zeitserien verwendet werden kann. Abbildung B.10 zeigt ein entsprechendes EN.



**Abbildung B.10:** Elman-Netz mit zwei Schichten und einer Rückkopplung innerhalb der ersten Schicht.

Bei der Implementierung der ENs in [35] wird in der ersten Schicht eine Fermi-Funktion (siehe Gleichung (B.5)) und in der zweiten Schicht eine lineare Transferfunktion (siehe Gleichung (B.4)) verwendet. Hiermit können beliebige Funktionen (mit einer endlichen Anzahl von Unstetigkeiten) mit beliebiger Genauigkeit approximiert werden. Einzige Voraussetzung ist eine ausreichende Anzahl von Neuronen bei zunehmender Komplexität der Funktion, die das Netz nachbilden soll.

### B.3.3.2 Training von Elman-Netzen

Als Trainingsalgorithmen für ENs werden in [35] zwei verschiedene Varianten des Back-Propagation-Algorithmus angegeben. Im einen Fall wird in jeder Epoche eine vollständige Sequenz von Eingabevektoren an das Netz angelegt, eine Sequenz von Fehlern durch Vergleich der Ausgabe mit der Zielklassifikation erzielt und schließlich der Back-Propagation-Algorithmus für alle Einträge in der Fehlersequenz angewendet, wobei die Rückkopplung ignoriert wird. Im zweiten Fall wird in jeder Epoche ein einzelner Trainingsvektor an das Netz angelegt und der Back-Propagation-Algorithmus sofort angewandt, wobei ebenfalls die Rückkopplung ignoriert wird. Da bei dieser Methode sich das Netz mit jeder Epoche zunehmend anpasst, wird diese auch als *Adaption* bezeichnet.

### B.3.4 Netze mit radialen Basisfunktionen

Bei den bisher betrachteten Netzmodellen berechnen die Neuronen als Ausgabe die Werte einer nicht-linearen Funktion für Skalarprodukte aus Eingabevektoren und Gewichtungsvektoren. Hierbei wird der Abstand der Eingabevektoren von der Trennebene bewertet. Alternativ lässt sich auch der Abstand zwischen Eingabevektor und einem gegebenen Zielvektor im Merkmalsraum bestimmen und als Aktivierung des Neurons ausgeben. Hierzu dienen *radiale Basisfunktionen (RBFs)*. Neuronale Netze, die Neuronen mit RBFs zur Berechnung ihrer Aktivierung verwenden, werden als *RBFNs* bezeichnet. Eine ausführliche Darstellung von RBFs und RBFNs findet sich [15].

### B.3.4.1 Radiale Basisfunktionen

Die ursprüngliche Anwendung von RBFs stammt aus dem Bereich der exakten Interpolation von mehrdimensionalen Merkmalsvektoren. Hierbei ist das Ziel für eine Menge  $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_{N_D}, y_{N_D})\}$  von Merkmalsvektoren  $\mathbf{x}_d \in \mathbb{R}^D$  mit zugehörigen Zielwerten  $y_d \in \mathbb{R}$ , eine Funktion  $h: \mathbb{R}^D \rightarrow \mathbb{R}$  mit der Eigenschaft

$$h(\mathbf{x}_d) = y_d \quad \text{für } d = 1, \dots, N_D \quad (\text{B.17})$$

zu finden. Die Methode der RBFs definiert zu diesem Zweck  $N_D$  nicht-lineare Basisfunktionen  $\phi: \mathbb{R}^D \rightarrow \mathbb{R}$ , deren Argument üblicherweise der Euklidische Abstand  $\|\mathbf{x} - \mathbf{x}_d\|$  eines Vektors  $\mathbf{x} \in \mathbb{R}^D$  vom jeweiligen Kontrollpunkt  $\mathbf{x}_d$  ist. Der Wert von  $h$  ergibt sich dann aus der Linearkombination dieser Basisfunktionen mit Gewichtungen  $\mathbf{w}$ :

$$h(\mathbf{x}) = \sum_{d=1}^{N_D} w_d \phi(\|\mathbf{x} - \mathbf{x}_d\|). \quad (\text{B.18})$$

Neben anderen Basisfunktionen (siehe [15]) werden am häufigsten Gauß-Funktionen

$$\phi(x) = e^{-\frac{1}{2\sigma^2}x^2} \quad (\text{B.19})$$

zur Erzeugung der Interpolationsfunktion in Gleichung (B.18) verwendet, die so eine differenzierbare Fläche repräsentiert. Die Glattheit der Fläche kann über die Varianz  $\sigma^2$  bestimmt werden.

### B.3.4.2 Berechnung der Ausgabe in RBFNs

Im Gegensatz zum Interpolationsproblem müssen bei der Erstellung von Netzen mit Hilfe der RBFs folgende Anforderungen an die Funktion zur Berechnung der Ausgabe eines NNs berücksichtigt werden:

1. Die Anzahl  $N_M$  der Basisfunktionen entspricht nicht der Anzahl  $N_D$  der gegebenen Trainingsvektoren  $\mathbf{x}_1, \dots, \mathbf{x}_{N_D}$ . Typischerweise ist  $N_M \ll N_D$ .
2. Die Zentren der Basisfunktionen sind nicht durch die Trainingsvektoren  $\mathbf{x}_1, \dots, \mathbf{x}_{N_D}$  gegeben, sondern stellen andere geeignete Zentren  $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_{N_M}$  dar.
3. Anstelle eines gemeinsamen Parameters  $\sigma^2$  für alle RBFs, besitzt jede RBF eine eigene Varianz  $\sigma^2$ .
4. Es werden wie bei den bisherigen Netzen zusätzliche Bias-Parameter  $w_0$  benötigt.

Gleichung (B.18) lässt sich leicht modifizieren, so dass die angegebenen Bedingungen für die Berechnung einer einzelnen Ausgabe eines RBFNs gelten:

$$y = \sum_{j=1}^{N_M} w_j \phi_j(\mathbf{x}) + w_0 \quad \text{mit} \quad \phi_j(\mathbf{x}) = e^{-\frac{1}{2\sigma_j^2} \|\mathbf{x} - \boldsymbol{\mu}_j\|^2}. \quad (\text{B.20})$$

Eine Verallgemeinerung der betrachteten RBFs mit einem Parameter  $\sigma^2$  stellen Basisfunktionen mit beliebiger Kovarianzmatrix  $\mathbf{U}$  dar. Diese entsprechen den Mischverteilungskomponenten in Abschnitt 2.3.2. Anstelle der Basisfunktion in Gleichung (B.20) wird stattdessen

$$\phi_j(\mathbf{x}) = e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_j)^\top \mathbf{U}_j^{-1}(\mathbf{x}-\boldsymbol{\mu}_j)} \quad (\text{B.21})$$

zur Berechnung des Aktivierungszustandes verwendet.

### B.3.4.3 Architektur und Training von RBFNs

RBFNs bestehen üblicherweise aus zwei Schichten, die zur Berechnung der Netzausgabe in Gleichung (B.20) verwendet werden. Die erste Schicht beinhaltet die Neuronen zur Berechnung der RBFs. Jedes Neuron bestimmt den Abstand der Eingabevektoren vom gegebenen Kontrollvektor und bewertet ihn mit der RBF. In der zweiten Schicht befinden sich gewöhnliche Neuronen zur Gewichtung der Basisfunktionen und zur Summation. Die Neuronen dieser Schicht haben die gleiche Gestalt wie die bereits bei mehrschichtigen Netzen verwendeten Neuronen (siehe Abschnitt B.3.1). Als Transferfunktion wird für die Neuronen der zweiten Schicht üblicherweise eine lineare Transferfunktion wie in Gleichung (B.4) verwendet.

Analog zur Architektur erfolgt auch das Training der RBFNs. Im ersten Schritt werden die RBFs und ihre Parameter bestimmt. Hierzu wird ein Greedy-Algorithmus verwendet, der inkrementell RBFs für die Trainingsbeispiele einfügt, die am schlechtesten mit den bisherigen RBFs klassifiziert werden können. Es handelt sich hierbei um ein unüberwachtes Lernen, d. h. nur die vorgegebenen Trainingsvektoren werden berücksichtigt, nicht aber die vorgegebenen Zielwerte. Im zweiten Schritt werden die Gewichtungen für die RBFs ermittelt. Dazu wird der Fehlerterm in Abhängigkeit vom Klassifikationsergebnis zu den Trainingsvektoren  $\mathbf{x}_d$  und entsprechendem Zielwert  $y_{d,k}$  über allen Netzausgaben  $y_k(\mathbf{x}_d)$  verwendet:

$$E = \frac{1}{2} \sum_d \sum_k (y_k(\mathbf{x}_d) - y_{d,k})^2. \quad (\text{B.22})$$

Die Minimierung des Fehlers und damit die Bestimmung passender Gewichte kann durch Lösung eines LGS durchgeführt werden. Sowohl der erste als auch der zweite Schritt lassen sich schnell berechnen, wodurch das Training von RBFNs sehr effizient gegenüber konventionellen mehrschichtigen Netzen ist.

# Anhang C

---

## Support-Vector-Machines

---

### C.1 Grundlagen

*Support-Vector-Machines (SVMs)* stellen ein Verfahren zur Klassifikation von Daten dar, das sich aus der statistischen Lerntheorie [159] ergeben hat. Die statistische Lerntheorie befasst sich mit der Erkennung von Regularitäten in Beobachtungen aus komplexen Systemen. Der statistische Zugang hat wie bei der Klassifikation mit HMMs den großen Vorteil, dass eine explizite Modellierung der den Systemen zugrunde liegenden Prozessen überhaupt nicht oder nur schwer möglich ist, dennoch aber Gesetzmäßigkeiten aufgrund empirischer Beobachtungen festgestellt werden können.

Die empirischen Beobachtungen dienen den SVMs als Trainingsmaterial. Sie seien im Folgenden durch die zwei Größen  $\mathbf{x} \in \mathbb{R}^D$ , den Merkmalsvektoren der Beobachtungen, und  $y \in Y \subseteq \mathbb{R}$ , den zugehörigen Zielklassifikationen, beschrieben. Zwischen  $\mathbf{x}$  und  $y$  bestehe ein deterministischer Zusammenhang, der durch die SVM erfasst werden soll. Die Trainingsdaten  $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_{N_D}, y_{N_D})\}$  seien gemäß einer unbekanntem Wahrscheinlichkeitsverteilung mit Auftrittswahrscheinlichkeit  $P(\mathbf{x}, y)$  für die Beobachtung von  $\mathbf{x}$  mit Zielklassifikation  $y$  erzeugt. Der Zusammenhang zwischen  $\mathbf{x}$  und  $y$  soll mit Hilfe einer *Entscheidungsfunktion*  $f : \mathbb{R}^D \rightarrow Y$  durch die SVM geschätzt werden. Die beste Schätzung ist die, welche das *Risiko*

$$R_f = \int |f(\mathbf{x}) - y|^C dP(\mathbf{x}, y) \quad (\text{C.1})$$

minimiert. Für den Fehlerterm ist typischerweise die Konstante  $C \in \{1, 2\}$ .

### C.2 Verwendung von Kernen für SVMs

Die Grundidee von SVMs ist, dass sich die Trainingsdaten  $\mathbf{x}_1, \dots, \mathbf{x}_{N_D}$  in den gegebenen Merkmalsräumen  $\mathbb{R}^D$  nur schwer dafür aber in Merkmalsräumen  $\mathbf{V}$  mit wesentlich höherer Dimension als  $D$  erheblich besser klassifizieren lassen. Dazu wird die nichtlineare

Abbildung

$$\begin{aligned}\Phi : \mathbb{R}^D &\rightarrow \mathbf{V}, \\ \mathbf{x} &\mapsto \Phi(\mathbf{x})\end{aligned}\tag{C.2}$$

definiert, welche die Trainingsdaten in den hochdimensionalen Merkmalsraum transformiert. Ziel ist es, in  $\mathbf{V}$  zwei gegebene Trainingsdaten durch eine Hyperebene  $\mathbf{w}$  mit minimaler VC-Dimension (siehe [158]) separieren zu können. Der Raum  $\mathbf{V}$  wird daher auch als *Linearisierungsraum* bezeichnet. Die Entscheidungsfunktion  $f$  lautet folglich

$$f(\mathbf{x}) = \text{sgn}(\langle \mathbf{w}, \Phi(\mathbf{x}) \rangle + b).\tag{C.3}$$

Die Berechnung des Skalarproduktes  $\langle \mathbf{w}, \Phi(\mathbf{x}) \rangle$  kann sehr aufwendig sein. Dieses Problem wird durch so genannte *Kernfunktionen* gelöst, die eine effiziente Auswertung der Entscheidungsfunktion ermöglichen. Zur Erläuterung wird an dieser Stelle zunächst ein einfaches Beispiel aus [143] verwendet, bei dem zweidimensionale Merkmalsvektoren mittels Monome in einen dreidimensionalen Merkmalsraum übertragen werden. Mit Hilfe der Abbildung

$$\begin{aligned}\Phi : \quad \mathbb{R}^2 &\rightarrow \mathbb{R}^3, \\ (x_1, x_2) &\mapsto (x_1^2, \sqrt{2}x_1x_2, x_2^2)\end{aligned}\tag{C.4}$$

kann in diesem Beispiel das Skalarprodukt  $\langle \Phi(\mathbf{x}), \Phi(\mathbf{y}) \rangle$  mit  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^2$  durch

$$\begin{aligned}\langle \Phi(\mathbf{x}), \Phi(\mathbf{y}) \rangle &= (x_1^2, \sqrt{2}x_1x_2, x_2^2)(y_1^2, \sqrt{2}y_1y_2, y_2^2)^\top \\ &= ((x_1, x_2)(y_1, y_2)^\top)^2 \\ &= \langle \mathbf{x}, \mathbf{y} \rangle^2 \\ &=: k(\mathbf{x}, \mathbf{y})\end{aligned}\tag{C.5}$$

berechnet werden.

Das Beispiel lässt sich wie folgt verallgemeinern. Für beliebige Vektoren  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^D$  mit  $D \in \mathbb{N}$  berechnet der Kern

$$k(\mathbf{x}, \mathbf{y}) := (\mathbf{x}, \mathbf{y})^n\tag{C.6}$$

ein Skalarprodukt im Raum der Monome mit Ordnung  $n \in \mathbb{N}$  zu den gegebenen Koordinaten. Ist  $k : M_C \times M_C$  ein stetiger Kern eines stetigen Integraloperators im Hilbertraum  $L_2(M_C)$  für eine kompakte Menge  $M_C \subset \mathbb{R}^D$ , d. h.

$$\forall f \in L_2(M_C) : \int_{M_C} k(\mathbf{x}, \mathbf{y}) f(\mathbf{x}) f(\mathbf{y}) d\mathbf{x} d\mathbf{y} \geq 0,\tag{C.7}$$

dann existiert ein Raum  $\mathbf{V}$  und eine Abbildung  $\Phi : \mathbb{R}^D \rightarrow \mathbf{V}$ , so dass  $k(\mathbf{x}, \mathbf{y}) = \langle \Phi(\mathbf{x}), \Phi(\mathbf{y}) \rangle$ . Dies kann direkt mit Hilfe des *Theorems von Mercer* (siehe [158]) nachgewiesen werden, das besagt, dass ein Kern eines positiven Integraloperators in seine Eigenfunktionen  $\Psi_j$  mit

$$k(\mathbf{x}, \mathbf{y}) = \sum_{j=1}^D \eta_j \Psi_j(\mathbf{x}) \Psi_j(\mathbf{y}) \quad \text{mit} \quad \eta_j > 0 \quad \text{und} \quad D \leq \infty\tag{C.8}$$

entwickelt werden kann. In diesem Fall ist

$$\Phi : \mathbf{x} \mapsto (\sqrt{\eta_1}\Psi_1(\mathbf{x}), \sqrt{\eta_2}\Psi_2(\mathbf{x}), \dots) \quad (\text{C.9})$$

die gesuchte Funktion, welche die Vektoren  $\mathbf{x}$  in den hochdimensionalen Raum  $\mathbf{V}$  abbildet. Neben den hier vorgestellten *polynomiellen Kernfunktionen* aus Gleichung (C.6) gibt es noch eine Vielzahl weiterer Familien von Kernfunktionen. Die am häufigsten eingesetzten Vertreter sind Gaußsche radiale Basisfunktionen (RBFs)

$$k(\mathbf{x}, \mathbf{y}) = e^{-\frac{1}{2\sigma^2}\|\mathbf{x}-\mathbf{y}\|^2} = e^{-C\|\mathbf{x}-\mathbf{y}\|^2} \quad \text{mit } C \in \mathbb{R}, \quad (\text{C.10})$$

*sigmoidale Kernfunktionen*

$$k(\mathbf{x}, \mathbf{y}) = \tanh(C_1 \langle \mathbf{x}, \mathbf{y} \rangle + C_2) \quad \text{mit } C_1, C_2 \in \mathbb{R} \quad (\text{C.11})$$

und *invers multiquadratische Kernfunktionen*

$$k(\mathbf{x}, \mathbf{y}) = \frac{1}{\sqrt{\|\mathbf{x} - \mathbf{y}\|^2 + C^2}} \quad \text{mit } C \in \mathbb{R}_+. \quad (\text{C.12})$$

Bemerkenswert ist, dass durch die Verwendung der Kernfunktionen aus Gleichung (C.10) und (C.11) sowie der resultierenden Merkmalsräume Spezialfälle von SVMs in Form von RBFNs- bzw. NNs aus Abschnitt B.3.4 entstehen. Welche Kernfunktion für eine bestimmte Klassifikationsaufgabe am besten geeignet ist, wird in der Praxis empirisch ermittelt. Hinweise hierzu sind in [67] zu finden.

## C.3 Bestimmung der Parameter zur Klassifikation

Die in Abschnitt C.1 beschriebene Entscheidungsfunktion  $f$  aus Gleichung (C.3) zur Separierung zweier Klassen von Trainingsdaten lässt sich mit Hilfe der Kern-Methode aus dem vorigen Abschnitt im Eingaberaum auswerten, ohne die Skalarprodukte im hochdimensionalen Merkmalsraum  $\mathbf{V}$  aufwendig zu bestimmen. Die Kern-Methode spielt ebenfalls eine entscheidende Rolle beim Einlernen der SVMs.

Für die Trainingsdaten  $\mathbf{x}_1, \dots, \mathbf{x}_{N_D}$  mit zugehörigen Zielwerten  $y_1, \dots, y_{N_D}$  kann die Entscheidungsfunktion geschrieben werden als

$$y_d = \text{sgn}(\langle \mathbf{w}, \mathbf{x}_d \rangle + b) \quad (\text{C.13})$$

mit der durch  $\mathbf{w} \in \mathbb{R}^D$  gegebenen separierenden Hyperebene. Die Bedingung, mit der die Klassifikationsentscheidung herbeigeführt werden kann, lautet dann für die gegebenen Trainingsdaten

$$y_d(\langle \mathbf{w}, \mathbf{x}_d \rangle + b) \geq 1, \quad d = 1, \dots, N_D. \quad (\text{C.14})$$

Für die Darstellung im hochdimensionalen Merkmalsraum ergibt sich

$$y_d(\langle \mathbf{w}, \Phi(\mathbf{x}_d) \rangle + b) \geq 1, \quad d = 1, \dots, N_D \quad (\text{C.15})$$

für die perfekte Klassifikation ohne Trainingsfehler anhand der durch  $\mathbf{w} \in \mathbf{V}$  gegebenen Hyperebene. Entsprechend müssen die Parameter  $\mathbf{w}$  und  $b$  für die SVM eintrainiert bzw. bestimmt werden.

Zur Bestimmung der Parameter wird in [143] das Polynom

$$L(\mathbf{w}, b, \zeta) = \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{d=1}^{N_D} \zeta_d (y_d \langle \mathbf{w}, \Phi(\mathbf{x}_d) \rangle + b) - 1, \quad (\text{C.16})$$

eingeführt, das sich aus dem bezüglich der VC-Dimension bzw. des Risikos in Gleichung (C.1) zu minimierenden Komplexitätsterm  $\frac{1}{2} \|\mathbf{w}\|^2$  und den durch Lagrange-Multiplikatoren  $\zeta_d \geq 0, d = 1, \dots, N_D$  gewichteten Bedingungen in Gleichung (C.15) ergibt. Der Ansatz zur Minimierung von Gleichung (C.16) bzgl. der Variablen  $\mathbf{w}$  und  $b$  führt zu den Gleichungen

$$\frac{\partial L}{\partial b} = 0 \quad \text{und} \quad \frac{\partial L}{\partial \mathbf{w}} = 0, \quad (\text{C.17})$$

die ausgerechnet

$$\sum_{d=1}^{N_D} \zeta_d y_d = 0 \quad \text{und} \quad \mathbf{w} = \sum_{d=1}^{N_D} \zeta_d y_d \Phi \mathbf{x}_d \quad (\text{C.18})$$

ergeben. Mit Hilfe der rechten Gleichung lässt sich  $\mathbf{w}$  durch einen Ausdruck mit  $\Phi(\mathbf{x}_d)$  in Gleichung (C.16) substituieren. Somit wird eine von  $\mathbf{w}$  unabhängige Darstellung erzielt. Mit Hilfe der Kernmethode lässt sich zusätzlich das Produkt  $\langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$  durch die Kernfunktion  $k(\mathbf{x}_i, \mathbf{x}_j)$  substituieren. Durch Einsetzen der linken Gleichung von (C.18) und der abschließenden Maximierung der Gleichung (C.16) bezüglich der Koeffizienten  $\zeta_{d=1, \dots, N_D}$  erhält man das Optimierungsproblem

$$\max_{\zeta} \left( \sum_{d=1}^{N_D} \zeta_d - \frac{1}{2} \sum_{i,j=1}^{N_D} \zeta_i \zeta_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) \right) \quad (\text{C.19})$$

mit den Nebenbedingungen

$$\zeta_d \geq 0 \quad \text{und} \quad \sum_{d=1}^{N_D} \zeta_d y_d = 0. \quad (\text{C.20})$$

Durch Lösung des Optimierungsproblems in Gleichung (C.19) erhält man die gesuchten Koeffizienten  $\zeta_{d=1, \dots, N_D}$ . Insgesamt lässt sich damit die Entscheidungsfunktion aus Gleichung (C.14) schreiben als

$$\begin{aligned} f(\mathbf{x}_d) &= \text{sgn} \left( \sum_{d=1}^{N_D} y_d \zeta_d \langle \Phi(\mathbf{x}), \Phi(\mathbf{x}_d) \rangle + b \right) \\ &= \text{sgn} \left( \sum_{d=1}^{N_D} y_d \zeta_d k(\mathbf{x}, \mathbf{x}_d) + b \right). \end{aligned} \quad (\text{C.21})$$

# Abbildungsverzeichnis

2.1	Beispiel für den Graphen einer Markov-Kette mit $N_S = 5$ Zuständen und die zugehörige Übergangsmatrix $\mathbf{A}$ . . . . .	9
2.2	Abhängigkeit der Zustandsübergänge und Beobachtungen eines HMMs ausgerollt über mehrere Zeitschritte. . . . .	10
2.3	Weg des Bohrers durch den Pedikel in den Wirbelkörper. . . . .	22
2.4	Schnitt durch den Wirbel mit Bereichen unterschiedlicher Dichte. . . . .	23
2.5	Raummikrofon CCM 2 der Fa. Schoeps und Oberflächenmikrofon C 562 BL der Fa. AKG. . . . .	24
2.6	Halterung für das Oberflächenmikrofon an einer Operationsklemme und Anbringung direkt am Wirbel. . . . .	25
2.7	Spektrogramm einer ausgewählten Bohrung. . . . .	26
2.8	Manuelle Klassifikation der Bohrgeräusche in einem ausgewählten Datensatz. . . . .	28
2.9	Ausgabe des NNs für den ausgewählten Datensatz. . . . .	29
2.10	Ausgabe der SVM für den ausgewählten Datensatz. . . . .	31
2.11	Ausgabe der HMMs für den ausgewählten Datensatz. . . . .	33
2.12	Unterscheidung der Klassen für Bohren im Knochen und Perforation des Wirbelkanals sowie die Modellwahrscheinlichkeiten zur Erkennung des Übergangs zwischen den Klassen. . . . .	34
3.1	Schematische Darstellung verschiedener Zustandsübergangsgraphen für HMMs mit $N_S = 4$ Zuständen. . . . .	41
3.2	Transmission, Reflexion und Brechung von Schallwellen an einer glatten ebenen Grenzfläche zwischen verschiedenen Geweben. . . . .	46
3.3	Artefakte in der medizinischen Ultraschallbildgebung. . . . .	52
3.4	Ultraschallbilder für die Analyse mit HMMs. . . . .	55
3.5	Beobachtungswahrscheinlichkeiten $\log P_{\lambda_T}(O)$ und $\log P_{\lambda_B}(O)$ für die Bildspalten der Einzelschichten aus Abb. 3.4. . . . .	58
3.6	Ergebnisse der harten und der weichen Klassifikation. . . . .	59

3.7	Darstellung der Ergebnisse der weichen Klassifikation durch Gewichtungen des Rot- und des Grünkanals in den originalen Ultraschallbildern. . . . .	59
3.8	HMM-Zustände zur Rekonstruktion von Gewebetypen. . . . .	60
3.9	HMM-Zustände zur Rekonstruktion von Gewebetypen mit Anwendung des Calderón-Operators im direkten Vergleich zu Abb. 3.8. . . . .	63
4.1	Aufbau zur Bestimmung von Objektkoordinaten aus Bildkoordinaten mit Hilfe der DLT. . . . .	69
4.2	Proband bei der Manipulation eines Objektes in sechs verschiedenen Kamerasichten. . . . .	70
4.3	Benutzeroberfläche von Simi-Motion zur Selektion von Markern in verschiedenen Kamerasichten. . . . .	71
4.4	Magnetisches Tracking. . . . .	72
4.5	Sensoren im Datenhandschuh und Erfassung von Arm und Hand. . . . .	74
4.6	Winkelverläufe für das rechte Kniegelenk bei der Gehbewegung zweier Probanden. . . . .	77
4.7	Strichdarstellung zweier Volleyballspieler beim Sprungaufschlag. . . . .	79
4.8	Unterschiedliche Skalierungsstufen des rechten Ellenbogenwinkels und zugehörige Wavelet-Koeffizienten für zwei Volleyballspieler beim Sprungaufschlag. . . . .	80
4.9	Detailinformationen (Wavelet-Koeffizienten) auf den Skalierungsstufen 1 bis 5 zu den Winkelverläufen aus Abb. 4.6 . . . . .	82
4.10	Extrahierte Merkmale zu den Winkelverläufen aus Abb. 4.6. . . . .	83
4.11	Winkelverläufe des Ellenbogens bei einer Greif- und Verschiebebewegung mit zugehörigen Wavelet-Koeffizienten auf den Skalierungsstufen 1 bis 5. . . . .	83
4.12	Extrahierte Merkmale zu den Winkelverläufen aus Abb. 4.11. . . . .	84
4.13	Erkennung einer unergonomischen Gehbewegung mit Hilfe der erweiterten Merkmale. . . . .	85
4.14	Bilder vom Versuchsaufbau und von ausgewählten Bewegungsphasen der untersuchten Alltagsbewegung. . . . .	87
4.15	Visualisierung der Komponenten des Modells für die ausgewählten Bewegungsphasen in Abb. 4.14. . . . .	88
4.16	Ausgewählte Gelenkwinkel einer Bewegung. . . . .	88
4.17	Erkennungsraten bei der Personenerkennung mit Beobachtungssequenzen unterschiedlicher Länge. . . . .	92
4.18	Ergebnisse der Phasenklassifikation mit HMMs mit mittlerer Erkennungsrate. . . . .	98
4.19	Ergebnisse der Phasenklassifikation mit HMMs mit hoher Erkennungsrate. . . . .	99

---

4.20	Ergebnisse der Phasenklassifikation mit RBFNs mit 1000 RBFs bei mittlerer Erkennungsrate. . . . .	100
4.21	Ergebnisse der Phasenklassifikation mit RBFNs mit 1000 RBFs bei hoher Erkennungsrate. . . . .	101
4.22	Zyklisches Modell für Gehbewegungen. . . . .	103
4.23	HMM-Zustände und Bewegungsphasen für das rechte Bein. . . . .	106
4.24	Ergebnisse der Phasenerkennung für das linke Bein im Vergleich zu denen für das rechte Bein in Abb. 4.23. . . . .	107
4.25	Ergebnisse der Phasenklassifikation mit einem zusammengesetzten HMM. Der Datensatz entspricht dem von Abb. 4.18. . . . .	115
4.26	Ergebnisse der Phasenklassifikation mit einem zusammengesetzten HMM. Der Datensatz entspricht dem von Abb. 4.19. . . . .	115
5.1	Verknüpfung und Gruppierung von Modulen auf der Benutzeroberfläche von VAMOS. . . . .	120
5.2	Schematischer Aufbau eines Moduls in VAMOS. . . . .	122
5.3	Aufbau und Verwaltung von Frame-Daten in VAMOS. . . . .	124
5.4	Ein Beispiel für die partielle Duplikation von Frames mit der Klasse <code>ShareableList</code> . . . . .	124
5.5	Bildschirmdarstellung eines Schemas zur Bewegungsanalyse in VAMOS mit Benutzeroberflächen der eingesetzten Module. . . . .	125
B.1	Schematische Darstellung eines Neurons, der elementaren Einheit eines künstlichen Neuronalen Netzes. . . . .	143
B.2	Transferfunktionen mit hartem Schwellwert. . . . .	145
B.3	Begrenzt lineare Transferfunktionen. . . . .	145
B.4	Sigmoidale Transferfunktionen. . . . .	146
B.5	Ein Perzeptron mit 10 Eingabe- und 3 Ausgabeneuronen. . . . .	147
B.6	Berechnung Boolescher Funktionen durch lineare Separierung an Hyper-ebenen. . . . .	148
B.7	Mehrschichtiges NN zur Berechnung von $x_1$ XOR $x_2$ . . . . .	148
B.8	Sigmoidale Transferfunktionen und deren Überlagerung. . . . .	148
B.9	Zweischichtiges NN mit den Bezeichnungen und Indizierungen für den Back-Propagation-Algorithmus. . . . .	150
B.10	Elman-Netz mit zwei Schichten und einer Rückkopplung innerhalb der ersten Schicht. . . . .	152



# Tabellenverzeichnis

2.1	Die Klassen 1–9 von Bohrgeräuschen für die manuelle Klassifikation. . . . .	28
2.2	Erkennungsraten $P(i, j)$ des 2-schichtigen NNs bei ausgegebener Klasse $j$ und vorgegebener Klasse $i$ über alle Datensätze und Geräuschklassen $i, j \in \{1, \dots, 9\}$ . . . . .	29
2.3	Erkennungsraten $P(i, j)$ der mit 32 Merkmalen trainierten SVM bei ausgegebener Klasse $j$ und vorgegebener Klasse $i$ über alle Datensätze und Geräuschklassen $i, j \in \{1, \dots, 9\}$ . . . . .	30
2.4	Erkennungsraten $P(i, j)$ der HMMs bei ausgegebener Klasse $j$ und vorgegebener Klasse $i$ über alle Datensätze und Geräuschklassen $i, j \in \{1, \dots, 9\}$ . . . . .	33
2.5	Erkennungsraten $P(i, j)$ der HMMs bei ausgegebener Klasse $j$ und vorgegebener Klasse $i$ über alle Datensätze und die zusammengefassten Geräuschklassen. . . . .	33
3.1	Gewebespezifische Werte für Schallgeschwindigkeit $c$ , Dichte $\rho$ , Schallimpedanz $Z$ und Dämpfung (Quelle: [111]). . . . .	47
3.2	Typische Eindringtiefen und Auflösungen bei vorgegebenen Einstrahlfrequenzen in der Ultraschalldiagnostik (Quelle: [111]). . . . .	49
4.1	Das verwendete Handmodell: Längen der Fingersegmente in mm. . . . .	75
4.2	Wahrscheinlichkeiten für die Erkennung von Personen durch HMMs mit $N_S = 20$ Zuständen. . . . .	90
4.3	Wahrscheinlichkeiten für die Erkennung von Personen durch HMMs mit $N_S = 1$ Zustand. . . . .	91
4.4	Wahrscheinlichkeiten für die Erkennung der Rahmenbedingung (Füllstand der Kaffeekanne) durch HMMs mit $N_S = 40$ Zuständen. . . . .	93
4.5	Mittlere Erkennungsraten bei der Identifikation von Rahmenbedingungen mit unterschiedlichen Anzahlen von Zuständen $N_S$ pro HMM. . . . .	93
4.6	Erkennungsraten für Elman-Netze mit 15 Neuronen. . . . .	94
4.7	Wahrscheinlichkeiten für korrekte Klassifikation bei verschiedenen Anzahlen von Neuronen $N_N$ in der ersten Schicht der Elman-Netze. . . . .	95

---

4.8	Erkennungsraten bei der Klassifikation von Bewegungsphasen bei HMMs mit unterschiedlichen Parametrisierungen. . . . .	97
4.9	Erkennungsraten von RBFNs zur Phasenklassifikation bei unterschiedlichen Anzahlen von Basisfunktionen. . . . .	100
4.10	Vergleich der Erkennungsraten von Elman-Netzen, RBFNs und HMMs für einzelne Bewegungsphasen. . . . .	102
4.11	Mittlere Längen und Übergangswahrscheinlichkeiten der Bewegungsphasen.	113
4.12	Erkennungsraten bei der Klassifikation von Bewegungsphasen durch zusammengesetzte HMMs mit unterschiedlichen Parametrisierungen. . . . .	114
4.13	Erkennungsraten bei der Klassifikation des Füllstandes der Kaffeekanne durch zusammengesetzte HMMs mit unterschiedlichen Parametrisierungen.	116
A.1	Filterkoeffizienten für $\varphi_D$ (Quelle: [91]) . . . . .	140

# Symbolverzeichnis

$\forall$	Allquantor
$\exists$	Existenzquantor
$\{x_1, x_2, \dots, x_N\}$	Menge aus $N$ Elementen
$x_1 x_2 \dots x_N$	geordnete Folge aus $N$ Elementen
$ M $	Kardinalität einer Menge $M$
$\bar{M}$	Abschluss einer Menge $M$
$(x, y)^n$	Monome der Ordnung $n$ mit Variablen $x, y$
$\bar{x}$	komplexes Konjugat von $x$
$ \cdot $	Absolutbetrag für Skalarwerte
$\ \cdot\ $	Euklidische Vektornorm
$\langle \cdot, \cdot \rangle$	Skalarprodukt
$\mathbf{M}^{-1}$	Inverse zu Matrix $\mathbf{M}$
$\mathbf{M}^\top$	Transponierte zu Matrix $\mathbf{M}$
$\mathbf{x}^\top$	transponierter Vektor $\mathbf{x}$
$\oplus$	Direkte Summe auf Vektorräumen
$f * g$	Faltung von $f$ mit $g$
$\hat{f}$	Fourier-Transformierte der Funktion $f$
$\hat{\hat{f}}$	inverse Fourier-Transformierte der Funktion $\hat{f}$
$\hat{f}_w$	gefensterte Fourier-Transformierte der Funktion $f$
$a$	Aktivierungszustand eines Neurons
$a_p$	Übergangswahrscheinlichkeit der Phase $p$
$a_{i,j}$	Eintrag in Zustandsübergangsmatrix $\mathbf{A}$ für Zustände $i, j = 1, \dots, N$
$\tilde{a}_{i,j}$	Neuabschätzung für $a_{i,j}$

$\check{a}_{i,j}$	Eintrag in Adjazenzmatrix $\mathbf{A}$
$\mathbf{A} = (a_{i,j})$	Zustandsübergangsmatrix
$\check{\mathbf{A}}$	Adjazenzmatrix
$\mathbf{A}_p$	elementare Zustandsübergangsmatrix zu Phase $p$
$\alpha_t(i)$	Vorwärtswahrscheinlichkeit zum Zeitpunkt $t$ für Zustand $s_i$
$\check{\alpha}_t(i)$	skalierte Vorwärtsvariable
$\alpha'_t(i)$	Hilfsvariable zur Bestimmung der skalierten Vorwärtsvariablen
$b_i(\mathbf{o})$	Wahrscheinlichkeit für die Beobachtung von $\mathbf{o}$ in Zustand $s_i$
$b_{j,k}$	Eintrag in Lookup-Table für Beobachtungswahrscheinlichkeit des Symbols $v_k$ in Zustand $s_j$
$\tilde{b}_{j,k}$	Neuabschätzung für $b_{j,k}$
$\mathbf{B} = (b_{j,k})$	Matrix bzw. Lookup-Table für Beobachtungswahrscheinlichkeiten
$\text{BE}_{\nu_i, \nu_{i+1}}(t)$	Bandenergie im Frequenzbereich $[\nu_i, \nu_{i+1}]$ zum Zeitpunkt $t$
$\text{BER}_{\nu_i, \nu_{i+1}}(t)$	Bandenergieverhältnis im Frequenzbereich $[\nu_i, \nu_{i+1}]$ zum Zeitpunkt $t$
$\beta_t(i)$	Rückwärtswahrscheinlichkeit zum Zeitpunkt $t$ für Zustand $s_i$
$\check{\beta}_t(i)$	skalierte Rückwärtsvariable
$\text{BW}(t)$	Bandbreite zum Zeitpunkt $t$
$c$	bei Physik des Schalls: Schallgeschwindigkeit, sonst: Index für Klasse oder Rahmenbedingung
$c_{\text{hard}}(O)$	harte Klassifikation einer Bildspalte $O$
$c_{\text{soft}}(O)$	weiche Klassifikation einer Bildspalte $O$
$\check{c}_T$	Produkt über Skalierungen der Vorwärtsvariablen
$c_t$	Skalierung für Vorwärtsvariable zum Zeitpunkt $t$
$C$	eine Konstante
$C_A$	Absorptionskoeffizient
$C_\alpha$	Konstante für Abschätzung der Pseudodifferential-Eigenschaft von Calderón-Operatoren
$C_{\text{diag}}$	heuristischer Wert für Hauptdiagonale
$C_\epsilon$	Konstante für das Abklingverhalten von Wavelets
$C_{\text{LR}}$	Lernrate bei Neuronalen Netzen
$C_R$	Reflexionskoeffizient
$C_T$	Transmissionskoeffizient

$\mathbb{C}$	Körper der komplexen Zahlen
$\mathcal{C}^n$	Menge der $n$ -fach differenzierbaren Funktionen
$\mathcal{C}_{\alpha, \Omega_e}$	Calderón-Operator
$d$	Index für Test- oder Trainingsdaten
$D$	Dimension oder Ordnung
$\Delta P(x)$	Verhältnis betrachteter Modellwahrscheinlichkeiten für Bildspalte $O_x$
$\Delta w_{j,i}$	Abweichung zur Bestimmung der korrekten Gewichte von Neuronalen Netzen
$\delta_t(j)$	Wahrscheinlichkeit für Übergang in Folgezustand $s_j$ zum Zeitpunkt $t$
$\delta$	Hilfsvariable zur Berechnung des Fehlers bei Neuronalen Netzen
$e$	Eulersche Zahl
$E$	Gesamtklassifikationsfehler über alle Trainingsdaten
$E_d$	Klassifikationsfehler bei Trainingsdatum mit Index $d$
$\mathcal{E}$	Kantenmenge
$e$	Einstrahlrichtung
$\varepsilon$	ein kleinster Wert $> 0$
$\epsilon$	Exponent für das Abklingen von Wavelets
$\eta_j$	Skalierung zur Berechnung von $\Phi$ mit $\Psi_j$
$f$	eine Funktionen
$f_{\cos}$	Kosinus-Quetschfunktion
$f_{\text{Fermi}}$	Fermi-Funktion
$f_{\text{hard}}$	Transferfunktion mit hartem Schwellwert
$f_{\text{linear}}$	begrenzt lineare Transferfunktion
$f_{\text{tanh}}$	Transferfunktion basierend auf Tangens hyperbolicus
$\mathcal{F}$	Fourier-Transformation
$\mathcal{F}_w$	gefensterte Fourier-Transformation
$\mathcal{F}^{-1}$	inverse Fourier-Transformation
$g$	eine Funktion
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$	Graph mit Knotenmenge $\mathcal{V}$ und Kantenmenge $\mathcal{E}$
$\gamma_t(i)$	Wahrscheinlichkeit für Zustand $s_i$ zu Zeitpunkt $t$
$\gamma_{m,t}(i)$	Wahrscheinlichkeit für Zustand $s_i$ zu Zeitpunkt $t$ bzgl. Mischverteilungskomponente $m$

$\gamma_{d,m,t}(i)$	$\gamma_{m,t}(i)$ für Beobachtungssequenz $O_d$
$h$	eine Funktion
$i$	Zählindex
$\iota$	imaginäre Einheit
$j$	Zählindex
$J$	Schallintensität
$k$	Zählindex
$k(\cdot, \cdot)$	Kernfunktion
$\kappa$	Markov-Modell
$l$	Zählindex
$L_{1,\dots,11}$	Kameraparameter der DLT-Methode
$L_2(M_C)$	Hilbertraum auf einer kompakten Menge $M_C$
$L$	Hilfspolynom zur Parameterbestimmung der SVM
$L^2(\mathbb{R})$	Raum der quadratisch integrierbaren Funktionen auf $\mathbb{R}$
$\mathcal{L}$	Schicht eines Neuronalen Netzes
$\lambda$	bei Physik des Schalls: Wellenlänge, sonst: generelle Bezeichnung für ein HMM
$\tilde{\lambda}$	trainiertes HMM zur Unterscheidung vom untrainierten Modell $\lambda$
$\lambda_B$	HMM für Strukturen ohne Tumoren
$\lambda_c$	HMM zur Erkennung von Rahmenbedingung $c$
$\lambda_p$	HMM zur Erkennung von Bewegungsphase $p$
$\lambda_u$	HMM zur Erkennung von Person $u$
$\lambda_T$	HMM für Strukturen mit Tumoren
$\lambda_{\text{trans}}$	HMM zur Erkennung des Übergangs zwischen Bohrphasen
$\lg$	dekadischer Logarithmus
$\log$	natürlicher Logarithmus
$m$	Index für Mischverteilungskomponente
$M_C$	eine kompakte Menge
$\mu$	Erwartungswert
$\boldsymbol{\mu}$	Erwartungsvektor, Zentrum einer radialen Basisfunktion oder Mischverteilungskomponente
$n$	Zählindex

$N_C$	Anzahl der Klassen oder Rahmenbedingungen
$N_D$	Anzahl der Test- oder Trainingsdaten
$N_\lambda$	Anzahl der HMMs
$N_M$	Anzahl der radialen Basisfunktionen oder der Mischverteilungskomponenten
$N_P$	Anzahl der Bewegungsphasen
$N_S$	Anzahl der Zustände
$N_U$	Anzahl der Testpersonen
$N_V$	Anzahl der beobachtbaren Symbole
$\mathcal{N}$	Normalverteilung
$\mathbb{N}$	Menge der natürlichen Zahlen
$\mathcal{N} = (\mathbf{x}, \mathbf{w}, f)$	ein Neuron mit Eingangsvariablen $\mathbf{x}$ , Gewichtungen $\mathbf{w}$ und Transferfunktion $f$
$\mathcal{N}_{\text{in}}$	Eingabeneuron
$\mathcal{N}_{\text{out}}$	Ausgabeneuron
$\nu$	eine Frequenz
$\bar{\nu}(t)$	mittlere Frequenz zum Zeitpunkt $t$
$o$	skalare Beobachtungsvariable
$\mathbf{o}$	vektorwertige Beobachtungsvariable
$\mathbf{O}$	Trainings- oder Testdatenmenge
$O_d$	$d$ -te Observablensequenz aus einer Trainings- oder Testdatenmenge
$\mathbf{o}_{d,t}$	Observable zum Zeitpunkt $t$ in Observablensequenz $O_d$
$\mathcal{O}$	Landau-Symbol
$\Omega_{\mathbf{e}}$	Richtungsgewichtung für Calderón-Operatoren mit Vorzugsrichtung $\mathbf{e}$
$\boldsymbol{\omega}$	Vektor im Fourier-Raum
$p$	Index für Bewegungsphase
$P(A)$	Wahrscheinlichkeit für Ereignis $A$
$P(A B)$	bedingte Wahrscheinlichkeit für Ereignis $A$ bei gegebenem Ereignis $B$
$P(A, B)$	Verbundwahrscheinlichkeit für Ereignisse $A$ und $B$
$P_c(O, t)$	Wahrscheinlichkeit für die korrekte Erkennung einer Klasse $c$ anhand einer Beobachtungsfolge $O$ zum Zeitpunkt $t$
$P(i, j)$	Wahrscheinlichkeit für die Erkennung eines Merkmalsvektors der Klasse $j$ bei vorgegebener manueller Klassifikation $i$

$P_i(T)$	Wahrscheinlichkeit für das Verweilen in Zustand $s_i$ mit Dauer $T$
$P_\kappa(Q)$	Wahrscheinlichkeit für die Beobachtung der Zustandsfolge $Q$ mit Markov-Modell $\kappa$
$P_\lambda(O)$	Wahrscheinlichkeit für die Beobachtung von $O$ mit HMM $\lambda$
$P_\lambda(Q)$	Wahrscheinlichkeit für die Zustandsfolge $Q$ in HMM $\lambda$
$P_\lambda(Q O)$	A-posteriori-Wahrscheinlichkeit für die Zustandsfolge $Q$ bei Beobachtung von $O$ in HMM $\lambda$
$P_S$	rückgestreute Leistung
$P(\mathbf{x}, y)$	Auftrittswahrscheinlichkeit für die Beobachtung von $\mathbf{x}$ mit Zielklassifikation $y$
$\phi$	Gauß-Funktion
$\varphi$	Skalierungsfunktion für Wavelets
$\Phi$	Funktion zur Transformation in hochdimensionalen Merkmalsraum
$\pi$	Ludolfsche Zahl
$\boldsymbol{\pi}$	initiale Zustandswahrscheinlichkeiten
$\tilde{\pi}_i$	Neuabschätzung für $\pi_i$
$\psi$	Wavelet-Funktion
$\psi_t(j)$	Merker für Zustand bei Übergang in Folgezustand $s_j$ zum Zeitpunkt $t$
$\Psi_j$	Eigenfunktion zur Berechnung von $\Phi$
$q$	Zustandsvariable
$Q = q_1 q_2 \dots q_T$	Zustandsfolge
$r$	Radius
$r_S$	Abstand vom Streuer
$R_f$	Risikio für Schätzung durch Entscheidungsfunktion $f$
$\mathbf{R}$	Rotationsmatrix
$\mathbb{R}$	Körper der reellen Zahlen
$\mathbb{R}_+$	Menge der positiven reellen Zahlen
$\rho$	Dichte
$s$	ein Zustand
$S = \{s_1, \dots, s_N\}$	Zustandsraum
$\mathbf{s}$	Translationsvektor
sgn	Vorzeichenfunktion

$SO_3$	Gruppe der Rotationsmatrizen im $\mathbb{R}^3$
$\sigma$	Standardabweichung
$\sigma^2$	Varianz
$\sigma_S$	Streuquerschnitt
$\sigma_{\text{soft}}$	Signifikanz der weichen Klassifikation
$t$	zeitlicher Index, Zeitpunkt
$T$	größter zeitlicher Index, Dauer
$\bar{T}_i$	Mittlere Verweildauer in Zustand $s_i$
$T_{\text{Median}}$	Länge des verwendeten Medianfilters
$T_{\text{Offset}}$	Offset für Fensterung
$T_p$	Länge der Bewegungsphase $p$
$T_{\text{Shift}}$	Anzahl der Frames für Fensterverschiebung
$T_w$	Länge eines Fensters $w$
$T_{w_B}$	Länge des Blackman-Filters $w_B$
$T_W$	Fensterlänge
$\tau$	zeitlicher Hilfsindex
$\theta$	ein Winkel
$\vartheta$	schnell abklingende Funktion zur Bestimmung der Ableitungseigenschaft von Wavelets
$\theta(\boldsymbol{\omega}, \mathbf{e})$	Winkel zwischen den Richtungsvektoren $\mathbf{e}$ und $\boldsymbol{\omega}$
$\theta_{\text{dist}}$	Flexionswinkel des distalen Fingergelenkes
$\theta_{\text{prox}}$	Flexionswinkel des proximalen Fingergelenkes
$\theta_S$	Streuwinkel
$\tilde{\vartheta}_s$	skalierte Version von $\vartheta$
$u$	Index für Person
$\mathbf{U}$	Kovarianzmatrix
$v$	beobachtbares Symbol
$v_{n,k}$	Skalierungskoeffizient bei Wavelet-Transformation
$V = v_1, \dots, v_{N_V}$	Menge der beobachtbaren Symbole
$\mathbf{V}$	allgemeiner Vektorraum oder Unterraum, (hochdimensionaler) Merkmalsraum

---

$\mathcal{V}$	Knotenmenge
$\text{Vol}(t)$	Audio-Volumen zum Zeitpunkt $t$
$w$	Gewichtung oder Fensterfunktion
$w_B$	Blackman-Fenster
$\mathbf{w}$	Gewichtungsvektor
$w_{n,k}$	Wavelet-Koeffizient bei Wavelet-Transformation
$\mathbf{W}$	allgemeiner Vektorraum oder Unterraum, Merkmalsraum
$\mathcal{W}$	Wavelet-Transformation
$x$	Index für Bildspalten
$X$	maximaler Spaltenindex in einem Bild
$\mathbf{x}$	Merkmals- oder Trainingsvektor
$\xi_t(i, j)$	Wahrscheinlichkeit für Zustände $i$ und $j$ zu den Zeitpunkten $t$ und $t+1$
$y$	Aktivierungszustand eines Neurons bzw. Ausgabe eines Elman-Netzes
$y_d$	Zielfunktion für ein Elman-Netz bei Trainingsdatum mit Index $d$
$Y$	Wertebereich einer Transferfunktion
$\mathbf{y}$	Merkmals- oder Ausgabevektor
$\mathbf{y}_d$	Zielvektor beim Training
$\mathbf{y}(\mathbf{x})$	Ausgabevektor in Abhängigkeit von $\mathbf{x}$
$z$	Summe der gewichteten Eingaben eines Neurons
$Z$	Schallimpedanz
$\mathbb{Z}$	Menge der ganzen Zahlen
$\zeta_d$	Lagrange-Multiplikatoren zur Parameterbestimmung von SVMs

# Abkürzungsverzeichnis

<b>A-Mode</b>	Amplitudenmodus zur Darstellung von Ultraschalldaten
<b>AR</b>	autoregressive Modelle bzw. Prozesse
<b>ARIMA</b>	Erweiterung von ARMA, engl. <i>autoregressive integrated moving average</i>
<b>ARMA</b>	Kombination aus AR und MA
<b>B-Mode</b>	Grauwertmodus zur Darstellung von Ultraschalldaten, engl. <i>brightness mode</i>
<b>BLTP</b>	Aufbereitungsverfahren für Ultraschalldaten, engl. <i>binarization low-pass threshold propagation</i>
<b>CORBA</b>	engl. <i>common object request broker architecture</i>
<b>CT</b>	Computertomographie
<b>DGC</b>	Tiefenkompensation, engl. <i>depth gain compensation</i>
<b>DGL</b>	Differentialgleichung
<b>DLT</b>	engl. <i>direct linear transformation technique</i>
<b>DOF</b>	Freiheitsgrad, engl. <i>degree of freedom</i>
<b>DWT</b>	diskrete Wavelet-Transformation
<b>EDA</b>	explorative Datenanalyse
<b>EMG</b>	Elektromyogramm
<b>EM</b>	engl. <i>expectation maximization</i>
<b>EN</b>	Elman-Netz
<b>FIR-Filter</b>	Filter mit korrespondierendem Faltungskern endlicher Länge, engl. <i>finite impulse response filter</i>
<b>FFT</b>	Schnelle Fourier-Transformation, engl. <i>fast Fourier transform</i>
<b>HMM</b>	Hidden-Markov-Modell
<b>LED</b>	Leuchtdiode, engl. <i>light emitting diode</i>
<b>MA</b>	Modelle bzw. Prozesse mit <i>Moving-Average</i>

<b>MAP</b>	Maximum-a-posteriori-Schätzung
<b>ML</b>	Maximum-Likelihood
<b>MRT</b>	Magnetresonanztomographie
<b>MSA</b>	Multiskalenanalyse
<b>NN</b>	Neuronales Netz
<b>ONB</b>	Orthonormalbasis
<b>RBFN</b>	Neuronales Netz mit radialen Basisfunktionen
<b>RBF</b>	radiale Basisfunktion
<b>STFT</b>	gefensterte Fourier-Transformation, engl. <i>short-time Fourier transform</i>
<b>SVD</b>	Singulärwertzerlegung, engl. <i>singular value decomposition</i>
<b>SVM</b>	engl. <i>support vector machine</i>
<b>TM-Mode</b>	zeitliches Auftragen von Grauwertspalten zur Darstellung von Ultraschalldaten, engl. <i>time motion mode</i>
<b>VAMOS</b>	Visualisierungs-, Analyse- und Modellierungssystem
<b>XML</b>	engl. <i>extensible markup language</i>

# Literaturverzeichnis

- [1] ABDEL-MALEK, A. ; DENTINGER, A.: Enhanced Method for Reducing Ultrasound Speckle Noise Using Wavelet Transform. In: *Wavelet Applications in Signal and Image Processing, SPIE*, 1997, S. 92–103
- [2] ACHIM, A. ; BEZERANIOS, A. ; TSAKALIDES, P.: Novel Bayesian Multiscale Method for Speckle Removal in Medical Ultrasound Images. In: *IEEE Transactions on Medical Imaging* 20 (2001), Nr. 8, S. 772–783
- [3] AIAZZI, S. ; ALPARONE, L. ; BARONTI, S.: Multiresolution Local Statistics Speckle Filtering Based on a Ratio Laplacian Pyramid. In: *IEEE Transaction on Geoscience and Remote Sensing* 36 (1998), Nr. 5, S. 1466–1476
- [4] ALLARD, P. (Hrsg.) ; STOKES, I. A. F. (Hrsg.) ; BLANCHI, J.-P. (Hrsg.): *Three-Dimensional Analysis of Human Movement*. Human Kinetics, Champaign, 1995
- [5] ALPARONE, L. ; BARONTI, S. ; CARLA, R.: Adaptive Rank-Conditioned Median Filter for Edge-Preserving Image Smoothing. In: *Proceedings of the SPIE* Bd. 2955, 1996, S. 244–256
- [6] ASHIDA, K. ; LEE, S.-J. ; ALLBECK, J. ; SUN, H. ; BADLER, N. ; METAXAS, D.: Pedestrians: Creating Agent Behaviours through Statistical Analysis of Observation Data. In: *Computer Animation 2002 (CA '01), Seoul*, 2001, S. 84–92
- [7] ASHRAF, G. ; WONG, K.: Constrained Framespace Interpolation. In: *Computer Animation 2002 (CA '01), Seoul*, 2001, S. 61–72
- [8] BEHRENDTS, E.: *Introduction to Markov Chains*. Vieweg, Braunschweig/Wiesbaden, 2000 (Advanced Lectures in Mathematics)
- [9] BENNEWITZ, M. ; THRUN, W. Burgard S.: Learning Motion Patterns of Persons for Mobile Service Robots. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2002, S. 3601–3606
- [10] BETH, T.: *Verfahren der schnellen Fourier-Transformation*. Teubner-Verlag, 1984

- [11] BETH, T. ; BOESNACH, I. ; HAIMERL, M. ; MOLDENHAUER, J. ; BÖS, K. ; WANK, V.: Analyse, Modellierung und Erkennung menschlicher Bewegungen. In: *1. SFB-Aussprachetag Human Centered Robotics (HCRS) 2002, 5.–6. Dezember 2002, Tagungsband*, Universität Karlsruhe (TH), 2002, S. 17–24.
- [12] BETH, T. ; BOESNACH, I. ; HAIMERL, M. ; MOLDENHAUER, J. ; BÖS, K. ; WANK, V.: Analyse, Modellierung und Erkennung menschlicher Bewegungen. In: *E.I.S.S.-Report Bd. 1*, Europäisches Institut für Systemsicherheit, Universität Karlsruhe (TH), 2003
- [13] BETH, T. ; BOESNACH, I. ; HAIMERL, M. ; MOLDENHAUER, J. ; BÖS, K. ; WANK, V.: Characteristics in Human Motion – From Acquisition to Analysis. In: *International Conference on Humanoid Robots HUMANOIDS 2003, 1.–3. Oktober 2003, Karlsruhe/München, Conference Documentation*, VDI/VDE-GMA, 2003, S. 56
- [14] BEYER, R. (Hrsg.) ; LETCHER, S. (Hrsg.): *Pure and Applied Physics*. Bd. 32: *Physical Ultrasonics*. Academic Press, New York, London, 1969
- [15] BISHOP, C.: *Neural Networks for Pattern Recognition*. Oxford University Press, Oxford, 1998
- [16] BISSACCO, A. ; CHIUSO, A. ; MA, Y. ; SOATTO, S.: Recognition of Human Gaits. In: *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition 2001* Bd. 2, 2001, S. 52–58
- [17] BOBICK, A. ; IVANOV, Y.: Action Recognition Using Probabilistic Parsing. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Juni 1998*, 1998, S. 196–202
- [18] BOBICK, A. ; IVANOV, Y.: Action Recognition Using Probabilistic Parsing. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22 (2000), August, Nr. 8, S. 852–872
- [19] BOESNACH, I. ; HAHN, M. ; HAIMERL, M. ; MOLDENHAUER, J.: Aufbereitung medizinischer Bilddaten – Überblick über aktuelle Arbeiten des IAKS im Bereich medizinische Bildverarbeitung. In: *E.I.S.S.-Report Bd. 3*, Europäisches Institut für Systemsicherheit, Universität Karlsruhe (TH), 2001
- [20] BOESNACH, I. ; HAHN, M. ; MOLDENHAUER, J. ; BETH, T. ; SPETZGER, U.: Analysis of Drill Sound in Spine Surgery. In: *Perspective in Image Guided Surgery, Proceedings of the Workshop on Medical Robotics, Navigation and Visualization MRNV 2004, 11.–12. März 2004, Remagen*, World Scientific, Singapur, 2004, S. 77–84
- [21] BOESNACH, I. ; MOLDENHAUER, J. ; BETH, T.: VAMOS – Visualisierungs-, Analyse- und Modellierungssystem. In: SPATH, D. (Hrsg.) ; HAASIS, K. (Hrsg.) ; KLUMPP, D. (Hrsg.): *Aktuelle Trends in der Softwareforschung, Tagungsband zum doIT Software-Forschungstag am 29. Oktober 2004, Stuttgart*, Fraunhofer IRB Verlag, 2005, S. 124–134

- [22] BOESNACH, I. ; MOLDENHAUER, J. ; BURGMER, C. ; BETH, T. ; WANK, V. ; BÖS, K.: Classification of Phases in Human Motions by Neural Networks and Hidden Markov Models. In: *Conference on Cybernetics and Intelligent Systems CIS 2004, 1.–3. Dezember 2004, Singapur*, 2004
- [23] BRAND, M. ; HERTZMANN, A.: Style Machines. In: *SIGGRAPH 2000, Computer Graphics Proceedings*, ACM Press, 2000, S. 183–192
- [24] BRAUSE, R.: *Neuronale Netze*. B. G. Teubner, Stuttgart, 1991 (Leitfäden und Monographien der Informatik)
- [25] BROOKS, M.: *VOICEBOX: Speech Processing Toolbox für MATLAB*. Department of Electrical & Electronic Engineering, Imperial College, London. <http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html>
- [26] CAWLEY, G. C.: *MATLAB Support Vector Machine Toolbox*. University of East Anglia, School of Information Systems, Norwich, Norfolk, U.K. NR4 7TJ. <http://theoval.sys.uea.ac.uk/~{gcc}/svm/toolbox/>. Version: 2000
- [27] CHANG, C.-C. ; LIN, C.-J.: *LIBSVM: a Library for Support Vector Machines*. Department of Computer Science and Information Engineering; National Taiwan University, Taipei 106, Taiwan. <http://www.csie.ntu.edu.tw/~cjlin/papers/libsvm.pdf>. Version: 2004
- [28] CHO, Z.-H. ; JONES, J. ; SINGH, M.: *Foundations of Medical Imaging*. John Wiley & Sons, New York, 1993
- [29] CHUI, C. K.: *Wavelets: A Mathematical Tool for Signal Analysis*. SIAM, Philadelphia, 1997 (SIAM Monographs on Mathematical Modeling and Computation)
- [30] CHUI, C. K. ; CHEN, G.: *Springer Series on Information Science*. Bd. 17: *Kalman Filtering*. Springer, Berlin, 1987
- [31] DAINTY, J. (Hrsg.): *Topics in Applied Physics*. Bd. 9: *Laser Speckle and Related Phenomena*. Springer, Berlin, Heidelberg, 1984
- [32] DANNENMEYER, K. ; KROSCHER, K.: *Digitale Signalverarbeitung*. B. G. Teubner, Stuttgart, 1996
- [33] DASGUPTA, A. ; NAKAMURA, Y.: Making Feasible Walking Motion of Humanoid Robots from Human Motion Capture Data. In: *Proceedings of the International Conference on Robotics and Automation 1999*, 1999, S. 1044–1049
- [34] DAVY, M. ; GODSILL, S.: Audio Information Retrieval: a Bibliographical Study / Signal Processing Group, Cambridge University Engineering Department, Cambridge, UK. 2002 (CUED/F-INFENG/TR.429). – Forschungsbericht

- [35] DEMUTH, H. ; BEALE, M.: *Neural Network Toolbox User's Guide*. Version 4. The MathWorks Inc., Natick, 2001
- [36] DONOHO, D.: De-Noising by Soft-Thresholding. In: *IEEE Transactions on Information Theory* 41 (1995), Nr. 3, S. 613–627
- [37] DÖSSEL, O.: *Bildgebende Verfahren in der Medizin*. Springer, Berlin, Heidelberg, 2000
- [38] DÄUBER, S. ; HAIMERL, M. ; HOPPE, H. ; KREMPIEN, R. ; MENDE, U. ; MOLDENHAUER, J. ; WÖRN, H.: Aufbereitung medizinischer Bilddaten. In: WÖRN, H. (Hrsg.) ; MÜHLING, J. (Hrsg.) ; VAHL, C. (Hrsg.) ; MEINZER, H.-P. (Hrsg.): *Rechner- und Sensorgestützte Chirurgie (Workshop-Proceedings)*, Gesellschaft für Informatik, Bonn, 2001, S. 189–202
- [39] DURWARD, B. R. ; BAER, G. D. ; ROWE, P. J.: *Functional Human Movement*. Butterworth Heinemann, Oxford, 2001
- [40] ELMAN, J.: Finding Structures in Time. In: *Cognitive Science* (1990), Nr. 14, S. 179–211
- [41] FAN, J. ; YAO, Q.: *Nonlinear Time Series: Nonparametric and Parametric Methods*. Springer, Berlin, Heidelberg, New York, 2002 (Springer, Series in Statistics)
- [42] *Field II Ultrasound Simulation Program*. <http://www.es.oersted.dtu.dk/staff/jai/field>
- [43] FINK, G.: *Mustererkennung mit Markov-Modellen*. B. G. Teubner, Wiesbaden, 2003 (Leitfäden der Informatik)
- [44] FISCHER, A. ; STEIN, T. ; BÖS, K. ; WANK, V. ; BOESNACH, I. ; MOLDENHAUER, J. ; BETH, T.: Guidelines for Motion Planning of Humanoid Robots: Analysis of Intra- and Interindividual Variations in Human Movements. In: *5th International Symposium Computer Science in Sport, IACSS 2004, 25.–28 Mai 2005, Hvar, Kroatien* Croatian Association on Computer Science in Sport & International Association on Computer Science, 2005, S. 41
- [45] FLOHR, M.: Operationssaal der Zukunft: Chirurgie im Cyberspace. In: *Chip* 5 (2004), S. 108–114
- [46] FRICK, F.: Der vermessene Athlet. In: *Spektrum der Wissenschaft* 8 (2004), S. 90–93
- [47] GEHRIG, D.: *Kalibrierung eines Datenhandschuhs anhand von Anthropomatikdaten*, Universität Karlsruhe, Diplomarbeit, 2005. – in Erscheinung
- [48] GLOSSOP, N.: Motion in Vertebral Bodies during Spine Surgery. In: *Computer Assisted Orthopaedic Surgery – CAOS '96, Bern, 1996*

- [49] GRÜTZNER, P. ; HEBECKER, A. ; WAELTI, H. ; VOCK, B. ; NOLTE, L.-P. ; WENTZENSEN, A.: Klinische Studie zur registrierungsfreien 3D-Navigation mit dem mobilen C-Bogen SIREMOBIL Iso-C (3D). In: *electromedica* 71 (2003), Nr. 1, S. 58–67
- [50] HAAR, A.: Zur Theorie der orthogonalen Funktionensysteme. In: *Mathematische Annalen* (1910), Nr. 69, S. 331–371
- [51] HAERTEN, R. ; MÜCK-WEYMANN, M.: *Doppler- und Farbdoppler-Sonographie*. 2. Aufl. Siemens AG, Erlangen, 1994
- [52] HAIMERL, M.: *Modalitätsspezifische Aufbereitung medizinischer Bilddaten*, Universität Karlsruhe, Diss., 2003
- [53] HAIMERL, M. ; MOLDENHAUER, J. ; BETH, T. ; MENDE, U.: Locally Adaptive Filtering of Ultrasound Images. In: H., Hamza M. (Hrsg.): *IASTED International Conference on Biomedical Engineering BioMed 2003, Juni 2003, Salzburg*, Biomedical Engineering, ACTA Press, Anaheim, USA, 2003, S. 198–203
- [54] HAIMERL, M. ; MOLDENHAUER, J. ; MENDE, U.: Zielgerichtete Aufbereitung und Visualisierung dreidimensionaler medizinischer Ultraschall Datensätze. In: HANDELS, H. (Hrsg.) ; HORSCH, A. (Hrsg.) ; LEHMANN, T. (Hrsg.) ; MEINZER, H.-P. (Hrsg.): *Tagungsband "Bildverarbeitung für die Medizin 2001"*, Lübeck 5./6. März 2001, Informatik aktuell, Springer Verlag, 2001, S. 117–121
- [55] HAIMERL, M. ; MOLDENHAUER, J. ; MENDE, U. ; BETH, T.: Comparative Analysis of Locally Adaptive Image Enhancement for 3D Ultrasound Images. In: U., Boenick (Hrsg.) ; A., Bolz (Hrsg.): *Biomedizinische Technik (Biomedical Engineering) "Beiträge zur 36. Jahrestagung der Deutschen Gesellschaft für Biomedizinische Technik (DGBMT) im VDE, 25.–27. September 2002, Universität Karlsruhe"* Bd. 47, Ergänzung 1, Teil 2, Fachverlag Schiele & Schön, Berlin, 2002, S. 629–632
- [56] HAIMERL, M. ; MOLDENHAUER, J. ; MENDE, U. ; BETH, T.: Merkmalsinduzierte Aufbereitung medizinischer Ultraschallbilddaten. In: M., Meiler (Hrsg.) ; D., Saupe (Hrsg.) ; F., Kruggel (Hrsg.) ; H., Handels (Hrsg.) ; T., Lehmann (Hrsg.): *Tagungsband "Bildverarbeitung für die Medizin 2002"*, Leipzig 11./12. März 2002, Informatik aktuell, Springer Verlag, 2002, S. 99–102
- [57] HAMILTON, J.: *Time Series Analysis*. Princeton University Press, New Jersey, 1994
- [58] HAO, X. ; GAO, S. ; GAO, X.: A Novel Multiscale Nonlinear Thresholding Method for Ultrasonic Speckle Suppressing. In: *IEEE Transactions on Medical Imaging* 18 (1999), Nr. 9, S. 787–794
- [59] HARMS, J. ; TABASSO, G.: *Instrumented Spinal Surgery: Principles and Technique*. Thieme, 1999

- [60] HAUSCHILD, H. W.: Head Trajectories of Restrained Child Dummy in Sled Tests over 56 Kph Delta-v. In: *Annual Proceedings of the Association for the Advancement of Automotive Medicine* Bd. 44, 2000, S. 299–308
- [61] HAY, J. G. ; YU, B.: Critical Characteristics of Technique in Throwing the Discus. In: *Journal of Sports Science* 13 (1995), Nr. 2, S. 125–140
- [62] HERMANN, H.: *LNCS*. Bd. 2428: *Interactive Markov Chains*. Springer, Berlin, 1998
- [63] HERRING, J. ; DAWANT, B. ; MAURER, C. ; MURATORE, D. ; GALLOWAY, R. ; FITZPATRICK, J.: Surface-Based Registration of CT Images to Physical Space for Image-Guided Surgery of the Spine: A Sensitivity Study. In: *Medical Imaging* 17 (1998), Nr. 5, S. 743–752
- [64] HERTZ, J. ; KROGH, A. ; PALMER, R. G.: *Santa Fe Institute studies in the science complexity, Lecture Notes*. Bd. 1: *Introduction to the Theory of Neural Computation*. Addison-Wesley, Redwood City, 1991
- [65] HÄNSLER, E.: *Statistische Signale: Grundlagen und Anwendungen*. 3. Aufl. Springer, Berlin, Heidelberg, New York, 2002
- [66] HOFFMANN, R.: *Signalanalyse und -erkennung*. Springer, Berlin, Heidelberg, New York, 1998
- [67] HSU, C.-W. ; CHANG, C.-C. ; LIN, C.-J.: *A Practical Guide to Support Vector Classification*. Department of Computer Science and Information Engineering; National Taiwan University, Taipei 106, Taiwan. <http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf>. Version: 2004
- [68] HUBBARD, B. B.: *The World According to Wavelets*. A K Peters, Wellesley, 1996
- [69] HUCKLE, T. ; SCHNEIDER, S.: *Numerik für Informatiker*. Springer, Berlin, Heidelberg, New York, 2002
- [70] IJSPEERT, A. ; NAKANISHI, J. ; SCHAAL, S.: Movement Imitation with Nonlinear Dynamical Systems in Humanoid Robots. In: *Proceedings of the IEEE International Conference on Robotics and Automation 2002 (ICRA 2002)*, 2002, S. 1398–1403
- [71] INAMURA, T. ; TANIE, H. ; NAKMURA, Y.: From Stochastic Motion Generation and Recognition to Geometric Symbol Development and Manipulation. In: *Proceedings of the IEEE Humanoids 2003, Karlsruhe*, 2003, S. 9
- [72] JAIN, A. K.: *Fundamentals of Digital Image Processing*. Prentice Hall, Englewood Cliffs, 1989 (Information and System Sciences Series)
- [73] JENKINS, O. C.: *Data-Driven Derivation of Skills For Autonomous Humanoid Agents*, University of Southern California, Diss., 2003

- [74] JENSEN, A. ; COUR-HARBO, A. Ia: *Ripples in Mathematics – The Discrete Wavelet Transform*. Springer, Berlin, 2000
- [75] JENSEN, J. ; NIKOLOV, S.: Fast Simulation of Ultrasound Images. In: *IEEE International Ultrasonics Symposium, Puerto Rico*, 2000, S. 1721–1724
- [76] JOSKOWICZ, L. ; KNAAN, D. ; LIVYATAN, H. ; YANIV, Z. ; KHOURY, A. ; MOSHEIFF, R. ; LIEBERGALL, M.: Anatomical Image-Based Rigid Registration Between Fluoroscopic X-Ray and CT: Methods Comparison and Experimental Results. In: LEMKE, H. (Hrsg.) ; VANNIER, M. (Hrsg.) ; INAMURA, K. (Hrsg.) u. a.: *Computer Assisted Radiology and Surgery - CARS 2003*, Elsevier, 2003 (International Congress Series 1256), S. 419–425
- [77] KAMMEYER, K. ; KROSCHEL, K.: *Digitale Signalverarbeitung*. B. G. Teubner, Stuttgart, 2002
- [78] KARAMAN, M. ; KUTAY, M. A. ; BOZDAGI, G.: An Adaptive Speckle Suppression Filter for Medical Ultrasonic Imaging. In: *IEEE Transactions on Medical Imaging* 14 (1995), Juni, Nr. 2, S. 283–292
- [79] KAUS, M. ; STEINMEIER, R. ; SPORER, T. ; GANSLANDT, O. ; FAHLBUSCH, R.: Technical Accuracy of a Neuronavigation System Measured with a High-Precision Mechanical Micro-Manipulator. In: *Neurosurgery* 41 (1997), S. 1431–1437
- [80] KIM, C. ; PARK, H.: Preprocessing and Efficient Volume Rendering of 3-D Ultrasound Images. In: *IEICE Transactions on Information and Systems* 83 (2000), Nr. 2, S. 259–264
- [81] KINSLER, L. ; FREY, A.: *Fundamentals of Acoustics*. 2. Aufl. John Wiley & Sons, New York, 1962
- [82] KOVAR, L. ; GLEICHER, M.: Flexible Automatic Motion Blending with Registration Curves. In: *Proceedings of the 2003 ACM Eurographics/SIGGRAPH Symposium on Computer Animation, San Diego*, 2003, S. 214–224
- [83] KOVAR, L. ; GLEICHER, M. ; PIGHIN, F.: Motion Graphs. In: *Proceedings of the 29<sup>th</sup> Annual Conference on Computer Graphics and Interactive Techniques, San Antonio, 2002* Bd. 21, 2002, S. 472–482
- [84] KUCH, J. ; HUANG, T.: Vision Based Hand Modelling and Tracking for Virtual Teleconferencing and Telecooperation. In: *Proceedings of 5th International Conference on Computer Vision, Cambridge, MA*, 1995, S. 666–671
- [85] LEE, J.: *A Hierarchical Approach to Motion Analysis and Synthesis for Articulated Figures*, Korea Advanced Institute of Science and Technology, Diss., 2000

- [86] LEE, J. ; CHAI, J. ; REITSMA, P. ; HODGINS, J. ; POLLAND, N.: Interactive Control of Avatars Animated with Human Motion Data. In: *Proceedings of the ACM SIGGRAPH 2002*, 2002, S. 491–500
- [87] LEE, J. ; KUNII, T.: Model-Based Analysis of Hand Posture. In: *IEEE Computer Graphics and Applications* 15 (1995), September, Nr. 5, S. 77–86
- [88] LEHMANN, T. ; OBERSCHELP, W. ; PELIKAN, E. ; REPGES, R.: *Bildverarbeitung für die Medizin*. Springer, Berlin, Heidelberg, 1997
- [89] LI, Y. ; WANG, T. ; SHUM, H.-Y.: Motion Texture: A Two-Level Statistical Model for Character Motion Synthesis. In: *Proceedings of the ACM SIGGRAPH 2002*, 2002, S. 465–472
- [90] LIU, C. ; POPOVIĆ, Z.: Synthesis of Complex Dynamic Character Motion from Simple Animations. In: *Proceedings of the ACM SIGGRAPH 2002*, 2002, S. 408–416
- [91] LOUIS, A. K. ; MAASS, P. ; RIEDER, A.: *Wavelets*. 2. Aufl. B. G. Teubner, Stuttgart, 1998 (Studienbücher Mathematik)
- [92] LOUPAS, T. ; MCDICKEN, N. ; ALLAN, P.: An Adaptive Weighed Median Filter for Speckle Suppression in Medical Ultrasonic Images. In: *IEEE Transactions on Circuits and Systems* 36 (1989), Januar, Nr. 1, S. 129–135
- [93] MACDONALD, I. ; ZUCCHINI, W.: *Hidden Markov and Other Models for Discrete-valued Time Series*. Chapman & Hall, London, 1997 (Monographs on Statistics and Applied Probability 70)
- [94] MALLAT, S.: *A Wavelet Tour of Signal Processing*. Academic Press, San Diego, 1998
- [95] *Matlab*. Fa. Mathworks. <http://www.mathworks.com/products/matlab/>
- [96] *Matlab Neural Network Toolbox*. Fa. Mathworks. <http://www.mathworks.com/products/neuralnet/>
- [97] *Mazor – Surgical Technologies*. <http://www.mazorst.com/home.htm>
- [98] MEDVED, V.: *Measurement of Human Locomotion*. CRC Press, Boca Raton, 2001
- [99] MEHLDAU, H.: *Waveletbasierte Rauschreduktion in medizinischen Ultraschalldaten*, Universität Karlsruhe, Diplomarbeit, 1998
- [100] MEHLDAU, H. ; ZERFOWSKI, D. ; KLAPPENECKER, A.: Waveletbasierte Rauschreduktion in medizinischen Ultraschallbildern. In: *Proceedings zum Workshop: Bildverarbeitung für die Medizin 1998, Aachen*, Springer, Berlin, 1998, S. 229–233

- [101] MENDE, U. ; PIENTA, S. ; HAIMERL, M. ; MOLDENHAUER, J. ; WÖRN, H. ; ZUNA, I.: 3D-Ultraschall bei Knochenmalignomen. In: *Ultraschall 2003, 27. Dreiländertreffen der ÖGUM, DEGUM, SGUM, 10.–13. September 2003, Bregenz am Bodensee*, Ultraschall in der Medizin, 24. Jahrgang, S1 Sonderheft, Thieme Verlag, 2003, S. 45
- [102] MENDE, U. ; WELZEL, T. ; HASSFELD, S. ; FLECHTENMACHER, C. ; KREMPIEN, R. ; MOLDENHAUER, J. ; WÖRN, H.: Lymph Node Ultrasound. In: *EUROSON 2004 – XVI European Congress on Ultrasound in Medicine and Biology, 5.–8. Juni 2004, Zagreb* Bd. 126 (Suppl. 2), Lijec Vjesn, 2004, S. 50
- [103] MENDE, U. ; WELZEL, T. ; KREMPIEN, R. ; HASSFELD, S. ; FLECHTENMACHER, C. ; MOLDENHAUER, J. ; WÖRN, H. ; ZUNA, I.: 3D Ultrasound in Oncology. In: *EUROSON 2004 – XVI European Congress on Ultrasound in Medicine and Biology, 5.–8. Juni 2004, Zagreb* Bd. 126 (Suppl. 2), Lijec Vjesn, 2004, S. 51
- [104] MÜLLER, K.-R. ; MIKA, S. ; RÄTSCH, G. ; TSUDA, K. ; SCHÖLKOPF, B.: An Introduction to Kernel-Based Learning Algorithms. In: *IEEE Transactions on Neural Networks* 12 (2001), März, Nr. 2, S. 181–202
- [105] MOLDENHAUER, J.: *Dreidimensionale Visualisierung medizinischer Daten*, Universität Karlsruhe (TH), Studienarbeit, 1998
- [106] MOLDENHAUER, J.: *Visualisierung von Ultraschall-Volumendatensätzen mit konventionellen und waveletbasierten Verfahren*, Universität Karlsruhe (TH), Diplomarbeit, 1999
- [107] MOLDENHAUER, J. ; BETH, T. ; MENDE, U.: Hidden Markov Models for the Analysis and Classification of Ultrasound Data. In: LEMKE, H. (Hrsg.) ; VANNIER, M. (Hrsg.) ; INAMURA, K. (Hrsg.) ; FARMAN, A. (Hrsg.) ; DOI, K. (Hrsg.) ; REIBER, J. (Hrsg.): *Computer Assisted Radiology and Surgery CARS 2004, Proceedings of the 18th International Congress and Exhibition, 23.–26. Juni 2004, Chicago*, International Congress Series 1268, Elsevier, Amsterdam, 2004, S. 231–236
- [108] MOLDENHAUER, J. ; BOESNACH, I. ; BETH, T. ; WANK, V. ; BÖS, K.: Analysis of Human Motion for Humanoid Robots. In: *Proceedings of the IEEE International Conference on Robotics and Automation ICRA 2005, 18.–22. April 2005* IEEE, CINME, Barcelona, 2005, S. 312–317
- [109] MOLDENHAUER, J. ; HAIMERL, M. ; MENDE, U.: Wavelet-basierte Visualisierung von Ultraschall-Volumendaten. In: HORSCH, A. (Hrsg.) ; LEHMANN, T. (Hrsg.): *Tageband "Bildverarbeitung für die Medizin 2000"*, München, 13./14. März 2000, Informatik aktuell, Springer Verlag, 2000, S. 143–147
- [110] MOLDENHAUER, J. ; HAIMERL, M. ; WALZ, M. ; WEISSER, G.: Datenschutz und Systemsicherheit für die medizinische Informationsverarbeitung. In: WÖRN, H. (Hrsg.) ; MÜHLING, J. (Hrsg.) ; VAHL, C. (Hrsg.) ; MEINZER, H.-P. (Hrsg.): *Rechner-*

- und Sensorgestützte Chirurgie (Workshop-Proceedings)*, Gesellschaft für Informatik, Bonn, 2001, S. 156–164
- [111] MORNEBURG, H. (Hrsg.): *Bildgebende Systeme für die medizinische Diagnostik*. 3. Aufl. Publicis MCD Verlag, Erlangen, 1995
- [112] MURPHY, K.: The Bayes Net Toolbox for Matlab. In: *Computing Science and Statistics* 33 (2001), S. 331–350
- [113] MURPHY, K.: *Dynamic Bayesian Networks: Representation, Inference and Learning*, University of California, Berkley, Diss., 2002
- [114] NIEMINEN, A. ; HEINONEN, P. ; NEUVO, Y.: A New Class of Detail-Preserving Filters for Image Processing. In: *IEEE Transaction on Pattern Analysis and Machine Intelligence* 9 (1987), Nr. 1, S. 74–90
- [115] NIGG, B. M. (Hrsg.) ; HERZOG, W. (Hrsg.): *Biomechanics of the Musculo-skeletal System*. J. Wiley, New York, 1998
- [116] NORDIN, M. ; FRANKEL, V. H.: *Basic Biomechanics of the Musculoskeletal System*. 3. Aufl. Lippincott Williams & Wilkins, Philadelphia, 2001
- [117] O'BRIEN, J. F. ; BODENHEIMER, R. E. ; BROSTOW, G. J. ; HODGINS, J. K.: Automatic Joint Parameter Estimation from Magnetic Motion Capture Data. In: *Proceedings of Graphics Interface 2000, Montreal Canada*, 2000, S. 53–60
- [118] ORALKAN, Ö. ; ERGUN, A. ; CHENG, C.-H. ; JOHNSON, J. ; KARAMAN, M. ; LEE, T. ; KHURI-YAKUB, B.: Volumetric Ultrasound Imaging Using 2-D CMUT Arrays. In: *Transactions on Ultrasonics, Ferroelectronics, and Frequency Control* 50 (2003), November, Nr. 11, S. 1581–1594
- [119] PENNEY, G.: *Registration of Tomographic Images to X-ray Projections for Use in Image Guided Interventions*, King's College London, Diss., 1999
- [120] POLASEK, W.: *EDA – Explorative Datenanalyse*. Springer, Berlin, Heidelberg, New York, 1994
- [121] POTT, P. ; KÖPFLE, A. ; WAGNER, A. ; BADREDDIN, E. ; MÄNNER, R. ; WEISER, P. ; SCHARF, H.-P. ; SCHWARZ, M.: State of the art of surgical robots. In: *Perspective in Image Guided Surgery, Proceedings of the Workshop on Medical Robotics, Navigation and Visualization MRNV 2004, 11.–12. März 2004, Remagen*, World Scientific, Singapur, 2004
- [122] POTT, P. ; SCHWARZ, M. ; KÖPFLE, A. ; SCHILL, M. ; BADREDDIN, E. ; MÄNNER, R. ; WEISER, H.-P. ; SCHARF, H.-P.: Handgehaltener Operationsroboter – Grundlagen, Spezifikationen und Lösungsentwurf. In: *CURAC 2002, Computer und Roboter Assistierte Chirurgie, Leipzig*

- [123] PRESS, W. H. ; FLANNERY, B. P. ; TEUKOLSKY, S. A. ; VETTERLING, W. T.: *Numerical Recipes in C*. Cambridge University Press, 1991
- [124] PULLEN, K.: *Motion Capture Assisted Animation: Texturing and Synthesis*, Stanford University, Diss., 2002
- [125] PULLEN, K. ; BREGLER, C.: Motion Capture Assisted Animation: Texturing and Synthesis. In: *Proceedings of the ACM SIGGRAPH 2002*, 2002, S. 501–507
- [126] PUTZ, R. (Hrsg.) ; PABST, R. (Hrsg.): *Sobotta*. Bd. 2. 20. Aufl. Urban & Schwarzenberg, München, 1993
- [127] PUTZ, R. (Hrsg.) ; PABST, R. (Hrsg.): *Sobotta*. Bd. 1. 20. Aufl. Urban & Schwarzenberg, München, 1993
- [128] R. ELLIOT, J. M.: *Hidden Markov Models: Estimation and Control*. Springer, Berlin, 1995 (Applications of mathematics, 29)
- [129] RABINER, L.: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. In: *Proceedings of the IEEE 77* (1989), Feb., Nr. 2, S. 257–286
- [130] RABINER, L. ; JUANG, B.-H.: *Fundamentals of Speech Recognition*. Prentice Hall, 2003
- [131] RITTER, D. ; MITSCHKE, M. ; GRAUMANN, R.: Intraoperative Soft Tissue 3D Reconstruction with a Mobile C-Arm. In: LEMKE, H. (Hrsg.) ; VANNIER, M. (Hrsg.) ; INAMURA, K. (Hrsg.) u. a.: *Computer Assisted Radiology and Surgery – CARS 2003*, Elsevier, 2003 (International Congress Series 1256), S. 200–206
- [132] ROGALLA, O.: *Abbildung von Benutzerdemonstrationen auf variable Roboterkonfigurationen*, Universität Karlsruhe, Diss., 2002
- [133] *RoMed: Roboter und Manipulatoren für Medizinische Einsatzfelder*. <http://www.ar.ipk.fhg.de/ProjekteundProdukte/WISA-ROMED/ROMED-index-d.html>. – Kooperationsprojekt der Fraunhofer Institute IPK Berlin, IPA Stuttgart, IBMT St. Ingbert und IGD Darmstadt
- [134] ROSENBLATT, M.: *Gaussian and Non-Gaussian Linear Time Series and Random Fields*. Springer, New York, 2000 (Springer Series in Statistics)
- [135] RUSSEL, S. ; NORVIG, P.: *Künstliche Intelligenz*. Pearson Education Deutschland, München, 2004
- [136] SAISAN, P. ; BISSACCO, A.: Image-based Modelling of Human Gaits with Higher-Order Statistics. In: *Proceedings of the International Workshop on Dynamic Scene Analysis, Kopenhagen, 2002*, 2002

- [137] SAKAS, G. ; SCHREYER, L. ; GRIMM, M.: Preprocessing and Volume Rendering of 3D Ultrasonic Data. In: *IEEE Transactions on Computer Graphics and Applications* 15 (1995), Nr. 4, S. 47–54
- [138] SAKAS, G. ; WALTER, W.: Extracting Surfaces from Fuzzy 3D-Ultrasound Data. In: *SIGGRAPH 1995*, 1995, S. 465–474
- [139] SAWANT, A. ; ZEMAN, H. ; MURATONE, D. ; SAMANT, S. ; DIBIANCA, F.: Adaptive Median Filter Algorithm to Remove Impulse Noise in X-Ray and CT Images and Speckle Ultrasound Images. In: *Proceedings of the SPIE* Bd. 3661, 1999, S. 1263–1274
- [140] SCHAAL, S. ; IJSPEERT, A. ; BILLARD, A.: Computational Approaches to Motor Learning by Imitation. In: *Philosophical Transactions of the Royal Society of London* Bd. 358, 2003, S. 537–547
- [141] SCHENKEL, M.: *Series in Microelectronics*. Bd. 45: *Handwriting Recognition using Neural Networks and Hidden Markov Models*. Hartung-Verlag Konstanz, 1995
- [142] SCHÖLKOPF, B. ; MÜLLER, K.-R. ; SMOLA, A.: Lernen mit Kernen. In: *Informatik Forschung und Entwicklung* (1999), Nr. 14, S. 154–163
- [143] SCHÖLKOPF, B. ; SMOLA, A. ; MÜLLER, K.-R.: Nonlinear Component Analysis as a Kernel Eigenvalue Problem. In: *Neural Computation* 10 (1998), S. 1299–1319
- [144] SELLER, K. ; WILD, A. ; URSELMANN, L. ; KRAUSPE, R.: Prospektive Schraubenfehlagenanalyse nach konventioneller und navigierter Pedikelschraubenimplantation. In: *Biomedizinische Technik* 50 (2005), Nr. 9, S. 287–292
- [145] *DFG Sonderforschungsbereich 588 "Humanoide Roboter – Lernende und kooperierende multimodale Roboter"*. <http://www.sfb588.uni-karlsruhe.de>
- [146] SHUMWAY, R. ; STOFFER, D.: *Time Series Analysis and Its Application*. Springer, New York, 2000 (Springer Texts in Statistics)
- [147] SIPOS, E. ; TEBO, S. ; ZINREICH, S. ; LONG, D. ; BREM, H.: In Vivo Accuracy Testing and Clinical Experience with the ISG Viewing Wan. In: *Neurosurgery* 39 (1996), S. 194–204
- [148] SMYTH, M. (Hrsg.) ; WING, A. (Hrsg.): *The Psychology of Human Movement*. Academic Press, 1984
- [149] STARNER, T. ; PENTLAND, A.: Visual Recognition of American Sign Language Using Hidden Markov Models. In: *International Workshop on Automatic Face and Gesture Recognition*, 1995, S. 189–194
- [150] STARNER, T. ; PENTLAND, A.: Real-Time American Sign Language Recognition Using Desk and Wearable Computer Based Video. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20 (1998), Dezember, Nr. 12, S. 1371–1375

- [151] STEIN, T. ; FISCHER, A. ; BÖS, K. ; WANK, V. ; BOESNACH, I. ; MOLDENHAUER, J. ; BETH, T.: Erfassung und Analyse menschlicher Bewegungen zur Bewegungsplanung für humanoide Roboter. In: HUBER, G. (Hrsg.) ; MORLOCK, E. Schneider M. (Hrsg.): *Jahrestagung der Deutschen Gesellschaft für Biomechanik – Biomechanica V, 17.–19. März 2005, Hamburg* Deutsche Gesellschaft für Biomechanik DGfB, 2005, S. 112
- [152] STEINMANN, J. ; HERKOWITZ, H. ; EL-KOMMOS, H. ; WESOŁOWSKI, D.: Spinal Pedicle Fixation: Confirmation of an Image-Based Technique for Screw Placement. In: *Spine* 18 (1993), S. 1856–1861
- [153] STOLCKE, A. ; SEGAL, J.: Precise n-Gram Probabilities from Stochastic Context-free Grammars. In: *Proceedings of the 32nd Annual Meeting of the Association of Computational Linguistics, Juni 1994, Las Cruces, New Mexico*, 1994, S. 74–79
- [154] TREVES, F.: *Introduction to Pseudodifferential and Fourier Integral Operators*. Bd. 1. Plenum Press, New York, 1980
- [155] UDE, A. ; MAN., C. ; RILEY, M. ; ATKESON, C.: Automatic Generation of Kinematic Models for the Conversion of Human Motion Capture Data into Humanoid Robot Motion. In: *Proceedings of the IEEE/RAS International Conference on Humanoid Robots, Cambridge, Massachusetts, September 2000*, 2000
- [156] *Ultrasim – A Toolbox for Ultrasound Field Simulation*. [http://www.ifi.uio.no/\\$\sim\\$ultrasim](http://www.ifi.uio.no/$\sim$ultrasim)
- [157] *Homepage von VAMOS*. Institut für Algorithmen und Kognitive Systeme, Universität Karlsruhe (TH). <http://iaks-www.ira.uka.de/home/vamos/index.html>
- [158] VAPNIK, V.: *The Nature of Statistical Learning Theory*. Springer, Berlin, 1995
- [159] VAPNIK, V.: *Statistical Learning Theory*. Wiley, New York, Weinheim, 1998
- [160] VASEGHI, S. V.: *Advanced Signal Processing*. John Wiley & Sons, Chichester, 1996
- [161] VASILESCU, M.: Human Motion Signatures: Analysis, Synthesis, Recognition. In: *Proceedings of the International Conference on Pattern Recognition (ICPR 2002), Quebec, August 2002* Bd. 3, 2002, S. 456–460
- [162] VOGLER, C. ; METAXAS, D.: Parallel Hidden Markov Models for American Sign Language. In: *Proceedings of the Seventh International Conference on Computer Vision, Los Alamitos, USA* Bd. 1, 1999, S. 116–122
- [163] WALDMANN, K.-H. ; STOCKER, U.: *Stochastische Modelle*. Springer, Berlin, Heidelberg, New York, 2003

- [164] WANG, J. ; BODENHEIMER, B.: An Evaluation of a Cost Metric for Selecting Transactions between Motion Segments. In: *Proceedings of the 2003 ACM Eurographics/SIGGRAPH Symposium on Computer Animation, San Diego*, 2003, S. 232–238
- [165] WANG, L. ; HU, W. ; TAN, T.: Recent Developments in Human Motion Analysis. In: *Pattern Recognition* 36 (2003), S. 585–601
- [166] WANK, V. ; FISCHER, A. ; BÖS, K. ; BOESNACH, I. ; MOLDENHAUER, J. ; BETH, T.: Similarities and Varieties in Human Motion Trajectories of Predefined Grasping and Disposing Movements. In: *International Conference on Humanoid Robots HUMANOIDS 2004, 10.–12. November 2004, Santa Monica*, 2004, S. 311–321
- [167] WEESE, J. ; GÖCKE, R. ; PENNEY, G. ; DESMEDT, P. ; BUZUG, T. ; SCHUMANN, H.: Fast Voxel-Based 2D/3D Registration Algorithm Using a Volume Rendering Method Based on the Shear Warp Factorization. In: *SPIE International Symposium on Medical Imaging, San Diego, USA*, 1999
- [168] WEESE, J. ; PENNEY, G. ; DESMEDT, P. ; BUZUG, T. ; HILL, D. ; HAWKES, D.: Voxel-based 2D/3D Registration For Image Guided Surgery. In: *IEEE Trans. Inf. Tech. Biomed.* 1 (1997), Nr. 4, S. 284–293
- [169] WEISS, M. ; KULIKOWSKI, C.: *Computer Systems that Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning and Expert Systems*. Morgan Kaufman, 1991
- [170] WINTER, D.: *Biomechanics and Motor Control of Human Movement*. 2. Aufl. John Wiley & Sons, 1990
- [171] WINTER, S. ; BRENDE, B. ; RICK, A. ; STOCKHEIM, M. ; SCHMIEDER, K. ; ERMERT, H.: Registration of Bone Surfaces, Extracted from CT-Datasets, with 3D Ultrasound. In: *Biomedizinische Technik* Bd. 47, 2002, S. 57–60
- [172] XIAO, G. ; BRADY, M. ; NOBLE, J. ; ZHANG, Y.: Segmentation of Ultrasound B-Mode Images With Intensity Inhomogeneity Correction. In: *IEEE Transactions on Medical Imaging* 21 (2002), Nr. 1, S. 48–57
- [173] YAMANE, K.: *Simulating and Generating Motions of Human Figures*. Springer, Berlin, 2004
- [174] YIGIT, S. ; BURGHART, C. ; WÖRN, H.: Specific Combined Control Mechanisms for Human Robot Co-operation. In: *International Conference on Intelligent Systems and Control (ISC 2003), Salzburg IASTED*, 2003
- [175] YIN, P. ; ESSA, I. ; REHG, J. M.: Asymmetrically Boosted HMM for Speech Reading. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2004)*, IEEE Computer Society, Juni 2004, S. II755–761

- 
- [176] YOUNG, M.: *Optik, Laser, Wellenleiter*. Springer, Berlin, Heidelberg, 1997
- [177] YOUNG, S. ; RUSSEL, N. ; THORNTON, J.: Token Passing: A Simple Conceptual Model for Connected Speech Recognition Systems / Cambridge University, Engineering Department. 1989 (Technical Report CUED/F-INFENG/TR38). – Forschungsbericht. Technical Report
- [178] ZÖLLNER, R. ; ASFOUR, T. ; DILLMANN, R.: Programming by Demonstration: Dual-Arm Manipulation Tasks for Humanoids. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2004), Sendai, Japan, September 2004*, 2004
- [179] ZONNEVELD, F.: Intra-operative CT: Implementation of the Tomoscan M. In: *Medicamundi* 42 (1998), Nr. 1, S. 6–11



# Eigene Arbeiten

- [1] BOESNACH, I. ; MOLDENHAUER, J. ; BETH, T.: VAMOS – Visualisierungs-, Analyse- und Modellierungssystem. In: SPATH, D. (Hrsg.) ; HAASIS, K. (Hrsg.) ; KLUMPP, D. (Hrsg.): *Aktuelle Trends in der Softwareforschung, Tagungsband zum doIT Software-Forschungstag am 29. Oktober 2004, Stuttgart*, Fraunhofer IRB Verlag, 2005, S. 124–134
- [2] MOLDENHAUER, J. ; BOESNACH, I. ; BETH, T. ; WANK, V. ; BÖS, K.: Analysis of Human Motion for Humanoid Robots. In: *Proceedings of the IEEE International Conference on Robotics and Automation ICRA 2005, 18.–22. April 2005* IEEE, CINME, Barcelona, 2005, S. 312–317
- [3] STEIN, T. ; FISCHER, A. ; BÖS, K. ; WANK, V. ; BOESNACH, I. ; MOLDENHAUER, J. ; BETH, T.: Erfassung und Analyse menschlicher Bewegungen zur Bewegungsplanung für humanoide Roboter. In: HUBER, G. (Hrsg.) ; MORLOCK, E. Schneider M. (Hrsg.): *Jahrestagung der Deutschen Gesellschaft für Biomechanik – Biomechanica V, 17.–19. März 2005, Hamburg* Deutsche Gesellschaft für Biomechanik DGfB, 2005, S. 112
- [4] FISCHER, A. ; STEIN, T. ; BÖS, K. ; WANK, V. ; BOESNACH, I. ; MOLDENHAUER, J. ; BETH, T.: Guidelines for Motion Planning of Humanoid Robots: Analysis of Intra- and Interindividual Variations in Human Movements. In: *5th International Symposium Computer Science in Sport, IACSS 2004, 25.–28 Mai 2005, Hvar, Kroatien* Croatian Association on Computer Science in Sport & International Association on Computer Science, 2005, S. 41
- [5] BOESNACH, I. ; MOLDENHAUER, J. ; BURGNER, C. ; BETH, T. ; WANK, V. ; BÖS, K.: Classification of Phases in Human Motions by Neural Networks and Hidden Markov Models. In: *Conference on Cybernetics and Intelligent Systems CIS 2004, 1.–3. Dezember 2004, Singapur*, 2004
- [6] WANK, V. ; FISCHER, A. ; BÖS, K. ; BOESNACH, I. ; MOLDENHAUER, J. ; BETH, T.: Similarities and Varieties in Human Motion Trajectories of Predefined Grasping and Disposing Movements. In: *International Conference on Humanoid Robots HUMANOIDS 2004, 10.–12. November 2004, Santa Monica*, 2004, S. 311–321
- [7] MOLDENHAUER, J. ; BETH, T. ; MENDE, U.: Hidden Markov Models for the Analysis and Classification of Ultrasound Data. In: LEMKE, H. (Hrsg.) ; VANNIER, M. (Hrsg.) ; INAMURA, K. (Hrsg.) ; FARMAN, A. (Hrsg.) ; DOI, K. (Hrsg.) ; REIBER, J. (Hrsg.): *Computer Assisted Radiology and Surgery CARS 2004, Proceedings of the 18th International*

*Congress and Exhibition, 23.–26. Juni 2004, Chicago*, International Congress Series 1268, Elsevier, Amsterdam, 2004, S. 231–236

[8] MENDE, U. ; WELZEL, T. ; HASSFELD, S. ; FLECHTENMACHER, C. ; KREMPIEN, R. ; MOLDENHAUER, J. ; WÖRN, H.: Lymph Node Ultrasound. In: *EUROSON 2004 – XVI European Congress on Ultrasound in Medicine and Biology, 5.–8. Juni 2004, Zagreb* Bd. 126 (Suppl. 2), Lijec Vjesn, 2004, S. 50

[9] MENDE, U. ; WELZEL, T. ; KREMPIEN, R. ; HASSFELD, S. ; FLECHTENMACHER, C. ; MOLDENHAUER, J. ; WÖRN, H. ; ZUNA, I.: 3D Ultrasound in Oncology. In: *EUROSON 2004 – XVI European Congress on Ultrasound in Medicine and Biology, 5.–8. Juni 2004, Zagreb* Bd. 126 (Suppl. 2), Lijec Vjesn, 2004, S. 51

[10] BOESNACH, I. ; HAHN, M. ; MOLDENHAUER, J. ; BETH, T. ; SPETZGER, U.: Analysis of Drill Sound in Spine Surgery. In: *Perspective in Image Guided Surgery, Proceedings of the Workshop on Medical Robotics, Navigation and Visualization MRNV 2004, 11.–12. März 2004, Remagen*, World Scientific, Singapur, 2004, S. 77–84

[11] HAIMERL, M. ; MOLDENHAUER, J. ; BETH, T. ; MENDE, U.: Locally Adaptive Filtering of Ultrasound Images. In: H., Hamza M. (Hrsg.): *IASTED International Conference on Biomedical Engineering BioMed 2003, Juni 2003, Salzburg*, Biomedical Engineering, ACTA Press, Anaheim, USA, 2003, S. 198–203

[12] MENDE, U. ; PIANTA, S. ; HAIMERL, M. ; MOLDENHAUER, J. ; WÖRN, H. ; ZUNA, I.: 3D-Ultraschall bei Knochenmalignomen. In: *Ultraschall 2003, 27. Dreiländertreffen der ÖGUM, DEGUM, SGUM, 10.–13. September 2003, Bregenz am Bodensee*, Ultraschall in der Medizin, 24. Jahrgang, S1 Sonderheft, Thieme Verlag, 2003, S. 45

[13] BETH, T. ; BOESNACH, I. ; HAIMERL, M. ; MOLDENHAUER, J. ; BÖS, K. ; WANK, V.: Analyse, Modellierung und Erkennung menschlicher Bewegungen. In: *E.I.S.S.-Report* Bd. 1, Europäisches Institut für Systemsicherheit, Universität Karlsruhe (TH), 2003

[14] BETH, T. ; BOESNACH, I. ; HAIMERL, M. ; MOLDENHAUER, J. ; BÖS, K. ; WANK, V.: Characteristics in Human Motion – From Acquisition to Analysis. In: *International Conference on Humanoid Robots HUMANOIDS 2003, 1.–3. Oktober 2003, Karlsruhe/München, Conference Documentation*, VDI/VDE-GMA, 2003, S. 56

[15] HAIMERL, M. ; MOLDENHAUER, J. ; MENDE, U. ; BETH, T.: Merkmalsinduzierte Aufbereitung medizinischer Ultraschallbilddaten. In: M., Meiler (Hrsg.) ; D., Saupe (Hrsg.) ; F., Kruggel (Hrsg.) ; H., Handels (Hrsg.) ; T., Lehmann (Hrsg.): *Tagungsband "Bildverarbeitung für die Medizin 2002", Leipzig 11./12. März 2002*, Informatik aktuell, Springer Verlag, 2002, S. 99–102

[16] HAIMERL, M. ; MOLDENHAUER, J. ; MENDE, U. ; BETH, T.: Comparative Analysis of Locally Adaptive Image Enhancement for 3D Ultrasound Images. In: U., Boenick (Hrsg.) ; A., Bolz (Hrsg.): *Biomedizinische Technik (Biomedical Engineering) "Beiträge zur 36. Jahrestagung der Deutschen Gesellschaft für Biomedizinische Technik (DGBMT) im VDE, 25.–27. September 2002, Universität Karlsruhe"* Bd. 47, Ergänzung 1, Teil 2, Fachverlag Schiele & Schön, Berlin, 2002, S. 629–632

- [17] BETH, T. ; BOESNACH, I. ; HAIMERL, M. ; MOLDENHAUER, J. ; BÖS, K. ; WANK, V.: Analyse, Modellierung und Erkennung menschlicher Bewegungen. In: *1. SFB-Aus-sprachetag Human Centered Robotics (HCRS) 2002, 5.-6. Dezember 2002, Tagungsband*, Universität Karlsruhe (TH), 2002, S. 17–24.
- [18] HAIMERL, M. ; MOLDENHAUER, J. ; MENDE, U.: Zielgerichtete Aufbereitung und Visualisierung dreidimensionaler medizinischer Ultraschall Datensätze. In: HANDELS, H. (Hrsg.) ; HORSCH, A. (Hrsg.) ; LEHMANN, T. (Hrsg.) ; MEINZER, H.-P. (Hrsg.): *Tagungsband "Bildverarbeitung für die Medizin 2001"*, Lübeck 5./6. März 2001, Informatik aktuell, Springer Verlag, 2001, S. 117–121
- [19] BOESNACH, I. ; HAHN, M. ; HAIMERL, M. ; MOLDENHAUER, J.: Aufbereitung medi-zinischer Bilddaten – Überblick über aktuelle Arbeiten des IAKS im Bereich medizinische Bildverarbeitung. In: *E.I.S.S.-Report Bd. 3*, Europäisches Institut für Systemsicherheit, Universität Karlsruhe (TH), 2001
- [20] DÄUBER, S. ; HAIMERL, M. ; HOPPE, H. ; KREMPIEN, R. ; MENDE, U. ; MOLDENHAUER, J. ; WÖRN, H.: Aufbereitung medizinischer Bilddaten. In: WÖRN, H. (Hrsg.) ; MÜHLING, J. (Hrsg.) ; VAHL, C. (Hrsg.) ; MEINZER, H.-P. (Hrsg.): *Rechner- und Sensor-gestützte Chirurgie (Workshop-Proceedings)*, Gesellschaft für Informatik, Bonn, 2001, S. 189–202
- [21] MOLDENHAUER, J. ; HAIMERL, M. ; WALZ, M. ; WEISSER, G.: Datenschutz und Systemsicherheit für die medizinische Informationsverarbeitung. In: WÖRN, H. (Hrsg.) ; MÜHLING, J. (Hrsg.) ; VAHL, C. (Hrsg.) ; MEINZER, H.-P. (Hrsg.): *Rechner- und Sensor-gestützte Chirurgie (Workshop-Proceedings)*, Gesellschaft für Informatik, Bonn, 2001, S. 156–164
- [22] MOLDENHAUER, J. ; HAIMERL, M. ; MENDE, U.: Wavelet-basierte Visualisierung von Ultraschall-Volumendaten. In: HORSCH, A. (Hrsg.) ; LEHMANN, T. (Hrsg.): *Tagungs-band "Bildverarbeitung für die Medizin 2000"*, München, 13./14. März 2000, Informatik aktuell, Springer Verlag, 2000, S. 143–147
- [23] MOLDENHAUER, J.: *Visualisierung von Ultraschall-Volumendatensätzen mit kon-ventionellen und waveletbasierten Verfahren*, Universität Karlsruhe (TH), Diplomarbeit, 1999
- [24] MOLDENHAUER, J.: *Dreidimensionale Visualisierung medizinischer Daten*, Universi-tät Karlsruhe (TH), Studienarbeit, 1998



# Index

## A

A-Bild ..... 50  
 Abschattung..... 52  
 Absorption ..... 48  
 Absorptionskoeffizient ..... 48  
 Adaption ..... 152  
 Adjazenzmatrix ..... 8, 39, 109  
 Aktivierungsschwelle ..... 143  
 Aktivierungszustand ..... 143  
 akustische Information ..... 23  
 Akzeptanz ..... 65  
 Alltagsbewegung ..... 82  
 alternative Zustandsfolge ..... 109  
 Anatomie ..... 22, 43, 76  
 Anthropomatik ..... 1, 43  
 Anthropomatikdaten..... 5, 113, 119  
 Anthropomatiksystem .. 1, 10, 23, 39, 65  
 Apertur ..... 48  
 Approximationskoeffizient ..... 137  
 Approximationsraum ..... 136  
 AR..... *siehe* Prozess, autoregressiv  
 ARIMA..... *siehe* ARMA  
 ARMA..... *siehe* Prozess, autoregressive  
     moving average  
 Artefakte ..... 51  
 Audiodatenverarbeitung ..... 24  
 Audiokarte ..... 24  
 Audiomerkmale ..... 24  
 Aufenthaltswahrscheinlichkeit ..... 13  
 Ausgabeneuron ..... 146  
 Automat, probabilistisch ..... 39, 103

## B

B-Bild ..... 50, 55  
 Back-Propagation-Algorithmus.. 28, **149**  
 Bakis-Modell ..... **41**, 103  
 Bandbreite ..... 26  
 Bandenergieverhältnis ..... 26  
 Bandscheibenvorfall..... 22  
 Baum-Welch-Algorithmus .... 14, **15**, 21,  
     32, 40  
 Benutzerschnittstelle..... 120  
 Beobachtung ..... 10, 30, 126  
     empirisch ..... 155  
 Beobachtungsgleichung ..... 6  
 Beobachtungssequenz ..... **10**, 11, 12, 42,  
     57, 126  
     Länge..... 91  
 Beobachtungswahrscheinlichkeit ..... 10  
     des Modells .... 11, 12, 14, 20, 30, 57  
 beschreibende Statistik ..... 5  
 Beugung ..... 48  
 Bewegungsanalyse..... 65, 123  
 Bewegungsapparat ..... 66  
 Bewegungserfassung  
     magnetisch ..... 71, 87  
     videobasiert ..... 69  
 Bewegungsmerkmal ..... 66, 76, 126  
 Bewegungsmodell ..... 86  
 Bewegungsphase ..... 76, 81, 84, 95, 106,  
     123  
     elementar..... 66, 86  
 Bewegungssynthese ..... 123

Bewegungstrajektorie ..... 65, 67, 68, 76,  
77, 79, 93, 95, 104, 124, 128  
Bewegungszustand ..... 86  
Bias-Gewicht ..... 144, 153  
Bildgebungsprozess ..... 43  
Binärvektor ..... 28  
Blackman-Fenster ..... **25**, 89  
BodyViewer ..... 127  
Bohrgeräusch ..... **21**  
Boltzmann-Maschine ..... 151  
Boolesche Funktion ..... 147  
Brechung ..... 46  
Brechungsgesetz ..... 46

### C

C-Bogen ..... 22  
Calderón-Operator ..... 61  
Cepstral-Koeffizient ..... 27  
Cepstrogramm ..... 26  
Chapman-Kolmogorov-Gleichung ..... 8  
Chirurgieroboter ..... 23  
Client-Server-Modul ..... 121  
Computertomographie ..... 45  
Convex-Array ..... 50  
CORBA ..... 121  
cosine squasher ..... *siehe*  
Kosinus-Quetschfunktion  
Crashtest ..... 69  
CT ..... *siehe* Computertomographie

### D

Datenakquisition ..... 24, 49, 68  
Datenerfassung ..... 120  
Datenfluss ..... 119  
Datenflusskontrolle ..... 120  
Datenhandschuh ..... **74**, 87  
Datenquader ..... 50  
Datensicht ..... 120  
Daubechies-Wavelets ..... *siehe* Wavelet  
Dehnmessstreifen ..... 74, 76  
Dekomposition ..... 137  
depth-gain-compensation ..... *siehe*  
Tiefenkompensation

Detailkoeffizient ..... 137  
Detailraum ..... 137  
Device ..... 121  
DeviceConfiguration ..... 122  
DeviceData ..... 122  
DeviceManager ..... 122  
DeviceThread ..... 122  
DeviceUI ..... 122  
DGC ..... *siehe* Tiefenkompensation  
Dilatationsgleichung ..... 137  
Direct-Linear-Transformation-Technique  
69  
direkte Kinematik ..... 75, 77  
DLT ..... *siehe* Direct-Linear-  
Transformation-Technique  
DOF ..... *siehe* Freiheitsgrad  
Doppler-Effekt ..... 49  
DWT .... *siehe* Wavelet-Transformation,  
diskret  
dynamisches Laden ..... 121  
dynamisches Programmieren ..... 42

### E

EDA .... *siehe* explorative Datenanalyse  
Eindringtiefe ..... 48  
Einfallswinkel ..... 46  
Eingabeneuron ..... 146  
Elektromyogramm ..... 76  
Elman-Netz ..... 93, 98, **151**  
EM ..... *siehe* Expectation-Maximization  
EMG ..... *siehe* Elektromyogramm  
Energieverteilung ..... 25  
Entscheidungsfunktion ..... 155  
Epoche ..... 151  
Erkennung  
von Beobachtungssequenzen ..... 12  
von Bewegungsphasen ..... 95  
von Bohrgeräuschen ..... 21  
von Gewebetypen ..... 21, 59, 113  
von Personen ..... 89  
von Rahmenbedingungen ..... 92  
Event ..... 122  
Expectation-Maximization ..... 16

explorative Datenanalyse ..... 5

## F

FeatureGenerator ..... 126  
 Feed-Forward-Netz ..... 151  
   mehrschichtig ..... 28  
 feedback neural network ..... *siehe*  
   Neuronales Netz, rückgekoppelt  
 Feldgenerator ..... 72  
 Fenster ..... 127  
 Fensterung ..... 25, 78, 113  
 Fensterverschiebung ..... 27  
 Fermi-Funktion ..... 28, **145**  
 Fernbereich ..... 49  
 FFT ..... *siehe* Schnelle  
   Fourier-Transformation  
 Fingerbewegung ..... 74  
 Fingergelenk ..... 74  
 FIR-Filter ..... 25  
 Fokusbereich ..... 49  
 Forward-Algorithmus ..... *siehe*  
   Vorwärts-Algorithmus  
 Fourier-Transformation ..... 135  
   gefenstert ..... 25  
   invers ..... 26  
   schnell ..... 61  
 Frame ..... 123  
 FrameElement ..... 123  
 FrameInfo ..... 123  
 FrameReader ..... 124  
 Freiheitsgrad ..... 73, 74  
 Frequenzanalyse ..... 25

## G

GaitMapper ..... 127  
 Gangbewegung ..... 81, 84, 124  
 Gauß-Funktion ..... 153  
 Gauß-Verteilungen ..... 11  
 Gedächtnislosigkeit ..... *siehe*  
   Markov-Eigenschaft  
 Gelenkpunkt, Rekonstruktion ..... 73  
 Gelenkwinkel ..... 76  
   der Finger ..... 74

Gewebedichte ..... 46  
 Gewebetyp ..... 21, 59, 113  
 Gittersuchmethode ..... 30  
 Glättung ..... 77, 79  
 Goniometer ..... 76  
 Grammatik  
   kontextfrei ..... 68, 109  
   probabilistisch ..... 68  
 Graph ..... 8, 40, 119, 146, 151  
   ergodisch ..... 40

## H

Haar-Wavelet ..... *siehe* Wavelet  
 Handbohrer, chirurgisch ..... 24  
 Handgelenk ..... 74  
 haptische Information ..... 23  
 hidden layer ... *siehe* verborgene Schicht  
 Hidden-Markov-Modell ..... 7, 9, 30, 86  
   diskret ..... **10**, 16, 56  
   elementar ..... 106  
   ergodisch ..... 40  
   faktoriell ..... 68  
   gekoppelt ..... 68  
   Komposition ..... 105  
   kontinuierlich ..... **10**, 17, 31  
   linear ..... 40  
   parallel ..... 68  
   Struktur ..... 39  
   Training ..... **15**, 40  
 Hilfszustand ..... 103  
 HMM ..... *siehe* Hidden-Markov-Modell  
 hochdimensionaler Merkmalsraum .. 156  
 Hochpassfilter ..... 137  
 Hopfield-Netz ..... 151  
 humanoider Roboter ..... 65  
 Hyperebene ..... 30, 95, 98, 156, 157

## I

Implementierung ..... 119  
 Infrarot-Kamera ..... 75  
 initiale Zustandsverteilung .... 8, 42, 89,  
   104, 109  
 Iris ..... 54

Iterationsschritt ..... 17

## J

Java..... 121  
 Java COMM-API ..... 121  
 Java3D ..... 121  
 JMF ..... 121

## K

Kadaverpräparat ..... 24  
 Kaffeekanne ..... 86, 92, 109  
 Kalibrierkörper ..... 70  
 Kalibrierung ..... 74  
 Kalman-Filter ..... 7  
 Kameraparameter ..... 69  
 Kameransicht ..... 69  
 Kern ..... 30, 155  
 Kernfunktion ..... 156  
   invers multiquadratisch ..... 157  
   polynomiell ..... 157  
   radiale Basisfunktion ..... 157  
   sigmoidal ..... 157  
 Kernloch ..... 22  
 Kernmethode ..... 155  
 Klassifikation  
   hart ..... 58  
   manuell ..... 27, 96  
   weich ..... 58  
 Klassifikationsfehler ..... 99, 149  
 Klettband ..... 73  
 Kniegelenk ..... 77, 124  
 Knochendichte ..... 23  
 Knochenwand ..... 23  
 Körpermodell ..... 67, 87, 127  
 Kopplungsstück ..... 54  
 Kortikalis ..... 23  
 Kosinus-Quetschfunktion ..... 145  
 Kreuzvalidierung ..... 90  
 Krümmungsverhalten ..... 82, 104, 126  
 Kurvenplot ..... 121

## L

Lagrange-Multiplikator ..... 158

Layer ..... *siehe* Schicht  
 Leaving-one-out ..... 90  
 LED-Lauffichtleiste ..... 70  
 Lernrate ..... 150  
 Linear-Array ..... 50  
 lineare Modelle ..... 6  
 Linearisierungsraum ..... 156  
 Links-Rechts-Modell ..... 31, 40, 56, 103  
 logarithmische Wahrscheinlichkeit ..... 21  
 logarithmisches Spektrogramm ..... 26  
 Lookup-Table ..... 10, 56

## M

Magnetfeldsensor ..... 72, 75, 87  
 Magnetresonanztomographie ..... 45  
 Majoritätsprinzip ..... 14, 31, 32, 39, 58,  
   63, 68, 89, 90, 94, 96, 106, 113  
 Marker ..... 71, 75  
   optisch ..... 69  
   Position ..... 87  
 Markov-Eigenschaft ..... 7, 42  
 Markov-Kette ..... 8  
 Markov-Modell ..... 7, 11  
 Markov-Prozess ..... 7  
   einfach ..... 8  
   Ordnung ..... 8  
 Matlab ..... 27, 119, 126  
 Matrix  
   diagonaldominant ..... 41, 56  
   semistochastisch .. 8, 10, 40, 56, 104,  
     108, 110  
 Maximum-Likelihood ..... 16  
 Merkmalerweiterung ..... 77  
 Merkmalsgenerierung ..... 65, 126  
 Mikrofon ..... 24  
 Mischverteilung ..... 11, 126  
 Mischverteilungsklassifikator . 15, 32, 34  
 Mischverteilungskomponente ..... 154  
 mittlere Frequenz ..... 26  
 Mixture ..... *siehe* Mischverteilung  
 ML ..... *siehe* Maximum-Likelihood  
 ModelData ..... 123  
 Modell ..... 5

- Modellkonstruktion ..... 65  
 Modellparameter ..... 11, 15, 101  
 Modellstruktur ..... 101  
 Modul ..... 119  
   Ausgang ..... 121  
   Eingang ..... 121  
   Ereignisse ..... 121  
   Gruppe ..... 120  
 Monom ..... 156  
 MRT ..... *siehe*  
   Magnetresonanztomographie  
 MSA ..... *siehe* Multiskalenanalyse  
 Multiskalenanalyse ..... 135  
 Mutter-Wavelet ..... 137
- N**
- Nachbearbeitung ..... 71  
 Nahbereich ..... 49  
 Nervenzelle ..... 143  
 Neuabschätzung ..... 16, 21  
 Neurochirurgie ..... 21  
 Neuron ..... 143  
 Neuronales Netz.. 7, 27, 93, 98, **143**, 157  
   einschichtig ..... 147  
   mehrschichtig ..... 147  
   mit RBFs ..... 95, 98, 152, 157  
   rückgekoppelt ..... 151  
   Training ..... 149  
 NN ..... *siehe* Neuronales Netz  
 Normalverteilung ..... 11  
   mehrdimensional ..... 11  
 Nulldurchgang ..... 81  
 Nulleintrag ..... 40
- O**
- Oberflächenmikrofon ..... 24  
 Observable ..... *siehe* Beobachtung  
 ONB ..... *siehe* Orthonormalbasis  
 Operationsklemme ..... 24  
 Operationslampe ..... 24  
 Operationsplanungssystem ..... 22  
 Operationstisch ..... 24  
 Orthonormalbasis ..... 135
- Orthopädie ..... 84
- P**
- Parallelverarbeitung ..... 5  
 partielles Duplizieren ..... 123  
 PC ..... 24, 54, 126  
 Pedikel ..... 22  
 Pedikelschraube ..... 22  
 Perforation ..... 28  
 Perzeptron ..... 147  
 PhaseGenerator ..... 126  
 Piezo-Technik ..... 49  
 Plotter ..... 126  
 Prozess ..... 5, 23  
   autoregressiv ..... 6  
   autoregressive moving average ..... 6  
   kausal ..... 7  
   stationär ..... 7  
   stochastisch ..... 10, 31  
 Pseudodifferentieloperator ..... 61
- Q**
- quadratisch integrierbare Funktion.. 136
- R**
- radiale Basisfunktion . 30, **152**, 153, 157  
 Rahmenbedingungen ..... 76  
 Rampenfunktion ..... 144  
 Raummikrofon ..... 24  
 Rauschreduktion ..... 55  
 Rayleigh-Streuung ..... 48  
 RBF ..... *siehe* radiale Basisfunktion  
 RBFN ..... *siehe* Neuronales Netz mit  
   RBFs  
 recurrent neural network ..... *siehe*  
   Neuronales Netz, rückgekoppelt  
 Reflexion ..... 46  
 Reflexionsgesetz ..... 46  
 Reflexionskoeffizient ..... 47  
 Reibungseffekt ..... 48  
 Reiz ..... 143  
 Rekonstruktion ..... 137  
 Risiko ..... 155

Risikobereich ..... 23, 29  
 Röntgenbild ..... 22  
 Rückwärtsvariable ..... **13**, 16, 20

## S

Sättigung ..... 144  
 Scan ..... 50  
 Schallfeld ..... 46  
 Schallgeschwindigkeit ..... 46  
 Schallimpedanz ..... 46  
 Schallkeule ..... 47  
 Schallkopf ..... 49  
 Schallverstärkung ..... 52  
 Schallwelle ..... 24, 45  
 Schicht ..... 146  
 Schienbein ..... 71  
 Schnelle Fourier-Transformation ..... 61  
 Schwellwertfunktion  
   hart ..... 144  
   weich ..... 145  
 Segmentierung ..... 27, 50, 53, 54, 61, 96  
 ShareableList ..... 123  
 Short-Time Fourier-Transform ..... *siehe*  
   Fourier-Transformation  
 Signal ..... 25  
 Signalanteil ..... 25  
 Simulation ..... 54  
 Singulärwertzerlegung ..... 70, 73  
 SinkPort ..... 122  
 Skalenindex ..... 136  
 Skalierung ..... 20  
 Skalierungsfunktion ..... 136  
 Skalierungskoeffizient ..... 137  
 Skalierungsstufe ..... 78, 81  
 Sonographie ..... 43, 45  
 SourcePort ..... 122  
 Speckle ..... **51**  
 Spektrogramm ..... 25  
 Spinalkanal ..... 22  
 Spongiosa ..... 23  
 Sportwissenschaft ..... 81, 84  
 Sprungaufschlag ..... 78  
 Sprunggelenk ..... 85

statistische Lerntheorie ..... 155  
 Sternum ..... 54  
 STFT ..... *siehe* Fourier-Transformation  
 Streuung ..... 47  
 Strichdiagramm ..... 78  
 Stützphase ..... 81  
 subsampling ..... *siehe* Unterabtastung  
 Support-Vector-Machine ... 7, 27, 95, 98,  
   155  
 SVD ..... *siehe* Singulärwertzerlegung  
 SVM ..... *siehe* Support-Vector-Machine  
 Symbol ..... 10  
 Synchronisation ..... 70

## T

Teilband ..... 26  
 Testdaten ..... 27  
 Theorem von Mercer ..... 156  
 Tiefenkompensation ..... 49, 52, 55  
 Tiefpassfilterung ..... 137  
 Timer ..... 125  
 Tischdecken ..... 86  
 TM-Bild ..... 50  
 Token-Passing ..... 43  
 Tracking ..... *siehe* Bewegungserfassung  
 Trainingsdaten ..... 27  
 Transferfunktion ..... 28, 143, 144  
   begrenzt linear ..... 144  
   binär ..... 144  
   Fermi-Funktion ..... 145  
   Kosinus-Quetschfunktion ..... 145  
   linear ..... 75  
   sigmoidal ..... 82, 105, 145  
   Tangens hyperbolicus ..... 145  
 Transmissionskoeffizient ..... 47  
 Tumor ..... 54

## U

Ultraschall ..... **43**, 113  
   Physik ..... 45  
   Scanner ..... 45  
 Ultraschallbild ..... 39, **43**  
 Unterabtastung ..... 138

## V

VAMOS ..... 119  
 Variation  
   inter-individuell ..... 76  
   intra-individuell ..... 66, 76  
 verborgene Schicht ..... 147  
 Verdeckung ..... 70  
 Verfeinerungsgleichung ..... 137  
 verteiltes Rechnen ..... 121  
 Verweildauer ..... 9, 41  
 Verweilen ..... 9, 40, 56, 103  
 Video-Kamera ..... 70  
 virtuelle Umgebung ..... 67  
 Visualisierung ..... 87, 119, 121  
 Viterbi-Algorithmus .... **42**, 60, 111, 126  
 Volleyball ..... 78  
 Volumen ..... 26, 50  
 Volumendatensatz ..... 22  
 Volumenelement ..... 50  
 Vorwärts-Algorithmus .... **12**, 14, 32, 42,  
   57  
 Vorwärts-Rückwärts-Algorithmus ..... 14  
 Vorwärtsvariable ..... **13**, 16, 20  
 Vorwärtswahrscheinlichkeit ..... 42  
 Vorzeichenfunktion ..... 82, **144**  
 Voxel ..... *siehe* Volumenelement

## W

Wavelet ..... 135, 137  
   biorthogonal ..... 138  
   Daubechies-Wavelet ..... 139  
   Differentialoperator ..... 141  
   Haar-Wavelet ..... 78, **141**  
 Wavelet-Basis ..... 139  
 Wavelet-Familie ..... 141  
 Wavelet-Koeffizient .... 78, 85, 103, 126,  
   137  
 Wavelet-Transformation ..... 77, 135  
   diskret ..... 78, 126  
   redundant ..... 138  
 Wellenausbreitung ..... 54  
 Wettkampfsport ..... 69  
 Winkelgeschwindigkeit ..... 77, 78, 81

Winkeltrajektorie ..... 77, 88, 104, 128  
 Wirbelfraktur ..... 22  
 Wirbelkörper ..... 22  
 Wirbelsäule ..... 21  
 Wirbelverschiebung ..... 22

## X

XML ..... 121, 126

## Z

Zeilensummenkriterium ... **8**, 21, 40, 110  
 Zeitreihe ..... 5  
 Zeitreihenanalyse ..... 5  
 Zielausgabe ..... 28  
 Ziliarkörper ..... 54  
 Zugang ..... 22  
 Zustand ..... 7  
   Anzahl ..... 97  
   versteckt ..... 10  
 Zustandsgleichung ..... 6  
 Zustandsraum ..... 39  
 Zustandssequenz .... **10**, 11, 42, 59, 107,  
   109, 126  
 Zustandsübergangsmatrix ..... **8**, **16**, 31,  
   **39**, 42, 56, 89, 102, **107**  
 Zweiskalengleichung ..... 137

In dieser Arbeit werden adaptive Methoden zur Analyse von Anthropomatikdaten entwickelt. Zielsetzung ist die automatische Erkennung von Systemzuständen mit Hidden-Markov-Modellen. Anwendungsbeispiele sind Bohrgeräusche aus der Wirbelsäulenchirurgie, medizinische Ultraschallbilder und menschliche Bewegungsdaten. Neben dem Vergleich mit anderen Klassifikationsverfahren werden Merkmalsgenerierung, geeignete Modellstrukturen, Optimierung der Zustände und Aspekte der Implementierung besprochen.

ISBN-13: 978-3-86644-068-5

ISBN-10: 3-86644-068-5

---

[www.uvka.de](http://www.uvka.de)