

# **Markerlose Ganganalyse mit einem Multikamerasystem**

## **Dissertation**

zur Erlangung des akademischen Grades

**Doktoringenieur  
(Dr.-Ing.)**

von Dipl.-Ing. Roman Calow

geb. am 08.01.1972 in Lutherstadt Wittenberg

genehmigt durch die Fakultät für Elektrotechnik und Informationstechnik  
der Otto-von-Guericke-Universität Magdeburg

Gutachter:

Prof. Dr.-Ing. Bernd Michaelis

Prof. Dr.-Ing. Rudolf Mester

Promotionskolloquium am 04.03.2005



## Zusammenfassung

Gegenstand dieser Arbeit ist die markerlose Ganganalyse mit einem Multikamerasystem. Dafür wird das Analyse-durch-Synthese-Konzept zur Bildsequenzauswertung eines gehenden Menschen erweitert. Nach der Erläuterung der Parameterschätzung durch Ausgleichsrechnung, wird die Modellierung des Skeletts, der Körperform und der Kameras beschrieben. Neben Methoden für die Approximation des 3D-Modells, werden Verfahren zur Korrespondenzfindung zwischen Modell und Bild vorgestellt. Ein (quasi-)periodisches Modell der Bewegung wird eingeführt, das die Anzahl der zu schätzenden Parameter je Bildfolge verringert. Dadurch lässt sich die Robustheit und Genauigkeit des Ansatzes erhöhen. Das (quasi-)periodische Bewegungsmodell wird weiterhin dazu verwendet, die vom Algorithmus benötigten Näherungswerte bereitzustellen. Die Reproduzierbarkeit der Ergebnisse und die Eignung des Systems zur Ganganalyse wird demonstriert.

## Danksagung

Die vorliegende Arbeit entstand während meiner Tätigkeit am Institut für Elektronik, Signalverarbeitung und Kommunikationstechnik (IESK) der Otto-von-Guericke-Universität Magdeburg und bei Innomed e.V. Für die Betreuung, die freundliche Unterstützung und die wertvollen Anregungen bedanke ich mich herzlich bei Herrn Prof. Dr.-Ing. habil. B. Michaelis. Weiterhin möchte ich Prof. Dr.-Ing. Rudolf Mester für die Übernahme des Korreferates danken.

Dank sagen möchte ich allen Kollegen des IESK, die einen Anteil am Entstehen dieser Arbeit hatten. Ganz besonders herzlich möchte ich mich bei Dipl.-Ing. Robert W. Kuhn für die fruchtbare Zusammenarbeit, die freundschaftliche Arbeitsatmosphäre und fachlichen Diskussionen bedanken, genauso bei Frau Dipl.-Ing. Alexandra Lehmann und Beatrice E. Stark für das Lesen der Arbeit. Herrn Dipl.-Ing. Hans Rodiek und den Mitarbeitern der Werkstatt möchte ich für das Design und den Aufbau des Systems zur Synchronisierung der Multikameraanordnung danken.

Der größte Teil der Arbeit wurde durch Innomed e.V. und die INB Vision AG sowie dem BMBF finanziert (EU 0046KE0000 und BMBF 03i0404A), mein Dank gilt in diesem Zusammenhang Prof. Dr. Günther Gademann.

Magdeburg, den 25. Mai 2004

Roman Calow

In Liebe, für **Bea**.

# Inhaltsverzeichnis

<b>1. Einleitung</b>	<b>9</b>
<b>2. Grundlagen</b>	<b>11</b>
2.1. Der menschliche Gang . . . . .	11
2.1.1. Der Laufzyklus . . . . .	12
2.1.2. Überlegungen zur Bildaufnahmefrequenz . . . . .	14
2.2. Ansätze zur Bildfolgenauswertung . . . . .	16
2.2.1. Ansätze ohne explizites 3D-Modell . . . . .	16
2.2.2. Ansätze mit expliziten 3D-Modellen . . . . .	19
2.2.3. Der verwendete Ansatz . . . . .	23
<b>3. Modellierung des menschlichen Skeletts</b>	<b>29</b>
3.1. Festlegung des Weltkoordinatensystems . . . . .	29
3.2. Ein (zu) einfaches Skelettmodell . . . . .	30
3.3. Verbessertes Skelettmodell . . . . .	38
3.4. Grundlagen der Koordinatentransformationen . . . . .	39
3.4.1. Kinematische Ketten . . . . .	40
3.5. Ein kinematisches Modell der Beine . . . . .	42
3.6. Gelenkwinkelgrenzen . . . . .	46
<b>4. Modellierung der visuellen Erscheinung</b>	<b>47</b>
4.1. Möglichkeiten der 3D-Formbeschreibung . . . . .	49
4.2. Formbeschreibung durch Konturtemplates . . . . .	50
4.3. Formbeschreibung durch Kegelstümpfe . . . . .	54
4.4. Formbeschreibung durch Dreiecksnetze . . . . .	57
4.4.1. Skelett Deformation . . . . .	60
4.5. Modellierung der Oberflächenfarben . . . . .	61
4.6. Bewertung der Methoden . . . . .	62
<b>5. Modellierung der Bildaufnahme</b>	<b>63</b>
5.1. Kameraaufbau . . . . .	63
5.2. Kameramodell . . . . .	64
5.2.1. Projizieren von Punkten . . . . .	65
5.2.2. Projizieren von Differenzialen . . . . .	66
5.3. Erzeugung synthetischer Bilder . . . . .	68

5.3.1. Sichtbarkeit von Körperteilen . . . . .	69
5.3.2. Erkennen von Selbstverdeckungen . . . . .	72
<b>6. Approximation des Modells</b>	<b>73</b>
6.1. Übersicht geeigneter Approximationsverfahren . . . . .	73
6.2. Approximieren des Modells an die 3D-Punkte . . . . .	75
6.3. Approximieren des Modells an die Bildpunkte . . . . .	77
6.4. Approximieren des Modells an die Konturen im Bild . . . . .	79
6.5. Gegenüberstellung der Gleichungen . . . . .	81
<b>7. Das Korrespondenzproblem und Lösungsansätze</b>	<b>83</b>
7.1. Problemanalyse . . . . .	84
7.2. Möglichkeiten der Segmentierung des Probanden . . . . .	85
7.3. Labeling der Bildzeilen . . . . .	87
7.4. Disparität zur Unterscheidung des linken oder rechten Beines . . . . .	88
7.5. 2D-Templatmatching mit Hilfe eines genetischen Algorithmus . . . . .	89
<b>8. (Quasi-)periodisches Bewegungsmodell</b>	<b>93</b>
8.1. Approximation des Bewegungsmodells mit einem nichtlinearen Körpermodell . . . . .	100
8.1.1. Experiment . . . . .	101
8.1.2. Berechnung der Periodendauer in einer Ausgleichung . . . . .	103
8.2. Methoden der Repräsentation quasiperiodischer Bewegungsgrößen . . . . .	104
8.2.1. Separation von periodischem und aperiodischem Anteil . . . . .	105
8.2.2. Ein separates Bewegungsmodell für jeden Schritt . . . . .	105
8.2.3. Fourierreihe mit vielfacher Periodendauer der Schrittfrequenz . . . . .	107
8.3. Die Schrittzahl . . . . .	108
8.4. Schlussfolgerungen . . . . .	109
<b>9. Gewinnung von Näherungswerten</b>	<b>111</b>
9.1. Annahmen und Vereinfachungen . . . . .	111
9.2. Initialisierung der Translationsparameter . . . . .	114
9.3. Initialisierung der Orientierungsparameter . . . . .	116
9.4. Methoden zur Initialisierung der Gelenkwinkel . . . . .	116
9.4.1. Manuelle Vorgabe einer Schrittfrequenz und einer Schrittzahl . . . . .	116
9.4.2. Manuelles Markieren der Sprunggelenke . . . . .	117
9.4.3. Automatische Initialisierung der Schrittzahlen . . . . .	118
<b>10. Experimente, aufgetretene Probleme und deren Lösungen</b>	<b>123</b>
10.1. Experimente mit dem einfachen Skelettmodell . . . . .	123
10.2. Effekte bei der Approximation . . . . .	126
10.3. Manuelle Initialisierung, Einzelschrittauswertung . . . . .	131
10.4. Approximation mit quasiperiodischem Bewegungsmodell . . . . .	134
10.4.1. Problem der zeitlichen Synchronisierung des Bewegungsmodells . . . . .	137

10.4.2. Genaue Berechnung jeder einzelnen Schrittzahl . . . . .	140
10.4.3. Fourierreihe mit vielfacher Periodendauer der Schrittfrequenz . . .	144
<b>11. Ergebnisse des Messsystems</b>	<b>145</b>
11.1. Definition der Messgrößen . . . . .	145
11.2. Messergebnisse der optischen Ganganalyse . . . . .	146
11.3. Reproduzierbarkeit der Messungen und Simulation von Gangfehlern . . .	152
11.3.1. Gegenüberstellung der Laufrichtung . . . . .	152
11.3.2. Gegenüberstellung von gesundem Gang und simulierter Erkrankung	156
11.4. Bewertung der Ergebnisse . . . . .	159
<b>12. Zusammenfassung</b>	<b>161</b>
<b>Anhang</b>	<b>164</b>
<b>A. Details des Laufzyklus</b>	<b>165</b>
A.1. Standphase . . . . .	165
A.1.1. Heel Strike - Initial Contact (Fersenkontakt) . . . . .	165
A.1.2. Foot Flat (Fuß flach) . . . . .	166
A.1.3. Mid Stance (Mittelstand-Phase) . . . . .	166
A.1.4. Heel Off (Ferse hebt ab) . . . . .	166
A.1.5. Toe Off (Zehen heben ab) . . . . .	167
A.2. Schwungphase . . . . .	167
A.2.1. Initial Swing - Acceleration (Beschleunigung) . . . . .	167
A.2.2. Mid Swing (Mittelschwung) . . . . .	168
A.2.3. Bremsphase/Deceleration/Terminal Swing . . . . .	168
<b>B. Lösung eines Gleichungssystems mit Hilfe der Singulärwertzerlegung</b>	<b>169</b>
<b>C. Standardabweichungen der Unbekannten</b>	<b>173</b>
<b>D. Homogene Transformationsmatrizen</b>	<b>177</b>
D.1. Elementare Rotationsmatrizen . . . . .	177
D.2. Inverse elementare Rotationsmatrizen . . . . .	178
D.3. Elementare Translationsmatrizen . . . . .	178
D.4. Inverse elementare Translationsmatrizen . . . . .	179
<b>E. Automatische Ableitung kinematischer Ketten</b>	<b>181</b>
<b>F. Numerische Berechnung der Ableitungen</b>	<b>183</b>
<b>G. Kamerakalibrierung</b>	<b>185</b>
<b>H. Formelzeichen und Abkürzungen</b>	<b>191</b>
<b>Literaturverzeichnis</b>	<b>192</b>

## *Inhaltsverzeichnis*



# 1. Einleitung

Von der immer größeren Leistungsfähigkeit verfügbarer Rechner haben die Bereiche Bildverarbeitung und Computer Vision erheblich profitiert. Mit jeder neuen Rechnergeneration rücken immer komplexere Algorithmen in die praktische Anwendbarkeit. Bezogen auf die Analyse von Bildfolgen erscheint die Zeit reif, den zum Teil erheblichen manuellen Aufwand zur Bildauswertung durch automatische Algorithmen zu ersetzen.

Der Wunsch bestand, aus den Bildfolgen einer oder mehrerer Kameras automatisch den Bewegungsablauf einer abgebildeten Person zu gewinnen. So allgemein formuliert ist dieses Ziel momentan nicht zu erreichen. Realisierbar erscheint aber zunächst die medizinische Ganganalyse, die derzeitig entweder durch einen Arzt oder selten mit automatischen Systemen zur Bewegungsanalyse durchgeführt wird.

Eine subjektive Beurteilung ist immer problematisch, im Besonderen, wenn die Analysen in längeren Zeitabständen erfolgen sollen, weil es an einer präzisen Aufzeichnung und Speicherung der Daten fehlt. Weiterhin ist die Reproduzierbarkeit der Aussagen des Bewertenden fraglich. Auch ein Vergleich mit anderen Probanden gestaltet sich häufig schwierig. Die pure Aufzeichnung von Bildsequenzen mit einer oder mehreren Kameras und einer geeigneten Archivierung wäre zwar ein wichtiger Schritt in die richtige Richtung, für eine Objektivierung der Analysen ist damit aber noch nicht viel gewonnen, auch wäre der benötigte Speicherbedarf für die Rohdaten hoch.

Durch eine automatische Erfassung einer Anzahl relevanter Bewegungsparameter wären einige der Probleme gelöst. Da zum einen die Archivierung erleichtert und zum anderen ein objektiver, da automatisierbarer Vergleich dieser wenigen Parameter ermöglicht wird. Dadurch werden neue Operationen mit den gewonnenen Daten, wie Suchen, Sortieren und die Verwaltung in einer Datenbank möglich. Das würde völlig neue Möglichkeiten der medizinischen Statistik erlauben.

Allerdings erfordern die derzeitig verfügbaren automatischen Systeme eine aufwendige Anbringung von Markern an der Haut des Probanden oder wenigstens einen eng anliegenden entsprechend markierten Anzug. Marker sind zwar etabliert und existierende Messsysteme arbeiten zuverlässig, jedoch erfordert das Anbringen Zeit und sie stehen im Verdacht, den Bewegungsablauf des Probanden zu beeinflussen. Nicht zuletzt ist das Anbringen der Marker wieder eine Quelle von Subjektivität, die es zu reduzieren gilt. Spätere Anwendungen wie Identifikations- oder Überwachungssysteme würden durch ein markerloses Verfahren überhaupt erst ermöglicht.

Daraus resultiert die wichtigste Nebenbedingung dieser Arbeit: Das optische Messsystem

## 1. Einleitung

soll ohne Marker und möglichst automatisch in einer Klinik-Umgebung arbeiten können. Durch Weglassen der Marker gehen jedoch wichtige Informationen für die Bildauswertung verloren. Das erschwert die Ableitung der Bewegungsgrößen, wie der Gelenkwinkelverläufe oder Geschwindigkeiten der Körperteile in erheblicher Weise. Ein Teil der verloren gegangenen Informationen lässt sich durch a priori Wissen in die eigentliche Messung einbringen. Zur Repräsentation bietet sich ein maßstabsgerechtes dreidimensionales Körpermodell eines Menschen an.

Viel Know-How für die synthetische Darstellung eines menschlichen Körpers kann dabei aus der Computeranimation übernommen werden. So ist zu erwarten, dass die optische Messtechnik in erheblicher Weise von den Entwicklungen dort profitieren kann, wenn die Brücke zwischen der Animation eines Modells und der Schätzung der Bewegungsparameter geschlagen ist. Dazu wird ein Analyse-durch-Synthese-Ansatz als Ausgangspunkt verwendet, der bereits aus der Videokompression bekannt ist. Für die Auswertung einer Bildsequenz eines gehenden Menschen ist dieses Verfahren in der ursprünglichen Fassung nicht einsetzbar und muss an das Problem angepasst werden. Es werden Methoden der Ausgleichsrechnung aus der Photogrammetrie verwendet, die nicht nur die gesuchten Parameter schätzen können, sondern auch eine Aussage über ihre Genauigkeit erlauben.

Die große Anzahl möglicher Anwendungen in der Mensch-Maschine Interaktion, Identifikation, Überwachung, Sport und Humanmedizin motivierte zahlreiche Publikationen zum Thema markerlose Bewegungserfassung. Auch mangelt es nicht an Übersichtsartikeln [Sch00], [MG01], [AC99], [GD99], die selbst zum Teil den Umfang von Dissertationen haben. Dies unterstreicht die Bedeutung der Thematik, aber auch die Problematik einer effizienten Lösung. Ein 'State-of-the-Art' zu den Details wird an den jeweiligen Stellen im Text gegeben, vor allem aber von Kapitel 2 bis Kapitel 5.

Im Kapitel 2 werden die Grundlagen der Parameterschätzung erläutert. Weiterhin wird begründet, weshalb der modellbasierte Ansatz verfolgt wird. In dieser Arbeit werden unter anderem zwei große Themenbereiche behandelt:

- Modellierung des Probanden (Kapitel 3 und Kapitel 4) und der Bildaufnahme (Kapitel 5),
- Parameterschätzung durch Modell-Approximation (Kapitel 6), Gewinnung der Korrespondenz (Kapitel 7) und der Bereitstellung guter Näherungswerte (Kapitel 9).

Die klassische Ausgleichsrechnung wurde auf das Problem der Bewegungsanalyse angewendet, so ist jetzt auch eine Angabe der Genauigkeit geschätzter Parameter möglich. In Kapitel 8 wird ein neues quasiperiodisches Bewegungsmodell des Laufzyklus vorgestellt, das eine Verringerung der Parameteranzahl und damit eine Steigerung von Robustheit und Genauigkeit gestattet. In Kapitel 10 wird das System in verschiedenen Experimenten getestet. Aufgetretene Probleme werden dargestellt und gefundene Lösungen gezeigt. Ergebnisse zu Untersuchungen der Reproduzierbarkeit der Messungen werden in Kapitel 11 diskutiert. Eine Zusammenfassung der Arbeit wird in Kapitel 12 gegeben. Im Anhang sind interessante Details zu finden, die im Text den 'roten Faden' gestört hätten.

## 2. Grundlagen

Eine sehr gute Einführung in die Dynamik der menschlichen Bewegung ist das Buch von Winter: 'Biomechanics and motor control of human movement' [Win90]. Am Anfang werden zahlreiche Sensorprinzipien zur Erfassung der Bewegung vorgestellt. Im hinteren Teil des Buches findet man zunächst einfache, dann kompliziertere Verfahren zur Berechnung von Gelenkkräften und Momenten, wenn die Bewegung des Probanden und andere Daten bekannt sind. Auch wird die Synthese menschlicher Bewegung mit Hilfe der Mehrkörperphysik beschrieben. Ein automatischer Algorithmus zur Simulation der Mehrkörperdynamik ist in [Wit77] zu finden. Eine Software-Bibliothek zur Mehrkörperdynamik mit Quelltext und die dazu gehörigen Veröffentlichungen findet man auf der Web-Site von [LAZ98].

Ein Standardwerk speziell zur Ganganalyse ist [VDO92]. Es ist ein Softwarepaket zur Ganganalyse mit einer sehr ausführlichen Dokumentation. Übersichtlich werden die einzelnen Gangphasen und grundlegende Prinzipien zur Ganganalyse dargestellt. Die Erfassung der Bewegung basiert auf Markern, die an bestimmten Stellen des Probanden angebracht werden.

### 2.1. Der menschliche Gang

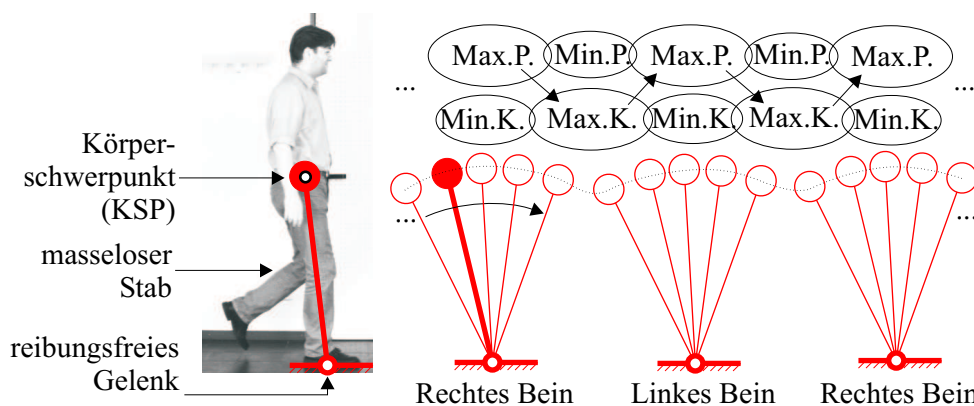


Abbildung 2.1.: Modell des inversen Pendels, Maximum potenzielle Energie (Max.P.), Maximum kinetische Energie (Max.K.), Minimum potenzielle Energie (Min.P.), Minimum kinetische Energie (Min.K.)

## 2. Grundlagen

Der aufrechte Gang stellt eine besondere Herausforderung an das Individuum, das gezwungen ist, Balance im Gravitationsfeld der Erde durch geschickte Steuerung des Bewegungsapparates zu erreichen. Ansatzweise werden die grundlegenden Zusammenhänge des aufrechten Ganges häufig durch das physikalische Modell eines inversen Pendels erläutert. Dabei spielt die nahezu periodische Umwandlung von kinetischer Energie in potenzielle Energie eine wichtige Rolle. Abbildung 2.1 verdeutlicht das.

### 2.1.1. Der Laufzyklus

Die nahezu periodische Natur des menschlichen Ganges ist ein wichtiges Element für dessen Analyse, deshalb soll hier eine Betrachtung des zeitlichen Ablaufes erfolgen. Wichtige Elemente wurden dafür aus [VDO92] entnommen.

Normalerweise wird die Bewegung eines Beines betrachtet. Demzufolge beziehen sich auch alle Angaben, sofern nicht anders angegeben, jeweils auf ein Bein. Es besteht ein einfacher Zusammenhang zwischen der *Schrittfrequenz*  $f$  und der *Periodendauer*  $T$  des Probanden.

$$f = 1/T \quad (2.1)$$

Die konstante *mittlere Geschwindigkeit*  $v$  eines gehenden Probanden hängt von der Schrittfrequenz und der *Schrittlänge*  $s_l$  (engl. Stride Length) ab. Dabei bezeichnet die Schrittlänge immer den Abstand zweier aufeinander folgender Standpunkte eines Beines.

$$v = w \cdot f = \frac{s_l}{T} \quad (2.2)$$

Eine höhere Geschwindigkeit kann also nach Gl.(2.2) durch eine Erhöhung der Schrittfrequenz  $f$  und eine Erhöhung der Schrittlänge  $s_l$  erreicht werden.

In der Signalverarbeitung werden periodische Größen häufig in Radian gekennzeichnet. Bei der Ganganalyse handelt es sich um die Betrachtung weitgehend periodischer Größen. Die Angabe des Timings könnte also ebenfalls in Radian oder ggf. in Grad erfolgen ( $2\pi$  oder  $360^\circ$  je Periode). In der Ganganalyse ist es aber üblich, das Timing innerhalb eines Laufzyklus in Prozent (%) anzugeben. Auf jeden Fall kann jede Periode mit einer fortlaufenden *Schrittnummer* (einheitslose Ganzzahl) versehen werden. Für die Betrachtung der Bewegung über eine längere Zeitspanne wären dann immer Schrittnummer und Prozentangabe erforderlich. Eine einzige fortlaufende Zahl wäre für Algorithmen der Bildverarbeitung aber viel besser zu handhaben, die sowohl die Schrittnummer als auch das Timing innerhalb eines Schrittes repräsentieren kann. Dafür wird in dieser Arbeit die einheitslose reelle *Schrittzahl*  $\theta$  verwendet. Hat sich die Schrittzahl um eins erhöht, dann wurde der Laufzyklus genau einmal durchlaufen. Die ganzzahlige Schrittnummer lässt sich leicht durch Abrunden der Schrittzahl ermitteln. Wird der Nachkommaanteil der Schrittzahl mit 100 % multipliziert, kann das Timing der Ereignisse wie gewohnt in Prozent erfolgen.

## 2.1. Der menschliche Gang

Bei einem normalen gesunden Gang, durchlaufen die beiden Beine einen Laufzyklus (siehe Abbildung 2.2) mit einem zeitlichen Versatz von etwa 50 %. Der Laufzyklus unterteilt sich grob in die Stand- und die Schwungphase. Neben dieser groben Einteilung des Laufzyklus lassen sich weitere charakteristische Phasen und Zeitpunkte definieren. In Abbildung 2.2 sind diese zu sehen. Phasen, in denen beide Beine Bodenkontakt haben, wurden grau hinterlegt (erste und zweite Double-Support-Phase). Durch Identifizierung von Ereignissen des Laufzyklus werden Operationen zur zeitlichen Normierung eines jeden Schrittes möglich. Das wiederum erlaubt Vergleiche gewonnener Daten:

- des linken und des rechten Beines
- eines Schrittes mit anderen Schritten des Probanden
- bei verschiedenen Geschwindigkeiten
- von verschiedenen Probanden

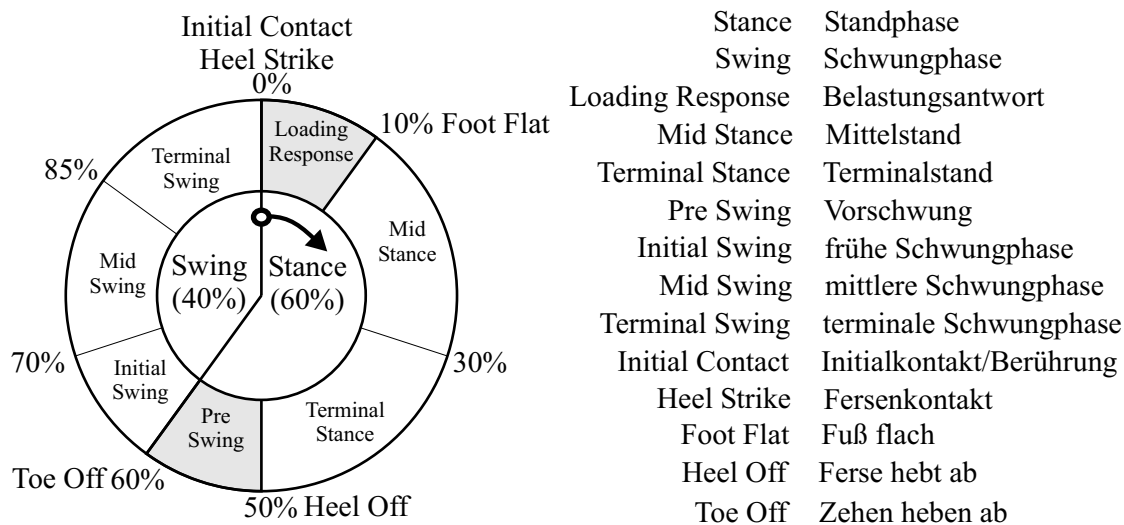


Abbildung 2.2.: Zu sehen ist das Timing eines normalen Laufzyklus (im Uhrzeigersinn). Die Übersetzung der Begriffe wurde [Bär01] entnommen.

Weiterhin eröffnen sich dadurch neue Möglichkeiten der zeitlichen Modellierung des menschlichen Ganges. Allerdings lassen sich nur für einige Ereignisse wirklich exakte Zeitpunkte bestimmen, da Kameras zu diskreten Abtastzeitpunkten (meist äquidistant) je ein Bild aufzeichnen. Wenn ein Ereignis jedoch zwischen zwei Abtastzeitpunkten eingetreten ist, dann lässt sich das Timing meist nicht genauer als die Abtastzeit selbst angeben.

## 2. Grundlagen

### Standphase

Man unterscheidet Stand- und Schwungphase. Hat ein Bein Kontakt mit dem Boden, (zum Beispiel Ferse, Sohle oder Zehen) befindet es sich in der Standphase. Diese lässt sich zunächst weiter unterteilen in:

- erste Doppelstandphase (First double support)
- Einzelstandphase (Single stance)
- zweite Doppelstandphase (Second double support).

Nach der zweiten Doppelstandphase geht das Bein in die Schwungphase, es hat dann keinen Kontakt mit dem Boden und wird zum nächsten Standpunkt bewegt. Danach beginnt der ganze Laufzyklus erneut. Die Standphase nimmt etwa 60 % des Laufzyklus ein. Eine genauere Darstellung der Details der Standphase ist dem Anhang A zu entnehmen.

### Schwungphase

Die Schwungphase ist für die eigentliche Bewegung des Probanden nicht relevant. Sie ist aber notwendig, um das Schwungbein zum nächsten Standpunkt zu führen. Für das Individuum ist es vorteilhaft, diesen Vorgang möglichst energiearm zu gestalten. Für die automatische Bildauswertung ist die Schwungphase etwas problematisch, weil hier die größten Geschwindigkeiten und Beschleunigungen auftreten, gleichzeitig aber auch die meisten Selbstverdeckungen der beiden Beine stattfinden. Für einen Bildverarbeitungsalgorithmus, der lediglich die Konturen der Beine auswertet, besteht in dieser Phase die größte Verwechslungsgefahr, weil die Beine natürlich sehr ähnlich aussehen und sich gleichzeitig auch noch sehr nahe kommen. Eine genauere Darstellung der Details der Schwungphase ist dem Anhang A zu entnehmen.

#### 2.1.2. Überlegungen zur Bildaufnahmefrequenz

Normalerweise wird die Bildwiederholrate eines Gerätes, wie einem Monitor oder einer Grafikkarte in Bilder je Sekunde (engl. *frames per second, fps*) angegeben. Bei der Bildaufnahme mit einem synchronisierten Multikamerasystem könnte diese Einheit irreführend sein, denn zu jedem Aufnahmezeitpunkt werden so viele Bilder geliefert, wie Kameras verwendet werden. Deshalb wird in dieser Arbeit der Begriff *Bildaufnahmefrequenz* oder kurz *Bildfrequenz* mit der Einheit Hertz (1 Hz=1/s) benutzt. Die maximal mögliche Bildfrequenz hängt von der verwendeten Hardware (Kameras, Framegrabber, Host) ab.

Aus technischen Gründen wurden in dieser Arbeit die Bildsequenzen mit einer Bildfrequenz von 10 Hz oder 20 Hz aufgezeichnet. Es hat sich aber gezeigt, dass für derartig dynamische Bewegungen, wie dem menschlichen Gang deutlich höhere Bildfrequenzen wünschenswert sind. Bei steigender Frequenz werden die Änderungen in den Bildern

## 2.1. Der menschliche Gang

kleiner und es vereinfacht sich die Korrespondenzfindung von Zeitschritt zu Zeitschritt. Bei der Ganganalyse treten die höchsten Geschwindigkeiten in der Schwungphase des Beines auf. Dabei ist die größte theoretisch zu erwartende Geschwindigkeit des Beines etwa  $2.5 = \frac{1}{0.4}$  mal höher als die mittlere Geschwindigkeit des Probanden. Das hängt mit dem Timing des Laufzyklus zusammen, da die Standphase 60 % und die Schwungphase 40 % der Zeit in Anspruch nimmt. Das Sprunggelenk muss also den selben Weg in nur etwa 40 % der Zeit zurücklegen. Nach ersten Experimenten wurde der Faktor 2.5 auf 2.7 korrigiert, da durch Beschleunigungs- und Bremsphase die tatsächlich gemessenen Geschwindigkeiten des Sprunggelenkes etwas höher als erwartet ausfallen.

$$v_{max} = 2.7 \cdot v \quad (2.3)$$

In der Seitenansicht besitzt das Bein am Sprunggelenk eine Breite von etwa 0.08 m. Für die Korrespondenzfindung wäre es günstig, wenn sich von Zeitschritt zu Zeitschritt das Bein nicht mehr als die Hälfte seiner Breite bewegt, damit die Korrespondenz der Kantenpunkte von Zeitschritt zu Zeitschritt einfach berechnet werden kann.

Wenn ein Fußgänger jedoch eine Durchschnittsgeschwindigkeit von 1.388 m/s (5 km/h) besitzt, dann ist für das Sprunggelenk eine Höchstgeschwindigkeit von 3.75 m/s zu erwarten. Bei 10 Hz bewegt sich das Schwungbein je Zeitschritt um 0.375 m, das ist deutlich mehr als seine Breite. Voraussichtlich lassen sich bereits bessere Ergebnisse ab 100 Hz aufwärts erzielen. Denn dann bewegt sich das Schwungbein etwa 0.0375 m je Zeitschritt, etwa 46 % seiner Breite am Sprunggelenk.

Aufgrund seiner geringen Breite wären für das Sprunggelenk maximale Wege von etwa  $s = 0.03 \text{ m}$  je Zeitschritt wünschenswert. Die erforderliche Bildfrequenz  $f_{max}$  kann bei einer gegebenen mittleren Probandengeschwindigkeit  $v$  leicht mit folgender Gleichung berechnet werden.

$$f_{max} = \frac{v_{max}}{s} = \frac{2.7 \cdot v}{0.03 \text{ m}} = \frac{90}{m} \cdot v \quad (2.4)$$

Für hochdynamische Laufbewegungen, wie sie in der Sportmedizin bei der Beobachtung eines Sprinters (10 m/s) auftreten, müssten demzufolge mindestens 900 Bilder je Sekunde aufgezeichnet werden! Um Bewegungsunschärfe bei derartigen Geschwindigkeiten zu reduzieren, müssen jedoch auch die Belichtungszeiten entsprechend kurz sein. Für kurze Belichtungszeiten wird jedoch eine erhebliche Lichtmenge benötigt, sodass eine entsprechend leistungsfähige Beleuchtung vorhanden sein muss. Nicht zuletzt besteht die technische Herausforderung auch darin, alle beteiligten Kameras entsprechend zu synchronisieren. Aus den genannten Gründen versucht das hier gezeigte Verfahren mit deutlich geringeren Bildfrequenzen auszukommen. Es benötigt deshalb aber in jedem Zeitschritt entsprechend gute Näherungswerte, damit eine gute Korrespondenz zwischen den Kantenpunkten des Modells und denen des Bildes hergestellt werden kann.

### 2.2. Ansätze zur Bildfolgenauswertung

Ganz allgemein geht es bei der Bildfolgenauswertung zur Bewegungsanalyse darum, aus den großen Datenmengen möglichst relevante Informationen zu gewinnen. Häufig sind das wenige Parameter, die sich für verschiedene Anwendungen aufbereiten lassen. Dabei wird oft für jeden Zeitpunkt oder für eine Zeitspanne ein Satz von Parametern berechnet, der die Bewegung repräsentiert. Meist sind diese Parameter einfach Zahlen, die sich zu einem Vektor zusammenfassen lassen.

Die Haltung des Probanden zu einem bestimmten Zeitpunkt  $t$  wird dann durch diesen Vektor beschrieben. Er wird in dieser Arbeit als *Zustandsvektor*  $\mathbf{x}_t$  bezeichnet. Im Zustandsvektor werden zum Beispiel Größen, wie Weg, Gelenkwinkel oder Formparameter repräsentiert. Auf eine Repräsentation der Ableitungen dieser Größen, wie Geschwindigkeiten oder Beschleunigungen wird in dieser Arbeit verzichtet, da mit Kameras lediglich Momentaufnahmen in jedem Zeitschritt gemacht werden. Mit einem Messsystem, das nur aus Kameras besteht, lassen sich nur reine Haltungen, also ein Satz statischer Parameter für jeden Zeitschritt direkt erfassen. In einem gewissen Rahmen lässt sich diese Lücke für bestimmte Anwendungen durch numerische Differenzierung der gemessenen Folge von Zustandsvektoren schließen.

#### 2.2.1. Ansätze ohne explizites 3D-Modell

Eine mögliche Strategie zur automatischen Bewegungsanalyse wäre die reine bottom-up Erkennung möglichst vieler Einzelkomponenten des sich bewegenden Probanden in den Kamerabildern. Anschließend könnten diese Einzelkomponenten zusammengesetzt und aus ihrer Lage zueinander der dahinter liegende Zustandsvektor berechnet werden. Dazu können verschiedene Informationen aus den aufgenommenen Bildern verwendet werden. Allerdings erfolgt die Bildauswertung für jeden Kamerakanal unabhängig voneinander. Das Problem dieses Ansatzes dürfte generell die Integration der Daten der einzelnen Kamerakanäle sein.

Die Bilder lassen sich in Konturen, Farben, Bewegungsmuster usw. zerlegen. Aus diesen Informationen müssten dann visuell konstante Komponenten zum Beispiel Fuß, Unterschenkel, Oberschenkel oder Hüfte zu größeren Einheiten gruppiert werden. Entlang der Zeitachse könnten dann die typischen Bewegungseigenschaften und zusammenhängende Objekte ermittelt werden. Mit mehreren Kameras ließen sich theoretisch auch die 3D-Positionen und Orientierungen im Raum schätzen. Aus der Lage zueinander ergeben sich dann die Gelenkwinkel. Ein explizites 3D-Modell wird nicht benötigt, lediglich die Topologie des Körpers. Standardkameras bilden bei einer Ganzkörperansicht die einzelnen Körperteile nur mit geringer Auflösung ab. Meist sind zu wenig Informationen vorhanden, um eine ausreichende Erkennungssicherheit zu gewährleisten. Hinzu kommen Variationen der Erscheinung durch Perspektive, Deformationen und Beleuchtungsänderungen. Bei den Beinen handelt es sich um relativ schlanke Körperteile mit wenig Eigentextur. Die wenigen messbaren Bildelemente (zum Beispiel Kanten) liegen



dann an den Rändern der Körperteile. Damit unterliegen sie wiederum der Interaktion mit dem jeweiligen Hintergrund des Körperteils. Im einfachen Fall ist dies ein statischer Hintergrund, aber im ungünstigen Fall können dort auch bewegende Körperteile sein. Weiterhin entsteht hier das Problem der (Selbst-)Verdeckungen.

Ein System zur Erkennung von Körperteilen wurde in [MPN98] vorgestellt. Die Erkennungssicherheit für die Beine liegt bei 70 %. Eine echtzeitfähige Erkennungsstrategie wurde von [BH94] sowie [Bau95] vorgestellt und von [Sie03] zum Tracking mehrerer Personen erweitert. Ein Bewegungs-Detektor erkennt Fußgänger, deren Silhouetten über der Zeit verfolgt werden. Durch Hauptkomponentenanalyse von Trainingsdatensätzen wurden die wichtigsten Variationen der Silhouetten ermittelt. Zur Repräsentation der typischen Eigenschaften von Silhouetten reichen ca. 18 Parameter. So kann die Dimension des Zustandsvektors auf diese Zahl begrenzt werden. Mit diesen Parametern können die wesentlichen Variationen der 2D-Silhouette dargestellt werden. Für die Vorhersage (Prediktion) des jeweils nächsten Bewegungszustandes wird ein Kalmanfilter verwendet.

Entlang der Normalenvektoren<sup>1</sup> der predizierten Silhouette wird nun nach Kantenpunkten gesucht. Punkte mit hohem Kontrast innerhalb eines Suchbereiches ( $\pm 3$  Pixel) dienen als Input für den Kalmanfilter. Das System ist echtzeitfähig und damit sehr geeignet für automatische Überwachungseinrichtungen. Die Ergebnisse der Analyse sind leider sehr an die visuelle Erscheinung und wenig an die physikalisch ursächlichen Gelenkwinkel geknüpft. Es müsste also quasi eine (Rück-)Übersetzung zwischen der Repräsentation der visuellen Erscheinung und den Gelenkwinkeln erfolgen.

In [OG99] wird dies realisiert. Durch Anlernen von Bewegungszuständen, gefolgt von einer hierarchischen Hauptachsentransformation, werden die visuellen Zustandsvektoren in physikalisch begründete übersetzt, sodass sie für die Steuerung eines Skelettmodells verwendet werden können. Unglücklicherweise ist die Abbildung nichtlinear und mehrdeutig. Durch die hohe Dimension des Zustandsvektors ist der Zustandsraum auch noch sehr groß. Der Ansatz unterstützt zwar mehrere Kameras, aber nur relativ wenige Freiheitsgrade des Oberkörpers.

Um die Nichtlinearität zwischen visuellem und physikalischen Zustandsraum zu überbrücken, bieten sich Lernverfahren an. Es muss eine ausreichend große Datenbank bereitgestellt werden, die den Zustandsraum hinreichend repräsentiert. Eine neue Indexmethode (Locality-Sensitive Hashing - LSH [GIM99]) zur Adressierung der Datensätze wurde in [SVD03] weiterentwickelt und zur Haltungsanalyse eingesetzt.

Silhouetten aufeinander folgender Zeitschritte werden auch in [Bra99b] als Input für die Erkennung des dahinterliegenden physikalischen Zustandes verwendet. Dafür müssen in einer Trainingsphase Zustandsnachbarschaften aus vorgegebenen Sequenzen ermittelt werden. Der Codierungsaufwand wird durch Entropieminimierung [Bra99a] klein gehalten. Die Auswertung erfolgt in Echtzeit durch einen Viterbi-Algorithmus [For73]. Mehrdeutigkeiten, die bei Verwendung von Silhouetten immer auftreten, werden dadurch im zeitlichen Kontext aufgelöst. Die Linearkombination der Zustandsnachbarn ermöglicht

---

<sup>1</sup>genauer: Entlang einer optimierten Suchrichtung (Mahalanobis optimal search direction [Bau96])

## 2. Grundlagen

die Interpolation von vorher nicht antrainierten Zwischenzuständen und damit sanfte Bewegungsabläufe. Es können nur antrainierte Folgen und Kombinationen daraus erkannt werden. Demonstriert wird dies an einigen typische Bewegungsfolgen. Das Verfahren arbeitet monokular, kann aber auf Multikerasysteme erweitert werden. Als Ergebnis liefert das Online-System Zustandsvektoren mit physikalischer Bedeutung.

Das System von [RS98] erlaubt die Verfolgung von mehreren Personen. Es basiert auf der Extraktion von Silhouetten durch Vorder-Hintergrundtrennung und kann die Verdeckungen aneinander vorbeilaufender Personen überbrücken. Eine Messung der 3D-Entfernung ist möglich, allerdings werden keine Details der Bewegung wie zum Beispiel Gelenkwinkel erfasst.

In [BY95] wurden deformierbare, zunächst voneinander unabhängige 2D-Patches zur Modellierung der Bewegung von Gesichtspartien eingesetzt. Der Versatz zwischen Grauwerten von Patch und Bild wird mittels des Optischen Flusses minimiert, indem die Lageparameter der Patches geändert werden. Damit kann die Bewegung zum Beispiel von Augenbrauen, Augenlidern und Mund bestimmt werden. In [JBY96] wurden diese Patches federnd miteinander verbunden, sodass ihre Freiheitsgrade nicht mehr unabhängig voneinander sind, was der Robustheit zugute kommt. Er zeigt mit 'Cardboard People', dass sich dieses Prinzip auch zur Analyse von Beinbewegungen eines Probanden anwenden lässt. Dies wird unter relativ konstanten Beleuchtungsbedingungen eines Probanden auf einem Laufband demonstriert. Es werden voraussichtlich gute Kontraste auf den Oberflächen der Körperteile benötigt, da das Verfahren letztlich auf dem Optischen Fluss basiert. Wichtige Informationen wie die Grauwertgradienten an den Körperteilgrenzen müssen wahrscheinlich ignoriert werden, da sie sich über der Zeit durch Interaktion mit dem jeweiligen Hintergrund oder anderen Körperteilen ändern. Der Optische Fluss basiert aber auf dem grundlegenden Ansatz, dass die Intensitätsfunktion im Bild über der Zeit konstant bleibt.

[BM98] verwendet bereits ein Objektkoordinatensystem zur Beschreibung der Bewegung des Torsos ( $3 \times$  Rotation,  $2 \times$  Translation,  $1 \times$  Skalierung) und kinematische Ketten als Bewegungsmodell der Extremitäten (je Körperteil einen Gelenkwinkel). Ein einfaches Kameramodell (Parallelprojektion) und die Beschreibung der Bewegung relativ zum vorhergehenden Zeitschritt erlaubt ein unkompliziertes Aufstellen des Gleichungssystems. Es basiert im Grundprinzip wieder auf dem Optischen Fluss, diesmal können jedoch kinematische Randbedingungen, wie Gelenkfreiheitsgrade berücksichtigt werden. Als Modell für die Körperteile verwendet er dreidimensionale Ellipsoiden (Blobs). Die Lösung einer quadratischen Gleichung erlaubt dabei die Berechnung des Schnittpunktes zwischen Sehstrahl und Blob. Form und Selbstverdeckungen der Körperteile werden durch sogenannte Support Maps (Layer [WA94]) berücksichtigt. Diese beschreiben, mit welcher Wahrscheinlichkeit ein Pixel zu welchem Körperteil oder dem Hintergrund gehört. Dieses Modell wurde von [JB93b] sowie [JB93a] vorgestellt und wird 'EM-based layered representation' genannt. EM<sup>2</sup> steht hierbei für Expectation Maximization [DLR77]. Da-

---

<sup>2</sup>Zu diesem Algorithmus findet man viel im Internet, kurz und verständlich sind die Gleichungen in [SW98] zusammengefasst.

mit werden die Zugehörigkeiten eines Pixels zu jedem Layer geschätzt. Diese Schätzung sowohl der Freiheitsgrade als auch der Pixelzugehörigkeiten erfolgt abwechselnd iterativ.

[HLF00] erweiterte das Verfahren und rekonstruierte die 3D-Bewegungen des ganzen Körpers aus monokularen Videosequenzen und bestimmt die Grauwertstruktur der Patches nicht nur in einem ausgewählten Zeitschritt, sondern durch gewichtete Mittelwertbildung verschiedener vorausgehender Zeitschritte.

Bei den meisten Ansätzen ohne ein explizites 3D-Modell ist es nur schwer möglich, bekanntes a priori Wissen in die Bildanalyse einfließen zu lassen. Da aber keine Marker verwendet werden dürfen, gehen wesentliche Informationen verloren, die gerade durch a priori Wissen ersetzt werden müssen.

### 2.2.2. Ansätze mit expliziten 3D-Modellen

Im Gegensatz zu den Algorithmen des vorherigen Abschnitts erlauben Ansätze mit expliziten 3D-Modellen eine gute Repräsentation von vorhandenem Wissen über die Körpergeometrie. Diese Ansätze gehen davon aus, dass ein Modell, also eine Beschreibung des zu verfolgenden Objektes, im 3D-Raum vorhanden ist. Aus der Computeranimation sind zahlreiche Methoden der Modellierung bekannt. Ausgefeilte Kameramodelle mit Renderingmethoden, die in Hardware realisiert sind, erlauben bereits die Synthese photorealistischer Bilder mit erheblicher Szenenkomplexität in Echtzeit. Vor allem die Mensch-Modellierung ist ein gut untersuchter Bereich [Rat98]. In Computerspielen werden sie als sogenannte Character zur Animation von Spielfiguren eingesetzt. Zur Ermittlung eines Körpermodells und der Erscheinung des Probanden können kommerzielle Body-Scanner eingesetzt werden [Ste04]. Ein solches Körpermodell diente in [SBB00] als Grundlage zur Bewegungsanalyse.

Die verwendeten Modelle bestehen meist aus einem Skelett und einem Oberflächenmodell. Das Skelett dient der Beschreibung der Bewegungen der Komponenten des Modells. Es übersetzt die Komponenten des Zustandsvektors in Translationen und Rotationen des Oberflächenmodells. Das Oberflächenmodell dient der Visualisierung. Es enthält alles, was zur Beschreibung der visuellen Eigenschaften und zur Objektverfolgung benötigt wird.

Zur Anwendung der gut ausgearbeiteten Modellierungskonzepte in der Bildverarbeitung wird meist eine Form des Analyse-durch-Synthese Konzeptes eingesetzt [AHS89], [LBK90], [Bus89] und [Koc93]. Abbildung 2.3 zeigt eine Übersicht. Es wird eine Struktur ähnlich einem Regelkreis eingesetzt, um die Abweichungen zwischen synthetischem Bild und realem Bild iterativ zu minimieren. Der Zustandsvektor des Modells muss so lange geändert werden, bis es der abgebildeten Person im Kamerabild entspricht. Die meisten Verfahren maximieren eine Gütefunktion oder minimieren den Fehler zwischen synthetischem Bild und realem Kamerabild.

Ursprünglich wurde das Verfahren für die Bildcodierung entwickelt. In dieser Anwen-

## 2. Grundlagen

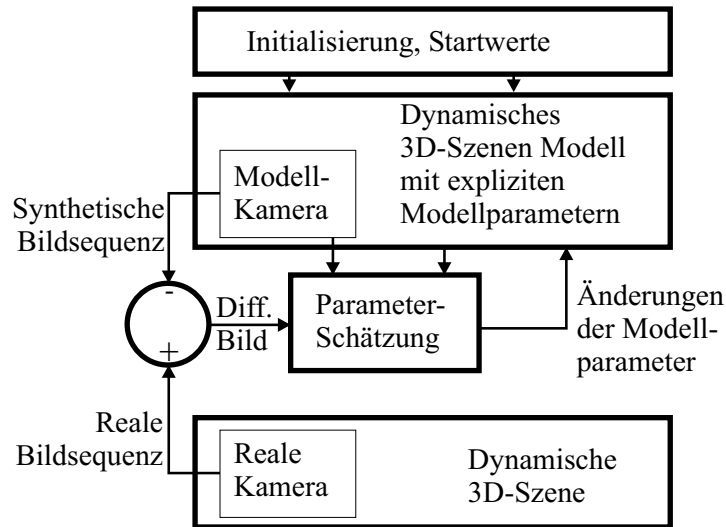


Abbildung 2.3.: Analyse-durch-Synthese Konzept (vereinfacht)

Es geht es in erster Linie um eine exakte Synthese der Grauwerte. Als Fehlerkriterium kommt dann die pixelweise Differenz der reinen Grauwerte in Frage, die mit einem Ansatz ähnlich dem Optischen Fluss minimiert wird. In der Praxis werden lediglich die Kantenpunkte verwendet, die ausreichenden Kontrast bieten. Da das Verfahren differenziell arbeitet, hat dies jedoch gerade bei hohen Ortsfrequenzen den Nachteil, dass verglichen mit der Ortsfrequenz nur relativ langsame Bewegungen mit kleinen Verschiebungen im Bild auftreten dürfen. Für die Bildcodierung (zum Beispiel Netmeeting) ist dieser Ansatz interessant, weil sich häufig die Haltung von sprechenden Personen in Bildsequenzen von Zeitschritt zu Zeitschritt nur wenig ändert und deshalb bereits gute Näherungen des Zustandsvektors aus dem vorhergehenden Zeitschritt vorhanden sind. In den ursprünglichen Veröffentlichungen ging es zum Beispiel darum, die kleinen Bewegungen eines sitzenden Sprechers mit gutem Grauwert-Kontrast ohne Verdeckungen zu modellieren. Dies ist ein verhältnismäßig einfacher Fall, verglichen mit den erheblichen Geschwindigkeiten und Beschleunigungen, die im Beinbereich bereits bei einem langsamen Gang auftreten (siehe Abschnitt 2.1.2). Zusätzliche Schwierigkeiten stellen Selbstverdeckungen der Beine und der geringe Kontrast der Haut in diesen Regionen dar.

Eine exakte Synthese der Grauwerte ist für die Bewegungsanalyse jedoch zweitrangig. Eigentlich sind hier nur die Verschiebungen zwischen Modell und Bild zu berechnen. Später wird beschrieben, wie dies genau geschieht und welche Abweichungen zwischen Modell und Bild letztlich minimiert werden.

Wesentlich größere Abweichungen lassen sich durch Suchverfahren mit Vorwärts (Forward) Kinematik (VK) verringern (siehe Abbildung 2.4). Mit wahrscheinlichkeitsbasierten Ansätzen wie Genetischen Algorithmen oder Partikelfiltermethoden lassen sich die Zustandsvektoren gewinnen. Zum Beispiel stammt der CONDENSATION Algorithmus [BI98] aus den aktiven Konturmodellen und verfolgt mehrere Hypothesen der Bewegung

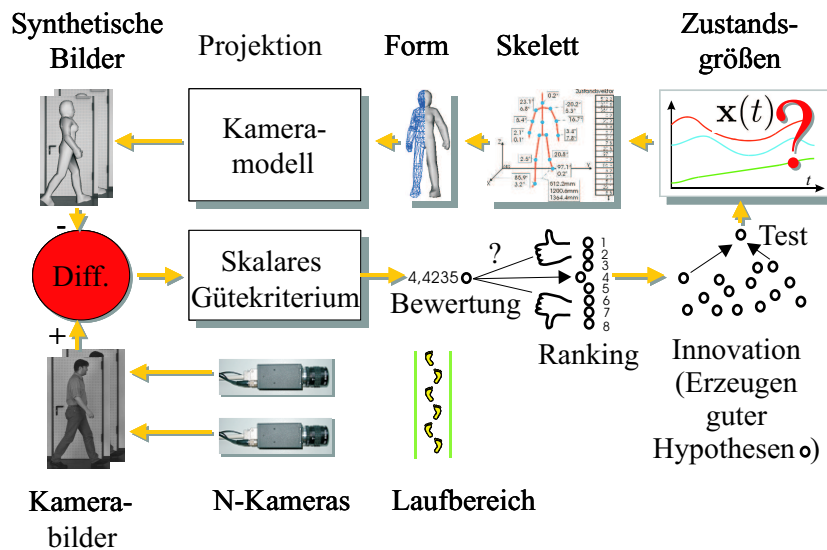


Abbildung 2.4.: Analyse-durch-Synthese Konzept: Zur Bestimmung der unbekannt Parameter wird eines der zahlreichen Suchverfahren eingesetzt. Dadurch kann unter Umständen der Schritt einer expliziten Korrespondenzfindung eingespart werden. Allerdings ist der Suchraum bei einem Mensch-Modell hochdimensional und damit sehr groß.

parallel. Nach einigen Zeitschritten erfolgt eine Selektion. Es wird dann die Hypothese mit der größten Wahrscheinlichkeit ausgewählt [DDR01]. Dadurch gewinnen die Verfahren an Robustheit und die Möglichkeit, visuelle Mehrdeutigkeiten aufzulösen. Allerdings muss man bedenken, dass der Zustandsraum bei einem menschlichen Skelett hochdimensional ist und damit der Suchraum bereits für einfache Modelle schnell sehr groß wird. Für eine vernünftige Modellierung der Verteilungsfunktion wird dann aber eine sehr große Anzahl an Partikeln oder Individuen (bei Genetischen Algorithmen) benötigt. Jedes Individuum repräsentiert dann einen Zustandsvektor, der jedes Mal die gesamte Synthesekette durchlaufen muss. Deshalb erlauben diese Verfahren meist keine Echtzeit, ihre Implementierung ist jedoch häufig einfacher als die im Folgenden beschriebenen Verfahren.

Besser geeignet erscheinen Algorithmen der inversen Kinematik und Trajektorienplanung. Beides sind gut studierte Bereiche der Robotik und mittlerweile auch der Computervisualistik. [TGB00] stellt die verschiedenen Ansätze für echtzeitfähige Systeme vor. Für die Bildauswertung müssen diese Verfahren etwas angepasst werden. Voraussetzung dafür ist die korrekte Korrespondenz zwischen Körperpartien des Modells und denen des Probanden im Bild. Verschiebungen werden dadurch messbar und lassen sich verwenden, um den Zustandsvektor gezielt zu ändern, Abbildung 2.5 zeigt dies. Eine explizite Korrespondenzfindung war in den beiden vorhergehenden Ansätzen nicht notwendig. Beim ersten Ansatz (Abbildung 2.3) wurde gefordert, dass die Lageabweichungen zwischen Mo-

## 2. Grundlagen

dell und Bild sehr klein sein müssen und die Korrespondenz dadurch bereits existiert. Beim zweiten Ansatz nach Abbildung 2.4 wurden alle möglichen Korrespondenzen durch das Suchverfahren quasi implizit gebildet.

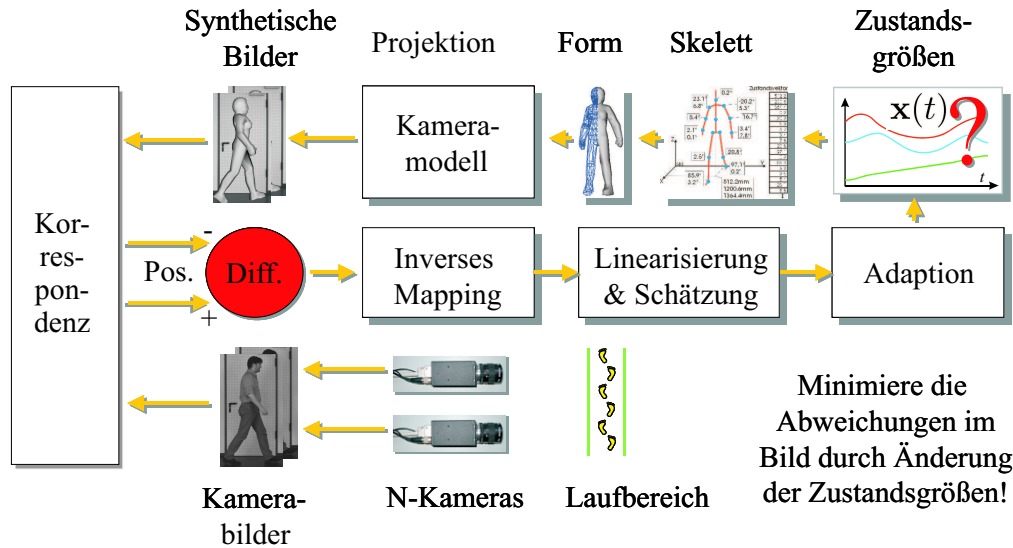


Abbildung 2.5.: Analyse-durch-Synthese Konzept: Zur Bestimmung der unbekannter Parameter wird ein iterativer Prozess ähnlich eines Regelkreises verwendet.

Die Suche nach einzelnen Körperteilen im Bild wird durch gute Näherungswerte für die Haltungparameter des Modells erleichtert. Im synthetischen Bild ist die Lage aller Körperpartien des Modells bekannt und es wird davon ausgegangen, dass sich im realen Bild die Körperteile des Probanden in ihrer Nähe befinden müssen. Oft wird als Näherungswert der Zustandsvektor des vorhergehenden Zeitschrittes verwendet oder eine Vorhersage daraus berechnet. Für das erste Bild der Sequenz stellt sich aber immer die Frage, wie die Näherungswerte des Zustandsvektors und der äußeren Erscheinung ermittelt werden können. Dies wird als Bootstrappingproblem [GD96] bezeichnet und von den meisten bestehenden Systemen einfach durch manuelle Eingaben überbrückt. Das Problem stellt sich ebenfalls innerhalb einer Bildsequenz, wenn aus verschiedenen Gründen das kontinuierliche Tracking unterbrochen wurde. Zur automatischen Lösung des Bootstrappingproblems sei auf die direkten bildbasierten Ansätze verwiesen [MG01]. Der Bereitstellung von guten Näherungswerten für den Spezialfall Ganganalyse wurde in dieser Arbeit das Kapitel 9 gewidmet.

Die Modellierung der einzelnen Elemente der Synthesekette ist aus der Robotik, der Photogrammetrie und Computergrafik bekannt. Damit lässt sich viel a priori Wissen für die Bildauswertung einbringen. Theoretisch lässt sich das Modell soweit verfeinern, dass durch Berücksichtigung von Trägheitseigenschaften des Probanden eine dynamische Modellierung mittels Mehrkörperphysik denkbar erscheint. Dies erfordert unter den gegebenen Umständen jedoch einen erheblichen Aufwand, der für die reine Bildauswertung

wenig Nutzen bringt.

Natürlich müssen für den Ansatz jeweils passende Modelle bereit gestellt werden. Will man statt eines menschlichen Probanden die Bewegung von Tieren analysieren, muss ein Körpermodell mit entsprechender Geometrie vorhanden sein. Die Algorithmen selbst lassen sich oft für andere Problemstellungen verwenden. Für die konkrete Zielsetzung der Ganganalyse erscheint das Körpermodell eines Menschen ausreichend, sofern sich einige wichtige Parameter anpassen lassen. Dadurch können zum Beispiel die anthropometrischen Besonderheiten zwischen männlichen und weiblichen Probanden berücksichtigt werden. Im einfachsten Fall lässt sich zumindest Körpergröße und Körpermasse ändern.

Der modellbasierte Ansatz bietet weitere Vorteile. Da jeder einzelne Schritt der Synthesekette bekannt ist, können Selbstverdeckungen, Beleuchtungs- und Formänderungen vorhergesagt und beachtet werden. Weiterhin lassen sich physikalische Randbedingungen wie Massenträgheit und begrenzte Beschleunigungen formulieren.

### 2.2.3. Der verwendete Ansatz

Die bereits genannten Gründe sprechen für einen modellbasierten Ansatz. Deshalb soll er auch in dieser Arbeit verfolgt werden. Nach einer Einführung in die mathematischen Grundlagen wird der Pseudocode eines Algorithmus angegeben und Überlegungen zur Genauigkeit angestellt. Die hier noch nicht genannten Probleme des Ansatzes werden im späteren Verlauf der Arbeit gezeigt und genauer analysiert. Gefundene Lösungen werden dargestellt.

### Das mathematische Rüstzeug

In dieser Arbeit werden Vektoren und Matrizen fettgedruckt dargestellt. Matrizen werden durch große und Vektoren durch kleinen Buchstaben gekennzeichnet. Es wird davon ausgegangen, dass sich alle Bewegungsgrößen des Systems in einem Zustandsvektor  $\mathbf{x}$  darstellen lassen. Hängt der Zustandsvektor von der Zeit ab, wird er  $\mathbf{x}(t)$  geschrieben. Punkte im Weltkoordinatensystem erhalten normalerweise das Formelzeichen  $\mathbf{w}$ . Ein tiefgestellter Index, zum Beispiel  $\mathbf{w}_i$  beschreibt die Punktkoordinaten genauer.  $\mathbf{w}_i = [x_i, y_i, z_i]^T$  wird geschrieben, wenn die Vektor-Komponenten des Punktes in den Koordinatenrichtungen explizit benötigt werden. Die für das Verständnis des Ansatzes zunächst unwichtigen Gewichtungsfaktoren zur Schätzung der Genauigkeit werden erst einmal weggelassen, im Anhang C sind diese aber berücksichtigt.

Entsprechend Abbildung 2.5 lässt sich eine abstrakte Formulierung für die Bildsynthese aufstellen. Jedes Kästchen wird hier als Funktion dargestellt. Das Modell des Skeletts eines Probanden wird als eine nichtlineare Funktion  $\mathbf{s}(\mathbf{x}, \mathbf{S})$  formuliert, die aus dem Zustandsvektor die Bewegung der Gelenkpunkte oder Körperteile berechnet. In Kapitel 3 wird das Skelettmodell näher beschrieben. Modellwissen über die genaue Länge der Elemente des Skeletts sei in der konstanten Matrix  $\mathbf{S}$  repräsentiert. Der Zusammenhang

## 2. Grundlagen

zwischen Zustandsvektor und Lage des Skeletts im 3D-Raum sei durch eine nichtlineare Funktion  $\mathbf{s}(\dots)$  dargestellt.

Das Skelett wird verwendet, um eine Formbeschreibung  $\mathbf{f}(\mathbf{s}(\mathbf{x}, \mathbf{S}), \mathbf{F})$  des Probanden entsprechend dem vorgegebenen Zustandsvektor zu deformieren. Möglichkeiten der Formbeschreibung werden in Kapitel 4 vorgestellt. Vorwissen über die genaue Form des Probanden sei zunächst in der konstanten Matrix  $\mathbf{F}$  angenommen. Für jeden beliebigen Zustandsvektor lassen sich nun die 3D-Koordinaten eines jeden Punktes der Modelloberfläche ermitteln.

$$\mathbf{w}(\mathbf{x}) = \mathbf{f}(\mathbf{s}(\mathbf{x}, \mathbf{S}), \mathbf{F}) \quad (2.5)$$

Mit einem Kameramodell  $\mathbf{p}(\mathbf{x}) = \mathbf{k}(\dots, \mathbf{K})$  lassen sich diese Punkte in jedes Kamerabild projizieren. Dies wird in Kapitel 5 näher erläutert. Weil die Kameras als statisch angenommen werden, ist die Matrix  $\mathbf{K}$  konstant, sie enthält alle Parameter, die zur Beschreibung einer Kamera benötigt werden (zum Beispiel Lage, Orientierung, Brennweite, Hauptpunkt usw.).

$$\mathbf{p}(\mathbf{x}) = \mathbf{k}(\mathbf{f}(\mathbf{s}(\mathbf{x}, \mathbf{S}), \mathbf{F}), \mathbf{K}) \quad (2.6)$$

Da einige/alle innere Funktionen nichtlinear sind, ist auch  $\mathbf{p}(\mathbf{x})$  nichtlinear.

In jedem Kamerabild werden Zielpunkte  $\mathbf{z}_i$  in einem Kamerabild vorgegeben. Der Zustandsvektor  $\mathbf{x}$  wird so bestimmt, dass für jeden Zielpunkt die folgende Bedingung erfüllt sein soll:

$$\mathbf{p}(\mathbf{x}, i) = \mathbf{z}_i \quad (2.7)$$

Durch Umformen  $\mathbf{z}_i - \mathbf{p}(\mathbf{x}, i) = \mathbf{0}$  erhält man ein Gleichungssystem der Form  $\mathbf{f}(\mathbf{x}) = \mathbf{0}$ . Solche Gleichungen lassen sich nach [BSG<sup>+</sup>96] mit der Methode von *Newton-Kantorowitsch* lösen. Dabei erfolgt eine Taylorreihenentwicklung um einen vorzugebenden Näherungswert  $\mathbf{x}_k$  mit ( $k=0,1,\dots$ ):

$$\mathbf{f}(\mathbf{x}) = \mathbf{f}(\mathbf{x}_k) + \frac{\partial \mathbf{f}(\mathbf{x}_k)}{\partial \mathbf{x}_k} (\mathbf{x} - \mathbf{x}_k) + \hat{\mathbf{r}}(\mathbf{x}) \quad (2.8)$$

Und es wird iterativ ein besserer Vektor  $\mathbf{x}_{k+1}$  berechnet:

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{\partial \mathbf{f}(\mathbf{x}_k)}{\partial \mathbf{x}_k}^{-1} \mathbf{f}(\mathbf{x}_k) \quad (2.9)$$

Aus [BSG<sup>+</sup>96] stammt das folgende Zitat:

Die Methode von Newton-Kantorowitsch konvergiert sehr rasch für Startwerte, die bereits hinreichend nahe an der Lösung liegen. Die Konvergenzgeschwindigkeit ist dann mindestens quadratisch. Für schlechte Startwerte kann jedoch dieses Verfahren völlig versagen.



## 2.2. Ansätze zur Bildfolgenauswertung

Aus diesem Ansatz hat sich in der Photogrammetrie der im Folgenden beschriebene Lösungsansatz zur Bündelblockausgleichung etabliert, bei dem die Lösung meist durch das *Gauss-Newton* und manchmal auch durch das *Levenberg-Marquardt-Verfahren* ermittelt wird. Der Hauptvorteil des Gauss-Newton-Verfahrens ist die Abschätzung der Genauigkeit der gefundenen Parameter (siehe Anhang C.9). Weiterhin besitzt es eine rasche Konvergenz. Zur Steigerung der Genauigkeit wird versucht, möglichst wesentlich mehr Zielpunkte vorzugeben als es unbekannte Parameter im Vektor  $\mathbf{x}$  gibt. Es lässt sich eine klassische Fehlerfunktion nach der L2-Norm (Methode der kleinsten Fehlerquadrate) definieren, die es dann zu minimieren gilt.

$$e(\mathbf{x}) = \sum_{i=1}^M \|\mathbf{z}_i - \mathbf{p}(\mathbf{x}, i)\|^2 \rightarrow \min \quad (2.10)$$

Die Nichtlinearität von  $\mathbf{p}(\mathbf{x}, i)$  bereitet noch Probleme bei der Suche nach dem Minimum.

Wird ein guter Näherungswert des Zustandsvektors  $\mathbf{x}_k$  vorgegeben, lässt sich die Funktion  $\mathbf{p}(\mathbf{x}, i)$  mittels einer Taylorreihenentwicklung linearisieren:

$$\mathbf{p}(\mathbf{x}, i) = \mathbf{p}(\mathbf{x}_k, i) + \frac{\partial \mathbf{p}(\mathbf{x}_k, i)}{\partial \mathbf{x}_k} \cdot (\mathbf{x} - \mathbf{x}_k) + \mathbf{r}(\mathbf{x}, i) \quad (2.11)$$

Dabei ist  $\mathbf{r}(\mathbf{x}, i)$  das *Restglied*, in dem die Terme höherer Ordnung zusammengefasst werden. Voraussetzung für die Reihenentwicklung ist die Möglichkeit der Berechnung der ersten Ableitung  $\frac{\partial \mathbf{p}(\mathbf{x}, i)}{\partial \mathbf{x}}$  am Näherungswert  $\mathbf{x}_k$ . Die Ableitung ist eine Matrix mit so vielen Spalten, wie es Elemente im Zustandsvektor  $\mathbf{x}$  gibt. Für einen Zielpunkt im Bild besitzt diese Matrix zwei Zeilen. Diese Matrix wird *Jacobische Matrix* genannt:

$$\mathbf{J}_i(\mathbf{x}_k) = \frac{\partial \mathbf{p}(\mathbf{x}_k, i)}{\partial \mathbf{x}_k} \quad (2.12)$$

Mit:

$$\Delta \mathbf{x} = \mathbf{x} - \mathbf{x}_k \quad (2.13)$$

lässt sich die Taylorreihe wie folgt umschreiben:

$$\tilde{\mathbf{p}}(\Delta \mathbf{x}, i) = \mathbf{p}(\mathbf{x}_k, i) + \mathbf{J}_i(\mathbf{x}_k) \cdot \Delta \mathbf{x} + \mathbf{r}(\mathbf{x}_k + \Delta \mathbf{x}, i) \quad (2.14)$$

Wird dies für jeden Punkt  $i$  getan, entsteht durch Untereinanderschreiben ein lineares Gleichungssystem, bei dem der Vektor  $\mathbf{z}$  die Zielpunktkoordinaten  $\mathbf{z}_i$  und der Vektor  $\mathbf{p}(\mathbf{x}_k)$  die Modellpunktkoordinaten  $\mathbf{p}(\mathbf{x}_k, i)$  im Bild bei einem gegebenen Näherungswert  $\mathbf{x}_k$  enthalten. Dabei ist  $\mathbf{z}$  als *Soll-Wert* und  $\mathbf{p}(\mathbf{x}_k)$  als *Ist-Wert* der Struktur (Regelkreis) nach Abbildung 2.5 aufzufassen.

Störungen, zum Beispiel durch Rauschen, sollen im Vektor  $\mathbf{n}$  repräsentiert werden.

$$\mathbf{p}(\mathbf{x}_k) + \mathbf{J}(\mathbf{x}_k) \cdot \Delta \mathbf{x} + \mathbf{r}(\dots) = \mathbf{z} - \mathbf{n} \quad (2.15)$$

$$\mathbf{J}(\mathbf{x}_k) \cdot \Delta \mathbf{x} = \mathbf{z} - \mathbf{p}(\mathbf{x}_k) - \mathbf{n} - \mathbf{r}(\dots) \quad (2.16)$$

## 2. Grundlagen

Die Abweichung zwischen Soll- und Ist-Wert des Regelkreises wird im Vektor  $\Delta \mathbf{y} = \mathbf{z} - \mathbf{p}(\mathbf{x}_k)$  repräsentiert. Das Restglied  $\mathbf{r}(\dots)$  und das Rauschen  $\mathbf{n}$  werden als unbekannte Größen zum Vektor der *Verbesserungen*  $\mathbf{v} = -\mathbf{n} - \mathbf{r}(\dots)$  zusammengefasst.

$$\mathbf{J}(\mathbf{x}_k) \cdot \Delta \mathbf{x} = \Delta \mathbf{y} + \mathbf{v} \quad (2.17)$$

Durch Ignorieren der Verbesserung erhält man ein sehr übersichtliches Gleichungssystem:

$$\mathbf{J}(\mathbf{x}_k) \cdot \Delta \mathbf{x} \approx \Delta \mathbf{y} \quad (2.18)$$

$$\mathbf{J}(\mathbf{x}_k) \cdot \Delta \mathbf{x} \approx \mathbf{z} - \mathbf{p}(\mathbf{x}_k) \quad (2.19)$$

Dabei ist  $\Delta \mathbf{x}$  die Änderung des bisherigen Zustandsvektors  $\mathbf{x}_k$  und ergibt sich durch Lösung von Gl.(2.18). Zur Berechnung des nächsten (besseren) Zustandsvektors  $\mathbf{x}_{k+1}$  wird Gl.(2.13) umgeformt:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta \mathbf{x} \quad (2.20)$$

Zwei Gründe verbieten in Gl.(2.18) und Gl.(2.19) die Verwendung des '=' Zeichens. Erstens wurde das Restglied  $\mathbf{r}(\dots)$  weggelassen und zweitens ist das Gleichungssystem (2.18) je nach Punktzahl überbestimmt. Dann lässt sich Gl.(2.15) selbst mit Restglied nicht exakt aufrecht erhalten, dies wurde zuvor durch den Vektor  $\mathbf{n}$  berücksichtigt. Beide Terme wurden zu einem Vektor der *Verbesserungen*  $\mathbf{v}$  zusammengefasst und es wird gefordert, dass das Quadrat der Verbesserungen  $\mathbf{v}^T \mathbf{v}$  minimal wird.

$$\mathbf{J}(\mathbf{x}_k) \cdot \Delta \mathbf{x} - \Delta \mathbf{y} = \mathbf{v} \quad (2.21)$$

$$\mathbf{v}^T \mathbf{v} \rightarrow \min \quad (2.22)$$

Dabei entspricht  $\mathbf{v}^T \mathbf{v} \rightarrow \min$  der Gl.(2.10), jedoch in linearisierter Form. Aus [Lin99] stammen die folgenden Gleichungen<sup>3</sup>, von denen nur die letzte wirklich verwendet wird:

$$\mathbf{v}^T \mathbf{v} = (\Delta \mathbf{x}^T \mathbf{J}^T - \Delta \mathbf{y}^T)(\mathbf{J} \Delta \mathbf{x} - \Delta \mathbf{y}) = \Delta \mathbf{x}^T \mathbf{J}^T \mathbf{J} \Delta \mathbf{x} - 2 \Delta \mathbf{x}^T \mathbf{J} \Delta \mathbf{y} + \Delta \mathbf{y}^T \Delta \mathbf{y} \quad (2.23)$$

Zur Bestimmung des Minimums dieser quadratischen Form ist die partielle Ableitung von Gl.(2.23) nach  $\Delta \mathbf{x}$  zu berechnen und diese zu Null zu setzen. Man erhält das lineare Gleichungssystem (Normalgleichungen)

$$\mathbf{J}^T \mathbf{J} \Delta \mathbf{x} - \mathbf{J}^T \Delta \mathbf{y} = \mathbf{0} \quad (2.24)$$

mit der geschätzten Änderung des Zustandsvektors:

$$\underline{\Delta \mathbf{x}} = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \Delta \mathbf{y} \quad (2.25)$$

Dies ist ein mögliches Lösungsverfahren für nichtlineare Gleichungen, das nicht zwingend konvergieren muss. In der Praxis hat sich dieser Ansatz aber als sehr brauchbar

<sup>3</sup> Der Ausdruck  $(\mathbf{x}_k)$  für die Abhängigkeit der Jacobimatrix  $\mathbf{J}(\mathbf{x}_k)$  vom aktuellen Zustandsvektor  $\mathbf{x}_k$  wurde aus Gründen der Übersichtlichkeit weggelassen.

herausgestellt. Alternative Methoden wie der klassische Gradientenabstieg wurden nicht untersucht, da das vorgestellte Verfahren keine wirklichen Konvergenzprobleme zeigte. Wie der weitere Verlauf dieser Arbeit noch zeigen wird, ist nicht die Minimierung der nichtlinearen Funktion das eigentliche Problem des gewählten Ansatzes, sondern die Bereitstellung korrekter Zielpunkte (Korrespondenzproblem) und die dafür erforderlichen Näherungswerte des Zustandsvektors  $\mathbf{x}_0$ .

### Der verwendete Algorithmus

Das mathematische Rüstzeug besteht im Wesentlichen aus linearer Algebra. Die Schwierigkeit besteht zunächst in der Gewinnung des Modells  $\mathbf{p}(\mathbf{x}_k)$ . Außerdem müssen die partiellen Ableitungen  $\frac{\partial \mathbf{p}(\mathbf{x}_k)}{\partial \mathbf{x}_k}$  berechnet werden, damit die Jacobimatrix  $\mathbf{J}(\mathbf{x}_k)$  aufgestellt werden kann. Dies wird Gegenstand der drei Kapitel über die Modellierung sein.

Für die Modellapproximation werden durch die Software die folgenden Schritte durchgeführt:

1. Stelle einen möglichst guten Näherungswert für den Zustandsvektor  $\mathbf{x}_0$  bereit
2. Stelle das/ein Vorwärtsmodell (zum Beispiel  $\mathbf{p}(\mathbf{x}_k)$ ) auf, möglichst unter Berücksichtigung von Verdeckungen
3. Stelle die Korrespondenz zwischen Modell und Bild her (Bereitstellung von  $\mathbf{z}$ ) und berechne die Abweichungen  $\Delta \mathbf{y}$  zwischen Modell und Bild (im Wesentlichen Verschiebungen)
4. Leite die Modellpunktkoordinaten nach dem Zustandsvektor ab ( $\frac{\partial \mathbf{p}(\mathbf{x}_k)}{\partial \mathbf{x}_k}$ ) und baue damit die Jacobimatrix  $\mathbf{J}(\mathbf{x}_k)$  auf
5. Wiederhole die Schritte 2 bis 4, für eine große Anzahl von Punkten, damit Gl.(2.18) ausreichend überbestimmt ist
6. Löse das Gleichungssystem mit Gl.(2.25) und erhalte  $\Delta \mathbf{x}$
7. Ändere den Zustandsvektor mit Gl.(2.20) und verwende ihn als besseren Näherungswert
8. Wiederhole Schritte 2 bis 7, bis Konvergenz erreicht ist

Das Ganze wird aufwendig, weil die analytische Ableitung (Schritt 4) des Modells nach den Zustandsvektorkomponenten relativ kompliziert ist. Es ist aber jederzeit möglich, die Ableitungen (zum Beispiel zur Probe) numerisch nach Anhang F durchzuführen. Je nach Anzahl der Zustandsvektorkomponenten und der verwendeten Modelle ist dies aber rechenzeitintensiv. Besser ist die analytische Berechnung der Differenziale, zumal in der Literatur ein automatischer Formalismus existiert (siehe Anhang E). Deshalb werden die Ableitungen für jede Stufe der Modellierung angegeben. So lässt sich das Modell zunächst im 3D-Raum ableiten und diese Ableitungen mit dem Kameramodell nach Abschnitt 5.2.2 quasi ins Bild projizieren.

## 2. Grundlagen

### Parameteranzahl vs. Genauigkeit

In dieser Arbeit wird häufig ein überbestimmtes Gleichungssystem  $\mathbf{J} \cdot \Delta \mathbf{x} = \Delta \mathbf{y}$  zur Schätzung des unbekanntem Zustandsvektors  $\mathbf{x}$  verwendet. Erfolgt die Parameterschätzung durch Ausgleichsrechnung (Zusammenfassung in Anhang C aus [Luh00]), lässt sich die Standardabweichung einer Unbekannten des Zustandsvektors  $\mathbf{x}$  mit Gl.(2.26) berechnen.

$$\hat{\sigma}_i = \hat{\sigma}_0 \cdot \sqrt{q_{ii}} \quad (2.26)$$

Dabei ist  $q_{ii}$  das Hauptdiagonalelement der Matrix  $\mathbf{Q} = (\mathbf{J}^T \mathbf{J})^{-1}$ , aus der bekannten Formel zur Lösung überbestimmter Gleichungssysteme  $\Delta \mathbf{x} = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \cdot \Delta \mathbf{y}$  (siehe auch Anhang B). Eine kleine Standardabweichung der Unbekannten  $\mathbf{x}$  ist ein häufig verwendetes Indiz für eine hohe Genauigkeit der geschätzten Parameter.

Die Standardabweichung a posteriori (nach der Ausgleichung) berechnet sich durch:

$$\hat{\sigma}_0 = \sqrt{\frac{\mathbf{v}^T \mathbf{v}}{r}} \quad (2.27)$$

$$\mathbf{v} = \mathbf{J} \cdot \Delta \mathbf{x} - \Delta \mathbf{y} \quad (2.28)$$

mit der Redundanz:

$$r = n - u \quad (2.29)$$

Dabei ist  $n$  die Anzahl der Beobachtungen (Gleichungen) und  $u$  die Anzahl der unbekanntem Parameter. Nun ist leicht einzusehen, dass sich eine kleine Standardabweichung durch eine hohe Redundanz  $r$  erreichen lässt. Deshalb ist man bestrebt, für die Modellapproximation möglichst deutlich mehr Beobachtungen als unbekanntem Modellparameter zu erhalten. Es ist bekannt, dass mit einer hohen Redundanz nicht nur die Genauigkeit der zu schätzenden Parameter, sondern auch die Robustheit gegen Störungen oder Ausreißer steigt.

Die Anzahl unabhängiger Beobachtungen  $n$  ist leider durch die verfügbaren Informationen begrenzt, zum Beispiel durch die Eigentextur des Messobjektes, die Bildauflösung oder die Anzahl der Kameras. Gemäß Gl.(2.29) lässt sich die Redundanz auch erhöhen, indem die Anzahl der unbekanntem Parameter  $u$  verkleinert wird. Das ist auch der Grund, weshalb man bestrebt ist, möglichst einfache Modelle mit möglichst wenigen Parametern zu verwenden. Leider können zu einfache Modelle die physikalischen Gegebenheiten nur begrenzt wiedergeben und es entstehen systematische Fehler. Das äußert sich dann in einer Vergrößerung von  $\mathbf{v}^T \mathbf{v}$  und gemäß Gl.(2.27) damit auch wieder in einer steigenden Standardabweichung a posteriori  $\hat{\sigma}_0$ .

Um eine möglichst hohe Genauigkeit zu erreichen, muss also ein Kompromiss für die optimale Parameteranzahl gefunden werden. Zum einen darf sie nicht zu klein werden, weil das Modell auf das Problem angepasst bleiben muss und zum anderen darf sie nicht größer werden, als Bild-Information zur Verfügung steht, da sonst die Redundanz des Gleichungssystems und damit auch wieder die Genauigkeit sinkt.

### 3. Modellierung des menschlichen Skeletts

Das Analyse-durch-Synthese Konzept wurde bereits in Abschnitt 2.2.2 erläutert. Der Hauptvorteil ist die Möglichkeit, a priori Wissen durch eine detaillierte Modellierung einzubringen. Modelliert werden der menschliche Körper bestehend aus Skelett und Oberflächenbeschreibung, sowie das abbildende System (Aufbau von Kameras).

#### 3.1. Festlegung des Weltkoordinatensystems

Große Teile der verwendeten Koordinatentransformationen wurden aus [Wac97] übernommen. Die absolute Lage aller Komponenten des Messsystems, wie Kameras und Messobjekte, werden in einem globalen rechtshändigen Koordinatensystem angegeben. Es wird *Weltkoordinatensystem (WKS)* genannt und so definiert, dass die xy-Ebene exakt auf einem als eben angenommenen Boden liegt. Dabei zeigt die z-Achse nach oben (Abbildung 3.1) und die Koordinate  $z=0$  kennzeichnet einen Punkt auf dem Boden. Die Hauptbewegungsrichtung eines Probanden erfolgt entlang der x-Achse des Weltkoordinatensystems entweder in positiver oder in negativer Richtung.

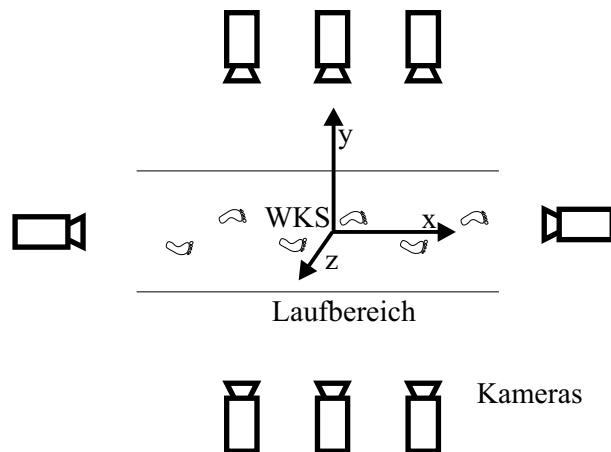


Abbildung 3.1.: Laufbereich und Festlegung des Weltkoordinatensystems (WKS)

Aus dieser Festlegung resultieren die folgenden Vorteile:

- Ein 2D-Lageplan des Messfeldes mit Kameras und Probanden ergibt sich durch Weglassen der z-Koordinate.

### 3. Modellierung des menschlichen Skeletts

- Dadurch ist es relativ einfach, manuell Werte vorzugeben oder zu interpretieren.
- Die Höhenangaben sind an der z-Achse abzulesen.
- Die Orientierung von Körperteilen kann relativ zum Boden (WKS) angegeben werden.
- Der Verlust eines Freiheitsgrades<sup>1</sup> von Rotationsmatrizen kann weitgehend vermieden werden.
- Kollisionen des Körpermodells mit dem Boden sind einfach zu überprüfen ( $z < 0$ ).

## 3.2. Ein (zu) einfaches Skelettmodell

Der folgende Abschnitt beschreibt ein sehr einfaches Skelettmodell. Es ist etwa in einem Tag zu implementieren. Am Ende des Abschnittes werden jedoch seine Grenzen gezeigt und um es vorweg zu nehmen, dass hier beschriebene Skelettmodell wird später nicht weiter verwendet. Aber es ist geeignet die Grundlagen zu erläutern.

Aus der Technischen Mechanik sind Finite-Elemente-Methoden zur Modellierung von Feder-Masse-Systemen bekannt. Dieses Modell besteht aus Massepunkten, die durch masselose Federelemente miteinander verbunden sind. Es ist sehr leicht möglich, mit der Grundgleichung der Mechanik die Bewegung, der gekoppelten Massepunkte zu simulieren. Für die reine Modellapproximation kann auf die Modellierung der dynamischen Eigenschaften verzichtet werden. Für ein ganz einfaches Skelettmodell reicht es aus, dass Gelenkpunkte beweglich sind und bestimmte Abstände zwischen ihnen eingehalten werden.

Auf diese Weise lässt sich mit wenig Aufwand ein interaktives Skelettmodell eines Menschen mit einer beachtlichen Qualität implementieren, das nahezu keine Nichtlinearität aufweist. Es kommt ohne Koordinatentransformation aus, da alle Koordinatenangaben im Weltkoordinatensystem erfolgen können. Die mathematischen Grundlagen dafür sind sehr einfach. Das Modell besteht aus 13 Punkten  $\mathbf{w}_1 \dots \mathbf{w}_{13}$ , die durch 18 Strecken miteinander verbunden sind. Die Koordinaten der Punkte werden direkt im Zustandsvektor  $\mathbf{x}$  gespeichert, der demzufolge  $13 \cdot 3$  Komponenten enthält. Für jede der 18 Strecken wird eine Länge entsprechend den Proportionen des Probanden vorgegeben.

Die Komponenten des Zustandsvektors sind aber noch vollständig unabhängig voneinander. Damit die Proportionen entsprechend der vorgegebenen Streckenlängen eingehalten werden, wird ein Gleichungssystem mit Streckenbeobachtungen aufgestellt.

Zur Verdeutlichung des Prinzips sei hier ein Beispiel angegeben. Angenommen, das Skelett besteht lediglich aus zwei Gelenkpunkten  $i$  und  $j$ , die durch eine Strecke mit bekannter Länge  $s_0$  miteinander verbunden sind. Der Zustandsvektor besitzt bei zwei Punkten

---

<sup>1</sup>wird in der engl. Literatur [FDH97] als Gimbal-Lock-Problem bezeichnet

### 3.2. Ein (zu) einfaches Skelettmodell

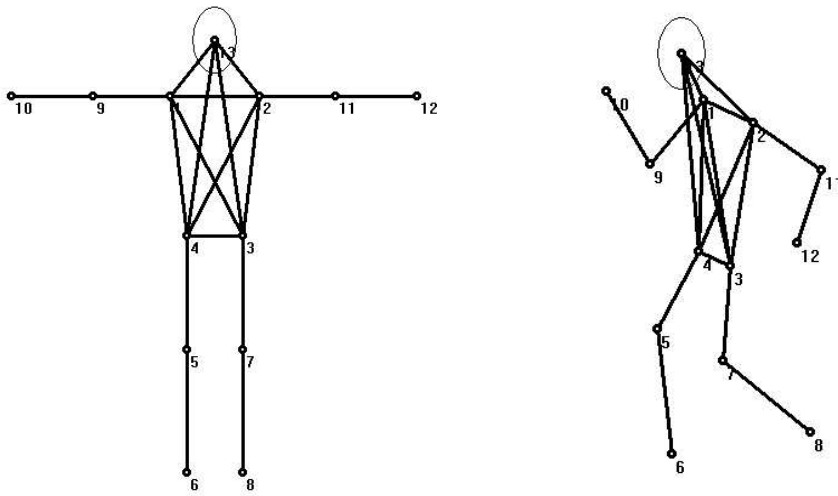


Abbildung 3.2.: Sehr einfaches Skelettmodell aus 13 Punkten und 18 Strecken

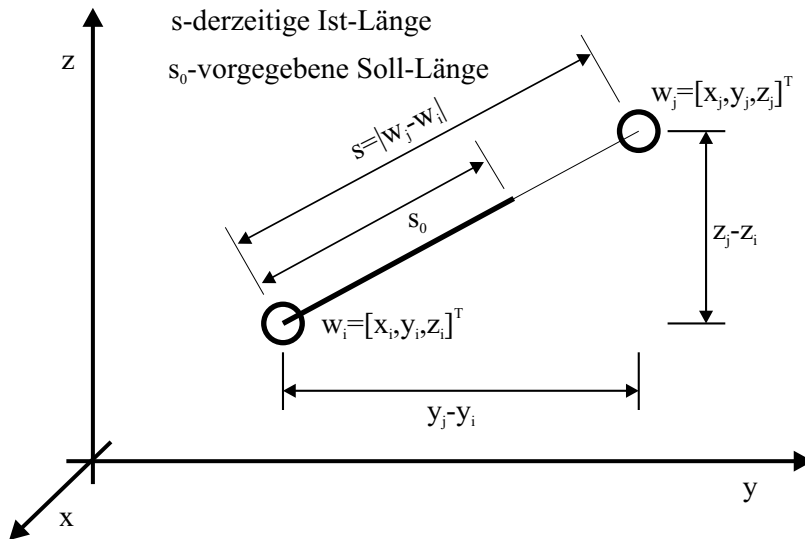


Abbildung 3.3.: Zwei Punkte durch eine Strecke verbunden. Ziel ist es zwar die Punkte  $w_i$  und  $w_j$  beweglich zu lassen, aber die Soll-Streckenlänge  $s_0$  via Gl.(3.12) einzuhalten.

### 3. Modellierung des menschlichen Skeletts

lediglich sechs Komponenten.

$$\mathbf{x} = \begin{bmatrix} \mathbf{w}_i \\ \mathbf{w}_j \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \\ z_i \\ x_j \\ y_j \\ z_j \end{bmatrix} \quad (3.1)$$

Allerdings sollen die derzeitigen Punktkoordinaten ( $\mathbf{w}_i$  und  $\mathbf{w}_j$ ) noch nicht den richtigen Abstand  $s$  besitzen. In Abbildung 3.3 ist zu sehen, dass der Abstand noch größer ist als durch die Streckenlänge  $s_0$  vorgegeben. Der aktuelle Abstand lässt sich mit dem Satz des Pythagoras berechnen:

$$s(\mathbf{w}_i, \mathbf{w}_j) = |\mathbf{w}_j - \mathbf{w}_i| = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2} \quad (3.2)$$

Eine Näherung ergibt sich durch die Taylorreihenentwicklung und Abbruch nach dem linearen Term. Dafür sind die folgenden partiellen Ableitungen der Streckenlänge nach den Punktkoordinaten erforderlich Gl.(3.2):

$$s = |\mathbf{w}_j - \mathbf{w}_i| \quad (3.3)$$

$$\frac{\partial s}{\partial x_j} = \frac{1}{2} \cdot \frac{1 \cdot (2x_j - 2x_i)}{\sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2}} = \frac{x_j - x_i}{s} = a \quad (3.4)$$

$$\frac{\partial s}{\partial x_i} = -\frac{x_j - x_i}{s} = -a \quad (3.5)$$

$$\frac{\partial s}{\partial y_j} = \frac{y_j - y_i}{s} = b \quad (3.6)$$

$$\frac{\partial s}{\partial y_i} = -\frac{y_j - y_i}{s} = -b \quad (3.7)$$

$$\frac{\partial s}{\partial z_j} = \frac{z_j - z_i}{s} = c \quad (3.8)$$

$$\frac{\partial s}{\partial z_i} = -\frac{z_j - z_i}{s} = -c \quad (3.9)$$

In einem Gleichungssystem sollen nun Lageänderungen

$$\Delta \mathbf{x} = \begin{bmatrix} \Delta \mathbf{w}_i \\ \Delta \mathbf{w}_j \end{bmatrix} = \begin{bmatrix} \Delta x_i \\ \Delta y_i \\ \Delta z_i \\ \Delta x_j \\ \Delta y_j \\ \Delta z_j \end{bmatrix} \quad (3.10)$$

für die Gelenkpunkte berechnet werden, damit sie näherungsweise die Soll-Länge  $s_0$  erreichen, gleichzeitig aber gegebene Nebenbedingungen einhalten, zum Beispiel ihre



### 3.2. Ein (zu) einfaches Skelettmodell

eigenen Koordinaten möglichst wenig ändern. Aus [Kru83] wurde die folgende Gleichung<sup>2</sup> für eine Streckenbeobachtung entnommen. Sie bewirkt, dass vorgegebene Punktabstände bestmöglich eingehalten werden.

$$a \cdot \Delta x_j + b \cdot \Delta y_j + c \cdot \Delta z_j - a \cdot \Delta x_i - b \cdot \Delta y_i - c \cdot \Delta z_i = s_0 - s \quad (3.11)$$

In diesem einfachen Beispiel bestünde das Gleichungssystem eigentlich nur aus einer Gleichung, besitzt aber  $2 \cdot 3 = 6$  unbekannte Parameter. Mit Standardmethoden wäre die Gleichung also noch nicht zu lösen. Deshalb werden zusätzliche Nebenbedingungen mit einem kleinen Gewicht aufgestellt ( $\mu \approx 0.01$ ), um die Änderungen der Punktkoordinaten möglichst klein zu halten. Sie wirken quasi wie 'Luftreibung' auf die Gelenkpunkte<sup>3</sup>. Dadurch erhält man ein Gleichungssystem, dessen Lösbarkeit sichergestellt ist.

$$\begin{bmatrix} -a & -b & -c & a & b & c \\ \mu & 0 & 0 & 0 & 0 & 0 \\ 0 & \mu & 0 & 0 & 0 & 0 \\ 0 & 0 & \mu & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu \end{bmatrix} \cdot \begin{bmatrix} \Delta \mathbf{w}_i \\ \Delta \mathbf{w}_j \end{bmatrix} = \begin{bmatrix} s_0 - s \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (3.12)$$

Die Lösung des Gleichungssystems erfolgt durch Ausgleichsrechnung nach dem Schema aus Abschnitt 3. Nach jedem Iterationsschritt werden die Änderungen am Zustandsvektor angebracht, indem  $\Delta \mathbf{w}_i$  und  $\Delta \mathbf{w}_j$  zu den bisherigen Punktkoordinaten  $\mathbf{w}_i$  und  $\mathbf{w}_j$  addiert werden:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta \mathbf{x} \quad (3.13)$$

Interaktiv wird das Modell dadurch, dass zusätzliche Beobachtungsgleichungen aufgestellt werden können, welche die Gelenkpunkte an die gewünschten Positionen verschieben. Um zum Beispiel den Punkt  $j$  mit den Koordinaten  $\mathbf{w}_j = [x_j, y_j, z_j]^T$  zu den Koordinaten  $[4, 7, -2]^T$  zu verschieben, würde man das Gleichungssystem 3.12 um drei Zeilen verlängern:

$$\begin{bmatrix} -a & -b & -c & a & b & c \\ \mu & 0 & 0 & 0 & 0 & 0 \\ 0 & \mu & 0 & 0 & 0 & 0 \\ 0 & 0 & \mu & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \Delta \mathbf{w}_i \\ \Delta \mathbf{w}_j \end{bmatrix} = \begin{bmatrix} s_0 - s \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 4 - x_j \\ 7 - y_j \\ -2 - z_j \end{bmatrix} \quad (3.14)$$

<sup>2</sup>Sie entsteht durch Ableiten des Satzes des Phytagoras nach den Punktkoordinaten.

<sup>3</sup> Vergl. Levenberg-Marquardt-Verfahren im Anhang B, hier würde  $\mu$  ggf. nach jeder Iteration angepasst werden

### 3. Modellierung des menschlichen Skeletts

Der ganze Vorgang kann wiederholt werden, bis  $\Delta \mathbf{x}$  gegen null geht.

Für jeden zusätzlichen Punkt im Skelett wird das Gleichungssystem um drei Spalten erweitert. Genaugenommen auch um je drei Zeilen, wenn man die Stabilitätsgleichungen mit  $\mu$  dazurechnen will. Für jede weitere Strecke vergrößert sich das System um eine Zeile nach Gl.(3.11). Werden die Gleichungen für das ganze Skelettmodell mit 13 Punkten aufgestellt, dann werden bereits 39 Spalten benötigt. 18 Gleichungen entstehen durch die Streckenbeobachtungen. Das System kann jedoch sehr schnell gelöst werden, wenn man zur Lösung die Libsparse [KSV86] verwendet.

Der Oberkörper des Skelettmodells wird durch eine schiefe Pyramide gebildet, dessen Spitze in der Nähe des Kopfes liegt. Durch diagonale Strecken wird erreicht, dass der Oberkörper seine Form behält und die Schultergelenke relativ zu den Hüftgelenken stabil bleiben. Das Modell ist im Raum vollständig beweglich, es benötigt keine einzige Rotationsmatrix und leidet daher nicht an Singularitäten. Das Modell lässt sich in jede

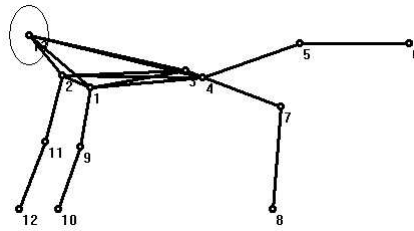


Abbildung 3.4.: Ohne weitere Nebenbedingungen ist das Modell im Raum vollständig beweglich und lässt sich auch in Haltungen bringen, die beim normalen aufrechten Gang nicht vorkommen.

beliebige Position bringen, da es bisher keinerlei Zwangsbedingungen für einen aufrechten Gang gibt (Abbildung 3.4). Für die Ganganalyse ist das Modell in dieser Form noch ungeeignet. Mit wenig Aufwand lassen sich aber Gleichungen finden, die einen aufrechten Oberkörper erzwingen. Es ist leicht einzusehen, dass sich die  $z$ -Koordinaten der beiden Schultergelenke bei aufrechtem Oberkörper nur wenig unterscheiden. Dasselbe gilt für die beiden Hüftgelenke. Das System lässt sich um zwei Gleichungen erweitern, die dafür sorgen, dass die  $z$ -Koordinaten von je zwei Gelenkpunkten nahezu auf gleicher Höhe bleiben. Angenommen, das linke Schultergelenk hätte den Index  $i$  und das rechte Schultergelenk den Index  $j$ , dann sorgt die folgende Gleichung für nahezu identische  $z$ -Koordinaten ( $z_i$  und  $z_j$ ).

$$\begin{bmatrix} 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \Delta \mathbf{w}_i \\ \Delta \mathbf{w}_j \end{bmatrix} = \begin{bmatrix} z_i - z_j \end{bmatrix} \quad (3.15)$$

Die Gleichung allein reicht jedoch noch nicht aus, Abbildung 3.4 demonstriert diesen Fall. So besitzen zwar die Gelenkpunkte der Hüfte die gleiche  $z$ -Koordinate, dass gilt auch für die Gelenkpunkte der Schulter, ein aufrechter Oberkörper wird dadurch allerdings noch nicht erzwungen.

### 3.2. Ein (zu) einfaches Skelettmodell

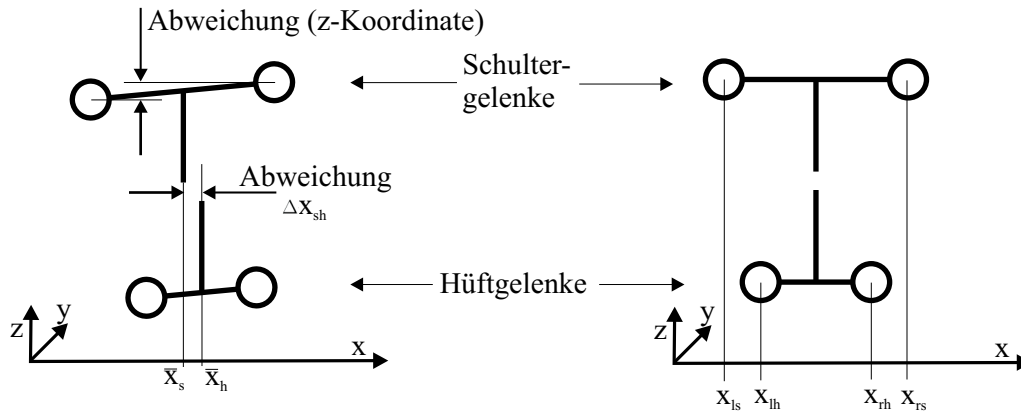


Abbildung 3.5.: Nebenbedingungen für den aufrechten Oberkörper. Die Kreise zeigen die Gelenkpunkte der linken und rechten Schulter sowie der linken und rechten Hüfte. Die vertikalen Linien stellen die arithmetischen Mittelwerte der beiden Gelenkkoordinaten (xy) dar. Links ist zu erkennen, was passiert, wenn der Oberkörper leicht geneigt wird.

In Abbildung 3.5 ist zu erkennen, dass es eine weitere Nebenbedingung für den aufrechten Gang gibt. Werden die arithmetischen Mittelwerte der x-Schulterkoordinaten ( $\bar{x}_s$ ) und der Hüftkoordinaten ( $\bar{x}_h$ ) berechnet, dann müssten diese in etwa identisch sein (Abbildung 3.5 rechts). Wird der Oberkörper jedoch leicht geneigt, entsteht eine Abweichung (vertikale Linien). Dies lässt sich zum Aufstellen weiterer Nebenbedingungen verwenden. Dasselbe gilt für die y-Koordinaten.

$$\bar{x}_h = \frac{(x_{lh} + x_{rh})}{2} \quad (3.16)$$

$$\bar{x}_s = \frac{(x_{ls} + x_{rs})}{2} \quad (3.17)$$

$$\bar{y}_h = \frac{(y_{lh} + y_{rh})}{2} \quad (3.18)$$

$$\bar{y}_s = \frac{(y_{ls} + y_{rs})}{2} \quad (3.19)$$

Mit

$$\Delta x_{sh} = \bar{x}_h - \bar{x}_s \rightarrow \min \quad (3.20)$$

$$\Delta y_{sh} = \bar{y}_h - \bar{y}_s \rightarrow \min \quad (3.21)$$

ergibt sich die Gleichung für die Nebenbedingung zu:

$$\begin{bmatrix} \frac{1}{2} & 0 & 0 & \frac{1}{2} & 0 & 0 & -\frac{1}{2} & 0 & 0 & -\frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 & \frac{1}{2} & 0 & 0 & -\frac{1}{2} & 0 & 0 & -\frac{1}{2} & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{w}_{ls} \\ \Delta \mathbf{w}_{rs} \\ \Delta \mathbf{w}_{lh} \\ \Delta \mathbf{w}_{rh} \end{bmatrix} = \begin{bmatrix} \Delta x_{sh} \\ \Delta y_{sh} \end{bmatrix} \quad (3.22)$$

### 3. Modellierung des menschlichen Skeletts

Es lässt sich eine weitere Nebenbedingung mit geringem Gewicht einführen, die den Oberkörper in der Nähe einer bestimmten Höhe hält. Dies ist aber nicht unbedingt erforderlich.

Zu diskutieren sind einige Eigenschaften dieses einfachen Skelettmodells. Die Koordinaten der Gelenkpunkte sind die einzigen freien Parameter des Modells, die Gelenkwinkel zwischen den Körperteilen sind jedoch nicht direkt aus dem Zustandsvektor ablesbar. Es ist aber leicht möglich, den kleinsten eingeschlossenen Winkel zwischen zwei Strecken (drei Gelenkpunkten) und damit zwischen zwei Körperteilen zu berechnen. Deutlich schwieriger ist es jedoch Winkelgrenzen zwischen zwei Körperteilen als Nebenbedingung zu formulieren. Dadurch lassen sich die Extremitäten des Modells in Gelenkwinkelbereiche bringen, die anatomisch unmöglich wären. Dies ist wohl auch der Hauptnachteil des einfachen Modells.

Da alle Koordinatenangaben absolut im Weltkoordinatensystem erfolgen, ist eine Modellierung der Bewegung mit einem Bewegungsmodell (zum Beispiel Periodizität) zumindest schwierig. Jede Translation und Rotation des Probanden ändert stets alle Koordinatenangaben der Gelenkpunkte. Besser wäre eine Modellierung der Gangbewegung in einem Koordinatensystem, das die Bewegungen der Körperteile relativ zur aktuellen Lage und Orientierung des Probanden beschreibt. Dadurch wäre dann eine getrennte Beschreibung des periodischen Gangmusters und den sich ständig ändernden Lage-Parametern zu erhalten.

Ein weiterer Nachteil wird erst im hinteren Teil dieses Kapitels deutlich. Das Skelett definiert zwar eindeutig die Rotationsachsen jedes Körperteiles, aber es macht keine Aussage über die Rotation des Körperteiles auf dieser Achse. Das ist so lange kein Problem, wie mit rotationssymmetrischen Formbeschreibungen der Körperteile gearbeitet werden soll. Dies ist dann die Einsatzgrenze des Modells. Eine detailliertere Beschreibung ist dann mit dem Modell nicht mehr möglich. Möchte jemand dennoch mit diesem Skelettmodell weiterarbeiten, so muss er überlegen, wie die verloren gegangenen Orientierungen der Körperteile rekonstruiert werden könnten. Der Koordinatenursprung eines jeden Körperteilkoordinatensystems lässt sich leicht in den proximalen<sup>4</sup> Gelenkpunkt legen. Eine Achse des Koordinatensystems könnte dann in Richtung distalem<sup>5</sup> Gelenkpunkt zeigen. Aber in welche Richtung genau die zweite Achse und damit auch die dritte zeigen soll, bleibt unklar und ist mit diesem Modell wahrscheinlich nicht zu klären. Eine Möglichkeit wäre vielleicht, durch drei Punkte des Oberkörpers (Kreuzprodukt) einen Vektor zu definieren, der nach vorn zeigt. Dieser Richtungsvektor könnte dann womöglich verwendet werden, um auch in jedem lokalen Körperteilkoordinatensystem zu definieren, wo genau 'vorne' ist. Die dritte Koordinatenachse erhält man durch das Kreuzprodukt. Allerdings ist dies nicht mehr möglich, wenn die Rotationsachse selbst nach vorne zeigt.

Trotz allem lassen sich bereits mit diesem einfachen Skelettmodell beachtliche Modellierungsqualitäten erreichen (Abbildung 3.6).

---

<sup>4</sup> med. zur Körpermitte hin gelegen

<sup>5</sup> med. weiter von der Körpermitte entfernt

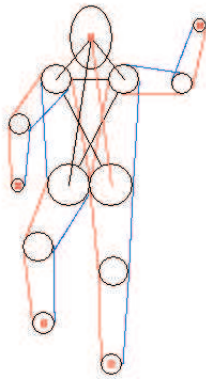


Abbildung 3.6.: Das einfache Skelettmodell mit einer Formbeschreibung nach Abschnitt 4.3, die Kreise verdeutlichen den Konusdurchmesser am jeweiligen Gelenkpunkt

#### Übersicht der Vor- und Nachteile des Modells

Das vereinfachte Skelettmodell erlaubt es, die Proportionen eines Probanden verhältnismäßig einfach zu modellieren. Es besteht aus Gelenkpunkten, deren Abstände zueinander durch Nebenbedingungen eingehalten werden.

Die Vorteile des Modells sind:

- Es ist sehr einfach und innerhalb eines Tages zu implementieren.
- Es bleibt selbst im unterbestimmten Fall stabil.
- Es gibt keine Rotationsmatrizen und dadurch keine Singularitäten.
- Es lassen sich leicht weitere Nebenbedingungen formulieren.
- Dadurch kann eine aufrechte Körperhaltung erzwungen werden.
- Sparse-Matrix-Algebra ermöglicht eine sehr schnelle Lösung der Gleichungen.

Demgegenüber stehen die folgenden Nachteile:

- Es gibt relativ viele Unbekannte.
- Absolute Koordinatenangaben erschweren ein höheres Bewegungsmodell.
- Die Gelenkwinkel sind nicht direkt ablesbar.
- Die anatomischen Gelenkwinkelgrenzen lassen sich nicht einfach formulieren.
- Es können ohne weitere Zusatzbedingungen keine lokalen Koordinatensysteme für jedes Körperteil festgelegt werden.
- Deshalb wird die Formbeschreibung der Körperteile normalerweise rotationssymmetrisch erfolgen.

### 3. Modellierung des menschlichen Skeletts

Die genannten Nachteile machen eine bessere Formulierung des Skeletts wünschenswert. Das ist allerdings deutlich komplizierter und aufwändiger zu implementieren. Deshalb werden im folgenden Abschnitt zunächst die Grundlagen erläutert. Anschließend wird ein Beispiel für ein einfaches Modell der Beine vorgestellt.

### 3.3. Verbessertes Skelettmodell

Das Skelettmodell dient der Beschreibung der kinematischen Eigenschaften des Modells. Das Skelett besteht aus Knochen (*Bones*) und Gelenken (*Joints*). Unter einem *Bone* wird ein lokales Koordinatensystem verstanden, das zur Definition geometrischer Größen eines Körperteiles verwendet wird. Diese geometrischen Größen könnten zum Beispiel die Konturpunktkoordinaten eines Knochens in diesem Körperteil sein. Auf diese Weise werden am Skelett weitere Komponenten des Modells (zum Beispiel Formbeschreibung) quasi befestigt, die ihre Haltung so einnehmen sollen, wie sie durch den Zustandsvektor vorgegeben sind. Jeder Freiheitsgrad entspricht dabei einem Element des Zustandsvektors. Wird eine Komponente des Zustandsvektors geändert, so bewegen sich die korrespondierenden Teile des Skeletts und die daran befestigten Modellteile gemäß dem auftretenden Freiheitsgrad.

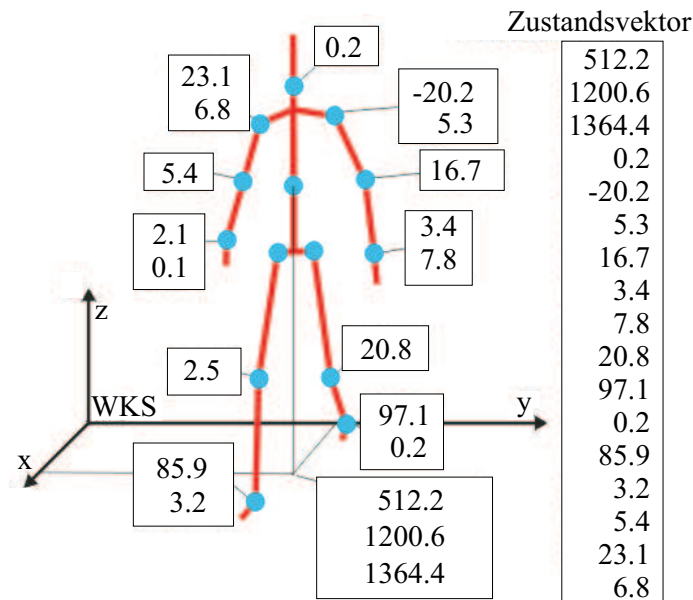


Abbildung 3.7.: Hierarchisches Skelettmodell eines Menschen, Parameter wie Gelenkwinkel oder Position im Raum werden zu einem Zustandsvektor zusammengefasst. Es wurden willkürliche Beispieldaten zur Darstellung verwendet.

Bones definieren lokale Koordinatensysteme für jedes Körperteil. Die Form eines Körperteiles lässt sich dann leicht in diesem lokalen Koordinatensystem beschreiben. Zu-

### 3.4. Grundlagen der Koordinatentransformationen

sammenhängende Punkte mit den selben Rotations- und Translationseigenschaften (zum Beispiel ein Unterschenkel) werden mit dem lokalen Koordinatensystem durch den Raum bewegt. Das können Teile der sichtbaren Hülle sein oder ein Gelenk. An diesem Gelenk lässt sich ein weiteres Bone befestigen. Nach [Vel00] wird es Child-Bone genannt, weil es den Bewegungen des Vorgänger-Bones (Parent) folgt. Die Lage eines Gelenkes wird dabei im lokalen Koordinatensystem des Parent-Bones angegeben. Beim Aufbau eines solchen Skeletts entstehen dann ausgehend vom Rumpf zu den Extremitäten hierarchische Transformationsketten.

Dieser hierarchische Aufbau erlaubt bereits eine Vorwärts (Forward) Kinematik. Sie erleichtert die Arbeit mit dem Modell. Es reicht, zum Beispiel aus den Winkelverlauf im Schultergelenk vorzugeben. Der Ober-, der Unterarm und die Hand bewegen sich dann in korrekter Weise mit.

Die Drehungen in den Gelenken werden durch Rotationsmatrizen beschrieben. Diese enthalten die relativen Drehungen im aktuellen Gelenk. Sie können unter Angabe von drei Winkeln oder eines Quaternions berechnet werden. Zum Beispiel sind das Schultergelenk und des Hüftgelenk jeweils Kugelgelenke, bei denen alle drei Winkel modelliert werden müssten. Oft sind die Gelenke aber in ihrer Bewegungsfreiheit eingeschränkt. Sie besitzen zum Teil weniger als drei Freiheitsgrade und/oder es sind nicht die vollen Winkelausschläge erlaubt. Diese Einschränkungen werden für das Modell dem biologischen Vorbild entsprechend gewählt. Das schafft zusätzliche Nebenbedingungen und verkleinert den Suchraum für die Bildsequenzauswertung. Ein Skalierungsfaktor lässt sich verwenden, um näherungsweise die Länge der Gliedmaßen festzulegen, wenn lediglich die Körpergröße bekannt ist.

Der kinematische Zustand des Systems wird in einem *Zustandsvektor*  $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$  zusammengefasst. Die Komponenten des Zustandsvektors enthalten Parameter, wie Lage und Orientierung des Probanden sowie einzelne Gelenkwinkel. Es ist ebenfalls möglich, direkt Koordinatenangaben der Gelenke im Zustandsvektor zu repräsentieren. Denkbar sind auch Größenangaben wie Körperhöhe oder Durchmesser eines Beines. Soll eine dynamische Modellierung erfolgen, lässt sich der Zustandsvektor um abgeleitete Größen, wie Geschwindigkeiten und Beschleunigungen erweitern.

## 3.4. Grundlagen der Koordinatentransformationen

Die Koordinatentransformationen können sehr kompakt mit homogenen Koordinaten<sup>6</sup> [BSG<sup>+</sup>96] erfolgen. Dabei werden Punktkoordinaten im 3D-Raum durch eine vierte Koordinate = 1 ergänzt. Die Beschreibung der Form eines dreidimensionalen Körperteiles erfolgt zunächst nicht direkt in Weltkoordinaten, sondern in einem lokalen Koordinatensystem, das fest mit dem Körperteil verbunden ist. Ein Punkt in lokalen Körperteilkoordinaten wird durch den Vektor  $\mathbf{l}_a$  gekennzeichnet. Dabei ist  $a$  der Index des Körperteiles.

---

<sup>6</sup>Grundlagen findet man im Internet oder in [FDFH97]

### 3. Modellierung des menschlichen Skeletts

Die Modellierung der Bewegung des Körperteiles wird dann durch Translation und Rotation (Orientierung) dieses lokalen Koordinatensystems erreicht. Beide lassen sich zu einer homogenen Transformationsmatrix mit der Dimension  $4 \times 4$  zusammenfassen.

Homogene Transformationsmatrizen  $\mathbf{T}_{0a}(\mathbf{x})$  repräsentieren dann die aktuelle Lage und Orientierung des Körperteiles. Die Klammer  $(\mathbf{x})$  deutet an, dass die Lage des Körperteiles vom aktuellen Systemzustand und damit vom Zustandsvektor  $\mathbf{x}$  abhängt. Ist die homogene Transformationsmatrix  $\mathbf{T}_{0a}(\mathbf{x})$  eines Körperteiles  $a$  bekannt, können Punkte aus den lokalen Körperteilkoordinaten  $\mathbf{l}_a = [x, y, z, 1]^T$  in Weltkoordinaten  $\mathbf{w}$  umgerechnet werden. Durch Umstellen der Gl.(3.23) (inverse Matrix  $\mathbf{T}_{0a}^{-1}$ ) lassen sich Punkte auch wieder von Weltkoordinaten in lokale Körperteilkoordinaten transformieren.

$$\mathbf{w} = \mathbf{T}_{0a}(\mathbf{x}) \cdot \mathbf{l}_a \quad (3.23)$$

$$\mathbf{l}_a = \mathbf{T}_{0a}^{-1}(\mathbf{x}) \cdot \mathbf{w} \quad (3.24)$$

Die Indizes der Transformationsmatrizen  $\mathbf{T}_{ba}$  nennen zuerst das Zielkoordinatensystem  $b$  und an zweiter Stelle das Quellkoordinatensystem  $a$ . In Gl.(3.23) wurde bereits  $b$  durch 0 ersetzt, weil das Ziel das Weltkoordinatensystem  $b = 0$  ist. Mit diesen Gleichungen sind Umrechnungen zwischen den Koordinatensystemen zweier beliebiger Körperteile ( $a$  und  $b$ ) möglich.

Die homogenen Transformationsmatrizen  $\mathbf{T}_{ba}(\mathbf{x})$  werden aus beliebigen Kombinationen der *Elementaroperationen*

- Translation (zum Beispiel  $\mathbf{E}_{tx}(x)$  Translation entlang X-Achse)
- Rotation (zum Beispiel  $\mathbf{E}_{ry}(\phi)$  Rotation um die Y-Achse)
- Skalierung

multiplikativ zusammengesetzt. Im Anhang sind diese Elementaroperationen aus [Wac97] noch einmal wiedergegeben.

Im dreidimensionalen Raum gibt es sechs Freiheitsgrade. Drei Koordinatenangaben für die Lage und drei für die Orientierung. Soll der Bewegungszustand eines Körperteiles vollständig beschrieben werden, muss man die Transformationsmatrix  $\mathbf{T}_{0a}(\mathbf{x})$  aus sechs Elementaroperationen  $\mathbf{E}_{..}$  zusammensetzen. Diese sind dem Anhang D zu entnehmen.

$$\begin{aligned} \mathbf{T}_{0a}(\mathbf{x}) &= \mathbf{T}_{0a}([x_1, x_2, \dots, x_6]^T) \\ &= \mathbf{E}_{tx}(x_1) \cdot \mathbf{E}_{ty}(x_2) \cdot \mathbf{E}_{tz}(x_3) \cdot \mathbf{E}_{rz}(x_6) \cdot \mathbf{E}_{ry}(x_5) \cdot \mathbf{E}_{rx}(x_4) \end{aligned} \quad (3.25)$$

#### 3.4.1. Kinematische Ketten

Grundlagen der kinematischen Ketten findet man in Büchern über Mechanik, Robotik und Computeranimation wie [NH02], [SZM85] sowie [FDFH97]. In gekoppelten mechanischen Systemen sind die Bewegungen der einzelnen Elemente nicht unabhängig voneinander. Allerdings sind die Grundelemente biologischer Systeme, wie Gelenke, Bänder



### 3.4. Grundlagen der Koordinatentransformationen

und Knochen bei weitem nicht so starr ausgeführt, wie bei technischen Systemen. Dennoch lässt sich das Prinzip der Robotik in gewissen Grenzen auch für biologische Systeme verwenden. Wird beispielsweise der Oberarm über das Schultergelenk bewegt, dann ändert sich in gleichem Maße auch die Lage und Orientierung von Unterarm und Hand. Die Körperteile des Armes bilden also eine *kinematische Kette*.

Eine vollständige Beschreibung der Bewegung eines jeden Kettengliedes mit sechs Freiheitsgraden wäre aufwendig und redundant. In einigen markerbasierten kommerziellen Motion-Capture-Systemen wird dies dennoch implementiert. Das erlaubt dann jedem Körperteil völlige Bewegungsfreiheit unabhängig von der Bewegung benachbarter Körperteile mit einer großen Anzahl unbekannter Parameter. In der Praxis sind die Körperteile jedoch über Gelenke mehr oder weniger flexibel miteinander verbunden. Effektiver ist es deshalb, lediglich relative Transformationen benachbarter Körperteile anzugeben. Realistisch wirkende Bewegungsabläufe können dann bereits mit deutlich weniger Freiheitsgraden beschrieben werden. Auch in dieser Arbeit wurde das realisiert.

Ein Beispiel: Die Haltung des Armes bestehend aus Ober- und Unterarm lässt sich mit  $6 + 6 = 12$  Parametern beschreiben. Dadurch kann sich jedes Körperteil völlig frei bewegen. Aber weil das Schultergelenk ein Kugelgelenk ist, lässt sich die Transformation zwischen Oberkörper und Oberarm bereits durch 3 Winkelfreiheitsgrade realisieren. Der Ellenbogen ist ein Scharniergelenk mit lediglich einem Winkelfreiheitsgrad. Damit lässt sich die Parameteranzahl für den Arm bereits von 12 auf 4 reduzieren. Bezogen auf die Bildsequenzbewertung ergibt sich eine deutliche Reduktion der Anzahl der Unbekannten. Mit positiven Auswirkungen auf die Robustheit und Genauigkeit.

Durch Matrixmultiplikation lassen sich dann beliebig lange Ketten aufstellen. Zu beachten ist, dass nicht alle Parameter der kinematischen Kette freigegeben und damit im Zustandsvektor repräsentiert sein müssen. Normalerweise ändert sich der Abstand zwischen Schultergelenk und Ellenbogen im Verlauf einer Messung nicht. Dann ist die Elementarmatrix dieser Translation konstant. Sie muss nicht jedes Mal neu aufgestellt werden. Sind zwei oder mehr aufeinander folgende Elementarmatrizen konstant, lassen sich diese ebenfalls zu einer Matrix zusammenfassen. Die Gelenk-Position eines Nachfolgers in einer kinematischen Kette wird jeweils im Koordinatensystem des Vorgängers durch Translationen angegeben.

Bei der Implementierung wird für jedes Körperteil  $a$  nur das Ergebnis aller Transformationen vom Weltkoordinatensystem bis zum Körperteil gespeichert. Für jedes Körperteil muss dabei nicht die gesamte Kette neu berechnet werden. Es reicht aus, die Transformationsmatrix  $\mathbf{T}_{0a}(\mathbf{x})$  des Vorgängers  $a$  mit der relativen Transformationsmatrix  $\mathbf{T}_{ab}(\mathbf{x})$  zu multiplizieren und im Nachfolger  $b = a + 1$  zu speichern. Durch rekursive Anwendung von Gl.(3.26) lassen sich beliebig komplizierte Modelle in Ketten und Baumstrukturen formulieren.

$$\mathbf{T}_{0b}(\mathbf{x}) = \mathbf{T}_{0a}(\mathbf{x}) \cdot \mathbf{T}_{ab}(\mathbf{x}) \quad (3.26)$$

Sehr effektiv können nun Punkte  $\mathbf{l}_a$  aus den lokalen Koordinatensystemen der Körperteile in das Weltkoordinatensystem transformiert werden. Speichert man für jedes Bone

### 3. Modellierung des menschlichen Skeletts

auch die inverse Transformationsmatrix  $\mathbf{T}_{0a}^{-1}(\mathbf{x})$ , so ist auch der umgekehrte Rechenweg effizient möglich.

## 3.5. Ein kinematisches Modell der Beine

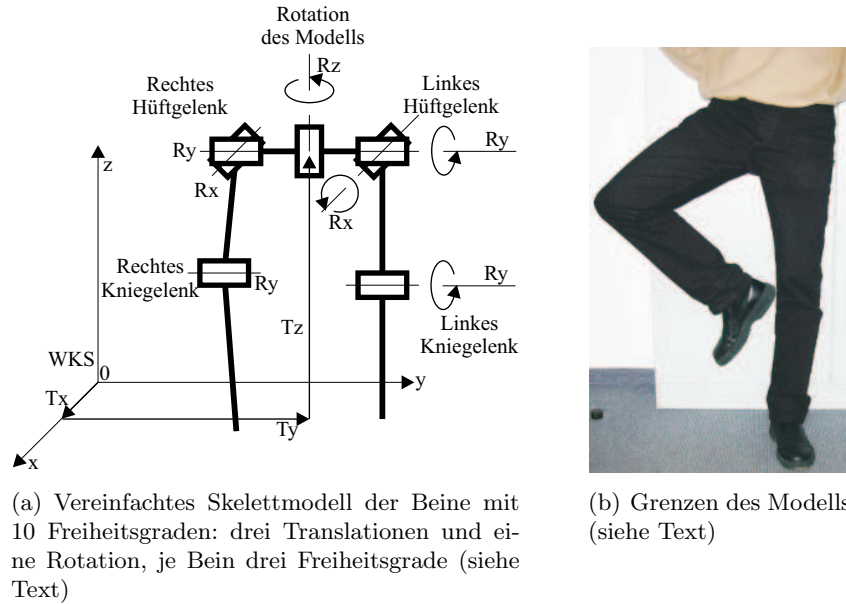


Abbildung 3.8.: Modell und Modellgrenzen

In dem implementierten System werden die 3D-Modelle durch Textdateien in einer Skriptsprache beschrieben. Im Folgenden soll anhand eines einfachen Beispiels die Verwendung der kinematischen Ketten gezeigt werden. Es werden einige Maße angegeben, sie wurden mit einem Maßband mehr oder weniger genau gemessen und stammen von einem 1830 Millimeter großen Probanden.

Die Kinematik der Ober- und Unterschenkel lässt sich bereits mit einem sehr einfachen Modell approximieren. Abbildung 3.8(a) zeigt dies. Das Modell besitzt 10 Freiheitsgrade, die im Zustandsvektor  $\mathbf{x} = [x_1, x_2, \dots, x_{10}]^T$  repräsentiert sind.

Die Beschreibung von Lage und Orientierung erfolgt durch vier Elementarmatrizen. Für Rotationsfreiheitsgrade enthalten die Elementarmatrizen jeweils Winkelfunktionen, dass lässt den Zusammenhang zwischen den Elementen der Elementarmatrizen und dem Zustandsvektor nichtlinear werden. Die Elementarmatrizen können mit Hilfe der Angaben in Anhang D erzeugt werden.

$$\mathbf{T}_{0h} = \mathbf{E}_{tx}(x_1) \cdot \mathbf{E}_{ty}(x_2) \cdot \mathbf{E}_{tz}(x_3) \cdot \mathbf{E}_{rz}(x_4) \quad (3.27)$$

Ein Punkt im lokalen Koordinatensystem der Hüfte lässt sich nun bereits mit Gl.(3.23)

### 3.5. Ein kinematisches Modell der Beine

ins Weltkoordinatensystem transformieren, indem die Matrix  $\mathbf{T}_{0h}$  (Index  $h$  steht für Hüfte) verwendet wird. Der Koordinatenursprung des Koordinatensystems für die Hüfte ist der Mittelpunkt zwischen beiden Hüftgelenken. Egal welcher Winkel in  $x_4$  vorgegeben wird, die folgende Gleichung (3.23) muss immer die Weltkoordinaten des Modells ergeben:

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ 1 \end{bmatrix} = \mathbf{T}_{0h} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (3.28)$$

Damit ist eine freie Translation der Hüfte in jede Richtung möglich. Eigentlich müsste die Orientierung des Probanden durch drei Winkel beschrieben werden. Da jedoch von einem aufrechten Gang ausgegangen wird, wurden zwei Winkel weggelassen. Damit wird eine waagerechte Haltung der Hüfte erzwungen und es werden bereits zwei Freiheitsgrade eingespart. Beide Hüftgelenke besitzen dadurch automatisch die selbe Höhe.

Für jedes Bein wird eine eigene kinematische Kette aufgestellt. Sie werden als Child-Bones hinter das Parent-Bone Hüfte  $\mathbf{T}_{0h}$  angefügt. Das Hüftgelenk ist anatomisch als Kugelgelenk ausgeführt. Demzufolge müsste es eigentlich mit drei Winkelfreiheitsgraden implementiert werden. In der Praxis können aber zu viele Freiheitsgrade nur schwer bestimmt werden. Man ist bestrebt, möglichst wenig Freiheitsgrade zuzulassen. Es ist möglich, auf einen Winkel zu verzichten, wenn keine extremen Bewegungen, wie in Abbildung 3.8(b) modelliert werden sollen. Die Beinstellung des abgebildeten Probanden ist nur dadurch möglich, weil das Hüftgelenk ein Kugelgelenk ist. Dadurch kann die Rotationsachse des Knies nahezu um  $90^\circ$  gedreht werden. Dies ist mit diesem einfachen Skelettmodell nicht möglich.

Da das Standbein in der Hüfte drehbar gelagert ist, ist es dem Menschen möglich 'um die Ecke' zu laufen, ohne den Fuß relativ zum Boden verdrehen zu müssen. Wird aber lediglich eine Analyse eines geraden Ganges mit nur einer Seitenansicht des Probanden durchgeführt, könnte wahrscheinlich selbst das Hüftgelenk mit nur einem Freiheitsgrad (für das Beugen und Strecken) beschrieben werden, weil sich der genaue Winkel für das Abspreizen und Heranziehen des Beines ohnehin nur ungenau erfassen lässt. Einen guten Kompromiss zwischen Modelltreue und Bestimmbarkeit der Parameter ist die Ausführung des Hüftgelenkes mit zwei Freiheitsgraden. In der Technischen Mechanik würde es als Kreuz- oder Kardangelenk bezeichnet werden. Es muss jedoch beachtet werden, dass mit einem Kreuzgelenk nicht alle praktisch möglichen Konfigurationen repräsentierbar sind und dadurch die Einsatzmöglichkeit dieses einfachen Modells eingeschränkt ist.

Um einen Punkt aus dem lokalen Koordinatensystem eines linken Oberschenkels in das Weltkoordinatensystem zu transformieren, verwendet man die Gl.(3.23) mit der Matrix  $\mathbf{T}_{0lo}$  (Index  $lo$  steht für den linken Oberschenkel,  $ro$  für den rechten). Diese Matrix

### 3. Modellierung des menschlichen Skeletts

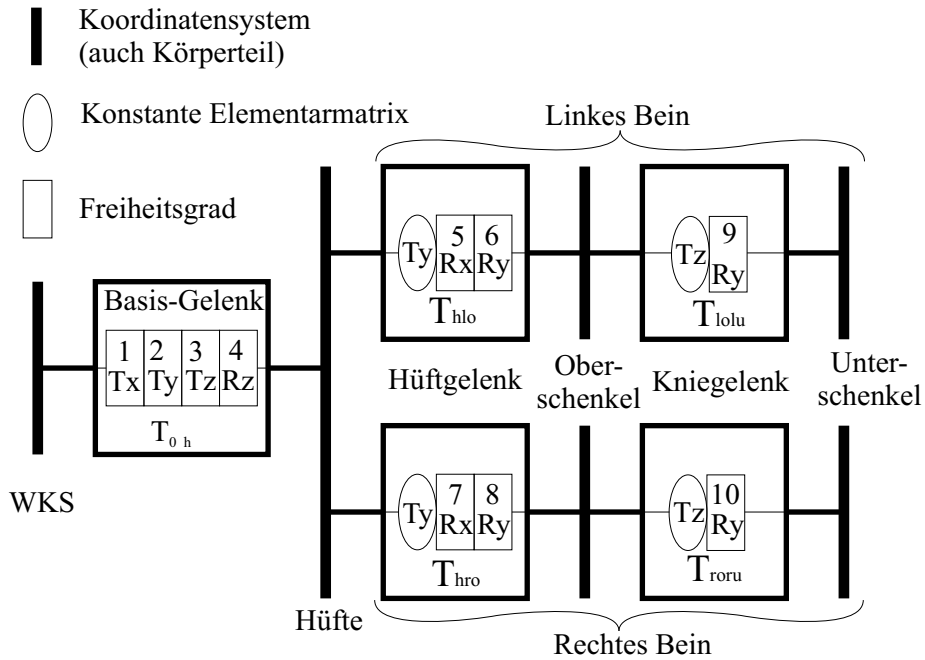


Abbildung 3.9.: Topologie des einfachen Beinmodells

berechnet sich unter Verwendung von Gl.(3.25) und Gl.(3.26):

$$\mathbf{T}_{0lo} = \mathbf{T}_{0h} \cdot \mathbf{E}_{ty}(L_h) \cdot \mathbf{E}_{rx}(x_5) \cdot \mathbf{E}_{ry}(x_6) \quad (3.29)$$

$$\mathbf{T}_{0ro} = \mathbf{T}_{0h} \cdot \mathbf{E}_{ty}(-L_h) \cdot \mathbf{E}_{rx}(x_7) \cdot \mathbf{E}_{ry}(x_8) \quad (3.30)$$

Für den rechten Oberschenkel gilt die zweite Gleichung. Der Abstand eines jeden Hüftgelenkes vom Koordinatenursprung der Hüfte wird durch die Länge<sup>7</sup>  $L_h$  ausgedrückt.

Es ist leicht einzusehen, dass für das rechte Bein die Länge  $L_h$  negativ in die Elementarmatrix  $\mathbf{E}_{ty}(-L_h)$  eingetragen werden muss. Während der Bewegung ändert sich der Abstand gewöhnlich nicht, deshalb ist auch die Elementarmatrix  $\mathbf{E}_{ty}(\pm L_h)$  konstant.

In vielen Rechenschritten werden die Weltkoordinaten der beiden Hüftgelenke berechnet. Das Hüftgelenk bildet den Koordinatenursprung des Oberschenkels. Mit Hilfe der Gl.(3.23) berechnen sich die Weltkoordinaten des linken ( $lh$ ) und rechten Hüftgelenkes

<sup>7</sup>Für einen 1830[mm] großen Probanden kann etwa  $L_h = 100[mm]$  verwendet werden.

### 3.5. Ein kinematisches Modell der Beine

(*rh*):

$$\mathbf{w}_{lh} = \begin{bmatrix} x_{lh} \\ y_{lh} \\ z_{lh} \\ 1 \end{bmatrix} = \mathbf{T}_{0lo} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (3.31)$$

$$\mathbf{w}_{rh} = \begin{bmatrix} x_{rh} \\ y_{rh} \\ z_{rh} \\ 1 \end{bmatrix} = \mathbf{T}_{0ro} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (3.32)$$

Das Knie verbindet den Unterschenkel mit dem Oberschenkel. Es wird als Scharniergelenk mit einem Freiheitsgrad modelliert. Der Unterschenkel ist ein Child-Bone des Oberschenkels. In Gl.(3.33) und Gl.(3.34) zeigt sich die Länge des Oberschenkels  $L_o = 410[mm]$  als konstante Elementarmatrix  $\mathbf{E}_{tz}(-L_o)$ .

$$\mathbf{T}_{0lu} = \mathbf{T}_{0lo} \cdot \mathbf{E}_{tz}(-L_o) \cdot \mathbf{E}_{ry}(x_9) \quad (3.33)$$

$$\mathbf{T}_{0ru} = \mathbf{T}_{0ro} \cdot \mathbf{E}_{tz}(-L_o) \cdot \mathbf{E}_{ry}(x_{10}) \quad (3.34)$$

Demzufolge berechnen sich die Weltkoordinaten der Kniegelenke (*lk* und *rk*) mit:

$$\mathbf{w}_{lk} = \begin{bmatrix} x_{lk} \\ y_{lk} \\ z_{lk} \\ 1 \end{bmatrix} = \mathbf{T}_{0lu} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (3.35)$$

$$\mathbf{w}_{rk} = \begin{bmatrix} x_{rk} \\ y_{rk} \\ z_{rk} \\ 1 \end{bmatrix} = \mathbf{T}_{0ru} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (3.36)$$

In dieser Weise könnten nun auch die Sprunggelenke und die Füße modelliert werden. Um das Beispiel aber einfach zu halten, werden die Füße zunächst ignoriert. Damit dennoch die Oberflächenbeschreibung erfolgen kann, ist hier die Formel zur Berechnung der Weltkoordinaten der Sprunggelenke (*ls* und *rs*) angegeben:

$$\mathbf{w}_{ls} = \begin{bmatrix} x_{ls} \\ y_{ls} \\ z_{ls} \\ 1 \end{bmatrix} = \mathbf{T}_{0lu} \cdot \begin{bmatrix} 0 \\ 0 \\ -L_u \\ 1 \end{bmatrix} \quad (3.37)$$

$$\mathbf{w}_{rs} = \begin{bmatrix} x_{rs} \\ y_{rs} \\ z_{rs} \\ 1 \end{bmatrix} = \mathbf{T}_{0ru} \cdot \begin{bmatrix} 0 \\ 0 \\ -L_u \\ 1 \end{bmatrix} \quad (3.38)$$

### 3. Modellierung des menschlichen Skeletts

Im Koordinatensystem der Unterschenkel besitzen die Sprunggelenke die lokalen Koordinaten  $[0, 0, -L_u, 1]^T$ , wobei  $L_u = 440[mm]$  die Länge des Unterschenkels ist.

Mit diesen Gleichungen lässt sich die Vorwärts Kinematik der sechs Gelenkpunkte berechnen. Das bedeutet, ein Zustandsvektor  $\mathbf{x}$  wird vorgegeben und daraus die Weltkoordinaten der sechs Gelenkpunkte  $\mathbf{w}_i$  berechnet. Die Vorwärts Kinematik eignet sich hervorragend zur Visualisierung von Bewegungsdaten. Für die Modellapproximation ist aber der umgekehrte Weg viel interessanter, also mittels inverser Kinematik einen unbekannten Zustandsvektor zu schätzen. Dies wird in Abschnitt 6.2 beschrieben.

## 3.6. Gelenkwinkelgrenzen

Für die Gelenke des Skelettmodells werden die Gelenkwinkelgrenzen als Vorwissen eingeführt. Damit werden viele kinematisch unmögliche Bewegungen und so potenzielle Mehrdeutigkeiten des Bildauswertungsprozesses ausgeschlossen.

Die Winkelbereiche der Gelenke variieren von Proband zu Proband erheblich. Es wäre sehr aufwendig, für jeden Probanden seine persönlichen Gelenkwinkelgrenzen vor jeder Messung zu bestimmen, zumal die Gelenkwinkelgrenzen lediglich zum Ausschluss absolut unmöglicher Modellkonfigurationen verwendet werden sollen. Für jeden Gelenkwinkel wird deshalb eine minimale und eine maximale Grenze angegeben, die dem Modell zwar die notwendige Bewegungsfreiheit lässt, gleichzeitig aber den Suchraum auf vernünftige Weise einschränkt. Die genauen Werte sind eher unkritisch, weil die Gelenkwinkelgrenzen nicht als Nebenbedingungen in der Ausgleichung formuliert wurden. Sie werden lediglich bei der Anwendung der Lösung nach einer Ausgleichung überprüft, um absolut unmögliche Körperhaltungen während des Iterationsprozesses zu verbieten. Sollte selbst bei der letzten Iteration eine Gelenkwinkelgrenze überschritten werden, so deutet dies auf einen Fehler bei der Ausgleichung hin.

Selbst bei Angabe der Gelenkwinkelgrenzen im Zustandsraum besteht immer noch die Möglichkeit der Kollision der eigenen Körperteile im Ortsraum. Diese Fälle können durch eine Kollisionserkennung detektiert und behandelt werden. Dafür sind effektive Algorithmen aus der Computeranimation bekannt. In dieser Arbeit wurden sie jedoch nicht implementiert, weil der Aufwand dafür den Nutzen einer Kollisionserkennung bei weitem übersteigt. Es müssten für jede der verwendeten Formbeschreibungen eigene Algorithmen der Kollisionserkennung entwickelt werden, um dann eine lediglich kleine Reduktion des Suchraumes zu erhalten. Vor allem ist ungeklärt, wie die Ergebnisse der Kollisionserkennung in dem hier verwendeten Ansatz zur Modellapproximation als Nebenbedingungen verwendet werden können.

## 4. Modellierung der visuellen Erscheinung

Mit dem zuvor beschriebenen Skelettmodell ist die Repräsentation der Bewegung von Körperteilen möglich. Dies geschieht zum Beispiel beim einfachen Skelettmodell durch direkte Bewegung der Gelenkpunkte oder beim kinematischen Skelettmodell indirekt durch Ansteuerung der Gelenkwinkel. Zur Bildauswertung fehlt neben einer Ansteuerung des Modells noch eine Repräsentation der konkreten Erscheinung des Probanden. Damit wird ein Soll-Ist-Vergleich von Modell und Proband im Kamerabild überhaupt erst möglich. Die visuelle Erscheinung besteht im Wesentlichen aus der Form, der Helligkeit (Grauwerte) und der Farbe der Oberfläche sowie der Interaktion mit dem Hintergrund der Szene.

Ein interessanter Punkt ist die Frage, wie detailliert die äußere Erscheinung eines Probanden eigentlich modelliert werden muss, damit ein Soll-Ist-Vergleich zwischen Modell und Bild zuverlässig erfolgt. Der notwendige Detailgrad hängt von der Metrik ab, die zum Vergleich von Modell und Bild verwendet wird. Ein Matchingverfahren, das Grauwert für Grauwert vergleicht, benötigt voraussichtlich eine deutlich genauere Modellierung als ein Verfahren, das auf abstrakteren Repräsentationsformen aufbaut. Im Folgenden soll es erst einmal um die prinzipiellen Möglichkeiten der Modellierung gehen.

Zunächst wäre sogar eine rein bildbasierte Modellierung der Erscheinung denkbar. Im einfachsten Fall lässt sich für jedes Körperteil in jeder Kamera ein 2D-Template verwenden. In [JBY96] wurden affine Deformationen der Templates erlaubt. Die Korrespondenzfindung erfolgt indirekt mit dem Ansatz des Optischen Flusses. Der Ansatz arbeitet jedoch rein bildbasiert ohne ein 3D-Skelettmodell. Deshalb ist der Einsatz auf monokular Systemen beschränkt. Allerdings zeigt dies, dass bereits wesentliche Elemente der Erscheinung eines Probanden durch diese einfache Art der Modellierung wiedergegeben werden können.

Wird für jede Kamera ein Satz dieser Templates vorgesehen, deren Lage durch ein gemeinsames 3D-Skelettmodell bestimmt wird, dürften sogar Auswertungen in Multi-Kamera-Systemen möglich werden (siehe Abbildung 4.1(a)). Eine geringfügig verbesserte Repräsentation wird erreicht, indem die Templates als Ebenen im 3D-Raum modelliert werden (siehe Abbildung 4.1(b)). Dadurch lassen sich Einflüsse der perspektivischen Projektion berücksichtigen. Ein wie auch immer geartetes Templatematching erlaubt in beiden Fällen die Korrespondenz zwischen Skelett, Template und schließlich Bild. Es ist zu erwarten, dass ein solches Verfahren versagt, wenn sich der Proband dreht, da die Rotation der Körperteile um ihre Achse durch ein einfaches Template nur schwer wiedergegeben werden kann. Eine Möglichkeit ist, das Aussehen der Templates je nach

#### 4. Modellierung der visuellen Erscheinung

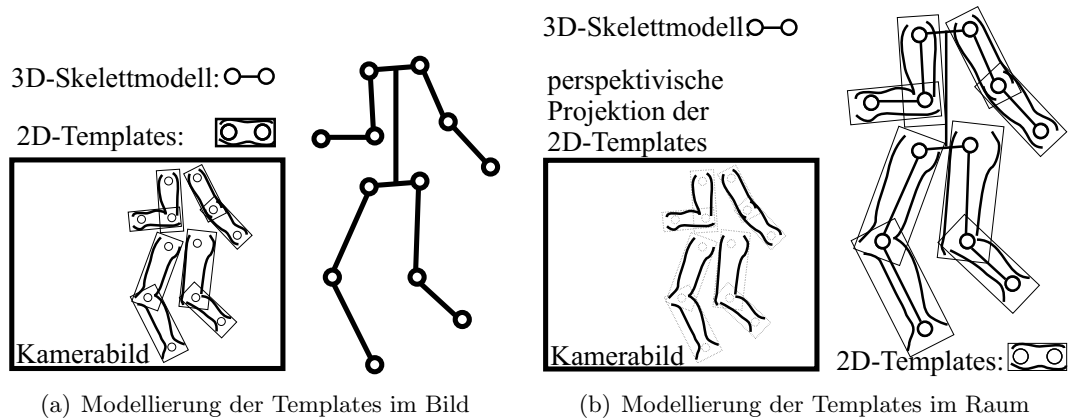


Abbildung 4.1.: Repräsentation der Erscheinung durch Templates. Mit einem Template-matching wird die Korrespondenz zwischen Modell und Bild hergestellt.

Blickrichtung der Kameras zu verändern.

Ohne Kleidung erscheinen die zu analysierenden Beine in Hautfarbe, also einfarbig mit wenig Struktur und demzufolge wenig Informationsgehalt. Deshalb sind die Texturinformationen, die sich aus einfarbigen Beinen gewinnen lassen, unzuverlässig. Sie entstehen im Wesentlichen durch Licht und Schatten und erzeugen dadurch sehr variable Bildanteile. Es existieren Methoden zur Modellierung von Licht und Schatten, sie basieren aber darauf, dass die genaue Geometrie der Szene bereits bekannt ist. Die genaue Bewegung des Probanden und damit auch die Geometrie der Szene in jedem Zeitschritt sind aber noch unbekannt. Aus diesem Grund wird auf eine Modellierung der Eigentextur der Oberfläche in dieser Arbeit verzichtet. Zumindest denkbar wäre aber ein iteratives Verfahren, das zunächst die Probandenbewegung näherungsweise bestimmt, um dann mit den daraus berechenbaren Geometriedaten eine Simulation von Licht und Schatten durchzuführen, die dann eine noch genauere Vermessung der Bewegungsdaten erlaubt.

Im Abschnitt 3 wurde der Zusammenhang zwischen der Anzahl der Modellparameter, der erreichbaren Genauigkeit und der verfügbaren Bildinformation bereits erläutert. Es wurde dargelegt, dass es ein Optimum der Parameteranzahl für die jeweils verfügbare Bildinformation gibt. Mehr Bildinformation kann demzufolge entweder zur Steigerung der Genauigkeit bzw. zur Erhöhung der Anzahl der Parameter des Modells verwendet werden. Gegenwärtig werden weitergehende Untersuchungen durchgeführt, den Texturanteil auf der Hautoberfläche zu vergrößern. Es bietet sich eine spezielle Kleidung mit einem bestimmten Muster an, was den klassischen auf Marker basierenden Systemen wieder sehr nahe kommt. Untersucht wird aber vor allem die Möglichkeit, zum Beispiel durch Projektion von strukturiertem Licht(Muster) den Informationsgehalt auf der Probandenoberfläche zu erhöhen. Konkrete Ergebnisse liegen jedoch noch nicht vor.

Für die Verwendung von Farbinformation gilt ähnliches, denn ein einfarbig-hautfarbener Körper dürfte die Differenzierung von Körperteilen allein basierend auf Farbinformation



erschweren. Anders ist die Situation bei der Verwendung bunter Kleidung oder farbiger Muster. In diesem System wurden sie jedoch nicht berücksichtigt.

Deshalb wird zur Bildauswertung im Wesentlichen Kanteninformation verwendet, die ihren Ursprung in der Form des Probanden hat. Der Schwerpunkt in diesem Kapitel liegt demzufolge auf der Formbeschreibung des 3D-Modells.

### 4.1. Möglichkeiten der 3D-Formbeschreibung

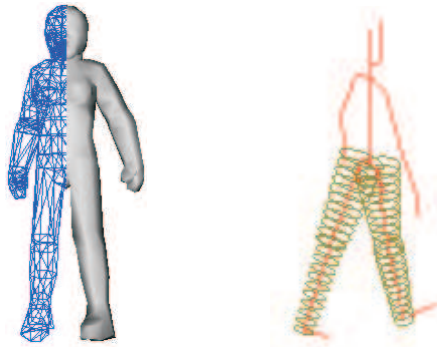


Abbildung 4.2.: Zwei Möglichkeiten zur Beschreibung der Körperform. Links: Formbeschreibung mit Dreiecksnetz [Lan99], Rechts: Formbeschreibung durch ein einfaches Konusmodell

Die Körperform kann durch Zusammensetzen geometrischer Primitiven, wie zum Beispiel Kugel, Kegelstümpfe und/oder Ellipsoiden erfolgen. In der Computeranimation haben sich aber komplexe Dreiecksnetze, die zum Beispiel durch ein Skelett deformiert werden, weitgehend durchgesetzt.

Meist wird über geometrische Primitiven, wie Zylinder oder Ellipsoide eine Annäherung an die Körperform versucht. Eine flexiblere Repräsentation ist durch ein Dreiecksnetz möglich. Es werden Punkte (Vertices) und Listen mit Dreiecken auf diesen Punkten angegeben. Einfache Mensch-Modelle lassen sich bereits ab 100 Dreiecken darstellen, diese könnten sogar von Hand generiert werden. Sofern ausreichend Information verfügbar ist, ist die Detailstufe der Formrepräsentation nach oben nur durch die verfügbare Rechenleistung und den Fähigkeiten des modellbildenden Prozesses begrenzt. In der Computeranimation wird diese Information vorher durch aufwendige Modellierungstechniken bereitgestellt, die einen erheblichen manuellen Aufwand erfordern.

Als Quelle für das Formmodell kann aber auch ein kommerzieller 3D-Body Scanner [Ste04] eingesetzt werden. Meistens kommt eines der aktiven Triangulationsverfahren (Lichtschnittverfahren) zum Einsatz. Zum Beispiel: Mehrere Laser projizieren eine Lichtebene, der Körper schneidet diese und es bildet sich ein heller Streifen auf der Körperoberfläche. Verschiedene Kameras bilden dann diesen Streifen ab. Somit lässt sich

#### 4. Modellierung der visuellen Erscheinung

eine dreidimensionale Kontur berechnen. Der gesamte Aufbau aus Kameras und Laserprojektoren wird dann durch Linearmotoren über die Körperoberfläche geführt und der Vorgang wiederholt. Aus den einzelnen Konturen lässt sich ein dichtes und detailreiches Oberflächenmodell berechnen.

Ein Multikamerasystem soll hauptsächlich zur Bewegungserfassung benutzt werden. Es könnte aber genauso zur Erfassung der Oberflächenform mit dem Volumenschnittverfahren (Shape-from-Silhouettes) [Nie99], [Wei00] verwendet werden. Mit den Kameras wird das Messobjekt oder der Proband aus verschiedenen Richtungen aufgenommen und eine Objektmaske (Silhouette) berechnet. Jeder Voxel eines Volumenmodells wird in jede Kamera projiziert und getestet ob er in allen Kameras innerhalb der Objektmaske liegt. Wenn ja, wird der Voxel auf einen Wert ungleich null gesetzt, andernfalls wird er gelöscht. Anschließend bleiben nur die Voxel im Volumenmodell gesetzt, die in allen Kameras innerhalb der Silhouette liegen. Ein schneller Marching-Cube-Algorithmus [LC87] extrahiert dann aus dem Voxelmmodell die Oberfläche des Körpervolumens. Allerdings lassen sich keine konkaven Oberflächen erfassen. Die Rekonstruktion von konkaven Details des Körpers ist mit diesem Verfahren nicht möglich.

## 4.2. Formbeschreibung durch Konturtemplates

Eine einfache Möglichkeit der Formbeschreibung stellt das Konturtemplate dar.

Am Beispiel eines Unterschenkels soll erläutert werden, wie sich die Stabilität und Genauigkeit der Approximation steigern lässt, wenn eine bessere Formbeschreibung verwendet wird. Zunächst erfolgt die Approximation lediglich in einem Kamerabild (2D), indem ein Template bestehend aus zwei aktiven Konturmodellen an das Bein eines Probanden approximiert wird. Die Kamera zeigt den Probanden in Seitenansicht.

Für dieses Experiment werden im Grunde die Gleichungen nach Abschnitt 3.2 verwendet. Da diesmal jedoch eine 2D-Modellierung erfolgt, werden die Gleichungen für die z-Koordinaten einfach ignoriert. Zur Beschreibung der Lage des Körperteils werden die beiden Gelenkpunkte des Knies  $\mathbf{p}_k = [x_k, y_k]^T$  und des Fußgelenkes  $\mathbf{p}_f = [x_f, y_f]^T$  verwendet. Ein gegebener Gelenkabstand  $s_0$  zwischen beiden wird wieder wie in Abschnitt 3.2 über eine Streckenbeobachtung eingehalten.

$$a \cdot \Delta x_k + b \cdot \Delta y_k - a \cdot \Delta x_f - b \cdot \Delta y_f = s_0 - s \quad (4.1)$$

$$a = \frac{x_k - x_f}{s_0} \quad ; \quad b = \frac{y_k - y_f}{s_0} \quad (4.2)$$

$$s = |\mathbf{p}_k - \mathbf{p}_f| \quad (4.3)$$

## 4.2. Formbeschreibung durch Konturtemplates

$$\begin{bmatrix} -a & -b & a & b \\ \mu & 0 & 0 & 0 \\ 0 & \mu & 0 & 0 \\ 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & \mu \end{bmatrix} \cdot \begin{bmatrix} \Delta \mathbf{p}_f \\ \Delta \mathbf{p}_k \end{bmatrix} = \begin{bmatrix} s_0 - s \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (4.4)$$

Punkte zwischen beiden Gelenkpunkten lassen sich interpolieren, indem eine Variable  $u$  von 0 bis 1 durchlaufen wird.

$$\mathbf{p}_m(u) = \mathbf{p}_k \cdot (1 - u) + \mathbf{p}_f \cdot u \quad (4.5)$$

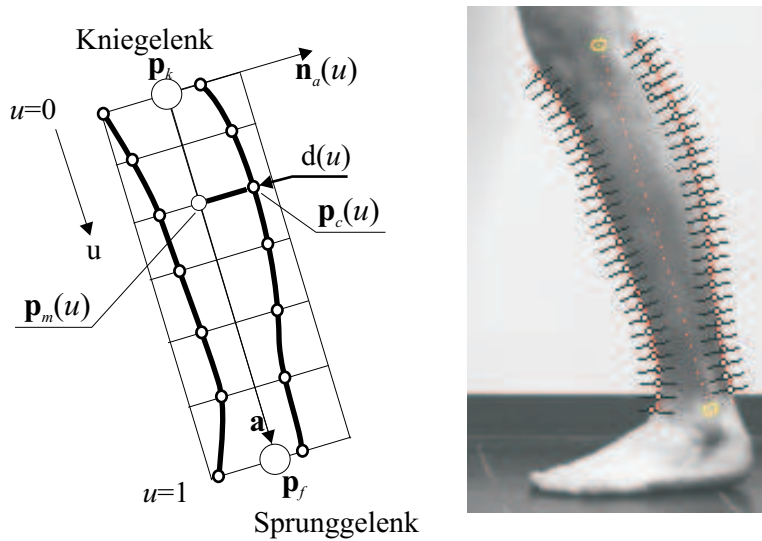


Abbildung 4.3.: Präzisere Modellierung der 2D-Form eines Körperteils

Der Richtungsvektor  $\mathbf{a} = \mathbf{p}_f - \mathbf{p}_k$  zwischen beiden Punkten lässt sich normieren und zum Aufbau eines Koordinatensystems für die Formbeschreibung verwenden. Abbildung 4.3 zeigt die Modellierung. Es werden zwei Konturen durch die Funktion  $d(u)$  beschrieben. Dabei gibt  $d(u)$  den Abstand der Kontur vom zugehörigen Mittelpunkt  $\mathbf{p}_m(u)$  auf der Konusachse an. Dafür definiert ein Normalenvektor  $\mathbf{n}_a = [x_{na}, y_{na}]^T$  (mit  $1 = |\mathbf{n}_a|$ ) rechtwinklig zur Körperteilachse  $\mathbf{a}$  die Richtung. Die Lage eines Konturpunktes kann dann berechnet werden mit:

$$\mathbf{p}_c(u) = \mathbf{p}_m(u) + \mathbf{n}_a \cdot d(u) \quad (4.6)$$

Zur Repräsentation der Form wird eine Funktion  $d(u)$  benötigt. Hier wurde ein B-Spline [Pie96] mit zum Beispiel  $N = 5$  Kontrollpunkten  $c_i$  verwendet. Ein Funktionswert des

#### 4. Modellierung der visuellen Erscheinung

B-Spline berechnet sich dann mit Hilfe der B-Spline-Basisfunktionen  $b_i(u)$ :

$$d(u) = \sum_{i=1}^N b_i(u) \cdot c_i \quad (4.7)$$

Die Gleichungen (4.5), (4.6) und (4.7) zusammengesetzt ergeben für einen Konturpunkt:

$$\mathbf{p}_c = \mathbf{p}_k \cdot (1 - u) + \mathbf{p}_f \cdot u + \mathbf{n}_a \cdot \sum_{i=1}^N b_i(u) \cdot c_i \quad (4.8)$$

Die partiellen Ableitungen werden zur Parameterschätzung benötigt:

$$\frac{\partial \mathbf{p}_c}{\partial x_k} = \begin{bmatrix} 1 - u \\ 0 \end{bmatrix} \quad ; \quad \frac{\partial \mathbf{p}_c}{\partial y_k} = \begin{bmatrix} 0 \\ 1 - u \end{bmatrix} \quad (4.9)$$

$$\frac{\partial \mathbf{p}_c}{\partial x_f} = \begin{bmatrix} u \\ 0 \end{bmatrix} \quad ; \quad \frac{\partial \mathbf{p}_c}{\partial y_f} = \begin{bmatrix} 0 \\ u \end{bmatrix} \quad (4.10)$$

$$\frac{\partial \mathbf{p}_c}{\partial c_i} = \begin{bmatrix} x_{na} \\ y_{na} \end{bmatrix} \cdot b_i(u) \quad (4.11)$$

Vor der Messung müssen die formbeschreibenden Kontrollpunkte  $c_i$  berechnet werden. Dazu werden die beiden Gelenkpunkte  $\mathbf{p}_c$  und  $\mathbf{p}_f$  manuell vorgegeben. Während der Messung hingegen wird die Form konstant gehalten und die Lage der Gelenkpunkte berechnet.

In beiden Fällen kann für jeden Konturpunkt das Suchstrahlverfahren nach [BI98] eingesetzt werden, um ausreichend Beobachtungsgleichungen aufzustellen. Dabei wird ausgehend von den Konturpunkten des Formmodells  $\mathbf{p}_c$  entlang einer gebenden Normalenrichtung  $\mathbf{n}_s$  nach der betragsmäßig größten Bildkante gesucht. Abbildung 4.4 zeigt links das Konturtemplate mit einem Punkt auf der Kontur. Rechts ist dieser Bereich vergrößert dargestellt. Zu sehen ist das Suchintervall. In diesem Intervall wird das Bild abgetastet. Es entsteht ein 1D-Grauwertprofil des Suchintervalls. Hier wird nun nach dem Punkt mit dem größten Gradienten gesucht. Im Ergebnis der Suche wird ein Abstand  $a$  entlang der Normalenrichtung  $\mathbf{n}_s$  erhalten.

Diese Suche wird für jeden Konturpunkt  $j$  durchgeführt. Zur Parameterschätzung wird die quadratische Summe der Abstände  $a_j$  minimiert.

$$e(c_1, c_2, \dots, c_i, \dots, c_N) = \sum_{j=1}^M (a_j)^2 \rightarrow \min \quad (4.12)$$

$$e(\mathbf{p}_k, \mathbf{p}_f) = (s_0 - |\mathbf{p}_k - \mathbf{p}_f|)^2 + \sum_{j=1}^M (a_j)^2 \rightarrow \min \quad (4.13)$$

Dabei minimiert Gl.(4.12) die Abstände  $a_j$  durch Veränderung der Form und Gl.(4.13) minimiert die Abstände  $a_j$  durch Veränderung der Lage des Templates. Der Term  $(s_0 -$

## 4.2. Formbeschreibung durch Konturtemplates

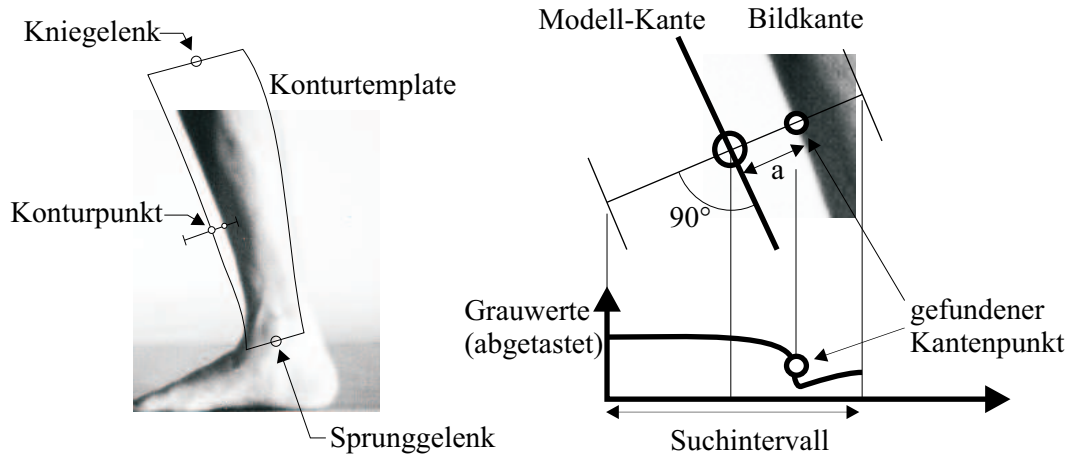


Abbildung 4.4.: Suchstrahlmethode nach [BI98]

$|\mathbf{p}_k - \mathbf{p}_f|^2$  sorgt dafür, dass ein vorgegebener Gelenkabstand  $s_0$  eingehalten wird (siehe Gl.(4.1)).

Die Parameterschätzung selbst erfolgt dann durch Ausgleichsrechnung. Dabei wird für jeden Konturpunkt eine der folgenden Beobachtungsgleichungen verwendet, je nachdem, ob die Form oder die Lage des Templates berechnet werden soll.:

$$\mathbf{n}_{sj}^T \cdot \left[ \frac{\partial \mathbf{p}_c}{\partial c_1} \cdot \Delta c_1 + \frac{\partial \mathbf{p}_c}{\partial c_2} \cdot \Delta c_2 + \dots + \frac{\partial \mathbf{p}_c}{\partial c_i} \cdot \Delta c_i + \dots + \frac{\partial \mathbf{p}_c}{\partial c_N} \cdot \Delta c_N \right] = a_j \quad (4.14)$$

$$\mathbf{n}_{sj}^T \cdot \left[ \frac{\partial \mathbf{p}_c}{\partial x_k} \cdot \Delta x_k + \frac{\partial \mathbf{p}_c}{\partial y_k} \cdot \Delta y_k + \frac{\partial \mathbf{p}_c}{\partial x_f} \cdot \Delta x_f + \frac{\partial \mathbf{p}_c}{\partial y_f} \cdot \Delta y_f \right] = a_j \quad (4.15)$$

Bei der Berechnung der Lage darf Gl.(4.1) nicht vergessen werden.

Für die Schätzung der Lage der Gelenkpunkte ist es wichtig, die in der Form des Konturtemplates codierte Information auch zu verwenden. Deshalb darf die Suche nicht für jeden Konturpunkt in die selbe Richtung erfolgen, weil durch Wiederverwendung der selben Normalenrichtung (zum Beispiel indem einfach  $\mathbf{n}_a$  als Suchrichtung verwendet wird) die Beobachtungsgleichungen linear abhängig werden. Die Berechnung der Normalenrichtung  $\mathbf{n}_s$  kann durch numerische Differenzierung benachbarter Konturpunkte erfolgen.

In Abbildung 4.5 wurde die Lage des Konturtemplates berechnet, die geschätzten Gelenkpunktkoordinaten wurden durch je drei Konfidenzellipsen mit skalierten Standardabweichungen (Sigma: 3, 30 und 300) markiert. Links wurden die Kontrollpunkte  $c_i$  lediglich näherungsweise vorgegeben. Es fehlt also noch an einer genauen Formbeschreibung und es ist zu sehen, dass dadurch die Lage der Gelenkpunkte nur sehr ungenau geschätzt werden kann. Die Iterationen wurden auch nicht richtig stabil und das Konturtemplate 'rutscht' nach oben. Das liegt vor allem daran, dass die Normalenvektoren der Konturen nahezu in die selbe Richtung zeigen.

#### 4. Modellierung der visuellen Erscheinung

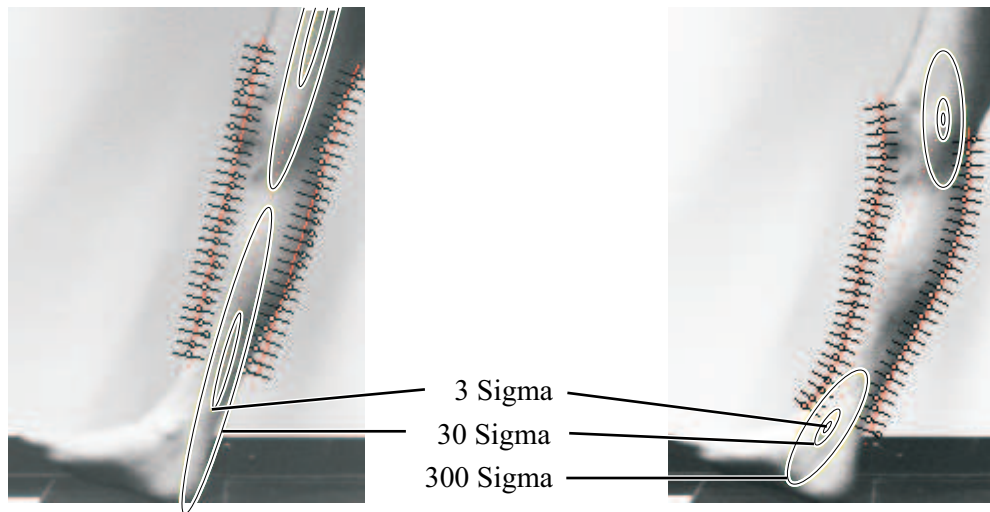


Abbildung 4.5.: Konfidenzellipsen der Gelenkpunkte nach Approximation des Konturtemplates, links sehr ungenaue Formbeschreibung (durch zwei einfache Linien), genaue Formbeschreibung durch ein B-Spline Polynom

Rechts in Abbildung 4.5 ist das Ergebnis zu sehen, nachdem die Kontrollpunkte  $c_i$  durch Approximation an Kantenpunkte im Bild berechnet wurden. Erst durch die genauere Formbeschreibung entstehen ausreichend unabhängige Gleichungen und es lassen sich die Gelenkpunktkoordinaten genau berechnen. Zu erkennen ist dies an deutlich kleineren Konfidenzellipsen auf der rechten Seite des Bildes.

Generell besitzen Konturtemplates den Nachteil, dass sie lediglich die Form aus einer bestimmten Kamerablickrichtung repräsentieren können. Um also die Form eines Unterschenkels aus verschiedenen Richtungen (Multikamerasystem) repräsentieren zu können, müsste für jede Kamera ein Satz Konturtemplates erzeugt werden. Für die Bildsequenzauswertung funktioniert der Ansatz aber nur dann gut, wenn sich die Orientierung des Probanden relativ zur Kamera nicht wesentlich ändert. Das ist zum Beispiel bei der Ganganalyse der Fall, wenn der Proband lediglich in einer Richtung auf einer geraden Bewegungsbahn läuft. Zu überlegen wäre, ob sich die Form der Konturtemplates nicht in Abhängigkeit vom Kamerablickwinkel beschreiben ließe, um auch Rotationen des Probanden zuzulassen. Sehr groß wäre der Aufwand dafür voraussichtlich nicht.

### 4.3. Formbeschreibung durch Kegelstümpfe

Geometrische Primitiven sind unter anderem Zylinder, (elliptische) Kegelstümpfe, Kugeln und Ellipsoiden. Sie dienen der rigiden Beschreibung von Körperteilen. In vielen modellbasierten Ansätzen, die auf die Verfolgung bewegter Menschen zielen, werden die Körperteile durch Zylinder oder Kegelstümpfe approximiert. Sie werden gern verwen-

### 4.3. Formbeschreibung durch Kegelstümpfe

det, weil die Projektion ins Bild einfach und schnell zu realisieren ist. Berechnungen der Tangentialebenen, der sichtbaren Grenzkanten zwischen Vordergrundobjekt und Hintergrund, ergeben sich durch mehr oder weniger einfache Operationen.

[Wac97] verwendet einen geraden elliptischen Kegelstumpf als Modell der Körperteile. Dort wurden die Gleichungen für die Projektion präzise hergeleitet. Ein etwas vereinfachter Weg soll hier gezeigt werden. Die Radien der Grundflächen ( $r_o$  und  $r_u$ ) beschreiben die

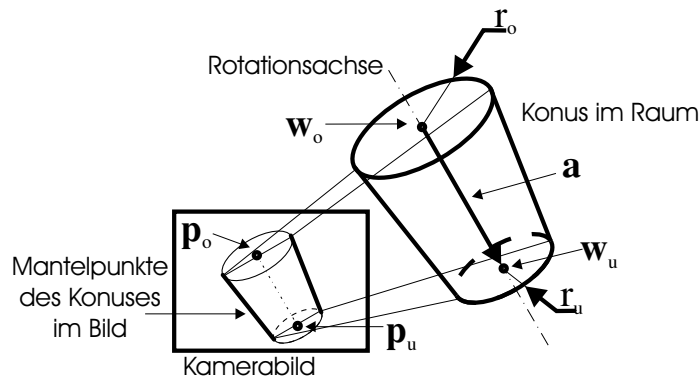


Abbildung 4.6.: Wird ein Konus ins Bild projiziert, werden zur Modellapproximation nur die Silhouettenpunkte des Konusmantels verwendet.

Dicke des Kegelstumpfes. Seine Lage im 3D-Raum kann durch die beiden Mittelpunkte der Grundflächen  $w_o$  und  $w_u$  beschrieben werden. Sie legen gleichzeitig die Rotationsachse  $a = w_u - w_o$  des Konus fest. Abbildung 4.6 zeigt die einzelnen Größen. Verwendet man als Punktkoordinaten  $w_o$  und  $w_u$  die Gelenkpunkte des Skeletts, ergibt sich bereits ein sehr einfaches Modell der unteren Extremitäten. Dieses ist in Abbildung 4.7 zu sehen. Eine Modellierung der Körperform in der Nähe der Gelenke, also dem Übergang

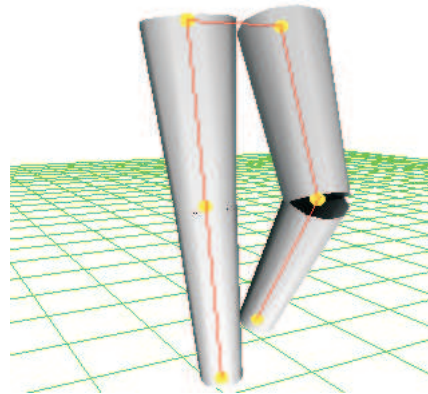


Abbildung 4.7.: Kinematisches Skelett der Beine nach Abschnitt 3.5 mit einer Formbeschreibung durch je einen Konus für jedes Beinsegment.

zwischen zwei Kegelstümpfen, ist mit diesem einfachen Modell zumindest schwierig und

#### 4. Modellierung der visuellen Erscheinung

wurde deshalb nicht realisiert. Weil die Form der FüÙe erheblich von der eines Kegelstumpfes abweicht, erscheint ihre Modellierung mit einem Konusmodell nicht sinnvoll.

Wird für einen Konus ein Parameterwert  $u = 0 \dots 1$  vorgegeben, können Radius  $r(u)$  und ein Punkt auf der Konusachse  $\mathbf{w}_a(u)$  durch Linearkombination ermittelt werden.

$$\mathbf{w}_a(u) = \mathbf{w}_o \cdot (1 - u) + (u) \cdot \mathbf{w}_u \quad (4.16)$$

$$r(u) = r_o \cdot (1 - u) + (u) \cdot r_u \quad (4.17)$$

Dabei sind  $r_o$  und  $r_u$  die Radien der Grundflächen des Kegelstumpfes.

Für die Bildauswertung werden die Mantelpunkte des Kegelstumpfes im Bild benötigt. Folgende Annahme erlaubt eine einfache Berechnung der Mantelpunkte  $\mathbf{w}_l(u)$  und  $\mathbf{w}_r(u)$  im 3D-Raum. Wenn der Abstand der Kamera groß im Verhältnis zum Konusradius ist, dann lässt sich eine vereinfachte Formel zur Berechnung der Mantelpunkte formulieren. In Abbildung 4.8 sind die geometrischen Zusammenhänge verdeutlicht.

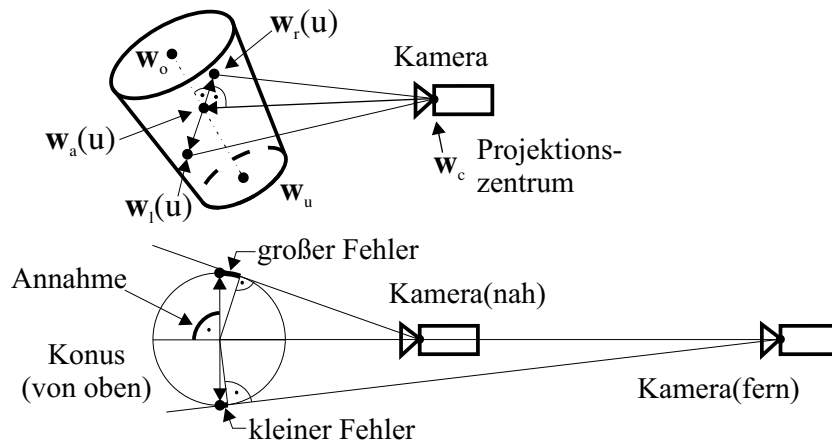


Abbildung 4.8.: Berechnung der 3D-Mantelpunkte  $\mathbf{w}_l(u)$  und  $\mathbf{w}_r(u)$  aus der Konusachse, dem Sehstrahl zwischen  $\mathbf{w}_a(u)$  und dem Projektionszentrum  $\mathbf{w}_c$  der Kamera bei gegebenem Parameterwert  $u$ .

Gesucht sind die Richtungen  $\mathbf{d}_{ml}$  und  $\mathbf{d}_{mr}$  zwischen Mittelpunkt und den beiden Mantelpunkten  $\mathbf{w}_l(u)$  und  $\mathbf{w}_r(u)$ . Angenommen, der Konusradius  $r(u)$  ist klein im Verhältnis zum Abstand  $|\mathbf{w}_a(u) - \mathbf{w}_c|$  zur Kamera, dann liegen  $\mathbf{w}_a(u)$ ,  $\mathbf{w}_l(u)$  und  $\mathbf{w}_r(u)$  näherungsweise auf einer Geraden, die orthogonal zur Konusachse und dem Sehstrahl steht. Somit lässt sich Richtung der Vektoren  $\mathbf{d}_{ml}$  und  $\mathbf{d}_{mr}$  aus dem Kreuzprodukt des Sehstrahles  $(\mathbf{w}_a(u) - \mathbf{w}_c)$  mit der Konusachse  $(\mathbf{w}_u - \mathbf{w}_o)$  berechnen.

$$\mathbf{d}_m = (\mathbf{w}_u - \mathbf{w}_o) \times (\mathbf{w}_a(u) - \mathbf{w}_c) \quad (4.18)$$

$$\mathbf{d}_{mn} = \mathbf{d}_m \cdot \frac{1}{|\mathbf{d}_m|} \quad (4.19)$$

$$\mathbf{w}_l(u) = \mathbf{w}_a(u) + \mathbf{d}_{mn} \cdot r(u) \quad (4.20)$$

$$\mathbf{w}_r(u) = \mathbf{w}_a(u) - \mathbf{d}_{mn} \cdot r(u) \quad (4.21)$$



#### 4.4. Formbeschreibung durch Dreiecksnetze

Für die Modellapproximation werden die partiellen Ableitungen der Mantelpunkte nach den beiden Mittelpunkten der Grundflächen  $\mathbf{w}_o$  und  $\mathbf{w}_u$  benötigt. Weil die Modellapproximation ohnehin iterativ erfolgt, reichen Näherungslösungen aus. Die Differenziale sind nur für die x-Koordinate von  $\mathbf{w}_o$  und  $\mathbf{w}_u$  dargestellt, sie sind für die anderen Koordinatenrichtungen identisch.

$$\frac{\partial \mathbf{w}_l}{\partial w_{ox}} \approx (1 - u) \quad (4.22)$$

$$\frac{\partial \mathbf{w}_l}{\partial w_{ux}} \approx u \quad (4.23)$$

$$\frac{\partial \mathbf{w}_r}{\partial w_{ox}} \approx (1 - u) \quad (4.24)$$

$$\frac{\partial \mathbf{w}_r}{\partial w_{ux}} \approx u \quad (4.25)$$

Sollen bei der Modellapproximation auch die Radien der Grundflächen des Kegelstumpfes ermittelt werden, so benötigt man die folgenden partiellen Ableitungen der Mantelpunkte:

$$\frac{\partial \mathbf{w}_l}{\partial r_o} = \mathbf{d}_{mn} \cdot (1 - u) \quad (4.26)$$

$$\frac{\partial \mathbf{w}_l}{\partial r_u} = \mathbf{d}_{mn} \cdot (u) \quad (4.27)$$

$$\frac{\partial \mathbf{w}_r}{\partial r_o} = -\mathbf{d}_{mn} \cdot (1 - u) \quad (4.28)$$

$$\frac{\partial \mathbf{w}_r}{\partial r_u} = -\mathbf{d}_{mn} \cdot (u) \quad (4.29)$$

#### 4.4. Formbeschreibung durch Dreiecksnetze

Dreiecksnetze werden in der Computergrafik seit langem verwendet, weil sie eine detaillierte Beschreibung der Oberflächenform und -farben erlauben. Die Grundlage bilden Punktwolken im 3D-Raum. Diese Stützstellen werden Vertices genannt. Indexlisten verbinden je drei Vertices zu einem Dreieck. Damit definieren sie Nachbarschaften und beschreiben das Dreiecksnetz. Werden diese Vertices eines jeden Dreieckes in definierter Reihenfolge angegeben, lässt sich mit Hilfe des Kreuzproduktes automatisch auch der Normalenvektor berechnen, da drei Punkte im 3D-Raum immer in einer Ebene liegen.

Das Einführen von Oberflächenfarben ist möglich. Jedes Dreieck erhält je Eckpunkt drei 2D-Texturkoordinaten, die einen dreieckigen 2D-Bildausschnitt festlegen. Er enthält die Bildinformationen, die auf das Dreieck projiziert wird. Dreiecksnetze werden selten manuell, sondern meistens automatisch erzeugt. Das kann durch Abtasten geometrischer Primitiven erfolgen (Zylinder, Ebenen, Freiformflächen durch Dreiecke approximieren) oder durch eine Flächenrückführung gescannter 3D-Oberflächendaten.

#### 4. Modellierung der visuellen Erscheinung

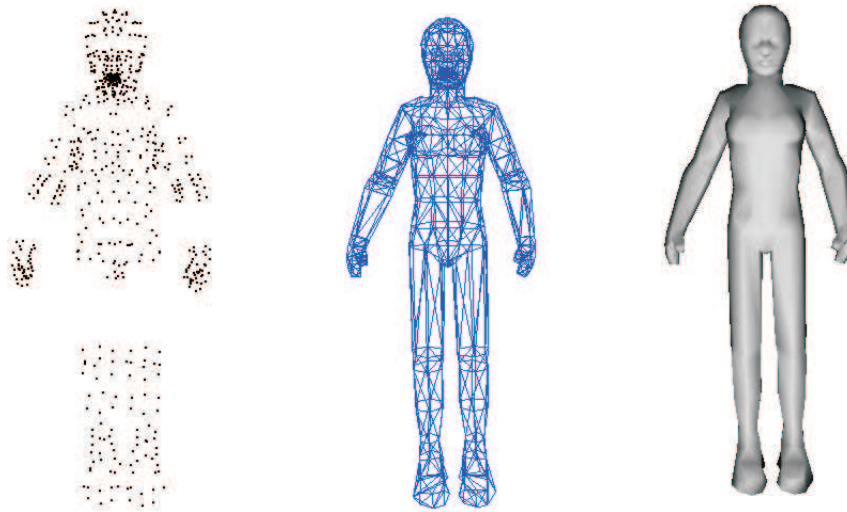


Abbildung 4.9.: Die Punktwolke (links) bildet die Stützstellen für eine Formbeschreibung durch ein Dreiecksnetz (mittig) nach [Lan99]. Rechts ist eine beleuchtete Darstellung zu sehen.

In der Bewegungsanalyse wurden Dreiecksnetze und die Möglichkeit, Grauwert- oder Farbinformationen auszuwerten, zum Beispiel von [WN99], [WL01] und [Wac97] eingesetzt. Prinzipiell repräsentiert ein Dreiecksnetz zunächst eine statische Formbeschreibung. Die Vertices des Netzes lassen sich jedoch entsprechend einer vorgegebenen Bewegung deformieren. Eine Methode dafür ist im folgenden Abschnitt 4.4.1 beschrieben. Finite-Elemente-Methoden (FEM) erlauben eine noch realistischere Simulation der Deformationen weichen Gewebes, wurden aber in dieser Arbeit nicht implementiert.

Der Vorteil einer detailreichen Modellierung durch Dreiecksnetze ist auch gleichzeitig ihr Nachteil. Denn jedes Vertices geht theoretisch mit drei unbekanntem Parametern in eine mögliche Formschatzung ein. Ein Ganzkörpermodell ist aber mit weniger als 150 Vertices nicht zu modellieren. Eine Verringerung der Parameteranzahl wäre sinnvoll. Das könnte zum Beispiel dadurch geschehen, dass für jedes Vertices nur bestimmte Freiheitsgrade zugelassen werden (zum Beispiel Normalenrichtung). Nachbarschaftsbeziehungen zwischen den Vertices könnten dann eine glatte Oberfläche sicherstellen. In jedem Fall bleibt jedoch die Zahl der zu bestimmenden Parameter hoch.

Sollten dennoch pure Dreiecksnetze zur Formbeschreibung verwendet werden, müssten diese Parameter vor einer Messung für einen Probanden bestimmt werden. Das könnte ein kommerzieller Body-Scanner ermöglichen, er leistet die Erfassung der Körperform in einer bestimmten Körperhaltung mit einer sehr großen Anzahl von Punkten, die notfalls ausgedünnt werden können. Das reicht jedoch nicht aus. Derzeitig liefert ein Scannvorgang eine Punktwolke und eventuell noch Nachbarschaftsbeziehungen zwischen den Punkten. Zum einen fehlt noch jegliche Zuordnung der Punkte zu einem bestimmten Körperteil (lokales Körperteilkoordinatensystem). Zum anderen muss auch die innere

#### 4.4. Formbeschreibung durch Dreiecksnetze

Struktur des Skeletts erfasst werden. Dazu gehört zum Beispiel die genaue Lage (und Abstände) der Gelenkpunkte sowie ihrer Rotationsachsen innerhalb der Formbeschreibung. Dies ist jedoch mit einem Body-Scanner bisher nicht möglich. Unglücklicherweise sind Formbeschreibung und Skelettmodell voneinander abhängig, sodass weitere Nebenbedingungen gefunden werden müssen. Eine direkte Animation der gescannten Punktwolke und damit eines ggf. daraus abgeleiteten Dreiecksnetzes erscheint derzeit nicht sinnvoll.

Möglicherweise ließe sich dies durch Body-Scans verschiedener Haltungen des Probanden realisieren. Die derzeit verfügbaren Body-Scanner erfordern einen ruhig stehenden Probanden. Deshalb erhält man eine oder mehrere statische Formbeschreibungen. Die genauen Formvariationen der Oberfläche in einer Bewegung, wie Muskelkontraktion oder Bewegung von Fettgewebe können mit der derzeit verfügbaren Technik ebenfalls nicht erfasst werden.

Steht eine sehr große Anzahl von Probanden und ein Bodyscanner zur Verfügung, können aber zumindest Statistiken über deren Form aufgestellt werden. Mit einer Hauptachsentransformation (Eigenvektorzerlegung) lassen sich dann Eigenformen von Körperteilen berechnen, welche die wesentlichen Formänderungen der Probanden beschreiben. Durch Weglassen der höheren Eigenformen ergibt sich voraussichtlich eine erhebliche Parameterreduktion. Im Ramsis-Projekt<sup>1</sup> wurde dies realisiert.

Die Frage ist, wie weit sich der Aufwand zur Parameterbestimmung des Mensch-Modells für einen Probanden verringern lässt. Existiert bereits eine Bibliothek mit sehr häufig auftretenden Körperformen, reicht womöglich die Angabe von Figurtyp, Geschlecht und Alter, um ein grobes Körpertemplate auszuwählen, das durch Angabe von Größe und Gewicht genau genug beschrieben ist. Das Ramsis-System stand jedoch zur Projektlaufzeit nicht zur Verfügung, so ist zum einen die Abschätzung des Aufwandes schwierig und zum anderen kann keine Aussage über die Eignung des Systems für die Bildauswertung mit dem hier gewählten Ansatz gemacht werden. Es existieren aber bereits Arbeiten, die das Ramsis-Modell zur Bewegungsanalyse verwenden [SBB00] und [SB01].

Frei mit Quelltext verfügbar ist eine Implementierung eines weiblichen Oberflächenmodells von [Lan99]. Allerdings ist damit auch die Formbeschreibung festgelegt und Änderungen daran sind aus den genannten Gründen nicht mehr möglich. Diese Implementierung wurde aber genutzt, um die wesentlichen Prinzipien zur Modellapproximation mit Dreiecksnetzen zu demonstrieren. Für die eigentlichen Messungen ist dieses Modell allerdings nicht zu verwenden, weil eine Anpassung der Körperproportionen des [Lan99] Modells an einen real existierenden Probanden schwierig ist.

Obwohl einige theoretische Überlegungen angestellt werden, erfolgt die Formbeschreibung in dieser Arbeit nicht durch ein Dreiecksnetz, nicht zuletzt im Hinblick auf eine spätere Anwendung im medizinischen Bereich. Es sollen die Bewegungsparameter mit möglichst hoher Genauigkeit und nicht primär die Form des Probanden erfasst werden. Der notwendige Aufwand zur Formerfassung sollte demzufolge möglichst gering gehalten

---

<sup>1</sup>[http://www.human-solutions.com/produkte\\_ramsis\\_e.php](http://www.human-solutions.com/produkte_ramsis_e.php)

## 4. Modellierung der visuellen Erscheinung

werden. Die verwendete Formbeschreibung soll nur so detailliert sein, dass die Genauigkeit der zu messenden Bewegungsgrößen nicht wesentlich zu verringert wird.

### 4.4.1. Skelett Deformation

Details zur Skelett Deformation und Quelltexte in OpenGL findet man in [Lan99]. Am Ende von Abschnitt 3.4 wurde beschrieben, wie Punkte aus dem Weltkoordinatensystem in lokale Körperteilkoordinaten umgerechnet werden können. Dieser Ablauf lässt sich verwenden, um die Oberflächenform des 3D-Modells nicht in lokalen Koordinatensystemen angeben zu müssen, sondern im Weltkoordinatensystem. Das Verfahren wird in der Computergrafik als *Skelett Deformation* (engl.: Skeletal Deformation) bezeichnet. Es ist interessant, weil sich dann die Form zum Beispiel des Menschen über einen 3D-Body Scanner erfassen und dann entsprechend der gewünschten Haltung deformieren lässt. Damit bietet es eine gute Möglichkeit a priori Wissen in die Bildsequenzauswertung einfließen zu lassen. Dazu arbeitet man mit mindestens zwei Haltungen des Skeletts:

- Eine Grundstellung  $\mathbf{T}_{0a}(\mathbf{x}_{t_G})$ . Dabei bezeichnet  $\mathbf{x}_{t_G}$  den Zustandsvektor der Grundstellung.
- Eine aktuelle Haltung  $\mathbf{T}_{0a}(\mathbf{x}_t)$ . Dabei bezeichnet  $\mathbf{x}_t$  den Zustandsvektor der Haltung zum Zeitpunkt  $t$ .

Die Grundstellung beschreibt, welche Körperhaltung der Mensch zum Zeitpunkt  $t_G$  des statischen Scannvorganges eingenommen hatte. Die inversen Transformationsmatrizen  $\mathbf{T}_{0a}^{-1}(\mathbf{x}_{t_G})$  dieser Haltung erlauben die Umrechnung der gescannten Punktwolke vom Weltkoordinatensystem in die lokalen Koordinatensysteme der Körperteile.

Das Modell lässt sich jetzt in eine beliebige Haltung bringen. Dazu werden neue Gelenkwinkel in den Zustandsvektor eingetragen und die entsprechenden Transformationsmatrizen aktualisiert. Nun lassen sich die Punkte aus den lokalen Körperteilkoordinaten zurück in das Weltkoordinatensystem transformieren. So ist eine Deformation der gescannten statischen Oberflächenbeschreibung in jeden beliebigen Bewegungszustand möglich.

Allerdings müssen die eingescannten Punkte zuvor den einzelnen Körperteilen (lokalen Koordinatensystemen) zugeordnet worden sein. Es muss also festgelegt werden, welcher Punkt in welches lokale Koordinatensystem transformiert werden soll. Das könnte manuell über ein Benutzerinterface erfolgen. Dies ist jedoch aufwendig, deshalb wird ein anderer Weg vorgeschlagen. Man bringt das Mensch-Modell (Skelett und Oberfläche) in die Grundstellung. Dann wird die Oberflächenbeschreibung des Modells an die reale Punktwolke approximiert. So entsteht eine Oberflächenbeschreibung, deren Form der eingescannten Punktwolke entspricht und deren Vertices bereits korrekt den Körperteilen zugeordnet sind. Des Weiteren kann der komplizierte Schritt der Flächenrückführung, also die Berechnung einer Dreiecksvermaschung der gemessenen Punktwolken, entfallen.

## 4.5. Modellierung der Oberflächenfarben

Bei der Implementierung lassen sich Speicher und Matrixoperationen sparen, indem für die aktuelle Haltung gleich das Produkt aus inverser Transformationsmatrix der Grundstellung mit der Transformationsmatrix der aktuellen Haltung gebildet wird. Dieses Produkt erlaubt dann die direkte Umrechnung eines gescannten Punktes der Grundstellung  $\mathbf{w}_{t_G}$  in Körperteilkoordinaten  $\mathbf{l}_a$  (siehe Gl.(3.23)) und von dort in die aktuelle Haltung  $\mathbf{w}_t$  (siehe Gl.(3.24)).

$$\mathbf{w}_t = \mathbf{T}_{0a}(\mathbf{x}_t) \cdot \mathbf{T}_{0a}^{-1}(\mathbf{x}_{t_G}) \cdot \mathbf{w}_{t_G} \quad (4.30)$$

An den Grenzen zweier Körperteile  $a$  und  $b$  können die Vertices  $\mathbf{w}_{t_G}$  durch zwei Interpolationskoeffizienten  $u_a$  und  $u_b$  gewichtet transformiert werden, sodass sich die Transformation dieser Vertices durch eine Linearkombination der beiden benachbarten Koordinatensysteme ergibt. Dadurch ist eine verbesserte Modellierung von Deformationen der Oberfläche in der Nähe der Gelenke möglich [Vel00].

$$\mathbf{w}_t = [u_a \cdot \mathbf{T}_{0a}(\mathbf{x}_t)\mathbf{T}_{0a}^{-1}(\mathbf{x}_{t_G}) + u_b \cdot \mathbf{T}_{0b}(\mathbf{x}_t)\mathbf{T}_{0b}^{-1}(\mathbf{x}_{t_G})] \cdot \mathbf{w}_{t_G} \quad (4.31)$$

$$1 = u_a + u_b \quad (4.32)$$

Zur professionellen Mensch-Modellierung [Rat98] werden häufig mathematische Funktionen, wie B-Splinepolynome [Pie96] verwendet. Sie erlauben stetige Übergänge zwischen den Körperteilen und eine beliebige Verfeinerung der Abtastpunkte durch Dreiecksnetze. Ihre Topologie ist aber nur in aufwändiger manueller Kleinarbeit zu konstruieren.

All diese Maßnahmen dienen der besseren Formbeschreibung, d.h. die äußere Erscheinung wird realistischer modelliert. Allerdings bleibt die Frage offen, ob medizinisch relevante Daten dadurch tatsächlich genauer erfasst werden können. Dies ist eigentlich zu erwarten, aber voraussichtlich ist der Aufwand für einen täglichen Routineeinsatz zu hoch. Wie noch zu sehen sein wird, liegen die Hauptprobleme bei der markerlosen Ganganalyse woanders und es ist gerechtfertigt, an dieser Stelle zunächst keinen erhöhten Aufwand bei der Körpermodellierung zu betreiben.

## 4.5. Modellierung der Oberflächenfarben

Die Modellierung der Oberflächenfarben ist geeignet, die Qualität des Matchings zwischen Modell und Bild zu erhöhen (intensitäts- oder farbbasierte Matchingstrategien) oder die Ergebnisse einer Messung fotorealistisch zu präsentieren.

Für eine fotorealistische Darstellung werden die Farbwerte der Oberfläche benötigt. Je nach Art der aufnehmenden Kameras erfolgt die Repräsentation durch Farb- oder Grauwerte (kurz: Textur). Dazu wird das vorhandene Formmodell erweitert, die Textur mehrerer Dreiecke eines Oberflächenbereiches wird in einem 2D-Bild gespeichert. Die Texturinformation wird durch Rückprojektion eines korrekt positionierten Oberflächenmodells in die Kamerabilder gewonnen.

#### 4. Modellierung der visuellen Erscheinung

Die meisten optischen Verfahren zur Erfassung der Oberflächenform können gleichzeitig auch zur Erfassung der Texturinformation dienen. Je nach verwendetem Messverfahren wird neben den Bildern zur reinen Oberflächenvermessung ein weiteres Bild mit homogener Beleuchtung aufgenommen. Dann erfolgt die Oberflächenvermessung und es liegen die statische Punktwolke des Probanden und dazu kalibrierte Kamerabilder vor. Nun wird das 3D-Mensch-Modell an die Punktwolke approximiert und die genaue Oberflächenform bestimmt. Es erfolgt eine Verdeckungs- und Sichtbarkeitsanalyse. Das bedeutet, für jedes Dreieck des Oberflächenmodells wird das Kamerabild bestimmt, das die besten Texturinformationen liefern kann. Dazu muss überprüft werden, ob ein Dreieck vollständig im Kamerabild sichtbar ist. Wenn ja, muss das Kamerabild gefunden werden, welches das Dreieck in einem möglichst optimalen Winkel gesehen hat. Dazu bestimmt man den Normalenvektor des Dreieckes und berechnet den Winkel zwischen Normalenvektor und der Blickrichtung der Kamera. Es wird das Kamerabild verwendet, das einen möglichst kleinen Winkel besitzt. Es kann auch eine Mittelwertbildung oder eine Ausgleichsrechnung zwischen den Informationen verschiedener Kamerabilder erfolgen.

### 4.6. Bewertung der Methoden

Forminformation des Körpers kann auf verschiedenen Wegen repräsentiert werden. Die allgemein gültigste Modellierung wird durch das klassische Dreiecksnetz erreicht. Die große Anzahl von Freiheitsgraden des Netzes erfordert jedoch ein Messverfahren zur Gewinnung dieser Parameter, weil eine manuelle Erfassung nahezu unmöglich ist.

Im Hinblick auf eine spätere Anwendung im medizinischen Bereich scheint dies auch der Hauptnachteil der Dreiecksnetze zu sein. Es sollen ja die Bewegungsparameter mit möglichst hoher Genauigkeit und nicht primär die Form des Probanden erfasst werden. Allerdings ließe sich das vorhandene Multikamerasystem zur Ganganalyse auch zur Formbestimmung verwenden, sofern entsprechende Algorithmen zur Verfügung stehen. Die Entwicklung dieser Verfahren könnte Gegenstand einer weiteren Dissertation sein. Gegenwärtig ist der zu betreibende Aufwand für die Formfassung jedoch groß, dass eine Verwendung der Dreiecksnetze für die Ganganalyse nicht praktikabel erscheint.

Demgegenüber stehen einfachere Methoden mit deutlich weniger Freiheitsgraden. Bei der Formbeschreibung durch ein einfaches Konusmodell lassen sich alle Parameter noch mit einem Maßband oder gar während einer Messung aus den Bilddaten bestimmen. Natürlich können damit keine Details der Oberflächenstruktur repräsentiert werden, die aber für die angestrebte Anwendung auch nicht erforderlich erscheinen. Das Konusmodell bietet die Möglichkeit der Erweiterung, indem ein Zylinderkoordinatensystem mit einem variablen Radius verwendet wird.

Eine Repräsentation mit mittlerer Genauigkeit stellen 2D-Konturtemplates dar. Sie besitzen deutlich weniger Freiheitsgrade als ein Dreiecksnetz, können aber bereits relevante Merkmale der Oberflächenform im Bild repräsentieren.

## 5. Modellierung der Bildaufnahme

Ein wichtiges Element der Synthesekette ist das abbildende System. Es wird im Folgenden beschrieben.

### 5.1. Kameraaufbau

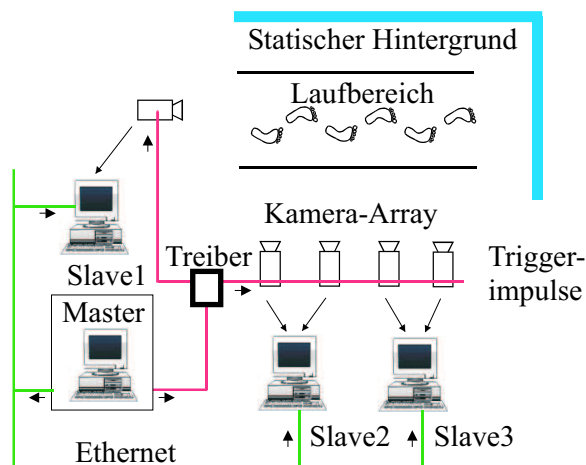


Abbildung 5.1.: Master-Slave-Anordnung zur synchronen Bildaufnahme

Zur Bildaufnahme wird eine Anordnung von mehreren Kameras verwendet, sie ist in Abbildung 5.1 zu sehen. Es werden höchstens zwei Kameras mit je einem Slave-Rechner verbunden. Ein Master-Rechner überträgt ASCII-Kommandos an seine Slaves, die daraufhin Operationen wie Bildaufnahme, Filterung, Speicherung und vieles mehr durchführen. Dadurch ist eine Skalierbarkeit des Systems durch eine gleichmäßige Auslastung jedes einzelnen Rechners gewährleistet. Einfache Operationen der Bildverarbeitung lassen sich sogar in Echtzeit auf jedem Slave berechnen. Dies erlaubt einen möglicherweise neuen Ansatz für die Kamerakalibrierung (siehe Anhang G), bei dem die Lokalisierung einer blinkenden Leuchtdiode (LED) auf jedem Slave erfolgt und lediglich die gemessenen Bildkoordinaten gespeichert werden.

Um eine synchrone Bildaufnahme aller Kameras zu gewährleisten, wurden diese vorher in den Extern-Trigger-Modus gebracht. Jede Kamera wartet dann auf die Low-Flanke

## 5. Modellierung der Bildaufnahme

eines Triggerimpulses. Auf das Signal hin, das durch eine Impulsgeber-Karte in einem der Computer erzeugt wurde, löst jede Kamera eine Bildaufnahme aus und überträgt das Bild an seinen Slave. Die Slaves halten die empfangenen Bilder zunächst im Hauptspeicher. Verkleinerte Versionen der aufgezeichneten Bildsequenzen können vom Master zur Darstellung angefordert werden, sodass der Bediener entscheiden kann, ob die Bildaufnahme gelungen ist und ob er diese Bildsequenzen dann verwerfen, speichern oder auswerten will.

Überlegungen zur benötigten Bildfrequenz wurden bereits in Abschnitt 2.1.2 angestellt.

### 5.2. Kameramodell

Für die Grundlagen der Kameramodellierung und allgemeine fotogrammetrische Probleme sei auf [Luh00], [AK89] und [Kra97] verwiesen. Es ist möglich, grundlegende Abbildungseigenschaften der Aufnahmebedingungen zu modellieren. Soll eine fotorealistische Darstellung der Szene erfolgen, so ist dies möglich. Details sind in [FDFH97], [Vel00] zu finden. In dieser Arbeit war das jedoch nicht der Schwerpunkt. Es wird lediglich ein einfaches Kameramodell verwendet.

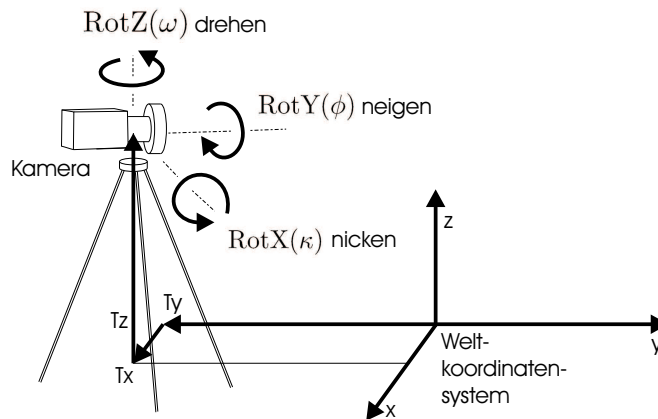


Abbildung 5.2.: Externe Parameter des Kameramodells (Translation und Rotation)

Zunächst muss für jede Kamera ein lokales Koordinatensystem, das Kamerakoordinatensystem, definiert werden. Dabei kann das Konzept der Transformationsketten aus Abschnitt 3.4 verwendet werden. Zunächst werden drei Transformationen (Translation) durchgeführt, um die Position der Kamera im Raum zu beschreiben. Die Angabe der Transformationsparameter erfolgt dabei im Weltkoordinatensystem, damit ihre Interpretation erleichtert wird.

Drei Transformationen (Rotation) erlauben der Kamera, in eine bestimmte Richtung zu schauen. Immer wenn die Orientierung durch Rotationsmatrizen dargestellt wird, deren Elemente unter Angabe von drei Winkeln gewonnen werden, kann es zu Singularitäten



kommen. Dies ist zum Beispiel dann der Fall, wenn einer der drei Winkel so groß geworden ist, dass zwei Rotationsachsen nahezu in die selbe Richtung zeigen. Sowohl bei dem Mensch-Modell als auch bei den Kameras stellt sich dieses Problem. Es wird Gimbal-Lock genannt. Bei unbewegten Kameras sind während der Messung keine Probleme zu erwarten. Der Verlust eines Freiheitsgrades in bestimmten Konfigurationen kann aber zu Problemen während der Kamerakalibrierung führen, die vor der Messung erfolgen muss. Das Problem lässt sich durch Verwendung von Quaternionen<sup>1</sup> endgültig lösen. Allerdings sind diese Parameter schwer zu interpretieren. Bedeutend einfacher ist es, die Rotationsreihenfolge so zu wählen, dass es bei üblicher Kamerageometrie und der in Kapitel 3.1 gezeigten Festlegung des Koordinatensystems (alle Kameras schauen nahezu horizontal) zu keinen Singularitäten kommt. Für die Rotation bietet sich deshalb die folgende Reihenfolge an:

1. RotZ( $\omega$ ) Drehen der Kamera
2. RotY( $\phi$ ) Neigen der Kamera
3. RotX( $\kappa$ ) Nicken der Kamera
4. RotX(90 °) Aufrichten der Kamera dadurch bleibt  $\kappa$  in der Nähe von null

Beschrieben werden Translation und Rotation mit einer kinematischen Kette nach Abschnitt 3.4 und 3.4.1.

$$\begin{aligned} \mathbf{T}_{0k}(\mathbf{x}) &= \mathbf{T}_{0k}([x, y, z, \omega, \phi, \kappa]^T) \\ &= \mathbf{E}_{tx}(x) \cdot \mathbf{E}_{ty}(y) \cdot \mathbf{E}_{tz}(z) \cdot \mathbf{E}_{rz}(\kappa) \cdot \mathbf{E}_{ry}(\phi) \cdot \mathbf{E}_{rx}(\omega) \cdot \mathbf{E}_{rx}(90^\circ) \end{aligned} \quad (5.1)$$

Dabei werden die Elementarmatrizen  $\mathbf{E}_{..}(\cdot)$  gemäß Anhang D bereitgestellt.

### 5.2.1. Projizieren von Punkten

Nachdem die Transformationskette aufgebaut wurde, kann mit Gl.(3.24) oder Gl.(5.2) ein beliebiger Punkt des Weltkoordinatensystems  $\mathbf{w}$  in das lokale Koordinatensystem  $\mathbf{l}_k = [l_{kx}, l_{ky}, l_{kz}, 1]^T$  der Kamera  $k$  gebracht werden. Gl.(5.3) beschreibt ein einfaches lineares Modell einer Lochkamera. Es berechnet die perspektivische Projektion  $\mathbf{p}_p$  des Punktes  $\mathbf{l}_k$ :

$$\mathbf{l}_k = [l_{kx}, l_{ky}, l_{kz}, 1]^T = \mathbf{T}_{0k}^{-1} \cdot \mathbf{w} \quad (5.2)$$

$$\mathbf{p} = \begin{bmatrix} p_x \\ p_y \end{bmatrix} = \mathbf{h} + \frac{c}{l_{kz}} \cdot \begin{bmatrix} l_{kx} \\ s_y \cdot l_{ky} \end{bmatrix} \quad (5.3)$$

Gl.(5.3) verwendet die Kammerkonstante  $c$  in Pixel, berücksichtigt die Koordinaten des Hauptpunktes  $\mathbf{h}$  und erlaubt eine Skalierung des Bildes in y-Richtung. Dies könnte interessant sein, wenn mit TV-Kameras im Zeilensprung-Verfahren gearbeitet wird, aber

<sup>1</sup> Zur Beschreibung von Rotationen im 3D-Raum mit Hilfe von Quaternionen sind vier Zahlen erforderlich. Diese sind jedoch voneinander abhängig, was ihre Nutzung in dem hier verwendeten Ansatz erschwert.

## 5. Modellierung der Bildaufnahme

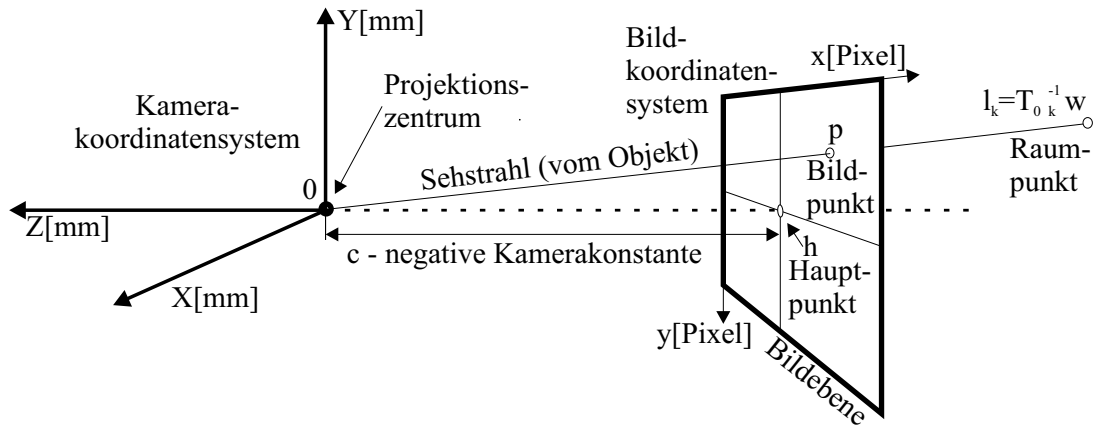


Abbildung 5.3.: Weitere Parameter des Kameramodells

lediglich Halb-Bilder ausgewertet werden. Dadurch lassen sich Bildfrequenzen von bis zu 60 Hz mit reduzierter vertikaler Auflösung realisieren. Dies ist durchaus sinnvoll, da bei der Ganganalyse keine sehr hohe vertikale Bildauflösung benötigt wird, denn die meisten Konturen des menschlichen Körpers verlaufen von oben nach unten und es kommt im wesentlichen auf eine gute horizontale Bildauflösung an. Der Skalierungsfaktor beträgt bei normalen Laborkameras (noninterlaced, quadratische Pixel) etwa  $s_y = -1.0$ , wenn die erste Zeile von oben und  $s_y = 1.0$ , wenn die erste Zeile von unten gezählt wird. Auf eine Modellierung der Verzeichnung des Objektivs wird in dieser Arbeit zunächst verzichtet, sie ist aber mit relativ wenig Aufwand zu implementieren.

### 5.2.2. Projizieren von Differenzialen

Zwei Probleme müssen gelöst werden: zum einen die Kalibrierung der Kameras vor der eigentlichen Messung und zum anderen die Approximation des Mensch-Modells an den Bildinhalt während der Messung. Die Kamerakalibrierung erfolgt normalerweise getrennt vor den eigentlichen Messungen.

Für die Kamerakalibrierung muss das Kameramodell nach den einzelnen Kameraparametern abgeleitet werden. Dies und die Kalibrierung selbst sind im Anhang G beschrieben.

Für die Modellapproximation während einer Messung müssen die Bildkoordinaten eines Punktes des Modells nach den Freiheitsgraden des Modells abgeleitet werden. Das Modell des Probanden  $\mathbf{w}(\dots)$  berechnet die Lage von Körperteilpunkten im Weltkoordinatensystem, wenn ein Zustandsvektor  $\mathbf{x}$  gegeben ist. Mit Hilfe des zuvor beschriebenen Kameramodells  $\mathbf{p}(\dots)$  können diese Punkte ins Bild projiziert werden.

$$\mathbf{p} = \mathbf{p}(\mathbf{l}_k(\mathbf{w}(\mathbf{x}))) \quad (5.4)$$

Für die Ableitung nach dem Zustandsvektor gilt demzufolge die Kettenregel. Für die

innere Funktion gilt:

$$\mathbf{l}_k = \mathbf{l}_k(\mathbf{w}(\mathbf{x})) \quad (5.5)$$

$$\frac{\partial \mathbf{l}_k}{\partial \mathbf{x}} = \frac{\partial \mathbf{l}_k}{\partial \mathbf{w}} \cdot \frac{\partial \mathbf{w}}{\partial \mathbf{x}} \quad (5.6)$$

Für die äußere Funktion:

$$\mathbf{p} = \mathbf{p}(\mathbf{l}_k(\dots)) \quad (5.7)$$

$$\frac{\partial \mathbf{p}}{\partial \mathbf{x}} = \frac{\partial \mathbf{p}}{\partial \mathbf{l}_k} \cdot \frac{\partial \mathbf{l}_k}{\partial \mathbf{x}} \quad (5.8)$$

Gl.(5.6) in Gl.(5.8) eingesetzt ergibt:

$$\frac{\partial \mathbf{p}}{\partial \mathbf{x}} = \frac{\partial \mathbf{p}}{\partial \mathbf{l}_k} \cdot \frac{\partial \mathbf{l}_k}{\partial \mathbf{w}} \cdot \frac{\partial \mathbf{w}}{\partial \mathbf{x}} \quad (5.9)$$

Die ersten beiden Terme in Gl.(5.9) lassen sich bereits aus dem Kameramodell ableiten. Aus Gl.(5.2) folgt der erste Term:

$$\frac{\partial \mathbf{p}}{\partial \mathbf{l}_k} = \begin{bmatrix} c/l_{kz} & 0 & -cl_{kx}/(l_{kz}^2) & 0 \\ 0 & cs_y/l_{kz} & -cs_y l_{ky}/(l_{kz}^2) & 0 \end{bmatrix} \quad (5.10)$$

Aus Gl.(5.3) folgt der zweite Term:

$$\frac{\partial \mathbf{l}_k}{\partial \mathbf{w}} = \mathbf{T}_{0k}^{-1} \quad (5.11)$$

Bleibt noch der dritte Term  $\frac{\partial \mathbf{w}}{\partial \mathbf{x}}$  offen. Er enthält die Ableitungen eines Punktes  $\mathbf{w}$  nach den Freiheitsgraden des 3D-Körpermodells. Dieser Term hängt natürlich stark vom verwendeten Modell ab und wo genau der Punkt am Modell 'befestigt' ist. Für ein Modell bestehend aus kinematischen Ketten ist im Anhang E ein automatischer Algorithmus zur Berechnung von  $\frac{\partial \mathbf{w}}{\partial \mathbf{x}}$  beschrieben.

Viel interessanter ist aber die Schlussfolgerung, die sich aus Gl.(5.9) ziehen lässt. Da sich zwei der drei Terme aus dem Kameramodell ergeben, kann man damit die Differentiale  $\frac{\partial \mathbf{w}}{\partial \mathbf{x}}$  aus dem Weltkoordinatensystem ins Kamerabild quasi projizieren. Das erleichtert die Erstellung eines automatischen Programmes zur Approximation eines beliebigen Modells an die Kamerabilder. Für einen Körperteilpunkt des Modells müssen seine Weltkoordinaten vorliegen und die Ableitungen  $\frac{\partial \mathbf{w}}{\partial \mathbf{x}}$  nach den Freiheitsgraden des Modells. Es lässt sich dann sowohl der Punkt selbst als auch seine Ableitungen ins Kamerabild projizieren. Letzteres geschieht durch eine einfache Matrixmultiplikation, weil sich die  $2 \times 4$  Matrix  $\frac{\partial \mathbf{p}}{\partial \mathbf{w}}$  bei statischen Kameras vorab berechnen und speichern lässt:

$$\frac{\partial \mathbf{p}}{\partial \mathbf{w}} = \begin{bmatrix} c/l_{kz} & 0 & -cl_{kx}/(l_{kz}^2) & 0 \\ 0 & cs_y/l_{kz} & -cs_y l_{ky}/(l_{kz}^2) & 0 \end{bmatrix} \cdot \mathbf{T}_{0k}^{-1} \quad (5.12)$$

$$\frac{\partial \mathbf{p}}{\partial \mathbf{x}} = \frac{\partial \mathbf{p}}{\partial \mathbf{w}} \cdot \frac{\partial \mathbf{w}}{\partial \mathbf{x}} \quad (5.13)$$

## 5. Modellierung der Bildaufnahme

In Abschnitt 6.2 wird gezeigt werden, dass sich die Ableitungen  $\frac{\partial \mathbf{w}}{\partial \mathbf{x}}$  zur Approximation des Modells an Punktwolken im Weltkoordinatensystem verwenden lassen. Es lassen sich nun Modellpunkte auch an Zielpunkte im Kamerabild approximieren. Dies wird in Abschnitt 6.3 gezeigt werden. Für eine Komponente  $x_i$  des Zustandsvektors  $\mathbf{x}$  schreibt sich Gl.(5.13)

$$\frac{\partial \mathbf{p}}{\partial x_i} = \frac{\partial \mathbf{p}}{\partial \mathbf{w}} \cdot \frac{\partial \mathbf{w}}{\partial x_i} \quad (5.14)$$

### 5.3. Erzeugung synthetischer Bilder

Die fotorealistische Darstellung von synthetischen Szenen ist aus der Computergrafik bekannt. Deshalb wird hier nur kurz erläutert, welche Erweiterungen für dieses Projekt vorgenommen werden. Im Gegensatz zur Computergrafik, bei der eine möglichst hohe Bildwiederholfrequenz auf einem Bildschirm angestrebt wird, muss für die Bildsequenzbewertung auf die synthetischen Bilddaten zugegriffen werden können. Das ist z.T. ein kritischer Aspekt, da sich die meisten Grafikkarten nur mit geringer Performance auslesen lassen. OpenGL bietet weiterhin die Möglichkeit, als Ausgabegerät eine Bitmap anzugeben [Rog95]. Damit kann auch der Hauptprozessor auf die synthetischen Bilddaten zugreifen. Allerdings waren derzeit auch mit diesem Verfahren keine sehr hohen Bildwiederholraten zu erreichen.

Nach der Kalibrierung des Messsystems stehen für jede Kamera die Kameraparameter der äußeren und inneren Orientierung (Hauptpunkt) zur Verfügung. Nun wird für jede physische Kamera eine virtuelle Kamera mit den selben Kameraparametern und einem eigenen Bildspeicher erzeugt. Das ermöglicht nun die maßstabsgerechte Synthese der 2D-Projektion des 3D-Modells. Ziel ist ja, durch Modifikation des Modells ein synthetisches Bild zu erzeugen, das dem realen Kamerabild optimal entspricht. Da die meisten Renderingsysteme keine Objektiv-Verzeichnung modellieren, müsste das synthetische Bild durch eine Warming-Funktion so verzerrt werden, dass es der realen Verzeichnung entspricht. Umgekehrt wäre es ebenfalls denkbar das reale Bild zu entzerren. Zum derzeitigen Stand des Projektes waren solche Genauigkeitsanforderungen jedoch nicht gestellt, sodass keine Verzeichnungskorrektur erfolgt. Zur Minimierung der Fehler wurden Objektive mit relativ großer Brennweite ( $\geq 10$  mm) und einer kleinen Verzeichnung verwendet.

Die derzeit verwendeten Algorithmen zum Vergleich von synthetischem Bild und dem realen Bild sind sehr einfach. Da die Oberfläche der Beine eines gehenden Probanden zu wenig Eigentextur oder Farbwerte besitzt, stehen für den Anwendungsfall Ganganalyse zu wenig Informationen zur Verfügung. Deshalb erfolgte eine Beschränkung des Vergleiches auf Bild-Merkmale. Es wurde lediglich eine Suche ausgehend von Kantenpunkten des Modells nach Kantenpunkten im realen Bild realisiert. Anders als bei vielen anderen Verfahren zum Vergleich von Bilddaten, erübrigt sich dadurch auch die Notwendigkeit, realistische Grauwert- oder Farbbilder zu synthetisieren.

### 5.3.1. Sichtbarkeit von Körperteilen

Auswahl eines Pixels im synthetischen Bild:

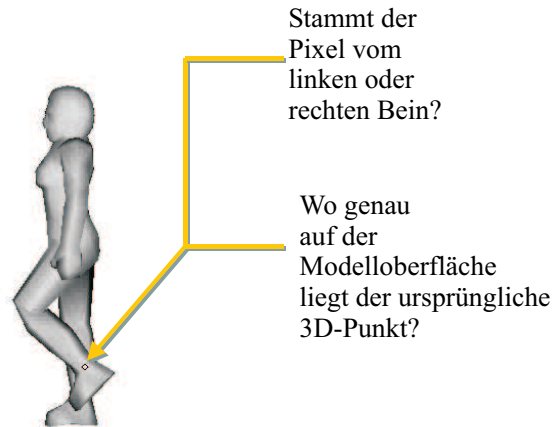


Abbildung 5.4.: Darstellung des Verdeckungsproblems

Ein großes Problem im Analyse-Synthese-Konzept ist die Beantwortung der Frage, welche Teile des Modells wo im Bild zu sehen sind und ob diese Bereiche überhaupt sichtbar, also nicht von anderen Teilen des Modells verdeckt sind. Es gibt einen sehr eleganten Weg, diese Frage zu beantworten[CMH03]. In jeder synthetischen Kamera wird nicht nur das fotorealistische Bild gerendert, sondern noch ein Indexpuffer und ein Interpolationspuffer.

Bei den meisten verfügbaren Rendering-Systemen erfolgt die Synthese der Bilddaten unter Verwendung des RGB-Farbraums mit den Grundfarben Rot (R), Grün (G) und Blau (B). Jede Primitive (wie Dreieck oder Linie) des Modells wird im Index-Bild mit einer eigenen Farbe gerendert. Das erste Dreieck zum Beispiel mit  $(R=0, G=0, B=1)$ . Das zweite Dreieck mit der Farbe  $(R=0, G=0, B=2)$  und so weiter. Je Farbkomponente stehen normalerweise 256 Farbwerte zur Verfügung. Im RGB Farbraum können damit also  $256 \cdot 256 \cdot 256 = 16777216$  verschiedene Dreiecke unterschieden werden. Mit diesem Trick ist es nun möglich, für jeden Pixel des realen Bildes auch das Dreieck zu bestimmen, das durch diesen Pixel abgebildet wird. Dazu wird die Adresse des Pixels im Indexpuffer berechnet und der Farbwert ausgelesen. Steht dort der Farbwert  $(R=0, G=0, B=2)$ , so wurde dieser Pixel von Dreieck 2 gerendert.

Dieses Verfahren ist mit minimalem Aufwand zu implementieren und liefert für alle Pixel gleichzeitig die korrekten Modell-Primitiven. Über Tabellen kann sehr schnell auf die entsprechenden Software-Objekte zugegriffen werden, die einzelne Bildregionen synthetisiert haben.

Damit das Verfahren funktioniert, muss das Rendering-System korrekt eingestellt werden. Normalerweise ändert das Rendering-System noch einmal alle vorgegebenen Farb-

## 5. Modellierung der Bildaufnahme

werte, um Beleuchtung, Materialeigenschaften und Schattenwurf zu simulieren. Diese Operationen müssen abgeschaltet werden, da sonst die vorgegebenen Farbwerte nicht mehr den Modell-Primitiven entsprechen würden!

Der Hintergrund wird im Index- und Interpolationspuffer mit dem Farbwert (R=0, G=0, B=0) gefüllt. Werden für jeden Pixel des Puffers 32 Bit verwendet, ist ein Zugriff auf einen Pixel des Puffers über einen ganzzahligen Datentyp<sup>2</sup> möglich. Das erlaubt dann eine sehr schnelle Berechnung der Silhouette des Modells, weil der Test, ob ein Pixel gleich oder ungleich Null ist, mit nur einem (unsigned long) statt mit drei Vergleichen (für jeweils R, G, B) durchgeführt werden kann. Pixel ungleich Null gehören zum Vordergrundobjekt und Pixel gleich Null gehören zum Hintergrund. Sehr leicht lassen sich dann auch die Konturen der Silhouette extrahieren. Sie liegen dort, wo zwei benachbarte Pixel jeweils zum Vordergrund und Hintergrund gehören.

Mit dem Indexpuffer können nun zwar die einzelnen Primitiven unterschieden werden, es kann jedoch noch keine Aussage über die Lage innerhalb einer Primitive getroffen werden. Für eine noch genauere Unterscheidung der Pixel einer Primitive wird ein Interpolationspuffer benötigt. In diesem Puffer wird wieder die gesamte Szene gerendert. Diesmal jedoch wird jedes Dreieck mit einem RGB-Farbverlauf gezeichnet. Der Farbverlauf eines Dreieckes wird durch die Farbwerte seiner Vertices bestimmt zum Beispiel:

1. Eckpunkt  $\mathbf{V}_1$  mit Farbwert (R=255, G= 0, B= 0)
2. Eckpunkt  $\mathbf{V}_2$  mit Farbwert (R= 0, G=255, B= 0)
3. Eckpunkt  $\mathbf{V}_3$  mit Farbwert (R= 0, G= 0, B=255)

Für Linien kann das in ähnlicher Weise erfolgen, es werden jedoch nur zwei Farbkomponenten benutzt.

Nun kann für jeden Pixel im realen Bild nicht nur die Modell-Primitive, sondern auch die ungefähre Lage innerhalb der Primitive (Dreieck oder Linie) berechnet werden. Dazu normiert man die Summe der gelesenen Farbwerte auf 1

$$\begin{aligned} r &= R/(R + G + B) \\ g &= G/(R + G + B) \\ b &= B/(R + G + B) \end{aligned} \tag{5.15}$$

Die drei floating-point Zahlen (r,g,b) können als Interpolationskoeffizienten verwendet werden. Durch Linearkombination der drei Vertices des Dreieckes kann jetzt der 3D-Raumpunkt ( $\mathbf{w}$ ) des Pixels näherungsweise gebildet werden.

$$\mathbf{w} = r \cdot \mathbf{w}_1 + g \cdot \mathbf{w}_2 + b \cdot \mathbf{w}_3 \tag{5.16}$$

Zu beachten ist jedoch, dass die Genauigkeit, mit der der Farbverlauf wiedergegeben wird, lediglich ganzzahlig ist und auch durch das verwendete Interpolationsverfahren

---

<sup>2</sup>in der Programmiersprache C zum Beispiel unsigned long

beim Rendering bestimmt wird. Das ist nicht immer bekannt und hängt von der Implementierung des Rendering-Systems ab. Es ist deshalb günstig, die Dreiecke im Modell nicht zu groß werden zu lassen, damit sie im Bild ebenfalls nur wenige Pixel bedecken.

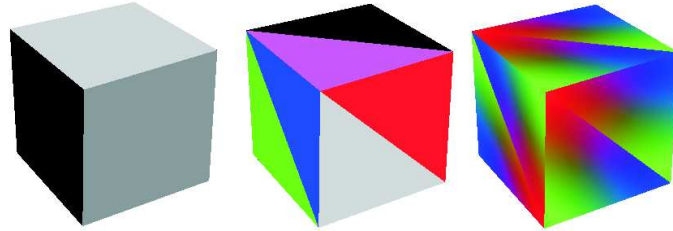


Abbildung 5.5.: Würfel im synthetischen Bild (Links), Indexpuffer (Mitte), Interpolationpuffer (Rechts)

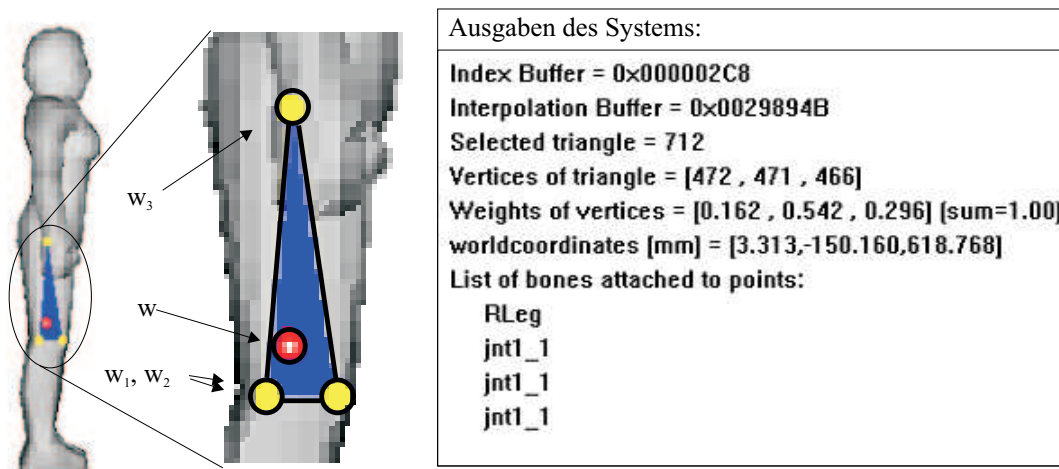


Abbildung 5.6.: Auswahl eines Pixels im synthetischen Bild, Ermittlung der angrenzenden Vertices des gefundenen Dreieckes, Berechnung des 3D-Punktes aus den Interpolationskoeffizienten

### 5.3.2. Erkennen von Selbstverdeckungen

Selbstverdeckungen sind ein Problem in der Bildauswertung. Oft ist es günstig zu wissen, wo im Bild ein Objekt ein anderes Objekt verdeckt, weil dies eine Vorhersage der Sichtbarkeit von Objekten im nächsten Zeitschritt erlaubt. Der Indexpuffer erlaubt auch die Erkennung von Objektgrenzen im synthetischen Bild, die zum Beispiel bei Selbstverdeckungen typisch sind.

Der Indexpuffer erfordert die Nummerierung der Dreiecke. Im Oberflächenmodell (Dreiecksnetz) besitzen die Dreiecke eine wohl definierte Nachbarschaft. Zwei Dreiecke sind dann benachbart, wenn sie zwei gemeinsame Vertices besitzen. Liegen im Indexpuffer

## 5. Modellierung der Bildaufnahme

zwei Dreiecke nebeneinander, die im Dreiecksnetz nicht benachbart liegen, dann muss diese neue Nachbarschaft durch den Abbildungsvorgang zustande gekommen sein. Also liegt eine Objektgrenze und mit hoher Wahrscheinlichkeit eine Selbstverdeckung vor, wenn sich eine neue Nachbarschaft ergeben hat, die nicht durch die Netztopologie gegeben ist.

Demzufolge muss ein Algorithmus zur Erkennung von Selbstverdeckungen zwei benachbarte Pixel des Indexpuffers überprüfen. Besitzen beide Pixel die selben Indexwerte, liegen sie auf dem selben Dreieck, es kann also keine Objektgrenze vorliegen. Besitzen sie zwei unterschiedliche Indexwerte, so könnte eine Objektgrenze vorliegen. Jetzt wird überprüft, ob die beiden Dreiecke im Dreiecksnetz benachbart sind. Sind sie benachbart, liegt keine Selbstverdeckung vor. Sind sie nicht benachbart, liegt eine Selbstverdeckung vor.

Wie kann die Frage der Nachbarschaft zweier Dreiecke effizient beantwortet werden? Nachdem ein Dreiecksnetz aufgebaut wurde, wird einmal eine Nachbarschaftsmatrix aufgestellt. Sie besitzt die Größe  $N \times N$ , wobei  $N$  die Anzahl der Dreiecke im Netz ist. Die Matrix wird zunächst als Einheitsmatrix initialisiert. Nun wird das Netz durchlaufen. Wenn zwei Dreiecke benachbart sind, dann wird der entsprechende Eintrag in der Matrix auf 1 gesetzt. Zeile  $n$  entspricht Dreieck  $n$  und Spalte  $m$  entspricht Dreieck  $m$ . Natürlich ist die Matrix symmetrisch, weil Dreieck  $n$  Nachbar von  $m$  ist und umgekehrt. Für zwei benachbarte Pixel werden die Index-Werte ( $m$  und  $n$ ) gelesen und diese als Zeilenindex und Spaltenindex in der Nachbarschaftsmatrix verwendet. Steht dort eine 1, zeigen die beiden Pixel zwei benachbarte Dreiecke und es liegt keine Körperteilkante vor. Steht dort hingegen eine 0, dann zeigen die beiden Pixel zwei nicht benachbarte Dreiecke und es liegt eine Objektgrenze vor, die auf eine Selbstverdeckung hindeutet.



## 6. Approximation des Modells

Das hier verwendete Approximationsverfahren besteht aus zwei Phasen, der Erkennungsphase und der Anpassungsphase. Die Erkennungsphase dient der Verknüpfung zwischen Modell und Bildinformation. Sie erlaubt die Zuordnung der erforderlichen Verschiebungen im Bild oder im 3D-Raum, nachdem eine möglichst eindeutige Korrespondenz zwischen Punkten des Modells und Punkten des Bildes gefunden wurde. Dies ist Gegenstand von Kapitel 7. Dafür werden Algorithmen der Bildverarbeitung verwendet.

Die Verringerung dieser Verschiebungen ist Aufgabe der Anpassungsphase. Meist werden hybride Ansätze verwendet, in denen sich beide Phasen gegenseitig beeinflussen, weil gute Näherungswerte des Modells den Suchraum verkleinern und damit die Korrespondenzfindung vereinfachen. Weiterhin erlauben gute Näherungswerte die Verwendung von Entscheidungskriterien zur Ausreißerbeseitigung und das Auflösen visueller Mehrdeutigkeiten.

### 6.1. Übersicht geeigneter Approximationsverfahren

Zur Vereinfachung der Anpassungsphase lässt sich der Zustandsraum auch diskretisieren [GD96]. Dabei werden Winkel mit  $5^\circ$  bis  $15^\circ$  und Translationen mit 5 cm bis 10 cm quantisiert. Die Projektion des kompletten Modells ist allerdings aufwendig und gegenwärtig nicht in Echtzeit durchführbar. Jede Projektion stellt dabei eine Hypothese für einen Zustandsvektor dar. Durch ein Ähnlichkeitskriterium erfolgt dann die Auswahl der besten Hypothese. Sie verwenden ein modifiziertes Champfer-Matching [Gus92] als Gütekriterium. Dabei werden Kanten extrahiert, binärisiert und anschließend wird eine Distanztransformation nach [RP66] und [Soi98] durchgeführt. Mit Hilfe des Distanzbildes lässt sich leicht der mittlere Abstand zwischen Kanten des Modells und den Konturen im Bild berechnen, den es zu minimieren gilt.

In [DF99] und [DF01] werden die Prinzipien der Technischen Mechanik zur Simulation der Bewegung von Mehrkörpersystemen auch zur Modell-Approximation verwendet. Gemessene Abweichungen zwischen Modellsilhouette und Bildsilhouette werden dabei zunächst in Federkräfte umgerechnet, die über Massenträgheit und Trägheitsmoment als Beschleunigungen und Winkelbeschleunigungen die Lage der miteinander verbundenen Körperteile des Modells verändern. Als Beobachtungen dienen Messungen im Bild oder 3D-Punktvolken, die aus Tiefenkarten gewonnen wurden. Demonstriert wird das Verfahren an einem Mensch-Modell und dem Modell einer Hand. Die Arbeit mit Kräften

## 6. Approximation des Modells

erfordert die Lösung der Bewegungsgleichungen. Dies ist mit einem rekursiven Algorithmus ([Wit77], [RS88]) effizient möglich. Eine Implementierung ist mit [LAZ98] frei verfügbar, jedoch fehlt es an dem konkreten Mensch-Modell. Die Anwendung der Algorithmen zur Gangsimulation und damit zur Synthese ist in [Hen02] genau beschrieben. Die Komplexität beträgt  $O(N)$  wobei  $N$  die Anzahl der starren Körperteile ist. Es wird aber explizit darauf hingewiesen, dass die Bewegungsgleichungen nicht zur Modellierung der Probandenbewegung dienen, sondern lediglich als mathematisches Mittel zur Minimierung der Abweichungen. Mit diesem Verfahren ist jedoch eine Aussage über die Genauigkeit der geschätzten Parameter nicht möglich.

Wird der Zustandsvektor vorgegeben und die korrekte Haltung berechnet, nennt man dies Vorwärts Kinematik. Wird das Modell mit den Transformationsketten korrekt aufgebaut, ergibt sich die Vorwärts Kinematik von selbst. Inverse Kinematik (IK) hingegen dient der Umkehrung dieses Vorganges, also der Berechnung des Zustandsvektors bzw. der Zustandsvektoren bei vorgegebener Haltung oder vorgegebenen Bewegungsablauf. In [TGB00] sind eine ganze Reihe von echtzeitfähigen Methoden zur Berechnung der inversen Kinematik aufgeführt.

Für die Approximation eines Mensch-Modells sind die Grundlagen der inversen Kinematik deshalb so wichtig, weil sich die Algorithmen gut für die Anpassungsphase verwenden lassen. Mit kalibrierten Kameras können Bewegungsabläufe im Bild vorgegeben werden. Die Inverse Kinematik berechnet die korrespondierende Folge von Zustandsvektoren.

Das nichtlineare Problem der inversen Kinematik wird in [Wac97] iterativ durch Linearisierung der Transformationskette im aktuellen Zustand gelöst. Die Approximation wird bei ihm als Schätzaufgabe aufgefasst. Er verwendet ein iterierten erweiterten Kalmanfilter zur Schätzung des Zustandsvektors und der zugehörigen Kovarianzmatrix.

Es ist darauf zu achten, dass Distanzkriterien verwendet werden, die auch eine Verschiebung des Modells längs zu Konturen erlauben. Diese sind zum Beispiel in [BI98] und [Wac97] gezeigt. So werden zwischen Konturen nur die Abstände entlang ihrer Normalenvektoren minimiert.

Verschiedene Ansätze verwenden bereits ein Modell der Bewegung. In [Wac97] wird bereits ein erweiterter iterierter Kalmanfilter verwendet. In [Mül01] wird die Methode noch einmal verbessert, indem ein iterativ erweiterter Kalmanfilter unter Verwendung einer Levenberg-Marquardt-Optimierung eingesetzt wird. Ein Bewegungsmodell erlaubt gewöhnlich eine Vorhersage von Näherungswerten für den aktuellen Zeitschritt unter Berücksichtigung eines bereits berechneten Bewegungsablaufes.

Diese Verfahren kommen in dieser Arbeit nicht zum Einsatz. Das Hauptargument gegen den Kalmanfilter: Es besteht zunächst keine Echtzeitanforderung für die Bildsequenzauswertung. Die bisherigen Ansätze verwenden aber den Kalmanfilter hauptsächlich, um eine Schätzung der Bewegungsgrößen in Echtzeit zu realisieren. Das Kalmanfilter ist ein rekursiver Filter, der für jeden Zeitschritt einen Zustandsvektor schätzt. Dabei werden zur Parameterschätzung jedoch nur Informationen der Vorgeschichte einschließlich des aktuellen Zeitschrittes verwendet. Der Kalmanfilter 'kennt' also in jedem Zeitschritt

nur eine Seite der Zeitachse. Das hat zur Folge, dass die geschätzten Bewegungsgrößen zwangsläufig mit einer Phasenverschiebung behaftet sind. Es entsteht also ein unvermeidlicher Fehler dadurch, dass dem Kalmanfilter zur Auswertung nur eine Seite der Zeitachse zur Parameterschätzung zur Verfügung steht.

Weil aber hier keine unmittelbare Echtzeitanforderung besteht, ist es möglich, eine Parameterschätzung zu verwenden, die gleichzeitig Informationen aller Zeitschritte berücksichtigt. Deshalb müsste sie theoretisch in der Lage sein, die Bewegungsgrößen ohne eine Phasenverschiebung zu realisieren. Das hier verwendete Verfahren durch Ausgleichsrechnung ist deshalb von den bisherigen Ansätzen abzugrenzen.

## 6.2. Approximieren des Modells an die 3D-Punkte

Mit den bisher beschriebenen Systemkomponenten kann das Mensch-Modell in eine beliebige Haltung gebracht werden, sofern der Zustandsvektor dieser Haltung bekannt ist. Dazu werden die Transformationsmatrizen  $\mathbf{T}$  berechnet und das Oberflächenmodell mittels Skelettdeformation verändert. Damit lassen sich synthetische Bildsequenzen erzeugen. Bei der Bildauswertung ist aber genau der Zustandsvektor eines jeden Zeitschrittes unbekannt und soll bestimmt werden.

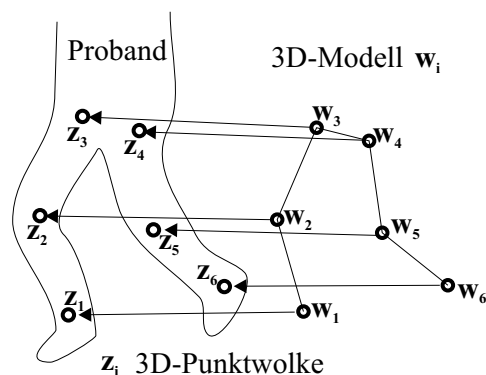


Abbildung 6.1.: Die Abweichung (Pfeile) zwischen Modellpunkten  $w_i$  und Zielpunkten  $z_i$  soll im 3D-Raum minimiert werden.

Zur Positionierung einer kinematischen Kette eines Industrieroboter wird für das letzte Kettenglied, dem sogenannten Endeffektor (Arbeitswerkzeug), die 3D-Sollposition und 3D-Sollorientierung vorgegeben. Inverse Kinematik berechnet nun den Zustandsvektor, der die kinematische Kette so einstellt, dass der Endeffektor sein Ziel (Werkstück) erreicht.

In der klassischen inversen Kinematik wird die Soll-Orientierung und Soll-Position *eines* Arbeitswerkzeuges vorgegeben, die der Industrieroboter zu erreichen hat. Die genaue Lage der anderen Elemente des Roboters spielt dabei nur eine untergeordnete Rolle. Für

## 6. Approximation des Modells

die Bewegungsanalyse müssen aber *alle* Körperteile eines Probanden korrekt positioniert werden. Bei dem vorgeschlagenen Konzept ergibt sich die Orientierung der einzelnen Körperteile aus ihrer Geometrie. Wenn zum Beispiel der Ort des Hüftgelenkes, des Knies und des Sprunggelenkes bekannt ist, dann ist auch die Orientierung des Oberarmes eindeutig bestimmt, da das Knie lediglich einen Winkelfreiheitsgrad besitzt (Scharniergelenk). Deshalb und weil sich mit Orientierungen bei der Bildauswertung schlecht rechnen lässt, soll als Eingang der Inversen Kinematik nur mit Punktpositionen, nicht aber mit Orientierungen gerechnet werden. Die Aufgabe besteht dann darin, den Zustandsvektor  $\mathbf{x}$  so zu bestimmen, dass  $M$  Punkte des Modells  $\mathbf{w}_i(\mathbf{x})$  (siehe Gl.(4.30)) ihre Zielpunkte  $\mathbf{z}_i$  erreichen.

Die Zielpunkte können zum Beispiel aus Tiefenkarten von Abstandssensoren gewonnen werden [WL01], [DF01]. Als Ergebnis entstehen meist Verschiebungsvektoren oder Distanzfunktionen im 3D-Raum, die es zu minimieren gilt: Die folgende Fehlerfunktion  $e(\mathbf{x})$  soll dann minimiert werden.

$$e(\mathbf{x}) = \sum_{i=1}^M (|\mathbf{z}_i - \mathbf{w}_i(\mathbf{x})|)^2 \rightarrow \min \quad (6.1)$$

Alle Koordinatenangaben liegen hier im Weltkoordinatensystem vor, deshalb ist es sehr übersichtlich und gut geeignet, die grundlegenden Elemente der Parameterschätzung zu demonstrieren.

Im Abschnitt 3.5 wurde bereits ein kinematisches Modell der Beine mit 10 Freiheitsgraden beschrieben. Es ist sehr interessant, mit Hilfe inverser Kinematik den Zustandsvektor  $\mathbf{x}$  zu schätzen, wenn die Zielkoordinaten  $\mathbf{z}_i$  der sechs Gelenkpunkte vorgegeben wurden. Da die Elementarmatrizen der Rotationsfreiheitsgrade Sinus und Cosinus enthalten, ist das System nichtlinear. Deshalb muss es linearisiert werden. Dabei wird üblicherweise ein Näherungswert für den Zustandsvektor vorgegeben, linearisiert und iterativ ein besserer Zustandsvektor berechnet. Dieses Rechenschema wurde bereits in Abschnitt 3.2 verwendet, mit dem Unterschied, dass jetzt nicht die Gelenkpunkte, sondern die Translationsparameter und Gelenkwinkel im Zustandsvektor repräsentiert sind.

Die Linearisierung erfolgt in einer Taylorreihe (Abbruch nach dem linearen Term). Dabei werden die Koordinaten der Modellpunkte nach den einzelnen Zustandsvektorkomponenten abgeleitet. Die Ableitungen  $\frac{\partial \mathbf{w}_i}{\partial \mathbf{x}}$  werden entweder numerisch (Anhang F) oder mit dem automatischen Algorithmus aus Anhang E berechnet.

$$\mathbf{w}_i(\mathbf{x}) + \frac{\partial \mathbf{w}_i}{\partial \mathbf{x}} \cdot \Delta \mathbf{x} = \mathbf{z}_i \quad (6.2)$$

$$\frac{\partial \mathbf{w}_i}{\partial \mathbf{x}} \cdot \Delta \mathbf{x} = \mathbf{z}_i - \mathbf{w}_i(\mathbf{x}) \quad (6.3)$$

Für jeden Punkt lassen sich dann 3 Gleichungen aufstellen. Bei sechs Gelenk- und sechs Zielpunkten ergeben sich bereits 18 Gleichungen. Die Änderungen der 10 unbekanntenen Freiheitsgrade  $\Delta \mathbf{x}$  lassen sich durch Ausgleichsrechnung berechnen.

### 6.3. Approximieren des Modells an die Bildpunkte

Um die Gleichungen zu testen, kann ein beliebiger Zustandsvektor vorgegeben und die Koordinaten der sechs Gelenkpunkte berechnet werden. Anschließend verändert man den Zustandsvektor und approximiert das Modell mittels inverser Kinematik an die zuvor berechneten Gelenkpunkte. Es muss dann wieder ein Zustandsvektor erhalten werden, der dem zuvor vorgegeben entspricht.

Zusammen mit der Formbeschreibung durch Konusmodelle (Abschnitt 4.3) entsteht ein kompaktes Modell der Beine zur Ganganalyse. Dabei werden je zwei der sechs Gelenkpunkte zur Beschreibung der Lage eines Kegelstumpfes verwendet.

Ein System müsste dann zunächst eine dreidimensionale Punktwolke des Probanden erzeugen. Mit bekannten Triangulationsverfahren, wie Stereovision und Photogrammetrie könnte dies geschehen. Es würden dann aus korrespondierenden Bildpunkten zunächst die dreidimensionalen Zielpunkte  $\mathbf{z}_i$  berechnet werden. Nicht unproblematisch ist dabei die Bereitstellung der korrekten Korrespondenz zwischen den Kameraansichten, weil die Extremitäten meist schmal und wenig texturiert sind.

Anschließend müsste die Korrespondenz zwischen Modell und Punkten gelöst werden. Dadurch stellt sich das Korrespondenzproblem gleich zweimal, zunächst zur Berechnung der Punktwolke (Korrespondenz zwischen zwei oder mehr Kamerabildern) und dann zur Berechnung der Korrespondenz zwischen Punktwolke und dem Modell. Eine Abschätzung der Genauigkeit der berechneten Winkelgrößen dürfte in diesem Fall schwierig sein. Das Verfahren würde auf Multikameraanordnungen mit spezieller Kamerageometrie beschränkt sein. Die Kameras müssten wenigstens paarweise aufgebaut werden, um jeweils eine Punktwolke aus einer Ansicht zu erzeugen. Ebenfalls problematisch wäre die Berücksichtigung von (Selbst-)Verdeckungen. In dieser Arbeit wurde deshalb ein anderer Weg eingeschlagen.

### 6.3. Approximieren des Modells an die Bildpunkte

Die Minimierung muss nicht unbedingt im dreidimensionalen Raum erfolgen. Bei der Auswertung von Bilddaten besteht ja gerade die Aufgabe darin, das Modell entsprechend der Bildinformation zu positionieren. Abbildung 6.2 zeigt diesen Fall. Daraus ergeben sich folgende Vorteile:

- direkte und vereinfachte Korrespondenz zwischen Modell und Bild
- Berücksichtigung von (Selbst-)Verdeckungen
- die Schätzung der Genauigkeit der Parameter ist direkt möglich (siehe Ausgleichsrechnung Anhang C)
- Skalierbarkeit von 1 bis N Kameras und
- Teilinformationen durch Bildkanten können genutzt werden.

## 6. Approximation des Modells

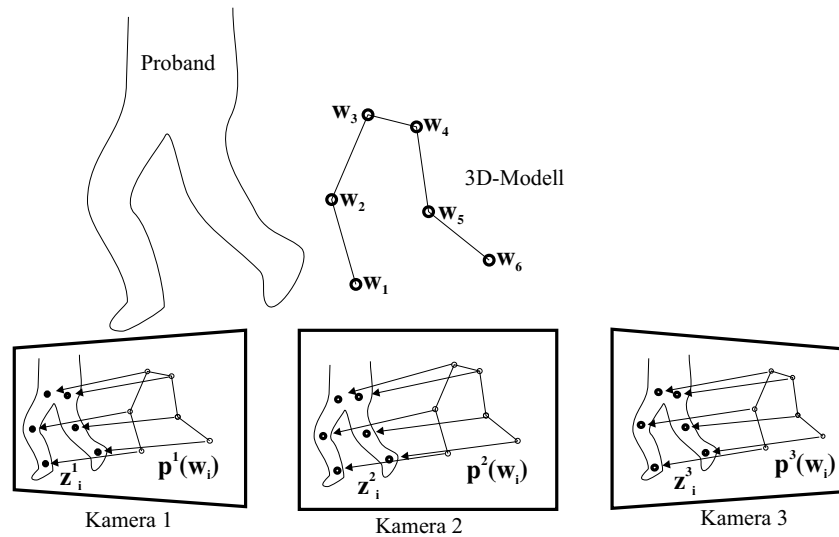


Abbildung 6.2.: Die Länge der Pfeile soll minimiert werden. Zielpunkte  $(z_i^1, z_i^2, z_i^3)$  werden direkt in den drei Kamerabildern vorgegeben.

In [Wac97] wird gezeigt, wie vorzugehen ist, wenn zweidimensionale Zielpunkte  $z_i^k$  in einem Bild der Kamera  $k$  vorliegen. Dabei ist  $p^k(w_i(x))$  einfach die projektive Abbildung des Punktes  $w_i(x)$  mit dem Kameramodell<sup>1</sup>  $p^k()$  des Bildes, in dem  $z_i^k$  vorgegeben wurde. Es sind also gleichzeitig die Fehler in allen beteiligten Kameras zu minimieren.

$$e(\mathbf{x}) = \sum_{k=1}^N \sum_{i=1}^M (|z_i^k - p^k(w_i(\mathbf{x}))|)^2 \rightarrow \min \quad (6.4)$$

Gl.(5.14) aus dem Ende von Abschnitt 5.2.2 kann verwendet werden, um das Modell direkt an gegebene Punkte in den Kamerabildern heranzuführen.

$$\frac{\partial p^k}{\partial \mathbf{x}} = \frac{\partial p^k}{\partial \mathbf{l}_k} \cdot \mathbf{T}_{0k}^{-1} \cdot \frac{\partial w_i}{\partial \mathbf{x}} \quad (6.5)$$

Die Matrix  $\frac{\partial p^k}{\partial \mathbf{x}}$  besitzt zwei Zeilen und so viele Spalten, wie es Freiheitsgrade im Zustandsvektor  $\mathbf{x}$  gibt. Jede Spalte kann als ein Richtungsvektor im Kamerabild verstanden werden, der in die Richtung zeigt, in die sich der projizierte Modellpunkt  $p^k(w_i(\mathbf{x}))$  bewegen würde, wenn sich der korrespondierende Freiheitsgrad um einen kleinen Betrag vergrößern würde. Zur Fehlersuche kann es hilfreich sein, diese Vektoren an  $p^k(w_i(\mathbf{x}))$  einzuzichnen und zu überlegen, ob ihre Richtung plausibel ist.

Die  $2 \times 4$  Matrix  $\frac{\partial p^k}{\partial \mathbf{l}}$  erhält man durch Gl.(5.10) aus Abschnitt 5.2. Matrix  $\mathbf{T}_{0k}^{-1}$  stammt ebenfalls aus dem Kameramodell und rotiert das Differenzial  $\frac{\partial w_i}{\partial \mathbf{x}}$  ins Kamerakoordina-

<sup>1</sup>siehe Abschnitt 5

## 6.4. Approximieren des Modells an die Konturen im Bild

tensystem der Kamera  $k$ .

$$\mathbf{p}^k(\mathbf{w}_i(\mathbf{x})) + \frac{\partial \mathbf{p}^k}{\partial \mathbf{x}} \cdot \Delta \mathbf{x} = \mathbf{z}_i^k \quad (6.6)$$

$$\frac{\partial \mathbf{p}^k}{\partial \mathbf{x}} \cdot \Delta \mathbf{x} = \mathbf{z}_i^k - \mathbf{p}^k(\mathbf{w}_i(\mathbf{x})) \quad (6.7)$$

Die Vektor-Gl.(6.7) liefert zwei Gleichungen im Gleichungssystem.

### 6.4. Approximieren des Modells an die Konturen im Bild

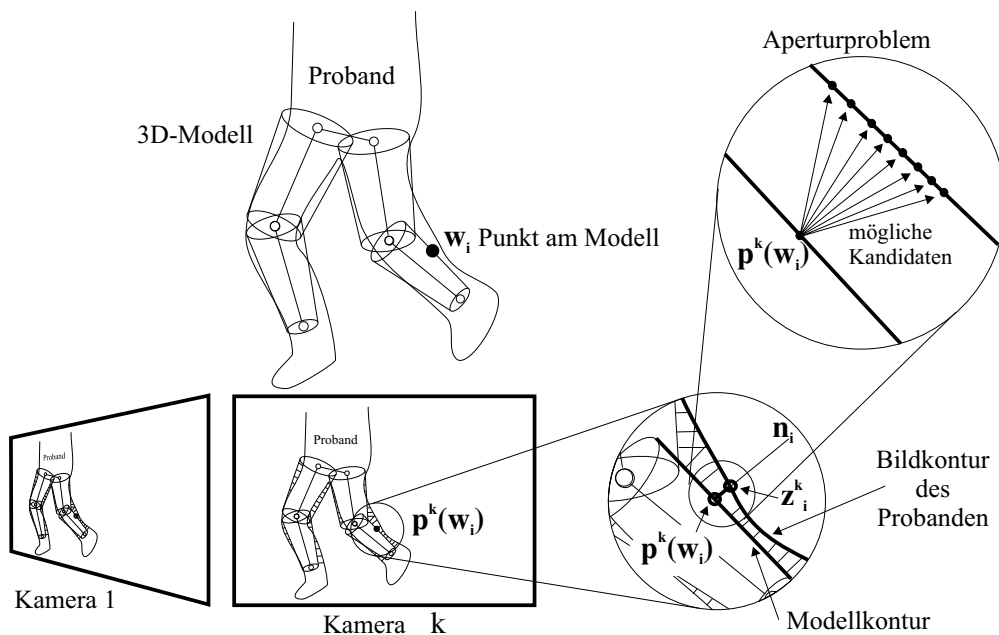


Abbildung 6.3.: Minimiert werden die Abweichungen zwischen Modellkontur und Kontur des Probanden im Bild. Gemessen wird ausgehend vom projizierten Punkt  $\mathbf{p}^k(\mathbf{w}_i(\mathbf{x}))$  entlang einer vorgegebenen Normalenrichtung  $\mathbf{n}_i$ .

In Abbildung 6.3 wurden bereits gute Näherungswerte für das Modell vorgegeben, sodass es sich bereits in die Nähe des Probanden befindet. Das kann zum Beispiel durch Approximation des Modells an manuell vorgegebene Punkte geschehen (siehe Abschnitt 6.3).

In Abschnitt 4.3 wurde gezeigt, wie sich Kegelstümpfe zur Formbeschreibung verwenden lassen. Im einfachsten Fall wird das Skelettmodell nach Abschnitt 3.5 bestehend aus sechs Punkten verwendet. Die Lage eines jeden Kegelstumpfes wird durch je zwei Punkte beschrieben. Mit Hilfe der Koordinaten des Projektionszentrums einer Kamera lassen sich die konturbildenden Mantelpunkte des Kegelstumpfes finden, die dann mit dem Kameramodell nach Abschnitt 5 in jedes Bild projiziert werden.

## 6. Approximation des Modells

Wenn Abweichungen zwischen Kantenpunkten des Modells und Kantenpunkten des Bildes minimiert werden sollen, dann darf nach [BI98] nur eine skalare Gleichung aufgestellt werden, da eine Kante lediglich Abstandsmaße quer zur Kantenrichtung liefert. Das Modell bleibt dadurch längs der Kantenrichtung unbestimmt und im übertragenen Sinne beweglich. Eine Anwendung von Gl.(6.7) wäre zwar auf den ersten Blick denkbar, ist aber falsch, weil das Modell dann auch längs zur Kantenrichtung in seiner Beweglichkeit eingeschränkt würde. Gl.(6.10) berücksichtigt diesen Zusammenhang und lässt dem Modell einen Freiheitsgrad entsprechend der Normalenrichtung  $\mathbf{n}_i$ . Erst wenn mehrere dieser Gleichungen mit verschiedenen Normalenrichtungen aufgestellt wurden, ist die Lage des Modells tatsächlich bestimmt. Genauso, wie es bereits beim klassischen Aperturproblem der Fall ist. Üblicherweise wird ausgehend von einem Kantenpunkt des Modells in Normalenrichtung  $\mathbf{n}_i$  (mit  $1 = |\mathbf{n}_i|$ ) nach einem Kantenpunkt in Kamera  $k$  gesucht und der Abstand  $a_i^k$  gemessen. Die Normalenrichtung  $\mathbf{n}_i$  wird dabei durch die Modellkontur vorgegeben.

$$a_i^k = \mathbf{n}_i^T (\mathbf{z}_i^k - \mathbf{p}^k(\mathbf{w}_i(\mathbf{x}))) \quad (6.8)$$

$$e(\mathbf{x}) = \sum_{k=1}^N \sum_{i=1}^M (a_i^k)^2 \rightarrow \min \quad (6.9)$$

$$\mathbf{n}_i^T \cdot \frac{\partial \mathbf{p}^k}{\partial \mathbf{l}} \cdot \mathbf{T}_{0k}^{-1} \cdot \frac{\partial \mathbf{w}_i}{\partial \mathbf{x}} \cdot \Delta \mathbf{x} = \mathbf{n}_i^T \cdot (\mathbf{z}_i^k - \mathbf{p}^k(\mathbf{w}_i(\mathbf{x}))) = a_i^k \quad (6.10)$$

Nach den ersten Experimenten (siehe Kapitel 10) mit diesen Gleichungen zeigte sich das Aperturproblem. Es drückte sich dadurch aus, dass mit einem einfachen Konusmodell der Ober- und Unterschenkel die vertikale Position des Probanden in den Double-Support-Phasen nicht geschätzt werden konnte. Das Modell wurde immer dann instabil, wenn der Proband beide Beine gestreckt hatte. Erst nach relativ langen Untersuchungen zeigte sich, dass es zu einem Rangabfall bei der Lösung der Normalengleichungen kommt. Ein Rangabfall tritt zum Beispiel auf, wenn mehrere Gleichungen linear abhängig voneinander sind. Mit Hilfe des Modells konnten zwar mehr Gleichungen als unbekannte Parameter aufgestellt werden, aber die Gleichungen repräsentierten quasi die selbe Information.

Verdeutlicht werden kann dies, wenn die Vergrößerung rechts oben in Abbildung 6.3 betrachtet wird. Sie zeigt einen Punkt auf dem Modell und gemäß dem Aperturproblem mehrere mögliche Kandidaten auf der Kontur des Probanden. Es besteht also Unklarheit darüber, mit welchem der Kandidaten der Punkt tatsächlich korrespondiert. Auf jeden Fall kann aber der Normalenvektor der Modellkontur berechnet werden und es lässt sich für den Modellpunkt Gl.(6.10) aufstellen. Weil eine Gleichung aufgestellt wurde, könnte auch ein unbekannter Parameter des Modells geschätzt werden. Theoretisch wäre es nun möglich, oberhalb oder unterhalb des Modellpunktes eine beliebige Anzahl weiterer Gleichungen aufzustellen. Denkbar, dass sich dadurch eine beliebige Anzahl von Modellparametern berechnen ließe. Natürlich ist dies praktisch nicht möglich, weil sich



## 6.5. Gegenüberstellung der Gleichungen

für jeden weiteren Punkt auf der selben Modellkontur das Aperturproblem in der selben Weise stellt. Dies wird dann durch den Algorithmus zur Lösung des Gleichungssystems mit einem Rangabfall quittiert, weil die scheinbar neuen Gleichungen in Wirklichkeit linear abhängig sind.

Die beste Maßnahme gegen den Rangabfall ist die Bereitstellung von mehr Information, indem Gl.(6.10) mit Normalenvektoren  $\mathbf{n}_i$  in sehr unterschiedlichen Richtungen aufgestellt wird. Das könnte zum Beispiel dadurch geschehen, dass eine detailliertere Modellkontur verwendet wird, bei der nicht alle Normalenvektoren in die selbe Richtung zeigen, wie dies beim einfachen Konus der Fall ist. Weiterhin wäre es denkbar, den Probanden durch gemusterte Kleidung mit einer festen Eigentextur zu versehen. Auch dadurch könnte Gl.(6.10) mit verschiedenen Normalenrichtungen aufgestellt werden.

In dieser Arbeit stellte sich jedoch heraus, dass es bereits durch die Verwendung eines einfachen Konturtemplates für die Füße in Verbindung mit dem Skelettmodell zu einer deutlichen Verbesserung kommt. Weil sich der Fuß aus Konturen sehr unterschiedlicher Richtung zusammensetzt, kann die Lage des Sprunggelenkes im Bild zuverlässig bestimmt werden. Daraus lässt sich auch auf die Lage des Kniegelenkes im Bild schließen, das Mensch-Modell gibt dafür den Abstand zwischen Knie- und Sprunggelenk vor. Die noch fehlende Richtungsinformation ergibt sich aus den Konturen der Unterschenkel.

## 6.5. Gegenüberstellung der Gleichungen

Die Parameter des Modells werden durch Minimierung einer Gütefunktion geschätzt. Drei Möglichkeiten der Formulierung von Gütefunktionen wurden vorgestellt. Viel interessanter für die Parameterschätzung ist aber nicht die Gütefunktion selbst, sondern die Gleichungen, die sich daraus für eine Ausgleichsrechnung ergeben. Sie sind hier noch einmal zusammengestellt.

$$\frac{\partial \mathbf{w}_i}{\partial \mathbf{x}} \cdot \Delta \mathbf{x} = \mathbf{z}_i - \mathbf{w}_i(\mathbf{x}) \quad (6.11)$$

$$\frac{\partial \mathbf{p}^k}{\partial \mathbf{l}_k} \cdot \mathbf{T}_{0k}^{-1} \cdot \frac{\partial \mathbf{w}_i}{\partial \mathbf{x}} \cdot \Delta \mathbf{x} = \mathbf{z}_i^k - \mathbf{p}^k(\mathbf{w}_i(\mathbf{x})) \quad (6.12)$$

$$\mathbf{n}_i^T \cdot \frac{\partial \mathbf{p}^k}{\partial \mathbf{l}_k} \cdot \mathbf{T}_{0k}^{-1} \cdot \frac{\partial \mathbf{w}_i}{\partial \mathbf{x}} \cdot \Delta \mathbf{x} = \mathbf{n}_i^T \cdot (\mathbf{z}_i^k - \mathbf{p}^k(\mathbf{w}_i(\mathbf{x}))) = a_i^k \quad (6.13)$$

Sollen Differenzen zwischen 3D-Modellpunkten  $\mathbf{w}_i$  und 3D-Zielpunkten  $\mathbf{z}_i$  minimiert werden, wird je eine Vektor-Gl.(6.11) aufgestellt. Im Gleichungssystem expandieren sie zu drei Beobachtungsgleichungen, für jede Koordinatenrichtung im Weltkoordinatensystem eine Gleichung. Ein Anwendungsfall ist die Approximation des Modells an eine vorher berechnete Punktwolke.

Sollen hingegen Differenzen zwischen 2D-Modellpunkten  $\mathbf{p}^k(\mathbf{w}_i(\mathbf{x}))$  und 2D-Zielpunkten  $\mathbf{z}_i^k$  in einem Kamerabild  $k$  minimiert werden, wird je eine Vektor-Gl.(6.12) aufgestellt.

## 6. Approximation des Modells

Daraus werden im Gleichungssystem zwei Gleichungen expandiert. Diese Gleichung wird genutzt, wenn ein Anwender zum Beispiel Gelenkpunkte in Kamerabildern vorgegeben hat und das Modell so gesteuert werden soll, dass es diese Gelenkpunkte erreicht.

Wenn Abweichungen zwischen Kantenpunkten des Modells und Kantenpunkten des Bildes minimiert werden sollen, ist das Aperturproblem zu berücksichtigen. In diesem Fall darf nur eine skalare Gl.(6.13) aufgestellt werden, da eine Kante lediglich Abstandsmaße quer zur Kantenrichtung liefert. Üblicherweise wird ausgehend von einem Kantenpunkt des Modells in Normalenrichtung  $\mathbf{n}_i$  nach einem Kantenpunkt im Bild gesucht und der Abstand  $a$  gemessen. Die Normalenrichtung  $\mathbf{n}_i$  wird gewöhnlich durch die Modellkontur vorgegeben (zum Beispiel durch numerische Ableitung).

Mit Hilfe dieser Gleichungen ist es möglich, grundlegende Approximationsverfahren zu implementieren.

## 7. Das Korrespondenzproblem und Lösungsansätze

In Abschnitt 2.2.3 wurde gezeigt, wie die Parameter des Modells durch Linearisierung des Modells in einem Arbeitspunkt und anschließender Lösung eines Gleichungssystems geschätzt werden können. Im Kapitel 6 wurden Möglichkeiten zur Aufstellung der notwendigen Beobachtungsgleichungen diskutiert, wenn Zielpunkte<sup>1</sup> oder Ziel-Konturen gegeben sind. Die Gewinnung der Zielpunkte ist nun Gegenstand dieses Kapitels.

Die bisherigen Systeme zur Bewegungsanalyse verwenden Marker, um eine eindeutige Korrespondenz zwischen Modell und Bild zur Verfügung zu stellen. Sollen aber keine Marker zum Einsatz kommen, muss die Korrespondenz auf anderem Wege hergestellt werden. Die richtige Korrespondenz zwischen Modell und dem Bild des Probanden ist deshalb die eigentliche Herausforderung bei der markerlosen Bewegungsanalyse.

Die wohl einfachste Möglichkeit ist die manuelle Vorgabe von Punkten in jedem Kamerabild. Allerdings ist dies sehr zeitaufwendig und eine erhebliche Quelle von Subjektivität, da sich ohne spezielle Markierungen (Gelenk-)Punkte nur schwer lokalisieren lassen.

Aufgabe einer automatischen Korrespondenzfindung ist es, zu einem gegebenen Körperteil des Modells die entsprechenden Bildregionen oder Bildmerkmale in jedem Kamerabild zu finden. Der Aufwand zur Suche eines Körperteils steigt dabei mit der Größe der in Frage kommenden Bildregion, in der es sich befinden kann. Mit einer größer werdenden Bildregion steigt aber auch die Gefahr von Mehrdeutigkeiten. Es besteht deshalb ein interessanter Zusammenhang zwischen dem Aufwand, der zur automatischen Korrespondenzfindung betrieben werden muss und den zur Verfügung stehenden Näherungswerten der Bewegungsparameter des Modells. Gute Näherungswerte machen kleine Suchbereiche möglich und reduzieren die Wahrscheinlichkeit von Mehrdeutigkeiten. In Kapitel 9 wird eine Strategie zur Bereitstellung von guten Näherungswerten erläutert.

Zur Approximation des Modells werden oft Kanten- und Konturfeatures extrahiert. Dabei kann eine präzise Modellierung des Grauwertverlaufes in den Bildern erfolgen, wie es im Modellgradientenverfahren von [Wac97] erfolgt. Das Verfahren wird eine hohe Genauigkeit aufweisen, sofern die genauen Abbildungseigenschaften der Kameras bekannt sind. Es ist jedoch zu erwarten, dass der Fangbereich dieses Verfahrens klein ist, da zwar eine Modellierung der Abbildungsfunktion erfolgt, aber keine Suche nach Kantenmerkmalen in einem größeren Bereich.

---

<sup>1</sup>im 2D-Bild oder im 3D-Raum

## 7. Das Korrespondenzproblem und Lösungsansätze

Diesen Nachteil besitzt die die Suchstrahlmethode nicht. Dabei wird ausgehend von Modellkanten entlang der Normalenrichtung in einem definierten Intervall nach einem Kantenpunkt gesucht. Dazu wird die Bildfunktion entlang des Suchstrahls äquidistant abgetastet. Es entsteht ein eindimensionales Grauwertprofil der Bildfunktion, hierin wird nach dem größten Gradienten gesucht. Dieser Ansatz stammt aus den Arbeiten über aktive Konturmodelle [BI98]. Da keine präzise Modellierung des Grauwertverlaufes erfolgt, könnte die erreichbare Genauigkeit geringer sein, als bei dem zuvor beschriebenen Verfahren.

Ein einfaches Blockmatching wird von [SBB00], [SB01] verwendet. Dazu wird das Modell im ersten Zeitschritt manuell positioniert. Dann werden Punkte von der Modelloberfläche abgerufen und ins Bild projiziert. Dort können Grauwertblöcke ausgeschnitten und Verschiebungsvektoren zum nächsten Zeitschritt berechnet werden. Das Modell wird entsprechend angepasst. Es existieren bereits robustere Methoden zum Beispiel [Mec99]. Eine bildbasierte Kalmanfilterung erlaubt Erkennung und Überbrückung von Verdeckungen und Störungen. Allen Verfahren, die auf der Korrelation von Oberflächengrauwerten basieren, ist gemeinsam, dass sie eine ausreichende Menge an Eigentextur benötigen. Sie versagen, wenn diese Informationen nicht zur Verfügung stehen, wie es bei unbekleideter Haut der Fall ist.

In [Wac97] werden neben den Kantenfeatures auch Oberflächengrauwerte verwendet. Seine Annahme ist: Oberflächengrauwerte des Modells bleiben entlang der Zeit konstant. Oberflächengrauwerte lassen sich in die Kamerabilder zurückprojizieren und mit den Grauwerten dieser Bildregion vergleichen. Ein Framework zur Modellierung ist in [STB00] und [SBF00] gezeigt. [Koc93] und [WN99] erläutern die notwendigen mathematischen Grundlagen und zeigen erste Ergebnisse. Es ist anzunehmen, dass dieses Verfahren im Wesentlichen durch die Kantenmerkmale stabil wird und dass die Verwendung der Oberflächengrauwerte lediglich bei bekleideten und gut texturierten Oberflächen einen Gewinn darstellen.

In [WADP97] wird ein Echtzeitverfahren zur Verfolgung von Menschen vorgestellt. Sie verwendeten Blobs als statistische Beschreibung der Oberflächenfarbwerte der Körperteile. Farbwerte werden durch Lagekomponenten im Bild ergänzt. Im Merkmalsraum bilden sich dann für jedes Körperteil Cluster, die sich entlang der Zeit nur langsam ändern. Ein Cluster wird durch seinen Mittelwert und eine mehrdimensionale Gaussverteilung beschrieben. Jeder Pixel des aktuellen Kamerabildes kann schnell auf Zugehörigkeit zu einem Körperteil überprüft werden, indem sein Abstand im Merkmalsraum relativ zu jedem Cluster bestimmt wird.

### 7.1. Problemanalyse

Im Prinzip geht es 'nur' darum, zu Körperteilen des Modells eindeutig korrespondierende Bildmerkmale zuzuordnen. Idealerweise entstehen eindeutige Punkt-zu-Punkt Korrespondenzen zwischen Modell und Bild. Da die Körperteile einfarbige längliche Strukturen

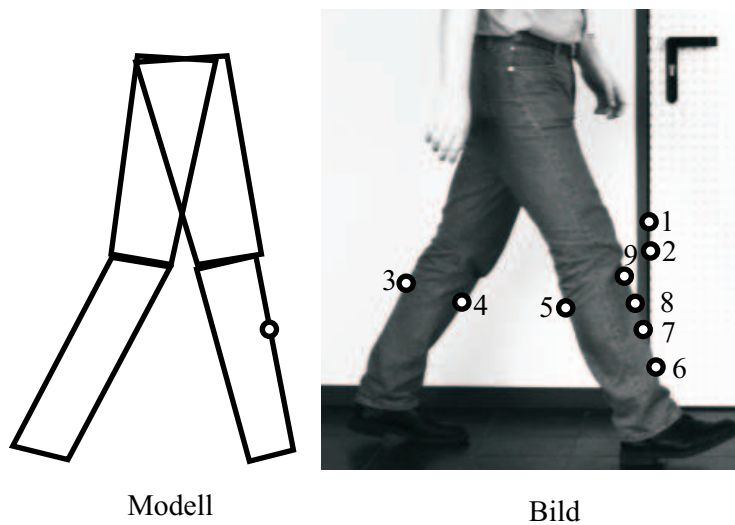


Abbildung 7.1.: Korrespondenzproblem zwischen Modell und Bild

sind, werden relevante Informationen in den Körperkanten codiert. Deshalb soll in dieser Arbeit die Korrespondenz nur entlang von Kantenpunkten hergestellt werden.

In Abbildung 7.1 ist das Korrespondenzproblem dargestellt. Zu sehen ist links ein Punkt des Modells und eine Auswahl von Kandidaten im Bild. Kandidaten 1 und 2 gehören zum statischen Hintergrund. Die Kandidaten 3 und 4 gehören zum anderen (falschen) Bein. Kandidat 5 liegt zwar bereits auf dem richtigen Körperteil, allerdings auf der falschen Seite. Die Kandidaten 6,7,8 und 9 liegen zwar bereits auf der richtigen Seite des korrekten Körperteils, jedoch ist die Korrespondenz in Längsrichtung schwierig. Die Kandidaten 6,7,8 und 9 verdeutlichen also das klassische Aperturproblem.

Viele Informationen für die Ganganalyse, wie Geschwindigkeit, Gelenkwinkel von Hüfte, Knie und Sprunggelenk zeigen sich bereits in den Seitenansichten des Probanden. Deshalb werden sie sehr gern zur Auswertung verwendet. Leider entstehen hier auch die meisten Selbstverdeckungen. Dabei lassen sich im Wesentlichen drei Verdeckungsregionen feststellen, wie in Abbildung 7.2 zu sehen ist:

1. Ständige Selbstverdeckungen in der Oberschenkel-Hüft-Gegend,
2. Periodische Verdeckung der Hüfte durch den Arm,
3. Periodische Verdeckungen in Knie-Unterschenkel-Fuß-Gegend während der Schwungphasen.

## 7. Das Korrespondenzproblem und Lösungsansätze

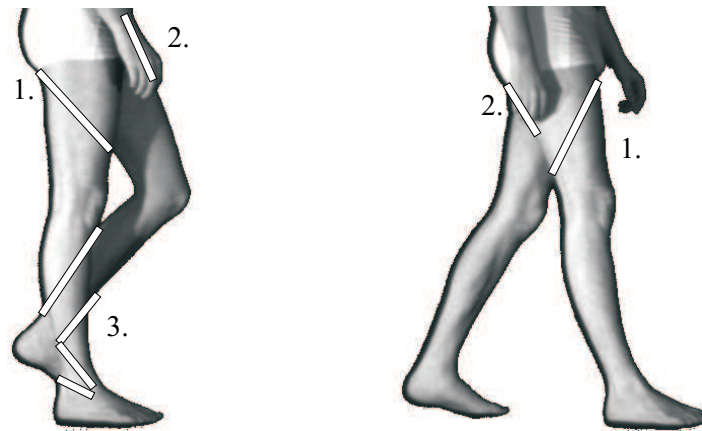


Abbildung 7.2.: Verdeckungen in der Seitenansicht, ihre Nummerierung ist im Text erläutert.

### 7.2. Möglichkeiten der Segmentierung des Probanden

Zunächst sollen die Kantenpunkte des Hintergrundes unterdrückt werden, um möglichst früh die Anzahl an Korrespondenz-Kandidaten zu reduzieren.

Das Ziel ist erreicht, wenn eine Vordergrund-Maske des Probanden erzeugt wurde. Zur Definition der Maske kommen verschiedene Informationen in Frage:

- Grauwert- oder Farbinformation (zum Beispiel Color-Keying)
- Bewegungsinformation
- Tiefeninformation [Höh01].

Die Aufbereitung der Informationen zu einer Maske ist ein Segmentierungsproblem, dessen optimale Lösung Gegenstand zahlreicher Publikationen ist. Der Aufwand zur Lösung hängt in erheblichen Maße von den genauen Eigenschaften des gegebenen Hintergrundes ab. Eine deutliche Vereinfachung der Algorithmen ergibt sich durch die Verwendung statischer Kameras vor einem statischen Hintergrund.

Zur Auswertung von Grauwert- und Farbinformation werden oft die Abweichungen zwischen einem Modell des Hintergrundes und dem aktuellen Kamerabild gemessen. Vor der eigentlichen Messung wird eine Bildsequenz der Szene ohne Probanden aufgezeichnet und daraus ein Modell des Hintergrundes berechnet. Dann lässt man den Probanden den Messbereich betreten und für jeden Zeitschritt und jede Kamera wird überprüft, ob Bildpixel nun noch die Eigenschaften des den Hintergrundmodells erfüllen oder ob die Abweichungen so groß sind, dass sie bereits zum Vordergrundobjekt gehören. Damit lassen sich Silhouetten des Vordergrundobjektes oder des Probanden berechnen.

Problematisch sind diese Verfahren immer dort, wo die Bildeigenschaften des Probanden mit denen des Hintergrundmodells übereinstimmen, dort entstehen dann Löcher in der

Vordergrundmaske.

Bei ungünstiger Beleuchtung wirft der Proband einen mehr oder weniger scharf umrandeten Schatten auf den Hintergrund. Dadurch können erhebliche Probleme für die Segmentierung auftreten, weil sich die Bildeigenschaften des Hintergrundes ändern. Im einfachsten Fall ändern sich lediglich die Intensitätswerte der Hintergrundpixel. Wenn jedoch eine scharfe Schattengrenze vorhanden ist, entstehen auch neue Kantenpunkte die kantenbasierte Suchverfahren stören können. Ein beleuchtungsinvariantes Verfahren wurde von [MAD01] vorgestellt, es basiert auf einem statistischen Kolinearitätskriterium.

Die Algorithmen können meistens deutlich vereinfacht werden, wenn der Hintergrund homogen und einfarbig ist. Das bekannteste und in der TV-Technik oft verwendete Verfahren ist das Color-Keying, bei dem ein Hintergrund aus mittlerem Blau oder Grün bereitgestellt wird. Es ist wichtig, den Hintergrund selbst gut und aus verschiedenen Richtungen auszuleuchten, weil Schattenwurf des Probanden die Segmentierung stören kann.

Um zunächst Einflüsse durch eine unvollständige oder fehlerhafte Segmentierung auszuschließen, wurde in dieser Arbeit ein noch einfacheres Prinzip für einige Untersuchungen verwendet. Der Hintergrund und Fußboden wurde mit einem schwarzem matten Stoff aus der Bühnentechnik bedeckt. Im Verhältnis zu den Kosten des Multikameraaufbaus fallen die Kosten für schwarzen Stoff nicht ins Gewicht. Der Nutzen ist jedoch enorm. Es werden keine aufwendigen Segmentierungsstrategien benötigt und es gibt keine Probleme durch Schattenwurf, weil der schwarze Stoff jegliches Licht absorbiert. Ein fester Schwellwert wird während der Kalibrierung festgelegt und eine rechteckige 'Region of Interest' für jedes Kamerabild ergibt nahezu optimale Vordergrund-Masken. Voraussetzung ist eine sehr gute Ausleuchtung des Probanden. Im Gegensatz zu den kantenbasierten Verfahren lässt sich für jeden Pixel klären, ob er zum Probanden oder zum Hintergrund gehört. Das steigert die Zuverlässigkeit der Korrespondenz zwischen Modell und Maske erheblich.

Für die Anwendung in einer natürlichen Klinikumgebung ist das Verhängen mit schwarzem Stoff voraussichtlich nicht akzeptabel. In diesem Fall muss ein maskenbildendes Segmentierungsverfahren angewendet werden, dass zum Beispiel die oben genannten Bildinformationen auswertet.

### 7.3. Labeling der Bildzeilen

Ein Vorschlag zur Verbesserung der Korrespondenz zwischen Modell und Bild ist das zeilenweise Abzählen der Kantenpunkte gemäß Abbildung 7.3. Für eine Region of Interest werden in jeder Zeile die Kantenpunkte indiziert. Lediglich wenn zwei bzw. vier Kantenpunkte je Zeile gefunden wurden, wird diese Zeile verwendet. Dies ermöglicht eine Art parity-check, der die Zuverlässigkeit dadurch erhöht, dass Zeilen mit einer ungeraden Anzahl von Punkten und damit voraussichtlich fehlerhaftem Labeling nicht verwendet

## 7. Das Korrespondenzproblem und Lösungsansätze

werden. Das Verfahren liefert dann die besten Ergebnisse, wenn ein guter Kontrast zwischen Hintergrund und Vordergrund vorhanden ist und wenn keinerlei Eigentextur auf den Beinen das Labeling erschwert.

Allerdings ist dieses Verfahren nicht ganz unproblematisch in den Single-Stance-Phasen, also wenn sich die Beine gegenseitig verdecken. Das Labeling nummeriert dort die Kantenpunkte nicht korrekt. In Abbildung 7.3 ist ein Beispiel gezeigt. Die linke Kante erhält in der Double-Support-Phase korrekterweise das selbe Label und gehört auch zum selben Bein. In der single-stance-phase wurde die linke Kante zwar immer noch mit dem selben Labelwert versehen, aber der obere Bereich der Kante gehört zum rechten Bein und der untere Bereich der Kante gehört zum linken.

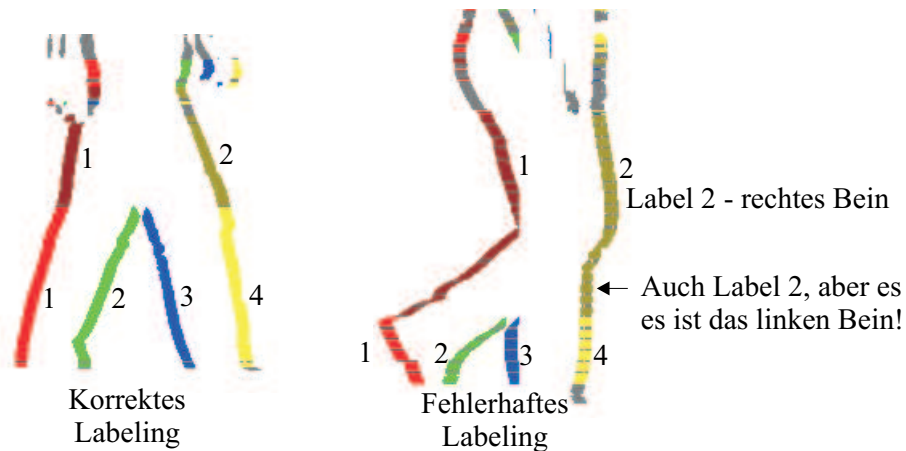


Abbildung 7.3.: Labeling der Kantenpunkte, die Nummerierung ist im Text erläutert.

Das Verfahren bringt einige weitere Probleme mit sich. Es vergibt die Label in Abhängigkeit von der Blickrichtung des Probanden zur Kamera. Weiterhin ändern sich Zuordnungen zwischen Label und Modell mit jedem Schritt, sodass eine mehr oder weniger aufwendige Zuordnungsstrategie der Labelwerte zu den Bereichen des Modells notwendig wird.

### 7.4. Disparität zur Unterscheidung des linken oder rechten Beines

In Abschnitt 7.1 wurde bereits diskutiert, wie wichtig die Seitenansichten des Probanden für die Ganganalyse sind. Leider entstehen hier auch die meisten Selbstverdeckungen. Aus Sicht einer Kamera verdeckt das der Kamera nahe Bein jeweils das der Kamera ferne Bein. Welches nun das linke oder rechte Bein ist, hängt wiederum von der Orientierung des Probanden ab.



#### 7.4. Disparität zur Unterscheidung des linken oder rechten Beines

Ist die Bewegungsrichtung des Probanden bereits bekannt und geht man davon aus, dass der Proband nicht etwa rückwärts oder seitlich läuft, lässt sich aus der Bewegungsrichtung auch die näherungsweise Orientierung des Probanden ermitteln. Gelingt es für jeden Kantenpunkt, den Abstand zur Kamera zu berechnen, ist es möglich, Kantenpunkte des kameranahen Beines von denen des kamerafernen Beines zu differenzieren. Damit ist dann über den Abstand auch die Zugehörigkeit zum linken oder rechten Bein gegeben.

Mit einer Stereo-Kamera-Anordnung ist es möglich, Entfernungen der Kantenpunkte zu berechnen. Werden zwei Kameras im sogenannten Normalfall angeordnet, lässt sich über eine einfache Formel der Zusammenhang zwischen der Disparität  $d$  und dem Abstand zur Kamera  $z$  angeben, wobei  $c$  die Kamerakonstante und  $b$  die Basisbreite ist:

$$z = \frac{c \cdot b}{d} \quad (7.1)$$

Der Normalfall hat den weiteren Vorteil, dass korrespondierende Punkte lediglich entlang einer Bildzeile gesucht werden müssen. Ist weiterhin der näherungsweise Abstand des Probanden bekannt, kann der Suchbereich innerhalb einer Zeile erheblich eingeschränkt werden. Die folgenden Werte sollen dies verdeutlichen:

$Z_{min}$	=	4000	[mm]	Abstand nahes Bein
$Z_{max}$	=	4200	[mm]	Abstand fernes Bein
$b$	=	300	[mm]	Basisbreite
$c$	=	1650	[Pixel]	(Brennw. $\approx 10.0$ [mm], Sensorel. 0.0060[mm])
$d_{Zmin}$	=	123.75	[Pixel]	Disparität nahes Bein
$d_{Zmax}$	=	117.80	[Pixel]	Disparität fernes Bein

Es ist zu erkennen, dass der Unterschied zwischen minimaler und maximaler Disparität für einen Probanden etwa 6 Pixel beträgt. Zum einen ist sie groß genug, dass auf eine Subpixel genaue Bestimmung der Kantenpunkte verzichtet werden kann und zum anderen ist sie klein genug, dass die Wahrscheinlichkeit der Verwechslung mit anderen Kantenpunkten eher gering ist. Das macht diese Variante der Korrespondenzfindung außerordentlich attraktiv.

Dabei ist es nicht einmal erforderlich, für jeden Kantenpunkt den tatsächlichen Abstand zur Kamera zu bestimmen. Die Zuordnung von Kantenpunkten zum kameranahen und kamerafernen Bein kann bereits durch die gemessene Disparität erfolgen. Es ist lediglich erforderlich, einen Disparitäts-Schwellwert anzugeben, unterhalb dem ein Kantenpunkt dem Kamera fernen Bein zuzuordnen ist. Dieser Schwellwert kann entweder durch Umstellung von Gl.(7.1) berechnet werden, indem eine  $z$ -Koordinate aus einer genäherten Modellposition vorgegeben wird. Oder es wird ein Histogramm der gemessenen Disparitäten verwendet und ein Schwellwert zum Beispiel durch Mittelwertbildung festgelegt.

## 7.5. 2D-Templatemarking mit Hilfe eines genetischen Algorithmus

In Kapitel 4 wurden 2D-Templates als eine sehr einfache Möglichkeit der Formbeschreibung vorgestellt. Sie können entweder direkt im Bild vorgegeben sein oder aus einem gegebenen 3D-Formmodell erstellt werden. Dafür werden die Körperkanten eines Körperteiles in das Kamerabild projiziert und die dabei entstandenen Konturen zu einem 2D-Template zusammengefasst. Die Lage und Orientierung des Templates kann zunächst durch zwei Gelenkpunkte des Skelettmodells vorgegeben werden. Die Korrespondenz zwischen jedem Template und dem Bild erfolgt durch Minimierung einer Kostenfunktion. Wurden optimale Lageparameter für jedes Template berechnet, können nun für jedes Template auch Zielpunkte für die beiden Gelenke berechnet und das Skelettmodell entsprechend approximiert werden.

Das Chamfer-Matching<sup>2</sup> [Gus92] verwendet ein schnelles Verfahren zur Berechnung des Abstandes zwischen Modell- und Bildkanten. Zunächst werden Kantenpunkte des Kamerabildes extrahiert und durch Anwendung eines Schwellwertes zu einer binären Maske verarbeitet. Diese Maske wird nun als Input für eine Distanztransformation [Soi98] benutzt. Durch eine Hin- und eine Rückiteration entsteht ein Bild (Distanzfunktion), in dem der Grauwert eines Pixels gleich dem Abstand des Pixels zu seinem nächsten Kantenpunkt ist. Die Distanzfunktion wird demzufolge null, wenn der Pixel direkt auf einer Bildkante liegt. Zur Berechnung des mittleren Abstandes zwischen Modell- und Bildkante wird die Modellkontur diskretisiert und die Distanzfunktion an den diskretisierten Stützstellen der Modellkontur abgetastet. Wird die Lage des Modells verändert, verändert sich demzufolge auch der mittlere Abstand zwischen Modellkonturen und den binärisierten Kantenpunkten des Kamerabildes.

Die Lage eines jeden Templates lässt sich nun ändern und jeweils der mittlere Abstand berechnen. Durch ein Optimierungsverfahren können nun optimale Templatepositionen im Bild berechnet werden. Ein kleiner Suchraum für jedes Template bewirkt eine Verringerung von Mehrdeutigkeiten und Rechenaufwand. Im einfachsten Fall wird für jedes Template ein dreidimensionaler Suchraum durch zwei Translationen und eine Rotation definiert. In Abbildung 7.4 wurde ein einfacher Genetischer Algorithmus verwendet, um optimale Templatepositionen zu finden. Zu sehen ist die Distanzfunktion, Grauwert=Distanz=0 wurde eingefärbt. Vor der ersten Iteration sind alle 500 Individuen gleichmäßig im Suchraum verteilt. Durch Kreuzung und Mutation mit anschließender Selektion entstehen nach 30 Generationen die ersten Ansammlungen in der Nähe lokaler Minimas. Nach bereits 60 Generationen wurden die beiden Füße sicher gefunden. Was nicht zu sehen ist: lässt man die Optimierung weiter laufen, dann konzentriert sich die ganze Generation auf einem Fuß. Ein Vorschlag zur Lösung des Problems in den Double-Support-Phasen ist die Einschränkung des Suchbereiches in der Nähe der tatsächlichen Fußposition. Es stellt sich dann aber auch wieder die Frage der Näherungswerte.

---

<sup>2</sup>engl. Chamfer - Hohlrinne oder Kerbe

## 7.5. 2D-Templatematching mit Hilfe eines genetischen Algorithmus

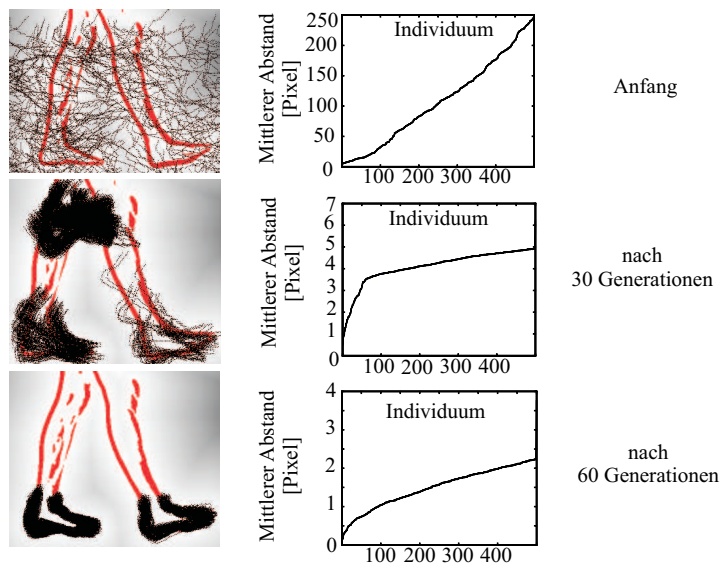


Abbildung 7.4.: 2D-Templatematching mit einem genetischen Algorithmus

Das Verfahren ist relativ unempfindlich gegen Teil-Verdeckungen, wie man in Abbildung 7.5 erkennen kann. Es wurde auch der zu gut einem Drittel verdeckte Fuß gefunden.

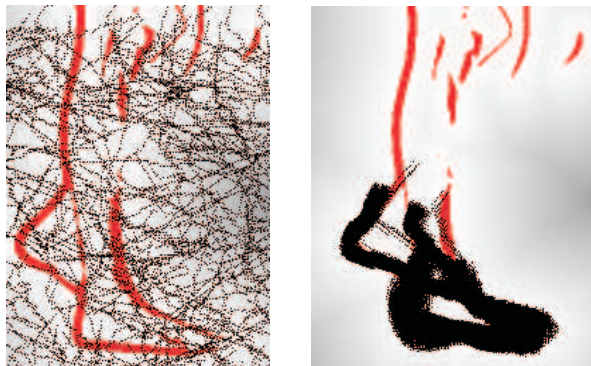


Abbildung 7.5.: Weitgehende Invarianz des Genetischen Algorithmus gegen Teil-Verdeckungen (Links: Anfang; Rechts: nach 50 Generationen)

In Abbildung 7.6 ist eine Population eines Unterschenkel-Templates zu erkennen. Es konnten beide Unterschenkel detektiert werden, allerdings konzentriert sich die Population auf den linken Unterschenkel. Interessant ist ein weiterer Effekt: Obwohl die Form des Templates den Unterschenkel gut wiedergibt, wird eine zu hohe Position geschätzt. Dieses Problem zeigt sich später auch bei der Approximation des Mensch-Modells mittels Ausgleichsrechnung. Es wird dann auf das klassische Aperturproblem zurückgeführt werden.

## 7. Das Korrespondenzproblem und Lösungsansätze

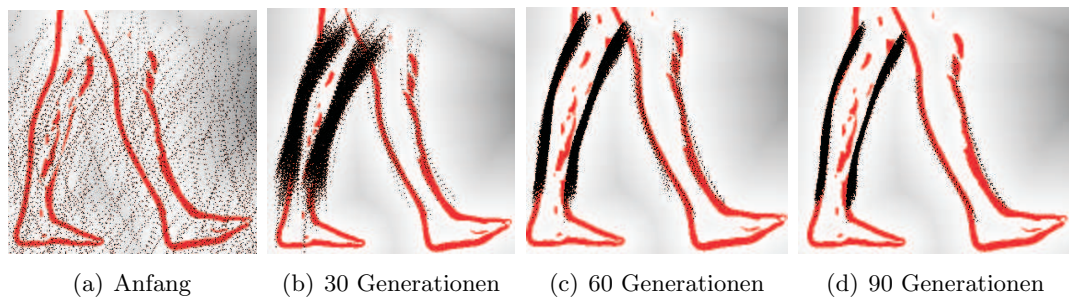


Abbildung 7.6.: Approximation eines Templates des Unterschenkels.

## 8. (Quasi-)periodisches Bewegungsmodell

Das Ziel der Bildsequenzauswertung ist es, einen Parametersatz des Mensch-Modells für jeden Zeitschritt zu schätzen. Die Anzahl der unbekannt Parameter wird zwar durch die Freiheitsgrade des Modells in jedem Zeitschritt festgelegt. Über die gesamte Sequenz betrachtet, bestimmt aber auch die Sequenzlänge die Anzahl der unbekannt Parameter.

Über die Sequenz betrachtet wäre es also denkbar, die Anzahl der Unbekannten zu verkleinern, wenn Zusammenhänge der Modell-Parameter zwischen den Zeitschritten existieren. Für den menschlichen Gang existiert ein Laufzyklus, dessen Details im Kapitel 2.1 erläutert wurden. Mit der Periodizität des Ganges ist also ein solcher Zusammenhang zwischen den Zeitschritten gefunden. Aufgrund begrenzter Muskelkräfte sind die Beschleunigungen limitiert, begrenzte Wertebereiche der Parameter (zum Beispiel Gelenkwinkelgrenzen) lassen die Geschwindigkeiten der Körperteile nicht endlos ansteigen. Deshalb sind die Haltungparameter benachbarter Zeitschritte ebenfalls ähnlich. Damit ist noch ein zweiter Zusammenhang gefunden. Über eine Zeitspanne betrachtet, müsste sich demzufolge die Anzahl der Unbekannten reduzieren lassen, indem eine Modellierung der Bewegung erfolgt.

Ein solches Bewegungsmodell sollte primär für jeden Zeitschritt  $t$  einen Zustandsvektor  $\mathbf{x}_t$  synthetisieren können. Es muss mit den bisher diskutierten Verfahren der Modellapproximation zusammenarbeiten können und sollte möglichst einfach sein. Deshalb soll die Synthese der Zustandsvektoren durch eine einfache lineare Gleichung erfolgen, in welcher der Vektor  $\mathbf{c}$  eine zeitunabhängige Beschreibung der Bewegung darstellt.

$$\mathbf{B}_t = \mathbf{B}(t) \tag{8.1}$$

$$\mathbf{x}_t = \mathbf{B}_t \cdot \mathbf{c} \tag{8.2}$$

Die Berechnung der Matrix  $\mathbf{B}_t = \mathbf{B}(t)$  richtet sich nach dem verwendeten Interpolationsverfahren.

Es ist bis jetzt nicht sicher geklärt, inwieweit der Gang als periodisch angenommen werden kann. Da der menschliche Gang einigen Störgrößen der Umwelt und Schwankungen körperinnerer Parameter unterliegt, ist anzunehmen, dass nicht jeder Schritt identisch zum nächsten ist, zumal es sich bei einem biologischen System nicht um ein starres technisches Gerät handelt. Eine genaue Aussage, wie groß die Parameterschwankungen eines gehenden Probanden von Schritt zu Schritt tatsächlich sind, konnte in dieser Arbeit noch nicht getroffen werden.

## 8. (Quasi-)periodisches Bewegungsmodell

Dennoch wäre es für viele weitere Anwendungen wünschenswert, lediglich den periodischen Anteil des Ganges zu verwenden, der für den Probanden durch Mittelwertbildung in jedem Schritt der Sequenz berechnet wird, da sich damit der mittlere Gang eines Probanden durch einen relativ kleinen Parametersatz beschreiben lässt. Je kleiner dieser Parametersatz ist, desto höher ist die Robustheit gegen Störungen, die Genauigkeit und desto einfacher dürften sich auf diesen Daten aufbauende Anwendungen gestalten lassen. Deshalb geht diese Arbeit in weiten Teilen davon aus, dass es diesen periodischen Anteil tatsächlich gibt und dass die Abweichungen von diesem periodischen Modell klein sind.

Unter Annahme eines periodischen Ganges ist zur Berechnung der Matrix  $\mathbf{B}_t$  auch eine periodische Funktion zu verwenden. Neben periodischen B-Splinefunktionen bietet sich auch eine klassische Fourierreihe an. Aus [BSG<sup>+</sup>96] stammt die folgende Gleichung dafür:

$$x(t) = \frac{a_0}{2} + \sum_{k=1}^K \left[ a_k \cos\left(\frac{2\pi kt}{T}\right) + b_k \sin\left(\frac{2\pi kt}{T}\right) \right] \quad (8.3)$$

Die Funktion  $x(t)$  besitzt die Periode  $T$ , es gilt  $f(t+T) = f(t)$ . Dies könnte zum Beispiel der Gelenkwinkelverlauf eines Kniegelenkes oder die Amplitude des Fußgelenkes sein. Die Fourierkoeffizienten  $a_k$  und  $b_k$  beschreiben die Charakteristik der Bewegung und sollen durch Parameterschätzung ermittelt werden. Sie lassen sich im Vektor  $\hat{\mathbf{c}}$  zusammenfassen. Die Funktionswerte von  $\cos\left(\frac{2\pi kt}{T}\right)$  und  $\sin\left(\frac{2\pi kt}{T}\right)$  an einer vorgegebenen Stelle  $t$  können vorab berechnet werden und lassen sich gemeinsam mit der Konstanten  $\frac{1}{2}$  in einer einzeiligen Matrix  $\mathbf{b}_t$  darstellen. Dann ergibt sich der Funktionswert der Fourierreihe an der Stelle  $t$  in Matrixschreibweise:

$$\hat{\mathbf{c}} = [a_0, \dots, a_k, b_k, \dots]^T \quad (8.4)$$

$$\mathbf{b}_t = \left[ \frac{1}{2}, \dots, \cos\left(\frac{2\pi kt}{T}\right), \sin\left(\frac{2\pi kt}{T}\right), \dots \right] \quad (8.5)$$

$$x_t = \mathbf{b}_t \cdot \hat{\mathbf{c}} \quad (8.6)$$

Gl.(8.6) liefert einen Skalar, es kann also nur eine Vektorkomponente des Zustandsvektors  $\mathbf{x}_t = [x_{t1}, x_{t2}, \dots, x_{ti}]^T$  repräsentiert werden. Um den ganzen Zustandsvektor  $\mathbf{x}_t$  darzustellen, werden die Fourierkoeffizienten einer jeden Zustandsvektorkomponente  $\hat{\mathbf{c}}_1 \dots \hat{\mathbf{c}}_i$  untereinander geschrieben. Die Blockdiagonalmatrix  $\mathbf{B}_t$  besitzt dann so viele Zeilen  $i$ , wie es Komponenten im Zustandsvektor gibt. Gl.(8.2) schreibt sich dann wie folgt:

$$\mathbf{x}_t = \begin{bmatrix} x_{t1} \\ x_{t2} \\ \vdots \\ x_{ti} \end{bmatrix} = \mathbf{B}_t \cdot \mathbf{c} = \begin{bmatrix} \mathbf{b}_{t1} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{b}_{t2} & & \mathbf{0} \\ \vdots & & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{b}_{ti} \end{bmatrix} \cdot \begin{bmatrix} \hat{\mathbf{c}}_1 \\ \hat{\mathbf{c}}_2 \\ \vdots \\ \hat{\mathbf{c}}_i \end{bmatrix} \quad (8.7)$$

Zur Schätzung der Parameter des Bewegungsmodells  $\mathbf{c}$  muss eine Folge von Zustandsvektoren  $\mathbf{z}_t$  vorgegeben werden, die durch das Bewegungsmodell approximiert werden

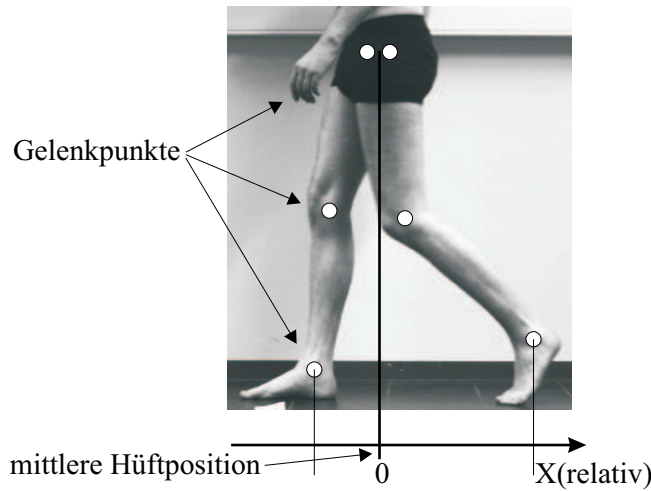


Abbildung 8.1.: Die Gelenkpunktkoordinaten werden im Folgenden relativ zur mittleren Hüftposition angegeben.

sollen. Dabei ist  $L$  die Länge der Sequenz.

$$e(\mathbf{c}) = \sum_t^L \|\mathbf{x}_t - \mathbf{z}_t\|^2 = \sum_t^L \|\mathbf{B}_t \cdot \mathbf{c} - \mathbf{z}_t\|^2 \rightarrow \min \quad (8.8)$$

Die Lösung erfolgt wieder durch Aufstellen und Lösen eines überbestimmten Gleichungssystems.

$$\begin{bmatrix} \mathbf{B}_1 \\ \mathbf{B}_2 \\ \vdots \\ \mathbf{B}_t \end{bmatrix} \cdot \mathbf{c} = \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_t \end{bmatrix} \quad (8.9)$$

Um die Approximation zu testen, wurden manuell die Hüftgelenkspositionen in verschiedenen Bildern markiert und deren Raumpunkte berechnet. Aus den beiden Gelenkpunkten der Hüfte lässt sich durch Mittelwertbildung die mittlere Hüftposition berechnen. Abbildung 8.1 zeigt die Definition relativer Gelenkpunktkoordinaten. In Abbildung 8.2 sind die absoluten sowie relativen x-Koordinaten der Trajektorien der Sprunggelenke und der Hüftposition zu sehen.

Zur Demonstration des Ansatzes wurde als Soll-Zustandsvektor  $\mathbf{z}_t$  die x-Koordinate des linken Fußgelenkes relativ zur Hüfte vorgegeben und eine Approximation des Bewegungsmodells durchgeführt. Abbildung 8.3 zeigt die Approximationsergebnisse, wenn die Anzahl der Fourierkoeffizienten und die Periodenlänge des Bewegungsmodells geändert werden. Es ist deutlich zu sehen, dass die Anzahl der Fourierkoeffizienten eher unkritisch ist. Einen erheblichen Einfluss auf das Approximationsergebnis hat aber die

## 8. (Quasi-)periodisches Bewegungsmodell

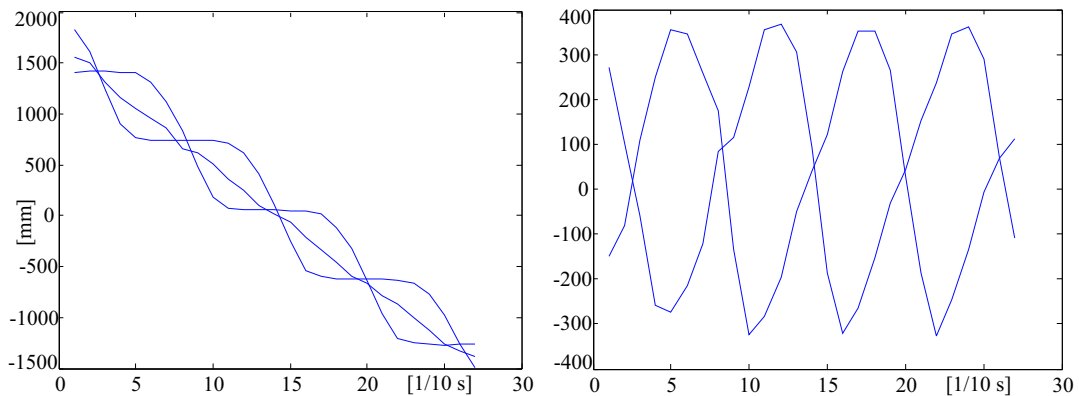


Abbildung 8.2.: Links: x-Koordinaten des linken und rechten Sprunggelenkes (alternierend), sowie der Hüfte (nahezu gerade) in Bewegungsrichtung; Rechts: Koordinaten des linken und rechten Sprunggelenkes relativ zur Hüfte

Periodenlänge der Fourierreihe. Sie muss sehr präzise vorgegeben werden, um ein optimales Approximationsergebnis zu erhalten. Unerwartet ist das Diagramm rechts oben. Es zeigt ein starkes Überschwingen, hervorgerufen durch zu viele Fourierkoeffizienten in Verbindung mit einer falschen Periodenlänge. Dieses hier zunächst nur angedeutete Problem wird ab Abschnitt 10.4.1 genauer untersucht.

In Abbildung 8.4 wurden die vier x-Koordinaten der beiden Sprung- und Kniegelenke mit der Methode der kleinsten Fehlerquadrate durch das beschriebene Bewegungsmodell approximiert. Misstrauisch macht vor allem die Asymmetrie der approximierten Knie-trajektorien. Die Hauptursache ist ein Ausreißer (drittes Sample des linken Knies). Geht man von einem normalen symmetrischen Gang aus, spricht nichts dagegen, die Anzahl der Parameter des Bewegungsmodells  $\mathbf{c}$  dadurch zu halbieren. Dies kann geschehen, indem die selben Fourierkoeffizienten für das linke und rechte Bein verwendet werden, da ja bekannt ist, dass die Bewegung der beiden Beine etwa  $180^\circ$  (oder 50 %) phasenverschoben verläuft. Natürlich können dann tatsächliche Asymmetrien des Ganges eines Probanden nicht mehr repräsentiert werden. Da es für medizinische Untersuchungen aber auch um Aussagen zur Symmetrie geht, ist ein derart vereinfachtes Modell ungeeignet. Für die Gewinnung von Näherungswerten könnte es dennoch hilfreich sein.

In Abbildung 8.5 wurde die Approximation mit den selben mit Ausreißern behafteten Daten durchgeführt, allerdings unter Verwendung des symmetrischen Bewegungsmodells. Deutlich zu erkennen ist nun die Symmetrie der Bewegungsabläufe. Wie zu erwarten war, steigt die Robustheit gegenüber Ausreißern durch Halbierung der Anzahl unbekannter Parameter.

Es wurde festgestellt, dass die exakte Periodendauer ein kritischer Parameter für die Qualität der Approximation ist. Um eine genauere Aussage treffen zu können, wurde in Abbildung 8.6 eine volle Suche über die Periodenlänge  $T$  durchgeführt. Ein Entfernen des Ausreißers änderte die Lage des Minimums bei  $T_{min} = 11.8[1/10s]$  nicht.



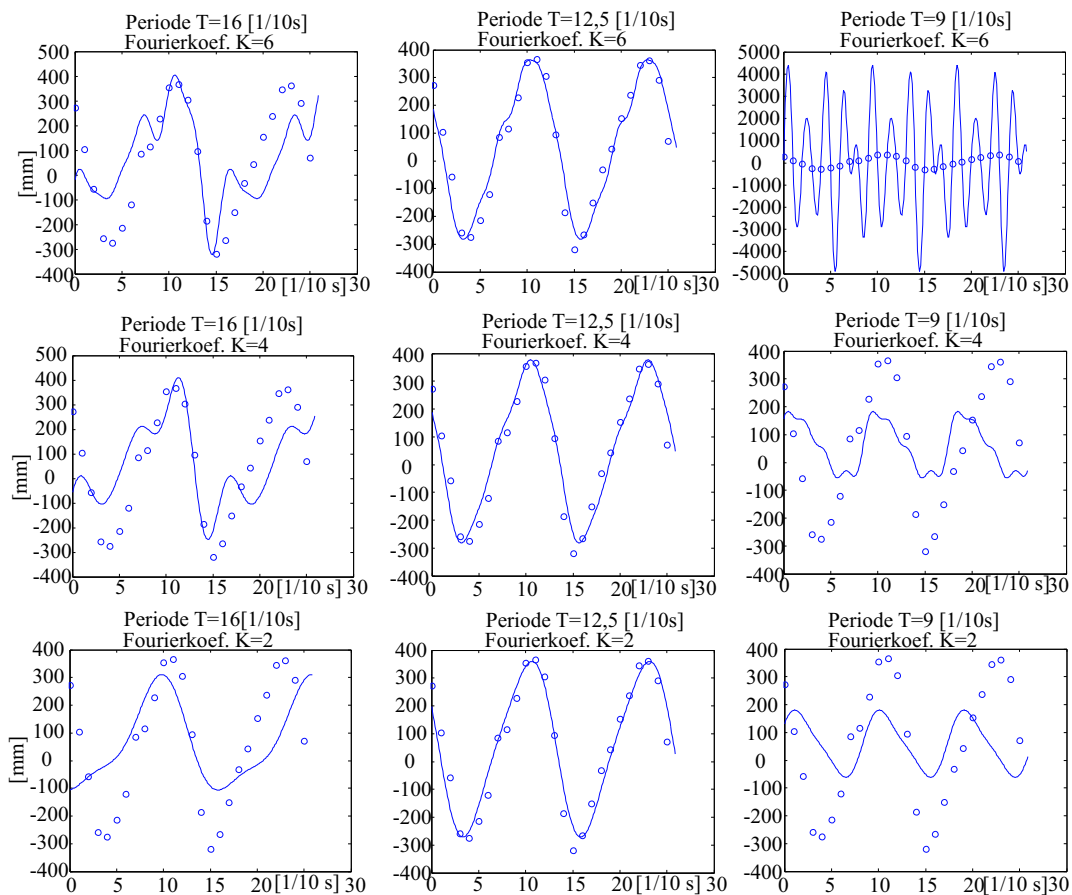


Abbildung 8.3.: Approximation relativen x-Koordinate des linken Fußgelenkes (Kreise) durch eine Fourierreihe mit verschiedenen Parametern (durchgezogen). Variiert wurden die Periodenlänge  $T$  (von links nach rechts) und die Anzahl der Fourierkoeffizienten  $K$  (von oben nach unten).

8. (Quasi-)periodisches Bewegungsmodell

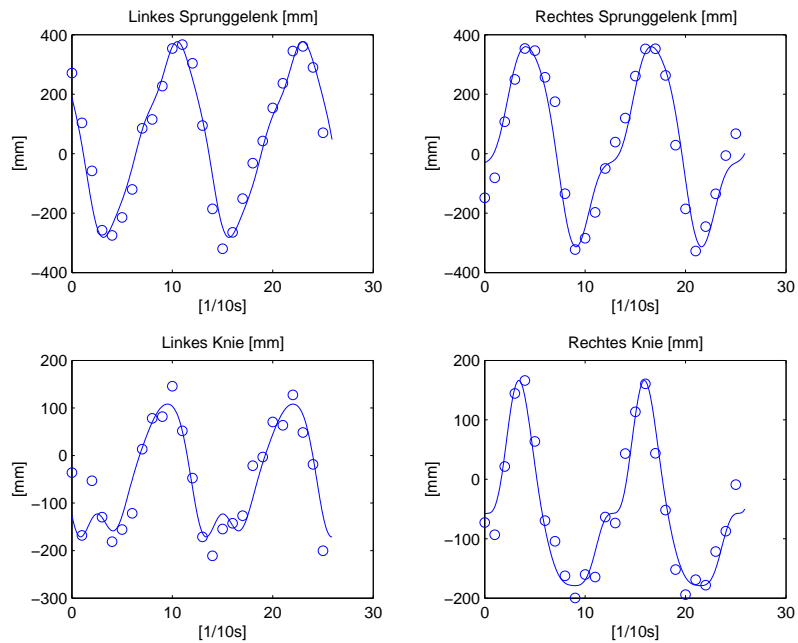


Abbildung 8.4.: Approximation der relativen Bewegung von Knie und Sprunggelenk ( $K=4$  und  $T=12.5[1/10s]$ )

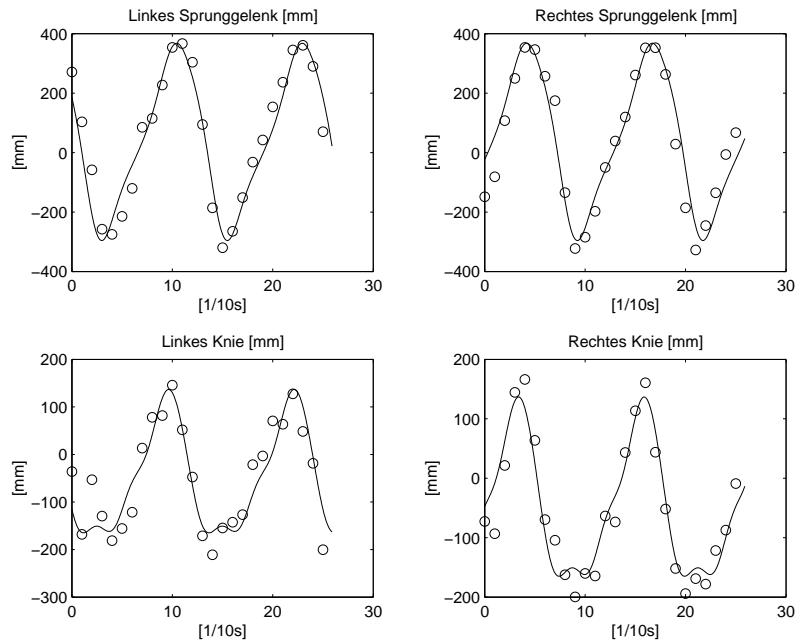


Abbildung 8.5.: Approximation der relativen Bewegung von Knie und Sprunggelenk unter Verwendung eines symmetrischen Bewegungsmodells. Es wurden nur die Hälfte der Parameter von Abbildung 8.4 im Vektor  $\mathbf{c}$  repräsentiert ( $K=4$  und  $T=12.5[1/10s]$ )

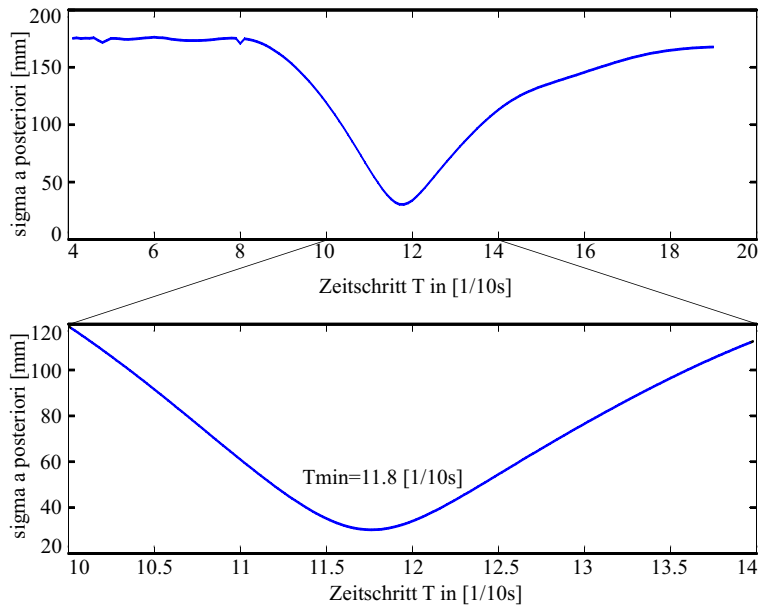


Abbildung 8.6.: Volle Suche über die Periodendauer  $T$ : Ein eindeutiges Minimum wurde bei  $T_{min} = 11.8[1/10s]$  gefunden ( $K=4$ ). Als Ursache für die kleineren Nebenminimas stellte sich ein Rangabfall bei der Lösung des Gleichungssystems heraus.

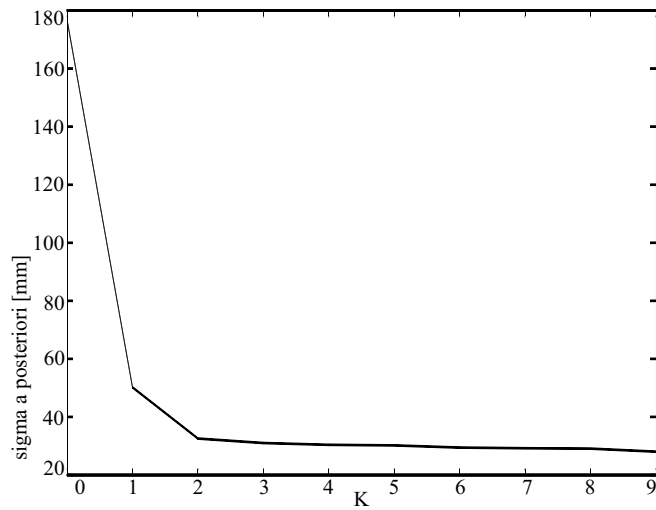


Abbildung 8.7.: Approximationsfehler bei unterschiedlicher Länge der Fourierreihe ( $K$ )

## 8.1. Approximation des Bewegungsmodells mit einem nichtlinearen Körpermodell

Es wurde gezeigt, dass sich ein gegebener Bewegungsablauf durch eine Fourierreihe mehr oder weniger genau approximieren lässt. Es wurde bisher kein Skelettmodell verwendet, statt dessen wurden direkt die x-Koordinaten der gemessenen Gelenkpunkte durch das Bewegungsmodell approximiert. Gewöhnlich lassen sich durch eine Fourierreihe gut periodische Bewegungen um einen Mittelwert (DC oder Gleichanteil) repräsentieren. Um einen konstanten Mittelwert zu erhalten, mussten alle Koordinaten relativ zum Hüftgelenk angegeben werden. Erleichtert wurde dies durch eine Hauptbewegungsrichtung des Probanden entlang der x-Achse. Durch dieses Modell wäre zum Beispiel ein im Kreis laufender Proband nicht so einfach darstellbar. Es müsste ein Koordinatensystem erzeugt werden, das die wesentlichen Bewegungen, wie Translation und Rotation relativ zum Weltkoordinatensystem kompensiert und in dem sich alle Gelenkpunkte relativ darstellen lassen. Hauptproblem dabei dürfte die Kompensation der Rotation des Probanden um die z-Achse sein. Möglich wäre das vielleicht unter Auswertung des Geschwindigkeitsvektors des Probanden.

Zwar wurde sowohl das sehr einfache Skelettmodell, bei dem Gelenkpunkte durch Strecken miteinander verbunden sind, als auch das beschriebene hierarchische Skelett differenziell formuliert, aber eine relative Formulierung der Bewegungsgrößen erlaubt nur das hierarchische Skelett. Das hierarchische Skelett verwendet als Repräsentation der Bewegung der Gliedmaßen direkt die Gelenkwinkel. Da diese relativ zum Vorgänger-Körperteil formuliert sind, wird eine Rotation des Probanden bereits durch das Modell kompensiert. Die zuvor gezeigte differenzielle Formulierung erlauben es, direkt Zielpunkte in den Kamerabildern anzugeben. Das wiederum gestattet es selbst die Gelenkpunkte zu verwenden, die nur in einer einzigen Kamera beobachtet werden konnten. Würde das Modell an 3D-Raumpunkte approximiert werden müssen, könnte diese Information nicht verwendet werden, weil sich die Koordinaten eines Gelenkpunktes nicht durch eine Kamera alleine berechnen ließen.

Um das periodische Bewegungsmodell mit dem bisherigen Mensch-Modell verwenden zu können, müssen die vorhandenen Gleichungen etwas umgestellt werden. Die lineare Gl.(8.2) des periodischen Bewegungsmodells ist auch für differenzielle Änderungen  $\Delta \mathbf{c}$  gültig.

$$\Delta \mathbf{x}_t = \mathbf{B}_t \cdot \Delta \mathbf{c} \quad (8.10)$$

$$\mathbf{x}_t + \Delta \mathbf{x}_t = \mathbf{B}_t \cdot [\mathbf{c} + \Delta \mathbf{c}] \quad (8.11)$$

$$\mathbf{x}_t + \Delta \mathbf{x}_t = \mathbf{B}_t \cdot \mathbf{c} + \mathbf{B}_t \cdot \Delta \mathbf{c} \quad (8.12)$$

Neben dem linearen Bewegungsmodell Gl.(8.2) existiert ein nichtlineares Körpermodell mit einem nichtlinearen Kameramodell. Ausgehend von einem Näherungswert des Zustandsvektors  $\mathbf{x}_t$  können damit die Koordinaten der Modellpunkte in den Kamerabildern  $\mathbf{p}_t = \mathbf{p}(\mathbf{x}_t)$  synthetisiert werden. Zur Schätzung des unbekanntes Zustandsvektors

## 8.1. Approximation des Bewegungsmodells mit einem nichtlinearen Körpermodell

$\mathbf{x}_{tn} = \mathbf{x}_t + \Delta\mathbf{x}_t$  aus seinem Näherungswert  $\mathbf{x}_t$  wird klassisch der funktionale Zusammenhang durch Linearisierung in einer Jacobimatrix ( $\mathbf{A}_t = \frac{\partial \mathbf{p}(\mathbf{x}_t)}{\partial \mathbf{x}_t}$ ) aufgestellt. Für jeden Zeitschritt kann dann ein überbestimmtes Gleichungssystem nach  $\Delta\mathbf{x}_t$  gelöst werden, wenn die Sollkoordinaten  $\mathbf{s}_t$  der Modellpunkte in den Kamerabildern gegeben sind.

$$\mathbf{A}_t \cdot \Delta\mathbf{x}_t = \mathbf{s}_t - \mathbf{p}_t \quad (8.13)$$

Dies ist die Standardmethode, bei der jeder Zeitschritt unabhängig von den anderen berechnet wird. Entsprechend hoch ist die Anzahl der Unbekannten. Eine Verringerung wird erreicht, wenn die Anzahl der Vektorkomponenten in  $\mathbf{c}$  kleiner ist als die Anzahl der Vektorkomponenten von  $\mathbf{x}_t$  multipliziert mit der Sequenzlänge. Durch Einsetzen von Gl.(8.10) in Gl.(8.13) entsteht für jeden Zeitschritt die folgende Gleichung:

$$[\mathbf{A}_t \cdot \mathbf{B}_t] \cdot \Delta\mathbf{c} = \mathbf{s}_t - \mathbf{p}_t \quad (8.14)$$

Diese Gleichung wird nun für jeden Zeitschritt der Sequenz in einem gemeinsamen Gleichungssystem aufgestellt. Dieses lässt sich mit der Methode der kleinsten Fehlerquadrate nach  $\Delta\mathbf{c}$  auflösen. Ausgehend von einem anfänglichen Bewegungsmodell  $\mathbf{c}_n$  entsteht nach einer Iteration ein besseres Bewegungsmodell  $\mathbf{c}_{n+1} = \mathbf{c}_n + \Delta\mathbf{c}$ .

Bei der Lösung des überbestimmten Gleichungssystems mit dem Algorithmus aus dem Anhang C berechnen sich die Standardabweichungen der unbekannt Parameter nach Gl.(C.9). Dazu gehören auch die Parameter des Bewegungsmodells  $\mathbf{c}$ , deren Standardabweichungen zu einem Vektor  $\sigma_c = [\hat{s}_1, \hat{s}_i, \dots, \hat{s}_u]^T$  zusammengefasst werden können. Leichter als die Parameter  $\mathbf{c}$  des Bewegungsmodells sind aber die Gelenkwinkel  $\mathbf{x}_t$  zu interpretieren. Es ist deshalb auch interessant, die Standardabweichungen  $\sigma_{x_t}$  der Gelenkwinkel anzugeben. Nach dem Fehlerfortpflanzungsgesetz lassen sich die Standardabweichungen  $\sigma_{x_t}$  der Gelenkwinkel eines Zeitschrittes  $t$  aus den Standardabweichungen der Parameter des Bewegungsmodells berechnen:

$$\sigma_{x_t} = \mathbf{B}_t \cdot \sigma_c \quad (8.15)$$

### 8.1.1. Experiment

Beim Aufstellen der nichtlinearen Beobachtungsgleichungen schleichen sich leicht Fehler ein. Um die zuvor beschriebenen Gleichungen zu testen, wurden zunächst die Gelenkpunkte der Hüft-, Knie- und Sprunggelenke in allen Kamera-Bildern und in allen Zeitschritten manuell vorgegeben. Die Näherungswerte für Translation und Rotation des Skeletts wurden durch den Schwerpunkt der Gelenkpunkte und seinem geglätteten Geschwindigkeitsvektor (Bewegungsrichtung=Blickrichtung) ermittelt.

Mit Hilfe der beschriebenen Gleichungen wurden sechs Gelenkpunkte des hierarchischen Skelettmodells an die im Bild vorgegebenen Punkte iterativ approximiert. Nach etwa 10 Iterationen änderten sich die geschätzten Bewegungsparameter nicht mehr wesentlich. Bei der Ausgleichung wurden 33 Zeitschritte aus vier Kameras berücksichtigt.

## 8. (Quasi-)periodisches Bewegungsmodell

Die Approximationen wurden zweimal durchgeführt, einmal ohne das beschriebene periodische Bewegungsmodell und einmal mit dem periodischen Bewegungsmodell ( $K=2$ ,  $T=11.8$ ). Im letzteren Fall wurden als unbekannte Parameter verwendet:

- sequenzvariabel: Translation in x- und y-Richtung
- sequenzvariabel: Rotation um z
- zyklisch: Translation entlang der z-Richtung (Auf- und Abwärtsbewegung)
- zyklisch: je zwei Mal Hüft- und Kniegelenkwinkel (Beugen und Strecken)
- sequenzvariabel: je zwei Hüftwinkel (Abspreizen und Heranziehen)

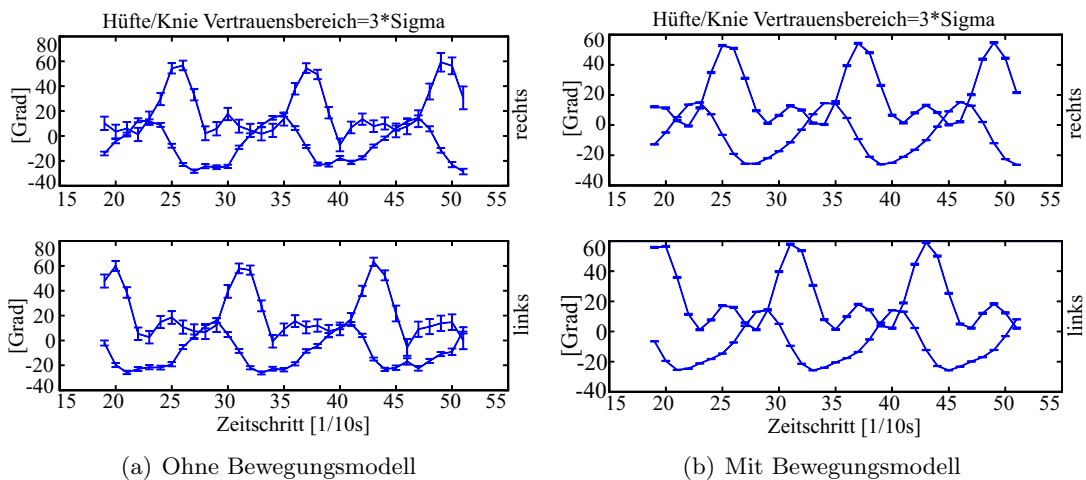


Abbildung 8.8.: Gelenkwinkelverläufe ohne (a) und mit zyklischem (b) Bewegungsmodell. Zu sehen ist, dass durch das Bewegungsmodell die Standardabweichungen deutlich gesenkt werden können.

In Abbildung 8.8 sind die Ergebnisse zu sehen. Es ist zu erkennen, dass die Vertrauensintervalle  $3 \cdot \sigma$  der Gelenkwinkel durch das Bewegungsmodell deutlich verkleinert werden können. Ganz allgemein wirken die Gelenkwinkelverläufe auch etwas glatter, ob dies jedoch der Realität entspricht, lässt sich daraus leider nicht ableiten.

Das Bewegungsmodell approximiert die Gelenkwinkel recht gut (Abbildung 8.9). Die maximale Abweichung zwischen beiden beträgt etwa  $10^\circ$ , ist im Durchschnitt aber deutlich kleiner. Zeitschritt 46 scheint ein Ausreißer zu sein, da er sowohl im Hüftgelenkwinkel, als auch im Kniewinkel schlecht approximiert werden konnte. Zu bemerken ist, dass Messwerte ab Zeitschritt 43 unsicher sind, weil dort der Proband lediglich von zwei Kameras gesehen werden konnte.

## 8.1. Approximation des Bewegungsmodells mit einem nichtlinearen Körpermodell

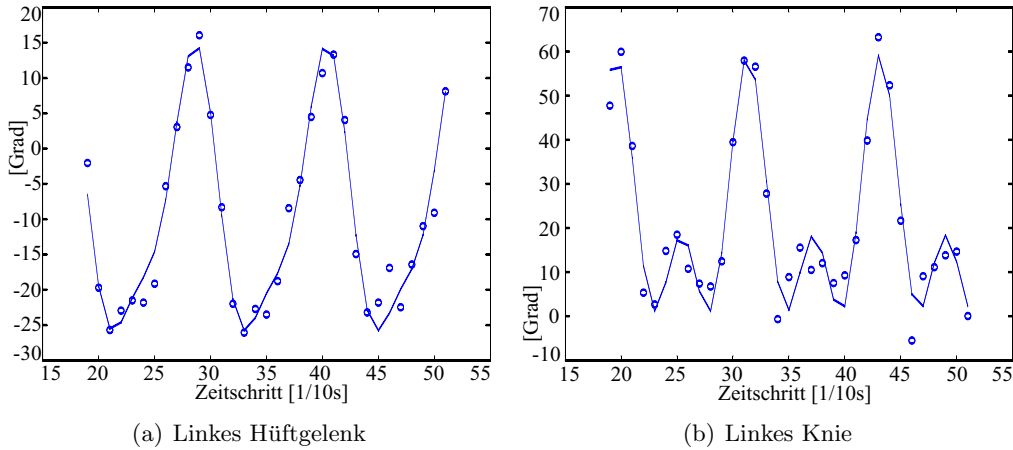


Abbildung 8.9.: Gelenkwinkelverläufe ohne (Kreise) und mit (Linie) zyklischem Bewegungsmodell. Die maximale Abweichung zwischen beiden beträgt etwa  $15^\circ$ . Unsicher sind die Werte ab Zeitschritt 43, da sich der Proband bereits außerhalb des Sichtbereiches von zwei Kameras bewegte und demzufolge nur noch von zwei Kameras beobachtet werden konnte.

### 8.1.2. Berechnung der Periodendauer in einer Ausgleichung

Die Versuchsreihen in Abbildung 8.3 haben gezeigt, dass das Bewegungsmodell empfindlich auf eine falsche Periodendauer reagiert. Es ist möglich die Periodendauer selbst als Unbekannte in die Ausgleichsrechnung einzuführen, indem ein guter Näherungswert vorgegeben wird. Dazu werden die Punktkoordinaten analytisch oder numerisch nach der Periodendauer abgeleitet.

$$\mathbf{x}_t = \mathbf{x}(t, T) = \mathbf{B}(t, T) \cdot \mathbf{c} \quad (8.16)$$

$$\mathbf{p}(t, T) = \mathbf{p}(\mathbf{x}_t) = \mathbf{p}(\mathbf{B}(t, T) \cdot \mathbf{c}) \quad (8.17)$$

$$\frac{\partial \mathbf{p}(t, T)}{\partial T} = \frac{\partial \mathbf{p}(\mathbf{x}_t)}{\partial \mathbf{x}_t} \cdot \frac{\partial \mathbf{x}(t, T)}{\partial T} = \mathbf{A}_t \cdot \frac{\partial \mathbf{x}(t, T)}{\partial T} \quad (8.18)$$

$$\frac{\partial \mathbf{x}(t, T)}{\partial T} \approx \frac{\Delta \mathbf{x}(t, T)}{\Delta T} = \left[ \frac{\mathbf{B}(t, T + \Delta T) - \mathbf{B}(t, T)}{\Delta T} \right] \cdot \mathbf{c} \quad (8.19)$$

$$\frac{\partial \mathbf{p}(t, T)}{\partial T} \approx \mathbf{A}_t \cdot \left[ \frac{\mathbf{B}(t, T + \Delta T) - \mathbf{B}(t, T)}{\Delta T} \right] \cdot \mathbf{c} \quad (8.20)$$

Ein Problem mit linearen Abhängigkeiten zwischen Periodendauer und Bewegungsparametern hat sich nicht gezeigt. Die Periodendauer erlaubt anschließend die Berechnung der mittleren Schrittfrequenz.

## 8.2. Methoden der Repräsentation quasiperiodischer Bewegungsgrößen

Für den Fall, dass der menschliche Gang sich nur schlecht durch ein periodisches Modell repräsentieren lässt, sollen einige Alternativen diskutiert werden. Die Versuchsreihen zuvor haben gezeigt, dass das Bewegungsmodell empfindlich auf eine falsche Periodendauer reagiert. Es lässt sich daraus Schlussfolgern, dass eine sehr genaue Synchronisierung des Bewegungsmodells mit der Bewegung des Probanden erfolgen muss. Die zu Grunde liegende Annahme war jedoch weiterhin die Existenz eines periodischen Laufzyklus, der sich durch eine zeitliche Synchronisierung mit der Probandenbewegung in Übereinstimmung bringen lässt. Die genaue *zeitliche* Synchronisierung ist jedoch nur die eine Seite. Auf der anderen Seite muss auch beachtet werden, dass sich die *Amplitude* der Gelenkwinkel von Schritt zu Schritt ebenfalls ändern kann. Der bisherige Ansatz vernachlässigt das, denn es werden die selben Fourierkoeffizienten, welche die Amplitude codieren, für die gesamte Sequenz verwendet.

Die Amplitude der Gelenkwinkel dürfte dabei in einem engen Zusammenhang mit der Schrittlänge  $s_l$  stehen. Die Schrittlänge  $s_l$  ließe sich zum einen aus der Differenz der Koordinaten des Sprunggelenkes zweier Standpunkte berechnen. Zum anderen existiert ein Zusammenhang zwischen Schrittlänge  $s_l$ , der Schrittfrequenz  $f_s$  und der mittleren Geschwindigkeit  $v$  des Probanden (siehe Gl.(2.2)).

$$w = \frac{v}{f_s} \quad (8.21)$$

Unter diesen Bedingungen könnte ein Korrekturterm verwendet werden, der die Fourierkoeffizienten und damit die Amplitude der Gelenkwinkelverläufe, entsprechend der Schrittlänge des Probanden skaliert. Dazu muss aber der Zusammenhang zwischen Schrittlänge und den Fourierkoeffizienten bekannt sein. Der Zusammenhang konnte in dieser Arbeit leider noch nicht aufgezeigt werden. Eigentlich ist es weniger die Frage, ob ein Zusammenhang existiert (das liegt eigentlich auf der Hand) als vielmehr, welche Komplexität die Modelle für diesen Zusammenhang besitzen müssen. Im einfachsten Fall kann versuchsweise ein linearer Zusammenhang angenommen werden. Im Ergebnis entsteht ein normierter Parametersatz, der für unterschiedliche Schrittfolgen und Schrittlängen konstant bleibt. Die Vergleichbarkeit der Messreihen wird dadurch vereinfacht, da die sehr variablen Größen, wie Schrittfolgen und Schrittlänge, nur noch geringen Einfluss auf die Parametersätze haben.

Für den Fall, dass der Gang nur annähernd periodisch ist, darf für jeden Schritt nicht einfach der selbe Parametersatz verwendet werden. Die Anzahl der Parameter für den aufgezeichneten Gang wird sich dann aber deutlich erhöhen und damit sowohl die Robustheit als auch die Genauigkeit der Parameterschätzung verkleinern. Im Folgenden sollen einige alternative Lösungen diskutiert werden. Sie basieren wieder auf der Repräsentation des Ganges im Frequenzraum:

- Separieren von periodischem und aperiodischem Anteil.



- Für jeden Schritt ein eigenes Bewegungsmodell verwenden.
- Eine Fourierrepräsentation mit einer vielfachen Periodendauer der *Schrittdauer* des Probanden.

### 8.2.1. Separation von periodischem und aperiodischem Anteil

Die Separation von periodischem und aperiodischem Anteil ist ein interessanter Ansatz, weil sie eine getrennte Verarbeitung der beiden Größen ermöglicht. Dazu wird zunächst eine Ausgleichsrechnung mit periodischem Bewegungsmodell durchgeführt, wie zuvor beschrieben. Im Ergebnis entsteht der periodische Anteil der Bewegung, als relativ kleiner Parametersatz des Bewegungsmodells im Frequenzraum. Aus den Fourierkoeffizienten wird nun eine periodische Bewegung des 3D-Modells synthetisiert. Das 3D-Modell ist dadurch in jedem Zeitschritt gut initialisiert. Ausgehend von diesen Startwerten, werden die Parameter des 3D-Modells in jedem Zeitschritt separat ausgeglichen und die Änderungen akkumuliert. Für jeden Zeitschritt entsteht dadurch ein eigener Parametersatz, der die Abweichungen zum periodischen Modell beschreibt. Damit können also direkt die Abweichungen zwischen periodischen Bewegungsmodell und aperiodischen Komponenten berechnet werden. Der Preis ist jedoch eine geringere Genauigkeit der aperiodischen Bewegungsgrößen, da zur Berechnung des aperiodischen Anteils keine zeitliche Integration der Informationen der Bildsequenz erfolgen kann. Abweichungen zum periodischen Modell lassen sich durch Nebenbedingungen klein halten, das dürfte die Berechnung des aperiodischen Anteils stabilisieren.

### 8.2.2. Ein separates Bewegungsmodell für jeden Schritt

Theoretisch kann für jeden Schritt ein eigenes Bewegungsmodell verwendet werden. Dann darf jeder Schritt ein Unikat darstellen. Es ist dadurch möglich die Amplitude der Gelenkwinkel für jeden Schritt variabel zu halten. Änderungen der Schrittweite können somit deutlich leichter erfasst werden. Diese Form ist voraussichtlich deutlich weniger empfindlich gegen eine ungenaue Angabe der Periodendauer  $T_s$ , weil der Definitionsbereich des Bewegungsmodells jeweils auf eine Periode beschränkt bleibt. Die Interpretation der Fourierkoeffizienten bleibt relativ einfach, weil für jeden Schritt ein separater Parametersatz berechnet wird. Es entsteht keine Vermischung der Fourierkoeffizienten aufeinander folgender Schritte, im Gegensatz zu den Methoden, die in den folgenden Abschnitten erläutert werden. Unter der Annahme, dass aufeinander folgende Schritte zumindest ähnliche Fourierkoeffizienten besitzen, lassen sich aus einem berechneten Schritt voraussichtlich sehr gute Näherungswerte für den nächsten Schritt erzeugen. So dass selbst bei einem sehr ausgefallenen Gangmuster die automatische Auswertung einer langen Sequenz ermöglicht wird, wenn der erste Schritt unter Verwendung manueller Punkteingaben berechnet wurde. Da jeder Schritt in einem eigenen Gleichungssystem ausgeglichen wird, muss nicht mehr die ganze Sequenz mit einem Mal ausgewertet wer-

## 8. (Quasi-)periodisches Bewegungsmodell

den. Es ist nun möglich selbst sehr lange Bildsequenzen, wie sie unter Verwendung eines Laufbandes entstehen können, Schritt für Schritt auszuwerten.

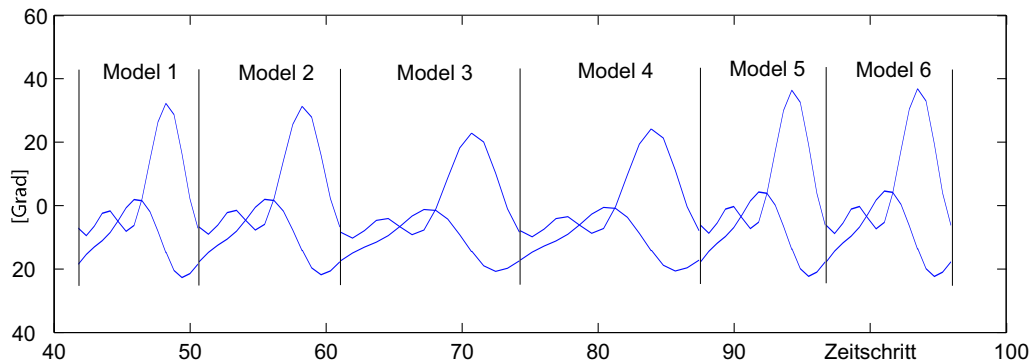


Abbildung 8.10.: Es wird ein separates Bewegungsmodell für jeden Probandenschritt verwendet und für jede Periode ein separater Satz von Fourierkoeffizienten berechnet.

Demgegenüber stehen einige Nachteile. Die vorhandenen Redundanzen zwischen aufeinander folgenden Schritten können nicht genutzt werden und es wird für jeden Schritt ein neuer Parametersatz benötigt. Dadurch steigt die Anzahl der unbekannt Parameter erheblich und es ist eine kleinere Genauigkeit/Robustheit zu erwarten, als bei Verwendung eines Bewegungsmodells für die ganze Bildsequenz. Das Bewegungsmodell sollte jedoch gerade zur Verringerung der Parameteranzahl verwendet werden. Somit ist eine Reduktion der Parameteranzahl eigentlich nur noch möglich, wenn die Abtastfrequenz des Bildaufnahmesystems sehr viel größer als die Schrittfrequenz ist.

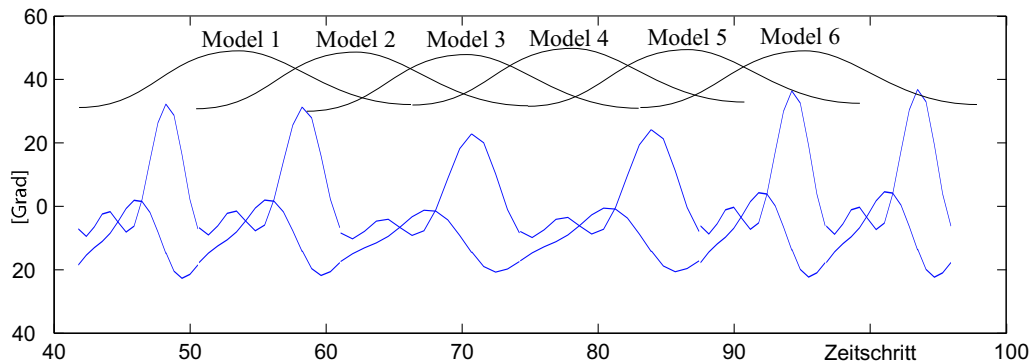


Abbildung 8.11.: Das Konzept der Fensterung aus der Signalverarbeitung erlaubt Bewegungsmodelle die sich überlappen. Die Überlappungsbereiche können beliebig groß gewählt werden, damit lässt sich die zu berechnende Anzahl der Bewegungsparameter in einem weiten Bereich variieren.

In einer leicht abgewandelten Form ist der Ansatz in der Lage das aus der Signalverarbeitung bekannte Konzept der Fensterung auf die Ganganalyse zu übertragen (siehe

Abbildung 8.11). Die Auswertung der Bildsequenz erfolgt dann in so genannten Zeit-’Fenstern’, die sich gegenseitig gut überlappen. Bei der Berechnung der Fourierkoeffizienten werden dann die Zeitschritte an den Rändern des Zeit-Fensters deutlich weniger gewichtet, als die Zeitschritte in der Mitte eines Zeit-Fensters. Durch die Überlappungen der Zeitfenster ändern sich nach einer Korrektur des Phasenwinkels (Verschiebungssatz) die Fourierkoeffizienten von Fenster zu Fenster nur wenig. Das liegt an der zeitlichen Nähe aufeinander folgender Fenster. Dadurch lassen sich deutlich bessere Näherungswerte für die Auswertung des nächsten Zeitfensters bereitstellen.

Nach Auswertung einer Bildsequenz entsteht nicht nur ein Laufzyklus, sondern eine ganze Reihe von Laufzyklen, die dann geeignet visualisiert und weiterverarbeitet werden müssen.

### 8.2.3. Fourierreihe mit vielfacher Periodendauer der Schrittfrequenz

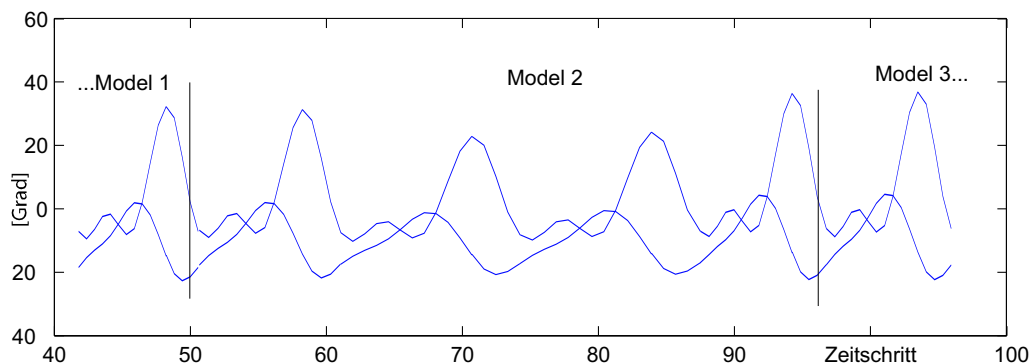


Abbildung 8.12.: Es wird nur ein Bewegungsmodell mit der Periode eines Vielfachen einer mittleren Schrittfrequenz verwendet.

Es ist möglich die Fourierreihe mit dem Vielfachen der Schrittfrequenz und einer erhöhten Anzahl von Fourierkoeffizienten zu verwenden. In diesem Fall würde wieder das ursprüngliche Bewegungsmodell nach Gl.(8.3) verwendet werden. Diesmal jedoch mit einer Periodendauer, die dem Vielfachen einer zu berechnenden mittleren Schrittfrequenz entspricht.

$$T = k \cdot \frac{1}{f_s} \quad \text{mit} \quad k > 1 \quad (8.22)$$

Dadurch werden zusätzliche ’Unterwellen’ des Signales erlaubt. Das Problem der zeitlichen Synchronisierung des Bewegungsmodells wird in diesem Fall zwar nicht gelöst, für jeden Schritt können jedoch geringfügige Variationen zwischen den Schritten repräsentiert werden. Es ist zu hoffen, dass dadurch auch in jeder Periode eine gewisse ’Synchronisierung’ quasi indirekt erfolgt.

Der Hauptvorteil dieser Methode besteht darin, dass sie nicht extra implementiert werden muss, eine Verdoppelung der Periodendauer reicht aus. Weiterhin werden die vor-

## 8. (Quasi-)periodisches Bewegungsmodell

handenen Redundanzen, die zwischen benachbarten Schritten bestehen, auch tatsächlich benutzt. Außerdem können Schwankungen der Amplitude der Gelenkwinkel durch eine unterschiedliche Schrittlänge repräsentiert werden. Soll jedoch eine normierte Darstellung der Gelenkwinkel gemäß dem Laufzyklus erfolgen, ist eine Repräsentation bezogen auf einen Schritt unumgänglich. Dies erfordert dann doch wieder eine genaue Berechnung der Schrittdauer  $T_s$ . Das kann dann jedoch nach der eigentlichen Ausgleichung durch Zeitanpassung der gewonnen Gelenkwinkelverläufe geschehen.

### 8.3. Die Schrittzahl

Die Abtastfrequenz des Multikamerasystems und die Schrittfrequenz des Probanden sind unabhängig voneinander. Weiterhin ist die Schrittfrequenz des Probanden über der Zeit nicht unbedingt als konstant anzunehmen. Im Gegensatz zum Multikamerasystem verwendet ein Proband keinen hochpräzisen Taktgenerator. Soll jedoch mit einem periodischen Modell der Bewegung gearbeitet werden, dann sollte dieses Modell mit der Schrittfrequenz des Probanden und nicht mit der Taktfrequenz des aufnehmenden Systems synchronisiert werden.

Zuvor wurde einfaches periodisches Bewegungsmodell beschrieben. Es verwendet für jede Vektorkomponente eine Fourierreihe nach Gl.(8.3). Der Term  $\frac{t}{T}$  erzeugt für jeden Zeitpunkt  $t$  eine reelle Zahl  $\theta_t$  für das Bewegungsmodell.

$$\theta_t = \frac{t}{T} \quad (8.23)$$

Wird diese Zahl abgerundet, lässt sie sich als eine fortlaufende Nummerierung der Perioden oder als auch als *Schrittnummer* interpretieren (vergl. Abschnitt 2.1.1). Mit Hilfe von  $\theta_t$  lässt sich die Fourierreihe nach Gl.(8.3) etwas anders schreiben:

$$x(\theta_t) = \frac{a_0}{2} + \sum_{k=1}^K [a_k \cos(2\pi k \cdot \theta_t) + b_k \sin(2\pi k \cdot \theta_t)] \quad (8.24)$$

Da jedoch  $\theta_t$  eine reelle Zahl ist, wäre der Begriff (Schritt-) *Nummer* für  $\theta_t$  irreführend, weil unter einer Nummer im allgemeinen eine ganze Zahl verstanden wird. Es dürfte deshalb besser sein, den Parameter  $\theta_t$  als *Schrittzahl* zu bezeichnen. In der Literatur zur Ganganalyse wird das Timing von Ereignissen innerhalb eines Laufzyklus gewöhnlich in Prozent angegeben (vergleiche Abschnitt 2.1). Wird der Nachkommaanteil von  $\theta_t$  mit 100 % multipliziert, ist auch dies direkt und sehr einfach möglich.

Die Schrittzahl kann während der Ausgleichung geschätzt werden. Dazu wird für jeden Zeitschritt die Schrittzahl in das Gleichungssystem zur Modellapproximation als Unbekannte eingeführt. Das Bewegungsmodell kann dafür nach jeder Schrittzahl  $\theta_t$  numerisch

abgeleitet ( $\Delta\theta = 0.01$ ) werden.

$$\mathbf{x}(\theta_t) = \mathbf{B}(\theta_t) \cdot \mathbf{c} \quad (8.25)$$

$$\frac{\partial \mathbf{p}(\theta_t)}{\partial \theta_t} = \frac{\partial \mathbf{p}(\theta_t)}{\partial \mathbf{x}} \cdot \frac{\partial \mathbf{x}(\theta_t)}{\partial \theta_t} \quad (8.26)$$

$$\frac{\partial \mathbf{x}(\theta_t)}{\partial \theta_t} \approx \frac{\Delta \mathbf{x}(\theta_t)}{\Delta \theta_t} = \left[ \frac{\mathbf{B}(\theta_t + \Delta\theta) - \mathbf{B}(\theta_t)}{\Delta\theta} \right] \cdot \mathbf{c} \quad (8.27)$$

Eine Nebenbedingung mit kleinem Gewicht wurde aufgestellt und in jedem Zeitschritt die numerische zweite Ableitung der Schrittzahl zu null gesetzt. Das stellt dann je nach Gewicht der Gleichung eine stärkere oder schwächere Stetigkeit (Linearität) der Schrittzahl sicher.

$$\frac{\theta_{t+1} - 2 \cdot \theta_t + \theta_{t-1}}{\Delta t^2} = 0 \quad (8.28)$$

Weil die Ausgleichung aber iterativ nach dem Ansatz in Abschnitt 2.2.3 erfolgt, muss diese Gleichung wie folgt umgestellt werden.

$$\frac{(\theta_{t+1} + \Delta\theta_{t+1}) - 2 \cdot (\theta_t + \Delta\theta_t) + (\theta_{t-1} + \Delta\theta_{t-1})}{\Delta t^2} = 0 \quad (8.29)$$

$$\frac{1}{\Delta t^2} \cdot \Delta\theta_{t+1} - \frac{2}{\Delta t^2} \cdot \Delta\theta_t + \frac{1}{\Delta t^2} \cdot \Delta\theta_{t-1} = -\frac{\theta_{t+1} - 2 \cdot \theta_t + \theta_{t-1}}{\Delta t^2} \quad (8.30)$$

Durch weitere Nebenbedingungen (Addition der gewichteten Einheitsmatrix, siehe Anhang B Levenberg-Marquardt-Verfahren) konnte auch eine korrekte Lösung gefunden werden. Allerdings waren die geschätzten Standardabweichungen der Gelenkwinkelverläufe zunächst sehr schlecht. Das lag daran, dass es eine Abhängigkeit zwischen Schrittzahl und den Fourierkoeffizienten des Bewegungsmodells besteht. Die Abhängigkeit konnte durch eine weitere Nebenbedingung vermieden werden, die *eine* Schrittzahl in der Mitte der Sequenz konstant hält. Damit ist die Schrittzahl in der Mitte der Sequenz quasi willkürlich festzulegen, aber Zahlenwerte relativ zu dieser Schrittzahl werden durch das Ausgleichungssystem präzise bestimmt. Eine manuelle Änderung der mittleren Schrittzahl bewirkt dann eine Phasenverschiebung der Fourierkoeffizienten des Bewegungsmodells. Auf jeden Fall ist das Verfahren in der Lage, Schwankungen der Schrittfrequenz entlang der Sequenz zu berücksichtigen und zu kompensieren. Die Methode hat jedoch den Nachteil, dass sich die Anzahl der Unbekannten um die Sequenzlänge vergrößert. Damit steigt jedoch auch die Größe der zu invertierenden Matrix.

## 8.4. Schlussfolgerungen

Zur Reduktion der Anzahl unbekannter Parameter lässt sich ein Bewegungsmodell verwenden. Die Parameter lassen sich auch noch Bestimmen, wenn ein komplexes 3D-Körpermodell zur Repräsentation der Geometrie verwendet wird. Periodische Bewegungen lassen sich

## 8. (Quasi-)periodisches Bewegungsmodell

durch Fourierreihen approximieren, ihre Periodendauer muss jedoch sehr genau vorgegeben werden. Es wurde eine Methode vorgestellt die Periodendauer gleichzeitig mit den Bewegungsparametern in einer Ausgleichung zu berechnen. Nicht exakt periodische Bewegungen können ebenfalls durch Fourierreihen repräsentiert werden, dafür wurden verschiedene Methoden vorgestellt. Eine wichtige Größe zur zeitlichen Synchronisierung des Bewegungsmodells mit der Bewegung des Probanden ist die Schrittzahl  $\theta_t$ . Die Periodendauer kann lediglich die Schrittfrequenz des Probanden, also exakt periodische Bewegungen repräsentieren. Alternativ dazu erlaubt die Schrittzahl eine beliebige zeitliche Anpassung.

## 9. Gewinnung von Näherungswerten

Der Zusammenhang zwischen dem zu betreibenden Aufwand für die automatische Korrespondenzfindung und den Näherungswerten der Bewegungsparameter des Modells wurde bereits am Anfang des Kapitels 7 erläutert. Durch gute Näherungswerte werden kleine Suchbereiche ermöglicht, die sowohl den Suchaufwand als auch die Gefahr von Mehrdeutigkeiten reduzieren.

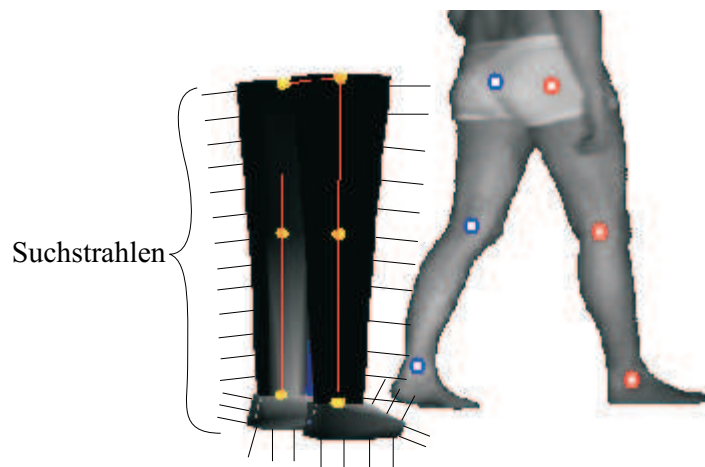


Abbildung 9.1.: Messung der Abweichungen zwischen Modell und Bild. Hier erkennt man, dass ausgehend von Kantenpunkten des Modells nach Kantenpunkten im Bild gesucht wird. Da das Modell aber noch zu schlechte Näherungswerte besitzt, ist eine korrekte Korrespondenz in diesem Zustand noch nicht zu erreichen.

### 9.1. Annahmen und Vereinfachungen

Ziel der Arbeit ist eine markerlose Ganganalyse. In der angestrebten Anwendung kann deshalb von folgenden Vereinfachungen ausgegangen werden[CMH03]:

- Es sind lediglich Laufbewegungen zu erkennen und keine beliebigen Bewegungen.
- Der Hintergrund einer jeden Kamera bleibt konstant.

## 9. Gewinnung von Näherungswerten

- Der Proband läuft in einem eingeschränkten Messvolumen, mit hoher Wahrscheinlichkeit sogar immer in die selbe Richtung.
- Es läuft immer nur ein Proband.
- Es existiert ein mittlerer Gang des Menschen der zur Initialisierung der Gelenkwinkel verwendet werden kann.

Daraus resultieren interessante Einschränkungen für den zu betreibenden Aufwand bei der Erkennung.

1. Wenn sich etwas im Bild bewegt, ist es der Proband.
2. Der Kopf des Probanden ist immer oben, ändert sein Aussehen nicht und bewegt sich mit einer mittleren Geschwindigkeit.
3. Laufbewegungen sind weitgehend periodisch.
4. Die Bewegungsrichtung des Probanden verläuft in seiner Sagittalebene.
5. Die Seitenansicht eines gehenden Probanden wiederholt sich weitgehend periodisch und annähernd symmetrisch für das linke und rechte Bein.

Das 3D-Modell erlaubt eine kombinierte Top-Down und Bottom-Up-Strategie zur Erkennung. Der grobe Ablauf der Erkennung ist folgender: Zunächst wird der bewegende Proband vom ruhenden Hintergrund segmentiert (Bottom-Up). Das Aussehen des Kopfes variiert nur wenig. Deshalb bietet er sich zur Berechnung der reinen Translation des Probanden an. Steht die Kopfposition in jedem Zeitschritt und jedem Kamerakanal fest, kann dessen dreidimensionale Bewegungsbahn näherungsweise berechnet werden. Einschränkung 4 erlaubt nun die Berechnung der näherungsweisen Orientierung des Probanden aus der Bewegungsbahn.

Durch erneutes Abtasten der Bildinformation wird die Translation des Probanden in den Bildern kompensiert. In den neu entstandenen Bildsequenzen bewegt sich der Proband fast so, als liefe er auf einem Laufband vor einer statischen Kamera. Dies erleichtert die Rekonstruktion der Beinbewegungen des Probanden. Die Beinstellung ist ein wichtiges Merkmal zur Rekonstruktion des Laufzyklus. Es wird zunächst nach Schlüssel-Bildern gesucht, die den Probanden mit möglichst geöffneten Beinen zeigen. Diese Schlüssel-Bilder erlauben die Initialisierung aktiver Konturmodelle, welche an die Bewegungsbahnen der Beine approximiert werden.

Ein typischer Laufzyklus (Top-Down) wird nun zeitlich so gestreckt oder gestaucht, dass sich das Mensch-Modell bereits synchron mit dem Probanden bewegt. In jedem Zeitschritt existieren also ausreichend gute Näherungswerte für eine Approximation zunächst an die Konturmodelle und anschließend an den Bildinhalt. In einer letzten Phase werden die genauen Amplituden der Beinbewegungen und eventuell Korrekturen der Modellform bestimmt.



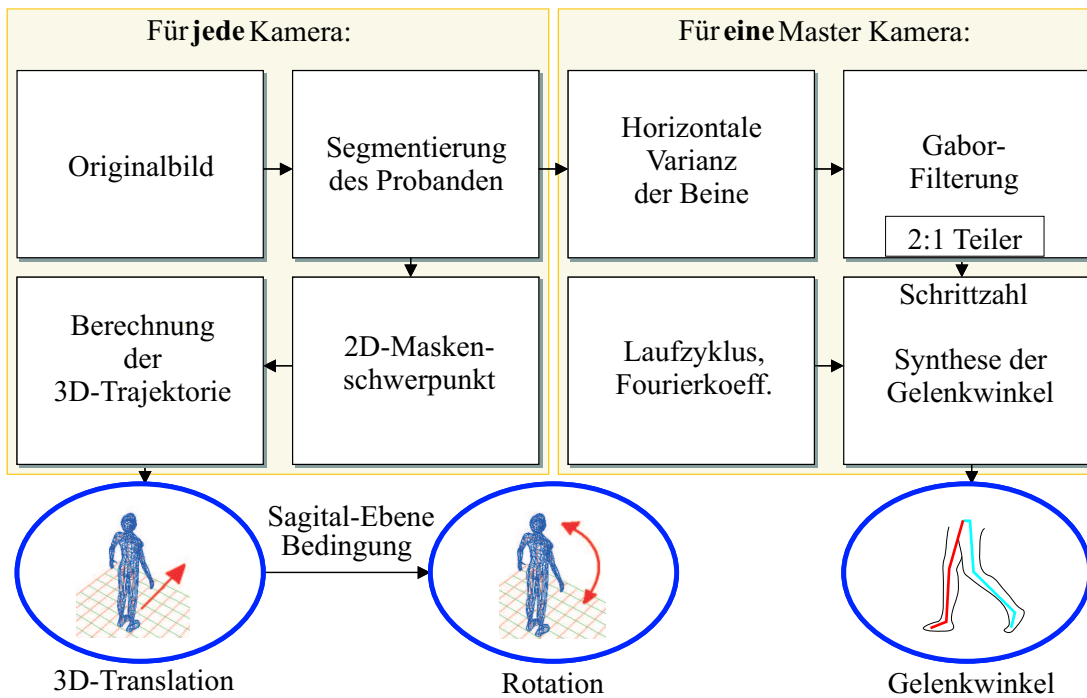


Abbildung 9.2.: Die Strategie zur automatischen Initialisierung des Modells. In jeder Kamera wird zunächst der Proband segmentiert. Daraus lässt sich die Trajektorie des Probanden im Raum ermitteln. Die Annahme, dass die Bewegungsrichtung mit der Sagittalebene zusammenfällt, erlaubt die Orientierung des Modells. Aus dem Kameraarray kann nun eine Master-Kamera gewählt werden, die den Probanden von der Seite zeigt. Ein vorgegebener Laufzyklus erlaubt nun die Initialisierung der Gelenkwinkel des Modells.

## 9.2. Initialisierung der Translationsparameter

Grundlage für diesen Schritt bilden die Vordergrundmasken  $V(x,y,t)$  des segmentierten Probanden. Wenn die Masken  $V(x,y,t)$  lediglich die Kantenpunkte des Probanden repräsentieren, reicht das für diesen Schritt aus. Für jede Bildzeile wird jetzt zunächst die 'Masse' und der (Zeilen-)Schwerpunkt berechnet.

$$m(y, t) = \sum_{x=0}^W V(x, y, t) \quad (9.1)$$

$$s(y, t) = \frac{1}{m(y, t)} \sum_{x=0}^W V(x, y, t) \cdot x \quad (9.2)$$

$t$ -Zeitschritt

$x$ -Spaltenindex, horizontale Position im Bild

$y$ -Zeilenindex, vertikale Position im Bild

$W$ -Breite des Bildes

$V(x, y, t)$ -Pixel der Vordergrundmaske (Proband) der Spalte  $x$ , Zeile  $y$  und Zeitschritt  $t$

$m(y, t)$ -Gradientenmasse der Zeile  $y$

$s(y, t)$ -Schwerpunkt der Gradientenmasse der Zeile  $y$

Der Kopf des Probanden habe im Bild die Höhe  $H$  Bildzeilen (bei Verwendung von Standardkameras z.B.  $H = 50$ ). Beginnend mit der obersten Bildzeile, wird für  $H$  aufeinander folgenden Zeilen überprüft, ob die Masse der Vordergrundgradienten  $m(y, t)$  einen bestimmten Schwellwert überschreitet. Ist dies mindestens für 90 % Zeilen der Fall, so wurde der Kopf mit hoher Wahrscheinlichkeit gefunden und die Suche kann abgebrochen werden. Durch Mittelwertbildung der Zeilenschwerpunkte ergibt sich die näherungsweise Kopfposition. Diese Suche lässt sich mit wenig Aufwand für jedes Bild der Sequenz durchführen. Außerdem erlaubt dieses Verfahren eine Aussage darüber, ob das untersuchte Bild einen Probanden enthält oder nicht. Der Suchbereich kann zeilen- oder spaltenweise eingeschränkt und damit Mehrdeutigkeiten ausgeschlossen werden. Zur genauen Bestimmung der Translation des Probanden ist dieses Verfahren jedoch zu ungenau.

Mittels Blockmatching lässt sich die geschätzte Kopfposition verbessern. Dazu wird ein allgemeines Kopftemplate in der Nähe der vorhergesagten Kopfposition mit dem aktuellen Bild der Vordergrundgradienten korreliert. Zum Matching kann die effiziente Phasenkorelation [Bau00] eingesetzt werden oder einfach die mittelwertfreie normierte Kreuzkorrelation [Mec99]. Der Korrelationskoeffizient zwischen Template  $r$  und Bildaus-

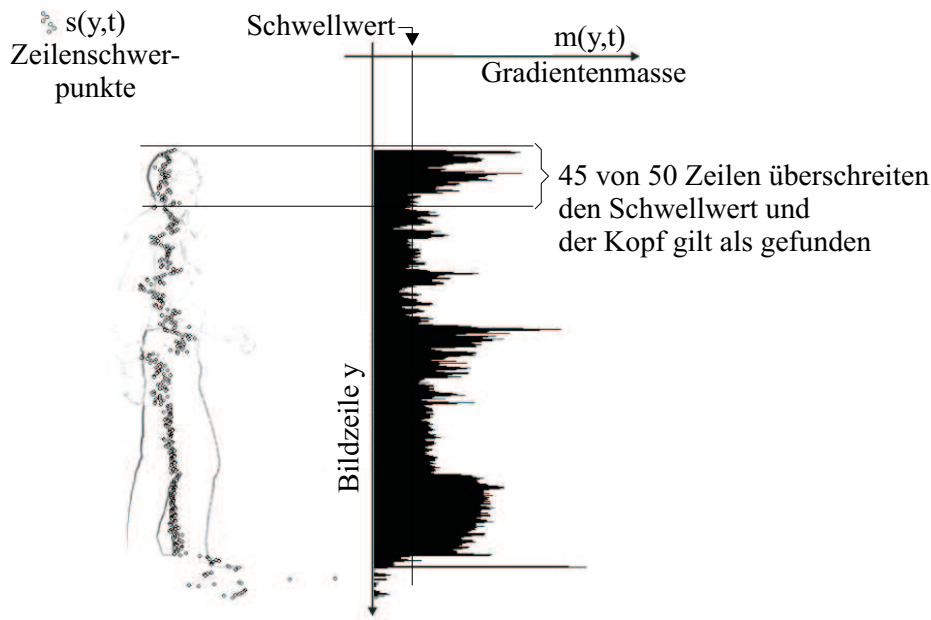


Abbildung 9.3.: Masse und Zeilenschwerpunkte der Vordergrundmaske eines Probanden

schnitt  $s(x, y)$  berechnet sich wie folgt:

$$k(x, y) = \frac{\sum_{j=-N}^N \sum_{i=-M}^M [r(i, j) - \bar{r}] \cdot [s(x + i, y + j) - \overline{s(x, y)}]}{\sqrt{\sum_{j=-N}^N \sum_{i=-M}^M [r(i, j) - \bar{r}]^2} \cdot \sqrt{\sum_{j=-N}^N \sum_{i=-M}^M [s(x + i, y + j) - \overline{s(x, y)}]^2}} \quad (9.3)$$

In Gl.(9.3) sind  $x$  und  $y$  die Koordinaten der zu untersuchenden Kopfposition.  $\bar{r}$  und  $\overline{s(x, y)}$  sind die Mittelwerte der zu vergleichenden Bildausschnitte. Die Breite  $M$  und die Höhe  $N$  des Kopftemplates begrenzen ihre Laufvariablen  $u$  und  $v$ . Es entsteht eine Gütefunktion  $k(x, y)$  deren Maximum die gefundene Kopfposition kennzeichnet. Durch Subpixelinterpolation des Maximums [Mec99] lässt sich die Genauigkeit erheblich steigern.

Mit der bekannter Kopfposition kann die Bewegung des Probanden, durch ein erneutes Abtasten der Bildinformation, kompensiert werden. Im Ergebnis lassen sich neue Bildsequenzen erzeugen, die den Probanden quasi aus einer mitlaufenden Kamera zeigen. In den eigenen Versuchen wurden Bildsequenzen im Format 256x256 Pixel erzeugt. Bei bekannter Kamerageometrie lässt sich aus der Höhe des Kopfes im Bild und dem Abstand des Probanden von einer Kamera des Multikamerasystems auch die Größe des Probanden schätzen. Das würde zumindest theoretisch die Möglichkeit bieten, die Größe des benötigten Mensch-Modells automatisch zu berechnen.

### 9.3. Initialisierung der Orientierungsparameter

Stehen die Kopfpositionen in allen Kameras fest, dann lässt sich durch einfache Triangulation auch die 3D-Kopfposition im Raum für jeden Zeitschritt berechnen. Im Ergebnis entsteht also eine 3D-Trajektorie (Bewegungsbahn) des Kopfes. Dabei können für die weitere Verarbeitung die z-Koordinaten der Trajektorie zu null gesetzt werden. Nach einem Glättungsfilter auf den Koordinaten der Trajektorie, ist die Position des Probanden in der xy-Ebene näherungsweise für jeden Zeitschritt bestimmt. Unter der Annahme, dass die Bewegungsrichtung des Probanden näherungsweise in seiner Sagittalebene liegt, lässt sich aus der Bewegungsbahn auch die Orientierung des Probanden ermitteln. Damit steht sowohl Lage als auch Orientierung des Probanden annähernd fest und es kann aus den beteiligten Kameras diejenige ausgewählt werden, die den Probanden bestmöglich von der Seite zeigt. Dies ist die Master-Kamera. Sie liefert die meiste Information für die Erkennung des Laufzyklus.

### 9.4. Methoden zur Initialisierung der Gelenkwinkel

Die wichtigste Vereinfachung dieser Arbeit ist die Annahme, dass der menschliche Gang näherungsweise periodisch ist (siehe Abschnitt 2.1.1). Weiterhin wird angenommen, dass es eine näherungsweise Übereinstimmungen des Ganges verschiedener Probanden gibt. In diesem Fall lassen sich Näherungswerte der Gelenkwinkel mit Hilfe eines allgemeinen Modells der Bewegung erzeugen, sofern es gelingt, den aktuellen Bewegungszustand des Probanden in jedem Zeitschritt zu ermitteln. Für die Repräsentation des Bewegungszustandes, in dem sich der Proband befindet, wurde in Abschnitt 2.1.1 bereits die reelle Schrittzahl  $\theta$  erwähnt und in Abschnitt 8.3 näher erläutert. Mit dem (quasi-)periodischen Bewegungsmodell nach Kapitel 8 steht ein Werkzeug zur Verfügung, das erlaubt die Schrittzahl in Gelenkwinkel des 3D-Modells umzurechnen. Sofern die Fourierkoeffizienten eines mittleren Ganges bereits bekannt sind, ist das Problem der Initialisierung der Gelenkwinkel auf die Berechnung einer reellen Schrittzahl für jeden Zeitschritt zu reduzieren. Verschiedene Methoden dafür werden im Folgenden vorgestellt.

#### 9.4.1. Manuelle Vorgabe einer Schrittfrequenz und einer Schrittzahl

Zur zeitlichen Charakterisierung einer exakt periodischen Messgröße reicht es aus, die Frequenz und die Phasenlage des Signales zu kennen. Zur Initialisierung des Bewegungsmodells des Probanden muss demzufolge eine Schrittzahl und die Periodendauer für die gesamte Bildsequenz vorgegeben werden. In der Praxis kann das dadurch geschehen, indem in der Mitte der Bildsequenz nach dem Zeitschritt gesucht wird, der den Probanden im Heel Strike des rechten Beines zeigt. Für diesen Zeitschritt wird dann die Schrittzahl null vorgegeben. Jetzt wird nach den Zeitschritten des vorhergehenden und nachfolgenden Heel Strike gesucht und daraus die Periodendauer ermittelt. Die Schrittzahlen

aller dazwischen liegenden Zeitschritte lassen sich daraus durch lineare Interpolation berechnen. Das Modell ist damit in den meisten Fällen gut initialisiert. Für die genaue Berechnung der Bewegungsparameter durch Ausgleichsrechnung ist aber eine Verbesserung zumindest der Periodendauer durch Ausgleichung erforderlich.

### 9.4.2. Manuelles Markieren der Sprunggelenke

In Abbildung 9.4 ist das Weg-Zeit-Diagramm des linken und rechten Sprunggelenkes dargestellt. Das Bild stellt den Bewegungsablauf einer neuen Bildsequenz dar, es dient also nicht zum direkten Vergleich mit den zuvor gezeigten Daten, sondern es soll zur Veranschaulichung eines einfachen Ansatzes zur Synchronisation des Bewegungsmodells mit der Probandenbewegung verwendet werden.

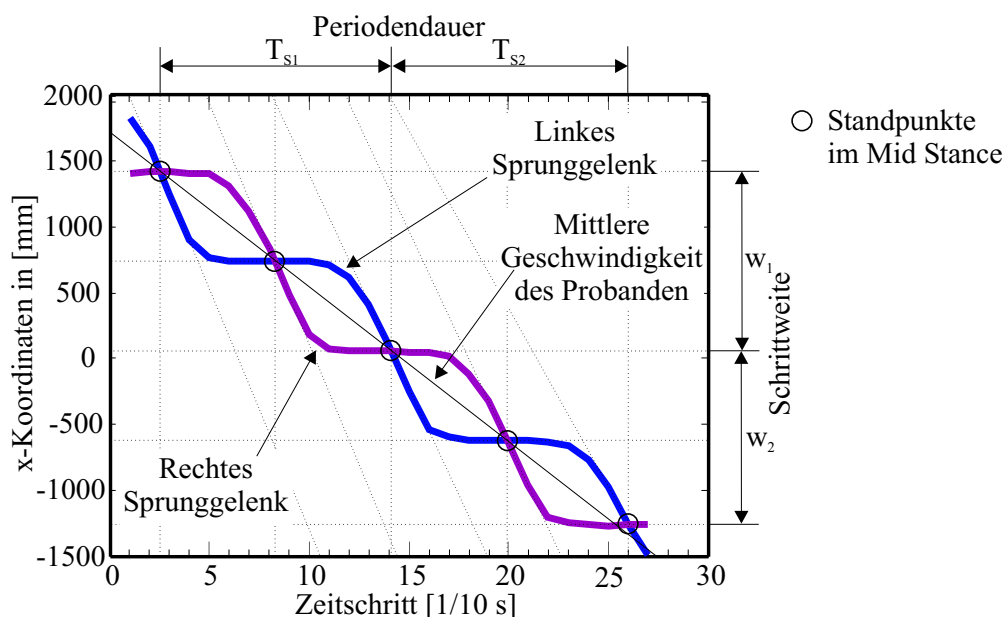


Abbildung 9.4.: Weg-Zeit-Diagramm der Sprunggelenke

Aus diesem Weg-Zeit-Diagramm lassen sich einige interessante Parameter ablesen. Der Proband bewegte sich in negativer Richtung entlang der x-Koordinatenachse des Weltkoordinatensystems. Der Anstieg der durchgezogenen Linie zeigt die mittlere Geschwindigkeit gemessen über einige Standpunkte. Um diese Linie 'schlängeln' sich die Koordinaten der beiden Sprunggelenke. In den Standphasen eines Beines ist der Anstieg im Weg-Zeit-Diagramm nahezu null, da das Standbein gewöhnlich nicht über den Boden gleitet. Die Schwungphasen eines Beines zeichnen sich durch einen besonders starken Anstieg im Weg-Zeit-Diagramm aus. Das liegt daran, dass hier die höchsten Geschwindigkeiten des Sprunggelenkes auftreten. Diese Geschwindigkeiten wurden noch einmal durch diagonale Linien verdeutlicht. Auf der rechten Ordinatenachse ist die Schrittlänge eines Beines

## 9. Gewinnung von Näherungswerten

dargestellt. Sie resultiert aus dem gemessenen Koordinatenabstand zweier aufeinander folgender Standpunkte eines Beines. Zur Festlegung des Timings des Laufzyklus könnten genau die Schnittpunkte der beiden Graphen dienen. Sie wurden durch je einen Kreis hervorgehoben. Aus der Schrittlänge und der Periodendauer entsteht das Gitter im Hintergrund des Diagramms. An der Abszisse ist deutlich zu erkennen, dass die Bewegung des Probanden (logischerweise) nicht synchronisiert mit dem Takt des Kamerasystems verläuft.

Interessant ist, dass sich aus diesen Punkten im Weg-Zeit-Diagramm das Timing des Laufzyklus in der kritischen Schwungphase berechnen lassen könnte. Dabei werden nicht einmal die vollständigen Trajektorien der Sprunggelenke benötigt. Zur Berechnung eines Schnittpunktes dürfte es vollkommen ausreichen, die Sprunggelenkskoordinaten des Standbeines in einem Zeitschritt des Mid Stance zu markieren und in die Sprunggelenkskoordinaten des Schwungbeines vor und nach dem Mid Stance. Der genaue Zeitpunkt des Mid Stance errechnet sich dann durch lineare Interpolation. Abbildung 9.5 verdeutlicht die Zusammenhänge.

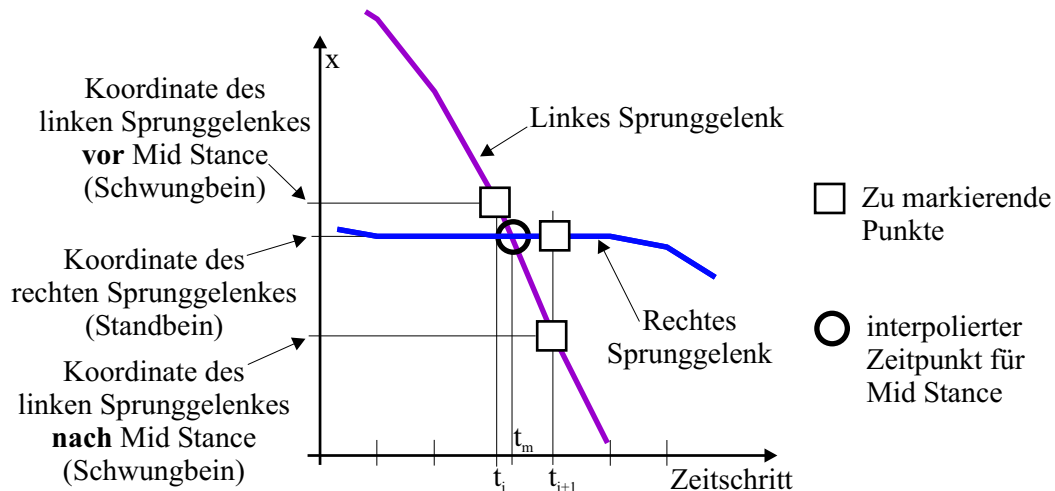


Abbildung 9.5.: Interpolation des genauen Zeitpunktes für den Mid Stance aus zwei benachbarten Zeitschritten

### 9.4.3. Automatische Initialisierung der Schrittzahlen

In Abbildung 9.6 sind zwei Vordergrundgradientenbilder aus der 'mitlaufenden' Kamera dargestellt. Links ist ein Proband mit geöffneten Beinen und rechts (9 Zeitschritte später) mit geschlossenen Beinen dargestellt. Für jede Bildzeile wurde ein Streuungsmaß  $(v(y, t))$  berechnet.

$$v(y, t) = \sum_{x=0}^W V(x, y, t) \cdot (x - m(y, t))^2 \quad (9.4)$$

#### 9.4. Methoden zur Initialisierung der Gelenkwinkel

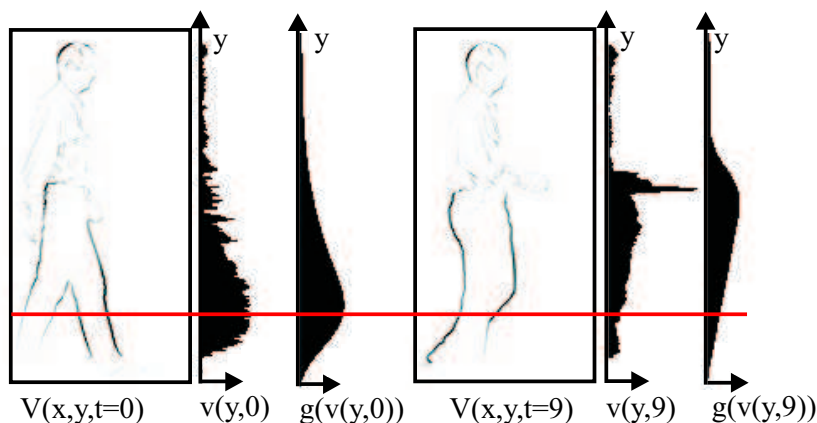


Abbildung 9.6.: Links: Zeitschritt 0 (Double-Support-Phase); Rechts: Zeitschritt 9 (Single-Stance-Phase) daneben: ungefilterte Zeilenvarianz, geglättete Zeilenvarianz

Das Ergebnis und eine geglättete Version (Gaussfilter) ist links neben jedem Bild zu sehen. Für die Erkennung der Laufphase wird nur das Streuungsmaß einer Bildzeile weiter verwendet. Diese Zeile wurde durch eine horizontale Linie hervorgehoben.

In den Bildzeilen der Beine wird der Zeilenschwerpunkt der Gradienten und deren Varianz berechnet.

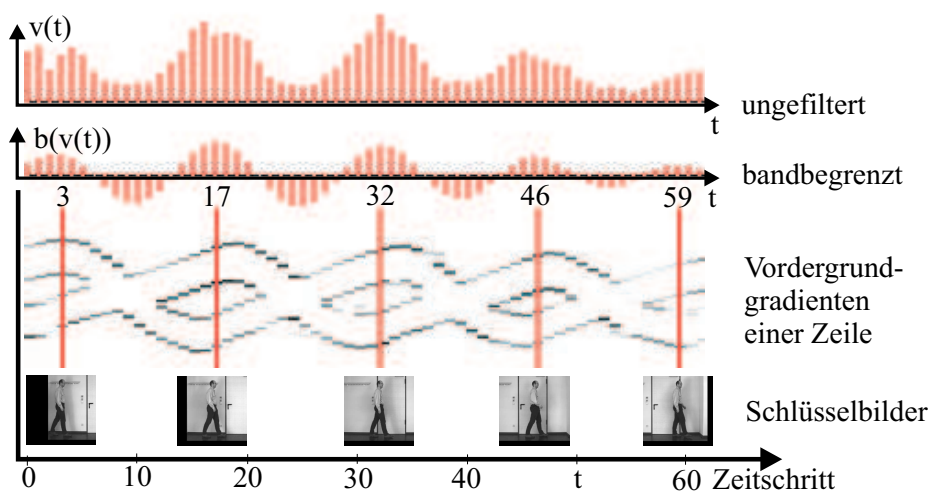


Abbildung 9.7.: Oben ist das ungefilterte Varianzkriterium zu sehen, in der Mitte die Bandpass gefilterte Version und unten die durch den Algorithmus gefundenen Key-Frames.

Über der Zeit entsteht somit ein periodisches Signal, das seine Maximas in den Zeitschritten besitzt, in denen der Proband die Beine geöffnet hatte (beide Double support Phasen). Zwei Methoden zur automatischen Berechnung von Näherungswerten der

## 9. Gewinnung von Näherungswerten

Schrittzahlen  $\theta_t$  der Sequenz aus diesem Signal sollen im Folgenden vorgestellt werden.

### Key-Frame-Technik

Die Aufgabe kann als gelöst bezeichnet werden, wenn die Zeitschritte der Sequenz bekannt sind, in denen sich der Proband im Heel Strike befunden hat. In den anderen Zeitschritten sollen dann mittels Key-Frame-Technik die Haltungsparameter interpoliert werden.

Das Rohsignal Zeilenvarianz ist dieses Signal noch recht unzuverlässig und besitzt viele lokale Nebenmaximas, deren genauer Zeitpunkt anfällig für Fehler ist. Deshalb erfolgt eine Bandpassfilterung des Signales im Frequenzraum. Es ist wichtig, keine Filter zu verwenden, die eine Phasenverschiebung des Signales bewirken. Am einfachsten ist es eine Bibliothek einer FFT zu benutzen, das Signal in den Frequenzraum zu transformieren und alle Fourierkoeffizienten zu null zu setzen, die im Sperrbereich des Bandpassfilters liegen. Der Sperrbereich des Bandpassfilters lässt sich berechnen, indem man die höchst mögliche Schrittfrequenz und die langsamste zu erwartende Schrittfrequenz bestimmt.

Nach der Rücktransformation der Fourierkoeffizienten in den Ortsraum steht ein sauberes Signal zur Verfügung. Dieses erlaubt eine Synchronisation des Modells mit dem Probanden. Die Maximas korrespondieren mit den Frames, in denen der Proband die Beine geöffnet hatte. Minimas sind dort, wo der Proband die Beine geschlossen hatte. Nun kann nach den Minimas und Maximas des Signales gesucht werden, für diese Stellen lässt sich die Schrittzahl festlegen. Dies sind die Key-Frames der zu interpolierenden Bewegung.

Es gibt zwei verschiedene Maximas, die sich jedoch sehr ähnlich sehen, weil der gesunde menschliche Lauf normalerweise symmetrisch ist. In jedem zweiten Maxima ist das linke Bein vorn und in dem dazwischen liegendem Maxima immer das rechte. Dazwischen kann die Schrittzahl interpoliert werden. Im Ergebnis bewegt sich das Modell mit dem typischen Bewegungsmuster näherungsweise synchron zur Bildsequenz.

### Gaborfilterung

Eine ähnliche Methode zur Synchronisation der Bewegung des Modells mit der des Probanden ist unter Verwendung eines Gaborfilters möglich. Im Gegensatz zur eben vorgestellten Methode wird kein separater Interpolationsschritt benötigt. Es wird ein komplexer Phasenwinkel aus dem Signal  $v(t)$  berechnet, die Methode dafür stammt aus [HS96]. Dazu wird der Filterkern  $G(t; \sigma, \omega)$  eines komplexen Gaborfilters verwendet:

$$G(t; \sigma, \omega) = \frac{1}{\sqrt{2\pi\sigma}} \cdot e^{-0.5\left(\frac{t}{\sigma}\right)^2} \cdot e^{-j\omega t} \quad (9.5)$$

Da es lediglich auf den Phasenwinkel, nicht aber auf den Betrag des Filterausganges ankommt, kann der Term  $\frac{1}{\sqrt{2\pi\sigma}}$  weggelassen werden. Dabei ist  $t$  die Zeit eines Abtastpunktes



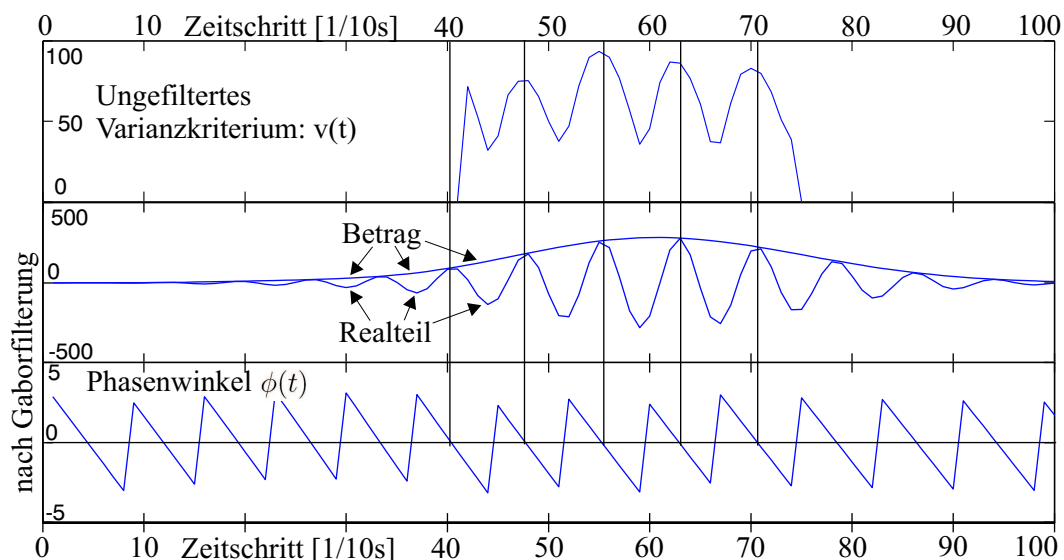


Abbildung 9.8.: Gaborfilterung der Zeilenvarianz zur Bestimmung des Phasenwinkels

der Filtermaske. Die Mittenfrequenz (Kreisfrequenz)  $\omega$  des Gaborfilters berechnet sich aus der minimalen  $f_{min}$  und maximalen  $f_{max}$  Frequenz, die möglichst genau vorzugeben ist:

$$\omega = 2\pi \cdot \left( \frac{f_{min} + f_{max}}{2} \right) \quad (9.6)$$

Das zu filternde Varianzkriterium besitzt die *doppelte* Schrittfrequenz, dies ist bei der Angabe von  $f_{min}$  und  $f_{max}$  zu beachten!

Der Parameter  $\sigma$  beeinflusst die Bandbreite des Gaborfilters. Für eigene Experimente wurde der dieser Wert aus der minimalen und maximalen Frequenz berechnet:

$$\sigma = \frac{1}{2 \cdot (f_{max} - f_{min})} \quad (9.7)$$

Die Filterung des Signales  $v(t)$  selbst erfolgt klassisch durch eine Faltung. Für einen Zeitpunkt  $t$  berechnet sich der komplexe Filterausgang.

$$f(t) = \sum_{t_g=-T}^T v(t+t_g) \cdot G(t_g; \sigma, \omega) \quad (9.8)$$

Die Größe von  $T$  sollte so gewählt werden, dass keine signifikanten Komponenten der Filtermaske verloren gehen ( $T \geq 3\sigma$ ). Die Faltung kann auch sehr effizient im Frequenzraum erfolgen, dann ist die Größe der Filtermaske ohnehin größer als die Sequenzlänge und deshalb nicht kritisch.

## 9. Gewinnung von Näherungswerten

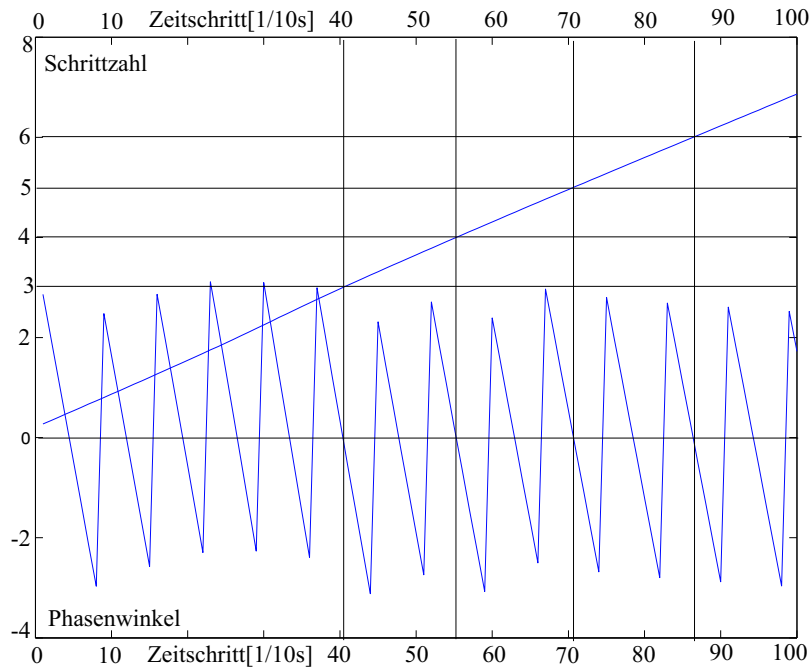


Abbildung 9.9.: Phasenwinkel und die daraus berechnete Schrittzahl

Der Gaborfilter  $f(t)$  berechnet für jeden Zeitschritt  $t$  einen Zeiger in der komplexen Zahlenebene, der mit nahezu doppelter Schrittfrequenz rotiert. Es lässt sich ein Phasenwinkel  $\phi_t$  für den komplexen Zeiger berechnen<sup>1</sup>, der aber ebenfalls bezogen auf die doppelte Schrittfrequenz angegeben ist.

$$\phi_t = \phi(t) = \text{arg}f(t) \quad (9.9)$$

Die Winkeldifferenz zweier aufeinander folgender Zeitschritte  $\Delta\phi_t = \phi_{t-1} - \phi_t$  ist nahezu proportional zur Schrittfrequenz des Probanden<sup>2</sup>.

Es lässt sich nun die reelle Schrittzahl  $\theta_t$  für jeden Zeitschritt berechnen.

$$\theta_t = \theta_{t-1} + \frac{1}{2} \cdot \frac{\Delta\phi_t}{2\pi} \quad (9.10)$$

Dabei sorgt  $\frac{1}{2}$  für eine zwei zu eins Untersetzung des Timings des Gaborfilters. Diese Zahl erhöht sich mit jedem Zeitschritt in der Weise, dass sie je Schritt des Probanden um eins größer wird.

Steht die Schrittzahl  $\theta_t$  für jeden Zeitschritt zur Verfügung lassen sich die Gelenkwinkel des 3D-Modells initialisieren.

<sup>1</sup> In Matlab oder C verwendet man dazu die Funktion  $\phi = \text{atan2}(im, re)$ .

<sup>2</sup> Bei der Berechnung ist auf den Grenzübergang  $-\pi$  nach  $+\pi$  zu achten, an dieser Stelle wird  $\Delta\phi_t < 0$  in diesem Fall ist  $\Delta\phi_t$  durch Addition von  $2\pi$  zu korrigieren.

# 10. Experimente, aufgetretene Probleme und deren Lösungen

In diesem Kapitel werden die Experimente beschrieben, die mit den vorgestellten Modellen durchgeführt wurden. Es werden die dabei festgestellten Probleme dokumentiert und gefundene Lösungen vorgestellt.

## 10.1. Experimente mit dem einfachen Skelettmodell

Im Folgenden sollen einige Beispielberechnungen mit realen Kamerabildern durchgeführt werden. Um die Gleichungen nicht zu unübersichtlich werden zu lassen, wird hier zunächst auf die Modellierung kinematischer Ketten nach Abschnitt 3.4.1 verzichtet. Eine deutliche Vereinfachung der Gleichungen des Modells wird erreicht, wenn jedes Gelenk als 3D-Raumpunkt aufgefasst wird, das über eine bekannte Strecke mit anderen Gelenken mehr oder weniger starr verbunden ist. Sehr ausführlich wurde dies in Abschnitt 3.2 beschrieben.

Die Lage eines Raumpunktes  $\mathbf{w}_i = [x_i, y_i, z_i]^T$ , der in 2 oder mehr Kameras  $k$  beobachtet wurde, berechnet sich durch Aufstellen der Beobachtungsgleichungen in ein überbestimmtes Gleichungssystem, das dann mit den Standardmethoden der Fehler- und Ausgleichsrechnung gelöst wird. Mit Hilfe des Kameramodells aus Abschnitt 5.2.1 berechnet sich die Projektion  $\mathbf{p}^k(\mathbf{w}_i)$  des Punktes  $i$ . Für jede Kamera  $k$ , in der ein zu erreichender Zielpunkt  $\mathbf{z}_i^k$  gegeben ist, lassen sich die beiden Beobachtungsgleichungen für die x- und y-Bildkoordinate aufstellen:

$$\begin{aligned}
 \begin{bmatrix} \frac{\partial p_x^k(\mathbf{w}_i)}{\partial x_i} & \frac{\partial p_x^k(\mathbf{w}_i)}{\partial y_i} & \frac{\partial p_x^k(\mathbf{w}_i)}{\partial z_i} \\ \frac{\partial p_y^k(\mathbf{w}_i)}{\partial x_i} & \frac{\partial p_y^k(\mathbf{w}_i)}{\partial y_i} & \frac{\partial p_y^k(\mathbf{w}_i)}{\partial z_i} \end{bmatrix} \cdot \begin{bmatrix} \Delta x_i \\ \Delta y_i \\ \Delta z_i \end{bmatrix} &= \begin{bmatrix} z_{ix}^k - p_x^k(\mathbf{w}_i) \\ z_{iy}^k - p_y^k(\mathbf{w}_i) \end{bmatrix} = \\
 \begin{bmatrix} \frac{\partial \mathbf{p}^k(\mathbf{w}_i)}{\partial x_i} & \frac{\partial \mathbf{p}^k(\mathbf{w}_i)}{\partial y_i} & \frac{\partial \mathbf{p}^k(\mathbf{w}_i)}{\partial z_i} \end{bmatrix} \cdot \begin{bmatrix} \Delta x_i \\ \Delta y_i \\ \Delta z_i \end{bmatrix} &= \mathbf{z}_i^k - \mathbf{p}^k(\mathbf{w}_i) = \\
 \mathbf{A}_i^k \cdot \Delta \mathbf{w}_i &= \Delta \mathbf{y}_i^k
 \end{aligned} \tag{10.1}$$

Da Gl.(10.1) eine 2D-Vektorgleichung ist, entsteht bereits ab zwei Kameras ein überbestimmtes Gleichungssystem. Die Berechnung von  $\mathbf{w}_i$  erfolgt wie üblich durch Ausgleichsrechnung.

## 10. Experimente, aufgetretene Probleme und deren Lösungen

Angenommen, zwei Gelenkpunkte  $i$  und  $j$  sind durch eine Strecke mit bekannter Länge  $s_0$  miteinander verbunden. Unter der Nebenbedingung Punktabstand =  $s_0$  sollen in einem Gleichungssystem nun zwei Punktkoordinaten ( $\mathbf{w}_i$  und  $\mathbf{w}_j$ ) berechnet werden. Aus [Kru83] wurde die folgende Gleichung für eine Streckenbeobachtung entnommen:

$$a \cdot \Delta x_j + b \cdot \Delta y_j + c \cdot \Delta z_j - a \cdot \Delta x_i - b \cdot \Delta y_i - c \cdot \Delta z_i = s_0 - s \quad (10.2)$$

$$\begin{bmatrix} -a & -b & -c & a & b & c \end{bmatrix} \cdot \begin{bmatrix} \Delta \mathbf{w}_i \\ \Delta \mathbf{w}_j \end{bmatrix} = s_0 - s \quad (10.3)$$

mit  $a = \frac{x_j - x_i}{s_0}$ ;  $b = \frac{y_j - y_i}{s_0}$ ;  $c = \frac{z_j - z_i}{s_0}$  und  $s = |\mathbf{w}_j - \mathbf{w}_i|$

Zur Aufstellung der Beobachtungsgleichungen für Kantenpunkte wird jede Modellkontur diskretisiert und für jeden Kantenpunkt  $i$  mit den Koordinaten  $\mathbf{p}^k(\mathbf{w}_i)$  ein Normalenvektor  $\mathbf{n}_i = [n_{ix}, n_{iy}]^T$  (mit  $1 = |\mathbf{n}_i|$ ) vorgegeben. Entlang dieser Normalenrichtung wird nach Kantenpunkten gesucht. Aus Normalenvektor und gefundener Distanz  $a_i$  könnte für jeden Kantenpunkt auch der 2D-Vektor  $\mathbf{z}_i^k - \mathbf{p}^k(\mathbf{w}_i) = \mathbf{n}_i \cdot a_i$  berechnet und zwei Beobachtungsgleichungen aufgestellt werden. Gemessen wurde aber nur eine skalare Größe, die Distanz  $a_i$  zwischen Modell und Bild. Stellt man jedoch zwei Gleichungen je Kantenpunkt auf, so lässt sich das Modell nur noch schwer längs der Kante bewegen. Die beiden ersten Gleichungen 10.1 werden deshalb durch Multiplikation mit dem Normalenvektor zu einer skalaren Beobachtungsgleichung transformiert:

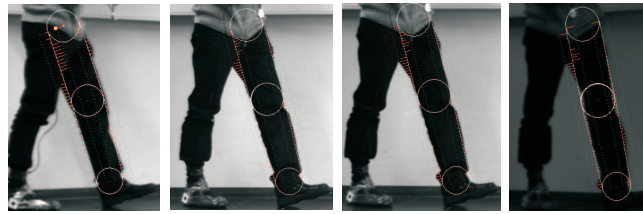
$$\mathbf{n}_i^T \cdot \mathbf{A}_{ci} \cdot \Delta \mathbf{w}_i = \mathbf{n}_i^T \cdot \Delta \mathbf{y}_i^k = a_i \quad (10.4)$$

Als Experiment wurde ein Bein aus drei Gelenkpunkten und drei Radien modelliert. Das Modell wurde von Hand näherungsweise positioniert und die neun unbekannt Parameter wurden geschätzt. Abbildung 10.1 zeigt die Bildausschnitte und einen Auszug aus dem Messprotokoll. Man sieht, das Bein ist gestreckt. In dieser Konfiguration liegen nicht genügend Informationen vor, das Konusmodell entlang der Konusachsen zu stabilisieren. Vor allem dann nicht, wenn die Konusradien ebenfalls unbekannt sind. Erst nachdem näherungsweise Koordinaten des Hüftgelenkpunktes in einer Kamera manuell vorgegeben wurden, steht ausreichend Information zur Verfügung, um Konvergenz zu erreichen. Die genaue Position des Konusmodells entlang der Konusachsen basiert dann aber einzig auf der Vorgabe des Anwenders und ist demzufolge subjektiv.

Werden zwei Beine modelliert, gelangt man zu ähnlichen Resultaten. Abbildung 10.2 zeigt die Ergebnisse. Die beiden Punkte der Hüftgelenke wurden durch eine Streckenbeobachtung miteinander verbunden. Normalerweise ist die Höhendifferenz beider Hüftgelenkpunkte minimal. Durch Gleichsetzen der z-Koordinaten der beiden Hüftgelenkpunkte lässt sich eine weitere Beobachtungsgleichung formulieren, die beide Punkte noch besser in der selben Höhe hält und somit das Modell weiter stabilisiert.

Bei der Approximation des zweibeinigen Modells treten verschiedene Probleme auf. In Abbildung 10.3 sind einige dieser Probleme dargestellt. Zunächst wäre die Vergrößerung des Beinradius in der Nähe des Hüftgelenkes zu nennen. Ursache dafür dürften die Beobachtungsgleichungen sein, die im Selbstverdeckungsbereich aufgestellt wurden. Deutlich sind die Ausreißer in diesen Bereichen zu erkennen.

## 10.1. Experimente mit dem einfachen Skelettmodell



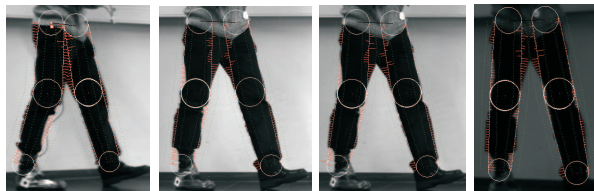
```

Vertrauensbereich = 3.00*sigma
Standardabweichung a priori           = 1.00
Standardabweichung a posteriori        = 1.68
Standardabweichung Streckenbeobachtungen = 1.00 [mm]
Standardabweichung Normalenbeobachtungen = 3.00 [mm]
Standardabweichung manuelle Punktbeobachtungen = 1.00 [Pixel]

Werte  Vertrauensbereiche
-490.40 ±7.51 x-[mm] linkes Sprunggelenk
383.57 ±17.41 y-[mm] linkes Sprunggelenk
126.32 ±17.85 z-[mm] linkes Sprunggelenk
-606.41 ±6.48 x-[mm] linkes Knie
412.38 ±13.27 y-[mm] linkes Knie
549.78 ±17.11 z-[mm] linkes Knie
-722.33 ±7.30 x-[mm] linkes Hüftgelenk
464.53 ±16.09 y-[mm] linkes Hüftgelenk
930.58 ±16.13 z-[mm] linkes Hüftgelenk
69.79 ±4.85 r-[mm] Radius Bein, höhe Fussknöchel
83.82 ±3.82 r-[mm] Radius Bein, höhe Knie
84.43 ±5.29 r-[mm] Radius Bein, höhe Hüftgelenk

Streckenlängen nach Ausgleichung:
soll=440.00  ist=440.00  dif=-0.000015  [mm][mm] Länge Unterschenkel
soll=410.00  ist=410.00  dif=-0.004516  [mm][mm] Länge Oberschenkel
    
```

Abbildung 10.1.: Approximation eines Beines an den Bildinhalt von 4 Kameras. Oben Bildausschnitte der Kameras, unten Auszug aus dem Messprotokoll.



```

Vertrauensbereich = 3.00*sigma
Standardabweichung a priori           = 1.00
Standardabweichung a posteriori        = 1.94
Standardabweichung Streckenbeobachtungen = 1.00 [mm]
Standardabweichung Normalenbeobachtungen = 3.00 [mm]
Standardabweichung manuelle Punktbeobachtungen = 1.00 [Pixel]
sigma UpwardH = 1.00
sigma UpwardU = 1.00

Werte  Vertrauensbereiche
-1008.38 ±8.43 x-[mm] linkes Sprunggelenk
433.27 ±18.94 y-[mm] linkes Sprunggelenk
132.58 ±18.61 z-[mm] linkes Sprunggelenk
-901.14 ±6.36 x-[mm] linkes Knie
336.56 ±14.57 y-[mm] linkes Knie
548.24 ±17.40 z-[mm] linkes Knie
-878.12 ±6.45 x-[mm] linkes Hüftgelenk
343.52 ±16.03 y-[mm] linkes Hüftgelenk
955.84 ±16.71 z-[mm] linkes Hüftgelenk
-488.32 ±8.77 x-[mm] rechtes Sprunggelenk
382.35 ±20.22 y-[mm] rechtes Sprunggelenk
134.24 ±18.76 z-[mm] rechtes Sprunggelenk
-612.93 ±7.00 x-[mm] rechtes Knie
420.37 ±15.54 y-[mm] rechtes Knie
554.52 ±18.08 z-[mm] rechtes Knie
-692.76 ±7.67 x-[mm] rechtes Hüftgelenk
420.60 ±19.27 y-[mm] rechtes Hüftgelenk
956.33 ±17.50 z-[mm] rechtes Hüftgelenk
63.06 ±3.96 r-[mm] Radius Bein, Höhe Fussknöchel
83.87 ±3.80 r-[mm] Radius Bein, Höhe Knie
71.74 ±4.09 r-[mm] Radius Bein, Höhe Hüftgelenk

Streckenlängen nach Ausgleichung:
soll=440.00  ist=440.04  dif=-0.035871  [mm] Länge linker Unterschenkel
soll=410.00  ist=408.31  dif=1.693325  [mm] Länge linker Oberschenkel
soll=440.00  ist=440.02  dif=-0.016731  [mm] Länge rechter Unterschenkel
soll=410.00  ist=409.66  dif=0.340053  [mm] Länge rechter Oberschenkel
soll=200.00  ist=200.75  dif=-0.746493  [mm] Abstand linkes-rechtes Hüftg
    
```

Abbildung 10.2.: Approximation beider Beine an den Bildinhalt von 4 Kameras. Oben Bildausschnitte der Kameras, unten Auszug aus dem Messprotokoll.

## 10. Experimente, aufgetretene Probleme und deren Lösungen

Das Bild in der Mitte der unteren Reihe zeigt das Durchdringen des Modells durch den Boden, wenn das Gewicht der manuellen Punktbeobachtung des Hüftgelenkpunktes so klein gemacht wird, wie es für eine manuelle Punktbeobachtung korrekt wäre (10 Pixel Standardabweichung). Der Einfluss dieser manuellen Beobachtung auf das Ausgleichsergebnis wird dann zu klein, um das Modell in der richtigen Höhe zu halten.

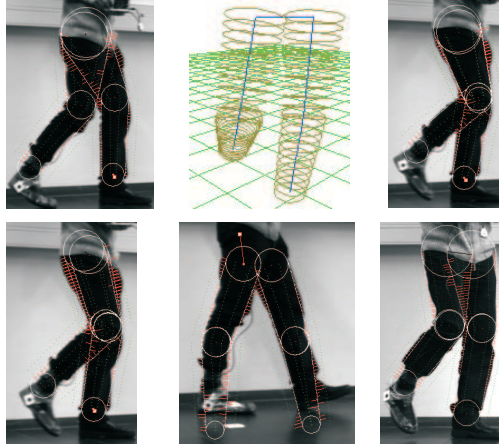


Abbildung 10.3.: Zwei Beinmodell und Probleme bei der Approximation an den Bildinhalt

### 10.2. Effekte bei der Approximation

Für alle folgenden Untersuchungen wird nicht mehr das vereinfachte Skelettmodell, sondern das hierarchische Skelettmodell mit kinematischen Ketten nach Abschnitt 3.3 verwendet. Zunächst wurden die Näherungswerte der Gelenkpunkte manuell vorgegeben. Mit Hilfe des Verfahrens nach Abschnitt 6.3 wurden die Beine des Modells in einem Zeitschritt an vier Kamerabilder approximiert. Auf der Basis dieser guten Näherungswerte lassen sich typische Effekte bei der Approximation von Konusmodellen mit dem Suchstrahlverfahren nach [BI98] zeigen.

Im ersten Versuch wurden nicht die manuellen Beobachtungsgleichungen für jeden Gelenkpunkt aufgestellt, sodass eine Approximation des Modells allein basierend auf der Kanteninformation stattgefunden hat. Bereits nach einer Iteration kommt es zu einem 'Abheben' des Modells, wie in Abbildung 10.4 zu sehen ist. Dies passiert immer in den Zeitschritten, in denen der Proband seine Beine gestreckt hat (Double-Support). Hauptursache dürfte ein singuläres Gleichungssystem sein, da lediglich Kantenpunkte in einer Richtung vorhanden sind. Im Grunde stellt sich hier das klassische Aperturproblem. Eine Fixierung des Modells in vertikaler Richtung ist in diesen Fällen nicht möglich. Abhilfe können weitere Beobachtungsgleichungen schaffen, die zum Beispiel Gradienteninformation in vertikaler Richtung auswerten. In Frage kommen Konturpunkte des Kopfes, der

Schultern oder auch der Fußsohlen.

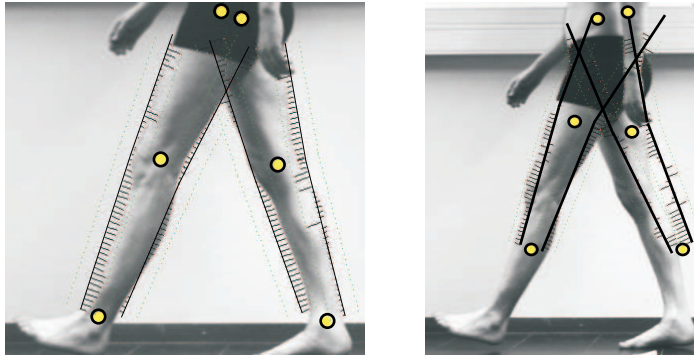


Abbildung 10.4.: Fehlerhafte Approximation: Modell vor einer Iteration (links) und nach einer Iteration (rechts)

Eine andere Möglichkeit wäre die Berücksichtigung weiterer Annahmen:

- Die Koordinaten des Sprunggelenkes ändert sich in der Standphase nicht.
- Die z-Koordinate des Probanden ist nahezu konstant.

Wird ein Bewegungsmodell eines laufenden Probanden eingesetzt, lässt sich vorhersagen, wann welches Bein auf dem Boden steht. Gegen die erste Annahme spricht aber, dass die Probleme hauptsächlich in der Doppel-Stand-Phase auftreten, vorwiegend in den Zeitschritten, in denen die Gewichtsverlagerung von einem auf das andere Bein erfolgt. Die Abrollbewegung des Probanden führt dann doch zu einer geringfügigen Bewegung der beiden Sprunggelenke, da der eine Fuß noch nicht wirklich in Ruhe gekommen ist, aber der andere Fuß bereits auf dem Weg zum nächsten Standpunkt ist.

Die zweite Annahme ist nur dann nützlich, wenn man auf die Erfassung der Höhe des Probanden gänzlich verzichten will. Ausgegeben würde dann lediglich eine mittlere Höhe des Probanden.

Um die Approximation zunächst zu stabilisieren, wurden die Beobachtungsgleichungen der manuell markierten Gelenkpunkte wieder in die Ausgleichung einbezogen. Dadurch lassen sich typische Effekte beobachten.

In Abbildung 10.5 sind vier Effekte gezeigt, die bei der Approximation von Konusmodellen an die Beine des Probanden auftreten. Die Gelenkpunkte des Modells sind als Kreise dargestellt. Schwarze Linien zeigen die Abweichungen zwischen Modell und gefundenem Kantenpunkt.

Die Detailprobleme in Abbildung 10.5 werden im Folgenden erläutert:

**a:** Probleme bereiten immer die Verdeckungsbereiche zwischen linkem und rechtem Bein. Hinzu kommt ein Mangel an Kontrast, sodass selbst die Kantenpunkte des verdeckenden Beines nicht sicher gemessen werden können. Bei guten Näherungswerten sollte ein

## 10. Experimente, aufgetretene Probleme und deren Lösungen

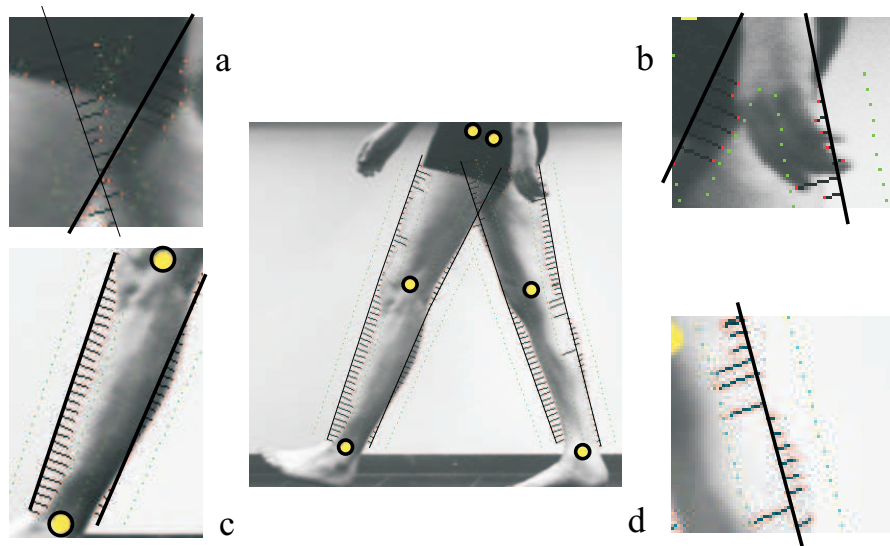


Abbildung 10.5.: Probleme bei der Approximation des Konusmodells an den Bildinhalt nach dem Verfahren von [BI98], verschiedene Ursachen von Fehlmessungen a: durch Mangel an Kontrast und Selbstverdeckungen, b: Durchschwingen der Hand, c: Asymmetrie der Form des Unterschenkels, d: Ausreißer durch Mangel an Kontrast zwischen Bein und Hintergrund

Verfahren zu entwickeln sein, das die problematischen Kantenpunkte vorhersagt und ausschließt. So würden lediglich die Silhouettenpunkte berücksichtigt werden.

**b:** Zu sehen sind Fehlmessungen im Bereich der Hand. Genauso wie im Fall **a** treten Verdeckungen auf, diesmal allerdings zwischen Hand und Bein. Da das Modell bisher nicht auf den Oberkörper erweitert ist, ließen sich diese Verdeckungen derzeit nicht einmal vorhersagen. Zur Lösung könnte entweder der Oberkörper modelliert werden, um auch die Bewegung der Hand zu erfassen, oder man verzichtet generell auf die Messung von Kantenpunkten im oberen Bereich des Oberschenkels.

**c:** Obwohl die beiden Gelenkpunkte von Knie- und Sprunggelenk nahezu optimal positioniert sind, sind deutlich größere Abweichungen an der linken Kante (Schienbein) zu erkennen als an der rechten Kante (Wade). Es ist zu vermuten, dass dadurch das Sprunggelenk des Modells nach rechts (vom Proband aus gesehen nach hinten) verschoben wird. In Abbildung 10.6 ist dann der nächste Iterationsschritt zu sehen und die Wirkung der zu ungenauen Formbeschreibung. Wie vermutet, äußert sie sich in einer Verschiebung der Sprunggelenke. Weiterhin deuten sich in Abbildung 10.6 neue Probleme an. Durch die Verschiebung der Sprunggelenke nach hinten kommt es nun zu einer fehlerhaften Korrespondenz, indem die Schienbein-Kantenpunkte des Modells an die Kantenpunkte der Wade approximiert werden. Das wiederum führt nun zu widersprüchlichen Beobachtungen für den Durchmesser des Kegelstumpfes am Sprunggelenk und es kommt einige



Iterationen später zu einem viel zu kleinen Radius nahe 0. In Abbildung 10.7 ist der Effekt nach einigen Iterationen deutlich zu sehen.

**d:** Fehlmessungen durch Mangel an Kontrast der Körperkante, bzw. durch stärkere Gradienten auf dem Bein. Die einfache Maximumsuche wird dadurch bereits leicht irritiert. Der Effekt ließe sich womöglich reduzieren, indem der Suchbereich verkleinert wird. Dadurch verkleinert sich aber auch der Fangbereich des Modells, also werden noch genauere Näherungswerte erforderlich.

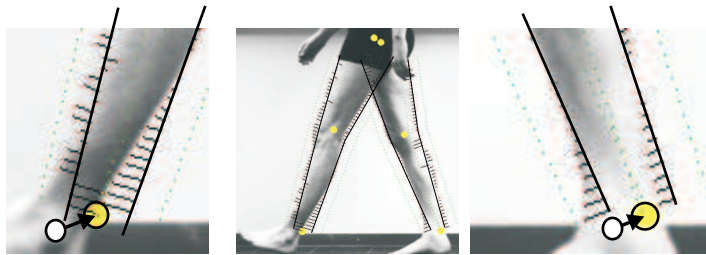


Abbildung 10.6.: Durch das verwendete Konusmodell kommt es zu systematischen Fehlern. Dies äußert sich in einer Verschiebung der Sprunggelenke nach rechts (vom Probanden aus gesehen nach hinten). Dies hat später eine fehlerhafte Korrespondenz zur Folge. Ursache ist die asymmetrische Form des Unterschenkels.

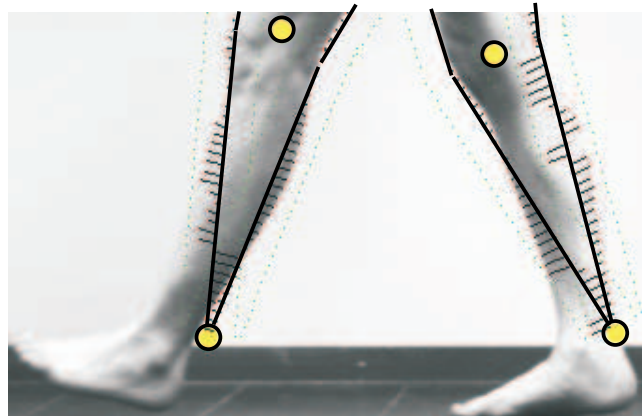


Abbildung 10.7.: Nach einigen Iterationen liegen die beiden Sprunggelenke des Modells vollständig außerhalb des Körpers. Der Konusdurchmesser in der Höhe des Sprunggelenkes wird dann irrtümlich sehr klein geschätzt. Ursache dafür ist eine unzureichende Formbeschreibung des Unterschenkels durch einen symmetrischen Konus (siehe Text Punkt c).

### Schlussfolgerung

Neben dem Mangel an vertikaler Fixierung des Modells kommt es im Wesentlichen zu vier Effekten, die analysiert und für die Lösungsvorschläge unterbreitet wurden. Die einfache Kombination von Konusmodell und Kantenpunktsuche ist nicht zufrieden stellend. Die Formbeschreibung durch ein zu einfaches Konusmodell ist unzureichend und führt zu erheblichen systematischen Fehlern die wiederum nach einigen Iterationen zu einer fehlerhaften Korrespondenz zwischen Modell und Bild führen. Die einzig mögliche Schlussfolgerung: Verbesserung des Modells und eine Suche nach Algorithmen, die eine bessere Korrespondenz erlauben. Es ist noch zu erwähnen, dass sich die selben Effekte auch zeigen, wenn statt dem einfachen Skelettmodell das hierarchische Modell mit kinematischen Ketten verwendet wird.

### Lösungen

Im vorhergehenden Abschnitt wurde beschrieben, welche Probleme auftreten, wenn lediglich zwei einfache Kegelstümpfe für die Beine verwendet werden. In den Double-Support-Phasen ist es dann nicht mehr möglich, die vertikale Position des Modells eindeutig zu bestimmen.

Deshalb wurde das vorgestellte Modell um zwei Koordinatensysteme erweitert und damit eine grobe Formbeschreibung der Füße durch ein einfaches 2D-Konturtemplate ermöglicht. Die Beine des Modells erhalten dadurch neue Kanten in horizontaler Richtung und es können weitere Gleichungen aufgestellt werden, die neue Informationen in die Ausgleichung einbringen. Die Höhe des Modells (z-Koordinate) lässt sich somit auch in den Double-Support-Phasen berechnen.

Es wurde darauf hingewiesen, dass die Symmetrie der Unterschenkel des Konusmodells zu einer falschen Berechnung des Kniewinkels führt. Deshalb wurde für die Unterschenkel und jede Kamera ein eigenes Konturtemplate verwendet, dessen Beschreibung im 3D-Raum erfolgte. Da das Konturtemplate die Asymmetrie der Unterschenkel repräsentiert, können nun auch die Kniegelenkwinkel korrekt berechnet werden.

Es wurde ja bereits festgestellt, dass die korrekte Korrespondenz der Kantenpunkte durch Eigentextur des Probanden erschwert wird. Weiterhin lassen sich mit dem Suchstrahlverfahren nach [BI98] Kantenpunkte links und rechts des Beines nicht automatisch differenzieren. Dieses Problem lässt sich durch eine Vorverarbeitung der Bilddaten beheben. Durch Segmentierung wird für jedes Bild zunächst eine binäre Maske erzeugt, die einen Pixel  $\neq 0$  trägt, wenn der Pixel den Probanden zeigt und  $= 0$  wenn der Pixel zum Hintergrund gehört (siehe Abschnitt 7.2). Die Modellapproximation wird dann nicht mehr mit dem Originalbild, sondern mit Kantenpunkten der Maske durchgeführt. Die Suche nach Kantenpunkten erfolgt weiterhin mit der Suchstrahlmethode, allerdings unterscheiden sich Kantenpunkte links und rechts vom Bein nun eindeutig durch ihr

Vorzeichen (Kantenrichtung). Dadurch wurde die Korrespondenzfindung erheblich verbessert.

### 10.3. Manuelle Initialisierung, Einzelschrittauswertung

Unter manueller Initialisierung soll verstanden werden, dass die sechs Gelenkpunkte der Hüfte, der Knien und der Sprunggelenke mit der Maus in jedem Zeitschritt und jeder Kamera mehr oder weniger genau vorgegeben wurden. Der Zeitaufwand für die Auswertung einer einfachen Bildsequenz mit 100 Zeitschritten (Bildaufnahmezeit: 10 Sekunden bei 10 Hz) ist recht erheblich. Die Gelenkpunkte des 3D-Modells lassen sich danach mit den vorgestellten Algorithmen an die vorgegebenen Punkte in den Kamerabildern iterativ heranführen.

Dies kann für jeden Zeitschritt separat geschehen. Es entsteht dann für jeden Zeitschritt ein Zustandsvektor des 3D-Mensch-Modells.

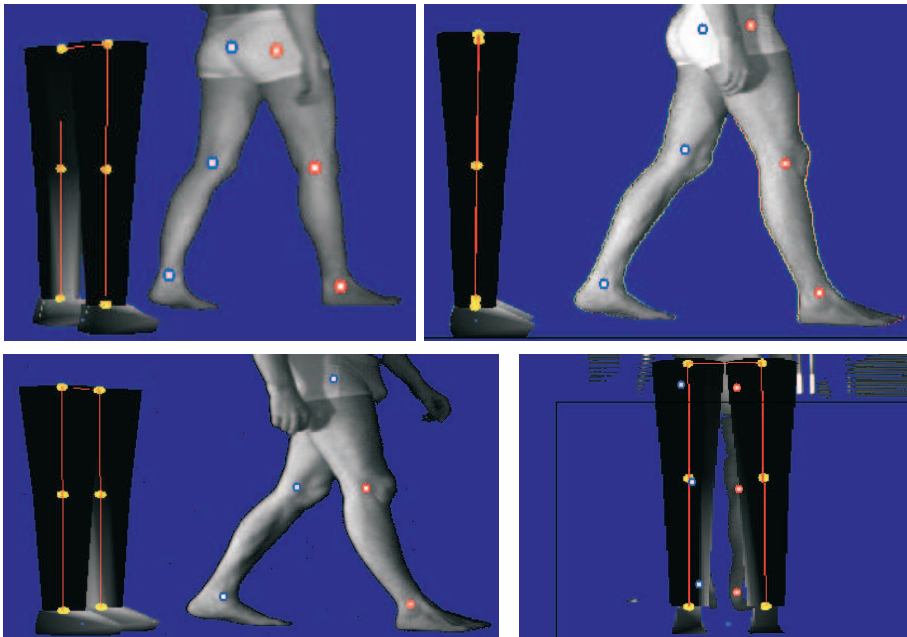


Abbildung 10.8.: Modell vor der manuellen Initialisierung: Durch schlechte Näherungswerte befindet sich das Modell nicht in allen Kamerabildern auf dem Probanden

Das Modell wird vor der Initialisierung in der Mitte des Messfeldes positioniert (siehe Abbildung 10.8). Nach etwa fünf Iterationen wurde das Modell bereits sehr gut an die manuell vorgegebenen Gelenkpunkte approximiert (siehe Abbildung 10.9).

In Abbildung 10.10 ist das Ergebnis der manuellen Initialisierung zu sehen. Im Hintergrund sind die Positionen der sieben Kameras angedeutet. Im Vordergrund ist das

## 10. Experimente, aufgetretene Probleme und deren Lösungen

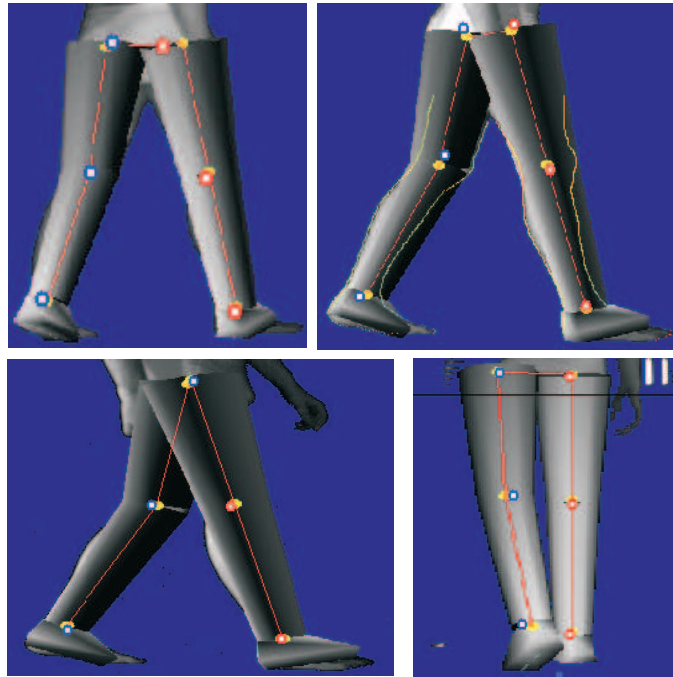


Abbildung 10.9.: Modell nach der manuellen Initialisierung: Das Modell verdeckt in allen Kamerabildern den Probanden.

3D-Modell bestehend aus vier Kegelstümpfen und zwei einfachen Dreiecksnetzen für die Füße zu sehen. Das Skelettmodell wurde durch die manuelle Eingabe von Gelenkpunkten in jedem Zeitschritt näherungsweise positioniert. Die Skelettlinien des Modells sind für alle Zeitschritte im 3D-Raum dargestellt. Die Gelenkwinkel des Sprunggelenkes werden während der Initialisierung noch nicht berechnet.

Einige der Gelenkwinkelverläufe nach der manuellen Initialisierung sind in Abbildung 10.11 zu sehen.

Auf diese Weise ist das Modell dann bereits so gut initialisiert, dass jetzt die Ausgleichsrechnung ohne manuelle Punktbeobachtungen allein basierend auf der verfügbaren Kanteninformation erfolgen kann. Zunächst wird nun ausgehend von den Kantenpunkten des Modells nach Kantenpunkten der Vorder-Hintergrund-Maske gesucht und je Kantenpunkt eine Beobachtungsgleichung für die Parameter des Modells aufgestellt.

Ausgehend von den guten Näherungswerten werden jetzt die Bewegungsparameter verbessert. Mit Hilfe der Konturen stehen jetzt auch Informationen an den Füßen zur Verfügung, und es können nun die Winkel der beiden Sprunggelenke berechnet werden. In diesem Zusammenhang ist zu erwähnen, dass die Iterationen häufig nicht konvergierten, wenn die beiden Winkel für die Sprunggelenke konstant gehalten wurden, sie scheinen für die Modellapproximation also eine erhebliche Rolle zu spielen.

Abbildung 10.12 zeigt das Modell vor und nach der Approximation. Die Minimierung

### 10.3. Manuelle Initialisierung, Einzelschrittauswertung

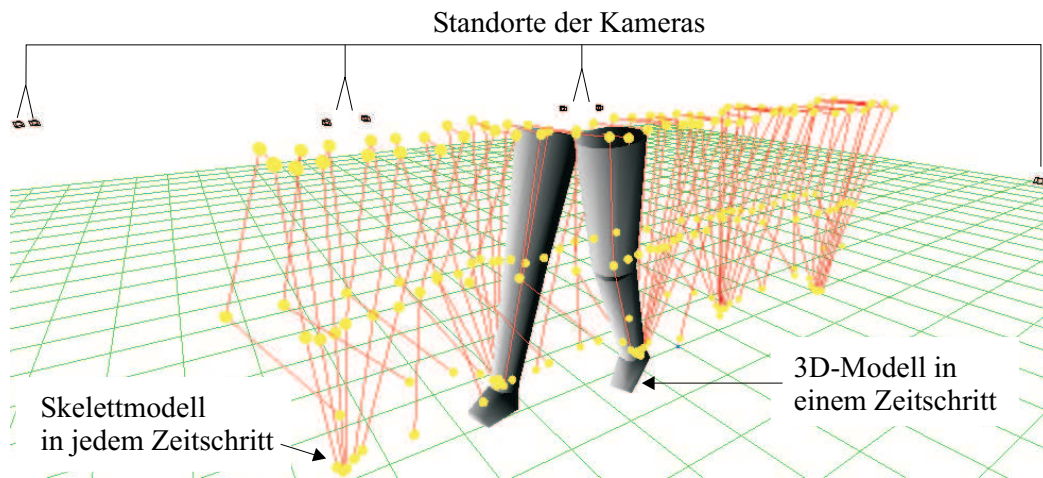


Abbildung 10.10.: Ergebnis der manuellen Initialisierung des 3D-Modells, Darstellung im virtuellen 3D-Raum

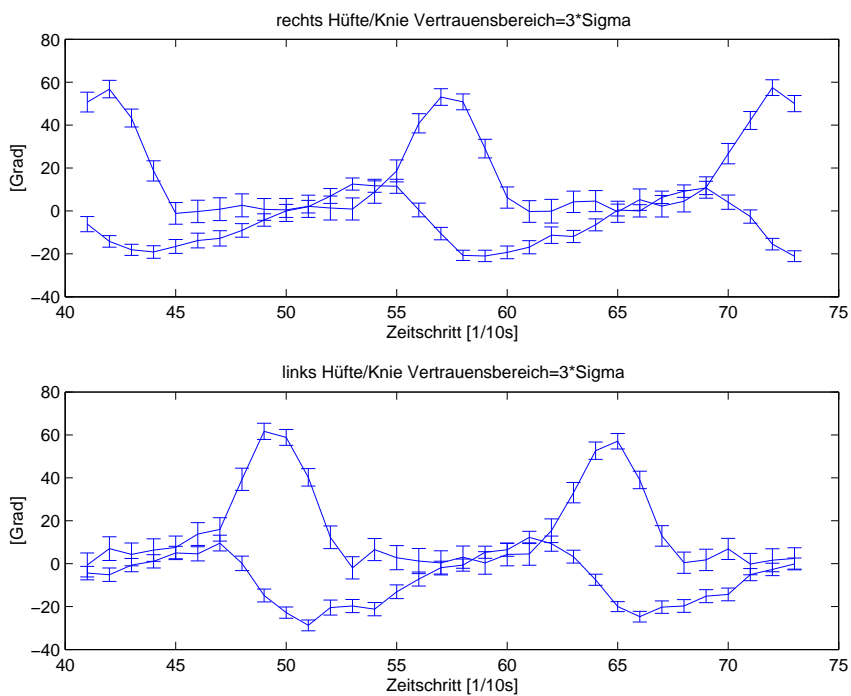


Abbildung 10.11.: Ergebnis der manuellen Modellinitialisierung, Darstellung der Gelenkwinkel im Zustandsraum mit Abschätzung der Genauigkeiten

## 10. Experimente, aufgetretene Probleme und deren Lösungen

der Abweichungen ist vor allem bei den Füßen zu erkennen. Das liegt im Wesentlichen daran, dass die manuellen Näherungswerte für die anderen Gelenke recht gut waren. Viel wichtiger ist aber der Aspekt, dass die Iterationen konvergierten und die daraus resultierende Schlussfolgerung, dass der gewählte modellbasierte Ansatz überhaupt für dieses Problem anwendbar ist.

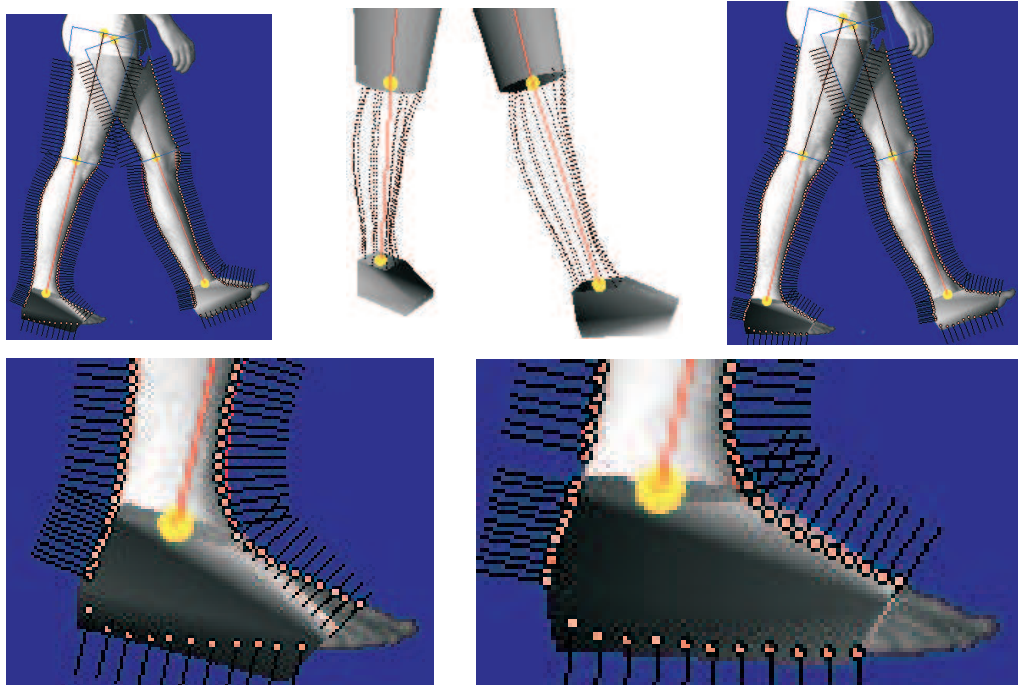


Abbildung 10.12.: Links oben und unten: vor der Approximation des Modells an die Kantenpunkte, Rechts oben und unten: Modell nach der Approximation an die Kantenpunkte, die Unterschiede sind vor allem an den Füßen sichtbar (unten). In der Mitte (oben) sind die verwendeten Punkte aller Konturtemplates einer jeden Kamera im 3D-Raum noch einmal dargestellt.

In Abbildung 10.13 sind die Ergebnisse der Ausgleichsrechnung auf der Basis von Kantenpunkten zu erkennen. Es ist vor allem eine Verkleinerung der geschätzten Unsicherheiten gegenüber der manuellen Initialisierung zu beobachten. Dies liegt wohl an der höheren Redundanz des Gleichungssystems, die durch die große Anzahl der Kantenpunkte entsteht.

### 10.4. Approximation mit quasiperiodischem Bewegungsmodell

Die Genauigkeit und Robustheit der Ausgleichung sollte sich steigern lassen, indem die Parameteranzahl der Unbekannten reduziert wird. Dies soll durch ein quasiperiodisches

#### 10.4. Approximation mit quasiperiodischem Bewegungsmodell

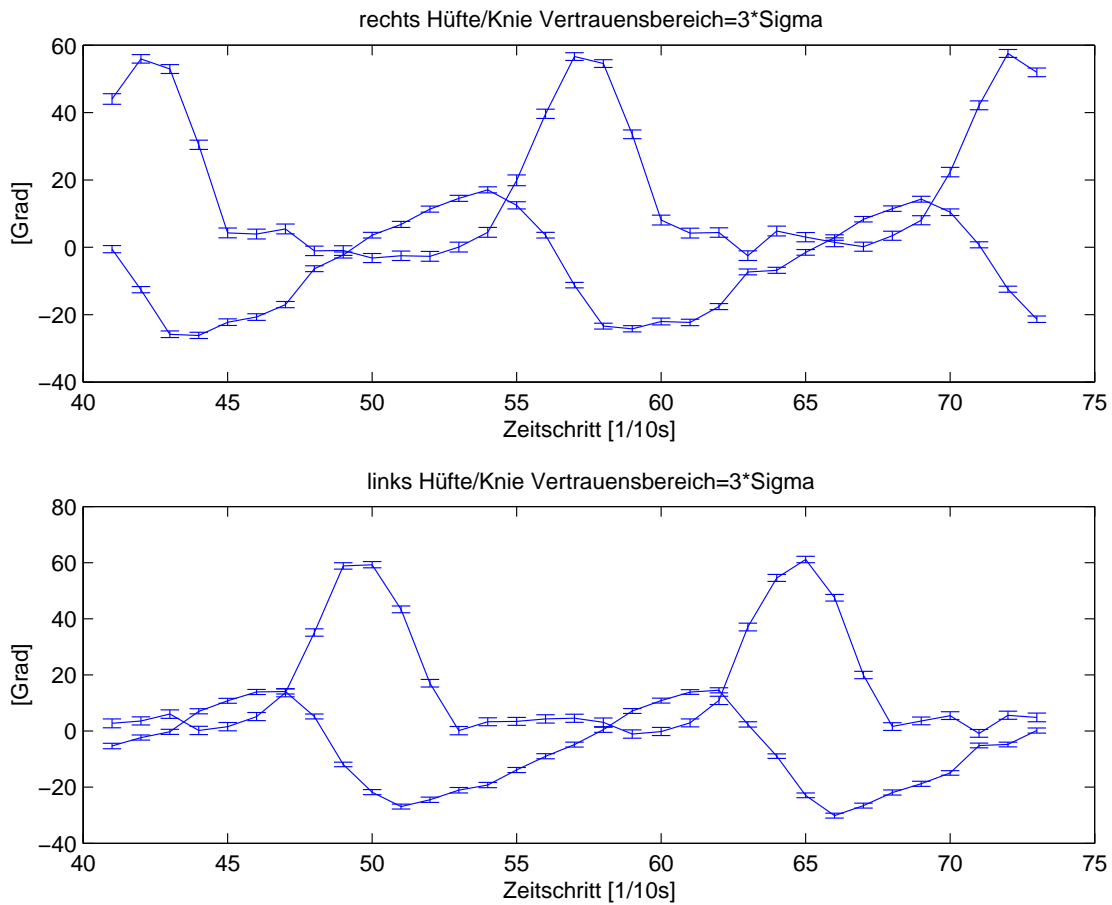


Abbildung 10.13.: Ergebnis der Ausgleichsrechnung auf der Basis von Kantenpunkten, Darstellung der Gelenkwinkel im Zustandsraum mit Abschätzung der Genauigkeiten

## 10. Experimente, aufgetretene Probleme und deren Lösungen

Modell der Bewegung geschehen. Die Ausgleichung erfolgt dann für alle Zeitschritte simultan in einem großen Gleichungssystem mit den selben manuellen Punkten, wie im Abschnitt zuvor. Abbildung 10.14 zeigt das Ergebnis der Berechnung. Verglichen mit

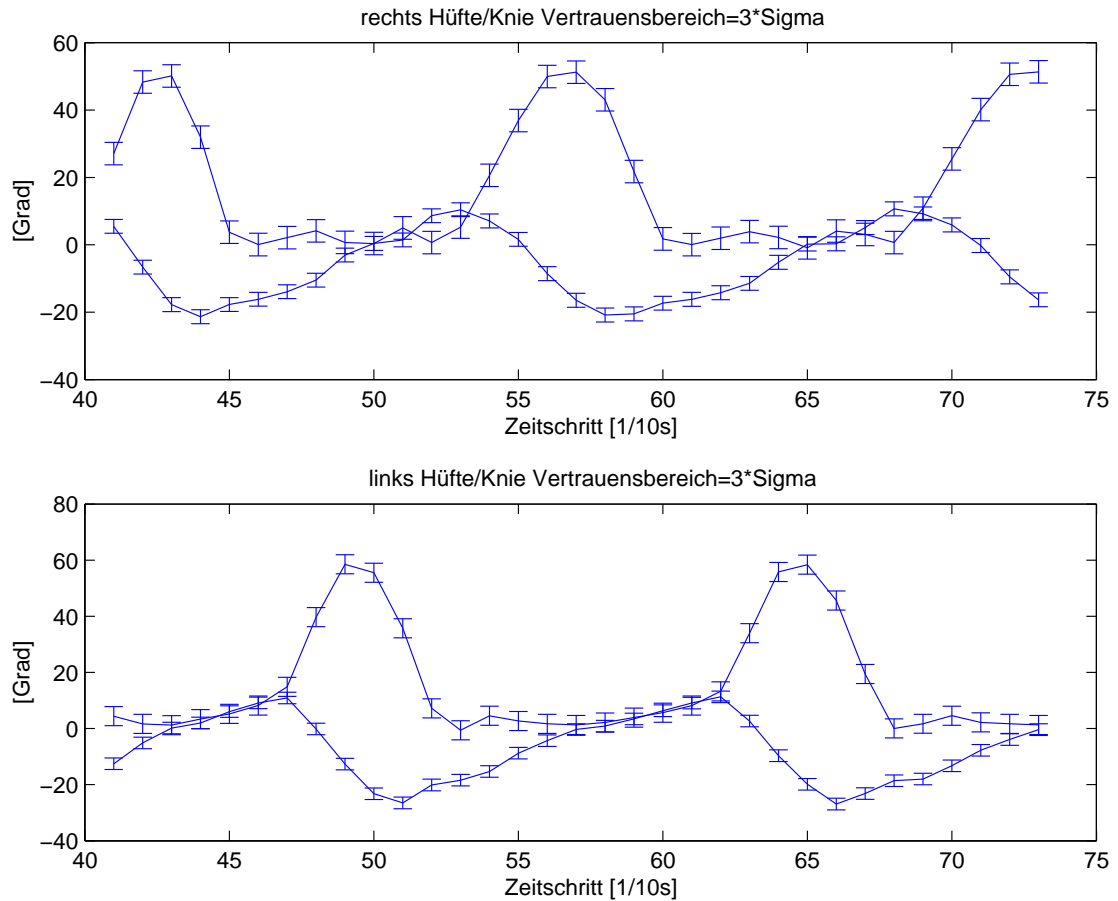


Abbildung 10.14.: Approximation des Modells mit Hilfe des periodischen Bewegungsmodells. Um Störungen durch schlechte Korrespondenz der Kantenpunkte auszuschließen, erfolgte die Approximation zunächst wieder an manuell vorgegebene Punkte. Die zeitliche Synchronisierung erfolgte mit der Gaborfiltermethode nach Kapitel 9.

Abbildung 10.11 ist zwar eine kleine, aber keine deutliche Verringerung der geschätzten Parameterunsicherheit zu erkennen. Das verwundert zunächst, deshalb muss hier eine genauere Analyse erfolgen.

### 10.4.1. Problem der zeitlichen Synchronisierung des Bewegungsmodells

Betrachtet man die Ergebnisse der Approximation in einem Kamerabild der Seitenansicht, so fällt auf, dass Probleme der Modellanpassung vor allem in der Schwungphase



#### 10.4. Approximation mit quasiperiodischem Bewegungsmodell

auftreten. Abbildung 10.15 zeigt drei problematische Zeitschritte.

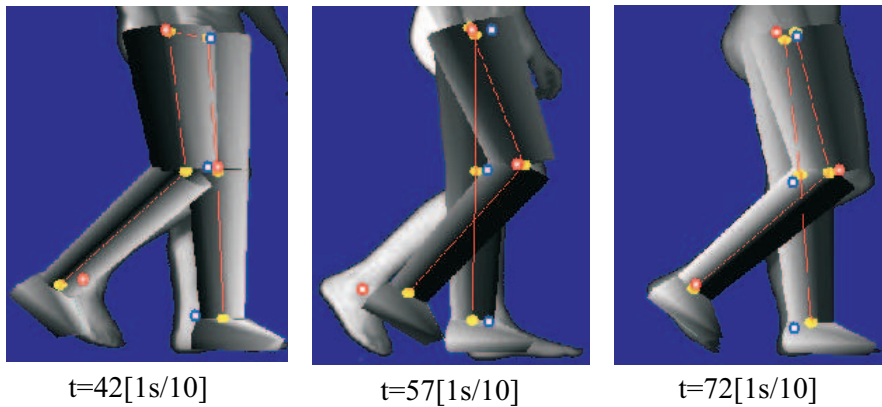


Abbildung 10.15.: Probleme des periodischen Bewegungsmodells in den Schwungphasen

Das Problem lässt sich noch deutlicher im Zustandsraum zeigen. In Abbildung 10.16 sind die Gelenkwinkelverläufe (Knie und Hüfte) des rechten Beines dargestellt. Vergleichend ist je ein Winkelverlauf visualisiert, einmal wurden die Winkel ohne und einmal mit Bewegungsmodell geschätzt. Vor allem die Abweichungen der Winkel in den Zeitschritten 42, 57 und 72 sind interessant.

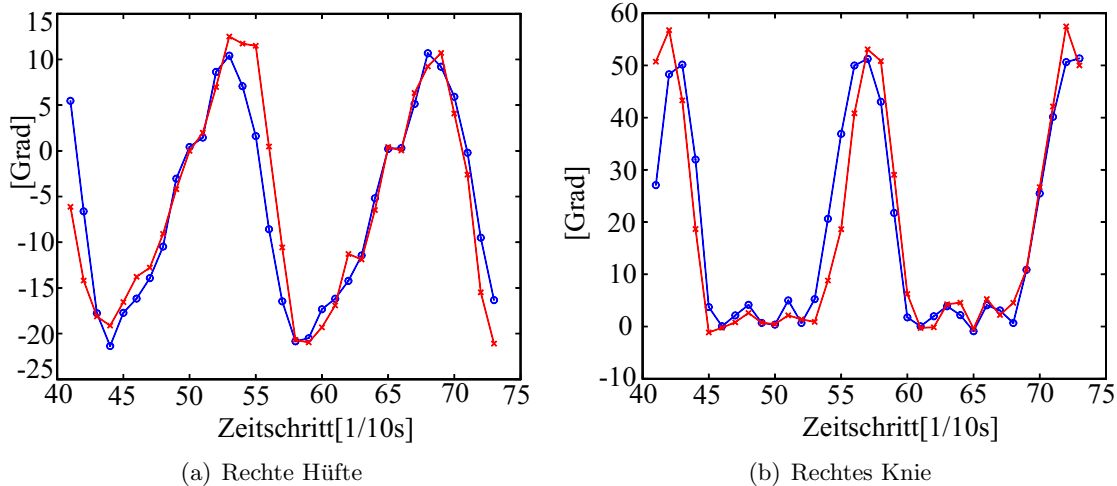


Abbildung 10.16.: Probleme der genauen zeitlichen Synchronisation des Bewegungsmodells mit der Bewegung des Probanden, (Bewegungsmodell - blau und 'o'; Ohne Bewegungsmodell - rot und 'x')

In diesen Zeitschritten treten erhebliche Winkelbeschleunigungen und Winkelgeschwindigkeiten auf. Das führt dann zu einem relativ steilen Anstieg der Verläufe im Winkel-

## 10. Experimente, aufgetretene Probleme und deren Lösungen

Zeit-Diagramm, was ein Anzeichen für hohe Frequenzanteile des Signales ist. Anscheinend führen bereits kleinere Fehler in der zeitlichen Synchronisation des Bewegungsmodells zu erheblichen Winkelfehlern, die wiederum zu großen Fehlern im Ortsraum führen. Dies scheint die Ursache für die Abweichungen in Abbildung 10.15 zu sein. Zur Verdeutlichung soll hier eine Beispielrechnung erfolgen.

Der Hüftwinkel (Amplitude  $30^\circ$ ) ändert sich um  $22^\circ$  in 0.2 s, also mit  $110^\circ/\text{s}$ . Der Kniewinkel (Amplitude  $50^\circ$ ) ändert sich um  $45^\circ$  in 0.2 s, also mit  $225^\circ/\text{s}$ .

Bei einer Bildfrequenz von 10 Hz ergeben sich bereits erhebliche Winkeländerungen von Zeitschritt zu Zeitschritt. Der Hüftwinkel ändert sich etwa um  $11^\circ$  je Zeitschritt und der Kniewinkel um  $22.5^\circ$  je Zeitschritt. Kleinere zeitliche Abweichungen in der Synchronisierung des Bewegungsmodells führen dann zu erheblichen Fehlern. Im Beispiel Abbildung 10.16 beträgt der Synchronisationsfehler etwa einen Zeitschritt ( $1/10$  s). Bezogen auf die Amplitude der Gelenke führt dieser minimale zeitliche Unterschied bereits zu 36 % Amplitudenfehler für die Hüfte und zu 45 % Amplitudenfehler für das Knie.

Zum genaueren Verständnis soll noch einmal ein Blick auf die bisher implementierte Synchronisierung des Bewegungsmodells geworfen werden, mit der auch die zuvor gezeigten Daten berechnet wurden.

Die Synchronisierung des Bewegungsmodells mit der Schrittfrequenz des Probanden erfolgte bisher bei allen Beispielen in diesem Kapitel mit Hilfe der Gaborfilterung der Beinvarianz nach Abschnitt 9.4. Theoretisch sollte sich das Bewegungsmodell nach Gl.(8.24) damit zeitlich näherungsweise mit der Probandenbewegung synchronisieren lassen.

Ein Ergebnis der automatischen Schrittzahlberechnung ist in Abbildung 10.17 dargestellt. Das Timing innerhalb eines Laufzyklus wird normalerweise in Prozent angegeben, wobei 100 % ein vollständiger Laufzyklus *eines* Schrittes *eines* Beines entspricht. In diesem Beispiel wurden deshalb zwei Ordinatenachsen verwendet. Die absolute Schrittzahl wurde in Abbildung 10.17(a) auf der linken Ordinatenachse dargestellt und an der rechten das Timing in Prozent. Beide Größen beziehen sich in den hier gezeigten Beispielen auf das rechte Bein. Normalerweise sind Zahlenwerte für den Laufzyklus über 100 % natürlich nicht üblich, auf einen Umbruch zu 0 % zu Beginn des neuen Schrittes wurde der Einfachheit halber verzichtet. In Abbildung 10.17(b) ist die aus der absoluten Schrittzahl abgeleitete Schrittfrequenz  $f_{st}$  dargestellt. Sie ergibt sich durch den Differenzenquotient zweier aufeinander folgender Schrittzahlen nach folgender Gleichung:

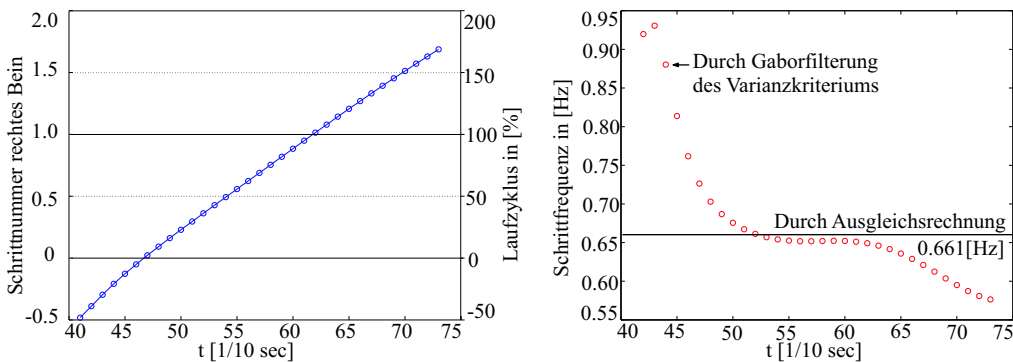
$$f_{st} = \frac{\theta_t - \theta_{t-1}}{\Delta t} \quad (10.5)$$

$$T_{st} = \frac{1}{f_{st}} \quad (10.6)$$

Aus der Schrittfrequenz lässt sich mit Gl.(10.6) auch die Periodendauer eines Schrittes, also die Schrittdauer  $T_{st}$  berechnen. Die horizontale Linie stellt eine mittlere Schrittfrequenz dar, die durch Ausgleichsrechnung über alle Zeitschritte der Sequenz berechnet wurde (siehe weiter unten), sie wird als korrekt angenommen. Mit diesem Wert konnten

#### 10.4. Approximation mit quasiperiodischem Bewegungsmodell

auch die besten Approximationen des Bewegungsmodells an die manuell vorgegebenen Punkte erreicht werden. Es ist zu erkennen, dass der Gaborfilter in der Mitte der Sequenz die Schrittfrequenz nahezu korrekt berechnet hat. Weiterhin ist zu sehen, dass am Anfang und am Ende der Sequenz erhebliche Abweichungen von diesem Wert auftreten. Zu Begründen ist dies mit einem relativ kleinen Bildausschnitt, in denen der Proband von einer Kamera beobachtet werden konnte. Das Varianzkriterium der Vordergrundmaske kann dann lediglich für diesen kleinen Bildausschnitt berechnet werden. Dadurch kommt es bei der Gaborfilterung zu Randeffekten, die wahrscheinlich zu den gezeigten Abweichungen führen.



(a) Durch den Gaborfilter berechnete Schrittzahl für jeden Zeitschritt

(b) Aus der Schrittzahl abgeleitete Schrittfrequenz für jeden Zeitschritt

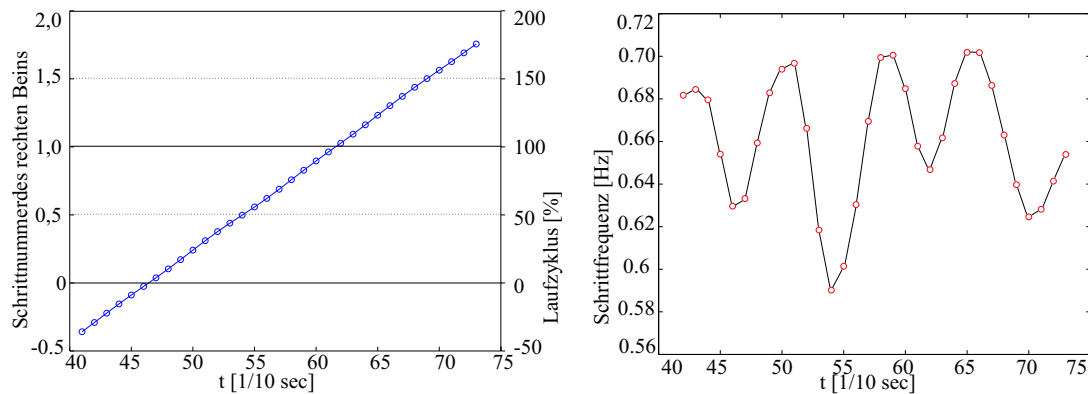
Abbildung 10.17.: Die Schätzungen wurden durch Gaborfilterung der Beinvarianz erhalten (siehe Kapitel 9).

Bei dem in Abbildung 10.14, Abbildung 10.15 und Abbildung 10.16 gezeigten Problem konnte in der Schwungphase die Schrittzahl offensichtlich nicht genau genug berechnet werden. Für die automatische Berechnung von Näherungswerten führt das dann dazu, dass der Kniegelenks- und der Hüftgelenkwinkel falsch in das Modell eingetragen werden. Das periodische Bewegungsmodell sollte aber gerade für die automatische Bereitstellung von Näherungswerten verwendet werden. Wenn sich jedoch die Schrittzahl durch Gaborfilterung nicht genau genug bestimmen lässt, dann ist ihre Verwendung zur automatischen Modellinitialisierung mit Hilfe des Bewegungsmodells fraglich. Da ja bei derartig großen Abweichungen im Ortsraum keine korrekte Korrespondenz zwischen Formmodell und realem Bild mehr möglich ist. Ausgerechnet in diesen problematischen Zeitschritten der Schwungphase kommen sich beide Beine auch noch sehr nahe. Zusätzlich treten die meisten Selbstverdeckungen auf, was die korrekte Korrespondenz noch weiter erschwert, dass wiederum wird zu falschen Ausgleichsergebnissen führen. Aus diesen Gründen ist eine deutlich genauere Synchronisierung des Bewegungsmodells mit dem Bewegungszustand des Probanden erforderlich, als die bisher verwendeten.

### 10.4.2. Genaue Berechnung jeder einzelnen Schrittzahl

In Abschnitt 8.3 wurde beschrieben, wie die Schrittzahlen für jeden Zeitschritt innerhalb einer Ausgleichung berechnet werden kann.

Das Ergebnis ist in Abbildung 10.18 zu sehen. Die Schrittzahlen verlaufen nun weitgehend linear.



(a) Durch Ausgleichsrechnung geschätzte Schrittzahlen für jeden Zeitschritt

(b) Aus der Schrittzahl abgeleitete Schrittfrequenz für jeden Zeitschritt

Abbildung 10.18.: Die genaue Schrittzahl wurde gemeinsam mit den Gelenkwinkeln durch Ausgleichsrechnung ermittelt.

Das erstaunliche Ergebnis der geschätzten Gelenkwinkelverläufe bei gleichzeitig berechneter Schrittzahl ist in Abbildung 10.19 zu sehen. Die geschätzten Standardabweichungen sind nun deutlich kleiner als bei der Einzelschrittausgleichung ohne Bewegungsmodell unter Verwendung der manuell vorgegebenen Punkte (vergleiche mit Abbildung 10.11). Weiterhin sind die Fehlerschranken auch viel kleiner als mit Bewegungsmodell, wenn direkt die durch Gaborfilterung ermittelten Schrittzahlen verwendet werden (vergleiche mit Abbildung 10.14). Die in Abbildung 10.15 gezeigten Probleme in den Schwungphasen wurden somit gelöst. Die Gelenkwinkelverläufe sind glatt und sauber. Erstaunlich ist auch die Symmetrie der Gelenkwinkelverläufe des linken und rechten Beines, wie sie bei einem gesunden Gang auch zu erwarten ist. Zu erwähnen ist, dass sich dieses Ergebnis bereits *ohne* eine spezielle Symmetriebedingung einstellte.

In Abbildung 10.18(b) ist der Verlauf der Schrittfrequenz über der Zeit dargestellt, die Schwankungen betragen etwa 10 % der mittleren Schrittfrequenz. Da die Schwankungen einen deutlich sichtbaren periodischen Anteil haben, ist zu vermuten, dass die Anzahl der Fourierkoeffizienten noch nicht ausreicht, um die Bewegungen des Probanden ausreichend genau zu repräsentieren. Voraussichtlich drücken sich dadurch die Unzulänglichkeiten des Bewegungsmodells in einer geringfügig falschen Schrittzahl aus. Dann müsste sich die Amplitude der **Schrittfrequenz** durch eine Vergrößerung der Anzahl der Fourierkoeffizienten verkleinern. Dies war jedoch nicht der Fall. Abbildung 10.20 zeigt die

### 10.4. Approximation mit quasiperiodischem Bewegungsmodell

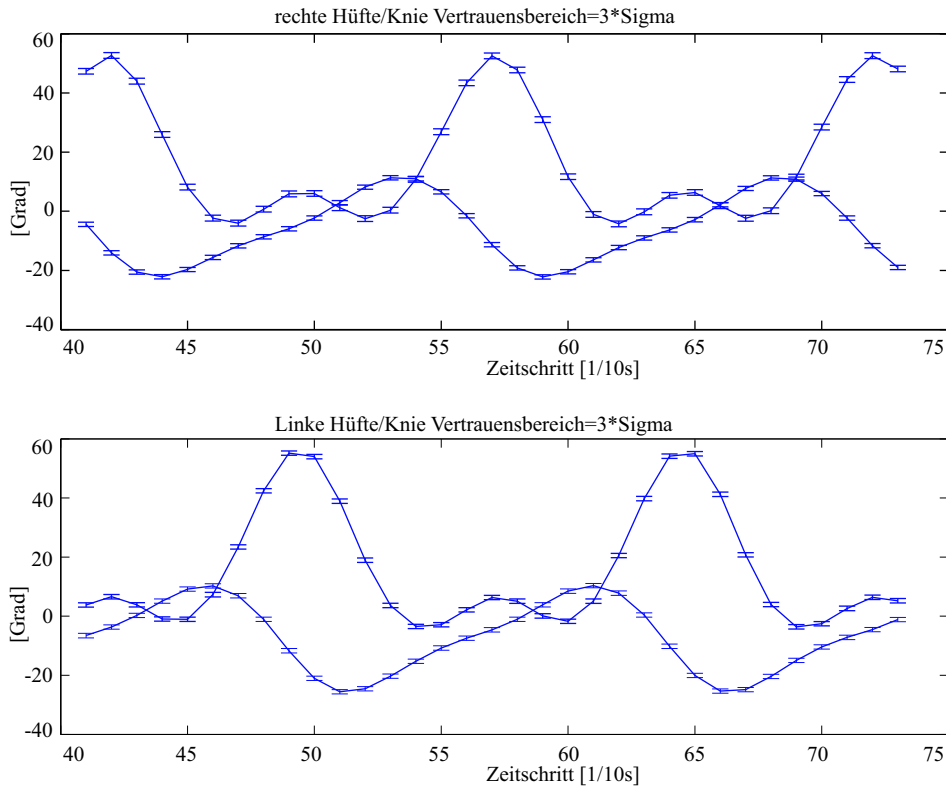
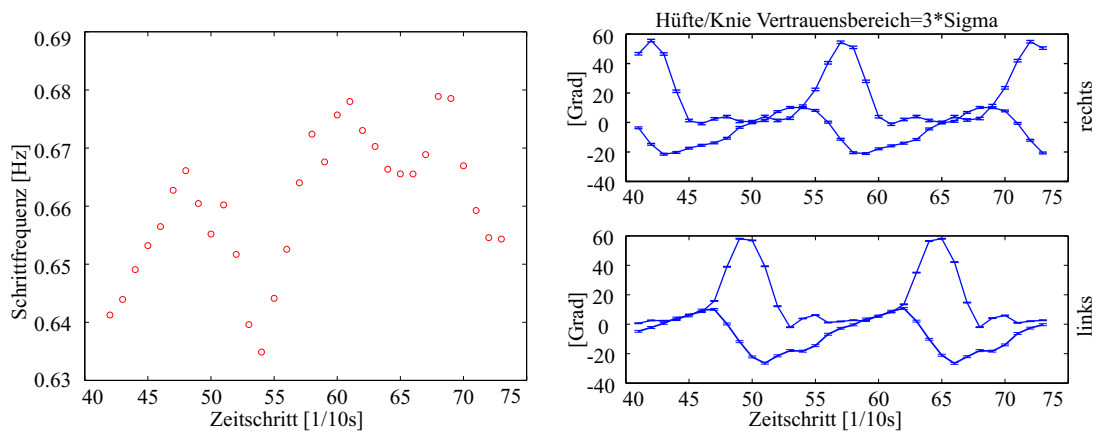


Abbildung 10.19.: Approximation an manuelle Punkte mit Hilfe des periodischen Bewegungsmodells bei gleichzeitig geschätzter Schrittzahl.



(a) Aus der Schrittzahl abgeleitete Schrittfrequenz für jeden Zeitschritt

(b) Gelenkwinkelverläufe

Abbildung 10.20.: Bewegungsmodell mit mehr Fourierkoeffizienten als im vorhergehenden Beispiel

## 10. Experimente, aufgetretene Probleme und deren Lösungen

Ergebnisse. Es änderten sich zwar die konkreten Werte der Schrittfrequenzen, jedoch nicht deren Amplitude.

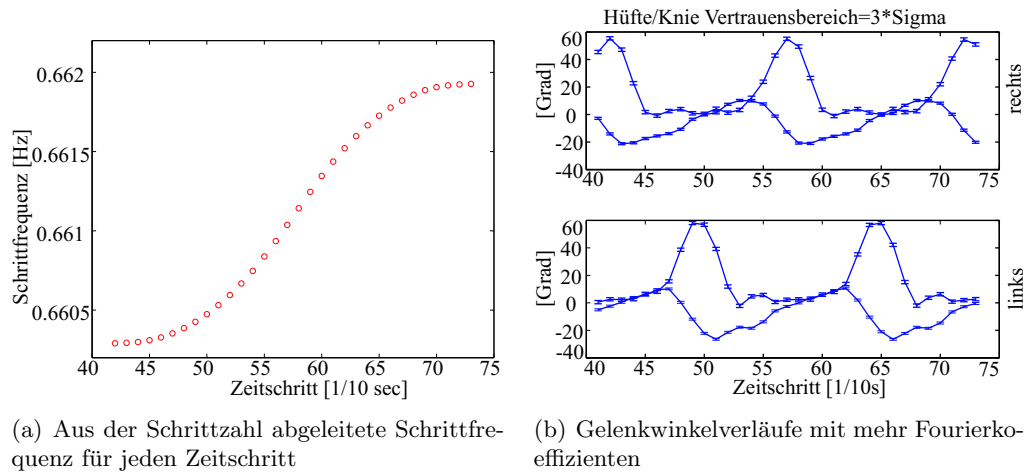


Abbildung 10.21.: Bewegungsmodell mit mehr Fourierkoeffizienten und stärkerer Glättung

Die Ausgleichung wurde deshalb noch einmal durchgeführt, diesmal jedoch mit einer deutlich stärkeren Gewichtung von Gl.(8.30), das Ergebnis zeigt Abbildung 10.21. Es ist eine deutlich bessere Glättung der Schrittfrequenz zu erkennen. Ihre Amplitude ist auch deutlich kleiner geworden (siehe Abbildung 10.21).

Allerdings steigen damit dann auch wieder die Standardabweichungen der geschätzten Gelenkwinkel geringfügig an. Dies ist ein Indiz dafür, dass sich Abweichungen der Probandenbewegung von den repräsentierbaren Bewegungen einer Fourierreihe in einem nichtlinearen Verlauf der Schrittzahl ausdrücken. In beiden Fällen, in denen eine große Anzahl Fourierkoeffizienten verwendet wurde, geht der symmetrische Eindruck der Gelenkwinkelverläufe wieder verloren. Das könnte ein Hinweis darauf sein, dass durch die hohe Parameteranzahl bereits eine Verringerung der Robustheit gegen Ausreißer zu beobachten ist. Es könnte jedoch sein, dass die Bewegung tatsächlich asymmetrisch war, aber ein Bewegungsmodell mit zu wenigen Fourierkoeffizienten diese Asymmetrie nicht wiedergeben kann.

### 10.4.3. Fourierreihe mit vielfacher Periodendauer der Schrittfrequenz

Das Ergebnis des in Abschnitt 8.2.3 beschriebenen Verfahrens ist in Abbildung 10.22 zu sehen.

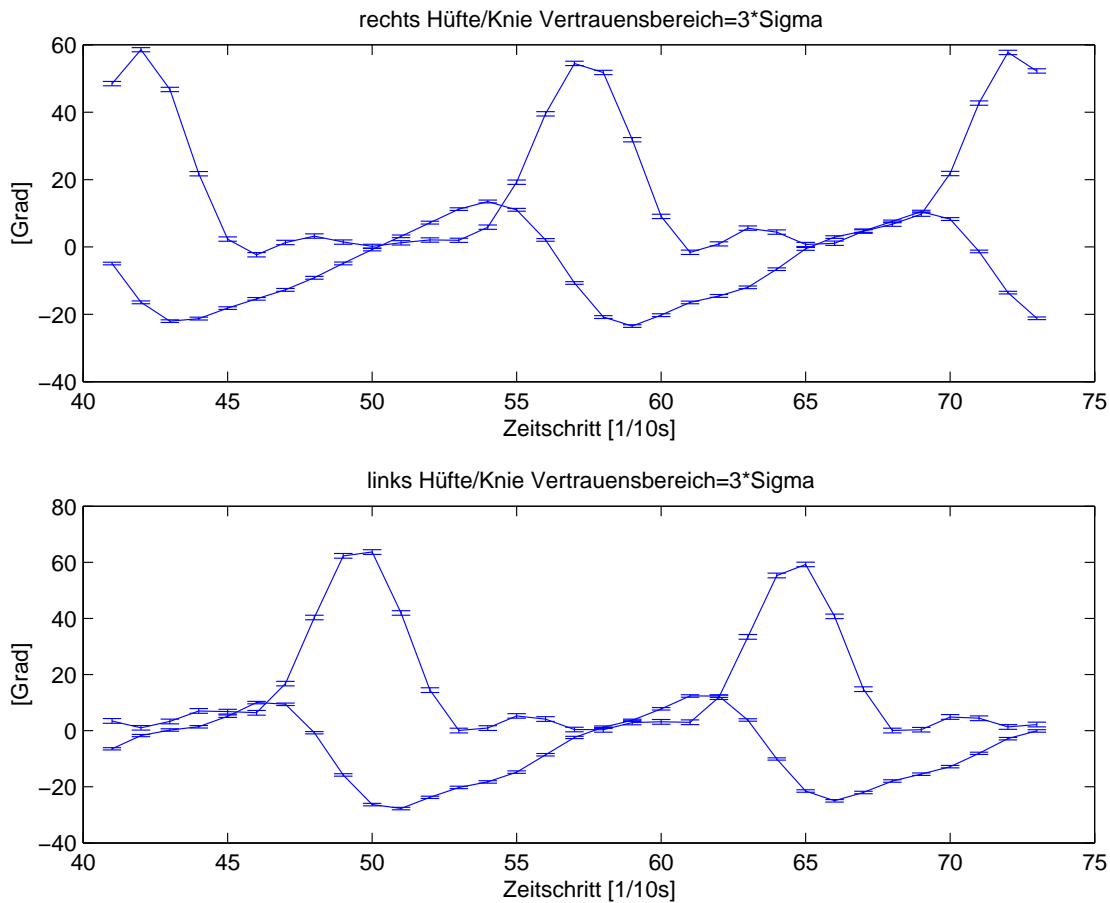


Abbildung 10.22.: Das Bewegungsmodell wurde mit der doppelten Periodendauer der Schrittfrequenz initialisiert (Abschnitt 8.2.3). Es sind geringfügige Abweichungen zwischen beiden Schritten zu erkennen. Dabei wurde auch die mittlere Periodendauer durch Ausgleichsrechnung bestimmt.

*10. Experimente, aufgetretene Probleme und deren Lösungen*



# 11. Ergebnisse des Messsystems

## 11.1. Definition der Messgrößen

Für Testmessungen wurde der nachstehend beschriebene Messaufbau verwendet. Zum Einsatz kam ein Multikamerasystem mit sechs Kameras, sie waren gemäß Abbildung 11.1(a) angeordnet. In diesem Kapitel werden Diagramme der wichtigsten mit dem System erfassbaren Größen vorgestellt.

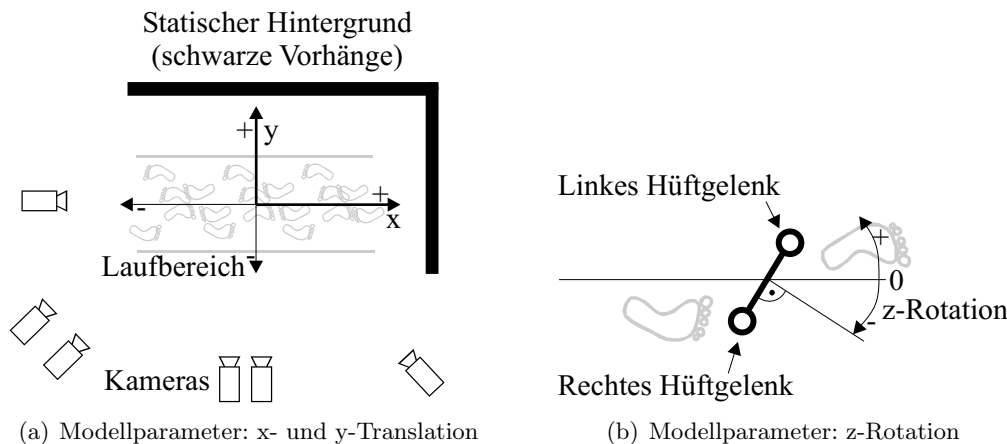


Abbildung 11.1.: Verwendeter Messaufbau und gemessene Größen

Vorzeichen und Richtung der Messgrößen werden durch das Weltkoordinatensystem und durch das 3D-Modell festgelegt. In Abbildung 11.1(a) sind die Translationsparameter des Modells in Draufsicht eingezeichnet. Dabei gibt die x-Translation die Bewegung in Laufrichtung und die y-Translation die Bewegung des Probanden quer zur Laufrichtung an. Abbildung 11.1(b) zeigt die beiden Hüftgelenke eines nach links (in positive x-Richtung) gehenden Probanden. Der Winkel z-Rotation gibt an, wie weit das Modell relativ zur x-Achse des Weltkoordinatensystems verdreht ist.

Sehr wichtig für die Interpretation der nachfolgenden Diagramme ist die Definition der Gelenkwinkel nach Abbildung 11.2. Die Gelenkwinkelverläufe können entweder direkt über der Zeitachse aufgetragen werden, dann besitzt die Abszisse die Einheit Sekunden. Das hat jedoch den Nachteil, dass keine direkte Vergleichbarkeit der Bewegungsparameter möglich ist, weil die Messreihen dann streng gekoppelt an das Timing des Aufnahmesystems dargestellt werden. Im hinteren Teil des Kapitels werden die Messgrößen in

## 11. Ergebnisse des Messsystems

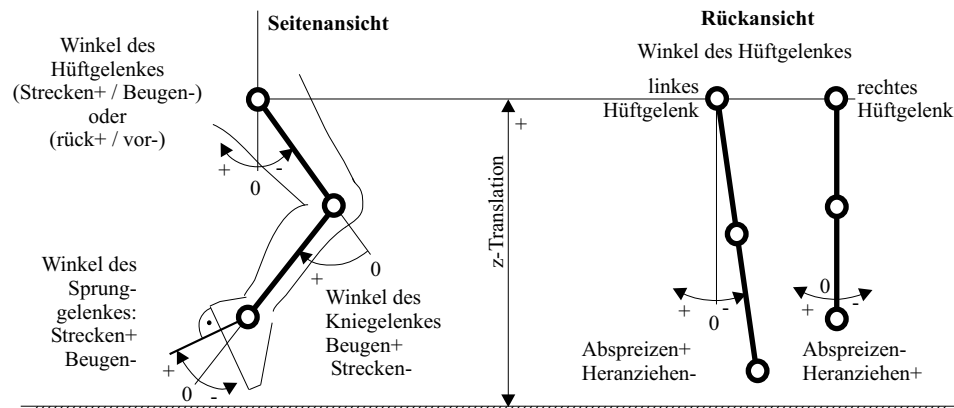


Abbildung 11.2.: Vorzeichendefinition der gemessenen Größen am Modell

Prozent normiert auf einen Laufzyklus angegeben. Denkbar ist auch eine direkte Visualisierung der wenigen Fourierkoeffizienten, allerdings sind diese durch einen menschlichen Beobachter nur schwer zu interpretieren.

Mit dem 3D-Modell ist auch eine Visualisierung der Gelenkpunktkoordinaten im Weltkoordinatensystem möglich. In dem implementierten System kann das 3D-Modell mit der zuvor aufgezeichneten Gangbewegung animiert werden. Allerdings ist eine Normierung und ein Vergleich von verschiedenen Gangmustern aufgrund der Komplexität der Darstellung nur schlecht möglich. Einen Eindruck davon kann man aus Abbildung 10.10 gewinnen. Deshalb werden in diesem Kapitel lediglich die Gelenkwinkelverläufe und die wichtigsten Translationsparameter dargestellt.

### 11.2. Messergebnisse der optischen Ganganalyse

Zur Demonstration, welche Parameter mit dem System erfasst werden können, wurde der Gang eines Probanden zwei Mal kurz nacheinander aufgezeichnet. Der Proband bewegte sich beide Male in die selbe Richtung. Bei diesen Untersuchungen wurde zunächst kein besonderer Wert auf eine Reproduzierbarkeit der Bewegungsparameter gelegt. Nachdem das Modell näherungsweise durch anklicken der Gelenkpunkte initialisiert wurde, basierte die Modellapproximation ausschließlich auf der Beobachtung von Konturpunkten. Die Ergebnisse der Ganganalyse sind im Folgenden dargestellt.

In Abbildung 11.3 und Abbildung 11.4 sind die typischen Gelenkwinkelverläufe des Probanden zu sehen. Die geschätzten Vertrauensbereiche in diesem Kapitel werden mit 3-facher Standardabweichung ( $3\sigma \hat{=} 99.73\%$ ) angegeben. Dargestellt ist der Gelenkwinkelverlauf für das Beugen und Strecken des Hüft- und Kniegelenkes. Das sind die Parameter, die gemäß [CNC03] sehr gut reproduzierbar sein sollten. Bis auf einen zeitlichen Versatz der Gelenkwinkelverläufe erscheinen diese ähnlich. Der zeitliche Versatz ist darauf zurückzuführen, dass als Zeiteinteilung auf der Abszisse der absolute Zeitschritt verwendet

## 11.2. Messergebnisse der optischen Ganganalyse

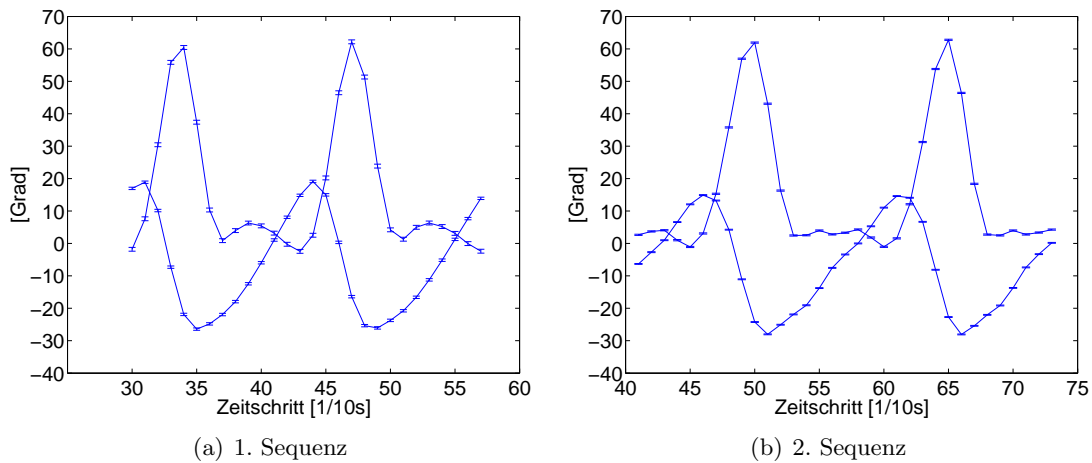


Abbildung 11.3.: Gelenkwinkelverläufe des linken Knies und des linken Hüftgelenkes eines Probanden

wurde, der durch das Aufnahmesystem festgelegt wird. Weiter hinten in diesem Kapitel erfolgt dann eine Normierung bezogen auf den Laufzyklus, die den zeitlichen Versatz bei der Darstellung eliminiert.

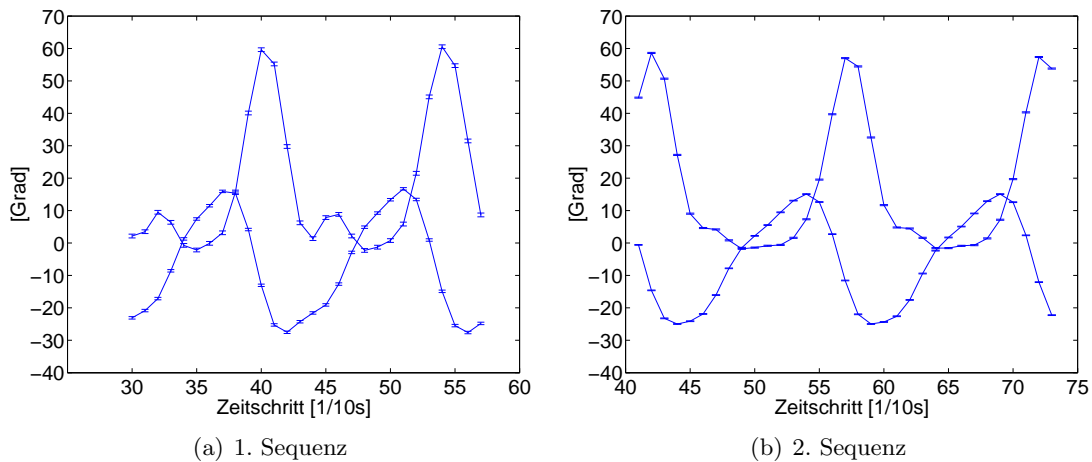
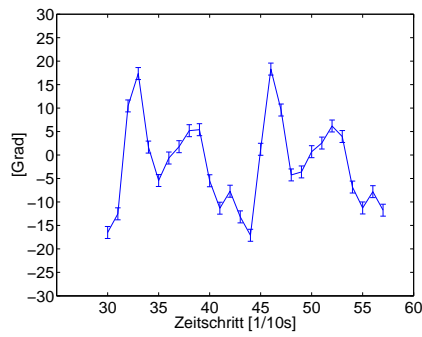


Abbildung 11.4.: Gelenkwinkelverläufe des rechten Knies und des rechten Hüftgelenkes eines Probanden

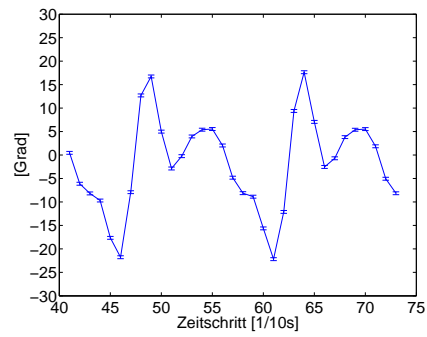
In Abbildung 11.5 sind die Gelenkwinkelverläufe der Sprunggelenke zu sehen. Die Verläufe sind zwar zwischen rechtem und linkem Bein ähnlich, sie unterscheiden sich aber, wenn die erste mit der zweiten Sequenz verglichen wird. Diese Winkel lassen sich lediglich mit mittlerer Genauigkeit bestimmen.

In Abbildung 11.6 ist der zurückgelegte Weg des Probanden über der Zeit aufgetra-

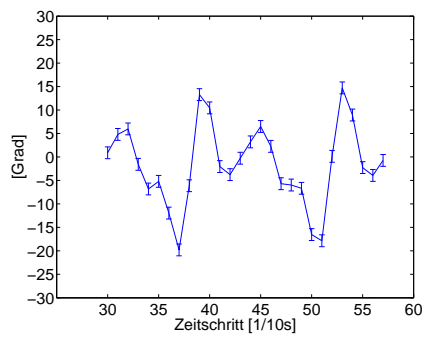
## 11. Ergebnisse des Messsystems



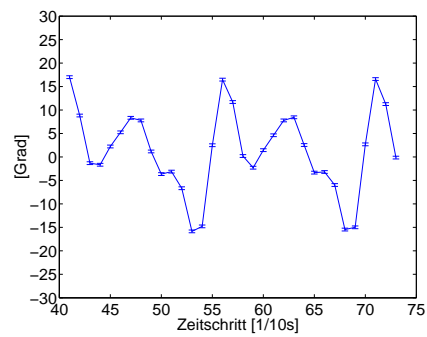
(a) 1. Sequenz(L)



(b) 2. Sequenz(L)

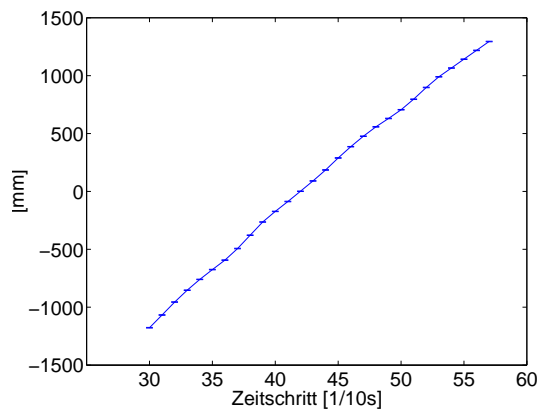


(c) 1. Sequenz(R)

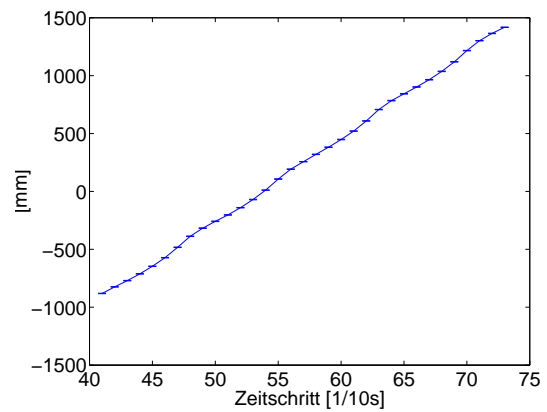


(d) 2. Sequenz(R)

Abbildung 11.5.: Gelenkwinkelverläufe des linken und rechten Sprunggelenkes eines Probanden



(a) 1. Sequenz



(b) 2. Sequenz

Abbildung 11.6.: Translation in x-Richtung (nahezu Laufrichtung)

## 11.2. Messergebnisse der optischen Ganganalyse

gen (Weg-Zeit-Diagramm). Das Bild zeigt die Translation des Probanden in x-Richtung des Weltkoordinatensystems. Da die Hauptbewegungsrichtung des Probanden mit der x-Richtung des Weltkoordinatensystems zusammenfällt, ist ein nahezu linearer Anstieg des zurückgelegten Weges über der Zeit zu erkennen. Überlagert wird er durch eine nahezu periodische Größe mit geringer Amplitude. Dies ist typisch für eine Gangbewegung und hat seine Ursache in der Umwandlung von potenzielle in kinetische Energie (und umgekehrt).

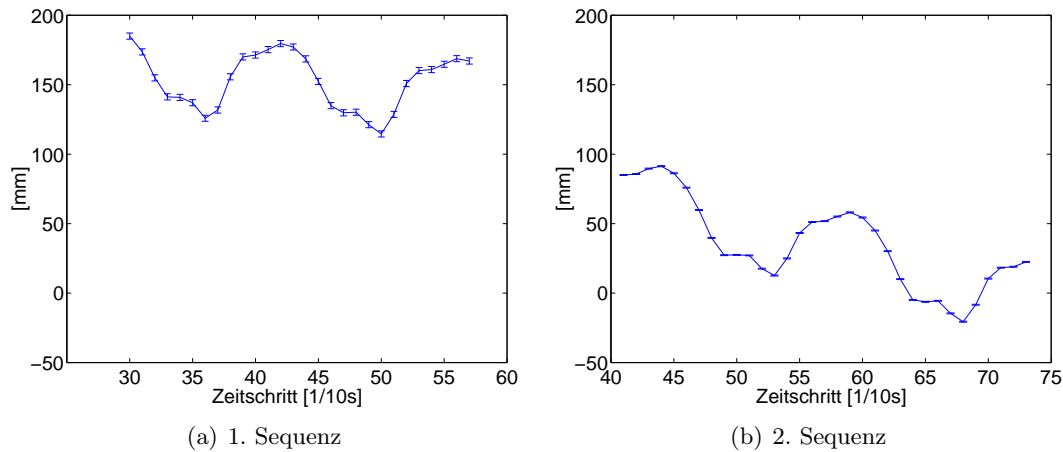


Abbildung 11.7.: Translation in y-Richtung (nahezu quer zur Laufrichtung)

Die Translation in y-Richtung ist in Abbildung 11.7 zu sehen. Es ist eine Ähnlichkeit der Abbildungen links und rechts zu erkennen. Offensichtlich bewegte sich der Proband beide Male etwas auf die Kameras zu (in negative y-Koordinatenrichtung). Das ist an dem langsamen Abfallen der Kurven zu erkennen. Überlagert wird diese Bewegung durch einen periodischen Anteil von etwa 50 mm quer zur Laufrichtung.

## 11. Ergebnisse des Messsystems

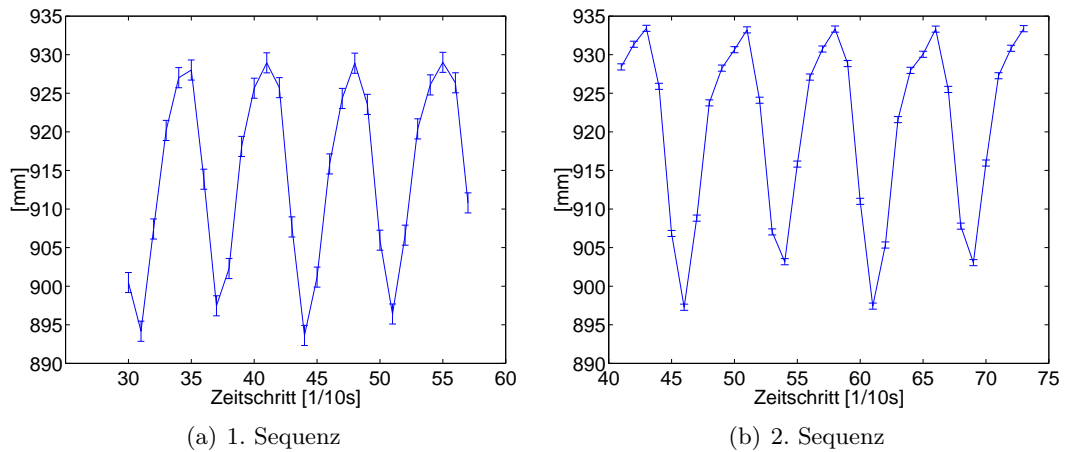


Abbildung 11.8.: Translation in z-Richtung (Aufwärts- und Abwärtsbewegung)

In Abbildung 11.8 ist die Auf- und Abwärtsbewegung des Probanden dargestellt. Beide Male sind ähnliche Verläufe zu sehen, mit einer Amplitude von etwa 30 mm.

Die Rotation der Hüfte um die z-Achse ist in Abbildung 11.9 zu sehen. Dieser Winkel gehört zu den Größen, die nur sehr ungenau berechnet werden können.

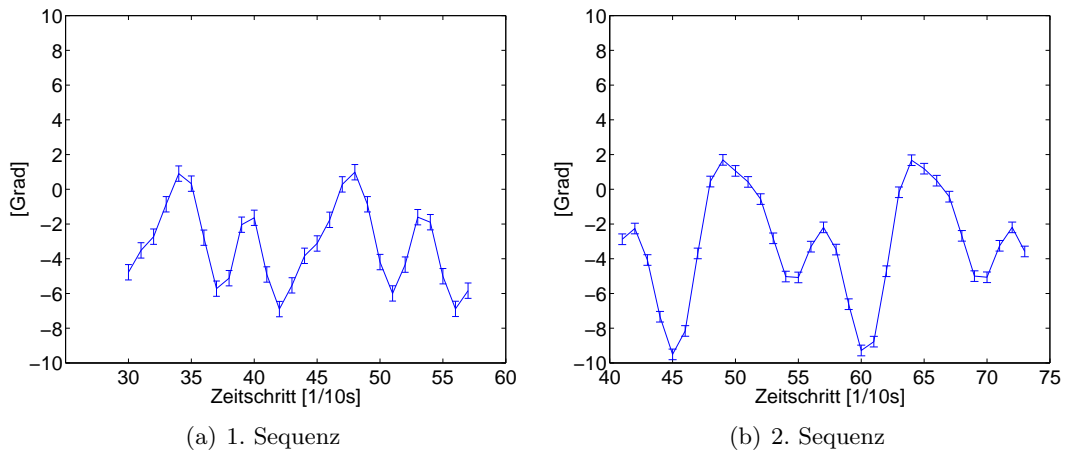


Abbildung 11.9.: Rotation um z-Achse (Körperlängsachse)

## 11.2. Messergebnisse der optischen Ganganalyse

In Abbildung 11.10 ist das Abspreizen und Heranziehen des linken Hüftgelenkes zu sehen. Aufgrund der Kamerakonfiguration können diese Winkel nur ungenau berechnet werden. Weiterhin ist die Amplitude dieser Winkel nur klein, das führt zu einer scheinbar großen Streuung der Parameter. Dasselbe gilt für das Abspreizen und Heranziehen des rechten Hüftgelenkes, dies ist in Abbildung 11.11 zu sehen.

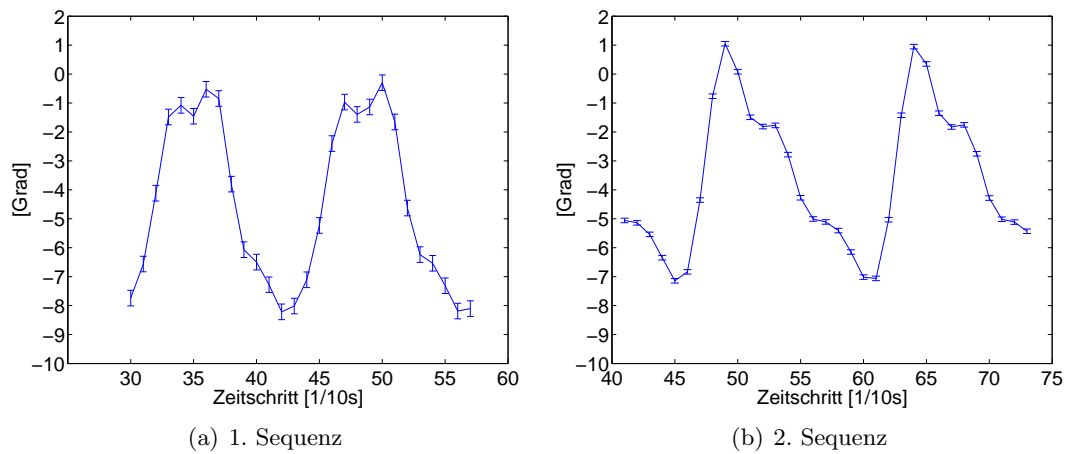


Abbildung 11.10.: Abspreizen und Heranziehen des linken Hüftgelenkes

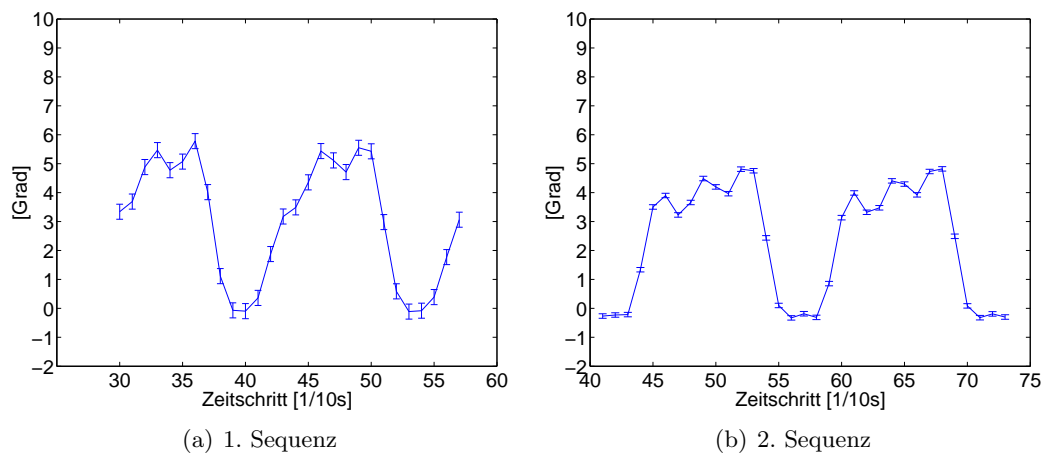


Abbildung 11.11.: Abspreizen und Heranziehen des rechten Hüftgelenkes

### 11.3. Reproduzierbarkeit der Messungen und Simulation von Gangfehlern

Besser als die zuvor gezeigten absoluten Zeitangaben, ist eine normierte Darstellung der Bewegungsparameter bezogen auf den Laufzyklus nach Abschnitt 2.1.1 und Anhang A. Dadurch wird die Darstellung unabhängig von den Zeitvorgaben des Aufnahmesystems. Wie in der Literatur üblich, erhält die Abszisse dann die Einheit Prozent (%). Die Bewegungsparameter verschiedener Messungen können dadurch direkt miteinander verglichen werden. Durch eine Verschiebung der Gelenkwinkelverläufe eines Beines um 50 %, wird eine einheitliche Darstellung für das linke und das rechte Bein erreicht. Damit können dann leicht Vergleiche der Gangsymmetrie angestellt werden. Es sind jedoch kleinere Abweichungen in zeitlicher Richtung zu erwarten. Sie sind darauf zurückzuführen, dass die Genauigkeit der Schrittzahl durch die manuelle Initialisierung auf einen Zeitschritt begrenzt war. Die Bildsequenzen wurden mit einer Frequenz von 20 Hz, statt bisher mit 10 Hz aufgezeichnet. Ein Zeitschritt der Bildfolge (20 Hz) bedeutet bei einer mittleren Schrittfrequenz von 0.74 Hz etwa 3.7 % des Laufzyklus.

Zur Initialisierung wurde in der Bildsequenz ein Zeitschritt ausgewählt, der den Probanden im Fersenkontakt (Initial Contact) des rechten Beines zeigt. Eine mittlere Schrittfrequenz wurde für jede Bildfolge zunächst manuell vorgegeben. Sie berechnet sich, indem die mittlere Periode des vorhergehenden Initial Contact und des nachfolgenden Initial Contact ermittelt wurde. Durch Ausgleichsrechnung wurden die Translationsparameter, Fourierkoeffizienten eines Laufzyklus und die Schrittzahl für jeden Zeitschritt geschätzt. Die mittleren Abweichungen zwischen Modell und Bild ( $\sigma$  a posteriori) lagen zwischen 0.8-0.98 Pixel, das Modell konnte also gut an die Bilddaten approximiert werden. Das Bewegungsmodell erlaubt die Umrechnung der geschätzten Fourierkoeffizienten in die hier dargestellten Gelenkwinkelverläufe des Laufzyklus.

#### 11.3.1. Gegenüberstellung der Laufrichtung

Der Proband wurde angewiesen, sechs Mal möglichst gleichmäßig und reproduzierbar den Laufbereich zu durchqueren. Ziel war es, einen Eindruck der Reproduzierbarkeit der gewonnenen Bewegungsparameter zu erhalten. Dafür wurden die Gelenkwinkelverläufe mit unterschiedlichen Farben gekennzeichnet. In den Diagrammen wurden zwei Farben verwendet:

- rot: der Proband bewegte sich von links nach rechts über den Laufbereich (in positive x-Koordinatenrichtung)
- schwarz: der Proband bewegte sich von rechts nach links über den Laufbereich (in negative x-Koordinatenrichtung)

In Abbildung 11.12 und Abbildung 11.13 sind die gut reproduzierbaren Gelenkwinkelverläufe der Hüfte und des Knies zu sehen.



11.3. Reproduzierbarkeit der Messungen und Simulation von Gangfehlern

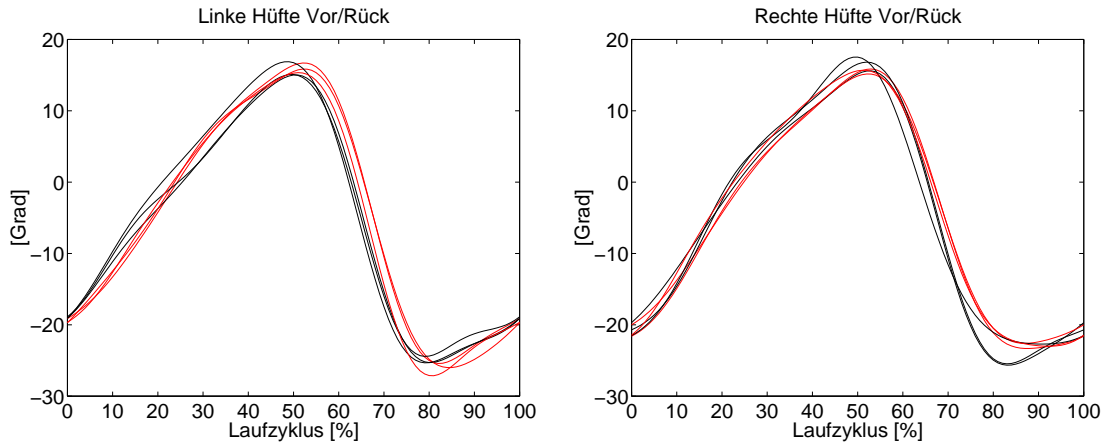


Abbildung 11.12.: Reproduzierbarkeit der Gelenkwinkelverläufe des Hüftgelenkwinkels

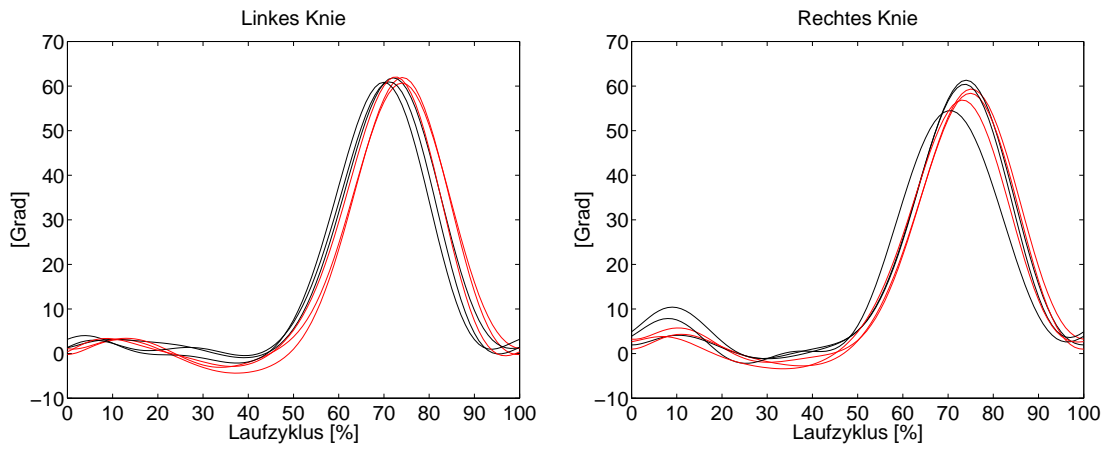


Abbildung 11.13.: Reproduzierbarkeit der Gelenkwinkelverläufe des Knies

## 11. Ergebnisse des Messsystems

Anfällig gegen einen Wechsel der Laufrichtung zeigen sich in Abbildung 11.14 die Gelenkwinkelverläufe der Sprunggelenkwinkel. Es kommt sowohl zu einem zeitlichen Offset des Gelenkwinkelverlaufes, als auch zu einem Winkeloffset. Mit hoher Wahrscheinlichkeit sind Selbstverdeckungen des Fußes die Ursache für diese Abweichungen. Aber auch reale biologische Variationen sind denkbar.

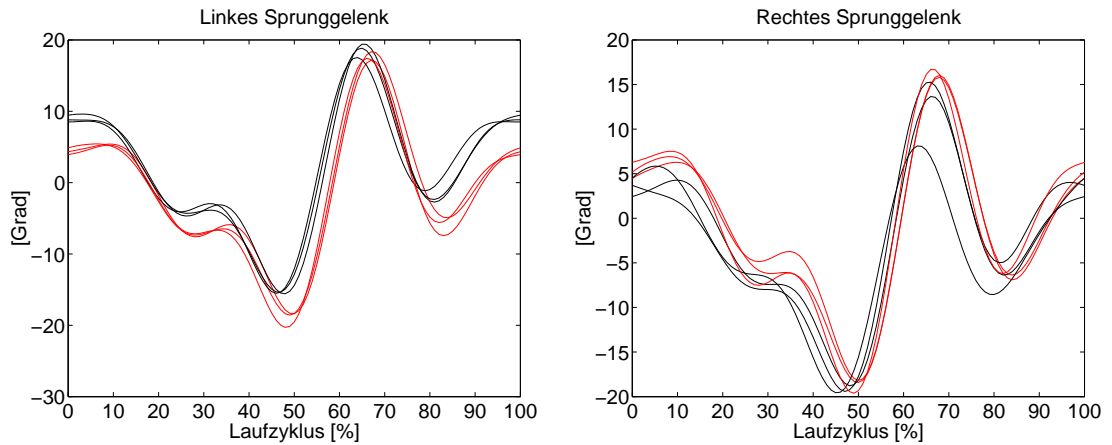


Abbildung 11.14.: Reproduzierbarkeit der Gelenkwinkelverläufe des Sprunggelenkwinkels

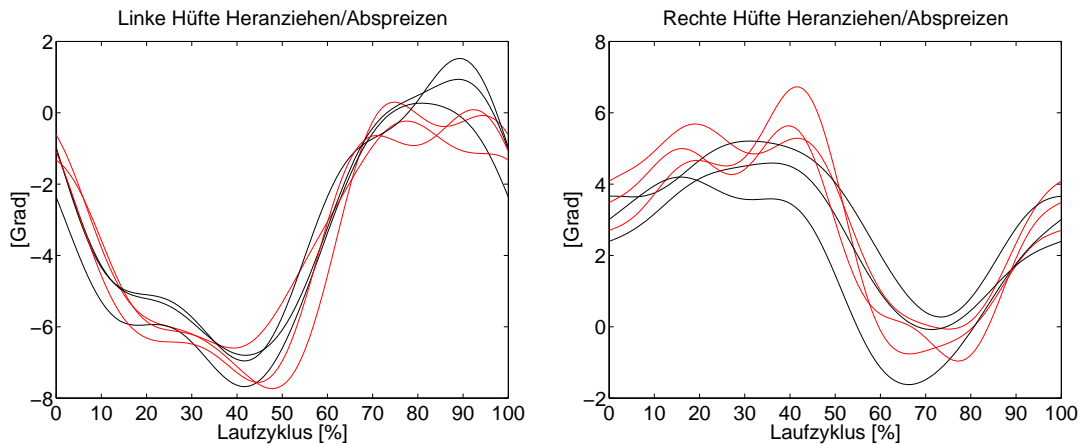


Abbildung 11.15.: Die Gelenkwinkelverläufe des Hüftgelenkes für das Abspreizen und Heranziehen sind nicht gut reproduzierbar.

Genauso schlecht reproduzieren lassen sich die Gelenkwinkelverläufe des Hüftgelenkes für das Abspreizen und das Heranziehen des Beines. Das deutete sich bereits im vorhergehenden Abschnitt an. Dort wurden bereits relativ große Standardabweichungen für diese Parameter vorhergesagt. In Abbildung 11.15 sind die Gelenkwinkelverläufe zu sehen.

### 11.3. Reproduzierbarkeit der Messungen und Simulation von Gangfehlern

Die Streuung der Translationsparameter ist in Abbildung 11.16 zu sehen. In Abbildung 11.16(a) ist die Bewegung quer zur Laufrichtung zu sehen. Derzeitig unterscheidet sich das Vorzeichen der Bewegung in Abhängigkeit von der Laufrichtung. Mit einem etwas größeren Aufwand bei der Modellierung ließe sich dies korrigieren. Aber das wurde in dieser Arbeit noch nicht getan. Dagegen ist eine gute Reproduzierbarkeit der Auf- und Abwärtsbewegung des Probanden in Abbildung 11.16(b) zu erkennen.

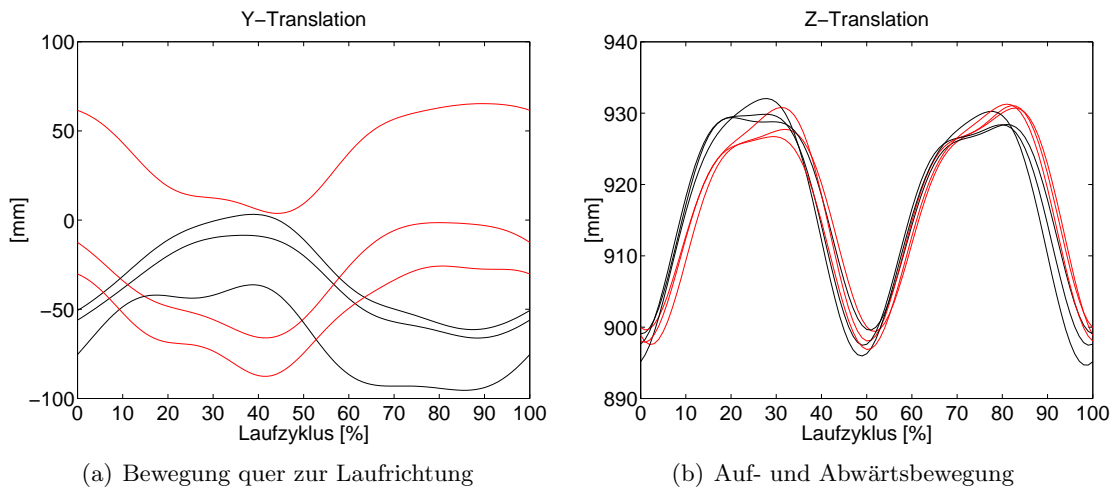


Abbildung 11.16.: Translationsparameter

Die Angaben aus der Literatur [CNC03] können mit diesen Messreihen bestätigt werden: am besten reproduzierbar erscheint der Hüftgelenks- und der Kniegelenkwinkel.

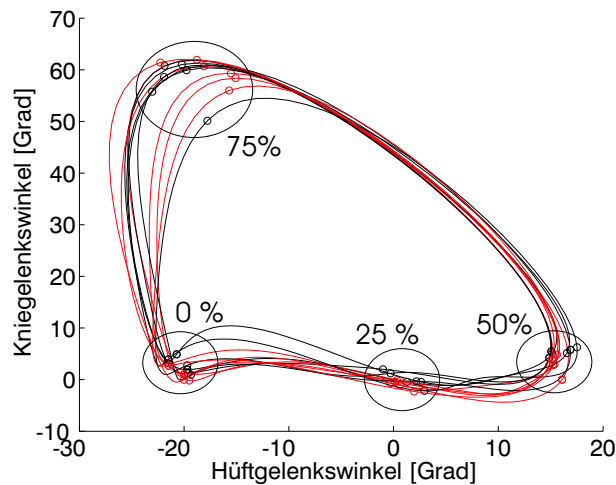


Abbildung 11.17.: Trajektorien des linken und rechten Beines im Zustandsraum

### 11.3.2. Gegenüberstellung von gesundem Gang und simulierter Erkrankung

Die im vorhergehenden Abschnitt bereits gezeigten Daten werden in diesem Abschnitt noch einmal verwendet. Für die Aufzeichnung dieser Bildsequenzen durchquerte der Proband den Laufbereich sechs Mal möglichst reproduzierbar. Insgesamt wurden aber neun Bildsequenzen mit abwechselnder Laufrichtung aufgezeichnet. In drei der Bildsequenzen simulierte der Proband eine Störung des Bewegungsapparates des rechten Beines. Einmal wurde der rechte Fuß ständig gestreckt, um eine Verkürzung der Achillessehne zu simulieren. Danach wurde zweimal das rechte Knie ständig gestreckt.

Für den Nachweis der Anwendbarkeit des Verfahrens zur Feststellung einer Störung des Bewegungsapparates, wurden die gesunden Gelenkwinkelverläufe (schwarz) den simulierten Erkrankungen (farbig) gegenübergestellt. Dabei bedeuten:

- x-blau: rechte Achillessehne verkürzt, Bewegung in positive x-Richtung
- o-rot: rechte Achillessehne verkürzt, Bewegung in negative x-Richtung
- .-margenta: rechtes Knie dauerhaft gestreckt, Bewegung in negative x-Richtung
- -schwarz: normaler Gang in verschiedene Richtungen, siehe Abschnitt 11.3.1

Es ist deutlich zu sehen, dass die Gelenkwinkelverläufe einer simulierten Erkrankung von denen eines gesunden Probanden zum Teil signifikant abweichen. Somit ist eine Möglichkeit der Erkennung von größeren Störungen des Bewegungsapparates nachgewiesen worden.

Da die Störungen immer das rechte Bein betrafen, sind dort auch die größten Abweichungen der Gelenkwinkel zu erkennen. Es ist aber auch festzustellen, dass sich eine Störung des rechten Beines auch in einem leicht veränderten Gangbild des linken Beines ausdrückt. Details müssen einer nachfolgenden Arbeit unter intensiver Mitarbeit von Medizinern vorbehalten sein. Hier sollte deren prinzipielle Eignung und die Empfindlichkeit des Verfahrens nachgewiesen werden.

### 11.3. Reproduzierbarkeit der Messungen und Simulation von Gangfehlern

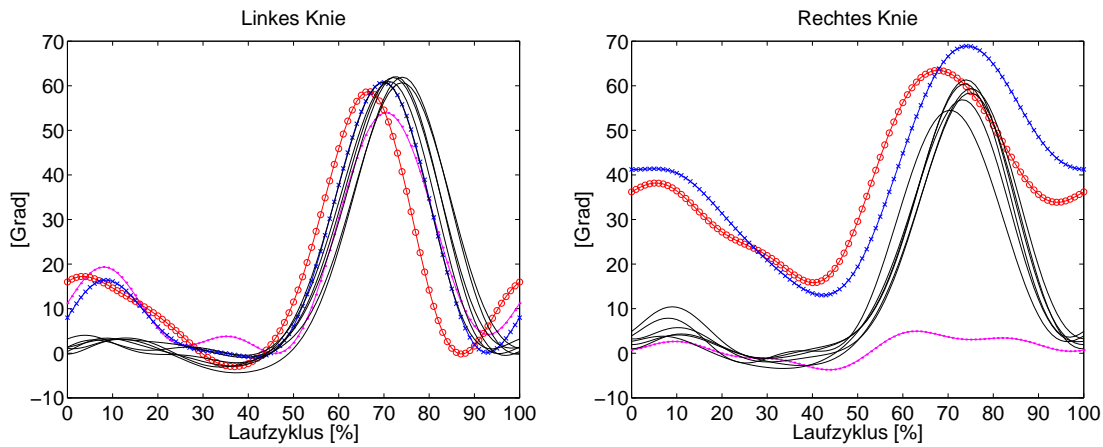


Abbildung 11.18.: Reproduzierbarkeit der Gelenkwinkelverläufe des Knies (schwarz: durchgezogen), Simulierte Gangstörungen (farbig: Punkte, Kreise, Kreuze)

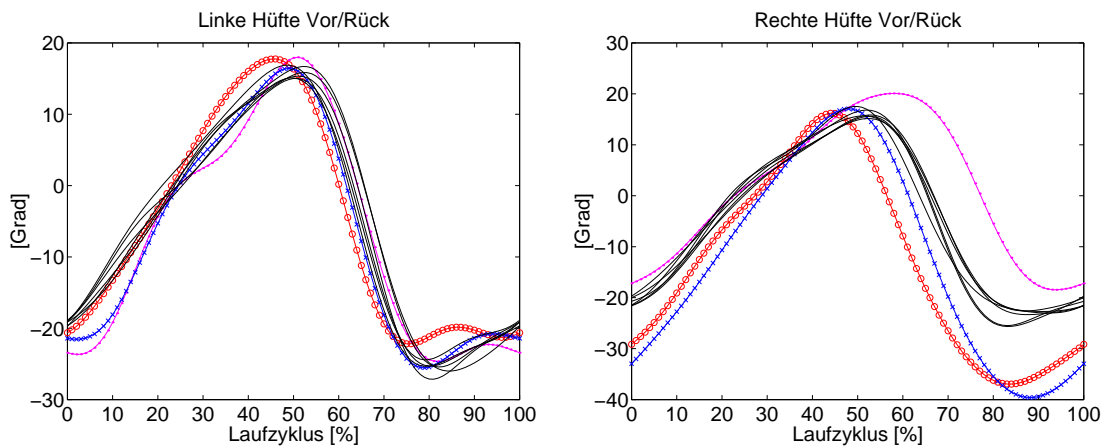


Abbildung 11.19.: Reproduzierbarkeit der Gelenkwinkelverläufe des Hüftgelenkwinkels (schwarz: durchgezogen), Simulierte Gangstörungen (farbig: Punkte, Kreise, Kreuze)

## 11. Ergebnisse des Messsystems

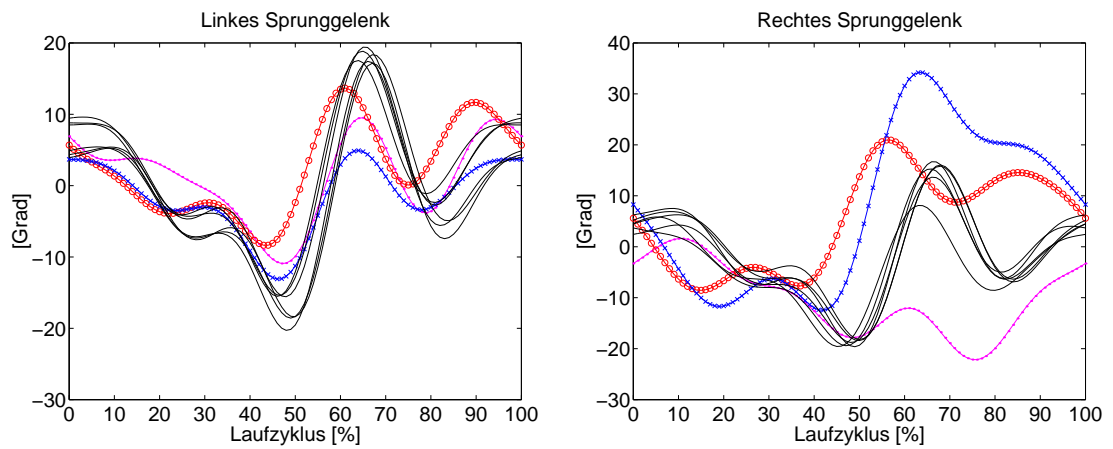
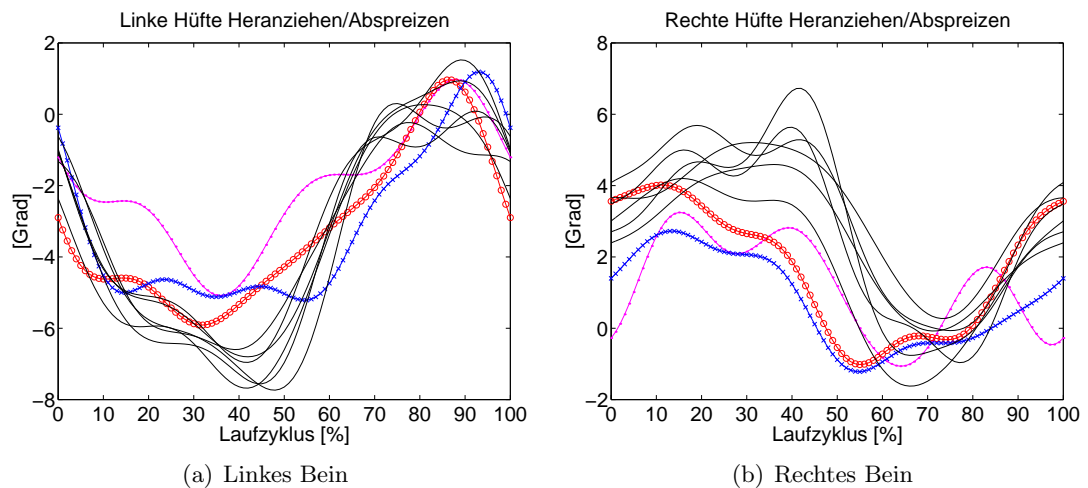


Abbildung 11.20.: Reproduzierbarkeit der Gelenkwinkelverläufe des Sprunggelenkwinkels (schwarz: durchgezogen), Simulierte Gangstörungen (farbig: Punkte, Kreise, Kreuze)



(a) Linkes Bein

(b) Rechtes Bein

Abbildung 11.21.: Die Gelenkwinkelverläufe des Hüftgelenkes für das Abspreizen und Heranziehen sind anscheinend nicht gut reproduzierbar. Normaler Gang (schwarz: durchgezogen), Simulierte Gangstörungen (farbig: Punkte, Kreise, Kreuze)

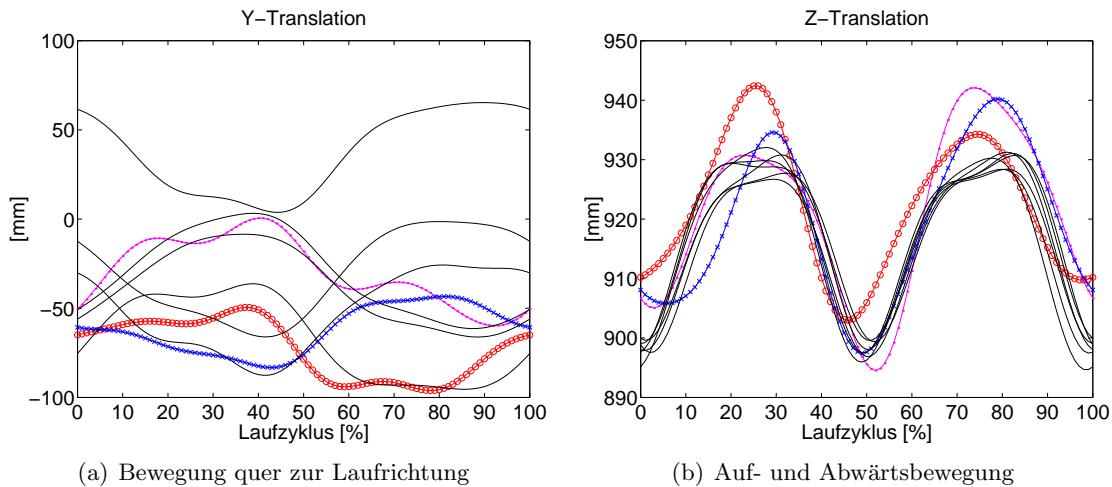


Abbildung 11.22.: Normaler Gang (schwarz: durchgezogen), Simulierte Gangstörungen (farbig: Punkte, Kreise, Kreuze)

## 11.4. Bewertung der Ergebnisse

Bei der optischen Ganganalyse gehen mit dem Weglassen von Markern relevante Informationen verloren, die sich zum Teil durch ein 3D-Modell ersetzen lassen.

Am Anfang der Arbeit wurde gezeigt, dass ein enger Zusammenhang zwischen der verfügbaren Bildauswertung und der Anzahl zuverlässig zu bestimmender Parameter besteht. Unter Berücksichtigung, dass bei einer Konturauswertung nur wenig Bildinformation verfügbar ist, wurde ein sehr einfaches 3D-Modell der unteren Extremitäten mit nur wenigen freien Parametern implementiert. Für die Parameterbestimmung der unteren Extremitäten auf der Basis von Bildkonturen, ist es sehr wichtig, die Füße des Probanden grob zu modellieren. Nur dadurch kann eine ausreichende Anzahl Konturpunkte in horizontaler Richtung erfasst werden, die wichtig sind, um die vertikale Position des 3D-Modells zu bestimmen.

Neu an dieser Arbeit ist die Verwendung eines (quasi-)periodischen Modells der Bewegung, das über der Zeit betrachtet, die Anzahl der freien Parameter weiter senkt. Die Parameterschätzung erfolgt nicht klassisch durch ein Tracking von Zeitschritt zu Zeitschritt, sondern für einen Teil der Bildsequenz gleichzeitig. Dadurch können relativ geringe Bildfrequenzen verwendet werden. Allerdings werden gute Näherungswerte für die Modellparameter benötigt, um eine sichere Korrespondenz zu gewährleisten.

Ebenfalls neu ist die Bereitstellung der Näherungswerte durch das (quasi-)periodische Bewegungsmodell. Dafür werden die zuvor bereit zu stellenden Koeffizienten einer mittleren Gangbewegung verwendet und zeitlich mit der zu erfassenden Probandenbewegung synchronisiert. Automatische Algorithmen dafür wurden vorgestellt und implementiert. Erfolgreich getestet wurde bisher eine sehr einfache Synchronisierung. Sie benötigt die

## 11. Ergebnisse des Messsystems

Angabe von zwei möglichst weit auseinander liegenden Zeitschritten des Heel Strikes. Daraus lässt sich die Phase und Periodendauer des Laufzyklus näherungsweise ermitteln. Eine genauere Schätzung der Periodendauer oder der Schrittzahl erfolgt während der Ausgleichung.

Es konnte gezeigt werden, dass sich die Gelenkwinkelverläufe der Hüfte und des Knies genau und reproduzierbar bestimmen lassen. Gut bestimmbar ist auch die Translationsbewegung in x- und z-Richtung. Eine mittlere Genauigkeit ist bei der Bestimmung des Sprunggelenkwinkels zu erreichen. Weniger Reproduzierbar lässt sich das Abspreizen und Heranziehen des Hüftgelenkes erfassen. Eine Ursache kann die normale biologische Variabilität des Ganges sein, weiterhin beträgt die Amplitude dieser Winkel nur wenige Grad und zeigt die normale Auflösungsgrenze des Systems.

Wie bei jedem Messsystem bleiben Unsicherheiten bei der Parameterbestimmung. Gezeigt wurde, dass größere Defekte des Bewegungsapparates aus diesen Parameterunsicherheiten deutlich hervortreten. Die dabei ermittelten Parameter weichen signifikant von den Streuungen eines gesunden Ganges ab. In einer Phase der praktischen Erprobung wird sich dann zeigen, wie sich die gewonnenen Daten für die Diagnose und Therapie verwenden lassen.

Ausgehend von dieser Präzisierung der Anforderungen in Zusammenarbeit mit Anwendenden, muss der Routinebetrieb wesentlich erleichtert werden. Das gilt insbesondere für die weitgehend automatische Adaption des Systems an häufig wechselnde Probanden. Auch eine Modifikation der gegenwärtig sehr einfachen Formbeschreibung des Modells der unteren Extremitäten kann die Erfassung klinisch relevanter Größen noch verbessern. Dies muss aber nachfolgenden Arbeiten vorbehalten bleiben.



## 12. Zusammenfassung

Die exakte Erfassung der Bewegung eines menschlichen Probanden ermöglicht viele neue Anwendungen. Der menschliche Gang als Sonderfall der menschlichen Bewegung lässt sich in wohl definierte Bewegungsphasen unterteilen. Das Gangbild eines Menschen ist von vielen Parametern abhängig, die individuell variieren. Eine automatische Ganganalyse ist hauptsächlich für zwei Anwendungen interessant. Die Identifikation eines Probanden anhand seines Gangbildes und die medizinische Ganganalyse zur Diagnose von Störungen des Bewegungsapparates. Die Aufgabe des Messsystems ist eine möglichst automatische Berechnung der Bewegungsparameter eines menschlichen Probanden aus einer Anzahl von synchron aufgezeichneten Bildsequenzen. Die weitere Verarbeitung der Bewegungsdaten für die einzelnen Anwendungen ist nicht mehr Gegenstand der Arbeit.

Das Mehrkamerasystem besteht aus einem Aufbau statischer Kameras, die über ein gemeinsames Triggersignal ausgelöst werden. So ist eine simultane Bildaufnahme möglich. Die Haltung des Probanden wird dadurch zu einem bestimmten Zeitpunkt aus verschiedenen Richtungen aufgezeichnet. Vor der eigentlichen Messung müssen die Kameras in einem gemeinsamen Koordinatensystem kalibriert werden. Zur Kalibrierung wurde ein spezielles Verfahren entwickelt, das ohne einen aufwändigen Kalibrierkörper auskommt. Die Kalibrierung selbst erfolgt durch klassische Bündelblockausgleichung.

Es wurden Untersuchungen zur Realisierung eines Systems zur Bewegungsanalyse durchgeführt, das die Bilddaten des Mehrkamerasystems verwendet. Bisher war es zur Bewegungsanalyse eines menschlichen Probanden notwendig die Person zu markieren. Wenn keine Marker eingesetzt werden sollen, gehen jedoch wichtige Informationen für die automatische Bildfolgenauswertung verloren. Das Hauptanliegen dieser Arbeit war es deshalb ein System zu entwickeln, das selbst ohne diese Informationen brauchbare Ergebnisse liefert.

Die fehlenden Informationen können zum Teil durch ein 3D-Modell des Probanden ersetzt werden. Deshalb wurde zur Bildauswertung ein modellbasierter Ansatz gewählt. Der Ansatz basiert auf dem Analyse-durch-Synthese-Konzept aus der Bildcodierung. Ein wichtiges Element des Systems ist eine Kette zur Synthese künstlicher Bilddaten, wenn ein geometrisches Modell der physischen Gegebenheiten zur Verfügung steht. Die Bestimmung der Parameter des Modells, erfolgt durch einen Soll-Ist-Vergleich der synthetischen Bilddaten (Ist) mit den real aufgenommenen (Soll) und einer iterativen Minimierung der Abweichungen.

Der Soll-Ist-Vergleich lässt sich dann auf einer höheren Abstraktionsebene durchführen, was auch eine exakte Pixel-für-Pixel Synthese erübrigt. Es lassen sich dann vereinfachte

## 12. Zusammenfassung

Modelle verwenden, die z.B. lediglich Konturmerkmale synthetisieren.

Die Bewegungsanalyse soll eine möglichst genaue Schätzung der Bewegungsparameter leisten. Für eine Weiterverarbeitung der Bewegungsdaten in anderen Anwendungen ist eine Abschätzung der Genauigkeit der Parameter sehr wichtig. Das kann mit Methoden der Fehler- und Ausgleichsrechnung erfolgen, die aus der Photogrammetrie bekannt sind. Mit dieser Arbeit wurden deshalb solche Methoden für den Analyse-durch-Synthese-Ansatz angepasst.

Bei der Implementierung und ersten Versuchen wurden vielfältige Probleme sichtbar. Dazu gehört die Modellbildung selbst und die Gewinnung von guten Näherungswerten der Bewegung. Um das Problemfeld einzuschränken wurde das System deshalb zunächst für den speziellen Anwendungsfall der medizinischen Ganganalyse konzipiert. Das erlaubt vereinfachende Annahmen hauptsächlich bei der Modellbildung und -initialisierung.

Für die Ganganalyse sind die Bewegungen der unteren Extremitäten eines Probanden interessant. Da es sich um weitgehend periodische Bewegungen handelt, lassen sich Nebenbedingungen formulieren, die zu einem näherungsweise periodischen Modell der Bewegung führen. Dabei werden die Gelenkwinkel durch eine Fourierreihe approximiert, so dass eine Reduktion der Parameterzahl im Frequenzraum erfolgen kann. Die Parameterschätzung erfolgt dann nicht mehr in jedem Zeitschritt separat, sondern gemeinsam für mehrere Zeitschritte eines Teils der Sequenz. Das hat positive Auswirkungen auf die Robustheit und Genauigkeit der Parameterschätzung. Es konnten erste Untersuchungen mit Hilfe eines periodischen Modells der Bewegung erfolgen.

Weiterhin wird durch das periodische Bewegungsmodell eine automatische Initialisierung des Menschmodells möglich, weil eine gute Näherung der zu analysierenden Bewegung bereits als Bewegungsmodell vorhanden ist. Zur Bereitstellung der Näherungswerte müssen die Gelenkwinkel des Menschmodells für jeden Zeitschritt aus dem Bewegungsmodell abgerufen werden. Für jeden Zeitschritt kann dann zumindest näherungsweise der innere Zustand des Probanden vorgegeben werden. Nach einer guten Initialisierung des Modells kann nun die genaue Schätzung der Parameter mit dem Analyse-durch-Synthese-Ansatz erfolgen.

Die Arbeit widmet sich der Modellierung, der Korrespondenzfindung und der eigentlichen Parameterschätzung durch Ausgleichsrechnung. Neben dem Kameramodell wird das Modell des Skeletts, der Form und der Bewegung des Probanden detailliert beschrieben.

Das Modell des Probanden besteht aus dem Skelett und der Formbeschreibung. Es wurden zwei verschiedene Formulierungen für das Modell des Skeletts untersucht. Eine sehr einfache Formulierung basiert auf der Beschreibung durch Gelenkpunktkoordinaten direkt im Weltkoordinatensystem. Nebenbedingungen zwischen den Gelenkpunktkoordinaten sorgen für die Einhaltung der Körperproportionen. Dieses Modell zeichnet sich durch eine bemerkenswert einfache Implementierung aus. Allerdings besitzt es auch deutliche Einsatzgrenzen. Ein Nachteil des Modells ist, dass die Formbeschreibung auf Rotationskörper beschränkt bleibt und eine bessere Formbeschreibung weitergehende

Untersuchungen erfordert. Weiterhin ist es unmöglich anatomische Begrenzungen, wie Gelenkwinkelgrenzen mit wenig Aufwand zu berücksichtigen. Auch ein höheres Bewegungsmodell ist mit diesem Skelett nur schwer zu realisieren. Hinzu kommt die relativ große Anzahl freier Parameter, die sich wiederum nachteilig auf die Genauigkeit und Robustheit auswirkt.

Diese Nachteile motivierten die Entwicklung eines hierarchischen Skelettmodells. Es besteht aus sogenannten kinematischen Ketten, die aus der Robotik bekannt sind. Der Hauptvorteil ist die Formulierung von Gelenkwinkeln relativ zwischen zwei Körperteilen und die Entkopplung der (Translations-)Bewegung des Probanden von den Freiheitsgraden der Gelenke. Dies erlaubt die Einführung von höheren Bewegungsmodellen. Die direkte Repräsentation der Gelenkwinkel im Zustandsvektor erlaubt nun auch die Überprüfung von Gelenkwinkelgrenzen. Weiterhin definiert das Skelettmodell für jedes Körperteil ein eigenes Koordinatensystem und es entfällt die Beschränkung auf rotations-symmetrische Formbeschreibungen.

Verschiedene Methoden der Formbeschreibung werden vorgestellt. Am einfachsten ist die Verwendung von 2D-Konturtemplates. Sie können entweder direkt im Bild vorgegeben oder im 3D-Raum formuliert sein. Sie besitzen jedoch den Nachteil, dass sie keine Änderungen der Ansichtsrichtung zwischen Kamera und Körperteil repräsentieren können. Deshalb muss für jede Kamera ein Satz dieser Templates bereitgestellt werden und der Proband darf seine Orientierung zur Kamera während der Bewegung nicht wesentlich ändern.

Aus diesem Grund wurde auch der klassische Ansatz zur Formbeschreibung durch je einen Konus je Körperteil realisiert. Das ist ein verhältnismäßig einfaches Modell der Form, bei dem nur zwei Durchmesser, zur Vorgabe von Anfangswerten, vor der eigentlichen Messung näherungsweise bestimmt werden müssen. Dies wird normalerweise bereits mit einem Maßband möglich sein. Die Projektion eines jeden Konusses ins Bild kann in vereinfachter Weise erfolgen, da zur Parameterschätzung lediglich die beiden Begrenzungskanten des Konusses im Bild benötigt werden.

Die detaillierteste Formbeschreibung erlaubt ein Dreiecksnetz, das durch das Skelett deformiert wird. Dieses Konzept ist aus der Computeranimation bereits länger bekannt. Allerdings besitzt das Dreiecksnetz eine erhebliche Anzahl freier Parameter, die manuell nicht mehr bestimmbar sind. Weil auch während der Bildsequenzauswertung keine ausreichende Information zur Verfügung steht, müssten die Formparameter vor der eigentlichen Messung bestimmt werden. Das erfordert jedoch derzeit noch einen erheblichen Aufwand, der den Rahmen dieser Arbeit gesprengt hätte.

Die Bewegungsparameter des Menschmodells werden geschätzt, indem die gemessene Differenz zwischen Soll und Ist minimiert wird. Die Steuerung des hierarchischen Skeletts ist aber relativ kompliziert, weil der Zusammenhang zwischen Gelenkwinkeln und Modellpunkten im Bild nichtlinear ist. Als Ansatz dienen differenzielle Algorithmen der inversen Kinematik. Für die Bildauswertung war eine Anpassung dieser Algorithmen erforderlich. Auch dies wurde in dieser Arbeit realisiert. Im Gegensatz zur klassischen

## 12. Zusammenfassung

inversen Kinematik ist es nun möglich, Zielpunkte oder Zielkonturen auch in Kamerabildern anzugeben. Damit ist ein direkter Soll-Ist-Vergleich von Bestandteilen des Modells und Merkmalen im Bild möglich. Das erlaubt dann auch die Berechnung von Konfidenzintervallen für die zu schätzenden Bewegungsparameter. Die Aufstellung der Beobachtungsgleichungen erfolgt differenziell. Dafür werden Punkte des Modells ins Kamerabild projiziert und deren Lage nach den einzelnen Freiheitsgraden des Modells partiell abgeleitet. Dafür existieren aus der Literatur automatische Formalismen, die in dieser Arbeit noch einmal zusammengestellt sind. Liegen die Punkte des Modells im synthetischen Bild vor, müssen korrespondierende Punkte im realen Bild gefunden werden.

Da keine Marker zum Einsatz kommen sollen, müssen andere Methoden der Korrespondenzfindung verwendet werden. Die einfarbigen Beine hinterlassen im Bild gewöhnlich nicht viel mehr Information, als ihre Begrenzungskanten. Deshalb wird auf eine Berücksichtigung von Texturinformationen derzeit verzichtet. Die Suche nach Kantenpunkten im Bild erfolgt ausgehend von Kantenpunkten des Menschmodells. Damit dieses Verfahren funktioniert, müssen jedoch gute Näherungswerte vorhanden sein. Es wurden Methoden zu deren Bereitstellung diskutiert.

Im letzten Kapitel der Arbeit werden verschiedene Experimente dokumentiert. In einigen Versuchsreihen wird die Reproduzierbarkeit der Messergebnisse des Systems demonstriert. Es wird die Anwendbarkeit des Ansatzes für die Ganganalyse gezeigt.

# A. Details des Laufzyklus

## A.1. Standphase

Man unterscheidet Stand- und Schwungphase. Hat ein Bein Kontakt mit dem Boden (zum Beispiel Ferse, Sohle oder Zehen) befindet es sich in der Standphase. Diese lässt sich zunächst weiter unterteilen in:

- erste Doppelstandphase (First double support)
- Einzelstandphase (Single stance)
- zweite Doppelstandphase (Second double Support)

Nach der zweiten Doppelstandphase geht das Bein in die Schwungphase, es hat dann keinen Kontakt mit dem Boden und wird zum nächsten Standpunkt bewegt. Danach beginnt der ganze Laufzyklus erneut. Die Standphase nimmt etwa 60 % des Laufzyklus ein.

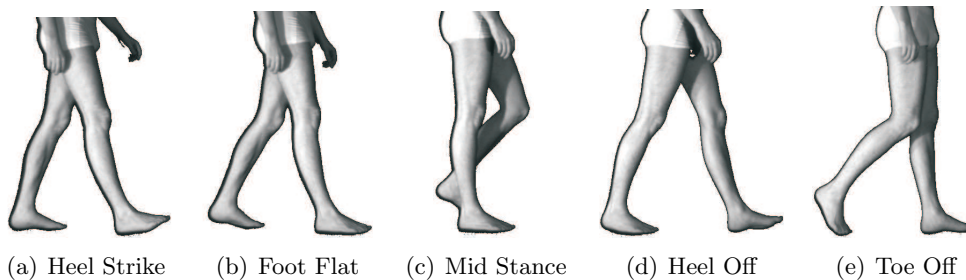


Abbildung A.1.: Elemente der Standphase

### A.1.1. Heel Strike - Initial Contact (Fersenkontakt)

Durch den ersten Kontakt der Ferse mit dem Boden am neuen Standpunkt wird nach der Schwungphase eines Beines ein neuer Laufzyklus eingeleitet. Das Knie ist weit durchgestreckt. Bezogen auf den gesamten Laufzyklus befindet sich der Kopf in der tiefsten Position. Dem Modell des inversen Pendels entsprechend ist die potenzielle Energie minimal. Damit besitzt die kinetische Energie ein Maximum, was sich in der höchsten horizontalen Geschwindigkeit ausdrückt.

## *A. Details des Laufzyklus*

Für die manuelle Bildauswertung ist das Ereignis Heel Strike interessant, weil es sich durch einfaches Betrachten zweier aufeinander folgender Bilder einer Sequenz zeitlich und örtlich eingrenzen lässt. Dadurch ist der Beginn der Standphase eindeutig gekennzeichnet. Für dieses Bein beginnt jetzt die erste Doppelstandphase. Das andere Bein trägt noch das gesamte Körpergewicht, aber seine Ferse hebt sich bereits. Aus Sicht des anderen Beines beginnt jetzt die zweite Doppelstandphase.

### **A.1.2. Foot Flat (Fuß flach)**

Nach dem ersten Kontakt wird der Fuß gestreckt, um ihn flach auf den Boden zu bringen. Jetzt wird auch ein großer Teil des Körpergewichtes auf diesen Fuß und damit den neuen Standpunkt übertragen. Damit entstehen bereits die ersten bremsenden Kräfte für den Oberkörper. Die Massenträgheit des Körpers schiebt ihn auf das Bein. Das Knie hat sich bereits leicht gebeugt und der Oberkörper beginnt aufzulaufen und sich leicht zu heben. Der Körperschwerpunkt wird bereits wieder angehoben. Dies ist ein Zeichen dafür, dass bereits kinetische Energie in potenzielle umgewandelt wird. Der projizierte Körperschwerpunkt bewegt sich in Richtung neuer Standpunkt. In den Bildern ist dieses Ereignis zeitlich nicht mehr ganz so deutlich zu identifizieren, weil es schwierig ist zu erkennen, wann genau der Fuß exakt flach auf dem Boden liegt. Die Zehen des anderen Beines haben gerade noch Kontakt mit dem Boden.

### **A.1.3. Mid Stance (Mittelstand-Phase)**

Ein Bein trägt das ganze Körpergewicht. Der Kopf hat die höchste Position im ganzen Laufzyklus, also ist die potenzielle Energie des Körpers maximal. Die Geschwindigkeit und damit die kinetische Energie des Körpers ist minimal. Das Lot des Körperschwerpunktes besitzt einen minimalen Abstand zum Fuß des Standbeines. Der Vorgang der Energieumwandlung kehrt sich ab jetzt wieder um und das Lot des Körperschwerpunktes entfernt sich vom Fuß des Standbeines. Das andere Bein befindet sich in der Schwungphase (Mid Swing). Außerdem treten in dieser Phase viele Selbstverdeckungen des von der Kamera entfernten Beines auf.

### **A.1.4. Heel Off (Ferse hebt ab)**

Beide Beine sind gestreckt. Das Knie des hinteren Beines fängt gerade an sich zu beugen. Weil sich der Körper bereits mit erhöhter Geschwindigkeit vom letzten Standpunkt wegbewegt und das Bein bereits gestreckt ist, hebt sich die Ferse an. Ein Großteil des Körpergewichtes liegt jedoch noch auf diesem Fuß und wird demzufolge vom Ballen und den Zehen getragen. Es beginnt das Abrollen über die Fußsohle und die Beschleunigung des Oberschenkels und des Knies. Das andere Bein befindet sich zwischen Initial Contact und Foot Flat. Das Ereignis Heel Off ist zeitlich eng begrenzt, deshalb ist es gut für die zeitliche Synchronisation des Modells mit den Bildsequenzen geeignet.

### A.1.5. Toe Off (Zehen heben ab)

Das Ende des Abrollens und damit das Ende der Standphase ist erreicht. Die Zehen verlassen gerade den Boden und das Bein geht in die Schwungphase. Im Bild lässt sich dieses Ereignis auf einen Zeitschritt genau eingrenzen. Das Knie ist bereits gebeugt, aber noch nicht maximal, der Oberschenkel/Knie ist bereits deutlich beschleunigt. Das andere Bein befindet sich zwischen Foot Flat und Mid Stance.

## A.2. Schwungphase

Die Schwungphase ist für die eigentliche Bewegung des Individuums nicht relevant. Sie ist aber notwendig, das Schwungbein zum nächsten Standpunkt zu bewegen. Für das Individuum ist es vorteilhaft, diesen Vorgang möglichst Energie sparend zu gestalten. Für die automatische Bildauswertung ist die Schwungphase etwas problematisch, weil hier die größten Geschwindigkeiten und Beschleunigungen auftreten, gleichzeitig aber auch viele Selbstverdeckungen der beiden Beine stattfinden. Außerdem besteht die größte Verwechslungsgefahr bei der Suche nach Kantenpunkten.

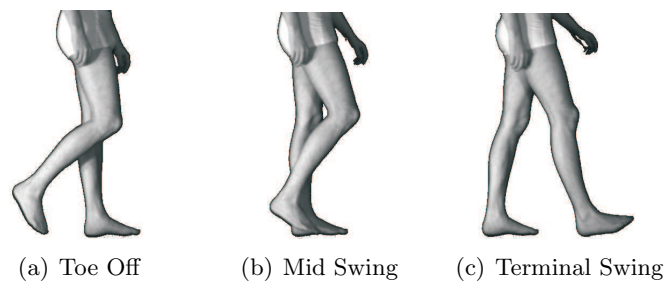


Abbildung A.2.: Elemente der Schwungphase

### A.2.1. Initial Swing - Acceleration (Beschleunigung)

Im Augenblick des Anhebens der Zehen ('Toe Off') wurde der Oberschenkel bereits durch Abstoßen des Fußes und passives Beugen des Knies deutlich beschleunigt. Dieser Vorgang wird durch Toe Off beendet. Durch Anheben des Fußes verlassen die Zehen den Boden und die Muskeln des Hüftbeugers beschleunigen das Bein vorwärts. Die Bewegungsenergie des Oberschenkels wird nun auch auf den Unterschenkel übertragen und so das ganze Bein in Richtung des neuen Standpunktes beschleunigt.

## *A. Details des Laufzyklus*

### **A.2.2. Mid Swing (Mittelschwung)**

In Mid Swing wird der Fuß direkt unter dem Körper hindurch bewegt. Es lässt sich ein genauer Zeitpunkt ermitteln, indem darauf geachtet wird, wann das Sprunggelenk das Standbein passiert. Der Kniewinkel wird zunächst nur wenig geändert. Der Hüftgelenkwinkel ändert sich mit näherungsweise konstanter Geschwindigkeit und das Bein wird zum neuen Standpunkt bewegt.

### **A.2.3. Bremsphase/Deceleration/Terminal Swing**

Die Bremsphase beschreibt die Muskelaktionen, die den Fuß abbremsen und stabilisieren, um den nächsten Heel Strike vorzubereiten. Für einen Moment ist die Fußsohle nahezu parallel zum Boden, sodass eigentlich nur noch ein Absenken des Fußes notwendig wäre. Heel Strike selbst beendet die Brems- und damit auch die Schwungphase.



## B. Lösung eines Gleichungssystems mit Hilfe der Singulärwertzerlegung

In nahezu allen Beispielen zur Modellapproximation wird zur Schätzung der unbekannt Parameter  $\mathbf{x}$  ein Gleichungssystem aufgestellt.

$$\mathbf{A} \cdot \mathbf{x} = \mathbf{y} \quad (\text{B.1})$$

Je nach Anzahl der Gleichungen, der Unbekannten und dem Vorhandensein von linearen Abhängigkeiten ist das Gleichungssystem überbestimmt, exakt bestimmt oder unterbestimmt. Exakt bestimmte Gleichungssysteme können mit dem klassischen Gauss-Jordan-Algorithmus [SK00] oder durch Berechnung der inversen Matrix gelöst werden.

$$\mathbf{x} = \mathbf{A}^{-1} \cdot \mathbf{y} \quad (\text{B.2})$$

Wenn die Matrix  $\mathbf{A}$  nur schwach besetzt (sparse) ist, kann LIBSPARSE [KSV86] zur effizienten Lösung verwendet werden. Überbestimmte Gleichungssysteme werden klassisch durch Ausgleichsrechnung mit Hilfe der Pseudoinversen gelöst:

$$\mathbf{x} = (\mathbf{A}^T \mathbf{A})^{-1} \cdot (\mathbf{A}^T \mathbf{y}) \quad (\text{B.3})$$

Da oft nicht abzusehen ist, wie viel Bildinformation und damit Gleichungen bei der Bildauswertung zur Verfügung stehen werden, wird immer mit der Pseudoinversen gerechnet. Das Produkt  $(\mathbf{A}^T \mathbf{A})$  ist eine quadratische Matrix mit so vielen Zeilen und Spalten, wie es Unbekannte gibt. Entsprechend ist der Term  $(\mathbf{A}^T \mathbf{y})$  ein Vektor der Länge  $n$ .

Jedoch kann die Matrix  $(\mathbf{A}^T \mathbf{A})$  singulär werden und lässt sich dann nicht mehr invertieren. Das kann passieren, wenn viele lineare Abhängigkeiten zwischen den Gleichungen bestehen. Die Berechnung der inversen Matrix kann dann mit Standardmethoden nicht mehr erfolgen und die Parameter können dann nicht mehr genau geschätzt werden, weil es theoretisch unendlich viele Lösungen für das Gleichungssystem gibt.

Dieser Fall kann durch die Berechnung des Ranges der quadratischen Matrix  $(\mathbf{A}^T \mathbf{A})$  erkannt werden. Bei den hier vorgestellten Methoden der Parameterschätzung durch Modellapproximation deutet ein solcher Rangabfall darauf hin, dass ein zu kompliziertes 3D-Modell mit mehr unbekannt Parameter verwendet wurde, als Bildinformation zur Verfügung steht. Mögliche Alternativen sind:

- vereinfachen des Modells, sodass weniger unbekannt Parameter berechnet werden müssen.

## B. Lösung eines Gleichungssystems mit Hilfe der Singulärwertzerlegung

- Bereitstellen von neuen Gleichungen (mit zusätzlichen Informationen oder Nebenbedingungen), die keine linearen Abhängigkeiten zu den bisherigen Gleichungen besitzen.
- Berechnung einer Lösung durch Addition eines Vielfachen der Einheitsmatrix.
- Verzicht auf eine hohe Genauigkeit und verwenden einer beliebigen der vielen Lösungen des unterbestimmten Gleichungssystems.

Mit der Singulärwertzerlegung (SVD, [Pre86] und [WR71]) ist ein leistungsfähiges Werkzeug zur Beurteilung der Lösbarkeit eines Gleichungssystems und der Berechnung der inversen Matrix [Boe97]. Implementierungen der Singulärwertzerlegung sind verfügbar (zum Beispiel Matlab oder in C++ mit der Bibliothek 'Newmat' sie verwendet die SVD nach [WR71]).

Die SVD kann zur Invertierung von Matrizen, selbst bei unterbestimmten Gleichungssystemen verwendet werden, indem sie eine quadratische Matrix  $\mathbf{N}$  in die drei Matrizen  $\mathbf{U}$ ,  $\mathbf{S}$  und  $\mathbf{V}^T$  zerlegt.

$$\mathbf{N} = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V}^T \quad (\text{B.4})$$

$\mathbf{U}$  und  $\mathbf{V}^T$  sind orthogonale Matrizen, deren Spaltenvektoren jeweils orthonormal sind. Deshalb ist die inverse Matrix gleich der transponierten und es gilt  $\mathbf{U}^{-1} = \mathbf{U}^T$  sowie  $(\mathbf{V}^T)^{-1} = \mathbf{V}$ .

$\mathbf{S}$  ist eine Diagonalmatrix und enthält absteigend sortiert die Singulärwerte  $s_1$  bis  $s_n$  von  $\mathbf{N}$ . Für den Spezialfall das  $\mathbf{N} = \mathbf{A}^T \mathbf{A}$  gilt  $\mathbf{U} = \mathbf{V}$  und es können die Eigenwerte  $\lambda_i$  der Matrix  $\mathbf{N}$  aus den Singulärwerten  $s_i$  berechnet werden

$$\lambda_i = s_i^2 \quad (\text{B.5})$$

Mit Hilfe der Matrix  $\mathbf{S}$  ist die Berechnung des Ranges<sup>1</sup> und der Konditionszahl des Gleichungssystems möglich. [Mat96]:

Die Konditionszahl  $\text{cond}(\mathbf{X})$  (ist auch eine eingebaute Funktion) ist der Quotient aus größtem und kleinstem Singularwert: Diese beiden Größenordnungen werden durch  $\mathbf{U}$  und  $\mathbf{V}$  gemischt, und bei endlicher Genauigkeit wie bei Gleitkommazahlen ist dann oft die Inversion nicht-singulärer Matrizen nicht mehr möglich. Konditionszahlen von größer als  $10^3$  sind meist schon bedenklich, es sei denn, die Matrix ist diagonal dominant, d.h. die größten Elemente stehen sowieso auf der Diagonale.

Zur Invertierung einer Diagonalmatrix müssen lediglich ihre Diagonalelemente invertiert werden.

$$\mathbf{S}^{-1} = \begin{pmatrix} s_1 & & 0 \\ & \ddots & \\ 0 & & s_n \end{pmatrix}^{-1} = \begin{pmatrix} \frac{1}{s_1} & & 0 \\ & \ddots & \\ 0 & & \frac{1}{s_n} \end{pmatrix} \quad (\text{B.6})$$

---

<sup>1</sup> Der Rang ist die Anzahl der Singulärwerte, die größer als ein vorgegebener Schwellwert  $tol$  sind. Beispielsweise verwendet Matlab für die Funktion  $\text{rank}()$ :  $tol = n \cdot s_1 \cdot 2.2204 \cdot 10^{-16}$

Ist das Gleichungssystem unterbestimmt, dann werden die letzten Elemente  $s_n$  sehr klein. Es würde also zu einer Division durch null bei der Berechnung der letzten  $\frac{1}{s_n}$  kommen. Diese Elemente werden dann  $\frac{1}{s_n} := 0$  gesetzt. Damit ist jetzt auch die Invertierung der Matrix  $\mathbf{N}$  im Falle eines unterbestimmten Gleichungssystems möglich (Pseudoinverse).

Durch Umstellen von B.4 ist ersichtlich, wie aus den Ergebnissen der Singulärwertzerlegung die inverse Matrix  $\mathbf{A}^{-1}$  berechnet werden kann.

$$\mathbf{A} \cdot \mathbf{A}^{-1} = \mathbf{I} \quad (\text{Einheitsmatrix}) \quad (\text{B.7})$$

$$\mathbf{U}\mathbf{W}\mathbf{V}^T \cdot \mathbf{A}^{-1} = \mathbf{I} \quad (\text{B.8})$$

$$\mathbf{W}\mathbf{V}^T \cdot \mathbf{A}^{-1} = \mathbf{U}^T \quad (\text{es gilt: } \mathbf{U}^{-1} = \mathbf{U}^T) \quad (\text{B.9})$$

$$\mathbf{V}^T \cdot \mathbf{A}^{-1} = \mathbf{W}^{-1}\mathbf{U}^T \quad (\text{B.10})$$

$$\mathbf{A}^{-1} = \mathbf{V}\mathbf{W}^{-1}\mathbf{U}^T \quad (\text{B.11})$$

Eine Lösung des unterbestimmten Gleichungssystems lässt sich auch erhalten, indem ein Vielfaches  $\mu$  der Einheitsmatrix  $\mathbf{I}$  zur Matrix  $(\mathbf{A}^T\mathbf{A})$  addiert wird, dadurch lässt sich ebenfalls eine inverse Matrix berechnen.

$$\mathbf{x} = (\mathbf{A}^T\mathbf{A} + \mu\mathbf{I})^{-1} \cdot (\mathbf{A}^T\mathbf{y}) \quad (\text{B.12})$$

Bezogen auf die in dieser Arbeit vorgestellten differenziellen Approximationsalgorithmen wirkt die Einheitsmatrix auf das Mensch-Modell wie 'mechanische Bremsen' in den Gelenken. Eine stärkere Gewichtung der Einheitsmatrix durch Erhöhen von  $\mu$  hält die Änderungen in den Gelenkwinkeln je Iterationsschritt klein.

Dieser Ansatz ist ähnlich dem *Levenberg-Marquardt-Verfahren*, bei dem das  $\mu$  in Abhängigkeit vom Ergebnis verändert wird. Wurde ein besseres Ergebnis erhalten, wird  $\mu$  um einen Faktor (zum Beispiel 10) verkleinert und mit der nächsten Iteration begonnen. Wurde ein schlechteres Ergebnis erhalten, wird  $\mu$  um einen Faktor (zum Beispiel 10) vergrößert und es muss die selbe Iteration wiederholt werden, indem Gl.(B.12) noch einmal berechnet wird.

*B. Lösung eines Gleichungssystems mit Hilfe der Singulärwertzerlegung*



### C. Standardabweichungen der Unbekannten

$$p_i = \frac{\sigma_0^2}{\sigma_i^2} \quad : \quad \text{Gewicht einer Beobachtung } i \quad (\text{C.4})$$

$$\begin{aligned}
 (1) \quad & \mathbf{P} = \mathbf{Q}_{ll}^{-1} && : \text{ Gewichtsmatrix} \\
 (2) \quad & \mathbf{N} \cdot \hat{\mathbf{x}} - \mathbf{n} = 0 && : \text{ Normalgleichungen} \\
 & \mathbf{N} = \mathbf{A}^T \cdot \mathbf{P} \cdot \mathbf{A} && : \text{ Normalgleichungsmatrix} \\
 & \mathbf{n} = \mathbf{A}^T \cdot \mathbf{P} \cdot \mathbf{l} && : \text{ Absolutglied} \\
 (3) \quad & \mathbf{Q} = \mathbf{N}^{-1} && : \text{ Auflösen der Normalgleichungen} \\
 & \hat{\mathbf{x}} = \mathbf{Q} \cdot \mathbf{n} && \\
 & = (\mathbf{A}^T \cdot \mathbf{P} \cdot \mathbf{A})^{-1} \cdot \mathbf{A}^T \cdot \mathbf{P} \cdot \mathbf{l} && : \text{ mit } \mathbf{Q}: \text{ Kofaktormatrix der Unbekannten} \\
 (4) \quad & \mathbf{v} = \mathbf{A} \cdot \hat{\mathbf{x}} - \mathbf{l} && : \text{ Verbesserungen} \\
 (5) \quad & \hat{\mathbf{l}} = \mathbf{l} + \mathbf{v} && : \text{ ausgeglichene Beobachtungen} \\
 (6) \quad & \hat{\mathbf{L}} = \mathbf{L} + \mathbf{v} && \\
 & \hat{\mathbf{X}} = \mathbf{X}^0 + \hat{\mathbf{x}} && \\
 (7) \quad & \hat{\sigma}_0 = \sqrt{\frac{\mathbf{v}^T \cdot \mathbf{P} \cdot \mathbf{v}}{n-u}} && : \text{ Standardabweichung a posteriori} \\
 (8) \quad & \mathbf{K} = \hat{\sigma}_0^2 \cdot \mathbf{Q} && \\
 (9) \quad & \hat{\mathbf{L}} = \varphi(\hat{\mathbf{X}}) && : \text{ Schlussprobe}
 \end{aligned} \quad (\text{C.5})$$

Liegen für nichtlineare Problemstellungen (zum Beispiel Bündelblockausgleichung) lediglich grobe Näherungswerte der Unbekannten vor, lässt sich das Ausgleichungssystem nur iterativ lösen. Dabei werden die in Schritt (6) einer Iteration  $k$  verbesserten Näherungswerte solange wiederum als Startwerte für das linearisierte funktionale Modell der nächsten Iteration  $k+1$  eingesetzt, bis die Summe der Unbekanntenzuschläge einen Grenzwert unterschreitet:

$$\mathbf{X}_{k+1}^0 = \mathbf{X}_k^0 + \hat{\mathbf{x}}_k \quad (\text{C.6})$$

Die in Schritt (3) durchgeführte Auflösung des unter (2) aufgestellten Normalgleichungssystems setzt eine spaltenreguläre Designmatrix  $\mathbf{A}$  voraus, d.h.:

$$r = \text{rang}(A) = u \quad (\text{C.7})$$

Dies bedeutet, dass die eingehenden Beobachtungen eine eindeutige Lösung des Unbekanntenvektors zulassen und die Inverse der Normalgleichungsmatrix  $\mathbf{N}$  existiert. Ausgleichungsprobleme, in denen einige Beobachtungen zur eindeutigen Bestimmung der Unbekannten fehlen, besitzen einen Rangdefekt  $d$  mit

$$d = u - r \quad (\text{C.8})$$

(Zitat Ende)

Die Standardabweichung  $\hat{\sigma}_i$  einer Unbekannten mit dem Index  $i$  berechnet sich aus der a posteriori Standardabweichung und dem Diagonalelement  $q_{ii}$  der Kofaktormatrix  $\mathbf{Q}$

$$\hat{\sigma}_i = \hat{\sigma}_0 \cdot \sqrt{q_{ii}} \quad (\text{C.9})$$

Die a posteriori Standardabweichung  $\hat{\sigma}_0$  wird in der nächsten Iteration als a priori Standardabweichung  $\sigma_0$  verwendet.

*C. Standardabweichungen der Unbekannten*



## D. Homogene Transformationsmatrizen

Im Abschnitt 3.3 wird ein Skelettmodell basierend auf kinematischen Ketten vorgestellt. Das Aufstellen der Ketten erfordert die Bereitstellung der elementaren Transformationsmatrizen. Deshalb sind hier die homogenen Transformationsmatrizen und deren Ableitungen wiedergegeben. Die hier aufgeführten Matrizen sind vollständig aus [Wac97] entnommen. Winkel in den Rotationsfreiheitsgraden müssen in Radiant angegeben sein.

### D.1. Elementare Rotationsmatrizen

Operation	Matrix $\mathbf{E}(\alpha)$	Ableitung $\frac{\partial \mathbf{E}(\alpha)}{\partial \alpha}$
$\mathbf{E}_{rx}$ (Rotation X-Achse)	$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) & 0 \\ 0 & \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & -\sin(\alpha) & -\cos(\alpha) & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$
$\mathbf{E}_{ry}$ (Rotation Y-Achse)	$\begin{pmatrix} \cos(\alpha) & 0 & \sin(\alpha) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\alpha) & 0 & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} -\sin(\alpha) & 0 & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 0 \\ -\cos(\alpha) & 0 & -\sin(\alpha) & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$
$\mathbf{E}_{rz}$ (Rotation Z-Achse)	$\begin{pmatrix} \cos(\alpha) & -\sin(\alpha) & 0 & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} -\sin(\alpha) & -\cos(\alpha) & 0 & 0 \\ \cos(\alpha) & -\sin(\alpha) & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$

## D.2. Inverse elementare Rotationsmatrizen

Die inversen Matrizen ergeben sich aus der transponierten Matrix.

Operation	Matrix $(\mathbf{E}(\alpha))^{-1}$	Ableitung $\frac{\partial(\mathbf{E}(\alpha))^{-1}}{\partial\alpha}$
$(\mathbf{E}_{rx})^{-1}$ (Rotation X-Achse) <sup>T</sup>	$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha) & \sin(\alpha) & 0 \\ 0 & -\sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & -\sin(\alpha) & \cos(\alpha) & 0 \\ 0 & -\cos(\alpha) & -\sin(\alpha) & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$
$(\mathbf{E}_{ry})^{-1}$ (Rotation Y-Achse) <sup>T</sup>	$\begin{pmatrix} \cos(\alpha) & 0 & -\sin(\alpha) & 0 \\ 0 & 1 & 0 & 0 \\ \sin(\alpha) & 0 & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} -\sin(\alpha) & 0 & -\cos(\alpha) & 0 \\ 0 & 0 & 0 & 0 \\ \cos(\alpha) & 0 & -\sin(\alpha) & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$
$(\mathbf{E}_{rz})^{-1}$ (Rotation Z-Achse) <sup>T</sup>	$\begin{pmatrix} \cos(\alpha) & \sin(\alpha) & 0 & 0 \\ -\sin(\alpha) & \cos(\alpha) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} -\sin(\alpha) & \cos(\alpha) & 0 & 0 \\ -\cos(\alpha) & -\sin(\alpha) & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$

## D.3. Elementare Translationsmatrizen

Operation	Matrix $\mathbf{E}(d)$	Ableitung $\frac{\partial\mathbf{E}(d)}{\partial d}$
$\mathbf{E}_{tx}$ (Translation X-Achse)	$\begin{pmatrix} 1 & 0 & 0 & d \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$
$\mathbf{E}_{ty}$ (Translation Y-Achse)	$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$
$\mathbf{E}_{tz}$ (Translation Z-Achse)	$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$

**D.4. Inverse elementare Translationsmatrizen**

Operation	Matrix $(\mathbf{E}(d))^{-1}$	Ableitung $\frac{\partial(\mathbf{E}(d))^{-1}}{\partial d}$
$(\mathbf{E}_{tx})^{-1}$	$\begin{pmatrix} 1 & 0 & 0 & -d \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$
$(\mathbf{E}_{ty})^{-1}$	$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -d \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$
$(\mathbf{E}_{tz})^{-1}$	$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -d \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$

*D. Homogene Transformationsmatrizen*

## E. Automatische Ableitung kinematischer Ketten

Im Abschnitt 3.3 wurde ein Skelettmodell basierend auf kinematischen Ketten vorgestellt. Zur Berechnung der partiellen Ableitungen wurde dort aber noch nicht viel geschrieben. Das wird hier nachgeholt.

Um einen Punkt  $\mathbf{l}_a$  von lokalen Körperteilkoordinaten  $a$  in Weltkoordinaten  $\mathbf{w}$  zu transformieren, wird Gl.(3.23) aus Abschnitt 3.4 verwendet.:

$$\mathbf{w}(\mathbf{x}) = \mathbf{T}_{0a}(\mathbf{x}) \cdot \mathbf{l}_a = \mathbf{E}_1(\mathbf{x}) \cdot \dots \cdot \mathbf{E}_p(\mathbf{x}) \cdot \mathbf{l}_a \quad (\text{E.1})$$

Die Transformationskette  $\mathbf{T}_{0a}(\mathbf{x})$  wird aus den Elementarmatrizen  $\mathbf{E}_1(\mathbf{x}) \cdot \dots \cdot \mathbf{E}_p(\mathbf{x})$  multiplikativ zusammengesetzt. Für die Approximation des Modells werden die Ableitungen der Punktkoordinaten  $\mathbf{w}(\mathbf{x})$  nach den Komponenten  $x_i$  des Zustandsvektors  $\mathbf{x} = [x_1, x_2, \dots, x_i, \dots, x_n]^T$  benötigt. Dabei ist  $\frac{\partial \mathbf{w}(\mathbf{x})}{\partial x_i}$  ein Vektor, der in die Richtung zeigt, in die sich  $\mathbf{w}$  bewegen würde, wenn  $x_i$  ein wenig geändert würde. Die kinematischen Ketten müssen so konstruiert sein, dass eine eindeutige Zuordnung zwischen einem Element  $x_i$  und seiner Elementarmatrix  $\mathbf{E}_q(x_i)$  existiert. Das bedeutet: keine der anderen Elementarmatrizen (außer  $\mathbf{E}_q(x_i)$ ) darf von  $x_i$  abhängig sein.

$$\frac{\partial \mathbf{w}(\mathbf{x})}{\partial x_i} = \mathbf{E}_1(\mathbf{x}) \cdot \dots \cdot \mathbf{E}_{q-1}(\mathbf{x}) \cdot \frac{\partial \mathbf{E}_q(x_i)}{\partial x_i} \cdot \mathbf{E}_{q+1}(\mathbf{x}) \cdot \dots \cdot \mathbf{E}_p \cdot \mathbf{l}_a \quad (\text{E.2})$$

Zur Vollständigkeit der Darstellung werden noch die inversen Beziehungen angegeben. Sie werden eigentlich nur zur Kamerakalibrierung benötigt. Der Ausdruck  $(\mathbf{x})$  für die Abhängigkeit vom Zustandsvektor wurde aus Platzgründen weggelassen:

$$\mathbf{l}_a = \mathbf{T}_{0a}^{-1} \cdot \mathbf{w} = \mathbf{E}_p^{-1} \cdot \dots \cdot \mathbf{E}_1^{-1} \cdot \mathbf{w} \quad (\text{E.3})$$

$$\frac{\partial \mathbf{l}_a}{\partial x_i} = \mathbf{E}_p^{-1} \cdot \dots \cdot \mathbf{E}_{q+1}^{-1} \cdot \frac{\partial (\mathbf{E}_q(x_i))^{-1}}{\partial x_i} \cdot \mathbf{E}_{q-1}^{-1} \cdot \dots \cdot \mathbf{E}_1^{-1} \cdot \mathbf{w} \quad (\text{E.4})$$

Zum automatischen Aufstellen der Ableitungsmatrizen (loc2dif) könnte folgender C++ Code verwendet werden (frei nach[Wac97]):

## E. Automatische Ableitung kinematischer Ketten

```
// M4X4: Klasse f. homogene Transformationsmatrizen (Dim 4x4)
class BONE //Repraesentation eines Koerperteiles (nicht vollstaendig)
{
public:
    M4X4    E; //    aktuelle Elementarmatrix des Koerperteils
    M4X4    dE; // abgeleitete Elementarmatrix des Koerperteils
    int     dof; // Freiheitsgrad (Spalte im Gleichungssystem)
    BONE *  pre; // Vorgaenger des Koerperteils
    ableiten(M4X4 & Links , M4X4 Rechts);
};

M4X4 loc2dif[MAXBONES]; // Koerperteilkoordinaten in dw/dx
int Spalte[MAXBONES]; // Spalten der Elemente(!=0) im Gl.syst
int nonzero; //Anzahl Spalten der Elemente(!=0) im Gl.syst

// Links : Produkt E1 * E2 * E3 * ...
// Rechts: Produkt ... * E(P-1) * EP
void BONE::ableiten(D4X4 & Links , D4X4 Rechts)
{ if(pre) pre->ableiten( Links , E * Rechts);
  loc2dif[nonzero]= Links * dE * Rechts ;
  Spalte[nonzero]= dof;
  nonzero++;
  Links = Links * E;
}

void PrintDiff(void)
{ BONE Skelett[MAXBONES]; // Skelettmodell
  Load(Skelett , "Human.txt"); // Modell laden
  BONE * fuss = Find(Skelett , "Fuss"); // Koerperteil suchen

  nonzero=0; // noch keine Elemente(!=0) der Gleichung
  M4X4 Links(IDENTITY);
  M4X4 Rechts(IDENTITY);
  fuss->ableiten(Links , Rechts);

  // Punkt(la) in lokalen Koerperteilkoordinaten(a)
  P3D la(20.0 , 43.7 , 17.4);
  for(int i=0;i<nonzero;i++)//jeder Freiheitsgrad des Fusses
  { // Berechnung des Differenzialvektors dw/dx
    P3D d = loc2dif[i] * la;
    printf("\nSpalte=%i:_dx=%lf_dy=%lf_dz=%lf",
           Spalte[i] , d.x ,d.y ,d.z);
  }
}
```

## F. Numerische Berechnung der Ableitungen

In dieser Arbeit kommen häufig nichtlineare Modelle der Form  $\mathbf{f}(\mathbf{x})$  zum Einsatz. Der übliche Ansatz (siehe Abschnitt 2.2.3) zur Parameterschätzung des Vektors  $\mathbf{x}$  ist eine Linearisierung des Modells in einem gegebenen Arbeitspunkt mit einer Taylorreihe und der anschließenden Lösung eines möglichst überbestimmten Gleichungssystems.

Zur Linearisierung werden jedoch die partiellen Ableitungen des Modells benötigt, deren analytische Berechnung häufig kompliziert ist und sehr schnell unübersichtlich wird.

In [Luh00] wird darauf hingewiesen, dass sich diese Differenziale auch numerisch berechnen lassen. Dazu werden nacheinander kleine Zuschläge  $\Delta x_j$  auf die Zustandsvektor-Komponente  $x_j$  des Arbeitspunktes  $\mathbf{x}_0$  addiert und subtrahiert. Es entstehen zwei neue Zustandsvektoren.

$$\mathbf{x}_+ = \mathbf{x}_0 + \Delta \mathbf{x} = [x_0, x_1, \dots, x_j + \Delta x_j, \dots, x_n]^T \quad (\text{F.1})$$

$$\mathbf{x}_- = \mathbf{x}_0 - \Delta \mathbf{x} = [x_0, x_1, \dots, x_j - \Delta x_j, \dots, x_n]^T \quad (\text{F.2})$$

Anschließend kann der Differenzenquotient für den Freiheitsgrad  $x_j$  numerisch berechnet werden.

$$\frac{\partial \mathbf{f}(\mathbf{x}_0)}{\partial x_j} \approx \frac{\mathbf{f}(\mathbf{x}_+) - \mathbf{f}(\mathbf{x}_-)}{2\Delta x_j} \quad (\text{F.3})$$

Bei der Wahl von  $\Delta x_j$  muss etwas experimentiert werden. Durch zu große Zahlenwerte wird das Ergebnis ungenau, weil  $\mathbf{f}(\mathbf{x})$  nichtlinear ist. Bei zu kleinen Zahlenwerten kann es passieren, dass die Funktionswerte  $\mathbf{f}(\mathbf{x}_+)$  und  $\mathbf{f}(\mathbf{x}_-)$  numerisch nicht genau genug berechnet werden und dadurch das Ergebnis verfälscht wird.

Dieses Verfahren ist leicht zu implementieren und die Funktion  $\mathbf{f}(\mathbf{x})$  kann als Black-Box betrachtet werden, d.h. Kenntnis über die genaue innere Struktur von  $\mathbf{f}(\mathbf{x})$  ist für die Berechnung der Differenziale nicht mehr notwendig.

## *F. Numerische Berechnung der Ableitungen*



## G. Kamerakalibrierung

Vor der Bildauswertung müssen die Parameter des in Kapitel 5 beschriebenen Kamera-modells für jede Kamera bestimmt werden. Es sind verschiedene Strategien zur Kamerakalibrierung bekannt. In [LZN02] wurde ein Verfahren zur Kalibrierung einer Kamera anhand eines gehenden Menschen, vorgestellt. Allerdings arbeitet das Verfahren nur mit erheblichen Abstand der Kamera zum Probanden, wie es bei normalen Verkehrsszenen der Fall ist. Für eine Anwendung im Nahbereich ist dieses Verfahren ungeeignet, deshalb wurde der klassische Bündelblockausgleich verwendet. Sehr übersichtlich ist dies in [Luh00] erläutert.



Abbildung G.1.: Antasten von Gitterpunkten zur Kalibrierung

Es besteht die Aufgabe, alle beteiligten Kameras in ein gemeinsames Weltkoordinatensystem zu bringen. Dazu ist zunächst eine Datumsfestlegung notwendig, d.h. es wird festgelegt, wo sich der Koordinatenursprung befindet und in welche Richtung die Koordinatenachsen zeigen. Die Kameras sollen möglichst nicht erst paarweise kalibriert werden, um dann die einzelnen Kamera-paare über eine Transformation zu einer Kameraanordnung zusammensetzen. Das führt oft dazu, dass sich mehrere kleine Fehler der relativen Transformationen zu einem großen absoluten Fehler am Ende der Transformationskette aufsummieren. Weiterhin hätte dies den Nachteil, dass das Weltkoordi-

## G. Kamerakalibrierung

natensystem willkürlich durch ein ausgewähltes Kamerapaar festgelegt würde. Für eine einfachere Interpretation der Messergebnisse ist es aber sehr wünschenswert, das Weltkoordinatensystem definiert an den Laufbereich zu knüpfen. So fällt die  $xy$ -Ebene des Koordinatensystems mit dem Boden zusammen und die  $z$ -Achse zeigt nach oben. Aus diesem Grunde ist es besser, alle Kameras gleich von Anfang an in einem überbestimmten Gleichungssystem auszugleichen. Dabei sind die Genauigkeitsanforderungen derzeit nicht besonders hoch (2-3 Pixel), da zum einen das derzeit verwendete Körpermodell nicht sehr präzise ist und zum anderen meist relative Größen wie:

- relative Gelenkwinkel (und ihre Ableitungen) zwischen Körperteilen
- Geschwindigkeit, als relative Größe zwischen zwei absoluten Raumpunkten
- Beschleunigungen

interessieren. Die einzige interessante absolute Größe wäre wohl die Höhe des Fußes über dem Boden (Überprüfung auf Durchdringung).

Durch die genannte Datumsfestlegung lässt sich das Problem der Singularitäten der Rotationsmatrizen sowohl für die Kameras als auch für das Mensch-Modell elegant umgehen. Denn die Kameras schauen nahezu horizontal und nicht vertikal von der Decke. Legt man die  $xy$ -Achse auf den Boden, ist auch die Bereitstellung von Näherungswerten für die Kamerapositionen relativ einfach mit einem Maßband möglich. Im speziellen Fall war sogar ein Kachelfußboden mit einem Gitterabstand von 250 mm vorhanden. Durch Kennzeichnen und Abzählen der Gitterlinien ist die Orientierung im 'Weltkoordinatensystem' sehr einfach. Die Blickwinkel der Kameras kann man grob schätzen.

Das Problem ist nun die Gewinnung einer ausreichenden Anzahl von Beobachtungsgleichungen, um die unbekannt Parameter zu berechnen. Wird ein Punkt im Raum markiert, so werden in jeder Kamera, die diesen Punkt sieht, zwei Koordinaten gemessen. Deshalb lassen sich für jede dieser Kameras zwei Gleichungen aufstellen. Doch wie verhält sich die Anzahl der Unbekannten? Nach dem zuvor beschriebenen Kameramodell besitzt jede Kamera 6 unbekannte Parameter der äußeren Orientierung und 2 bis 6 unbekannte Parameter der inneren Orientierung.

Jeder Raumpunkt wird durch 3 Koordinaten beschrieben. Je nachdem, wie viele Koordinaten des Punktes unbekannt sind, unterscheidet man:

- Passpunkte, alle Koordinaten  $XYZ$  bekannt
- Lagepasspunkt  $XY$  bekannt
- Höhenpasspunkt  $Z$  bekannt
- Verknüpfungspunkte, keine der Koordinaten bekannt

Eine möglichst hohe Redundanz  $r$  ist wünschenswert. Die Redundanz des Gleichungssystems berechnet sich aus der Anzahl der unbekannt Parameter  $u$  und der Anzahl der

Gleichungen  $n$  zu:

$$r = n - u \quad (\text{G.1})$$

Jeder Verknüpfungspunkt erhöht die Anzahl der Unbekannten um 3. Von jeder Kamera, in der der Punkt gemessen werden konnte, werden 2 Gleichungen bereitgestellt. Bereits ab zwei Kameras ( $2 \cdot 2 = 4$  Gleichungen) erhöht ein Verknüpfungspunkt die Redundanz des Gesamtsystems um eins und es lohnt sich, ihn zu berücksichtigen. Mit jeder weiteren Kamera erhöht sich die Redundanz für diesen Punkt um zwei. Schlussfolgerung: Im Idealfall werden Verknüpfungspunkte von allen beteiligten Kamera gesehen.

Klassische Instrumente zur Markierung eines Punktes im Raum sind Kreismarken. Diese lassen sich im Bild weitgehend automatisch und mit hoher Genauigkeit lokalisieren. Werden sie mit einer Codierung versehen, so ist auch eine automatische Zuordnung zwischen gemessenem Bildpunkt und Raumpunkt möglich. Allerdings können Kreismarken nur aus einem eingeschränkten Blickwinkel mit hoher Genauigkeit gemessen werden. Im vorherigen Abschnitt wurde aber erläutert, weshalb die gleichzeitige Beobachtung ein und desselben Raumpunktes günstig ist. Die Verwendung einer Kugel anstatt einer Kreismarke würde dieses Problem lösen. Möchte man jedoch direkt die Gitterpunkte des Bodens antasten, so führt der Radius der Kugel zwangsläufig zu einem Offset, der korrigiert werden muss.



Abbildung G.2.: LED-Tastkopf

Die Verwendung einer Punktlichtquelle wird aufgrund der verhältnismäßig schlechten Genauigkeit bei der Lokalisierung im Bild relativ selten in Betracht gezogen. Im Gegensatz zur kugelförmigen Markern lassen sich aber Punkte auf dem Boden fast gänzlich ohne Offset markieren. Als Punktlichtquelle wurde nach einigen Versuchen eine sehr helle LED verwendet, die weißes Licht emittiert. Dabei wurde auf einen sehr großen Abstrahlwinkel geachtet. Mittlerweile sind sehr kleine Bauformen als SMD-LED's verfügbar. Sie

## G. Kamerakalibrierung

lassen sich an den Rückseiten verbinden. Das ganze Paket ist im Durchmesser ca. 2mm. Dies ist wenig, wenn man bedenkt, dass bei der gegebenen Anwendung ein Pixel im Objektraum ca. 5 mm breit ist.

Eine automatische Lokalisierung des Lichtpunktes in allen Kamerabildern erscheint möglich, wenn der Hintergrund der LED nicht das Maximum des Dynamikbereiches der Kameras verlässt. Meist haben Wände jedoch einen weißen Anstrich. Das Problem der Hintergrundbeleuchtung lässt sich durch folgende Maßnahmen lösen:

1. Hintergrund oder ganzen Raum abdunkeln (Beleuchtung, Vorhänge)
2. Verkürzen der Shutter-Zeit der Kameras, bis nur noch die LED zu sehen ist
3. LED Flussstrom zeitlich codieren und aufzeichnen einer Bildfolge, in der Bildfolge wird dann nach Pixeln mit der selben zeitlichen Codierung gesucht.

Es wäre auch denkbar, die Blende des Objektivs soweit zu schließen, bis lediglich die leuchtende LED im Bild sichtbar ist. Davon muss jedoch abgeraten werden, weil nach der Kalibrierung die Blende zur normalen Bildaufnahme geöffnet werden müsste. Erfahrungsgemäß ändert sich dadurch aber auch Lage, Orientierung und Hauptpunkt der Kameras. Aus dem selben Grund sollten auch keine automatisch fokussierenden/zoomenden Objektive verwendet werden. Die Shutter-Zeit lässt sich bei modernen Kameras über ein serielles Protokoll ändern, so bleibt die Kamera selbst unangetastet und es ist zu erwarten, dass der Einfluss auf die berechneten Parameter des Kameramodells gering ist.

Punkt 3 ist aus zwei Gründen innovativ: Zum einen lässt sich der Flussstrom einer LED im Pulsbetrieb und damit die Leuchtdichte erhöhen. Zum anderen erlaubt eine zeitliche Codierung eine deutlich robustere Suche nach der LED im Bild. Voraussetzung dafür ist jedoch, dass sich die Lage der LED während der Bildaufnahme nicht signifikant ändert. Bei dem implementierten System steht das Triggersignal der Kameras zur Verfügung, um die zeitliche Codierung der LED mit der Bildaufnahme zu synchronisieren. Es reicht ein sehr einfacher Code aus, dabei wird die LED in den geraden Zeitschritten der Sequenz gepulst und in den ungeraden Zeitschritten bleibt die LED aus. Eine zeitliche Bandpassfilterung eines jeden Pixels  $g(t, x, y)$  der synchron aufgezeichneten Bildfolge leistet eine zuverlässige Unterdrückung aller Pixel, die nicht die LED zeigen. Es ist dann sehr einfach nach dem Pixel  $(x_m, y_m)$  mit dem betragsmäßig größtem Bandpassfilterausgang zu suchen.

Für die Bandpassfilterung wurde folgende Gleichung verwendet:

$$F(x, y) = \sum_{t=0}^T g(t, x, y) \cos(i \cdot \pi) \quad (\text{G.2})$$

Bis auf einen Skalierungsfaktor kann, Gl.(G.2) als Realteil eines Koeffizienten einer Fast-Fourier-Transformation (FFT) oder eines Koeffizienten einer Diskreten-Cosinus-Transformation (DCT) aufgefasst werden. Demnach wird jedes gerade Bild mit +1 mul-

tipliziert und jedes ungerade Bild mit  $-1$ . Das Ergebnis wird in einem Bild  $F(x, y)$  aufsummiert. Dafür ist ein geeigneter Datentyp zu wählen (zum Beispiel unsigned long). Bereits mit einer kurzen Bildsequenz mit  $T = 10$  Zeitschritten wurden sehr gute Ergebnisse erzielt.

Wird das Vorzeichen von  $F(x_m, y_m)$  in jedem Kamerabild betrachtet, ist es möglich, Probleme der Synchronisierung der Kamerakanäle bereits während der Kalibrierung zu erkennen. Läuft die Aufnahme zum Beispiel in einem Kamerakanal um einen Zeitschritt versetzt, so wechselt das Vorzeichen von  $F(x_m, y_m)$ . Laufen alle Kamerakanäle synchron, so muss das Vorzeichen von  $F(x_m, y_m)$  für alle Kanäle gleich sein. Ein zeitlicher Versatz eines oder mehrerer Kamerakanäle kann meist bereits an unterschiedlichen Vorzeichen erkannt werden.

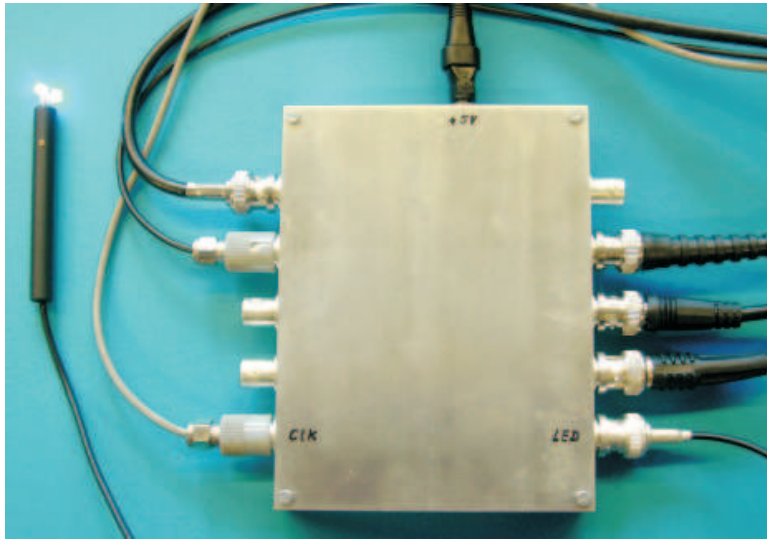


Abbildung G.3.: Steuereinheit zur Synchronisierung aller Kameras und zur Ansteuerung der LED, die links im Bild zu sehen ist

Im Prinzip besteht die Möglichkeit, ein ausgedehntes Passpunktfeld aus robusten Profilen aufzubauen. Es könnten LEDs mit einer weiteren zeitlichen Codierung angesteuert werden. Die Nummerierung der gefundenen LED-Positionen im Bild kann dann ebenfalls automatisch erfolgen. Dies bietet dann auch die Möglichkeit der Schätzung aller Parameter des Kameramodells. Für die Ganganalyse müsste der Kalibrierkörper jedoch recht groß sein und seine Handhabung dürfte in der Praxis kompliziert werden.

Statt dessen wurde nur eine LED nacheinander an definierte Punkte bewegt. Das natürlich gegebene Gitter des Kachelbodens macht dies sehr einfach. In jedem Slave Rechner erfolgte die automatische Suche nach der blinkenden LED. Die gefundenen Punktpositionen wurden von jedem Slave an den Master übertragen und von diesem zusammen mit den 3D-Raumpunktkoordinaten (falls angegeben) gespeichert.

Die LED wird von einer Person positioniert. Nach einem Zeichen löst eine zweite Person

## G. Kamerakalibrierung

die Bildaufnahme und damit die Suche nach der LED in jedem Kamerabildern aus. Das Programm fragt nun nach den 3D-Koordinaten der LED und speichert diese zusammen mit den gemessenen Bildkoordinaten eines jeden Kamerakanals in einer Textdatei. Günstig ist es, einige Koordinaten vorher auf dem Boden zu markieren.

Dieser Vorgang könnte auch durch eine Person erfolgen, wenn die LED-Punktlichtquelle mit einem Taster ausgestattet wird, der die Bildaufnahme auslöst. Über eine Sprachausgabe würden vom System die 3D-Koordinaten ausgegeben, die mit der LED als nächstes zu markieren sind. Höhenpasspunkte und Passpunkte, die nicht in der xy-Ebene liegen, könnten mit einem Lot an der Punktlichtquelle markiert werden. Damit lassen sich dann auch Passpunkte in verschiedenen Höhen definieren.

Für Verknüpfungspunkte werden keine 3D-Koordinaten benötigt. Es ist jedoch günstig, die LED so zu positionieren, dass in den Kamerabildern eine gute Punktverteilung erreicht wird. Im Idealfall ergibt sich eine gleichmäßige Punktverteilung über das ganze Messvolumen.

Bei eigenen Experimenten wurden zwischen 20 und 40 Positionen mit der LED angefahren. Dabei waren etwa 12 Punkte reine Passpunkte, 8 Höhenpasspunkte und der Rest reine Verknüpfungspunkte waren.

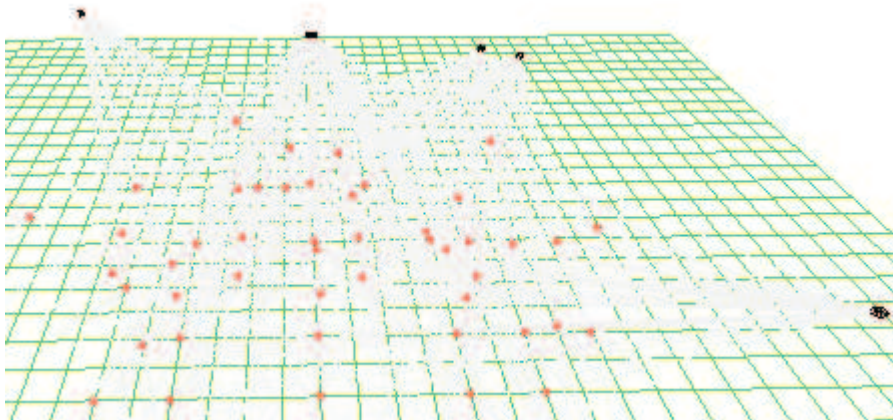


Abbildung G.4.: Ergebnis der Kalibrierung

## H. Formelzeichen und Abkürzungen

- $s$  -  $[m]$ ,  $[mm]$  Weggrößen, Streckenlängen, wenn vektoriell dann  $\mathbf{s}$
- $s_0$  -  $[m]$ ,  $[mm]$  Soll-Streckenlänge, Soll-Gelenkpunktabstand im 3D-Raum (Abschnitt 3.2)
- $s_0$  -  $[Pixel]$  Soll-Gelenkpunktabstand im Bild (Abschnitt 10.1)
- $s_l$  -  $[m]$ , Schrittlänge (engl. Stride Length), Abstand zwischen zwei Standpunkten desselben Beines
- $s_y$  - Skalierungsfaktor des Kameramodells (Abschnitt 5.2.1)
- $\sigma$  - Standardabweichung, Zeitkonstante des Gaborfilters
- $\sigma^2$  - Varianz
- $v$  -  $[m/s]$ ,  $[km/h]$ , Geschwindigkeiten, wenn vektoriell dann  $\mathbf{v}$
- $a$  -  $[m/s^2]$  Beschleunigungen, wenn vektoriell dann  $\mathbf{a}$
- $t$  -  $[s]$ ,  $[1/10s]$ , Zeitangaben, wenn  $t$  tiefgestellt, dann ganzzahliger Index des Zeitschrittes
- $f_s$  -  $[Hz]$ ,  $[1/s]$ , Schrittfrequenz eines Beines
- $T_s$  -  $[s]$  Periodendauer des Schrittes eines Beines, es gilt  $T_s = \frac{1}{f_s}$
- $\theta$  - reelle Schrittzahl, abgerundet ergibt sie die Schrittnummer (Abschnitt 2.1.1)
- $\phi$  - Phasenwinkel einer Komplexen Zahl (am Ausgang des Gaborfilters, Abschnitt 9.4)
- $\mathbf{E}_{tx}(x)$  Homogene Elementarmatrix: Translation entlang der X-Achse um den Weg  $x$
- $\mathbf{E}_{rx}(x)$  Homogene Elementarmatrix: Rotation um X-Achse im Winkel von  $x$
- $\mathbf{T}_{0i}(\mathbf{x})$  Homogene Transformationsmatrix von lokalen Körperteilkoordinaten des Körperteils  $i$  in Weltkoordinaten
- $\mathbf{T}_{ji}(x)$  Homogene Transformationsmatrix von Körperteil  $i$  nach Körperteil  $j$
- $\mathbf{J}$  Jacobimatrix, enthält die partiellen Ableitungen des Modells nach dem unbekanntem Zustandsvektor

## H. Formelzeichen und Abkürzungen

- **A** Kennzeichnet den funktionalen Zusammenhang eines (meist überbestimmten) Gleichungssystems  $\mathbf{A} \cdot \mathbf{x} = \mathbf{y}$ .
- **Q** inverse Matrix, wird zur Lösung eines überbestimmten Gleichungssystems berechnet  $\mathbf{Q} = (\mathbf{A}^T \mathbf{A})^{-1}$ .
- **B(t)** Matrix, enthält das Bewegungsmodell eines Zeitschrittes
- **c** Vektor enthält die (Fourier-) Koeffizienten des Bewegungsmodells
- $c_i$  - formbeschreibender Kontrollpunkt eines Spline-Polynoms (Abschnitt 4.2)
- $c$  - [Pixel] skalare Kamerakonstante, ähnlich der Brennweite eines Objektivs (Abschnitt 5.2.1)
- **p** Punkt des Modells in einem Kamerabild
- **p(x)** Punkt des Modells in einem Kamerabild, Abhängigkeit vom Zustandsvektor  $\mathbf{x}$
- WKS - Weltkoordinatensystem, in dieser Arbeit immer xy-Ebene auf dem Boden, z-Achse nach oben
- FFT - Fast Fourier Transformation, Schnelle Fourier Transformation
- FEM - Finite-Elemente-Methode
- DCT - Discrete-Cosinus-Transformation, Diskrete Kosinustransformation
- LED - Light-Emitting-Diodes, Leuchtdioden
- VK - Vorwärts Kinematik, Berechnung der Körpergeometrie aus dem Zustandsvektor
- IK - Inverse Kinematik, Berechnung des Zustandsvektors aus einer gegebenen Zielkonfiguration.
- TV - Television, Fernseh...
- SVD - Singular-Value-Decomposition, Singulärwertzerlegung
- fps - frames per second, Anzahl der Bilder je Sekunde, Einheit für die Bildaufnahme oder Bildwiedergabe
- Hz - Hertz, Einheit für die Frequenz
- ° - Grad, Einheit für Winkel
- [1/10s] - eine zehntel Sekunde, korrespondiert zur hier häufig verwendeten Bildaufnahme rate (10 fps)
- % - Prozent



# Literaturverzeichnis

- [AC99] AGGARWAL, J. K. ; CAI, Q.: Human Motion Analysis: A Review. In: *Computer Vision and Image Understanding* 73 (1999), March, Nr. 3, S. 428–440
- [AHS89] AIZAWA, K. ; HARASHIMA, H. ; SAITO, T.: Model-based analysis synthesis coding for a persons face. In: *Signal Processing: Image Communication* 1 (2) (1989), Oct., S. 139–152
- [AK89] ALBERTZ, Jörg ; KREILING, Walter: *Photogrammetrisches Taschenbuch*. Karlsruhe : Herbert Wichmann Verlag GmbH, 1989
- [Bau95] BAUMBERG, A. M.: *Learning Deformable Models for Tracking Human Motion*, The University of Leeds, Diss., October 1995
- [Bau96] BAUMBERG, A. M.: Hierarchical shape fitting using an iterated linear filter. In: *Proceedings of the 7th British Machine Vision Conference (BMVC96)*, 1996, S. 313–322
- [Bau00] BAUMBACH, Torsten: *'Shift Detection by Restoration' Signalbasiertes Bild-matching und spezielle Anwendungen*, FSU Jena, Lehrstuhl für digitale Bild-verarbeitung, Diss., 2000
- [BH94] BAUMBERG, A.M. ; HOGG, D.C. *An Efficient Method for Contour Tracking using Active Shape Models*. University of Leeds, Research Report Series, Report 94.11. 1994
- [BI98] BLAKE, Andrew ; ISARD, Michael: *Active Contours*. Springer Verlag, 1998
- [BM98] BREGLER, C. ; MALIK, J.: Tracking People with Twists and Exponential Maps. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1998
- [Boe97] DE BOER, Ingo H.: *Dreidimensionale Lokalisation einer Elektrodenanordnung mittels Analyse von Stereofarbaufnahmen*, Institut für Biomedizinische Technik, Universität Karlsruhe, Diplomarbeit, 1997
- [Bär01] BÄR, Jürgen: *Ganganalyse zur bewegungstherapeutischen Beurteilung des Konzeptes der MASAI BARFUSS TECHNOLOGIE*, Medizinisch-technische Akademie für den physiotherapeutischen Dienst am Ausbildungszentrum West für Gesundheitsberufe TILAK GmbH, Diplomarbeit, 2001

- [Bra99a] BRAND, M.E.: Pattern discovery via entropy minimization. In: *Uncertainty 99: International Workshop on Artificial Intelligence and Statistics*, 1999. – TR98-21
- [Bra99b] BRAND, M.E.: Shadow Puppetry. In: *International Conference on Computer Vision*, 1999. – IEEE 0-7695-0164-8/99, S. 1237 ff
- [BSG<sup>+</sup>96] BRONSTEIN ; SEMENDJAJEW ; GROSCHE ; ZEIDLER ; HACKBUSCH ; SCHWARZ: *Teubner-Taschenbuch der Mathematik*. Teubner Verlagsgesellschaft Leipzig, 1996
- [Bus89] BUSCH, H.: Automatic Modelling of Rigid 3D Objects Using an Analysis by Synthesis System. In: *SPIE Conf. On Visual Communications and Image Processing IV, 1199*. Philadelphia, USA, 1989, S. 356ff
- [BY95] BLACK, M.J. ; YACOOB, Y.: Tracking and Recognizing Rigid and Non-rigid Facial Motions Using Local Parametric Models of Image Motion. In: *ICCV*, 1995, S. 374–381
- [CMH03] CALOW, Roman ; MICHAELIS, Bernd ; HAMADI, Ayoub A.: Solutions for Model-Based Analysis of Human Gait. In: *Pattern Recognition 25th DAGM Symposium Magdeburg*, Springer, September 2003 (LNCS 2781), S. 540–547
- [CNC03] CUNADO, David ; NIXON, Mark S. ; CARTER, John N.: Automatic extraction and description of human gait models for recognition purposes. In: *Computer Vision and Image Understanding 90* (2003). – [www.elsevier.com/locate/cviu](http://www.elsevier.com/locate/cviu)
- [DDR01] DAVISON, A. J. ; DEUTSCHER, J. ; REID, I. D.: Markerless Motion Capture of Complex Full-Body Movement for Character Animation. In: *Proceedings of the Eurographics Workshop on Animation and Simulation*, Springer-Verlag LNCS, 2001
- [DF99] DELAMARRE, Q. ; FAUGERAS, O.: 3D Articulated Models and Multi-View Tracking with Silhouettes. In: *IEEE International Conference on Computer Vision ICCV99*, 1999
- [DF01] DELAMARRE, Q. ; FAUGERAS, O.: 3D Articulated models and multi-view tracking with physical forces. In: *CVIU journal* 81 (2001), S. 328–357
- [DLR77] DEMPSTER, A.P. ; LAIRD, N.M. ; RUBIN, D.B.: Maximum-likelihood from incomplete data via the EM algorithm. In: *Journal of the Royal Statistical Society* B39 (1977), S. 1–38
- [FDFH97] FOLEY, D. J. ; v. DAM, A. ; FEINER, S. K. ; HUGHES, J. F.: *Computer Graphics: Principles and Practice*. second edition. Addison-Wesley, 1997
- [For73] FORNEY, G.: The Viterbi algorithm. In: *Proceedings of the IEEE*, 6:268-278, 1973

- [GD96] GAVRILA, D.M. ; DAVIS, L.S.: Tracking of humans in action: a 3-D model-based approach. In: *ARPA Image Understanding Workshop*. Palm Springs, Feb. 1996
- [GD99] GAVRILA, D.M. ; DAVIS, L.S.: The Visual Analysis of Human Movement: A Survey. In: *Computer Vision and Image Understanding (CVIU)* 73 (1999), Nr. 1, S. 82–98. – Academic Press
- [GIM99] GIONIS, A. ; INDYK, P. ; MOTWANI, R.: Similarity search in high dimensions via hashing. In: KAUFMANN, Morgan (Hrsg.): *Proceedings of the 25th International Conference on Very Large Data Bases (VLDB'99)*, 1999, S. 518–529
- [Gus92] GUSE, W.G.: *Objektorientierte Bewegungsanalyse in Bildfolgen*, TH Aachen, Diss., 1992. – (D82)
- [Hen02] HENZE, A.: *Dreidimensionale biomechanische Modellierung und die Entwicklung eines Reglers zur Simulation zweibeinigen Gehens*, Eberhard-Karls-Universität zu Tübingen, Fak. f. Physik, Diss., 2002
- [Höh01] HÖHNE, Dirk: *Keysignalerzeugung durch stereoskopische Entfernungsmessung*, Fakultät für Maschinenbau und Elektrotechnik der technischen Universität Carola-Wilhelmina zu Braunschweig, Diss., 2001
- [HLF00] HOWE, R. ; LEVENTON, M.E. ; FREEMAN, W.T.: Bayesian Reconstruction of 3D Human Motion from Single-Camera Video. In: *MIT Press* (2000)
- [HS96] HANSEN, M. ; SOMMER, G.: Real-time vergence control using local phase differences. In: *Machine Graphics and Vision* 5 (1996), S. 51–63
- [JB93a] JEPSON, A. ; BLACK, M.: Mixture Models for Optical Flow Computation / University of Toronto. 1993 ( RBCV-TR-93-94). – Technical Report
- [JB93b] JEPSON, A. ; BLACK, M.J.: Mixture models for optical flow computation. In: *IEEE Conf. Computer Vision Pattern Recognition* (1993), S. 760–761
- [JBY96] JU, Shanon ; BLACK, M.J. ; YACOOB, Y.: Cardboard People: A Parameterized Model of Articulated Image Motion. In: *International Conference of Automatic Face and Gesture Recognition*, 1996
- [Koc93] KOCH, K.: Dynamic 3D Scene Analysis through Synthesis Feedback Control. In: *IEEE Trans. Patt. Anal. Mach. Intell., Special issue on analysis and synthesis* 15 (1993), June, Nr. 6, S. 556–568
- [Kra97] KRAUS, Karl: *Photogrammetrie*. Bd. 1 Grundlagen und Standardverfahren. sechste Auflage. Bonn : Ferd. Dümmler's Verlag, 1997
- [Kru83] KRUCK, E.: *Lösung großer Gleichungssysteme für photogrammetrische Blockausgleichungen mit erweitertem funktionalem Modell*, Universität Hannover (Fachrichtung Vermessungswesen), Diss., 1983

- [KSV86] KUNDERT, Kenneth ; SANGIOVANNI-VINCENTELLI, Alberto: *Sparse 1.3 (A Sparse Linear Equation Solver)*. University of California, Berkeley. 1986. – See Internet: SPARSE 1.3 Library in C
- [Lan99] LANDER, Jeff: Over My Dead, Polygonal Body. In: *Game Developer Magazine* (1999). – Source code Skeletal Deformation in OpenGL available
- [LAZ98] LEGNANI, G. ; ADAMINI, R. ; ZAPPA, B. *Spacelib in C*. <http://bsing.ing.unibs.it/~legnani/>. July 1998
- [LBK90] LIEDTKE, C.E. ; BUSCH, H. ; KOCH, R.: Automatic Modelling of 3D Moving Objects from a TV Image Sequence. In: *SPIE Conf. on Sensing and Reconstruction of 3D-Objects and Scenes*. Santa Clara USA, Feb. 1990. – Vol. 1260, S. 230–239
- [LC87] LORENSEN, W. ; CLINE, H.: Marching cubes: A high-resolution 3D surface construction algorithm. In: *SIGGRAPH' 87*, 1987, S. 163–169
- [Lin99] LINKE, Janette: *Zur Analyse der Geometrie und Beurteilung von geodätischen und mechanischen Netzen unter Verwendung von Balancierungsfaktoren*, Technische Universität Clausthal, Diss., 1999
- [Luh00] LUHMANN, Thomas: *Nahbereichsphotogrammetrie: Grundlagen, Methoden und Anwendungen*. erste Auflage. Heidelberg : Wichmann Verlag, 2000
- [LZN02] LV, Fengjun ; ZHAO, Tao ; NEVATIA, Ram: Self-Calibration of a camera from video of a walking human. In: *ICPR*, 2002
- [MAD01] MESTER, Rudolf ; AACH, Til ; DÜMBGEN, Lutz: Illumination-invariant change detection using a statistical colinearity criterion. In: *Proceedings Deutsche Arbeitsgemeinschaft für Mustererkennung (DAGM2001)*, 2001
- [Mat96] MATUTTIS, Hans-Georg. *Praktische Mathematik mit MATLAB (Version 2.0)*. <http://sun.uni-regensburg.de/matlab-5.3.1/html/script/mscript.html>. 1996
- [Mec99] MECKE, R.: *Grauwertbasierte Bewegungsschätzung in monokularen Bildsequenzen unter besonderer Berücksichtigung bildspezifischer Störungen*. Shaker Verlag, Aachen, Otto-von-Guericke Universität Magdeburg, Diss., 1999
- [MG01] MOESLUND, T.B. ; GRANUM, E.: A Survey of Computer Vision-Based Human Motion Capture. In: *Computer Vision and Image Understanding (CVIU)* 81 (2001), March, Nr. 3, S. 231–268
- [Mül01] MÜLLER, T.: *Modellbasierte Lokalisation und Verfolgung für sichtsystemgestützte Regelungen*, Universität Karlsruhe (TH), Fakultät für Informatik, Diss., 2001
- [MPN98] MEYER, D. ; PÖSL, J. ; NIEMANN, H.: Gait Classification with HMMs for

- Trajectories of Body Parts Extracted by Mixture Densities. In: *British Machine Vision Conference (BMVC)*, 1998, S. 456–468
- [NH02] NWOKAH, Osita D. I. ; HURMUZLU, Yildirim: *The Mechanical Systems Design Handbook*. erste Auflage. Boca Raton, Florida, USA: CRC Press LCC, 2002
- [Nie99] NIEM, Wolfgang: *Rekonstruktion starrer dreidimensionaler Objekte aus Kamerabildern*, Universität Hannover, Diss., 1999
- [OG99] ONG, Eng-Jon ; GONG, Shaogang: Tracking Hybrid 2D-3D Human Models from Multiple Views. In: *ICCV*, 1999
- [Pie96] PIEGL, L.: *The Nurbs Book*. 2nd Edition. Springer-Verlag Berlin Heidelberg New York, 1996
- [Pre86] PRESS, William H.: *Numerical Recipes*. University Press, Cambridge, 1986
- [Rat98] RATNER, P.: *3-D Human Modelling and Animation*. New York : John Wiley and Sohns, Inc, 1998
- [Rog95] ROGERSON, Dale. *OpenGL VI: Rendering on DIBs with PDF\_DRAW\_TO\_BITMAP*. Microsoft Developer Network Technology Group. April 1995
- [RP66] ROSENFELD, A. ; PFALTZ, J.: Sequential operations in digital picture processing. In: *J. Assoc. Comp. Mach.* 13 (1996), Nr. 4, S. 471–494
- [RS88] ROBERSON, R. ; SCHWERTASSEK, R.: *Dynamics of Multibody Systems*. Springer-Verlag, 1988
- [RS98] ROSALES, R. ; SCLAROFF, S.: Improved Tracking of Multiple Humans with Trajectory Prediction and Occlusion Modelling. In: *IEEE CVPR98*, 1998
- [SB01] SEITZ, T. ; BUBB, H.: Bewegungsmessung beim Menschen ohne Marker. In: *Tagungsband zum 47. Kongress der Gesellschaft für Arbeitswissenschaft*, 2001
- [SBB00] SEITZ, T. ; BALZULAT, J. ; BUBB, H.: Anthropometry and measurement of posture and motion. In: *International Journal of Industrial Ergonomics* (2000), Nr. 25
- [SBF00] SIDENBLADH, H. ; BLACK, M.J. ; FLEET, D.J.: Stochastic Tracking of 3D Human Figures Using 2D Image Motion. In: *6th ECCV*. Dublin, 2000
- [Sch00] SCHMIDT, C.O.: Der bewegte Mensch. In: *Ct* 23 (2000)
- [Sie03] SIEBEL, Niels T.: *Design and Implementation of People Tracking Algorithms for Visual Surveillance Applications*. UK, The University of Reading, Diss., March 2003. – <http://www.cvg.cs.rdg.ac.uk/~nts/PeopleTracking/>

## Literaturverzeichnis

- [SK00] SCHEID, H. ; KINDINGER, D.: *Schülerduden Mathematik 2*. Dudenverlag, Mannheim, 2000
- [Soi98] SOILLE, P.: *Morphologische Bildverarbeitung*. Berlin : Springer Verlag, 1998
- [STB00] SIDENBLADH, H. ; LA TORRE, F. D. ; BLACK, M.J.: A Framework for Modelling the Appearance of 3D Articulated Figures. In: *Proc. 4th IEEE International Conference on Automatic Face and Gesture Recognition*, 2000, S. 368–375
- [Ste04] STEIN, Dr. I. *Vitus 3D-Bodyscanner*. <http://www.vitronic.de/>. 2004
- [SVD03] SHAKHNAROVICH, G. ; VIOLA, P. ; DARRELL, T.: *Fast Pose Estimation with Parameter Sensitive Hashing*. [www.ai.mit.edu](http://www.ai.mit.edu). April 2003. – AI Memo 2003-009
- [SW98] SINGER, Y. ; WARMUTH, M.K. *A New Parameter Estimation Method for Gaussian Mixtures*. <http://citeseer.nj.nec.com/singer98new.html>. 1998
- [SZM85] SCHWARZ, W. ; ZECHA, M. ; MEYER, G.: *Industrierobotersteuerungen*. erste Auflage. Berlin: VEB Verlag Technik, 1985
- [TGB00] TOLANI, D. ; GOSWAMI, A. ; BADLER, N.I.: Real-Time Inverse Kinematics Techniques for Anthropometric Limbs. In: *Graphical Models* 62 (2000), Sept., Nr. 5, S. 353–388
- [VDO92] VAUGHAN, Christopher L. ; DAVIS, Brian L. ; O'CONNOR, Jeremy C.: *Gait lab Gait analysis laboratory: an interactive book and software package*. Champaign, Ill.: Human Kinetics Publ., 1992
- [Vel00] VELSZ, I.: *3D Studio MAX R3, Grundlagen und Praxis der 3D-Visualisierung und Animation*. München : Addison-Wesley, 2000
- [WA94] WANG, J.Y.A. ; ADELSON, E.H.: Representing Moving Images with Layers. In: *IEEE Transactions on Image Processing (Image Sequence Compression)* (1994)
- [Wac97] WACHTER, Stefan: *Verfolgung von Personen in monokularen Bildfolgen*, Universität Karlsruhe, Diss., 1997
- [WADP97] WREN, C. ; AZARBAYEJANI, A. ; DARRELL, T. ; PENTLAND, A.: Pfänder: Real-Time Tracking of the Human Body. In: *IEEE Pattern Analysis and Machine Intelligence* (1997)
- [Wei00] WEIK, Sebastian: Passive Full Body Scanner Using Shape from Silhouettes. In: *Proceedings of the ICPR*, Barcelona, Spain, 2000
- [Win90] WINTER, David A.: *Biomechanics and motor control of human movement*. 2. Wiley, New York, 1990 (A Wiley-Interscience publication)

- [Wit77] WITTENBURG, J.: *Dynamics of Systems of Rigid Objects*. B. G. Teubner Stuttgart, 1977
- [WL01] WEIK, S. ; LIEDTKE, C.-E.: Hierarchical 3D Pose Estimation for Articulated Human Body Models from a Sequence of Volume Data. In: *Robot Vision*. Auckland, New Zealand, 2001
- [WN99] WEIK, S. ; NIEMEYER, O.: 3D Motion Estimation for Articulated Human Templates Using a Sequence of Stereoscopic Image Pairs. In: *Proceedings of Visual Communications and Image Processing (VCIP99)* 3653 (1999). – Proceedings of SPIE
- [WR71] WILKINSON ; REINSCH: *Handbook of Automatic Computation*. Springer-Verlag, 1971