Lehrstuhl für Mensch-Maschine-Kommunikation
Technische Universität München

Personenverfolgung und Gestenerkennung in Videodaten

Sascha Schreiber

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

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender:	Univ.-Prof. Dr.techn. Josef A. Nossek
Prüfer der Dissertation:	1. Univ.-Prof. Dr.-Ing. habil. Gerhard Rigoll
	2. Univ.-Prof. Dr.-Ing. Kristian Kroschel, Universität Karlsruhe (TH)

Die Dissertation wurde am 22.10.2008 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 16.03.2009 angenommen.

Kurzfassung

Die computergestützte Analyse von Bild- und Videodaten gewinnt seit nunmehr zwei Jahrzehnten immer mehr an Bedeutung. Als ein Teilgebiet stellt dabei die automatische Detektion und Verfolgung von Objekten die fundamentale Grundlage für zahlreiche weiterführende Aufgaben aus dem Bereich der Videoanalyse dar.
Die vorliegende Arbeit beschäftigt sich mit der Entwicklung einer neuartigen Architektur zur automatisierten Personenverfolgung mit Fokus auf Besprechungsszenarien. Angelehnt an das physiologische Verständnis der menschlichen Szenenanalyse wird hierzu die Problemstellung des Personentrackings sowohl aus der bottom-up sowie gleichzeitig aus der top-down Sichtweise betrachtet. Ziel hierbei ist eine hybride Systemarchitektur, welche basierend auf einer datengetriebenen Optimierung von Zustandshypothesen eine effiziente Technik zur simultanen Verfolgung mehrerer Personen anhand deren Köpfen realisiert. Unter Nutzung von verschiedenen Objektmodellierungen werden im Rahmen dieser Arbeit diverse Architekturen implementiert, gegeneinander evaluiert und die erzielten Ergebnisse anhand definierter Metriken diskutiert.
Die erhaltenen Trackingergebnisse bilden die Basisinformation für die weiterführende Gestenerkennung. In realen Besprechungen werden Personen häufig von anderen Teilnehmern verdeckt, so dass die ausgeführten Gesten in der zweidimensionalen Bildebene vom System nur partikulär erfasst werden können. Aus diesem Grund wird in dieser Arbeit weiter untersucht, wie sich unterschiedliche Störungen auf die Erkennungsrate von Gesten auswirken. Darauf aufbauend werden Systemkonzepte, die eine Kompensation dieser Störungen erlauben, entworfen und evaluiert.

Inhaltsverzeichnis

1 Einleitung **1**
- 1.1 Motivation . . . 2
- 1.2 Aufbau der Arbeit . . . 6

2 Grundlagen der Personenverfolgung **9**
- 2.1 Vorverarbeitung . . . 11
  - 2.1.1 Detektion von Hautfarbe . . . 12
  - 2.1.2 Hintergrundsegmentierung . . . 18
- 2.2 Personendetektion . . . 23
  - 2.2.1 Gesichtsdetektion mittels Neuronaler Netze . . . 26
  - 2.2.2 Waveletbasierte Gesichtsdetektion . . . 31
- 2.3 Trackinglogik . . . 38
  - 2.3.1 Bestimmung der Personenkorrespondenzen . . . 39
  - 2.3.2 Trajektorienberechnung . . . 40
  - 2.3.3 Prädiktion der Objekteigenschaften . . . 41

3 Videobasierte hybride Personenverfolgung in Besprechungsszenarien **43**
- 3.1 Hypothesenbasiertes Tracking . . . 45
- 3.2 Einzelpersonenverfolgung . . . 50
  - 3.2.1 Modellierung von Köpfen mittels Ellipsen . . . 50
  - 3.2.2 Active Shape Modelle . . . 53
- 3.3 Mehrpersonenverfolgung . . . 63
  - 3.3.1 Mehrschichtiger Partikelfilter . . . 64
  - 3.3.2 Simulated Annealing . . . 70

4 Tracking-Evaluierung **79**
- 4.1 Historie der Tracking-Evaluierung . . . 79
- 4.2 Datenbank . . . 80
- 4.3 Evaluierungsschema . . . 84
  - 4.3.1 Das Zuordnungsproblem . . . 84
  - 4.3.2 Beurteilung von Trackingfehlern bezüglich der Personenkonfiguration . . . 86

4.3.3 Beurteilung von Trackingfehlern bezüglich der Personenidentitäten . . . 87
4.3.4 Prägnante Größen zur Bewertung von Trackingergebnissen auf Videosequenzen . . . 89
4.4 Evaluation Einzelpersonenverfolgung . . . 91
4.4.1 Evaluierte Systeme zur Einzelpersonenverfolgung . . . 91
4.4.2 Diskussion der Evaluationsergebnisse . . . 93
4.4.3 Zusammenfassung der Ergebnisse . . . 100
4.5 Evaluation Mehrpersonenverfolgung . . . 101
4.5.1 Evaluierte Systeme zur Mehrpersonenverfolgung . . . 101
4.5.2 Diskussion der Evaluationsergebnisse . . . 102
4.5.3 Zusammenfassung der Ergebnisse . . . 108

5 Gesten- und Aktionserkennung 111
5.1 Datenbank . . . 112
5.2 Merkmale . . . 113
5.3 Merkmalsextraktion und Aufbereitung . . . 116
5.3.1 Erzeugung rauschbehafteter Merkmale . . . 117
5.3.2 Modell der Merkmalsextraktion . . . 117
5.3.3 System der Merkmalsaufbereitung . . . 119
5.4 Experimente und Ergebnisse . . . 122
5.4.1 Erkennung von ungestörten Gesten . . . 122
5.4.2 Erkennung von rauschbehafteten Gesten . . . 126

6 Zusammenfassung 131
6.1 Hybride Trackingarchitektur . . . 131
6.2 Gestenerkennung . . . 133
6.3 Weitere Anwendungsgebiete . . . 134

A Abkürzungen 137

B Formelzeichen 139

C Theorie der eindimensionalen Hidden Markov Modelle 143
C.1 Doppelt stochastische Prozesse . . . 143
C.1.1 Kontinuierliche HMM . . . 145
C.1.2 Diskrete HMM . . . 146
C.2 Training eines HMM . . . 147
C.3 Klassifikation mittels HMM . . . 148
C.4 Viterbi-Algorithmus . . . 149

D Theorie des Kalmanfilters **151**

Literaturverzeichnis **155**

Kapitel 1

Einleitung

Seit zwei Jahrzehnten rückt die automatische Analyse von Bild- und Videodaten immer mehr in den Blickpunkt internationaler Forschungstätigkeiten. Ausgangspunkt hierfür war die in den frühen 80er Jahren einsetzende rasante Verbreitung von Systemen zur Videoüberwachung öffentlicher sowie privater Plätze und Gebäude, eingeleitet durch die enormen Fortschritte im Bereich der Computerhardware – vor allem in Bezug auf Datenspeicher und Rechenleistung – sowie der Entwicklung von hochauflösenden und rauscharmen optischen Sensoren zu erschwinglichen Preisen. Anfangs beschränkte sich hierbei die Aufgabe von Videoüberwachungssystemen lediglich auf die Bereitstellung von Daten ohne diese jedoch auch unmittelbar zu interpretieren.
Die bedingt durch die wachsende Zahl an Kameras anfallende Datenflut verlangte aber schon bald nach einer automatisierten Auswertung und Aufbereitung der in den Videodaten enthaltenen Informationen. Während vor allem in industriellen Anwendungen, wie beispielsweise der Ablaufsteuerung oder der Qualitätsprüfung, die automatische Bildanalyse aufgrund der relativ definierten Umgebungsbedingungen schon sehr schnell zielführende Ergebnisse liefern konnte, stellt die robuste Auswertung von Videos für natürliche Szenarien, bei denen keine oder nur wenig Einflussmöglichkeiten auf die Rahmenbedingungen gegeben sind, eine teils immer noch große Herausforderung dar. Typische Aufgabenstellungen im Rahmen solcher natürlichen Umgebungen sind:

- Fahrerintentionserkennung in der Automobildomäne
- Unfallprävention im Straßenverkehr durch Fußgängerdetektion
- Verkehrskontrollsysteme (z. B. Stadt- oder Autobahnmautsysteme)
- Bildgebende Verfahren der Medizintechnik
- Personenidentifikation in sicherheitskritischen Anwendungen
- Verhaltensanalyse von Menschenmassen (z. B. zur Gestaltung von Fluchtwegen)

- Detektion verdächtiger Verhaltensmuster zur Erkennung von Bedrohungsszenarien auf öffentlichen Plätzen oder in Gebäuden
- Automatische Auswertung von Besprechungen
- usw.

Systeme, die solch eine automatische Videoanalyse leisten und den Nutzer mit bereits interpretierten Informationen versorgen, stehen im Fokus aktueller Forschung und sind gemeinhin unter dem Begriff „intelligente Überwachungssysteme“[1] bekannt. Speziell für Sicherheitsdienste und Banken sind derlei Systeme von großem Interesse, da sie die Arbeit des Sicherheitspersonals erleichtern und somit Potential für eine effizientere Kontrolle sicherheitskritischer Bereiche[2] bieten. Einen weiteren Beleg dafür, dass hinter dieser Forschungsarbeit auch ein enormes kommerzielles Interesse steht, liefert zudem eine vom Marktforschungsinstitut JP Freeman durchgeführte Studie, die dem Markt für intelligente Überwachung ein Umsatzwachstum von 7 Milliarden Dollar im Jahr 2005 auf über 13 Milliarden Dollar im Jahr 2010 voraussagt.

1.1 Motivation

Bedingt durch die fortschreitende Globalisierung der Märkte spielt der Austausch von Menschen untereinander, sei es zur Konfliktbewältigung, zum Wissenstransfer oder zur Knüpfung sozialer Kontakte, eine immer bedeutendere Rolle. Eine Tatsache, die sich einer Vielzahl von wirtschaftswissenschaftlichen Studien[3] zufolge auch in der Organisation von Besprechungen niederschlägt. So nahm seit 1960 die durchschnittliche Zeitdauer, die pro Woche von einem Mitarbeiter auf mittlerer Managementebene für Besprechungen aufgewendet werden muß, kontinuierlich von ca. 3,5 h auf mehr als 10 h gegen Mitte der 90er Jahre zu. Obwohl Umfragen ergaben, dass viele Teilnehmer solcher Meetings die Produktivität und Effektivität der Besprechungen als eher niedrig einstufen, erwartet dennoch die große Mehrheit der Befragten in Zukunft eine weiter steigende Zahl an Besprechungen. Als problematisch erweist sich hierbei, dass sich Termine

[1] Engl. „*smart video surveillance systems*“

[2] Studien zufolge (vgl. Green [38]) nimmt die Aufmerksamkeit eines Individuums beim gleichzeitigen Sichten mehrerer Monitore aufgrund der monotonen Tätigkeit bereits nach 20 Minuten rapide ab, so dass eine Identifikation möglicher Bedrohungsszenarien nur mehr sehr unzureichend sichergestellt werden kann.

[3] Eine Zusammenfassung der Ergebnisse zahlreicher Veröffentlichungen zu dem Thema wurde von Romano u. Nunamaker [85] publiziert.

überschneiden oder aufgrund der möglicherweise sehr weit voneinander entfernten Veranstaltungsorte erst gar nicht wahrgenommen werden können. Personen, die aus den genannten Gründen nicht an der Versammlung teilnehmen, aber dennoch Interesse an den Beschlüssen haben, können sich bisweilen nur unter Rückgriff auf die Besprechungsprotokolle über den Verlauf des Meetings informieren. Allerdings müßten diese Protokolle, um einen authentischen und gesamtheitlichen Eindruck des Meetings widerspiegeln zu können und damit auch für Außenstehende die Anbahnung von getroffenen Entscheidungen besser nachvollziehbar werden zu lassen, sehr viel detaillierter abgefasst sein und beispielsweise Emotionen oder exakte Formulierungen bestimmter Aussagen beinhalten. Bedingt jedoch durch die Tatsache, dass Protokolle von menschlichen Beobachtern erstellt werden und daher immer einer mehr oder weniger subjektiven Bewertung unterliegen, die von dieser Person unmittelbar in der konkreten Mitschrift zum Ausdruck gebracht wird, kann selbst ein um diese Merkmale erweitertes Protokoll keine letztlich neutrale und somit objektive Informationsquelle darstellen (vgl. Schultz u. a. [95]). Darüber hinaus wäre aber auch allein aus Kosten-Nutzen Aspekten die Anfertigung eines solch umfassenden Protokolls von Hand nicht erfüllbar, da bereits die heutzutage üblicherweise angefertigten Ergebnisprotokolle einen enormen zeitlichen und finanziellen Aufwand darstellen.
Um dennoch eine Lösung für das Problem zu finden, Meetings produktiver zu gestalten und deren Ergebnisse in derartiger Weise aufzubereiten, dass sich absente Personen in kurzer Zeit auf den aktuellen Informationsstand der restlichen Versammlungsteilnehmer bringen können, beschäftigen sich zahlreiche Projekte auf internationaler Ebene mit der computergestützten, multimodalen Analyse von Besprechungen[4]. Im Jahr 2001 wurde dazu in Amerika vom „National Institute of Standards and Technology“ (NIST) ein Programm namens „Meeting Room Project“ [5] initiiert mit dem Ziel, eine Datenbank an Besprechungen aufzubauen und darauf basierend Technologien zu entwickeln, um Sprache in Text zu wandeln und zusammen mit den aus den Videodaten extrahierten Informationen in verwertbares Wissen zu transformieren.
Etwa zur gleichen Zeit begann man auf europäischer Ebene mit dem Start des Forschungsprojektes M4 (MultiModal Meeting Manager, [4]), diese Thematik zu untersuchen und einen intelligenten Besprechungsraum aufzubauen, der mit einer Vielzahl von visuellen und akustischen Sensoren bestückt ist. Resultat dieses Projektes war schließlich ein Demonstrationssystem, mit dem man archivierte, automatisch analysierte Besprechungen nach Inhalten durchforsten und ge-

[4]Unter dem Kontext der Besprechung soll hier neben typischen Firmenmeetings auch dazu verwandte Veranstaltungen wie Vorlesungen, Seminare oder auch formlose Gruppenbesprechungen verstanden werden.

wünschte Szenen sowohl in Textform sowie als Video detailliert betrachten kann. Von Seiten der in der Grundlagenforschung entwickelten Algorithmen waren allerdings für dieses Projekt noch sehr starre Rahmenbedingungen vorgegeben, wie beispielsweise:

- Fixe Teilnehmerzahl von vier Personen über die gesamte Dauer der Besprechung
- Genau geplante Phasenabfolge der Themen, wodurch zwar eine sehr gute, aber auch unnatürliche Strukturierung des Meetings gegeben ist
- Nahezu keine Störgeräusche und damit optimale Verhältnisse für die automatische Spracherkennung
- Keinerlei störende Objekte im Hintergrund der Personen

Darüber hinaus war – bedingt durch die verwendeten Technologien – eine Auswertung des Meetings nicht in Echtzeit möglich.
In einem weiteren Projekt namens AMI (Augmented Multiparty Interaction, [2]), welches im Jahr 2004 startete, bestand das Augenmerk insbesondere darin, die im Rahmen von M4 entwickelten Technologien robuster gegenüber potentiellen Störquellen zu gestalten, um die im Zuge des M4-Projektes aufgestellten Restriktionen aufzulösen, und gleichzeitig die verwendeten Algorithmen in Richtung Echtzeitfähigkeit zu optimieren. Einhergehend mit der realzeitfähigen Verarbeitung visueller und akustischer Daten, auch auf semantisch höherwertiger Ebene, sollte damit zusätzlich die Möglichkeit geschaffen werden, auch Teilnehmer via Videokonferenz mit in die Analyse einzubeziehen. Das Ziel dieses Projektes bestand darin, rechnergestützte Gruppenarbeit (CSCW[5]) in der Hinsicht zu ermöglichen, dass – sich mitunter auch an unterschiedlichen geographischen Orten aufhaltende – Personen in der von ihnen sonst praktizierten Weise mit einem Maximum an Komfort kollaborieren können und dadurch die Produktivität signifikant gesteigert wird. Als ein wichtiges Mittel zur Umsetzung dieses anspruchsvollen Ziels wurde dabei die computerseitige Erkennung und Interpretation von Emotionen erachtet, die durch den Einsatz neuartiger Technologien ermöglicht werden soll.
Als ein weiteres Indiz für die enorme Bedeutung, die man der Thematik CSCW in Zusammenhang mit intelligenten Räumen beimisst, kann die Tatsache interpretiert werden, dass zeitgleich mit dem Start des Projektes AMI ebenfalls im Rahmen eines groß angelegten EU-Forschungsprogramms das CHIL[6]-Konsortium damit begann, computergestützte Systeme zu entwickeln, die anhand aller aus dem optischen und akustischen Kanal verfügbaren Informationen mit Hilfe neu-

[5]Engl. „*Computer Supported Collaborative Work*“
[6]CHIL - Computers in the Human Interaction Loop, [3]

artiger Algorithmen zwischenmenschliche Interaktionen zu deuten wissen. Ziel hierbei ist es, den Computer direkt in diese Interaktionskette einzubinden in der Weise, dass er möglichst unauffällig integriert in die jeweilige Umgebung den Menschen in seiner Handlung unterstützt und er sich den Bedürfnissen seines Benutzers entsprechend anzupassen vermag.
Die direkte Fortführung der in AMI bereits erfolgreich entwickelten Technologien mündete in das im Herbst 2007 begonnene Projekt AMIDA (Augmented Multi-party Interaction with Distance Access, [1]). Der Fokus wird hierbei insbesondere auf die Erweiterung der Funktionalitäten vor allem in den Bereichen Telefon- sowie Videokonferenzen durch beispielsweise interaktive Schnellsuche in archivierten Daten oder personalisierte Unterstützungsoptionen von Seiten des Computers während eines Meetings gelegt. Die Vision des Projektes besteht vor allem darin, mit einem Besprechungsassistenten ein System zu entwickeln, welches aufgrund aktueller Geschehnisse während der laufenden Konferenz in der Lage ist, Dritte zu benachrichtigen, sobald ein für sie relevantes Thema diskutiert wird, oder Personen selbständig über den bisherigen Verlauf des Meetings zu informieren mit dem Ziel, die – räumlich entfernte – Zusammenarbeit weiter zu verbessern und dadurch die Notwendigkeit von zeitaufwändigen Reisen zu minimieren.
Als eine fundamentale Grundlage zur Umsetzung sämtlicher Ideen innerhalb der angesprochenen Projekte wird dabei Wissen in Form von Angaben über die Position sowie Orientierung aller an der Konferenz partizipierenden Personen vorausgesetzt, um darauf aufbauend beispielsweise die Identität einer Person festzustellen, deren Emotionen zu erkennen oder von ihr ausgeführte Aktionen zu bewerten und dadurch in einer weiterführenden Prozesskette schlussendlich die ehrgeizigen, oben erläuterten Projektziele realisieren zu können. Zur Ermittlung der grundsätzlich benötigten Aufenthaltskoordinaten bedienen sich die im Zuge der genannten Projekte entwickelten Methodiken prinzipiell des visuellen und gegebenenfalls des akustischen[7] Kanals. Während es sich hierbei für den Menschen als sehr einfach gestaltet, aus der zweidimensionalen Bildprojektion der realen Welt Objekte zu lokalisieren und deren Lagebeziehungen zu bestimmen, ist die computerbasierte Analyse einer Szene aus monokularen Bilddaten im Kontext des Bildverstehens auch heutzutage noch nicht allgemeingültig gelöst (vgl. Shen u. a. [98]). Erst durch explizite Einbeziehung von anwendungsspezifischem Vorwissen werden Algorithmen überhaupt dazu in die Lage versetzt,

[7]Speziell im Kontext von Konferenzen erweist sich die alleinige Nutzung des akustischen Kanals zur Positionsbestimmung eines Teilnehmers als nicht zielführend, da oftmals nur eine Person spricht und dadurch der momentane Aufenthaltsort der anderen Personen nicht feststellbar wäre.

erfolgreich im Sinne einer Szenenanalyse Objekte von den restlichen Bildbereichen zu segmentieren und beispielsweise über den zeitlichen Fortschritt für diese Objekte die Trajektorie[8] zu bestimmen. Dieses Vorwissen kann in seiner rudimentärsten Form lediglich in der geometrischen Information über das Objekt, welches in einem konkreten Anwendungsfall von Interesse ist, bestehen oder aber ergänzt werden um statisches Wissen über die Szene selbst, wie z. B. Raumgeometrien oder überhaupt für ein bestimmtes Objekt mögliche Erscheinungsorte. Während diese Information in einem mehr oder weniger aufwendigen Prozess zur Verfügung gestellt und damit maßgeblich selbst beeinflusst werden kann, muss beim Entwurf eines Systems zur Objektverfolgung darüber hinaus ebenso der Einfluss extrinsischer Umgebungsparameter, wie beispielsweise Beleuchtungsschwankungen über die Zeit Berücksichtigung finden.
Speziell im Hinblick auf eine automatische Besprechungsanalyse kann mit der Information über aktuell von einer jeweiligen Person ausgeführte Aktionen und Gesten ein wichtiges Merkmal zur Verfügung gestellt werden. Auch hierfür ist die Kenntnis der genauen Aufenthaltsposition derjenigen Person, deren Gesten identifiziert werden sollen, insofern hilfreich, als dass damit eventuell im Bild als Rauschquellen in Erscheinung tretende Bereiche erfolgreich eliminiert werden können. Basierend auf den ermittelten Gesten kann dann wiederum auf semantisch höherer Ebene durch Ansätze von Reiter u. a. [81] sowie Al-Hames u. Rigoll [6] über das Gruppenverhalten der Status der laufenden Konferenz kategorisiert werden.

1.2 Aufbau der Arbeit

Das Thema der vorliegenden Arbeit ist die Entwicklung, Implementierung und Evaluierung von neuartigen Verfahren zur Analyse von Besprechungen auf Ebene einzelner Personen. Hauptaugenmerk liegt hierbei auf der Extraktion der wesentlichen Basisinformationen über die Person wie Aufenthaltsort, Identität oder getätigte Aktionen.
In Kapitel 2 werden zunächst die allgemeinen Grundlagen von Systemen zur Personenverfolgung erläutert. In diesem Zusammenhang werden neben bewährten Techniken zur Personendetektion, welche die Kernkomponente solcher Systeme bilden, auch die Methoden zur Vorverarbeitung des Videosignals sowie nachgelagerte Prozessschritte zur eindeutigen Bestimmung von Trajektorien beleuchtet. Das Konzept der hybriden Personenverfolgung wird anschließend in Kapitel 3

[8] Unter der Trajektorie versteht man die Koordinaten des momentanen Aufenthaltsortes aufgetragen über dem zeitlichen Verlauf, auch als Bewegungspfad bezeichnet.

präsentiert. Hierzu wird in Abschnitt 3.1 zunächst die hypothesengestützte Objektverfolgung erklärt, bevor in Abschnitt 3.2 mit einer kurzen Einführung möglicher Modellierungen von Personen anhand ihrer Köpfe fortgefahren wird. Mittels dieser Modelle wird anschließend über einen stochastischen Partikelfilter ein System zur robusten Verfolgung von Einzelpersonen realisiert. Das vorgestellte Verfahren zur Einzelpersonenverfolgung wird in Abschnitt 3.3 durch strukturelle neuartige Maßnahmen erweitert, um zeitgleich mehrere Personen im Bild zu verfolgen. Insbesondere bei komplexen Szenarien, in denen sich ein System allein basierend auf einem einzigen Partikelfilter nur bedingt als zielführend erweist, kann durch einen hierarchischen Aufbau die Qualität der vom Algorithmus geleisteten Ergebnisse maßgeblich gesteigert werden.
Kapitel 4 befasst sich dann ausführlich mit der Evaluierung und kritischen Hinterfragung der Ergebnisse, die durch die in den beiden vorangegangenen Kapiteln vorgestellten Methoden erzielt wurden. Die angestellten Untersuchungen basieren dabei notwendigerweise auf einem aufwendigen Schema, anhand dessen zahlreichen Fehlergrößen, die hierfür zu Beginn des Kapitels definiert werden, eine detaillierte Analyse der durch die einzelnen Algorithmen erzeugten Objekthypothesen vorgenommen werden kann. In diesem Zusammenhang werden darauf aufbauend mögliche Ursachen für fehlerhafte Ergebnisse eruiert und positive Aspekte der einzelnen Verfahren herausgearbeitet.
Im Anschluss daran wird in Kapitel 5 mit der Gesten- und Aktionserkennung ein erstes Anwendungsfeld für die in den durch die Personenverfolgung ermittelten Aufenthaltsorte der Besprechungsteilnehmer untersucht. Ausgehend von den Positionen der Personen werden dazu Bewegungsmerkmale extrahiert. Um eventuell vorhandenen Störungseinflüssen zu begegnen, werden Systeme präsentiert, die eine Kompensation dieser Einflüsse auf die Erkennungsleistung ermöglichen.
Abschließend werden in Kapitel 6 die in dieser Arbeit erzielten Ergebnisse in einem kurzen Fazit nochmals zusammengefasst.

Kapitel 2

Grundlagen der Personenverfolgung

Die Verfolgung generell von Objekten (OT[1]) ist für eine Vielzahl von Aufgaben aus dem Bereich der Videoanalyse wie beispielsweise der automatischen, bildbasierten Überwachung, der Mensch-Maschine Interaktion oder der computergestützten Fahrzeugnavigation von fundamentaler Bedeutung (vgl. Yilmaz u. a. [121]): Erst durch Wissen über die Position von Objekten können anspruchsvollere Probleme wie eine Kollisionswarnung oder die automatische Erkennung von untypischen Situationen angegangen und gelöst werden. Für eine Vielzahl von Anwendungen spielt dabei der Mensch bzw. von ihm ausgeführte Aktionen – sei es nun im Dialog mit anderen Menschen oder etwa im Zuge der Bedienung von Maschinen – eine zentrale Rolle. Gerade aus diesem Grund konnte sich mit der *Personenverfolgung* eine eigene Disziplin innerhalb des weiten Forschungsbereiches des OT etablieren.
Typischerweise gliedern sich technische Systeme, die eine vollautomatische Lokalisation und Verfolgung von Personen leisten, dabei wiederum in eine Vielzahl einzelner, subsidiärer Algorithmen. In Abbildung 2.1 ist das grundlegende und im Kern so oftmals in der Literatur anzutreffende Aufbauprinzip eines Systems zur Personenverfolgung als Blockdiagramm skizziert. Beginnend mit einer Vorverarbeitung der Eingangsdaten werden in einem ersten Schritt Bildbereiche, die potenziell Kandidaten für die gesuchte Objektklasse enthalten könnten, aufgrund meist sehr einfach zu berechnender Merkmale vorab ermittelt. Durch die Personendetektion werden anschließend basierend auf einer entsprechenden Modellierung der Objektklasse Bildbereiche, die jeweils das gesuchte Objekt zeigen, segmentiert. Die aus dieser Detektionsstufe gewonnenen Erkenntnisse werden abschließend in der Trackinglogik ausgewertet, wodurch dann für jedes Objekt über den Zeitverlauf ein Bewegungspfad bestimmbar wird. Während einfachste

[1] Engl. „*object tracking*“; da sich mittlerweile auch im deutschen Sprachgebrauch für dieses Themenfeld die Bezeichnung *Tracking* etabliert hat, wird im weiteren Verlauf der Arbeit auch dieser Begriff als Synonym für *Verfolgung* benutzt.

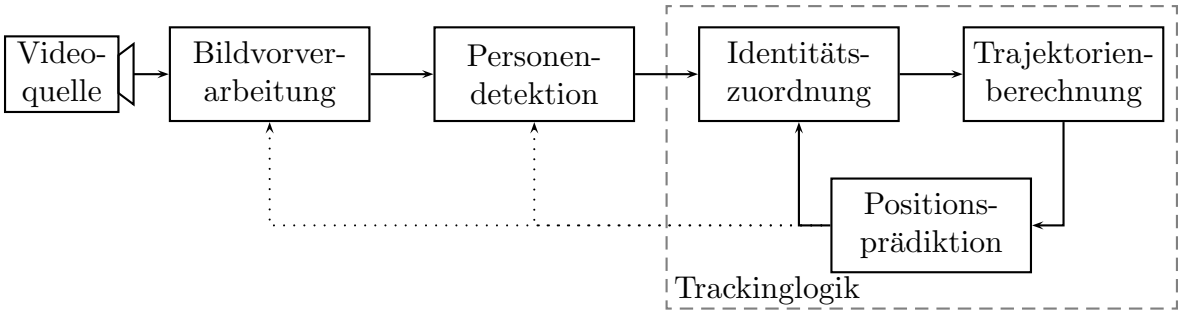


Abbildung 2.1 – Schematische Darstellung eines Systems zur automatischen Personenverfolgung.

Ansätze (vgl. Haritaoglu u. a. [41], Racine u. a. [80])[2] die aus dem vorhergehenden Bild stammende Objektinformation ausschließlich zur Bestimmung von Objektkorrespondenzen verwenden und die eigentliche Detektion ohne jegliches Vorwissen durchführen, nutzen andere Ansätze (vgl. Birchfield [17], Wren u. a. [117]) diese Information darüber hinaus sowohl für die Bildvorverarbeitung als auch die Personendetektion selbst (in Abbildung 2.1 durch die gepunktete Linie angedeutet), indem sämtliche innerhalb des Systems verfügbare bzw. in vorangegangenen Zeitschritten gewonnene Information als zusätzliche Wissensquelle zur Verfügung steht.
Bei dem in Abbildung 2.1 gezeigten Schaubild handelt es sich aber letztlich um eine sehr stark abstrahierte Darstellung, die insbesondere die Trackinglogik nur sehr allgemein durch die rudimentäre Andeutung der Funktionsblöcke zu erklären vermag. Konkret finden sich in der Literatur bei der technisch-algorithmischen Umsetzung von Trackingsystemen zwei grundsätzlich verschiedene Funktionsprinzipien wieder, die sich maßgeblich auf die Realisierung der Trackinglogik auswirken.
Die erste Gruppe bilden die datengetriebenen Ansätze, auch als bottom-up[3] Verfahren bezeichnet. Hierbei werden zunächst für das gesamte Bild nach einer Vorverarbeitung Merkmale berechnet, anhand derer mittels eines entsprechenden Modells Objekte detektiert werden. Aufgrund der für jedes Bild neuerlich durchzuführenden Merkmalsextraktion für die Personendetektionsstufe ist bei bottom-up Ansätzen jegliche Form von (Re-)Initialisierungsproblem a-priori ausgeschlossen. Jedoch erweist sich diese Art des Vorgehens als nachteilig in genau

[2]Das von Racine u. a. [80] beschriebene System wird allerdings nicht zur Verfolgung von Personen, sondern von fluoreszierenden Objekten verwendet.

[3]Bottom-up bezeichnet das diesen Ansätzen zugrunde liegende Prinzip, ausgehend von Rohdaten, also der untersten Ebene der Verarbeitungskette, durch eine immer feinere Betrachtung der Daten Information zu extrahieren.

solchen Fällen, in denen die Detektionsstufe mangelhaft arbeitet, da dann Fehler durch die gesamte Prozesskette propagiert werden. Im Gegensatz dazu berücksichtigen top-down Architekturen, die sich auf eine Abtastung des Bildraumes beschränken und damit auf Hypothesen für mögliche Objektkonstellationen basieren, zusätzlich den zeitlichen Kontext und ermöglichen dadurch auch bei zeitweise fehlerhaften Detektionen eine robuste Personenverfolgung, die allerdings aufgrund einer meist hohen Zahl an Hypothesen zu Lasten der Rechenzeit realisiert wird. Wegen der durchaus sehr vielfältigen Ausgestaltung der einzelnen Prozessschritte in der Literatur wird im Folgenden jeder der Blöcke aus Diagramm 2.1 vor dem Hintergrund des aktuellen Forschungsstandes detailliert diskutiert.

2.1 Vorverarbeitung

Gerade im Hinblick auf eine zeiteffiziente Realisierung eines Trackingsystems wird oftmals in einer vorgelagerten Stufe auf das gegebene Bild eine Vorverarbeitung angewandt. Diese hat zum Ziel, anhand geeigneter Merkmale all diejenigen Bereiche eines Bildes zu bestimmen, in denen sich aktuell keine Person aufhält. Grundgedanke dieses Vorgehens ist es, einerseits den Suchraum für den folgenden Detektionsschritt einzuschränken sowie andererseits Bereiche, die potentiell eine Quelle für mögliche Fehler durch die Personendetektion darstellen, zu eliminieren. Selbstverständlich können für diese Vorverarbeitung nur solche Merkmale des Bildes in Frage kommen, die aufgrund ihrer einfachen Berechenbarkeit die Systemressourcen nur mäßig beanspruchen und somit keine merkliche Zeitverzögerung in der Prozesskette verursachen[4]. Wegen der exponierten Position der Vorverarbeitung gleich zu Beginn der Videoanalyse und den dadurch bedingten Auswirkungen auf sämtliche folgenden Schritte liegt das Hauptaugenmerk bei der durch die Merkmale geleisteten Vorsegmentierung insbesondere auf einer hohen Verlässlichkeit, d.h. einer möglichst geringen Rate an fälschlicherweise vorab ausselektierten Personen, bei Vernachlässigung der sonst durchaus ebenso wichtigen Falsch-Positiv-Rate. Im Laufe der Entwicklung haben sich allgemein die Hautfarbe sowie die Segmentierung in Vorder- und Hintergrund als zwei sehr brauchbare Merkmale herauskristallisiert, die in nahezu jedem heutzutage veröffentlichten System zur Personenverfolgung genutzt oder sogar zwingend vorausgesetzt werden (vgl. Baumberg [12], Haritaoglu u. a. [41]).

[4] Man spricht daher auch oft von sog. *low-level* Merkmalen, wobei die Beurteilung, wann ein Merkmal als low-level zu bezeichnen ist, stark vom jeweiligen Kontext abhängt.

2.1.1 Detektion von Hautfarbe

Obwohl das Wissen über die Verteilung von hautfarbenen Bereichen in einem Bild nicht unmittelbar die Existenz einer Person nach sich ziehen muss[5] und damit als alleiniges Kriterium für eine Personendetektion ausscheidet, so bietet sich auf Grundlage dieses sehr einfachen Basismerkmals doch meist die Möglichkeit, die Effizienz von anspruchsvolleren Detektionsverfahren bezüglich Leistung und Ressourcen erheblich zu steigern. Alle in der Literatur beschriebenen Ansätze zur Detektion von hautfarbenen Bereichen in Bildern lassen sich nach Vezhnevets u. a. [109] grundsätzlich in pixel- und bereichsbasierte Verfahren unterteilen. Während die bereichsbasierten Techniken auch die räumliche Konstellation der Hautfarbenpixel mit betrachten und dadurch zwar einerseits eine bessere Detektionsrate erzielen, andererseits aber auch einen allgemein höheren Rechenaufwand mit sich bringen, modellieren pixelbasierte Methoden jeden Bildpunkt unabhängig von seiner Nachbarschaft. Da die Detektion von Hautfarbe aber nicht direkt zur Findung von Gesichtern, sondern letztlich nur unterstützend in Form einer Initialschätzung für die eigentliche Personendetektion angewandt wird, und somit nicht die Forderung nach einer exakten Identifizierung aller hautfarbenen Pixel erfüllt sein muss, werden zur Generierung des Basismerkmals in der Literatur oftmals pixelbasierte Segmentierungsverfahren aufgrund der effizienteren Berechnung bevorzugt. Grundsätzlich gliedert sich nach Kakumanu u. a. [53] die Problematik, hautfarbene Pixel von anderen zu unterscheiden, in zwei Teilbereiche: die Wahl sowohl des Farbraumes, welcher der Betrachtung zugrunde gelegt werden soll, als auch einer Klassifizierungsmethode, mit der Hautfarbe im gewählten Farbraum detektiert werden kann. Diese beiden Bereiche erweisen sich vor allem insofern als kritisch, da Hautfarbe wesentlich beeinflusst wird durch Beleuchtungsänderungen, Kameraparametrierung sowie personenspezifischen Eigenschaften, wie beispielsweise Alter oder Ethnizität. Es gilt daher, Hautfarbe einerseits durch geschickte Wahl eines geeigneten Farbraumes, andererseits mit Hilfe einer robusten Klassifizierungsmethode in möglichst generalisierter Form zu beschreiben, so dass der Einfluss vorherrschender Umgebungsbedingungen auf die Qualität der Detektion gemildert wird.

Wahl des Farbraumes

Die gemäß der Literatur (vgl. Kakumanu u. a. [53], Vezhnevets u. a. [109]) gebräuchlichsten Farbräume basieren auf einer Darstellung der Farbwerte mittels

[5] Wegen der großen Bandbreite, die unterschiedlichste Hauttypen im Farbspektrum einnehmen und daher modelliert werden müssen, können auch zahlreiche andere Objekte, die einen hautähnlichen Farbton aufweisen, mitunter als hautfarben erkannt werden.

RGB-Koordinaten, intensitätsnormalisierter rg-Chromawerte sowie HSV oder YC_rC_b Komponenten. Im Folgenden werden die Grundzüge dieser Farbräume vorgestellt und deren Eignung zur Detektion von Hautfarbe diskutiert.

RGB Der RGB-Farbraum bildet die native Darstellungsform von Farben im Bereich der digitalen Bildverarbeitung und entstammt der CRT-Monitortechnik, bei der Farben als Superposition des durch drei unterschiedliche Typen von Phosphor emittierten Lichtes entstehen. Wegen der starken Abhängigkeit von der Beleuchtung[6] eignet sich dieser Farbraum nur bedingt zur Modellierung von Hautfarbe, wird aber dennoch aufgrund der direkten Anwendbarkeit von Klassifikationsregeln auf den als RGB-Farbwerte unmittelbar vorliegenden Pixelwerten von einigen Autoren wie Wark u. Sridharan [114] oder auch Kovac u. a. [60] verwendet, um Hautfarbe in Bildern zu detektieren.

rg-Chroma Um die beim RGB-Farbraum störende, starke Beleuchtungsabhängigkeit der Merkmale zu mildern, werden die einzelnen Komponenten des Farbraumes durch die Intensität normiert:

$$r = \frac{R}{R+G+B} \qquad g = \frac{G}{R+G+B} \tag{2.1}$$

Für Betrachtungen in diesem Farbraum geht bedingt durch die Helligkeitsnormierung automatisch auch eine Dimensionsreduktion (Informationsverlust) einher, da anhand einfacher mathematischer Überlegungen sofort ersichtlich ist, dass sich die fehlende dritte Komponente (normierter Blaukanal) aus den beiden ersten berechnen läßt. In dem so erzeugten Chroma-Raum weisen hautfarbene Pixel weit weniger Varianz bei Veränderungen in der Beleuchtung oder auch in Bezug auf ethnische Eigenheiten auf, als dies beim originären RGB-Raum der Fall ist, weswegen sich dieser Farbraum in besonderer Weise zur Klassifizierung von Haut eignet und in zahlreichen Publikationen (vgl. z. B. Brown u. a. [20], Soriano u. a. [102], Stoerring u. a. [105]) Verwendung findet.

HSV Eine Variante, die sich an der perzeptiven Wahrnehmung von Farben orientiert, stellt der HSV-Farbraum[7] dar. Über eine nichtlineare Transformation

[6] Eine Helligkeitsänderung wirkt sich in diesem Farbraum auf alle drei Farbkanäle aus, wodurch eine robuste Detektion von Hautfarbe über einen großen Dynamikbereich der Beleuchtung nur begrenzt möglich ist.

[7] Die Bezeichnung HSV entstand aus der Abkürzung für die jeweiligen Komponenten Farbton (engl. „*hue*“), Sättigung (engl. „*saturation*“) und Helligkeit (engl. „*value*“).

werden hierbei die RGB-Werte übersetzt in einen Farbwinkel H, welcher die dominante Farbe angibt, die Sättigung S, welche die Ausgeprägtheit der Farbe repräsentiert, und die Helligkeit V, die gleichermaßen die Intensität widerspiegelt:

$$\begin{aligned} V &= \max(R,G,B) \\ S &= \begin{cases} (V - \min(R,G,B)) \cdot 255/V, & \text{wenn } V \neq 0 \\ 0, & \text{sonst} \end{cases} \\ H &= \begin{cases} (G-B) \cdot 60/S, & \text{wenn } V = R \\ 180 + (B-R) \cdot 60/S, & \text{wenn } V = G \\ 240 + (R-G) \cdot 60/S, & \text{wenn } V = B \end{cases} \end{aligned} \tag{2.2}$$

Der entscheidende Vorteil dieser Transformation liegt darin, dass die Werte H, S und V laut Skarbek u. Koschan [99] unempfindlich auf Glanzlicht oder ambiente Beleuchtung reagieren und daher einen geeigneten Farbraum zur Modellierung von Hautfarbe bilden. Aus diesem Grund greifen etliche Autoren (vgl. z. B. Wang u. Yuan [112], Zhu u. a. [125]) in ihren Arbeiten auf diesen Farbraum zur Detektion von Hautfarbe zurück.

YC_rC_b/YUV Ebenfalls auf den perzeptiven Eigenschaften beruht das YC_rC_b- sowie das YUV-Modell. Das RGB-Farbsignal, welches vorab mit einem exponentiellen Korrekturfaktor[8] beaufschlagt wird, läßt sich aufspalten in die Komponenten Luminanz Y und Chrominanz C_b und C_r bzw. U und V mit dem Ziel, die in den RGB-Farbkanälen enthaltene Redundanz zu vermindern:

$$\begin{pmatrix} Y \\ C_r \\ C_b \end{pmatrix} = \begin{pmatrix} 0,299 & 0,587 & 0,114 \\ 0,701 & -0,587 & -0,114 \\ -0,299 & -0,587 & 0,886 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad \text{bzw.} \tag{2.3}$$

$$\begin{pmatrix} Y \\ U \\ V \end{pmatrix} = \begin{pmatrix} 0,299 & 0,587 & 0,114 \\ -0,147 & -0,289 & 0,436 \\ 0,615 & -0,515 & 0,100 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \tag{2.4}$$

Die Farbinhalte werden durch die Chrominanzwerte kodiert, die Helligkeitsinformation entsprechend in der Luminanzkomponente. Aufgrund der

[8] Der Korrekturfaktor dient zur Berücksichtigung des nichtlinearen Helligkeitsempfindens des menschlichen Auges und wird gemeinhin als *Gammakorrektur* bezeichnet.

expliziten Aufteilung zwischen Luminanz und Chrominanz stellen diese Farbräume eine der bevorzugten Wahlen (vgl. Hsu u. a. [45], Phung u. a. [73], Wren u. a. [117]) für hautfarbenbasierte Ansätze dar.

Wahl der Klassifizierungsmethode

Eng mit der Wahl des Farbraumes ist die zur Detektion von Hautfarbe verwendete Modellierungs- und Klassifizierungsmethode verknüpft. Vezhnevets u. a. [109] sowie Kakumanu u. a. [53] fassen die am häufigsten in publizierten Ansätzen benutzten Strategien zur Klassifizierung von Bildpunkten zusammen.

Regelbasierte Modellierung Ein trotz seiner Einfachheit mit guten Ergebnissen überzeugender Ansatz basiert auf einer expliziten Beschreibung des hautfarbenen Bereiches im gewählten Farbraum mit Hilfe eines Regelwerkes. In der Praxis erweist es sich hierbei als schwierig, das für eine möglichst hohe Erkennungsrate notwendige, optimale Zusammenspiel zwischen einem geeigneten Farbraum einerseits, sowie den zu ermittelnden Begrenzungen des Hautfarbenbereiches andererseits, empirisch zu bestimmen. Aus diesem Grund wurden vor allem um die Jahrtausendwende mannigfaltige Ansätze publiziert, die sich auf jeweils unterschiedlichste Kombinationen aus Farbraum und den daraus zur Begrenzung herangezogenen Dimensionen stützen. Während nur vereinzelt Autoren (vgl. beispielsweise Kovac u. a. [60]) aufgrund der bereits angesprochenen Probleme dennoch eine Erkennung im RGB-Farbraum vollziehen, konzentriert sich die Mehrheit der Forscher bei den regelbasierten Verfahren auf die drei anderen im vorigen Abschnitt eingeführten Farbräume (vgl. Chai u. Ngan [23], Soriano u. a. [102], Wang u. Yuan [112]). Wegen der starren und generellen Beschreibungsform bietet sich dieser regelbasierte Klassifizierungsansatz vor allem in solchen Situationen an, in denen wenig Vorwissen über die konkreten Rahmenbedingungen, wie beispielsweise Beleuchtung oder dergleichen bekannt ist. Allerdings setzt diese Vorgehensweise stets eine präzise Farbkalibrierung des Kamerasystems voraus.

Histogrammbasierte Modellierung Bei diesem Ansatz wird der Farbraum in der Form quantisiert, dass für eine Vielzahl von Positivbeispielen, also Bildern, die ausschließlich hautfarbene Bildpunkte zeigen, die Zahl $N_{\text{pos}}(\vec{I})$ der jeweils vorkommenden Farbtupel[9] $\vec{I}$ in dem zugrunde liegenden Farbraum

[9] In der Praxis wird oftmals die Helligkeitskomponente (falls explizit durch eine eigene Größe beschrieben) vernachlässigt, wodurch statt des Farbtripels dann nurmehr der verbleibende Tupel zur Modellierung hautfarbener Pixel herangezogen wird.

in einer Histogramm-Matrix kumuliert werden. Durch anschließende Normierung mit der gesamten Zahl an Bildpunkten $N_{\text{Pix},1}$, die in der Matrix erfasst wurden, kann so mit der Gleichung

$$p(\vec{I}|\text{„Hautfarbe“}) = \frac{N_{\text{pos}}(\vec{I})}{N_{\text{Pix},1}} \tag{2.5}$$

die Wahrscheinlichkeit dafür berechnet werden, dass ein Bildpunkt mit dem Farbtupel $\vec{I}$ einen hautfarbenen Pixel markiert. Über eine binäre Schwellwertentscheidung können so für $p(\vec{I}) \geq \Theta$ hautfarbene Bereiche in Bildern detektiert werden.
Unter Berücksichtigung der a-priori Wahrscheinlichkeit $p(\vec{I})$ jedes hautfarbenen Pixels läßt sich eine weitere Steigerung der Klassifikationsleistung erreichen. Dazu wird in einem weiteren Histogramm, ähnlich wie bereits für die Positivbeispiele geschehen, die Zahl $N_{\text{neg}}(\vec{I})$ aller $N_{\text{Pix},2}$ Bildpunkte in Negativbeispielen für alle Farbtupel $\vec{I}$ erfasst. Über die beiden Wahrscheinlichkeiten

$$p(\vec{I}|\text{„Hautfarbe“}) = \frac{N_{\text{pos}}(\vec{I})}{N_{\text{Pix},1}} \tag{2.6}$$

$$p(\vec{I}|\overline{\text{„Hautfarbe“}}) = \frac{N_{\text{neg}}(\vec{I})}{N_{\text{Pix},2}} \tag{2.7}$$

kann mit Hilfe eines Naive-Bayes-Klassifikators ein Bildpunkt genau dann als hautfarben kategorisiert werden, wenn dessen Farbtupel die Bedingung

$$\frac{p(\vec{I}|\text{„Hautfarbe“})}{p(\vec{I}|\overline{\text{„Hautfarbe“}})} \geq \Theta \tag{2.8}$$

erfüllt. Der extrinsische Parameter Θ kann hierbei je nach Anforderung über eine ROC-Kurve[10] eingestellt werden. Insbesondere die Generierung der beiden Histogramm-Matrizen kann mitunter sehr viel Zeit in Anspruch nehmen, da – um einen repräsentativen Querschnitt aller in realen Bildern auftauchenden Farbtupel zu erhalten – eine große Zahl an Beispielen vonnöten ist. Sobald jedoch diese Matrizen vorliegen, können unbekannte Pixel über das Auslesen nur zweier Tabelleneinträge sehr zeiteffizient klassifiziert werden.

[10] Engl. „*receiver operator characteristic*“; bei dieser Kurve wird durch Variation eines Parameters, im vorliegenden Fall der Variablen Θ, die Zahl der korrekterweise als Treffer klassifizierten Bildpunkte über den fälschlicherweise als Treffer klassifizierten Bildpunkten angetragen.

Gaußmodelle Empirische Untersuchungen von Yang u. a. [119] haben gezeigt, dass im normalisierten rg-Farbraum Hautfarbe über Gauß'sche Verteilungskurven approximiert werden kann. Dies bietet den Vorteil, dass mit nur wenigen Parametern bei geringem Speicherbedarf eine schnelle und vor allem generalisierte Erkennung von hautfarbenen Pixeln erfolgen kann. Für einfach gestaltete Hintergrundszenarien mit nur wenig Beleuchtungsschwankung genügt hierbei zur Modellierung des Hautfarbenbereiches oftmals eine einzige zweidimensionale Gaußverteilung

$$p(\vec{I}|\text{„Hautfarbe“}) = \frac{1}{2\pi\sqrt{|\underset{\sim}{\Sigma}|}} \exp\left(-\frac{1}{2}(\vec{I}-\vec{\mu})^T \underset{\sim}{\Sigma}^{-1}(\vec{I}-\vec{\mu})\right), \tag{2.9}$$

wobei $\vec{\mu}$ den Mittelwert und $\underset{\sim}{\Sigma}$ die Kovarianzmatrix aller hautfarbenen Pixel darstellt. Basierend auf dieser Wahrscheinlichkeitsverteilung kann wiederum über eine Schwellwertentscheidung eine entsprechende Klassifizierung eines vorliegenden Farbtupels vorgenommen werden (vgl. Hsu u. a. [45]). Insbesondere für anspruchsvolle Umgebungsbedingungen mit wechselnden Lichtverhältnissen kann die Annahme einer unimodalen Verteilung von Hautfarbe nicht mehr aufrechterhalten werden. Aus diesem Grund wird oftmals ein erweiterter Ansatz basierend auf einer Modellierung mittels Gaußmixturen benutzt. Die dadurch beschriebene Verteilungsfunktion gestaltet sich als gewichtete Superposition von N_{Mix} einzelnen Gaußkurven $\mathcal{N}_i(\vec{\mu}_i, \underset{\sim}{\Sigma}_i)$ gemäß der Gleichung

$$p(\vec{I}|\text{„Hautfarbe“}) = \sum_{i=1}^{N_{\text{Mix}}} w_i \frac{1}{2\pi\sqrt{|\underset{\sim}{\Sigma}_i|}} \exp\left(-\frac{1}{2}(\vec{I}-\vec{\mu}_i)^T \underset{\sim}{\Sigma}_i^{-1}(\vec{I}-\vec{\mu}_i)\right). \tag{2.10}$$

Die Parameter $\vec{\mu}_i, \underset{\sim}{\Sigma}_i$ und w_i der Gaußmixturen können iterativ mit Hilfe des Expectation-Maximization (EM) Algorithmus (vgl. Bilmes [15], Yang u. Ahuja [120]) ermittelt werden. Die hierfür notwendige Initialschätzung kann dabei beispielsweise durch ein k-Means Clusterverfahren nach Kanungo u. a. [56] bereitgestellt werden. In der Praxis gestaltet sich üblicherweise die Bestimmung der optimalen Zahl zu verwendender Gaußmixturen als überaus schwierig und kann nicht allgemeingültig gelöst werden. So bewegt sich diese Zahl bei den in der Literatur zu findenden Ansätzen im Bereich von $N_{\text{Mix}} = 2$ (vgl. Yang u. Ahuja [120]) bis hin zu $N_{\text{Mix}} = 16$ Mixturen (vgl. Jones u. Rehg [52]). Durch einen Schwellwertvergleich läßt sich auch hier erneut die Klassifikation eines Farbtupels anhand der multivariaten Verteilungsdichtefunktion $p(\vec{I}|\text{„Hautfarbe“})$ erreichen.

2.1.2 Hintergrundsegmentierung

In zahlreichen Systemen zur Objektverfolgung bildet die Segmentierung von Vordergrundobjekten und Bildhintergrund den Beginn der Verarbeitungskette (vgl. Baumberg [12], Haritaoglu u. a. [41]), da sich hierdurch bereits in einem sehr frühen Stadium Bereiche des Bildes, die im Zuge des OT als uninteressant erachtet werden können, feststellen und ausblenden lassen, wodurch sich die weiteren Verarbeitungsprozesse effizienter gestalten lassen. Gerade aber aufgrund ihres frühen Eingreifens in die Bildanalyse, verbunden mit den unmittelbaren Auswirkungen auf die Ergebnisse der folgenden Prozessschritte, kommt der Wahl des entsprechenden Modellierungsverfahrens eine gewichtige Bedeutung zu. Hierbei sind allgemein von den Verfahren eine hohe Adaptionsgeschwindigkeit an plötzlich auftretende Veränderungen im Bild bei einer gleichzeitig qualitativ möglichst hochwertigen Segmentierungsleistung, speziell auch im Hinblick auf sich nur sehr langsam bewegende oder gar unbewegte Vordergrundobjekte[11], zu fordern. Prinzipiell zerfällt die Aufgabe der Segmentierung zwischen Vorder- und Hintergrund in zwei Teilbereiche (vgl. Abbildung 2.2): Die eigentliche Modellierung des Hintergrundes sowie die darauf aufsetzende, konkrete Bestimmung von Vordergrundbereichen. In den letzten beiden Jahrzehnten wurden hierzu mehrere integrierte Ansätze entwickelt, die teilweise beide Bereiche kombiniert betrachten.

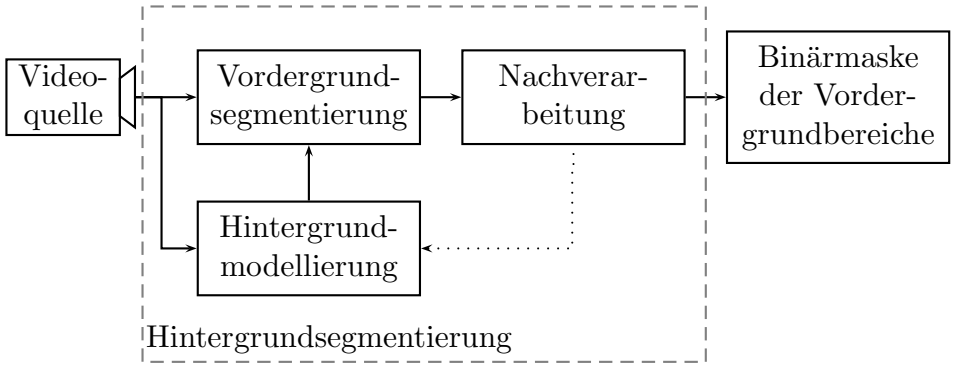


Abbildung 2.2 – Blockdiagramm allgemein für die Bildhintergrundsubtraktion, angelehnt an die Darstellung in Cheung u. Kamath [24].

[11] In der Praxis besteht bei der Hintergrundmodellierung die Problematik, dass Vordergrundobjekte, die sich nur langsam oder gar nicht bewegen, oftmals über den Zeitverlauf in den Hintergrund übernommen werden.

Nicht-rekursiver zeitlicher Mittelwert Cucchiara u. a. [28] verwenden zur Modellierung des Bildhintergrundes in ihren Veröffentlichungen eine zeitliche Mittelwertbildung

$$\underset{\sim}{\mu}_t = \sum_{\tau=1}^{N_\mathrm{B}} \underset{\sim}{I}_{t-\tau}, \tag{2.11}$$

über die letzten N_B Bilder einer Sequenz. Durch Subtraktion des so berechneten Hintergrundes vom aktuellen Eingangsbild $\underset{\sim}{I}_t$ entsteht ein Differenzbild $\underset{\sim}{D}_t = \underset{\sim}{I}_t - \underset{\sim}{\mu}_t$, aus dem nach einer Schwellwertoperation und evtl. folgenden morphologischen Operatoren[12] Bereiche resultieren, die dann potentielle Vordergrundobjekte darstellen. Obwohl diese Methode sehr intuitiv erscheint und bedingt durch die einfachen Operationen sehr zeiteffizient eingesetzt werden kann, so gestaltet sich zum einen die Wahl eines geeigneten Schwellwertes sehr schwierig und zum anderen zeigen sich gravierende Mängel insbesondere im Hinblick auf die Adaptionsgeschwindigkeit bzw. die Segmentierungsleistung[13].

Rekursive zeitliche Mittelwertbildung Im Gegensatz zu obigem Ansatz mindern die Ideen von Wren u. a. [117] sowie Koller u. a. [59] die Anforderungen an die Systemressourcen insofern, als dass sie den aktuell benötigten Mittelwert $\underset{\sim}{\mu}_t$ iterativ berechnen:

$$\underset{\sim}{\mu}_t \;=\; \alpha \underset{\sim}{I}_t + (1-\alpha)\underset{\sim}{\mu}_{t-1} \tag{2.12}$$

Zwar wird hierdurch nicht mehr nur die kürzere Vergangenheit der Bildsequenz repräsentiert, allerdings läßt sich der Einfluss sehr weit zurückliegender Bilder auf den aktuellen Mittelwert durch die Lernrate α entsprechend justieren. Auch bei diesem Ansatz wird prinzipiell die Entscheidung, welche Bereiche des aktuellen Bildes den Hintergrund darstellen, über eine Differenz $\underset{\sim}{D}_t = \underset{\sim}{I}_t - \underset{\sim}{\mu}_t$ getroffen. Zur Bestimmung eines geeigneten Schwellwertes wird in Wren u. a. [117] vorgeschlagen, neben dem Mittelwert $\underset{\sim}{\mu}_t$ auch die Varianz $\underset{\sim}{\Sigma}_t = \underset{\sim}{\mathbb{1}}\vec{\sigma}_t^2$ analog zu Gleichung 2.12 zu bestimmen. Anhand dieser Information lassen sich jeweils situationsbezogen durch die Bedingung $|\underset{\sim}{D}_t(\vec{p})| > |\kappa\vec{\sigma}_t|$ all diejenigen Pixel $\vec{p} = (x, y)^T$ im Differenzbild $\underset{\sim}{D}_t$ abhängig von einer Proportionalitätskonstanten κ ermitteln, die nicht zum Hintergrund zählen. Obwohl bei dieser Methode die Wahl zweier Parameter

[12] Durch morphologische Grundoperatoren wie beispielsweise Dilatation oder Erosion läßt sich eine glättende Wirkung auf dem zugrunde liegenden Binärbild erzielen.

[13] Je nach der Anzahl an Bildern N_B, über die gemittelt wird, kann entweder die Adaptionsgeschwindigkeit oder die Segmentierungsleistung optimiert werden, jedoch immer zu Lasten des jeweils anderen Kriteriums.

(Schwellwert und Länge der zeitlichen Mittelwertbildung) obsolet wird, so besteht trotzdem nach wie vor speziell bei raschen Änderungen im Bild abhängig von der Lernrate α die Problematik einer verlangsamten Adaption des Bildhintergrundes und daraus resultierend die Gefahr, durch die zeitliche Mittelwertbildung den tatsächlichen Bildhintergrund mit den bisher vorgestellten Methoden nicht mehr modellieren zu können.

Gauß-Mixtur-Modelle Speziell bei nicht mehr unimodalen Hintergrundstrukturen zeigen die bisher genannten Verfahren Schwächen. Um diese zu umgehen, verwenden andere Ansätze Gauß-Mixtur-Modelle (GMM), wie sie bereits im Rahmen der Hautfarbendetektion vorgestellt wurden, zur Modellierung des Bildhintergrundes (vgl. Power u. Schoonees [77], Stauffer u. Grimson [103]). Über in der Regel zwischen $N_{\text{Mix}} \in \{3, 4, 5\}$ gewichtete Gaußkurven $\mathcal{N}_i(\vec{\mu}_{t,i}, \underset{\sim}{\Sigma}_{t,i})$ wird für jeden Bildpunkt $\vec{p} = (x, y)^T$ durch

$$p(\vec{I}_t(\vec{p})|\mathcal{N}) = \sum_{i=1}^{N_{\text{Mix}}} w_{t,i} \frac{\exp\left(-\frac{1}{2}(\vec{I}_t(\vec{p}) - \vec{\mu}_{t,i})^T \underset{\sim}{\Sigma}_{t,i}^{-1} (\vec{I}_t(\vec{p}) - \vec{\mu}_{t,i})\right)}{\sqrt{(2\pi)^d |\underset{\sim}{\Sigma}_{t,i}|}} \tag{2.13}$$

eine Verteilung beschrieben, mit deren Hilfe eine Wahrscheinlichkeit dafür angegeben werden kann, dass der betreffende Bildpunkt den d-dimensionalen Wert $\vec{I}_t(\vec{p})$ annimmt. Durch den EM-Algorithmus lassen sich die notwendigen Parameter jeweils in einer iterativen Prozedur (vgl. Power u. Schoonees [77]) gemäß den Gleichungen[14]

$$w_{t,i} = (1 - \alpha) w_{t-1,i} + \alpha p(i|\vec{I}_t(\vec{p}), \mathcal{N}) \tag{2.14}$$

$$\vec{\mu}_{t,i} = (1 - \rho_{t,i}) \vec{\mu}_{t-1,i} + \rho_{t,i} \vec{I}_t(\vec{p}) \tag{2.15}$$

$$\underset{\sim}{\Sigma}_{t,i} = (1 - \rho_{t,i}) \underset{\sim}{\Sigma}_{t-1,i} + \mathbb{1} \rho_{t,i} \left(\left(\vec{I}_t(\vec{p}) - \vec{\mu}_{t,i} \right) \circ \left(\vec{I}_t(\vec{p}) - \vec{\mu}_{t,i} \right) \right) \tag{2.16}$$

mit einer Lernrate α bestimmen, wobei gilt:

$$\rho_{t,i} = \frac{1}{w_{t,i}} \alpha p(i|\vec{I}_t(\vec{p}), \mathcal{N}). \tag{2.17}$$

Zur rechenzeiteffizienten Abschätzung der grundsätzlich benötigten Wahrscheinlichkeit $p(i|\vec{I}_t(\vec{p}), \mathcal{N})$ kann dabei folgende Approximation, die durch

[14] Selbstverständlich handelt es sich bei den zu berechnenden Werten $w_{t,i}, \vec{\mu}_{t,i}$ und $\underset{\sim}{\Sigma}_{t,i}$ um jeweils ortsabhängige Größen. Lediglich aus Gründen der besseren Lesbarkeit wurde in den Formeln 2.14-2.16 auf die explizite Kennzeichnung dieser Ortsabhängigkeit verzichtet.

empirische Beobachtungen von Stauffer u. Grimson [103] legitimiert wird, benutzt werden:

$$p(i|\vec{I}_t(\vec{p}),\mathcal{N}) \approx \begin{cases} 1 & \text{wenn } ||\Sigma_{t,i}^{-1}\left(\vec{I}_t(\vec{p}) - \vec{\mu}_{t,i}\right)||_2 < 2,5 \\ 0 & \text{sonst} \end{cases} \tag{2.18}$$

Um der Stochastizitätsbedingung $\sum_{i=1}^{N_{\text{Mix}}} w_{t,i} = 1,\ \forall\, t$ zu genügen werden die Gewichte der Gaußkurven normiert. Prinzipiell wird durch die Gaußkurven bzw. konkreter durch die dadurch beschriebenen Verteilungen nicht zwischen Vorder- und Hintergrund unterschieden. Aufgrund der spezifischen Eigenschaften des Hintergrundes, dass dieser üblicherweise einerseits häufiger im Bild zu sehen ist als Objekte im Vordergrund und damit Gaußkurven mit besonders hohen Gewichten $w_{t,i}$ produziert, andererseits wenig Änderung[15] aufweist und dadurch die zum Hintergrund gehörenden Bildpunkte eine geringe Varianz $\vec{\sigma}_{t,i}^2 = \text{diag}(\underset{\sim}{\Sigma}_{t,i})$ zeigen, kann nach Power u. Schoonees [77] eine Kategorisierung zwischen Vorder- und Hintergrund vorgenommen werden, indem die Gaußkurven $\mathcal{N}_i$ bezüglich deren Verhältnis $\frac{w_{t,i}}{||\vec{\sigma}_{t,i}||_2}$ angeordnet werden und anschließend diejenigen $N_{\text{Mix}_{\text{eff}},t}$ Gaußkurven als den Hintergrund beschreibend erachtet werden, welche die Bedingung

$$N_{\text{Mix}_{\text{eff}},t} = \underset{1\leq N\leq N_{\text{Mix}}}{\text{argmin}} \left(\sum_{i=1}^{N} w_{t,i} > \Theta\right) \tag{2.19}$$

erfüllen, also deren Gewichte eine vorgegebene Schwelle Θ kumulativ überschreiten.

Kernelbasierte Dichteschätzung Untersuchungen von Elgammal u. a. [31] haben gezeigt, dass sich insbesondere für sehr frequente Variationen im Hintergrund die Wahrscheinlichkeitsverteilung für die d-dimensionalen Werte $\vec{I}_t(\vec{p})$, die ein betrachteter Bildpunkt annimmt, sehr schnell über den zeitlichen Verlauf ändert. Um diesem Umstand gerecht werden zu können, wird der gerade vorgestellte GMM-Ansatz verallgemeinert, indem nicht mehr jeder Bildpunkt über die zeitliche Vergangenheit eine Wahrscheinlichkeitsverteilung aufspannt, sondern sich diese zu jedem Zeitpunkt als Summation einer über jeweils einen fixen zeitlichen Bereich N_{B} zurückreichenden Kernelfunktion ausdrücken läßt. Hierzu wird in der Literatur (vgl. Elgammal

[15] Diese Bedingung trifft insbesondere auf die in dieser Arbeit betrachteten Innenraum-Umgebungen zu.

u. a. [31]) für die Kernelfunktion oftmals eine Normalverteilung gewählt. Auf diese Weise läßt sich durch die Gleichung

$$p(\vec{I}_t(\vec{p})) = \frac{1}{N_\mathrm{B}} \sum_{i=1}^{N_\mathrm{B}} \prod_{j=1}^{d} \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp\left(-\frac{(\vec{I}_t(\vec{p}) - \vec{I}_{t-i}(\vec{p}))^2}{2\sigma_j^2} \right) \tag{2.20}$$

die Wahrscheinlichkeit dafür berechnen, dass der betrachtete Bildpunkt $\vec{I}_t(\vec{p})$ zum Hintergrundbereich zählt. Die hierfür benötigte Kernelbandbreite

$$\sigma_j = \frac{m_j}{0,68\sqrt{2}} \tag{2.21}$$

wird für jeden Farbkanal j basierend auf dem Median m_j über die paarweisen Betragsdifferenzen $||\vec{I}_t(\vec{p}) - \vec{I}_{t-1}(\vec{p})||_2$ für die letzten N_B Bilder ermittelt. Wie Elgammal u. a. [31] in ihrer Publikation zeigen, läßt sich durch diesen Modellierungsansatz gegenüber GMM bei gleicher Falsch-Positiv-Rate die Detektionsrate bereits bei mäßigen Kontrastwerten zwischen Vorder- und Hintergrund signifikant steigern.

Kalman-Filter Ein anderer Ansatz, der sich auf die Annahme einer unimodalen Verteilung von Intensitätswerten stützt, nutzt die theoretischen Grundlagen des Kalman-Filters, wie sie in Anhang D beschrieben sind, zur Modellierung des Bildhintergrundes. Der Systemzustand ist bei Ridder u. a. [82] gegeben als Vektor, der zum einen den geschätzten Intensitätswert $\hat{\underset{\sim}{G}}_t(\vec{p})$ und zum anderen dessen zeitliche Ableitung $\hat{\dot{\underset{\sim}{G}}}_t(\vec{p})$ umfasst; den Messwert bildet der zum Zeitpunkt t tatsächlich vorliegende Intensitätswert $\underset{\sim}{G}_t(\vec{p})$ des betrachteten Bildpunktes an Position $\vec{p}$. Damit kann das dynamische Modell durch die Gleichung

$$\begin{bmatrix} \hat{\underset{\sim}{G}}_t(\vec{p}) \\ \hat{\dot{\underset{\sim}{G}}}_t(\vec{p}) \end{bmatrix} = \underset{\sim}{\underline{A}} \begin{bmatrix} \hat{\underset{\sim}{G}}_{t-1}(\vec{p}) \\ \hat{\dot{\underset{\sim}{G}}}_{t-1}(\vec{p}) \end{bmatrix} + \underset{\sim}{\underline{K}}_t(\vec{p}) \left(\underset{\sim}{G}_t(\vec{p}) - \underset{\sim}{\underline{H}}\,\underset{\sim}{\underline{A}}\,\hat{\underset{\sim}{G}}_{t-1}(\vec{p}) \right) \tag{2.22}$$

beschrieben werden, wobei die Matrix $\underset{\sim}{\underline{A}}$ die Systemdynamik und $\underset{\sim}{\underline{H}}$ die Messmatrix repräsentiert. Der Kalman Gain $\underset{\sim}{\underline{K}}_t(\vec{p}) = (\varsigma, \varsigma)^T$ dient zur Regelung der Adaptionsgeschwindigkeit des Hintergrundmodells. Um ein allzu schnelles Adaptieren des Hintergrundes an statische Vordergrundobjekte zu verhindern, wird hierbei der Parameter ς je nach aktueller Zugehörigkeit des Pixels zu Vorder- oder Hintergrund unterschiedlich gewählt

$$\varsigma = \begin{cases} \alpha_1 & \text{wenn Pixel } \in \text{ Vordergrund, d. h. } |G_t(\vec{p}) - \hat{G}_t(\vec{p})| > \Theta \\ \alpha_2 & \text{sonst} \end{cases}, \tag{2.23}$$

wobei $\alpha_1 < \alpha_2$ gilt.

Nachverarbeitung

Für alle vorgestellten Techniken gilt, dass sie nahezu dieselben Unzulänglichkeiten aufweisen: Zum einen wird, wie unmittelbar anhand des jeweiligen Modellierungsansatzes ersichtlich ist, jeder Pixel unabhängig von möglicherweise vorhandenen räumlichen Korrelationen betrachtet. Dies bedeutet, dass benachbarte Bildpunkte unabhängig voneinander zu Vorder- oder Hintergrund gehören können, was in der Praxis zu zufällig verstreuten punktuellen Fehlern in der binären Hintergrundmaske führt. Um dies zu vermeiden, kann durch morphologische Operationen dieser Nachbarschaftskontext nachträglich noch rudimentär berücksichtigt werden, indem isolierte Pixel innerhalb einer geschlossenen Fläche eliminiert werden. Zum anderen stellt der Schattenwurf von (bewegten) Objekten häufig ein Problem dar, da dieser je nach Situation die gleiche Gestalt haben kann, wie das verursachende Objekt selbst und damit für konturbasierte Detektionsmethoden mitunter Schwierigkeiten verursachen kann. Aus diesem Grund widmen sich zahlreiche Publikation auch der Detektion von Schatten[16]. Letztgenanntes Problem ist jedoch in wesentlichem Umfang nur bei Szenarien in Außenbereichen von Belang, so dass auf eine nähere Betrachtung von Schatteneffekten in den gut ausgeleuchteten Besprechungsräumen verzichtet werden kann.

2.2 Personendetektion

In seiner einfachsten Form wird durch das OT nur die Trajektorie, also die Position eines Objektes über den zeitlichen Verlauf, bereitgestellt. Abhängig von der konkreten Aufgabenstellung kann darüber hinaus jedoch auch zusätzliche Information wie z. B. die Silhouette des Objektes oder dessen Orientierung ermittelt werden. Grundlage hierfür ist eine entsprechende Repräsentation des zu verfolgenden Objektes, die gemeinhin wahlweise auf dessen Kontur oder aber den ansichtsbasierten[17] Eigenschaften beruht. Yilmaz u. a. [121] fassen dabei die in der Literatur gängigen Strategien zur Repräsentation von Objekten speziell im Rahmen der Personenverfolgung, wie in Tabelle 2.1 dargestellt, zusammen. Die jeweils geeignete Repräsentation für ein Anwendungsszenario hängt dabei einerseits ab von dem zu verfolgenden Objekt selbst (z. B. formveränderlich oder starr), anderseits aber auch von den zu erwartenden äußeren Rahmenbedingungen, unter welchen OT eingesetzt werden soll, insbesondere z. B. von der Qualität

[16] Eine gute Zusammenfassung hierzu liefert die Publikation von Prati u. a. [78].

[17] Unter ansichtsbasierten Eigenschaften eines Objektes wird die gleichzeitige Nutzung der Information über Kontur sowie Textur verstanden.

des optischen Sensors, dem Auftreten von Beleuchtungsschwankungen oder der Tatsache, dass Objekte durch andere Gegenstände verdeckt werden. Für Besprechungsszenarien erscheint der häufig gewählte Ansatz, den Menschen von Kopf bis Fuß zu modellieren, nur bedingt praktikabel, da sich in realen Besprechungen Personen überwiegend in der Nähe von bzw. direkt an Tischen aufhalten und daher bei einer vertretbaren Zahl an Kameras oftmals nur ab der Hüfte aufwärts erfasst werden. Aus diesem Grund war in der vorliegenden Arbeit Ziel des OT der Kopf als derjenige Teil des Menschen, der am wahrscheinlichsten bei physikalischer Anwesenheit der Person im Besprechungsraum auch in der Kameraperspektive sichtbar ist.
In engem Bezug zur Objektrepräsentation steht die Wahl passender Merkmale, anhand derer eine eindeutige Separierung zwischen Objektklasse und dem restlichen Merkmalsraum getroffen werden kann. Yilmaz u. a. [121] identifizierten dabei in der Literatur vier grundsätzliche und häufig benutzte Basismerkmale:

Farbe Aufgrund der Möglichkeit einer einfachen Bestimmung stellt Farbe ein sehr beliebtes Merkmal dar. Die Farbe eines Objektes ist dabei im Wesentlichen durch zwei physikalische Faktoren, nämlich die spektrale Leistungsdichte des Strahlers und die Oberflächenbeschaffenheit des Objektes, maßgeblich festgelegt. Je nach Aufgabenstellung bieten sich hierbei unterschiedliche Farbräume an mit jeweils individuellen Eigenschaften, wie beispielsweise eine erhöhte Robustheit gegenüber Beleuchtungsschwankungen oder eine physiologisch bessere Modellierung der menschlichen Farbwahrnehmung.

Kanten Information über eine mögliche räumliche Begrenzung von Objekten wird offenbart durch Kanten. Insbesondere bei Konturmodellen dienen Kanten als unerlässliches Merkmal zur Objektbeschreibung. Ein wesentlicher Vorteil von kantenbasierten Merkmalen liegt vor allem in der Tatsache, dass diese weit weniger anfällig auf Beleuchtungsschwankungen reagieren als beispielsweise Farbmerkmale.

Optischer Fluss Ein Vektorfeld, welches die 2D-Projektion der Bewegungsrichtung und -geschwindigkeit für sämtliche Pixel zweier aufeinanderfolgender Bilder einer Videosequenz wiedergibt, wird als optischer Fluss bezeichnet. Grundlegende Annahme hierfür ist die Beibehaltung der Helligkeit eines Pixels in den zwei betrachteten Bildern, für die der optische Fluss berechnet werden soll (vgl. Horn u. Schunck [44]). Aufgrund dieser Annahme ist es unmittelbar klar, dass diese Art von Merkmal auf Variation der Beleuchtung sehr empfindlich reagiert.

Objektrepräsentation	Erklärung
Punkte	Ein Objekt wird repräsentiert durch einen einzelnen Schwerpunkt oder aber durch einen Satz von (aussagekräftigen) Punkten innerhalb des Objektes.
Einfache geometrische Strukturen	Wird ein Objekt durch geometrische Primitive wie beispielsweise Rechtecke oder Ellipsen approximiert, können dadurch dessen Ausmaße mit erfasst werden.
Silhouette und Kontur	Ein Objekt wird hier maßgeblich durch seinen Rand charakterisiert, der wahlweise durch eine kontinuierliche Begrenzungslinie oder diskrete Abtaststellen modelliert wird. Hieraus kann die durch das Objekt belegte Fläche (Silhouette) abgeleitet werden, die dann ebenso als mögliche Repräsentation dienen kann.
Zusammengesetzte Modelle (Articulated Models)	Das Objekt wird aufgefasst als Aneinanderreihung einzelner Körperteile, wie beispielsweise Arme, Beine und Torso. Diese Bestandteile selbst können dann wiederum auf Basis von geometrischen Strukturen modelliert werden und als kinematische Kette zulässige Bewegungen von Gliedmaßen erfassen.
Skelettmodelle	Die Gliedmaßen eines Objektes werden reduziert auf Linienstücke, die dann eine kinematische Kette bilden und damit Formveränderungen des Objektes durch unterschiedliche Lagebeziehungen der Linienstücke erlauben.
Wahrscheinlichkeitsverteilung der Textur	Die Textur des Objektes wird entweder durch parametrische Wahrscheinlichkeitsverteilungen wie beispielsweise Gauß-Mixtur-Modelle oder durch nichtparametrisierte Modelle basierend auf z. B. Histogrammen approximiert.
Prototypen (Templates)	Objekte werden hierbei durch Positivbeispiele repräsentiert. Basierend auf den Gemeinsamkeiten in diesen Beispielen wird ein Prototyp erzeugt.
Active Appearance Modelle (AAM)	Objekte werden ganzheitlich beschrieben durch gleichzeitige Einbeziehung von Textur- und Konturinformation. Ähnlich wie bei den Prototypen werden anhand von Positivbeispielen objektspezifische Eigenheiten erlernt. Allerdings erlaubt die Repräsentation mit Active Appearance Modellen aufgrund der statistischen Modellierung eine größere Objektvielfalt.

Tabelle 2.1 – Generelle Möglichkeiten zur Repräsentation von Objekten, angelehnt an die Darstellung von Yilmaz u. a. [121].

Textur Die Textur beschreibt die Oberflächeneigenschaften eines Objektes. Sie stellt bezüglich Beleuchtungsänderungen ein nahezu ebenso robustes Merkmal dar wie Kanten. Ähnlich wie der optische Fluss findet auch dieses Merkmal überwiegend Anwendung bei ansichtsbasierten Techniken zur Objektbeschreibung, wird jedoch in der jüngeren Literatur nur mehr selten erwähnt.

Eine abgeschlossene Objektbeschreibung mittels der gerade beschriebenen Merkmale ist nur in Ausnahmefällen möglich. Daher werden zusätzlich meist komplexere Merkmale benutzt, um damit einerseits die Detektionsrate weiter zu erhöhen und andererseits die Zahl der fälschlicherweise als Objekt detektierten Bildbereiche zu minimieren. Auch diese Merkmale müssen wiederum solche Eigenschaften des Objektes beschreiben, für die eine möglichst zuverlässige Identifikation als Objekt im Merkmalsraum gewährleistet werden kann. Ziel zahlreicher Ansätze ist es daher, in einem automatisierten Prozess derartige Merkmale zu bestimmen. Dies kann zum einen mit Hilfe extern vorgegebener Forderungen nach beispielsweise einer möglichst geringen Korrelation verschiedener Merkmale geschehen, wie dies z. B. im Zuge der Gesichtsdetektion mittels der Hauptkomponentenanalyse bei Menser u. Muller [68] umgesetzt wurde, zum anderen aber auch datengetrieben in einem „Black-Box"-Verfahren[18] vonstatten gehen.
In den folgenden beiden Abschnitten werden exemplarisch zwei Verfahren vorgestellt, mit denen Personen in Bildern detektiert werden können und die sich in der Forschungsgemeinschaft als Stand der Technik etablieren konnten.

2.2.1 Gesichtsdetektion mittels Neuronaler Netze

Einen bedeutenden Ansatz, der sehr erfolgreich auf das Problem der Gesichtsdetektion angewandt wurde und mittlerweile als eines der Standardverfahren für diese Aufgabenstellung erachtet werden kann, stellt das von Rowley u. a. [86] im Jahre 1998 veröffentlichte Verfahren basierend auf Neuronalen Netzen (NN) dar.

Vorverarbeitung

Da bei dem Verfahren nach Rowley unmittelbar die Grauwertinformation des Bildes zur Detektion eines Gesichtes benutzt wird, bedarf es zum Ausgleich beeinflussender Faktoren wie Kontrast oder Helligkeit einer vorhergehenden Aufbereitung der zu untersuchenden Bilddaten. Das zu diesem Zweck in einem ersten Normierungsschritt angewandte Vorgehen ist dabei in seinen wesentlichen

[18] Als typische Ansätze für derartige datengetriebene Verfahren könnten in diesem Zusammenhang Neuronale Netze oder auch boostingbasierte Algorithmen angeführt werden.

Grundzügen der bereits von Sung u. Poggio [106] beschriebenen Methodik entlehnt und in Abbildung 2.3 schematisch skizziert. Aufgrund der in einer ersten

(a) (b) (c) (d) (e)

Abbildung 2.3 – Vorverarbeitung eines Bildausschnittes für eine nachfolgende Klassifizierung: Die 20×20 Pixel umfassenden Bildausschnitte (a) werden zunächst maskiert, um störende Hintergrundbereiche auszublenden (b). Durch eine entsprechende adaptive Modellierung der Beleuchtungsverhältnisse (c) lassen sich Einflüsse wie beispielsweise Schlagschatten merklich reduzieren (d). Eine abschließende Histogrammnormalisierung (e) gewährleistet definierte Bedingungen für eine erfolgreiche Erkennung des Bildausschnittes.

Näherung als oval angenommenen Gesichtsstruktur wird, um sich dem störenden Einfluss von eventuell im zu untersuchenden, auf 20×20 Pixel skalierten Bildausschnitt (vgl. Abbildung 2.3a) befindlichen Hintergrundpixeln zu entledigen, eine ellipsenförmige Filtermaske über das Bild gelegt (vgl. Abbildung 2.3b). Der verbleibende Teil des Bildes stellt dann das zu klassifizierende Objekt im Sinne der Personendetektion dar. Bedingt durch die in realen Szenarien typischerweise sehr stark ausgeprägten Beleuchtungsvariationen, die einerseits von einer wechselnden Helligkeit der Lichtquelle sowie andererseits von einer veränderlichen Positionierung derselben relativ zum Objekt herrühren, können jedoch unterschiedliche Objekte der Klasse Gesicht in der dargestellten Form aufgrund einer sehr hohen intra-Klassenvarianz nicht zufriedenstellend erkannt werden. Daher werden die durch die Beleuchtungsvariation hervorgerufenen Effekte, wie beispielsweise Schlagschatten, mittels einer entsprechenden Modellierung der Lichtquelle kompensiert. In Anlehnung an das von Waring u. Liu [113] vorgeschlagene Verfahren wird hierzu das Bild in 5×5 Blöcke unterteilt und in jedem Block der minimale Intensitätswert ermittelt, womit für jeden Block jeweils eine 3×3 Matrix gefüllt wird. Durch eine bilineare Interpolation wird die resultierende 12×12 Matrix dann wieder auf eine Größe von 20×20 skaliert (vgl. Abbildung 2.3c) und vom Bildausschnitt subtrahiert (vgl. Abbildung 2.3d). Abschließend wird das resultierende Bild zur Kontrastverbesserung durch einen Histogrammausgleich normalisiert (vgl. Abbildung 2.3e).

Training des Neuronalen Netzes

Grundlage für die Klassifizierung zwischen den Klassen „Nicht-Gesicht“ und „Gesicht“ bildet ein dreischichtiges Multi-Layer Perzeptron, an dessen Eingangsschicht ein 20×20 großes Grauwertbild G gelegt wird[19] und dessen Ausgangssignal durch einen Wert im Bereich [0;1] die Wahrscheinlichkeit für das Vorliegen eines Gesichtes im angelegten Bildausschnitt widerspiegelt. Von der Eingangs-

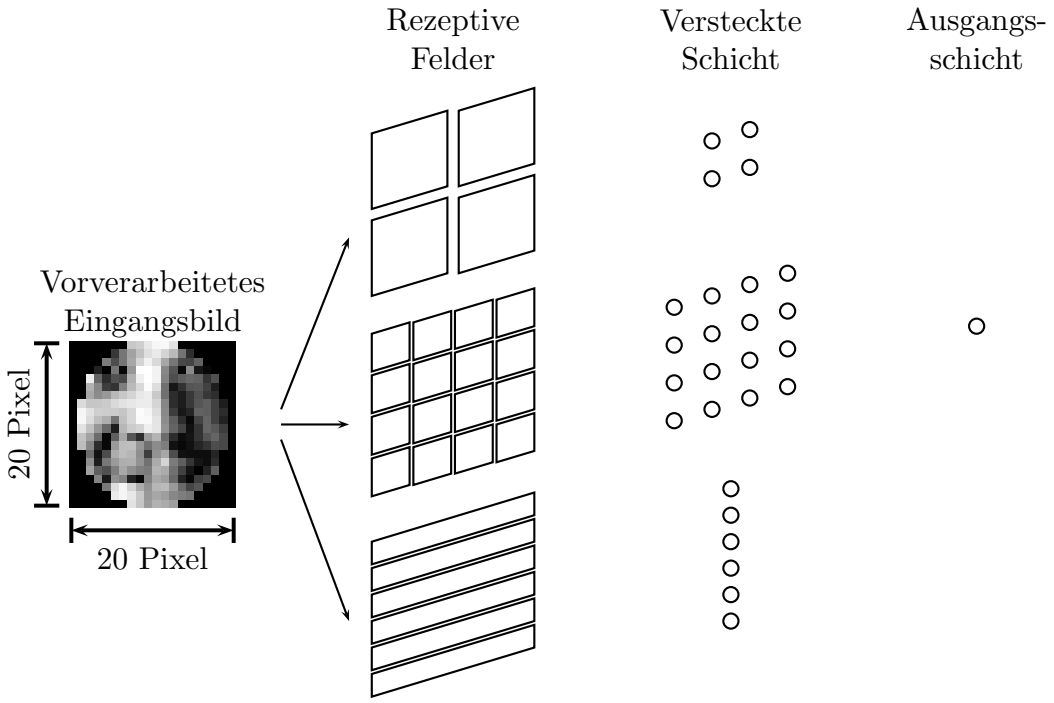


Abbildung 2.4 – Struktur des Neuronalen Netzes gemäß Rowley u. a. [86]: Ein Eingangsbild wird in 26 rezeptive Felder zerlegt und dem entsprechend ein Ensemble von Neuronen der Eingangsschicht jeweils genau einem der 26 Neuronen der versteckten Schicht zugeordnet. Über diese versteckte Schicht wird ein Zusammenhang innerhalb von Gesichtsbildern gelernt. Das reellwertige Ausgangsneuron zeigt schließlich das Ergebnis der Klassifikation.

zur Ausgangsschicht wird das Signal über unterschiedliche Typen rezeptiver Felder propagiert, wodurch die biologischen Vorgänge bei der Informationsaufnahme und -vorverarbeitung auf der Retina entsprechend nachgebildet werden sollen. Hierzu werden, wie in Abbildung 2.4 dargestellt, vier Einheiten aus jeweils 10×10 Neuronen gebildet, die jeweils 25 % des Eingangsbildes analysieren, weitere 16 Einheiten der Größe 5×5 Pixel, um Merkmale wie Augen, Nase oder

[19] Jeder Intensitätswert eines Pixels wird dabei mit genau einem Neuron assoziiert.

Mundwinkel zu detektieren, sowie sechs Einheiten, die jeweils 5 Pixel breite, überlappende Streifen des Bildes nach Mund oder Augenpaaren durchsuchen. Um dieses Netz zu trainieren, bedarf es sowohl positiver Beispiele, d. h. Bilder, die ein Gesicht zeigen, sowie negativer Beispiele. Da aber a-priori unklar ist, welche Art von negativen Beispielen repräsentativ für die Klasse „Nicht-Gesicht“ ist, wird über ein Bootstrapping-Verfahren (vgl. Sung [107]) die Menge der negativen Trainingsbeispiele anhand des Lernerfolges des Netzes iterativ erhöht und angepasst:
Als Ausgangsbasis dient eine Menge an positiven Beispielen, die zur Minimierung der intra-Klassenvarianz, wie in Abbildung 2.5 gezeigt, anhand der Augen- sowie der Mundposition ausgerichtet wurden. Darüber hinaus wird ein weiteres Set

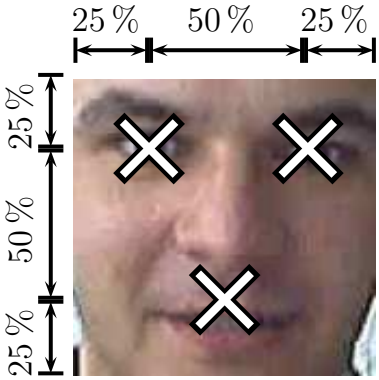


Abbildung 2.5 – Normierungsvorschrift, nach der die zum Training des Neuronalen Netzes verwendeten Positiv-Beispiele aus Bildern erzeugt werden: Einheitlich gilt dabei, dass die Breite des Ausschnittes durch den Augen-Augen Abstand, die Höhe durch den Abstand der Augenlinie zum Mund (Oberlippe) festgelegt wird.

bestehend aus 1000 synthetischen, zufällig erzeugten Bildern generiert, die als Negativ-Beispiele dienen. Nachdem sämtliche Bilder die beschriebene Vorverarbeitungskette durchlaufen haben, werden auf Basis derselben die Gewichte des Neuronalen Netzes mittels des RPROP-Algorithmus trainiert (vgl. Riedmiller u. Braun [83]). Die Lernprozedur terminiert, sobald die Summe der quadratischen Fehler zwischen tatsächlicher Objektklasse und der durch das NN ermittelten, am Ausgangsknoten anliegenden kontinuierlichen Größe auf einem vorher festgelegten Validierungsdatensatz ansteigt, was auf eine Überanpassung[20] des Netzwerkes an die Trainingsdaten hindeutet. Das erzeugte Netz wird dann auf

[20]Engl. „*overfitting*“

reale Bilder, welche kein Gesicht enthalten, angewandt. Aus allen auf diesen Bildern fälschlicherweise als Gesicht klassifizierten Bereichen werden zufällig N_B ausgewählt, die dem Trainingskorpus als weitere Negativ-Beispiele hinzugefügt werden. Anschließend wird das Netz erneut in der geschilderten Weise trainiert. Dies wiederholt sich solange, bis die Zahl der negativen Beispiele diejenige der positiven Bilder übersteigt.

Detektion von Gesichtern

Anhand des trainierten Netzes können nunmehr vorgegebene Bildausschnitte robust klassifiziert werden. Um mit diesem Netz Gesichter in unbekannten Bildern detektieren zu können, wird das zu untersuchende Bild zunächst in einzelne Ausschnitte unterteilt. Hierfür wird ein quadratisches Abtastfenster der Größe 20×20 mit einem Überlappungsgrad von $90\,\% - 95\,\%$ über das Bild verschoben und jeder der so erzeugten Bereiche nach der Vorverarbeitung, wie sie in Abbildung 2.3 beschrieben ist, dem Neuronalen Netz zur Klassifizierung übergeben. Diese Prozedur wiederholt sich anschließend in der geschilderten Weise auf weiteren Bildern, die dadurch entstehen, dass das Ausgangsbild in mehreren Schritten mit konstantem Faktor herunterskaliert wird (siehe Abbildung 2.6).

Abbildung 2.6 – Prinzip der pyramidalen Abtastung: Ausgehend vom Originalbild (links) werden weitere Bilder in der sog. *Gaußpyramide* dadurch erzeugt, indem mittels eines konstanten Faktors das Ausgangsbild unterabgetastet und durch bilineare Interpolation geglättet wird. Innerhalb eines jeden Bildes wird anschließend in der beschriebenen Vorgehensweise auf Basis von 20×20 Pixel umfassenden Ausschnitten nach Gesichtern gesucht.

Erst dadurch wird gewährleistet, dass auch Gesichter, die im Bild einen Bereich von mehr als 20×20 Pixel einnehmen, überhaupt detektiert werden können. Bei

einem solchen blockbasierten Detektionsverfahren wie diesem ist es unmittelbar ersichtlich, dass sich – bedingt durch diese pyramidale Abtastung – schon bei geringen Bildgrößen ein erheblicher Rechenaufwand ergibt, so dass sich beispielhaft für ein Bild in einer Standard-VGA[21] Auflösung bei einem Überlappungsgrad von 90 % und einem Skalierungsfaktor von 83,3 % rund 230000 einzelne Bildausschnitte ergeben. Gerade für derartige Verfahren kann über die oben erläuterten Vorverarbeitungsmethoden (vgl. Abschnitte 2.1.1 und 2.1.2) der relevante Suchraum drastisch[22] eingeschränkt werden, wodurch schließlich auch eine Detektion in Echtzeit möglich wird.

2.2.2 Waveletbasierte Gesichtsdetektion

Ein neuartiger Ansatz zur Gesichtsdetektion in Echtzeit wurde 2001 von Viola u. Jones [110] veröffentlicht. Die Grundidee des von ihnen beschriebenen Algorithmus ist es, aus einer Vielzahl von sehr einfach zu berechnenden Merkmalen, die jeweils für sich betrachtet das Eingangssignal für einen schwachen Klassifikator[23] darstellen, mehrere sog. starke Klassifikatoren zu bilden, die dann – in einer Kaskadenstruktur angeordnet – gelernte Objekte in einem gegebenen Bildausschnitt detektieren.

Merkmalsberechnung

Der Merkmalsberechnung zugrunde liegen dabei die in Abbildung 2.7 zusammengefassten Rechteckstrukturen[24], welche angelehnt sind an die sog. Haar-Wavelets[25] und bereits in ähnlicher Form von Papageorgiou u. a. [71] verwendet wurden. Für jede der rechteckförmigen Masken wird ein korrespondierender

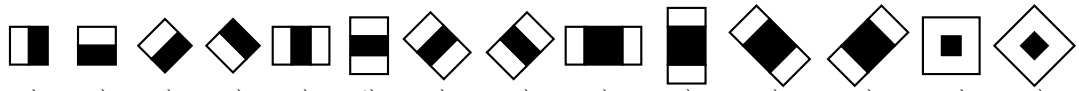


Abbildung 2.7 – Überblick über die zur Detektion von Gesichtern verwendeten Haar-ähnlichen Basismerkmale nach Lienhart u. Maydt [62].

[21] 640×480 Bildpunkte

[22] Typischerweise verbleibt für gewöhnlich ein Suchbereich von ca. 50 %, kann aber erfahrungsgemäß auch Größenordnungen von nurmehr 10 % bis 20 % annehmen.

[23] Schwach soll in diesem Zusammenhang bedeuten, dass der betreffende Klassifikator eine im Mittel nur knapp oberhalb der Ratewahrscheinlichkeit liegende Erkennungsleistung liefert.

[24] Die in der Abbildung gezeigten Merkmale stellen den in einer späteren Arbeit von Lienhart u. Maydt [62] erweiterten Satz an Wavelets dar.

[25] Haar-Wavelets wurden 1909 von Alfred Haar eingeführt und stellen die ersten und mit die einfachsten in der Literatur bekannten Wavelets dar.

Merkmalswert $f_j(x, y, s)$ in Abhängigkeit des Ortes $\vec{p} = (x, y)^T$ und der Skalierung s berechnet, indem die Helligkeitswerte sämtlicher Pixel im grauwertgewandelten Originalbild G, welche durch den schwarzen Bereich der Maske überdeckt werden, aufsummiert und von der Summe der durch den weißen Bereich der Schablone abgedeckten Helligkeitswerte subtrahiert werden. Auf diese Weise können in einem unbekannten Bildbereich objekttypische Intensitätsverläufe, wie sie beispielsweise im Gesicht zwischen Augenpartie und Nasenrücken auftreten (vgl. hierzu Abbildung 2.8), detektiert werden. Je nach Art der Merkmale reagieren

Abbildung 2.8 – Beispiel für die durch die Wavelets beschriebenen Gesichtsmerkmale: horizontaler Übergang von Auge-Nasenrücken-Auge.

diese sensitiv insbesondere auf Kanten (Abbildung 2.7a-d), Linien (Abbildung 2.7e-l) oder Punktflächen (Abbildung 2.7m,n). Der Vorteil bei Verwendung derartiger Merkmale besteht vor allem darin, dass sie sich sehr effizient über sog. Integralbilder berechnen lassen. Je nach Typus des zugrunde liegenden Merkmals wird hierzu für die horizontal bzw. vertikal ausgerichteten Masken ein Integralbild $\mathcal{G}_{\mathrm{SAT}}$ sowie für die diagonal angeordneten Merkmale ein Integralbild $\mathcal{G}_{\mathrm{RSAT}}$ generiert[26]:

$$\mathcal{G}_{\mathrm{SAT}}(x, y) = \sum_{x' \leq x, y' \leq y} G(x', y') \tag{2.24}$$

$$\mathcal{G}_{\mathrm{RSAT}}(x, y) = \sum_{x' \leq x, x' \leq x - |y - y'|} G(x', y'). \tag{2.25}$$

Mit Hilfe dieser Matrizen, welche die Stammfunktion zum Originalbild darstellen, kann so für ein durch die Punkte $(x_A, y_A), (x_B, y_B), (x_C, y_C)$ und (x_D, y_D) definiertes Rechteck, wie es Grundlage der Haar-ähnlichen Merkmale ist, durch nur vier Tabellenzugriffe die kumulierte Helligkeit

$$G^*_{\mathrm{Kum}} = \mathcal{G}_{\mathrm{SAT}}(x_B, y_B) - \mathcal{G}_{\mathrm{SAT}}(x_A, y_A) - \mathcal{G}_{\mathrm{SAT}}(x_C, y_C) + \mathcal{G}_{\mathrm{SAT}}(x_D, y_D) \tag{2.26}$$

[26] Engl. „*summed area table*“ (SAT) bzw. engl. „*rotated summed area table*“ (RSAT)

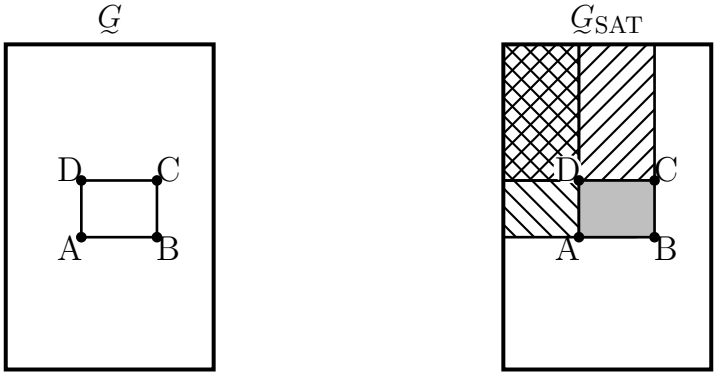


Abbildung 2.9 – Effiziente Merkmalsberechnung mit Hilfe des Integralbildes: Die Summe aller Helligkeitswerte, die durch eine rechteckförmige Struktur in einem Bild (links) wie beispielsweise dem weißen oder schwarzen Teilbereich der Haar-ähnlichen Merkmale abgedeckt werden, läßt sich – unabhängig von der Fläche des Rechtecks – über das zugehörige Integralbild (rechts) mit nur vier Tabellenzugriffen bestimmen.

in dem durch das Rechteck definierten Bildausschnitt $\underset{\sim}{G}^*$ bestimmt werden (vgl. Abbildung 2.9). Durch die konkrete Gestalt der verwendeten Schablonen aus Abbildung 2.7, die sich aus jeweils zwei bzw. drei einzelnen Rechtecken zusammensetzen, ergibt sich hierdurch eine sehr performante Berechnung sämtlicher Merkmale in konstanter Zeit.

Beleuchtungsausgleich

Um den Einfluss von Beleuchtungsschwankungen auf die Merkmale selbst möglichst gering halten zu können, erfolgt eine lokale Normalisierung des betrachteten Bildausschnittes $\underset{\sim}{G}^*$ bezüglich der Varianz σ^2, die mathematisch über den Zusammenhang

$$\sigma^2 = \frac{1}{N_{\mathrm{Pix}}} \sum_{x,y} (\underset{\sim}{G}^*(x,y))^2 - \left(\frac{1}{N_{\mathrm{Pix}}} \sum_{x,y} \underset{\sim}{G}^*(x,y) \right)^2 \tag{2.27}$$

aus dem Mittelwert der quadratischen Helligkeitswerte und dem quadrierten Mittelwert über alle Helligkeitswerte errechnet werden kann. Während letzterer sofort durch das bereits vorhandene Integralbild G_{SAT} für einen gegebenen Bildausschnitt wiederum in konstanter Zeit ermittelt werden kann, bedarf es zur effizienten Berechnung des Terms $\frac{1}{N_{\mathrm{Pix}}} \sum_{x,y} (\underset{\sim}{G}^*(x,y))^2$ eines weiteren Integralbildes, welches über die quadratischen Helligkeitswerte kumuliert. Liegt auch ein solches vor, so läßt sich die Varianz σ^2 ebenso für jeden Bildausschnitt in konstanter Zeit ermitteln.

Training des waveletbasierten Detektors

Analog zum Training des NN werden definierte Bildausschnitte, die sowohl positive als auch negative Beispiele zeigen, nun aber in der Größe 24×24 benutzt, um objektklassenspezifische Gemeinsamkeiten zu lernen, allerdings nicht mehr basierend auf den Grauwertinformationen der einzelnen Bildpunkte, sondern auf den berechneten Merkmalen. Da jedoch innerhalb des Bildausschnittes die Merkmalsfilter jeweils in diversen Skalierungen s an unterschiedlichen Positionen (x, y) platziert werden können, ergibt sich bereits für ein 24×24 Basisfenster ein ca. 118000 dimensionaler Merkmalsvektor, der die Anzahl vorhandener Pixel in dem zugrunde liegenden Fenster und somit die native Dimension des Ausschnittes bei weitem übersteigt. Aus diesem Grund gilt es eine geeignete Auswahl unter den zu nutzenden Merkmalen zu treffen. Durch den Einsatz von Boosting (vgl. Freund [33], Freund u. Schapire [34]) kann die Dimensionalität dieses Vektors durch eine repräsentative Auswahl bestimmter Vorkommnisse von Merkmalsfiltern (insbesondere bzgl. deren Lagebeziehung und Skalierung) entscheidend reduziert werden. Originäres Ziel des Boostings ist es, durch Kombination verschiedener schwacher Klassifikatoren mit jeweils einer korrekten Klassifikationsrate von knapp oberhalb der Ratewahrscheinlichkeit einen starken Klassifikator zu erzeugen, der dann auf Basis eines gewichteten Mehrheitsentscheides arbeitet. Dieses Prinzip läßt sich in adaptierter Form zur Auswahl geeigneter Merkmale für die Gesichtsdetektion wie folgt anwenden:

Über eine binäre Entscheidungsfunktion k_j wird anhand der Schwelle Θ_j jedes Merkmal $f_j(x, y, s)$ auf eine der beiden Klassen „Gesicht“ bzw. „Nicht-Gesicht“ abgebildet:

$$k_j = \begin{cases} 1 & \text{wenn } \; p_j f_j(x, y, s) < p_j \Theta_j \\ -1 & \text{sonst} \end{cases} \tag{2.28}$$

Über die Parität p_j kann hierbei die Ungleichheitsbeziehung des Schwellwertentscheides gesteuert werden, je nachdem, ob für die Mehrzahl der Positivbeispiele ein Schwellwert über- oder unterschritten wird. Bedingt durch die Parameter p_j und Θ_j resultieren bei Vorliegen von N_{Bsp} Trainingsbeispielen für jedes Merkmal $2 \cdot N_{\text{Bsp}}$ verschiedene schwache Klassifikatoren[27]. Über das in Algorithmus 1 skizzierte AdaBoost-Verfahren wird anschließend ein starker Klassifikator K konstruiert.

Hierzu werden zunächst initiale Gewichte $w_{1,i}, i \in \{1, \ldots, N_{\text{Bsp}}\}$ für alle Trainingsbeispiele festgelegt. In jeder von insgesamt N_{Iter} Iterationen wird anschließend derjenige Klassifikator k_t ermittelt, der bei gegebener Gewichtung der Beispiel-

[27] Für N_{Bsp} Trainingsbeispiele lassen sich jeweils N_{Bsp} unterschiedliche Intervalle für einen Schwellwert Θ_j definieren, der wahlweise unter- oder überschritten werden kann.

Algorithmus 1 AdaBoost

Benötigt:

Menge an Trainingsbildern $\{\underline{G}^*_1, \ldots, \underline{G}^*_{N_{\text{Bsp}}}\}$ mit jeweils zugehörigem Klassenlabel $\{y_1, \ldots, y_{N_{\text{Bsp}}}\}$ mit $y_i \in \{-1, 1\}$

procedure

Initialisiere Gewichte $w_{1,i} = \frac{1}{N_{\text{Bsp}}} \quad \forall \quad i \in \{1, \ldots, N_{\text{Bsp}}\}$

for $(t = 1, \ldots, N_{\text{Iter}})$ **do**

Normalisiere die Gewichte $\tilde{w}_{t,i} = \frac{w_{t,i}}{\sum_{j=1}^{N_{\text{Bsp}}} w_{t,j}}$, um damit eine Wahrscheinlichkeitsverteilung zu erhalten

Pro Merkmal $f_j(x, y, s)$ wird ein schwacher Klassifikator k_j trainiert und der resultierende Fehler $\epsilon_{t,j} = \sum_{i=1}^{N_{\text{Bsp}}} w_{t,i} |k_j(\underline{G}^*_i) - y_i|$ berechnet

Wähle denjenigen Klassifikator k_t, für den der Fehler $\epsilon_{t,j}$ minimal wird

Aktualisiere die Gewichte:

$w_{t+1,i} = \tilde{w}_{t,i} e^{-\chi_t k_t(\underline{G}^*_i) y_i}$ mit $\chi_t = \frac{1}{2} \ln\left(\frac{1 - \min_j \epsilon_{t,j}}{\min_j \epsilon_{t,j}}\right)$

end for

end procedure

bilder den kleinsten Klassifizierungsfehler $\epsilon_{t,j}$ ausweist. Am Ende jeder Iteration werden abschließend die Gewichte auf Basis der Erkennungsergebnisse des ausgewählten Klassifikators k_t angepasst, so dass falsch klassifizierte Beispiele mit einer entsprechend höheren Gewichtung in die nächste Iteration gehen. Die aus den Iterationen resultierenden N_{Iter} Klassifikatoren bilden als Linearkombination den starken Klassifikator

$$K = \begin{cases} 1 & \text{wenn} \quad \sum_{t=1}^{N_{\text{Iter}}} \chi_t k_t \geq 0,5 \sum_{t=1}^{N_{\text{Iter}}} \chi_t \\ -1 & \text{sonst} \end{cases} . \tag{2.29}$$

Motiviert durch die in der Praxis zu beobachtende Tatsache, dass oftmals der Großteil eines Bildes und somit die Mehrheit der Abtastfenster ausschließlich Hintergrund zeigt, werden zur Klassifizierung der Bildausschnitte mehrere starke Klassifikatoren unterschiedlicher Komplexität zu einer Kaskade seriell verschaltet (siehe Abbildung 2.10). Diesem Vorgehen liegt dabei die Idee zugrunde, dass eine Vielzahl der Abtastfenster, welche kein zu detektierendes Objekt enthalten, bereits durch einen relativ einfachen, starken Klassifikator K, bestehend aus einer Ansammlung nur weniger Merkmalsfilter, verworfen werden kann und damit eine sehr zeiteffiziente Vorselektion eventueller Objekt-Kandidaten ermöglicht wird. Dieser Prozess wiederholt sich in sämtlichen N_{Kask} Kaskadenstufen,

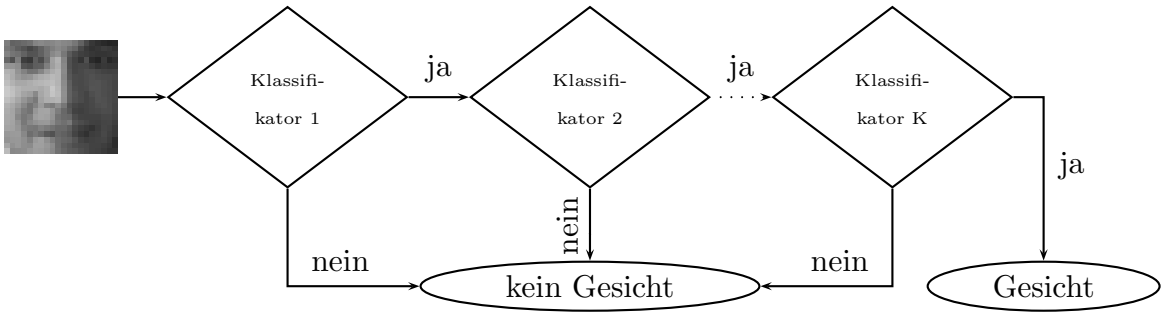


Abbildung 2.10 – Kaskadendetektor bestehend aus N_{Kask} starken Klassifikatoren: In der frühen Phase der Kaskade bestehen die Klassifikatoren aus nur wenigen Merkmalen (bei Viola u. Jones [110] z. B. jeweils zwei Merkmalen), so dass eine sehr schnelle Klassifikation von Bildausschnitten erfolgt. Ermöglicht durch die damit praktizierte Vorselektion von den zu validierenden Bildausschnitten, die potentiell ein Gesicht zeigen, werden diese Ausschnitte im Verlauf der Kaskade mit immer komplexeren – beispielsweise wie bei Viola u. Jones [110] aus 200 Merkmalen zusammengesetzte – Klassifikatoren analysiert.

so dass nach erfolgreichem Durchlaufen der letzten Stufe der vorliegende Bildausschnitt endgültig als Gesicht klassifiziert wird.
Zum Aufbau der Kaskade mittels des in Algorithmus 2 skizzierten Schemas werden die einzelnen Klassifikatorstufen jeweils nur noch auf denjenigen Daten, die von der unmittelbar vorhergehenden Stufe positiv bewertet wurden, trainiert. Ein derartiges Vorgehen resultiert in einer am Ausgang der Kaskade gültigen Detektionsrate von

$$p_{\text{TP,ges}} = \prod_{i=1}^{N_{\text{Kask}}} p_{\text{TP},i}, \tag{2.30}$$

wobei $p_{\text{TP},i}$ die Detektionsrate der i-ten Stufe der Kaskade repräsentiert. Für eine hohe Detektionsgüte der gesamten Kaskadenstruktur ist es daher notwendig, in jeder Stufe eine möglichst hohe Detektionsrate zu gewährleisten, was durch die zusätzliche Addition einer Konstanten in Gleichung 2.29 einzustellen ist. Damit einhergehend wird zwar zeitgleich ein Anstieg der Falsch-Positiv-Rate $p_{\text{FP,ges}}$ verursacht, wobei dies aufgrund der Propagierung über die einzelnen Stufen der Kaskade durchaus in Kauf genommen werden kann und letztlich z. B. für eine Falsch-Akzeptanzrate von 30 % in jeder Stufe zu einer gesamten Falsch-Klassifikation bei einer zehnstufigen Kaskade in der Größenordnung 10^{-6} führt.

Algorithmus 2 Training der Klassifikationskaskade

Benötigt:

Maximum für die gerade noch akzeptierte falsch-positiv-Rate p_{FP} pro Kaskadenstufe
Minimum für die gerade noch akzeptierte Detektionsrate p_{TP} pro Kaskadenstufe
Gewünschte falsch-positiv-Rate $p_{\mathrm{FP,ges}}$ über die gesamte Kaskade
Menge der positiven Trainingsbilder $\mathcal{P}$
Menge der negativen Trainingsbilder $\mathcal{N}$
Unabhängige Menge an Bildern $\mathcal{V}$ zur Validierung

procedure
  Initialisiere $p_{\mathrm{FP},0} = 1.0$, $p_{\mathrm{TP},0} = 1.0$ und $i = 0$
  while $(p_{\mathrm{FP,ges},i} > p_{\mathrm{FP,ges}})$ **do**
    Inkrementiere i
    Setze $N = 0$ und $p_{\mathrm{FP,ges},i} = p_{\mathrm{FP,ges},i-1}$
    while $(p_{\mathrm{FP,ges},i} > p_{\mathrm{FP}} \cdot p_{\mathrm{FP,ges},i-1})$ **do**
      Inkrementiere N
      Trainiere einen Klassifikator K mittels AdaBoost, bestehend aus N Merkmalen anhand der Trainingsbilder $\mathcal{P}$ und $\mathcal{N}$
      Bestimme $p_{\mathrm{FP,ges},i}$ und $p_{\mathrm{TP,ges},i}$ für den trainierten Klassifikator K anhand der Validierungsbilder $\mathcal{V}$
      Reduziere den Schwellwert des aktuellen Klassifikators solange, bis dieser eine Detektionsrate von $p_{\mathrm{TP}} \cdot p_{\mathrm{TP,ges},i-1}$ erreicht.
    end while
    Setze $\tilde{\mathcal{N}} = \{\}$
    Falls $p_{\mathrm{FP,ges},i} > p_{\mathrm{FP,ges}}$, so evaluiere den aktuellen Klassifikator K erneut auf der Menge $\mathcal{N}$ und befülle die Menge $\tilde{\mathcal{N}}$ mit all denjenigen Bildern, die hierbei als positiv klassifiziert wurden
    Setze $\mathcal{N} = \tilde{\mathcal{N}}$
  end while
end procedure

Detektion von Gesichtern

Wie allgemein bei einer Vielzahl von Detektionstechniken üblich, wird auch hier zunächst das zu verarbeitende Bild durch überlappende Fensterung abgetastet. Um auch Objekte unterschiedlicher Größe detektieren zu können, wird jedoch – anders als bei der in zahlreichen Verfahren zur Gesichtsdetektion oftmals ange-

wandten Pyramiden-Technik[28] – bei diesem Algorithmus nicht das Bild selbst, sondern vielmehr die Merkmale skaliert, wodurch eine neuerliche Berechnung der Integralbilder vermieden wird. Daher wird bei diesem Ansatz das Abtastfenster in unterschiedlicher Skalierung über das Ausgangsbild geschoben, wodurch eine sehr zeiteffiziente Detektion von – auch in der Größe variierenden – Gesichtern in unbekannten Bildern ermöglicht wird.

2.3 Trackinglogik

Basierend auf einer gewählten Modellierung wäre es prinzipiell denkbar, in jedem Einzelbild einer vorhandenen Videosequenz völlig unabhängig von jeglichem zeitlichen Vorwissen zu verfolgende Objekte neuerlich durch eine vollständige Suche zu detektieren. Neben der Tatsache, dass sich eine Zuordnung von Objekten aus aufeinanderfolgenden Bildern dann als sehr aufwendig erweist, erscheint für praktische Anwendungen ein derartiges Vorgehen oftmals aus zweierlei Gründen als ungeeignet:
Zum einen beansprucht eine vollständige Suche nach Objekten in jedem Einzelbild für die Mehrheit der Detektionsverfahren auch auf modernen Rechnerarchitekturen ein hohes Maß an Systemressourcen, so dass ein realzeitfähiger Einsatz als nicht garantiert erscheint. Darüber hinaus kann durch keines der derzeit bekannten Verfahren eine fehlerlose Detektion[29] unabhängig von den gerade vorherrschenden Rahmenbedingungen, wie sie bereits angesprochen wurden, gewährleistet werden, so dass die Qualität der Personenverfolgung stark darunter leiden würde.
Aus diesem Grund wird eine übergeordnete Steuerung eingeführt, die neben der Zuordnung von Objekten in aufeinander folgenden Bildern einer Videosequenz auch die Objektverwaltung (insbesondere inkl. deren Eigenschaften) und die Trajektorienberechnung übernimmt. Wie eingangs durch das Blockdiagramm schon angedeutet, zerfällt diese Steuerlogik im Wesentlichen in drei Teile: Bestimmung der Personenkorrespondenzen, Trajektorienberechnung, sowie Prädiktion der Objekteigenschaften.

[28] Vgl. hierzu beispielsweise das Vorgehen bei dem im vorangegangenen Abschnitt vorgestellten Ansatz nach Rowley u. a. [86].

[29] Fehlerlos bedeutet in diesem Kontext insbesondere, dass eine 100 %ige Detektionsrate einhergeht mit einer 0 %igen Falsch-Akzeptanz Rate.

2.3.1 Bestimmung der Personenkorrespondenzen

Aufgabe dieses Moduls ist es, über den zeitlichen Verlauf aktuell im Bild detektierte Objekte eindeutig den aus dem vorangegangenen Zeitschritt erhaltenen Objekten zuzuweisen bzw. neu in der Szene erscheinende Objekte als solche zu identifizieren. Erst über diese Zuordnung ist es möglich, in einer nachgelagerten Prozedur die Trajektorie eines Objektes zu bestimmen.
Zur Bestimmung der Objektkorrespondenz wird hierbei meist ein Abstandsmaß definiert, welches für gewöhnlich auf der räumlichen Lagebeziehung (Position sowie Größe und evtl. Rotation) und darüber hinaus auf den Texturmerkmalen des Objektes basiert. Gerade die Hinzunahme der Texturähnlichkeit zweier Objekte wirkt sich hierbei positiv auf die Störempfindlichkeit der Objektverfolgung aus, da die alleinige Nutzung der Lagebeziehung von Objekten vor allem in Situationen kurz vor oder nach einer gegenseitigen Verdeckung eine nur unzureichende Informationsquelle für die eindeutige Zuordnung von Objekten darstellt. Die Messung der Texturähnlichkeit zweier Objekte stützt sich dabei häufig auf eine Histogrammdarstellung der zu vergleichenden Texturen und kann nach Cha u. Srihari [22] im Wesentlichen in die Kategorien vektor- oder wahrscheinlichkeitsbasierte Ansätze unterteilt werden. Bei ersteren werden die Grauwertstatistiken zweier Bilder hierbei direkt auf Vektoren $\vec{H}_1$ und $\vec{H}_2$ der Länge b abgebildet, die dann über bekannte Abstandsmaße wie die Manhattan ($D_M(\vec{H}_1, \vec{H}_2)$) oder die Euklid'sche Distanznorm ($D_E(\vec{H}_1, \vec{H}_2)$) bzw. mitunter auch über eine Schnittmengenbetrachtung ($D_I(\vec{H}_1, \vec{H}_2)$) elementweise miteinander verglichen werden können:

$$D_M(\vec{H}_1, \vec{H}_2) = \sum_{i=1}^{b} |\vec{H}_1(i) - \vec{H}_2(i)| \quad \text{bzw.} \tag{2.31}$$

$$D_E(\vec{H}_1, \vec{H}_2) = \sqrt{\sum_{i=1}^{b} (\vec{H}_1(i) - \vec{H}_2(i))^2} \quad \text{bzw.} \tag{2.32}$$

$$D_I(\vec{H}_1, \vec{H}_2) = \sum_{i=1}^{b} \min(\vec{H}_1(i), \vec{H}_2(i)) \tag{2.33}$$

In der Praxis erweisen sich derlei Maße jedoch oftmals als sehr sensitiv gegenüber Bildrauschen oder anderen Störungen (vgl. Huet u. Hancock [46]). Darüber hinaus ist man im Zuge des Trackingproblems vielmehr an einer probabilistischen Form der Ähnlichkeit interessiert, weil hieraus unmittelbar ein normiertes Wertemaß resultiert. Deswegen stellt die Gruppe der wahrscheinlichkeitsbasierten Ähnlichkeitsbewertungen die bevorzugte Variante des Histogrammvergleiches

dar (vgl. Comaniciu u. Meer [26]). Die entsprechenden Maße werden mittels der normierten Histogramme $\vec{H}_1^*$ und $\vec{H}_2^*$ auf Basis der Kullback-Leibler-Distanz $D_K(\vec{H}_1^*, \vec{H}_2^*)$ oder der Bhattacharyya-Distanz $D_B(\vec{H}_1^*, \vec{H}_2^*)$ ermittelt (vgl. Kang u. a. [55]):

$$D_K(\vec{H}_1^*, \vec{H}_2^*) = \sum_{i=1}^{b} (\vec{H}_1^*(i) - \vec{H}_2^*(i)) \log \frac{\vec{H}_1^*(i)}{\vec{H}_2^*(i)} \quad \text{bzw.} \tag{2.34}$$

$$D_B(\vec{H}_1^*, \vec{H}_2^*) = -\log \sum_{i=1}^{b} \sqrt{\vec{H}_1^*(i)\vec{H}_2^*(i)} \tag{2.35}$$

Durch eine Schwellwertentscheidung kombiniert mit einem Maximumsentscheid kann eine Zuordnung von Objekten unterschiedlicher Zeitschritte unmittelbar anhand dieser Messgrößen getroffen werden.

2.3.2 Trajektorienberechnung

Sowohl bei den bottom-up als auch den hypothesengetriebenen Ansätzen liegen Aussagen über mögliche Objektpositionen häufig auf Basis von Wahrscheinlichkeiten vor. Um eine robuste Personenverfolgung zu realisieren, wird häufig in einem nachgelagerten Prozess die Position von Objekten durch eine Mittelung über die vorliegenden Objekthypothesen bestimmt. Hierbei bilden sowohl die unbekannte Anzahl an Objekten als auch die Bestimmung all derjenigen Hypothesen, die ein- und dasselbe Objekt repräsentieren, die zentralen Probleme. Zahlreiche Ansätze (vgl. u. a. Gatica-Perez u. a. [35], Isard u. Maccormick [48]) verwenden hierzu eine speziell für das simultane Verfolgen mehrerer Objekte erweiterte Fassung eines Partikelfilters, bei dem die Hypothesen auch den Kontext eines Szenarios mit erfassen, wodurch gleichzeitig die Zahl der zu detektierenden Objekte und deren Lagebeziehung automatisch bestimmt wird. Erst durch diesen Schritt wird es möglich, für jedes der durch die Hypothesen erfasste Objekt auch seine dazugehörige Trajektorie zu ermitteln.
Obwohl prinzipiell durch die Zuordnung der Objekte aus unterschiedlichen Zeitschritten bereits die Trajektorie bestimmbar ist, kann diese eventuell für die nachfolgende Anwendung aufgrund von Rauschen oder Messfehlern so nicht unmittelbar genutzt werden. Aus diesem Grund wird die Trajektorie im einfachsten Fall durch einen zeitlichen Mittelwert über mehrere Zeitschritte oder aber auch durch den Einsatz eines Kalman-Filters geglättet.

2.3.3 Prädiktion der Objekteigenschaften

Sowohl bei den hypothesengetriebenen Ansätzen, als auch den bottom-up Verfahren, ist eine Prädiktion der aktuellen Objekteigenschaften $\vec{h}_t$ essentiell im Hinblick auf eine robuste Personenverfolgung. Hierzu wird häufig über ein mit normalverteiltem, mittelwertfreiem Rauschen $\vec{u}_t$ (Kovarianz $\underset{\sim}{\Sigma}$) beaufschlagtes, lineares Bewegungsmodell

$$\vec{h}_{t+1} = \underset{\sim}{A}\vec{h}_t + \vec{u}_t, \tag{2.36}$$

dessen Bewegungsmatrix $\underset{\sim}{A}$ vorab anhand von annotierten Daten erstellt oder aufgrund empirischer Beobachtungen geschätzt wurde, die zu erwartenden Eigenschaften $\vec{h}_{t+1}$ und damit u. a. auch die Position im nächsten Zeitschritt bestimmt.

Kapitel 3

Videobasierte hybride Personenverfolgung in Besprechungsszenarien

Obwohl gerade auf dem Gebiet der Personenverfolgung in den letzten 15 Jahren bereits sehr viel Forschung betrieben wurde, so gestaltet sich dennoch die szenarienunabhängige, automatische Verfolgung von Personen in monokularen Bildsequenzen als äußerst schwierig. Dies begründet sich vor allem in den zahlreichen Einflüssen, die unmittelbar auf die Qualität der Ergebnisse eines Trackingsystems einwirken (vgl. Javed u. Shah [50]). Die Personendetektion als das Kernstück eines jeden Systems zur Personenverfolgung reagiert hierbei oftmals sehr anfällig auf (Teil-)Verdeckungen, die durch andere Personen, Gegenstände im Raum oder speziell in Innenraum-Szenarien durch eine kameranahe Position verursacht werden können. Hier ist es das Bestreben der Forschung, einerseits durch die Art der Modellierung einer Person dieser Sensitivität vorzubeugen, sowie andererseits durch eine entsprechende Systemarchitektur geeignete Vorkehrungen zu treffen, um auch in einer solchen, für gewöhnlich nur vorübergehend vorherrschenden Situation die robuste Verfolgung einer Person zu ermöglichen. Während die Verdeckungsproblematik überwiegend die Detektionsstufe betrifft, wirken andere Effekte auf sämtliche Module der Prozesskette ein. Insbesondere sich schnell verändernde Beleuchtungsbedingungen sowie Schatteneffekte beeinflussen hierbei mitunter je nach gewählter Modellierung unterschiedlich stark die Leistung der Personendetektion, sind aber auch durch die gängigen Algorithmen der Vorverarbeitung nur selten ausreichend zu beseitigen. Weiter stellen sich in diesem Zusammenhang auch eine schlechte Aufnahmequalität der Videoquelle oder ein stark strukturierter Bildhintergrund als ebenso hinderlich im Sinne einer erfolgreichen Personenverfolgung heraus.
In Anbetracht dieser Tatsache ist es wenig überraschend, dass kommerziell bisher

nur Systeme eingesetzt werden können, die sehr anwendungsspezifisch entwickelt wurden und dabei konkret definierte Umgebungsbedingungen voraussetzen. Den Systemen wird dabei als Expertenwissen extrinsische Information zur Verfügung gestellt, welches dann in sämtlichen Modulen unterstützend zur Erfüllung ihrer jeweiligen Aufgabe verwendet werden kann. Während jedoch der Mensch intuitiv sein über Jahre erworbenes Kontextwissen bezogen auf die jeweils zu lösende Aufgabe, wie beispielsweise die Detektion von Personen, einsetzt und damit häufig sehr erfolgreich auch in ihm unbekannten Szenarien agiert, muss der Rechner gemeinhin mit einem wesentlich geringeren Vorwissen auskommen. Aus diesem Grund ist es daher wichtig, dass sämtliche im Laufe des Verarbeitungsprozesses gewonnenen Daten Berücksichtigung bei der zum aktuellen Zeitschritt erfolgenden Analyse einer Szene finden. Zu diesem Zweck wird mit der in dieser Arbeit

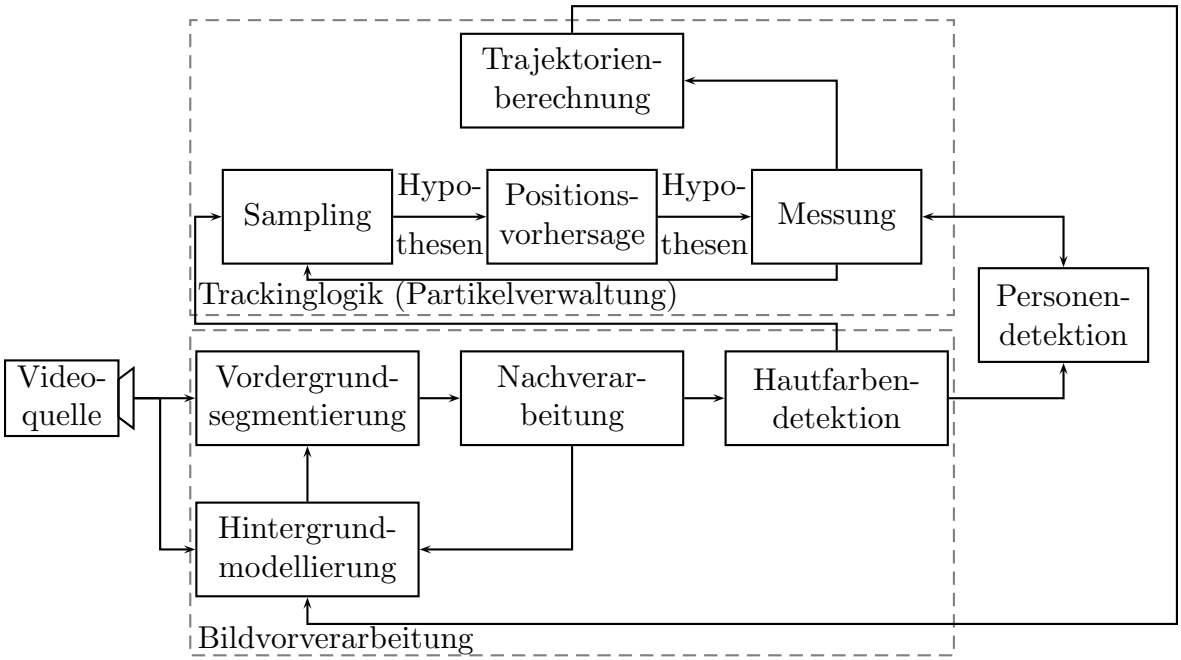


Abbildung 3.1 – Schematische Visualisierung des in der vorliegenden Arbeit entworfenen hybriden Systems zur Personenverfolgung: Die üblicherweise unidirektional (vgl. hierzu die allgemeine Darstellung in Abbildung 2.1) ausgestaltete Messung der durch den Partikelfilter generierten Hypothesen wurde um einen Rückkanal erweitert. Durch die zusätzliche Rückführung der im Zuge des Trackings gewonnenen Informationen kann auch bereits der Vorverarbeitung wesentliches Kontextwissen bereitgestellt werden.

vorgestellten Systemarchitektur versucht, gemäß dem psychologischen Verständnis der visuellen Informationsverarbeitung beim Menschen (vgl. hierzu Hochstein

u. Ahissar [43]), wonach die Szenenanalyse sowohl in einem bottom-up als auch gleichzeitig in einem top-down Vorgehen vonstatten geht, ein biologisch motiviertes Vorgehensprinzip auf algorithmischer Ebene nachzuempfinden. Eines der Ziele der vorliegenden Arbeit ist es daher, die vorrangig datengetriebene Sichtweise des Trackingproblems, wie sie vor allem bis gegen Ende der 90er Jahre vorherrschte (vgl. Bobick u. a. [19], Haritaoglu u. a. [40], Yamane u. a. [118]), mit der modernen, hypothesengesteuerten Technik, die oftmals auf Kalmanfilterung (vgl. Zhao u. a. [122], Zhao u. Nevatia [123]) oder stochastischen Abtaststrategien (z. B. Partikelfilter, vgl. Gatica-Perez u. a. [35], Isard u. Maccormick [48]) beruht, zu vereinen.
Der Entwurf eines solchen hybriden Systems betrifft hierbei maßgeblich die Schnittstelle zwischen Personendetektionsstufe und Trackinglogik, die dann nicht mehr unidirektional ausgestaltet sein kann, sondern notwendigerweise durch die vorliegenden Bilddaten verursachte Zustandsänderungen in den Hypothesen ermöglichen muss. Wie in Abbildung 3.1 visualisiert, dient dann jede Hypothese zwar einerseits als Ausgangsbasis für die lokale Bildanalyse, kann jedoch gleichzeitig auf Basis der Bildinformation in den beschreibenden Eigenschaften verändert bzw. optimiert werden. Grundvoraussetzung hierfür ist allerdings, dass die Personendetektion auf einem Modell beruhen muss, das in einer geeigneten Form eine Adaption auf vorliegende Daten erlaubt. Um die Idee der ganzheitlichen Wissensverarbeitung auch auf alle Ebenen des entwickelten Trackingsystems auszuweiten, wird durch Rückführung des – während des aktuellen Zeitschrittes – eruierten Wissens über die Objekte im Bild an die Hintergrundmodellierung sichergestellt, dass bereits in der Vorverarbeitungsphase jegliche Information, die dem System vorliegt, schon sehr früh in den Analyseprozess einbezogen und damit umfassend genutzt werden kann.
In den folgenden Abschnitten wird das Prinzip der hypothesengesteuerten Objektverfolgung erläutert und im Sinne der hybriden Trackingarchitektur geeignete Ansätze zur Personenmodellierung vorgestellt. Für dieses, in seiner eingeführten Form zunächst nur zur Einzelpersonenverfolgung anwendbare System werden in den anschließenden Abschnitten sinnvolle Erweiterungen aufgezeigt, um auf Grundlage dieses Systems eine hybride Mehrpersonenverfolgung zu realisieren.

3.1 Hypothesenbasiertes Tracking

Stochastisch formuliert bezeichnet Tracking im Sinne der Objektverfolgung das Problem, den Zustand $\vec{x}_t$, bzw. die a-posteriori-Wahrscheinlichkeitsdichte $p(\vec{x}_t) \equiv p(\vec{x}_t|\mathcal{I}_t)$ mit $\mathcal{I}_t = \{\underset{\sim}{I}_1, \ldots, \underset{\sim}{I}_t\}$ eines dynamischen Systems zum Zeitpunkt t anhand

der gesamten bis zum aktuellen Zeitpunkt in Form von Bilddaten verfügbaren Information $\mathcal{I}_t$ zu schätzen. In den meisten Fällen existiert zur Berechnung der Wahrscheinlichkeitsverteilung $p(\vec{x}_t|\mathcal{I}_t)$ jedoch keine geschlossene Form, so dass das Problem unter Verwendung der Bayes'schen Regel umformuliert wird zu

$$\begin{aligned} p(\vec{x}_t|\mathcal{I}_t) &= \frac{p(\mathcal{I}_t|\vec{x}_t)p(\vec{x}_t)}{p(\mathcal{I}_t)} = \frac{p(\underset{\sim}{I}_t|\vec{x}_t,\mathcal{I}_{t-1})p(\vec{x}_t,\mathcal{I}_{t-1})}{p(\mathcal{I}_t)} = \\ &= \frac{p(\underset{\sim}{I}_t|\vec{x}_t,\mathcal{I}_{t-1})p(\vec{x}_t|\mathcal{I}_{t-1})}{p(\underset{\sim}{I}_t|\mathcal{I}_{t-1})}. \end{aligned} \tag{3.1}$$

Für die Annahme, dass das aktuelle Bild $\underset{\sim}{I}_t$ statistisch unabhängig von der vorangegangenen Bildsequenz $\mathcal{I}_{t-1}$ ist, also $p(\underset{\sim}{I}_t|\mathcal{I}_{t-1}) = p(\underset{\sim}{I}_t)$ gilt, vereinfacht sich die Gleichung 3.1 weiter zu

$$p(\vec{x}_t|\mathcal{I}_t) = \frac{p(\underset{\sim}{I}_t|\vec{x}_t)p(\vec{x}_t|\mathcal{I}_{t-1})}{p(\underset{\sim}{I}_t)}. \tag{3.2}$$

Da der Nenner in dieser Gleichung unabhängig von den Zuständen ist, wird er durch eine zeitvariable Proportionalitätskonstante $\kappa_t = p(\underset{\sim}{I}_t)^{-1}$ zur Einhaltung der Stochastizitätsbedingung ersetzt. Der Term $p(\vec{x}_t|\mathcal{I}_{t-1})$ im Zähler der Gleichung 3.2 kann in der Regel nicht in geschlossener Form gelöst werden. Wird jedoch angenommen, dass die Zustandsübergänge durch eine Markov-Kette 1. Ordnung modelliert werden können, also der aktuelle Zustand $\vec{x}_t$ nur vom unmittelbar vorhergehenden Zustand $\vec{x}_{t-1}$ abhängt und dadurch $p(\vec{x}_t|\vec{x}_1,\ldots,\vec{x}_{t-1}) = p(\vec{x}_t|\vec{x}_{t-1})$ impliziert, so kann mittels Marginalisierung über die vorherigen Zustände $\vec{x}_{t-1}$ dieser Term überführt werden in

$$p(\vec{x}_t|\mathcal{I}_{t-1}) = \int_{-\infty}^{\infty} p(\vec{x}_t,\vec{x}_{t-1}|\mathcal{I}_{t-1})\mathrm{d}\vec{x}_{t-1} = \int_{-\infty}^{\infty} p(\vec{x}_t|\vec{x}_{t-1})p(\vec{x}_{t-1}|\mathcal{I}_{t-1})\mathrm{d}\vec{x}_{t-1}. \tag{3.3}$$

Eingesetzt in Gleichung 3.2 ergibt dies nunmehr die Möglichkeit, die gewünschte Wahrscheinlichkeitsdichte über dem Zustandsraum rekursiv durch die Berechnungsvorschrift

$$p(\vec{x}_t|\mathcal{I}_t) = \kappa_t\ p(\underset{\sim}{I}_t|\vec{x}_t) \int_{-\infty}^{\infty} p(\vec{x}_t|\vec{x}_{t-1})p(\vec{x}_{t-1}|\mathcal{I}_{t-1})\mathrm{d}\vec{x}_{t-1} \tag{3.4}$$

zu erhalten. Beschrieben durch obige Gleichung läßt sich somit die a-posteriori Wahrscheinlichkeitsdichte $p(\vec{x}_{t-1}|\mathcal{I}_{t-1})$ zum Zeitpunkt $t-1$ durch eine Zustandsübergangswahrscheinlichkeit $p(\vec{x}_t|\vec{x}_{t-1})$ zur a-priori Wahrscheinlichkeitsdichte

$p(\vec{x}_t|\mathcal{I}_{t-1})$ entwickeln und daraus über eine Messung $p(\underset{\sim}{I}_t|\vec{x}_t)$ die a-posteriori Wahrscheinlichkeitsdichte $p(\vec{x}_t|\mathcal{I}_t)$ für den aktuellen Zeitpunkt t ermitteln. Zur konkreten Ausgestaltung der Wahrscheinlichkeitsdichte $p(\vec{x}_t|\vec{x}_{t-1})$ wird auf Bewegungsmodelle der Art

$$\vec{x}_t = \underset{\sim}{A}\vec{x}_{t-1} + \vec{u}_t \tag{3.5}$$

zurückgegriffen, um mithilfe einer Bewegungsmatrix $\underset{\sim}{A}$ und einem Rauschanteil $\vec{u}_t$, welcher als normalverteilt $\mathcal{N}(0, \underset{\sim}{\Sigma}_u)$ angenommen wird[1], den aktuellen Zustand auf Basis des vorhergehenden zu prädizieren. Bei Wahl eines solchen Bewegungsmodells gilt für die Zustandsübergangswahrscheinlichkeit

$$p(\vec{x}_t|\vec{x}_{t-1}) \propto \exp\left(-\frac{1}{2}(\vec{x}_t - \underset{\sim}{A}\vec{x}_{t-1})^T \underset{\sim}{\Sigma}_u^{-1}(\vec{x}_t - \underset{\sim}{A}\vec{x}_{t-1})\right). \tag{3.6}$$

Die kontinuierliche Wahrscheinlichkeitsdichtefunktion $p(\vec{x}_t|\mathcal{I}_t)$ läßt sich in der Praxis nur in Ausnahmefällen analytisch ermitteln. Kann für die Dichtefunktion $p(\vec{x}_t|\mathcal{I}_t)$ a-priori von einer unimodalen Verteilung ausgegangen werden, so erlaubt dies die Verwendung eines Kalmanfilters zur rekursiven Abschätzung der a-posteriori Wahrscheinlichkeitsdichte $p(\vec{x}_t|\mathcal{I}_t)$. Im Allgemeinen ist jedoch diese Annahme gerade im Zuge des Trackingproblems bei komplexeren Videodaten mit beispielsweise stark strukturierten Hintergrundbereichen nicht aufrecht zu erhalten. Daher wird für gewöhnlich über numerische Näherungsverfahren wie z. B. Partikelfilter (auch bekannt als sequentielle Monte-Carlo Simulation) versucht, die Wahrscheinlichkeitsdichte $p(\vec{x}_t|\mathcal{I}_t)$ zu approximieren. Als ein Vertreter dieser stochastischen Simulationsmethoden wurde 1998 der Condensation-Algorithmus[2] von Isard u. Blake [47, 49] veröffentlicht. Dieser basiert wesentlich auf dem Prinzip des *Factored Sampling* (vgl. Grenander u. a. [39]), wonach allgemein eine Funktion $f(x)$, deren zwei Faktoren $f_1(x)$ und $f_2(x)$ bekannt sind, mit Hilfe von N_S Stützwerten $\vec{h}_i$, im weiteren Partikel genannt, angenähert werden kann. Wie in Abbildung 3.2 visualisiert, entsteht zunächst auf Basis des Faktors $f_1(x)$ ein Satz von N_S Partikeln dadurch, dass N_S Werte zufällig gemäß der Funktion $f_1(x)$ ausgewählt werden[3]. Jedem dieser Partikel $\vec{h}_i$ wird über die Funktion

[1] Sowohl die Bewegungsmatrix als auch der Parameter $\underset{\sim}{\Sigma}_u$ können anhand von repräsentativen und annotierten Videosequenzen ermittelt werden. In der Praxis werden jedoch diese Parameter oftmals aufgrund plausibler Überlegungen gewählt, wodurch sich meist vergleichbare Ergebnisse einstellen.

[2] Condensation stellt eine Wortneuschöpfung dar, die von dem diesem Algorithmus zugrunde liegenden Prinzip der Conditional Density Propagation herrührt.

[3] Bei diesem als *Abtasten* (engl. „*Sampling*“) bezeichneten Schritt wird nach dem Prinzip „Ziehen mit Zurücklegen“ gehandelt, es können somit die gleichen x-Werte auch mehrmals unter den Partikeln auftauchen.

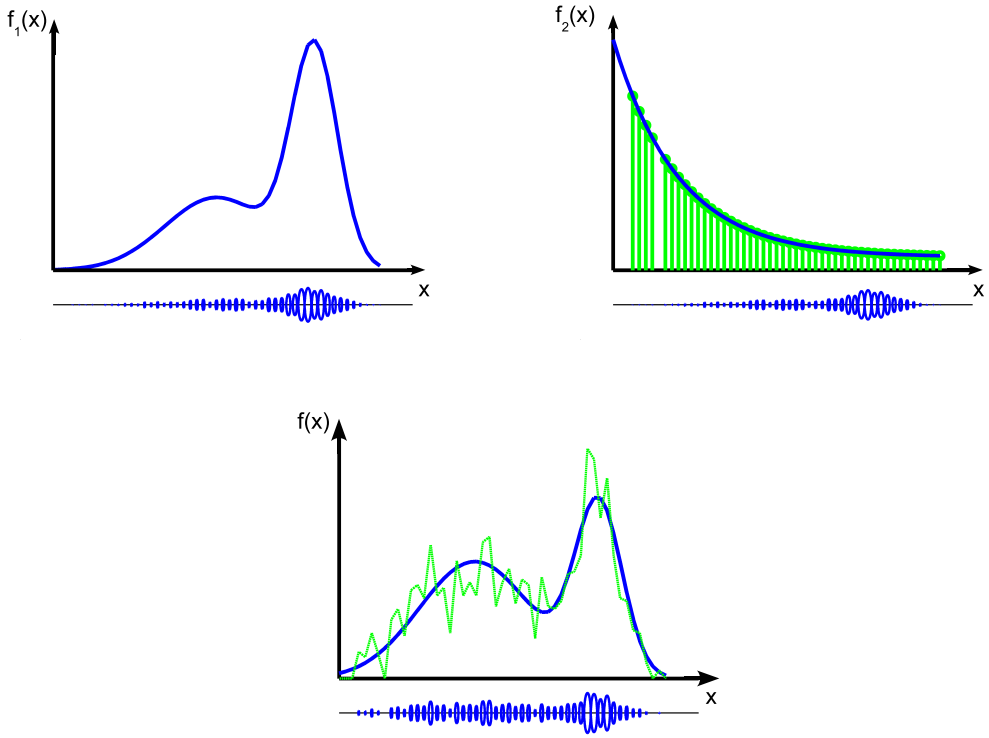


Abbildung 3.2 – Prinzip des *Factored Sampling*: Eine Funktion, die faktorisierbar ist in zwei Terme $f_1(x)$ und $f_2(x)$ kann durch einen Satz an Partikeln (vgl. jeweils die unter den Graphen angetragenen Ellipsen) approximiert werden, indem zunächst zufällig gemäß der Funktion $f_1(x)$ Partikel erzeugt werden. In der Darstellung repräsentiert die Größe der Ellipse die Häufigkeit der jeweiligen Partikel. Anschließend wird jedem Partikel ein Gewicht gemäß der Funktion $f_2(x)$ zugewiesen (vgl. jeweils Höhe der grünen Linien) und von dem dadurch beschriebenen Partikelsatz erneut zufällig, nun aber gemäß der Verteilung der Partikelgewichte, ein Satz von Partikeln gezogen. Die Häufigkeit dieser neuen Partikel stellt dann unmittelbar die Approximation der Funktion $f(x)$ dar.

$f_2(x)$ ein Gewicht

$$\pi_i = \frac{f_2(\vec{h}_i)}{\sum_{j=1}^{N_S} f_2(\vec{h}_j)} \tag{3.7}$$

zugewiesen, resultierend in einem Partikelset $\mathcal{S} = \{\vec{h}_i, \pi_i\}$ mit $i \in \{1, ..., N_S\}$. Mittels erneutem zufälligem „Ziehen mit Zurücklegen“ von diesem Partikelset $\mathcal{S}$ entsteht schließlich ein Partikelset $\tilde{\mathcal{S}}$, dessen Verteilung die Funktion $f(x)$ approximiert und für die Grenzbetrachtung $N_S \to \infty$ exakt wiedergibt. Übertragen auf das Ausgangsproblem, dargestellt durch Gleichung 3.4, ergibt sich demnach

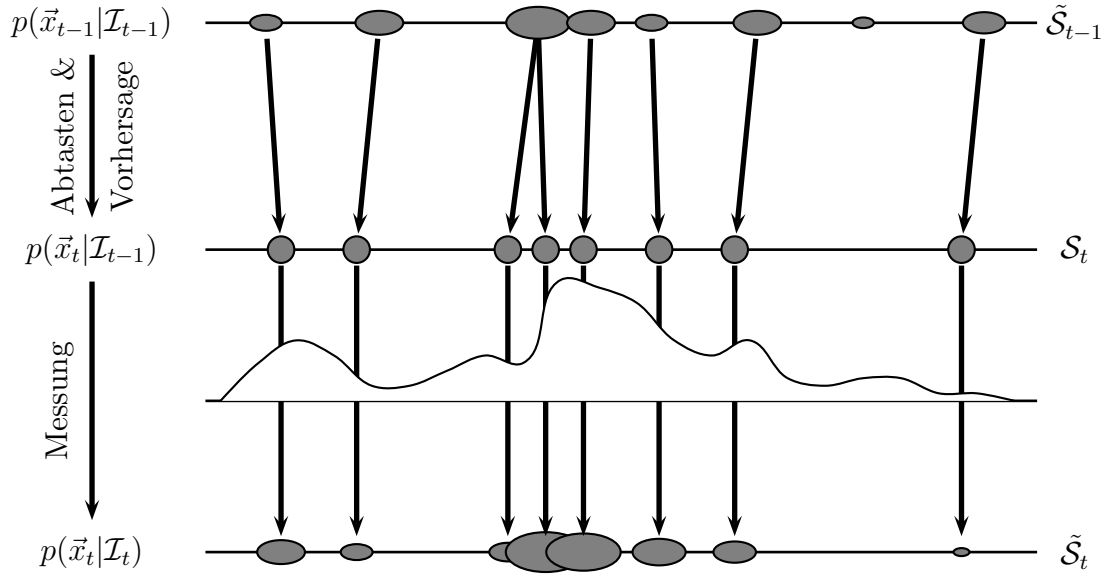


Abbildung 3.3 – Visualisierung der für jeden Zeitschritt im Zuge des Condensation-Algorithmus vollzogenen iterativen Prozesse „Messen“, „Abtasten“ und „Vorhersagen“. Die Größe der Ellipsen zeigt auch hier wiederum die Höhe des zugehörigen Gewichtes an.

folgende rekursive Berechnungsmöglichkeit auf Basis diskreter Zustände (vgl. hierzu auch Abbildung 3.3):
Ausgehend von einem Partikelset $\tilde{\mathcal{S}}_{t-1}$, dessen Zustandsverteilung die Wahrscheinlichkeitsdichte $p(\vec{x}_{t-1}|\mathcal{I}_{t-1})$ approximiert, läßt sich durch eine zufällige Selektion („Abtasten“) von N_S Partikeln gemäß der Gewichte $\pi_{t-1,i}$, gefolgt von einer Prädiktion („Vorhersage“) derselben mittels eines rauschbehafteten Bewegungsmodells (vgl. Gleichung 3.5) ein neues, ungewichtetes Partikelset $\mathcal{S}_t$ erzeugen, welches die Verteilung

$$p(\vec{x}_t|\mathcal{I}_{t-1}) = \int_{-\infty}^{\infty} p(\vec{x}_t|\vec{x}_{t-1}) p(\vec{x}_{t-1}|\mathcal{I}_{t-1}) \mathrm{d}\vec{x}_{t-1} \tag{3.8}$$

repräsentiert. Anschließend werden gemäß der Factored Sampling Vorschrift Gewichte mittels einer „Messung“ erzeugt, indem für jedes Partikel $\vec{h}_{t,i} \in \mathcal{S}_t$ die Wahrscheinlichkeit $p(\underline{I}_t|\vec{x}_t = \vec{h}_{t,i})$ evaluiert wird. Hieraus resultiert ein aktualisiertes Set $\tilde{\mathcal{S}}_t$, welches die gesuchte Wahrscheinlichkeitsverteilung $p(\vec{x}_t|\mathcal{I}_t)$ approximiert.
Gerade im Zuge der Einzelpersonenverfolgung, wie sie nachfolgend erläutert wird, bietet es sich aufgrund des a-priori Wissens, dass nur eine einzige Person im Bild sichtbar ist, speziell für Messfunktionen mit einer sehr unscharfen De-

tektionscharakteristik an, durch eine abschließende, gewichtete Mittelung über alle Hypothesen eine sehr stabile Lokalisation der Person zu realisieren.

3.2 Einzelpersonenverfolgung

Während in zahlreichen Ansätzen für eine Vielzahl von Anwendungen Personen oftmals in ihren gesamten Ausmaßen von Kopf bis Fuß modelliert werden, erweist sich ein derartiges Vorgehen, speziell vor dem Hintergrund der in dieser Arbeit im Fokus stehenden Besprechungsszenarien, als nicht zielführend. Dies begründet sich vor allem in der Tatsache, dass Besprechungsräume typischerweise flächenmäßig nur sehr geringe Ausmaße annehmen und sich Personen darin überwiegend in der Nähe von oder direkt an den Konferenztischen aufhalten, wodurch Personen oftmals erst von der Hüfte aufwärts im Kamerabild sichtbar sind. Da ferner für eine ebenfalls interessierende Emotionserkennung oder Personenidentifikation Wissen über das Gesicht der Personen, insbesondere betreffend die Position und Größe, von grundlegender Bedeutung ist, konzentrieren sich die meisten Verfahren, die eine videobasierte Detektion und Verfolgung von Personen in Besprechungsszenarien zum Ziel haben (vgl. Bernardin u. Stiefelhagen [14], Potucek u. a. [76], Schreiber u. Rigoll [90, 91], Smith u. a. [101]), überwiegend auf das Gesicht respektive den Kopf als dasjenige Merkmal eines Menschen, welches aufgrund seiner Bedeutung für die zwischenmenschliche Kommunikation als besonders repräsentativ für die Person selbst erachtet werden kann und daher im Folgenden jeweils als Synonym für Person verwendet wird[4]. Es werden daher in den folgenden Abschnitten mehrere Modellierungsmöglichkeiten für den menschlichen Kopf vorgestellt, die geeignet sind, in einem hybriden Trackingsystem die Rolle der Detektionsstufe zu übernehmen und somit als Messfunktion für die Ermittlung der Partikelgewichte zu fungieren.

3.2.1 Modellierung von Köpfen mittels Ellipsen

Ein Modell, welches durch seine extreme Einfachheit besticht, basiert auf den Veröffentlichungen von Birchfield [16, 17]. Hierbei macht man sich die Tatsache zu Nutze, dass der Kopf in einer groben Näherung eine ovale Form aufweist, und verwendet eine formfeste Ellipse mit einem fixen Achsenverhältnis, um omnidirektionale Kopfansichten ohne jeglichen Trainingsaufwand mit einem einzigen,

[4] Die – streng genommen nicht korrekte – Gleichsetzung der Begriffe Kopf und Person hat sich im Forschungsbereich Tracking durchgesetzt (vgl. beispielsweise Gatica-Perez u. a. [35]) und wird in dieser Arbeit ebenso verwendet.

aber aufgrund der einfachen Annahme nur bedingt anpassungsfähigen Modell zu beschreiben. Ausgehend von der allgemeinen Ellipsengleichung

$$\left(\frac{x-t_x}{s}\right)^2+\left(\frac{y-t_y}{sr}\right)^2=1 \tag{3.9}$$

läßt sich ein derartiges Modell bei festem Achsenverhältnis r durch einen Satz von Parametern, nämlich den Euklid'schen Transformationsparametern Translation $\vec{t}=(t_x,t_y)^T$ und Skalierung s vollständig beschreiben. Birchfield realisierte basierend auf diesem Modell einen sehr einfachen bottom-up Trackingansatz, indem er die Parameter der Ellipse mittels eines linearen Bewegungsmodells zunächst prädiziert und anschließend eine erschöpfende Suche nach dem in Abschnitt 2.2.2 beschriebenen Grundprinzip innerhalb eines lokalen Bereichs um den prädizierten Ort der Ellipse durchführte. Dazu wird eine Bewertungsfunktion

$$\Omega_{\mathrm{Ell}}=\frac{1}{N_{\mathrm{Pkt}}}\sum_{i=1}^{N_{\mathrm{Pkt}}}|\vec{n}(\vec{p}_i)^T\vec{g}(\vec{p}_i)| \tag{3.10}$$

eingeführt, bei der für jeden Pixel $\vec{p}_i=(x_i,y_i)^T,\ \ i\in\{1,...,N_{\mathrm{Pkt}}\}$ entlang des Ellipsenrandes das Skalarprodukt zwischen der durch den jeweiligen Pixel verlaufenden Normalen $\vec{n}(\vec{p}_i)$ und des an der jeweiligen Position vorliegenden Gradienten $\vec{g}(\vec{p}_i)$ berechnet und über alle N_{Pkt} Randpixel gemittelt wird. Das dafür benötigte Gradientenbild wird über eine Sobelfilterung mittels einer 3×3 Matrix aus dem Grauwertbild $\mathcal{G}$ gemäß der folgenden Faltungsvorschrift erzeugt:

$$\mathcal{G}_x=\begin{pmatrix}1&0&-1\\2&0&-2\\1&0&-1\end{pmatrix}*\mathcal{G}\qquad \mathcal{G}_y=\begin{pmatrix}1&2&1\\0&0&0\\-1&-2&-1\end{pmatrix}*\mathcal{G} \tag{3.11}$$

Über diese beiden Teilbilder $\mathcal{G}_x$ und $\mathcal{G}_y$ ergibt sich pixelweise das betragsmäßige Gradientenbild

$$\mathcal{G}(x,y)=\sqrt{(\mathcal{G}_x(x,y))^2+(\mathcal{G}_y(x,y))^2} \tag{3.12}$$

und der zugehörige Richtungswinkel

$$\eta(x,y)=\arctan\frac{\mathcal{G}_y(x,y)}{\mathcal{G}_x(x,y)}. \tag{3.13}$$

Über einen Schwellwertoperator wird abschließend für jede der innerhalb des Suchraumes auf Basis der Kostenfunktion Ω_{Ell} evaluierten Ellipse eine binäre

Entscheidung über das Vorliegen eines Kopfes mit den durch die Ellipsenparameter gegebenen Eigenschaften herbeigeführt.
Die Idee eines hybriden Trackingsystems verfolgend, wurde der Ansatz im Zuge dieser Arbeit dahingehend modifiziert, dass dieses Ellipsenmodell zum einen in eine Partikelfilterarchitektur integriert wurde und zum anderen entgegen der von Birchfield beschriebenen Wirkungsweise eine Adaption auf die zugrunde liegenden Bilddaten erlaubt. Hierzu werden die im Abschnitt 3.1 noch allgemein formulierten Systemzustände $\vec{x}_t$ als Realisierungen von Ellipsen interpretiert, womit jedes aus dem Zustandsraum abgetastete Partikel $\vec{h}_{t,i} = (t_x, t_y, s)^T$ durch die Parameter einer konkreten Ellipse spezifiziert ist. Ziel des hybriden Tracking-

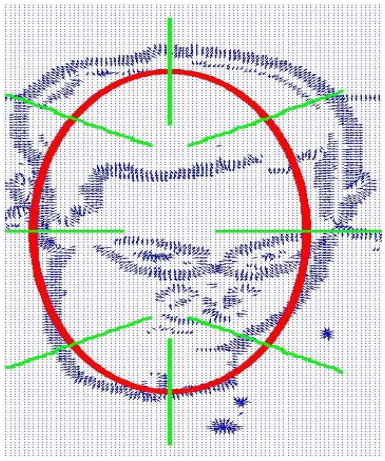


Abbildung 3.4 – Normiertes Gradientenbild, wie es durch die Sobelfilterung entsteht. Die blauen Pfeile darin repräsentieren die jeweilige Richtung des Gradienten. Das Ellipsenmodell (rot) wird anschließend anhand der Gradienten derart modifiziert, dass die Richtung der Normalen (grün) durch die Randpunkte des Modells (in der Abbildung exemplarisch für acht Stützpunkte visualisiert) möglichst parallel zu den Gradienten, die sich im Bild an den Positionen der jeweiligen Randpunkte ergeben, ausgerichtet ist.

prozesses ist es, das Modell oder genauer dessen durch ein jeweiliges Partikel gegebenen Parameter aufgrund der Bilddaten so anzupassen, dass es eine evtl. vorhandene Kopfposition zu beschreiben vermag (vgl. Abbildung 3.4). Grundlage für den Adaptionsprozess der Parameter ist hierbei eine Kostenfunktion

$$\Omega'(\vec{p}_i, \vec{p}_j) = |\vec{n}(\vec{p}_i)^T \vec{g}(\vec{p}_j|. \tag{3.14}$$

Um basierend auf dem Bildinhalt eine Korrektur der Parameter der Ellipse zu erreichen, wird für jeden Pixel $\vec{p}_i$ auf dem Ellipsenrand eine Gerade $\vec{l}$ orthogonal

zur Ellipse aufgestellt und innerhalb einer δ-Umgebung zum Punkt $\vec{p}_i$ entlang dieser Geraden derjenige Pixel aus $\vec{p}_j \in \vec{l}$ ermittelt, der die Kostenfunktion Ω' maximiert:

$$\hat{\vec{p}}_i = \underset{\vec{p}_j \in \vec{l}}{\operatorname{argmax}} \left(\Omega'(\vec{p}_i, \vec{p}_j) \right) . \tag{3.15}$$

Durch die Gesamtheit aller so erhaltenen Schätzpunkte $\hat{\vec{p}}_i$ werden anschließend die Parameter einer modifizierten Ellipse derart berechnet, dass die Summe der quadratischen Abstände zwischen den Punkten auf dem Rand der Ellipse und den Werten $\hat{\vec{p}}_i$ minimal wird. Iterativ wird damit eine Anpassung des Modells an die vorliegenden Bilddaten erzielt, bis sich schließlich die Ellipsenparameter nicht mehr signifikant ändern und damit der Prozess terminiert. Abschließend erfolgt eine qualitative Evaluierung der erhaltenen Ellipse auf Basis der Bildinformation mittels der Bewertungsfunktion Ω_{Ell}, welche sich als Wahrscheinlichkeit für das tatsächliche Vorliegen eines Kopfes mit der durch die Ellipse beschriebenen Gestalt interpretieren läßt und damit unmittelbar als Maß für die Messung $p(\mathcal{I}_t|\vec{x}_t = \vec{h}_{t,i})$ eines Partikels genutzt wird.

3.2.2 Active Shape Modelle

Eine weitere Möglichkeit zur Modellierung von Köpfen besteht darin, diese nicht als formfeste, sondern vielmehr als in ihrem Erscheinungsbild veränderliche Objekte nachzubilden. Ziel hierbei ist es wiederum, omnidirektionale Kopfansichten in einem einzigen Modell zu erfassen. Deswegen wird für diesen Ansatz erneut die Form des Kopfes, nun aber eben nicht als starr angenommen, in Verbindung mit den Schultern verwendet. Eine Methodik, die sich speziell zur Parametrisierung solcher formveränderlichen Objekte eignet, sind die von Cootes u. a. [27] vorgestellten *Active Shape Modelle* (ASM)[5]. Bei diesem Verfahren wird in einem vorgelagerten Trainingsprozess anhand zahlreicher Beispielbilder zuerst objektspezifisches Wissen extrahiert und in einem statistischen Formmodell gelernt. Ein Objekt wird dazu ausschließlich über seine Objekthülle[6] definiert, d. h. über all diejenige geometrische Information, welche nach Beseitigung der Euklid'schen Parameter (Translation, Rotation und Skalierung) verbleibt. Im Gegensatz dazu soll mit Objektkontur die mit den Euklid'schen Transformationsparametern behaftete Objekthülle bezeichnet werden. Im Folgenden wird zunächst der Aufbau

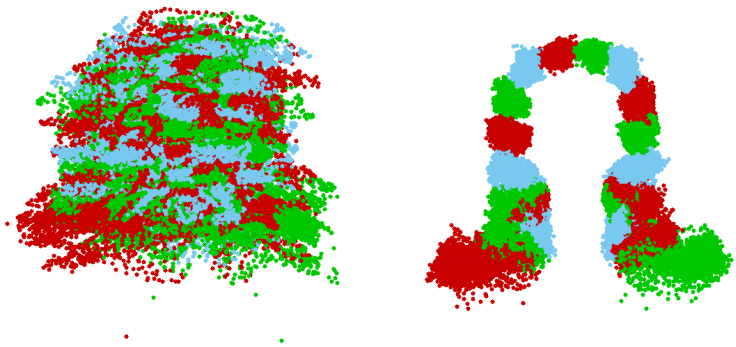
[5] Active Shape Modelle sind die logische Weiterentwicklung der *Snakes/Active Contours* von Kass u. a. [57] und werden deshalb auch als *Smart Snakes* bezeichnet.

[6] Die Hülle eines Objektes wird im Englischen auch als *Shape* bezeichnet, wodurch sich der Ausdruck Active Shape Modell ableitet.

eines solchen Modells skizziert, bevor auf die Vorgehensweise zur Lokalisation von Objekten mittels dieser Modelle eingegangen wird.

Aufbau eines statistischen Formmodells

Wie bereits angedeutet, bilden Objektkonturen die Ausgangsbasis zur Erstellung des statistischen Formmodells. In einem Annotationsprozess wird hierfür zunächst in N_{Bsp} Beispielbildern die Kontur durch einen Satz von N_{Pkt} Stützpunkten[7] $\vec{p}_j = (x_j, y_j)^T$, die in unterschiedlichen Konturbeispielen jeweils derselben geometrischen Position auf dem Objekt entsprechen, in diskreter Weise beschrieben. Auf diese Art erhält man für jede Beispielkontur i eine geordnete Punktmenge $\vec{P}_i = (\vec{p}_{i,1}^T, \ldots, \vec{p}_{i,N_{\text{Pkt}}}^T)^T$, die jedoch aufgrund der beschriebenen Vorgehensweise noch transformationsbehaftet ist (vgl. Abbildung 3.5a). Um



(a) (b)

Abbildung 3.5 – Typische Verteilung der einzelnen Stützpunkte (jeweils in unterschiedlicher Farbe dargestellt) einer Kontur, wie sie unmittelbar nach dem Annotieren, d.h. transformationsbehaftet, vorliegt (a) und das nach dem Ausrichten der Bilder zueinander, also durch Entfernen der Euklid'schen Transformationsparameter erhaltene Resultat (b). Während in den originären Daten noch keinerlei statistischer Zusammenhang erkennbar ist, offenbart die transformationsfreie Darstellung der Trainingsbeispiele diese Information unmittelbar.

beim Aufbau des Formmodells nur die tatsächliche Information der Objekthülle nutzen zu können, werden die Punktmengen $\vec{P}_i$ sämtlicher Trainingsobjekte

[7] In Anlehnung an den im Englischen gebräuchlichen Ausdruck werden die Stützpunkte im Folgenden auch als *Landmarks* bezeichnet.

mittels der *Generalisierten Prokrustes Analyse* (GPA) nach Gower [37] aufeinander ausgerichtet. Bei der GPA handelt es sich um eine multivariate statistische Analysemethode, mit deren Hilfe die Summe der quadratischen Abstände

$$D = \sum_{i=1}^{N_{\text{Bsp}}} ||\vec{P}_i - \vec{P}'||_2^2 \tag{3.16}$$

von N_{Bsp} Punktmengen $\vec{P}_i$ zum Mittelwert $\vec{P}' = \frac{1}{N_{Bsp}} \sum_{i=1}^{N_{\text{Bsp}}} \vec{P}_i$ aller Punktmengen durch Bestimmung der jeweiligen Transformationsparameter Translation $t_i = (t_{i,x}, t_{i,y})^T$, Rotation θ_i und Skalierung s_i minimiert wird. Wie anhand der Gleichung unmittelbar ersichtlich, kann diese Minimierung und somit die Bestimmung der Euklid'schen Transformationsparameter nur iterativ erfolgen, da der finale Mittelwert $\vec{P}'$ a-priori unbekannt ist und sich dieser ändert, falls wenigstens eine der Punktmengen $\vec{P}_i$ eine Transformation erfährt. Wird o. B. d. A. davon ausgegangen, dass sich der Schwerpunkt sämtlicher Punktmengen bereits deckt und damit $t_{i,x} = t_{i,y} = 0$ für alle $i \in \{1, \ldots, N_{\text{Bsp}}\}$ gilt, so lassen sich die gesuchte Skalierung s_i und Rotation θ_i für die i-te Punktmenge auf Basis einer Transformationsabbildung

$$T(\vec{p}) = s \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \vec{p} \tag{3.17}$$

mittels des quadratischen Fehlers

$$E_i = \sum_{j=1}^{N_{\text{Pkt}}} ||T(\vec{p}_{i,j}) - \vec{p}_j{}'||_2^2 \quad \text{für} \quad \vec{p}_{i,j} \in \vec{P}_i, \quad \vec{p}_j{}' \in \vec{P}' \tag{3.18}$$

durch Nullsetzen der partiellen Ableitungen $\frac{\partial E_i}{\partial s_i}$ sowie $\frac{\partial E_i}{\partial \theta_i}$ mit $(x_{i,j}, y_{i,j}) \in \vec{P}_i$ und $(x'_j, y'_j) \in \vec{P}'$ bestimmen zu:

$$s_i = \frac{1}{||\vec{P}_i||_2^2} \sqrt{\left(\sum_{j=1}^{N_{\text{Pkt}}} x_{i,j} y'_j - y_{i,j} x'_j \right)^2 + \left(\sum_{j=1}^{N_{\text{Pkt}}} x_{i,j} x'_j + y_{i,j} y'_j \right)^2} \tag{3.19}$$

$$\theta_i = \arctan \frac{\sum_{j=1}^{N_{\text{Pkt}}} x_{i,j} y'_j - y_{i,j} x'_j}{\sum_{j=1}^{N_{\text{Pkt}}} x_{i,j} x'_j + y_{i,j} y'_j}. \tag{3.20}$$

Die korrespondierenden Landmarks der nunmehr transformationsfreien Punktmengen $\vec{P}_i^*$ streuen – wie in Abbildung 3.5b dargestellt – jeweils mit einer gewissen Varianz um die Mittelwertkontur. Diese Streuung rührt, da die betrachteten Punktmengen transformationsfrei sind, offensichtlich von der Formvariabilität des zu modellierenden Objektes her. Zur Erfassung dieser Formvariabilitäten werden über die *Hauptachsentransformation* (PCA) die Eigenvektoren

$\vec{\psi}_i,\ \ i \in \{1, ..., N_{\text{Bsp}}\}$ mitsamt der zugehörigen Eigenwerte $\lambda_i, i \in \{1, ..., N_{\text{Bsp}}\}$ aus der Kovarianzmatrix

$$\underset{\sim}{\Sigma} = \frac{1}{N_{\text{Bsp}} - 1} \sum_{i=1}^{N_{\text{Bsp}}} (\vec{P}_i^* - \vec{P}')(\vec{P}_i^* - \vec{P}')^T \tag{3.21}$$

ermittelt. Im Zuge der Modellbildung werden nun all diejenigen N Eigenvektoren, für die nach der Größe ihrer Eigenwerte in absteigender Reihenfolge sortiert die Bedingung

$$\sum_{i=1}^{N} \lambda_i \geq 0,98 \sum_{i=1}^{N_{\text{Bsp}}} \lambda_i \tag{3.22}$$

gerade noch erfüllt ist[8], zu einer Matrix $\underset{\sim}{\Psi} = (\vec{\psi}_1, \ldots, \vec{\psi}_N)$ zusammengefasst. Nach

Abbildung 3.6 – Exemplarische Darstellung der durch variierende Gewichtung zwischen $-3\sqrt{\lambda_i}$ und $3\sqrt{\lambda_i}$ der ersten drei Eigenvektoren erzielbaren Formänderungen: Während der erste Eigenvektor augenscheinlich die Kopfform von schmal bis rundlich beeinflusst, zeichnet sich der zweite Eigenvektor maßgeblich für die Erfassung von Kopfdrehungen verantwortlich. Der dritte Eigenvektor bildet hauptsächlich die Schulterpartie ab, die einerseits von der Körperstatur abhängt, mitunter aber auch beispielsweise durch die Drehung ins Profil verändert wird.

Rückprojektion der gewichteten Eigenvektoren in den Ursprungsraum lassen sich durch die Modellgleichung

$$\vec{P}^* \approx \vec{P}' + \underset{\sim}{\Psi}\vec{b} \tag{3.23}$$

neue Objekthüllen synthetisieren, wobei die Formveränderungen über den Gewichtungsvektor $\vec{b}$ gesteuert werden können (siehe Abbildung 3.6). Zur Vermeidung klassenuntypischer Muster, wie sie durch eine weitere Aussteuerung der

[8] Der Faktor $0,98$ wird hierbei in der Literatur (vgl. Cootes u. a. [27]) als oftmals ausreichend erachtet, um einerseits ein großes Spektrum an Variationen des Modells zu erlauben, aber gleichzeitig auch Rauschen, welches beim Annotieren mitunter entsteht, zu mindern.

jeweiligen Eigenvektoren entstehen würden, wird eine Variation der einzelnen Gewichte hierbei nur in einem festen Wertebereich zugelassen, der durch die jeweiligen Eigenwerte, also die Variation in Richtung des Eigenvektors, festgelegt wird[9].
Durch Hinzunahme der Euklid'schen Transformationsparameter lassen sich somit über die Beziehung

$$\vec{p}_{i,j} = s_i \begin{pmatrix} \cos\theta_i & -\sin\theta_i \\ \sin\theta_i & \cos\theta_i \end{pmatrix} \vec{p}^{\,*}_{i,j} + \vec{t}_i, \quad \vec{p}^{\,*}_{i,j} \in \vec{P}^{\,*}_i \tag{3.24}$$

innerhalb der Modellrestriktion zulässige Objektkonturen erzeugen.

Objektlokalisation

Das trainierte formveränderliche Modell läßt sich neben der reinen Synthetisierung von Objektkonturen ebenso zur Lokalisation von Objekten in Bildern anwenden. Lokalisation soll hierbei im Gegensatz zu den bisher im Rahmen dieser Arbeit vorgestellten Detektionstechniken, die gemeinhin auf Vorkommnisse von Objekten in einer erschöpfenden Suche mittels einer Abtaststrategie schließen, bedeuten, dass die Position eines Objektes ausgehend von einer Initialisierung durch Adaption ermittelt wird. Hierfür ist eine möglichst präzise Startschätzung nötig, die a-priori jedoch oftmals nicht gegeben ist. Aus diesem Grund bietet sich die Objektlokalisation basierend auf ASM als äußerst geeignet für die Verwendung in einer hybriden Trackingarchitektur an, sodass mit jedem Partikel automatisch eine Initialschätzung gegeben ist, welche eindeutig festgelegt wird durch den Partikelzustand $\vec{x}_t = \vec{h}_{t,i} = (t_x, t_y, s_i, \theta_i, \vec{b}_i)^T$, bestehend aus den Euklid'schen Parametern, sowie dem Formparameter $\vec{b}_i$, mit dem die Gestalt des ASM beeinflusst wird. Ausgehend von den Initialparametern wird durch

$$\vec{P}^{\,*}_i = \vec{P}' + \underset{\sim}{\Psi}\vec{b}_i \tag{3.25}$$

gefolgt von der durch Gleichung 3.24 beschriebenen Transformation eine Objektkontur generiert, die anschließend aufgrund der vorliegenden Bilddaten modifiziert wird. In ihrer Veröffentlichung schlagen Cootes u. a. [27] hierfür ein grauwertbasiertes, lokales Optimierungsverfahren vor. Hierzu wird für sämtliche Trainingsbilder neben der eigentlichen Position der Landmarks zusätzlich der Grauwertverlauf entlang der Normalen durch den Landmark innerhalb einer δ-Umgebung mit erfasst. Werden sämtliche Verläufe über alle N_{Bsp} Trainingsbeispiele für jeden Landmark gemittelt, so ergeben sich N_{Pkt} stützpunktspezifische

[9] In der einschlägigen Literatur (vgl. Cootes u. a. [27]) hat sich hier die empirisch ermittelte Faustformel $|b_i| \leq 3\sqrt{\lambda_i},\ \ i \in \{1, \ldots, N_{\mathrm{Bsp}}\}$ als sehr brauchbarer Grenzwert erwiesen.

Grauwert-Templates. Für die bilddatengetriebene Optimierung des Konturmodells wird anschließend iterativ jeweils entlang der durch das vorliegende Modell bestimmten Stützpunkt-Normalen für jeden Landmark derjenige Pixel ermittelt, für den die Korrelation zwischen Grauwertverlauf im Bild und dem gelernten Template maximiert wird. Ein derartiges Vergehen erweist sich jedoch insbesondere für stark strukturierte Hintergründe sowie sehr variable Texturen innerhalb der durch das Konturmodell beschriebenen Objekte oftmals als nachteilig. Daher wurden im Rahmen dieser Arbeit zwei alternative Strategien verfolgt:

Gradientenbasierte Suche Die von Cootes u. a. [27] vorgeschlagene Methodik weist für natürliche Szenarien einige Unzulänglichkeiten auf, die durch den im Folgenden beschriebenen Ansatz umgangen werden. Wie in Abbildung 3.7 an einem einfachen Beispiel gezeigt, würde ein Vergleich von gelernten und den tatsächlich im Bild vorliegenden Grauwertverläufen jeweils entlang der Normalen durch den Landmark „A“ die Position „B“ als neuen Landmark der Kontur ermitteln und somit zu einem nicht gewünschten Ergebnis führen. Stattdessen erscheint es plausibler, explizit die Richtung von Kanten (in der Abbildung 3.7 durch grüne Pfeile dargestellt) für eine Neupositionierung der Stützpunkte zu verwenden. Aus diesem Grund wird die Kostenfunktion in dieser Arbeit definiert über das Skalarprodukt zwischen dem Normalenvektor $\vec{n}(\vec{p})$ und dem Gradientenvektor $\vec{g}(\vec{p})$:

$$\Omega_{\mathrm{ASM_1}} = \sum_{j=1}^{N_{\mathrm{Pkt}}} (\vec{n}(\vec{p}_j)^T \vec{g}(\vec{p}_j)). \tag{3.26}$$

Iterativ ergibt sich damit durch eine – äquivalent zu Gleichung 3.15 – in einer δ-Umgebung zum betrachteten Pixel $\vec{p}_i$ entlang der Normalen $\vec{l}$ vorzunehmenden Suche die neue Landmarkposition

$$\hat{\vec{p}}_i = \underset{\vec{p}_j \in \vec{l}}{\mathrm{argmax}} \left(\Omega'(\vec{p}_i, \vec{p}_j) \right). \tag{3.27}$$

Bedingt durch diese Modifikation werden Kanten im Gradientenbild, welche parallel zum jeweiligen Normalenvektor verlaufen, entsprechend höher bewertet. Dadurch wird vor allem bei stark strukturierten Hintergründen ein besseres Konvergenzverhalten erzielt, was sich auch in den im Kapitel 4 geschilderten Evaluierungen entsprechend widerspiegelt.

Gabor-Wavelet basierte Suche Eine weitere Technik zur Adaption eines ASM an vorliegende Bilddaten basiert auf Gabor-Wavelet[10] Merkmalen (vgl. Ar-

[10] Bei Gabor-Wavelets handelt es sich um biologisch motivierte Faltungskernel, deren Filterantworten denen der einfachen Zellen des visuellen Kortex ähneln (vgl. Daugman [29]).

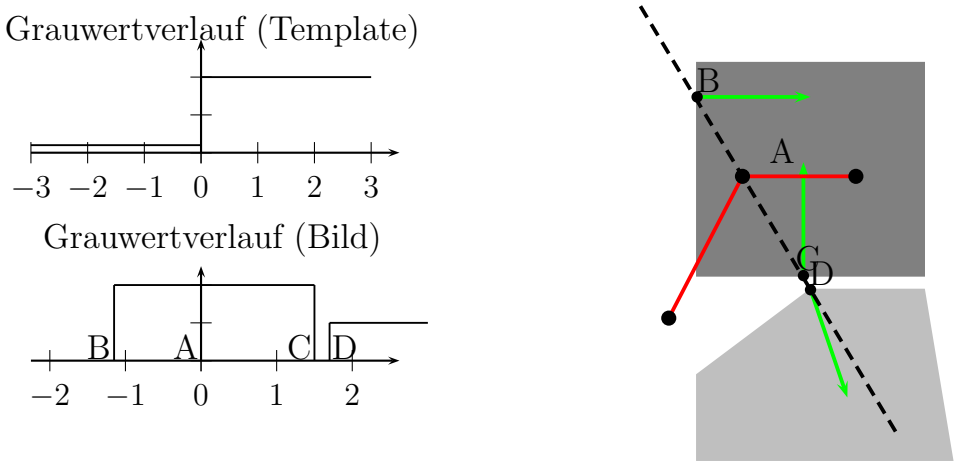


Abbildung 3.7 – Visualisierung der Adaptionsstrategie für den Landmark „A“ einer Modellkontur (rot) an einem synthetischen Beispiel mit zwei Objekten (grau bzw. hellgrau angedeutet): Entlang der Normalen (gestrichelte Linie im Schaubild rechts) durch den Landmark „A“ ergibt sich ein Grauwertverlauf, wie er im Diagramm links unten dargestellt ist. Anhand dieses Verlaufs wird derjenige Pixel ermittelt, für den die Korrelation zwischen trainiertem (Template links oben) und dem an betreffender Position tatsächlich vorliegendem Grauwertverlauf maximiert wird; in vorliegendem Beispiel wäre dies der Punkt „B“. Vor diesem Hintergrund erscheint es plausibler, durch Beachtung der Information über die Kantenrichtungen (durch Pfeile repräsentierter Bildgradient der auf der Normalen befindlichen Pixel) denjenigen Pixel zu lokalisieren, der eine zur Normalen möglichst parallele Ausrichtung des dort zugrunde liegenden Bildgradienten hat. Ein derartiges Vorgehen ermittelt für den vorliegenden Fall eine Verschiebung des Punktes „A“ zur Position „D“, was zu einer offensichtlich besseren Übereinstimmung von Modellkontur und Objektkante führt als im Fall einer Verschiebung des Punktes „A“ nach „B“.

ca u. a. [8], Jiao u. a. [51]). Diese in den 80er Jahren von Daugman [29] erstmals auf Bildverarbeitungsprobleme angewandte Form von Wavelets zeigt dabei ein sehr gutes örtliches Frequenzauflösungsverhalten bei gleichzeitiger Berücksichtigung der Nachbarschaftsbeziehung zwischen den Bildpunkten. Beschrieben wird die Familie der Gabor-Wavelets durch die Kernelfunktion

$$\psi_j(\vec{p}) = \vec{k_j}^T \vec{k_j} \exp\left(-\frac{\vec{k_j}^T \vec{k_j} \vec{p}^T \vec{p}}{2\sigma^2}\right) \left[\exp\left(i\vec{k_j}^T \vec{p}\right) - \exp\left(-\frac{\sigma^2}{2}\right)\right], \qquad (3.28)$$

basierend auf einer gauß-gefensterten komplexwertigen Wellenfunktion[11] mit dem Wellenvektor

$$\vec{k}_j = \begin{pmatrix} k_{jx} \\ k_{jy} \end{pmatrix} = \begin{pmatrix} k_j \cos\phi_j \\ k_j \sin\phi_j \end{pmatrix} \quad \text{mit} \quad j \in \mathbb{N}, \tag{3.29}$$

der sowohl durch die Orientierung ϕ_j als auch die Frequenz k_j parametrisiert ist. Im Rahmen von Bildverarbeitungsaufgaben werden Merkmale bei Nutzung von Gabor-Wavelets häufig mittels einer Filterbank bestehend aus 40 Wavelets ψ_j (vgl. Jiao u. a. [51], Wiskott u. a. [116]), deren Wellenvektor $\vec{k}_j$ acht diskrete Orientierungen bei fünf unterschiedlichen Frequenzstufen annimmt (vgl. Abbildung 3.8), extrahiert:

$$\begin{aligned} k_j &= k_j(v) = 2^{-\frac{v+2}{2}}\pi \quad && \text{mit} \quad v = 0,\ldots,4 \\ \phi_j &= \phi_j(u) = u\tfrac{\pi}{8} && \text{mit} \quad u = 0,\ldots,7 \end{aligned} \quad \text{und} \quad j = u + 8v \tag{3.30}$$

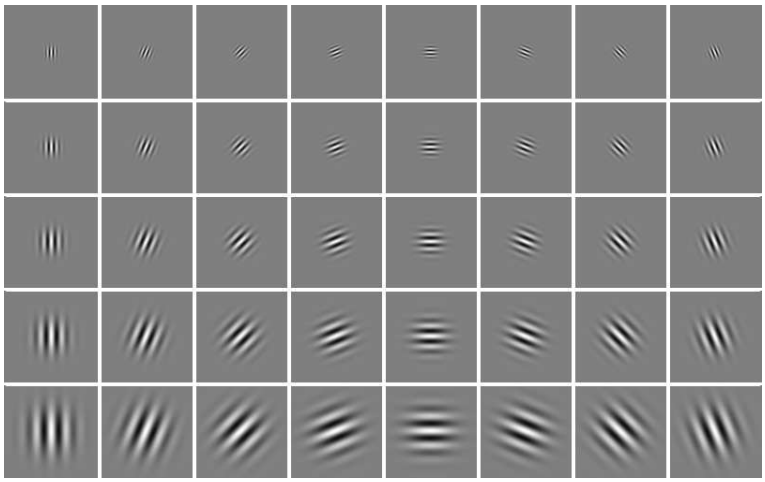


Abbildung 3.8 – Filterbank bestehend aus Gabor-Wavelets mit acht unterschiedlichen Orientierungen bei fünf verschiedenen Skalierungsstufen.

Basierend auf dieser Wavelet-Bank wird für einen Bildpunkt $\vec{p}$ mit Helligkeitswert $\mathcal{G}(\vec{p})$ ein 40-dimensionaler Merkmalsvektor $\vec{J}$, im Folgenden als Jet bezeichnet, definiert:

$$\vec{J}(\vec{p}) = \begin{pmatrix} J_1(\vec{p}) \\ \vdots \\ J_{40}(\vec{p}) \end{pmatrix} \quad \text{mit} \quad J_j(\vec{p}) = \int_{-\infty}^{\infty} \mathcal{G}(\vec{p}\,')\psi_j(\vec{p} - \vec{p}\,')\mathrm{d}\vec{p}\,' \tag{3.31}$$

[11] Der Term $\exp\left(-\frac{\sigma^2}{2}\right)$ stellt sicher, dass $\int \psi_j(\vec{p})\mathrm{d}\vec{p} = 0$ gilt und damit die Kernelfunktion keinen Gleichanteil aufweist.

Um anhand der Jets eine lokale Optimierung der Stützpunkte des Modells vornehmen zu können, werden während des Modellaufbaus durch das Annotieren nicht mehr nur die relativen Positionen der einzelnen Landmarks zueinander sowie deren Varianzen gelernt, sondern zusätzlich auch für jeden Landmark $\vec{p}_i$ ein mittlerer Jet

$$\vec{J}'(\vec{p}_i\,') = \frac{1}{N_{\mathrm{Bsp}}} \sum_{j=1}^{N_{\mathrm{Bsp}}} \vec{J}(\vec{p}_{i,j}) \quad \text{mit} \quad \vec{p}_{i,j} \in \vec{P}_i \tag{3.32}$$

über alle N_{Bsp} Trainingskonturen ermittelt. Während des Adaptionsprozesses des Modells an die vorliegenden Bildinformationen ist es Ziel, diejenigen Pixel $\hat{\vec{p}}_j$ zu finden, die für einen bestimmten Landmark $\vec{p}_j$ die Ähnlichkeit zwischen dem gelernten Jet $\vec{J}'(\vec{p}_j)$ und dem für die Position $\hat{\vec{p}}$ errechneten Jet $\vec{J}(\hat{\vec{p}})$ maximiert. Hierzu wird jedes Merkmal in seine äquivalente Darstellung mittels Betrag $\vec{a}$ und Phase $\vec{\Phi}$ überführt und damit eine Ähnlichkeitsfunktion

$$S(\vec{J}, \vec{J}') = \frac{\sum_{i=1}^{40} a_i a_i' \cos(\Phi_i - \Phi_i')}{\sqrt{\sum_{i=1}^{40} a_i^2 \sum_{i=1}^{40} a_i'^2}} \tag{3.33}$$

zwischen zwei Jets $\vec{J}$ und $\vec{J}'$ definiert. Während die Amplitude der Gabor-Merkmale zwar eine gewisse Unempfindlichkeit gegenüber einer mäßigen Translation sowie Rotation aufweist, reagiert die Phase bereits auf kleinste Euklid'sche Veränderungen. Aus diesem Grund eignet sich insbesondere die Phaseninformation der Faltungsantwort, um den räumlichen Verschiebungsvektor $\vec{t} = (t_x, t_y)^T$ zwischen den beiden Jets zu bestimmen. Zu diesem Zweck wird die Ähnlichkeitsfunktion $S(\vec{J}, \vec{\bar{J}})$ aus Gleichung 3.33 modifiziert zu

$$S(\vec{J}, \vec{J}') = \frac{\sum_{i=1}^{40} a_i a_i' \cos(\Phi_i - \Phi_i' - \vec{tk}_j)}{\sqrt{\sum_{i=1}^{40} a_i^2 \sum_{i=1}^{40} a_j'^2}} \tag{3.34}$$

und durch Nullsetzen der partiellen Ableitungen $\frac{\partial S}{\partial t_x} = \frac{\partial S}{\partial t_y} = 0$ der Verschiebungsvektor bestimmt:

$$\vec{t}(\vec{J}, \vec{J}') = \begin{pmatrix} t_x \\ t_y \end{pmatrix} = \frac{1}{\Gamma_{xx}\Gamma_{yy} - \Gamma_{xy}\Gamma_{yx}} \begin{pmatrix} \Gamma_{yy} & -\Gamma_{yx} \\ -\Gamma_{xy} & \Gamma_{xx} \end{pmatrix} \begin{pmatrix} \Phi_x \\ \Phi_y \end{pmatrix} \tag{3.35}$$

mit

$$\Phi_x = \sum_{i=1}^{40} a_i a_i' k_{jx} (\Phi_i - \Phi_i'), \tag{3.36}$$

$$\Phi_y = \sum_{i=1}^{40} a_i a_i' k_{jy} (\Phi_i - \Phi_i'), \tag{3.37}$$

$$\Gamma_{xx} = \sum_{i=1}^{40} a_i a_i' k_{jx} k_{jx}, \tag{3.38}$$

$$\Gamma_{yy} = \sum_{i=1}^{40} a_i a_j' k_{jy} k_{jy} \quad \text{und} \tag{3.39}$$

$$\Gamma_{xy} = \Gamma_{yx} = \sum_{i=1}^{40} a_i a_i' k_{jx} k_{jy} \tag{3.40}$$

Mittels dieser Feinkorrektur können Verschiebungsvektoren mit maximal der halben Wellenlänge des jeweils höchstfrequenten Kernels, der im Jet $\vec{J}$ Berücksichtigung findet, geschätzt werden[12]. Dadurch läßt sich mittels eines iterativen Prozesses für einen Jet $\vec{J}(\vec{p})$ an Position $\vec{p}$ diejenige Position $\vec{p}'$ in einer lokalen Umgebung von $\vec{p}$ ermitteln, für welche die Ähnlichkeit $S(\vec{J}(\vec{p}'), \vec{J}')$ zu einem gegebenen Jet $\vec{J}'$ maximiert wird. Hierzu wird für die gegebene Position $\vec{p}$ zunächst unter ausschließlicher Betrachtung aller Orientierungen der Frequenzstufe k_4 der Verschiebungsvektor $\vec{t}(\vec{J}(\vec{p}), \vec{J}')$ mittels Gleichung 3.35 ermittelt. Anschließend wird für den sich ergebenden modifizierten Ausgangspunkt $\hat{\vec{p}} = \vec{p} + \vec{t}(\vec{J}(\vec{p}), \vec{J}')$ ein neuer Jet $\vec{J}(\hat{\vec{p}})$ berechnet. Durch sukzessive Hinzunahme der jeweils acht Gaborkernel aus der nächst höherfrequenten Stufe wird über weitere vier Iterationen die Positionierung der Landmarks verfeinert. Nach Einbeziehung aller fünf Frequenzstufen in die Ähnlichkeitsberechnung resultiert für jeden Landmark ein Pixel $\hat{\vec{p}}$, dessen Gaborjet $\vec{J}(\hat{\vec{p}})$ maximale Ähnlichkeit zu demjenigen Jet $\vec{J}'$ aufweist, der durch Mittelung über die Trainingsdaten entstanden ist. Eine abschließende Bewertung der endgültig konvergierten Objektkontur wird schließlich durch einen Vergleich der Jets $\vec{J}(\hat{\vec{p}})$ dieser Kontur mit dem gemittelten Jet $\vec{J}'$ aus dem Training vorgenommen:

$$\Omega_{\mathrm{ASM}_2} = S(\vec{J}(\hat{\vec{p}}), \vec{J}') \tag{3.41}$$

[12] Bei der vorgestellten Wavelet-Bank können demnach Verschiebungsvektoren mit einer Länge von bis zu acht Pixel bei alleiniger Nutzung der Kernelfunktionen mit der Frequenz k_4 bestimmt werden.

Unabhängig von dem konkret zur Optimierung der Landmarks herangezogenen Verfahren stellen die neuen Stützpunkte $\hat{\vec{P}} = (\hat{\vec{p}}_1, \ldots, \hat{\vec{p}}_{N_{\mathrm{Pkt}}})^T$ im Allgemeinen keine gültige Objektmodellierung mehr dar. In einem nachgelagerten Schritt gilt es, denjenigen Gewichtungsvektor $\vec{b}^*$ des Konturmodells zu bestimmen, welcher der neuen Punktekonstellation am besten gerecht wird, d. h. die Distanz zwischen der neuen (transformationsbehafteten) Punktmenge $\hat{\vec{P}}$ und dem Modell $\vec{P}' + \underset{\sim}{\Psi}\vec{b}$ minimiert. Iterativ werden hierzu ausgehend von einer (im ersten Schritt mit willkürlich gewähltem Gewichtungsvektor $\vec{b}$ initialisierten) Modellkontur $\vec{P}' + \underset{\sim}{\Psi}\vec{b}$ zunächst die Euklid'schen Formparameter s, θ und $\vec{t}$ nach den aus der Modellgenerierung bekannten Gleichungen 3.19 und 3.20 bestimmt, welche die neue Punktekonstellation bestmöglich auf diese Modellkontur ausrichtet. Hieraus resultiert die transformierte Punktekonstellation $\vec{P}^*$. Anschließend wird durch Auflösen von Gleichung 3.25 der entsprechende Gewichtungsvektor zu

$$\vec{b}^* = \underset{\sim}{\Psi}^T(\vec{P}^* - \vec{P}') \tag{3.42}$$

bestimmt. Mit der durch diesen Gewichtungsvektor $\vec{b}^*$ neu entstandenen Modellkontur beginnt der Prozess wiederum von neuem und wiederholt sich solange, bis sowohl die Schätzung der Euklid'schen Transformationsparameter, als auch die Berechnung des Gewichtungsvektors konvergieren. Mittels einer abschließenden neuerlichen Bewertung des Modells anhand der Gleichungen 3.26 bzw. 3.41 kann somit die im Rahmen des Partikelfilterprozesses zu messende Wahrscheinlichkeit $p(\mathcal{I}_t|\vec{x}_t = \vec{h}_{t,i})$ zur Verfügung gestellt werden.

3.3 Mehrpersonenverfolgung

Wird das für die Einzelpersonenverfolgung entwickelte Verfahren entsprechend auf Szenarien mit mehreren Personen angewandt, so beobachtet man, dass nach einer anfänglichen Einschwingzeit sämtliche Partikel oftmals auf einer einzigen Position im Bild konvergieren und somit eine simultane Verfolgung mehrerer Personen unmöglich wird (vgl. Abbildung 3.9). Um ein derartiges Verhalten zu vermeiden, wenden Isard u. Maccormick [48] das Konzept der Partikelfilterung nicht mehr unmittelbar auf Objektzustände $\vec{x}_t = \vec{h}_{t,i}$ an, sondern auf Objektkonfigurationen $\mathcal{H}_t = \{N, \vec{h}_{t,1}, \ldots, \vec{h}_{t,N}\}$ variabler Länge an. Jede dieser Objektkonfigurationen besteht dabei aus der Anzahl N der darin enthaltenen Einzelobjekte und den Objektzuständen $\vec{x}_t = \vec{h}_{t,i}$. Basierend auf dieser Modellierung ergeben sich daraus für eine maximale Zahl $N_{\max}$ von zeitgleich zu verfolgenden Personen $2^{N_{\max}}$ unterschiedliche Objektkonfigurationen, wobei jede von diesen wiederum

Abbildung 3.9 – Exemplarische Visualisierung der im Zuge der simultanen Mehrpersonenverfolgung resultierenden Probleme: eine globale Mittelung über die im Bild verstreuten Hypothesen – wie im Fall der Einzelpersonenverfolgung zur Stabilisierung des Tracks angewandt – führt bei der simultanen Verfolgung mehrerer Personen zwangsläufig zu falschen Ergebnissen. Als wesentlich problematischer stellt sich jedoch die Tatsache dar, dass über die Zeit sämtliche Hypothesen mittel- bis langfristig auf der (einen) Position im Bild konvergieren, für welche die Messung die beste Gewichtung liefert. In der beispielhaft betrachteten Bildfolge, die im Abstand von 1,2 Sekunden aufgenommen wurde, konzentrieren sich die anfangs noch auf beiden Personen verteilten Hypothesen bedingt durch die Kopfdrehung der im Bild links befindlichen Person und damit einhergehend mit einer Verschlechterung der Hypothesengewichte ausschließlich auf die im Bild rechts sitzende Person.

durch verschiedene Parametrierungen der einzelnen Objektzustände $\vec{h}_{t,i}$ differieren kann. Um ein robustes Tracking zu erreichen, ist bei dieser Technik hierfür eine sehr hohe Zahl an Partikeln notwendig[13], was bei komplexen Objektmodellierungen mitunter zu einem erheblichen Rechenaufwand führen kann.
Zur Vermeidung solch hoher Rechenzeitanforderungen wurden im Rahmen dieser Arbeit zwei neuartige Ansätze entwickelt, die das zur Verfolgung einzelner Personen erfolgreich eingesetzte System entsprechend erweitern und einen überproportionalen Anstieg der benötigten Zahl an Partikeln umgehen.

3.3.1 Mehrschichtiger Partikelfilter

In einem ersten Ansatz (vgl. Schreiber u. Rigoll [90]) wurde ein hierarchisch strukturiertes Konzept ausgearbeitet, resultierend in einem Systemaufbau wie

[13] In ihrer Publikation nennen Isard u. Maccormick [48] eine Zahl von 10000 Partikel, um bis zu drei Personen in einem Szenario verfolgen zu können.

in Blockschaltbild 3.10 dargestellt. Im Gegensatz zu dem oben beschriebenen

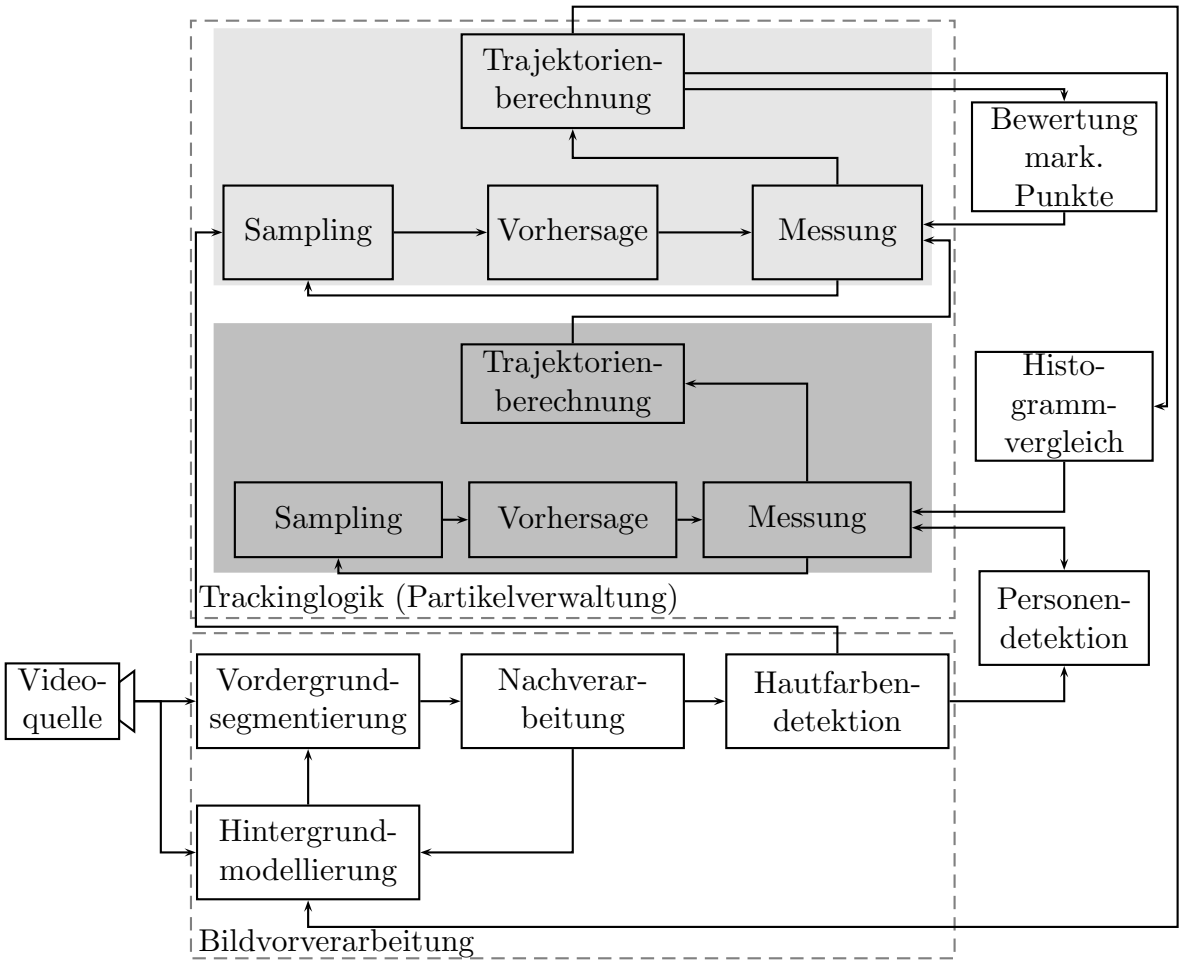


Abbildung 3.10 – Blockschaltbild des Trackingsystems nach Schreiber u. Rigoll [90] zur simultanen Verfolgung mehrerer Personen, realisiert als zweischichtige Partikelfilter-Struktur: während der hellgrau unterlegte Bereich die Detektion der Personenkonstellation koordiniert, wird durch den dunkelgrau markierten Bereich das Einzelpersonentracking vorgenommen.

Verfahren nach Isard u. Maccormick [48] wird die Fähigkeit zum simultanen Tracking mehrerer Personen dadurch hergestellt, dass in dem entwickelten System eine Partikelfilterung auf zwei Ebenen vollzogen wird. In dieser Weise wird durch die Separierung der Detektionsaufgabe von dem Problem der Bestimmung der vorliegenden Personenkonfiguration[14] eine signifikante Reduzierung der insgesamt benötigten Hypothesenzahl auf ca. $25 \cdot N_{\max}$ möglich, wobei $N_{\max}$ die

[14]Mit dem Begriff *Personenkonfiguration* wird die räumliche Verteilung von Personen in der 2D-Projektion des Kamerabildes bezeichnet.

maximale Zahl zeitgleich zu verfolgender Personen beschreibt. So obliegt es dem hierarchisch höhergelegenen Partikelfilter (in Abbildung 3.10 hellgrau unterlegt), das bei der Verfolgung von Einzelpersonen irrelevante Korrespondenzproblem, also die Zugehörigkeit von Partikeln zu einem bestimmten Objekt, zu lösen, während die Aufgabe der Basisschicht (in Abbildung 3.10 dunkelgrau hervorgehoben) nach wie vor die eigentliche Personenverfolgung bleibt.
Zu diesem Zweck wird das aus Abbildung 3.1 bekannte System um einen zusätzlichen Partikelfilter $\mathcal{S}_t^* = \{\vec{H}_{t,i}, \Pi_{t,i}\}$ mit $i \in \{1, \ldots, N_{S'}\}$ erweitert. Jeder Partikel $\vec{H}_{t,i}$ dieses Filters repräsentiert dabei selbst wiederum ein eigenes System $\mathcal{S}_{t,i}$ zur Einzelpersonenverfolgung mitsamt aller darin befindlichen aktuellen Hypothesen $\vec{h}_{t,i}$ und deren zugehörige Gewichte $\pi_{t,i}$. Ähnlich wie auf Ebene des Systems zur Einzelpersonenverfolgung werden neue Konfigurationszustände $H_{t,i}$ auf Basis der Hautfarbendetektion erzeugt, die dann den durch die Partikel repräsentierten Raum der möglichen Konfigurationen entsprechend erweitern. Während des Samplingschrittes können so neue Partikel in den Filterprozess Einzug halten und dadurch neue Konfigurationen evaluiert werden. Über ein lineares, rauschbehaftetes Bewegungsmodell mit $\mathcal{N}(0, \Sigma_u)$-verteilter Rauschgröße $\vec{u}_t$ werden diese Partikel für den nächsten Zeitschritt prädiziert:

$$\vec{H}_{t+1,i} = \underset{\sim}{A}\vec{H}_{t,i} + \vec{u}_t. \tag{3.43}$$

Die Aktualisierung der Gewichte $\Pi_{t,i}$ dieser Hypothesen $\vec{H}_{t,i}$ wird maßgeblich gesteuert durch die Partikelfilter zur Einzelpersonenverfolgung (Basisschicht), die der betrachteten Hypothese zugrunde liegen. Während jedoch im Fall der Einzelpersonenverfolgung ein Abdriften einzelner Partikel $\vec{h}_{t,i}$ aufgrund fehlender weiterer Objekte im Bild, die durch ein Partikel entsprechend repräsentiert werden können, hinnehmbar war, ist genau dies im Zuge des Mehrpersonentracking nicht mehr tragbar, da dann bei örtlicher Nähe zweier Objekte Partikel möglicherweise wiederum auf nur einem Objekt konvergieren und eine spätere Separierung nicht mehr möglich ist. Die alleinige Nutzung der durch die Personendetektion gewonnenen Messgröße vermag das Problem der Partikelallokation nicht endgültig zu lösen. Daher wird die Aktualisierung der Partikelgewichte $\Pi_{t,i}$ – um einem Abdriften der Partikel vorbeugen zu können – auf der Fusion dreier Teilmessungen basierend umgesetzt, die jeweils auf unterschiedlichen Merkmalen beruhen und das Kontextwissen aus dem vorherigen Trackingresultat einbeziehen. Die zur Fusion herangezogenen Messgrößen sind im Einzelnen:

Messung durch das Objektmodell Mittels der vorliegenden Bilddaten läßt sich anhand des zum Tracking eingesetzten Objektmodells eine Bewertung aller durch die Partikel $\vec{h}_{t,i}$ beschriebenen Bildausschnitte vornehmen, wie es

im vorangegangenen Abschnitt zur Einzelpersonenverfolgung beschrieben wurde. Resultat dieser Bewertung ist jeweils eine Messgröße Ω_i (beispielsweise durch die Gleichung 3.10 oder 3.26 bzw. 3.41), welche die Wahrscheinlichkeit für das Vorliegen eines Kopfes in dem betrachteten Bildausschnitt widerspiegelt. Durch Mittelung über alle N_S Partikel des jeweiligen Filters erhält man ein Maß für die Güte der durch die Partikel gegebenen Beschreibung der Bildinformation:

$$\Omega_\mathrm{Modell} = \frac{1}{N_\mathrm{S}} \sum_{i=1}^{N_\mathrm{S}} \Omega_i \tag{3.44}$$

Histogrammvergleich Für ein gegebenes Objekt $\vec{T}_j$, wie es durch die Mittelung über die Partikel $\vec{h}_{t,i}$ eines Einzelpersonenverfolgungssystems $\mathcal{S}_{t,j}$ beschrieben und mittels eines Histogrammes $\vec{H}_1$ charakterisiert ist, wird durch eine gemittelte Ähnlichkeitsmessung, basierend auf einem Histogrammvergleich für jedes Partikel $\vec{h}_{t,i}$ (mit Histogramm $\vec{H}_{2,i}$) eine Texturähnlichkeit gemäß der in Abschnitt 2.3.1 eingeführten Bhattacharyya-Distanz (vgl. Gleichung 2.35) über die b Histogrammeinträge berechnet:

$$\Omega_\mathrm{Hist} = \frac{1}{N_\mathrm{S}} \sum_{i=1}^{N_\mathrm{S}} \left(-\log \sum_{j=1}^{b} \sqrt{\vec{H}_1(j)\vec{H}_{2,i}(j)} \right) \tag{3.45}$$

Durch Mittelung über diese Messwerte lassen sich so Rückschlüsse auf die Streuung und die Stabilität der durch die einzelnen Partikel $\vec{h}_{t,i}$ erfassten Bildausschnitte und damit letztlich auf das tatsächliche Vorhandensein eines Kopfes ziehen.

Bewertung mittels markanter Punkte Während der Histogrammvergleich lediglich gesamtheitlich die dem Objekt $\vec{T}_j$ zugrunde liegende Textur in Form einer Statistik berücksichtigt, werden durch die Ähnlichkeitsbewertung anhand markanter Punkte[15] die geometrischen Lagebeziehungen von besonderen Merkmalen der Textur explizit modelliert. Hierzu werden innerhalb eines gegebenen Objektes aus dem vorhergehenden Zeitschritt über den Harrisoperator (vgl. Harris u. Stephens [42]) markante Punkte in der Weise detektiert, dass zunächst für jeden Pixel $\vec{p}$ des Bildausschnittes G^* zur

[15] Unter markanten Punkten sollen hierbei solche Punkte verstanden werden, die in einer lokalen Umgebung möglichst einzigartig sind. Im Folgenden wird sich hierbei im wesentlichen auf aussagekräftige Ecken beschränkt.

lokalen Beschreibung einer quadratischen Nachbarschaftsstruktur $\underset{\sim}{R}_1$ die Tensormatrix

$$\underset{\sim}{B}(\vec{p}) = \begin{pmatrix} \sum_{\vec{p}\,'\in\underset{\sim}{R}_1} \left(G_x(\vec{p}\,')\right)^2 & \sum_{\vec{p}\,'\in\underset{\sim}{R}_1} G_x(\vec{p}\,')G_y(\vec{p}\,') \\ \sum_{\vec{p}\,'\in\underset{\sim}{R}_1} G_x(\vec{p}\,')G_y(\vec{p}\,') & \sum_{\vec{p}\,'\in\underset{\sim}{R}_1} \left(G_y(\vec{p}\,')\right)^2 \end{pmatrix} \tag{3.46}$$

aufgestellt wird, wobei die Matrizen $\underset{\sim}{G}_x$ sowie $\underset{\sim}{G}_y$ die Richtungsableitungen des Bildausschnittes $\underset{\sim}{G}^*$ bedeuten. Eine Aussage über das Vorliegen einer Ecke an betreffender Position $\vec{p}$ liefert der Rang dieser Matrix, der hierfür notwendigerweise gleich der Dimension der Matrix selbst sein muss. Anstatt der hiermit lediglich binär möglichen Entscheidung für die Markanz eines Punktes wird eine feinere Auswahl aussagekräftiger Punkte durch ein Gütemaß

$$Q(\vec{p}) = \det(\underset{\sim}{B}(\vec{p})) - \kappa \left(\mathrm{spur}(\underset{\sim}{B}(\vec{p}))\right)^2 \tag{3.47}$$

erreicht, welches für sämtliche Pixel $\vec{p}$ auf Basis von Eigenwertbetrachtungen[16] den Grad der Markanz abhängig von einem empirisch ermittelten Faktor[17] κ in einer kontinuierlichen Größe bewertet. Um eine gewisse Mindestdistanz markanter Punkte zu wahren, werden mittels der sog. *Nicht-Maxima-Unterdrückung* innerhalb einer quadratischen Nachbarschaft $\underset{\sim}{R}_2$ alle Pixel ausgeblendet, deren Güte eine der Bedingungen

$$Q(\vec{p}) \;<\; \max_{\vec{p}\,'\in\underset{\sim}{R}_2} Q(\vec{p}\,') \quad \text{oder} \tag{3.48}$$

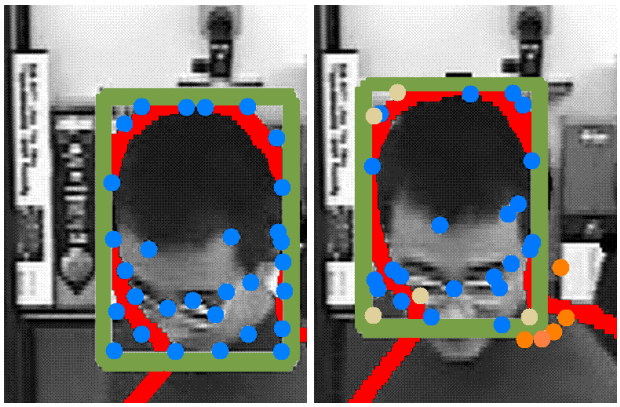
$$Q(\vec{p}) \;<\; \kappa' \max_{\vec{p}\,'\in\underset{\sim}{R}_1} Q(\vec{p}\,') \tag{3.49}$$

genügt. Die verbliebenen Pixel stellen dann im Sinne obiger Definition markante Punkte dar.

Um mittels dieser Punkte eine Ähnlichkeitsbeurteilung von unterschiedlichen Objekten durchzuführen, werden mit jeder Initialisierung eines neuen Objektes stets $N_{\mathrm{mP,max}}$ markante Punkte innerhalb des Objektes detektiert. Diese werden dann mittels eines Verfahrens zur Berechnung des optischen Flusses (z. B. nach Lucas u. Kanade [64]) anhand der Bilddaten des nächsten Zeitschrittes prädiziert, wobei hierdurch markante Punkte außerhalb der zu dem betreffenden Zeitschritt detektierten Objektkontur zu liegen kommen können (vgl. Abbildung 3.11). Alle prädizierten Punkte werden gemäß dem beschriebenen Vorgehen neuerlich anhand des Harrisoperators

[16] Statt die Eigenwerte β_1 und β_2 direkt zu berechnen, werden für das Gütemaß hierbei die Beziehungen $\mathrm{spur}(\underset{\sim}{B}) = \beta_1 + \beta_2$ und $\det(\underset{\sim}{B}) = \beta_1\beta_2$ genutzt.

[17] In der Literatur wird für gewöhnlich $\kappa = 0,04$ gewählt.



Zeitschritt t Zeitschritt $t+1$

Abbildung 3.11 – Bildausschnitte zweier aufeinanderfolgender Zeitschritte, in denen jeweils die detektierte Objektkontur (rot) mit ihrer umgebenden Box (grün) eingezeichnet sind. Jede Box enthält jeweils 30 markante Punkte (farbige Kreise). Wie im rechten Bild ersichtlich, befinden sich nach der Prädiktion mittels des optischen Flusses wiederum zahlreiche Punkte (blau) innerhalb der Box, einige der Punkte (orange) kommen jedoch außerhalb der Box zum Liegen. Diese werden anschließend in einer Reinitialisierungsphase durch neue Punkte (beige) innerhalb der Box ersetzt.

und den nachfolgenden Schritten evaluiert. Hierdurch kann es zum Verschwinden weiterer markanter Punkte innerhalb des Objektes kommen. Auf Basis der Zahl N_{mP} noch im Objekt verbliebener markanter Punkte wird ein Maß

$$\Omega_{\mathrm{mP}} = \frac{N_{\mathrm{mP}}}{N_{\mathrm{mP,max}}} \tag{3.50}$$

definiert, welches anzeigt, wie stark sich die durch das Objekt $\vec{T}_j$ beschriebene Textur in zwei aufeinander folgenden Bildern ändert, wodurch sich die sog. *Sprunghaftigkeit* geometrischer Texturmerkmale für das jeweils betrachtete Objekt qualitativ erfassen läßt. Abschließend wird für das betreffende Objekt in einer Initialisierungsphase mittels des Harrisoperators die Zahl der markanten Punkte wieder zu $N_{\mathrm{mP,max}}$ ergänzt.

Basierend auf diesen drei Messgrößen ergibt sich schließlich die Aktualisierung der Partikelgewichte $\Pi_{t,i}$ über die multiplikative Verknüpfung gemäß der Vorschrift

$$\Pi_{t,i} = \Pi_{t-1,i}\Omega_{\mathrm{Modell}}\Omega_{\mathrm{Hist}}\Omega_{\mathrm{mP}}. \tag{3.51}$$

3.3.2 Simulated Annealing

In einem weiteren Ansatz (vgl. Schreiber u. Rigoll [91]) wurde ein heuristisches Verfahren namens *Simulated Annealing*[18] (SA) benutzt, um eine Mehrpersonenverfolgung zu realisieren.

Grundprinzip des Simulated Annealing

Die Grundlagen für das simulierte Abkühlen wurden von Metropolis u. a. [69] im Jahr 1953 gelegt. Gegenstand seiner Abhandlung war die Suche nach einer allgemeingültigen Methodik zur Simulation des Abkühlprozesses von molekularen Substanzen zur Ableitung von Stoffeigenschaften. Ziel hierbei war es, das aus der Natur wohlbekannte Prinzip, dass Stoffe unter gegebenen Randbedingungen jeweils den günstigsten Energiezustand einzunehmen versuchen, nachzubilden. Dazu modellierte Metropolis Substanzen als zufällige Ansammlung von Teilchen (Atome/Moleküle), die willkürlich Zustandsänderungen vollziehen können. Betrachtet man innerhalb eines Zeitschrittes die Gesamtheit aller Zustandsänderungen bei einer während dieser Zeitspanne als konstant angenommenen Temperatur[19] ϑ, so resultiert daraus eine neue Anordnung der Teilchen, welche sich im Vergleich zur Ausgangskonfiguration durch eine Änderung ΔL in der Gesamtenergiebilanz der Substanz unterscheidet. Bei einer Verringerung der Energiebilanz, also für $\Delta L < 0$, wird diese neue Konfiguration der Teilchen in der Simulation als Ausgangssituation für den folgenden Zeitschritt betrachtet. Für den Fall, dass die Energieänderung ΔL positiv ist, wird die entstandene Anordnung der Partikel nur mit einer (Boltzmann-verteilten) Wahrscheinlichkeit

$$p = \exp\left(-\frac{\Delta L}{k_B \vartheta}\right) \tag{3.52}$$

als zulässig erachtet, wobei k_B die aus der Physik bekannte Boltzmann-Konstante darstellt. Wie aus Formel 3.52 unmittelbar ersichtlich ist, wird eine Energieerhöhung des Systems mit fortschreitendem Absinken der Temperatur ϑ immer unwahrscheinlicher, wodurch der mögliche Zustandsraum sukzessive eingeschränkt und letztlich eine Konvergenz erzwungen wird. Durch dieses Modell konnte der in der Natur beobachtbare Effekt beschrieben werden, dass langsame Abkühlvorgänge zu einer sehr regelmäßigen Struktur führen, während sich die Teilchen beispielsweise beim Abschrecken[20] nur unregelmäßig anordnen können und sich

[18] Wörtlich übersetzt bedeutet Simulated Annealing etwa „simuliertes langsames Abkühlen".

[19] Diese Annahme kann im Hinblick auf den dem Modell zugrunde liegenden, langsamen Abkühlprozess (annealing) als gerechtfertigt angesehen werden.

[20] *Abschrecken* ist der Fachbegriff für Härten von Metallen durch einen sehr raschen Abkühlvorgang.

das Material dann als extrem spröde erweist.
1983 wurde der Ansatz des Metropolis-Algorithmus von Kirkpatrick u. a. [58] weiterentwickelt und übertragen auf kombinatorische Optimierungsprobleme. Seine Idee bestand darin, die Energie, welche als Minimierungsfunktion bei Metropolis diente, auszutauschen gegen eine auf das jeweilige Optimierungsproblem zugeschnittene Kostenfunktion. Analog dazu sind die Zustandsänderungen als Nachbarlösungen und die Konfiguration der Teilchen als Lösung selbst zu interpretieren. Beginnend bei einer hohen „Temperatur“, welche bei den Optimierungsproblemen dann nurmehr als Systemparameter fungiert, wird in mehreren Zeitschritten das Ausgangsproblem optimiert. Nach und nach wird die Temperatur nun abgesenkt, womit - wie aus Gleichung 3.52 ersichtlich - eine Verminderung der Annahmewahrscheinlichkeit für Verschlechterungen der Kostenfunktion einhergeht. Dieser Prozess wird solange fortgeführt, bis sich keinerlei gültige Zustandsänderungen mehr ergeben und die Lösung somit als konvergiert betrachtet werden kann.

Anwendung auf das Problem des Mehrpersonentracking

Wie im vorigen Abschnitt bereits erläutert, stellt die korrekte intertemporäre Abbildung einer Menge von Objekten $\mathcal{T}_{t-1} = \{\vec{T}_{t-1,1}, \ldots, \vec{T}_{t-1,N_{t-1,\mathcal{T}}}\}$ auf eine andere Objektmenge $\mathcal{T}_t = \{\vec{T}_{t,1}, \ldots, \vec{T}_{t,N_{t,\mathcal{T}}}\}$ ein zentrales Problem dar. Während Racine u. a. [80] zur Lösung dieser Problematik ein System allein basierend auf dem Prinzip des SA vorschlagen, wurde im Rahmen dieser Arbeit ein neuartiger Ansatz durch die Kombination von Partikelfilter und Simulated Annealing entwickelt. Hierzu wurde das bestehende System zur Einzelpersonenverfolgung, welches in Kapitel 3.2 vorgestellt wurde, wie in Abbildung 3.12 gezeigt, um eine zusätzliche Kontrollebene erweitert, deren zentraler Bestandteil die Technik des SA ist.
Grundidee hierbei ist es, durch eine graphentheoretische Modellierung der Objektkonfiguration den zeitlichen Kontext zwischen je zwei aufeinanderfolgenden Bildern zu erfassen. Hierzu werden die durch den Trackingalgorithmus ermittelten Objekte $\mathcal{T}_t$ als Knoten V eines Graphen $G = (V, E)$ interpretiert, wobei jedes Objekt $\vec{T}_{t,j}$ charakterisiert wird durch die folgenden Eigenschaften:

a) Schwerpunkt $\vec{t}(\vec{T}_{t,j}) = (t_x, t_y)^T$ des Objektes

b) Fläche $F_{\mathrm{BB}}(\vec{T}_{t,j})$ des umschließenden Rechtecks (engl. „*Bounding Box*“)

c) Vom Objekt tatsächlich eingenommene Fläche $F_{\mathrm{Sil}}(\vec{T}_{t,j})$, festgelegt durch die Silhouette

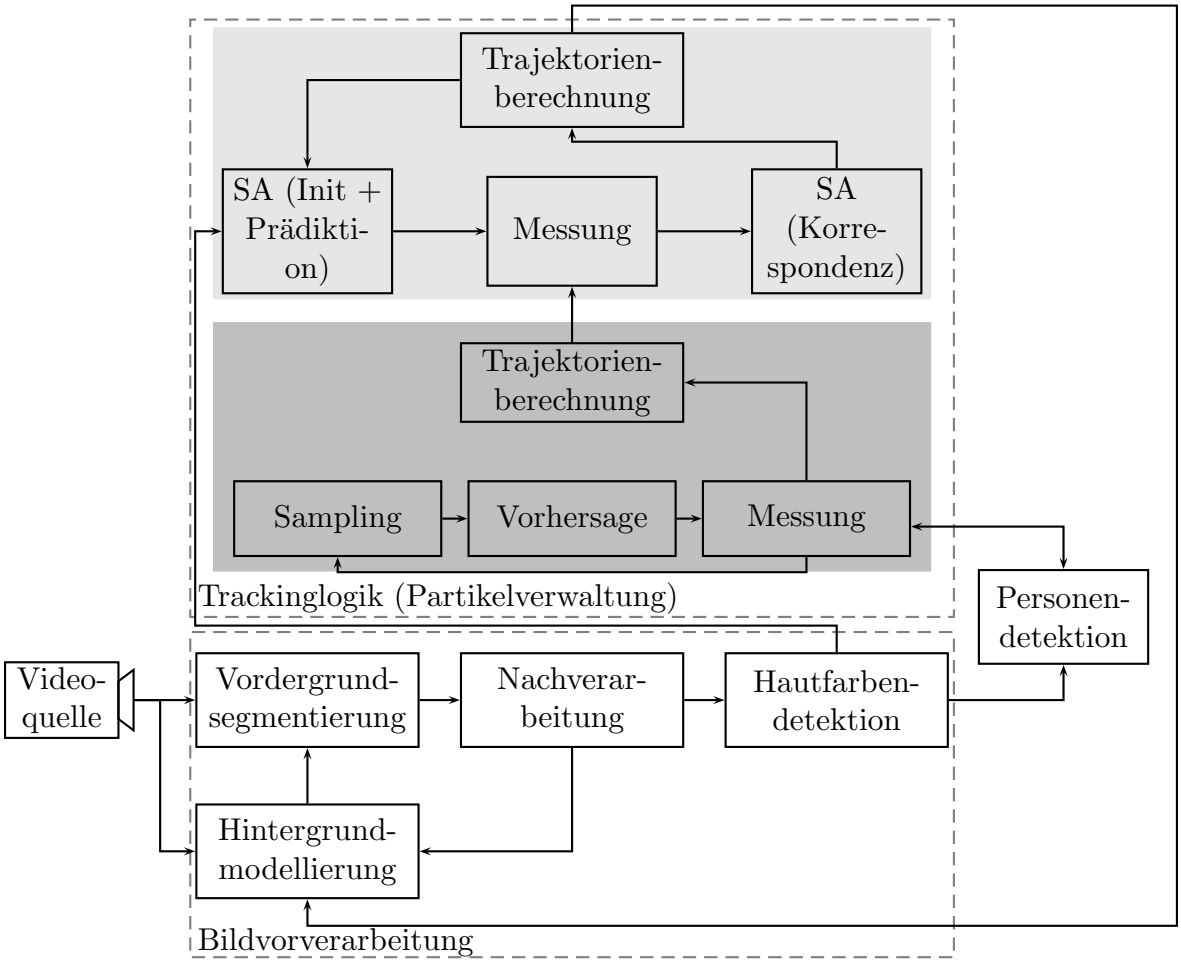


Abbildung 3.12 – Blockschaltbild des Trackingsystems (vgl. Schreiber u. Rigoll [91]) zur simultanen Verfolgung mehrerer Personen, basierend auf Simulated Annealing: Während durch den heuristischen Algorithmus das Objektkorrespondenzproblem gelöst wird (hellgrauer Bereich), kann davon losgelöst in einer tiefer liegenden, probabilistischen Schicht (dunkelgrau markiert) das Einzelpersonentracking vorgenommen werden.

d) Histogramm $\vec{H}(\vec{T}_{t,j})$, welches die statistische Häufigkeit der Grauwerte innerhalb des Objektes erfasst

Die Menge der Kanten E des Graphen wird gebildet aus – zu Beginn willkürlich gewählten – Paaren einander entsprechender Knoten, wobei der jeweilige Startknoten aus der Menge $\mathcal{T}_{t-1}$ und der Endknoten aus der Menge $\mathcal{T}_t$ stammen muss. Hierbei lassen sich grundsätzlich die in Abbildung 3.13 gezeigten fünf Basisereignisse anhand der beteiligten Teilmengen $\mathcal{A} \in \mathcal{T}_{t-1}$ und $\mathcal{B} \in \mathcal{T}_t$ unterscheiden:

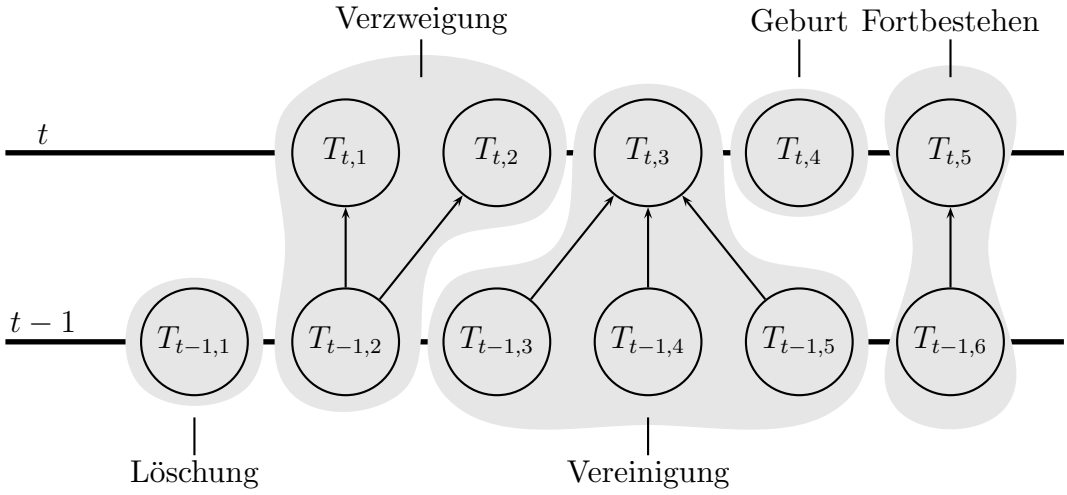


Abbildung 3.13 – Visualisierung der fünf möglichen Basistypen für Kantenkonstellationen, die in einem Graphen vorkommen können.

a) Löschung - Für ein Objekt wird im folgenden Zeitschritt keine Entsprechung gefunden. Es gilt: $\mathcal{A} = \{V_i\}$ und $\mathcal{B} = \{\}$.
b) Geburt - Im aktuellen Bild wurde ein Objekt festgestellt, welches keinen Vorgänger hat. Es gilt: $\mathcal{A} = \{\}$ und $\mathcal{B} = \{V_i\}$.
c) Vereinigung - Aus mehreren Objekten vom vorhergehenden Bild resultiert ein einziges Objekt im aktuellen Zeitschritt (z. B. bedingt durch eine gegenseitige Verdeckung). Es gilt: $\mathcal{A} = \{V_i, \ldots, V_j\}$ und $\mathcal{B} = \{V_k\}$.
d) Aufteilung - Ein Objekt aus dem Zeitschritt $t-1$ zerteilt sich in mehrere Objekte im aktuellen Bild. Es gilt: $\mathcal{A} = \{V_i\}$ und $\mathcal{B} = \{V_j, \ldots, V_k\}$.
e) Fortbestehen - Ein Objekt aus dem Zeitschritt $t-1$ ist im aktuellen Bild wieder zu finden. Es gilt: $\mathcal{A} = \{V_i\}$ und $\mathcal{B} = \{V_j\}$.

Auf dem durch die Knoten (Objekteigenschaften) und die Kanten (Ereignisse) eindeutig spezifizierten Graph G wird ein Energieäquivalent

$$L(G) = \sum_{i=1}^{N_{\text{Events}}} L_i(G) = \sum_{i=1}^{N_{\text{Events}}} L(\mathcal{A}_i, \mathcal{B}_i) \tag{3.53}$$

als Summation über alle Teilenergien $L(\mathcal{A}_i, \mathcal{B}_i)$, wie sie durch die N_{Events} im Graphen vorliegenden Basisereignisse beschrieben werden, definiert. Jede ereignisspezifische Teilenergie berücksichtigt dabei die Objekteigenschaften insofern, als dass zu deren Berechnung die folgenden Ähnlichkeitsmaße herangezogen wer-

den[21]:
Teilenergie bzgl. Schwerpunkte

$$L_{\vec{t}}(\mathcal{A},\mathcal{B}) = \begin{cases} 1 & \text{falls Ereignis Löschung oder Geburt} \\ \frac{||\vec{t}(\mathcal{A})-\vec{t}(\mathcal{B})||_2^2}{\sigma_{\vec{t}}^2} & \text{sonst} \end{cases} \tag{3.54}$$

Teilenergie bzgl. Fläche der umschließenden Rechtecke

$$L_{F_{\mathrm{BB}}}(\mathcal{A},\mathcal{B}) = \frac{F_{\mathrm{BB}}(\mathcal{A}) - F_{\mathrm{BB}}(\mathcal{B})}{\sigma_{F_{\mathrm{BB}}}^2} \tag{3.55}$$

Teilenergie bzgl. Fläche der Silhouetten

$$L_{F_{\mathrm{Sil}}}(\mathcal{A},\mathcal{B}) = \frac{F_{\mathrm{Sil}}(\mathcal{A}) - F_{\mathrm{Sil}}(\mathcal{B})}{\sigma_{F_{\mathrm{Sil}}}^2} \tag{3.56}$$

Teilenergie bzgl. Histogrammverteilungen

$$L_H(\mathcal{A},\mathcal{B}) = \frac{D_B(\vec{H}(\mathcal{A}),\vec{H}(\mathcal{B}))}{\sigma_H^2} \tag{3.57}$$

Um die unterschiedlich ausgeprägten Streuungen der einzelnen Messgrößen in einem ausgewogenen Verhältnis zueinander in eine Gesamtenergiebilanz einfließen lassen zu können, wird jede Messgröße durch die jeweilige aus einer Trainingssequenz gewonnene Varianz σ^2 normiert. Mittels dieser Ähnlichkeitsmaße wird die Teilenergie

$$L(\mathcal{A}_i,\mathcal{B}_i) = N_{\mathrm{Objekte}}\left(L_{\vec{t}}(\mathcal{A}_i,\mathcal{B}_i) + L_{F_{\mathrm{BB}}}(\mathcal{A}_i,\mathcal{B}_i) + L_{F_{\mathrm{Sil}}}(\mathcal{A}_i,\mathcal{B}_i) + L_H(\mathcal{A}_i,\mathcal{B}_i)\right) \tag{3.58}$$

für jedes Ereignis berechnet, wobei zur Berücksichtigung der Anzahl N_{Objekte} der am Ereignis beteiligten Objekte die Summe der Ähnlichkeitsmaße abschließend mit dieser Zahl multipliziert wird.
Auf Basis der so getroffenen Definition der Energie eines Graphen läßt sich mittels des erläuterten SA-Verfahrens ein im Sinne der Energiebilanz optimaler Graph iterativ bestimmen (vgl. Algorithmus 3). Hierzu wird – zu Beginn ausgehend von einem initialen Graphen G, dem willkürlich eine Energie von $L = 0$

[21] Für den Fall, dass eine der Mengen $\mathcal{A}$ oder $\mathcal{B}$ mehrere Elemente umfassen sollte, so werden zur Berechnung der Größen stets die Eigenschaften der auf Pixelebene zu bildenden Schnittmenge der Objekte verwendet.

zugewiesen wird – ein Nachbarschaftsgraph G' erzeugt, indem sowohl aus der Menge der Startknoten als auch aus der Menge der Zielknoten jeweils ein Knoten gewählt wird. Verbindet diese beiden Knoten bereits eine Kante, so wird diese gelöscht, im anderen Fall wird eine solche hinzugefügt (vgl. Abbildung 3.14).

Algorithmus 3 Graphenbasierte Lösung des Trackingproblems mittels SA

Benötigt: Menge der Startknoten $\mathcal{A}$, Menge der Endknoten $\mathcal{B}$, Schwelle Θ

procedure
  Generiere zufällig einen Graphen G aus Knoten $\mathcal{A}$ und $\mathcal{B}$
  Bestimme alle zulässigen Nachbarlösungen $\mathcal{G}$ zu G
  Setze $L = L_{opt} = 0$ und $G_{opt} = G$
  while ($\vartheta > \Theta$ oder $\mathcal{G} \neq \{\}$) **do**
    Wähle einen Graphen $G' \in \mathcal{G}$
    Bestimme Energiedifferenz ΔL zwischen G und G'
    if ($\Delta L \leq 0$ oder Zufallszahl ($r \in [0, \ldots, 1]$) $< \exp(-\frac{\Delta L}{\vartheta})$) **then**
      $G = G'$
      Bestimme alle zulässigen Nachbarlösungen $\mathcal{G}$ zu G
      Setze $L \leftarrow L + \Delta L$
      if ($L < L_{opt}$) **then**
        Setze $L_{opt} = L$ und $G_{opt} = G'$
      end if
      Setze $\vartheta \leftarrow \frac{\vartheta}{2}$
    else
      Setze $\mathcal{G} \leftarrow \mathcal{G} \backslash \{G'\}$
    end if
  end while
end procedure

Diese Modifikation des Graphen impliziert für die betreffenden Knoten neue Basisereignisse, die wiederum selbst eine Änderung der Energie des Graphen bewirken. Da im Zuge des Optimierungsverfahrens nur die Energiedifferenz ΔL zwischen dem alten Graph G und dem neuen Graph G' von Belang ist, genügt es, nur die Teilenergien $L_i(G)$ bzw. $L_i(G')$ für diejenigen N_{Event} Basisereignisse zu berechnen, die sich durch das Wegnehmen bzw. Hinzufügen der Kante geändert

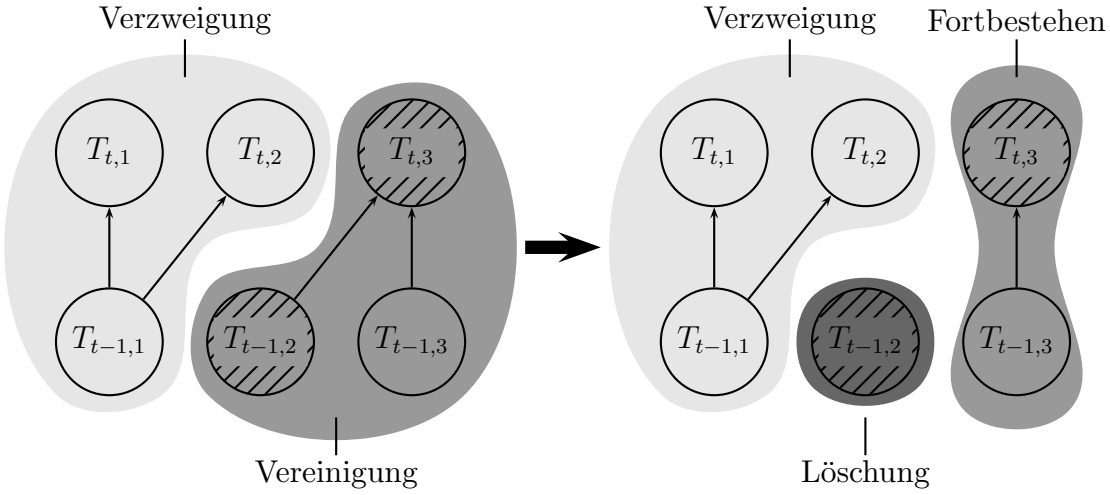


Abbildung 3.14 – Beispielhafte Darstellung von Graph (links) und Nachbarschaftsgraph (rechts): Die im Ausgangsgraph zwischen den zufällig gewählten Knoten (schraffiert) vorhandene Kante wird gelöscht. Damit geht unmittelbar einher, dass aus den zwei Basisereignissen (Verzweigung, Vereinigung) im neuen Graph drei Basisereignisse (Verzweigung, Löschung, Fortbestehen) werden.

haben. Gilt für die daraus resultierende Differenz

$$\Delta L = \sum_{i=1}^{N_{\text{Event}}} L_i(G') - L_i(G) < 0, \tag{3.59}$$

so wird der neue Graph bedingungslos als neue Ausgangslösung für die nächste Iteration übernommen. Andernfalls wird auf Basis des Kontrollparameters ϑ in Anlehnung[22] an Gleichung 3.52 eine Akzeptanzwahrscheinlichkeit $p = \exp(-\frac{\Delta E}{\vartheta})$ berechnet, mit der dieser Graph auch bei einer Verschlechterung der Energiebilanz dennoch als Ausgangslösung für den nachfolgenden Iterationsschritt verwendet wird. Wann immer sich die Ausgangslösung ändert, wird gleichzeitig der Kontrollparameter ϑ halbiert, so dass der iterative Prozess abgebrochen wird, sobald wahlweise ϑ einen vorgegebenen Schwellwert Θ unterschreitet oder aber sämtliche Nachbarlösungen zu einem Graphen evaluiert worden sind.
Auf Basis dieses graphentheoretischen Konzeptes wird durch die in Abbildung 3.12 dargestellte Trackingarchitektur eine Personenverfolgung von Zeitschritt $t-1$ nach t mittels folgendem Vorgehen geleistet:

[22] Im Kontext der Optimierung kann hierbei auf die Boltzmann-Konstante aus Gleichung 3.52 verzichtet werden.

In jedem Bild $\underset{\sim}{I}_t$ werden im rg-Chroma mit Hilfe einer Gaußmodellierung – wie in Abschnitt 2.1.1 beschrieben – hautfarbene Bereiche detektiert und somit eine Menge von Objekten $\mathcal{T}_t'$ ermittelt, welche als potentielle Köpfe in Betracht kommen. Desweiteren stehen aus dem vorhergehenden Bild $\underset{\sim}{I}_{t-1}$ die Menge $\mathcal{T}_{t-1}$ aller zu diesem Zeitschritt getrackten Objekte zur Verfügung. Um eine sinnvolle[23] Reinitialisierung der Partikelfilter (vgl. Abschnitt 3.1) auch im Falle des Mehrpersonentrackings zu ermöglichen, werden in einer ersten Stufe zunächst die Objekte $\mathcal{T}_t'$ den im letzten Zeitschritt ermittelten Tracks zugeordnet. Bedingt durch eine gelegentlich nicht optimal mögliche Detektion hautfarbener Pixel kann die Zahl der Objekte $\mathcal{T}_t'$ auch in kleinen Szenarien mit nur wenigen Personen dennoch schnell zweistellige Werte annehmen, weswegen diese Zuordnung auf Basis der im vorigen Abschnitt eingeführten Heuristik mittels SA erfolgt. Da die Objekte $\mathcal{T}_t'$ aus einem anderen Zeitschritt stammen als die Tracks $\mathcal{T}_{t-1}$, werden für eine möglichst stimmige Zuordnung die Tracks zuerst prädiziert, woraus die Objekte $\tilde{\mathcal{T}}_t$ resultieren. Bei der anschließenden Anwendung des im vorigen Abschnitt geschilderten iterativen Optimierungsverfahrens zur Zuordnung von Hautfarbenbereichen auf die Tracks muss – um wie erwähnt die Reinitialisierung sinnvoll gestalten zu können – eine surjektive Abbildung gewährleistet sein. Deswegen ist aus der Menge der Basisereignisse die Verzweigung a-priori zu eliminieren und es sind daher diejenigen im Zuge des Verfahrens generierten Nachbarschaftsgraphen als unzulässig zu betrachten, in welchen ein solches Verzweigungsereignis zu finden ist. Auch die vorherig beschriebene Art der Energieberechnung ist insoweit zu modifizieren, als dass eine Ähnlichkeit der Silhouetten zwischen Tracks, die Gesichter oder ganze Köpfe repräsentieren, und der Hautfarbenbereiche, die abhängig von der situationsbedingten Güte mitunter nur Teile des Gesichts widergeben, nicht sehr aussagekräftig ist und daher nicht in die Gesamtenergiebetrachtung einbezogen wird. Über die letztlich erhaltene Zuordnung stehen so für jeden Track – falls vorhanden – mögliche Objekte zur Reinitialisierung zur Verfügung, die unmittelbar in den zum Track gehörigen Partikelfilterprozess einfließen. Sämtliche Hautfarbenbereiche, die keinem Track zugeordnet werden konnten, stellen potentielle Kandidaten für neue Objekte in der Szene dar, weswegen für jeden dieser Bereiche ein neuer Partikelfilter und damit ein neues, potentielles Objekt initialisiert wird.
Anschließend ermittelt jeder Partikelfilter in der bekannten Weise zur Einzelpersonenverfolgung durch eine datengetriebene Adaption der Hypothesen eine vermeintliche Objektposition und deren zugehörige Güte Ω_{Modell} (vgl. Gleichung

[23] Sinnvoll bedeutet hier, dass jeder objektspezifische Partikelfilter nur auf diejenigen hautfarbenen Bereichen im Bild reinitialisiert werden soll, die vermeintlich auch zu dem betreffenden Objekt gehören.

3.44). Um vor allem temporäre Effekte, die mit einem spontanen Absinken der Güte der Partikel für einzelne Objekte verbunden sind, zu mildern, erfolgt im Sinne einer robusten Gestaltung der Personenverfolgung die Beurteilung der Objekte auf einer über die letzten N Zeitschritte gemittelten Güte $\overline{\Omega}_{\text{Modell}}$. All diejenigen Objekte, deren mittlere Güte $\overline{\Omega}_{\text{Modell}}$ eine Schwelle Θ überschreitet, werden mittels Simulated Annealing den prädizierten Tracks $\tilde{\mathcal{T}}_t$ zugeordnet. Erst durch diesen Schritt gelingt es, Situationen, in denen Personen von anderen verdeckt werden, zu kontrollieren und in die Einzelobjekte aufzulösen, sowie nach Beendigung der Verdeckung erfolgreich die identitätstreue Verfolgung der Objekte fortzuführen.

Kapitel 4

Tracking-Evaluierung

Trotz der vielfältigen Aktivitäten auf dem Gebiet der Objekt- und Personenverfolgung wurde erst in jüngerer Zeit damit begonnen, einheitliche Kriterien zu definieren, anhand derer die Ergebnisse verschiedener Trackingalgorithmen verglichen werden können. In den folgenden Absätzen soll zunächst die generelle Notwendigkeit einer systematischen Evaluierungsstrategie motiviert und auf Basis eines ausgewählten Bewertungsverfahrens die in den Kapiteln 3.2 sowie 3.3 beschriebenen Methoden zur Verfolgung von Personen bzw. insbesondere deren Köpfen gegeneinander verglichen werden. Ein grundlegender Anspruch an das gewählte Bewertungsverfahren muss dabei sein, dass damit ein Trackingergebnis nachvollziehbar durch objektive Messgrößen ausgedrückt wird, die der subjektiv wahrgenommenen Empfindung gerecht werden.

4.1 Historie der Tracking-Evaluierung

In der zweiten Hälfte der 90er Jahre wurden vereinzelt erste Ideen (z. B. von Pingali u. Segen [74]) publiziert, welche sich mit der Evaluierung von Trackingsystemen auseinandersetzten. Diese mündeten in den in Verbindung mit der Konferenz „Computer Vision and Pattern Recognition (CVPR)“ veranstalteten Arbeitskreis „Empirical Evaluation Methods in Computer Vision (EEMCV)“. Während jedoch diese Veranstaltung noch relativ breit ausgerichtet war, konnte eine erste internationale und spezialisiertere Plattform mit Themenschwerpunkt Performanzevaluierung von Trackingalgorithmen durch den im Jahr 2000 erstmalig veranstalteten Arbeitskreis „Performance Evaluation of Tracking and Surveillance (PETS)“ etabliert werden. Dabei wurde anfänglich das Hauptaugenmerk auf die Schaffung einer gemeinsamen Datenbasis gelegt, welche eine Vergleichbarkeit der Ergebnisse von verschiedenen Algorithmen ermöglichen sollte. Erst in den folgenden Veranstaltungen kristallisierte sich die zusätzliche Notwendigkeit von definierten Metriken heraus, um unterschiedliche Algorithmen nicht

mehr nur anhand sehr subjektiv geprägter, visueller Eindrücke von Trackingergebnissen gegenüberstellen zu können, sondern eine objektive Beurteilung diverser Algorithmen anhand prägnanter Zahlen zuzulassen. Dabei sollen geeignete Metriken nach Smith u. a. [100] einerseits möglichst allgemeingültig gehalten werden, d. h. die Option bieten, unterschiedlichste Trackingtechniken (basierend auf visuellen und/oder akustischen Merkmalen, 2D oder 3D Objektdarstellungen usw.) evaluieren zu können, andererseits die Zahl der Freiheitsgrade (Parameter, Schwellwerte) begrenzt halten, um eine praxistaugliche Anwendbarkeit der Metriken zu gewährleisten. Gleichzeitig aber müssen derartige Metriken auch die menschliche Wahrnehmung der Trackingergebnisse in ausreichendem Maße widerspiegeln und dabei stets leicht interpretierbar bleiben, was unter anderem nur einen limitierten Satz von Metriken zuläßt (vgl. Manohar u. a. [66]). Wie jedoch auch erst kürzlich publizierte Ansätze (vgl. beispielsweise die Arbeiten von Bashir u. Porikli [9], Bernardin u. a. [13], Black u. a. [18], Ellis [32], Lazarevic-McManus u. a. [61], Manohar u. a. [66], Schlogl u. a. [88], Zhu u. a. [126]) immer noch offenbaren, gelingt eine Erfüllung aller genannten Anforderungen nur mit teils mäßigem Erfolg.
Gerade jedoch unter dem Aspekt, dass durch die Einführung von Metriken neben einem objektiven Algorithmenvergleich darüber hinaus eine gezielte Analyse der Störanfälligkeit bzgl. bestimmter Situationen im Verlauf des Trackingprozesses erlaubt und dadurch die Weiterentwicklung bestehender Technologien erleichtert wird, zeigt sich die generelle Notwendigkeit eines Evaluierungsschemas, so dass sich bis zum heutigen Tag immer wieder neue Publikationen und sogar neu ins Leben gerufene Arbeitskreise[1] diesem Thema widmen.

4.2 Datenbank

Zur Durchführung einer Evaluierung der verschiedenen, in den Kapiteln 3.2 und 3.3 vorgestellten Methodiken für die Personenverfolgung wird eine Datenbank verwendet, welche im Rahmen des europäischen Projektes AMI (Augmented Multi-party Interaction) am schweizer Forschungsinstitut IDIAP (Institute Dalle Molle d'Intelligence Artificielle Perceptive) aufgezeichnet wurde[2]. Für diesen Datenkorpus wurde ein typischer Konferenzraum, wie er in Abbildung 4.1 dargestellt ist, nachgebildet. Im Gegensatz zu konventionellen Besprechungszimmern

[1] Beispiel für einen solchen sehr jungen Arbeitskreis, der 2006 erstmals veranstaltet wurde, ist der „CLEAR Evaluation Workshop“ (vgl. Stiefelhagen u. Garofolo [104]).

[2] Der hier betrachtete Datenkorpus trägt die Bezeichnung „AV16.7.ami“ und wurde eigens für die Arbeitsgruppe „Objektlokalisation und -verfolgung“ erstellt.

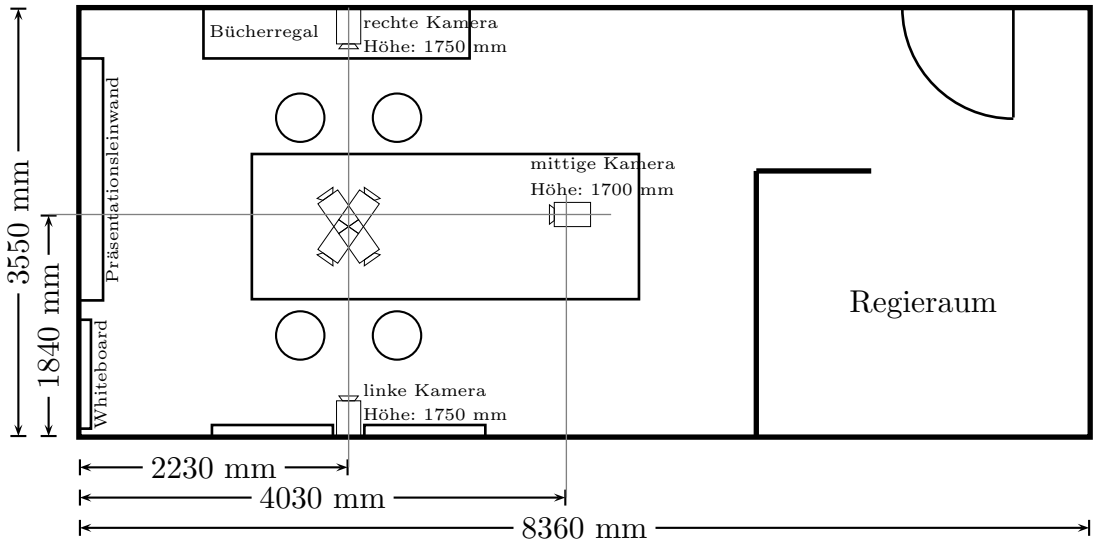


Abbildung 4.1 – Maßstabsgetreue Darstellung betreffend Aufbau und Abmessungen des zur Videoaufnahme genutzten und mit Kameras ausstaffierten Konferenzraumes.

rechte Kamera (R) *mittige Kamera (M)* *linke Kamera (L)*

Abbildung 4.2 – Beispiele für die durch die drei Hauptkamerasysteme im Konferenzraum abgedeckten Perspektiven.

wurde dieser Raum jedoch mit zusätzlichem technischen Equipment ausgestattet[3]:
An jeder der beiden Wände hinter den Teilnehmern befindet sich in 1,75 m Höhe eine Kamera, welche die jeweils gegenüberliegende Seite des Raumes filmt. Am unteren Ende des Konferenztisches wurde in 1,70 m Höhe eine weitere Kamera mit Blickrichtung auf die Präsentationsleinwand und das Whiteboard positioniert. Abbildung 4.2 zeigt für die durch die drei Kameras abgedeckten Perspektiven jeweils ein Beispiel. Weitere vier Kameras wurden in der Mitte des Tisches zwischen den Personen platziert, um Nahaufnahmen der einzelnen Sitzungsteil-

[3]In der englischsprachigen Literatur wird ein solcher Raum üblicherweise als *smart room* bezeichnet.

nehmer zu erhalten. Diese Kameras waren insbesondere für die Emotionserkennung von Bedeutung, spielten jedoch für das hier behandelte OT keine weitere Rolle.
Bei Analyse typischer Besprechungsabläufe stellt man fest, dass sich die teilnehmenden Personen die meiste Zeit der Sitzung auf ihren anfangs eingenommenen Plätzen am Tisch befinden. Besonders interessant aus Sicht der Personenverfolgung sind aber vor allem jene kritischen Momente, in denen Personen das Sichtfeld der Kamera betreten oder selbiges verlassen, durch andere Personen bzw. Gegenstände teilweise oder komplett verdeckt werden, sowie das Auftreten spontaner Bewegungen vor stark strukturiertem Hintergrund. Ziel bei der Akquise der 16 Videosequenzen des Datenkorpus war es deshalb, den Fokus auf eben genau diese Phänomene realer Besprechungen zu legen, weswegen die Teilnehmer angewiesen wurden, durch ihre Handlungen speziell derartige, für das OT kritische Situationen zu provozieren. Entstanden ist daraus eine Datenbank

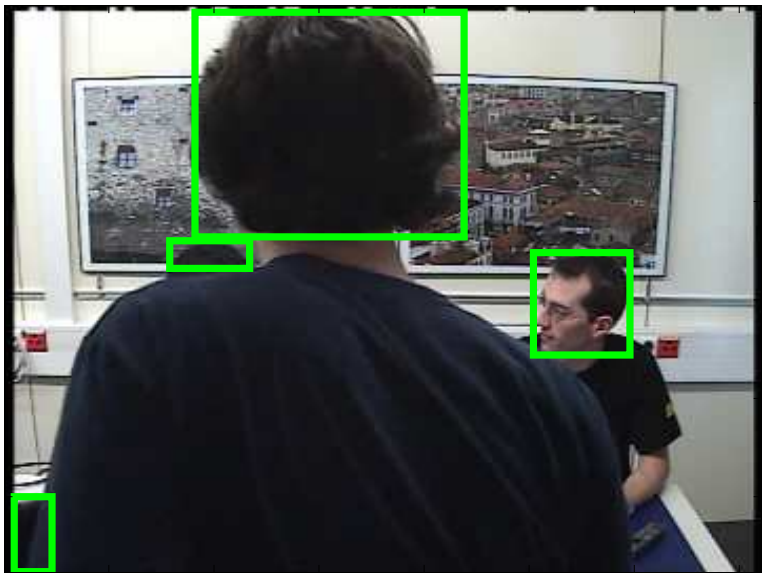


Abbildung 4.3 – Beispielhafte Annotation gemäß den Vorgaben: Selbst Köpfe, die aufgrund einer nur teilweise gegebenen Sichtbarkeit oder mangels Kontrast erst auf den zweiten Blick als solche auszumachen sind, wurden konsistent – repräsentiert durch die umschreibenden Rechtecke – als Referenzobjekte markiert.

von Beispielsitzungen mit variierender Teilnehmerzahl und einer Gesamtdauer von ca. 30 Minuten. Für sämtliche Sequenzen steht die für eine Beurteilung von Trackingresultaten essentiell notwendige Information[4] über die Position und

[4]In der einschlägigen Fachliteratur wird diese Information als *ground truth* bezeichnet.

Sequenz	Kameraperspektive	Dauer in s	Zahl der Teilnehmer/ frontal sichtbar/ Hinterkopfansicht	Verdeckungen von Köpfen?	Kameraperspektive verdeckt?	Person setzt sich?
01	L	63	1/1/1	nein	ja	nein
	R	63	1/1/1	nein	ja	nein
02	L	48	1/1/1	nein	ja	nein
	R	48	1/1/1	nein	ja	nein
03	L	208	1/1/1	nein	nein	ja
	R	208	1/1/1	nein	nein	ja
08	L	99	2/2/0	ja	ja	ja
	R	99	2/0/2	nein	ja	ja
09	L	69	2/2/0	ja	nein	nein
	R	69	2/0/2	ja	ja	nein
12	L	101	3/3/0	ja	nein	ja
	R	101	3/0/3	ja	ja	ja
13	L	94	3/3/0	ja	nein	ja
	R	94	3/0/3	ja	ja	ja
14	L	117	4/2/2	ja	ja	ja
	R	117	4/2/2	ja	ja	ja
16	L	88	4/4/4	ja	ja	nein
	R	88	4/2/4	nein	ja	nein

Tabelle 4.1 – Auflistung der zur Evaluierung herangezogenen Videosequenzen sowie der jeweils zu beobachtenden Phänomene, die im Zuge der Objektverfolgung von besonderem Interesse sind: Angegeben ist, unter welcher Ansicht Personen im Bild zu sehen sind (frontal oder nur der Hinterkopf wie z. B. in Video 08R), inwiefern es zu einer gegenseitigen Verdeckung von Köpfen kommt bzw. ob ein Kopf nicht mehr vollständig im Bild erfasst ist, weil sich die betreffende Person zu nahe vor einer Kamera befindet, sowie die Vorkommnisse, dass sich Personen setzen.

Größe von Köpfen in Form von umschreibenden Rechtecken zur Verfügung, wofür im Abstand von je 500 ms alle im Bild befindlichen Köpfe, auch wenn diese aufgrund einer Verdeckung eventuell nur zu einem Bruchteil sichtbar sein sollten, in einem manuellen Annotationsprozess präzise erfasst wurden. Durch dieses sehr strikt definierte Schema konnte gewährleistet werden, dass die Annotation sehr objektiv erfolgte und dadurch die Referenzdaten als äußerst verlässlich und vollständig erachtet werden können. Zeitgleich führt dies zu einer Einbeziehung von subjektiv als nahezu nicht mehr detektierbar einzustufenden Teilobjekten, wie sie beispielsweise in Abbildung 4.3 dargestellt sind, was jedoch in den angestellten Untersuchungen die Möglichkeit bietet, auch die Grenzbereiche der Leistungsfähigkeit betrachteter Algorithmen auszuloten.
Zum Zwecke einer ordnungsgemäßen wissenschaftlichen Evaluierung wird dieses Datenset aufgeteilt in einen Trainingsdatensatz, welcher zum Erlernen bzw. Anpassen evtl. notwendiger Parametrierungen in den Trackingalgorithmen herangezogen werden kann, und einen davon disjunkten Datensatz, der ausschließlich zur Evaluierung eingesetzt wird. In Tabelle 4.1 findet sich eine Zusammenstellung der zur Evaluierung verwendeten Sequenzen, sowie der darin zu beobachtenden Phänomene.

4.3 Evaluierungsschema

Um eine Vergleichbarkeit zwischen diversen Trackingstrategien zu erreichen, wird in dieser Arbeit ein im Jahr 2005 von Smith u. a. [100] veröffentlichtes Evaluierungsschema (vgl. hierzu auch Schreiber u. Gatica-Perez [89]) zugrunde gelegt, welches insbesondere durch einen Satz von intuitiven Fehlermaßen eine dem subjektiven Empfinden sehr gut entsprechende Bewertung von Trackingergebnissen liefert und dennoch aufgrund der umfassenden Messgrößen eine detaillierte Analyse der Ergebnisse ermöglicht. Ausgangspunkt für die Evaluierung bildet die Menge der $N_{t,\mathcal{O}}$ Referenzobjekte $\mathcal{O}_t = \{\vec{O}_{t,1}, \ldots, \vec{O}_{t,N_{t,\mathcal{O}}}\}$ mit $\vec{O}_{t,i} = (t_x, t_y, s, \xi)^T$, welche zu jedem Zeitschritt t für jedes Bild $\underset{\sim}{I}_t$ die Position (Schwerpunkt t_x, t_y), Größe (Skalierung s) und Identität ξ der zu verfolgenden Objekte durch ein umschreibendes Rechteck repräsentieren.

4.3.1 Das Zuordnungsproblem

Bevor die Analyse der Trackingergebnisse mittels spezieller Metriken erfolgen kann, muss zuerst das Zuordnungsproblem zwischen den Referenzobjekten $\mathcal{O}_t$ und den vom Algorithmus ausgegebenen $N_{t,\mathcal{T}}$ Objekten $\mathcal{T}_t = \{\vec{T}_{t,1}, \ldots, \vec{T}_{t,N_{t,\mathcal{T}}}\}$ –

im Folgenden auch als Tracks $\vec{T}_{t,i} = (t_x, t_y, s, \xi)^T$ bezeichnet – gelöst werden[5]. Während in einigen Publikationen (vgl. Bashir u. Porikli [9], Black u. a. [18]) hierfür im Wesentlichen nur die Position und eventuell die Bewegungsrichtung für die beiderseitige Zuordnung zwischen Tracks und Referenzobjekten herangezogen werden oder allenfalls die räumliche und zeitliche Überlappung zwischen Track und Referenz im Verhältnis zur Fläche der Referenz berücksichtigt wird (vgl. Senior u. a. [97]), stützt sich die Evaluierung in dieser Arbeit auf einen umfassenderen Ansatz. Hierfür werden die beiden Maße Genauigkeit P_{ij} bzw. die Vollständigkeit R_{ij} betrachtet, welche die überlappende Fläche zweier Objekte ($|\vec{O}_{t,i} \cap \vec{T}_{t,j}|$) in das Verhältnis zu der alleinigen Fläche des Tracks ($|\vec{T}_{t,j}|$) bzw. des Referenzobjektes ($|\vec{O}_{t,i}|$) setzen:

$$P_{t,ij} = \frac{|\vec{O}_{t,i} \cap \vec{T}_{t,j}|}{|\vec{T}_{t,j}|} \tag{4.1}$$

$$R_{t,ij} = \frac{|\vec{O}_{t,i} \cap \vec{T}_{t,j}|}{|\vec{O}_{t,i}|} \tag{4.2}$$

Idealerweise sind für eine gegenseitige Zuordnung von Objekt $\vec{T}_{t,j}$ und Referenzobjekt $\vec{O}_{t,i}$ sowohl eine hohe Genauigkeit $P_{t,ij}$, als auch eine hohe Vollständigkeit $R_{t,ij}$ zu fordern. Um eine binäre Entscheidung bzgl. der Korrespondenz von Track und Referenzobjekt unter Beachtung des gerade genannten Aspektes herbeiführen zu können, bedient man sich der F-Bewertung[6], eines gewichteten harmonischen Mittels zwischen Genauigkeit und Vollständigkeit:

$$\mathrm{F}_{t,ij} = \frac{2R_{t,ij}P_{t,ij}}{R_{t,ij} + P_{t,ij}} \tag{4.3}$$

Wird dieses Maß für alle Kombinationen $(i, j), i \in \{1, \ldots, N_{t,\mathcal{O}}\}, j \in \{1, \ldots, N_{t,\mathcal{T}}\}$ berechnet, so erfolgt schließlich eine gegenseitige Zuordnung der betreffenden Objekte genau für diejenigen Fälle, in denen die F-Bewertung einen vorgegebenen Schwellwert überschreitet. In Anlehnung an die Arbeit von Lienhart u. a. [63], der einen Track genau dann einem Referenzobjekt zuweist, wenn der Abstand der Schwerpunkte weniger als 30 % der Breite des Referenzobjektes beträgt und die Ausdehnungsmaße nicht um mehr als ±50 % differieren, wurde dieser Schwellwert

[5] Kernproblem hierbei ist die Definition eines Kriteriums, anhand dessen entschieden werden kann, welche Tracks jeweils einem Referenzobjekt und ebenso umgekehrt, welche Referenzobjekte jeweils einem Track zuzuweisen sind.

[6] In der englischen Literatur ist diese Bewertung als sogenanntes *F-measure* bekannt (vgl. Van Rijsbergen [108]).

auf ein Äquivalent von $0,33$ gesetzt. Basierend auf der getroffenen Zuordnung wird anschließend mittels geeigneter Fehlermaße eine detaillierte Analyse von Trackingergebnissen vorgenommen.

4.3.2 Beurteilung von Trackingfehlern bezüglich der Personenkonfiguration

Im Hinblick auf die geschilderte Art der Zuordnung ist es offensichtlich, dass ein einwandfreies Trackingergebnis genau dann vorliegt, wenn jedem Referenzobjekt $\vec{O}_{t,i}$ eineindeutig ein Trackingobjekt $\vec{T}_{t,j}$ zugeordnet wurde. Um Zuordnungsfehler beschreiben zu können, werden Metriken für jede der möglichen Fehlerklassen, welche bei der Zuordnung auftreten können, eingeführt. Diese sind:

a) FN – Ein Referenzobjekt konnte keinem der Trackingobjekte zugeordnet werden.

b) FP – Ein vom Tracker erzeugtes Objekt konnte keinem der Referenzobjekte zugeordnet werden.

c) MO – Ein Trackingobjekt wurde mehreren Referenzobjekten zugeordnet. Hierbei wird für jedes zusätzliche Referenzobjekt jeweils ein weiterer MO-Fehler gewertet.

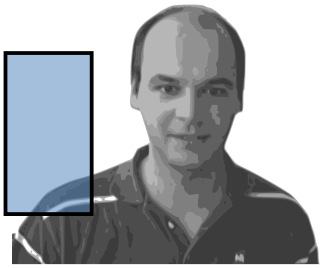
d) MT – Ein Referenzobjekt wurde mehreren Trackingobjekten zugeordnet. Hierbei wird für jeden zusätzlichen Track jeweils ein weiterer MT-Fehler gewertet.

Jeder der obigen Fehlertypen ist in Abbildung 4.4 exemplarisch dargestellt. Insbesondere bei den Größen MO sowie MT ist hierbei zu bemerken, dass der intuitive Eindruck eines menschlichen Betrachters ein Trackingergebnis umso schlechter bewertet, je mehr zusätzliche Objekte einem einzigen Referenz- respektive Trackingobjekt zugeordnet werden. Zur Berücksichtigung dieses Umstandes fließt daher die Zahl überschüssiger Objekte unmittelbar in diese Art von Fehlergröße ein. Die eben genannte physiologische Perzeption ist jedoch nicht nur limitiert auf Objektebene, sondern erstreckt sich ebenso auf die gesamte Szene, so dass ein steigender Anteil an fehlenden oder überschüssigen Tracks in einem Bild mit einem proportional zunehmenden Grad als falsch empfunden wird. Um der menschlichen Wahrnehmung darüber hinaus also weiter Rechnung zu tragen, wird deshalb mit der Konfigurationskompaktheit

$$\mathrm{CD} = N_{t,\mathcal{T}} - N_{t,\mathcal{O}} \tag{4.4}$$

ein weiteres Maß eingeführt, welches sich berechnet aus der Differenz zwischen der Zahl an Tracks ($N_{t,\mathcal{T}}$) und der Zahl an Referenzobjekten ($N_{t,\mathcal{O}}$).

(a) FN - Der zu sehende Kopf wurde vom Algorithmus nicht als solcher im aktuellen Bild erkannt.



(b) FP - Obwohl kein tatsächlich zu detektierendes Objekt an betreffender Stelle im Bild vorhanden ist, wird vom Algorithmus dennoch ein Track angezeigt.

(c) MT - Für ein Referenzobjekt liefert der Algorithmus mehr als einen Track.

(d) MO - Ein einziger Track umfasst mehr als nur ein Referenzobjekt.

Abbildung 4.4 – Exemplarische Visualisierung der im Kontext der Evaluierung einer Objektkonstellation erfassten Fehlertypen, die durch die Maße FN (a), FP (b), MT (c) und MO (d) beschrieben werden.

4.3.3 Beurteilung von Trackingfehlern bezüglich der Personenidentitäten

Neben der Analyse der Konfiguration stellt die über den Zeitverlauf konstante Zuordnung einer Identität zu jedem der ermittelten Tracks den zweiten wichtigen Aspekt einer Personenverfolgung dar. Aus diesem Grund befasst sich eine umfassende Evaluierung von OT-Algorithmen nicht nur mit den Konfigurationsfehlern sondern auch mit einer Auswertung der den einzelnen Tracks zugewiesenen Identitäten[7]. Gemäß dem Verständnis eines einwandfreien Trackingergebnisses sollte ein einziger Track über den gesamten Zeitverlauf genau einem Referenzobjekt zu-

[7] Augenmerk liegt hierbei nicht auf der Feststellung der wahren Identität des Referenzobjektes durch den Track, sondern in der stimmigen und konsistenten Vergabe eines Bezeichners von Seiten des Trackingalgorithmus. Diese Problemstellung wird in der Sprechererkennung auch als *Diarization* bezeichnet.

gewiesen sein und damit zu jedem Zeitpunkt über die Kenntnis der Identität des Tracks eineindeutig auf die Identität des Referenzobjektes geschlossen werden können. Falls, wie es in der Praxis insbesondere bei gegenseitigen Verdeckungen oder bei einem erneuten Eintreten einer – bereits aus einem früheren Teil der Videosequenz – bekannten Person in das Szenario mitunter passieren kann, die Identität eines Objektes durch Assoziation mit einem anderen Track geändert wird, so muss zunächst definiert werden, welcher Track die Identität eines Referenzobjektes über die gesamte Dauer der Videosequenz festlegt. Obwohl hierfür prinzipiell diverse Strategien – beispielsweise durch die zuerst oder letztmalig getroffene Zuordnung von Track und Referenzobjekt in der Sequenz – denkbar wären, so erscheint doch das Konzept (Assoziationsregel) am plausibelsten, die Identität desjenigen Tracks $\vec{T}_{t,j}$ als bestimmend für das Referenzobjekt $\vec{O}_{t,i}$ zu betrachten, welcher über die meiste Zeit der Videosequenz mit diesem assoziiert war. Im Folgenden wird ein solcher Track auch als identifizierender Track $\vec{T}_{\hat{j}_i}$, das entsprechende Referenzobjekt als identifiziertes Referenzobjekt $\vec{O}_{\hat{i}_j}$ bezeichnet. Hierfür sind im gewählten Evaluierungskonzept zwei weitere Metriken von zentraler Bedeutung, deren Definition zusätzlich graphisch in Abbildung 4.5 verdeutlicht ist:

a) FIT – Ein Referenzobjekt $\vec{O}_{t,i}$, welches nicht vom Track $\vec{T}_{t,j}$ identifiziert wird, wird dennoch zum aktuellen Zeitschritt diesem Track zugeordnet.

b) FIO – Ein Track $\vec{T}_{t,j}$, der nicht das Referenzobjekt $\vec{O}_{t,i}$ identifiziert, wird im aktuellen Zeitschritt dennoch diesem Referenzobjekt zugeordnet.

Ergänzt werden diese beiden Fehlermaße durch die Trackergüte $Q_{\mathcal{T}}$ und die Objektgüte $Q_{\mathcal{O}}$, um die zeitliche Konsistenz der Abbildung $\mathcal{T}_t \rightarrow \mathcal{O}_t$ sowie $\mathcal{O}_t \rightarrow \mathcal{T}_t$ über die gesamte Sequenz zu erfassen. Hierzu wird zunächst diejenige Zeitdauer $t_{j\hat{i}_j}$ bestimmt, für die ein Track $\vec{T}_j$ sein identifiziertes Referenzobjekt $\vec{O}_{\hat{i}_j}$ verfolgt, d. h. die dieser Track dem korrekten Referenzobjekt zugeordnet war, und ins Verhältnis zu der gesamten Lebensdauer t_{T_j} des Tracks gesetzt, wodurch man ein Maß für die Trackergüte erhält.

$$Q_{\mathcal{T},j} = \frac{t_{j\hat{i}_j}}{t_{T_j}} \tag{4.5}$$

Analog ist zu verfahren, um die Objektgüte zu ermitteln, wobei $t_{i\hat{j}_i} = t_{j\hat{i}_j}$ gilt.

$$Q_{\mathcal{O},i} = \frac{t_{i\hat{j}_i}}{t_{O_i}} \tag{4.6}$$

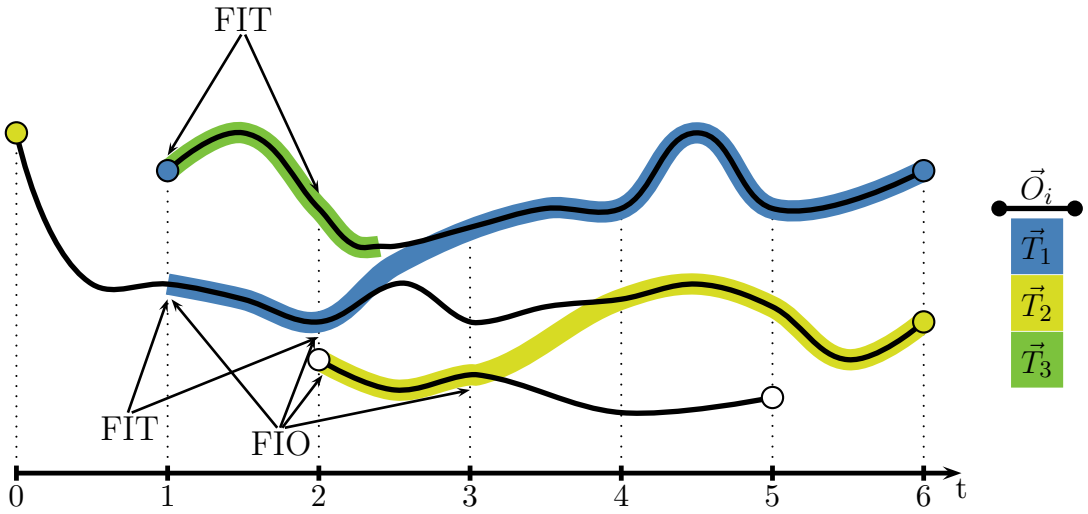


Abbildung 4.5 – Exemplarische Darstellung der im Zusammenhang mit der eindeutigen Zuordnung von Identitäten zwischen Tracks und Referenzobjekten auftretenden Fehler: Jedem der drei Referenzobjekte (schwarze Linien) wurde jeweils durch die Assoziationsregel ein Track (farbige Balken) zugeordnet, angedeutet durch die Färbung der Start- und Endknoten der Referenzobjekte. Wann immer die Identität, repräsentiert durch die Färbung, des Tracks nicht eineindeutig auf die des Referenzobjektes abgebildet wird, so indiziert dies zum aktuellen Zeitpunkt entweder einen Fehler FIT, wenn der betreffende Track nicht der identifizierende Track des Referenzobjektes ist, oder einen Fehler FIO, wenn der betreffende Track ein anderes Referenzobjekt identifiziert.

4.3.4 Prägnante Größen zur Bewertung von Trackingergebnissen auf Videosequenzen

Die Darstellung sämtlicher Fehlergrößen pro Zeitschritt ist einer qualitativen Beurteilung von Trackingergebnissen aufgrund der unüberschaubaren Fülle an Information nicht dienlich. Stattdessen werden auf Basis der eingeführten Metriken über die Sequenzlänge gemittelte Größen berechnet, die dann als prägnante Werte Aufschluß über die Leistungsfähigkeit eines Trackingansatzes geben können. Da über die Zeit die Zahl der in der Videosequenz sichtbaren Personen stark schwanken kann, genügt es hierbei nicht, unmittelbar die einzelnen Fehlergrößen über die Zeitschritte zu mitteln[8]. Vielmehr bedarf es zuerst einer Normalisie-

[8]So kann definitionsgemäß in einer Szene, in der keine Person erscheint, ein FN-Fehlertypus nicht auftreten. Dies muss entsprechend in den mittleren Fehlergrößen berücksichtigt werden.

Messgröße	Berechnungsvorschrift
F-Bewertung	$\overline{\mathrm{F}} = \frac{1}{T}\sum_{t=1}^{T}\frac{\mathrm{F}_t}{\max(N_{t,\mathcal{O}},1)}$
Mittlere Zahl an falsch-positiv Objekten	$\overline{\mathrm{FP}} = \frac{1}{T}\sum_{t=1}^{T}\frac{\mathrm{FP}_t}{\max(N_{t,\mathcal{O}},1)}$
Mittlere Zahl an falsch-negativ Objekten	$\overline{\mathrm{FN}} = \frac{1}{T}\sum_{t=1}^{T}\frac{\mathrm{FN}_t}{\max(N_{t,\mathcal{O}},1)}$
Mittlere Zahl an mehrfach assoziierten Referenzobjekten	$\overline{\mathrm{MO}} = \frac{1}{T}\sum_{t=1}^{T}\frac{\mathrm{MO}_t}{\max(N_{t,\mathcal{O}},1)}$
Mittlere Zahl an mehrfach assoziierten Tracks	$\overline{\mathrm{MT}} = \frac{1}{T}\sum_{t=1}^{T}\frac{\mathrm{MT}_t}{\max(N_{t,\mathcal{O}},1)}$
Gemittelte Konfigurationskompaktheit	$\overline{\mathrm{CD}} = \frac{1}{T}\sum_{t=1}^{T}\frac{\mathrm{CD}_t}{\max(N_{t,\mathcal{O}},1)}$
Mittlere Zahl an falsch identifizierten Referenzobjekten	$\overline{\mathrm{FIO}} = \frac{1}{T}\sum_{t=1}^{T}\frac{\mathrm{FIO}_t}{\max(N_{t,\mathcal{O}},1)}$
Mittlere Zahl an falsch identifizierenden Tracks	$\overline{\mathrm{FIT}} = \frac{1}{T}\sum_{t=1}^{T}\frac{\mathrm{FIT}_t}{\max(N_{t,\mathcal{O}},1)}$
Gemittelte Trackergüte (über alle N_{Tracks} unterschiedlichen, vom Algorithmus erzeugten Tracks)	$\overline{Q}_{\mathcal{T}} = \frac{1}{N_{\mathrm{Tracks}}}\sum_{j=1}^{N_{\mathrm{Tracks}}} Q_{\mathcal{T},j}$
Gemittelte Objektgüte (über alle N_{Refobj} Referenzobjekte)	$\overline{Q}_{\mathcal{O}} = \frac{1}{N_{\mathrm{Refobj}}}\sum_{i=1}^{N_{\mathrm{Refobj}}} Q_{\mathcal{O}_i}$

Tabelle 4.2 – Abschließende Gesamtübersicht der im Zuge der Evaluierung von Trackingergebnissen zugrunde gelegten Messgrößen. Um eine generelle qualitative Aussagekraft der Zahlen zu gewährleisten, werden die Größen entsprechend normiert.

rung der Fehlermaße zu jedem Zeitpunkt. Hierzu werden – wiederum durch die menschliche Physiologie motiviert – sämtliche Messgrößen[9] durch die zum Zeitpunkt t gegebene Zahl an Referenzobjekten $\max(N_{t,\mathcal{O}}, 1)$ dividiert[10]. Somit läßt sich abschließend die Berechnung von aussagekräftigen Werten für eine sinnvolle Bewertung der Leistungsfähigkeit verschiedener Trackingmethoden, wie in Tabelle 4.2 gelistet, zusammenfassen.

[9] Eine Ausnahme bilden lediglich die beiden Maße $Q_{\mathcal{T}}$ sowie $Q_{\mathcal{O}}$, da diese bereits objektspezifisch ausgewertet werden.

[10] Aus algebraischen Gründen wird durch das Maximum $\max(N_{t,\mathcal{O}}, 1)$ geteilt, um eine Division durch 0 zu vermeiden.

4.4 Evaluation Einzelpersonenverfolgung

Zur Beurteilung der Leistungsfähigkeit der in dieser Arbeit entwickelten Trackingarchitektur wurde diese zunächst im Zuge der Einzelpersonenverfolgung einem reinen bottom-up sowie einem top-down Referenzsystem gegenübergestellt.

4.4.1 Evaluierte Systeme zur Einzelpersonenverfolgung

Um den Einfluss der innerhalb der Architektur verwendeten Technik zur Personendetektion abgrenzen zu können, wurden alle der im Kapitel 3.2 vorgestellten, unterschiedlichen Methoden zur Personenmodellierung in das entwickelte Gesamtsystem integriert und gegeneinander evaluiert. Auf diese Weise resultierten insgesamt fünf unterschiedliche Systeme, die im Folgenden nochmals kurz zusammengefasst werden:

System A Als Referenzsystem für die Trackingaufgabe fungiert ein etablierter bottom-up Ansatz basierend auf dem Verfahren nach Viola u. Jones [110]. In Abbildung 4.6 ist das verwendete System als Blockschaltbild grob skizziert: Nach einer Vorverarbeitung, in der Bereiche mit Hautfarbe sowie

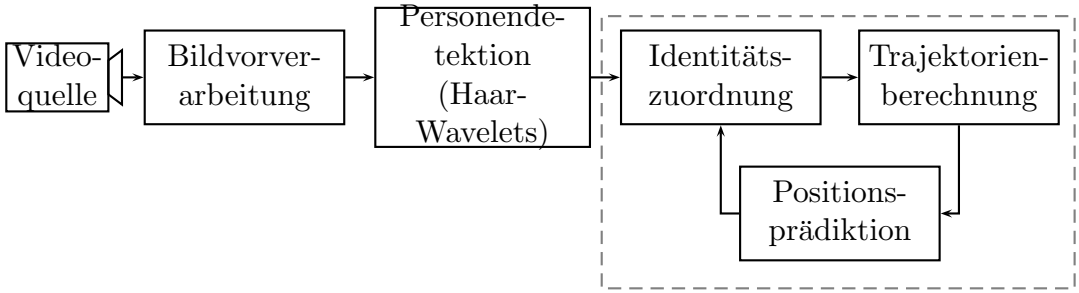


Abbildung 4.6 – Schematische Darstellung der verwendeten Architektur (System A) zur automatischen Personenverfolgung mittels des Detektionsverfahrens nach Viola u. Jones [110].

Vordergrundpixel detektiert werden, wird in dem verbleibenden Bild mittels der wavelet-basierten Klassifikationskaskade nach Gesichtern gesucht. Jeder Ausschnitt wird anschließend anhand eines Histogrammvergleiches entweder als eine bestimmte, aus dem vorhergehenden Bild bekannte Person identifiziert oder durch eine neue Identität als in die Szene eintretende Person markiert.

System B Ein reiner top-down Ansatz wurde basierend auf der Personendetektion mittels eines NN realisiert. Hierfür wurde die entwickelte Architektur insofern modifiziert (vgl. Abbildung 4.7), als dass jedes Partikel lediglich eine Bewertung durch die Personendetektion erfährt, jedoch dessen spezifische Parameter nicht aufgrund der Bilddaten verändert werden. Somit ist der Rückkanal zwischen Personendetektion und der Partikelverwaltung aufgebrochen, die Aufgabe der Personendetektion beschränkt sich daher ausschließlich auf die Bestimmung der Gewichte für die durch den Partikelfilter gegebenen Hypothesen.

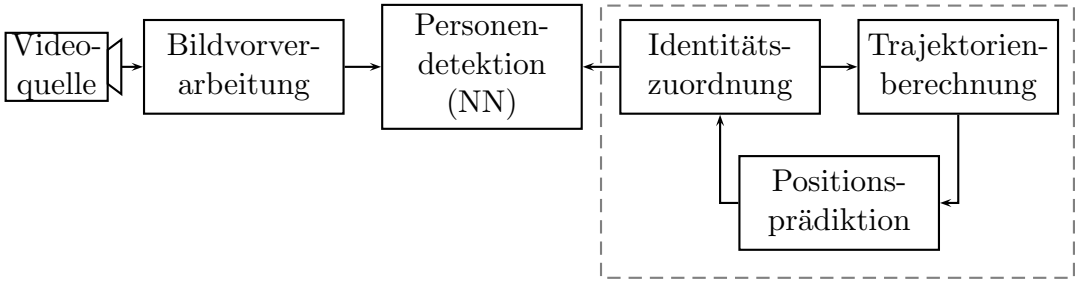


Abbildung 4.7 – Kompakte Darstellung der hypothesengetriebenen Architektur (System B) zur automatischen Personenverfolgung mittels des Detektionsverfahrens nach Rowley u. a. [86].

System C Die hybride Systemarchitektur, wie sie in Grafik 3.1 abgebildet ist, stellt die Grundlage für die in Kapitel 3.2.1 beschriebene Ellipsenmodellierung des menschlichen Kopfes dar. Bei diesem System gestaltet sich die Kommunikation zwischen Partikelfilter und Personendetektion bidirektional, so dass aufgrund der Bilddaten ein direkter Einfluss auf die Zustände der einzelnen Hypothesen möglich wird.

System D Wiederum als hybrides System wurde das formveränderliche Personendetektionsmodell basierend auf ASM umgesetzt. Hierzu wurde ein Kopf-Schulter Modell mit 20 Stützpunkten trainiert. Die datengetriebene Modelladaption basiert hierbei nicht wie von Cootes u. a. [27] vorgeschlagen auf einem Histogrammvergleich der Grauwerte entlang einer Geraden, sondern wie in Abschnitt 3.2.2 erläutert auf dem Gradientenbild, um vor allem Fehlern durch die bei den vorliegenden Besprechungsszenarien störenden Strukturen im Hintergrund entsprechend entgegenwirken zu können.

System E Der in System D gewählte Ansatz wurde abgeändert, so dass zwar auch hier ebenso Active Shape basierte Modelle zum Einsatz kommen, deren Anpassung allerdings nicht mehr auf dem Gradientenbild beruht, sondern mit Hilfe des in Abschnitt 3.2.2 beschriebenen Vergleiches von Gabor-Wavelets vorgenommen wird. Ziel ist es auch hierbei, durch die in den Wavelets kodierte Richtungsinformation von Kanten unempfindlicher gegenüber stark strukturierten Hintergrunddaten zu werden.

Bei allen Systemen kommt in der Bildvorverarbeitungsstufe sowohl ein adaptives Hintergrundmodell als auch eine Hautfarbendetektion zum Einsatz. Das Hintergrundmodell beruht dabei auf einem rekursiven zeitlichen Mittelwertmodell nach Gleichung 2.12, bei dem sämtliche Bereiche eines Bildes, in denen keine Tracks generiert wurden, zur Aktualisierung des Mittelwertes herangezogen werden. Die Detektion von Hautfarbe im Bild wird pixelbasiert mittels einer Schwellwertentscheidung vorgenommen. Hierzu wird, wie in Abschnitt 2.1.1 beschrieben, Hautfarbe im rg-Chroma anhand einer zweidimensionalen Gaußverteilung modelliert. Die auf einem Partikelfilter beruhenden Systeme B-E wurden in einem Modus betrieben, der die situative Anpassung der Hypothesenanzahl N_S abhängig von den modellbezogenen Messwerten erlaubt. Hierbei wurde der Wertebereich $15 \leq N_S \leq 30$ extern vorgegeben.

4.4.2 Diskussion der Evaluationsergebnisse

Alle fünf Systeme wurden in gleicher Weise auf all diejenigen Videosequenzen aus dem Validierungsset angewendet, in denen ausschließlich eine einzige Person zu sehen ist (Sequenzen 01L-03R). Anschließend wurden die von den jeweiligen Trackern erzielten Ergebnisse im Zuge einer Evaluierung basierend auf den im vorhergehenden Abschnitt 4.3 eingeführten Fehlermaßen gegenübergestellt.

Passgenauigkeit - die F-Bewertung

In Abbildung 4.8 ist die für jede Sequenz gemittelte F-Bewertung über der jeweiligen Videosequenz für jede Methode aufgetragen. Anhand dieser Darstellung ist unmittelbar ersichtlich, dass mit dem bottom-up Ansatz A aufgrund der Haar-waveletbasierten Personendetektion ein sehr präzises Trackingverfahren realisiert werden kann[11], welches den rein probabilistischen top-down Ansatz mit einem Neuronalen Netz als Messfunktion (Verfahren B) aufgrund der von

[11] Dieses Ergebnis erscheint insbesondere deswegen nicht weiter überraschend, da die Methodik nach Viola u. Jones [110] allgemein als sehr präzise im Sinne der ortsbezogenen Genauigkeit detektierter Bildausschnitte gilt.

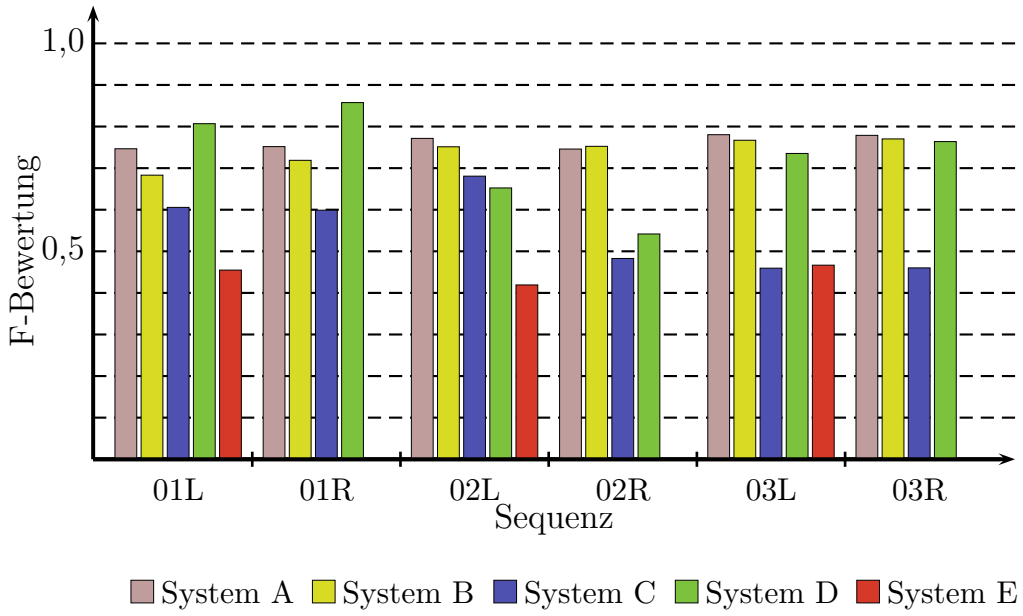


Abbildung 4.8 – Diagramm der mittleren F-Bewertung für sämtliche Systeme zur Einzelpersonenverfolgung, aufgetragen über der jeweils evaluierten Sequenz.

den Hypothesen fix vorgegebenen Position und der fehlenden Adaptionsmöglichkeit an die Bildinformation um durchschnittlich 3 % übertrifft. Darüber hinaus offenbart das Diagramm auch, dass ein hybrider, probabilistischer Systemansatz basierend auf ASM (vgl. Systeme D, E) die von den ausschließlich unidirektional kommunizierenden Trackingprinzipien vorgelegten Werte (mit Ausnahme der Sequenzen 02L und 02R, Erklärung siehe unten) in etwa zu erreichen vermag, wenn – wie für System D mittels der gradientenbasierten Methodik – die richtige Strategie der bilddatengetriebenen Modelladaption gewählt wird. Hier konnte die auf Gabor-Wavelets basierende Technik in System E aufgrund des stark strukturierten Hintergrundes sowie einer großen Variation der innerhalb des Kopfes befindlichen Textur (Profil-, Hinterkopf- und Frontalansichten) nur sehr unzureichende Ergebnisse bzgl. der Genauigkeit liefern und ergab im Falle der rechtsseitigen Kameraperspektiven sogar überhaupt keine ausreichende Übereinstimmung mit den annotierten Referenzdaten. Im Kontext des hybriden Partikelfilters führt eine Modellanpassung des ASM beruhend auf den Bildgradienten meist zu den besten F-Bewertungen. Lediglich für die Sequenzen 02L sowie 02R weist dieses Maß schlechtere Werte auf, was jedoch damit zu begründen ist, dass der Kopf der Person oftmals von hinten und aufgrund der kameranahen Position meist nur unvollständig im Bild sichtbar ist und hierbei in die Berechnung der F-Bewertung Tracks wie in Abbildung 4.9 einbezogen wurden, die von den

anderen Methoden technologisch bedingt erst gar nicht erfasst werden können und daher dann bei diesen zu erhöhten Werten der durchschnittlichen FN-Maße führen. Unter den hybriden Techniken zeigt sich ebenso im direkten Vergleich

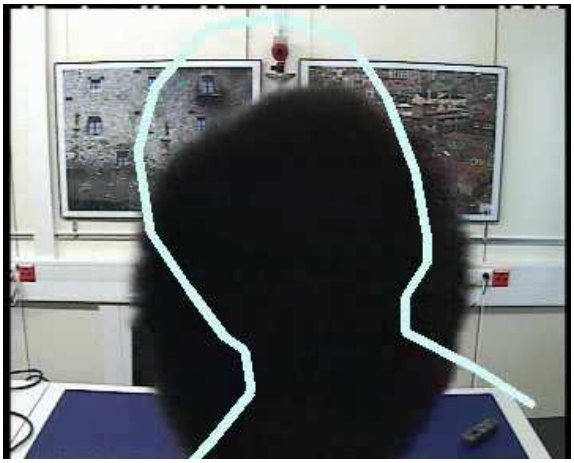


Abbildung 4.9 – Visualisierung eines Tracks (Mittel aller Hypothesen), wie er durch das gradientenbasierte ASM-Verfahren zur Personenverfolgung detektiert wurde. Obwohl der Kopf aufgrund der kameranahen Position und der nur kurzen Präsenz nicht exakt in seinen Ausmaßen erfasst wurde, so zeigt sich bei diesem Verfahren dennoch das Potential, auch in solchen Situationen erfolgreich zu agieren, wenngleich sich dies in einer Verschlechterung der F-Bewertung niederschlägt.

unterschiedlicher Modellierungen bei gleicher Adaptionsstrategie, dass die formfeste Ellipse des Systems C offenbar nur bedingt einer realen Rundumsicht des Kopfes genügt und letztlich den Vorteil des hybriden Partikelfilter nicht auszureizen vermag.

Personenkonfiguration

Wie im Zusammenhang mit der oben diskutierten Analyse der F-Bewertung bereits angeklungen ist, bedarf es zu einem genaueren Verständnis der Trackingresultate einer Betrachtung aller Fehlermaße, um Schwachstellen einzelner Verfahren diskutieren zu können. Hierzu sind in Tabelle 4.3 für genau diejenigen Sequenzen, welche nur eine einzige Person zeigen, sämtliche Fehler gelistet. Anhand der $\overline{\text{FN}}$-Fehler, welche die durchschnittliche Rate von nicht detektierten Personen pro Zeitschritt und Referenzobjekt ausweisen, lassen sich insbesondere Rückschlüsse auf die Personendetektionsstufe ziehen, die sich aufgrund der stets gegebenen Sensitivität auf äußere Störungen wie Beleuchtungsschwankung-

en oder Verdeckungen als maßgeblich verantwortlich für die meisten dieser Fehler zeichnet und durch die umgebende Architektur entsprechend korrigiert werden soll. Auch hier zeigt das System A, basierend auf Haar-ähnlichen Wavelets, seine Qualität, die jedoch damit erkauft wird, dass ein aufwendiges Training vorangehen musste. Ebenfalls zur Kategorie der trainingsbasierten Systeme zählt Ansatz B, welcher integriert in einen Partikelfilter etwa eine ähnliche Performanz zeigt. Wie in Tabelle 4.3 zu erkennen, kann bzgl. dieses Fehlermaßes ein formfestes Ellipsenmodell (System C), welches keines vorab durchzuführenden Trainingsprozesses bedarf, meist eine ebenso zuverlässige Detektion leisten. Vergleicht man mit dem formfesten Ansatz nach Viola u. Jones und den formadaptiven ASM zwei (trainingsbasierte) Techniken, so zeigen sich bei den $\overline{\text{FN}}$-Fehlern tendenziell Vorteile der Detektionsleistung auf Seiten der ASM. Diese sind hauptsächlich mit der Einbettung des Modells in einen hybriden Partikelfilter zu erklären, da so die Anpassungsfähigkeit der modellierten Konturen an die Bilddaten voll ausgespielt werden kann, wobei dennoch jede einzelne Kontur implizit gesteuert wird durch die Verteilung der einzelnen Hypothesen und damit bereits nahe von lokalen Minima im hochdimensionalen Suchraum platziert werden kann. Eine Gegenüberstellung der beiden ASM-basierten Techniken (vgl. System D und E) läßt zudem sofort erkennen, dass die Strategie, auf der die bilddatengetriebene Adaption von Statten geht, wesentlichen Einfluss auf die Qualität der Methodik hat. So erweist sich im vorliegenden Anwendungsfall, vor allem bedingt durch den teilweise stark strukturierten Hintergrund sowie die verschiedenen Kopfansichten, ein Gabor-Wavelet basierter Ansatz wegen des sehr schwach ausgeprägten Konvergenzverhaltens während des Adaptionsprozesses als wenig zielführend, was sich insofern bereits durch die niedrigen F-Bewertungen andeutete. Mit Ausnahme des Systems E implizieren die Werte speziell bei den Sequenzen 03L und 03R, dass die Personendetektionsstufe offenbar stark von einer probabilistischen Systemarchitektur profitieren kann und sich damit positiv auf die Zahl der vom Algorithmus nicht erfassten Personen auswirkt. Fälschlicherweise als Personen erkannte Bereiche des Bildes werden durch die Kenngröße $\overline{\text{FP}}$ erfasst. Anhand der Ergebnisse in Tabelle 4.3 ist unmittelbar ersichtlich, dass sich auch in dieser Fehlergröße wiederum die Qualität der Personendetektionsstufe spiegelt. Wie aus der Literatur bekannt, werden für den Detektor nach Viola u. Jones [110] generell nur sehr selten Bildbereiche irrtümlicherweise als Gesicht klassifiziert, was durch die für System A vorliegenden Ergebnisse über alle sechs Sequenzen erneut bestätigt wird. Als wesentlich störempfindlicher erweist sich hingegen das mittels eines NN trainierte Modell in System B, welches bei den von der linken Kameraperspektive aufgenommen Sequenzen in dem zu sehenden Bücherregal oftmals fälschlicherweise Gesichter detektiert. Für die anderen Sequenzen jedoch

		01L	01R	02L	02R	03L	03R
$\overline{\text{FN}}$	A	0.09	0.14	0.02	0.27	0.30	0.37
	B	0.11	0.14	0.04	0.27	0.21	0.32
	C	0.08	0.13	0.14	0.29	0.16	0.23
	D	0.08	0.11	0.08	0.33	0.12	0.14
	E	0.17	0.45	0.04	0.43	0.33	0.51
$\overline{\text{FP}}$	A	0.00	0.00	0.04	0.04	0.01	0.03
	B	0.91	0.00	0.96	0.02	1.00	0.01
	C	0.14	0.17	0.18	0.33	0.16	0.20
	D	0.09	0.16	0.12	0.43	0.03	0.07
	E	0.20	0.14	0.12	0.08	0.37	0.02
$\overline{\text{MT}}$	A	0.00	0.00	0.00	0.00	0.00	0.00
	B	0.00	0.00	0.00	0.00	0.00	0.00
	C	0.00	0.00	0.00	0.00	0.00	0.00
	D	0.00	0.00	0.00	0.00	0.00	0.00
	E	0.00	0.00	0.00	0.00	0.00	0.00
$\overline{\text{MO}}$	A	0.00	0.00	0.00	0.00	0.00	0.00
	B	0.00	0.00	0.00	0.00	0.00	0.00
	C	0.00	0.00	0.00	0.00	0.00	0.00
	D	0.00	0.00	0.00	0.00	0.00	0.00
	E	0.00	0.00	0.00	0.00	0.00	0.00
$\overline{\text{CD}}$	A	0.09	0.14	0.06	0.22	0.30	0.38
	B	0.89	0.14	0.92	0.29	0.79	0.32
	C	0.06	0.14	0.04	0.12	0.09	0.14
	D	0.05	0.14	0.08	0.14	0.11	0.16
	E	0.09	0.47	0.08	0.39	0.07	0.51
$\overline{\text{FIO}}$	A	0.00	0.00	0.00	0.00	0.00	0.00
	B	0.00	0.00	0.00	0.00	0.00	0.00
	C	0.00	0.00	0.00	0.00	0.00	0.00
	D	0.00	0.00	0.00	0.00	0.00	0.00
	E	0.00	0.00	0.00	0.00	0.00	0.00
$\overline{\text{FIT}}$	A	0.00	0.00	0.00	0.00	0.00	0.00
	B	0.02	0.00	0.00	0.00	0.00	0.00
	C	0.00	0.00	0.00	0.00	0.00	0.00
	D	0.00	0.00	0.00	0.00	0.00	0.00
	E	0.00	0.00	0.00	0.00	0.00	0.00
$\overline{Q}_{\mathcal{T}}$	A	1.00	1.00	0.88	0.80	0.94	0.91
	B	0.14	1.00	0.13	0.89	0.13	0.95
	C	0.61	0.66	0.10	0.30	0.64	0.59
	D	0.70	0.69	0.40	0.19	0.91	0.85
	E	0.38	0.00	0.50	0.00	0.25	0.00
$\overline{Q}_{\mathcal{O}}$	A	0.68	0.69	0.88	0.38	0.34	0.27
	B	0.58	0.69	0.75	0.38	0.52	0.37
	C	0.74	0.72	0.13	0.33	0.64	0.55
	D	0.74	0.76	0.50	0.24	0.73	0.72
	E	0.42	0.00	0.75	0.00	0.27	0.00

Tabelle 4.3 – Zusammenstellung der Ergebnisse nach Fehlertypen für sämtliche Systeme zur Einzelpersonenverfolgung, wie sie sich gemäß der Analyse des Evaluierungsschemas auf den sechs herangezogenen Videosequenzen ergeben haben.

ergibt sich ein ähnlich hohes Niveau wie für System A. Eine Evaluierung der konturbasierten Detektionsverfahren in den Systemen C, D und E zeigt hier im Gegensatz zu den erscheinungsbasierten Verfahren, die im Allgemeinen von der zusätzlichen Information über die Textur innerhalb des Objektes erwartungsgemäß profitieren, einen signifikanten – wenn auch in absoluten Zahlen betrachtet immer noch als gering einzustufenden – Anstieg der Fehlergröße. Jedoch zeigen sich über die Sequenzen hinweg im Vergleich zu dem NN-basierten Modell stabilere Ergebnisse, die auf eine geringere situative Abhängigkeit der Detektion schließen lassen.
Die Auswertung von $\overline{\text{MT}}$-Fehlern im Rahmen der Einzelpersonenverfolgung erübrigt sich insofern, als dass definitionsgemäß maximal nur ein einziger Track erzeugt wird und dadurch dieser Fehlertypus nicht auftreten kann. Ebenso entfällt die Diskussion der $\overline{\text{MO}}$-Fehler, da in den in diesem Abschnitt betrachteten Szenarien die aufgenommenen Sequenzen ausschließlich eine einzige Person zeigen, weswegen auch eine Zuweisung eines Tracks auf mehrere Referenzobjekte a-priori ausgeschlossen werden kann.
Wie aus Tabelle 4.3 abzulesen, wird mit dem Maß $\overline{\text{CD}}$ eine Größe zur Verfügung gestellt, die stark korreliert mit den restlichen gerade diskutierten Kennzahlen. Obwohl $\overline{\text{CD}}$-Fehler als alleiniges Maß für die Personenkonfiguration zwar nur bedingt aussagekräftig[12] sind, so bietet diese Größe im Allgemeinen dennoch einen ersten richtungsweisenden qualitativen Eindruck von Trackingergebnissen. Auch für den hier untersuchten Fall der Einzelpersonenverfolgung spiegelt dieses Maß den bereits subjektiv empfundenen Eindruck, dass System D messgrößenübergreifend in nahezu allen Sequenzen die besten Leistungen zeigt, gefolgt von den Systemen C und A. Zusammenfassend übertreffen demzufolge die hybriden Architekturen einfacher strukturierte Ansätze durch eine gesamtheitliche Betrachtung des Trackingproblems bzgl. der Erfassung der Personenkonfiguration.

Personenidentitäten

Desweiteren soll eine Beurteilung der Leistungsfähigkeit der Systeme bzgl. der Feststellung von Personenidentitäten durch eine genauere Betrachtung der zugehörigen Fehlertypen vorgenommen werden. Hierbei kann wiederum a-priori der Fehlertypus $\overline{\text{FIO}}$ von der Analyse ausgeschlossen werden, da durch die Beschränkung auf nur eine zu verfolgende Person (und somit nur ein einziges Referenzobjekt) derartige Fehler nicht auftreten können. Eine Analyse der Trackingergebnisse bezüglich der Identitätszuordnung konzentriert sich daher neben den

[12] Aufgrund der möglichen Fehlerauslöschung von $\overline{\text{FN}}$- sowie $\overline{\text{FP}}$-Fehlern kann für eine exakte Abbildung von Trackingergebnissen auf objektive Messgrößen das $\overline{\text{CD}}$-Maß nicht als alleinig repräsentativ betrachtet werden.

beiden Gütemaßen $\overline{Q}_{\mathcal{T}}$ und $\overline{Q}_{\mathcal{O}}$ auf die $\overline{\text{FIT}}$-Fehler, die prinzipiell vornehmlich dann auftreten, wenn einem Referenzobjekt aufgrund von instabilem Verhalten des Trackingalgorithmus und den daran anschließenden Neuinitialisierungen immer wieder vermeintlich eine neue Identität zugewiesen wird. Alternativ dazu werden $\overline{\text{FIT}}$-Fehler auch durch eine gegenseitige Verdeckung von Personen hervorgerufen, bei der nach Auflösen der Verdeckung die Identitäten miteinander vertauscht werden. Nachdem die zweite Quelle für derartige Fehler im vorliegenden Experiment a-priori ausgeschlossen werden kann, müssten jegliche $\overline{\text{FIT}}$-Fehler zwangsläufig auf die Instabilität des Trackingsystems zurückzuführen sein. Hier zeigt sich jedoch anhand der durchwegs vernachlässigbar niedrigen Zahlenwerte $\overline{\text{FIT}}$, dass sämtliche Systeme über den gesamten zeitlichen Verlauf der Sequenzen ein stabiles Verhalten zeigen und demnach durch die praktizierte histogrammbasierte Objektzuordnung anhand des zuletzt als sicher erkannten Tracks für ein Referenzobjekt ein Identitätswechsel nicht stattfindet.
Eine alleinige Betrachtung dieses Fehlertypus reicht – wie eingangs erläutert – jedoch im Sinne einer umfassenden Beurteilung der Fähigkeit eines Trackingsystems, Identitäten einwandfrei den gegebenen Referenzobjekten zuzuordnen, nicht aus. Weiteren Aufschluss bieten hier die Gütemaße $\overline{Q}_{\mathcal{T}}$ und $\overline{Q}_{\mathcal{O}}$. Während sich die bereits in der $\overline{\text{F}}$-Bewertung und in den Fehlermaßen $\overline{\text{FN}}$ sowie $\overline{\text{FP}}$ gezeigten Schwächen der Systeme B und E in einem signifikanten Abfall der Gütemaße $\overline{Q}_{\mathcal{T}}$ bzw. $\overline{Q}_{\mathcal{O}}$ niederschlagen, so kann insbesondere System A mit einer konstant hohen Trackergüte und somit einer sehr vertrauenswürdigen Identitätsbestimmung, die durch die hohe Präzision des Detektionsprinzips und einer damit einhergehenden konstanten Histogrammrepräsentation ermöglicht wird, den gewonnenen positiven Eindruck bestätigen. Im Vergleich mit System A sind bei den Systemen C und D mit ihren konturbasierten Personenmodellierungen Minderleistungen bzgl. der Trackergüte festzustellen. Als ursächlich für diese reduzierten Werte zeichnen sich hierbei nicht tatsächliche Probleme bei der Identitätsbestimmung[13], sondern vielmehr die Tatsache, dass die in diesen Systemen verwendete Art der Personenmodellierung verbunden mit dem datengetriebenen Adaptionsprozess der Modelle stets dazu führt, dass erst nach einem – wenngleich meist nur kurzen – Abklingverhalten ein Track, auch bei Verschwinden der realen Person im Bild, verspätet gelöscht wird. Bedingt durch die somit erhöhte Lebensdauer des Tracks reduziert sich damit unmittelbar die Trackergüte. Selbiges ist ebenso auf die Objektgüte der durch die Systeme C und D erzielten Ergebnisse zu übertragen, da selbstverständlich neben dem angesprochenen Abkling- auch ein entsprechendes Einschwingverhalten zu beobachten ist, was

[13] Diese Fehlerquelle, die sich prinzipiell auch in der Güte niederschlagen würde, kann mangels $\overline{\text{FIO}}$- sowie $\overline{\text{FIT}}$-Fehler ausgeschlossen werden.

auch an den $\overline{\text{FN}}$-Fehlern abzulesen ist. Insgesamt erweisen sich jedoch bzgl. der Objektgüte diejenigen Systeme, die auf einer hybriden Architektur basieren, mit Ausnahme von System E als geringfügig performanter gegenüber dem reinen bottom-up bzw. top-down Ansatz.

4.4.3 Zusammenfassung der Ergebnisse

Abschließend wird mit Diagramm 4.10 eine übersichtliche Gesamtbewertung der fünf evaluierten Systeme versucht, indem über alle Sequenzen entsprechend ihrer Länge eine Mittelung der Evaluationsgrößen vorgenommen wird. Die berechne-

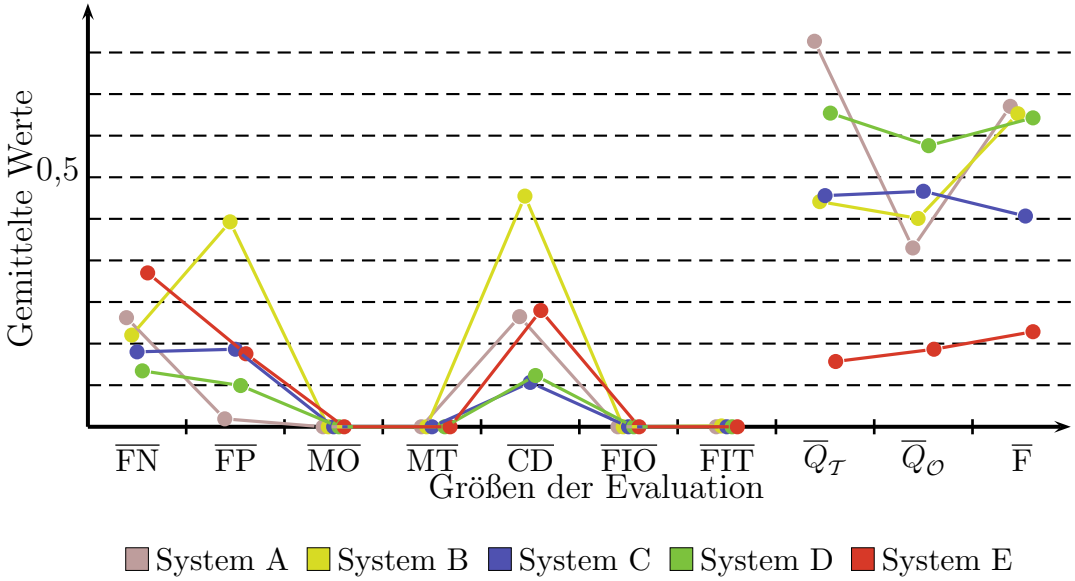


Abbildung 4.10 – Zusammenfassende Darstellung der Evaluationsergebnisse: Für jedes der fünf Systeme wurden die Fehlergrößen über alle sechs Sequenzen gewichtet mit der jeweiligen Länge gemittelt. Im linken Teil des Diagramms sind die Graphen über denjenigen Größen angetragen, die möglichst niedrige Werte annehmen sollen, im rechten Teil sollen die Werte möglichst hoch sein.

ten Werte sind für jedes System über der jeweiligen Fehlergröße angetragen. Da die Werte der Fehlermaße, welche die Personenkonfiguration betreffen, ebenso wie die $\overline{\text{FIT}}$- und $\overline{\text{FIO}}$-Werte im Sinne eines guten Trackingverfahrens möglichst klein, die restlichen Fehlergrößen hingegen Werte nahe eins erreichen sollten, wurden die Graphen jeweils im Diagramm unterteilt: je näher der linke Ast des Graphen an der Abszisse liegt und sich der rechte Ast von dieser entfernt,

desto besser das zugehörige Trackingsystem. Hierbei ergibt sich – als qualitative Bestätigung des sich bei der vorangegangenen Diskussion abzeichnenden Eindruckes – folgendes Gesamtbild: System D, welches auf einer hybriden Architektur basiert, kann sich insgesamt gegenüber den rein bottom-up (System A) bzw. top-down (System B) agierenden ansichtsbasierten Techniken behaupten. Ebenso kann insbesondere die in einer hybriden Architektur eingebettete formfeste Ellipsenmodellierung (System C) mit den von System B generierten Ergebnissen vergleichbare, mitunter sogar tendenziell bessere Leistungen erzielen. Abgeschlagen auf dem letzten Platz rangiert System E, bei dem aufgrund der nur mäßigen Detektionsleistung vor allem bei stark strukturierten Hintergründen keine sinnvolle Verfolgung von Personen möglich war.

4.5 Evaluation Mehrpersonenverfolgung

Eine weitere Evaluation (vgl. Schreiber u. Rigoll [91]) galt schließlich den in Abschnitt 3.3 präsentierten Architekturen zur simultanen Verfolgung mehrerer Personen, die basierend auf verschiedenen Detektionsmodulen einander gegenübergestellt wurden. Auch hier diente wiederum der Standard bottom-up Ansatz (System A) als Referenz.

4.5.1 Evaluierte Systeme zur Mehrpersonenverfolgung

Vor dem Hintergrund der aus dem vorherigen Abschnitt gewonnenen Erkenntnisse kamen für die Evaluierung der entwickelten Systemarchitekturen nur noch diejenigen Personenidentifikationsmodule zum Einsatz, für die sich ein entsprechendes Potential im Hinblick auf eine Eignung zum Mehrpersonentracking herauskristallisiert hatte. Konkret handelt es sich dabei um die folgenden vier Systeme:

System A Hierbei handelt es sich um das aus dem Abschnitt 4.4 bekannte System A, welches prinzipbedingt bereits die Fähigkeit zur simultanen Verfolgung mehrerer Personen aufweist und im Zuge dieser Evaluation erneut als Referenz dient.

System F Grundlage bildet die in Abschnitt 3.3.1 beschriebene hierarchische Hybrid-Architektur (vgl. Abbildung 3.10) bestehend aus zwei Partikelfiltersystemen. Zur Messung in der Personendetektionsstufe wird das formveränderliche ASM verwendet, welches im datengetriebenen Adaptionsprozess auf Basis der Gradienten im Bild optimiert wird.

System G Die in der Architektur aus Abschnitt 3.3.2 vollzogene Kombination von stochastischem Partikelfilter und heuristischem SA-Verfahren findet bei diesem System Anwendung. Die Gewichte der Hypothesen des Partikelfilters werden dabei durch ein auf Gesichter trainiertes NN nach dem Ansatz von Rowley u. a. [86] bestimmt. Aufgrund der fehlenden modelleigenen Adaptionsfähigkeit an die zugrunde liegenden Daten ist das System – abweichend von der Darstellung in Abbildung 3.12 – jedoch nur als top-down Ansatz gestaltet, d. h. dass der Rückkanal zwischen Personendetektion und Messung aufgebrochen wurde.

System H Ebenfalls auf derselben Architektur basiert dieses System, allerdings erfolgt die Gewichtung der Hypothesen aufgrund der Messwerte, die sich durch die gradientenbasierte Anpassung von ASM ergeben. Bedingt durch die lokale Adaptionsfähigkeit des Modells entspricht die Struktur dieses Systems dem Blockschaltbild in Abbildung 3.12 und stellt somit die in dieser Arbeit im Fokus stehende hybride Umsetzung eines Verfahrens zur simultanen und omnidirektionalen Verfolgung mehrerer Personen dar.

Bei allen Systemen wurden wie in Abschnitt 4.4 die Vorverarbeitungsschritte in der gleichen Weise beibehalten. Ebensolches gilt für die situative Adaption der Hypothesenanzahl in den Partikelfiltern zur Einzelpersonenverfolgung. Lediglich der Partikelfilter zur Bestimmung der Personenkonfiguration in System F arbeitet mit einer festen Hypothesenanzahl $N_S = 30$.

4.5.2 Diskussion der Evaluationsergebnisse

Grundlage für diesen Vergleich waren alle 18 der in Abschnitt 4.2 aufgelisteten und zum Zwecke der Evaluierung ausgewählten Videosequenzen. Mit aufsteigender Nummer der Sequenz nimmt dabei die Zahl der an der jeweiligen Besprechung teilnehmenden Personen von eins bis vier zu, womit einhergeht, dass Verdeckungen wahrscheinlicher werden und damit tendenziell Sequenzen mit höherer Nummer als anspruchsvoller erachtet werden können. Durch die folgende Analyse soll im Hinblick auf die in den Videodaten beinhalteten Herausforderungen Schwachpunkte sowie Stärken der einzelnen Systeme herausgearbeitet werden.

Passgenauigkeit - die F-Bewertung

Analog zum vorigen Abschnitt soll auch hier zunächst wiederum auf die Genauigkeit der Zuordnung von ermittelten Tracking- und Referenzobjekten eingegangen

werden. Hierzu ist in Abbildung 4.11 die F-Bewertung als Mittel über der jeweiligen Sequenz für jede der vier Methoden als Balkendiagramm gegeben. Wie dar-

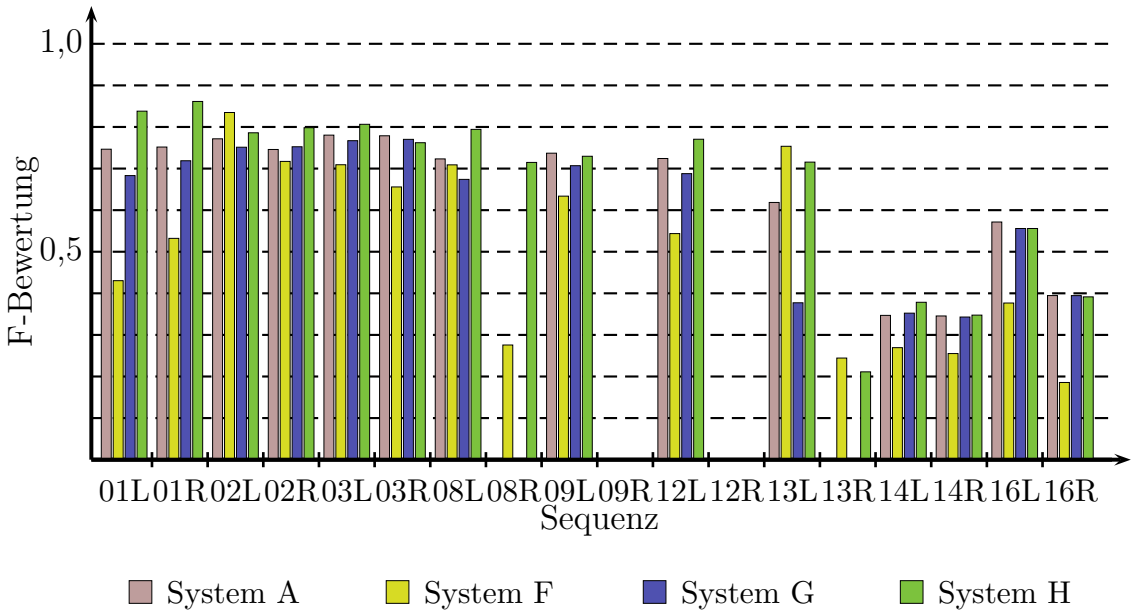


Abbildung 4.11 – Diagramm der mittleren F-Bewertung für sämtliche Systeme zur simultanen Verfolgung mehrerer Personen, aufgetragen über der jeweils evaluierten Sequenz.

aus unmittelbar hervorgeht, schneidet dabei die Kombination aus heuristischem Optimierungsverfahren und stochastischem Partikelfilter mit formadaptiver Personenmodellierung (System H) für mehr als die Hälfte aller Sequenzen am besten ab und übertrifft sogar die – betreffend der Detektionspräzision – als sehr exakt geltende Methode nach Viola u. Jones (System A) um durchschnittlich ca. 10 %. Bei einem direkten Vergleich von top-down (System G) und hybrider (System H) Trackingarchitektur, jeweils auf unterschiedlichen Strategien zur Personendetektion beruhend, ist der Vorteil der bilddatengetriebenen Hypothesenplatzierung mittels ASM wie auch schon bei den Systemen zur Einzelpersonenverfolgung erneut klar erkennbar. Als im Mittel schwächster Algorithmus zeigt sich die doppelschichtige Partikelfilterarchitektur (System F). Dies begründet sich darin, dass trotz des hierarchischen Aufbaus ein Konvergieren der Partikel unterschiedlicher Objekte – vermeintlicher sowie tatsächlicher Art – gelegentlich dennoch vorkommt und dadurch der jeweilige Track vom eigentlich zu verfolgenden Objekt wegdriftet. Während dieses üblicherweise langsam ablaufenden Driftprozesses wird zwar der Track immer noch dem zugehörigen Referenzobjekt zugeordnet, was sich aber wegen des abnehmenden Überlappungsgrades negativ

in einer nur mäßigen F-Bewertung niederschlägt.
Der klare Abfall in der F-Bewertung bei den Sequenzen ´08R´, ´09R´, ´12R´ sowie ´13R´ rührt von der Tatsache her, dass in allen vier Videoaufzeichnungen ausschließlich die Hinterköpfe der Personen zu sehen sind und sich diese darüber hinaus in unmittelbarer Nähe zur Kamera befinden, was zusätzlich erschwerend zu einer nur teilweisen Abbildung des Kopfes im Videobild führt. Während für die Systeme A und G, die auf den ansichtsbasierten Techniken zur Personendetektion beruhen, ein erfolgreiches Personentracking gänzlich misslingt, so erweist sich speziell in diesen Szenarien eine Personenmodellierung mittels eines flexiblen Modells wiederum vorteilhaft, wodurch sogar in zwei der vier genannten Sequenzen ein Tracking von Personen grundsätzlich ermöglicht wird.

Personenkonfiguration

Zur genaueren Analyse der durch die einzelnen Algorithmen erzielten Ergebnisse bedarf es neuerlich einer detaillierten Untersuchung der diversen Fehlergrößen. Vergleicht man die in Tabelle 4.4 gelisteten Ergebnisse der evaluierten Systeme zur Mehrpersonenverfolgung auf den Einzelpersonenszenarien (Sequenzen ´01L´ - ´03R´) mit denjenigen aus Tabelle 4.3, so bemerkt man, dass die Zahlenwerte allgemein nur unwesentlich differieren und daher offenbar sämtliche Architekturen die Zahl der im Video präsenten Personen zuverlässig ermitteln können, solange die Personendetektion verlässliche Ergebnisse liefert. Wie eingangs erläutert, nimmt mit steigender Sequenznummer die Zahl der Teilnehmer zu. Damit einhergehend ändert sich in grundlegender Weise auch die Verhaltensweise der Personen: Während Personen, die sich alleine im Besprechungszimmer befinden, überwiegend frontal durch die Kamera erfasst werden, agieren mit wachsender Zahl der Teilnehmer diese verstärkt untereinander, so dass die Blickrichtung einer Person mit jedem Wechsel des Gesprächspartners häufig geändert wird und in der Kameraperspektive somit oftmals auch Kopfansichten über das Profil hinaus vorkommen. Gerade in derlei Situationen versagen erwartungsgemäß die auf Frontal- sowie Halbprofilansichten trainierten Gesichtsmodelle nach Viola & Jones (als bottom-up Ansatz in System A) sowie nach Rowley (integriert in das heuristisch-probabilistische Trackingsystem G), was zu stark erhöhten Zahlenwerten des Fehlermaßes $\overline{\mathrm{FN}}$ führt. Im Gegensatz dazu gelingt es den formveränderlichen ASM in einer Vielzahl von Sequenzen, durch eine entsprechende Modelladaption Köpfe auch bei in der Tiefe gedrehten Ansichten zu verfolgen. In einem direkten Vergleich von System F und H kann hierbei gezielt der Einfluss der Trackingarchitektur auf die Qualität der Ergebnisse untersucht werden. Wie bereits im Zuge der Analyse der F-Bewertung im vorangegangenen Abschnitt diskutiert, kann durch den hierarchischen Partikelfilteransatz (System

		01L	01R	02L	02R	03L	03R	08L	08R	09L	09R	12L	12R	13L	13R	14L	14R	16L	16R
$\overline{\text{FN}}$	A	0.09	0.14	0.02	0.27	0.30	0.37	0.88	0.75	0.63	0.49	0.71	0.75	0.76	0.83	0.67	0.75	0.26	0.29
	F	0.28	0.41	0.02	0.20	0.20	0.26	0.75	0.73	0.47	0.49	0.78	0.75	0.69	0.81	0.81	0.84	0.64	0.34
	G	0.11	0.14	0.04	0.27	0.21	0.32	0.78	0.75	0.54	0.49	0.75	0.75	0.80	0.83	0.68	0.75	0.29	0.33
	H	0.08	0.13	0.02	0.14	0.12	0.16	0.03	0.73	0.13	0.49	0.38	0.75	0.50	0.81	0.43	0.55	0.29	0.18
$\overline{\text{FP}}$	A	0.00	0.00	0.04	0.04	0.01	0.03	0.03	0.30	0.08	0.00	0.03	0.11	0.06	0.08	0.11	0.06	0.12	0.02
	F	0.23	0.28	0.00	0.16	0.11	0.15	0.36	0.11	0.28	0.05	0.31	0.10	0.27	0.13	0.32	0.21	0.21	0.03
	G	0.91	0.00	0.96	0.02	1.00	0.01	0.50	0.00	0.70	0.01	0.51	0.00	0.48	0.00	0.48	0.00	0.57	0.01
	H	0.00	0.02	0.00	0.06	0.00	0.01	0.07	0.00	0.31	0.00	0.11	0.00	0.28	0.00	0.16	0.11	0.57	0.09
$\overline{\text{MT}}$	A	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00
	F	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	G	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	H	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.04	0.00	0.01	0.00	0.00	0.00	0.01	0.01	0.00	0.02
$\overline{\text{MO}}$	A	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	F	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	G	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	H	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
$\overline{\text{CD}}$	A	0.09	0.14	0.06	0.22	0.30	0.38	0.88	0.50	0.61	0.49	0.72	0.64	0.72	0.74	0.58	0.69	0.36	0.30
	F	0.08	0.13	0.02	0.16	0.14	0.12	0.42	0.62	0.19	0.44	0.50	0.65	0.52	0.67	0.51	0.63	0.42	0.31
	G	0.89	0.14	0.92	0.29	0.79	0.32	0.47	0.75	0.39	0.48	0.52	0.75	0.44	0.82	0.33	0.75	0.44	0.33
	H	0.08	0.14	0.02	0.20	0.12	0.17	0.11	0.73	0.35	0.49	0.29	0.75	0.35	0.81	0.30	0.44	0.44	0.26
$\overline{\text{FIO}}$	A	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.08	0.00	0.05	0.00	0.08	0.01	0.26	0.03
	F	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.01	0.00	0.06	0.00	0.11	0.00	0.04	0.02	0.07	0.00
	G	0.00	0.00	0.00	0.00	0.00	0.00	0.07	0.00	0.00	0.00	0.07	0.00	0.03	0.00	0.07	0.02	0.23	0.02
	H	0.00	0.00	0.00	0.00	0.00	0.00	0.06	0.00	0.01	0.00	0.00	0.00	0.05	0.00	0.00	0.00	0.23	0.09
$\overline{\text{FIT}}$	A	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.02	0.00	0.01	0.00	0.03	0.01	0.25	0.03
	F	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	G	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.02	0.00	0.01	0.00	0.04	0.02	0.23	0.02
	H	0.00	0.16	0.00	0.08	0.00	0.06	0.52	0.00	0.39	0.00	0.20	0.00	0.25	0.00	0.30	0.20	0.23	0.24
$\overline{Q}_{\mathcal{T}}$	A	1.00	1.00	0.88	0.80	0.94	0.91	0.80	0.00	0.50	0.00	0.61	0.00	0.57	0.00	0.47	0.61	0.44	0.86
	F	0.06	0.14	1.00	0.58	0.70	0.62	0.27	0.20	0.56	0.00	0.12	0.00	0.17	0.12	0.22	0.22	0.34	0.90
	G	0.14	1.00	0.13	0.89	0.13	0.95	0.19	0.00	0.26	0.00	0.14	0.00	0.14	0.00	0.21	0.79	0.45	0.92
	H	1.00	0.96	1.00	1.00	1.00	0.97	0.97	1.00	0.75	0.00	0.88	0.00	0.76	1.00	0.85	0.86	0.45	0.91
$\overline{Q}_{\mathcal{O}}$	A	0.68	0.69	0.88	0.38	0.34	0.27	0.05	0.00	0.20	0.00	0.15	0.00	0.14	0.00	0.39	0.35	0.36	0.57
	F	0.05	0.10	0.88	0.52	0.56	0.49	0.17	0.08	0.31	0.00	0.09	0.00	0.23	0.11	0.17	0.17	0.26	0.85
	G	0.58	0.69	0.75	0.38	0.52	0.37	0.15	0.00	0.26	0.00	0.10	0.00	0.17	0.00	0.37	0.33	0.36	0.54
	H	0.74	0.38	0.88	0.48	0.72	0.56	0.44	0.02	0.31	0.00	0.49	0.00	0.21	0.07	0.42	0.41	0.36	0.55

Tabelle 4.4 – Zusammenstellung der Evaluierungsergebnisse, die für jedes der vier Systeme (A, F, G, H) zur simultanen Mehrpersonenverfolgung und jede Sequenz (01L-16R) gelistet sind. Grundlage hierfür sind die im Kontext des Evaluierungsschemas definierten Fehlergrößen, jeweils gemittelt über die Länge einer Sequenz.

F) einem gelegentlichen Konvergieren von Partikeln unterschiedlicher Objekte an einer Position im Bild nicht vollends vorgebeugt werden, was letztlich gleichzeitig verstärkt $\overline{\text{FP}}$- sowie $\overline{\text{FN}}$-Fehler hervorruft[14]. Mit System H schließlich ist es gelungen, durch die Verknüpfung von stochastischem Partikelfilter mit heuristi-

scher Nachbarschaftssuche in Verbindung mit einem adaptiven Personenmodell die Zahl der Auslassungen (FN) nochmals signifikant zu verringern und sehr erfolgreiche Trackingergebnisse zu generieren. Desweiteren kann durch diese Architektur das im Zuge der Einzelpersonenverfolgung beobachtete problematische Einschwing- und Abklingverhalten der datengetriebenen Modelladaption offenbar sogar noch etwas gemildert werden, so dass sich hier nun – auch bei den überwiegend frontalen Ansichten in den Sequenzen ´01L´ bis ´03R´ – die Zahl der $\overline{\text{FP}}$-Fehler auf das Niveau von System A absenken läßt.
Während bei den Systemen im vorangegangenen Experiment zur Einzelpersonenverfolgung die Diskussion von MO- sowie MT-Fehlern a-priori obsolet war, können bei den evaluierten Systemen zur Mehrpersonenverfolgung diese beiden Fehlertypen prinzipiell auftreten. Die Analyse der Trackingergebnisse zeigt jedoch, dass sowohl MO-Fehler, die eine Repräsentation mehrerer Personen durch einen einzigen Track erfassen, als auch MT-Fehler, die auf eine jeweils multiple (und dann evtl. nur partikuläre) Erfassung eines Gesichts/Kopfes durch die Personendetektion schließen lassen, nur äußerst selten auftreten. Die Evaluierungen liefern somit die Erkenntnis, dass diese Fehlermaße bei einer vernünftigen Strategie der Personendetektion augenscheinlich eine nur untergeordnete Rolle bei Algorithmen zur Verfolgung von Personen spielen[15].
Auch in der Konfigurationskompaktheit $\overline{\text{CD}}$ spiegeln sich die gerade diskutierten Fakten wider: System H weist für neun der 16 Sequenzen die jeweils niedrigsten Werte aus, gefolgt von System F und und den Systemen A und G. Bei den beiden letztgenannten wirkt sich hierbei insbesondere die Tatsache negativ aus, dass vor allem bei den Sequenzen mit höherer Nummer überwiegend viele Ansichten vertreten sind, die Köpfe von hinten oder aufgrund von Schreibgesten stark gesenkt zeigen, und somit von den ansichtsbasierten Detektionsverfahren nur selten erfasst werden können.

Personenidentitäten

Gerade im Zuge des Mehrpersonentracking spielt neben der korrekten Positionsbestimmung einer Person die einwandfreie Zuordnung von Identitäten über die gesamte Laufzeit eine zentrale Rolle. Es ist daher auch für die evaluierten Systeme zur simultanen Verfolgung mehrerer Personen unerlässlich, die in Tabelle

[14] Durch das Konvergieren von Partikeln zweier unterschiedlicher Objekte an einer Position im Bild, die üblicherweise einer der Positionen der Objekte selbst entspricht, wird eines der beiden Referenzobjekte nicht mehr durch einen Track korrekt verfolgt und stattdessen seine Position an anderer Stelle vermutet.

[15] Dieser Sachverhalt ergab sich in ähnlicher Weise auch für andere Trackingsysteme bei einer im Rahmen des AMIDA-Projektes veröffentlichten Studie (vgl. Smith u. a. [101]).

4.4 gelisteten Zahlenwerte bzgl. der durch die Algorithmen ermittelten Personenidentitäten näher zu analysieren.
Wie sich bereits im vorangegangenen Experiment zur Einzelpersonenverfolgung abzeichnete, kann über die Zeitschritte hinweg durch die allein histogrammgestützte Identitätszuordnung (Systeme A und F) ein vermeintlicher Wechsel der Identität eines Referenzobjektes (FIT) für einen Großteil der Fälle vermieden werden. Anders gestaltet sich dies jedoch bei dem heuristisch-probabilistischen Ansatz in den Systemen G und H. Abhängig von der zur Personendetektion verwendeten Technik läßt sich hier – vor allem bei denjenigen Sequenzen mit stark strukturiertem Hintergrund – ein differierendes Systemverhalten bzgl. der Vergabe von Identitäten an die Tracks feststellen. Begründet werden kann dies maßgeblich durch die Beobachtung, dass die formveränderliche Modellierung von Personen mittels ASM erwartungsgemäß auch die Hinterkopfansicht einer Person detektiert, aber speziell vor stark strukturiertem Hintergrund bedingt durch ein geringfügiges Abdriften der entsprechenden Hypothesen des Partikelfilters nicht durchwegs exakt die Objektkontur erfasst wird. Dadurch wird das die Objekttextur repräsentierende Histogramm nicht unerheblichen Qualitätsschwankungen unterworfen, so dass ein Ähnlichkeitsvergleich zweier Histogramme, zusammen mit der Positions- und Größeninformation des Objektes, gelegentlich zu einem Verbleiben der heuristischen Optimierungsstrategie in lokalen Minima führt und damit Fehler in der Identitätszuordnung hervorruft. Die beabsichtigte und mit dem System H auch realisierte omnidirektionale Verfolgung von Personen, also die verbesserte Performanz in Bezug auf die Fehlergrößen der Personenkonfiguration, geht somit zu Lasten eines Anstiegs der $\overline{\text{FIT}}$-Fehler. Im Gegensatz dazu indiziert die ansichtsbasierte Detektionstechnik anhand eines NN nur dann überhaupt Personen, wenn die Texturmerkmale genügend ähnlich zum Trainingsmaterial sind, wodurch extreme Änderungen des Histogrammes a-priori begrenzt werden und damit die heuristische Optimierung nahezu immer erfolgreich agiert. Bei den $\overline{\text{FIO}}$-Fehlern hingegen zeigt sich, dass offenbar unabhängig von der gewählten Architektur und der Methodik zur Personendetektion diese Art des Fehlers keine entscheidende Größe für ein System zur simultanen Verfolgung von Personen darstellt.
Die Gütemaße $\overline{Q}_{\mathcal{T}}$ und $\overline{Q}_{\mathcal{O}}$ bilden nun entgegen der Situation in Abschnitt 4.4 nicht mehr nur im Wesentlichen die $\overline{\text{FN}}$- sowie $\overline{\text{FP}}$-Fehler ab, sondern werden auch durch die sich bei der Mehrpersonenverfolgung ergebenden $\overline{\text{FIO}}$- und $\overline{\text{FIT}}$-Fehler maßgeblich beeinflusst. In Bezug auf die Trackergüte $\overline{Q}_{\mathcal{T}}$ offenbart System F gravierende Mängel und bestätigt damit die bereits aus der Personenkonfiguration gewonnenen Erkenntnisse. Auch der in System G realisierte top-down Ansatz kann sich hierbei nur unwesentlich von System F im Hinblick auf die Leis-

tungsfähigkeit absetzen. Bei den Systemen A und H ist im direkten Vergleich festzustellen, dass zum einen durch die omnidirektionale Personenmodellierung mittels ASM, zum anderen durch die hybride heuristisch-probabilistische Architektur eine signifikante Steigerung der Performanz gegenüber einem Standard bottom-up Verfahren zu erzielen sind. Der noch klar zu verzeichnende Vorsprung des Systems H gegenüber den anderen Systemen reduziert sich bei Betrachtung der Objektgüte $\overline{Q}_{\mathcal{O}}$ bedingt durch die Zunahme der $\overline{\text{FIT}}$-Fehler. Desweiteren bestätigt sich anhand der - verglichen mit den Systemen A und G - höheren Gütewerte $\overline{Q}_{\mathcal{O}}$ des Systems H nochmals die bei der Diskussion der Ergebnisse zur Einzelpersonenverfolgung beobachtete Tendenz, dass die Objektgüte maßgeblich von einer hybriden Architektur profitiert.

4.5.3 Zusammenfassung der Ergebnisse

Analog zum Abschnitt 4.4 wird auch die Diskussion der Evaluierungsergebnisse zur simultanen Verfolgung mehrerer Personen mit dem Versuch eines gesamtheitlichen Überblickes abgeschlossen. Hierzu sind in Diagramm 4.12 wiederum für

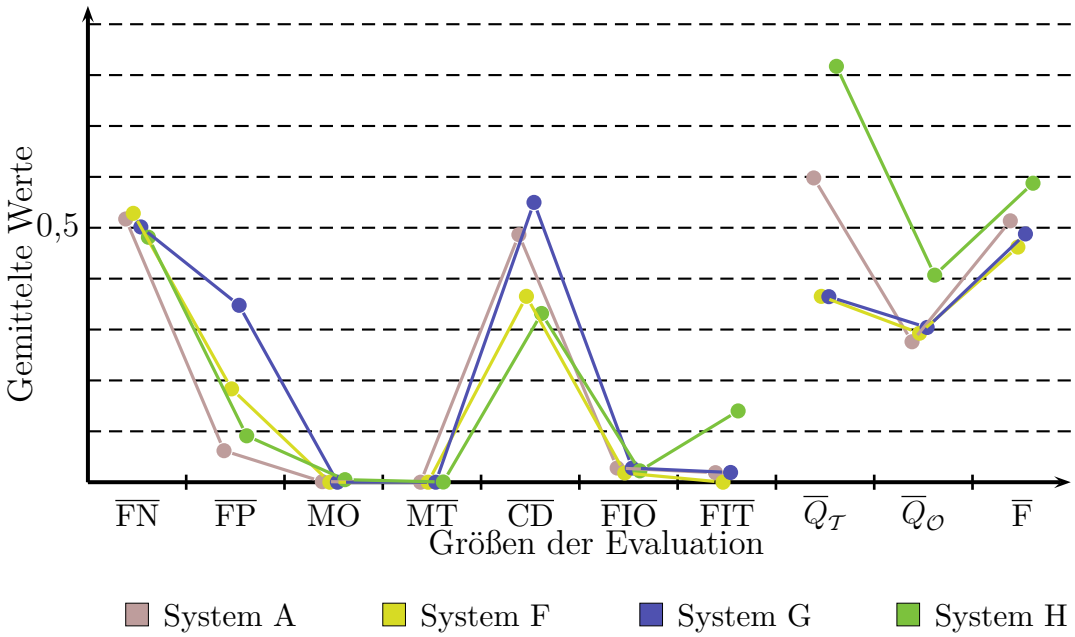


Abbildung 4.12 – Zusammenfassende Darstellung der Ergebnisse für jede der vier Methoden als gewichtetes Mittel.

alle vier evaluierten Systeme die über alle Sequenzen gemittelten Werte für die berechneten Fehlermaße angetragen. Auch hier bestätigt sich das durch die vorherige Diskussion gezeichnete Bild der im Fokus stehenden Systeme: Während System H und System A bzgl. den Maßen die Personenkonfiguration betreffend noch etwa gleichauf liegen, kann sich System H mit seiner hybriden heuristisch-probabilistischen Architektur letztlich bei den Gütemaßen und der $\overline{\mathrm{F}}$-Bewertung gegenüber dem bottom-up Ansatz klar absetzen. System F, welches ebenso wie System H eine formveränderliche Modellierung zur Detektion von Personen – jedoch eingebettet in eine hierarchische Partikelfilterstruktur – verwendet, zeigt durchwegs im Vergleich zu den beiden vorig genannten Systemen signifikant schlechtere Werte, kann sich jedoch vor allem wegen der besseren Werte bei den Fehlermaßen zur Personenkonfiguration noch vor System G behaupten.

Kapitel 5

Gesten- und Aktionserkennung

Die zwischenmenschliche Kommunikation gestaltet sich aufgrund des intuitiven Gebrauchs diverser multimodaler Fähigkeiten wie Körperbewegung, Sprache oder Schrift als sehr einfach und effizient (vgl. Geiser [36]). Durch den Einzug moderner Technik in unseren Alltag ist der Mensch jedoch immer öfter dazu gezwungen, mit maschinellen Systemen zu interagieren, wobei hier aufgrund technischer Restriktionen nur ein begrenztes Maß der genannten Mitteilungsformen zum bidirektionalen Informationsaustausch genutzt werden kann. Diese Problematik stellt einen der zahlreichen Forschungsschwerpunkte der Mensch-Maschine Kommunikation dar. Neben den bekannten Standardverfahren zur Interaktion mit technischen Systemen wie Tastatur und Computermaus konnte in den letzten Jahren mit der Sprache die Liste der möglichen Eingabemodalitäten erweitert werden. Gesten, die als der informationstragende Teil sämtlicher Körperbewegungen definiert werden können (vgl. Mitra [70]), bleiben indes als Informationsquelle immer noch nahezu unberücksichtigt. Dies dürfte sich jedoch in den nächsten Jahren durch eine wachsende Zahl an Anwendungsgebieten wie Virtual Reality oder Smart Rooms, bei denen sich durch eine Kombination aus Spracheingabe und Gesten ein sehr hoher Immersionsgrad realisieren läßt, wesentlich ändern (vgl. Schultheis [94]).
Zusätzlich zum reinen Informationsaustausch können Gesten systemintern auch zum Ausführen weiterer Aktionen benutzt werden. So läßt sich basierend auf Gesten eine Komprimierung von Datenströmen vorstellen oder das Verhalten von Automaten anpassen. Darüber hinaus kann die Erkennung von Gesten zur Ableitung von Aktionen auf semantisch höherwertiger Ebene dienen. In Besprechungsszenarien würden sich so Rückschlüsse auf die aktuelle Phase, beispielsweise Abstimmung oder Präsentation, ziehen lassen.

5.1 Datenbank

Die für die Erkennung von personenspezifischen Gesten zugrunde liegende Datenbank wurde innerhalb des M4-Projektes (MultiModal Meeting Manager) aufgezeichnet (vgl. McCowan u. a. [67]). Die Szenarien wurden in einem Raum durchgeführt, der dem im Zuge der Personenverfolgung verwendeten Aufbau ähnelt. Insgesamt wurden 59 Sitzungen von je ca. fünf Minuten Dauer abgehalten, wobei pro Sitzung genau vier Akteure teilnahmen. Auf diese Weise entstand ein Videokorpus mit einem Nettoumfang von 15 Stunden Datenmaterial. Im Gegensatz zu dem AMI-Datenkorpus wurde der Ablauf sämtlicher Besprechungen dieser Datenbank bereits vorab festgelegt. Hierfür wurden 10 unterschiedliche Gruppenaktionen wie beispielsweise „Monolog einer Person“, „Präsentation“, „Diskussion zwischen den Personen“ oder „Abstimmung“ definiert. Mittels eines ergodischen Hidden Markov Modells wurde für jede Besprechung die Abfolge dieser Aktionen bestimmt und durch einen Regisseur die zeitliche Einhaltung dieser Aktionen während der Aufnahme der Videos überwacht. All diese Aktionen können selbst wiederum als Summation bestimmter, von den einzelnen Besprechungsteilnehmern ausgeführter Basisgesten aufgefasst werden. Daher wurden für sämtliche Sitzungen dieser Datenbank neben den Gruppenaktionen zusätzlich die personenspezifischen Aktionen manuell annotiert. Daraus abgeleitet ergab sich ein Set aus sechs unterschiedlichen Basisgesten, die für eine nachfolgende Verarbeitung auf Gruppenebene von Interesse sein können:

Schreiben kann als fundamentaler Hinweis auf die Wissensvermittlung durch eine andere Person erachtet werden, wie dies beispielsweise in einer Präsentation geschieht; die Geste startet mit dem Aufsetzen des Stiftes auf dem Papier und endet mit dem sichtbaren Anheben des Stiftes.

Nicken wird gemeinhin als Geste der (stummen) Meinungsäußerung betrachtet und kann somit als Indikator für Abstimmungssituationen fungieren; Nicken muss insbesondere abgegrenzt werden zu unbewussten – und somit informationsirrelevanten – Bewegungen des Kopfes; aufgrund der sehr kurzen Dauer ist ein sehr präzises Annotieren von Beginn (= Start der ersten vertikalen Auslenkung des Kopfes) und Ende (= Beginn der Ruhestellung des Kopfes) notwendig.

Kopf schütteln als Geste kann ebenso wie Nicken als Meinungsäußerung interpretiert werden und dient somit mitunter auch als ein wichtiger Anzeiger für Abstimmungen oder Diskussionen; die Geste setzt ein mit dem Beginn

der horizontalen Auslenkung des Kopfes und endet mit der Rückkehr in die Ruheposition.

Zeigen veranlasst Personen, ihren Blick auf etwas zu richten, wodurch sich Information vermitteln läßt; die Geste beginnt mit Einnehmen der Zeigeposition und endet, sobald der Arm beginnt, sich wieder von dieser Position zu entfernen.

Aufstehen zeigt in erster Linie den Wechsel einer Gruppenaktion an; sobald die Person beginnt, sich zu erheben, setzt der Vorgang des Aufstehens ein und wird als abgeschlossen betrachtet, wenn die Person eine aufrechte Haltung einnimmt.

Sich setzen stellt das Pendant zum Aufstehen dar und weist ebenso auf den Zustandswechsel der Gruppenaktion hin; die Geste beginnt mit dem Verlassen der aufrechten Stehposition und endet, sobald sich die Person auf den Stuhl gesetzt hat.

In Tabelle 5.1 findet sich dazu nochmals eine Zusammenstellung der ausgewählten Basisgesten und einiger wichtiger statistischer Daten.

Basisgeste	Durchschnittl. Dauer [s]	Std.-abweichung Dauer [s]	Häufigkeit im Trainingsset	Testset
Schreiben	8,79	8,38	377	508
Nicken	1,97	1,38	465	236
Kopf schütteln	2,06	1,58	55	34
Zeigen	1,90	1,48	94	49
Aufstehen	1,85	0,75	10	12
Sich setzen	1,88	0,74	10	9

Tabelle 5.1 – Überblick über die für die Gestenerkennung relevanten Basisgesten und einige der wichtigsten statistischen Daten.

5.2 Merkmale

Für die Modellierung einzelner Gesten müssen passende Merkmale aus den Bilddaten extrahiert werden. Aufgrund der Tatsache, dass Gesten stets aus einer

Bewegung resultieren[1], werden diese über die Differenz $\underset{\sim}{D}'_t = \underset{\sim}{G}_t - \underset{\sim}{G}_{t-1}$ zweier aufeinanderfolgender, grauwertgewandelter Bilder $\underset{\sim}{G}_t$ sowie $\underset{\sim}{G}_{t-1}$ beschrieben. Um möglichst rauschfreie Merkmale zu erhalten, bedarf es einer entsprechenden Vorverarbeitung der Bilddaten. Aus diesem Grund wird zuerst durch Anwendung eines Schwellwertoperators (Schwelle Θ) auf das Differenzbild $\underset{\sim}{D}'_t$ eventuell vorhandenes Bildrauschen beseitigt, resultierend in einem Bild

$$\underset{\sim}{D}_t(x,y) = \begin{cases} 0 & |\underset{\sim}{D}'_t(x,y)| < \Theta \\ D'_t(x,y) & |\underset{\sim}{D}'_t(x,y)| > \Theta \end{cases}, \tag{5.1}$$

und anschließend mittels den morphologischen Operationen „opening“ und „closing“ verbleibende Artefakte beseitigt. Um nur von der aktuell betrachteten Person ausgeführte Bewegungen zu berücksichtigen, wird über die Definition eines Aktionsbereiches R, welcher anhand der durch das Tracking ermittelten Kopfposition $\vec{p}_{t,\text{Kopf}} = (t_x, t_y)^T$ festgelegt wird, der verbleibende Bereich des Bildes ausgeblendet. In Abbildung 5.1 ist das Resultat der vorangegangenen Operationen exemplarisch visualisiert. Auf dem nunmehr verbleibenden Bild werden

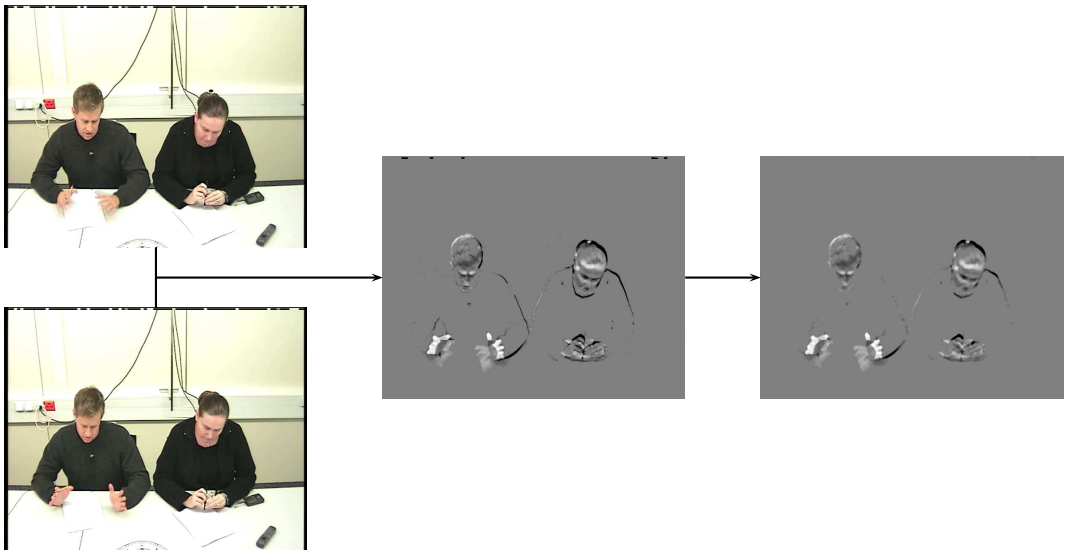


Abbildung 5.1 – Aus den zeitlich versetzten Bildern (links) wird durch Subtraktion das Differenzbild (mitte) erzeugt, welches durch eine Schwellwertoperation gefolgt von morphologischem „opening“ und „closing“ in das Ausgangsbild (rechts) überführt wird.

innerhalb des Aktionsbereiches R die sogenannten *Global Motion Merkmale*, angelehnt an die in Rigoll u. a. [84] beschriebene Darstellung, berechnet. Diese

[1] Vgl. hierzu die eingangs beschriebene Definition des Begriffes Gesten.

bestehen aus den in der Tabelle 5.2 zusammengefassten Größen. Pro Zeitschritt

Merkmal	Berechnungsvorschrift
Bewegungsschwerpunkt	$m_{t,x}=\frac{\sum_{x,y\in R}\lvert\mathcal{D}_t(x,y)\rvert x}{\sum_{x,y\in R}\lvert\mathcal{D}_t(x,y)\rvert}-t_{t,x}$ $m_{t,y}=\frac{\sum_{x,y\in R}\lvert\mathcal{D}_t(x,y)\rvert y}{\sum_{x,y\in R}\lvert\mathcal{D}_t(x,y)\rvert}-t_{t,y}$
Varianz der Bewegung	$\sigma^2_{t,x}=\frac{\sum_{x,y\in R}\lvert\mathcal{D}_t(x,y)\rvert(x-m_{t,x})^2}{\sum_{x,y\in R}\lvert\mathcal{D}_t(x,y)\rvert}$ $\sigma^2_{t,y}=\frac{\sum_{x,y\in R}\lvert\mathcal{D}_t(x,y)\rvert(y-m_{t,y})^2}{\sum_{x,y\in R}\lvert\mathcal{D}_t(x,y)\rvert}$
Änderung des Bewegungsschwerpunktes	$\Delta m_{t,x}=m_{t,x}-m_{t-1,x}$ $\Delta m_{t,y}=m_{t,y}-m_{t-1,y}$
Intensität der Bewegung	$\overline{G}_t=\frac{\sum_{x,y\in R}\lvert\mathcal{D}_t(x,y)\rvert}{\sum_{x,y\in R}1}$

Tabelle 5.2 – Global Motion Merkmale: Physikalische Größen und deren Berechnung.

$t \in \{1,\ldots,T\}$ und für jede Person P_i, $i \in \{1,\ldots,4\}$ wird auf diese Weise für jeden definierten Bereich $R_{t,i}$ ein 7-dimensionaler Vektor

$$\vec{F}_{t,i}(R_{t,i}) = (m_{t,x}, m_{t,y}, \sigma^2_{t,x}, \sigma^2_{t,y}, \Delta m_{t,x}, \Delta m_{t,y}, \overline{G}_t)^T \tag{5.2}$$

extrahiert. Zur Untersuchung der personenspezifischen Aktivitäten wurden zweierlei unterschiedliche Kombinationen von Bereichen definiert, aus denen schließlich die jeweiligen Merkmalsströme

$$\vec{M}_1 = \left(\vec{F}_{1,1}(R_{1,1}),\ldots,\vec{F}_{T,1}(R_{T,1}),\vec{F}_{1,2}(R_{1,2}),\ldots,\vec{F}_{T,4}(R_{T,4})\right) \quad \text{und} \tag{5.3}$$

$$\vec{M}_2 = \left(\vec{F}_{1,1}(R_{1,1,1}),\vec{F}_{1,1}(R_{1,1,2}),\ldots,\vec{F}_{T,4}(R_{T,4,1}),\vec{F}_{T,4}(R_{T,4,2}))\right) \tag{5.4}$$

resultieren. In Abbildung 5.2 sind die der Merkmalsextraktion zugrunde liegenden Bereichskonstellationen veranschaulicht: Während beim Merkmalsset $\vec{M}_1$ für jede beobachtete Person ein einziger Aktionsbereich festgelegt wurde, der Arm-

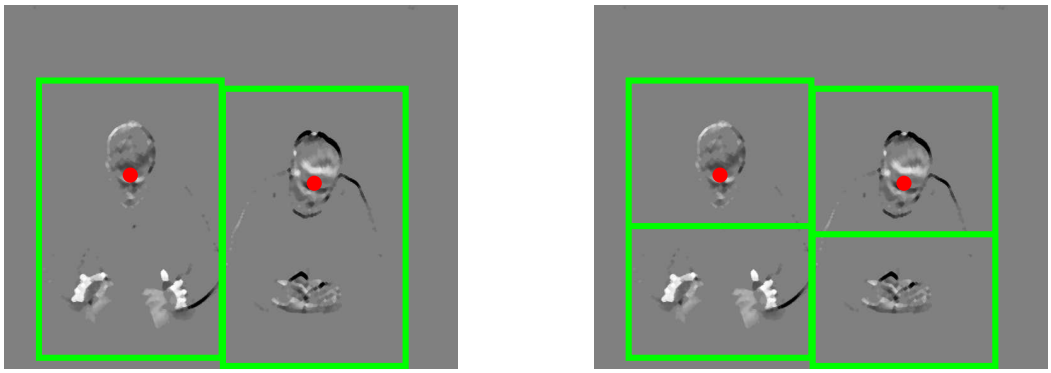


Abbildung 5.2 – Unterschiedliche Definition der für die Merkmalsextraktion betrachteten Aktionsbereiche, repräsentiert durch die grünen Rechtecke: (links) Gesamtheitliche Betrachtung der Person (Merkmalsstrom $\vec{M}_1$), (rechts) separate Betrachtung von Kopf- und Handbewegungen (Merkmalsstrom $\vec{M}_2$). Die roten Punkte deuten jeweils die aus dem Personentracking gewonnene Position des Kopfes der Person an, in deren Abhängigkeit die Aktionsbereiche festgelegt werden.

und Kopfbewegungen ganzheitlich betrachtet und auf nur einen Merkmalsstrom abbildet, wird in Merkmalsset $\vec{M}_2$ eine separate Betrachtung der Arm- und Kopfgesten dadurch vorgenommen, dass der Aktionsbereich relativ zur jeweiligen Kopfposition automatisch in die Bereiche $R_{t,i,1}$ und $R_{t,i,2}$ aufgeteilt wird.

5.3 Merkmalsextraktion und Aufbereitung

Im Idealfall handelt es sich bei den zu erkennenden Gesten um Bewegungsmuster, die von einem möglichst rauscharmen Sensor erfasst werden und in ihrer Gesamtheit beobachtet werden können. Während die erste der beiden Bedingungen maßgeblich durch die Wahl der Sensorik aktiv beeinflusst werden kann, unterliegt die Forderung nach einer beobachtbaren Bewegung nicht mehr unmittelbar dem Einfluss von außen. Eine derartige Störung in der Beobachtung des Bewegungsmusterablaufs erfolgt in der Praxis häufig durch eine teilweise oder gar zeitweise vollständige Verdeckung bedingt durch andere Personen, die sich vor der zu beobachtenden Person aufhalten. Um derartigen Störungen durch geeignete Maßnahmen begegnen zu können, bedarf es einer entsprechenden Modellierung der Merkmalsextraktion sowie der Auswirkung von Störungen auf die generierten Merkmale.

5.3.1 Erzeugung rauschbehafteter Merkmale

Zur Simulation der geschilderten Störquellen wurden die zur Merkmalsextraktion verwendeten Videos künstlichen Manipulationen unterzogen. Um hierbei die Gegebenheiten in realtypischen Szenarien zu berücksichtigen, wurde bei der Untersuchung der Erkennungsleistung bei unterschiedlichen Arten der Manipulation auf vier Situationen eingegangen[2]. Die in der Tabelle 5.3 unter den Bezeichnern OCC_1, OCC_2 und OCC_3 eingeführten Manipulationen repräsentieren jeweils Situationen, in denen sich eine im Kamerabild vor dem zu beobachtenden Sitzungsteilnehmer befindliche Person aufhält und somit eine ganzheitliche Beobachtung des Bewegungsmusters unmöglich macht. Mit Manipulation OCC_4 wird letztlich der Extremfall einer Verdeckung durch eine Person mit ausgebreiteten Armen simuliert. In diesem Fall verbleibt nur etwas weniger als die Hälfte der Bilddaten, um gestenspezifische Merkmale zu extrahieren. Zu beachten ist dabei, dass die daraus resultierenden Merkmale jedoch noch sehr viel weniger Information beinhalten, da gerade der zentrale Bereich, in dem Bewegungen stattfinden, verdeckt ist.

5.3.2 Modell der Merkmalsextraktion

Die geschlossene mathematische Formulierung eines Prozesses, welcher die im Abschnitt 5.2 beschriebene Extraktion der Merkmale exakt modelliert, stellt sich gerade unter dem Aspekt, dass die Auswirkungen von Störungen analysiert werden sollen, als sehr unpraktikabel dar. Aus diesem Grund bedient man sich einer vereinfachenden Betrachtung, bei der sich ein Merkmal $\vec{y}_t$ aus einer linearen Abbildung, der sog. Messmatrix $\underset{\sim}{H}_t$, eines verborgenen Zustandes $\vec{x}_t$ ergibt:

$$\vec{y}_t = \underset{\sim}{H}_t \vec{x}_t \tag{5.5}$$

Hierbei verbirgt sich hinter dem Systemzustand $\vec{x}_t$ gewissermaßen eine abstrahierte Darstellung der Bilddaten. Ebenso wie die Bilddaten eine zeitliche Abhängigkeit aufweisen, kann ein weiterer Prozess angenommen werden, der die zeitliche Interdependenz auf Zustandsebene, überlagert von einem Rauschprozess $\vec{u}_t$, modelliert:

$$\vec{x}_t = \underset{\sim}{A}_t \vec{x}_{t-1} + \vec{u}_t \tag{5.6}$$

Das durch die beiden Gleichungen 5.5 und 5.6 beschriebene System dient in der dargestellten Form als Modell für die Generierung von ungestörten Merkmalen.

[2] Diese Verdeckungen wurden zeitgleich auch von Zobl u. a. [127] benutzt und später auch bei der Analyse von Besprechungsszenarien auf semantisch höherwertiger Ebene zur Erzeugung rauschbehafteter Größen angewandt von Al-Hames u. Rigoll [6, 7].

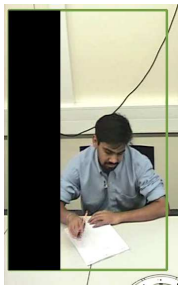
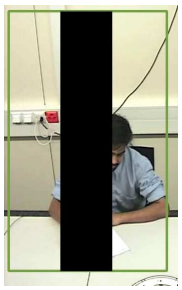
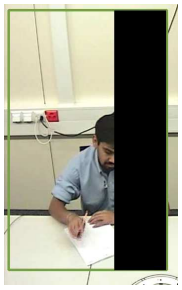
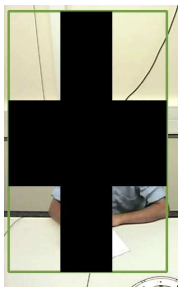
Bezeichner	Art der Manipulation	Beschreibung
OCC_1		Verdeckung des linken Drittels eines Bildes durch geschwärzte Fläche zur Simulation einer am linken Bildrand stehenden Person
OCC_2		Verdeckung des mittleren Drittels eines Bildes durch geschwärzte Fläche zur Simulation einer in Bildmitte stehenden Person
OCC_3		Verdeckung des rechten Drittels eines Bildes durch geschwärzte Fläche zur Simulation einer am rechten Bildrand stehenden Person
OCC_4		Verdeckung des zentralen Ausschnittes eines Bildes durch geschwärzte Fläche zur Simulation einer zentral in Bildmitte stehenden Person mit ausgebreiteten Armen

Tabelle 5.3 – Zusammenstellung der auf die Videodaten angewandten Manipulationen.

Um Störungen bei der Merkmalsextraktion mit einzubeziehen, erfasst man die zusätzlich vorhandenen Störungen durch eine Modifizierung des beschriebenen mathematischen Modells. Um auch hier das Modell möglichst einfach zu halten, wird die Störung der Merkmalsströme als ein gaußverteiltes Rauschsignal $\vec{v}_t$ ausschließlich auf Merkmalsebene angenommen, wodurch sich Gleichung 5.5 abändert zu:

$$\vec{y}_t^* = \underset{\sim}{H}_t \vec{x}_t + \vec{v}_t \tag{5.7}$$

In Blockschaltbild 5.3 ist die Modellierung der Merkmalsextraktion nochmals visualisiert.

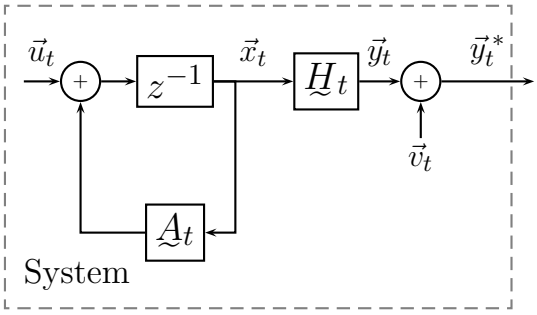


Abbildung 5.3 – Modell der Merkmalsextraktion: Aus den Zuständen $\vec{x}_t$ werden Beobachtungen $\vec{y}_t$ abgeleitet, die durch Beaufschlagung mit einem Störsignal $\vec{v}_t$ zu den rauschbehafteten Ausgangsgrößen $\vec{y}_t^*$ werden.

5.3.3 System der Merkmalsaufbereitung

Ziel einer möglichen Vorverarbeitung der gestörten Merkmalsströme ist es, den eventuell beaufschlagten Rauschanteil im Signal $\vec{y}_t^*$ zu eliminieren bzw. den verborgenen Prozesszustand $\vec{x}_t$ zu schätzen, um die ungestörten Merkmalsströme zu erhalten. Eine Methodik, die geeignet ist, für ein stochastisches lineares dynamisches System (LDS), wie es der gewählten Modellierung zugrunde liegt, aufgrund von u. U. rauschbehafteten Beobachtungen $\vec{y}_t^*$ den Prozesszustand $\vec{x}_t$ zu schätzen, ist der Kalmanfilter[3] (vgl. Kalman [54]). Als eine der zentralen Voraussetzungen benötigt der Kalmanfilter hierfür neben der Beobachtung $\vec{y}_t^*$ ebenso Informationen über das diese Merkmale generierende System, welche über

[3] Auf eine genauere Darstellung des Kalmanfilters sei an dieser Stelle auf den Anhang D verwiesen.

die Messmatrix $\underset{\sim}{H}_t$ und die Systemmatrix $\underset{\sim}{A}_t$ bereit gestellt werden. Um für die konkrete Anwendung die beiden Matrizen genauer spezifizieren zu können, ist vorab eine Definition der Systemzustände $\vec{x}_t$ vonnöten. Wie bereits angedeutet, handelt es sich bei diesen Zuständen um Bilddaten in abstrahierter Form, die direkt bei der Merkmalsextraktion nicht in Erscheinung treten und für deren konkrete Festlegung prinzipiell zahlreiche Freiheitsgrade existieren. Aus Plausibilitätsgründen bietet es sich jedoch an, die Zustände gleichzusetzen mit den ungestörten Beobachtungen und anzureichern um zusätzliches Wissen in Form der zeitlichen Ableitungen der ungestörten Merkmale. Die Zustände für einen beliebigen Zeitschritt t nehmen dann folgende Gestalt an[4]:

$$\vec{x}_t \;=\; \begin{pmatrix} \vec{y}_t \\ \vec{y}_t - \vec{y}_{t-1} \end{pmatrix} \in \mathbb{R}^{14} \tag{5.8}$$

Einhergehend mit der Definition der Zustände wird zugleich die Messmatrix $\underset{\sim}{H}_t$ festgelegt und kann somit als eine sich zeitlich nicht verändernde Größe

$$\underset{\sim}{H} \;=\; \left(\mathbb{1}|\mathbb{Q}\right) \in \mathbb{R}^{7\times 14} \tag{5.9}$$

betrachtet werden. Wird die zur Kalmanfilterung benötigte Systemmatrix $\underset{\sim}{A}_t$ ebenso als über die Zeit konstant angenommen, kann sie, da sie Bestandteil des Systemmodells ist, aus den Trainingsdaten z. B. mittels eines adaptiven linearen Netzwerkes (ADALINE) gemäß Widrow u. Hoff [115] gelernt werden. Dabei werden Paare von Systemzuständen $(\vec{x}_{t-1}, \vec{x}_t)$ benutzt, um die Gewichtsmatrix $\underset{\sim}{W} = (\vec{w}_1, \ldots, \vec{w}_{14})^T$ eines einschichtigen neuronalen Netzes wie in Abbildung 5.4 dargestellt zu lernen. Hierzu wird eine Fehlerfunktion

$$E = \frac{1}{2}\sum_{i=1}^{N_{\text{Bsp}}} ||\vec{e}_i||_2^2 = \frac{1}{2}\sum_{i=1}^{N_{\text{Bsp}}} ||\vec{x}_{t,i} - \underset{\sim}{W}\vec{x}_{t-1,i}||_2^2 \tag{5.10}$$

eingeführt, welche die quadratischen Abstände zwischen den aktuellen und den vorhergehenden Zuständen über alle Trainingsbeispiele N_{Bsp} aufsummiert. Um für diese Fehlerfunktion möglichst zügig das Minimum und somit eine optimale Gewichtsmatrix $\underset{\sim}{W} = \{\vec{w}_1, \ldots, \vec{w}_{14}\}$ zu bestimmen, wird für diese einschichtigen neuronalen Netze das Lernverfahren gemäß der Widrow-Hoff-Regel[5] angewandt.

[4] Die im Folgenden verwendete Notation bezieht sich auf Merkmale aus dem Merkmalsstrom $\vec{M}_1$ (Dimensionalität 7); für Merkmale aus dem Merkmalsstrom $\vec{M}_2$ beträgt die Dimensionalität 14.

[5] Diese Lernregel ist auch unter dem Namen δ-Regel gemäß dem zugrunde liegenden Verfahren bekannt.

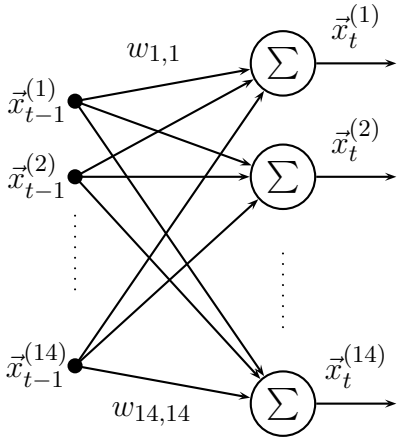


Abbildung 5.4 – Struktur eines adaptiven linearen Netzwerkes.

Hierbei werden in einem iterativen Gradientenabstiegsverfahren sämtliche Gewichte $\vec{w}_j$ mit einer Lernrate α gemäß der Vorschrift

$$\vec{w}_j \leftarrow \vec{w}_j + \alpha(\vec{x}_t^{(j)} - \vec{w}_j^T \vec{x}_{t-1})\vec{x}_{t-1} \tag{5.11}$$

solange angepasst, bis die multidimensionale Fehlerfunktion E ein Minimum erreicht hat. Über die dadurch gelernte Gewichtsmatrix kann nun im Kalmanfilter eine Prädiktion von Zuständen $\vec{x}_{t+1}$ bei gegebenem Zustand $\vec{x}_t$ vorgenommen werden. Aus den verbleibenden Prädiktionsfehlern $\vec{e}_i$ kann abschließend die für den Kalmanfilter notwendige Kovarianzmatrix des Rauschprozesses $\vec{u}_t$ bestimmt werden:

$$\underset{\sim}{\Sigma}_u = \frac{1}{N_{\text{Bsp}}} \sum_{i=1}^{N_{\text{Bsp}}} \vec{e}_i \vec{e}_i^T \tag{5.12}$$

Ebenso kann bei Wissen über die Art der Störung, also bei Vorliegen von zusammengehörigen Paaren von ungestörten $\vec{y}_t$ und gestörten Merkmalen $\vec{y}_t^*$, die Kovarianz des Rauschprozesses $\vec{v}_t$ ermittelt werden zu:

$$\underset{\sim}{\Sigma}_v = \frac{1}{N_{\text{Bsp}}} \sum_{i=1}^{N_{\text{Bsp}}} (\vec{y}_t^* - \vec{y}_t)(\vec{y}_t^* - \vec{y}_t)^T \tag{5.13}$$

Über den durch diese Größen vollständig beschriebenen Kalmanfilter werden, wie in Blockschaltbild 5.5 dargestellt, die rauschbehafteten Merkmale $\vec{y}_t^*$ aufbereitet und störungsfreie Merkmale $\hat{\vec{y}}_t$ geschätzt, welche anschließend zur Erkennung verwendet werden.

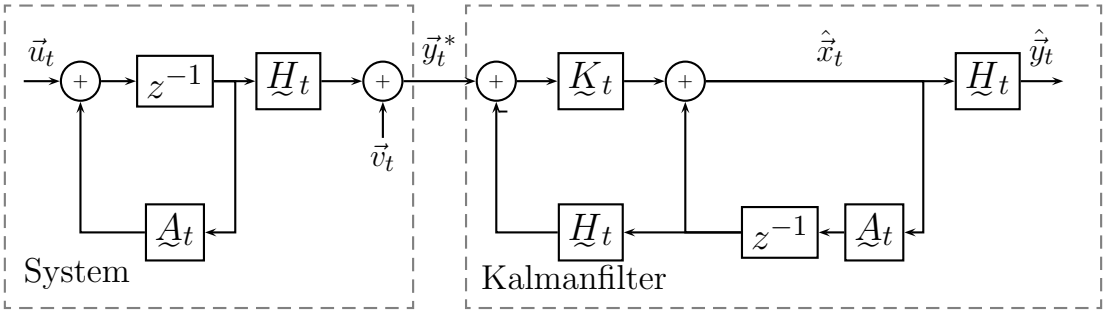


Abbildung 5.5 – Gesamtsystem zur Entstörung der Merkmale: Aus den durch das System generierten Merkmalen $\vec{y}_t^*$ werden durch Kalmanfilterung Schätzwerte für rauschfreie Beobachtungen $\hat{\vec{y}}_t$ geliefert.

5.4 Experimente und Ergebnisse

Im Folgenden werden die zur Erkennung der definierten Basisgesten in Besprechungsszenarien durchgeführten Experimente beschrieben. Als Erkenner fungierte in allen Versuchen eine Struktur bestehend aus 6 Hidden Markov Modellen[6], wobei jedes Modell zur Erkennung einer eigenen Geste (vgl. Abbildung 5.6) trainiert wurde.

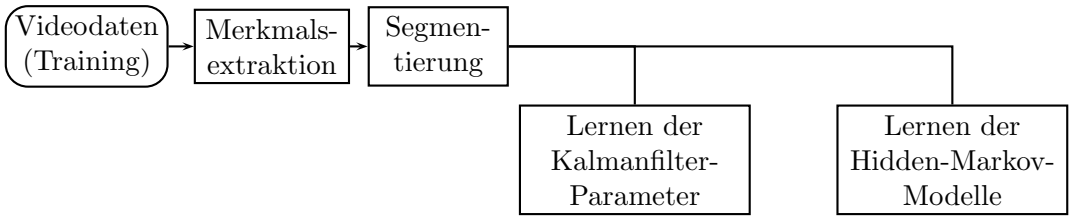


Abbildung 5.6 – Schematisierte Darstellung des Trainingsablaufs.

5.4.1 Erkennung von ungestörten Gesten

Die Basis zur Beurteilung der Güte jeglicher Maßnahmen zur Störungskompensation bildet das originäre filterlose Erkennungssystem. Es wurde daher in einem ersten Experiment ein Referenzsystem, wie in Abbildung 5.7 gezeigt, aufgebaut, um einen Merkmalsstrom $\vec{M}_1$ (ganzheitliche Merkmalsextraktion) bzw. $\vec{M}_2$ (Merkmalsextraktion für Kopf und Arme separat, vgl. Abbildung 5.2) aus einer

[6]Die Grundlagen der Theorie zu den Hidden Markov Modellen sind ausführlich in Anhang C dargelegt.

Videosequenz zu erzeugen, welcher sämtliche von der betrachteten Person ausgeführte Aktionen repräsentiert. Um die einzelnen Aktionen zu identifizieren,

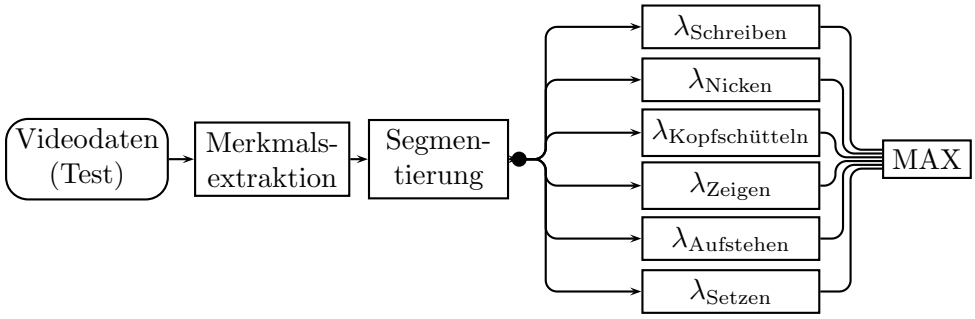


Abbildung 5.7 – Aufbau des Referenzsystems zur Erkennung von Gesten aus Videodaten.

muss der jeweilige Merkmalsstrom vorab zeitlich segmentiert werden. Prinzipiell könnte dies durch Verfahren basierend auf dem Bayes'schen Information Criterion[7] (BIC) (vgl. Schwarz [96]) automatisiert vorgenommen werden. Experimentelle Studien haben jedoch ergeben, dass für die vorliegenden Gesten eine Segmentierung mittels eines solchen Verfahrens aufgrund der sehr kurzen Dauer einzelner Typen von Gesten wie beispielsweise Kopfschütteln oder Nicken nur sehr mäßige Ergebnisse bzgl. der Exaktheit der Segmentgrenzen liefert. Um in den angestellten Untersuchungen den zusätzlichen Einfluss fälschlich erkannter Segmentgrenzen ausschließen zu können, wurden die Merkmalsströme basierend auf manuell annotierten Anfangs- und Endzeitpunkten in die Einzelgesten zerteilt. Diese wurden anschließend unverändert dem Erkenner zugeführt, welcher dann über einen Maximumsentscheid

$$\lambda^* = \max_{\lambda_j} p(\vec{M}_i|\lambda_j) \tag{5.14}$$

die Zuordnung eines Merkmalsmusters auf ein bestimmtes Modell λ^* trifft und damit die Geste entsprechend klassifiziert.
Ziel eines ersten Experimentes ist es, einerseits eine als Referenz dienende Erkennungsrate auf ungestörten Daten zu erhalten sowie andererseits die vorgestellten Merkmalsextraktionsverfahren $\vec{M}_1$ und $\vec{M}_2$ gegenüberzustellen. Wie in Tabelle 5.4 zusammengefasst, ergab sich hierbei für unterschiedliche Kombinationen

[7] Benannt nach Gideon E. Schwarz ist dieses Kriterium ebenso unter dem Namen Schwarz Information Criterion (SIC) bekannt.

von Zahl der Zustände und verwendeten Gaußmixturen in den kontinuierlichen HMM nahezu durchgehend eine um wenige Prozentpunkte schlechtere Erkennungsleistung bei Extraktion der Merkmale über das Verfahren, welches eine separate Betrachtung von Kopfregion und Tischregion durchführt. Dies dürfte

Zustände	Zahl der verwendeten Gaußmixturen									
	2		4		6		8		10	
	$\vec{M}_1$	$\vec{M}_2$	$\vec{M}_1$	$\vec{M}_2$	$\vec{M}_1$	$\vec{M}_2$	$\vec{M}_1$	$\vec{M}_2$	$\vec{M}_1$	$\vec{M}_2$
3	67,0	63,0	74,4	67,3	79,6	73,8	78,5	73,6	78,8	74,7
4	75,8	71,3	79,6	72,9	79,8	73,2	80,2	75,0	80,9	74,3
5	77,6	71,8	76,6	74,8	78,5	75,0	78,8	75,8	79,3	75,6
6	77,4	70,5	80,3	73,6	81,4	72,5	81,7	74,8	**82,3**	77,8
7	78,4	71,9	81,4	73,2	**82,3**	73,0	81,1	78,3	81,0	79,0

Tabelle 5.4 – Gegenüberstellung der für die beiden Merkmalsextraktionsverfahren $\vec{M}_1$ und $\vec{M}_2$ erzielten Erkennungsergebnisse in % auf jeweils ungestörten Videosequenzen bei unterschiedlicher Parametrierung der Hidden Markov Modelle.

auf zweierlei Tatsachen zurückzuführen sein:

a) Da die Unterteilung notwendigerweise relativ zur lokalisierten Kopfposition vorgenommen wird, diese aber selbst für unbewegte Köpfe immer einen – wenn auch geringen – Rauschanteil aufweist, wird ebenso die Position der Trennlinie einem Rauschprozess unterworfen. Hierdurch werden gerade diejenigen Pixel im Differenzbild $\underset{\sim}{D}_t$, welche sich sehr nahe an dieser Trennlinie befinden, in oftmaligem Wechsel zum oberen bzw. unteren Ausschnitt der Region gewertet, was sich in sämtlichen Global Motion Größen durch eine sehr große Variation äußert.

b) Desweiteren dürfte bei Verfahren $\vec{M}_1$ ein glättender Effekt zum Tragen kommen: Am Beispiel der Schreibgeste (siehe Abbildung 5.8) zeigt sich, dass es mitunter von einzelnen Personen unterschiedliche Ausführungsformen der Gesten gibt, bei denen einmal nur die Hand selbst, bei anderen der gesamte Arm während des Schreibens bewegt wird. Während bei Verfahren $\vec{M}_1$ die Bewegung des Armes sich in nur geringem Maße in einer Änderung der Global Motion Größen niederschlägt, wirkt sich dies bei einer Unterteilung des Aktionsbereiches sehr viel stärker sowohl in der Verschiebung des Bewegungsschwerpunktes als auch in einer größeren Varianz der Bewegung aus. Die globale Berechnungsvorschrift von Methodik $\vec{M}_1$ wirkt daher wie

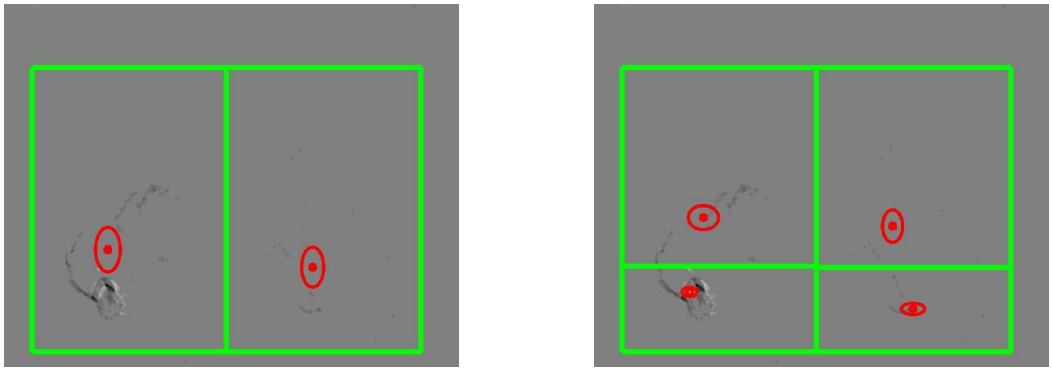


Abbildung 5.8 – Im Bild links werden Bewegungen der Person ganzheitlich durch nur einen Aktionsbereich erfasst: Obwohl die Person links außen bei der Schreibgeste den gesamten Arm und die neben ihm sitzende Person nur die Hand bewegt, sind der durch die rote Ellipse angedeutete Schwerpunkt und die Varianz in einem ähnlichen Wertebereich. Im Bild rechts hingegen werden Kopf- und Armbewegungen separiert betrachtet: Sowohl die einzelnen Global Motion Merkmale als auch die relative Position zwischen den Schwerpunkten von Arm- und Kopfbewegung differieren hierbei deutlich und führen letztlich zu einem größeren Varianzbereich, der in der Modellierung abgedeckt werden muss.

ein Filter, der diese Art von Rauschen aus den Merkmalen eliminiert, wohingegen die beschriebenen Effekte bei Verfahren $\vec{M}_2$ unmittelbar auf die Merkmale negativ einwirken.

Aufgrund der beobachteten, günstigeren Eignung des Merkmalsextraktionsverfahrens $\vec{M}_1$ wurden die folgenden Experimente bzgl. der Erkennung verdeckter Gesten nurmehr basierend auf diesen Merkmalen vorgenommen. Vergleicht man die beiden Konfusionsmatrizen (siehe Tabellen 5.5, 5.6) derjenigen HMM-Parametrierungen, für die sich ein Maximum in der Erkennungsleistung ergibt, so stellt man fest, dass bei 7 Zuständen und 6 Gaußmixturen die Gesten gleichmäßig gut erkannt werden, während für die HMM-Modellierung mit 6 Zuständen und 10 Gaußmixturen die gute Erkennungsleistung auf Kosten der beiden Gesten „Sich setzen“ und „Kopfschütteln“ geht. Der Grund hierfür liegt weniger in der Zahl der im HMM verwendeten Zustände, sondern vielmehr in der beschränkten Zahl an Vorkommnissen dieser beiden Gestentypen in den Trainingsdaten, weswegen die Parameter der Gaußkurven bei steigender Zahl von Mixturen nur noch schlecht geschätzt werden können und dadurch die Erkennungsleistung dieser selteneren Gesten sinkt. Weil daher generell für das vorliegende Datenmaterial eher

eine geringere Zahl an Gaußmixturen vernünftig erscheint, werden die weiteren Untersuchungen ausschließlich für HMM-Modelle mit 7 Zuständen angestellt.

	Schreiben	Setzen	Aufstehen	Nicken	Kopf-schütteln	Zeigen	
Schreiben	436	0	0	42	14	16	85,8
Setzen	2	1	2	1	0	3	11,1
Aufstehen	2	0	8	0	0	2	66,7
Nicken	6	0	0	196	30	4	83,0
Kopfschütteln	3	0	0	19	10	2	29,4
Zeigen	1	0	0	1	0	47	95,9

Tabelle 5.5 – Konfusionsmatrix für ein HMM mit 6 Zuständen und 10 Gaußmixturen bei Verwendung des Merkmalsextraktionsverfahrens $\vec{M}_1$.

	Schreiben	Setzen	Aufstehen	Nicken	Kopf-schütteln	Zeigen	
Schreiben	435	0	0	37	14	22	85,6
Setzen	1	5	2	0	0	1	55,6
Aufstehen	0	0	9	2	0	1	75,0
Nicken	7	0	2	189	33	5	80,1
Kopfschütteln	3	0	0	16	13	2	38,2
Zeigen	0	0	0	2	0	47	95,9

Tabelle 5.6 – Konfusionsmatrix für ein HMM mit 7 Zuständen und 6 Gaußmixturen bei Verwendung des Merkmalsextraktionsverfahrens $\vec{M}_1$.

5.4.2 Erkennung von rauschbehafteten Gesten

Im Gegensatz zum letzten Abschnitt werden in den folgenden Experimenten die Gesten nicht mehr unmittelbar aus den ursprünglichen Bilddaten extrahiert, sondern basierend auf den veränderten Bildinhalten, welche aus den in Tabelle 5.3 gezeigten Manipulationen resultieren.
Um die Auswirkungen der unterschiedlichen Manipulationen auf die Erkennungsleistung abschätzen zu können, werden die ungefilterten Merkmale auf das im vorigen Abschnitt benutzte Erkennersystem gegeben. Wie der Überblick in Tabelle 5.7 zeigt, sinken die Erkennungsraten für die betrachteten Manipulationen OCC_1, OCC_2 und OCC_4 ab, während sich für OCC_3 sogar eine tendenzielle Verbesserung der Erkennungsleistung im Vergleich zu den ungestörten Daten ergibt. Diese Verbesserung basiert mitunter auch darauf, dass die überwiegende

Mehrheit der Personen in den Besprechungsszenarien Rechtshänder ist und Bewegungen der linken Hand somit eher als Störquelle interpretiert werden können. Durch die Manipulation OCC_3 werden gerade diese Bewegungsanteile ausgeblendet, wodurch sich offensichtlich die generelle Qualität der Merkmale erhöht, was seinerseits wiederum zu einer Steigerung der Erkennungsleistung führt.

Mixturen	OCC_1	OCC_2	OCC_3	OCC_4
2	79,8	65,0	80,8	55,5
4	80,4	65,8	80,7	49,2
6	79,6	66,8	82,6	56,0

Tabelle 5.7 – Erkennungsleistung auf verrauschten Merkmalen für ein HMM-System ohne Einsatz eines Filters zur Rauschunterdrückung.

Globaler Kalmanfilter Um die durch die Manipulationen verursachten Störungen in den Merkmalen zu kompensieren, wurde in das bestehende System eine Kalmanfilterung integriert. Die für den Kalmanfilter benötigten Parameter wurden hierfür unmittelbar aus der Gesamtheit der ungestörten Merkmale der Trainingsdaten, also ohne Unterscheidung der jeweils vorliegenden Geste, bestimmt, woraus schließlich ein globaler Filter resultiert. Dieser Vorgehensweise liegt die Annahme zugrunde, dass jeder Gestentyp gleichermaßen von der Manipulation betroffen ist und sich dies in den Merkmalen auf ein- und dieselbe Art bemerkbar macht. In Abbildung 5.9 ist das betrachtete System zur Stö-

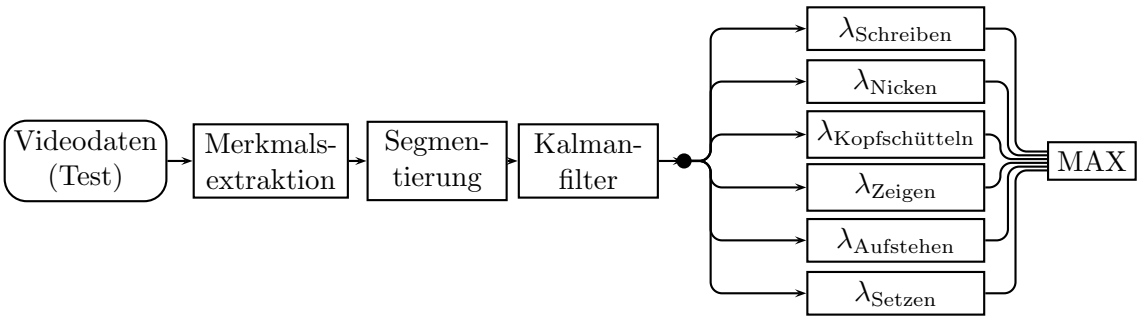


Abbildung 5.9 – Blockstruktur des Erkennungsprozesses basierend auf einer globalen Filterung.

rungskompensation als Blockschaltbild dargestellt: Basierend auf den aus den rauschbehafteten Daten extrahierten und segmentierten Merkmalen $\vec{y}_t$ werden

durch den Kalmanfilter Schätzwerte für die ungestörten Merkmale errechnet, die dann äquivalent zum Ausgangssystem durch einen Maximumsentscheid klassifiziert werden.
Aus Tabelle 5.8 ist ersichtlich, dass durch den Einsatz eines globalen Filters zur Störungskompensation mit Ausnahme von Manipulation OCC_1 sowie OCC_3[8] für alle untersuchten Störungsszenarien eine Verbesserung der Erkennungsleistung erzielt werden kann. Entsprechend der bereits im vorigen Abschnitt angedeuteten Vermutung, dass Bewegungen der linken Hand sich in den Merkmalen vielmehr als Störung denn als Nutzsignal auswirken, kann für Manipulation OCC_3 auch die Annahme, dass sich die Manipulation auf allen Gesten gleichermaßen in einer Veränderung der Merkmale niederschlägt, nur sehr bedingt aufrecht erhalten werden.

Mixturen	OCC_1	OCC_2	OCC_3	OCC_4
2	74,4	72,6	71,1	60,0
4	77,1	71,9	73,4	62,0
6	74,7	71,3	74,4	62,5

Tabelle 5.8 – Erkennungsleistung auf verrauschten Merkmalen für ein HMM-System bei Verwendung eines einzigen (globalen) Kalmanfilters zur Störungskompensation.

Spezifischer Kalmanfilter Da ein Beweis für die Zulässigkeit der für den Einsatz eines globalen Filters getroffenen Annahme unmittelbar nur sehr schwer zu erbringen ist und, wie gesehen, für einzelne Manipulationen nicht gehalten werden kann, wurde in einem weiteren Experiment für jede Geste unabhängig ein eigener Kalmanfilter aus ungestörten Merkmalen erzeugt. Dies wird wesentlich auf der Theorie begründet, dass die Merkmale jeder Geste durch die auf den Videodaten erfolgte Manipulation unterschiedlich variiert werden, demzufolge also unterschiedliche Störprozesse zugrunde liegen. Die darauf spezialisierten Kalmanfilter wurden in einem neuen Systemaufbau (vgl. Blockschaltbild 5.10) dann den einzelnen Hidden Markov Modellen vorgeschaltet, wodurch der jeweils anliegende Merkmalsstrom unabhängig von der tatsächlichen Geste mittels jeweils eines gestenspezifischen Kompensators gefiltert wird. Hierdurch wird erreicht, dass jeder Merkmalsstrom gleichermaßen in Richtung der sechs unterschiedlichen Gesten optimiert wird, wobei eine tatsächliche Störungskompensation selbstverständlich nur bei demjenigen Filter zu tragen kommt, welcher

[8] Bei den beiden genannten Manipulationen wirkt sich die Störung mithin nur geringfügig auf die Erkennungsrate aus.

mit der durch die Merkmalssequenz beschriebenen Geste korrespondiert. Die

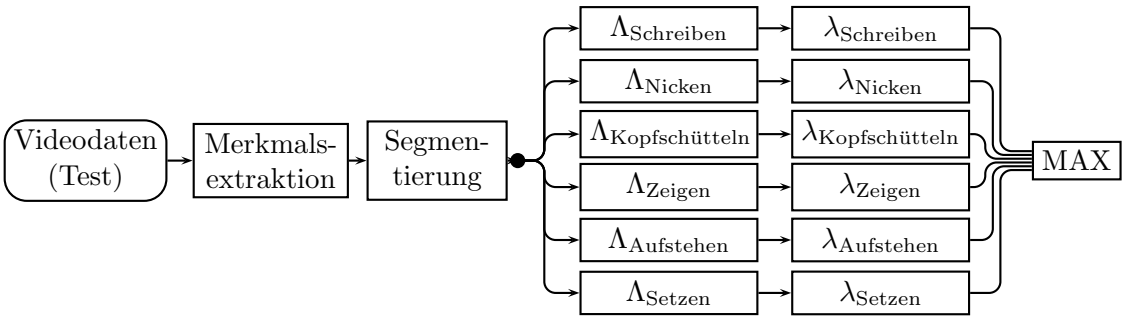


Abbildung 5.10 – Blockstruktur des Erkennungssystems basierend auf einer gestenspezifischen Vorfilterung des Merkmalsstroms.

Evaluierung dieses Systemaufbaus lieferte die in Tabelle 5.9 zusammengefassten Ergebnisse. Gegenüber der globalen Filterung erweist sich der Einsatz gesten-

Mixturen	OCC_1	OCC_2	OCC_3	OCC_4
2	81,3	70,8	74,2	64,3
4	83,5	68,4	74,4	64,9
6	82,2	69,1	75,9	65,5

Tabelle 5.9 – Erkennungsleistung auf verrauschten Merkmalen für ein HMM-System bei Einsatz gesten-spezifischer Kalmanfilter zur Störungskompensation.

spezifisch trainierter Kalmanfilter als durchwegs vorteilhaft, was sich an einer absoluten Steigerung der durchschnittlichen Erkennungsraten im Bereich 1 % -3 % zeigt. Dennoch bleiben auch bei der gestenspezifischen Filterung die für Manipulation OCC_3 erzielten Erkennungsraten deutlich hinter denen zurück, welche bei Verzicht auf jegliche Filterung erreicht werden konnten. Dies korrespondiert offensichtlich mit der eingangs unterstellten Vermutung, dass die Merkmale durch Bewegungen der linken Hand gestört werden, und untermauert die Feststellung, dass dieses Störsignal anstatt es mittels eines Kalmanfilters zu rekonstruieren besser unterdrückt wird, wie es durch Anwendung der Manipulation geschieht.

Kapitel 6

Zusammenfassung

Die vorliegende Arbeit beschäftigte sich mit der Entwicklung einer neuartigen Architektur zur robusten und stabilen Verfolgung von Personen in Innenraum-Umgebungen. Hauptaugenmerk lag hierbei im Wesentlichen auf der grundlegenden Weiterentwicklung gängiger bottom-up bzw. top-down Trackingtechnologien durch die physiologisch motivierte Betrachtung der Bild-/Szenenanalyse und hierbei insbesondere der Personenverfolgung als hybriden, d. h. sowohl datenwie auch gleichzeitig hypothesengetriebenen, Prozess. In diesem Zusammenhang entstand eine gesamtheitliche Architektur, mit der auch das simultane Verfolgen mehrerer Personen in einer Videosequenz ermöglicht wird. An einem Anwendungsbeispiel wurde anschließend die extrahierte Information über die Position von Personen genutzt, um verschiedene Basisgesten der Besprechungsteilnehmer zu erkennen.

6.1 Hybride Trackingarchitektur

Beginnend mit der Analyse gängiger Techniken zur Personenverfolgung wurde zunächst ein allgemeines Grundmodell des Trackingprozesses erläutert und ausgehend vom Stand der Technik die einzelnen Funktionseinheiten näher beleuchtet. Speziell für die Personendetektion, der im Zuge der hybriden Umsetzung eine entscheidende Rolle zukommt, wurden hierbei mit der Haar-Wavelet basierten Methodik nach Viola u. Jones [110] sowie mit dem von Rowley u. a. [86] veröffentlichten Ansatz auf einem Neuronalen Netz fußend zwei der etablierten und als sehr leistungsfähig geltenden Verfahren vorgestellt. Prinzipbedingt lassen sich diese beiden Methoden jedoch nicht unmittelbar im Rahmen der in dieser Arbeit verfolgten Idee des hybriden und omnidirektionalen Trackingprozesses nutzen, so dass hierfür andere Modellierungskonzepte für die Personendetektion notwendig waren. Aus diesem Grund kamen in der entwickelten Architektur Ellipsen- (vgl. Birchfield [17]) sowie statistische Formmodelle, basierend auf dem Prinzip von

Cootes u. a. [27], in ihrer Wirkweise jedoch modifiziert, zum Einsatz. Obwohl diese beiden Modellierungsansätze zwar generell als alleinstehende Personendetektoren in Einzelbildern nur mäßige Leistungen insbesondere im Vergleich zu den beiden erstgenannten Detektionsverfahren zeigen, so erweisen sie sich für die Nutzung in einem videobasierten und hypothesengesteuerten System aufgrund ihres datengetriebenen Konvergenzprinzips als besonders prädestiniert für die Umsetzung in einem hybriden Gesamtkonzept. Mittels dreier verschiedener Architekturen, die das Spektrum von rein daten- bzw. hypothesengetrieben sowie dem in dieser Arbeit verfolgten hybriden Konzept abdeckten, wurde in einer ersten Evaluierung die Leistungsfähigkeit von fünf unterschiedlichen Ansätzen auf sechs Videosequenzen, in denen jeweils nur eine einzige Person agierte, bewertet. Ermöglicht durch die zahlreichen Fehlermaße des zugrunde gelegten Evaluierungsschemas konnte eine tiefgehende Analyse der einzelnen Ansätze erfolgen. Diese legte die prinzipiellen Schwachpunkte der einzelnen evaluierten Architekturkonzepte offen und untermauerte – wie an den Ergebnissen ersichtlich – die Vorteile einer hybriden Betrachtungsweise des Trackingproblems.
Bedingt durch das im Zuge der hybriden Architektur angewandte Samplingprinzip des Partikelfilters ist eine simultane Verfolgung mehrerer Personen a-priori nicht möglich, da die Partikel meist innerhalb weniger Zeitschritte auf einer Person konvergieren. Aus diesem Grund wurde in einem zweiten Schritt die entworfene Architektur dahingehend erweitert, dass durch kontrollierte Steuerung der Partikel ein Konvergieren auf nur einer Person ausbleibt. Hierzu wurde im Gegensatz zu anderen Verfahren aus der Literatur versucht, durch eine bewusste Trennung zwischen der Hypothesenallokation auf einzelne Objekte und der Detektionsaufgabe eine rechenzeiteffiziente Lösung des Trackingproblems zu entwickeln. Ein erstes Konzept bildete dazu eine hierarchisch angeordnete Struktur zweier Partikelfilter. Wie anhand der Evaluation ersichtlich wurde, kann eine solche Architektur zwar ansprechende Ergebnisse liefern, die strikte und konstante Zuordnung von Hypothesen auf jeweils ein Objekt kann damit jedoch nicht durchwegs gewährleistet werden. Aus diesem Grund wurde ein weiteres Architekturkonzept entwickelt, welches die probabilistische Abtaststrategie des Partikelfilters kombiniert mit einem übergeordneten heuristischen Nachbarschaftssuchverfahren zur Allokation der einzelnen Partikelfilter. Die hierfür durchgeführte Evaluation konnte zeigen, dass diese Architektur – reduziert auf einen top-down Ansatz – Trackingergebnisse zu liefern vermag, die tendenziell gerade noch mit denen einer hierarchischen Partikelfilterstruktur verglichen werden können. Erst als hybride Architektur offenbart diese Form der Architektur eine sehr herausragende Performanz, die sich in allen Evaluationsmaßen positiv niederschlägt und

unter den getesteten Ansätzen klar als das System mit den besten Ergebnissen hervorgeht.

6.2 Gestenerkennung

Daneben wurde aufsetzend auf den Ergebnissen der Personenverfolgung ein System zur Erkennung von personenspezifischen Aktionen vorgestellt, wobei der zentrale Aspekt darauf lag, die durch Verdeckungen typischerweise in zu analysierenden Szenarien auftauchenden Störungen geeignet zu kompensieren und so eine – möglichst unabhängig von der konkreten Art der Verdeckung – störunempfindliche Gestenerkennung zu realisieren.
In unterschiedlichen synthetischen Szenarien wurden hierbei konkret vier verschiedene Verdeckungstypen simuliert, die zu einem signifikanten Abfall der Erkennungsleistung eines auf ungestörten Daten trainierten Systems führten. Ziel der Arbeit im Hinblick auf die Gestenerkennung war es nun, ein System zu entwickeln, mit dem eine sinnvolle und robuste Erkennung auch auf verrauschten Daten wieder möglich wird. Hierzu wurden die durch Verdeckungen verursachten Störungen im Bild bzw. vielmehr deren Auswirkung auf die Merkmalsextraktion zunächst in einem LDS modelliert und somit einer Kalmanfilterung zugänglich gemacht, wodurch letztlich die originalen, unverrauschten Merkmale geschätzt werden konnten. Gerade die Abbildung der physikalischen Merkmalsextraktion auf Modellebene konnte hierbei grundsätzlich auf zwei unterschiedliche Arten ausgestaltet werden: Eine erste These ging von einer einheitlichen Auswirkung der Störungen auf den Prozess der Merkmalsextraktion aus, weswegen der Rauschunterdrückung lediglich ein einziges Erzeugendensystem zugrunde gelegt wurde. Wie jedoch die Evaluierung zeigte, ließen sich hierdurch nur marginal verbesserte und insgesamt sehr uneinheitlich zu interpretierende Erkennungsergebnisse generieren. Aus diesem Grund wurde in einer zweiten These für jede zu untersuchende Geste ein eigenes Erzeugendensystem angenommen und demzufolge der Einfluss der Störquelle auf die Merkmale gestenspezifisch behandelt. In einer weiteren Evaluierung konnte diese Annahme weiter gestützt werden, da sich hier eine durchschnittliche relative Performanzsteigerung abhängig von der Art der Störquelle zwischen 3,0 % und 21,2 % gegenüber der Erkennung von verrauschten Daten ohne jegliche Störungskompensation eingestellt hat. Lediglich für einen einzelnen der simulierten Verdeckungstypen sank die Erkennungsrate um ca. 8,0 % ab, was insbesondere damit begründet werden konnte, dass diese Art der Störung offensichtlich in den Merkmalen bereits originär vorhandenes Rauschen kompensiert. Der Versuch einer nachgeschalteten Korrektur der Merk-

male durch die entwickelte Kalmanfilterstruktur fügt hier somit tendenziell eher einen neuerlichen Rauschanteil hinzu, der sich in einer reduzierten Erkennungsleistung niederschlägt.
Da offensichtlich eine generelle Störungskompensation unter diesem Aspekt suboptimal erscheint, könnten weiterführende Arbeiten durch eine Erkennung von Störungen bzw. der konkreten Art der Störungen die Qualität einer Gestenerkennung in natürlichen Szenarien weiter verbessern.

6.3 Weitere Anwendungsgebiete

Die Personenverfolgung in Videodaten stellt für eine Vielzahl von Aufgaben im Bereich der videogestützten Informationsverarbeitung eine grundlegende Voraussetzung dar. Mit der Thematik „Gestenerkennung“ wurde im Rahmen dieser Arbeit bereits ein Anwendungsgebiet für die von der entwickelten Architektur zur Personenverfolgung gelieferten Positionsinformationen aufgezeigt. Aufgrund der durch die datenadaptive Art der Personenmodellierung mittels Active Shapes sehr präzise erfassten Objektinformationen eignen sich diese Daten ebenso hervorragend zur Identifikation von Personen. Hierzu wurden bereits konzep-

Abbildung 6.1 – Beispielhafte Bilder der zur Personenidentifikation verwendeten Büroszenarien. Um derartige Szenarien realistisch nachzubilden, wurde bewusst auf hochwertige und somit teure Kameraausstattung verzichtet, weswegen sich die Aufnahmequalität der Sequenzen nur als mäßig bewerten läßt. Daneben ist auch die starke Schlagschattenbildung, die sich als wesentlicher Störfaktor bei der Klassifikation bemerkbar macht, im Gesicht erkennbar.

tionelle Experimente im Verbund mit der beschriebenen Architektur für Büroszenarien, welche gekennzeichnet waren von schlechter Aufnahmequalität und starken Beleuchtungsschwankungen (vgl. Abbildung 6.1), gestartet, deren Ziel die Untersuchung einer Identifikation von Personen in Rundumansicht (inklusive Hinterköpfe) war.
In einem ersten Entwurf (vgl. Schreiber u. a. [92]) wurde hierzu die von der

ASM-basierten Trackingarchitektur gelieferte Objektkontur als Initialschätzung für sog. *Active Appearance Modelle* (AAM)[1] genutzt, um über eine Modellierung der Textur innerhalb der Kontur Merkmale zu generieren. Bei dieser Art der Modellierung wird neben der im Zuge der ASM beschriebenen Modellierung der Objektkontur auch die darin enthaltene Textur in äquivalenter Weise erfasst und kann durch einen Gewichtungsvektor entsprechend modifiziert werden. Der sich nach der datengetriebenen Adaption einstellende Gewichtungsvektor dient unmittelbar als Merkmal, welches in einem einschichtigen NN klassifiziert wurde. Hierbei konnte das System selbständig abhängig von der Güte der Modelladaption ermitteln, für welche Merkmale eine Klassifikation voraussichtlich erfolgversprechend ist, wodurch sich für die herausfordernde Art der Daten Erkennungsraten von beispielsweise 92,5 % ergaben, wenn im Mittel ca. 2,5 Bilder pro Sekunde der Klassifizierung zugeführt wurden.
Ein zweites Konzept (vgl. Schreiber u. a. [93]) ging über den ersten Ansatz hinaus, indem eine aufwendige Datenaufbereitung mittels des von Pizer u. a. [75] veröffentlichten CLAHE-Verfahrens zur Kompensation der Beleuchtungseffekte im Gesicht der eigentlichen Erkennungsstufe vorgeschaltet wurde. Die anschließend für den durch die ASM beschriebenen Bildausschnitt generierten DCTmod2-Merkmale (vgl. Sanderson u. Paliwal [87]) wurden anschließend in einem zyklischen HMM trainiert, welches wie in Abbildung 6.2 dargestellt mittels einer Rundumansicht der jeweiligen Person initialisiert wurde. Die Evaluierung

Abbildung 6.2 – Rundumansicht einer Person aus dem Büroszenario, wie sie zur Initialisierung der zyklischen HMM-Struktur verwendet wurde.

dieses Systems konnte zeigen, dass eine Reklassifikation auf den zum Training benutzten Daten in nahezu 100 % der Bilder erfolgreich war und somit die Identifikation einer Person auch anhand des Hinterkopfes offensichtlich prinzipiell vorgenommen werden kann. Auf unbekannte Daten angewendet zeigte sich, dass

[1] AAM benötigen für eine qualitativ hochwertige Adaption an die zugrunde liegenden Bilddaten eine sehr präzise Startschätzung, wofür die durch das ASM ermittelte Kontur prädestiniert ist.

– erwartungsgemäß – die Erkennung frontaler Ansichten die besten Ergebnisse lieferte, zum Profil hin abfiel und sich für Hinterkopfansichten wieder leicht verbesserte.
Gerade für die hypothesengesteuerte sowie die hybride Personenverfolgung kann die tatsächliche Identität einer Person maßgeblich dazu beitragen, bei gegenseitigen Verdeckungen mittels des Kontextwissens Partikel entscheidend zu beeinflussen, so dass objektspezifisch beispielsweise durch eine individuelle Kopfform auch in Situationen mit teilweiser Verdeckung die Personenverfolgung durch bessere Messwerte stabilisiert wird. Dies könnte damit den Anstoss für weitere Arbeiten im Bereich der Integration der Personenidentifikation direkt in eine hybride Trackingarchitektur bilden.

Anhang A

Abkürzungen

AAM .... Active Appearance Modell
AMI ..... Augmented Multi-party Interaction
AMIDA .. Augmented multi-party Interaction with distance access
ASM ..... Active Shape Modell
BIC ..... Bayes'sches Information Criterion
CD ..... Configuration Distance (Konfigurationskompaktheit)
CHIL .... Computers in the Human Interaction Loop
CSCW ... Computer Supported Collaborative Work
EM ..... Expectation Maximization
FIT ..... False identifying tracker
FIO ..... False identified object
FN ..... False positive
FP ..... False negative
GMM .... Gauß-Mixtur-Modell
GPA ..... Generalisierte Prokrustes Analyse
HMM .... Hidden Markov Modell
IDIAP ... Institute Dalle Molle d'Intelligence Artificielle Perceptive
LDS ..... Lineares dynamisches System
M4 ..... MultiModal Meeting Manager
MO ..... Multiple tracker
MT ..... Multiple object
NN ..... Neuronales Netz
NIST .... National Institute of Standards and Technology
OT ..... Object tracking
PCA ..... Principal component analysis (Hauptachsentransformation)
ROC ..... Receiver operator characteristic
RSAT .... Rotated summed area table
SAT ..... Summed area table

Anhang B

Formelzeichen

$\underset{\sim}{\mathbb{Q}}$	Nullmatrix
$\underset{\sim}{\mathbb{1}}$	Einheitsmatrix
a_{ij}	Zustandsübergangswahrscheinlichkeit
$\vec{a}$	Betragsvektor
α	Lernrate
$\alpha_t(j)$	Vorwärtswahrscheinlichkeit
$\underset{\sim}{A}$	Systemmatrix
b_i	Ausgabedichte
$\beta_t(j)$	Rückwärtswahrscheinlichkeit
$\vec{b}$	Gewichtungsvektor für Active Shape Modelle
$\underset{\sim}{B}(\vec{p})$	Tensormatrix für den Bildpunkt $\vec{p}$
d	Dimensionalität
D_{M}	Manhattan-Distanz
D_{E}	Euklid'sche Distanz
D_{I}	Distanz auf Schnittmengenbetrachtung basierend
D_{K}	Kullback-Leibler-Distanz
D_{B}	Bhattacharyya-Distanz
$\underset{\sim}{D}$	Differenzbild
η	Richtungswinkel
ϵ	Klassifizierungsfehler
E	Kante eines Graphen
$f_j(x, y, s)$	Wert des Haar-ähnlichen Merkmales j an Position (x, y) mit Skalierung s
$\vec{\Phi}$	Phasenvektor
$F_{t,ij}$	F-Bewertung zweier Objekte
$\vec{F}$	Global Motion Merkmalsvektor
$\vec{g}$	Gradient
G	Grauwert (Intensität)
$\overline{G}$	Mittlere Intensität

$\hat{G}$	Geschätzter Grauwert
$\underset{\sim}{G}$	Grauwertbild
$\underset{\sim}{G}^*$	Ausschnitt aus einem Grauwertbild
$\underset{\sim}{G}_x, \underset{\sim}{G}_y$	In x- bzw. y-Richtung gefiltertes Grauwertbild
$\underset{\sim}{G}_{\mathrm{Kum}}$	Matrix mit den kumulierten Intensitäten
$\underset{\sim}{G}_{\mathrm{SAT}}$	Integralbild für horizontal und vertikal ausgerichtete Haar-ähnliche Merkmale
$\underset{\sim}{G}_{\mathrm{RSAT}}$	Integralbild für diagonal ausgerichtete Haar-ähnliche Merkmale
$\vec{h}_t$	Partikel (Objekteigenschaften) zum Zeitpunkt t
$\vec{H}$	Histogramm
$\vec{H}^*$	Normiertes Histogramm
$\underset{\sim}{H}$	Messmatrix
$\vec{I}$	Farbtupel/-tripel
$\vec{I}(\vec{p})$	Farbtupel/-tripel an Position $\vec{p} = (x, y)^T$ des Bildes $\underset{\sim}{I}$
$\underset{\sim}{I}$	Farbbild
$\mathcal{I}$	Menge an Einzelbildern
$\vec{J}$	Gabor-Jet
κ	Proportionalitätskonstante
k_j	Schwacher Klassifikator
$\vec{k}$	Wellenvektor
K	Starker Klassifikator
$\underset{\sim}{K}_t$	Kalman Gain
λ	Modellparameter eines Hidden Markov Modells
$\underset{\sim}{\Lambda}$	Übergangswahrscheinlichkeitsmatrix
$L(G)$	Energieäquivalent eines Graphen G
$\vec{\mu}, \underset{\sim}{\mu}$	Mittelwert
m	Median
m_x	Schwerpunkt der Bewegung in x-Richtung
Δm_x	Änderung des Schwerpunkts der Bewegung in x-Richtung
m_y	Schwerpunkt der Bewegung in y-Richtung
Δm_y	Änderung des Schwerpunkt der Bewegung in y-Richtung
$\vec{M}_1$	Merkmalsstrom bei gesamtheitlicher Betrachtung einer Person zur Berechnung der Global Motion Merkmale
$\vec{M}_2$	Merkmalsstrom bei separierter Betrachtung einer Person zur Berechnung der Global Motion Merkmale
$\vec{n}$	Normalenvektor
N_{B}	Zahl an Bildern
N_{Bsp}	Zahl an Beispielen

N_{Event}	Zahl an Ereignissen (Kanten)
N_{Iter}	Zahl an Iterationen
N_{Kask}	Zahl an Kaskadenstufen
N_{Mix}	Zahl an Gaußmixturen
$N_{\text{Mix}_{\text{eff}}}$	Effektiv genutzte Zahl an Gaußmixturen
$N_{\text{neg}}(k)$	Zahl an negativen Beispielen der Klasse k
N_{Objekte}	Zahl der an einem Ereignis beteiligten Objekte
N_{Pix}	Zahl an Pixel
N_{Pkt}	Zahl der Punkte
$N_{\text{pos}}(k)$	Zahl an positiven Beispielen der Klasse k
N_{Refobj}	Zahl unterschiedlicher Referenzobjekte in einer Videosequenz
N_{S}	Zahl an Hypothesen
$N_{t,\mathcal{O}}$	Zahl an Referenzobjekten zum Zeitpunkt t
$N_{t,\mathcal{T}}$	Zahl an Tracks zum Zeitpunkt t
N_{Tracks}	Zahl unterschiedlicher Tracks in einer Videosequenz
$\mathcal{N}(\vec{\mu}, \underset{\sim}{\Sigma})$	Normalverteilung mit Mittelwert $\vec{\mu}$ und Kovarianzmatrix $\underset{\sim}{\Sigma}$
$\mathcal{N}$	Menge der Negativbeispiele
$\vec{O}_t$	Referenzobjekt (Objekt aus der Menge $\mathcal{O}_t$)
$\mathcal{O}_t$	Menge an Referenzobjekten zum Zeitpunkt t
Ω_{Ell}	Bewertungsfunktion zur lokalen Optimierung des Ellipsenmodells
Ω_{ASM_1}	Bewertungsfunktion zur lokalen Optimierung des Active Shape Modells basierend auf Gradienten
Ω_{ASM_2}	Bewertungsfunktion zur lokalen Optimierung des Active Shape Modells basierend auf Gabor-Wavelets
$p(x)$	Wahrscheinlichkeit für Auftreten des Wertes x
p_{TP}	Detektionsrate
p_{FP}	Falsch-positiv Rate
$\vec{p}$	Position/Pixel
$\hat{\vec{p}}$	Geschätzte Position
$\mathcal{P}$	Menge der Positivbeispiele
$\vec{P}$	Geordnete Punktemenge (transformationsbehaftet)
$\vec{P}^*$	Geordnete Punktemenge (transformationsfrei)
$\vec{P}'$	Gemittelte Punktmenge
$P_{t,ij}$	Genauigkeit eines Referenzobjektes $O_{t,i}$ in Bezug auf den Track $T_{t,j}$
q_t	Zustand zum Zeitpunkt t
$\vec{Q}$	Zustandsfolge
$R_{t,ij}$	Vollständigkeit eines Tracks $T_{t,j}$ in Bezug auf das Referenzobjektes $O_{t,i}$

σ^2	Varianz
σ_x^2	Varianz der Bewegung in x-Richtung
σ_y^2	Varianz der Bewegung in y-Richtung
ς^2	Parameter für Kalmanfilter
S_i	Zustand eines Markovprozesses
$S(\vec{J}, \vec{J'})$	Ähnlichkeitsfunktion zwischen zwei Jets $\vec{J}$ und $\vec{J'}$
$\underset{\sim}{\Sigma}$	Kovarianzmatrix
s	Skalierung
$\mathcal{S}$	Menge an Partikeln
t	Zeitpunkt oder Bildnummer einer Videosequenz
$\vec{t}$	Translation
$\vec{T}_t$	Track (Objekt aus der Menge $\mathcal{T}_t$)
$\mathcal{T}_t$	Von einem Tracking-Algorithmus zum Zeitpunkt t ermittelte Menge an Objekten
θ	Rotation
Θ	Schwellwertparameter
$\vec{u}_t$	Normalverteiltes Systemrauschen
$\vec{v}_t$	Normalverteiltes Messrauschen
E	Knoten eines Graphen
$\mathcal{V}$	Validierungsmenge
w_i	Gewicht der i-ten Komponente
$\underset{\sim}{W}$	Gewichtsmatrix
$\vec{x}_t$	Systemzustand
ξ	Identitätsbezeichner für einen Track
$\vec{y}_t$	Beobachtung zum Zeitpunkt t
$\underset{\sim}{Y}$	Observationen
ψ	Gabor-Wavelet
$\vec{\psi}$	Eigenvektor
$\underset{\sim}{\Psi}$	Eigenvektorenmatrix

Anhang C

Theorie der eindimensionalen Hidden Markov Modelle

Die Klassifizierung extrahierter Merkmale stellt ein typisches Problem der Mustererkennung dar, welches sich mit der maschinellen Erkennung und Auswertung von dynamischen Mustern in Signalen beschäftigt. Als ein sehr mächtiges Werkzeug zur Lösung von dynamischen Mustererkennungsaufgaben hat sich für zahlreiche Anwendungsgebiete unterschiedlichster Disziplinen, die von der Sprach-/Sprechererkennung (vgl. Campbell [21]) sowie der Handschrifterkennung (vgl. Cole u. a. [25]) über die Gestenerkennung und die Gesichtserkennung (vgl. Mitra [70], Zhao u. a. [124]), die Genomanalyse (vgl. Pedersen u. Hein [72]) bis hin zur Zeitreihenanalyse in der Finanzmathematik (vgl. Mamon u. Elliott [65]) reichen, der Einsatz von *Hidden Markov Modellen* (HMM) bewährt.
Das Prinzip dieser Methodik ist es, anhand von verschiedenen Merkmalssequenzen[1] $\underset{\sim}{Y}$ einer Klasse k durch ein Training klassentypische Modelle λ_k auf Basis von bestimmten in den Daten auftauchenden Mustern zu erstellen. Für eine unbekannte Merkmalssequenz $\underset{\sim}{Y} = (\vec{y}_1, \ldots, \vec{y}_T)$ der Länge T wird anschließend in einem Klassifizierungsschritt jenes Modell λ_{k^*} bestimmt, für welches die Wahrscheinlichkeit $p(\underset{\sim}{Y}|\lambda_{k^*})$ maximal wird und von welchem somit die Observation am wahrscheinlichsten emittiert wurde.

C.1 Doppelt stochastische Prozesse

Grundlage für die Hidden Markov Modelle bildet die Theorie der diskreten Markov-Prozesse. Bei diesen Prozessen handelt es sich um stochastische Systeme, welche die Eigenschaft aufweisen, dass ein Vorliegen des Zustands S_i zum diskreten Zeitpunkt t, repräsentiert durch den Zustand q_t, nur abhängig von dessen unmittelbar vorherig eingenommenen Zustand S_j und somit unabhängig von

[1] Ebenso gebräuchlich ist die Bezeichnung Observationen.

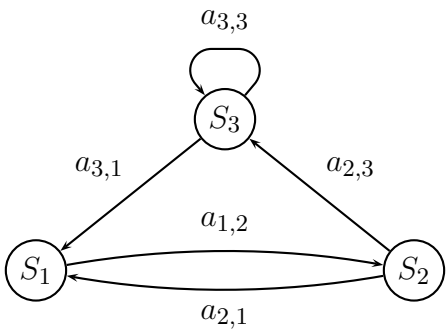


Abbildung C.1 – Exemplarische Markov-Kette mit drei Zuständen und den auftretenden Übergangswahrscheinlichkeiten $a_{i,j}$.

der weiteren Vergangenheit des Prozesses ist[2]:

$$p(q_t = S_j | q_{t-1} = S_i, q_{t-2} = S_h, \dots) = p(q_t = S_j | q_{t-1} = S_i) \tag{C.1}$$

Markov-Ketten wie in Abbildung C.1 stellen eine mögliche Realisierung solcher Markov-Prozesse dar. Bei der für die Praxis relevanten Gruppe der endlichen Markov-Ketten können insgesamt N verschiedene Zustände eingenommen werden. Hierbei findet zu jedem Zeitschritt ein Übergang statt, der in einen vom aktuellen Zustand S_i erreichbaren Zustand S_j mündet. Dieser Übergang ist Resultat eines stochastischen Prozesses und wird bestimmt durch die in Gleichung C.1 beschriebene Übergangswahrscheinlichkeit

$$a_{i,j} = p(q_t = S_j | q_{t-1} = S_i), \qquad \text{mit } a_{i,j} > 0 \text{ und } \sum_{j=1}^{N} a_{i,j} = 1 \tag{C.2}$$

Fasst man sämtliche Übergangswahrscheinlichkeiten zu einer Matrix

$$\underset{\sim}{\Lambda} = \begin{pmatrix} a_{1,1} & \dots & a_{1,N} \\ \vdots & \ddots & \vdots \\ a_{N,1} & \dots & a_{N,N} \end{pmatrix} \tag{C.3}$$

zusammen, so ist die Struktur, d.h. die Anzahl der Zustände und die möglichen Zustandswechsel, durch diese Matrix eindeutig definiert. Gemeinsam mit dem Vektor $\vec{\pi}$, der die Wahrscheinlichkeiten

$$\pi^{(i)} = p(q_1 = S_i) \qquad \text{mit } \sum_{i=1}^{N} \pi^{(i)} = 1 \tag{C.4}$$

[2]Diese Eigenschaft wird gemeinhin auch als Markov Eigenschaft bezeichnet.

beinhaltet, die Markov-Kette im Zustand S_i zu beginnen, kann ein stationärer Markov-Prozess durch ein Zweitupel $\lambda = (\vec{\pi}, \underset{\sim}{\Lambda})$ vollständig beschrieben werden. Bei dieser Art der Modellierung wird jedes beobachtbare Symbol auf genau einen Zustand abgebildet, wodurch bei gegebener Observation $\underset{\sim}{Y}$ direkt auf die Zustandsfolge $\vec{Q} = (q_1, \ldots, q_T)^T$ geschlossen werden kann. Daher läßt sich die Wahrscheinlichkeit, dass eine gegebene Observation $\underset{\sim}{Y}$ durch ein Modell λ generiert worden ist, durch folgende Gleichung ermitteln:

$$p(\underset{\sim}{Y}|\lambda) = p(\vec{Q}|\lambda) = p(q_1) \prod_{t=2}^{T} p(q_t|q_{t-1}) \tag{C.5}$$

Sind die Zustände der Markov-Kette nicht unmittelbar beobachtbar[3], sondern nur die im einem beliebigen Zustand S_i emittierte Observation $\vec{y}_t$, die selbst wiederum Resultat eines (weiteren) stochastischen Prozesses ist und mit einer bestimmten Ausgabedichte

$$b_i(\vec{y}_t) = p(\vec{y}_t|q_t = S_i) \tag{C.6}$$

emittiert wird, so spricht man von einem Hidden Markov Modell. Der aktuelle Zustand legt nun im Gegensatz zur Betrachtungsweise bei den Markov-Ketten nicht mehr die Observation selbst, sondern vielmehr die Emissionsdichte für die eigentliche Observation fest. Zur eindeutigen Beschreibung eines solchen doppelt stochastischen Prozesses muss nun das von der Markov-Kette bekannte Zweitupel erweitert werden um eine zusätzliche Information, nämlich den Ausgabedichten $b_i(\vec{y}_t)$ in den einzelnen Zuständen S_i, welche zu einer Matrix $\underset{\sim}{B}$ zusammengefasst werden. Demnach wird ein HMM durch ein Dreitupel $\lambda = (\vec{\pi}, \underset{\sim}{\Lambda}, \underset{\sim}{B})$ eindeutig festgelegt.
Für die in der Praxis anzutreffenden Problemstellungen handelt es sich bei den Observationen häufig um reellwertige Vektoren $\vec{y}_t \in \mathbb{R}^d$ der Dimension d. Zur Behandlung solcher Observationen wird in der Literatur (z. B. Rabiner [79]) zwischen den kontinuierlichen und den diskreten HMM unterschieden.

C.1.1 Kontinuierliche HMM

Die reellwertigen Ausgabevektorsequenzen können durch ein HMM mit kontinuierlichen Emissionsdichten erzeugt werden. Modelle dieser Art werden gewöhnlich auch als kontinuierliche HMM bezeichnet. Zur Modellierung der Emissionsdichte eines Zustands S_i wird dabei häufig auf eine kontinuierliche d-dimensionale

[3] Man bezeichnet die Markov-Kette in diesem Fall auch als versteckt (engl. „*hidden*“).

Normalverteilung[4]

$$\mathcal{N}(\vec{y}, \vec{\mu}_i, \underset{\sim}{\Sigma}_i) = \frac{1}{\sqrt{(2\pi)^d |\underset{\sim}{\Sigma}_i|}} \exp\left(-\frac{1}{2}(\vec{y} - \vec{\mu}_i)^T \underset{\sim}{\Sigma}_i^{-1} (\vec{y} - \vec{\mu}_i)\right) \tag{C.7}$$

mit Mittelwert $\vec{\mu}_i$ und Kovarianzmatrix $\underset{\sim}{\Sigma}_i$ zurückgegriffen. Durch eine gewichtete Überlagerung von N_{Mix} Normalverteilungen lassen sich schließlich auch beliebige Zusammenhänge in den Daten darstellen. Die Ausgabewahrscheinlichkeit läßt sich dann berechnen zu

$$b_i(\vec{y}) = \sum_{k=1}^{N_{\text{Mix}}} w_{i,k} \mathcal{N}(\vec{y}, \vec{\mu}_{i,k}, \underset{\sim}{\Sigma}_{i,k}), \tag{C.8}$$

wobei $w_{i,k}$ das jeweilige Gewicht angibt, mit dem die k-te Normalverteilung in die Summe eingeht, und den Stochastizitätsbedingungen

$$w_{i,k} \geq 0 \text{ und } \sum_{k=1}^{N_{\text{Mix}}} w_{i,k} = 1 \tag{C.9}$$

genügen muss.

C.1.2 Diskrete HMM

Sollen die Emissionsdichten hingegen durch diskrete Funktionen beschrieben werden, so können die Zustände nurmehr diskrete Observationen aus einem Alphabet $\vec{v} = (v_1, ..., v_M)^T$ erzeugen. Hierfür ist es notwendig, die Observationen anhand eines Codebuchs mit M Einträgen zu quantisieren. Dieses Codebuch kann beispielsweise über eine k-Means Vektorquantisierung aus den Beispielobservationen erzeugt werden. Die Ausgabewahrscheinlichkeit für eine Observation v_m im Zustand S_i ergibt sich dann zu

$$b_i(v_m) = p(v_m | q_t = S_i) \tag{C.10}$$

Diese lassen sich kompakt in einer Symbolemissionswahrscheinlichkeitsmatrix

$$\underset{\sim}{B} = \begin{pmatrix} b_1(v_1) & \dots & b_N(v_1) \\ \vdots & \ddots & \vdots \\ b_1(v_m) & \dots & b_N(v_m) \end{pmatrix} \tag{C.11}$$

darstellen.

[4] Auch bekannt unter der Bezeichnung Gaußverteilung.

C.2 Training eines HMM

Aufgabe des Trainings der klassenspezifischen Modelle ist es, das im vorigen Abschnitt vorgestellte charakteristische Parametertupel $\lambda_k = (\vec{\pi}_k, \underline{A}_k, \underline{B}_k)$ mit Hilfe mehrerer exemplarischer Mustersequenzen $\underset{\sim}{Y}$ der Klasse k so zu bestimmen, dass die Wahrscheinlichkeit für die Generierung der Beobachtungen $\underset{\sim}{Y}$ durch das Modell λ_k maximiert wird[5]. Da sich die Parameter mathematisch nicht in geschlossener Form ermitteln lassen, muss zur Berechnung ein iterativer Optimierungsansatz gewählt werden. Für das Training von HMM hat sich hier als sehr effizientes Verfahren der Expectation-Maximization Algorithmus (vgl. Dempster u. a. [30]), der auch unter dem Namen Baum-Welch Algorithmus (vgl. Baum u. Petrie [10], Baum u. a. [11]) bekannt ist, etabliert. Ausgehend von einer Initialschätzung für die Modellparameter wird bei diesem Algorithmus abwechselnd ein neues Parametertupel $\bar{\lambda}$ für den nächsten Iterationsschritt geschätzt und anschließend die Differenz zwischen der logarithmierten Produktionswahrscheinlichkeit des alten und des neuen Modells maximiert. Zur praktischen Durchführung dieser Maximierung wurden von Rabiner [79] folgende Gleichungen formuliert, die in der Publikation von Bilmes [15] ausführlich hergeleitet werden:

$$\begin{aligned} \bar{\vec{\pi}}(i) &= \text{Erwartete Häufigkeit des Prozessbeginns im Zustand } S_i = \\ &= p(q_1 = S_i|\underset{\sim}{Y}, \lambda) \end{aligned} \tag{C.12}$$

$$\begin{aligned} \bar{a}_{i,j} &= \frac{\text{Zahl der erwarteten Übergänge von Zustand } S_i \text{ nach } S_j}{\text{Zahl der erwarteten Sprünge aus } S_i} = \\ &= \frac{\sum_{t=1}^{T-1} p(q_t = S_i, q_{t+1} = S_j|\underset{\sim}{Y}, \lambda)}{\sum_{t=1}^{T-1} p(q_t = S_i|\underset{\sim}{Y}, \lambda)} = \frac{\sum_{t}^{T-1} p(q_t = S_i, q_{t+1} = S_j, \underset{\sim}{Y}|\lambda)}{\sum_{t}^{T-1} p(q_t = S_i, \underset{\sim}{Y}|\lambda)} \end{aligned} \tag{C.13}$$

$$\begin{aligned} \bar{b}_i(v_k) &= \frac{\text{Zahl der erwarteten Aufenthalte im Zustand } S_i \text{ und Beobachtung von } v_k}{\text{Zahl der Aufenthalte in } S_i} = \\ &= \frac{\sum_{t=1,\vec{y}_t=v_k}^{T} p(q_t = S_i|\underset{\sim}{Y}, \lambda)}{\sum_{t=1}^{T} p(q_t = S_i|\underset{\sim}{Y}, \lambda)} = \frac{\sum_{t=1,\vec{y}_t=v_k}^{T} p(q_t = S_i, \underset{\sim}{Y}|\lambda)}{\sum_{t=1}^{T} p(q_t = S_i, \underset{\sim}{Y}|\lambda)} \end{aligned} \tag{C.14}$$

Eine direkte Berechnung der Parameter anhand dieser Gleichungen würde exponentiell mit der Länge T der Observation $\underset{\sim}{Y}$ ansteigen und wäre somit in der Praxis nicht durchführbar. Durch Umformulieren der Gleichungen läßt sich

[5] Man spricht in diesem Zusammenhang aufgrund der Maximierungsvorschrift auch von Maximum Likelihood (ML) Schätzung.

dieser Umstand beseitigen, wodurch eine Berechnung mit einer Komplexität proportional zur Länge T möglich wird:

$$\begin{aligned} p(q_t = S_i, \underset{\sim}{Y}|\lambda) &= p(\vec{y}_{1:t}|q_t = S_i, \lambda)p(\vec{y}_{t+1:T}|q_t = S_i, \lambda)p(q_t = S_i|\lambda) = \\ &= p(\vec{y}_{1:t}, q_t = S_i|\lambda)p(\vec{y}_{t+1:T}|q_t = S_i, \lambda) \end{aligned} \tag{C.15}$$

$$\begin{aligned} p(q_t = S_i, q_{t+1} = S_j, \underset{\sim}{Y}|\lambda) &= p(\vec{y}_{1:t}|q_t = S_i, \lambda)p(\vec{y}_{t+1:T}|q_{t+1} = S_j, \lambda) \\ &\quad p(q_t = S_i, q_{t+1} = S_j|\lambda) = \\ &= p(\vec{y}_{1:t}, q_t = S_i|\lambda)p(\vec{y}_{t+1}|q_{t+1} = S_j, \lambda) \\ &\quad p(\vec{y}_{t+2:T}|q_{t+1} = S_j, \lambda)p(q_{t+1} = S_j|q_t = S_i, \lambda) \end{aligned} \tag{C.16}$$

Führt man für die Ausdrücke $p(\vec{y}_{1:t}, q_t = S_i|\lambda)$ bzw. $p(\vec{y}_{t+1:T}|q_t = S_i, \lambda)$ die Vorwärtswahrscheinlichkeit $\alpha_t(i)$ bzw. die Rückwärtswahrscheinlichkeit $\beta_t(i)$ ein, so gehen die Gleichungen C.15 und C.16 über in

$$\begin{aligned} p(q_t = S_i, \underset{\sim}{Y}|\lambda) &= \alpha_t(i)\beta_t(i) \\ p(q_t = S_i, q_{t+1} = S_j, \underset{\sim}{Y}|\lambda) &= \alpha_t(i)b_j(\vec{y}_{t+1})\beta_{t+1}(j)a_{i,j} \end{aligned} \tag{C.17}$$

Der Vorteil in der nunmehr gewählten Darstellung besteht darin, dass sich ein Training von HMM sehr effizient implementieren läßt, da sowohl die Vorwärts- als auch die Rückwärtswahrscheinlichkeit jeweils rekursiv berechnet werden kann mittels

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i)a_{i,j}\right] b_j(\vec{y}_{t+1}) \tag{C.18}$$

$$\beta_t(i) = \sum_{j=1}^{N} a_{i,j}b_j(\vec{y}_{t+1})\beta_{t+1}(j), \tag{C.19}$$

wobei gilt:

$$\alpha_1(i) = \vec{\pi}_i b_i(\vec{y}_1) \quad \forall i \in \{1, \ldots, N\} \tag{C.20}$$

$$\beta_T(i) = 1 \quad \forall i \in \{1, \ldots, N\} \tag{C.21}$$

C.3 Klassifikation mittels HMM

Nachdem für jede der insgesamt k Klassen ein separates Modell λ_k erstellt wurde, besteht die Aufgabe der Klassifizierung wie eingangs bereits erwähnt darin, dasjenige Modell zu finden, für welches die Produktionswahrscheinlichkeit, eine

Observation $\underset{\sim}{Y}$ zu erzeugen, maximiert wird. Dies kann mathematisch folgendermaßen formuliert werden:

$$\lambda^* = \underset{k}{\operatorname{argmax}}\, p(\underset{\sim}{Y}|\lambda_k) = \underset{k}{\operatorname{argmax}} \sum_{\forall q \in \vec{Q}} p(\underset{\sim}{Y}, q|\lambda_k), \tag{C.22}$$

wobei $\vec{Q}$ die Menge aller zulässigen Zustandsabfolgen durch das Modell bezeichnet. Ebenso wie beim Training der HMM ist eine direkte Berechnung wiederum aufgrund des mit der Länge der Observation exponentiell ansteigenden Aufwandes praktisch nicht handhabbar. Jedoch bietet sich auch hier durch die im vorigen Abschnitt eingeführte Vorwärtswahrscheinlichkeit $\alpha_t(i)$ die Möglichkeit, sehr effizient durch nur einmalige Berechnung eines jeden Teilpfades die Produktionswahrscheinlichkeit zu bestimmen:

$$p(\underset{\sim}{Y}|\lambda) = \sum_{i=1}^{N} \alpha_T(i) \tag{C.23}$$

Durch diesen sog. Vorwärtsalgorithmus läßt sich die Anzahl der nötigen Rechenschritte von $2TN^T$ auf nurmehr N^2T Operationen reduzieren.

C.4 Viterbi-Algorithmus

Häufig ist in praktischen Anwendungen nur die wahrscheinlichste Zustandsabfolge[6] $\hat{\vec{Q}}$ unter allen möglichen Sequenzen $\vec{Q}$ von Interesse, für die gilt, dass bei gegebener Beobachtungssequenz $\underset{\sim}{Y}$ für ein Modell λ die Wahrscheinlichkeit $p(\underset{\sim}{Y}, \vec{Q}|\lambda)$ maximiert wird. Zur Bestimmung dieser Zustandssequenz $\hat{\vec{Q}}$ kommt der sog. Viterbi-Algorithmus (vgl. Viterbi [111]) zum Einsatz. Anstatt der vorherig eingeführten Vorwärtswahrscheinlichkeiten ermittelt dieser Algorithmus zu jedem Zeitpunkt t die jeweils maximal erzielbaren Wahrscheinlichkeiten

$$\gamma_t(i) = \max_{Q} p(q_{1:t-1}, q_t = S_i, \vec{y}_{1:t}|\lambda) \tag{C.24}$$

über sämtliche Zustandsfolgen $\vec{Q}$, die im Zustand S_i enden. Ersetzt man in Gleichung C.18 die Summation durch den Maximierungsoperator, so ergibt sich analog zu den Vorwärtswahrscheinlichkeiten wiederum die Möglichkeit, über eine rekursive Berechnungsvorschrift die Wahrscheinlichkeiten $\gamma_t(j)$ zu ermitteln:

$$\gamma_{t+1}(j) = \max_{1<i\leq N} \gamma_t(i) a_{i,j} b_j(\vec{y}_{t+1}) \quad \text{mit} \tag{C.25}$$

$$\gamma_1(i) = \vec{\pi}_i b_i(\vec{y}_1) \tag{C.26}$$

[6] Diese Zustandsabfolge wird auch als Viterbi-Pfad bezeichnet.

Damit ergibt sich die Produktionswahrscheinlichkeit $p(\underset{\sim}{Y}|\lambda)$ durch den Viterbi-Algorithmus zu

$$p(\underset{\sim}{Y}, \hat{\vec{Q}}|\lambda) = \max_{1<j\leq N} \gamma_T(j). \tag{C.27}$$

Um letztlich auch die wahrscheinlichste Zustandsfolge $\hat{\vec{Q}}$ selbst ermitteln zu können, werden in einer Matrix $\underset{\sim}{\psi_t}$ für jeden Zeitpunkt t jeweils diejenigen Vorgängerzustände S_i erfasst, die im Zustand S_j die Wahrscheinlichkeit $\gamma_t(j)$ maximieren:

$$\psi_{t+1}(j) = \operatorname*{argmax}_{1<i\leq N} \gamma_t(i) a_{i,j} \tag{C.28}$$

Der wahrscheinlichste Zustand $\hat{q}_t$ zu einem Zeitpunkt t läßt sich somit dann unmittelbar aus der Matrix $\underset{\sim}{\psi}$ ablesen:

$$\hat{q}_t = \psi_{t+1}(\hat{q}_{t+1}) \quad \text{mit} \tag{C.29}$$

$$\hat{q}_T = \operatorname*{argmax}_{1<i\leq N} \gamma_T(i). \tag{C.30}$$

Anhang D

Theorie des Kalmanfilters

Im Jahre 1960 veröffentlichte der ungarisch-amerikanische Mathematiker Rudolf Emil Kalman einen Algorithmus (vgl. Kalman [54]) zur Lösung des Wiener Problems auf Basis eines rekursiven Verfahrens. Die zugrunde liegende Problemstel-

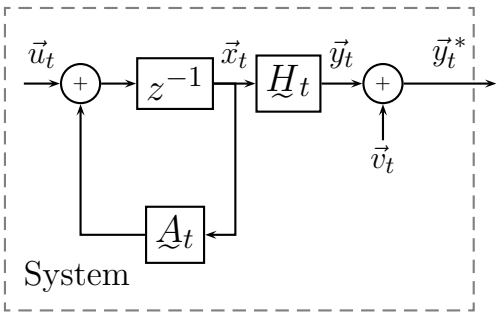


Abbildung D.1 – Modell eines linearen dynamischen stochastischen Prozesses.

lung lautete dabei, für ein dynamisches stochastisches Modell (siehe Abbildung D.1) der Form

$$\vec{x}_{t+1} = \underset{\sim}{A}_t \vec{x}_t + \vec{u}_t \tag{D.1}$$

$$\vec{y}_t^* = \underset{\sim}{H}_t \vec{x}_t + \vec{v}_t \tag{D.2}$$

mit den beiden gaußverteilten Störgrößen

$$\vec{u}_t \sim \mathcal{N}(0, \underset{\sim}{\Sigma}_u) \tag{D.3}$$

$$\vec{v}_t \sim \mathcal{N}(0, \underset{\sim}{\Sigma}_v) \tag{D.4}$$

einen Schätzwert $\hat{\vec{x}}_t$ für das Signal $\vec{x}_t$ bei gegebener Beobachtung des Signals $\underset{\sim}{Y} = (\vec{y}_1, \ldots, \vec{y}_t)$ zu bestimmen, so dass der Erwartungswert des Fehlers

$$\mathrm{E}\{\vec{e}_t^T \vec{e}_t\} = \mathrm{E}\{(\vec{x}_t - \hat{\vec{x}}_t)^T (\vec{x}_t - \hat{\vec{x}}_t)\} \tag{D.5}$$

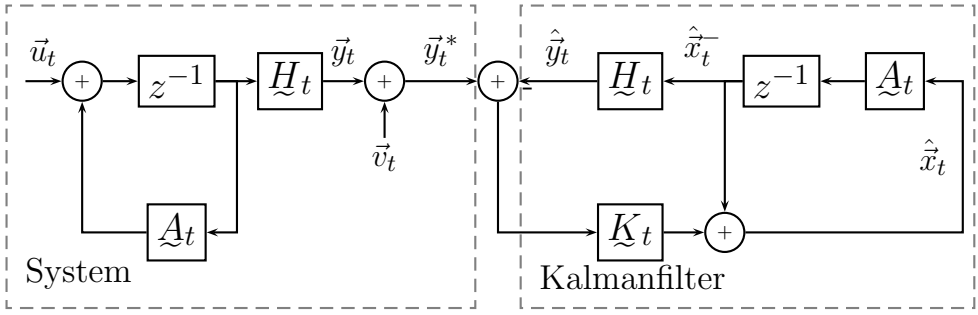


Abbildung D.2 – Modell zur Schätzung von unverrauschten Zustandsgrößen $\hat{\vec{x}}_t$.

minimal wird. Ein einfaches Modell zur Erzeugung dieses Schätzwertes würde darin bestehen, das durch die Gleichungen D.1 und D.2 beschriebene System zu verwenden, jedoch ohne die beiden unbekannten Störsignale $\vec{u}_t$ sowie $\vec{v}_t$ mit einzubeziehen:

$$\hat{\vec{x}}_{t+1} = \underset{\sim}{A}_t \hat{\vec{x}}_t \tag{D.6}$$

$$\hat{\vec{y}}_t = \underset{\sim}{H}_t \hat{\vec{x}}_t \tag{D.7}$$

Um dem System zusätzliche Information, welche mit dem beobachtbaren Signal $\underset{\sim}{Y}$ zur Verfügung steht, zu übergeben, wird das zum aktuellen Zeitschritt k vorliegende Residuum zwischen dem Signal $\vec{y}_t$ und dem Schätzwert $\hat{\vec{y}}_t$ über eine lineare Abbildung $\underset{\sim}{K}_t$, dem sog. Kalman Gain, in das System (siehe graphische Darstellung in D.2) eingekoppelt, wodurch sich folgende Filtergleichungen ergeben:

$$\hat{\vec{x}}_t^- = \underset{\sim}{A}_t \hat{\vec{x}}_{t-1} \tag{D.8}$$

$$\hat{\vec{x}}_t = \hat{\vec{x}}_t^- + \underset{\sim}{K}_t(\vec{y}_t^* - \hat{\vec{y}}_t) \tag{D.9}$$

$$\hat{\vec{y}}_t = \underset{\sim}{H}_t \hat{\vec{x}}_t^- \tag{D.10}$$

Bei Wissen über das die Beobachtungen $\underset{\sim}{Y}$ generierende System, welches durch die beiden Matrizen $\underset{\sim}{A}_t$ und $\underset{\sim}{H}_t$ vollständig spezifiziert ist, stellt der Kalman Gain $\underset{\sim}{K}_t$ den einzigen variablen Parameter des Filters dar. Ziel des Kalmanfilters ist es, die Fehlervarianz $\underset{\sim}{P}_t = \mathrm{E}\{\vec{e}_t \vec{e}_t^T\}$ durch geeignete Wahl der Kalman Gain Matrix $\underset{\sim}{K}_t$ zu minimieren. Die Einträge dieser Matrix können somit durch die partielle Ableitung der erwarteten Fehlervarianz $\underset{\sim}{P}_t$ nach den Einträgen in der Matrix $\underset{\sim}{K}_t$ bestimmt werden zu:

$$\frac{\partial \underset{\sim}{P}_t}{\partial \underset{\sim}{K}_t} = \frac{\partial \mathrm{E}\{(\vec{x}_t - \hat{\vec{x}}_t)(\vec{x}_t - \hat{\vec{x}}_t)^T\}}{\partial \underset{\sim}{K}_t} = 0$$

$$\underset{\sim}{K}_t = \underset{\sim}{P}_t^- \underset{\sim}{H}_t^T (\underset{\sim}{H}_t \underset{\sim}{P}_t^- \underset{\sim}{H}_t^T + \underset{\sim}{\Sigma}_v)^{-1} \tag{D.11}$$

Für die durch Gleichung D.11 gegebene Bestimmungsform des Kalman Gains wird Wissen über die zu erwartende a-priori Fehlervarianz $\underset{\sim}{P}_t^- = \mathrm{E}\{(\vec{x}_t - \hat{\vec{x}}_t^-)(\vec{x}_t - \hat{\vec{x}}_t^-)^T\}$ benötigt, die selbst wiederum die Kenntnis des tatsächlichen - aber unbekannten - Systemzustandes $\vec{x}_t$ voraussetzt. Über mehrere Umformungen läßt sich $\underset{\sim}{P}_t^-$ jedoch vereinfachen:

$$\begin{aligned}
\underset{\sim}{P}_t^- &= \mathrm{E}\{(\vec{x}_t - \hat{\vec{x}}_t^-)(\vec{x}_t - \hat{\vec{x}}_t^-)^T\} = \\
&= \mathrm{E}\{(\underset{\sim}{A}_t\vec{x}_{t-1} + \vec{u}_t)(\vec{x}_{t-1}^T\underset{\sim}{A}_t^T + \vec{u}_t^T)\} = \\
&= \mathrm{E}\{(\underset{\sim}{A}_t\vec{x}_{t-1}\vec{x}_{t-1}^T\underset{\sim}{A}_t^T + \underset{\sim}{A}_t\vec{x}_{t-1}\vec{u}_t^T + \vec{u}_t\vec{x}_{t-1}\underset{\sim}{A}_t^T + \vec{u}_t\vec{u}_t^T)\} \qquad \text{(D.12)}
\end{aligned}$$

Da das Rauschsignal als unabhängig von den Systemzuständen angenommen wurde, kann Gleichung D.12 schließlich überführt werden in

$$\underset{\sim}{P}_t^- = \underset{\sim}{A}_t\mathrm{E}\{\vec{x}_{t-1}\vec{x}_{t-1}^T\}\underset{\sim}{A}_t^T + \mathrm{E}\{\vec{u}_t\vec{u}_t^T\} = \underset{\sim}{A}_t\underset{\sim}{P}_{t-1}\underset{\sim}{A}_t^T + \underset{\sim}{\Sigma}_u \qquad \text{(D.13)}$$

Auch hierfür wird in Form der Fehlervarianz $\underset{\sim}{P}_{t-1}$ wiederum Wissen über die tatsächlichen Systemzustände benötigt, weswegen auch $\underset{\sim}{P}_t$ entsprechend umgeformt wird[1]:

$$\begin{aligned}
\underset{\sim}{P}_t &= \mathrm{E}\{(\vec{x}_t - \hat{\vec{x}}_t)(\vec{x}_t - \hat{\vec{x}}_t)^T\} = \\
&= \mathrm{E}\{\left(\vec{x}_t - ((\underset{\sim}{\mathbb{1}} - \underset{\sim}{K}_t\underset{\sim}{H}_t)\vec{x}_t^- + \underset{\sim}{K}_t\vec{y}_t^*)\right)\left(\vec{x}_t - ((\underset{\sim}{\mathbb{1}} - \underset{\sim}{K}_t\underset{\sim}{H}_t)\vec{x}_t^- + \underset{\sim}{K}_t\vec{y}_t^*)\right)^T\} = \\
&= \mathrm{E}\{(\underset{\sim}{\mathbb{1}} - \underset{\sim}{K}_t\underset{\sim}{H}_t)(\vec{x}_t - \hat{\vec{x}}_t^-)(\vec{x}_t - \hat{\vec{x}}_t^-)^T(\underset{\sim}{\mathbb{1}} - \underset{\sim}{K}_t\underset{\sim}{H}_t)^T + \underset{\sim}{K}_t\vec{v}_t\vec{v}_t^T\underset{\sim}{K}_t^T\} = \\
&= (\underset{\sim}{\mathbb{1}} - \underset{\sim}{K}_t\underset{\sim}{H}_t)\underset{\sim}{P}_t^{-1}(\underset{\sim}{\mathbb{1}} - \underset{\sim}{K}_t\underset{\sim}{H}_t)^T + \underset{\sim}{K}_t\underset{\sim}{\Sigma}_v\underset{\sim}{K}_t^T \qquad \text{(D.14)}
\end{aligned}$$

Substituiert man in dieser Gleichung das Matrizenprodukt $\underset{\sim}{K}_t\underset{\sim}{\Sigma}_v$ durch die aus Gleichung D.11 herleitbare Beziehung $\underset{\sim}{K}_t\underset{\sim}{\Sigma}_v = (\underset{\sim}{\mathbb{1}} - \underset{\sim}{K}_t\underset{\sim}{H}_t)\underset{\sim}{P}_t^-\underset{\sim}{H}_t^T$, so ergibt sich für $\underset{\sim}{P}_t$ die Gleichung:

$$\begin{aligned}
\underset{\sim}{P}_t &= (\underset{\sim}{\mathbb{1}} - \underset{\sim}{K}_t\underset{\sim}{H}_t)\underset{\sim}{P}_t^-(\underset{\sim}{\mathbb{1}} - \underset{\sim}{K}_t\underset{\sim}{H}_t)^T + (\underset{\sim}{\mathbb{1}} - \underset{\sim}{K}_t\underset{\sim}{H}_t)\underset{\sim}{P}_t^-\underset{\sim}{H}_t^T\underset{\sim}{K}_t^T = \\
&= (\underset{\sim}{\mathbb{1}} - \underset{\sim}{K}_t\underset{\sim}{H}_t)\underset{\sim}{P}_t^- \qquad \text{(D.15)}
\end{aligned}$$

Zusammenfassend läßt sich ein Kalmanfilter demnach aufteilen in zwei abwechselnd durchzuführende Operationen: In einer Prädiktion werden ausgehend von der zum letzten Zeitpunkt vorliegenden Zustandsschätzung $\hat{\vec{x}}_{t-1}$ und der daraus resultierenden Fehlervarianz $\underset{\sim}{P}_{t-1}$ die a-priori Werte

$$\begin{aligned}
\hat{\vec{x}}_t^- &= \underset{\sim}{A}_t\hat{\vec{x}}_{t-1} \qquad \text{(D.16)} \\
\underset{\sim}{P}_t^- &= \underset{\sim}{A}_t\underset{\sim}{P}_{t-1}\underset{\sim}{A}_t^T + \underset{\sim}{\Sigma}_u \qquad \text{(D.17)}
\end{aligned}$$

[1] Bei der Umformung wurde wiederum die stochastische Unabhängigkeit des Rauschsignals $\vec{v}_t$ von den Systemzuständen $\vec{x}_t$ sowie $\hat{\vec{x}}_t^-$ benutzt.

berechnet. Die für diesen iterativen Prozess notwendigen Anfangsbedingungen $\hat{x}_0$ und $\underset{\sim}{P}_0$ können hierbei nahezu beliebig initialisiert werden, wobei der Realität entsprechende Werteannahmen für die beiden Variablen das Einschwingverhalten des Filters entsprechend verkürzen können. Gefolgt von einem Korrekturschritt, in den die aktuelle Beobachtung $\vec{y}_t$ über den aktuellen Kalman Gain $\underset{\sim}{K}_t$ einfließt, werden die a-priori Werte $\hat{\vec{x}}_t^-$ und $\underset{\sim}{P}_t^-$ weiterentwickelt zu:

$$\underset{\sim}{K}_t = \underset{\sim}{P}_t^- \underset{\sim}{H}_t^T (\underset{\sim}{H}_t \underset{\sim}{P}_t^- \underset{\sim}{H}_t^T + \underset{\sim}{\Sigma}_v)^{-1} \tag{D.18}$$

$$\hat{\vec{x}}_t = \hat{\vec{x}}_t^- + \underset{\sim}{K}_t (\vec{y}_t - \underset{\sim}{H}_t \hat{\vec{x}}_t^-) \tag{D.19}$$

$$\underset{\sim}{P}_t = (\underset{\approx}{\mathbb{1}} - \underset{\sim}{K}_t \underset{\sim}{H}_t) \underset{\sim}{P}_t^- \tag{D.20}$$

Literaturverzeichnis

[1] *Augmented Multi-party Interaction with Distance Access.* – EU FP6-IST Programm IST-2005-033812

[2] *Augmented Multiparty Interaction.* – EU FP6-IST Programm IST-2002-506811

[3] *Computers in the Human Interaction Loop.* – EU FP6-IST Programm IST-2002-506909

[4] *MultiModal Meeting Manager.* – EU FP5-IST Programm IST-2001-34485

[5] *Nist Meeting Room Project.* `http://www.nist.gov/speech/test_beds/index.html`, Abruf: 31.08.2008

[6] Al-Hames, M.; Rigoll, G.: A Multi-modal Graphical Model for Robust Recognition of Group Actions in Meetings from Disturbed Videos. In: *Proceedings of the International Conference on Image Processing (ICIP)*, 2005

[7] Al-Hames, M.; Rigoll, G.: A Multi-Modal Mixed-State Dynamic Bayesian Network for Robust Meeting Event Recognition from Disturbed Data. In: *Proceedings of the 6th International Conference on Multimedia and Expo (ICME)*, 2005

[8] Arca, S. ; Campadelli, P. ; Lanzarotti, R.: A Face Recognition System Based on Local Feature Analysis. In: Kittler, J. (Hrsg.); Nixon, M. S. (Hrsg.): *Proceedings of the 4th International Conference on Audio- and Video-Based Biometrie Person Authentication (AVBPA)* Bd. 2688. Guildford, UK : Springer, June 2003 (Lecture Notes in Computer Science), S. 182–189

[9] Bashir, F.; Porikli, F.: Performance Evaluation of Object Detection and Tracking Systems. In: *Proceedings of the 9th IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS).* New York, NY, USA, June 2006, S. 7–14

[10] Baum, L.; Petrie, T.: Statistical Inference for Probabilistic Functions of Finite State Markov Chains. In: *The Annals of Mathematical Statistics* 37 (1966), S. 1554–1563

[11] Baum, L. ; Petrie, T. ; Soules, G. ; Weiss, N.: A Maximization Technique Occuring in The Statistical Analysis of Probabilistic Functions of Markov Chains. In: *The Annals of Mathematical Statistics* 41 (1970), Nr. 1, S. 164–171

[12] Baumberg, A.: *Learning Deformable Models for Tracking Human Motion*, University of Leeds, Diss., 1995

[13] Bernardin, K. ; Elbs, A. ; Stiefelhagen, R.: Multiple Object Tracking Performance Metrics and Evaluation in a Smart Room Environment. In: *Proceedings of the Sixth IEEE International Workshop on Visual Surveillance (ECCV-VS)*. Graz, Austria, May 2006

[14] Bernardin, K.; Stiefelhagen, R.: Audio-Visual Multi-Person Tracking and Identification for Smart Environments. In: *Proceedings of the 15th international conference on Multimedia (MULTIMEDIA)*. New York, NY, USA : ACM, 2007, S. 661–670

[15] Bilmes, J. A.: A gentle tutorial on the EM algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. 1997. – Forschungsbericht

[16] Birchfield, S.: An elliptical head tracker. In: *In Proceedings of the 31st Asilomar Conference on Signals, Systems and Computers*, 1997, S. 1710–1714

[17] Birchfield, S.: Elliptical Head Tracking Using Intensity Gradients and Color Histograms. In: *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*. Washington, DC, USA : IEEE Computer Society, 1998, S. 232

[18] Black, J. ; Ellis, T. ; Rosin, P.: A novel method for video tracking performance evaluation. In: *In Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS*, 2003, S. 125–132

[19] Bobick, A. F. ; Intille, S. S. ; Davis, J. W. ; Baird, F. ; Pinhanez, C. S. ; Campbell, L. W. ; Ivanov, Y. A. ; Schütte, A. ; Wilson, A.: The KidsRoom: A Perceptually-Based Interactive and Immersive Story Environment. In: *Presence: Teleoper. Virtual Environ.* 8 (1999), Nr. 4, S. 369–393. `http://dx.doi.org/http://dx.doi.org/10.1162/105474699566297`. – DOI http://dx.doi.org/10.1162/105474699566297. – ISSN 1054–7460

[20] Brown, D. ; Craw, I. ; Lewthwaite, J.: A SOM Based Approach to Skin Detection with Application in Real Time Systems. In: *in Proc. of the British Machine Vision Conference*, 2001

[21] Campbell, J. J.P.: Speaker recognition: a tutorial, Sep 1997, S. 1437–1462

[22] Cha, S.-H.; Srihari, S. N.: On measuring the distance between histograms. In: *Pattern Recognition* 35 (2002), June, Nr. 6, S. 1355–1370

[23] Chai, D.; Ngan, K.: Face segmentation using skin-color map in videophone applications. In: *Circuits and Systems for Video Technology, IEEE Transactions on* 9 (Jun 1999), Nr. 4, S. 551–564

[24] Cheung, S.-C. S.; Kamath, C.: Robust techniques for background subtraction in urban traffic video. In: Panchanathan, S. (Hrsg.); Vasudev, B. (Hrsg.): *Visual Communications and Image Processing 2004. Edited by Panchanathan, Sethuraman; Vasudev, Bhaskaran. Proceedings of the SPIE, Volume 5308, pp. 881-892 (2004).* Bd. 5308, 2004 (Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference), S. 881–892

[25] Cole, R. A. ; Chief, I. ; Mariani, J. ; Uszkoreit, H. ; Zaenen, A. ; Varile, G. ; Zampolli, A. ; Cole, R. ; Zue, V. ; Zue, V. ; Cole, R. (Hrsg.): *Survey of the state of the art in human language technology.* New York, NY, USA : Cambridge University Press, 1997. – ISBN 0–521–59277–1

[26] Comaniciu, D.; Meer, P.: Mean Shift: A Robust Approach Toward Feature Space Analysis. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24 (2002), Nr. 5, S. 603–619

[27] Cootes, T. F. ; Taylor, C. J. ; Cooper, D. H. ; Graham, J.: Active shape models - their training and application. In: *Computer Vision and Image Understanding* 61 (1995), Nr. 1, S. 38–59

[28] Cucchiara, R. ; Grana, C. ; Piccardi, M. ; Prati, A.: Detecting moving objects, ghosts, and shadows in video streams. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25 (Oct. 2003), Nr. 10, S. 1337–1342

[29] Daugman, J.: Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression. In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 36 (Jul 1988), Nr. 7, S. 1169–1179

[30] Dempster, A. ; Laird, N. ; Rubin, D.: Maximum Likelihood from Incomplete Data via the EM Algorithm. In: *Journal of the Royal Statistical Society B* 39 (1977), Nr. 1, S. 1–38

[31] Elgammal, A. M. ; Harwood, D. ; Davis, L. S.: Non-parametric Model for Background Subtraction. In: *ECCV '00: Proceedings of the 6th European Conference on Computer Vision-Part II*. London, UK : Springer-Verlag, 2000. – ISBN 3–540–67686–4, S. 751–767

[32] Ellis, T.: Performance metrics and methods for Tracking in Surveillance. In: *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS)*. Copenhagen, Denmark, 2002

[33] Freund, Y.: Boosting a weak learning algorithm by majority. In: *COLT '90: Proceedings of the third annual workshop on Computational learning theory*. San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 1990. – ISBN 1–55860–146–5, S. 202–216

[34] Freund, Y.; Schapire, R. E.: A brief introduction to boosting. In: *In Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, Morgan Kaufmann, 1999, S. 1401–1406

[35] Gatica-Perez, D. ; Odobez, J.-M. ; Ba, S. ; Smith, K. ; Lathoud, G.: Tracking People in Meetings with Particles. In: *Proceedings of the International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS), invited paper*, 2005

[36] Geiser, G.: *Mensch-Maschine-Kommunikation.* München : Oldenburg, 1990

[37] Gower, J.: Generalized procrustes analysis. In: *Psychometrika* 40 (1975), March, Nr. 1, S. 33–51

[38] Green, M. W.: The Appropriate and Effective Use of Security Technologies in U.S. Schools, A Guide for Schools and Law Enforcement Agencies / Sandia National Laboratories. 1999 (NCJ 178265). – Forschungsbericht

[39] Grenander, U. ; Chow, Y. ; Keenan, D. M.: *Hands: a pattern theoretic study of biological shapes*. New York, NY, USA : Springer-Verlag New York, Inc., 1991. – ISBN 0–387–97386–9

[40] Haritaoglu, I. ; Harwood, D. ; Davis, L. S.: W4: Who? When? Where? What? A Real Time System for Detecting and Tracking People. In: *FG*, 1998, S. 222–227

[41] Haritaoglu, I. ; Harwood, D. ; David, L. S.: W4: Real-Time Surveillance of People and Their Activities. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 22 (2000), Nr. 8, S. 809–830. `http://dx.doi.org/http://dx.doi.org/10.1109/34.868683`. – DOI http://dx.doi.org/10.1109/34.868683. – ISSN 0162–8828

[42] Harris, C.; Stephens, M.: A Combined Corner and Edge Detection. In: *Proceedings of The Fourth Alvey Vision Conference*, 1988, 147–151

[43] Hochstein, S.; Ahissar, M.: View from the top: hierarchies and reverse hierarchies in the visual system. In: *Neuron* 36 (2002), December, Nr. 5, S. 791–804

[44] Horn, B. K. P.; Schunck, B. G.: Determining Optical Flow. In: *Artificial Intelligence* 17 (1981), S. 185–203

[45] Hsu, R.-L. ; Abdel-Mottaleb, M. ; Jain, A. K.: Face detection in color images. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 24 (2002), Nr. 5, S. 696–706

[46] Huet, B.; Hancock, E. R.: Cartographic indexing into a database of remotely sensed images. In: *In Third IEEE Workshop on Applications of Computer Vision (WACV96*, 1996, S. 8–14

[47] Isard, M.; Blake, A.: Condensation – conditional density propagation for visual tracking. In: *International Journal of Computer Vision* 29 (1998), S. 5–28

[48] Isard, M.; Maccormick, J.: BraMBLe: a Bayesian multiple-blob tracker. In: *Proceedings of the 8th IEEE International Conference on Computer Vision (ICCV)* Bd. 2, 2001, 34–41 vol.2

[49] Isard, M.; Blake, A.: ICONDENSATION: Unifying Low-Level and High-Level Tracking in a Stochastic Framework. Version: 1998. `http://www.springerlink.com/content/g05d61nfx3jklu1q`. 1998, 893+

[50] Javed, O.; Shah, M.: Tracking and Object Classification for Automated Surveillance. In: *Proceedings of the 7th European Conference on Computer Vision-Part IV (ECCV)*. London, UK : Springer-Verlag, 2002. – ISBN 3–540–43748–7, S. 343–357

[51] Jiao, F. ; Li, S. ; Shum, H.-Y. ; Schuurmans, D.: Face alignment using statistical models and wavelet features. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)* 1 (18-20 June 2003), S. I–321–I–327 vol.1

[52] Jones, M. J.; Rehg, J. M.: Statistical Color Models with Application to Skin Detection. In: *International Journal of Computer Vision* 46 (1999), Nr. 1, 81-96. `citeseer.ist.psu.edu/jones99statistical.html`

[53] Kakumanu, P. ; Makrogiannis, S. ; Bourbakis, N.: A survey of skin-color modeling and detection methods. In: *Pattern Recognition* 40 (2007), Nr. 3, S. 1106–1122

[54] Kalman, R.: A New Approach to Linear Filtering and Prediction Problems. In: *Transactions of the ASME - Journal of Basic Engineering* (1960), Nr. 82, S. 35–45

[55] Kang, J. ; Cohen, I. ; Medioni, G. G.: Object Reacquisition Using Invariant Appearance Model. In: *Proceedings of the 17th International Conference on Pattern Recognition (ICPR)*, 2004, S. 759–762

[56] Kanungo, T. ; Mount, D. M. ; Netanyahu, N. S. ; Piatko, C. D. ; Silverman, R. ; Wu, A. Y. ; Member, S. ; Member, S.: An efficient k-means clustering algorithm: Analysis and implementation. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24 (2002), S. 881–892

[57] Kass, M. ; Witkin, A. ; Terzopoulos, D.: Snakes: Active contour models. In: *International Journal of Computer Vision* V1 (1988), January, Nr. 4, 321–331. `http://dx.doi.org/10.1007/BF00133570`. – DOI 10.1007/BF00133570

[58] Kirkpatrick, S. ; Gelatt, C. D. ; Vecchi, M. P.: Optimization by Simulated Annealing. In: *Science, Number 4598, 13 May 1983* 220, 4598 (1983), S. 671–680

[59] Koller, D. ; Weber, J. ; Huang, T. ; Malik, J. ; Ogasawara, G. ; Rao, B. ; Russell, S.: Towards robust automatic traffic scene analysis in real-time. In: *Proceedings of the International Conference on Pattern Recognition (ICPR)*. Israel, 1994

[60] Kovac, J. ; Peer, P. ; Solina, F.: Human skin color clustering for face detection. In: *EUROCON 2003. Computer as a Tool. The IEEE Region 8* 2 (2003), Sept., S. 144–148 vol.2

[61] Lazarevic-McManus, N. ; Renno, J. ; Jones, G. A.: Performance evaluation in visual surveillance using the F-measure. In: *VSSN '06: Proceedings of the 4th ACM international workshop on Video surveillance and sensor networks*. New York, NY, USA : ACM Press, 2006. – ISBN 1–59593–496–0, S. 45–52

[62] Lienhart, R.; Maydt, J.: An extended set of Haar-like features for rapid object detection. In: *Proceedings of the International Conference on Image Processing* 1 (2002), S. I–900–I–903 vol.1

[63] Lienhart, R. ; Kuranov, A. ; Pisarevsky, V.: Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection. In: *DAGM'03, 25th Pattern Recognition Symposium*. Magdeburg, Germany, Sep. 2003, S. 297–304

[64] Lucas, B.; Kanade, T.: An Iterative Image Registration Technique with an Application to Stereo Vision. (1981), S. 674–679

[65] Mamon, R. S.; Elliott, R. J.: *Hidden Markov Models in Finance*. Berlin : Springer, 2007

[66] Manohar, V. ; Soundararajan, P. ; Raju, H. ; Goldgof, D. ; Kasturi, R. ; Garofolo, J.: Performance Evaluation of Object Detection and Tracking in Video, 2006, S. II:151–161

[67] McCowan, I. ; Bengio, S. ; Gatica-Perez, D. ; Lathoud, G. ; Monay, F. ; Moore, D. ; Wellner, P. ; Bourlard, H.: Modeling human interaction in meetings, 2003, IV–748-51 vol.4

[68] Menser, B.; Muller, F.: Face detection in color images using principal components analysis. In: *In Seventh International Conf. on Image Processing and Its Applications*, 1999, S. 620–624

[69] Metropolis, N. ; Rosenbluth, A. W. ; Rosenbluth, M. N. ; Teller, A. H. ; Teller, E.: Equation of State Calculations by Fast Computing Machines. In: *J. Chem. Phys* 21 (1953), S. 1087–1092

[70] Mitra, T. S.; A. S.; Acharya: Gesture Recognition: A Survey. In: *IEEE Transactions on Systems, Man and Cybernetics, Part C: Applications and Reviews* 37 (May 2007), Nr. 3, S. 311–324

[71] Papageorgiou, C. P. ; Oren, M. ; Poggio, T.: 1998, A general framework for object detection. In: *Sixth International Conference on Computer Vision* (1998), S. 555–562

[72] Pedersen, J. S.; Hein, J.: Gene finding with a hidden Markov model of genome structure and evolution. In: *Bioinformatics*, 2003

[73] Phung, S. L. ; Bouzerdoum, A. ; Chai, D.: A novel skin color model in YCbCr color space and its application to human face detection. In: *Proceedings of the International Conference on Image Processing* 1 (2002), S. I–289–I–292 vol.1

[74] Pingali, S. G.; Segen, J.: Performance Evaluation of People Tracking Systems. In: *Proceedings of the 3rd IEEE Workshop on Applications of Computer Vision (WACV)*. Washington, DC, USA : IEEE Computer Society, 1996. – ISBN 0–8186–7620–5, S. 33

[75] Pizer, S. ; Johnston, R. ; Ericksen, J. ; Yankaskas, B. ; Muller, K.: Contrast-limited adaptive histogram equalization: speed and effectiveness. In: *Proceedings of the First Conference on Visualization in Biomedical Computing* (22-25 May 1990), S. 337–345

[76] Potucek, I. ; Beran, V. ; Sumec, S. ; Zemcik, P.: Evaluation and comparison of tracking methods using meeting omnidirectional images. In: *Workshop on Multimodal Interaction and Related Machine Learning Algorithms (MLMI)*, 2007, 12

[77] Power, W. P.; Schoonees, J. A.: Understanding Background mixture models for foreground segmentation. In: *Proceedings Image and Vision Computing New Zealand 2002*, 2002

[78] Prati, A. ; Mikic, I. ; Trivedi, M. M. ; Cucchiara, R.: Detecting moving shadows: Algorithms and evaluation. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 25 (2003), S. 918–923

[79] Rabiner, L. R.: A tutorial on hidden Markov models and selected applications in speech recognition. In: *Proceedings of the IEEE* 77 (1989), Nr. 2, S. 257–286

[80] Racine, V. ; Hertzog, A. ; Jouanneau, J. ; Salamero, J. ; Kervrann, C. ; B. Sibarita, J.: Multiple target tracking of 3D fluorescent objects based on simulated annealing. In: *Proc. Int. Symp. on Biomedical Imaging (ISBI'06)*. Arlington, USA, April 2006

[81] Reiter, S. ; Schreiber, S. ; Rigoll, G.: Multimodal Meeting Analysis by Segmentation and Classification of Meeting Events based on a Higher Level Semantic Approach. In: *Proceedings of the 30th International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. Philadelphia, USA, March 2005

[82] Ridder, C. ; Munkelt, O. ; Kirchner, H.: Adaptive background estimation and foreground detection using kalman-filtering, 1995, S. 193–199

[83] Riedmiller, M.; Braun, H.: A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP algorithm. In: *Proceedings of the International Conference on Neural Networks*. San Francisco, CA, 1993, 586–591

[84] Rigoll, G. ; Kosmala, A. ; Eickeler, S.: High Performance Real-Time Gesture Recognition Using Hidden Markov Models. In: *Lecture Notes in Computer Science* 1371 (1998), S. 69–80

[85] Romano, N.; Nunamaker, J.: Meeting Analysis: Findings from Research and Practice. In: *Proceedings of the 34th Annual Hawaii International Conference on System Sciences (HICSS)-Volume 1*. Washington, DC, USA : IEEE Computer Society, 2001. – ISBN 0–7695–0981–9, S. 1072

[86] Rowley, H. A. ; Baluja, S. ; Kanade, T.: Neural Network-Based Face Detection. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 20 (1998), Nr. 1, S. 23–38

[87] Sanderson, C.; Paliwal, K. K.: Fast features for face authentication under illumination direction changes. In: *Pattern Recogn. Lett.* 24 (2003), Nr. 14, S. 2409–2419

[88] Schlogl, T. ; Beleznai, C. ; Winter, M. ; Bischof, H.: Performance Evaluation Metrics for Motion Detection and Tracking. In: *Proceedings of the 17th International Conference on Pattern Recognition (ICPR)*. Washington, DC, USA : IEEE Computer Society, 2004, S. 519–522

[89] Schreiber, S.; Gatica-Perez, D.: Evaluation Scheme for Tracking in AMI / Augmented Multi-Party Interaction. 2006. – Forschungsbericht

[90] Schreiber, S.; Rigoll, G.: Robust Omni-directional Multi-cue Tracking for Multiple Person Meeting Scenarios. In: *The Sixth IEEE International Workshop on Visual Surveillance 2006 (ECCV-VS)*. Graz, Austria, May 2006

[91] Schreiber, S.; Rigoll, G.: Omni-directional Multiperson Tracking in Meeting Scenarios combining Simulated Annealing and Particle Filtering. In: *Proceedings of the 8th International Conference on Automatic Face and Gesture Recognition (FG)*. Amsterdam, The Netherlands, September 2008

[92] Schreiber, S. ; Störmer, A. ; Rigoll, G.: A Hierarchical ASM/AAM Approach in a Stochastic Framework for Fully Automatic Tracking and Recognition. In: *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. Atlanta, GA USA, October 2006, S. 1773–1776

[93] Schreiber, S. ; Störmer, A. ; Rigoll, G.: Omnidirectional tracking and recognition of persons in planar views. In: *Proceedings of the IEEE International Conference on Image Processing (ICIP)*. San Diego, California, U.S.A., October 2008

[94] Schultheis, A.: iPhone und das Prinzip der Einfachheit. In: *Wirtschaftsbild* 15 (2007), S. 2

[95] Schultz, T. ; Waibel, A. ; Bett, M. ; Metze, F. ; Pan, Y. ; Ries, K. ; Schaaf, T. ; Soltau, H. ; Westphal, M. ; Yu, H. ; Zechner, K.: The ISL Meeting Room System. In: *Proceedings of the Workshop on Hands-Free Speech Communication (HSC-2001)*, 2001

[96] Schwarz, G.: Estimating the Dimension of a Model. In: *The Annals of Statistics* 6 (1978), Nr. 2, S. 461–464

[97] Senior, A. W. ; Hampapur, A. ; Tian, Y.-L. ; Brown, L. ; Pankanti, S. ; Bolle, R. M.: Appearance Models for Occlusion Handling. In: *Second*

IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS). Kauai, Hawaii, USA, Dezember 2001

[98] Shen, C. ; Brooks, M. J. ; Hengel, A. van d.: Augmented particle filtering for efficient visual tracking. In: *IEEE International Conference on Image Processing (ICIP)*. Genoa, Italy, September 2005

[99] Skarbek, W.; Koschan, A.: Colour Image Segmentation — A Survey / Institute for Technical Informatics, Technical University of Berlin. 1994. – Forschungsbericht

[100] Smith, K. ; Gatica-Perez, D. ; Odobez, J.-M. ; Ba, S.: Evaluating Multi-Object Tracking. In: *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR) - Workshops*. Washington, DC, USA : IEEE Computer Society, 2005. – ISBN 0–7695–2372–2–3, S. 36

[101] Smith, K. ; Schreiber, S. ; Potucek, I. ; Beran, V. ; Rigoll, G. ; Gatica-Perez, D.: 2D Multi-Person Tracking: A Comparative Study in AMI Meetings. In: *First International Evaluation Workshop on Classification of Events, Activities and Relationships (CLEAR)*. Southampton, UK, April 2006

[102] Soriano, M. ; Martinkauppib, B. ; Huovinen, S. ; Laaksonen, M.: Adaptive skin color modeling using the skin locus for selecting training pixels. In: *Pattern Recognition* 36 (2003), March, Nr. 3, S. 681–690

[103] Stauffer, C.; Grimson, W. E. L.: Adaptive Background Mixture Models for Real-Time Tracking. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*, 1999, S. 2246–2252

[104] Stiefelhagen, R. (Hrsg.); Garofolo, J. S. (Hrsg.): *Multimodal Technologies for Perception of Humans, First International Evaluation Workshop on Classification of Events, Activities and Relationships, CLEAR 2006, Southampton, UK, April 6-7, 2006, Revised Selected Papers*. Bd. *4122*. Springer, 2007 (Lecture Notes in Computer Science). – ISBN 978–3–540–69567–7

[105] Stoerring, M. ; Kočka, T. ; Andersen, H. J. ; Granum, E.: Tracking regions of human skin through illumination changes. In: *Pattern Recognition Letters* 24 (2003), Nr. 11, S. 1715–1723

[106] Sung, K. K.; Poggio, T.: Example Based Learning for View-Based Human Face Detection. Cambridge, MA, USA : Massachusetts Institute of Technology, 1994. – Forschungsbericht

[107] Sung, K. K.: *Learning and example selection for object and pattern detection*, Diss., 1996. – Supervisor-Tomaso A. Poggio

[108] Van Rijsbergen, C. J.: *Information Retrieval, 2nd edition.* Dept. of Computer Science, University of Glasgow, 1979

[109] Vezhnevets, V. ; Sazonov, V. ; Andreeva, A.: A survey on pixel-based skin color detection techniques. In: *In Proceedings of GraphiCon*, 2003, 85–92

[110] Viola, P.; Jones, M. J.: Robust Real-time Object Detection. In: *Second IEEE ICCV Workshop on Statistical and Computational Theories of Vision* (2001), July

[111] Viterbi, A. J.: Error bounds for convolutional codes and an asymptotically optimal decoding algorithm. In: *IEEE Transactions on Information Theory* 13 (1967), S. 260–269

[112] Wang, Y.; Yuan, B. Z.: A novel approach for human face detection from color images under complex background. In: *Pattern Recognition* 34 (2001), October, Nr. 10, S. 1983–1992

[113] Waring, C.; Liu, X.: Face detection using spectral histograms and SVMs. In: *IEEE Transactions on Systems, Man, and Cybernetics, Part B* 35 (June 2005), Nr. 3, S. 467–476

[114] Wark, T.; Sridharan, S.: A syntactic approach to automatic lip feature extraction for speaker identification. In: *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on* 6 (12-15 May 1998), S. 3693–3696 vol.6. `http://dx.doi.org/10.1109/ICASSP.1998.679685`. – DOI 10.1109/ICASSP.1998.679685. – ISSN 1520–6149

[115] Widrow, B.; Hoff, M. E.: Adaptive switching circuits. (1988), S. 123–134. ISBN 0–262–01097–6

[116] Wiskott, L. ; Fellous, J.-M. ; Krüger, N. ; Malsburg, C. von d.: *Face Recognition by Elastic Bunch Graph Matching*

[117] Wren, C. R. ; Azarbayejani, A. ; Darrell, T. ; Pentland, A.: Pfinder: Real-Time Tracking of the Human Body. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19 (1997), Nr. 7, 780-785. `citeseer.ist.psu.edu/wren97pfinder.html`

[118] Yamane, T. ; Shirai, Y. ; Miura, J.: Person tracking by integrating optical flow and uniform brightness regions. In: *Proceedings of the IEEE International Conference on Robotics and Automation* 4 (1998), May, S. 3267–3272 vol.4

[119] Yang, J. ; Lu, W. ; Waibel, A.: Skin-Color Modeling and Adaptation. In: *Proceedings of the Third Asian Conference on Computer Vision-Volume II (ACCV)*. London, UK : Springer-Verlag, 1997. – ISBN 3–540–63931–4, S. 687–694

[120] Yang, M.-H.; Ahuja, N.: Gaussian mixture model for human skin color and its application in image and video databases. In: *Its Application in Image and Video Databases. Proceedings of SPIE 1999*, 1999, S. 458–466

[121] Yilmaz, A. ; Javed, O. ; Shah, M.: Object tracking: A survey. In: *ACM Computing Surveys* 38 (2006), Nr. 4

[122] Zhao, T. ; Nevatia, R. ; Lv, F.: Segmentation and Tracking of Multiple Humans in Complex Situations. In: *In Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)* (2001)

[123] Zhao, T.; Nevatia, R.: Tracking multiple humans in complex situations. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 26 (Sept. 2004), Nr. 9, S. 1208–1221

[124] Zhao, W. ; Chellappa, R. ; Phillips, P. J. ; Rosenfeld, A.: Face recognition: A literature survey. In: *ACM Computing Survey* 35 (2003), Nr. 4, S. 399–458

[125] Zhu, Q. ; Cheng, K.-T. ; Wu, C.-T. ; Wu, Y.-L.: Adaptive learning of an accurate skin-color model. In: *Proceedings of the 6th IEEE International Conference on Automatic Face and Gesture Recognition (FG)* (17-19 May 2004), S. 37–42

[126] Zhu, X. ; Collins, R. ; Teh, S. K.: An open source tracking testbed and evaluation web site. In: *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (WAMOP-PETS)*, 2005, S. 17–24

[127] Zobl, M. ; Laika, A. ; Wallhoff, F. ; Rigoll, G.: Recognition of Partly Occluded Person Actions in Meeting Scenarios. In: *Proceedings of the International Conference on Image Processing (ICIP)*, 2004