Lehrstuhl für Mensch-Maschine-Kommunikation
der Technischen Universität München

Entwicklung und Evaluierung neuartiger Verfahren zur automatischen Gesichtsdetektion, Identifikation und Emotionserkennung

Frank Wallhoff

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.-Ing. Joachim Hagenauer
Prüfer der Dissertation:	1.	Univ.-Prof. Dr.-Ing. habil. Gerhard Rigoll
	2.	Univ.-Prof. Dr.-Ing. Eckehard Steinbach

Die Dissertation wurde am 05.10.2005 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 21.03.2006 angenommen.

Vorwort

Die vorliegende Arbeit ist das Ergebnis meiner als wissenschaftlicher Mitarbeiter am Fachgebiet Technische Informatik des Fachbereichs Elektrotechnik an der Gerhard-Mercator-Universität Duisburg begonnenen und als wissenschaftlicher Assistent am Lehrstuhl für Mensch-Maschine-Kommunikation an der Technischen Universität München nahtlos fortgesetzten Tätigkeiten.

Dem Leiter der beiden Lehrstühle, Herrn Prof. Dr.-Ing. habil. Gerhard Rigoll, gilt mein außerordentlicher Dank für die Möglichkeit der Bearbeitung der vorgestellten Themen mit angemessenem methodischen Spielraum. Mit ruhiger und sicherer Hand sowie wertvollen Anregungen sorgte er stets für eine klare wissenschaftliche Zielsetzung, was erheblich zum Inhalt dieser Arbeit beigetragen hat.

Ebenso gilt mein Dank an dieser Stelle Univ.-Prof. Dr.-Ing. Eckehard Steinbach vom Lehrstuhl für Kommunikationsnetze an der Technischen Universität München für die Übernahme des Zweitgutachtens und den damit verbundenen Mühen.

Für das überdurchschnittlich angenehme Klima an beiden Lehrstühlen möchte ich mich bei jeweils allen Angehörigen bedanken. Dem kongenialen Kollegen Martin Herwig Zobl ein außerordentliches Dankeschön für die zahllosen nicht immer nur fachlich fundierten Gespräche: „Mach fertig!" Ebenso sind Björn W. Schuller und Dejan Arsić aufgrund der vielen Diskussionen nicht zu vergessen. An Peter Brand, Heiner Hundhammer sowie Ernst Ertl für die stets einsatzbereite Infrastruktur ein großes Lob.

An dieser Stelle seien auch Daniel Willett, Stefan Müller-Schneiders und Claus von Rücker aufgrund ihres Korrekturlesens und für manch nützlichen Tipp nicht vergessen.

Für ihren permanenten Beistand und ihre Geduld danke ich meiner Frau Silke, sowie meinen Kindern Joshua Lucas und Josefina Nele. Leider sind sie bei mir während der Ausarbeitungsphase oftmals nur auf geteilte Aufmerksamkeit gestoßen.

München, im Oktober 2005 Frank Wallhoff

Kurzfassung

Die Arbeit befasst sich mit der Konzeption, Integration und Beurteilung von neuen, rechnergestützten Methoden zur Findung und Erkennung von Gesichtern in Einzel- und Bewegtbildern, sowie der Emotionserkennung über deren dynamisches Mienenspiel.

Die verwendeten Methoden stammen aus dem Bereich der Mustererkennung und umfassen die Hauptachsentransformation, künstliche neuronale Netze, Support Vektor Maschinen, Hidden Markov Modelle sowie daraus abgeleitete hybride Systeme.

Nach der Untersuchung der involvierten Einzelkomponenten werden neuartige, integrierte Systeme zur blickwinkelunabhängigen Gesichtverfolgung für omnidirektionales Bildmaterial und zur robusten kontaktlosen Zugangskontrolle in Flugzeugen präsentiert. Darüber hinaus werden Lösungen zur Profilerkennung bei modellierter Frontalansicht, wie auch eine Personen unabhängige Erkennung spontaner Mimiken vorgestellt, und in Anlehnung an die Perzeptionsleistung des Menschen evaluiert.

Inhaltsverzeichnis

Kapitel 1

Einleitung

Zur Wahrnehmung der Umwelt stellt die Sinnesmodalität des Sehens mit ihrer enormen Leistungsfähigkeit einen der wichtigsten rezeptiven Kanäle des Menschen dar. So hätte der Verlust des Augenlichts neben dem Hörsinn gegenüber den meisten anderen Sinnen, wie dem Tast-, Geruchs- oder dem Geschmackssinn oftmals die gravierensten Änderungen der bisherigen Lebensgewohnheiten zur Folge [Fis87].

Bereits während der frühen Entwicklungsphase eines Menschen, innerhalb der ersten Lebensmonate, prägen sich die Fähigkeiten zur Wahrnehmung von Gesichtern. In einem Alter von ca. 3 Monaten ist ein Säugling zudem in der Lage, erste soziale Kontakte durch Lächeln aufzubauen [Hel76]. Auch später ist die visuelle Wahrnehmung zur Kommunikation und für soziale Kontakte von großer Bedeutung [Dar72]. Über das Sehen kann mit hoher Geschwindigkeit eine Vielzahl von Informationen aufgenommen werden, was umgangssprachlich durch das bekannte Sprichwort: „Ein Bild sagt mehr als 1000 Worte" ausgedrückt werden kann.

Ziel der vorliegenden Arbeit ist die rechnergestütze Implementierung der gesichtsbasierten Funktionen im Rahmen der visuellen Wahrnehmung, welche sind:

1. **Das Aufspüren von Gesichtern.** Diese nachzubildende Funktion liefert unabhängig vom gewählten Medium und ohne Vorwissen zuverlässige Informationen über die Position, Ausrichtung und Größe von Gesichtern. Die Medien können Grauwert- oder Farbbilder sowie Videosequenzen sein.

2. **Die Erkennung bekannter Personen.** Zweck einer visuell basierten Erkennung ist die sichere Zuordnung der Identität über ein durch Bilddaten gegebenes personenspezifisches Gesichtsmuster. Unterschieden wird zwischen einer frontalen Erkennung sowie der Zuordnung von Profilbildern zu Vorderansichten.

3. **Die Emotionserkennung über die Mimik.** Hier wird versucht, die aus Gesichtern ablesbare Mimik so auszuwerten, dass eine Aussage über den inneren Gemütszustand einer Person gemacht werden kann.

1.1 Motivation

Durch zuverlässige Lösungen zur Detektion, Identifikation und Emotionserkennung über das kontaktlose Messen und Auswerten von Gesichtern mit immer günstigeren und präziseren Kameras sowie bezahlbarer Rechenleistung eröffnet sich eine Vielzahl an technischen Anwendungsgebieten. So werden innerhalb des durch die Europäische Kommission geförderten *Face and Gesture*-Netzwerks (FGNET) visionäre Szenarien mit kommerziellen Anwendungen skizziert und formuliert [Wal02, Wal03a].

Neben der haptischen Bedienung technischer Geräte liefern die natürlichen Kommunikationskanäle des Menschen einen noch eher untergeordneten Beitrag zur heutigen Mensch-Maschine-Kommunikation. Zwar hat die verbesserte Leistungsfähigkeit von automatischen Spracherkennungssystemen eine zunehmende Bedeutung für die Bedienung von Mobiltelefonen oder Infotainmentsystemen in Kraftfahrzeugen erfahren, oft ist der Nutzer aber nicht frei in der Wahl der Interaktion. Neben Sprache kann auch die visuelle Kommunikation eine möglichst natürliche und intuitive Interaktion zwischen Mensch und Maschine gewährleisten. Die Emotionserkennung auf Gesichtsbasis, gekoppelt mit einer Rückmeldung über einen computeranimierten Avatar, kann so zum Beispiel zur Verbesserung des Komforts, der Ergonomie und Verlässlichkeit von Fahrerassistenzsystemen erheblich beitragen [Bar03].

Auch andere Anwendungen, wie gesichtsbasierte Verfahren für automatisierte Zugangs- und Kontrollsysteme, befinden sich in der Entwicklungsphase. Mit Hilfe solcher Systeme könnte die Anmeldung an einem Personalcomputer künftig über eine berührungslose Identifikation und Verifikation statt über die Eingabe einer Kombination aus Benutzername und Kennwort realisiert sein. Weiter wird zur Steigerung der Sicherheit der zukünftigen zivilen Luftfahrt im europäischen Luftraum ein Verifikationssystem bei Boardingprozessen im EU-Projekt SAFEE[1] benötigt [Gau05]. Die Einführung digitaler Bilder für die biometrische Gesichtserkennung in Reisepässen unterstreicht das Zukunftspotenzial dieser Forschungsrichtung [Bun05].

Neben Sicherheitsanwendungen erwächst ein immer größeres Interesse am Thema Gesichtserkennung auch im Kontext von Multimediaanwendungen. Dabei erweist sich neben der inhaltlichen Suche von Objekten [Mul99b] eine automatische Indexierung multimedialer Daten auf Basis der darin vorkommenden Personen bzw. Gesichter für zahlreiche Anwendungen als wünschenswert. So ist die vollautomatische Erzeugung des Sitzungsprotokolls Ziel eines Projekts mit dem Namen *Multimodal Meeting Manager* (M4), welches neben der gesichtsbasierten Identifikation und Verifikation auch die Erkennung von Gemütszuständen involviert [Ren05]. Weitere Unternehmungen beschäftigen sich mit der automatischen Generierung audiovisueller Metadaten im Kontext von MPEG-7 (AGMA) [Koh05].

[1] Akronym für: Security of Aircraft in the Future European Environment

1.2 Beiträge und Gliederung der Arbeit

Hauptbestandteile der vorliegenden Arbeit sind die Entwicklung, Implementierung und qualitative Beurteilung von neuartigen, rechnergestützten Methoden zur Findung und Erkennung von Gesichtern sowie der Mimikerkennung. Aufgrund der hohen Komplexität der aufgeführten Einzelfunktionen werden die drei hieraus resultierenden Problemstellungen separat untersucht.

Im Einzelnen ist die Arbeit folgendermaßen unterteilt:

- **Kapitel 2.** Nach einem kurzen Umriss der Eigenschaften der visuellen Perzeption des Menschen wird auf die notwendigen Grundlagen zur technischen Realisierung der Bilderfassung eingegangen. Anschließend werden Verfahren und Theorien aus dem Themenbereich der Musterkennung erläutert, welche zur Lösung der vorgestellten Aufgaben erforderlich sind: die Hauptachsentransformation, künstliche neuronale Netze, Support Vektor Maschinen und Hidden Markov Modelle.

- **Kapitel 3.** Dieses Kapitel beschäftigt sich mit der Suche nach und dem Finden von Gesichtern in Standbildern und Videosequenzen. Nach der Vorstellung einer einheitlichen Repräsentation frontal aufgenommener Gesichter werden verschiedene Ansätze für deren Detektion in Grauwert- und Farbbildern näher beleuchtet und qualitativ bewertet. Nach der Einführung einer neuen, blickwinkelunabhängigen Detektion wird abschließend die Implementierung eines robusten Systems zur Gesichtsverfolgung in Videosequenzen vorgestellt, welche nach einer geeigneten Entzerrung sogar für omnidirektionale Kameraaufnahmen eingesetzt werden kann.

- **Kapitel 4.** Hauptgegenstand ist hier die Erkennung und Zuordnung von Gesichtern. In diesem Zusammenhang werden Schritte zur Vorverarbeitung und Merkmalsextraktion erläutert. Neben der frontalen Gesichtserkennung mit diskreten stochastischen Modellen, werden im Besonderen verschiedene neuartige Systeme zur Erkennung von Profilbildern beschrieben und beurteilt. Dabei ergeben sich Kombinationen aus probabilistischen sowie neuronalen Verfahren, auch hybride Systeme genannt.

- **Kapitel 5.** Im letzten Themenkreis wird das Erzeugen und das Personen unabhängige Erkennen dynamischer Mimiken mittels verschiedener Verfahren vorgestellt und ausgewertet. Hierzu werden zwei Datenbanken entworfen, eine mit überzogenen, eine andere mit möglichst spontanen Gesichtsausdrücken.

- **Kapitel 6.** Geschlossen wird die Arbeit mit einer Zusammenfassung der vorgestellten Systeme.

Kapitel 2

Grundlagen zur Bildverarbeitung und Mustererkennung

In diesem Kapitel sollen zunächst die wesentlichen Eigenschaften und Verarbeitungsschritte der menschlichen visuellen Perzeption und ihrer maschinellen Synthese in Digitalrechnern in knapper Form vergegenwärtigt werden, da sie für die Anforderungen an eine natürliche Mensch-Maschine-Kommunikation von fundamentaler Bedeutung sind [Ste93]: „Das beste bisher bekannte Bildverarbeitungs- und Mustererkennungssystem ist immer noch das menschliche Auge in Verbindung mit der Bildauswertung durch das Gehirn." Hieran schließt die Vorstellung der im Rahmen dieser Arbeit verwendeten Modelle und Methoden der Mustererkennung an, wie neuronale Netze, Support Vector Machines sowie Hidden Markov Modelle.

2.1 Die visuelle Wahrnehmung des Menschen

Obwohl die genauen Prozesse und Mechanismen der internen Informationsverarbeitung innerhalb des visuellen Kortexes bisher nur in Ansätzen verstanden wurden, konnte eine Vielzahl von charakteristischen Eigenschaften der optischen Sensorik durch psychovisuelle Experimente erforscht werden [Dem60, Hau94]. Die Adaptionseigenschaften, die Auflösung sowie das Zeitfrequenz- und Ortsfrequenzverhalten des menschlichen Auges sind im Wesentlichen bekannt. Im Gegensatz zur oben angedeuteten, semantisch höherwertigen Verarbeitung, der visuellen Perzeption im visuellen Kortex, ist auch die Art und Weise der Wandlung der optischen Reize in durch das Gehirn verarbeitbare Signale weitestgehend bekannt.

Das Auge des Menschen und einer Vielzahl weiterer Säugetiere besteht in seinen wesentlichen Komponenten aus einer Linse, dem Ziliarmuskel, der Iris und der Retina, auch Netzhaut genannt. Durch Kontraktion bzw. Relaxation des Ziliarmuskels kann die Brennweite der Linse akkomodiert werden, um eine scharfe zweidimensionale Abbildung des betrachteten Gegenstandes auf den empfindlichsten Bereich der Netzhaut, die Fovea, zu projizieren. Durch die Öffnungsgröße der Iris kann die Intensität des Lichteinfalls beeinflusst werden. Die Aufmerksamkeitssteuerung wird durch die relative Position gegenüber der Augenhöhle

über den Augenmuskel eingestellt, was seinerseits vom visuellen Kortex vorgenommen wird.

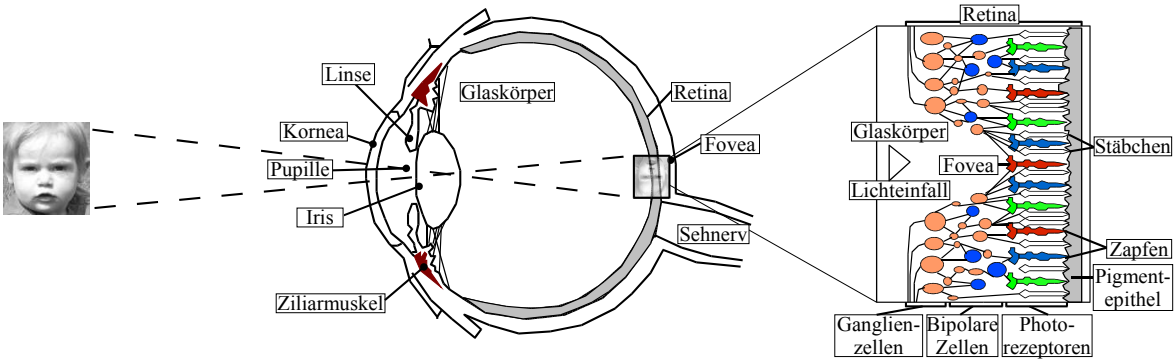


Abbildung 2.1: Querschnitt eines Auges nach [Bir99]

Wie im rechten Teil der Abbildung 2.1 veranschaulicht, besteht die Retina insgesamt aus drei Schichten. Dabei erfolgt die Wahrnehmung in der untersten Ebene über Photorezeptoren bzw. Lichtsinneszellen mit Stäbchen und Zapfen. Die photorezeptiven Stäbchen sind für das skotopische Sehen in der Dämmerung, auch bezeichnet als Helldunkel- oder Kontrastsehen verantwortlich [Ste93]. Die Zapfen decken die Funktion des photopischen Sehens in heller Umgebung, dem Farbsehen ab. Analog zur trichromatischen Theorie wird allgemein davon ausgegangen, dass es auf der Netzhaut jeweils Rezeptoren bzw. Zapfen für die Farben Rot, Grün und Blau gibt, welche jeweils auf verschiedene Wellenlängen des einfallenden Lichts reagieren [Hau94].

Die Informationen der beiden Sensortypen werden von einem neuralen Netz mit bipolaren Nervenzellen über ihre Eingänge, die Dendriten gebündelt. Über ihre auch Axonen genannten Ausgänge werden die verarbeiteten Information an die Dendriten der Ganglienzellen weitergeleitet und abermals gebündelt. Vereinfacht ausgedrückt, wird hier die Farbinformation von der RGB[1]-Darstellung in drei Gegenfarbkanäle transformiert. Des Weiteren wird bei der Bearbeitung in der Retina von einer adaptiven Kontrastoptimierung ausgegangen. Schließlich liegen die äußeren Reize in Form aufbereiteter Impulsfolgen am Sehnerv bzw. Nervus Opticus an, um in das zentrale Nervensystem eingeleitet zu werden. Durch die Informationsaufbereitung beider Augen ist überdies auch räumliches Sehen möglich. Im dahinter liegenden primären visuellen Kortex geschieht die Feinwahrnehmung von Mustern und Farben. Im sekundären visuellen Kortex werden Objekte lokalisiert und grobe Muster wahrgenommen. Außerdem wird von hier aus die Blickbewegung gesteuert. Verschiedene zeitlich aufeinander folgende Impulse können bei besten Lichtverhältnissen maximal bis zu einer Frequenz von ca. 60 Hertz unterschieden werden [Hau94].

Insgesamt ist der visuelle Apparat des Menschen mit den angeführten Mechanismen bekanntermaßen in der Lage sehr komplexe Aufgaben zu erledigen. So können üblicherweise mehrere hundert Gesichter in beliebig komplexen Szenarien gefunden und identifiziert werden. Auch ist eine sichere Interpretation von Gesichtsausdrücken und Emotionen unter schwierigen Beleuchtungsverhältnissen und Blickwinkeln möglich.

[1] Gängige Beschreibung für die Farbangabe durch die Intensitäten Rot, Grün und Blau

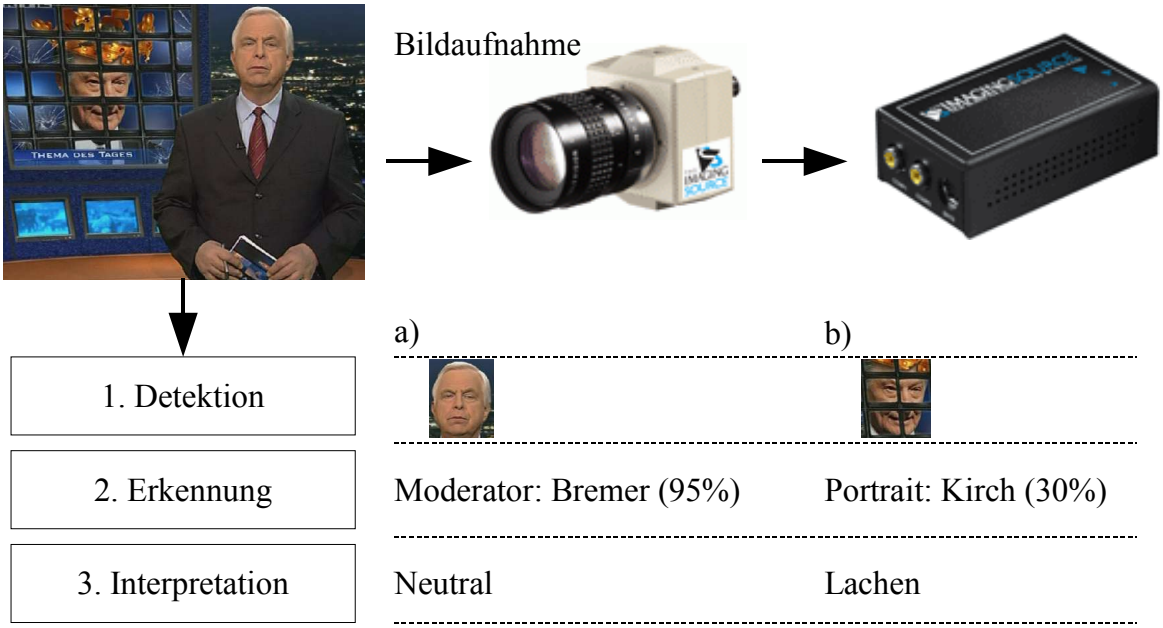


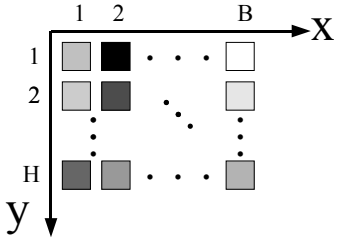
Abbildung 2.2: Die drei grundlegenden Themenkreise zur Verarbeitung des Musters Gesicht

Im Folgenden sollen technische Modelle und Systeme aus dem Bereich der Mustererkennung vorgestellt werden, um die drei in Abbildung 2.2 beschriebenen Grundfunktionen der visuellen Wahrnehmung nachzubilden. Zur vorgestellten Sensorik des menschlichen Auges wird zunächst eine äquivalente Realisierung zur digitalen Bildaufnahme vorgestellt. Die semantisch höherwertige Verarbeitung wird dabei mit Hilfe von Algorithmen der Mustererkennung angegangen.

2.2 Technische Sensoren und Repräsentation von Bildern

Obwohl beim menschlichen Auge eine Trennung von Bildaufnahme- und bearbeitung durch die darin befindlichen verschmolzenen neuronalen Netze nicht mehr eindeutig möglich ist, kann eine Aufspaltung der technischen Bilderfassung und Nachbearbeitung vorgenommen werden. So können beispielsweise die kontrastoptimierenden Eigenschaften der Retina während der Aufnahme durch eine nachgeschaltete Anwendung des Retinex-Algorithmus bei der Bilderfassung erfolgen [Fun04].

Analog zum Auge muss ein Bild der äußeren, dreidimensionalen Umwelt auch für die Verarbeitung in Digitalrechnern zunächst durch eine geeignete Optik in eine zweidimensionale Bildfunktion $f(x, y)$ überführt werden. Danach wird das einfallende und gebündelte Licht durch Rasterung und Quantisierung sowohl in räumlicher Hinsicht als auch im Wertebereich diskretisiert [Gon87]. Unter Annahme gleich großer Abtastfenster im Bildbereich, kann ein zweidimensionaler Bildausschnitt $I(x, y)$ mit der Breite B und der Höhe H in der Matrixschreibweise nach Gleichung 2.1 ausgedrückt werden.



$$I = \begin{bmatrix} I(1,1) & I(1,2) & \dots & I(1,B) \\ I(2,0) & I(2,2) & \dots & I(2,B) \\ \vdots & & & \\ I(H,0) & I(H,2) & \dots & I(H,B) \end{bmatrix} \tag{2.1}$$

Vereinfacht ausgedrückt entspricht die Intensität $I(x, y)$ eines Bildpunktes[2] der auf die Fotozelle eingefallenen Photonenenergie. In der Praxis werden zur Bilderfassung verschiedene Sensoren wie beispielsweise *Charged Coupled Devices* (CCD) oder neuerdings auch die preiswerteren *Complementary Metal Oxyd Semiconductor* Chips (CMOS) eingesetzt.

Bei Grauwertbildern entspricht $I(x, y)$ einem einzelnen Wert, bei Farbbildern einem Zahlentripel. Je nach Anzahl der Quantisierungsstufen der Grauwerte bzw. der Farben ist die Anzahl der Intensitätsstufen begrenzt. Ohne Beschränkung der Allgemeinheit soll im Folgenden von einer Auflösung von jeweils 8 Bit pro Kanal ausgegangen werden. Somit kann die normierte Intensität jeweils einen Wert zwischen 0 und $2^8 - 1 = 255$ annehmen. Diese Auflösung hat sich in der Praxis für nahezu alle Applikationen als ausreichend erwiesen.

Nach der trichromatischen Theorie erfolgt die Erfassung des eingefallenen Lichtes bei Farbbildern getrennt für die drei aktiven Primärfarben Rot, Grün und Blau [Hab91, Hau94]. Diese Sensorik ist in einer Näherung der spektral gewichteten Bestrahlungsstärke des menschlichen Auges mit seinen drei verschiedenen Zapfenarten nachempfunden. Durch vorgeschaltete optische Filter werden technisch nur die relevanten Spektralanteile auf die elektrischen Fotowandler gelassen. In Abbildung 2.3 sind typische Bereiche abgebildet, aus denen die Farbintensitäten bestimmt werden.

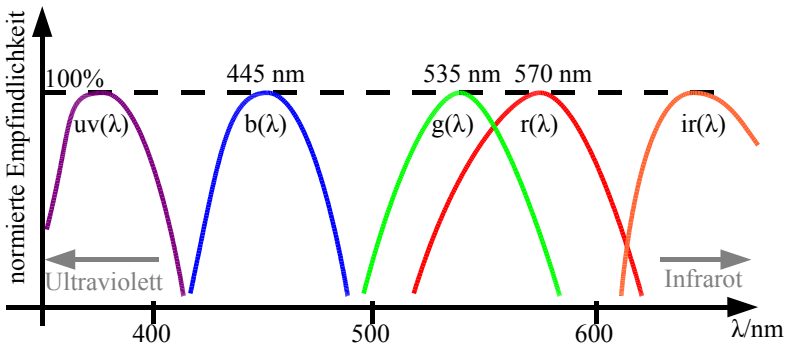


Abbildung 2.3: Normierte spektrale Empfindlichkeiten von Fotozellen

Neben den oben aufgeführten Farben ist es ebenfalls möglich, für das Auge nicht sichtbares Licht außerhalb des Wellenlängenbereichs zwischen 400 nm und 750 nm zu erfassen. Je nach verwendetem Sensor und seiner Empfindlichkeit können so auch ultraviolettes Licht (UV) (<400 nm) und Infrarotstrahlen (IR) (780 nm bis 1 mm) aufgezeichnet werden. Der Spektralbereich für Infrarot wird üblicherweise noch einmal unterteilt in die, dem sichtbaren Spektrum am nächsten liegende, kurzwellige IR-A Strahlung (780 nm - 1400 nm), die

[2] Auch Pixel genannt, aus dem Englischen für *Picture Element*

IR-B Strahlung (1400 nm - 3000 nm) und die IR-C Strahlung (3000 nm - 1 mm). Im Zusammenhang der Bildverarbeitung ist der IR-A Bereich zum Beispiel für Nachtsichtgeräte von großer praktischer Bedeutung.

Die Intensitäten $I_{r,g,b}$ für Rot, Grün und Blau am Ort (u, v) ergeben sich jeweils durch die Integration des einfallenden Lichtes $L(u, v, \lambda)$ multipliziert mit den frequenzabhängigen Empfindlichkeiten $S_{r,g,b}$ über die zu erfassenden Spektralbereiche λ.

$$I_{r,g,b}(u, v) = \int L(u, v, \lambda) \cdot S_{r,g,b}(\lambda) \mathrm{d}\lambda \tag{2.2}$$

Obwohl eine reale Bilderfassung rauschbehaftet ist, kann im Rahmen der Arbeit aufgrund des zu vernachlässigenden Einflusses von idealtypischen, rauschfreien Bildsensoren ausgegangen werden. Problematisch erweist sich die durch unterschiedlich normierte Empfindlichkeiten entstehende, fehlende Farbkonstantheit, welche in der Praxis durch einen Weißabgleich kompensiert werden muss. Diese Problematik wird später im Zusammenhang der Findung von Hautfarben im Abschnitt 3.2 weiter erörtert.

Die Umwandlung von Grauwert- in Farbmatrizen kann durch Duplizieren der Matrix erreicht werden. Anders herum kann ein Grauwert aus den drei Farbintensitäten nach Gleichung 2.3 bestimmt werden [Ste93].

$$I(x, y) = 0.3 \cdot I_r(x, y) + 0.59 \cdot I_g(x, y) + 0.11 \cdot I_b(x, y) \tag{2.3}$$

2.3 Automatische Mustererkennung

Oftmals ist es die Aufgabe der automatischen Mustererkennung, ein unbekanntes Muster dem ähnlichsten Repräsentanten einer Gruppe zuvor definierter oder unbekannter Beispiele zuzuordnen. Hierbei kann es sich bei den zu untersuchenden Mustern beispielsweise um Fingerabdrücke, Gesichtsbilder, Handschrift oder Sprachsignale handeln. Allgemein üblich wird der gesamte technische Prozess der Mustererkennung in die drei folgenden funktionalen Blöcke unterteilt [Jai00, Rus03a, Rig04a]: 1. Vorverarbeitung, 2. Merkmalsextraktion und 3. Klassifikation.

In Abbildung 2.4 ist diese Kette anhand eines Einzelzeichenerkenners exemplarisch dargestellt. Je nach Art, Dimensionalität und Beschaffenheit der zu verarbeitenden Muster[3] können die beiden ersten Module auch in kombinierter Form implementiert sein, so dass eine Trennung nicht zwangsläufig vorgenommen werden muss.

Vorverarbeitung Die Vorverarbeitung hat unabhängig vom untersuchten Muster die Aufgabe, eine kanonische Repräsentation der Informationen zu gewährleisten. Dies schließt üblicherweise eine gemeinsame Normalisierung des resultierenden Datenvektors $\mathbf{x}$ ein. Des Weiteren ist die Eliminierung von Störungen oftmals Bestandteil der Vorverarbeitung [Rus03b]. In der Praxis können hier das Entfernen von Rauschen oder das Ausschneiden relevanter

[3] Es soll im Folgenden davon ausgegangen werden, dass Mustervektoren unabhängig von ihrer ursprünglichen Dimension in Form von Spaltenvektoren dargestellt sind.

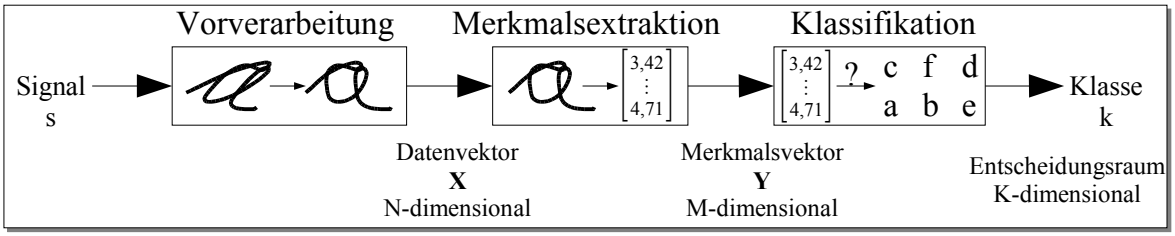


Abbildung 2.4: Schematischer Ablauf in der Mustererkennung

Bereiche von Interesse sein. Dazu gibt es im Besonderen für Bilder vielfältige Manipulationsmöglichkeiten. Im Beispiel wird das Signal $\mathbf{s}$ nach der Erfassung durch die Vorverarbeitung zur Normierung der Neigung in eine andere Repräsentationsebene der Dimension N gebracht.

Merkmalsextraktion Um die Charakteristik und die Trennbarkeit hervorzuheben, wird der vorliegende Datenvektor $\mathbf{x}$ in einen musterspezifischen Merkmalsvektor[4] $\mathbf{y}$ der Dimensionalität M überführt, im Beispiel durch das Symbol **a** repräsentiert. Mit der Merkmalsextraktion geht oftmals eine Datenreduktion durch Elimination von Irrelevanz einher [Rus03a], so dass $M << N$ gilt. Obwohl je nach Beschaffenheit des Quellsignals viele Möglichkeiten in Frage kommen, seien hier als mögliche Verfahren nur die Transformation in den Frequenzbereich (Fourier) oder die Zeit-Frequenz-Transformationen (Wavelet) genannt [Nie03].

Klassifikation Obwohl die beiden vorigen Punkte sicherlich zu einem Großteil an der Gesamtleistung eines Erkennungssystems verantwortlich sind, wird die eigentliche Zuordnung zwischen dem unbekannten Muster und den Referenzmustern erst bei der Klassifikation vorgenommen. In der nun parametrisch günstigeren Repräsentation des Musters kann die Zuordnung zu einer der bekannten k Klassen, respektive Modelle vorgenommen werden.

Grundsätzlich kann bei Klassifikationstechniken zwischen Schablonen basierten, statistischen, syntaktischen oder regelbasierten Ansätzen unterschieden werden [Jai00]. Im Folgenden werden die im Rahmen der Arbeit verwendeten numerischen Klassifikatoren resümiert: die Hauptachsentransformation, die Hidden Markov Modelle, die künstlichen neuronalen Netze, sowie Support Vector Machines[5]. Vor den komplexeren Klassifikationsverfahren wird zunächst ein repräsentativer Mustererkennungsprozess auf Basis eines Abstandsklassifikators vorgestellt.

2.4 Hauptachsentransformation und Klassifikation

Die Hauptachsentransformation[6], häufiger als *Principle Components Analysis* (PCA) bekannt, ist ein effizientes informationstheoretisch motiviertes Merkmalsextraktionsverfahren

[4] Engl.: Feature Vector

[5] Bei diesem Ausdruck hat sich die englische Fachbezeichnung durchgesetzt.

[6] Ebenfalls unter dem Namen Kahunen-Loeve-Transformation (KLT) geläufig

mit dem Hauptziel der drastischen Datenreduktion. Gleichzeitig werden die in den Ursprungsdaten vorhandenen Informationen maximal erhalten [Tur91b]. Dabei wird der originale hochdimensionale Datenraum mit Hilfe einer linearen Transformation in einen besser trennbaren, niederdimensionalen überführt, so dass die Muster anschließend in Richtung ihrer maximalen Streuung vorliegen. Im Verlauf der Arbeit wird die PCA zur Detektion und zur Erkennung von Gesichtern verwendet.

Geometrisch betrachtet werden zur Transformation zunächst alle Daten um den gemeinsamen Mittelwert $\bar{\mathbf{x}}$ verschoben, wie mit einer zweidimensionalen Verteilung in Abbildung 2.5 veranschaulicht. Anschließend wird eine Drehung mit einer Rotationsmatrix $\mathbf{U} = [\mathbf{u_1}, \ldots, \mathbf{u_N}]$ vorgenommen, welche durch die Hauptachsen bzw. Eigenvektoren gegeben ist. Beschrieben werden kann die Transformation formell durch:

$$\mathbf{y} = \mathbf{U}^T(\mathbf{x} - \bar{\mathbf{x}}). \tag{2.4}$$

Eine ausführliche Herleitung der Berechnung wird im Anhang B.1 vorgestellt. Je nach Kom-

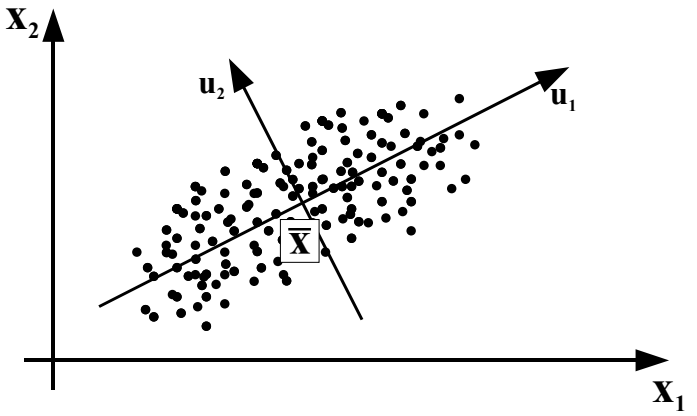


Abbildung 2.5: Geometrische Interpretation der PCA: 1. Verschiebung in den Mittelpunkt, 2. Drehung in Richtung der größten Streuung

pression und zugelassenem Fehler zwischen Original und reduziertem Vektor $\epsilon = \|\mathbf{x} - \tilde{\mathbf{x}}\|$, werden nur die ersten der ursprünglich N Projektionen y_i in Richtung der Eigenvektoren mit den K größten Eigenwerten λ_i berücksichtigt. Der verbleibende Rest ergibt sich somit zu

$$\epsilon = \Big\| \sum_{i=1}^{N} y_i \mathbf{u}_i - \sum_{i=1}^{K} y_i \mathbf{u}_i \Big\| = \Big\| \sum_{i=K+1}^{N} y_i \mathbf{u}_i \Big\| = \frac{1}{2} \sum_{i=K+1}^{N} \lambda_i \qquad ; K < N \tag{2.5}$$

Nach der Dimensionsreduktion können die reduzierten Mustervektoren $\mathbf{y}$ als Merkmalsvektoren interpretiert werden, da sie nun nur noch die wesentlichen Eigenschaften der Originaldaten verkörpern. Im Allgemeinen ist eine inhaltliche Interpretation der erhaltenen Merkmale nicht trivial. Ein unbekannter Vektor kann über die Suche nach dem nächsten Nachbarn[7] einem bekannten Referenzmuster $\mathbf{r}$ zugeordnet werden [Rig04a]. Werden mehrere Muster zu

[7] Engl. Nearest Neighbour Classifier

einem Referenzmuster zusammengefasst, kann dies bereits als überwachtes Training des Systems aufgefasst werden. Mathematisch wird der Minimale-Abstands-Klassifikator durch die Suche nach dem geringsten Euklidischen Abstand zwischen dem unbekannten Merkmalsvektor und allen I bekannten Referenzmustern formuliert. Die Referenzmuster, beispielsweise durch die Mittelwerte der Beispieldaten gegeben, repräsentieren die bekannten Musterklassen Ω. Die Euklidische Distanz zwischen zwei Vektoren berechnet sich über

$$d_E(\mathbf{y}, \mathbf{r}) = \sqrt{\sum_{i=1}^{K} (y_i - r_i)^2}. \tag{2.6}$$

Durch eine komponentenweise Gewichtung c kann man weitere statistische Eigenschaften der Daten in den Mustererkennungsprozess einbinden. Dies ist üblicherweise die Varianz σ bzw. der reziproke Eigenwert. Durch die Normierung wird dafür gesorgt, dass die Variationen in Richtung der verschiedenen Dimensionen als gleichmäßig signifikant bewertet werden. Dieses erweiterte Maß ist als Mahalanobis-Abstand bekannt, hier vereinfacht:

$$d_M(\mathbf{y}, \mathbf{r}) = \sqrt{\sum_{i=1}^{K} \sigma_i \cdot (y_i - r_i)^2} \qquad \text{bzw.} \qquad d_M(\mathbf{y}, \mathbf{r}) = \sqrt{\sum_{i=1}^{K} \frac{1}{\lambda_i} \cdot (y_i - r_i)^2}. \tag{2.7}$$

Die Suchvorschrift nach dem nächsten Nachbarn der I Klassen ist gegeben durch

$$\Omega_i = \underset{i \in I}{\operatorname{argmin}}\ d(\mathbf{y}, \mathbf{r_i}) = \underset{i \in I}{\operatorname{argmin}}\ \|\mathbf{y} - \mathbf{r_i}\|. \tag{2.8}$$

Demnach wird ein unbekannter Merkmalsvektor $\mathbf{y}$ der Klasse Ω_i zugeordnet, zu deren Referenzmuster $\mathbf{r_i}$ er den kleinsten Abstand hat. Eine Einschränkung dieses trivialen Klassifikationsverfahrens liegt in der Tatsache begründet, dass wie in Abbildung 2.6a gezeigt, nur lineare Klassengrenzen realisierbar sind. Abhilfe zur Verbesserung der Trennbarkeit der Klassen liegt beispielsweise in der Verwendung mehrerer Referenzmuster und in der Erweiterung zur Suche nach den K nächsten Nachbarn[8], was in Abbildung 2.6b dargestellt ist. Hierdurch können stückweise lineare Klassengrenzen auf Kosten eines stark gestiegenen Rechenbedarfes ermöglicht werden.

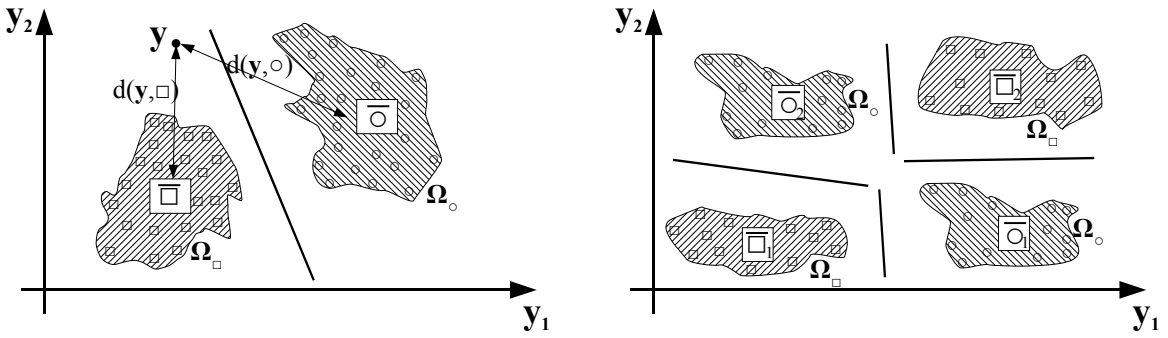
Im folgenden Kapitel werden künstliche neuronale Netzwerke mit flexibleren Klassengrenzen für komplexere Aufgabenstellungen vorgestellt.

2.5 Künstliche neuronale Netze

Künstliche neuronale Netzwerke[9] (NN) sind getaktete, konnektionistische, massiv parallel arbeitende Strukturen zur Verarbeitung und Klassifikation von Signalen bzw. Informationen [Lip89, Bis95, Pat96]. In den 80er Jahren galten NN als der Königsweg der künstlichen Intelligenz und sind bis heute für viele Aufgaben erfolgreich eingesetzt worden. Viele andere Verfahren können in NN-Strukturen überführt werden, anders herum ist der Einsatz von

[8] K-Nearest-Neighbor Regel

[9] Im Englischen: Artificial Neural Network



2.6a 2.6b

Abbildung 2.6: Nächster Nachbar Suche bei einem Zweiklassenproblem ($\Omega_\square$ / $\Omega_\circ$) mit a) zwei Referenzvektoren $\bar{\square}$ und $\bar{\circ}$, sowie b) mit 4 Referenzvektoren für die Klassenmittelpunkte $\bar{\square}_{1,2}$ und $\bar{\circ}_{1,2}$.

NN als Funktionsapproximierer anderer Klassifikatoren möglich [Jor97]. Je nach den verwendeten Topologien, Prozessorelementen und Lernverfahren kann ein umfassendes Spektrum möglicher Applikationen abgedeckt werden. So können NN beispielsweise zur Entfernung von Störungen bzw. Rauschen, als Assoziativspeicher oder für Klassifikationsaufgaben eingesetzt werden. Des Weiteren werden sie in der Diagnostik, Prädiktion von Ereignissen, zur Datenkomprimierung, für Steueraufgaben, zur multisensorischen Datenfusion sowie für das Erlernen beliebiger Abbildungsvorschriften eingesetzt. Eine umfassende Liste möglicher Einsatzgebiete technisch realisierter Lösungen befindet sich im Anhang B.3.1.

Wie in Kapitel 2.1 bereits angedeutet, wurde bei den ersten NN versucht, die Funktion biologischer Nervensysteme, insbesondere die der Retina von Säugetieren, durch das sogenannte Perzeptron nachzuahmen. In die heutigen Realisierungen gehen jedoch nur noch die grundlegenden Ideen ein. Zwar konnte die angestrebte hochgradige Dichte, Parallelität und Vernetzung der einzelnen informationsverarbeitenden Knoten bzw. Neuronen bis heute nicht annähernd erreicht werden, die Leistungsfähigkeit existierender technischer Systeme ist dennoch beachtlich.

2.5.1 Mathematische Formulierung künstlicher Neuronen

Zur Realisierung der oben genannten Funktionalitäten werden NN mit mathematisch beschreibbaren Verarbeitungsknoten verwendet [Rig94]. Analog zu ihrer biologischen Vorlage bestehen die rechnergestützten Neuronenmodelle, auch Prozessorelemente genannt, aus einem Eingangswerte beschreibenden Vektor $\mathbf{x}$, einem Gewichtungsvektor $\mathbf{w}$, einem Bias-Wert w_0, einer Propagierungsfunktion $G(x)$, sowie einer Aktivierungsfunktion $F(a) = F(G(\mathbf{x}))$ und zuletzt einem skalaren Ausgang y. Eine solche Verarbeitungseinheit ist in Abbildung 2.7 dargestellt.

Für eine vereinfachte Schreibweise wird der Bias-Wert w_0 in den Gewichtungsvektor $\mathbf{w}$ einbezogen: $\mathbf{w}^T = [w_0, w_1, \ldots, w_N]$. Konsequenterweise muss vor dem ersten Element des

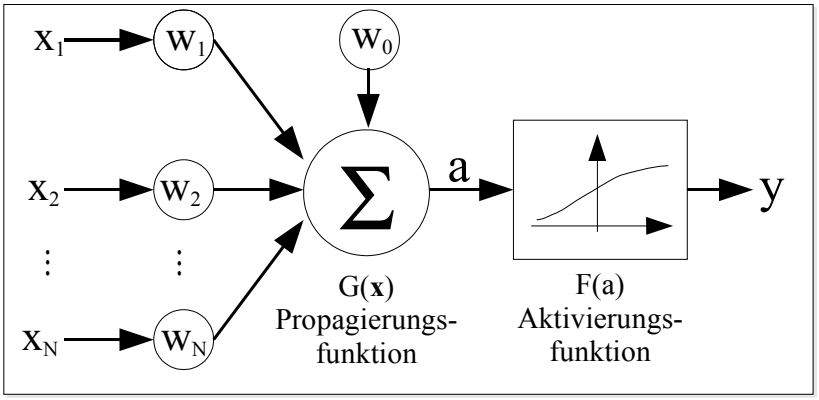


Abbildung 2.7: Mathematisches Modell eines künstlichen Neurons

Eingangsvektors der konstante Wert $x_0 = 1$ eingeführt werden, so dass fortan gilt $\mathbf{x}^T = [x_0 = 1, x_1, \ldots, x_N]$.

Obwohl theoretisch auch andere Verknüpfungen möglich sind, wird die Propagierungsfunktion $G(\mathbf{x})$ allgemein üblich durch die gewichtete Summe der Eingänge beschrieben:

$$a = G(\mathbf{x}) = \mathbf{w}^T \cdot \mathbf{x} \qquad \text{bzw.} \qquad G(\mathbf{x}) = \sum_{i=1}^{N} w_i \cdot x_i + w_0 = \sum_{i=0}^{N} w_i \cdot x_i \tag{2.9}$$

Das Ergebnis von $G(\mathbf{x})$ wird auch als Aktivierung a eines Neurons bezeichnet. Für die zugehörige Aktivierungsfunktion, auch Transferfunktion genannt, gibt es je nach Anwendung wieder verschiedene Optionen. Allgemein ist der Wert des Neuronenausgangs y gegeben durch:

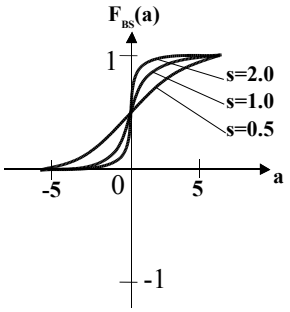
$$y = F(G(\mathbf{x})) = F(a) = F\left(\sum_{i=0}^{N} w_i \cdot x_i\right). \tag{2.10}$$

Je nach Anwendung und Art des Signals, wie kontinuierlich oder binär, kann es sich beispielsweise um einen symmetrischen harten Begrenzer, eine symmetrische oder gesättigte lineare Funktion, eine Sigmoid- oder auch die Tangenshyperbolikusfunktion nach Abbildung 2.8 handeln. Es existieren somit sowohl stetige als auch nicht stetig differenzierbare Funktionen, deren Wahl vom verwendeten Trainingsalgorithmus sowie der konkreten Aufgabenstellung abhängt.

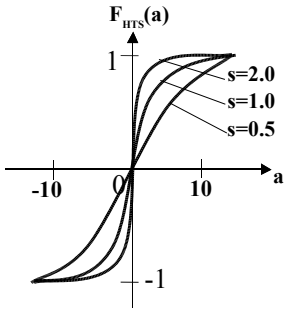
2.5.2 Netzwerktopologien

Durch die Verknüpfung der Ein- und Ausgänge werden die einzelnen Neuronen zu einem Gesamtsystem mit gewünschtem internen Informationsfluss verschaltet. Die Schicht bzw. Ebene, an welche die externen Informationen angelegt werden, nennt sich in der Literatur durchgängig Eingangsschicht. Die Neuronen, deren Ausgänge von außen gemessen und ausgewertet werden, sind unter dem Begriff Ausgabeschicht zusammengefasst.

Neuronen, die weder direkte Ein- noch Ausgänge haben, werden in versteckten Zwischenschichten, sogenannten *Hidden Layern* zusammengefasst. Ist keine verdeckte Schicht



$$F_{\text{BS}}(a) = \frac{1}{1 + e^{-as}} \qquad (2.11)$$



$$F_{\text{HTS}}(s) = \frac{1 - e^{-2as}}{1 + e^{-2as}} \qquad (2.12)$$

Abbildung 2.8: Häufig verwendete Aktivierungsfunktionen: links die Sigmoidfunktion, rechts der hyperbolische Tangens Sigmoid

vorhanden, spricht man von einem einschichtigen Netzwerk, da die Neuronen der Eingangsschicht üblicherweise keine signalverarbeitenden Funktionen vollführen. Falls weitere Schichten zwischen der Ein- und Ausgabeschicht vorhanden sind, wird die vorliegende Architektur als mehrschichtig bzw. als *Multi Layer Perzeptron* (MLP) bezeichnet. Die Schichtigkeit ist jeweils durch die Anzahl der Ebenen mit adaptierbaren Gewichten gegeben.

Je nachdem, ob die Ausbreitungsrichtung streng von den Ausgabeknoten einer Schicht auf die Eingänge der darauf folgenden Schichten gerichtet sind, kann zwischen vorwärts-[10] und rückwärtsgerichteten bzw. rekurrenten oder Feedback-Netzen unterschieden werden. In Abbildung 2.9 sind die zur Problemlösung häufig verwendeten ein- und zweidimensionalen vorwärtsgerichteten MLP skizziert, welche ihre Stärken hauptsächlich im Bereich von Vorhersagen, Klassifikation und Funktionsannäherung haben.

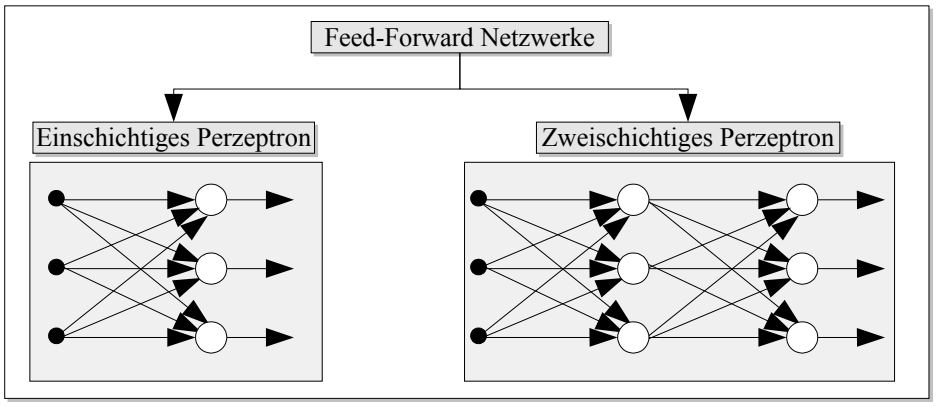


Abbildung 2.9: Beispielhafte vorwärtsgerichtete NN-Architekturen

[10] Engl. Feed-Forward

2.5.3 Lernverfahren

In der Praxis wird die Arbeit mit NN in zwei Phasen unterteilt: eine Trainingsphase und eine anschließende Gebrauchs- bzw. Anwendungsphase. Vor der Parameterbestimmung werden die Gewichtungen üblicherweise mit zuvor definierten oder zufälligen Startwerten initialisiert.

Vor der Wahl eines Lernprozesses muss zunächst eine Modellumgebung definiert werden, in welcher die verfügbaren Informationen definiert sind. Ferner muss eine Lernregel vorgegeben werden, die angibt wie die Gewichtungen beeinflusst werden. Ein Lernalgorithmus beschreibt die Prozedur, in der Lernregeln zur Anpassung der Gewichte verwendet werden.

Bei den sogenannten Lernparadigmen kann zwischen dem überwachten und unüberwachten, sowie dem hybriden Lernen unterschieden werden. Im Zusammenhang mit NN bedeutet Lernen die Bestimmung der freien Netzwerkparameter bzw. der optimalen Gewichtungen w_{ji} mittels einer Menge an Lernbeispielen. Eine aus diesem Sachverhalt ableitbare Eigenschaft ist, dass das Verhalten eines NN nur für die gelernten Musterdaten sowie die folgenden Testreihen exakt angegeben werden kann. Über das Verhalten bei beliebigen unbekannten Eingabemustern lassen sich lediglich statistische Angaben machen. Beim überwachten Lernen sind die zu den Eingabemustern gehörenden Ausgabemuster bzw. Funktionen bekannt. Beim unüberwachten Lernen existiert üblicherweise keine Rückmeldung über den gemachten Lernerfolg. Beim bestärkenden Lernen wird dem NN während der Trainingsphase von außen lediglich mitgeteilt, ob das Ergebnis richtig oder falsch ist.

Grundsätzlich kann zwischen vier Lernregeln unterschieden werden, wobei mit jeder Strategie üblicherweise eine bestimmte Netzstruktur assoziiert wird [Jai96]. So existieren neben dem Lernen durch Fehlerverbesserung noch die Hebbsche Regel, das konkurrierende sowie das Boltzmannsche Lernen.

Im Fall der Fehlerminimierung wird dem NN zu jedem Eingabemuster $\mathbf{x}^n$ ein passendes Ausgabemuster $\mathbf{t}^n$ präsentiert. Durch eine fortschreitende Parameteradaption kann eine Minimierung des auftretenden Fehlers zwischen den Sollwerten und den aktuellen Ausgaben erfolgen. Die Fehlerfunktion E eines MLP ist durch die Summe der quadrierten Differenzen zwischen den Zielwerten t_j und den aktuellen Ausgaben o_j aller j Neuronen für ein Wertepaar in der Ausgabeschicht gegeben. Zur Bestimmung des Gesamtfehlers E_{ges} werden die Einzelfehler $E = \frac{1}{2}\sum_j (t_j - o_j)^2$ aufsummiert.

$$E_{\text{ges}} = \sum_n E_n = \frac{1}{2}\sum_n \sum_j (t_j - o_j)^2 \tag{2.13}$$

Training durch Backpropagation Für Topologien mit differenzierbarer Aktivierungsfunktion existiert eine leistungsstarke, auf den Rechen- und Speicheraufwand bezogen, effiziente Methode zur Bestimmung zumindest lokal optimaler Gewichte und Bias-Werte. Dieser Algorithmus wird Backpropagation oder verallgemeinerte Delta-Regel genannt und ist aus der Perzeptronen Lernregel abgeleitet [Rig94]. Beim Backpropagation ist es Ziel, die Fehlerfunktion über ein Gradientenabstiegsverfahen mit fester Schrittweite zu minimieren [Bis95, Rig96].

Da die Fehlerfunktion nach Gleichung 2.13 offensichtlich von den zu optimierenden Parametern w_{ji} abhängig ist, müssen die Gewichte derart angepasst werden, dass der verbleibende Fehler gegen ein Minimum konvergiert[11], idealerweise dem globalen, in der Realität oftmals jedoch nur einem lokalen. Der Lösungsansatz des Verfahrens wird im Anhang B.3.2 detaillierter vorgestellt.

Beim Training kann unterschieden werden, ob die Gewichte nach jedem einzelnen Muster aktualisiert werden[12], oder ob erst alle Beispiele durchlaufen werden[13]. Alle Trainingsschritte in einer Epoche bzw. Iteration werden auf diese Weise mehrmals wiederholt, bis sich stabile Gewichte herauskristallisiert haben.

Elastisches Backpropagation Mögliche Probleme beim Training nach dem oben beschriebenen Error-Backpropagation sind die eventuell langsame Konvergenz in Abhängigkeit der Lernrate η, das Verbleiben in lokalen Minima, sowie das Auftreten numerischer Instabilitäten. Durch diesen Sachverhalt findet das Training nach dem reinen Backpropagationverfahren in der Praxis eher selten Anwendung. Statt dessen bedient man sich beispielsweise der Alternative über die Erweiterung mit einem Momentum-Term [Rig96, Ger04]. Hierbei behält die Gewichtsänderung in jeder Iteration einen Rest der Vorhergehenden bei. Die Änderungsregel B.7 wird somit um das Moment $\gamma[w_{ji}^n - w_{ji}^{n-1}]$ erweitert.

$$w_{ji}^{n+1} = w_{ji}^n + \Delta w_{ji} + \gamma[w_{ji}^n - w_{ji}^{n-1}] \tag{2.14}$$

Der Wertebereich von γ liegt zwischen 0 und 1. Durch diese Erweiterung kann die Konvergenz beschleunigt, sowie eine mögliche Oszillation um das Minimum verhindert werden.

In der Praxis wird zudem ein von Riedmiller und Braun [Rie93] eingeführter adaptiver Lernalgorithmus auf Basis eines optimierten Gradientenabstiegsverfahren verwendet. Hierzu wird das bekannte Propagationverfahren zum Resilient Propagation-Algorithmus (RPROP)[14] mit dem Ziel erweitert, die Lernrate dynamisch an den aktuellen Verlauf der Fehleroberfläche anzupassen. Im Gegensatz zu herkömmlichen Gradientenabstiegsverfahren wird beim RPROP jedoch nicht der Betrag der Gradienten, sondern nur das Vorzeichen $\left(\frac{\partial E}{\partial w_{ji}}\right)$ ausgewertet. Für die Anpassung wird somit lediglich geprüft, ob zwischen der vorhergehenden und der aktuellen Trainingsiteration ein Vorzeichenwechsel des Gradienten stattgefunden hat. Der Algorithmus wird im Anhang B.3.3 ausführlicher vorgestellt.

2.5.4 Lernverhalten neuronaler Netze

Durch die adaptive Änderung der Sprungweite Δw_n stellt sich durch den RPROP-Algorithmus in der Regel eine frühere Konvergenz als beim nicht adaptiven Gradientenabstiegsverfahren ein, was in Abbildung 2.10 schematisch dargestellt ist.

[11] Least-Mean-Square (LMS) Verfahren
[12] Online-Learning
[13] Batch-Learning
[14] Resilient PROPagation steht für elastische Fortpflanzung

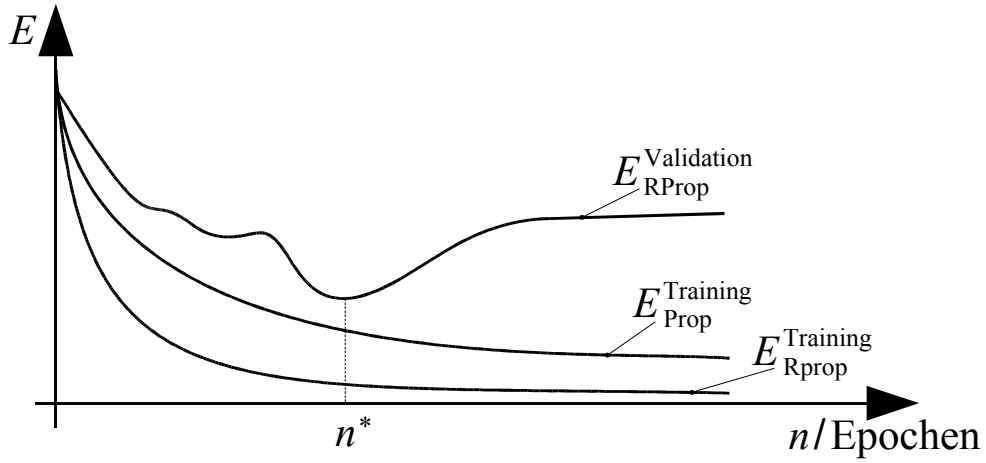
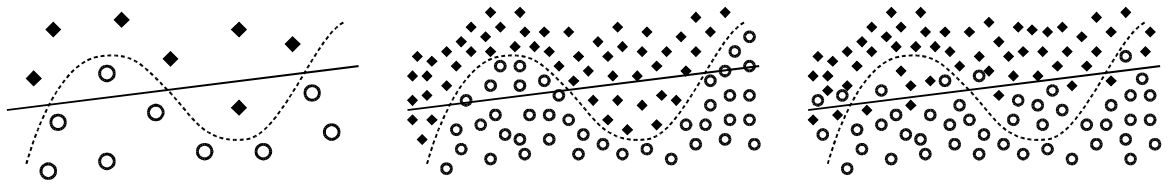


Abbildung 2.10: Trainingsfehler nach regulärem Backpropagation und RPROP

Gewöhnlich stellt sich durch das Kriterium der Fehlerminimierung nach dem Einschwingen ein monoton fallender Verlauf des Trainingsfehlers ein. Betrachtet man neben den Trainingsdaten noch einen weiteren Datensatz mit disjunkten, also unbekannten Beispielen, so gibt die Fehlerfunktion auf diese Validierungsdaten wichtige charakteristische Eigenschaften über das Lernverhalten. Hiermit kann ermittelt werden, wie viele Epochen bzw. Iterationen ein Training idealerweise andauern sollte, um die geforderte Funktionalität zu generalisieren. Ein ideales Kriterium für den Abbruch des Trainings ist beispielsweise durch das Minimum des Fehlers auf den Validierungsdaten gegeben [Bis95, Pat96, Ger04]. Betrachtet man dazu Abbildung 2.10, so läge der Zeitpunkt der optimalen Generalisierung bei Epoche n^*. Bei zu vielen Trainingsiterationen, der Überanpassung bzw. dem Overfitting, passt sich das System zu gut an die Beispieldaten an, lernt sie praktisch auswendig und modelliert eventuell vorhandenes Messrauschen mit.

Neben dem Trainings- und Validierungsdatensatz kommt für die Gebrauchsphase noch ein dritter Korpus hinzu: das Testset, bezüglich dessen Verteilung der Fehler minimal sein soll. Hieran lässt sich das Dilemma der Über- und Unteranpassung mit Hilfe eines Zweiklassenproblems und möglichen Klassengrenzen wie in Abbildung 2.11 verdeutlichen. Da die Testdaten im Vorfeld nicht bekannt sind, ist nicht klar, welche der beiden Trennlinien den kleineren Fehler aufweisen wird: die durchgezeichnete Gerade eines einfachen Perzeptrons oder die gestrichelte Trennlinie eines MLP.



2.11a 2.11b 2.11c

Abbildung 2.11: Beispielhafte Klassengrenzen eines Perzeptrons (durchgezogen) und eines MLP (getrichelt) für verschiedene Datensätze

Die größeren Symbole im linken Bereich der Abbildung repräsentieren die Trainingsdaten zur Bestimmung der Trennfunktion für die beiden Klassen ◯ und ⧫. Im skizzierten Fall wäre zunächst die gestrichelte Kurve vorzuziehen, da sie auf den Trainingsdaten ein optimales Klassifikationsergebnis gewährleistet. Bei der linearen Trennung bleiben beim Training wenige Restfehler.

Falls die tatsächliche Verteilung der Testdaten wie im mittleren Teil gegeben ist, würde man durch Wahl der linearen Klassengrenze von einer Unteranpassung sprechen. Nach einer Verteilung wie im rechten Teil würde man bei einer Wahl der gestrichelten Trennlinie von einer Überanpassung sprechen. Durch die Wahl eines möglichst großen und repräsentativen Trainingskorpus und einer Restriktion der Komplexität der Klassengrenzen kann dieses Dilemma jedoch weitgehend umgangen werden.

Neben dem geforderten Lernverhalten kann zudem gefolgert werden, dass für die Generalisierungseigenschaften auch die Netzwerktopologie von entscheidender Wichtigkeit ist. Falls zu viele Einheiten und somit zu viele freie Parameter vorhanden sind, wird ein NN einen gegebenen Datensatz während des Trainings eher auswendig lernen. Im anderen Extremfall, bei zu wenigen Einheiten also, wird das Netz keine ausreichenden Generalisierungseigenschaften aufweisen können. Die Wahl der Anzahl von verdeckten Ebenen und den darin enthaltenen Neuronen ist nicht trivial. Für eine Optimierung müssen üblicherweise mehrere Versuchsreihen durchlaufen und das Verhalten bezüglich des Fehlers sowohl auf den Trainings-, als auch auf den Validierungsdaten ermittelt werden.

2.5.5 Klassifikation mit neuronalen Netzen

In den implementierten Applikationen werden NN sowohl zur Abbildung von Funktionen als auch als Klassifikatoren für Ein- und Mehrklassenprobleme verwendet, für welche sie aufgrund ihrer diskriminativen Eigenschaften besonders geeignet sind. Ein entscheidender Vorteil bei der Klassifikation mit mehrschichtigen MLP ist, dass im Gegensatz zu den in Kapitel 2.4 vorgestellten Distanzklassifikatoren nicht nur lineare Klassengrenzen erfasst werden können. So sind in Abbildung 2.12 drei verschiedene Netztopologien mit möglichen Trennungseigenschaften für das *Exclusive-Oder* Problem der Booleschen Algebra und ein fiktives Weiteres dargestellt. Dabei werden die Klassen der zweidimensionalen Verteilung mit schwarzen und weißen Elementen repräsentiert.

Bereits durch die Verwendung eines einschichtigen Netzes mit Hard-Limitern kann eine lineare Hyperebene im Musterraum aufgespannt werden. Durch die Hinzunahme einer zweiten Schicht ist eine Und-Verknüpfung von Halbräumen möglich, wodurch komplexere, stückweise lineare Klassengrenzen ermöglicht werden. Durch eine dritte Schicht können diese Bereiche dann durch eine weitere Oder-Operation zu beliebig geclusterten Hyperbereichen verknüpft werden. Durch die Wahl von sigmoidalen Propagierungsfunktionen können die entstehenden Grenzen darüber hinaus weicher geartet sein.

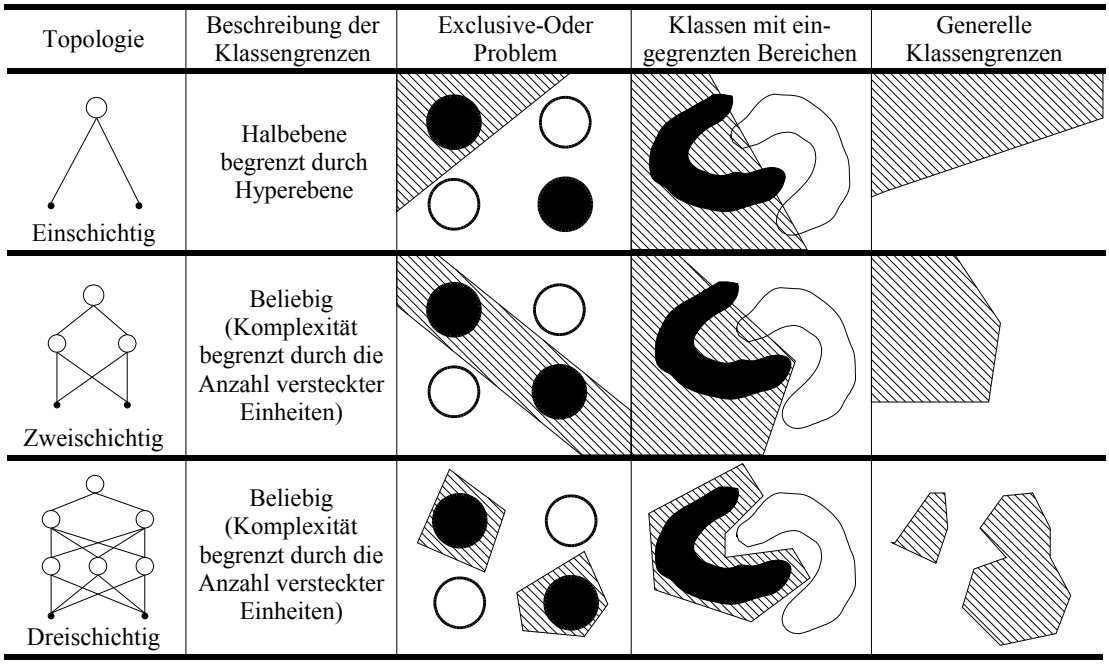


Abbildung 2.12: Geometrische Interpretation der Separierbarkeit bei verschiedenen MLP-Topologien nach Lippmann [Lip87]

2.6 Klassifikation mit Support Vektor Maschinen

In den letzten Jahren hat die Klassifikation mit Support Vector Machines[15] (SVM) zunehmend an Bedeutung gewonnen [Bur98, Sch02]. Hierbei müssen im Vorfeld zunächst keine statistischen Annahmen über die Verteilung der Trainingsdaten gemacht werden. Die Optimierung der Generalisierungseigenschaften und die Vermeidung des bereits in 2.5.3 vorgestellten Overfitting Dilemmas [Mul01a] durch Minimierung des empirischen Risikos mit Hilfe der sogenannten Vapnik-Chervonenkis-Dimension ist vielmehr eine wesentliche Eigenschaft von SVM [Nie03]. SVM haben sich aufgrund ihrer Eigenschaften, im Besonderen bei der Klassifikation von Zweiklassenproblemen hervorgetan. Anwendungsbeispiele in denen sich SVM gegenüber anderen Ansätzen als besonders leistungsfähig erwiesen haben, umfassen beispielsweise die Gesichtsdetektion, die Einzelzeichenerkennung sowie die inhaltliche Suche in Dokumenten und die Textkategorisierung [Hea98]. Prinzipiell können SVM auch als Sonderform von NN interpretiert werden [Cor95]. Ein wesentlicher Vorteil für die praktische Anwendung ist, dass in der Klassifikationsphase alle Berechnungen durch die Verwendung von Skalarprodukten ausgedrückt werden können. Die wesentlichen funktionalen Eigenschaften von SVM sind in den nächsten drei Unterpunkten zusammengefasst.

Lineare Separierbarkeit Analog zu den erwähnten Perzeptronen ist es Ziel des SVM-Trainings, möglichst optimale Flächen zur Trennung der Muster einer endlichen Stichprobe

[15] In dieser Arbeit wurde der englische Terminus beibehalten, da der Ausdruck Stützvektor Maschine/Methode eher ungebräuchlich ist.

$\mathbf{X}$ zu finden. Die Merkmalsvektoren $\mathbf{x}_m$ werden dabei zusammen mit dem Label y_m ihrer Klassenzugehörigkeit repräsentiert.

$$\mathbf{X} = \{(\mathbf{x}_m, y_m), m = 1, \ldots, M\} \text{ mit } \mathbf{x}_m \in \mathbb{R}^n \text{ und } y_m \in \{-1, 1\}\} \tag{2.15}$$

Die Klassenzugehörigkeit eines positiven Stichprobenbeispiels der Klasse Ω_1 ist durch das Label $y_m = +1$ gekennzeichnet, $y_m = -1$ für Klasse Ω_2 andernfalls. Zunächst wird davon ausgegangen, dass die beiden Klassen im Falle der linearen Separierbarkeit durch ebene Trennflächen ideal geteilt werden können. Eine Menge von Trennfunktionen ist durch orientierte Hyperebenen gegeben, welche durch Gleichung 2.16 mit dem senkrecht zur Fläche stehenden Richtungsvektor $\mathbf{w}$ und dem Abstand b der Fläche zum Ursprung des Koordinatensystems definiert sind.

$$d_{\mathbf{w},b}(\mathbf{x}) = \mathbf{x}^T\mathbf{w} + b \tag{2.16}$$

Mit dem Abstand $d_{\mathbf{w},b}$ eines Punktes zur Ebene kann entschieden werden, ob ein Mustervektor auf der positiven oder negativen Seite oder genau auf der Hyperebene liegt. Für eine optimale Hyperebene, welche die positiven Beispiele exakt von den negativen trennt, sollen für alle Muster der Stichprobe $x_m \in \mathbf{X}$ die Gleichungen 2.17-2.19 gelten.

$$\mathbf{x}_m^T\mathbf{w} + b \geq +1, \text{ wenn } y_m = +1 \tag{2.17}$$

$$\mathbf{x}_m^T\mathbf{w} + b \leq -1, \text{ wenn } y_m = -1 \tag{2.18}$$

$$y_m \left(\mathbf{x}_m^T\mathbf{w} + b\right) \geq 1, \forall\, \mathbf{x} \in \mathbf{X} \tag{2.19}$$

Der Abstand eines Punktes von der Ebene ist positiv, wenn der Punkt in Richtung der Normalen der Ebene liegt. Die Ebenenparameter $\mathbf{w}$ und b müssen als Konsequenz nun so skaliert werden, dass für die am nächsten zur Ebene liegenden Punkte $\mathbf{x}'$ die Beziehung $|\mathbf{x}'^T\mathbf{w} + b| = 1$ gilt. Alle Punkte, für die nach den Gleichungen 2.17 und 2.18 das Gleichheitszeichen gilt, liegen also auf einer der beiden Ebenen $\mathcal{H}^1$ und $\mathcal{H}^2$ parallel zur gesuchten Trennebene, entweder auf der positiven oder negativen Seite. Zwischen diesen beiden Ebenen befindet sich ein $\frac{2}{||\mathbf{w}||}$ breiter Korridor, in dem sich keine Vektoren befinden. Ein sinnvolles Optimalitätskriterium für eine sichere Trennung ist die Maximierung des Abstandes der beiden Trennebenen zueinander. $\mathcal{H}^*$ liegt somit exakt in der Mitte zwischen $\mathcal{H}^1$ und $\mathcal{H}^2$. Die Punkte bzw. Vektoren, welche auf den Trennebenen liegen, werden als die charakteristischen *Support Vektoren*[16] bezeichnet. Der Abstand der Ebenen zueinander wird in der Literatur kanonisch als *Margin*, die Trennebene dazwischen als *Maximum Margin Hyperebene* bezeichnet. Der linke Teil in Abbildung 2.13 fasst die oben genannten Eigenschaften für ein zweidimensionales Beispiel zusammen. Hierin sind die Stützvektoren für die positiven ⧫ und negativen ∘ Exemplare jeweils mit einem Kreis ◯ umrandet.

Alle weiteren Vektoren der gegebenen Stichprobenverteilung außer den Stützvektoren sind für die eigentliche Klassifikation im Folgenden irrelevant. Zur Lösung des Optimierungsproblems werden Lagrangesche Multiplikatoren eingeführt, welche zusammen mit den sogenannten notwendigen und hinreichenden Karush-Kuhn-Tucker Bedingungen gelöst werden können, welche ausführlich in der Literatur beschrieben sind [Bur98, Sch02].

[16] Im Sinne von Stützvektoren

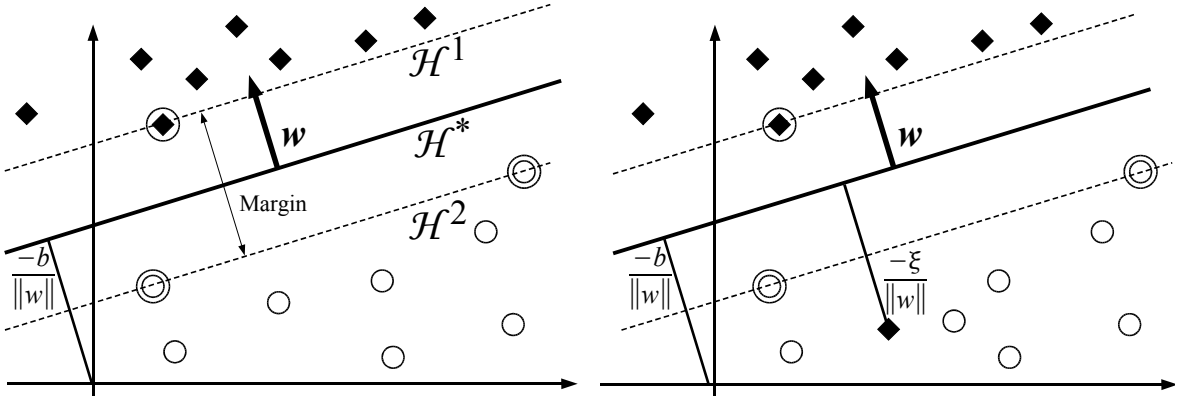


Abbildung 2.13: Maximum Margin Hyperebene sowie Stützvektoren, links für eine linear, rechts für eine nicht linear separierbare Verteilung

Nach der Berechnung der idealen Hyperebene können unbekannte Muster in der darauf folgenden Anwendungs- oder Testphase über die Entscheidungsregel 2.20 getrennt werden. Da in praktischen Anwendungen, z.B. aufgrund von Rauschen, nicht von einer idealen Trennung der beiden Klassen ausgegangen werden kann, muss der vorgestellte ideale Klassifikationsansatz auf nicht linear separierbare Stichproben verallgemeinert werden.

$$\mathbf{x} \in \begin{cases} \Omega_1 & : d_{\mathbf{w},b}(\mathbf{x}) \geq 0 \\ \Omega_2 & : d_{\mathbf{w},b}(\mathbf{x}) < 0 \end{cases} \tag{2.20}$$

Erweiterung auf nicht ideal trennbare Daten Zur Erweiterung wird folgende Modifikation vorgenommen. Zur Einhaltung obiger Gleichungen und Bedingungen werden zusätzlich zu den Labeln y_m sogenannte Schlupfvariablen $\xi_m \geq 0$ zur Abschwächung eines Fehlers eingeführt. Die Gleichungen 2.17 bis 2.19 werden somit erweitert zu:

$$\mathbf{x}_m^T \mathbf{w} + b \geq +1 - \xi_m, \quad \text{wenn } y_m = +1 \tag{2.21}$$

$$\mathbf{x}_m^T \mathbf{w} + b \leq -1 + \xi_m, \quad \text{wenn } y_m = -1 \tag{2.22}$$

$$y_m \left(\mathbf{x}_m^T \mathbf{w} + b \right) \geq 1 - \xi_m, \quad \forall\, \mathbf{x}_m \in \Psi \tag{2.23}$$

$$\xi_m \geq 0, \quad m = 1, \ldots, N \tag{2.24}$$

Dieses neue Klassifikationsproblem[17] ist rechts in Abbildung 2.13 veranschaulicht. Durch die Einführung der abschwächenden Schlupfvariablen ist es nun möglich, dass Vektoren auf der falschen Seite der Trennebene liegen können, ohne den etablierten Ansatz fundamental neu formulieren zu müssen. Die Trennebene kann unter Berücksichtigung einer Fehlerabschätzung $\sum_m \xi_m$ analog zu der oben erwähnten Lösungsmethode mit zusätzlichen Lagrange-Multiplikatoren optimiert werden. Für eine detaillierte Beschreibung sei wieder auf die Literatur verwiesen [Bur98, Sch02].

[17] Auch als Soft Margin Binary Classifier bezeichnet

Verallgemeinerung auf linear nicht lösbare Problemstellungen In einem dritten Schritt soll die Beschränkung der Trennfunktion auf Hyperebenen verallgemeinert werden. Ziel hierbei ist es, einen linear nicht trennbaren Eingaberaum der Dimension n, in einen höherdimensionalen Merkmalsraum der Dimension $\tilde{n}$ so zu transformieren, dass die Muster dann wieder linear trennbar sind: $\Phi : \mathbb{R}^n \rightarrow \mathbb{R}^{\tilde{n}}$, wobei $n < \tilde{n}$ gilt. Die Bestimmung der Hyperebene kann dann wie bekannt vorgenommen werden. In Abbildung 2.14 ist die grundlegende Idee skizziert.

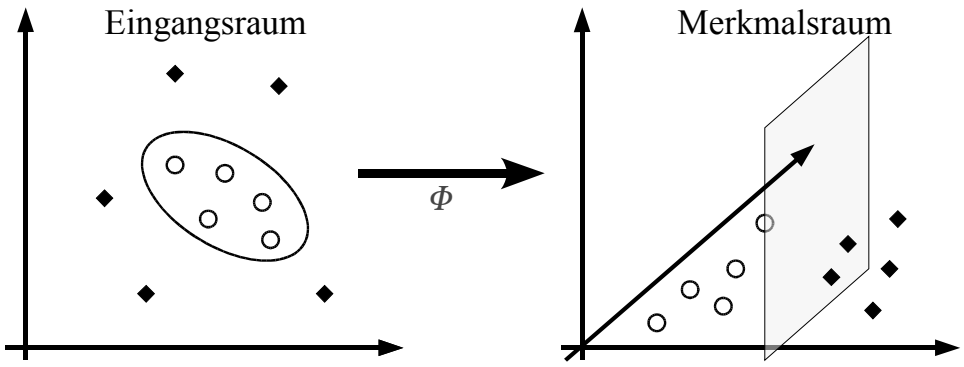


Abbildung 2.14: Transformation einer in $\mathbb{R}^2$ nicht linear separierbaren Menge (links) in einen dreidimensionalen Raum mit linearer Trennbarkeit (rechts)

Für die Klassifikation gilt analog $d_{\tilde{\mathbf{w}},\tilde{b}}(\mathbf{x}) = \Phi\mathbf{x}^T\tilde{\mathbf{w}} + \tilde{b}$, wobei die Berechnung des Skalarprodukts je nach gewählter Transferfunktion und Dimension recht kostspielig sein kann. Durch die Einführung nicht linearer Kernfunktionen[18] $k(\mathbf{x}, \mathbf{w})$ kann der Rechenaufwand bezüglich des Skalarprodukts auf die Dimension des Eingaberaumes beschränkt werden.

$$k(\mathbf{x}, \mathbf{w}) = \big(\Phi(\mathbf{x})^T \cdot \Phi(\mathbf{w})\big) \tag{2.25}$$

Zur Einhaltung obiger Forderung muss die Kernelfunktion neben der Eigenschaft der Symmetrie und Semi-Definitivität die Cauchy-Schwarzsche Ungleichung erfüllen. In der Praxis werden häufig die drei folgenden Kernfunktionen eingesetzt, wobei die günstigste Wahl in der Regel nicht vorhersagbar ist, aber durch empirische Tests ermittelt werden kann.

1. Polynomialkernel (Grad d): $k(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^T \cdot \mathbf{w})^d$
2. Gauß-Kernel[19] (Varianz σ): $k(\mathbf{x}, \mathbf{w}) = \exp(-||\mathbf{x} - \mathbf{w}||^d/2\sigma^2)$
3. Sigmoid-Kernel (mit Verstärkung K und Offset Θ): $k(\mathbf{x}, \mathbf{w}) = \tanh(K(\mathbf{x}^T \cdot \mathbf{w})) + \Theta)$

Da SVM soweit nur für die Lösung von Zweiklassenproblemen eingeführt wurden, sollen im Folgenden drei Möglichkeiten zur Lösung von Mehrklassenproblemen aufgezeigt werden [All01, Nie03]:

[18] Engl.: Kernel Function
[19] Oftmals auch Radiale Basis Funktion (RBF) genannt

1. Rückführung auf k binäre Einzelklassifikatoren, die jeweils zwischen einer aktiven Klasse und allen $k-1$ anderen Klassen unterscheiden: „Eine gegen alle Anderen." Nachdem alle k Klassifikationsergebnisse vorliegen, kann über den maximalen Abstand zur Hyperebene eine finale Entscheidung vorgenommen werden.

2. Rückführung auf $k(k-1)/2$ binäre Einzelklassifikatoren, wobei jede Klasse gegen jede andere diskriminiert wird :„Eine gegen eine." Hier kann die endgültige Wahl der erkannten Klasse durch das Zählen der positiv bewerteten Einzelentscheidungen für eine bestimmte Klasse gefunden werden. In der Praxis hat sich diese Variante durch bessere Erkennungsraten gegenüber der ersten als überlegen erwiesen. Jedoch ist sie dabei mit erhöhtem Berechnungsaufwand verbunden [Has98, Nie03].

3. Verwendung von Entscheidungsbäumen durch geeignete Zerlegung in ein hierarchisches Ensemble mehrerer Binärprobleme[20] [Sch04]. Hierzu kann beispielsweise in jeder Ebene eine Klasse gegen die verbleibenden Klassen diskriminiert werden. Durch die geschachtelten Vergleiche in mehreren Stufen kann die Anzahl der Klassifikatoren gegenüber dem vorigen Ansatz bei ähnlicher Erkennungsleistung stark verringert werden, so dass es zu maximal $k-1$ Klassifikationen kommt.

Im Rahmen dieser Arbeit werden SVM sowohl für die Gesichtsdetektion, als auch für die Erkennung dynamischer Mimiksequenzen eingesetzt und untersucht.

2.7 Klassifikation mit Hidden Markov Modellen

Im Weiteren wird auf die Klassifikation mit Hilfe sogenannter Hidden Markov Modelle[21] (HMM) eingegangen. Der Grund für die große Verbreitung der HMM ist der erfolgreiche Einsatz in einer breiten Palette von Klassifikationsanwendungen, insbesondere im Zusammenhang mit Zeitreihen. So wurden HMM insbesondere zur automatischen Spracherkennung (ASE) [ST95, Jel97], Handschrift- [Rig98] sowie der Piktogrammerkennung [Mul01b, Mul99a] mit großem Erfolg eingesetzt. Die erste der zwei herausragenden Eigenschaften von HMM ist die qualitative Variabilität, welche durch geeignete Modellierung Streuungen innerhalb ähnlicher Merkmalsvektoren zulässt. Die zweite, noch wesentlichere Eigenschaft ist die quantitative Variabilität, welche den Vergleich verschieden langer Vektorfolgen mit nicht linearer Verzerrung ermöglicht. In Abbildung 2.15 wird dies mit zwei unterschiedlich langen Sequenzen von Fingerstreifenabdrücken verdeutlicht.

Obwohl die Verarbeitung zeitlich verzerrter Muster nach diesem Beispiel auch mit Hilfe von lokalen Distanzen und Algorithmen wie der dynamischen Zeitverzerrung[22] bewerkstelligt werden könnte, haben sich doppelt stochastische Automaten zur Modellierung und Analyse realer Signalquellen als überlegen erwiesen.

Bei den hier vorgestellten HMM werden sequentielle Abfolgen von Beobachtungen durch einen ersten statistischen Prozess in Form sogenannter Markov Quellen realisiert. Die

[20] Verfahren auch als Multi-Layer SVM bekannt

[21] Auch hier hat sich der englische Ausdruck durchgesetzt

[22] Dynamic Time Warping (DTW)

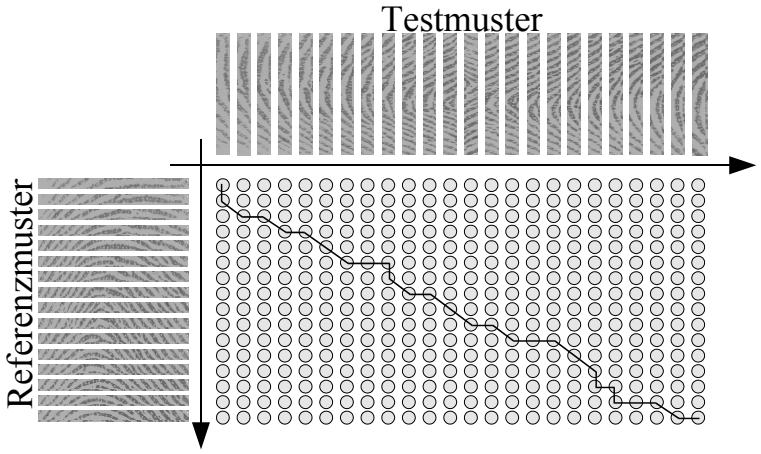


Abbildung 2.15: Nicht lineare Verzerrung zwischen Referenz- und Testmuster [Mor04]

jeweiligen zu einem Modellzustand gehörigen Mustervektoren werden bei kontinuierlichen HMM durch multivariate Normalverteilungen, andernfalls durch diskrete Wahrscheinlichkeiten beschrieben. Die Klassifikation eines unbekannten Musters erfolgt hiernach über die Wahl des Modells mit der größten Produktionswahrscheinlichkeit[23]. Zur Modellierung von Musterfolgen nach dem obigen Beispiel muss für das Signal die Bedingung der stückweisen Quasistationarität gelten.

Im einfachsten Fall wird während der Erkennung nach dem Modell λ_{l^*} aus einem Klasseninventar $\Lambda = \{\lambda_1, \ldots, \lambda_L\}$ vom Umfang L gesucht, welches ein zu klassifizierendes Muster $\mathbf{X} = \mathbf{x}_1, \ldots, \mathbf{x}_T$ mit der höchsten Produktionswahrscheinlichkeit generiert. Es wird also die Klasse gewählt, deren Modell die höchste A-Posteriori-Wahrscheinlichkeit bei vorgegebener Beobachtungssequenz $\mathbf{X}$ aufweist.

$$l^* = \underset{l}{\operatorname{argmax}} P(\lambda_l|\mathbf{X}) \tag{2.26}$$

Dieses Kriterium ist allgemein üblich unter dem Namen Maximum A-Posteriori bekannt. Über den Satz von Bayes kann die oben gesuchte Wahrscheinlichkeit nach Gleichung 2.27 angegeben werden.

$$P(\lambda_l|\mathbf{X}) = \frac{P(\mathbf{X}|\lambda_l)P(\lambda_l)}{P(\mathbf{X})} \tag{2.27}$$

Hierbei kann Kontextwissen über das Auftreten der Modelle über die a priori Wahrscheinlichkeit $P(\lambda_l)$ beschrieben werden. Für den allgemeinen Fall kann die Bestimmung und Schätzung der richtigen Verteilungen beliebig schwierig werden. Im Falle einer Gleichverteilung der vorkommenden Modelle gilt jedoch $P(\lambda_l) = 1/L$, womit für die Maximumsfindung nach Gleichung 2.26 nur noch ein nicht informativer Beitrag vorliegt. Die absolute Auftrittswahrscheinlichkeit der Beobachtung $P(\mathbf{X})$ ist offensichtlich keine Funktion des verwendeten Modells. Sie liefert somit ebenso keinen Beitrag zur Klassenentscheidung und kann ebenfalls unberücksichtigt bleiben. Es bleibt somit $P(\mathbf{X}|\lambda_l)$ statt $P(\lambda_l|\mathbf{X})$ zu bestim-

[23] Maximum Likelihood

men.

$$l^* = \underset{l}{\text{argmax}} P(\mathbf{X}|\lambda_l) \tag{2.28}$$

Eine Übersicht über das Erkennungsproblem mit HMM ist in Abbildung 2.16 gegeben.

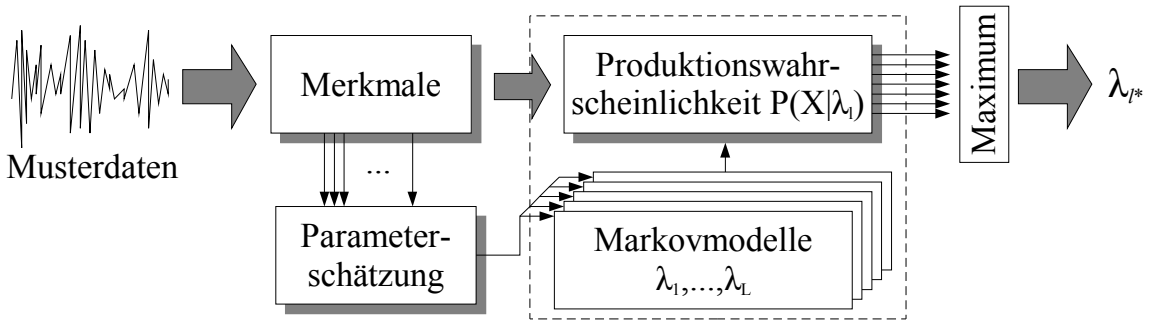


Abbildung 2.16: Typische Mustererkennungsaufgabe mit HMM

Da analog zu den vorgestellten NN nicht die Referenzmuster selbst zu einem Vergleich herangezogen werden, sollen die Parameter zur Repräsentation der unterliegenden Verteilung im nächsten Unterkapitel zunächst formell für eindimensionale Erkennungsaufgaben definiert werden. Hiernach wird auf die besonderen Eigenschaften zur Modellierung und Erkennung von zwei- und dreidimensionalen Mustern eingegangen.

2.7.1 Definition von Modellparametern

Zunächst wird zur Generierung eines eindimensionalen stochastischen Prozesses eine Markov Quelle mit einer endlichen Menge N innerer Zustände angenommen. Die Menge der Zustände eines Automaten ist gegeben durch $\mathcal{Q} = \{s_1, \ldots, s_N\}$. Weiterhin sei eine Folge $\mathbf{q}$ innerer Zustände q_t gegeben, die zu den diskreten Zeitpunkten $t = \{1, 2, \ldots, T\}$ einen Zustand aus $\mathcal{Q}$ annehmen.

$$\mathbf{q} = q_1, \ldots, q_T \ , \ \ q_t \in \ \mathcal{Q} \tag{2.29}$$

Der zu modellierende Prozess soll als kausal gelten. Ein Zustand hängt also nur von den vorausgehenden Zuständen ab. Darüber hinaus soll die Eigenschaft der Einfachheit[24] gelten, wonach die Wahl des aktuellen Zustandes nur vom unmittelbar vorherigen abhängig ist. Ferner soll das zu modellierende Signal stationär sein, also unabhängig von der absoluten Zeit t. Für einen einfachen, kausalen und stationären Prozess können die folgenden Übergangswahrscheinlichkeiten zwischen zwei Zuständen als $P(q_t|q_1, \ldots, q_{t-1}) = P(q_t|q_{t-1})$ festgehalten werden. Diese Sprungwahrscheinlichkeiten können in Form einer Matrix $\mathbf{A}$ der Dimension $N \times N$ geschrieben werden, deren Elemente a_{ij} angeben, wie wahrscheinlich ein Wechsel von Zustand q_i nach Zustand q_j ist. Dabei müssen die einzelnen Elemente den Stochastizitätsbedingungen $\sum_j a_{ij} = 1$ und $a_{ij} \geq 0$ genügen.

$$\mathbf{A} = [a_{ij}]_{N \times N} \ \text{ mit } \ a_{ij} = P(q_t = s_j | q_{t-1} = s_i) \tag{2.30}$$

[24] Markov-Quelle 1. Ordnung

Eine bisher beschriebene Markov Quelle kann nach Abbildung 2.17 in Form eines gerichteten Graphen veranschaulicht werden. Prinzipiell sind beliebige andere Übergänge zwischen den Zuständen möglich. Im Weiteren soll aber ohne Beschränkung der Allgemeinheit von den abgebildeten linearen Modellen ausgegangen werden, in denen nur Übergänge zum nachfolgenden Zustand oder Selbstübergänge, nicht jedoch rückwärtsgerichtete sowie zyklische Transitionen zugelassen sind. Durch weiße Kästchen in Matrix $\mathbf{A}$ nach Abbildung 2.17 werden Übergänge mit der Wahrscheinlichkeit Null symbolisiert. Graue Kästchen repräsentieren von Null verschiedene Werte.

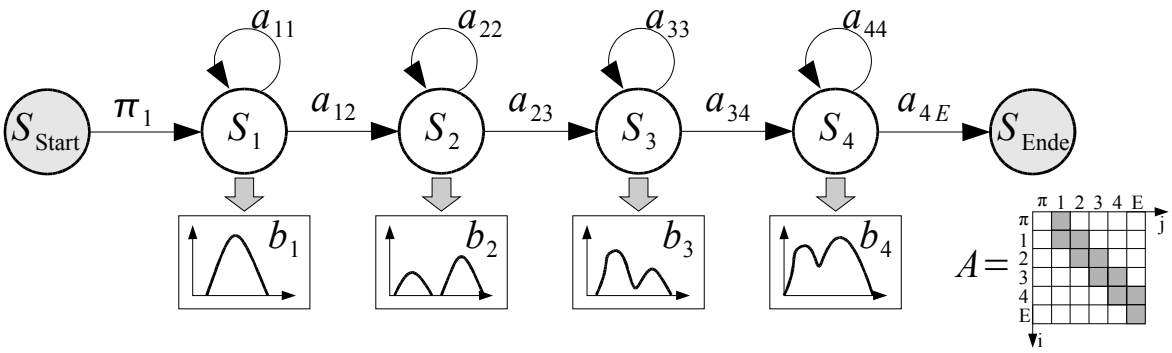


Abbildung 2.17: Lineares Modell mit 4 kontinuierlichen Zuständen und nicht emittierendem Start- S_{Start} und End-Zustand S_{Ende}

Bei Verwendung der betrachteten Markov Quelle würde nun in jedem Zustand ein festes Zeichen emittiert. Bei den HMM soll nun jedoch neben diesem ersten Prozess von einem zweiten Zufallsprozess ausgegangen werden, welcher zu jedem diskreten Zeitpunkt bzw. in jedem Zustand des ersten Prozesses ein diskretes Symbol oder eine kontinuierliche Ausgabe emittiert bzw. absorbiert und deren Produktionswahrscheinlichkeit bestimmt.

Einem Beobachter wird bei diesem zweifach probabilistischen Vorgang von außen jedoch lediglich eine zeitdiskrete Folge von Observationen $\mathbf{X} = \mathbf{x}_1, \ldots, \mathbf{x}_T$ sichtbar, während die Folge $\mathbf{q}$ der tatsächlich eingenommenen Zustände unbekannt bleibt. Diese wesentliche Eigenschaft bestimmt den allgemein gebräuchlichen Namen *Hidden* Markov Modell, da die Zustandsabfolge des Hintergrundprozesses nach Gleichung 2.29 versteckt also *Hidden* bleibt. Randbedingung ist, dass die Produktion eines Zeichens nur vom aktuell eingenommenen Zustand abhängig ist: $P(\mathbf{x}_t|\mathbf{x}_1, \ldots, \mathbf{x}_{t-1}, q_1, \ldots, q_t) = P(\mathbf{x}_t|q_t)$. Als Konsequenz bedeutet dies, dass die Beobachtungen untereinander unkorreliert sein müssen. Die einer Beobachtung $\mathbf{x}_t$ zugrunde liegende stochastische Verteilung in einem Zustand q_j wird durch die Produktionswahrscheinlichkeit $b_j(\mathbf{x}) = P(\mathbf{x}_t|q_t = s_j)$, mit $\mathbf{b} = [b_j]_{N\times 1}$ beschrieben.

Durch die Angabe des Parametersatzes $\lambda = (\mathbf{A}, \mathbf{b})$ sind alle Parameter eines HMM λ eindeutig spezifiziert. Im Folgenden werden die zwei typischsten Ausgabeverteilungen vorgestellt: kontinuierlich und diskret. Die in der Literatur genannten Schritte zum Training und zur Dekodierung sind im Anhang in Kapitel B.4.1 zusammengefasst.

2.7.2 Kontinuierliche Produktionswahrscheinlichkeiten

Bei Verwendung von HMM mit kontinuierlichen Wahrscheinlichkeitsdichtefunktion (WDF) wird davon ausgegangen, dass der zu modellierende Prozess in jedem emittierenden Zustand durch eine Mischverteilungsdichte aus multivariaten Normalverteilungen $\mathcal{N}$ approximiert werden kann. Zwar existieren prinzipiell auch andere Ansätze, in der Literatur dominieren jedoch die Gaußschen Mischverteilungsmodelle[25] (GMM). Durch eine hinreichend große Anzahl an gewichteten Überlagerungen ist prinzipiell die Annäherung an eine beliebige Dichtefunktionen möglich.

$$b_j(\mathbf{x}) = \sum_{k=1}^{K} w_{jk}\mathcal{N}(\mathbf{x}|\mu_{jk}, \mathbf{\Sigma}_{jk}) \text{ und } \mathcal{N} = \frac{1}{\sqrt{(2\pi)^K|\mathbf{\Sigma}_{jk}|}} \cdot e^{-\frac{1}{2}(\mathbf{x}-\vec{\mu}_{jk})^T\Sigma_{jk}^{-1}(\mathbf{x}-\vec{\mu}_{jk})} \quad (2.31)$$

Die zu schätzenden Parameter der kontinuierlichen Ausgabeverteilungen bestehen aus den Gewichtsfaktoren w_{jk} der K Normalverteilungen, den Mittelwertvektoren $\mu_{\mathbf{jk}}$ und der Kovarianzmatrix $\mathbf{\Sigma_{jk}}$. Mit einer solchen Realisierung lassen sich Signale mit abschnittsweise stationären Bereichen modellieren, so dass Folgen mit ähnlichen statistischen Eigenschaften durch denselben Zustand innerhalb des HMM ausgedrückt werden können, wie in Beispiel 2.17 gezeigt.

Durch die Überlagerung ergibt sich automatisch eine gewisse Robustheit gegenüber statistisch abweichenden Merkmalen und Rauschen. Nachteilig kann sich die erhöhte Anzahl zu schätzender Parameter auswirken. Zur Reduktion des Rechenaufwandes wäre es an dieser Stelle dienlich, die Elemente der Beobachtungsvektoren $\mathbf{x}_t$ untereinander statistisch unabhängig zu halten, wodurch die Kovarianzmatrix nur noch Elemente auf der Hauptdiagonalen hat. Insgesamt resultiert[26] eine reduzierte Anzahl von $K\cdot(2N+1)$ gegenüber $K\cdot(N^2+N+1)$ Parametern pro HMM-Zustand. In der Praxis hat es sich bei geringeren Mengen an Trainingsdaten als vorteilhaft erwiesen, die Varianzen der einzelnen Verteilungen nicht beliebig klein werden zu lassen, sondern so zu begrenzen, dass sie nicht unter eine untere globale Schranke c_{var}, den sogenannten *Variance Floor* fallen.

Für eine erfolgreiche Modellierung ist die Einhaltung der gemachten Forderungen unabdingbar. Außerdem hat sich in vielen praktischen Anwendungen gezeigt, dass für die robuste Schätzung aller Modellparameter eine hinreichend große Anzahl an Beispielen zur Verfügung stehen muss.

2.7.3 Diskrete Produktionswahrscheinlichkeiten

Insbesonders bei stark begrenzten Resourcen, wie Speicher- und Rechenkapazität, sowie bei geringer Anzahl an Trainingsbeispielen hat sich die Verwendung von diskreten Produktionswahrscheinlichkeiten als nützlich erwiesen [Neu99]. Vor der Schätzung der Auftrittswahrscheinlichkeiten der diskreten Label m_k muss zunächst ein Ausgabealphabet $\mathcal{M}$ erzeugt werden, so dass die kontinuierlichen Vektorfolgen $\mathbf{X}$ auf diskrete Reihen abgebildet werden können: $\mathcal{M} = \{m_1, \ldots, m_K\}$.

[25] Engl.: Gaussian Mixture Model

[26] je K Parameter in Σ_{jk} und μ_{jk} und einer für w_{jk}

Bei diskreten HMM wird der kontinuierliche Merkmalraum in M disjunkte Unterräume $\mathbf{M}_k$ aufgeteilt. Hiernach werden die vormals kontinuierlichen Merkmale $\mathbf{x}_t$ durch einen Vektorquantisierer (VQ) über diskrete Label in gewichtete Wahrscheinlichkeiten überführt, wie in Abbildung 2.18 skizziert. Ein Vektor $\mathbf{x}_t$ wird dann auf den Codebuch-Eintrag m_k mit Wahrscheinlichkeit w_k abgebildet, wenn er Element des Unterraums $\mathbf{M}_k$ ist.

$$b_j(\mathbf{x}) = \begin{cases} w_k & \text{wenn } \mathbf{x}_t \in \mathbf{M} \\ 0 & \text{sonst} \end{cases} \tag{2.32}$$

Für einen HMM-Zustand s_j existieren demnach M verschiedene Parameter, welche analog zu den oben genannten GMM die unterliegende Verteilung modellieren sollen. Die Bestimmung der lokalen Wahrscheinlichkeit kann während der Dekodierung über das Ablesen eines Codebucheintrages effizient gelöst werden. Mit dem Ziel einer weniger fehleranfälligen Modellierung während des Trainings wird in der Praxis auch hier eine untere Schranke c_{min} angegeben, unter welchen die diskrete Wahrscheinlichkeit nicht fallen darf. Das Code-

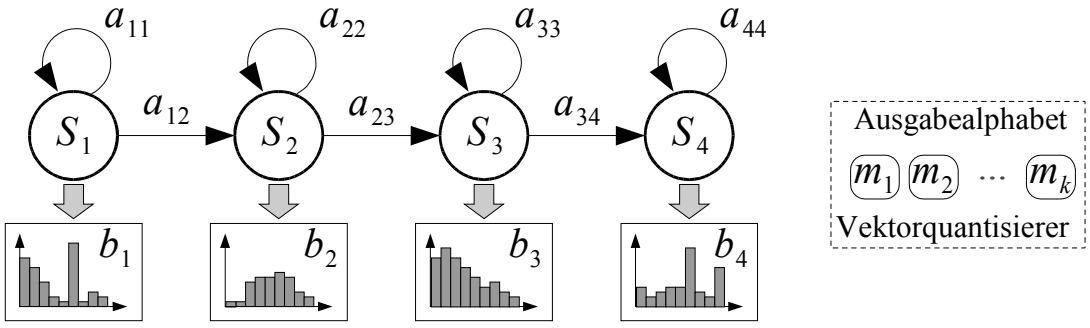


Abbildung 2.18: Lineares Modell mit 4 diskreten Zuständen

buch kann beispielsweise über einen K-Means-Cluster-Algorithmus konstruiert werden, wobei alternative Verfahren aufgrund ihrer gesteigerten Performanz oftmals vorzuziehen sind [Neu99]. Die Abbildung der Musterdaten zu den jeweiligen Vektoren kann nach dem Prinzip des nächsten Nachbarn erfolgen. Mit zunehmender Größe des Codebuchs wird prinzipiell eine feinere Partitionierung des Vektorraums möglich, was einem entstehenden Quantisierungsfehler entgegenwirkt.

2.7.4 Bildmodellierung mit pseudo-zweidimensionalen Hidden Markov Modellen

Die wesentliche Eigenschaft der Klassifikation mit HMM ist die Fähigkeit, dynamisch verzerrte Muster bearbeiten zu können. Es wird angestrebt, diese Eigenschaft auch bei Bilddaten beizubehalten, was zu dem erweiterten Begriff *Dynamic Plane Warping* (DPW) führt. Eine theoretisch ideale Möglichkeit zur Modellierung zweidimensionaler Bilddaten besteht in einer Verallgemeinerung von HMM, den Markov-Random-Fields. Aufgrund fehlender Algorithmen finden diese jedoch in der Praxis keine Anwendung [Mul02a].

Obwohl reale Bilddaten, in Abschnitt 2.2 über Matrizen eingeführt, keine kausalen Prozesse repräsentieren und somit prinzipiell eine Forderung der Anwendbarkeit von HMM verletzen, konnten sie in der Vergangenheit erfolgreich eingesetzt werden, wie auch im Zusammenhang der Gesichtserkennung später noch gezeigt wird.

Eine formale Restriktion auf eindimensionale Muster ergibt sich jedoch zunächst über die oben eingeführte Modellstruktur. Eine Möglichkeit zum Erhalt der zweidimensionalen Natur der Muster besteht im Aufstellen einer ergodischen, echt zweidimensionalen Übergangsmatrix. Dies führt aber schon bei kleinen Modellen und wenigen Zuständen zu einem immensen Rechenaufwand. Begründet liegt dies in der explosionsartig gestiegenen Anzahl an möglichen Wegen durch das Trellisdiagramm. Die Anwendung planarer bzw. echt zweidimensionaler Hidden Markov Modelle (2DHMM) scheidet somit für praktische Anwendungen ebenso aus.

Im Bereich der bildbasierten Zeichenerkennung[27] konnte von Kuo und Agazzi [Kuo94] eine Näherungslösung zur Modellierung von Bildern mit gedruckter Schrift eingeführt werden. Die Topologie der dabei verwendeten, hierarchischen Modelle wird als pseudozweidimensionales Hidden Markov Modell (P2DHMM) bezeichnet. Mit der Verwendung dieser Modelle geht allerdings der Verzicht auf die dynamische Verzerrung in eine Dimension einher, was durch den Begriff *pseudo* in der Namensgebung betont wird.

Nach Samaria [Sam94] nimmt man bei diesem Vorgehen vorab ein übergeordnetes lineares Modell mit Superzuständen an, in welches statt Emissionswahrscheinlichkeiten untergeordnete Modelle eingebettet werden. In Abbildung 2.19 ist ein solches P2DHMM mit 4 Superzuständen dargestellt.

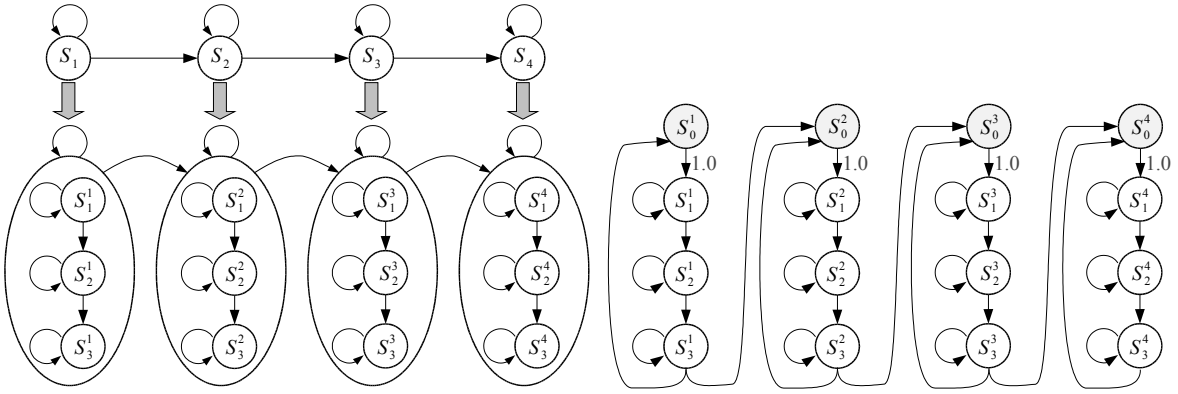


Abbildung 2.19: P2DHMM als lineare Folge von Superzuständen (links) und äquivalentes 1DHMM (rechts)

Die Wahl, ob die horizontale oder vertikale Dimension übergeordnet ist, wird hauptsächlich durch die Art der zu modellierenden Daten bestimmt. Im Folgenden wird von einem übergeordneten horizontalen Prozess ausgegangen. Zur Realisierung der oben eingeführten Problematik kann zum einen ein geschachtelter Viterbi-Algorithmus zum Zuge kommen. Alternativ kann die entstandene Struktur und deren Verarbeitung auf bekannte Verfahren und Algorithmen zurückgeführt werden [Sam94]. Ein äquivalentes eindimensionales HMM

[27] Optical Character Recognition (OCR)

(1DHMM) kann implementiert werden, indem man vor den Beginn der inneren Prozesse zusätzliche Markierungszustände einfügt, welche in den ersten Zustand der Spalte springen, also $a_{ij} = 1.0$. Des Weiteren müssen die jeweils involvierten Zustandsübergänge adäquat angepasst werden, d.h. der letzte Zustand eines Untermodells muss auf den eigenen sowie den folgenden Markierzustand zeigen, wie rechts in Abbildung 2.19 gezeigt. Zur besseren Übersicht soll die zweidimensionale Struktur durch die Zustandsanordnung erhalten bleiben. Die vorgestellte Erweiterung macht eine Modifikation der Merkmalsdaten notwendig, so dass zu Beginn eines Spaltenanfangs eine Markierung in den Merkmalstrom eingebracht werden muss. Durch die Wahl eines Symbols oder Wertes, der außerhalb des regulären Bereichs liegt, kann eine richtige Zuordnung zwischen Merkmalen und Zuständen erzwungen werden. Aus einem P2DHMM mit 4×3 States wird also ein gleichwertiges 1DHMM mit $(4 \times 3) + 4$ Zuständen, welches mit den im Anhang erläuterten Methoden trainiert und zur Klassifikation eingesetzt werden kann.

2.7.5 Modellierung von Bildfolgen mit pseudo-dreidimensionalen Hidden Markov Modellen

Angesichts der Erfolge bei der Verarbeitung zweidimensionaler Daten mit P2DHMM ist der Wunsch erwachsen, Bildfolgen mit HMM modellieren und klassifizieren zu können. Auch hier würden theoretisch die dreidimensionalen Markov-Random-Fields bzw. ergodische dreidimensionale Hidden Markov Modelle (3DHMM) Anwendung finden, was aber ebenfalls aus den oben genannten Gründen, wie fehlenden Algorithmen ausgeschlossen werden muss.

Zur Lösung werden die bewährten P2DHMM zu den neuartigen pseudo-dreidimensionalen Hidden Markov Modellen (P3DHMM) erweitert. Die formale Einführung dieser Erweiterung zeigte erste interessante Ergebnisse im Zusammenhang der Gestenerkennung und soll für die dynamische Mimikerkennung aufgegriffen werden [Mul00, Yal00, Mul02a]. Definiert seien P3DHMM über einen dreifach hierarchischen Prozess. Wie rechts in Darstellung 2.20 gezeigt, impliziert dies die wiederholte Kapselung von P2DHMM in Hyperzuständen eines obersten Prozesses.

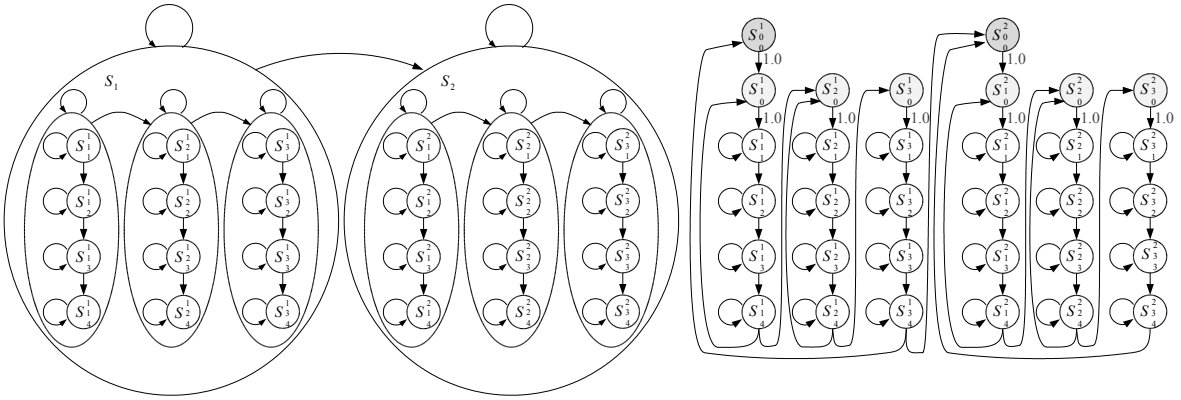


Abbildung 2.20: P3DHMM mit hierarchischer Kapselung (links) und als äquivalentes 1D-Modell (rechts)

Aus Gründen der Implementierbarkeit werden P3DHMM ebenfalls wieder in gleichwertige 1DHMM transformiert. Dies erfolgt durch konsequente Wiederholung aller erforderlichen Schritte der Überführung eines P2DHMM in ein 1DHMM. Im Detail wird dies durch die Platzierung weiterer Markierungen, den Hyperzuständen, erreicht. Diese zusätzlichen Zustände repräsentieren den Bildstart. Zusammen mit den eingekapselten P2DHMM können diese Zustände zu der im rechten Teil der Grafik 2.20 aufgezeigten Zustandsfolge konvertiert werden. Auch hier muss die Zustandübergangswahrscheinlichkeit vom jeweils letzten Zustand auf den Bildanfang sowie zum nächsten Frame angepasst werden. Damit auch bei den P3DHMM bzw. deren äquivalenten 1DHMM die richtige Zuordnung zwischen Bild- und Spaltenanfang gewährleistet werden kann, müssen ebenfalls Hypermerkmale mit einem Wert außerhalb des Wertebereichs in den Merkmalstrom integriert werden.

Aus einem P3DHMM mit 2 Hyperzuständen, je 3 Superzuständen und je 4 inneren Zuständen wird somit bereits ein Modell mit einer beachtlichen Zahl von $((4*3)+3)*2)+2 = 32$ Zuständen. Obwohl in der Literatur eine formelle Einführung des zweifach verschachtelten Viterbi-Algorithmus existiert [Mul02b], wird aus Gründen der Implementierbarkeit ebenfalls wieder auf die bekannten Verfahren zurückgegriffen.

2.8 Hybride Systeme

In diesem, die Grundlagen abschließenden Unterpunkt soll die Kombination von verschiedenen Paradigmen der Mustererkennung vorgestellt werden. In der Literatur werden Verknüpfungen zwischen beliebigen Verfahren prinzipiell als hybride Systeme bezeichnet, im Folgenden soll unter diesem Terminus jedoch speziell die gemeinsame Verwendung von NN und HMM verstanden werden [Rig02].

Ziel der Kopplung ist die simultane Nutzung der positiven Wesensmerkmale der involvierten Techniken. Auf der einen Seite sind dies die Vorteile von HMM zur Bearbeitung zeitlich oder räumlich verzerrter Muster, die diskriminativen Eigenschaften von NN beim Training und der Klassifikation andererseits. Ein Beispiel für den bewährten Einsatz dieser Kombination ist ein verwendeter NN-basierter Vektorquantisierer nach dem MMI[28] Kriterium, welcher kontinuierliche Merkmalsdaten auf diskrete Label abbildet, die anschließend zur Bestimmung von Produktionswahrscheinlichkeiten genutzt werden [Neu01].

Wie im Kontext des Trainings von HMM bereits erwähnt, können die Parameter der Normalverteilungen auch über NN geschätzt werden, was sich im Bereich der ASE als vorteilhaft erwiesen hat [Rot00]. Im Lauf dieser Arbeit wird dementsprechend ein Verfahren zum gemeinsamen Parametertraining im Anwendungsbereich der Gesichtserkennung über Profilbilder detaillierter vorgestellt.

[28] Maximierung der Mutual Information (MMI) bzw. der Transinformation

Kapitel 3

Detektion von Gesichtern

Wie eingangs bereits erläutert, widmet sich der erste der drei Themenkreise mit Relevanz für die visuelle Mensch-Maschine-Kommunikation der Findung von Gesichtsmustern in Bildern mit beliebigem Hintergrund.

Vor einer Personenerkennung über das Muster Gesicht müssen in einem ersten Schritt zunächst die charakteristischen Bereiche über die Gesichtsfindung isoliert werden. Der erste Teil dieses Kapitels behandelt daher die Findung von frontal aufgenommenen Gesichtern in Einzelbildern, wobei verschiedene Ansätze zur Detektion in statischen Bildern untersucht werden. Anschließend wird der leistungsfähigste Ansatz auf die Findung von in die Ebene gedrehten Gesichter erweitert. Im dritten Schritt werden die vorgestellten Einzelkomponenten durch Hinzunahme temporaler Informationen auf die Verfolgung von Gesichtern in Bildsequenzen erweitert.

Prinzipiell unterscheidet sich die Aufgabe ein Gesicht zu finden nicht grundsätzlich von anderen Segmentierungs- und Lokalisierungsaufgaben, wie beispielsweise der Personenfindung [Rig04b]. Durch die Adaption des Trainingsmaterials sind auch verwandte Aufgabenstellungen realisierbar.

Bedingung für eine robuste Detektion von Objekten und somit auch Gesichtsbereichen in Einzelbildern und Bildsequenzen ist eine hohe Invarianz gegenüber den folgenden Einflüssen:

- der Lage bzw. räumlichen Position innerhalb eines Bildes,
- der Auflösung, Größe bzw. Skalierung,
- Drehungen aus der sichtbaren Bildebene, wie in Abbildung 3.1a,
- Drehung bzw. Kopfneigung innerhalb der sichtbaren Bildebene, siehe Abbildung 3.1d,
- den äußeren Belichtungseinflüssen,
- sowie eventuell vorhandenen Verdeckungseffekten.

Die hohe Variabilität innerhalb der dreidimensionalen Klasse Gesicht und deren spezielle Erscheinung haben die Aufgabe der Gesichtsdetektion zu einer eigenen Disziplin im Bereich

der Mustererkennung werden lassen. Die besonderen Ansprüche an Gesichtsdetektoren ergeben sich durch:

- die Personenunabhängigkeit,
- Alter, Hautfarbe und Geschlecht,
- unterschiedliche Emotionen bzw. Gesichtsausdrücke,
- das mögliche Vorhandensein von Verdeckungen durch Brillen und Bärte,
- sowie einer Vielzahl kosmetischer Aspekte wie beispielsweise Frisuren.

In Zusammenstellung 3.1 sind mögliche Erscheinungsformen von Gesichtern sowie deren Variationen und den daraus resultierenden Problemen beispielhaft dargestellt. Wie aus

3.1a 3.1b 3.1c

3.1d 3.1e 3.1f

Abbildung 3.1: Verschiedene mögliche Erscheinungsformen der Klasse Gesicht

der Betrachtung dieser Bilder schnell gefolgert werden kann, erfolgt die Findung der darin enthaltenen Gesichter durch den visuellen Apparat des Menschen weitestgehend schnell und mühelos. Die Nachbildung dieser Leistung mit Hilfe von Digitalrechnern hat sich allerdings als nicht trivial herausgestellt. In den letzten Jahren sind verschiedene Ansätze und Verfahren zur Gesichtsdetektion entwickelt worden. Die wesentlichen Kategorien werden im folgenden Kapitel vorgestellt.

3.1 Überblick über Systeme und Techniken zur Gesichtsfindung

In der Literatur finden sich mehrere Systeme zur Gesichtsfindung in Standbildern und Bildsequenzen vor beliebigen Hintergründen. Hierbei wird üblicherweise zwischen den Termini Detektion und Lokalisierung differenziert. Die Lokalisation setzt dabei implizit das Vorhandensein genau eines Gesichtes voraus, dessen genauer Ort in einem gegebenen Bereich innerhalb eines einzelnen Farb- oder Grauwertbildes zu bestimmen ist. Der umfassendere Begriff Detektion deckt darüber hinaus auch die Fälle das Vorhandensein keines oder mehrerer Gesichter ab.

Die meisten der in der Literatur erwähnten Ansätze zur Findung von Gesichtern in Einzelbildern lassen sich nach Yang in vier Hauptkategorien untergliedern, wobei Überschneidungen nicht ausgeschlossen werden können [Yan02].

1. **Wissensbasierte Methoden** (Knowledge-Based Methods): Verfahren nach dieser Art verwenden menschliches Wissen in Form von Regeln, um die wesentlichen Bestandteile und Merkmale zu finden, die ein Gesicht ausmachen. Hauptsächlich werden sie zur Lokalisierung eingesetzt. Die Schwierigkeit liegt hierbei im Finden geeigneter Regeln. Dies können beispielsweise die Positionen von Augen, Nase und Mund sowie die Abstände untereinander sein.

2. **Methoden mit Hilfe von unveränderlichen Merkmalen** (Feature Invariant Approaches): Die hier verwendeten Techniken basieren auf strukturellen Eigenschaften und Merkmalen von Gesichtern, die unabhängig vom Blickwinkel, den Lichtverhältnissen sowie verschiedenen Gesichtsausdrücken vorhanden sind. Auch diese Verfahren sind hauptsächlich für die Lokalisation zu verwenden. Beispiele hierfür sind Ansätze unter Verwendung von Hautfarbe oder Texturen.

3. **Schablonen-basierte Methoden** (Template Matching Methods): Bei dieser Technik werden zunächst Schablonen von ganzen Gesichtern oder einzelnen Partien bestimmt. In einem zweiten Schritt können Gesichter mit Hilfe der Korrelation zwischen Schablone bzw. Vorlage und zu untersuchendem Bildbereich sowohl detektiert als auch lokalisiert werden.

4. **Suche nach Erscheinungsmustern** (Appearance-Based Methods): Im Gegensatz zu den Schablonen orientierten Ansätzen werden hier heuristische Modelle aus gegebenen Beispielen abgeleitet, die eine möglichst hohe Repräsentationsbandbreite aufweisen sollen. Verfahren dieser Art können für die Aufgabe der Detektion herangezogen werden. Typische Vertreter dieser Gattung verwenden Technologien wie neuronale Netze, Eigenfaces, Point Distribution Modelle, Support Vektor Maschinen sowie Hidden Markov Modelle.

Eine weitere Möglichkeit zur Einteilung von Findungssystemen auf Basis verwendeter Technologien aus dem Bereich der Mustererkennung ist im Diagramm 3.2 nach Hjelmaas dargestellt [Hje01]. Die in der Einteilung dunkel markierten Techniken werden im Folgenden

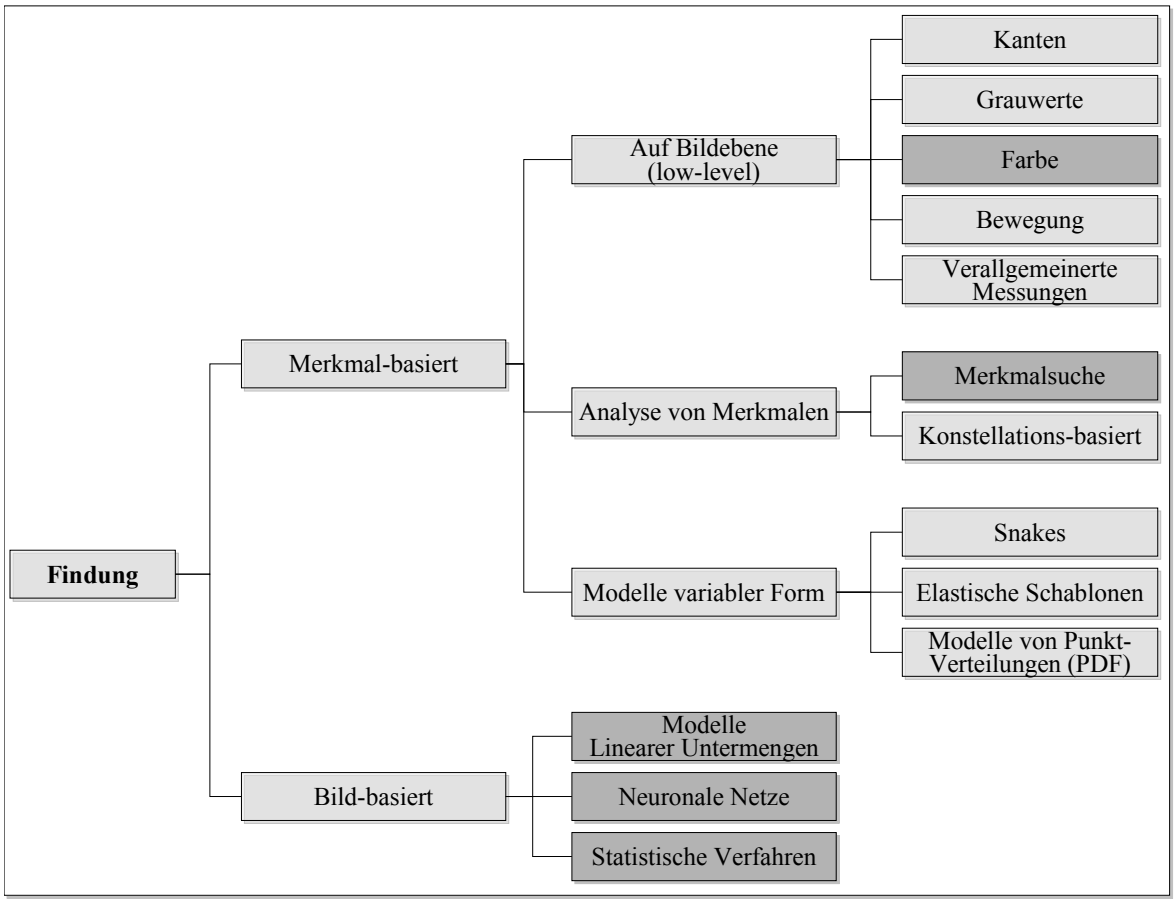


Abbildung 3.2: Einteilung verschiedener Gesichtsfindungsverfahren nach Techniken der Mustererkennung [Hje01]

zur Realisierung von Gesichtsfindungssystemen detaillierter vorgestellt und im Rahmen der Arbeit verglichen. Mit dem Ziel einer robusten Detektion werden zunächst zwei hautfarbenbasierte Ansätze erläutert, welche zur Einschränkung des gesamten Hypothesenraumes innerhalb eines Bildes verwendet werden können. Die Verwendung zusätzlicher Farbinformationen wirkt sich dabei nicht nur positiv auf die Steigerung der Zuverlässigkeit aus, sondern trägt auch zur Reduktion des Berechnungsaufwandes bei. Anschließend werden vier Verfahren nach Erscheinungsmustern zur frontalen Gesichtsdetektion vorgestellt und experimentell verglichen. Auf den Ergebnissen basierend wird im Anschluss daran ein neuartiges NN-System zur richtungsunabhängigen Detektion vorgestellt.

3.2 Segmentierung nach Hautfarben

Obwohl die Nutzung von Farbinformationen zur Gesichtsfindung über Hautfarben nicht immer ausreichend ist, wird sie aufgrund ihres geringen Rechenaufwandes oftmals für eine erste Segmentierung eingesetzt. Eine weitere positive Eigenschaft ist die Unempfindlichkeit bezüglich der Ausrichtung der zu findenden Gesichter.

Zur Modellierung von Hautfarbe existiert in der Literatur eine Vielzahl möglicher Farbräume [Ska93]. Gemeinsames Ziel aller Ansätze ist eine möglichst globale mathematische Beschreibung von Bereichen mit hautfarbenen Pixeln, unabhängig von äußeren Belichtungssituationen und Farbspektren. Die Verwendung des sogenannten intensitätsnormierten Farbraumes[1] hat sich hierbei oftmals als überlegen hervorgetan [Sto04]. Durch die folgende Transformation werden die Farbintensitäten R, G und B in die *RG*-Chrominanz-Ebene überführt:

$$r = \frac{R}{R+G+B} \qquad g = \frac{G}{R+G+B} \qquad b = \frac{B}{R+G+B} \tag{3.1}$$

Durch diese Transformation beinhalten die neuen Variablen nur noch die reine Farbinformation, jedoch keine Angabe über Intensitäten mehr. Da für alle Farben die Beziehung $r + g + b = 1$ gilt, werden üblicherweise nur die zwei Werte r und g berücksichtigt.

Nach der Transformation folgt nun die Aufgabe, die sich darin befindlichen hautfarbenen Bereiche zu beschreiben oder zu modellieren. Ein grundsätzliches Problem mit dem jedoch alle farbbasierten Systeme konfrontiert sind, ist die fehlende Farbkonstantheit, wonach der gleiche Hautbereich unter Beleuchtung mit zwei Lichtquellen verschiedener Farbspektren jeweils anders in Erscheinung tritt. Um ein möglichst breites Szenario abdecken zu können, werden im Rahmen einer schnellen und robusten Vorsegmentierung zwei verschiedene Verfahren auf ihre Einsetzbarkeit untersucht und implementiert:

1. **Modellierung mit Gaußschen Mixture Modellen.**

   Vorteil dieses ersten Ansatzes ist die gute Adaptierbarkeit an die verwendete Kamera und die gegebenen äußeren Belichtungseinflüsse. Durch Parameteränderung kann das Hautfarbenmodell an die Farbtemperatur der verwendeten Beleuchtung angepasst werden. Nachteil ist, dass ein solches Modell vor der Anwendung trainiert werden muss.

2. **Modellierung nach physikalischen Eigenschaften der Haut.**

   Hier begründet sich der Vorteil durch ein physikalisch motiviertes Modell der menschlichen Haut. Eine Adaptierung an die speziellen Gegebenheiten der Aufnahmen kann somit in erster Näherung entfallen und es bedarf keines Trainings mehr. Ein Nachteil liegt in der Notwendigkeit eines kalibrierten Kameramoduls.

3.2.1 Hautfarbenmodell mit Gaußschen Modellen

Zur Findung von hautfarbenen Pixeln wird ein GMM κ mit Hilfe des EM-Algorithmus und zuvor manuell selektierter Bereiche bzw. einzelner Bildpunkte trainiert [JM99, Yan99]. Hieraus resultiert die Bedingung, dass die Verhältnisse des Testszenarios gleiche farbliche Eigenschaften aufweisen müssen bzw. dass das verwendete Modell ansonsten adaptiert werden muss [Raj99]. In der Erkennungsphase kann für jeden Bildpunkt $I(x, y)$, ursprünglich bestehend aus einem Farbtripel R, G und B, die Hautfarbenwahrscheinlichkeit $P(I(x, y)|\Omega_{Haut})$

[1] Auch: Normalized Color Coordinates (NCC) Space

bestimmt werden, wobei $c = [r(x,y), g(x,y)]^T$.

$$S(x,y) = \begin{cases} 1 & T_{Haut} < P(I(x,y)|\Omega_{Haut}) = \frac{1}{2\pi\sqrt{||\Sigma_\kappa||}} e^{-\frac{1}{2}(c-\mu_\kappa)^T \Sigma_\kappa^{-1}(c-\mu_\kappa)} \\ 0 & sonst \end{cases} \tag{3.2}$$

Durch Schwellwertbildung über T_{Haut} kann eine korrespondierende binäre Hautfarbenmatrix S gebildet werden. Diese besagt an welchen Stellen im Originalbild Hautfarbe vorhanden ist und wo nicht. Im konkreten Anwendungsfall hat sich gezeigt, dass mit einem aus einer einzigen Normalverteilung bestehenden GMM schon hervorragende Ergebnisse erzielt werden können, was zudem zu einer beschleunigten Berechnung beiträgt. Für diesen Spezialfall degradiert die Parameterschätzung zu einer trivialen Berechnung der Mittelwerte und der Kovarianzmatrix. In Gleichung 3.3 ist ein experimentell bestimmter Parametersatz gegeben, welcher auf Basis von Fernsehbildern in Studioqualität bestimmt wurde.

$$\mu_\kappa = \begin{pmatrix} 0,44548 \\ 0,28935 \end{pmatrix} \quad , \quad \Sigma_\kappa = 10^{-3} \cdot \begin{bmatrix} 4,0916 & -0,3925 \\ -0,3925 & 1,53269 \end{bmatrix} \tag{3.3}$$

Im RG-Chrominanzraum belegt obiges GMM bei einer Schwelle von $T_{Haut} = 0.5$ die in Grafik 3.3 abgedruckte elliptische Fläche. Pixel in diesem Farbbereich werden als hautfarben eingestuft. Wie bereits betont, hängt die Leistungsfähigkeit des Ansatzes stark davon ab,

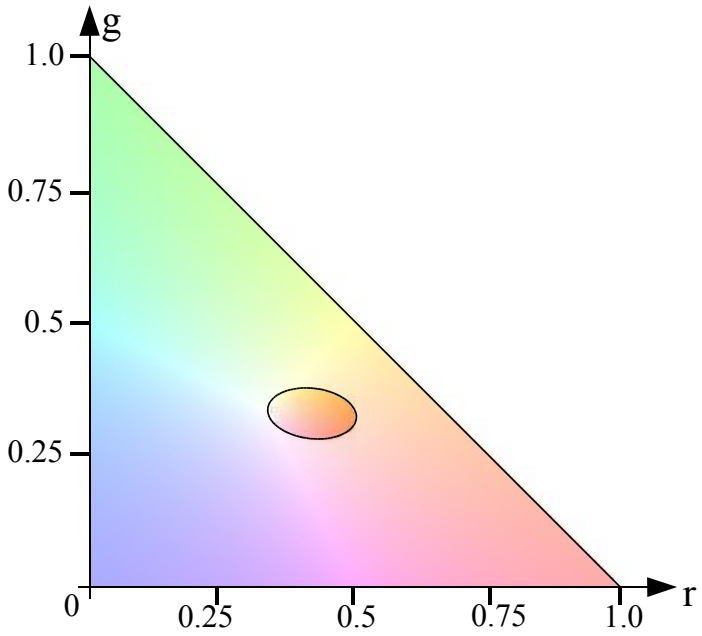


Abbildung 3.3: Hautfarben im RG-Chrominanzraum nach dem GMM-Ansatz

wie nah die farblichen Konditionen der Testaufnahme an denen der Trainingsdaten liegen. Eine gelungene Farbsegmentierung ist in Abbildung 3.4 demonstriert, in welcher die Gesichter der recht dunkel belichteten Zuschauer farblich ausnahmslos richtig segmentiert werden. Als problematisch zeigt sich die hautfarbenähnliche Oberbekleidung des zweiten Zuschauers von links in der obersten Reihe, welche offenbar ebenfalls im Hautfarbenraum liegt. Auf

eine mögliche Weiterverarbeitung, wie beispielsweise die Zusammenfassung benachbarter Punkte zu Flächen, den sogenannten *Blobs* sowie weiteren Heuristiken wird verzichtet, da im Rahmen der Arbeit nur die Auftrittshäufigkeit innerhalb eines zu testenden Bereichs von Relevanz sein wird.

Abbildung 3.4: Originales Studiobild (links) zusammen mit seiner Skinmask (rechts)

3.2.2 Physikalisch motiviertes Hautfarbenmodell

Die zweite betrachtete Möglichkeit, Bereiche mit hautfarbenen Pixeln zu definieren beruht auf einem physikalischen Modell der zweilagigen menschlichen Haut, repräsentiert durch die äußere Epidermis und die darunter liegende Dermis [Sto99]. Dazu wird zunächst der reine Bildentstehungsprozess betrachtet, nach welchem ein geringer Bruchteil des auf die Haut einfallenden Lichts mit der gleichen Farbverteilung der Quelle direkt an der Oberfläche reflektiert wird. Die Absorption außer Acht lassend, dringt der Hauptanteil in die Epidermis ein, wird materialcharakteristisch gefiltert und beim Auftreffen auf die Dermis wieder nach außen transmittiert. Die farblichen Filtereigenschaften der Epidermis hängen somit hauptsächlich von den spektralen Transmissionseigenschaften, bestimmt durch den sogenannten Dopa-Melanin Anteil ab, welcher je nach Hauttyp schwanken kann. Durch die Eigenschaft der intensitätsnormierenden Transformation nehmen sowohl braune, gelbe, dunkle als auch blasse Haut sehr nah beieinander liegende Positionen im RG-Chrominanzraum ein. Wenn die Farbtemperatur der Quelle bei konstanten Kameraparametern über das sichtbare Spektrum variiert wird, bildet sich für verschiedene Hauttypen letztendlich ein zusammenhängender Bereich innerhalb des RG-Chrominanzraumes, welcher in der Literatur als Skin Locus[2] bezeichnet wird [Mar02b, Sto04].

Bei Verwendung einer kalibrierten und weißabgeglichenen Kamera hat der Hautfarbenbereich in der RG-Chroma-Ebene die Form eines nach unten geöffneten Halbmondes, welcher durch zwei quadratische Funktionen g_{up} und g_{down} beschrieben werden kann. Zur Rückweisung weißer und grauer Punkte, die zunächst im eingeschlossenen Bereich liegen, wird noch ein dritter Kreis definiert, welcher seinen Mittelpunkt im Zentrum der Farbe Weiß hat, also

[2] Im Sinne von Hautfarben Ortskurve

um $r = 0.33$ und $g = 0.33$. In der Praxis haben sich hierzu bei kalibrierter Kamera Kreise mit den Parametrisierungen nach Gleichung 3.4–3.6 als hervorragend geeignet herausgestellt [Sor00].

$$g_{up} = -1,8423 * r^2 + 1,5294 * r + 0,0422 \tag{3.4}$$

$$g_{down} = -0,7279 * r^2 + 0,6066 * r + 0,1766 \tag{3.5}$$

$$W_r = (r - 0,33)^2 + (g - 0,33)^2 < 0,02^2 \tag{3.6}$$

Schließlich wird ein Pixel als hautfarben eingestuft, wenn es oberhalb der unteren und unterhalb der oberen Funktion sowie außerhalb des Weißkreises W_r liegt.

$$S(x,y) = \begin{cases} 1 & (g < g_{up}) \wedge (g > g_{down}) \wedge (W_r > 0.004) \\ 0 & sonst \end{cases} \tag{3.7}$$

Der so entstandene eingeschlossene Bereich ist in Abbildung 3.5 dargestellt. Ein Vergleich mit dem oben vorgestellten GMM zeigt, dass der Hautfarbenbereich im Farbraum eine größere Fläche einnimmt. Durch diese globale Modellierung kann bei kalibriertem Bildsensor

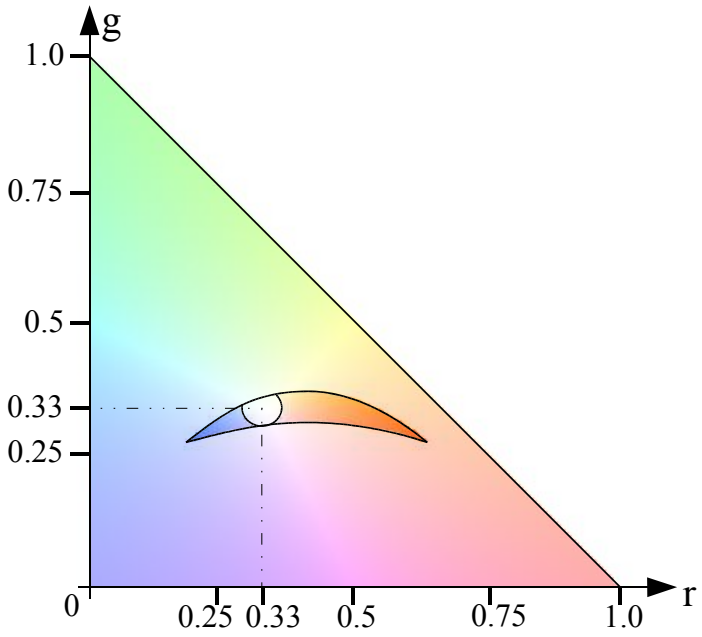


Abbildung 3.5: Skin Locus im RG-Chrominanzraum

zwar ein breiteres Spektrum an möglichen Szenarien abgedeckt werden, damit ist aber auch eine geringere Trenneigenschaft bei hautähnlichen Farben verbunden. Durch die beiden Bildmasken in 3.6 wird diese Problematik veranschaulicht. Die Hautsegmentierung über den Skin Locus ist zunächst prinzipiell ebenfalls akzeptabel, weist in Details jedoch Abweichungen zum GMM auf, welches auf dieses spezielle Beleuchtungsszenario optimiert wurde. Im Speziellen seien hier die nicht mehr so ausgeprägten Konturen innerhalb der Gesichter an Augen, Mündern und den Schatten der Nasen genannt.

Abbildung 3.6: Vergleich der Hautfarbenmasken nach dem Skin Locus Modell (links) und nach dem GMM (rechts)

Prinzipiell wäre die Lage des Skin Locus über Parametervariationen ebenfalls adaptierbar [Mar02b], würde aber wieder ein zusätzliches Training erfordern. Auf der anderen Seite könnte ein GMM mit einer genügend großen Anzahl von Trainingsdaten und überlagerten Normalverteilungen an die Form des Skin Lokus heranreichen.

Zusammenfassend sind GMM bei bekannten und bereits modellierten Farbverteilungen vorzuziehen, ohne Wissen über die Farbverteilung und bei Kameras mit automatischem Weißabgleich ist im allgemeinen Fall die Modellierung über den Skin Locus vorzuziehen.

3.3 Blockbasierte Abtastverfahren

Wie in obigem Beispiel gezeigt, birgt die Verwendung von Hautfarben zur Detektion von Gesichtern unabhängig von der speziell gewählten Methode selbst bei optimaler Modellierung grundsätzlich die Gefahr, dass sich Gesichter von hautfarbenen Bereichen nicht separieren lassen. In der zweiten verwendeten Gruppe zur Gesichtsdetektion, den blockbasierten Abtastverfahren, wird zudem die wichtige Eigenschaft der Unabhängigkeit gegenüber dem verwendeten Hintergrund angestrebt.

Die am häufigsten verwendete Lösung basiert auf einer sequentiellen Untersuchung des gesamten Bildes mit sich überlappenden Abtastfenstern und verschiedenen Skalen, wodurch dafür Sorge getragen wird, dass Gesichter an allen Bildpositionen mit beliebigen Skalierungen vorhanden sein können. Bei dieser sogenannten pyramidalen Abtastung ergibt sich aus der Art des Ansatzes zwangsläufig ein erheblicher Berechnungsaufwand. Bei einem Bild mit 320×240 Bildpunkten, einer Blocküberlappung von 18 Bildpunkten, einer Größe des Abtastfensters von 20×20 Pixeln und einer Skalierungsschrittweite von $s = 1.2$, entsteht somit nach Gleichung 3.9 unter Vernachlässigung des Randproblems die Anzahl F von ca. 50.000 zu untersuchenden Fenstern in K verschiedenen Skalierungen. Die Funktion *int* symbolisiert die Wandlung in die nächst kleinere Ganzzahl. Durch Hinzunehmen vorhandener Farbinformationen über die vorgestellten Ansätze kann der Suchbereich für praktische Anwendungen

aber bereits wieder stark reduziert werden, so etwa um ca. 80% bei Bild 3.4.

$$F = \sum_{i=0}^{K} \text{int}\left(\frac{s^{-i}\text{Breite}-\text{Überlappung}}{\text{Blockgröße}-\text{Überlappung}}\right) \times \text{int}\left(\frac{s^{-i}\text{Höhe}-\text{Überlappung}}{\text{Blockgröße}-\text{Überlappung}}\right) \tag{3.8}$$

$$K = \text{int}\left(\frac{\ln\left(\frac{\min(\text{Breite},\text{Höhe})}{\text{Blockgröße}}\right)}{\ln(s)}\right) \tag{3.9}$$

Die Parameter bei der pyramidalen Abtastung geben direkt die Eigenschaften des Klassifikators bezüglich Translation und Skalierung vor. Bei einem Detektor, welcher robust gegenüber Verschiebungen und Maßstabsänderungen ist, können die Schrittweiten entsprechend erhöht werden.

Wie aus Darstellung 3.7 ersichtlich, läuft die Auswertung aller Abtastfenster auf ein Zweiklassenproblem hinaus. Bei einer Vorlagengröße von 20×20 Grauwerten mit einer Grauwertkodierung von 8 Bit ($= 2^8$) ergeben sich bereits $256^{400} = 2^{3200}$ mögliche Grauwertkombinationen. Bei einer angenommenen Erdbevölkerungszahl im Jahr 2000 von ca. 6 Milliarden Menschen $\approx 2^{32}$ ergibt sich hieraus, dass man mit einer solchen Vorlage zumindest theoretisch 100 mal die gesamte Population bezüglich der Frontalbilder abbilden könnte. Für eine robuste Modellbildung sind somit bereits vor der Klassifikation geeignete diskriminative Maßnahmen zu treffen, um die Erscheinungsform der Klasse Gesicht möglichst kompakt werden zu lassen.

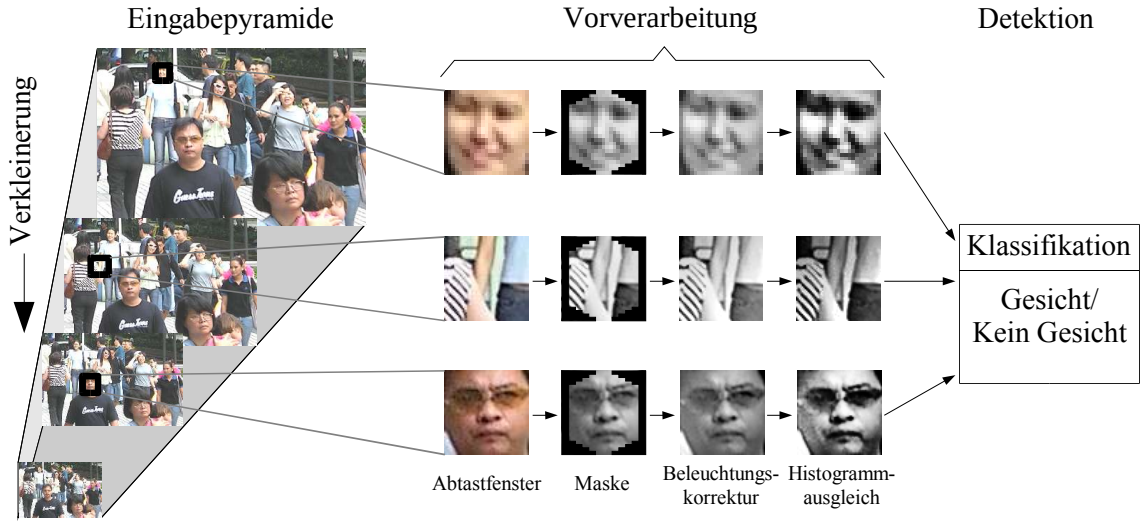


Abbildung 3.7: Eingabepyramide mit blockweiser Abtastung und Vorverarbeitung

3.3.1 Definition der Erscheinungsform Gesicht

Vor der Suche nach einem geeigneten Klassifikator muss im Vorfeld die Frage erörtert werden, welche Erscheinungsform zur Findung und Erkennung von frontalen Gesichtsbildern herangezogen werden kann und soll. Beispielsweise könnte noch die Frisur, die Ohren und ein Teil des Halses zum Gesichtsbereich hinzugezählt werden. Problem hierbei ist aber die breite Streuung der genannten Elemente, so dass vorzugsweise ein quadratischer Ausschnitt

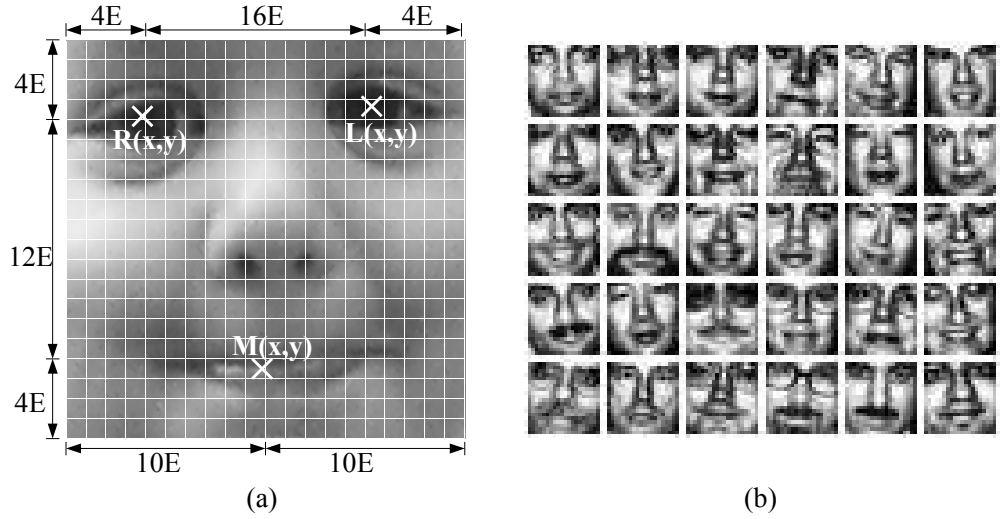


Abbildung 3.8: Idealtypischer Gesichtsausschnitt (a) sowie zufällig verschobene, skalierte und gedrehte Positivbeispiele der Klasse frontaler Gesichter (b)

um die Augen, die Nase und den Mund als Gesichtsbereich definiert wird. Dieser markante ideale Gesichtsbereich wird durch Ankerpunkte nach Grafik 3.8 definiert. Die Stützpunkte selbst basieren auf MPEG-4 konformen Markierungen, welche die Mittelpunkte der Augen und des Mundes repräsentieren [Ost98]. Eine vollständige Auflistung aller relevanten Punkte befindet sich im Anhang A.2. Die Nutzung der abgebildeten Ankerpunkte wie Augen und Mund sowie deren empirisch ermittelte Abstände haben sich als äußerst funktional und robust zur Repräsentation von normalisierten Gesichtsausschnitten erwiesen. Verfahren zur automatischen Suche und Findung der zugrunde liegenden Stützpunkte werden in einem späteren Kapitel ansatzweise vorgestellt. Die Koordinaten des linken Auges sind mit $L(x, y)$, die des rechten mit $R(x, y)$ bestimmt. Die Lage des Mittelpunktes des Mundes sei durch $M(x, y)$ gegeben. Dies entspricht den Punkten 3.5, 3.6 und 2.2 in der MPEG-4 konformen Notation. Unabhängig von der Bildgröße sollen die Augenmittelpunkte jeweils 4 Einheiten von den oberen äußeren Ecken eines quadratischen Blockes mit 20×20 Rastereinheiten gelegen sein. Der Mittelpunkt des Munds befindet sich in der Blockmitte mit einem Abstand von ebenfalls 4 Pixeln zum unteren Bildrand.

Bezüglich der in der Einleitung erläuterten Freiheitsgrade wird die ideale Erscheinungsform zur Detektion frontal aufgenommener Gesichter folgendermaßen erweitert:

- Eine Abweichung der Skalierung zwischen Raster und Bild darf bis zu 10% betragen.
- Eine Translation der einzelnen Koordinaten von maximal 2 Bildpunkten ist erlaubt.
- Die Kopfneigung innerhalb der Ebene darf bis zu $\pm 15^\circ$ abweichen.
- Rotationen um beide Achsen aus der Bildebene, also Azimut und Elevation, sind ebenfalls mit bis zu $\pm 15^\circ$ erlaubt.
- Rückweisung gleichmäßiger Texturen und Bereiche durch minimalen Kontrasthub von 40 Grauwerteinheiten.

- Zugelassen sind alle Personen unabhängig ihrer ethnischen Zugehörigkeit mit und ohne Bärte, Brillen etc. sowie allen natürlichen Gesichtsausdrücken.

In Abbildung 3.8b sind einige Positivbeispiele nach obiger Definition aufgezeigt. Besonders durch die eingeführte Variation bezüglich Translation und Skalierung wird erreicht, dass nicht alle Positionen mit allen Skalen innerhalb des gegebenen Bildes auf das Vorkommen von Gesichtern untersucht werden müssen. Eine zusätzliche Aufweichung der Definition würde zwar die Suche nochmals beschleunigen, gleichzeitig aber die Kompaktheit der Klasse dramatisch reduzieren. Der implementierte Ansatz hat sich als funktionaler Kompromiss zwischen kompakten Modellen und Reduktion des Rechenaufwands erwiesen.

Für das Training der Klasse Gesicht wurden ca. 10.000 Bilder mit den geforderten Eigenschaften aus verschiedenen verfügbaren Gesichtsdatenbanken bereitgestellt. Eine Liste der verwendeten Datenbanken befindet sich im Anhang A. Vor dem Training der Klassifikatoren wird versucht, die Erscheinungsform von Gesichtern durch Operationen im Bildbereich weiter zu vereinheitlichen bzw. normalisieren.

3.3.2 Vorverarbeitung von Bildausschnitten

Unter nicht idealtypischen Bedingungen erstellte Gesichtsaufnahmen werden neben einer nicht frontalen Position im Besonderen aufgrund der äußeren Beleuchtungseinflüsse stark variieren. So ist beispielsweise die absolute Helligkeit des Bildes zum einen unbekannt, zum anderen von Aufnahme zu Aufnahme nicht konstant. Eine weitere Problematik ergibt sich aufgrund der verschiedenen Positionen der Lichtquelle relativ zum aufgenommenen Objekt. So kann das Gesicht von der Seite, von oben oder direkt von vorn angestrahlt werden. Die Kombination beider Effekte wirkt der angestrebten geschlossenen Darstellungsform der Gesichtsklasse entgegen. Zur Milderung dieser störenden Einflüsse existieren jedoch bildbasierte Algorithmen zur Bildverbesserung, mit denen der beschriebene Einfluss in erster Näherung eliminiert werden kann. Obwohl eine globale Anwendung auf das zu untersuchende Bild weitaus effizienter wäre, müssen alle zu untersuchenden Bildausschnitte separat vorverarbeitet werden.

Im ersten Schritt wird zur Normalisierung der Lichtquellenposition eine zweidimensionale lineare Funktion nach Gleichung 3.14 angenähert, die mögliche Beleuchtungsgradienten in horizontaler sowie vertikaler Richtung ausgleicht [Sun98]. Je nach aktueller Position und mittlerer Abweichung der Intensitäten in der linken und rechten bzw. der oberen und unteren Bildhälfte nach den Gleichungen 3.10 bis 3.13 wird im Ausgangsbild positionsabhängig Pixel für Pixel eine entsprechende Intensität hinzugefügt bzw. subtrahiert.

$$m_x^l = \frac{\sum_{x=1}^{\text{Breite}/2} \sum_{y=1}^{\text{Höhe}} i(x,y)}{\text{Breite}/2 \cdot \text{Höhe}} \tag{3.10}$$

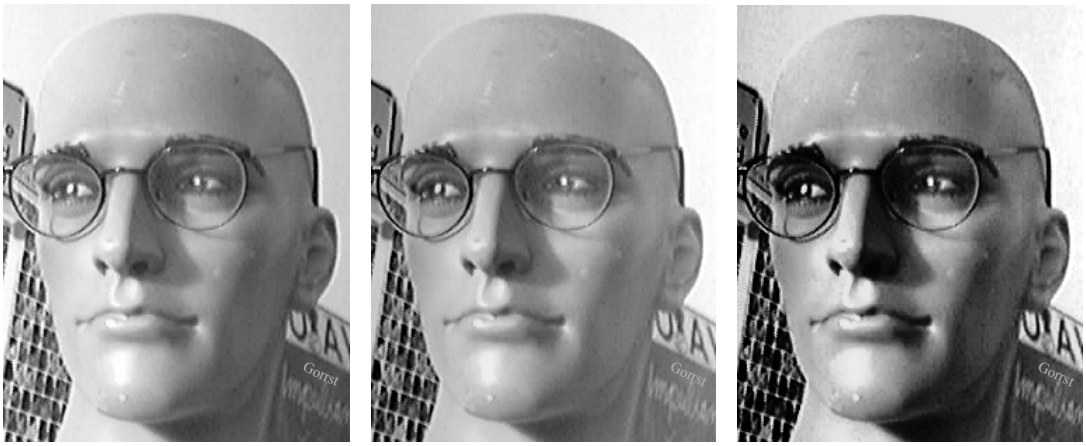
$$m_x^r = \frac{\sum_{x=\text{Breite}/2}^{\text{Breite}} \sum_{y=1}^{\text{Höhe}} i(x,y)}{\text{Breite}/2 \cdot \text{Höhe}} \tag{3.11}$$

$$m_y^o = \frac{\sum_{y=1}^{\text{Höhe}/2} \sum_{x=1}^{\text{Breite}} i(x,y)}{\text{Breite} \cdot \text{Höhe}/2} \tag{3.12}$$

$$m_y^u = \frac{\sum_{y=\text{Höhe}/2}^{\text{Höhe}} \sum_{x=1}^{\text{Breite}} i(x,y)}{\text{Breite} \cdot \text{Höhe}/2} \tag{3.13}$$

$$o(x,y) = i(x,y) - \frac{2(m_x^r - m_x^l)}{\text{Breite}} \cdot (x - \frac{\text{Breite}}{2}) - \frac{2(m_y^u - y_x^o)}{\text{Höhe}} \cdot (y - \frac{\text{Höhe}}{2}) \tag{3.14}$$

Da durch die oben getroffene Klassendefinition eine Präsenz von Hintergrundpixeln nicht immer ausgeschlossen werden kann, wird zur Vermeidung von Störeffekten eine in erster Näherung ovale Maske zur Ausblendung vorgeschaltet, wie in Abbildung 3.7 gezeigt. Der Einfluss der beschriebenen Beleuchtungsnormalisierung ist mit Hilfe einer seitlich bestrahlten Schaufensterpuppe in Abbildung 3.9 ersichtlich. Nach der ersten Verarbeitungsstufe wird



(a) (b) (c)

Abbildung 3.9: Beleuchtungsnormalisierung: (a) Originalbild, (b) beleuchtungskorrigiertes Bild (c) histogrammnormalisiertes und beleuchtungskorrigiertes Bild

zur Kontrastoptimierung noch die Normalisierung des Histogramms angewendet. Dieses Standardverfahren trägt dafür Sorge, dass die auftretenden Pixelintensitäten innerhalb des Histogramms gleichverteilt sind [Gon87]. Als Konsequenz wird der Kontrast innerhalb das aktuellen Testfensters auf ein Maximum angehoben. Der Einfluss der Histogrammnormalisierung ist ebenfalls im Abbildung 3.9c verdeutlicht.

3.3.3 Verschmelzen mehrerer Hypothesen

Durch die pyramidale Abtastung und Auswertung sich überlappender Bereiche kann es vorkommen, dass neben falschen Positionen nicht nur eine ideale Gesichtsposition, sondern eine Schar an Detektionsergebnissen im näheren Umfeld gefunden wird. Nach der Abtastung müssen die gefundenen Hypothesen somit zu einer, bei der Anwesenheit mehrerer Gesichter entsprechend mehreren Hypothesen zusammengefasst werden. Diese Zusammenfassung kann nach Rowley in vier Schritten über folgende Heuristik erfolgen [Row99]:

1. Bestimmung aller möglichen Einzeldetektionen an allen Positionen und Größen innerhalb des Bildes.

2. Zählung der Nachbarn einer Detektionen mit vorgegebener Abweichung der Entfernung innerhalb der gleichen Skalierung sowie über eine vorgegebene Größenänderung hinweg.

3. Hypothesen, welche mit ihrer Auftrittshäufigkeit über einer Schwelle liegen, werden durch Mittelung zusammengefasst.

4. Wegfall sich überlappender Detektionen durch Elimination der Gruppe mit geringerer Anzahl von Nachbarn bzw. akkumulierter Gesamtwahrscheinlichkeit.

Die Zusammenfassung mehrerer Hypothesen ist exemplarisch in Beispiel 3.10 dargestellt. Die Parametrisierung wurde dabei so gewählt, dass die verbleibenden Hypothesen mindestens drei direkte Nachbarn mit einem Mittelpunktabstand unter drei Pixeln, sowie einer Größenänderung um maximal den Faktor 1.2 aufweisen dürfen. Durch diese Heuristik können die gegebenen Einzelhypothesen optimal zu einer richtigen zusammengefasst werden. In

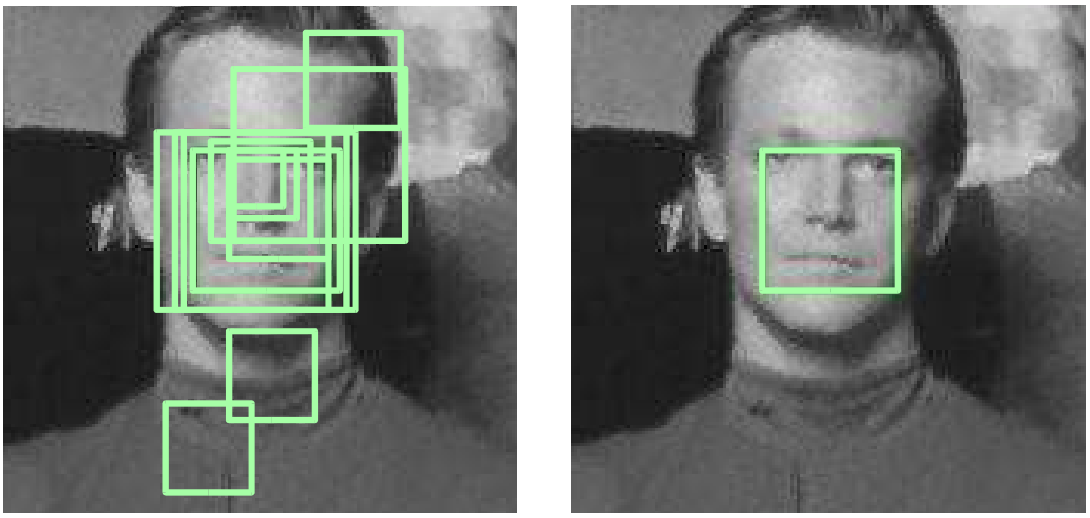


Abbildung 3.10: Zusammenfassung von multiplen Detektionen: Links alle Einzeldetektionen und rechts das Ergebnis nach der Zusammenfassung

den folgenden Kapiteln werden verschiedene Methoden zur Klassifikation von Bildausschnitten nach dem blockbasierten Ansatz evaluiert und auf die Verwendbarkeit für ein robustes Gesichtstracking untersucht.

3.3.4 Findung mit Eigenfaces

Eine der ältesten Ansätze zur Detektion von Gesichtern ist die Verwendung der sogenannten Eigenfaces [Sir87, Tur91a], welcher in die Gruppe der bildbasierten Verfahren mit Modellierung linearer Untermengen einzustufen ist.

Die auf Daten ähnlich den Beispielen nach Abbildung 3.8b berechneten Eigenvektoren repräsentieren hierbei die Achsen eines Koordinatensystems, in welchem Gesichter besonders effizient repräsentiert werden können. Bei einzelner Betrachtung der Eigenvektoren aus dem ursprünglichen System entstehen die sogenannten Eigenfaces.

Zur Bestimmung der Eigenfaces wird ein Ensemble mit N Prototypen benötigt. Zunächst werden die vorliegenden zweidimensionalen Bilddaten mit den Gesichtsausschnitten in Spaltenvektoren S umgewandelt, von denen dann der Mittelwert M bestimmt wird. Daraus ergibt sich die Matrix $A = [S_1 - M, \ldots, S_N - M]$, von der die Kovarianzmatrix $C = AA^T$ bestimmt wird. Im letzten Schritt werden die Eigenvektoren bzw. Eigenfaces der Kovarianzmatrix C mit Hilfe der in Kapitel 2.4 vorgestellten PCA bestimmt.

In der Praxis hat es sich gezeigt, dass zur Repräsentation von Gesichtern unter Tolerierung eines Fehlers nur die Gewichte der ersten Vektoren $w_k = U_k^T \cdot (Y - M)$ mit den größten Eigenwerten maßgebend sind. In den Eigenfaceraum transformierte Bilder können durch die Linearkombination $Y = \sum_{k=1}^{K} w_k \cdot U_k + M$ wieder in den Bildbereich zurück transformiert werden. Die Abbildung 3.11 zeigt die Zerlegung eines gegebenen Gesichtes in den Mittelwert sowie die ersten fünf gewichteten Eigenfaces. Zugrunde liegende Idee bei

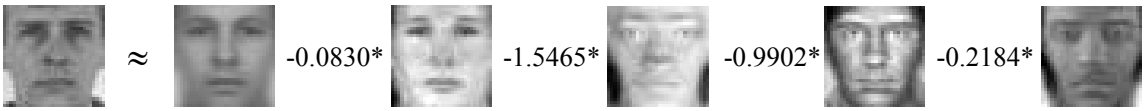


Abbildung 3.11: Zusammensetzung eines Gesichtsbildes aus einem Durchschnittsgesicht und den gewichteten ersten fünf Eigenfaces

der Gesichtsdetektion mit Eigenfaces ist, dass die Rücktransformierte eines gesichtsähnlichen Bildausschnitts durch die Linearkombination von Eigenfaces ebenfalls wieder wie ein Gesicht aussieht. Der Euklidische Abstand zum Originalbild ist somit gering. Nicht gesichtsähnliche Ausschnitte, deren Rücktransformierte ebenfalls durch die Linearkombination von Eigenfaces gebildet wird und zwangsläufig gesichtsähnliche Züge aufweist, müssen somit einen größeren Abstand zum Original aufweisen. Die Entscheidung *Gesicht vorhanden* bzw. *Kein Gesicht vorhanden* kann über eine Schwelle Θ nach der Bedingung 3.15 bestimmt werden.

$$\Theta > ||\Phi - \Phi_f|| \tag{3.15}$$

Dabei ist Φ ist der zu testende Bildausschnitt und Φ_f seine Rückprojektion. Über die Schwelle Θ kann die Empfindlichkeit des System gesteuert werden. Alternativ könnte auch die berechnungsintensivere Korrelation der beiden Ausschnitte herangezogen werden.

In der Praxis hat die Gesichtsdetektion mit Eigenfaces heute einen eher geringeren Verbreitungsgrad. Obwohl die Berechnung für einen zu untersuchenden Ausschnitt verhältnismäßig schnell ist, zeigt sich der Ansatz als äußerst empfindlich gegenüber Änderungen bezüglich Translation, Rotation oder Skalierung. Dies wirkt sich auf eine erschwerte Wahl der Schwelle Θ aus.

3.3.5 Gesichtsdetektion mit neuronalen Netzen

Anliegen bei diesem Verfahren ist das automatisierte, heuristische Erlernen Gesichtsmuster beschreibender Eigenschaften über gegebene Trainingsdaten. Das erfolgreichste System zur Detektion mit NN wurde von Rowley et al. eingeführt und wird iterativ in mehreren Schritten trainiert [Row98]. Nach der kanonischen Vorverarbeitung werden die durch pyramidale Abtastung entstandenen 20×20 elementigen Intensitätsmatrizen an die Eingangsschicht eines NN angelehnt. Nach dem Anlegen werden die Eingangswerte durch das Netzwerk propagiert, so dass in der Ausgangsschicht Werte erzeugt werden, die eine Aussage über das Vorhandensein eines Gesichtes zulassen.

In der Literatur wird für diesen Zweck ein NN mit mehreren verdeckten Schichten, den so genannten rezeptiven Feldern vorgeschlagen, die jeweils bestimmte Gesichtselemente innerhalb des gesamten präsentierten Gesichtsausschnitts beurteilen [Row98]. Diese retinale Struktur ist an den biologischen Aufbau der Netzhaut angelegt. Die Schichten des Netzes nach Abbildung 3.12 können in die folgenden drei Gruppen untergliedert werden:

1. Vier Felder unterteilen das Eingangsbild in quadratische Bereiche mit 10×10 Pixeln ohne Überlappung.

2. Die zweite Gruppe umfasst 16 ebenfalls quadratische Unterbereiche mit je 5×5 Pixeln.

3. Bei den letzten Feldern handelt es sich um 6 überlappende horizontale Streifen mit 20×5 Bildpunkten.

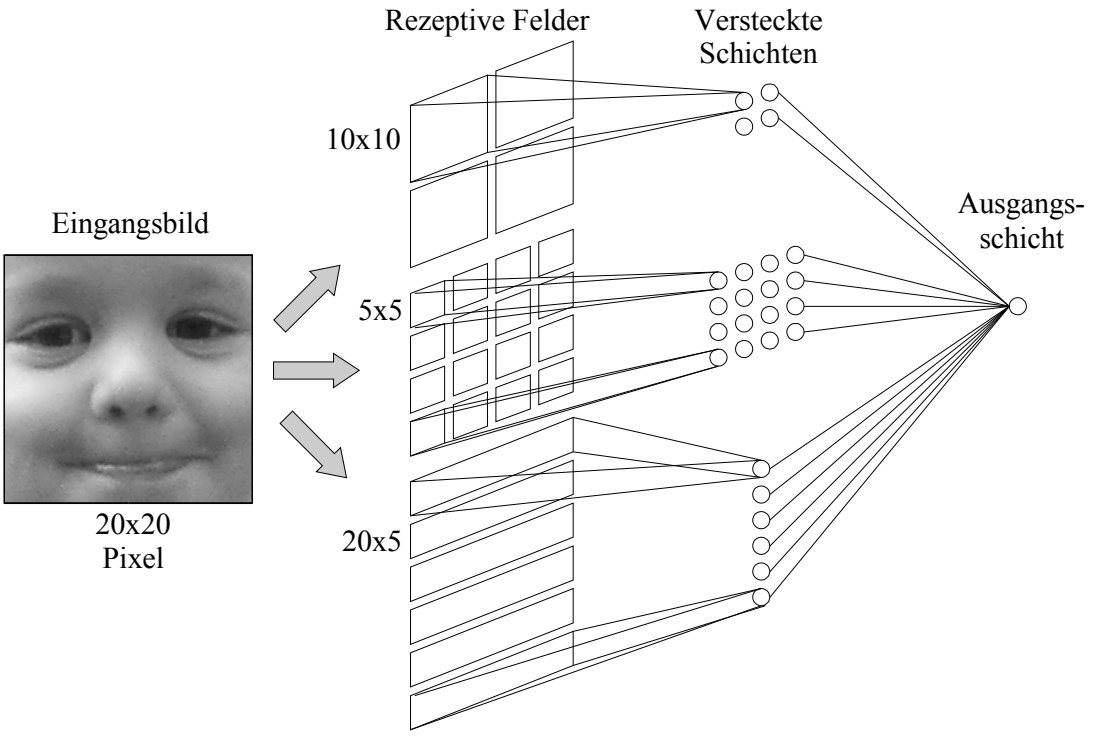


Abbildung 3.12: Retinale Netzstruktur bestehend aus 26 rezeptiven Feldern

Alle Ausgänge der insgesamt 26 Eingangsfelder werden jeweils auf eine verstecke Schicht bestehend aus neun Zellen geführt. Die Ausgangsschicht besteht aus zwei Neuronen. Die Ein- und Ausgänge der Teilnetze sind jeweils voll miteinander verknüpft. Bei allen Knoten in den Hidden Layern wird der hyperbolische Tangens Sigmoid als Aktivierungsfunktion verwendet.

Die Netzausgänge werden während eines iterativen Trainings so bestimmt, dass bei einem positiven Gesichtsbeispiel der Vektor $\mathbf{y} = [1, -1]^T$ und bei einem Negativbeispiel $\mathbf{y} = [-1, 1]^T$ ausgegeben wird. Durch die Erweiterung des negativen Ausgabeneurons kann

die Eigenschaft der Diskriminanz bezüglich der beiden zu unterschiedenen Klassen beträchtlich gesteigert werden. In der Anwendungsphase gilt ein Gesicht als erkannt, wenn der erste Ausgang größer und der zweite kleiner als Null sind.

Vor dem Training des Netzes müssen zunächst Positiv- und Negativbeispiele bereitgestellt werden. Neben der bereits diskutierten Datensammlung für die Klasse *Gesicht*, stellt sich die Frage, welche Exemplare für die Klasse *Nicht Gesicht* ausreichend repräsentativ sind. Abhilfe der Problematik einer fest vorgegebenen Menge an Beispielen kann durch die Anwendung eines Bootstrapping-Algorithmus [Sun98] geschaffen werden, bei dem iterativ neue Negativbeispiele zum Trainingskorpus hinzugefügt werden. Das Training läuft hiernach wie folgt ab:

1. Zunächst wird ein ausreichend großes Kontingent an Positivbeispielen auf die in Kapitel 3.3.1 beschriebene Weise bereitgestellt. Zudem werden zunächst 1.000 Bilder mit zufälligen Pixelintensitäten generiert. Darüber hinaus wird eine wesentlich kleinere Mustermenge mit repräsentativen Beispielen beider Klassen zur Validierung bereitgestellt.

2. Danach wird oben beschriebene Netzwerkarchitektur mit zufällig initialisierten Gewichten erstellt und mit Hilfe des RPROP-Algorithmus solange trainiert, bis der über 20 Iterationen gemittelte Fehler auf die Validierungsdaten ansteigend ist, wodurch ein Overfitting vermieden werden kann. Ein typischer Verlauf der Trainingsfehler ist im Diagramm 3.13a abgebildet.

3. Das bisher trainierte Netz wird nun auf einen Satz von Bildern angewendet, bei denen im Vorfeld bekannt ist, dass sie keine Gesichter enthalten. Das Material stammt aus einer Datenbank mit Standbildern, die aus dem Fernsehen aufgenommenen wurden. Der Inhalt deckt eine Vielzahl zufällig gewählter Sendungen und Anstalten ab. Bei allen Bereichen, bei denen das erste Ausgangsneuron einen Wert größer und das zweite einen Wert kleiner Null aufweisen, handelt es sich somit zwangsläufig um Falschdetektionen.

4. Von allen gemachten Falschdetektionen werden 1.000 zufällig ausgewählte Beispiele zur Trainingsdatenmenge hinzugefügt. Hiernach wird das bisher trainierte Netz mit den erweiterten Trainingsdaten nach Schritt 2 solange weiter trainiert, bis die Anzahl der negativen Trainingsbeispiele gleich groß der Anzahl an Gesichtsbeispielen ist.

Neben der Fehlerkurve der ersten Trainingsiteration ist in Abbildung 3.13 zudem der Verlauf während der zweiten Iteration unter Angabe der Datenmengen aufgetragen. Die gestrichelten Kurven repräsentieren den Verlauf des mittleren quadratischen Fehlers auf den Trainingsdaten, die helle durchgezogene den der Validierungsdaten. Der Verlauf der schwarzen Kurve ergibt sich aus dem Mittelwert der letzten 20 Werte des Fehlers der Validierungsdaten. In Abständen von 20 Iterationen wird geprüft, ob der mittlere Fehler ansteigend ist, wobei gegebenenfalls das Training terminiert wird. Dadurch kann die zuvor diskutierte Überanpassung an die Trainingsdaten verhindert werden.

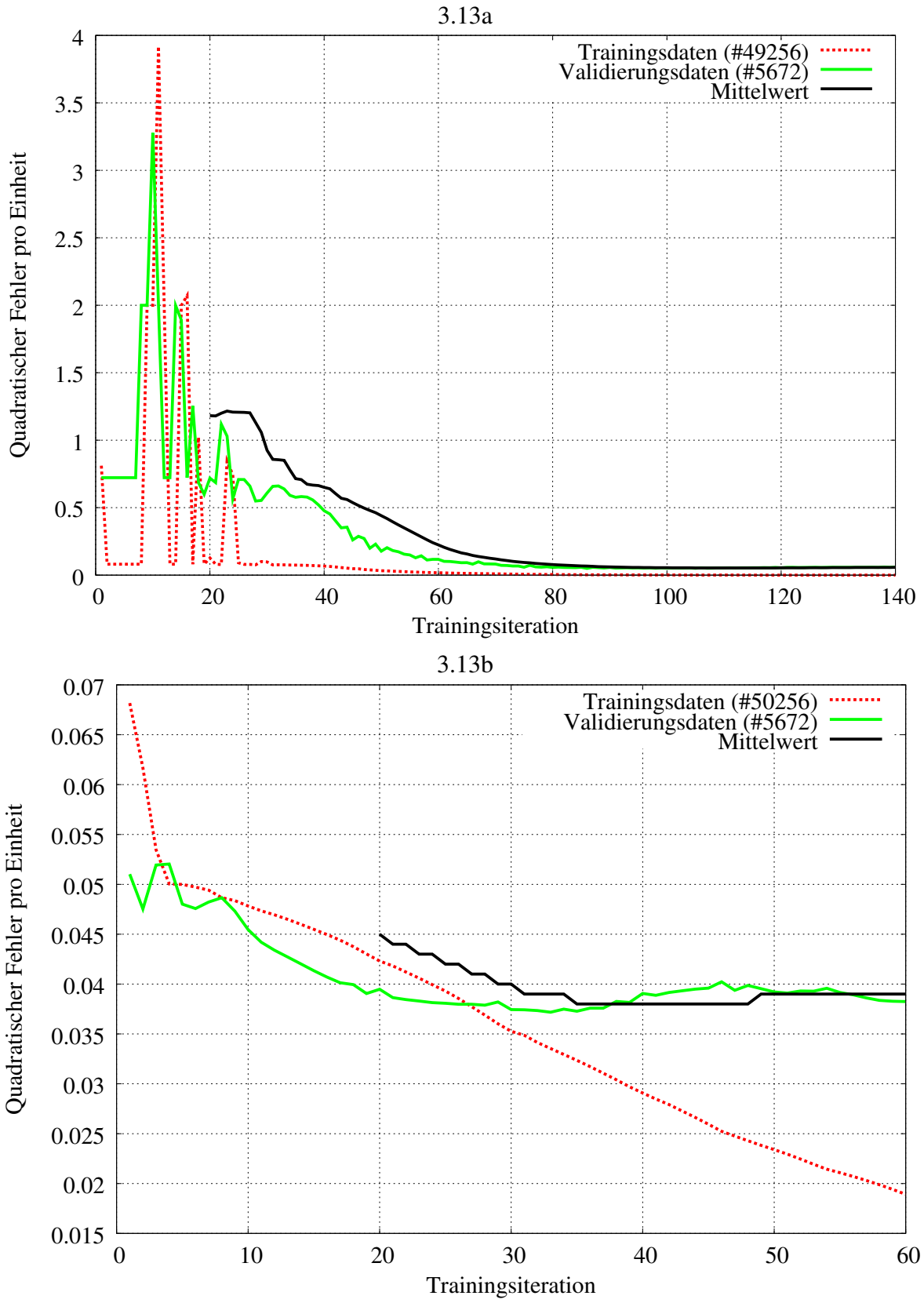


Abbildung 3.13: Trainingsfehler während (a) der ersten und (b) der zweiten Iteration

Mit einem auf diese Art trainierten NN können Gesichter auf robuste Weise detektiert werden. Ein Nachteil, der auf der Verwendung der NN basiert, wirkt sich in einem relativ hohen Rechenaufwand aus. Die Werte der Ausgabeneuronen können darüber hinaus so umgerechnet werden, dass eine Interpretation als A-Posteriori-Wahrscheinlichkeit zulässig ist:

$$p(\lambda_{\text{Gesicht}}|x) = \frac{y_1 - y_2 + 2}{4} \tag{3.16}$$

3.3.6 Gesichtsdetektion mit Integralbildern

Ein weiteres Verfahren in der Gruppe der Detektoren mit blockbasierter Abtastung basiert auf der Auswertung effizient und schnell zu berechnender Merkmale. Vorgestellt wurde dieses Verfahren von Viola und Jones und verwendet sogenannte einfache Klassifikatoren[3] mit niedriger Zuverlässigkeit auf Basis von Schwellwerten [Vio01]. Während einer Trainingsphase werden repräsentative Merkmale innerhalb des Abtastfensters ausgewählt und die Parameter für eine gewichtete Entscheidungsfunktion ermittelt.

Die zu findenden Merkmale sind bei dem verwendeten Verfahren nicht pixelbasiert sondern setzen sich aus erweiterten 2D Haar Basis Wavelets zusammen [Pap98]. Grundlage für diese Merkmale sind typische Intensitätsunterschiede bei Regionen innerhalb des Gesichtes, welche konstant heller bzw. dunkler als ein benachbarter Bereich sind. So ist beispielsweise der Augenbereich in der Regel dunkler als die Stirnpartie. Die zu bestimmenden Intensitätsunterschiede können mit fünf verschiedenen Basisfunktionen nach Abbildung 3.14a–e bestimmt werden.

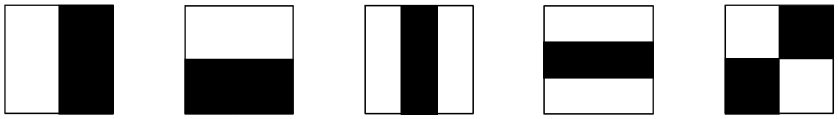


Abbildung 3.14: Fünf verschiedene Gruppen der Basismerkmale

Eine weiße Fläche repräsentiert positiv zu zählende Intensitäten, ein schwarzer Bereich dementsprechend negative. Abgeleitet aus den Basisfunktionen können die Merkmale an verschiedenen Positionen mit variierenden Skalen innerhalb des Abtastfensters liegen. Die Basisgröße des Abtastfensters beträgt auch hier wieder 20×20 Bildpunkte. Als Randbedingung gilt die Höhen- und Breitengleichheit der enthaltenen Flächen. Insgesamt ergibt sich eine Anzahl von 78.462 verschiedenen Varianten der 5 Basismerkmale [Lie02].

Zur effizienten Berechnung der obigen Haar-ähnlichen Merkmale werden Integralbilder herangezogen, welche an einem Punkt (x, y) jeweils die kumulierte Intensität ausgehend vom Ursprung beinhalten, siehe Skizze 3.15a. Für einen Punkt gilt demnach:

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \tag{3.17}$$

[3] aus dem Englischen von *Weak Classifier*

Über die rekursiven Berechnungsvorschriften nach 3.18 und 3.19 kann ein Integralbild in einem Durchlauf bestimmt werden.

$$s(x,y) = s(x,y-1) + i(x,y) \tag{3.18}$$

$$ii(x,y) = ii(x-1,y) + s(x,y) \tag{3.19}$$

Die akkumulierte Intensität einer rechteckigen Fläche D nach Abbildung 3.15b kann nun durch vier Werte des aufgestellten Integralbildes bestimmt werden. Dabei geben die Punkte 1–4 die Eckpunkte des zu bestimmenden Rechtecks an. Der Wert am Punkt 2 repräsentiert somit den Wert der Integralflächen $A + B$, der des Punktes 3 die Flächen $A + C$. Der Wert an der Stelle 4 kennzeichnet die Fläche $A + B + C + D$. Der Wert der Fläche D kann demnach durch die Berechnung $ii_D = ii(x_4, y_4) - ii(x_2, y_2) - ii(x_3, y_3) + ii(x_1, y_1)$ bestimmt werden. Analog können die oben vorgestellten, auf diese Flächen aufbauenden Merkmale, mit wenigen Zugriffen auf die erstelle Integralmatrix berechnet werden. Die Entscheidung,

Abbildung 3.15: Bestimmung eines Integralbildes (a) und Berechnung eines rechteckigen Bereiches (b)

ob im Abtastfenster x ein Gesicht vorhanden ist, kann auf Basis eines einzelnen Merkmals f_j durch die triviale Entscheidungsfunktion[4] h_j nach Gleichung 3.20 über einen Schwellwert bestimmt werden. Diese Schwellen Θ_j müssen in einer Trainingsphase zusammen mit einer Parität p_j geschätzt werden.

$$h_j = \begin{cases} 1 & \text{wenn } p_j f_j(x) < p_j \Theta_j \\ 0 & \text{andernfalls} \end{cases} \tag{3.20}$$

Unter der Verwendung einer Gesamtmenge von T gewichteten Merkmalen kann der folgende globale Klassifikator gefunden werden:

$$h(x) = \begin{cases} 1 & \text{wenn } \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & \text{andernfalls} \end{cases} \tag{3.21}$$

Neben der Findung der Parameter α, Θ und p müssen gleichzeitig die repräsentativsten Merkmale selbst gefunden werden. Die *intelligente* Selektion der Merkmale[5] wird mit Hilfe des

[4] In der Literatur als *Weak Classifier* bezeichnet

[5] Engl.: Feature Selection

AdaBoosting Algorithmus gelöst [Sch97]. Bei diesem Algorithmus werden die aussagekräftigsten Merkmale in T Schritten iterativ gefunden und durch Gewichtung zu einem leistungsfähigen Gesamtklassifikator[6] zusammengefasst. Der Boosting Algorithmus ist im Anhang B.5 abgedruckt.

Als weitere leistungssteigernde Maßnahme der Detektion wird eine Kaskadierung der vorgestellten Klassifikatoren vorgenommen. Motiviert ist dieser Ansatz durch die Tatsache, dass sich die Leistung des Detektors nach einer gewissen Anzahl an Merkmalen nicht mehr signifikant ändert. Mit dem Ziel der Weiterleitung nahezu aller Gesichtsausschnitte von einer an die nächste Stufe schaltet man mehrere Klassifikatoren seriell hintereinander, wobei in jeder Kaskade möglichst viele Bilder ohne gesichtsähnliche Muster herausgefiltert werden sollen. Beim Training werden daher jeder Kaskade nur noch die reduzierten Datensätze präsentiert, wodurch stärker spezialisierte Entscheidungsfunktionen gefunden werden.

Nach dem hier beschriebenen Vorgehen wird ein Detektor mit 40 Merkmalen in 8 Kaskaden trainiert [Ars03]. Vorteilhaft an dieser Detektion ist die besonders schnelle und effiziente Berechnung der Merkmale und der anschließenden Klassifikation. Aufgrund der äußert hohen Anzahl zu prüfender Merkmale während des vorgeschalteten Selektionsprozesses erweist sich das Training andererseits als kostspielig und langwierig.

3.3.7 Gesichtsdetektion mit Support Vektor Maschinen

Das letzte der vorgestellten blockbasierten Verfahren ähnelt dem bereits vorgestellten System auf Basis von NN nach Kapitel 3.3.5. Alternativ werden hier zur Klassifikation SVM verwendet, was durch die besonders guten diskriminativen Eigenschaften motiviert ist [Hea98].

Analog zum oben vorgestellten Bootstrapping wird zunächst ein Basiskorpus mit positiven und negativen Trainingsbeispielen erstellt. Mit diesem wird eine erste SVM initialisiert. Im folgenden Schritt werden ebenfalls wieder 1.000 zufällig ausgewählte Falschdetektionen zu einem erweiterten Trainingssatz hinzugefügt, mit welchem eine verbesserte SVM bestimmt wird. Terminiert wird auch hier spätestens, wenn die Anzahl der positiven und negativen Beispiele gleich groß ist, oder die Anzahl an Falschdetektionen unter eine gegebene Schwelle fällt. Der Verlauf einer zweidimensionalen Klassengrenze mit realen Beispieldaten ist schematisch in Abbildung 3.16 gezeigt. Als Kernfunktion werden polynomiale Funktionen verwendet. Während der Trainingsphase kann beobachtet werden, dass die Anzahl der Falschdetektionen bereits nach wenigen Iterationen beachtlich sinkt. Dies kann auf die Tatsache zurückgeführt werden, dass für eine robuste Schätzung der Trennfläche lediglich die involvierten Stützvektoren entscheidend sind. Es ergab sich insgesamt eine Anzahl von 1.077 Stützvektoren mit einer Dimension von je 400 Elementen.

Zusammenfassend können die besonders guten diskriminativen Eigenschaften der SVM zwischen beiden Klassen als herausragend angesehen werden. Problematisch zeigt sich hingegen der im Vergleich zu den einfachen Merkmalen ebenfalls als hoch anzusehende Rechenaufwand.

[6] In der Literatur als Strong Classifier bezeichnet

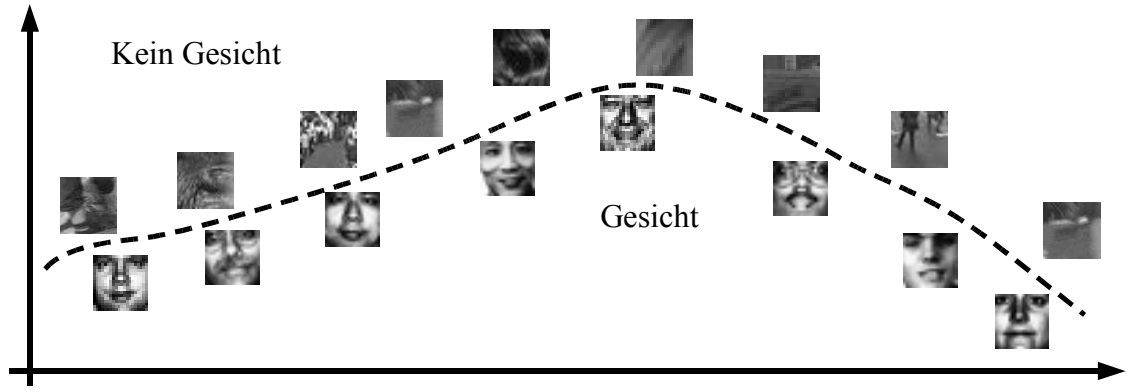


Abbildung 3.16: SVM Klassengrenze zwischen *Gesicht* und nicht *Gesicht*

3.3.8 Vergleich blockbasierter Ansätze

Vor der Erweiterung zu einem blickrichtungsinvarianten Gesichtsdetektor und zur Entwicklung eines Systems zur Verfolgung von Gesichtern wird der im Hinblick auf die folgenden Punkte optimale blockbasierte Detektor gesucht:

- Leistungsfähigkeit
- Komplexität / Schnelligkeit
- Erweiterbar- und Implementierbarkeit

Zur qualitativen Beurteilung der Ansätze wurden die zuvor trainierten Systeme in verschiedenen Entwicklungsumgebungen implementiert und auf einem Standardkorpus zur Gesichtsdetektion des *Massachusetts Institute for Technology* (MIT) mit aufrechten frontalen Gesichtern evaluiert [Sun98], siehe Anhang A. Bei den Untersuchungen hat sich bereits im Vorfeld gezeigt, dass die Detektion mit Eigenfaces aufgrund ihrer zu hohen Empfindlichkeit gegenüber Variationen in der Erscheinungsform des zu erkennenden Objekts nur von akademischen Interesse ist und als praktikable Möglichkeit ausgeschlossen werden muss. In einer nachgeschalteten Stufe eignet sie sich aber für eine zusätzliche Feinlokalisierung.

Wie zur Messung von Detektionsergebnissen üblich, wurde einerseits bewertet zu welchem Prozentsatz die enthaltenen Gesichter gefunden wurden, andererseits wurde die Anzahl der Falschdetektionen gemessen. Nach Lienhart [Lie02] gilt ein Gesicht dann als korrekt erkannt, wenn

1. die Höhe bzw. Breite der Detektion innerhalb $\pm 50\%$ der ursprünglichen Höhe bzw. Bereite liegt und
2. der geometrische Abstand der Mittelpunkte nicht mehr als 30% der Breite abweicht.

Andernfalls gilt der gefundene Bereich als Falschdetektion. Die nach diesem Bewertungsmaßstab in Tabelle 3.1 zusammengefassten Detektionsraten decken sich mit den in der Literatur vorgestellten Ergebnissen [Hea98, Row98, Vio01, Yan02]. Beim Integralbild basierten

Verfahren	Detektionsrate [%]	Falschdetektionen	Empfindlichkeit
NN	90,3	42	hoch
NN	79,9	5	niedrig
Integralbilder	89,0	50	hoch
Integralbilder	77,8	5	niedrig
SVMs	74,2	20	niedrig

Tabelle 3.1: Ergebnisse der vorgestellten blockbasierten Abtastverfahren mit verschiedenen Schwellen bzw. Empfindlichkeiten

Detektor weicht die Anzahl der Falschdetektionen jedoch aufgrund der aus Zeitgründen limitierten Anzahl an berechneten Kaskadenstufen ab [Ars03]. Letztendlich ist für praktische Anwendungen die Anzahl von Falschdetektionen bei akzeptabler Detektionsrate entscheidend. Wie angedeutet kann zur Minderung von *False Positives* eine nachgeschaltete Feinlokalisierung bzw. Rückweisung erfolgen.

Aufgrund der unterschiedlichen Entwicklungsumgebungen ist ein direkter Laufzeitvergleich bezüglich der optimalen Suchgeschwindigkeit im Rahmen dieser Evaluierung nicht möglich. Qualitativ hat sich der Ansatz auf Basis von Integralbildern als Schnellster hervorgehoben und war ca. um Faktor 5 schneller als die vorgestellten NN. Für ein Bild der Größe 320×240 wurde auf einem Linux System mit einem 1.3 GHz Pentium Centrino Prozessor durchschnittlich weniger als eine halbe Sekunde benötigt. Im Vergleich zu den SVM war das Verfahren sogar ca. 20 mal schneller. Das langsame Abschneiden der SVM ist hauptsächlich auf die hohe Anzahl an verwendeten Stützvektoren zurückzuführen.

Für die weiterführenden Untersuchungen wird die Verwendung des NN-gestützten Systems gewählt, da es bezüglich der geforderten Optimalitätskriterien, wie Berechnungsdauer, Leistungsfähigkeit und Erweiterbarkeit den besten Kompromiss darstellt.

3.3.9 Detektion in die Tiefe gedrehter Gesichter

Mit den vorgestellten Verfahren ist es bisher möglich, aufrecht und frontal aufgenommene Gesichter zu finden. Im Folgenden soll es das Ziel sein, in horizontaler Richtung aus der Bildebene gedrehte Gesichter unter Angabe des azimutalen Drehwinkels φ_{diskret} zu detektieren. Eine durch Drehung innerhalb der Ebene entstandene Kopfneigung nach Abbildung 3.1d soll hier nicht von Interesse sein, könnte aber zusätzlich zur pyramidalen Abtastung durch Rotation des zu untersuchenden Bildes gelöst werden. Da in den angestrebten praktischen Anwendungen Aufnahmen mit Elevationen bzw. vertikalen Drehungen aus der Ebene von mehr als $\pm 15^\circ$ eher selten sind, soll diese Problematik ebenfalls vernachlässigt werden.

Durch die Rotation des dreidimensionalen Objekts Kopf bekommt die Klasse Gesicht gegenüber der frontalen Definition eine wesentlich breitere Erscheinungsform. Aufgrund des oben beschriebenen langwierigen Prozesses der Merkmalsselektion wird zur blickrichtungsunabhängigen Detektion von der vergleichsweise schnellen Detektion mit Integralbildern Abstand genommen. Statt dessen wird des vorgestellte Verfahren mit NN zur Erweiterung

gewählt, da es sowohl eine günstige Erkennungsleistung als auch eine ideale Möglichkeit zur angestrebten Funktionalität gewährleistet.

Die Neuerung des vorgestellten Systems liegt in der Parallelisierung eines Gesichtsdetektors mit einer diskreten Schätzung der Blickrichtung in horizontaler Richtung. Hierzu wird der Ausgabevektor von 2 auf 16 Ausgabeneuronen erweitert, wodurch sich analog zu den verwendeten rezeptiven Strukturen eine Vielzahl zusätzlicher verdeckter Schichten ergibt.

Die beiden ersten Werte des Ausgabevektors repräsentieren nach Gleichung 3.16 noch immer das Maß für die Wahrscheinlichkeit, dass am Eingang ein Gesicht anliegt. Die verbleibenden Ausgaben repräsentieren wieder zwei komplementäre Wahrscheinlichkeiten $p(\lambda^{\varphi}_{\text{Gesicht}}|x)$ und $p(\bar{\lambda}^{\varphi}_{\text{Gesicht}}|x)$, dass das Gesicht in einem diskreten Winkel φ zur Ebene vorliegt. Die Wertigkeiten der modellierten Winkel sind $-90°$, $-45°$, $-22,5°$, $0°$, $22,5°$, $45°$ und $90°$. Diese diskreten Perspektiven werden gewählt, da in den verfügbaren Datenbanken ausreichend Material für ein überwachtes Training zur Verfügung steht. Die zusätzliche Schät-

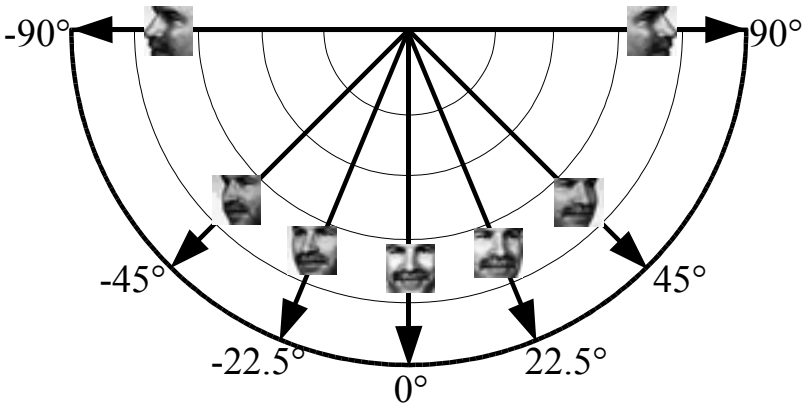


Abbildung 3.17: Diskrete Richtungsschätzung der azimutalen Kopfdrehung

zung der Blickwinkel macht ebenfalls eine Modifikation der Trainingsdaten erforderlich. Bei der Normierung der Ansichten mit Viertel- und Halbdrehung kann nach dem gleichen Verfahren wie bei den Frontalaufnahmen vorgegangen werden. Bei den Profilansichten mit $\pm 90°$ ist allerdings jeweils ein Auge verdeckt, so dass die Ausschnitte über andere Stützpunkte definiert werden müssen. Diese werden über die Ohren durch die Anker 10.4 und 10.10 nach Anhang A.2 charakterisiert. Die hierbei entstehenden Ausschnitte sind in Abbildung 3.17 aufgeführt. Auch hier werden die präsentierten Trainingsdaten zur Steigerung der Robustheit künstlich rotiert und verschoben.

Da eine zu schätzende Kopforientierung üblicherweise auch zwischen zwei idealtypisch gelernten Prototypen liegen kann, müssen die diskreten Ausgaben in kontinuierliche überführt werden. Bei einer einzelnen Hypothese berechnet sich der resultierende Winkel über die sieben mit den Produktionswahrscheinlichkeiten gewichteten Richungsvektoren nach Ausdruck 3.22. Bei zusammengefassten Hypothesen wird der Winkel des Bereichs mit der höchsten Gesichtswahrscheinlichkeit gewählt.

$$\varphi_{\text{kontinuierlich}} = \text{arc} \left[\sum_{\varphi_{\text{diskret}} \in \{0,\pm 22{,}5,\pm 45,\pm 90\}} p(\lambda^{\varphi_{\text{diskret}}}_{\text{Gesicht}}|x) \cdot e^{j\varphi_{\text{diskret}}} \right] \tag{3.22}$$

Für eine Beurteilung des Ansatzes wird das integrierte System auf einer Sequenz eines multimodalen Besprechungsszenarios mit zwei Sitzungsteilnehmern getestet [Ren05]. Hierbei können die Suchbereiche aufgrund der a-priori bekannten Sitzpositionen bereits im Vorfeld auf die rechte bzw. linke Bildhälfte beschränkt werden. In den Beispielen nach Abbildung 3.18 sind neben den gefundenen Gesichtern der Sitzungsteilnehmer die Winkel der geschätzten Kopforientierungen in Form von Vektorpfeilen dargestellt.

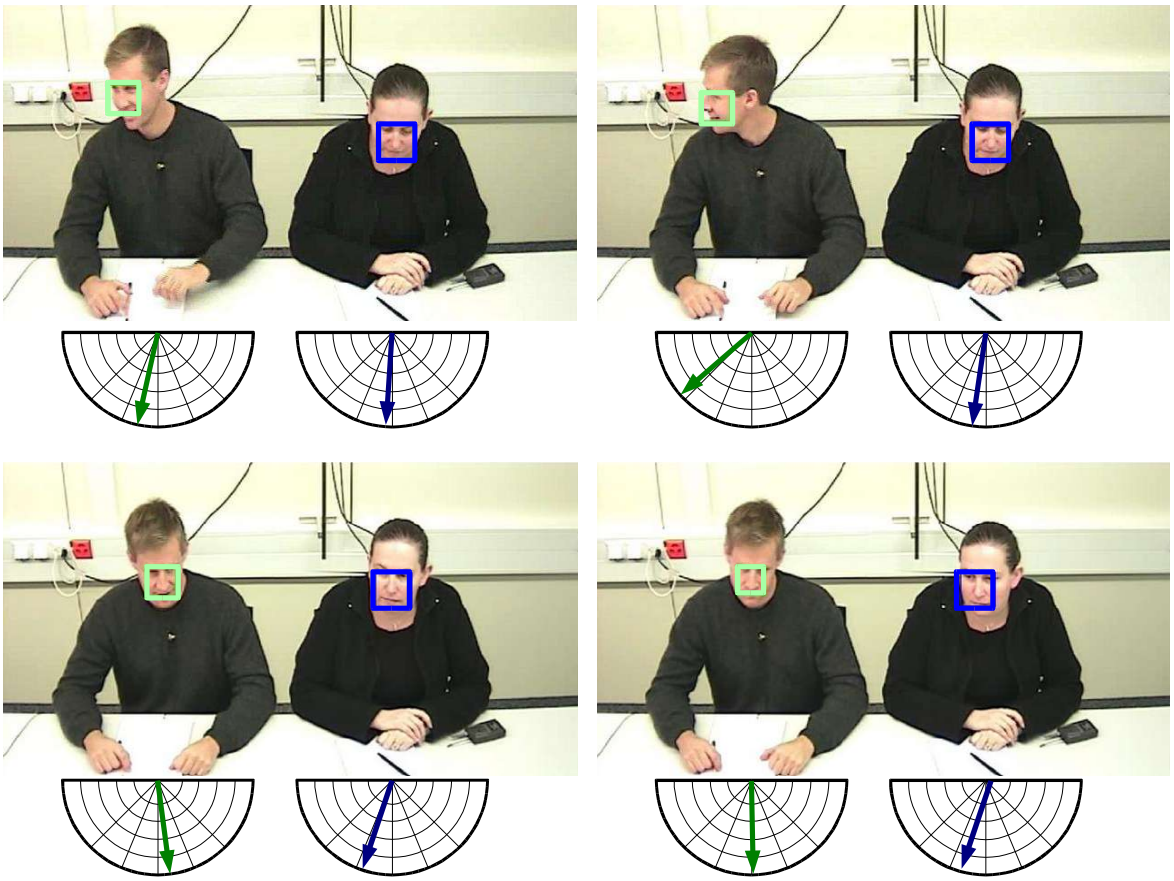


Abbildung 3.18: Detektionsergebnisse mit geschätzter Kopfdrehung für eine Sequenz mit zwei Besprechungsteilnehmern

Eine Auswertung der gefundenen Bereiche in dieser Bildsequenz ergibt eine ausgezeichnete Sicherheit bei der Lokalisierung der in der Bildfolge enthaltenen Gesichter sogar mit aus der Bildebene gedrehten Gesichtern. Die Angabe der Blickrichtung kann hingegen nur als eine grobe Schätzung des tatsächlichen Winkels interpretiert werden. Zum Teil ergaben sich Abweichungen zwischen tatsächlicher und bestimmter Blickrichtung von bis zu $20°$. Die mittlere gemessene Abweichung beträgt ca. $5°$. Eine feinere Unterteilung der diskreten Drehwinkel in Kombination mit mehr Trainingsmaterial kann hier in weiterführenden Versuchen Abhilfe schaffen.

3.4 Gesichtsdetektion in Bildsequenzen mit dem Condensation Algorithmus

Zur Beschleunigung des vorgestellten Systems wird der Suchbereich, in dem Gesichter vermutet werden, in einem zusätzlichen Schritt dramatisch reduziert. Die Idee basiert auf der Annahme, dass sich die Position und Drehung des Gesichts in benachbarten Bildern einer zusammenhängenden Videosequenz nach einer anfänglichen Detektion nur langsam ändert. Es ist somit unwahrscheinlich, dass ein Gesicht in einem Bild auftaucht und im nächsten wieder verschwunden ist. Ferner wird es nicht über eine große Distanz verschoben oder mit signifikant anderer Größe zu finden sein. Der einfachste Ansatz hierzu liegt in der Beschränkung der Suche auf räumlich benachbarte Bereiche. Wie bei der statistischen Personenverfolgung könnten zudem Kalman-Filter [Rig04b] zur Modellierung der Bewegung angewandt werden.

Im Weiteren wird stattdessen ein Ansatz eingeführt, welcher die Wahrscheinlichkeitsverteilungen von Gesichtern mit Hilfe der temporalen Information über Bildsequenzen hinweg verfolgt. Das aus der Literatur nach Isard und Blake zugrunde liegende Verfahren ist unter dem Neologismus *Condensation*[7] [Isa98] zu finden. Daneben sind ebenfalls die Ausdrücke *sequentielle Monte-Carlo-Methode*, *Random Sampling* oder *Partikelfilter* zu finden.

Nach einer anfänglichen Schätzung möglicher Gesichtspositionen in der Initialisierungsphase durch eine erschöpfende Suche über NN-basierte Verfahren pflanzen sich die sogenannten Partikel von einem Bild zum nächsten fort. Auf den vorherigen Beobachtungen beruhend wird dazu ein Satz von N hypothetischen, diskreten Partikelpositionen $s^1 \dots s^N$ vorhergesagt. Nach der Prädiktion werden die neuen Partikel mit Hilfe von Messungen ausgewertet und normalisiert. Typischerweise häufen sich die hypothetischen Partikel an den tatsächlichen Gesichtspositionen, bildhaft ausgedrückt *kondensieren* sie.

Konkret besteht ein Satz von Messwerten für einen hypothetischen Gesichtsbereich aus den folgenden 10 Elementen: den beiden Koordinaten der oberen linken Ecke des Bereichs zum diskreten Zeitpunkt k, $x(k)$ und $y(k)$ sowie der Größenskalierung $s(k)$, der mittleren Kopforientierung $\varphi(k)$ und ihrer Wahrscheinlichkeit $p(\varphi(k))$, dem Gewicht der Messung $\pi(k)$ und zusätzlich die entsprechenden Angaben zum vorhergehenden Zeitpunkt $k-1$ wie $x(k-1)$, $y(k-1)$ und $s(k-1)$. Die Messungen an den vorhergesagten Bereichen werden aus einer Kombination der MLP-Ausgaben und über die Häufigkeit an hautfarbenen Pixeln vorgenommen. Dabei werden jeweils die gemittelten Werte des Ausgabevektors $\mathbf{y} = [y_1, y_2]$ des MLP und die Hautfarbenhäufigkeiten als Wahrscheinlichkeiten für das Partikel i zum Zeitpunkt k interpretiert.

$$p(z_k^{\text{MLP}}|x_k^i) = \frac{y_1 - y_2 + 2}{4} \tag{3.23}$$

$$p(z_k^{\text{Haut}}|x_k^i) = \frac{n_{\text{Haut}}}{n_{\text{ges.}}} \tag{3.24}$$

Die Gesamtwahrscheinlichkeit für ein Gesicht in einem Bereich ergibt sich durch das Produkt der beiden Wahrscheinlichkeiten: $p(z_k|x_k^{(i)}) = p(z_k^{\text{MLP}}|x_k^i) \cdot p(z_k^{\text{Haut}}|x_k^i)$

[7] Aus dem Englischen abgeleiteter Terminus für CONditional DENSitiy PropagATION

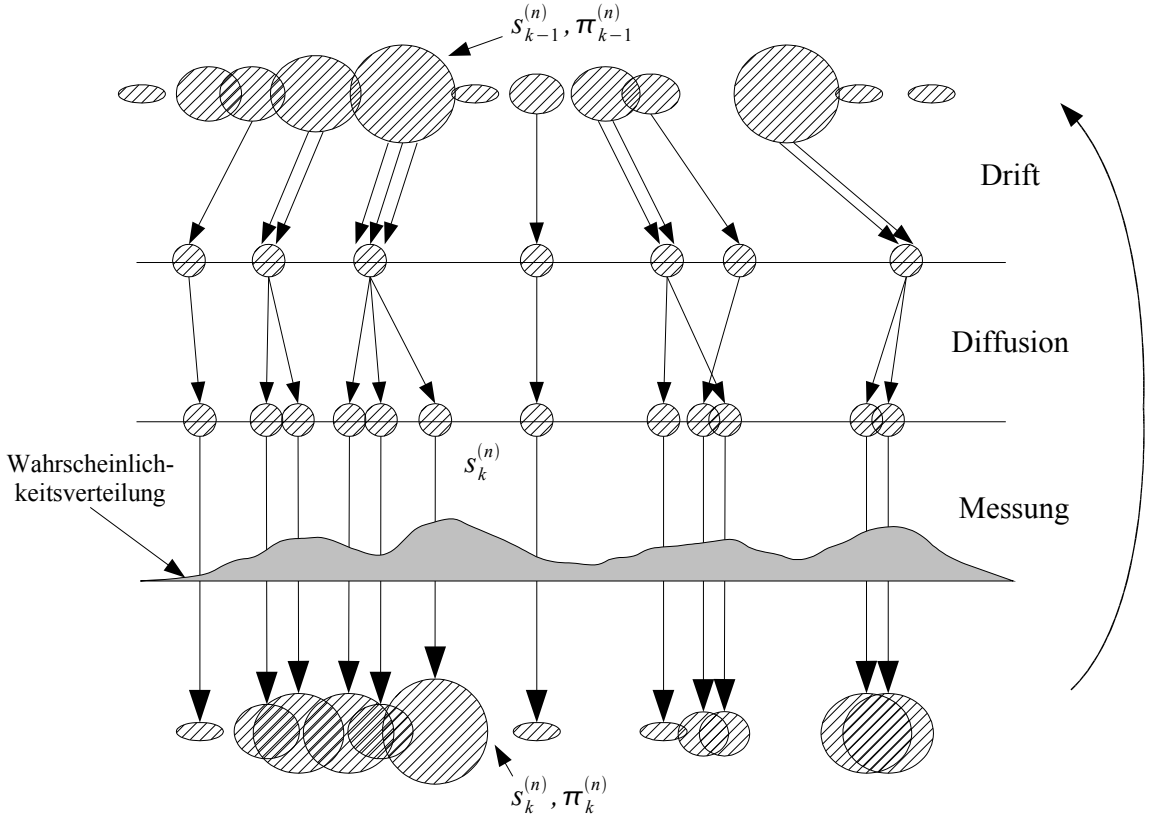


Abbildung 3.19: Iterationsschritt nach dem Condensation-Algorithmus [Isa98]

Basierend auf den Werten einer bereits bestimmten Verteilung oder der Initialisierung können die möglichen Positionen der zu testenden Bereiche über folgende Schritte einer iterativen Berechnungsvorschrift nach Darstellung 3.19 bestimmt werden:

- Deterministische Verschiebung der Partikel durch ein lineares Bewegungsmodell $s_k^{(n)} = As_{k-1}^{(n)}$. Die Parameter von $\mathbf{A}$ werden mit Hilfe gelabelter Daten über ein adaptives lineares Netzwerk (ADALINE) im Vorfeld geschätzt und geben somit die mögliche Dynamik vor. Die Partikel werden entsprechend ihrer Wahrscheinlichkeiten zufällig ausgewählt, andernfalls verworfen. Bei Unterschreitung der vorgegebenen Anzahl N können die Partikel mit den größten Gewichten auch mehrfach verwendet werden. Im Fall, dass keine Partikel mehr vorhanden sind, muss eine neue Initialisierung durchgeführt werden.

- Diffusion bzw. stochastische Drift, durch Addition von normalverteiltem Rauschen zur Modellierung von Systemrauschen und zur Vermeidung der Partikeldegeneration.

- Annäherung an die zugrunde liegende kontinuierliche Verteilung durch Messung an den prädizierten Stellen über die diskreten Wahrscheinlichkeiten $p(z_k^{\text{MLP}}|x_k^i)$ und $p(z_k^{\text{Haut}}|x_k^i)$

- Resampling der Partikelgewichte π_i durch Normalisierung der gemachten Beobachtungen:

$$\pi_i = \frac{p(z|x = s^i)}{\sum_{j=1}^{N} p(z|x = s^i)} \tag{3.25}$$

In der Darstellung 3.19 symbolisieren die Mittelpunkte der Ellipsen die Orte der Messungen, die Größe deren Gewicht. Eine zusätzliche Neuerung des Verfahrens besteht in der Ermöglichung der Findung neu hinzukommender Gesichter. Zu diesem Zweck werden nach einer Iteration 10% der vorgegebenen Partikel an zufällig gewählten Stellen und Größen innerhalb des Bildes initialisiert.

Das auf diese Weise implementierte System wurde ebenfalls im Zusammenhang mit Sitzungsszenarien erprobt [Hei03, Wal04a, Wal04b]. Zudem wird dieser Ansatz in Verbindung mit der Überwachung potentiell auffälligen Verhaltens in Flugzeugen eingesetzt. In der Bildzusammenstellung 3.20 sind beispielhaft die detektierten Gesichtsbereiche eines Fluggastes abgebildet. Die dunklen Quadrate in den Beispielbildern repräsentieren alle Partikel mit Gesichtern, die hellen die Gesamthypothesen nach der Zusammenfassung.

Eine Auswertung zeigt, dass mit dem Algorithmus und einer Anzahl von $N = 25$ Partikeln bereits eine echtzeitfähige Verfolgung von Gesichtern selbst mit geringen Anforderungen an die Hardware möglich wird. Zur Steigerung der Detektionsleistung ist aber eine Partikelanzahl von $N = 100$ anzusetzen. Gesichter können aufgrund der beiden unabhängigen Detektionsansätze über MLP und mit Hautfarben unabhängig von der Drehung des Kopfes in die Tiefe präzise lokalisiert werden. Falls es wie im unteren linken Beispiel dennoch zu Störungen kommen sollte, ist das System selbstständig in der Lage, die richtigen Positionen wiederzufinden. Zudem stellt die kurzzeitige Verdeckung von Gesichtsbereichen, beispielsweise durch Hände, keine nennenswerte Störung dar.

3.5 Gesichtsverfolgung in omnidirektionalen Aufnahmen

Im Besonderen bei Aufnahmen, die einen großen räumlichen Bereich umspannen, ist es aus Kostengründen und einem reduzierten Hardware- und Installationsaufwand wünschenswert, die gesamte Szene mit einer Aufnahmeeinheit abdecken zu können. Ebenfalls zur Aufnahme von Besprechungen wird ein zylindrischer, hyperbolischer Spiegel vor der Kameralinse angebracht, um alle horizontalen Richtungen innerhalb des gesamten Raumes aufnehmen zu können. Ein Beispiel für ein derartiges Bild ist in Abbildung 3.21a gezeigt. Als Konsequenz der verwendeten Spiegel entstehen nicht lineare Bildverzerrungen, welche in der Regel nicht vollständig rekonstruiert werden können. In einer abschließenden Versuchsreihe soll das bereits entstandene System zur Verfolgung von Gesichtern auf omnidirektionale Aufnahmen erweitert werden. Vor dem bekannten Tracking wird eine mehrstufige Entzerrung der Bilder vorgeschaltet:

1. Im ersten Schritt wird eine radiale Transformation angewendet, welche auf einer linearen Verteilung der Pixel beruht. Wie in Abbildung 3.21b gezeigt, geschieht dies, indem

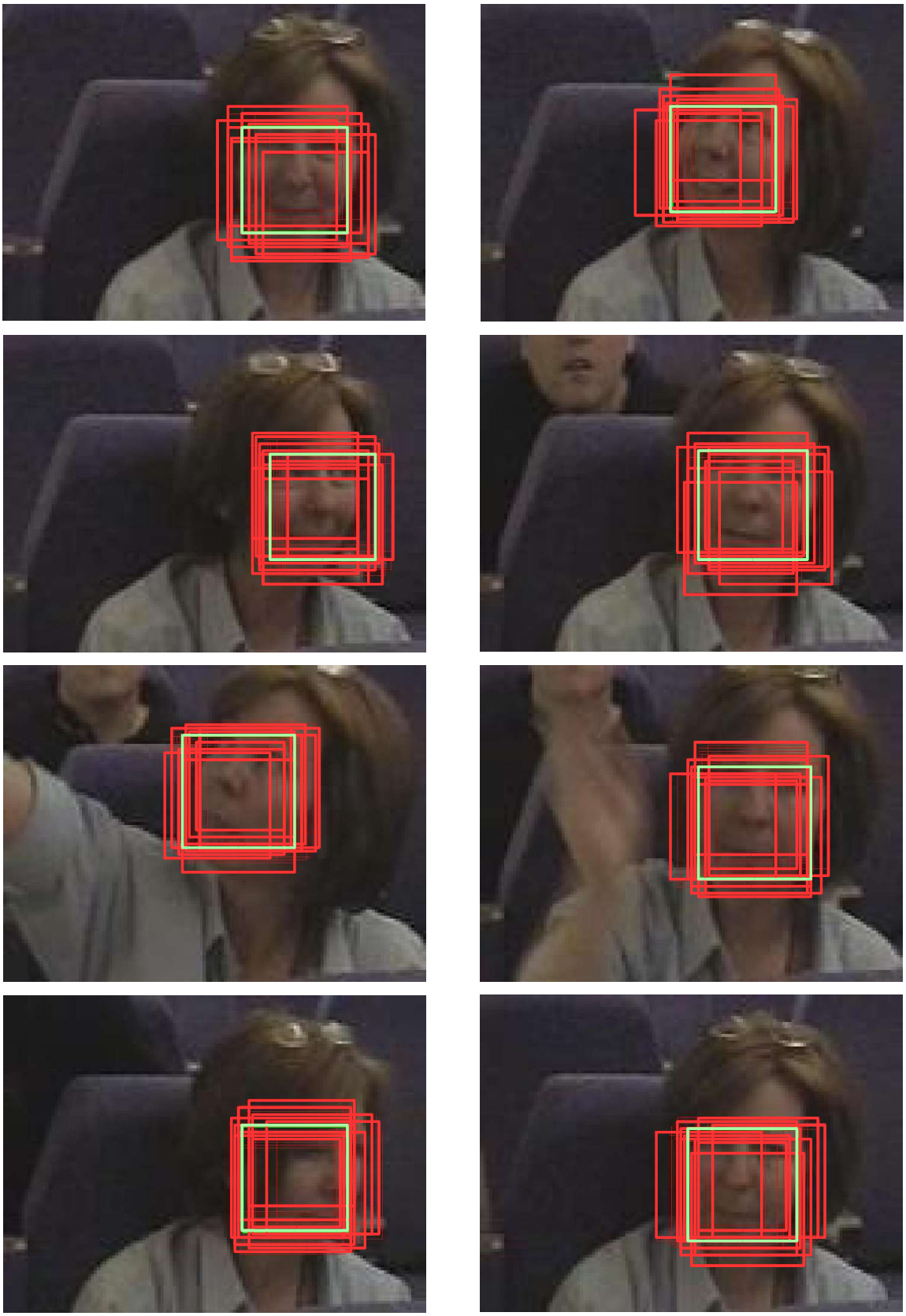
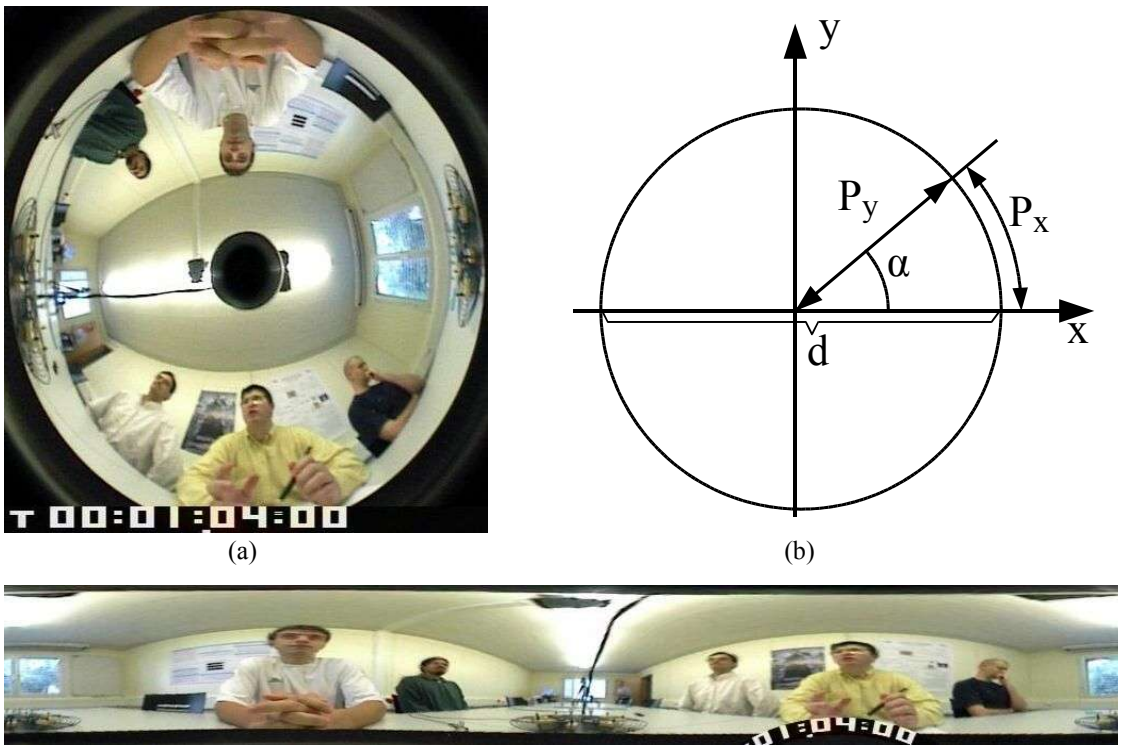


Abbildung 3.20: Beispielbilder einer Trackingsequenz in einem Überwachungsszenario

der Reihe nach Linien vom Mittelpunkt $(\text{Center}_x, \text{Center}_y)$ des omnidirektionalen Bildes O von innen nach außen hin abtastet werden. Nach diesem Abrollen entstehen zunächst Panoramabilder P nach Abbildung 3.21c. Die Koordinaten innerhalb dieses



(a) (b)

(c)

Abbildung 3.21: Transformation des omnidirektionalen Originalbilds (a) mit streifenweiser Abtastung (b) in ein Panoramabild (c)

Panoramabildes sind P_x und P_y. Bei der Transformation wird angenommen, dass die vormals räumlichen Koordinaten auf einen Zylinder mit dem Radius r projiziert wurden, welcher durch den Übergang des Bildes zum schwarzen Rand gegeben ist. Die Achse des angenommen Zylinders ist dabei identisch mit dem Mittelpunkt des Spiegels und des Kamerabilds [Svo99]. Der Reihe nach können die Spaltenvektoren des Panoramabildes aus dem omnidirektionalen abgetastet werden, wobei der Wert P_x den Winkel in O zu $\alpha = \frac{P_x}{r}$ vorgibt. Die Breite des Panoramabildes ergibt sich durch den Umfang des Zylinders $W_P = 2\pi r$. Die Pixelkorrespondenz an einem Punkt (P_x, P_y) im Ausgangsbild ergibt sich aus der Intensität des Eingangsbildes am Punkt (O_x, O_y) über:

$$O_x = (r - P_y) \cdot \cos(\alpha) + \text{Center}_x \tag{3.26}$$

$$O_y = (r - P_y) \cdot \sin(\alpha) + \text{Center}_y \tag{3.27}$$

Zur Vermeidung von zusätzlichen Artefakten und Verzerrungen durch die Umrechnung werden Methoden zur Bildverbesserung angewendet, welche auf die Mittelung benachbarter Pixel zurückgeführt werden können [Ste93]. Des Weiteren können der Randbereich und das Zentrum des Originalbildes weggelassen werden, da sie keine

inhaltliche Informationen enthalten. Außerdem wird im Folgenden nur noch der durch die räumliche Anordnung des Raumes interessante Bereich betrachtet. Das Ergebnis der Transformation ist in Abbildung 3.22 dargestellt.

Abbildung 3.22: Annäherung der Entzerrungsparameter durch Kreise

2. Nach dem ersten Schritt folgt die Korrektur der noch verbliebenen perspektivischen Verzerrungen, welche durch verschiedene Abstände der aufgenommenen Objekte zum Spiegel entstehen. Weiterführende Verbesserungsmaßnahmen zur Angleichung der Kopfgröße werden beispielsweise in [Cha05] erwähnt. Entzerrungsmaßnahmen können allerdings nur mit Kenntnis über die geometrischen Eigenschaften des Spiegels, des Raumes und die Positionen der zu detektierenden Besprechungsteilnehmer vorgenommen werden. Aus diesem Grund werden in jeweils einem Einzelbild der gewonnenen Bildausschnitte des Panoramabilds zwei repräsentative Kreise eingepasst, welche alle unbekannten Parameter der Aufnahme charakterisieren. Die Kreise werden durch die manuelle Markierung von drei Punkten gewonnen und entsprechen in erster Näherung dem Übergang zwischen Wand und Decke bzw. Tischkante, wie in Bild 3.22 ersichtlich.

   Mit Hilfe dieser Kreise können zum einen Deformationen in vertikaler Richtung sowie Verzerrungen durch die zylindrische Projektion ausgeglichen werden, was auf verschiedenen Spaltenbreiten in Abhängigkeit des Winkels zum Zylinder basiert, siehe Skizze 3.23. Die unterschiedlichen Spaltenbreiten können durch eine inverse, perspektivische

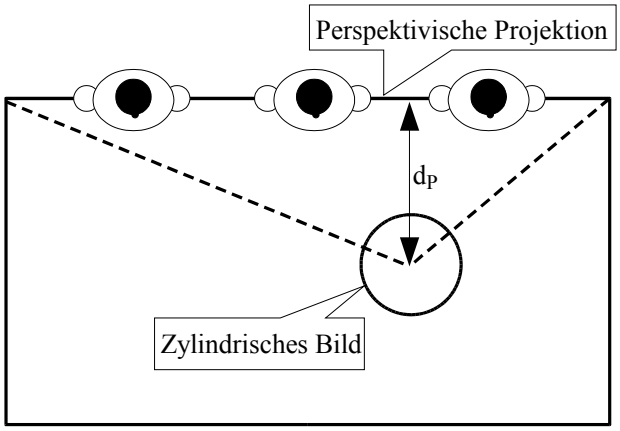


Abbildung 3.23: Zweite angewandte Transformation zur Entzerrung

Projektion nach Gleichung 3.28 ausgeglichen werden.

$$N_x = \arctan\left(\frac{x}{d_P}\right) \cdot \frac{W_N}{2\pi} \tag{3.28}$$

Dabei beschreibt N_x die Spalte im breitennormalisierten Bild und W_N die korrespondierende Breite. Der Wert x repräsentiert die Position im perspektivisch verzerrten Bild, d_P gibt den Abstand zwischen dem Mittelpunkt des Zylinders bzw. der Kamera und der Projektionsebene an.

Zur Vermeidung von Blockeffekten werden wiederholt einfache Interpolationsmethoden zur Bildrestauration verwendet [Ste93]. Abbildung 3.24 zeigt das Ergebnis nach allen Transformationen und Interpolationen zusammen mit einem aus einer anderen Perspektive aufgenommenen planaren Bild der gleichen Szene.

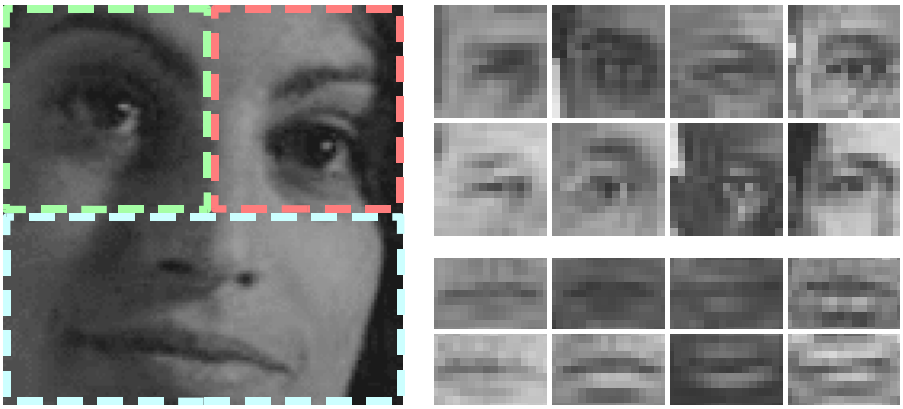
Abbildung 3.24: Linke Raumhälfte nach Entzerrung (oben), Originalbild (Mitte) und eintretende Person mit verbleibender Verzerrung im Randbereich (unten)

Nach der Verarbeitung aller Einzelbilder der vorliegenden Sequenz werden Gesichter mit Hilfe des vorgestellten Condensation Algorithmus verfolgt. Die Ergebnisse zeigen, dass trotz der verbliebenen räumlichen Verzerrung eine äußerst robuste Detektion basierend auf einer

Kombination aus Hautfarbe und MLP möglich ist [Wal04b]. Die Gesichtslokalisierung von Personen, die ihren Sitzplatz einnehmen oder verlassen, hat sich aufgrund unkompensierter Verzerrungen in den Randbereichen als problematisch erwiesen, wie in Bild 3.24 unten gezeigt.

3.6 Lokalisation der Augen und des Mundes in Gesichtern

Nachdem die Findung von Gesichtern selbst in komplexen Szenarien mit beliebigen Hintergründen über die vorgestellten Methoden prinzipiell gelöst ist, soll im Folgenden eine nachgeschaltete Lokalisierung einzelner Gesichtspartien, sogenannter *Viseme*, in Frontalbildern untersucht werden. Das Ziel ist dabei die Bestimmung der Stützpunkte von Augen und Mund für eine nachfolgende Merkmalsextraktion zur Gesichtserkennung [Bla02a], welche bereits in Abschnitt 3.3.1 vorgestellt wurde. Bei Vorlage eines durch Detektion vorsegmentierten Bildes kann die Suche nach Augen und Mund über eine Heuristik eingeschränkt werden. Abbildung 3.25a zeigt die verbleibenden Suchbereiche für die Augen und den Mund. Für die Augen ist dies jeweils das linke bzw. rechte obere Viertel. Für den Mund ergibt sich die untere Bildhälfte als möglicher Bereich. Wie vorgestellt, hat sich bei der Gesichtsdetektion



(a) (b)

Abbildung 3.25: Verbleibende Suchbereiche und Trainingsdaten für Augen und Mund

die Verwendung von MLP als besonders günstig erwiesen. Daher wird das erstellte System für die neue Aufgabenstellung mit angepassten Trainingsdaten und Netzstrukturen adaptiert. Analog zu Kapitel 3.3.5 werden hierzu zwei Netzwerke mit den Klassen *Mund* und *Auge* über die in Abbildung 3.25b dargestellten Positivbeispiele trainiert. Die Trainingsbeispiele für die Augen werden ähnlich wie die Gesichtsdaten ermittelt und können durch konzentrische Quadrate um den Augenmittelpunkt mit einer Kantenlänge des Augen-Augen-Abstandes gebildet und auf eine Größe von 20×20 Bildpunkten skaliert werden. Der Abstand der Augen bestimmt ebenfalls die Breite und das Doppelte der Höhe eines konzentrischen Rechtecks um den Mundmittelpunkt.

Das retinal verknüpfte MLP zur Augenfindung hat eine Dimension von 20×20 Eingangsneuronen, 2 Ausgangsneuronen und 26 verdeckte Schichten. Je 6 verdeckte Schichten

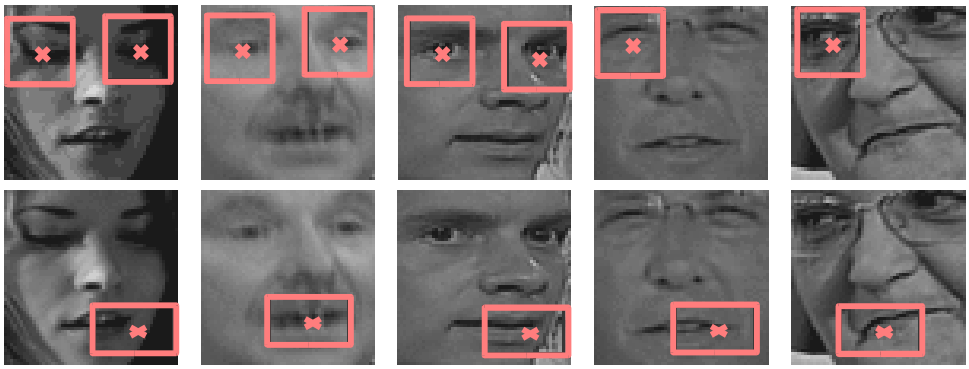


Abbildung 3.26: Detektionsergebnisse für Augen (oben) und für Münder (unten)

der Dimension 20×5 dienen der Erkennung von Augenbrauen und Wimpern, weitere 4 Schichten der Dimension 10×10 zur Erkennung der Pupille. Zur Erfassung der Augenumrisse und der Pupille mit einer gröberen Auflösung kommen weitere 16 Bereiche der Größe 5×5 zur Anwendung. Alle Schichten sind voll miteinander verknüpft.

Für die Detektion des Mundes wird ein Netz mit einer Eingangsschicht von 25×15 Neuronen eingesetzt. Auch hier wird das Netz in rezeptive Felder eingeteilt. Dies sind je 3 Schichten der Größe 25×5 zur Erkennung von Lippen und evtl. vorhandenen Zähnen und je 5 Felder der Größe 5×15 zur Erkennung von Grübchen und Mundwinkeln.

Zur Beurteilung der entstandenen Detektoren werden mit einem NN-basierten Gesichtsdetekor im ersten Schritt 50 Gesichtspositionen automatisch ermittelt. Im zweiten Schritt wird innerhalb der resultierenden Suchbereiche nach Augen respektive Mündern gesucht. Mehrfache Detektionen werden durch Mittelung zusammengefasst. Die jeweiligen Label der Viseme sind durch die Mittelpunkte der Detektionsbereiche in Abbildung 3.26 gegeben. Für eine quantitative Auswertung werden die Detektionsraten gemessen sowie die Nutzbarkeit der ermittelten Positionen beurteilt. Ein Label gilt als geeignet, wenn der Euklidische Abstand zwischen dem idealen Punkt und dem Mittelpunkt des gefundenen Bereichs weniger als 25% der Höhe beträgt. Die nach dieser Definition erhaltenen Ergebnisse sind in Tabelle 3.2 zusammengefasst. Die Detektionsraten der Augen sind demnach für diesen ersten Ansatz

Art	Detektionsrate [%]	Nutzbarkeitsrate [%]
Augen	72	72
Mund	36	18

Tabelle 3.2: Ergebnisse für die Lokalisierung von Augen und Mund

bereits vielversprechend. Nicht detektierte Augen lassen sich meist auf zu knapp bestimmte Gesichtsbereiche zurückführen. Für die Findung von Mündern scheint der vorgestellte Ansatz hingegen weniger geeignet zu sein, was vermutlich in der zu hohen Elastizität des Mundes begründet liegt. Alternativ zu einer expliziten Suche könnte der gesuchte Ankerpunkt für den Mund über ein gleichseitiges Dreieck durch die Augenmittelpunkte geschätzt werden.

Kapitel 4

Gesichtserkennung

Nachdem die Position und Größe eines Gesichts mit den im vorigen Kapitel vorgestellten Methoden als bekannt angesehen werden kann, befasst sich dieses Kapitel zunächst mit der Zuordnung eines unbekannten Bildausschnitts zu einem bekannten Modell, der sogenannten Identifikation. Anschließend werden Konfidenzmaße eingeführt, welche die von einem Klassifikator gemachten Entscheidungen in einer nachgeschalteten Verifikation überprüfen und gegebenenfalls revidieren.

Die Schwierigkeit der Erkennungsaufgabe liegt darin, dass die Variationen zwischen Bildern des gleichen Gesichts aufgrund der Änderung der Beleuchtungsverhältnisse oder des Betrachtungswinkels fast immer größer sind als die Änderung des Bildes aufgrund des Wechsels des Gesichts selbst [Mos94]. Daneben haben zudem Alterungsprozesse, die An- bzw. Abwesenheit von Brillen und Bärten, Wechsel der Mimik sowie kosmetische Veränderungen einen großen Einfluss auf die äußere Erscheinungsform einer Person. Diese Problematik ist prinzipiell die gleiche wie bei der Detektion in Kapitel 3 und wird für eine Person in Bildserie 4.1 veranschaulicht.

Abbildung 4.1: Erscheinungsformen eines Gesichts derselben Person [Phi00]

Eine der interessantesten Anwendungen für Gesichtserkennungssysteme ist die Identifizierung einer Person über die im Gesicht vorhandene Metrik, da aufgrund der berührungslosen Messung keine Interaktion mit einem System notwendig wird. Ein allgemeines Biometriesystem besteht, wie in Abbildung 4.2, aus einer Vielzahl von einzelnen Komponenten.

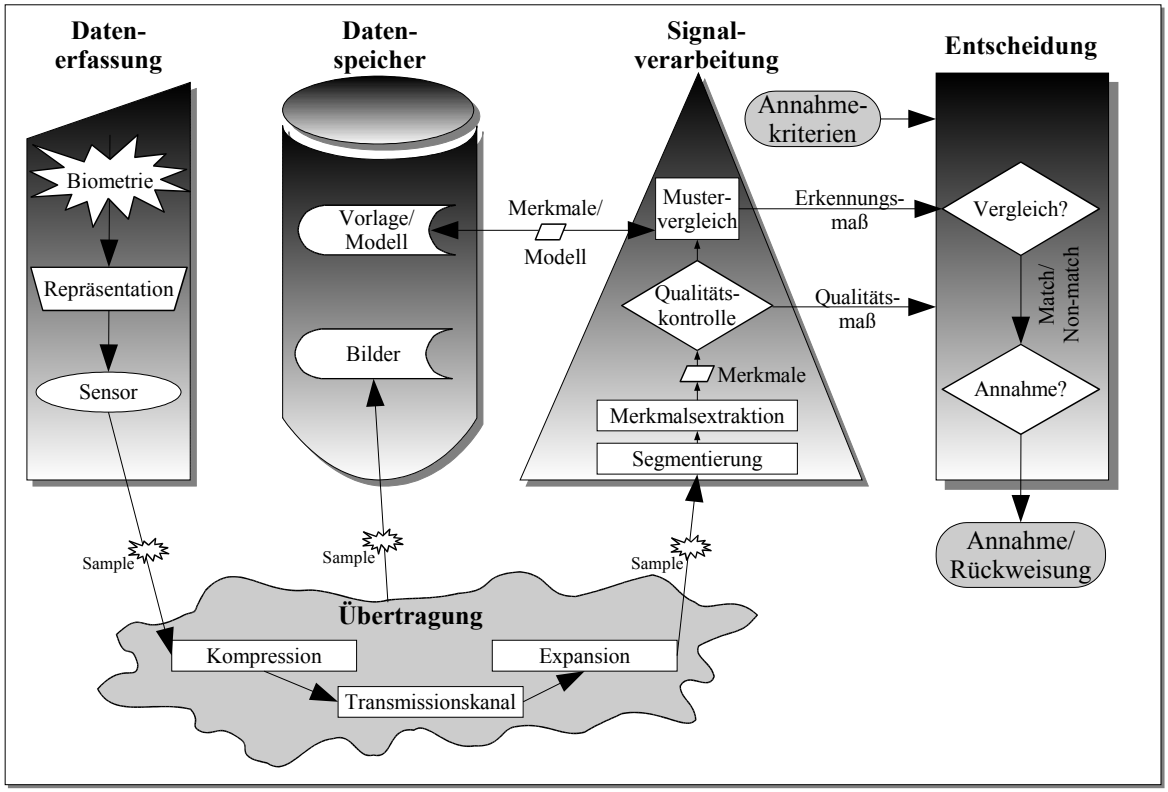


Abbildung 4.2: Diagramm eines generellen Biometriesystems [Man02]

Die wesentlichen Gruppen und Funktionen sind hierin:

- Die Datenerfassung, bei der Gesichtsbilder mit Hilfe optischer Sensoren aufgrund ihrer biometrischen Vorlagen erfasst werden. Die hohe Intraklassenvarianz der hierbei entstehenden Bilder beruht auf den erwähnten Repräsentationsmöglichkeiten.

- Die Datenübertragung, welche im Zusammenhang der vorliegenden Arbeit im Wesentlichen die Konvertierung in und aus komprimierten Bildformaten beinhaltet.

- Die Datenspeicherung zur Bereitstellung der Bilder sowie der Vorlagen und Modelle auf der Festplatte.

- Die Signalverarbeitung bestehend aus der Segmentierung, hier der Gesichtsfindung, der anschließenden Merkmalsextraktion, einer möglichen Qualitätskontrolle und dem Mustervergleich.

- Zuletzt erfolgt die Verifikation über eine abschließende Beurteilung des vorangegangenen Mustervergleiches.

In den folgenden Unterpunkten wird zunächst ein neuartiges, auf diskrete WDF erweitertes System zur Klassifikation frontal aufgenommener Einzelbilder auf Basis von P2DHMM unter Verwendung ideal aufgenommenen Bildmaterials eingeführt und quantitativ bewertet. Anschließend wird ein Gesamtsystem zur automatischen Gesichtserkennung in Bildsequenzen mit Verifikation vorgestellt. Neben der Verarbeitung vornehmlich frontal aufgenommener Bilder werden in diesem Kapitel abschließend neuartige hybride Systeme zur Profilbilderkennung präsentiert.

4.1 Klassifikation von Frontalbildern

Bei der Klassifikation von Frontalbildern wird in der Literatur prinzipiell zwischen erscheinungs- und modellbasierten Methoden unterschieden [Lu03, Zha03]. Bei der ersten Kategorie werden Bilder holistisch durch lineare sowie nicht-lineare Methoden beschrieben. Am weitesten verbreitet sind die zur Detektion bereits verwendeten Eigenfaces nach Turk und Pentland [Tur91a]. Bei der zweiten Kategorie werden Modelle über zuvor abgeleitete Merkmale gebildet. Hiernach kann weiter nach 2D und 3D Ansätzen unterschieden werden. Die drei bekanntesten zweidimensionalen Ansätze sind die sogenannten *Elastic Bunch Graphs* von Wiskott und von der Malsburg [Wis97], die *Active Appearance Models* nach Lanitis und Cootes [Lan95] und die Hidden Markov Modelle nach Samaria [Sam94]. Bei der dreidimensionalen Verarbeitung seien die *3D Morphable Models* nach Blanz und Vetter [Bla02b] als Vertreter genannt.

Im Folgenden werden unbekannte Gesichter durch zweidimensionale Vorlagen repräsentiert und mit P2DHMM klassifiziert. Da das Bildmaterial in den ersten Versuchen aus qualitativ hochwertigen Datenbanken stammt, kann die Vorverarbeitung auf die Normalisierung bzw. das Ausschneiden und Anpassen der Gesichtsbereiche beschränkt werden. Bei Datenbanken mit unterschiedlicher Bildqualität müssen zusätzlich beleuchtungsnormalisierende Maßnahmen ergriffen werden. Um die Leistung der verwendeten Klassifikatoren isoliert bewerten zu können, wird die Erkennung von der Suche nach Stützpunkten durch Vorgabe idealer Label entkoppelt. Zur Merkmalsextraktion nach dem in Grafik 4.2 vorgestellten Ablauf wird die zweidimensionale diskrete Cosinus Transformation (DCT) herangezogen.

4.1.1 Vorverarbeitung

Zur Entkopplung von Erkennung und Detektion beschränkt sich die Vorverarbeitung auf das Ausschneiden der zu erkennenden Gesichtsbereiche mit Hilfe der idealen Stützpunkte. Wie schon bei der Gesichtsdetektion in Kapitel 3.3.1 werden auch hier zur Definition der erkennungsrelevanten Gesichtsbereiche wieder die beiden Augenmittelpunkte $L'(x, y)$ und $R'(x, y)$ sowie der Mittelpunkt $M'(x, y)$ des Mundes verwendet [Ost98].

Neben der manuellen Lokalisation des Gesichtes dient dieser Schritt zudem der Normalisierung von Größe und Rotation unter Vermeidung von Verzerrungen. Obwohl sich die Modellierung mit P2DHMM prinzipiell als robust gegenüber dynamischen Bildverzerrungen erwiesen hat, sollen die Einflüsse durch Kopfneigung und Größenänderung auf die

zweidimensionale Repräsentation des eigentlich dreidimensionalen Originals minimiert werden [Mar02a]. Durch die Anwendung einer einheitlichen Vorverarbeitung auf alle Trainings- und Testbilder kann eine kanonische Repräsentation in Form standardisierter Gesichtsbereiche gewährleistet werden. In einer Arbeit von Chen et al. [Che98] konnte diesbezüglich gezeigt werden, dass bei einer reinen Gesichtserkennungsaufgabe alle irrelevanten Daten und Bereiche, wie Schultern, Hintergrund und Haare entfernt werden müssen. Obwohl eine ganzheitliche Erkennung von Gesichtern mit Umfeld möglich ist [Eic00b], sollen Haare und Hintergrund in dieser Arbeit so weit wie möglich ausgeschlossen werden. Die Normalisierung und die Einführung einer einheitlichen Pixelkorrespondenz kann durch die folgenden Schritte verwirklicht werden:

1. Normalisierung der Kopfneigung durch entgegengesetzte Rotation innerhalb der Bildebene. Dazu werden die vier Werte der Augenkoordinaten benötigt, welche nach Gleichung 4.1 den Drehwinkel γ ergeben, siehe Abbildung 4.3a. Nach der Drehung können die neuen Koordinaten $R(x, y)$, $L(x, y)$ und $M(x, y)$ berechnet werden, siehe Abbildung 4.3b.
$$\gamma = \arctan\left(\frac{R_y - L_y}{R_x - L_x}\right) \tag{4.1}$$

2. Unter Beibehaltung des Verhältnisses zwischen Höhe und Breite erfolgt nun das Ausschneiden geeigneter Bildbereiche aufgrund der transformierten Stützpunkte und den empirisch ermittelten Metriken. Die beiden charakteristischen Größen sind der Abstand der Augen d_{AA} und der vertikale Abstand d_{AM} zwischen dem Mundmittelpunkt und der Mitte einer Linie zwischen den Augen. Die ideale Breite B des Gesichtsausschnitts ist durch den doppelten Augenabstand, die ideale Höhe H durch den doppelten Abstand zwischen Augen und Mund über die Gleichungen 4.2–4.3 gegeben, siehe Abbildung 4.3c. Die Offset-Koordinaten der linken, oberen Ecke X_{off} und Y_{off} sind durch die Beziehung 4.4–4.5 gegeben.
$$B = 2 \cdot d_{AA} = 2 \cdot (R_x - L_x) \tag{4.2}$$
$$H = 2 \cdot d_{AM} = 2 \cdot (M_y - R_y) \tag{4.3}$$
$$X_{\text{off}} = L_x - \frac{d_{AA}}{2} \tag{4.4}$$
$$Y_{\text{off}} = L_y - \frac{d_{AM}}{2} \tag{4.5}$$

3. Aufgrund der später eingeführten DCT-Koeffizienten, welche Amplituden von Basisfunktionen mit festen Frequenzen repräsentieren, sollte ein Größenunterschied zwischen den Trainings- und Testbeispielen trotz der Elastizität der HMM nicht zu groß werden. Zur Vermeidung verfälschter Merkmale erfolgt somit im letzten Schritt der Vorverarbeitung entweder eine Skalierung auf eine feste Höhe H_0 wie in Abbildung 4.3d gezeigt, oder die Anpassung des endgültigen Bildausschnitts an eine vorgegebene Fläche wie in Abbildung 4.3e. Prinzipiell kann sich nach obiger Vorgabe ein beliebiges Verhältnis zwischen Breite und Höhe $S = \frac{H}{B}$ einstellen. Je nachdem ob eine feste Höhe oder eine Fläche zu generieren ist, ergibt sich zur Vermeidung von

Verzerrungseffekten eine variierende Breite B_0 bzw. ein symmetrischer Bereich um die gegebenen Stützpunkte. Aufgrund der Elastizität der verwendeten HMM verursachen die resultierenden Änderungen gegenüber dem idealen Ausschnitt jedoch keine Modellierungsprobleme.

Eine Breitenadaption kann nach Gleichung 4.6 über das Verhältnis S erreicht werden. Im hier verwendeten Beispiel wurde für die Höhe beispielsweise ein Wert von 80 Pixeln vorgegeben.

$$B_0 = H_0 \cdot \frac{B}{H} = \frac{H_0}{S} \tag{4.6}$$

Zum Vermeiden von Auslassungen muss bei der Anpassung an eine gegebene Fläche vorab unterschieden werden, welcher der beiden Parameter Höhe bzw. Breite dominierend ist. Wenn gilt $d_{AA} > S \cdot d_{AM}$, dann wird der ideale Abstand zwischen Augen und Mund angepasst zu $d_{AM} = S \cdot d_{AA}$. Bei $d_{AA} < S \cdot d_{AM}$ ergibt sich ansonsten $d_{AA} = S \cdot d_{AM}$. Die Dimensionen des ganzheitlichen Gesichtsauschnitts sind vor der Skalierung durch Verdoppelung der angepassten Distanzen gegeben. Ein typischer Wert für das Verhältnis zwischen Höhe und Breite ist $S = 1.5$ respektive eine Breite von $B_0 = 64$ und eine Höhe von $H_0 = 96$ Pixeln.

Durch die Einführung einer ovalen Maske bzw. dem Weglassen der Eckbereiche können im Bildausschnitt evtl. noch enthaltene Haar- oder Hintergrundbereiche zusätzlich ausgeblendet werden. Eine Sichtung der entstandenen Bildausschnitte ergab hierzu jedoch keinen dringenden Anlass. Ferner wurde von einer Maskierung aufgrund der damit verbundenen Merkmalsverfälschung im Randbereich Abstand genommen.

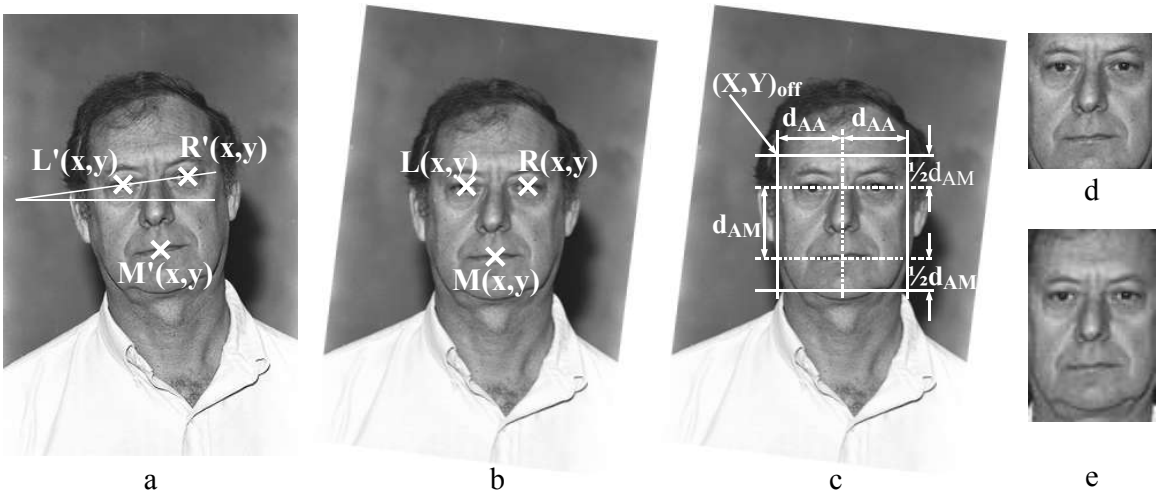


Abbildung 4.3: Kanonische Vorverarbeitung und Normalisierung von Gesichtern

4.1.2 Merkmalsextraktion

Wie im Kapitel für Grundlagen erläutert, sind die Ziele der Merkmalsextraktion die Datenreduktion und die Überführung in eine für die Modellierung parametrisch günstigere Re-

präsentationsform. Für eine optimale Funktionalität sollte darüber hinaus ein hohes Maß an Universalität, Einzigartigkeit, Beständigkeit und Erfassbarkeit der Objektmerkmale gelten.

Intensitäten als Merkmale. In ersten Ansätzen der HMM-basierten Gesichtserkennung wurden zunächst lineare HMM mit Spalten- bzw. Zeilenvektoren in Verbindung mit Grauwertintensitäten eingesetzt [Sam94]. Da ein Gesicht aufgrund seiner Anatomie in vertikaler Richtung flexibler ist, bietet sich die Modellierung mit Zeilenvektoren an.

In Analogie dazu könnten sequentiell abgetastete Grauwerte bereits als Merkmale zur Modellierung mit P2DHMM herangezogen werden. Arbeiten zu diesem Ansatz haben aber in erster Linie eine äußerst hohe Anfälligkeit gegenüber geringen Änderungen der Beleuchtung gezeigt, was den Einsatz für eine robuste Erkennung ausschließt [Sam94].

Segmentierungsbeispiele unter Verwendung von Pixelintensitäten sind in Darstellung 4.4 gezeigt, wobei im Vorfeld ein P2DHMM mit 5×5 Zuständen trainiert wurde. Zur Segmentierung wird der wahrscheinlichste Pfad durch die quasi-zweidimensionalen Verteilungen bestimmt. Aus Gründen der Übersichtlichkeit werden die emittierenden Zustände sowie die korrespondierenden Pixel durch verschiedene Helligkeiten gekennzeichnet. Die schraffierten Zustände entsprechen den Markierungszuständen für den Spaltenanfang. Aus der Illustration geht der Verlust des dynamischen Warpings in horizontaler Richtung deutlich hervor. Durch höherwertigere Merkmale, wie blockbasierte DCT-Koeffizienten mit Überlappung, kann diese Beschränkung für praktische Anwendungen bereits abgeschwächt werden.

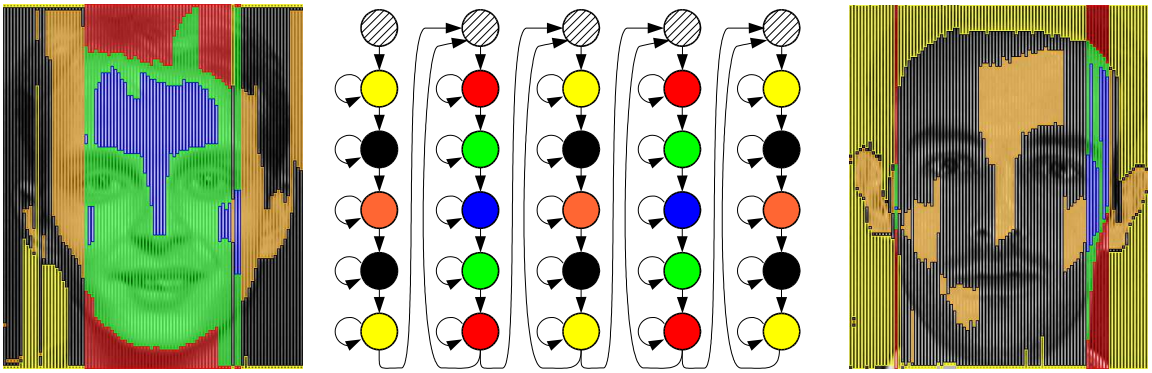


Abbildung 4.4: Gesichtssegmentierungen über Intensitäten (links und rechts) mit korrespondierendem P2DHMM (Mitte)

Diskrete Cosinus Transformation. Mit dem Ziel einer robusten, höherwertigen Repräsentation der Bildinhalte wird eine blockbasierte Merkmalsextraktion eingeführt. Ein weiterer Aspekt der verwendeten zweidimensionalen DCT ist zudem die Möglichkeit einer verlustbehafteten Datenreduktion und der damit verbundenen Möglichkeit einer kompakteren Modellierung. In mehreren Arbeiten im Zusammenhang mit HMM wie Nefian [Nef99] und Eickeler [Eic98a] hat sich der Einsatz der zweidimensionalen DCT gegenüber anderen Ansätzen als überlegen erwiesen.

Bei der Durchführung der DCT-Transformation werden Gewichtungskoeffizienten orthogonaler, zweidimensionaler Cosinus-Basisfunktionen berechnet. Die sich ergebenden Koef-

fizienten können durch die orthogonalen Basisfunktionen als unkorreliert angesehen werden, was neben einem hohen Informationsgehalt der einzelnen Koeffizienten darüber hinaus für die Modellbildung mit HMM vorteilhaft ist. Hierdurch besetzen die Kovarianzmatrizen kontinuierlicher HMM nur Werte auf der Hauptdiagonalen.

Durch die Verwendung dieser Transformation ergibt sich ein zusätzlicher praktischer Vorteil aufgrund der weiten Verbreitung des Bildkompressionsverfahrens nach dem JPEG-Standard[1], welcher fundamental auf der DCT beruht. In einer Arbeit von Eickeler konnte gezeigt werden, dass eine Gesichtserkennung ohne zusätzliche Konvertierung oder Umrechnung direkt mit bereits gespeicherten JPEG-Bilddaten möglich ist [Eic00b].

Analog zum ersten Schritt der JPEG-Kompression werden bei der Merkmalsextraktion der Reihe nach einzelne Blöcke mit Grauwertintensitäten durch Koeffizienten harmonischer Schwingungen unterschiedlicher Frequenzen repräsentiert. Obwohl theoretisch beliebige Blockgrößen möglich sind, haben Werte von $N = 8$ und $N = 16$ die größte Verbreitung. Die Amplituden $C(u, v)$ der Basisfunktionen werden in Form einer Matrix C mit den Indizes $u \in 0; N-1$ und $v \in 0; N-1$ arrangiert [Gon87].

$$C(u,v) = c(u,v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \cdot \cos \frac{(2x+1)u\pi}{2N} \cos \frac{(2y+1)v\pi}{2N} \tag{4.7}$$

In obiger Gleichung nehmen die Vorfaktoren $c(u, v)$ die folgenden Wertigkeiten an:

$$c(u,v) = \begin{cases} \frac{1}{N} & \text{für } u, v = 0 \\ \frac{2}{N} & \text{sonst} \end{cases} \tag{4.8}$$

Nach der Umformung symbolisiert der obere, linke Koeffizient $C(0, 0)$ den Gleichanteil des untersuchten Blocks. Zur Visualisierung ist in Abbildung 4.5 ein Standardtestbild (a) zusammen mit seiner Rekonstruktion (c) gezeigt. Der Teil (c) zeigt die Basisschwingungen, Teil (d) die Rekonstruktionen mit nur jeweils einem Koeffizienten. Wie aus dem abgedruckten Beispiel ersichtlich, haben die Koeffizienten der höheren Frequenzen mit höheren Indizes einen zunehmend kleineren Informationsgehalt. Mit dem Ziel einer Datenreduktion werden daher nur die wesentlichen informationstragenden Koeffizienten in den Merkmalvektor aufgenommen, deren Indizes der Beziehung $u + v \leq D$ gehorchen. Die Konstante D gibt die Nebendiagonale an, bis zu der die Koeffizienten übernommen werden. Im gezeigten Beispiel gilt $D = 3$. Die verbleibenden Werte werden somit in einem blockbeschreibenden 10-dimensionalen Vektor formiert. Aufgrund der Frequenzabhängigkeit der Koeffizienten würde mit einer signifikanten Bildgrößenänderung eine irreversible Verfälschung der Merkmale einhergehen. Im Falle der Verdopplung würden sich aufgrund der Frequenzhalbierung beispielsweise alle Indizes exakt um eine Position verschieben.

Zur Generierung quasi stationärer Merkmalssequenzen wird es nötig, aufeinander folgende Abtastfenster nicht um die komplette Blockgröße zu verschieben sondern lediglich um einen Bruchteil, wodurch überlappende Bereiche entstehen. In den Anwendungen kommen Überlagerungen von 25%, 50% und 75% zum Einsatz. Die Abtastung erfolgt dabei aufgrund

[1] Gebräuchliche Bezeichnung für das Dateiformat der Joint Photographic Expert Group (JPEG)

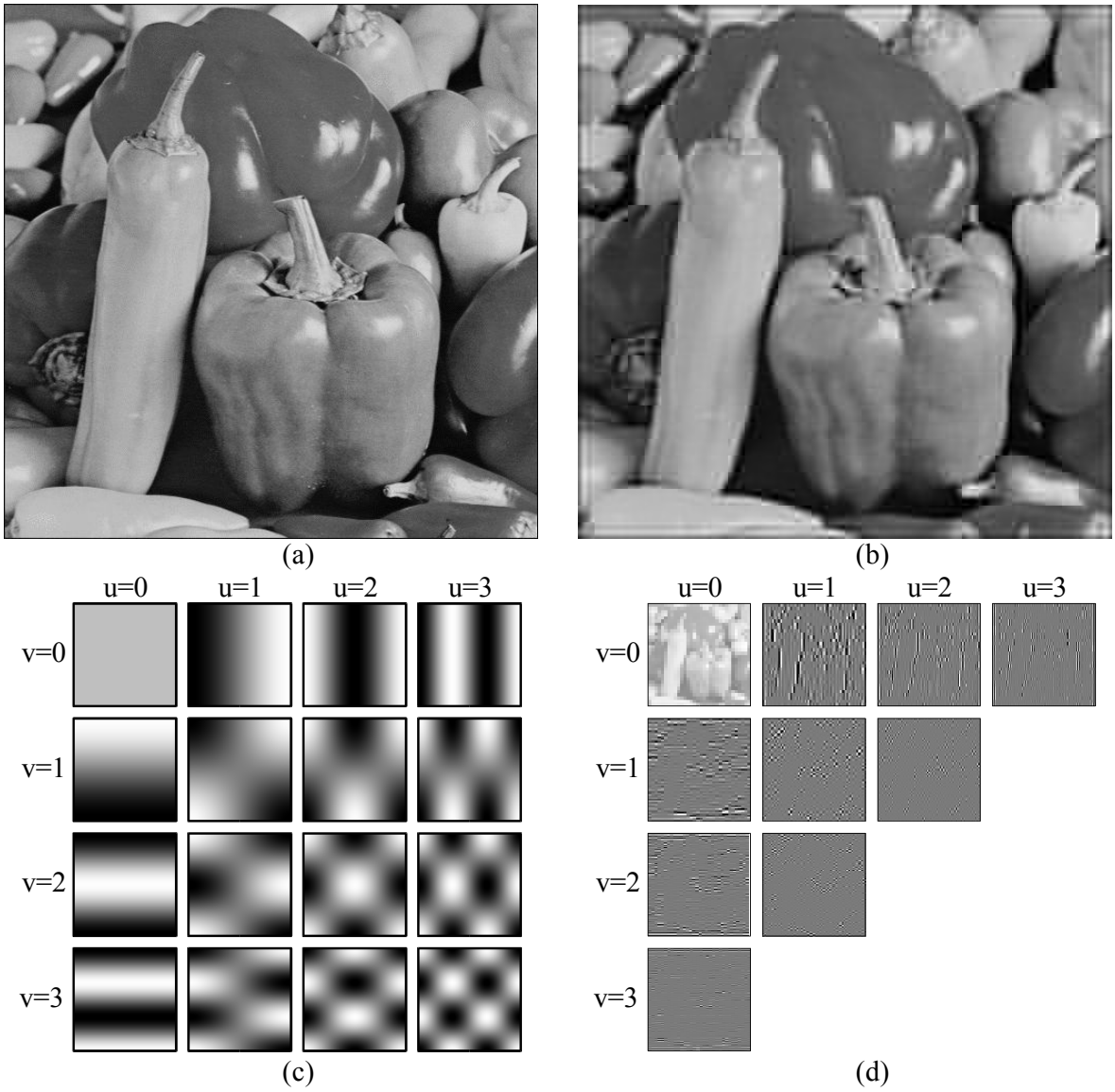


Abbildung 4.5: Originalbild (a) mit Rekonstruktion (b) sowie Basisfunktionen (c) und gewichteten Beiträgen (d)

der Anatomie menschlicher Gesichter zunächst von oben nach unten, dann von links nach rechts. Um eine Modellierung mit P2DHMM zu ermöglichen, werden vor dem ersten Block einer Spalte zu den Markerzuständen korrespondierende Observationen eingebracht. Damit alle Pfade durch den Zustandsraum bei denen die Marker nicht auf die Markerstates treffen eine Wahrscheinlichkeit von Null einnehmen, erhalten diese Zustände Wertigkeiten außerhalb des Bereiches real vorkommender DCT-Koeffizienten. In den später vorgestellten Versuchen erhält ein Markierungsvektor damit besonders große Zahlenwerte für alle Elemente des Vektors, hier 1000.

Der typische Verlauf einer sich ergebenden Segmentierung durch Verwendung von P2DHMM und DCT-Merkmalen unter Auslassung der Markerstates ist durch die Beispiele in Bild 4.6 gegeben. Analog zur Segmentierung mit Grauwerten wird auch hier ein trai-

niertes Modell mit 5×5 Zuständen verwendet. Zur besseren Visualisierung wurde auf eine Überlappung verzichtet. Die Größe der Abtastfenster beträgt 8×8 Pixel.

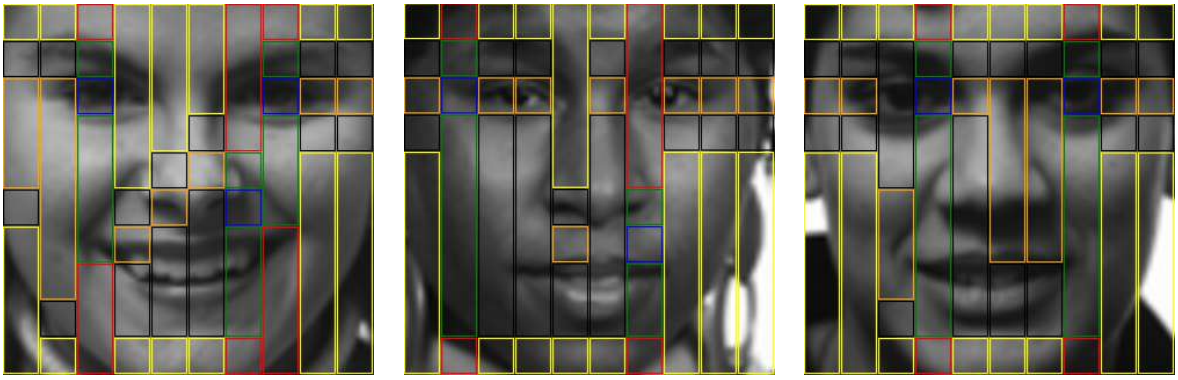


Abbildung 4.6: Segmentierungsbeispiele für die Modellierung mit DCT-Blöcken

4.1.3 P2DHMM-basierte Frontalbilderkennung

Zur Modellierung der sich ergebenden Merkmalssequenzen werden die beiden in den Kapiteln 2.7.2 und 2.7.3 beschriebenen Produktionswahrscheinlichkeiten verwendet: kontinuierlich und diskret. Der Trainingsablauf lässt sich dabei unabhängig von der verwendeten WDF in größtenteils ähnliche Schritte zusammenfassen, wie im Flussdiagramm 4.7 zu sehen ist.

Unter gemeinsamer Nutzung aller Bilder mit idealen Stützpunkten für Augen und Mund wird vor der eigentlichen Modellbildung die beschriebene Vorverarbeitung und Merkmalsextraktion angewendet. Um resultierenden, unterschiedlichen Amplituden der DCT-Koeffizienten $C(u, v)$ entgegenzuwirken, werden alle Observationen mit Hilfe einer, auf Basis der Trainingsdaten ermittelten, globalen Statistik normiert. Hierzu werden die Mittelwerte elementweise abgezogen und separat durch die entsprechende Varianz geteilt.

Danach muss unterschieden werden, ob es sich um ein kontinuierliches oder diskretes System handelt. Bei einem Kontinuierlichen sind zur Modellierung keine weiteren Maßnahmen notwendig, bei Verwendung einer diskreten WDF muss während des Trainings zunächst ein passendes Codebuch $\mathcal{M}$ für die Vektorquantisierung mit allen zur Verfügung stehenden Trainingsdaten ermittelt werden. Hierzu wird von einem neuronalen, die Transinformation optimierenden Verfahren, Gebrauch gemacht [Neu01]. Mit dem so entstandenen Codebuch werden die kontinuierlichen Vektoren in diskrete Label überführt. Zur Gewährleistung der zweidimensionalen Struktur werden unabhängig von der Art der WDF, Marker an den entsprechenden Stellen in die Merkmalsketten eingebracht. Unter Vorgabe eines prototypischen Modells mit gleichverteilten Zustandsübergangs- und Ausgabewahrscheinlichkeiten wird in der Trainingsphase vorerst ein globales Gesichtsmodell $\bar{\lambda}$ geschätzt. Danach werden die individuellen Personenmodelle λ_n mit den entsprechenden Trainingsbeispielen nachtrainiert.

In der Erkennungsphase werden die Produktionswahrscheinlichkeiten aller Modelle für die zu beschreibende Sequenz ermittelt. Das unbekannte Bild wird der Klasse mit der höchsten Produktionswahrscheinlichkeit respektive Ähnlichkeit zugeordnet.

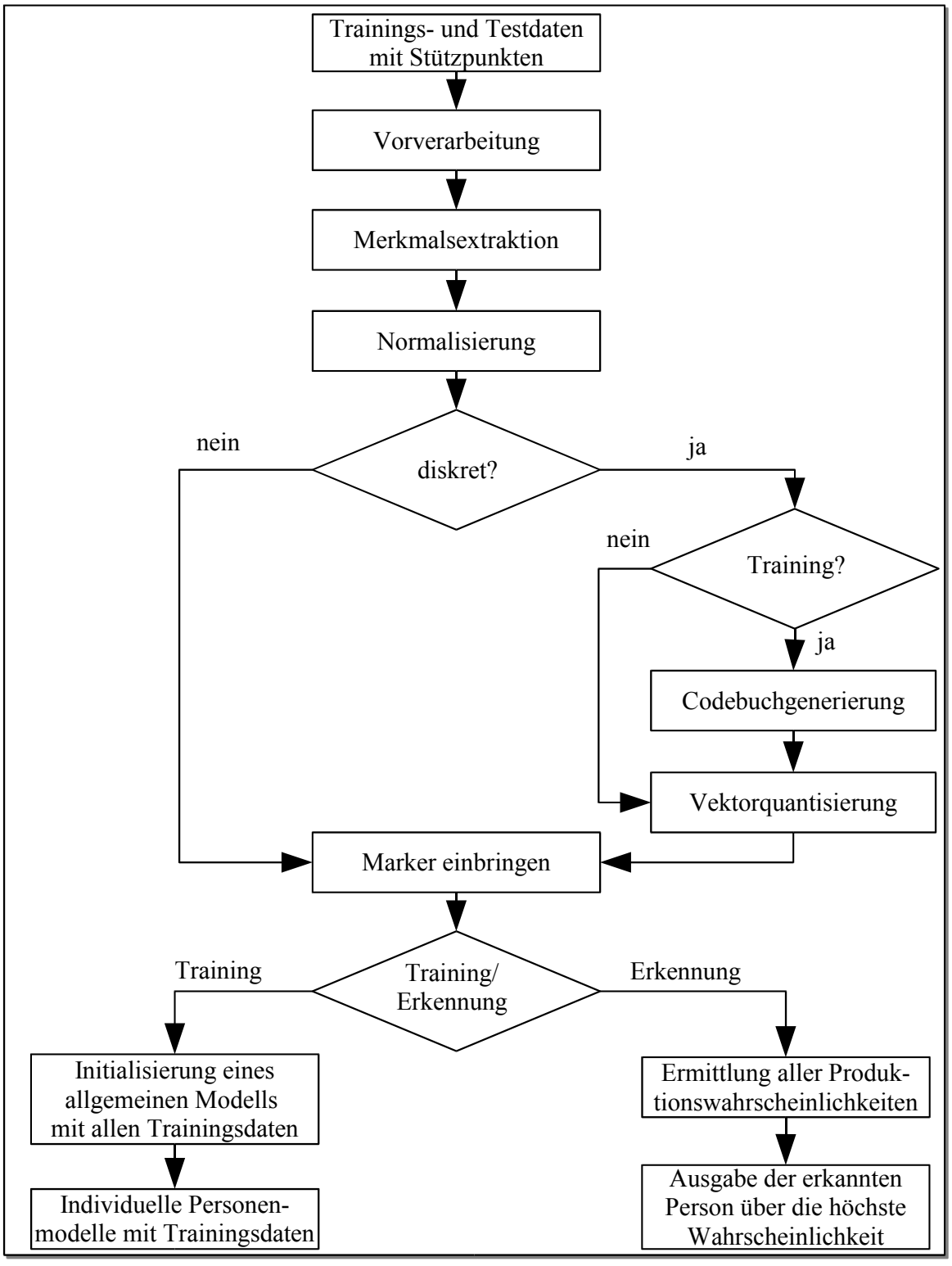


Abbildung 4.7: Flussdiagramm des Trainings- und Erkennungsablaufs

4.1.4 Experimente und Ergebnisse

Die gesamte Leistungsfähigkeit des vorgestellten Systems bestehend aus Vorverarbeitung, Merkmalsextraktion und Modellierung bzw. Klassifikation wird im Folgenden unter Zuhil-

fenahme dreier verschiedener Datenbasen mit unterschiedlichen Parametersätzen und Gesichtsausschnitten näher untersucht. Soweit nicht anders angegeben, werden bei der Mermalsextraktion 10 Koeffizienten und eine Blocküberlappung von 75% verwendet.

Da aufgrund des verwendeten Klassifikators systembedingt immer das ähnlichste Modell gefunden wird, bleibt zunächst nur die reine Identifikations- bzw. Retrievalleistung zu bewerten. In den vorliegenden Szenarien werden dementsprechend nur Bilder von Personen erkannt, welche in der Gruppe der bekannten Personen sind. Es kommen also zunächst keine Eindringlinge vor. Ferner steht der unmittelbare Vergleich von diskreter und kontinuierlicher Modellierung im Mittelpunkt der ersten Untersuchungen.

AT&T Datenbank. Die erste der verwendeten Datenbanken umfasst eine Personenmenge von 40 Individuen mit je 10 Aufnahmen und wurde ursprünglich von der Firma Olivetti[2] erstellt, neuerdings wird sie unter dem Namen AT&T geführt [Sam94]. Zum Training werden jeweils die ersten fünf Bilder einer Serie verwendet. Getestet wird anschließend mit den fünf verbleibenden Bildern der Person. Die Aufnahmen umfassen verschiedene emotionale Ausdrücke, Blickrichtungen sowie moderate Drehungen des Kopfes. Während der Aufnahmen herrschen gute Bedingungen der Beleuchtung, welche für alle Bilder konstant gehalten wurden. Die möglichen Erscheinungsformen einer Person sind in der Bildserie 4.8 vergegenwärtigt. Es werden beide in Kapitel 4.1.1 vorgestellten Verfahren zur Extraktion der Gesichtspartien verwendet.

Wie in der Literatur konnte unabhängig von den Ausschnitten sowohl mit kontinuierlicher als auch mit diskreter WDF die perfekte Erkennungsrate, also 100%, gemessen werden [Eic00b]. Daher wurde die Anzahl der Trainingsbeispiele in weiteren Versuchen bis auf ein Exemplar reduziert, wodurch immer noch keine Fehler in der Modellierung beobachtet wurden.

Abbildung 4.8: Exemplarische Erscheinungsformen in der AT&T Datenbank [Sam94]

[2] Auch Olivetti Research Laboratories (ORL) Datenbank genannt

FERET Datenbank. Da unter Verwendung der obigen vergleichsweise kleinen Datenbank aufgrund der idealen Erkennungsrate keine Verbesserungen durch eine diskrete Modellierung erkennbar wurden, wird ein weitaus größerer Testkorpus für die weiteren Untersuchungen verwendet. Zu diesem Zweck wird eine Teilmenge aus der frei verfügbaren FERET[3] Datenbank [Phi00] extrahiert.

Die FERET Datenbank enthält verschiedene Bildserien zur qualitativen Beurteilung von Systeminvarianzen gegenüber Beleuchtungseinflüssen, Gesichtsausdrücken und Alterungserscheinungen. Zur Untersuchung des Modellierungsverhaltens des vorgestellten Systems wird auf eine Menge von Frontalaufnahmen mit unterschiedlichen Gesichtsausdrücken zurückgegriffen, welche kurz hintereinander und unter guten Beleuchtungsverhältnissen aufgenommen wurden. Bis auf die Mimik und die Wahl der Bildausschnitte entspricht die Bildqualität den hohen Anforderungen der Bundesdruckerei für digitale Bilddaten in Ausweisen [Bun05]. Bei diesem Szenario mit der Bezeichnung *FB gegen FA* existieren pro Person exakt zwei Aufnahmen, so dass während des Trainings nur ein einziges Beispiel verfügbar ist.

Da für eine robuste Schätzung der stochastischen Modellparameter jedoch mehrere Beispiele unabdingbar sind, wird die Anzahl an Trainingsexemplaren durch stochastische Variation der idealen Ausschnitte künstlich erhöht ähnlich wie in Kapitel 3.3.1. Die Variationsparameter sind wie bei der Detektion auch hier die Koordinaten für den Offset, die Neigung sowie die Höhe und Breite. Eine Anzahl von insgesamt fünf Trainingsbeispielen hat sich in der Praxis als ausreichend erwiesen. Der verwendete Testkorpus umfasst zunächst 321 verschiedene Bildpaare, von denen exemplarisch fünf Beispiele in der oberen Reihe von Abbildung 4.9 abgebildet sind. In der Mitte der Abbildung befinden sich die korrespondierenden Testbilder zur Anfrage beim System.

Abbildung 4.9: Typische Bildpaare einer Person (obere und mittlere Zeile) sowie fälschlich zugeordnete Personen aus der FERET Datenbank (unten) [Phi00]

[3] Abkürzung für Face Recognition Technology

In den Versuchsreihen wird neben den grundsätzlichen Modellierungsparadigmen *diskret* und *kontinuierlich* auch zwischen verschiedenen Anzahlen an Zuständen der P2DHMM von 3×3 bis 7×7 sowie unterschiedlichen Blockgrößen $N = 16$ und $N = 8$ unterschieden.

Bei der Vorverarbeitung werden zunächst Gesichtsbereiche mit 64×96 Bildpunkten generiert. Für den Vergleich wird jeweils die Anzahl an richtig klassifizierten Testbildern gemessen. Über die sogenannte 10-Besten Liste, welche die Häufigkeit angibt, dass sich die richtige Person unter den 10 wahrscheinlichsten Personen befindet, kann darüber hinaus ein Trend der Modellierungseigenschaften abgelesen werden. Bei der Modellierung mit diskreten Modellen hat sich ein Codebuch mit $\mathcal{M} = 1.000$ Einträgen als ausreichend groß erwiesen. Eine Vergrößerung brachte keine zusätzlichen Verbesserungen. Dementsprechend konnte bei kontinuierlichen Modellen über mehrere Normalverteilungen vermutlich aufgrund der Limitation an Trainingsdaten ebenfalls keine Leistungssteigerung beobachtet werden. Die Ergebnisse der Simulationen sind in den Diagrammen 4.10a – 4.10d gegenübergestellt [Wal01a]. Die Reklassifikationsrate liegt bei allen Experimenten bei 100%, die beste gemessene Iden-

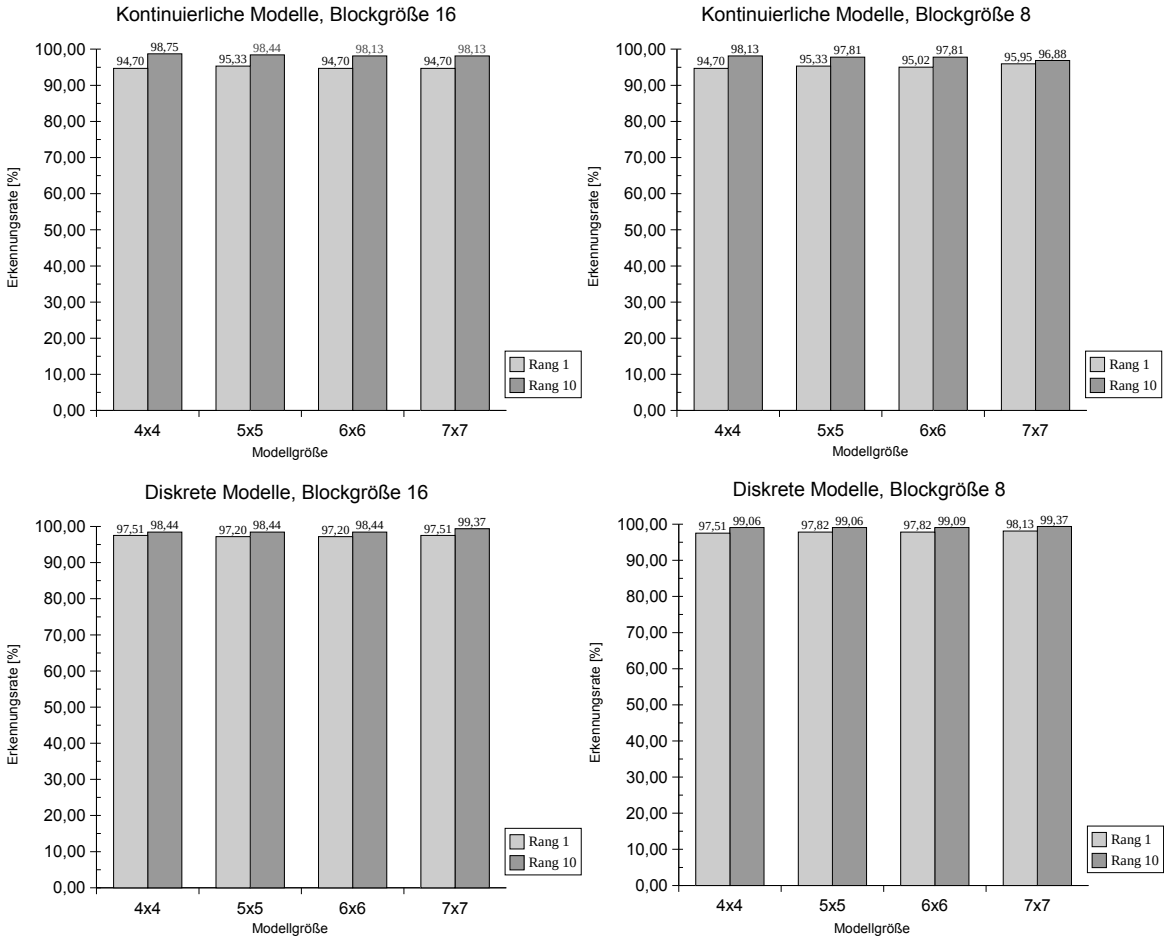


Abbildung 4.10: Erkennungsergebnisse bei Blockgrößen $N = 16$ und $N = 8$ bei kontinuierlichen sowie diskreten Modellen

tifikationsrate bei $98,13\%$. Ein Vergleich der Modellierungsverfahren indiziert eine Überlegenheit der Modelle mit diskreter WDF. Mit geringer werdender Größe des Abtastfensters

und höherer Anzahl an verwendeten Modellzuständen ist darüber hinaus eine steigende Zuordnungsrate bei gleichzeitig steigender Berechnungsdauer messbar.

Zur weiteren Beurteilung der Qualität des Gesamtsystems wird die Anzahl der unterschiedlichen Personen auf das ganze zur Verfügung stehende Material in Anlehnung an die Vorgaben des FRVT[4] 2000 Protokolls ausgedehnt [Bla05]. Insgesamt umfasst der erweiterte Korpus die Anzahl von 1.196 Trainingsbildern und 1.195 Testbildern. Die Simulationsergebnisse zur Messung der Identifizierungsleistung sind unter Angabe der relevanten Parameter in Tabelle 4.1 zusammengefasst.

P2DHMM Zustände	Überlappung	Blockgröße	Koeffizienten	Codebuch	Vorverarbeitung	Erkennungsrate [%]
3×3	6	8	10	500	Fläche 80×80	84,77
4×4	6	8	10	500	Fläche 80×80	85,77
5×5	6	8	10	500	Fläche 80×80	86,86
5×5	8	16	10	500	Fläche 64×96	80,25
5×5	12	16	10	1.000	Fläche 64×96	88,82
7×7	12	16	10	1.000	Fläche 64×96	92,80
7×7	6	8	10	2.000	Fläche 64×96	93,64
8×8	6	8	10	1.500	Fläche 64×96	93,64
8×8	6	8	10	2.000	Fläche 64×96	94,56
8×8	6	8	15	2.000	Fläche 64×96	98,58
10×10	6	8	21	2.000	Fläche 64×96	93,97

Tabelle 4.1: Identifikationsraten *FA vs. FB* mit verschiedenen Parametern

Aufgrund der Überlegenheit diskreter Modelle wurde im Weiteren auf den Vergleich mit kontinuierlichen verzichtet. Die zuvor beobachteten Trends können auch bei der zweiten Testreihe bestätigt werden: die Leistung von Modellen mit steigender Komplexität sind in der Regel bis zu einer gewissen Sättigung besser, kleinere Abtastfenster sind überlegen und eine Überlappung von 75% hat sich zur Generierung quasi stationärer Beobachtungssequenzen bewährt. Durch die Verwendung größerer Merkmalsvektoren, z.B. 15 Koeffizienten, kann eine weitere Leistungssteigerung beobachtet werden. Die Größe des Codebuchs ist stark von der Wahl der anderen Parameter abhängig. Insgesamt hat sich eine Größenordnung von 2.000 Einträgen in der Regel als ausreichend erwiesen. Die Wahl des Gesichtsausschnitts scheint aufgrund der Elastizität des Klassifikators keine herausragende Rolle einzunehmen.

Die beste Parameterkonstellation ergab eine Identifikationsrate von $98,58\%$[5], was in der Nähe des besten in der Literatur erwähnten kommerziellen Systems liegt [Phi00]. Eine abschließende Betrachtung der falsch zugeordneten Bilder nach Abbildung 4.9 vergegenwärtigt

[4] Abkürzung für Facial Recognition Vendor Test

[5] Zur robusteren Schätzung wurden bei diesem Experiment sogar 10 statt 5 künstlicher Trainingsbeispiele verwendet.

die Schwierigkeit der Gesichtserkennungsaufgabe mit nur einem Trainingsbeispiel. In der unteren Reihe befinden sich die den eigentlichen Bildpaaren fälschlicherweise zugeordneten Personen.

AR-Datenbank. In einer folgenden Versuchsreihe soll die Leistungsfähigkeit des diskreten Gesichtserkenners bezüglich realer Variationen der erscheinungsbasierten Klasse Gesicht eruiert werden. Hierzu wird eine ebenfalls frei verfügbare Datenbank mit 117 verschiedenen Personen verwendet, welche von **A**leix Martinez und **R**obert Benavente erstellt wurde, kurz die AR-Datenbank [Mar98]. Von jeder darin enthaltenen Person werden in einer Serie insgesamt 13 frontal aufgenommene Bilder bereitgestellt. Jede Serie beinhaltet dabei normal beleuchtete Aufnahmen mit 4 verschiedenen Gesichtsausdrücken, sowie starke Belichtungseinflüsse und Verdeckungen durch Sonnenbrillen bzw. Schals. Darüber hinaus werden in einer zweiten Serie mit einem zeitlichen Abstand von vierzehn Tagen alle Aufnahmen der Personen wiederholt. Abbildung 4.11 zeigt unter Angabe der Bildindizes eine beispielhafte Bildserie einer Person ohne Verdeckungen.

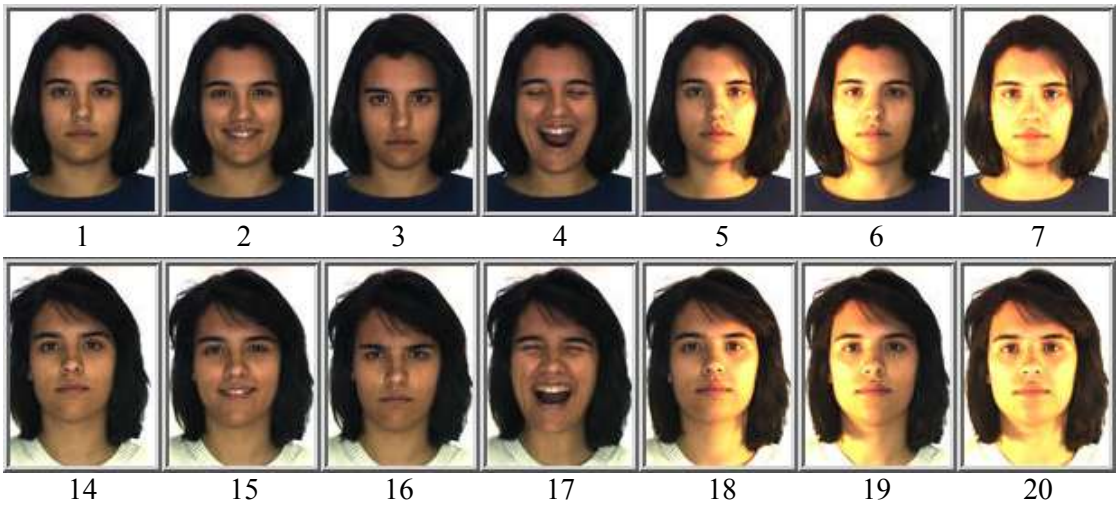


Abbildung 4.11: Beispielhafte Bildserie aus der AR-Datenbank [Mar98]

Neben dem Systemverhalten bei einer höheren Anzahl echten Trainingsmaterials mit zeitlichem Abstand zwischen den Aufnahmen wird zudem der Einfluss einer variierenden Belichtung in den Versuchen untersucht. Für alle Experimente gelten die gleichen Modellierungsparameter mit 7×7 Zuständen, einer Blockgröße von $N = 8$, einer Überlappung von 6 Pixeln und einem Codebuch von $\mathcal{M} = 2.000$. Die Unterschiede der Simulationen ergeben sich in der Konstellation der verwendeten Trainings- und Testdaten, welche sich aus den in Tabelle 4.2 genannten Exemplaren zusammensetzen.
Neben den verschiedenen Konstellationen der Trainings- und Testdaten wird zur Kompensation der Beleuchtungseinflüsse zusätzlich eine gradientenbasierte Helligkeitsnormierung nach Kapitel 3.3.2 vor die Merkmalsextraktion geschaltet. Mit Hilfe des im Anhang B.6 vorgestellten Retinex-Algorithmus wird zudem versucht, lokale Kontrastprobleme auszugleichen [Fun04], wie in Abbildung 4.12 gezeigt.

Bildindex	Beleuchtung	Mimik
1,14	normal	neutral
2,15	normal	lachend
3,16	normal	verärgert
4,17	normal	schreiend
5,18	Schlaglicht von links	neutral
6,19	Schlaglicht von rechts	neutral
7,20	beidseitiges Schlaglicht	neutral

Tabelle 4.2: Identifikationsraten der AR-Datenbank mit verschiedenen Bildkonstellationen

Abbildung 4.12: Unbearbeitete (oben) und optimierte (unten) Bilddaten [Mar98]

Mit dem Ziel helligkeitsinvarianter Merkmale werden darüber hinaus die DCT-Koeffizienten für den Gleichanteil eines Blockes in den Vektoren weggelassen, so dass sich in den Versuchen eine Dimension von 14 statt 15 Koeffizienten ergibt [San03, San04]. Ein Auszug der relevanten Versuche und Ergebnisse ist in Tabelle 4.3 aufgeführt.

Aus den Ergebnissen wird ersichtlich, dass der DC-Koeffizient C_{00} bei Bildern mit unterschiedlicher Belichtung einen störenden Einfluss auf die richtige Zuordnung hat. Bei nur einem Trainingsbeispiel mit Bildausschnitt über die volle Gesichtsbreite kann die Identifikationsrate durch Weglassen des Gleichanteils auf über 99% gesteigert werden. Wie schon bei den Ergebnissen der FERET-Datenbank kristallisiert sich auch bei diesen Tests eine von der Mimik nahezu unabhängige Erkennungsleistung heraus. Bei Erkennungsaufgaben mit nicht frontal beleuchteten Gesichtern kann die Identifikationsrate durch eine helligkeitsnormierende Vorverarbeitung ebenfalls gesteigert werden, wie ein Vergleich der entsprechenden Versuche zeigt. Selbst unter schwierigeren Belichtungsverhältnissen kann unter Verwendung einer beleuchtungsnormalisierenden Vorverarbeitung eine nahezu perfekte Erkennungsleistung erreicht werden.

Trainings-index	Test-index	Erkennungs-rate [%]	DC-Koeffizient C_{00}	Vorverarbeitung
1	14	91,51	ja	-
1	14	99,15	-	-
1-4	14-17	97,65	-	-
1-4	14-20	86,94	-	-
1-4	14-17	97,22	-	Gradientenausgleich+Retinex
1-4	14-20	86,45	-	Gradientenausgleich+Retinex
1-7	14-17	97,22	-	-
1-7	14-20	96,95	-	-
1-7	14-20	99,15	-	Gradientenausgleich+Retinex

Tabelle 4.3: Identifikationsraten mit verschiedenen Parameterkonstellationen und Erkennungsszenarien für die AR-Datenbank

4.1.5 Mehrheitsentscheidungen

Mit dem oben vorgestellten Trainings- und Testkorpus der FERET Datenbank mit 321 Personen wird in einer weiteren Versuchsreihe über eine nachgeschaltete Mehrheitsentscheidung versucht, eine Steigerung der Erkennungsleistung auf Entscheidungsebene herbeizuführen. In Analogie zur Visemdetektion werden dazu neben den ganzheitlichen Gesichtserkennern Systeme für einzelne Bildbereiche eingeführt, wie dem rechten und dem linken Auge sowie der Nase und dem Mund. Ähnlich zu den oben vorgestellten Empiriken ergeben sich die personenspezifischen Bildbereiche auch hier aus den Abständen der Augen, der Nase und des Mundes, welche wiederum durch ideale Stützpunkte gegeben sind. Typische Ausschnitte sind in Abbildung 4.13 dargestellt.

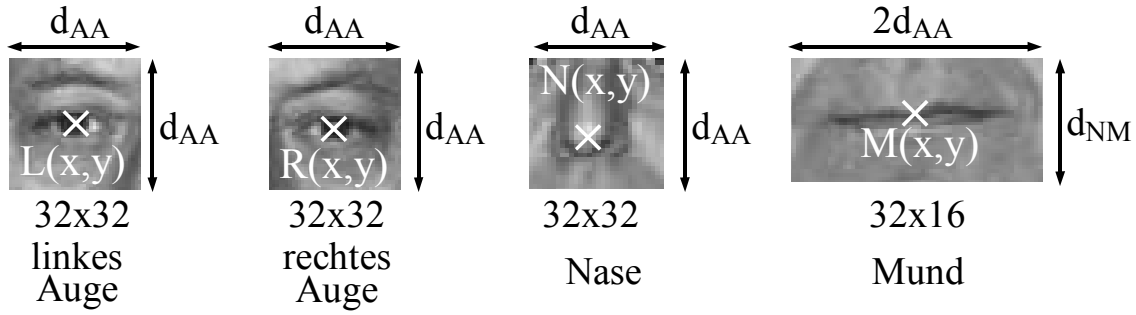


Abbildung 4.13: Typische Bildausschnitte und deren Metriken

Vor der Fusion werden die einzelnen Systeme zunächst nach den vorgestellen Abläufen trainiert und deren Einzelergebnisse gemessen, wie in Tabelle 4.4 angegeben. Zur Erkennung werden sowohl diskrete als auch kontinuierliche Modellierungsparadigmen mit unterschiedlichen Systemparametern angewandt.

Ausschnitt	Erkennungsrate [%]	Bemerkung
gesamtes Gesicht	98,13	diskret
linkes Auge	96,26	kontinuierlich
rechtes Auge	95,95	kontinuierlich
Nase	85,67	diskret
Mund	30,53	diskret

Tabelle 4.4: Identifikationsraten für verschiedene Bildausschnitte

Grundsätzlich weisen Systeme, die auf den Nasen- und Mundausschnitten basieren, eine vergleichsweise schwache Erkennungsleistung auf. Zurückgeführt werden kann diese offensichtliche Unterlegenheit auf die hohe Intraklassenvarianz, beispielsweise hervorgerufen durch verschiedene Gesichtsausdrücke. Auf informationsreicheren Augenbereichen basierende Klassifikatoren weisen hingegen beachtlich höhere Raten auf, welche nahe an die des gesamten Gesichtsausschnitts heranreichen.

In der nachgeschalteten Votingstufe werden die Einzelergebnisse für ein unbekanntes Muster über eine Mehrheitsentscheidung verknüpft, d.h. es gilt jeweils die Person als erkannt, welche am häufigsten gewählt wurde. Bei der Zusammenfassung von drei Systemen sind somit zwei gleichlautende Treffer notwendig. Eine funktionale Konstellation für eine Entscheidungsfindung ist die Berücksichtigung der Ergebnisse eines ganzheitlichen Erkenners, sowie jeweils die besten Systeme für das linke und rechte Auge. Hierdurch kann der verbleibende Erkennungsfehler immerhin von 1.87% auf 1.56% gesenkt werden.

Prinzipiell scheint die Steigerung der Zuverlässigkeit der vorgestellten HMM-basierten Identifikationssysteme über Mehrheitsentscheidungen bereits mit trivialen Regeln möglich. Die Hinzunahme weiterer Einzelergebnisse und die Einführung eines adaptierten Regelwerks stellt zudem ein enormes Potential bezüglich der Sicherheit des Systems dar. Für praxisnahe Anwendungen kann sich jedoch die Komplexität des Gesamtsystems als problematisch erweisen, da die Rechendauer in erster Näherung mit der Anzahl an verwendeten Einzelsystemen linear steigt. Alternativ zur Untergliederung des Gesichts in Viseme bietet sich auch ein Rescoring über die Gewichtung einzelner Modellzustände für verschiedene Bereiche wie Augen und Nase auf Featureebene an.

4.2 Verifikation und Konfidenzmaße

Nach der Identifikation respektive Zuordnung eines unbekannten Gesichtsbildes zu einer bekannten Klasse, genauer der Auswahl aus *Einem von Vielen*, dient die Verifikation der Überprüfung des gefundenen, möglicherweise fehlerbehafteten Ergebnisses. Dabei kann es auch vorkommen, dass das getestete Personenmuster nicht Bestandteil des vorher vereinbarten Personenkreises ist. Bei dem hier verwendeten Klassifikator nach dem Prinzip der größten Ähnlichkeit, wird jedoch systembedingt auch bei Eindringlingen immer das Modell der ähnlichsten Person ausgegeben. Die herausragende Anwendungsmöglichkeit der Verifikation ist

neben der Erkennung von Zuordnungsfehlern durch den *Einer-gegen-Einen* Vergleich gegeben, bei dem überprüft wird, ob ein vorliegendes Bild zu dem gegebenen Modell gehören kann.

In einem Identifikationsszenario sind bei der Einstufung einer Person aus einer Gruppe bekannter Personen prinzipiell die folgenden Ausgänge möglich [Eic00a]:

- Ein unbekannte Person, welche nicht im Bestand der bekannten Individuen ist, wird richtigerweise zurückgewiesen.
- Durch einen Erkennungsfehler wird eine Person fälschlicherweise einer anderen Klasse zugeordnet. In diesem Fall ist eine Rückweisung einer Annahme vorzuziehen.
- Trotz einer richtigen Erkennung wird eine Person vom System fälschlicherweise abgewiesen, was idealtypisch nicht passieren sollte.
- Ein bekannte Person wird vom System richtigerweise angenommen.

Für die Verifikation werden auf Basis der Klassifikationsergebnisse Konfidenzmaße abgeleitet, die die Richtigkeit der gemachten Zuordnung bzw. Vorgabe prüfen. Wenn das zu einem Ergebnis korrespondierende Konfidenzmaß über einer zuvor definierten Schwelle liegt, gilt die Vorgabe bzw. das bei der Identifikation gewonnene Zwischenergebnis als angenommen. Andernfalls wird die Person vom System als unbekannt zurückgewiesen bzw. das Ergebnis als Fehlzuordnung eingestuft. Zu diesem Zweck wird auf Basis der bereits bekannten Identifikationssysteme im Weiteren zunächst ein geeignetes Konfidenzmaß ermittelt. Das günstigste gefundene Maß wird im Anschluss für eine Verifikationsaufgabe parametrisiert.

4.2.1 Normalisierte A-Posteriori-Wahrscheinlichkeiten

In einem ersten Ansatz können die bei der Bildklassifikation mit P2DHMM bestimmten Produktionswahrscheinlichkeiten analog zur Spracherkennung als Grundlage für ein Sicherheitsmaß dienen [Wil98]. Im einfachsten Fall wird die Identifikation eines unbekannten Gesichtsbildes $\mathbf{O}$ zum Personenmodell λ_i zurückgewiesen, wenn die Produktionswahrscheinlichkeit $P(\lambda_i|\mathbf{O})$ unter der vorgegebenen Schwelle τ liegt:

$$P(\mathbf{O}|\lambda_i) < \tau \tag{4.9}$$

Durch Normierung auf die Summe der obersten C folgenden Produktionswahrscheinlichkeiten der Rangliste kann die Aussagekräftigkeit zusätzlich deutlich gesteigert werden [Eic00a, Eic02]. Bei den durchgeführten Versuchen wird die Anzahl $C = 25$ verwendet. Die Rückweisung der Erkennung erfolgt wenn gilt:

$$\frac{P(\mathbf{O}|\lambda_i)}{\sum_{j=1}^{C} P(\mathbf{O}|\lambda_j)} < \tau \tag{4.10}$$

4.2.2 Vergleich des getesteten und erkannten Bildes

Die diesem Ansatz zugrunde liegende Idee ist, dass das Verhältnis zwischen den Produktionswahrscheinlichkeiten des Testbildes und dem repräsentativsten Trainingsbild bei der gleichen Person nicht zu groß sein darf. Dieser neuartige Ansatz ist im Besonderen für die Verifikation einer behaupteten Identität geeignet. Mathematisch kann diese Formulierung in Gleichung 4.11 ausgedrückt werden.

$$\tau > 1 - \frac{|P(\mathbf{O}|\lambda_{i,test}) - P(\mathbf{O}|\lambda_{i,training})|}{P(\mathbf{O}|\lambda_{i,training})} \tag{4.11}$$

Im Einzelnen wird die Person in der Identifikationsphase zunächst über das wahrscheinlichste Modell ermittelt. Sind sich Trainings- und Testbild ähnlich, ist die Differenz der Produktionswahrscheinlichkeiten für das Personenmodell tendenziell klein. Durch Normierung auf die Wahrscheinlichkeit des Trainingsbildes ergibt sich das gesamte vorgestellte Sicherheitsmaß.

4.2.3 Summierte Rangunterschiede

Ähnlich zum ersten Ansatz werden auch bei diesem Maß nicht nur das Beste sondern die B besten, respektive wahrscheinlichsten Modelle ermittelt. Im Gegensatz zu den bereits vorgestellten Verfahren werden die Produktionswahrscheinlichkeiten zur Bewertung jedoch nicht herangezogen. Hintergrund hierbei ist vielmehr, dass bei der Identifikation einer zu erkennenden Person eine ähnliche B-besten Liste erzeugt wird, wie bei den Trainingsbildern der erkannten Klasse [Eic02]. Das Sicherheitsmaß ergibt sich durch die Abweichungen der Ranglisten für beide Bilder. Die Berechnung eines Wertes sei anhand des Beispiels nach Tabelle 4.5 vergegenwärtigt.

Rang	Trainings-bild	Test-bild	Differenz-betrag
1	P00035	P00035	$\|1-1\| = 0$
2	P00743	P01047	$\|2-3\| = 1$
3	P00244	P00743	$\|3-4\| = 1$
4	P00142	P00244	$\|4-5\| = 1$
5	P01047	P00142	$\|5-2\| = 3$
6	P00011	P00011	$\|6-6\| = 0$
			$\sum = 6$

Tabelle 4.5: Berechnung der summierten Rangunterschiede

Ausgehend von den sich ergebenden Ranglisten für das entsprechende Trainingsbild, gegeben durch die zuvor erkannten Klassen, werden die Positionen der wahrscheinlichsten Modelle zeilenweise über den Betrag ihrer Rangdifferenz bewertet und anschließend durch Summierung zusammengefasst. Wenn die Listen große Unterschiede aufweisen, wird die

ermittelte Summe über einem zuvor eingestellten Schwellwert liegen, woraufhin das gefundene Ergebnis zurückgewiesen wird.

4.2.4 Levenshtein Distanz

Abweichend von einer listenplatzweisen Auswertung der Positionen innerhalb der B-besten Liste, kann ein weiteres Sicherheitsmaß mit Hilfe der Levenshtein Distanz definiert werden. Die Levenshtein Distanz dient ursprünglich zum dynamischen Vergleich von Zeichenketten bzw. Symbolfolgen. Für die Anwendung als Sicherheitsmaß muss somit jeder Person der Trainingsmenge ein eigenes Symbol zugeordnet werden. Die beiden B-besten Listen für das Trainings- und Testbild werden dann als Zeichenketten interpretiert, deren Abstand zueinander ausgewertet werden soll. Die Berechnung sowie ausführlichere Erläuterungen zu diesem Distanzmaß können dem Anhang B.7 entnommen werden.

4.2.5 Experimente und Ergebnisse

Zur Eignungsprüfung dieser vier Sicherheitsmaße werden die im Kapitel 4.1.4 vorgestellten Datensätze der FERET und AT&T Datenbank verwendet, deren Bildqualitäten identisch sind. Von der FERET Datenbank dienen 321 Beispielbilder als dem System bekannte Personen, 40 Kandidaten des AT&T Korpus werden als Eindringlinge verwendet. Zum Aufdecken von Erkennungsfehlern sowie zur Erkennung von Eindringlingen wird ein nicht optimales diskretes Basissystem mit 5×5 Zuständen, einer Blockgröße von 16 Pixeln und einer Überlappung von 12 Bildpunkten als Grundlage für die Versuchsreihen herangezogen. Neben dem vorherigen Ausschneiden und einer Größennormierung auf 64×96 Pixel werden keine weiteren vorverarbeitenden Maßnahmen getroffen.

Zur Findung des günstigsten Konfidenzmaßes werden die Schwellwerte τ als freie Parameter über ihre möglichen Wertebereiche variiert. In den drei parametrisierten Diagrammen in Abbildung 4.14 ist die jeweilige prozentuale Fehlerrate über der Falschalarmrate aufgetragen. Ein falscher Alarm ergibt sich durch die fehlerhafte Rückweisung einer dem System bekannten Person. Bei den Erkennungsfehlern kann zwischen Eindringlingen und der Rate unerkannter Erkennungsfehler unterschieden werden. Zudem wird in einem weiteren Diagramm die Summe der beiden Fehler angegeben.

Je geringer die Fläche unterhalb der Kurve ist, desto leistungsfähiger ist das gewählte Rückweisungsmaß. Bei den vier vorliegenden Fällen setzen sich nach diesem Kriterium die beiden listenbasierten Ansätze ab. Von diesen beiden wiederum erweist sich das Maß über den Rangfolgenunterschied als überlegen.

Nach der Selektion des Maßes auf Basis des Rangfolgenunterschieds muss vor der Realisierung eines Zugangssystems der bis jetzt noch freie Parameter τ des Schwellwerts ermittelt werden. Da aufgrund obiger Diagramme eine optimale Wahl nicht ohne Weiteres vorgenommen werden kann und bei der angestrebten Verifikation interne Zuordnungsfehler nicht von Relevanz sind, wird zunächst eine Methode zur Beurteilung von Biometriesystemen vorgestellt.

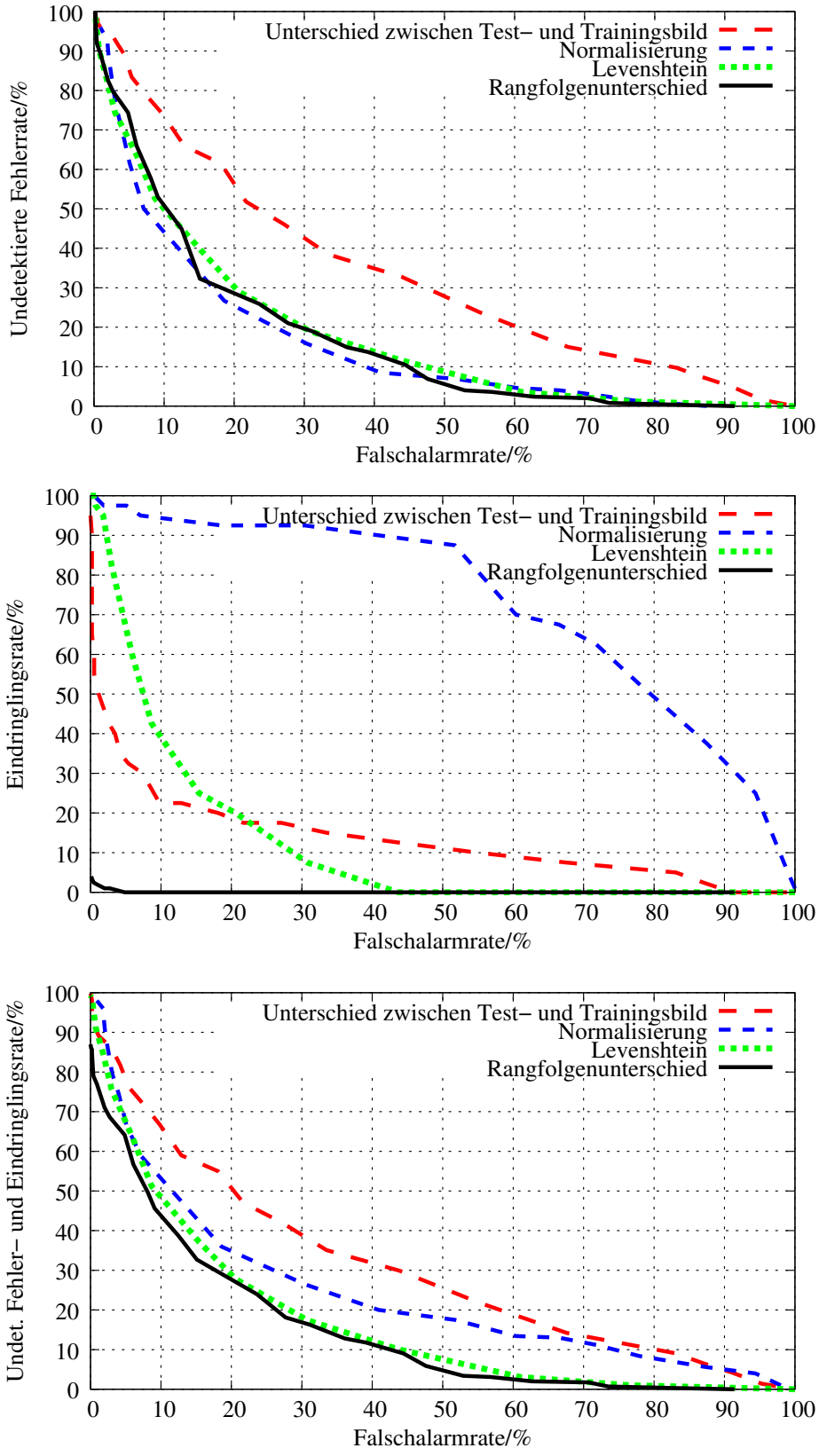


Abbildung 4.14: Rückweisung von Erkennungsfehlern (oben), von Eindringlingen (Mitte) sowie der Summe beider Fehler (unten), aufgetragen über der Falschalarmrate

Bei der Verifikation, dem *Eins-zu-Eins* Vergleich mit Bestätigung der behaupteten Identität, gibt es bei der Authentisierung an einem System ähnlich zum Identifikationsszenario prinzipiell vier verschiedene Situationen [Pet02]:

1. Eine berechtigte Person wird akzeptiert.
2. Eine berechtigte Person wird zurückgewiesen.
3. Eine nicht berechtigte Person wird akzeptiert.
4. Eine nicht berechtigte Person wird zurückgewiesen.

Offensichtlich sind die beiden Fälle 2 und 3 aus obiger Liste zu vermeiden. Die sogenannte Falschrückweisungsrate[6] (FRR) ist das Maß für die Häufigkeit, mit der eine berechtigte Persone n unberechtigterweise zurückgewiesen wurde. Die FRR(n) ist in der Regel ein Merkmal für die Benutzerakzeptanz, da falsche Abweisungen für den Benutzer lästig sind. Analog gibt die Falschannahmerate[7] (FAR) die Häufigkeit an, mit der nicht berechtigte Personen als berechtigt eingestuft werden. Da eine falsche Annahme im Regelfall zu einem Schaden führt, ist die FAR ein sicherheitsrelevantes Maß. Da die beiden Größen FAR(n) und FRR(n) nichtstationäre statistische Größen sind, weisen sie eine starke individuelle Abhängigkeit auf. Die Raten sind durch die folgenden Verhältnisse bestimmt:

$$FAR(n) = \frac{\text{Zahl der erfolgreichen Angriffe gegen eine Person } n}{\text{Gesamtzahl der Angriffe gegen eine Person } n} \tag{4.12}$$

$$FRR(n) = \frac{\text{Zahl der zurückgewiesenen Verifikationsversuche einer berechtigen Person } n}{\text{Zahl der Verifikationsversuche einer berechtigten Person } n} \tag{4.13}$$

Durch Mittelung der obigen Werte über alle Personen ergeben sich die personenunabhängigen Raten FAR und FRR. Zur qualtitativen Beschreibung eines biometrischen Systems und zur Findung eines geeigneten Schwellwerts werden diese beide Kurven gemeinsam in einem Diagramm als Funktion von τ aufgetragen. Der Schnittpunkt der beiden Kurven gibt den Wert der zu wählenden Schwelle vor. Der verbleibende Systemfehler, die sogenannte *Equal Error Rate* (EER), sollte idealerweise sehr klein sein.

Bei einer zu geringen Bildmenge kann es jedoch vorkommen, dass keine aussagekräftige Schwelle gefunden werden kann. So lag die EER bei der AT&T-Datenbank über ein großes Intervall hinweg bei 0%. Zur Einstellung des Arbeitspunktes wird daher der oben vorgestellte FERET-Korpus mit 1.196 Personen verwendet. Für die Ermittlung der FAR und FRR Kurven werden alle möglichen Testbilder der Reihe nach mit einem selektierten Modell verglichen und bewertet.

Wie aus dem Diagramm 4.15 ersichtlich wird, liegt der Schnittpunkt der beiden Kurven bei einer normierten Schwelle von etwa $\tau = 0.76$. Die EER ergibt sich beim verwendeten System zu etwa 3%. Durch die Verwendung anderer Konfidenzmaße konnte keine weitere Reduktion der EER erreicht werden.

[6] False Rejection Rate
[7] False Acceptance Rate

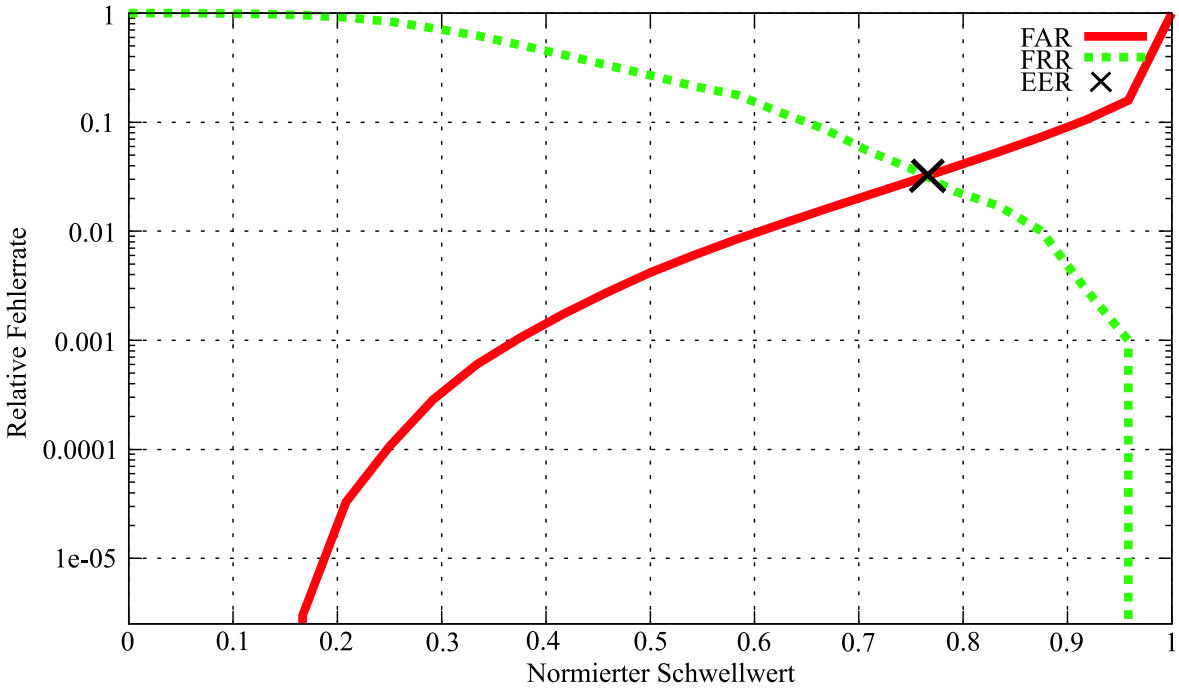


Abbildung 4.15: Logarithmierte FAR und FRR Kurven zur Ermittelung einer geeigneten Schwelle über die EER

4.3 Implementierung eines Systems zur Zugangskontrolle

Auf Basis der bisher entwickelten und parametrisierten Systembausteine wird zur Steigerung der Flugsicherheit in diesem Kapitel ein Eincheck-Überwachungssystem vorgestellt und prototypisch implementiert. Idee bei diesem Szenario ist, dass nach einer Identitätsprüfung am Abflugschalter ein Modell mit den Gesichtsdaten des Passagiers auf eine funkgesteuerte Boardingkarte[8] geschrieben wird. Zur Findung nicht autorisierter Fluggäste soll während des Betretens des Türbereich im Flugzeugs überprüft werden, ob die Daten des aufgenommenen Gesichts mit dem Modell auf dem Boardingpass identisch sind. Die automatische Detektion der Gesichtsbereiche wird jeweils durch das Personal bzw. die Präsenz eines RFID[9]-Tags im Eingangsbereich initiiert.

Zur Gesichtsdetektion wird das vorgestellte Condensation-Verfahren verwendet. Bei der Bestimmung des Gesichtsausschnitts für die Modellbildung und während der Klassifikation soll nun die Ausgabe des Detektors direkt herangezogen werden. Eine zusätzliche Findung der exakten Augen- und Mundkoordinaten nach Kapitel 3.6 wird aufgrund der erzielten Genauigkeit zunächst nicht in Betracht gezogen. Über eine Heuristik kann der Gesichtsbereich auf den in Kapitel 4.1.1 vorgestellten Ausschnitt hochgerechnet werden [Eic01]. Da das Gesicht des Passagiers während des Einchecken und Einsteigens üblicherweise über einen Zeitraum von mehreren Sekunden im Erfassungsbereich des Sensors verbleibt, können jeweils mehrere Aufnahmen zur Verarbeitung herangezogen werden. Um Erkennungsfehlern durch evtl. falsch oder unpräzise detektierte Gesichter entgegenzuwirken, werden alle gefundenen

[8] So genanntes Szenario mit Template-on-Card
[9] Abkürzung für Radio Frequency Identification (Device)

Bilder einem zweiten Feinlokalisationsschritt unterzogen. Ohne den Berechnungsaufwand gravierend zu erhöhen hat sich diese triviale Maßnahme in ersten Versuchen als äußerst effektiv herausgestellt. Beim Betreten des Flugzeugs wird über das Konfidenzmaß nach Listenunterschied entschieden, ob Modell und aktuelle Aufnahme jeweils übereinstimmen. In der späteren Praxis soll das System dem Flugbegleitpersonal die aktuelle Verifikationsentscheidung mitteilen, wodurch entsprechende Maßnahmen eingeleitet werden können [Gau05]. In Abbildung 4.16 ist eine typische Bildfolge des An-Bord-Gehens gezeigt.

Abbildung 4.16: Typische Bildfolge beim Betreten eines Flugzeugs

Die besonderen Schwierigkeiten im Zusammenhang dieser Problemstellung sind einerseits das Zusammenspiel zwischen Detektor und Verifikator sowie die diskutierten Einflüsse durch die Änderung der Belichtung. Die äußeren Einflüsse während des Check-Ins können zwar weitgehend kontrolliert werden, im Türbereich des Flugzeugs sind aber während des Einsteigens vielfältige Situationen denkbar. Alterungsprozesse können bei dieser Anwendung ausgeschlossen werden.

Aufgrund des Fehlens realistischer Daten wird zur Simulation eines Gesamtsystems im Vorfeld eine Datenbank mit achtzehn Personen erstellt. Pro Person werden jeweils vier unterschiedlich belichtete Bildserien aufgezeichnet, wobei die Personen von der Kamera frontal erfasst werden. In den aufgezeichneten Sequenzen wird kontinuierlich nach Gesichtern gesucht. Damit analog zu den Referenzdaten ebenfalls Aufnahmen mit Blick in die Kamera entstehen, wird die Kooperationsbereitschaft der Personen bei allen Aufnahmen vorausgesetzt. Zur Merkmalsextraktion werden die Gesichtsbereiche auf eine Größe von 80×80 Pixel verkleinert. Zur Modellierung werden jeweils zehn Bilder im zeitlichen Abstand von einer Sekunde verwendet, wodurch eine höhere Robustheit gewährleistet werden kann. Als Basissystem wird das oben vorgestellte FERET-System mit 1.196 Personen benutzt. Die Erkennungslisten der reklassifizierten Bilder werden für die anschließende Verifikation zur Beschleunigung bereits im Vorfeld erstellt. Aus den drei verbleibenden Bildsequenzen werden für die Tests ebenfalls fünf Einzelbilder im äquidistanten Abstand von einer Sekunde gewählt und jeweils gegen die trainierten Modelle der ersten Serie verglichen. Zur qualitativen Bewertung des Systems wird die im Zusammenhang der Signaldetektion bekannte Empfänger-Operationscharakteristik, geläufiger als *Receiver Operating Characteristic* (ROC) ermittelt, welche durch das Auftragen der FRR gegen die FAR in ein Diagramm gegeben ist.

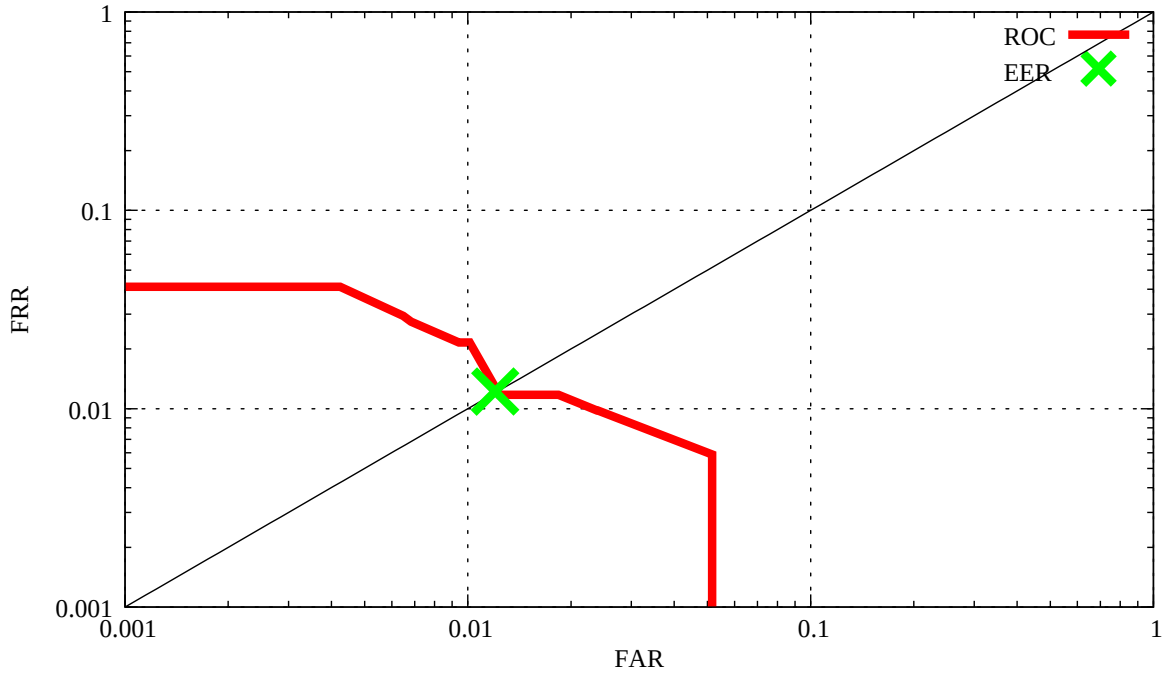


Abbildung 4.17: ROC-Kurve eines vollautomatischen Eingangskontrollsystems

Die gesuchte EER ergibt sich bei den verwendeten Systemkomponenten nach einer Kreuzvalidierung zu unter 2%, was bereits auf eine akzeptable Gesamtkonstellation hindeutet. Je nach tolerierbarer Benutzerakzeptanz bzw. Falschalarmrate kann die Eindringlingsrate bzw. das Sicherheitsrisiko sehr niedrig gehalten werden. Eine Sichtung der falsch klassifizierten Bilder weist hauptsächlich eine Schwäche bezüglich extrem unterschiedlicher Belichtungsverhältnisse zwischen den Trainings- und Testbildern auf. Darüber hinaus bestehen noch Abweichungen zwischen den automatisch gefundenen Gesichtsbereichen und den Referenzausschnitten des Basissystems. Die im Moment verbleibende Diskrepanz zwischen den idealen und gefundenen Gesichtsbereichen kann in folgenden Experimenten durch eine präzisere Feinlokalisierung auf Basis einzelner Koordinaten für Augen und Mund weiter minimiert werden. Eine abschließende Betrachtung für den Einsatz unter echten Bedingungen ist jedoch aufgrund des Fehlens realer Boardingsequenzen zu diesem Zeitpunkt noch nicht möglich.

4.4 Erkennung von Profilbildern

Nachdem die stochastische Erkennung von Frontalaufnahmen unter variierenden Belichtungseinflüssen mit beachtlichen Erkennungsraten erfolgreich implementiert werden konnte, soll im Folgenden die Gesichtserkennung mit einer Kopfdrehung von $90°$ um die Z-Achse zwischen Aufnahme und Modell exemplarisch untersucht werden. Neben einer starken Variation der Belichtungseinflüsse stellt die blickwinkelunabhängige Gesichtserkennung für alle existierenden Verfahren die größte Herausforderung dar [Fro97, Gro01, San04]. Es gibt untermauerte wahrnehmungspsychologische Theorien die besagen, dass ein Bildvergleich einer

beliebigen Zwischenansicht durch Interpolation bekannter Prototypen aus Vorderansicht und Profil erfolgen kann. Ferner gilt die blickwinkelunabhängige Erkennung von oft gesehenen Personen als sicherer gegenüber der selten gesehenen oder unbekannten Personen [Val97].

Die im Weiteren verfolgte Aufgabenstellung soll die Erkennung einer vormals *unbekannten* Person über die entsprechende Profilbildaufnahme sein, wenn nur *eine* Frontalaufnahme bekannt ist. Diese als äußerst schwierig anzusehende Erkennungsaufgabe beschränkt sich auf die Zuordnung von Bildpaaren aus der so genannten MUGSHOT-Datenbank. Sinnvoll kann dieser Vergleich für eine zusätzlich nachgeschaltete Verifikation oder zur Einschränkung des Suchbereichs in großen Datenbanken sein [Gor95]. Ferner wird durch die Profilbilderkennung die Verarbeitung von Bildmaterial von nicht kooperativen oder unaufmerksamen Personen ermöglicht.

Wie im Rahmen der Gesichtserkennung bereits angedeutet wurde, könnte zur Lösung dieser Aufgabe beispielsweise auf eine Kombination von dreidimensionalen Kopfmodellen mit Oberflächentexturen zurückgegriffen werden. Obwohl in der Literatur mehrere repräsentative Möglichkeiten vorgestellt wurden [Gro94, Vet98, Lav00, Che01, Bla02b], welche zumeist für Animationszwecke eingesetzt werden, soll im Rahmen dieser Arbeit von 3D Modellen Abstand genommen werden. Der Grund hierfür liegt im üblicherweise hohen Aufwand zur Erstellung der hochdimensionalen Modelle sowie der komplexen dreidimensionalen Objekterfassung. Stattdessen soll zur Erkennung der azimutal um 90° gedrehten Gesichter eine alternative, neuartige Möglichkeit Anklang finden. Hintergrund ist die Einführung einer vorgeschalteten Stufe zur Bild- bzw. Merkmalstransformation, welche mit Hilfe geeigneter Daten von einem NN bzw. MLP selbständig erlernt werden kann. Theoretisch angedeutet wurde die Verwendung von Operatoren zur perspektivischen Änderung eines Bildes in [O'R92, Wür94], die Änderung der Merkmale in [Mau95].

Zunächst ist es das konkrete Ziel das Bild einer zu erkennenden Person für den Vergleich über ein erscheinungsbasiertes Modell auf die bekannte Ansicht zurückzuführen. Nach der vorgestellten Transformation kann im Anschluss wieder ein vornehmlich auf Frontalbilder spezialisiertes HMM System angewendet werden. Zusätzlich zur rein bildbasierten Transformation sollen die aus einer MLP Vorstufe und einem HMM zur Erkennung bestehenden Systeme in eine hybride Struktur mit gemeinsamen Parametern überführt werden. Neben einem Operator im Bildbereich wird darüber hinaus von einer merkmalsbasierten Transformation mit den bereits vorgestellten Eigenfaces Gebrauch gemacht.

4.4.1 Verwendetes Bildmaterial

Prinzipiell sollen in der Mugshot-Erkennungsaufgabe Profilbilder mit Frontalansichten verglichen werden. Zu diesem Zweck muss folglich auf Bildmaterial zurückgegriffen werden, welches nicht nur Vorderansichten enthält, sondern Bildpaare mit beiden Ansichten. Daher wird neben der bereits vorgestellten FERET Datenbank mit ihrer beschränkten Anzahl von Profilansichten auf eine weitere Datenbank zurückgegriffen.

Diese Datenbank ist die *NIST*[10] *Special Database 18: Daten zur Mugshot Identifikation - Frontalansichten und Profile* [Wat94]. Der Begriff Mugshot stammt aus dem amerikanischen Sprachgebrauch und steht für die im Zusammenhang der Verhaftung von potentiell Kriminellen und Straftätern erstellten Identifikationsphotos. Die aus FBI-Archiven erstellte Datenbank umfasst Bildmaterial von insgesamt 1.573 Individuen in 8-Bit kodierten Grauwertbildern. Üblicherweise existieren jeweils zwei Bilder zu jeder Person: die Frontal- und die Profilansicht. Hauptsächlich handelt es sich bei den Bildpaaren um Gesichter von männlichen Personen, mit einem deutlich geringeren Anteil sind aber auch Frauen enthalten. Die Personen decken ein breites Spektrum bezüglich Herkunft und Hautfarbe sowie veränderlichen Sekundärmerkmalen wie Brille oder Bart ab.

Nach einer Auswahl der zur Erkennung qualitativ ausreichenden Bildpaare wird das verbleibende Material einer gemeinsamen Vorverarbeitung unterzogen. Analog zu den bereits vorgestellten Ansätzen bestehen die Verarbeitungsschritte auch hier aus dem manuellen Setzen der Koordinaten für Augen, Nase und Mund. Auf deren Basis werden alle Bilder einheitlich ausgerichtet, ausgeschnitten und auf eine Größe von 64×64 Punkten gebracht. Auszüge der Resultate dieser Maßnahme sind in Abbildung 4.18 ersichtlich. Ohne Beschränkung der Allgemeinheit wurden nur Profilansichten der rechten Gesichtshälfte bereitgestellt. Zur Bearbeitung der anderen Seite muss somit eine vertikale Spiegelung beider Aufnahmen vorgeschaltet werden.

Abbildung 4.18: Vorverarbeitete Gesichter aus der MUGSHOT Datenbank [Wat94]

Die vollständige Übersicht des eingesetzten Testkorpus kann dem Anhang A.3 entnommen werden. Insgesamt werden auf die beschriebene Weise für die Durchführung der weiteren Versuche drei disjunkte Datensätze zusammengestellt:

1. Ein Satz zur Schätzung der Parameter des Rotationsoperators bestehend aus 600 Bildpaaren, fortan DB-1 genannt.

2. Der eigentliche Testsatz bestehend aus 100 Bildpaaren, kurz DB-2.

3. Die Menge DB-3 besteht aus 100 Bildpaaren der FERET Datenbank und dient zur Validierung des MLP-Trainings.

[10] Abkürzung für *National Institute of Standards and Technology* (NIST)

Da aufgrund der Schwierigkeit der Aufgabe bereits im Vorfeld keine perfekte Erkennung zu erwarten ist, wurde die Zuordnungsfähigkeit des Menschen mit Hilfe von 15 Probanden unter vergleichbaren Bedingungen gemessen. Dies bietet eine passende Bewertungsgrundlage und ermöglicht eine adäquate Leistungseinordnung der rechnergestützten Erkennung.

Die Testpersonen bekommen dazu die Aufgabe der Reihe nach 15 zufällig gewählte Profilansichten je einem aus 100 gezeigten Gesichtern der DB-2 zuzuordnen. Dabei wurden insgesamt Identifikationsraten zwischen ca. 30% und 94% erreicht. Im Mittel lag die Rate der richtig zugeordneten Personen bei kaum mehr als 72%, was den Schwierigkeitsgrad der Erkennungsaufgabe noch einmal unterstreicht. Dieses mit kleiner Stichprobenzahl gewonnene Ergebnis deckt sich gut mit den von Valentin beschriebenen Perzeptionsversuchen [Val97].

Vor der Vorstellung spezialisierter Systeme werden im Anschluss die bekannten frontalen Gesichtserkennungssysteme auf ihre Eignung für die neue Erkennungsaufgabe getestet.

4.4.2 Direkte Profilbilderkennung mit HMM

Aufgrund der Elastizität von P2DHMM ist eine robuste Erkennung auch bei moderaten Änderungen des Blickwinkels möglich. So wurde von Eickeler berichtet, dass der verwendete Ansatz aufgrund seiner Dynamik in der Lage ist, Bilder mit Kopfdrehungen von bis ca. $20°$ problemlos zu erkennen [Eic02]. Belegt werden konnte dies durch nahezu perfekte Identifikationsraten auf entsprechenden Daten.

Vor den weiteren Untersuchungen stellt sich daher zunächst die Frage, ob und wie gut die erscheinungsbasierten Modelle bei einer Kopfdrehung von $90°$ funktionieren. Dazu werden die Frontalansichten der 100 Testpersonen nach den in Kapitel 4.1.4 erläuterten Schritten trainiert. Zur Ermittlung der Identifikationsrate werden dann jeweils die Profilansichten des Systems verwendet. Obwohl die Erkennung prinzipiell auch umgekehrt erfolgen könnte, soll im Weiteren nur noch die Erkennung im Profilbereich berücksichtigt werden.

Zur Modellierung der Gesichtsdaten werden P2DHMM mit 3×3 bis 5×5 Zuständen, einer Menge von 15 bis 28 DCT-Koeffizienten und einer Blocküberlappung zwischen 0% und 75% verwendet. Die Erkennungsrate variiert bei diesen Vorversuchen zwischen 4% und 10%. Grund für diese erwartet geringe Akkuratheit ist die offensichtlich zu große Abweichung zwischen der gelernten und der zu erkennenden Ansicht. Die Elastizität der P2DHMM reicht demnach für eine akzeptable Zuordnung bei Weitem nicht aus.

Zur Anpassung an die besonderen Gegebenheiten der Erkennungsaufgabe wird eine einfache Manipulation an den Modellen vorgenommen. Hierdurch müssen während der Erkennung nur die sichtbaren Bildbereiche durchlaufen werden. Ermöglicht wird dies durch Änderung der Wahrscheinlichkeiten der internen Zustandsabfolge auf die Weise, dass die Sprungwahrscheinlichkeit nach dem Durchlaufen des halben Modells statt auf die nächste auf die letzte Spalte gesetzt wird. Durch diese Modifikation kann zwar bereits eine leichte Steigerung der Leistung festgestellt werden, das beste zu beobachtende Ergebnis liegt allerdings immer noch bei nur 16%.

4.4.3 Synthese künstlicher Testbilder

Wie bereits erläutert soll die Abweichung zwischen den Blickwinkeln vor der Erkennung in einem nächsten Schritt durch eine Transformation im Bildbereich ausgeglichen werden. Primäres Ziel der Aufgabe bleibt die Erkennung. Die Synthese künstlicher Ansichten soll nur als Interimsergebnis betrachtet werden.

Bei der Entwicklung dieses nicht trivialen bildbasierten Rotationsoperators treten mehrere Detailprobleme wie beispielsweise Verdeckungen von Bildpartien auf. Darüber hinaus sind in den beiden orthogonalen Ansichten nicht alle Korrespondenzen zwischen den Bildpunkten erlernbar. So ist beispielsweise die Nasenkrümmung oder die Form des Kinns in der Profilansicht nicht aus der Frontalaufnahme abzuleiten. Die Skizze A.1 im Anhang auf Seite 131 verdeutlicht diese Problematik.

Trotz dieser Einzelheiten soll im Folgenden jedoch angenommen werden, dass zwischen den beiden Ansichten ein derart hoher Zusammenhang besteht, dass die jeweiligen Ansichten zumindest in erster Näherung aus den gegebenen Bildern prädizierbar sind. Nach dem Motto *Exempla docent* können unter Verwendung geeigneten Beispielmaterials zumindest grundlegende Zusammenhänge automatisch erlernt werden.

Zur Realisierung der nicht-linearen, komplexen Operatoren werden zwei Strategien verfolgt: im ersten Ansatz wird versucht, eine neuronale Struktur mit Grauwertintensitäten zu implementieren, der zweite Operator verwendet die aus dem Bereich der Gesichtserkennung bekannten Eigenfaces, auf deren Basis die Rotation durchgeführt wird.

Zeilenweise Transformation horizontaler Bereiche. Die hintergründige Idee für dieses Vorgehen ist die Tatsache, dass sich bei der zugelassenen Drehung um die Z-Achse nur die horizontale Komponente eines Punktes ändert. Die Elevation eines Punktes bleibt in allen planaren Aufnahmen eines konzentrisch aufgenommenen, beliebig komplexen, dreidimensionalen Objekts gleich. Dieser Sachverhalt wird in Abbildung 4.19 verdeutlicht.

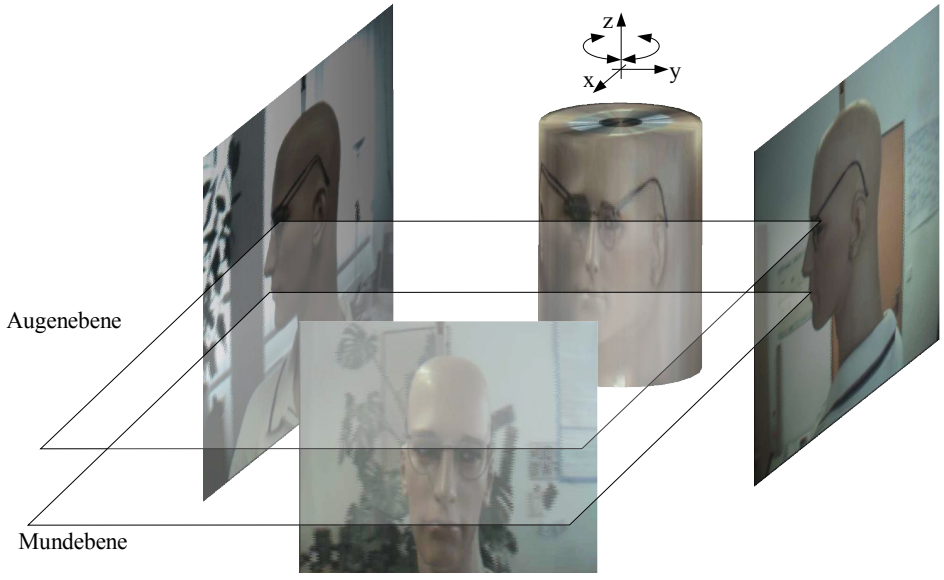


Abbildung 4.19: Erscheinungsformen eines Gesichts bei konzentrischen Aufnahmen

Für das weitere Vorgehen öffnet dieser offensichtliche Zusammenhang bereits eine Vielzahl möglicher neuronaler Architekturen [Wal00, Wal01b, Wal01c]. Verfolgt wird dabei jeweils ein zeilenweises Mapping zwischen den beiden korrespondierenden Gesichtsbereichen. Wichtig ist demnach die Tatsache, dass das Trainings- und Testmaterial möglichst keinen vertikalen Versatz aufweist, was durch die verwendete Vorverarbeitung als gegeben angesehen werden kann. Nur hierdurch kann gewährleistet werden, dass sich einzelne Zeilen auf die Zuordnungen zwischen beiden Ansichten spezialisieren können, wobei im speziellen keine exakte Pixelkorrespondenz anvisiert wird.

Technisch wird die Bildtransformation durch MLP mit entsprechender interner Konnektivität realisiert. Zur Steigerung der Robustheit und Prädizierbarkeit werden neben der eigentlichen Ursprungszeile zusätzlich die direkten Nachbarzeilen berücksichtigt. Eine erste lineare MLP-Architektur zur Generierung von Bildern mit seitlichem Blickwinkel ist in Bild 4.20 gezeigt. Die Pfeile in der Abbildung symbolisieren eine Vollvermaschung der Eingangs- und Ausgangsneuronen.

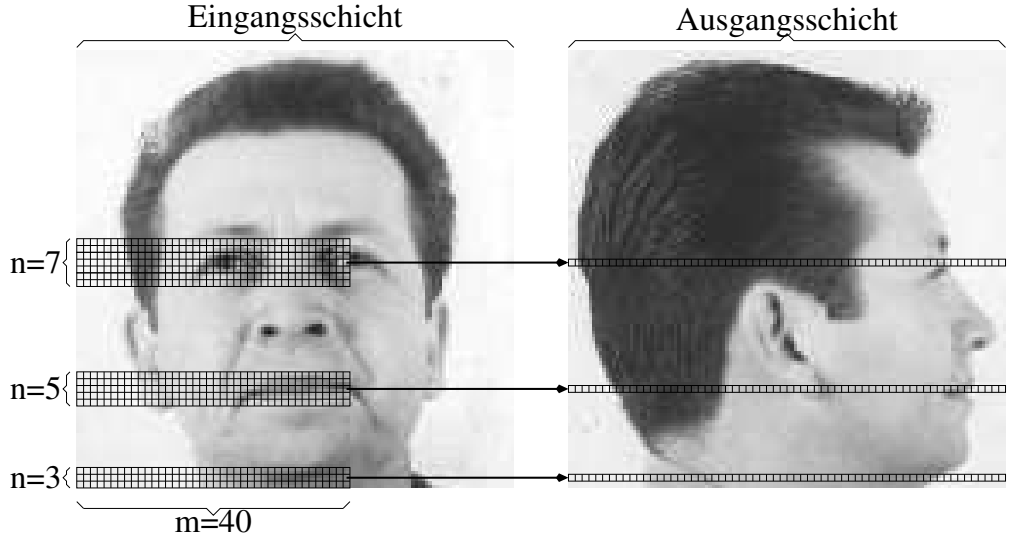


Abbildung 4.20: Zweischichtige Netzarchitektur zur Generierung künstlicher Ansichten

In Versuchen mit dieser Netzgattung werden jeweils Nachbarschaften von $n = [3, 5, 7]$ Zeilen genutzt. In den Randbereichen werden nur die verfügbaren Zeilen berücksichtigt. Der getestete horizontale Bereich umspannt Breiten aus der Menge $m = [32, 34, 36, 38, 40]$.

Neben den vorgestellten zweischichtigen MLP finden zudem auch Architekturen mit einer verdeckten Schicht nach Abbildung 4.21 Anwendung. Hierdurch wird der gesamten Struktur das Erlernen auch nicht-linearer Zusammenhänge ermöglicht. Innerhalb der verdeckten Schicht werden Sigmoidfunktionen eingesetzt.

Bezüglich des Trainings, stellen sich die in Kapitel 2.5.4 genannten, typischen Verläufe der zu erwartenden Fehler nach Abbildung 2.10 für die drei verwendeten Datenmengen ein. Nach einer zufallsbasierten Initialisierung der Gewichte, wird RPROP als Trainingsalgorithmus verwendet. Die Aktivierungen der einzelnen Neuronen eines trainierten Netzes sind exemplarisch in 4.22 abgebildet. Dunkle Intensitäten werden durch Kreise mit kleinem Radi-

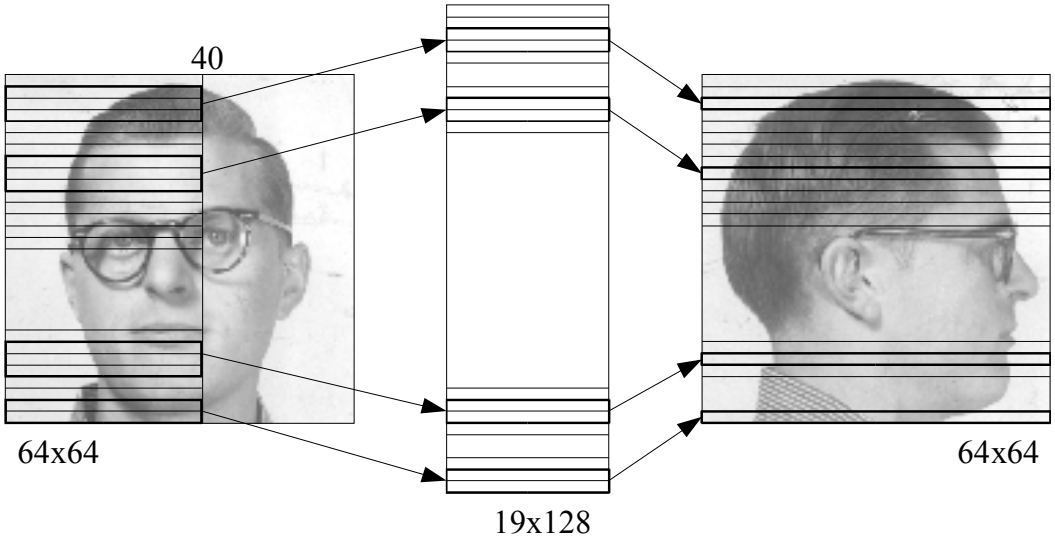


Abbildung 4.21: Erweitertes Rotationsnetz mit dreischichtiger Architektur

us repräsentiert. Wie hieran ersichtlich, lassen die Aktivierungen innerhalb der verborgenen Schicht keine Interpretationen über die gelernten Zusammenhänge zu.

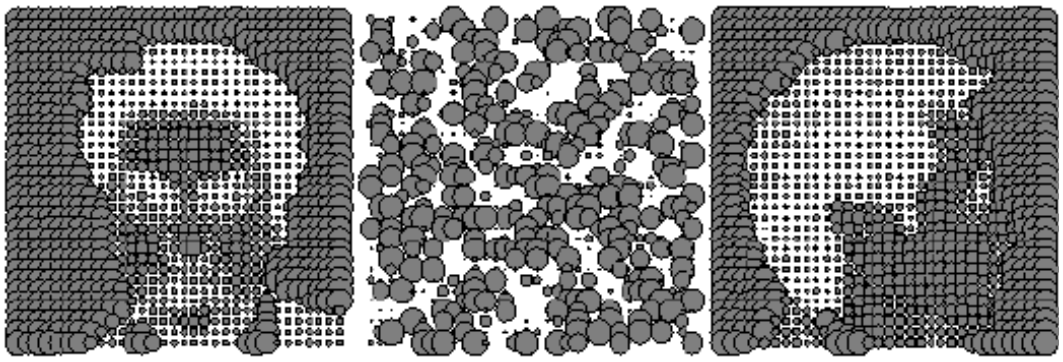


Abbildung 4.22: Aktivierungen der Neuronen eines dreischichtigen Netzwerks

Eine weitere Erhöhung der Anzahl an verdeckten Schichten wurde zwar erwägt und implementiert, in empirischen Versuchen zeigte sich jedoch, dass die Begrenztheit des Trainingsmaterials einer guten Generalisierung dieser großen Netze entgegenwirkt. In der Zusammenstellung nach Abbildung 4.23 sind originale Frontal- und Profilansichten zusammen mit typischen künstlichen Gegenstücken aufgeführt.
Offenbar weisen die synthetischen Ansichten zwar keine photorealistische Qualität auf, die wesentlichen Eigenschaften wie Haar- und Hautfarbe, sowie das Vorhandensein von Koteletten und Bärten können aber ansatzweise richtig wiedergegeben werden. Durch nachverarbeitende Maßnahmen mit Glättungsoperatoren oder Rauschfiltern konnte zwar eine leichte Verbesserung beobachtet werden, im Zusammenhang der später vorgestellten Erkennung brachte dies aber keine Steigerung. Mit dem Ziel hochwertigerer Bilder wird im Folgenden die Profilschätzung über Eigenmugshots untersucht.

Abbildung 4.23: Eingangsbilder mit künstlich erzeugten Profil- bzw. Frontalansichten

Bildtransformation über Eigenmugshots. Ähnlich zum vorhergehenden Rotationsoperator, werden auch in diesem Zusammenhang vorwärtsgerichtete NN zur Generierung von Profilansichten verwendet. Dies geschieht nun jedoch nicht mit Pixelintensitäten sondern mit den aus Bildern abgeleiteten Merkmalen [Mau95]. Die Merkmale werden im vorliegenden Fall durch die Projektionen der Gesichter in den Eigenfaceraum bestimmt [Wal03b]. Hierzu wird die im Grundlagenkapitel 2.4 erläuterte Hauptachsentransformation auf die Hilfsdatenbank DB-1 angewendet. Die resultierenden Eigenvektoren, fortan Eigenmugshots genannt, werden dabei getrennt für die Frontal- und Profilansichten bestimmt, welche in Abbildung 4.24 gezeigt sind. Dabei befinden sich in den beiden Spalten jeweils oben links die Mittelwerte zusammen mit den ersten vierzehn Eigenmugshots. Die Ordnung ist durch die höchsten Eigenwerte gegeben und verläuft im Bild von oben links nach unten rechts.

Abbildung 4.24: Mittelwerte und Eigenfaces der Frontal- sowie Profilansichten

Zur Transformation werden ebenfalls vollvermaschte MLP mit zwei-, drei- und vierschichtigen Architekturen eingesetzt. An die Eingänge werden die Gewichte der Frontalansichten gelegt, die Ausgänge liefern die Profilbildprojektionen. Die Anzahl der Eingangs- und Ausgangsneuronen variiert mit der Anzahl der verwendeten Gewichte zwischen 10 und 100. Mit der Verwendung dieser höherwertigen Merkmale geht gegenüber der bildbasierten Transformation eine stark reduzierte Anzahl benötigter Neuronen einher. Dadurch kann diesem Ansatz prinzipiell eine bessere Möglichkeit zur Generalisierung zugeschrieben werden.

In Abbildung 4.25 sind unter Verwendung der 30 relevantesten Gewichte beispielhaft einige Resultate abgedruckt. Das erste Bild zeigt dabei jeweils das Original, ein zweites die dazugehörige Rückprojektion auf Basis des durch DB-1 aufgespannten Eigenmugshotraums und das dritte die entsprechende künstlich geschaffene Ansicht.

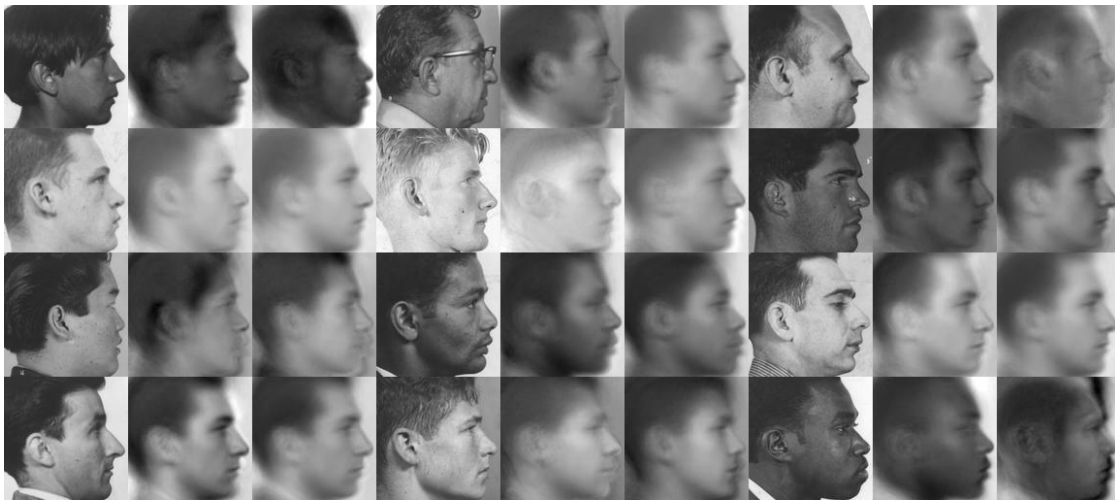


Abbildung 4.25: Auszug originaler und synthetisch erstellter Profilansichten

Wie aus den Beispielen ersichtlich, herrscht eine gewisse Diskrepanz zwischen den Originalbildern und den durch Rückprojektion gewonnenen Profilen. So kann beispielsweise die Brille im oberen, mittleren Bild nicht rekonstruiert werden. Die Ursache für die Abweichung scheint neben der Anzahl der verwendeten Eigenmugshots hauptsächlich darin begründet zu liegen, dass trotz der weitaus größeren Trainingsmenge nicht alle Details aus dem vormals unbekannten Testkorpus DB-2 durch die verfügbaren Eigenmugshots aus DB-1 wiederhergestellt werden können. Eine ideale Generalisierung durch das MLP konnte offenbar noch nicht voll erreicht werden. Die über das MLP geschätzten Profile weisen jedoch eine weitaus höhere Qualität auf. Zudem sind die künstlichen Profile den Rücktransformierten zumindest subjektiv weitestgehend ähnlich.

Eine Partitionierung der zu transformierenden Daten mit dem Ziel der Einführung mehrerer spezialisierter Netze für verschiedene Personengruppen hat bei beiden oben genannten Ansätzen zu keiner Verbesserung beitragen können. So hat bereits die Trennung von hell- und dunkelhäutigen Personen in zwei Gruppen eine Verringerung der Anzahl an Trainingsbeispielen um je ca. die Hälfte bedeutet. Der hieraus erwachsende Effekt sind schlechtere Generalisierungseigenschaften der Netze, was experimentell über den Trainingsfehler nachgewiesen wurde. Bei gleichzeitiger Beibehaltung der Anzahl an Beispielen für jede Untergruppe sollte die Aufspaltung aber prinzipiell möglich sein und zu merklichen Verbesserungen führen.

4.4.4 Hybride Profilbilderkennung mit HMM

Mit den auf die oben beschriebene Art entstandenen Bildern können jetzt kombinierte Erkennungssysteme nach Abbildung 4.26 konstruiert werden. Zur Zuordnung der durch bildbasierte Transformation entstandenen Bilder werden ausschließlich HMM verwendet. Zwar

ist nach der Bildrotation ebenfalls eine Klassifikation mit P2DHMM anzustreben, Experimente unter Verwendung künstlicher Profile ergaben dabei jedoch eine maximale Rate von lediglich 24% korrekt zugeordneten Personen. Ursache hierfür ist in erster Linie der beschriebene Rauscheffekt der zu klassifizierenden Bilder und der damit entstehenden Problematik der abgeleiteten DCT-Frequenzmerkmale. Aus diesem Grund wird für die Erkennung ein alternatives Vorgehen angewendet.

Statt der frequenzabhängigen DCT-Merkmale werden die Pixelintensitäten direkt modelliert. Durch spaltenweise Extraktion der Bildpunkte wird ein Merkmalsvektor ohne vertikale Freiheit formiert. Durch den Verlust der Dynamik in vertikaler Richtung existiert zwar nur noch die Möglichkeit einer horizontalen Variabilität, dadurch kann aber gleichzeitig der Einfluss des durch Synthese entstandenen Rauschens auf die Erkennung kompensiert werden.

Nunmehr wird eine Merkmalsequenz insgesamt durch die horizontale Spaltenabtastung über Pixelintensitäten gewonnen, welche dann über bekannte 1DHMM modelliert werden können. Insgesamt besteht eine Merkmalssequenz aufgrund der gegebenen Bilddimensionen aus 64 Spaltenvektoren mit je 64 Elementen. Bei der Abtastung wurde bewusst die horizontale Richtung gewählt, da das Rauschen durch die zeilenweise Verarbeitung hauptsächlich in vertikaler Richtung auftritt, wie aus den Beispielen in Abbildung 4.23 deutlich hervorgeht. Das Gesamtsystem kann zusammenfassend nach Diagramm 4.26 skizziert werden.

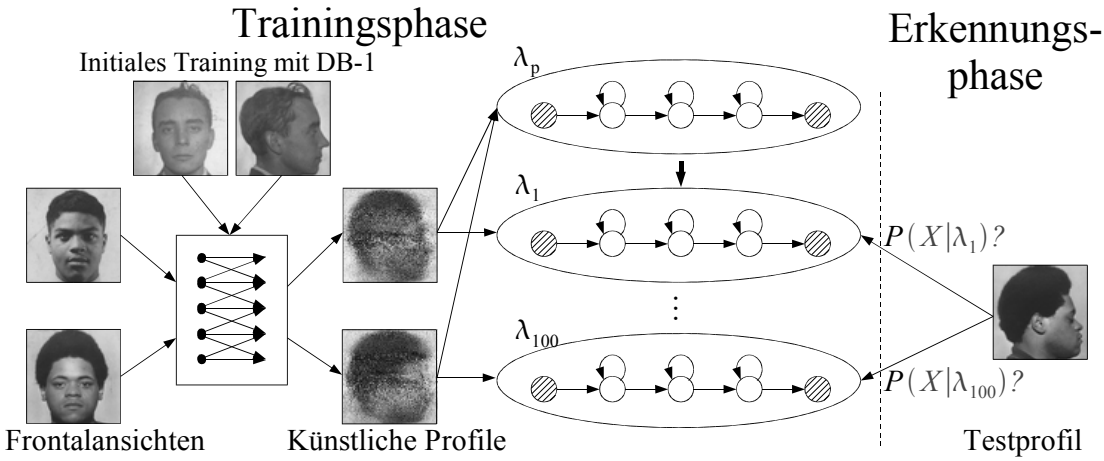


Abbildung 4.26: Gesamtsystem zur Zuordnung von Profilen zu Frontalansichten

Grundsätzlich werden bei der Nutzung des Gesamtsystems zwei Phasen unterschieden: die vorbereitende Trainingsphase, und die anschließende Erkennungsphase. Insgesamt ergibt sich der Ablauf der einzelnen Verarbeitungsschritte wie folgt:

1. Initiales Training des Rotations-MLP mit Hilfsdatenbank DB-1. Abbruchkriterium ist durch den minimalen Fehler der Validierungsdaten auf DB-3 gegeben.

2. Schätzung eines allgemeinen Profil-HMM λ_p mit sämtlichen synthetischen Profilansichten, entstanden durch das Anlegen der Frontalansichten aus Testkorpus DB-2.

3. HMM-Training mit Spezialisierung der Personenmodelle $\lambda_1 \ldots \lambda_{100}$ aus DB-2. Hierbei werden die Mittelwerte und Varianzen, nicht jedoch die Übergangswahrscheinlichkeiten neu geschätzt.

4. In der Erkennungsphase können unbekannte Profile über die maximale Produktionswahrscheinlichkeit identifiziert werden.

Mit Erkennungssystemen nach obiger Art eröffnet sich eine Vielzahl möglicher Parameterkonstellationen, wobei ein Satz mit optimaler Leistung im Vorfeld nicht vorherzusagen ist. Durch sukzessive Steigerung der Erkennungsrate über die Anpassung der einzelnen Werte können jedoch funktionale Kombinationen gefunden werden. Unter Angabe charakteristischer Parameter sind in Tabelle 4.6 ausgewählte Erkennungsergebnisse dieser Versuchsreihen aufgeführt.

MLP-Parameter			HMM-Parameter	
Eingang	Verdeckt	Typ	Zustände	Erkannt [%]
$m = 40, n = 7$	-	2D	3×3	24
$m = 40, n = 7$	-	1D	26	56
$m = 40, n = 3$	40×1	1D	22	48
$m = 34, n = 5$	40×2	1D	24	50
$m = 40, n = 3$	19×2	1D	26	60
$m = 38, n = 1$	40×2	1D	23	53

Tabelle 4.6: Übersicht der hybriden NN/HMM Profilerkennung

Zum Einen kann anhand der zwei obersten Simulationen die überlegene Erkennungsrate der 1DHMM gegenüber den P2DHMM bestätigt werden. Zum Anderen sind mit verschiedenen Architekturen Erkennungsraten von weit über 50% möglich. Das beste ermittelte System weist sogar die Anzahl von 60 richtig zugeordneten Personen auf. Eine Minderung oder Unterdrückung des überlagerten Rauschens brachte in keinem der gemachten Versuche eine Verbesserung mit sich.

4.4.5 Profilbilderkennung mit Eigenmugshots und Abstandsmaßen

Nach der Rücktransformation der geschätzten Ansichten aus dem Eigenmugshotraum, können die entstanden Bilder grundsätzlich auch mit den oben beschriebenen, statistischen Ansätzen klassifiziert werden. Da nach der Synthese von wenig verrauschten Bildern ausgegangen werden kann, würde sich hier wieder verstärkt die Verwendung der elastischeren P2DHMM anbieten.

Wie Eingangs erwähnt, hat sich zur Erkennung im Eigenfaceraum die auf klassischen Distanzen beruhende Suche nach dem nächsten Nachbarn durchgesetzt. Daher wird dieses Vorgehen auch im Kontext der Mugshoterkennungsaufgabe erprobt [Wal03b]. Hierzu bieten

sich die Euklidische Distanz und der Mahalanobis Abstand an. Die Verwendung von diskriminativen Verfahren, wie den SVM oder der Linearen Diskriminanz Analyse scheidet aufgrund der auf ein Exemplar beschränkten Datenmenge pro Klasse bereits im Vorfeld aus. Ein wesentliches Merkmal bei den verwendeten Distanzklassifikatoren ist, dass ein zusätzliches Training des Klassifikators in der Anwendungsphase entfallen kann.

Nach der Trennung in Trainings- und Testphase, mündet der Ablauf des in Schema 4.27 gezeigten Systems in die folgenden Schritte:

1. Bestimmung des Eigenmugshotraums und Projektion aller Bilder in diesen.

2. Initiales Training des Rotations-MLP mit Hilfsdatenbank DB-1 bis Fehler auf Validierungssatz minimiert ist.

3. Während der Erkennungsphase erfolgt die Zuordnung zwischen geschätzten Profilgewichten $\hat{W}_1 \ldots \hat{W}_{100}$ und den echten Gewichten der unbekannten Personen $\hat{W}$ über den geringsten Abstand zum Referenzvektor mit dem Index i.

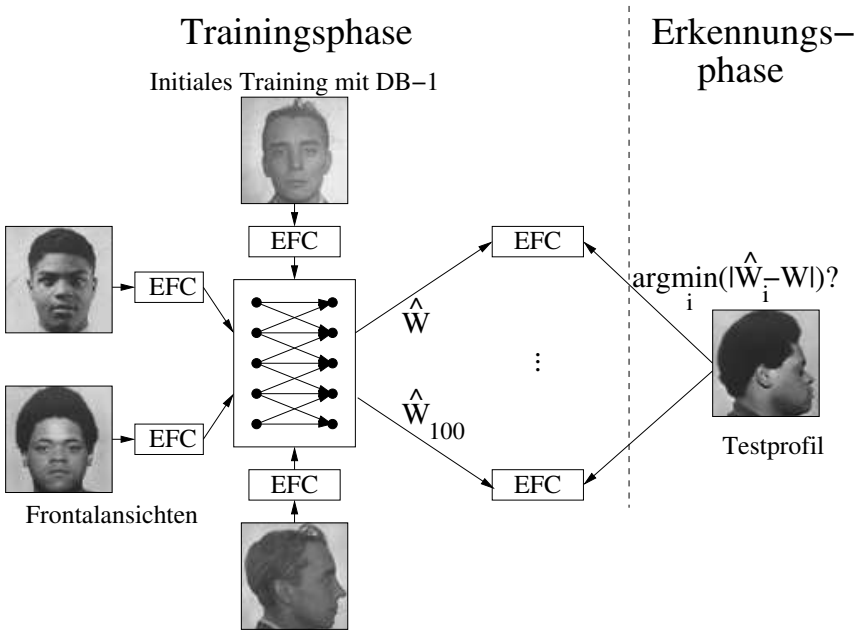


Abbildung 4.27: Überblick des hybriden Systems zur Erkennung künstlicher Profilmodelle

Zusammenfassend erweist sich die Verwendung des Mahalanobis Abstand dem Euklidischen in den Experimenten erwartungsgemäß als überlegen, wobei der Unterschied im Bereich von 2% liegt. Die in dieser Reihe gemessene maximale Erkennungsrate liegt bei 39%, gewonnen unter Verwendung von 100 Gewichtskoeffizienten und einem dreischichtigen MLP. Darüber hinaus soll neben der Erkennung über die geringste Distanz zur Steigerung der Erkennungsleistung ein MLP verwendet werden, welches im Folgenden vorgestellt wird.

4.4.6 Neuronale Profilbilderkennung mit Eigenmugshots

Unter der Voraussetzung, dass die Klassen der zu erkennenden Personen bereits im Vorfeld bekannt sind, und dass die Dimensionen aller zu klassifizierenden Muster statisch sind, kann zur Gesichtsklassifikation ebenfalls eine neuronale Struktur eingeführt werden [Pat96].

In diesem Fall werden die Koeffizienten der Hauptachsen zur Klassifikation an die Eingänge des MLP angelegt. Über eine verborgene Schicht können dann für jede Person am Ausgang zwei als Bayessche Schätzwahrscheinlichkeiten interpretierbare Werte abgerufen werden. Ein Knoten weist dabei eine positive Logik im Sinne *Ist Person*, der zweite eine negative Logik mit Bedeutung *Ist Nicht Person* auf. Wie schon bei der Detektion kann die Diskriminanz unter den zu erkennenden Personen auch hier durch die beiden komplementären Ausgaben während der Parameterschätzung zusätzlich vergrößert werden. Darüber hinaus ist von Vorteil, dass das zu optimierende Trainingskriterium, hier der minimale Trainingsfehler, bei den beiden verwendeten Verarbeitungsebenen Rotation und Erkennung identisch ist. Das Gesamtsystem lässt sich nach Abbildung 4.28 zusammenfassen.

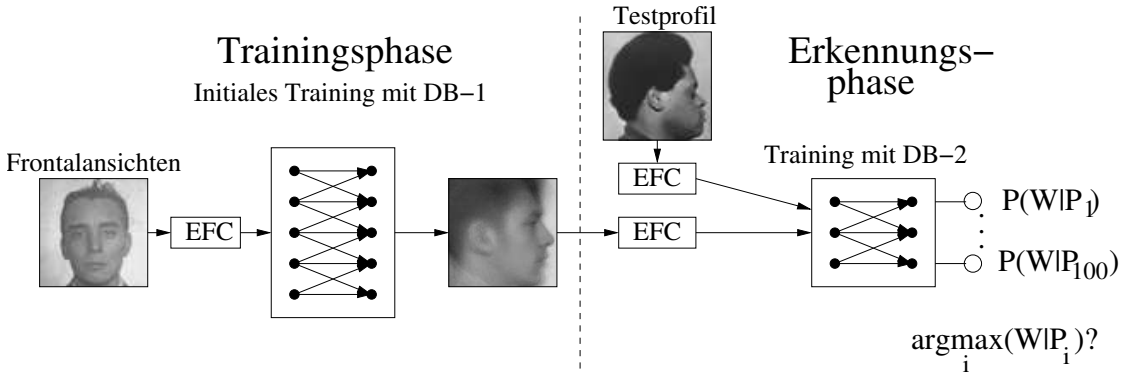


Abbildung 4.28: Kombinierte Netzstruktur zur Erkennung von Profilen

Ein kompletter Durchlauf des Trainings- und Erkennungszyklus besteht insgesamt aus vier Stufen:

1. Bestimmung des Eigenmugshotraums und Projektion aller Bilder in denselbigen.

2. Initiales Training des Rotations-MLP mit Hilfsdatenbank DB-1.

3. MLP-Training der zu erkennenden Profilbild-Projektionen aus DB-1.

4. Während der Erkennungsphase werden zunächst die geschätzten Profile zu den Frontalansichten erzeugt. Danach werden der Reihe nach alle Wahrscheinlichkeiten $P(\hat{W}_i|P_1) \ldots P(\hat{W}_i|P_{100})$ aus DB-2 berechnet. Die Erkennung erfolgt über den Index i des Neurons mit der höchsten Aktivierung.

Aufgrund der gemeinsamen Trainingsparadigmen ist es nun möglich, die beiden MLP Strukturen in eine zu überführen. Bei Vorliegen weiteren Trainingsmaterials kann die ganze Struktur nun gesamtheitlich trainiert werden. Bei einem freien Training der resultierenden

Gesamtstruktur erweist sich jedoch die Tatsache als problematisch, dass die Eigenmugshot-Koeffizienten im Inneren von ihrer ursprünglichen Bedeutung abweichen. Zur Vermeidung dieses Effektes ist es bei einem nachgeschalteten Training jedoch möglich, jeweils nur die erste oder zweite Teilstruktur anzupassen.

Zur besseren qualitativen Einordnung des vorgestellten Ansatzes werden die durch Linearkombination der Eigenmugshots zurücktransformierten Bilder mit der Erkennungsrate von 1DHMM und P2DHMM verglichen. In Tabelle 4.7 sind exemplarisch die repräsentativsten Erkennungsraten zusammengefasst.

Klassifikator	Parameter	Erkennungsrate [%]
Mahalanobis	-	39
MLP, Eigenmugshots	3 Schichten	55
1DHMM, Spalten	21 Zustände	49
P2DHMM, DCT	8×8 Zustände	30

Tabelle 4.7: Übersicht der Erkennungsergebnisse der Profilerkennung mit Eigenmugshots

Ein Vergleich der erhaltenen Ergebnisse lässt mehrere Schlüsse und Deutungen zu. Zunächst erweist sich die neuronale Klassifikation im Eigenmugshotraum den Distanzklassifikatoren mit 55% als weit überlegen. Außerdem kann durch die Verwendung stochastischer, aus Rückprojektionen gewonnen HMM keine zusätzliche Optimierung erzielt werden. Zwar ist die Erkennung über P2DHMM aufgrund der vergleichsweise glatten Erscheinungen mit 30% prinzipiell besser als bei der Erkennung der zeilenweise gewonnenen Bilder, fällt aber gegenüber den eindimensionalen HMM mit einer Rate von 49% weit zurück.

Motiviert durch die bisher beste Erkennungsleistung basierend auf 1DHMM mit vorgeschalteter Bildtransformation, wird ein integrierter NN/HMM Hybridansatz zur gemeinsamen Parameterschätzung weiterverfolgt.

4.4.7 Integrierter hybrider NN/HMM Ansatz

Ausgegangen wird bei der Integration des NN/HMM Ansatzes von dem hybriden Erkennungssystem nach Kapitel 4.4.4. Die beiden dort vorkommenden, voneinander unabhängigen Trainingsstufen basieren auf zwei unterschiedlichen Optimierungszielen. Beim MLP ist dies der kleinste Fehler und beim HMM die Optimierung der maximalen Ähnlichkeit. Mit dem Ziel der Steigerung des Informationsflusses zwischen Frontalbild und den gewonnenen Modellen soll durch die Integration beider Stufen in ein gesamtheitliches Konzept versucht werden, die beiden Trainingsphasen zu vereinheitlichen.

Implementiert werden kann die angestrebte Fusion durch die Kopplung der HMM Parameter mit den Ausgängen des MLP. Aufgrund der geringen Anzahl an Daten haben die Varianz und die Zustandsübergänge eine untergeordnete Rolle und weichen in der Regel nicht stark vom Allgemeinmodell ab, so dass die Mittelwertschätzung der Gaußverteilungen als ausreichend angesehen werden kann. In der Gebrauchsphase des MLP tritt die Generation der Profilmodelle nun direkt an die Stelle der Erzeugung synthetischer Ansichten. Hierbei

wird wieder auf die Struktur der bekannten 1DHMM zurückgegriffen. Eine resultierende Netzarchitektur zur Generierung der Mittelwertvektoren μ unter Verwendung einer verdeckten Ebene ist in Darstellung 4.29 gezeigt.

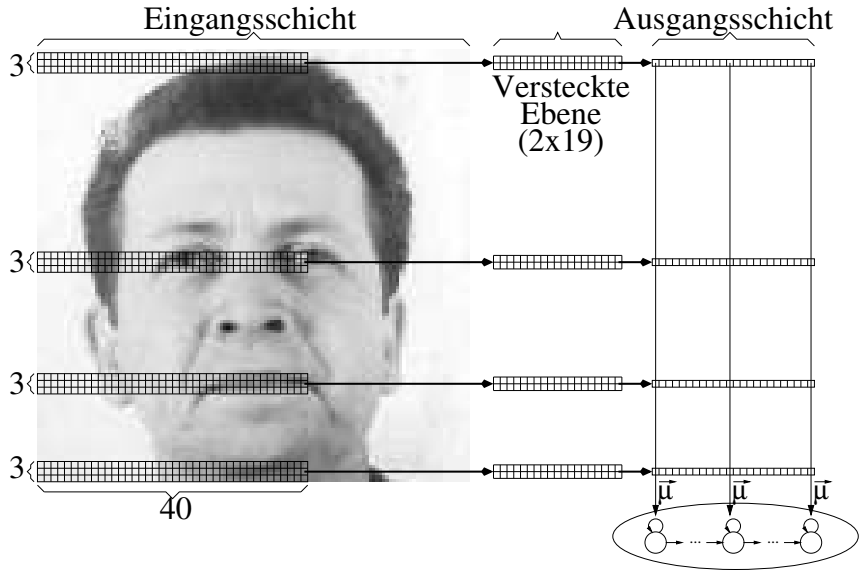


Abbildung 4.29: Netzstruktur zur Schätzung von Profilmodellmittelwerten

Bei der Abbildung werden auch hier horizontale Bereiche bestehend aus 3 Zeilen mit dem vorherzusagenden Mittelwerten des Profilmodells verknüpft. Die Größe der verborgenen Schicht beträgt beispielsweise 19×2 Neuronen, die verwendete Bildgröße 64×64 Punkte. Die Übertragung an die Ausgänge ist spaltenweise organisiert. Die Einbettung dieser Methode erfolgt nach Abbildung 4.30 über:

1. Training eines gemeinsamen Prototyps λ_p und Modellierung aller Profilansichten aus DB-1 in Form von 1DHMM.

2. Schätzung der Parameter des MLP: am Eingang liegen Frontalbilder von DB-1 an, am Ausgang die Mittelwertvektoren der zugehörigen HMM.

3. Erzeugen der Profilmodelle der Personen aus DB-1 ohne weiteres Training durch direktes Einfügen der geschätzten Werte in den gemeinsamen Prototyp.

4. Klassifikation über ML-Entscheidung.

Ein wesentliches Merkmal dieses Verfahrens ist, dass in der Anwendungsphase analog zu den einfachen Distanzklassifikatoren kein erneutes Training notwendig wird. Da auch bei den hier vorgestellten Strukturen keine optimale Parameterkonstellation vorhersagbar ist, wurden mehrere Variationen zur Parameterfindung durchlaufen. Die günstigste sich ergebende Kombination besteht wie im obigen Beispiel aus insgesamt drei Ebenen mit einer verdeckten Schicht von 19×2 Einheiten und weist eine Erkennungsrate von 49% auf.

Offensichtlich beschreiben die vom NN auf Basis des Satzes DB-1 geschätzten Mittelwertvektoren die Personen in DB-2 nicht ausreichend. Ein möglicher Grund hierfür ist das

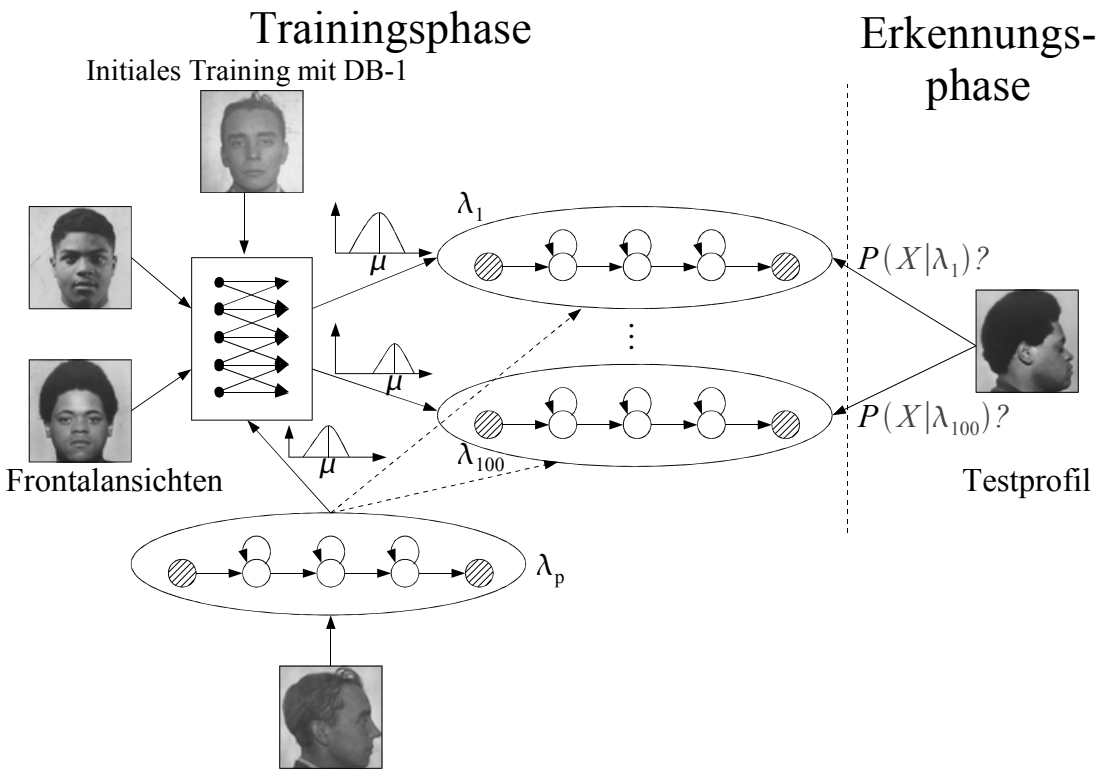


Abbildung 4.30: Hybrides Systems zur direkten Schätzung von Profilmodellen

Fehlen einer personenspezifischen Anpassung von Varianz und Übergangswahrscheinlichkeit. Die zu geringe Ähnlichkeit der Personen in beiden Datensätzen kann hierfür aber auch verantwortlich sein. Für weiterführende Experimente mit dem vorgestellten System sollte eine drastische Vergrößerung der Hilfsdatenbasis erwägt werden.

4.4.8 Zusammenfassung der Profilerkennung

Die repräsentativen Systemleistungen der verschiedenen Ansätze werden abschließend in Tabelle 4.8 zusammengefasst und diskutiert.

Methode	Erkannt [%]	Bemerkung
P2DHMM, DCT	10	Direkte Verwendung frontaler HMM
P2DHMM, DCT	24	Verwendung modifizierter HMM
MLP, Eigenmugshots	55	Klassifikation mit NN
1DHMM, Intensitäten	60	Bildbasierte Synthese
1DHMM, Mittelwertvektoren	49	Direkte Schätzung der Profilmodelle

Tabelle 4.8: Ergebniszusammenfassung der Profilerkennung

Zunächst kann festgehalten werden, dass der Einsatz der vorgestellten Maßnahmen zur Erkennung von Profilen aufgrund der unterlegenen Basisleistung frontaler P2DHMM ge-

rechtfertig ist. Insgesamt ist durch die Transformationen eine Steigerung der Rate von 10% auf 60% richtig erkannte Personen möglich geworden.

Durch die Einführung angepasster Frontalmodelle ist bereits eine Steigerung von 10% auf 24% Prozent möglich geworden. Durch den Einsatz der eingeführten Eigenmugshots in Kombination mit einer neuronalen Klassifikation konnte bereits eine Güte von 55% erreicht werden. Obwohl durch die Synthese erhebliche Artefakte im Bild entstanden sind, bietet der vorgestellte Ansatz mit 1DHMM auf Basis zeilenweise geschätzter Profilansichten die beste Erkennungsleistung mit 60%. Die erwünschte Verbesserung durch die Verschmelzung der Rotiererstufe mit dem Modelltraining über eine direkte Schätzung der Profilmodelle blieb aus. In weiterführenden Arbeiten bietet sich zur Optimierung an dieser Stelle der Einsatz von RBF-NN an.

Das erreichte Ergebnis kann aufgrund der einleitend genannten Problematik jedoch bereits als beachtlicher Erfolg gewertet werden, da von einer anzustrebenden Rate von ca. 72% auszugehen ist. Da auf Basis der verwendeten Datenbank keine weiteren Publikationen bekannt sind, ist ein direkter Vergleich mit anderen Arbeiten zu diesem Thema nicht möglich. Durch Bildsynthese mit dreidimensional gewonnenen Kopfmodellen und Texturen wird in einem ähnlichen Szenario auf anderen Daten jedoch ebenfalls nur eine maximale Zuordnungsrate von 65% erreicht [Vet98]. Durch eine Merkmalstransformation im Zusammenhang des Bunch Graph Matching werden bei der Zuordnung von Halbprofilen sogar nur ca. 50% der Personen richtig zugeordnet [Mau95].

In Evaluationen der beiden besten Systeme zur frontalen Gesichtserkennung im Face Recognition Vendor Test 2000[11] fällt die Erkennungsrate bei einer Kopfdrehung von $45°$ bereits auf 70 bis 80 Prozent [Bla05], bei Drehungen darüber hinaus sinkt sie sogar in den einstelligen Bereich [Gro01].

Zur Aufhebung der Beschränkung auf Profilansichten kann auf Basis der vorgestellten Verfahren ein Übergang zu einer blickwinkeltoleranten Erkennung angestrebt werden. Ein möglicher Ansatz besteht beispielsweise in der Kaskadierung von Rotationsoperatoren mit kleineren Abweichungen bezüglich der Betrachtungswinkel, z.B. $10°$. Da bei diesen geringeren Winkeländerungen von schwach bis ausbleibenden Verdeckungen auszugehen ist, kann eine weitaus bessere Bildqualität erwartet werden.

[11] Ein unabhängiger Leistungsvergleich für Gesichtserkennungssysteme verschiedener Hersteller

Kapitel 5

Dynamische Mimikerkennung

Neben den bisher behandelten Problemkreisen, nämlich der Findung von beliebigen Gesichtern vor variablen Hintergründen und der Erkennung bzw. Zuordnung von bekannten Gesichtern in definierten Szenarien, beschäftigt sich die vorliegende Aufgabenstellung mit der Erkennung personenunabhängiger Gesichtsausdrücke in Bildsequenzen. Hiermit wird in ersten Ansätzen versucht, das enorme Potential der nicht-verbalen Kommunikation für computergestützte Anwendungen zu nutzen [Dar72]. So stellt beispielsweise die vollautomatische Mimik-Transkription von Spielfilmen ein interessantes praxisnahes Anwendungsszenario dar. Aufgrund der Problematik einer Erkennung beliebig langer Mustersequenzen zeigt es sich in einem ersten Schritt als günstig, die einzelnen, kurzzeitig auftretenden Gesichtsausdrücke zunächst zeitlich vorzusegmentieren. Diese isolierten Bereiche können danach in einem zweiten Schritt klassifiziert werden.

Obwohl das menschliche Gesicht in der Lage ist, mehrere tausend Ausdrücke zu erzeugen, wird in einer Arbeit von Ekman und Friesen mit bedeutender Rolle im Rahmen der Ausdruckspsychologie gezeigt, dass allen mimischen Ausdrücken prinzipiell nur sechs verschiedene Basisemotionen zugrunde liegen [Ekm71]. Bei der Observation des menschlichen Mienenspiels kommt danach aus psychologischer Sicht neben dem neutralen Zustand eine der Emotionen wie Angst, Ärger, Ekel, Freude, Trauer oder Überraschung als Ursache in Betracht, welche als Kultur übergreifend interpretierbar disputiert werden. Im Rahmen der Definition von MPEG4-Metadaten wurde das in Abbildung 5.1 zusammengefasste Inventar zudem standardisiert. Bezüglich Fälschungssicherheit und Willkürlichkeit der ausgedrückten Mimik wird neben anderen emotionalen Erscheinungsformen angenommen, dass der artikulierte Inhalt im Regelfall stärker als die Stimme kontrolliert wird, gefolgt von der Gesichtsbewegung und zuletzt der Körperbewegung [Ekm74]. Daher können neben der emotionalen Ursache bis zu 70% der sichtbaren Mimik auch nicht emotionalen Ursprungs sein.

Ansätze zur Erkennung von Gesichtsausdrücken können prinzipiell sowohl in *statische* als auch *dynamische* Verfahren unterteilt werden. Im weiteren Verlauf der Arbeit werden Verfahren zur dynamischen Erkennung der oben genannten, in der Regel kurzzeitig auftretenden Basisemotionen vorgestellt und anhand zweier möglichst repräsentativer Datenbanken getestet und qualitativ bewertet. Ein wesentlicher Grund für das Verwenden dynamischer Methoden liegt zum einen in der Tatsache begründet, dass stochastisch basierte Ansätze anhand von variierenden Trainingsdaten die jeweils wesentlichen Eigenschaften der Beispiele über die

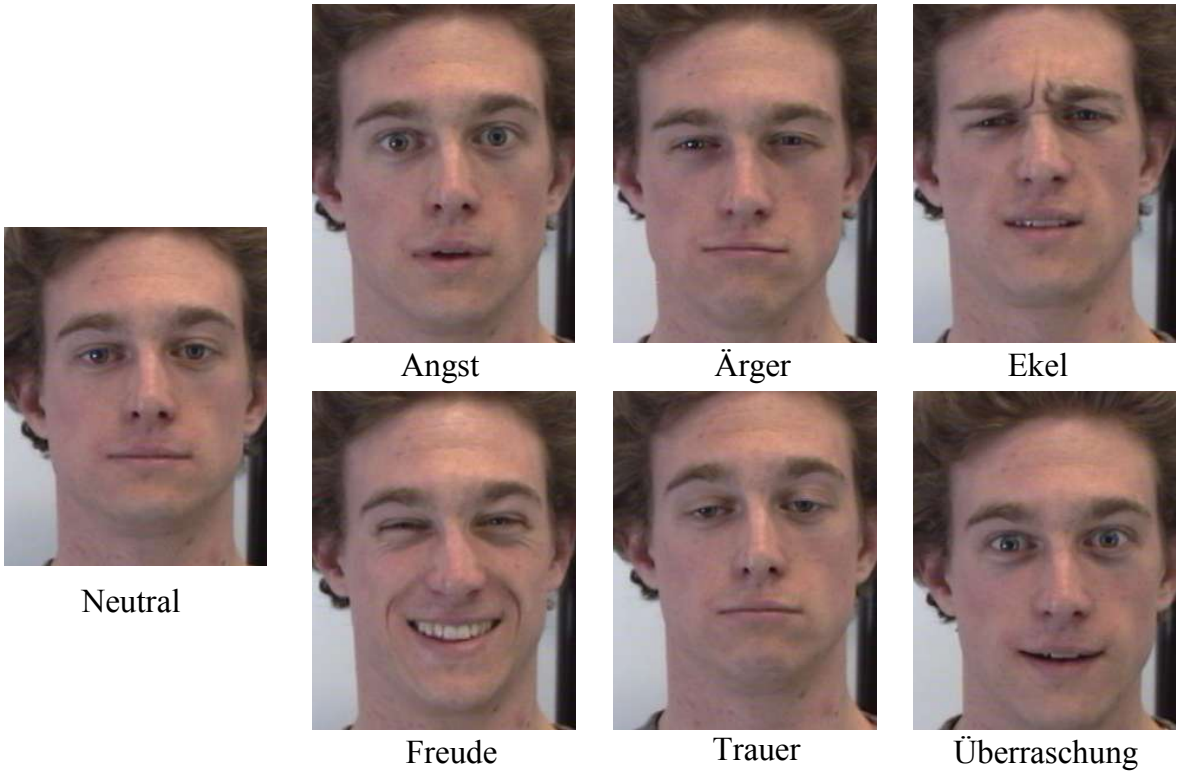


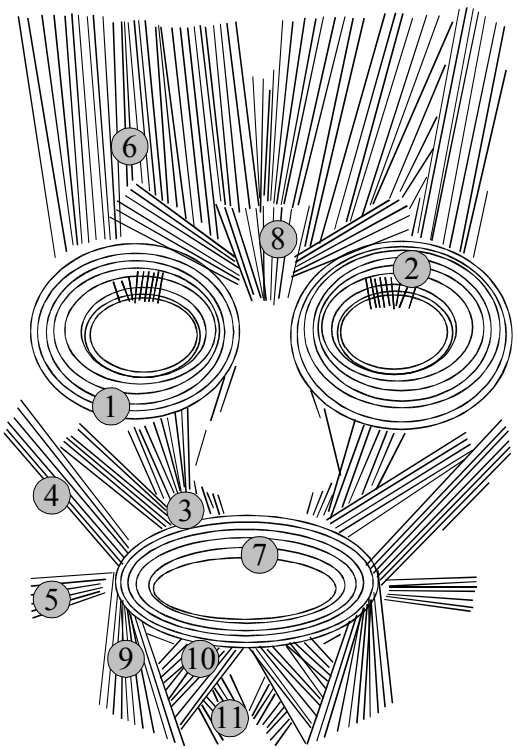
Abbildung 5.1: Beispiele der ausgeprägten Basisemotionen

Änderung selbst erlernen können. Eine aufwändige Vorgabe oder Suche der Gesichtssegmente über Stützpunkte und deren Abstände kann somit entfallen. Ein zweiter Vorteil liegt in der Möglichkeit den zu untersuchenden Gesichtsbereich ganzheitlich aufgrund seiner Dynamik zu betrachten, was an die physiologisch orientierte Klassifikation zeitlich ablaufender emotionaler Ausdrücke angelehnt ist. Eine ausführliche Diskussion alternativer Ansätze kann der Literatur entnommen werden [Pan03].

5.1 Biologische Hintergründe zur Mimikentstehung

Ohne auf die exakten psychologischen Hintergründe der Mimikentstehung einzugehen soll vereinfachend angenommen werden, dass die zugrunde liegende Emotion über die Bewegung der Muskulatur im Gesichtsbereich gemessen werden kann. Die Bewegungen einzelner Gesichtsmuskeln und deren Kombination rufen beobachtbare Veränderungen in der mimischen Erscheinung hervor. Unterschwellig dient die Anatomie der Mimik erzeugenden Gesichtsmuskulatur als Grundlage für eine geeignete Modellbildung. Obwohl im Gesichtsbereich insgesamt 26 Muskeln vorhanden sind, beteiligen sich nur elf an der Erzeugung der Mimik, welche in Übersicht 5.2 gezeigt sind.

Auf Grundlage dieser Gesichtsmuskeln wurde das im Bereich der Psychologie verbreitete *Facial Action Coding System* (FACS) von Ekman und Friesen entwickelt, welches sich als ein zuverlässiges und objektives Instrumentarium zur Erfassung des mimischen Ausdrucks



1. Augenringmuskeln
2. Levator palpebrae
3. Levator labii superioris
4. Großer Jochbeinmuskel
5. Risorius/Platysma
6. Stirnmuskel
7. Mundringmuskel
8. Augenbrauenrunzler
9. Depressor angulioris
10. Depressor labii inferioris
11. Mentalis

Abbildung 5.2: Die wichtigsten Gesichtsmuskeln [Fai90]

erwiesen hat [Ekm78]. Bei der Beschreibung der mimischen Veränderung sind die folgenden vier Kennwerte von Bedeutung:

1. *Qualität*: Identifikation der beobachtbaren Gesichtsbewegungen.
2. *Intensität*: Die Ausprägung der Bewegung.
3. *Lateralität*: Unterschiedliche Ausprägungsform auf den beiden Gesichtshälften.
4. *Zeit*: Lokalisierung der stärksten Ausprägung.

Unter Voraussetzung sich ändernder mimischer Aktivitäten sollen dynamische Modelle zur Beschreibung der Mimik und somit der Emotion trainiert werden. Aus Gründen der Beschränktheit des im Besonderen *frei erhältlichen* Trainings- und Testmaterials mussten vor Beginn der Arbeiten zunächst entsprechende Daten zur dynamischen Emotionserkennung akquiriert werden.

5.2 Datenbanken zur dynamischen Emotionserkennung

Für die Vorversuche werden in einer ersten Datenbank vier verschiedene Basisemotionen aufgezeichnet. Darin wiederholen sechs Personen auf künstliche und karikierte Art und Weise jeweils viermal die ausgewählten Emotionen Ärger, Ekel, Freude und Überraschung. Die Gesichtsausdrücke beginnen mit einem neutralen Ausdruck und enden in der stärksten Ausprägung, dem so genannten *Apex*. Freie Kopfbewegungen sind bei diesen Aufnahmen nicht

zugelassen. Zur Elimination von äußeren Einflüssen wird eine gleichmäßige, diffuse Beleuchtung und ein schwarzer Hintergrund gewählt. Die Sequenzen werden mit einer Rate von 25 Bildern pro Sekunde bei einer Größe von 320×240 Pixeln als unkomprimierte Grauwertbilder aufgezeichnet.

Zur Verallgemeinerung der Versuche wurde eine zweite Datenbank mit allen sieben Basisemotionen vor einem beliebigen Hintergrund erstellt. Die im Rahmen des EU-Projekts FGNET verfügbar gemachte Datenbank stieß auf große Resonanz und wurde bereits über siebzig mal angefordert, aufgrund der kurzen Verfügbarkeit wurden bisher jedoch noch keine Ergebnisse auf dem Material publiziert [Wal05]. Der erstellte Korpus umfasst 18 Versuchspersonen, die bei der Darbietung aller Ausdrücke jeweils dreimal aufgezeichnet wurden. Zur Elizitation[1] werden den Probanden kurze Stimulivideos auf einem Monitor gezeigt, über welchem die aufzeichnende Kamera angebracht ist. Durch die Wahl eines solchen Versuchaufbaus kann eine frontale Gesichtserfassung gewährleistet werden, siehe Abbildung 5.1. Die Aufzeichnungsrate beträgt auch hier 25 Hertz, wobei ebenfalls unkomprimierte Bilder mit einer Dimension 640×480 gespeichert werden.

Bei einem entspannten Ausdruck beginnend werden den Versuchspersonen Stimulivideos gezeigt, welche den gewünschten Gefühlszustand hervorrufen sollen. Die mimische Aktivität der zugehörigen Basisemotionen vollzieht sich idealtypisch vom Neutralzustand ausgehend bis hin zur maximalen Intensität. Da dieses Vorgehen jedoch nicht bei allen Klassen angewendet werden kann, wird alternativ versucht den inneren Gefühlszustand durch Nachempfinden zu erzeugen, was im Gegensatz zum Schauspielen zu weniger künstlichen Aufnahmen führt. Durch das parallele Abspielen der Auslöservideos während der Aufnahme erfolgt bei den meisten Sequenzen bereits im Vorfeld eine temporale Segmentierung. Bei den Basisemotionen konnte während der Aufzeichnungen folgendes festgestellt werden:

- **Freude:** Bei dieser, am einfachsten zu erzeugenden Emotion zeigt sich, dass ein kurzer lustiger Filmausschnitt bereits vortrefflich in der Lage ist, einen ausgeprägten heiteren Gemütszustand zu erzeugen.

- **Überraschung:** Der Zustand einer realistischen Überraschung hingegen ist schwer zu erzeugen. Daher war es Aufgabe der Probanden, eine überraschende Szene innerlich nachzuempfinden, wodurch die Natürlichkeit größtenteils erhalten bleibt.

- **Ärger:** Durch Hineinversetzen in eine ärgerliche Situation können ebenfalls wieder nahezu natürliche Daten mit verärgerten Gesichtsausdrücken gesammelt werden.

- **Angst:** Das Elizitieren der Emotion Angst hat sich während der Aufnahmen als schwierig erwiesen. Daher werden hier längere, Spannung aufbauende Filmszenen gezeigt, bei denen die Zuschauer im Moment der Spannungsentladung durch den ausgelösten Schreck für einen kurzen Moment Angst zeigen.

- **Traurigkeit:** Die schwierigste zu erzeugende Emotion ist Trauer. Durch das Einspielen längerer dramatischer Filmszenen wurde auch hier versucht die Videodaten durch Nachempfinden zu erhalten.

[1] Kontrollierte Erhebung spontaner Gefühlsäußerungen

- **Ekel:** Zum Hervorrufen dieser leicht zu erzeugenden Reaktion wird ebenfalls wieder ein kurzer Videoausschnitt gezeigt, beispielsweise mit arg verdorbenen Lebensmitteln.

- **Neutral:** Die Aufnahmen für den neutralen Emotionszustand mit entspannter Gesichtsmuskulatur werden während der Darbietung des Einführungsvideos aufgenommen.

Nach der Beurteilung des Materials durch mehrere befragte Personen und Experten, können die Emotionen Ärger, Ekel, Freude und Neutral in erster Näherung als spontan bzw. authentisch angesehen werden. Durch die Art der Erzeugung basieren die verbleibenden Gemütszustände Angst, Traurigkeit und Überraschung stark auf den schauspielerischen Leistungen der Probanden und zeigen starke Unterschiede bezüglich Qualität und Intensität. Die Dauer zwischen Grundzustand und maximaler Ausprägung lag bei den meisten Versuchspersonen und Emotionen bei etwa $2/3s$.

Wie schon in den Arbeiten von Ekman, haben sich die Emotionen Angst und Trauer bezüglich ihrer Eindeutigkeit als problematisch herauskristallisiert [Ekm78]. Dies ist wahrscheinlich auf die Schwierigkeiten einer spontanen Emotionserzeugung selbst zurückzuführen. Bedingt durch die personenabhängigen mimischen Ausprägungen, scheint zum Anderen keine eindeutige Zuordnung dieser Gemütszustände möglich zu sein.

Zur Beurteilung der Authentizität der gesammelten Aufnahmen und zur Verifikation der Eindeutigkeit soll die Erkennungsleistung des Menschen für die gesammelten Daten ermittelt werden. Dazu wurden jeweils dreißig zufällig ausgewählte Sequenzen aller sieben Emotionen von zwanzig unabhängigen, nicht auf die Problematik spezialisierten Personen beurteilt. Aufgrund der bekanntermaßen hohen, evolutionär hervorgebrachten Erkennungsleistung des Menschen soll das ermittelte Ergebnis als Referenz für die automatischen Erkennungssysteme herangezogen werden. Die Probanden hatten ein Alter zwischen 23 und 38 Jahren, deren Verteilung bestand in etwa je zur Hälfte aus Männern und Frauen. Die Auswertung dieses Perzeptionstests auf den vorliegenden Daten ergibt eine mittlere Erkennungsleistung von 61%. Das beste Einzelergebnis lag bei 93%, das Schlechteste bei 38%. In der Personengruppe mit Versuchsbeteiligten, wurde im Mittel eine entsprechend höhere Perzeptionsleistung von etwa 80% gemessen. Von ähnlichen Beobachtungen wurde auch im Kontext einer anderen Arbeit mit einer vergleichbaren Problemstellung berichtet [Mic03].

5.3 Modellierung mit globalen Bewegungsmerkmalen

Zunächst wird versucht die Gesamtheit der Muskelbewegung, respektive deren Auswirkungen, von einem Bild auf ein folgendes durch Differenzbilder zu approximieren, wie in Abbildung 5.3 gezeigt. Unter der Voraussetzung, dass die zu messende Bewegung innerhalb der zu erkennenden Sequenz nur durch die Kontraktion der Muskulatur und nicht durch die Bewegung des Gesichts selbst entsteht, soll die Dynamik der Mimik durch sieben repräsentative Momente in einem Merkmalsvektor zusammengefasst werden. Ein wesentlicher Vorteil bei diesem Verfahren liegt im Entfallen einer aufwändigen Suche von gesichtsspezifischen Stützpunkten bzw. den Labeln nach Anhang A.1 [Zha98, Bar03]. Diese Form der Modellierung zeitlich dynamischer Muster mit den in Kapitel 2.7 vorgestellten HMM hat

sich bereits im Rahmen der Gesten- und Personenaktivitätserkennung als äußerst effektiv erwiesen [Eic98b, Wal04a]. Im Gegensatz zu Verfahren mit optischem Fluss ist die schnelle Berechenbarkeit von Differenzbildern ein wesentlicher Vorteil dieses Ansatzes.

Abbildung 5.3: Interpretation des Differenzbilds als Muskelbewegung.

Die Muskelbewegung soll durch das Differenzbild I'_d zweier aufeinander folgender Bilder durch pixelweise Subtraktion repräsentiert werden.

$$I'_d(x,y,k) = I(x,y,k) - I(x,y,k+1) \tag{5.1}$$

Zur Elimination von Rauschen und sonstigen Artefakten, wird zusätzlich ein Schwellwertmechanismus vorgeschaltet:

$$I_d(x,y,k) = \begin{cases} 0 & |I'_d(x,y,k)| < T \\ I'_d(x,y,k) & \text{sonst} \end{cases} \tag{5.2}$$

Mit dem Ziel der Entkopplung der reinen Erkennungsleistung der Mimikerkennung von der eventuell vorgeschalteten Gesichtsdetektion, wird im Verlauf der Experimente nur von ideal lokalisierten Gesichtsbereichen Gebrauch gemacht. Dabei werden die in Abbildung 5.1 gezeigten Ausschnitte verwendet. Durch den Einsatz von Differenzbildern wirkt sich ein beliebiger, aber stationärer Hintergrund nicht auf den Inhalt der Merkmale aus. Die resultierende Verteilung im Bildbereich R_i soll als Bewegungsintensität in $x-$ und $y-$Richtung interpretiert werden. Die ersten beiden Merkmale zur holistischen Beschreibung der Mimik ergeben sich durch die Massenschwerpunkte $m_x(k)$ und $m_y(k)$:

$$m_x(k) = \frac{\sum\limits_{(x,y)\in R_i} x \cdot |I_d(x,y,k)|}{\sum\limits_{(x,y)\in R_i} |I_d(x,y,k)|} \tag{5.3}$$

$$m_y(k) = \frac{\sum\limits_{(x,y)\in R_i} y \cdot |I_d(x,y,k)|}{\sum\limits_{(x,y)\in R_i} |I_d(x,y,k)|} \tag{5.4}$$

Um die Unabhängigkeit der Merkmale gegenüber der Gesichtsposition innerhalb des untersuchten Ausschnitts zu gewährleisten, werden zentrierte Gesichtsbereiche verwendet. Zur

weiteren Beschreibung der Mimik werden die zeitlichen Änderungen der Schwerpunkte, $\Delta m_x(k)$ in $x-$ und $\Delta m_x(k)$ in $y-$Richtung zum Merkmalvektor hinzugefügt:

$$\Delta m_x(k) = m_x(k) - m_x(k-1) \quad ; \text{ab } k > 0, \text{ sonst } 0 \tag{5.5}$$

$$\Delta m_y(k) = m_y(k) - m_y(k-1) \quad ; \text{ab } k > 0, \text{ sonst } 0 \tag{5.6}$$

Die Ausgeprägtheit einer Bewegung wird durch die relative Standardabweichung gegenüber dem Bewegungsschwerpunkt beschrieben, wodurch sich die beiden Merkmale $\sigma_x(k)$ und $\sigma_y(k)$ ergeben. Mit diesen beiden Werten kann unterschieden werden, ob sich größere oder eher kleinere Partien der Gesichtsmuskulatur bewegen:

$$\sigma_x(k) = \frac{\sum\limits_{(x,y)\in R_i} |I_d(x,y,k)| \cdot (x - m_x(k))}{\sum\limits_{(x,y)\in R_i} |I_d(x,y,k)|} \tag{5.7}$$

$$\sigma_y(k) = \frac{\sum\limits_{(x,y)\in R_i} |I_d(x,y,k)| \cdot (y - m_y(k))}{\sum\limits_{(x,y)\in R_i} |I_d(x,y,k)|} \tag{5.8}$$

Mit dem letzten Merkmal wird die Gesamtheit der Intensität $i(k)$ der Bewegung erfasst, welche durch den Mittelwert der Intensitäten berechnet wird. Große Werte repräsentieren somit intensive Bewegungen, kleine einen stationären Zustand.

$$i(k) = \frac{\sum\limits_{(x,y)\in R_i} |I_d(x,y,k)|}{\sum\limits_{(x,y)\in R_i} 1}. \tag{5.9}$$

Durch die ganzheitliche Beschreibung der Muskelaktivitäten mit den obigen Merkmalen kann der hochdimensionale Mimikbereich unter Beibehaltung seiner Dynamik auf einen siebendimensionalen Vektor reduziert werden, was in Abbildung 5.4 visualisiert ist.

$$\mathbf{x}_k = [m_x, m_y, \Delta m_x, \Delta m_y, \sigma_x, \sigma_y, i]^T, \tag{5.10}$$

Der gesamte eindimensionale Musterverlauf $\mathbf{X}_n = [\mathbf{x}_1, ..., \mathbf{x}_n]$ entsteht durch die Aneinanderreihung der Merkmale $\mathbf{x}(k)$ aller Bilder einer Sequenz und kann auf unterschiedliche Weise klassifiziert werden:

1. Zum einen ist die Modellierung mit linearen 1DHMM möglich. Der Vorteil der Verwendung dieser Modelle ist die Beibehaltung der dynamischen Invarianz gegenüber zeitlichen Verzerrungen der Observationen.

2. Alternativ ist eine Klassifikation mit Hilfe von Mehrklassen-SVM denkbar. Da mit SVM jedoch nur Daten mit statischer Dauer, respektive Vektordimension verarbeitet werden können, müssen die Muster hierzu in einem vorgeschalteten Prozess auf eine einheitliche Länge gebracht werden. Diese Längennormalisierung kann beispielsweise derart geschehen, dass ausgehend vom ersten Bild einer messbaren Bewegung nur eine bestimmte Anzahl an Folgebildern berücksichtigt wird. In der Praxis hat sich gezeigt, dass ausgehend vom Neutralzustand nach etwa 16 Bildern eine Mimik nahezu maximal ausgeprägt ist.

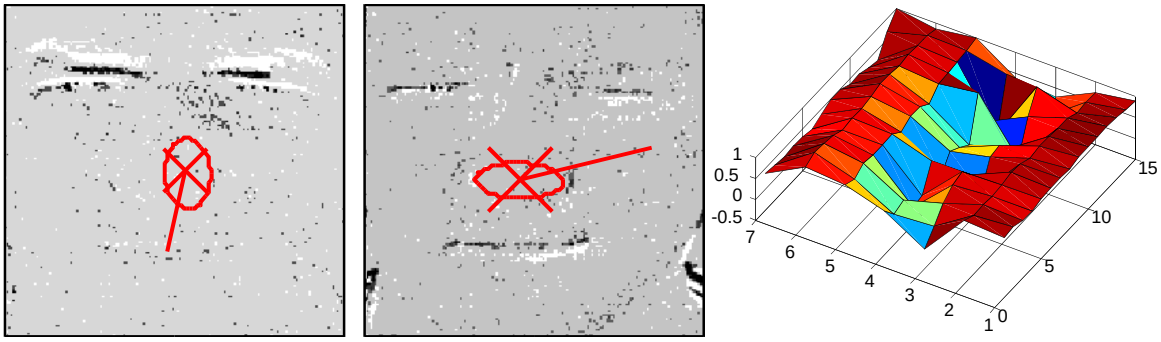


Abbildung 5.4: Visualisierung der Merkmale links: der Bewegungsschwerpunkt als Kreuz, die Varianz als Ellipse, die Abweichungen als Linie und die Intensität als Hintergrundton. Rechts: Zeitlicher Verlauf als Oberflächendiagramm

5.4 Differenzbildbasierte DCT

Nach der holistischen Bewegungsbeschreibung wird im zweiten Ansatz eine detaillierte Ortsauflösung der Bewegungsmodellierung angestrebt. Dies kann unter Anwendung des oben beschriebenen Vorgehens über eine Unterteilung des Bildbereichs erfolgen, zunächst soll jedoch eine spatio-temporale Abstraktion zur Beschreibung etabliert werden. Dazu werden die Differenzbilder nach Gleichung 5.2 einer zweidimensionalen DCT unterzogen. Hiermit ist es zum einen möglich die Dynamik einzelner Bereiche bzw. Gesichtspartien getrennt voneinander zu berücksichtigen, zum anderen kann die planare Elastizität der Mimik durch die blockweise Abtastung zusätzlich repräsentiert werden.

Zur Modellierung der zweidimensionalen Bildserien werden die im Kontext der Gestenerkennung bereits erfolgreich applizierten P3DHMM eingesetzt [Yal00, Mul02b], welche in Abschnitt 2.7.5 eingeführt wurden. Wie in Abbildung 5.5 illustriert, wird die äquivalente dreidimensionale Merkmalsequenz durch das Einbringen einer zusätzlichen Markierung am Bildanfang generiert.

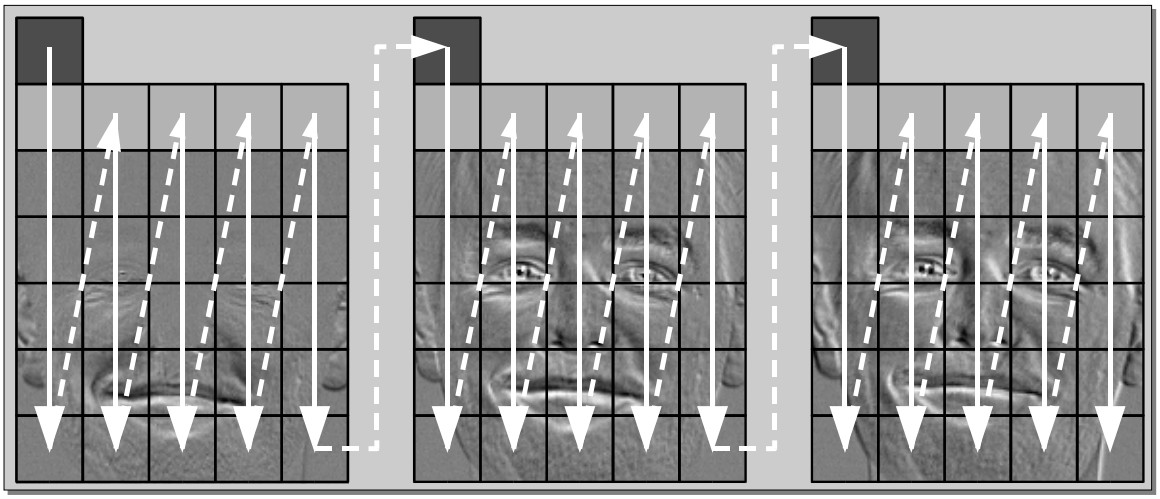


Abbildung 5.5: P3DHMM konforme Merkmalsequenz auf Differenzbildern basierend

Aufgrund der hohen Zustandsanzahl und der damit vorkommenden Rücksprungsmöglichkeiten, werden die Bildsequenzen zur Beschleunigung des komplexen Trainingsprozesses zunächst in Form von unabhängigen hintereinander geschalteten P2DHMM repräsentiert [Hul01]. Dazu werden die zu modellierenden Bildsequenzen in mehrere gleichlange Segmente unterteilt. Deren Anzahl ist durch die Menge an übergeordneten Modellzuständen gegeben. Die untergeordneten P2DHMM werden dann über diese Teilsequenzen initialisiert. Im nächsten Schritt werden die Modelle in ein P3DHMM eingebettet, welches anschließend unter Nutzung der kompletten, zusammenhängenden Sequenz nachtrainiert wird. Die hohe Rechenzeit ergibt sich hierbei hauptsächlich durch die Komplexität des während des Trainings verwendeten Forward-Backward Algorithmus, welcher proportional zum Ausdruck $N^2 \cdot T$ ist. Die Größe N wird durch die Anzahl der HMM-Zustände, und T durch die Anzahl der Vektoren in den Observationssequenzen bestimmt. Durch die vorgestellte Beschleunigung kann die Trainingszeit auf diese Weise in etwa halbiert werden.

5.5 Automatische Segmentierung mit dem Bayesian Information Criterion

Vor dem Training bzw. der Erkennung unbekannter Mimiksequenzen ist zunächst eine zeitliche Zerlegung[2] der Videodaten sinnvoll. Für dieses Anliegen wird ein effizienter Ansatz zur Audiosegmentierung nach Tritschler und Gopinath, das so genannte *Bayesian Information Criterion* (BIC) [Tri99], erweitert. Dieses Verfahren hat sich bereits im Zusammenhang mit globalen Bewegungsmerkmalen bei der Erkennung von Personenverhalten als funktionell erwiesen [Zob03, Wal04a, Zob04]. Bei diesem Verfahren ist hervorzuheben, dass die semantischen Inhalte der Bildfolgen im Vorfeld nicht bekannt sein müssen.

Zur Findung der Grenze zwischen mimischen Ausdrücken wird ein am Sequenzanfang beginnendes Abtastfenster mit der Länge n bezüglich einer semantischen Änderung untersucht. Genauer wird geprüft, ob es wahrscheinlicher ist, dass die Teilsequenz von einer Quelle Φ_1 respektive zwei unterschiedlichen Quellen Φ_{21} und Φ_{22} stammt. Das zu untersuchende Fenster wird in zwei Bereiche geteilt, welche für zwei verschiedene Ursachen stehen. Innerhalb der zum Zeitpunkt s beginnenden Sequenz $e_s, \ldots, e_{s+n}$ in Grafik 5.6 entstehen durch die Aufspaltung an der Stelle i die Teilfenster $e_s, \ldots, e_i$ und $e_{i+1}, \ldots, e_n$. Die kürzeste Fensterlänge beträgt 4 Einzelbilder, so dass gilt $i \in s+4, \ldots, s+n-4$. Dieser Vorgang wird solange wiederholt, bis entweder eine Grenze gefunden, oder die letzte mögliche Position erreicht wurde. Bei einem negativen Ergebnis wird das Abtastfenster vergrößert und die Suche solange neu initiiert, bis eine Grenze gefunden wurde. Anschließend wird der Anfang des neuen Suchfensters an das Ende des alten geschoben, und die Suche bis zum Ende der Bildsequenz fortgesetzt. Obwohl die Modellierung der zunächst unbekannten Inhalte prinzipiell über beliebig komplexe Mechanismen erfolgen kann, werden die verschiedenen Segmente aus Effizienzgründen durch die Kovarianzmatrizen Σ der Norm $\mathbf{e}_s$ der siebendimensionalen globalen Bewegungsmerkmale gebildet. Durch die Wahl dieser Modellierungsmethode kann

[2] Auch *Shot Boundary Detection* genannt.

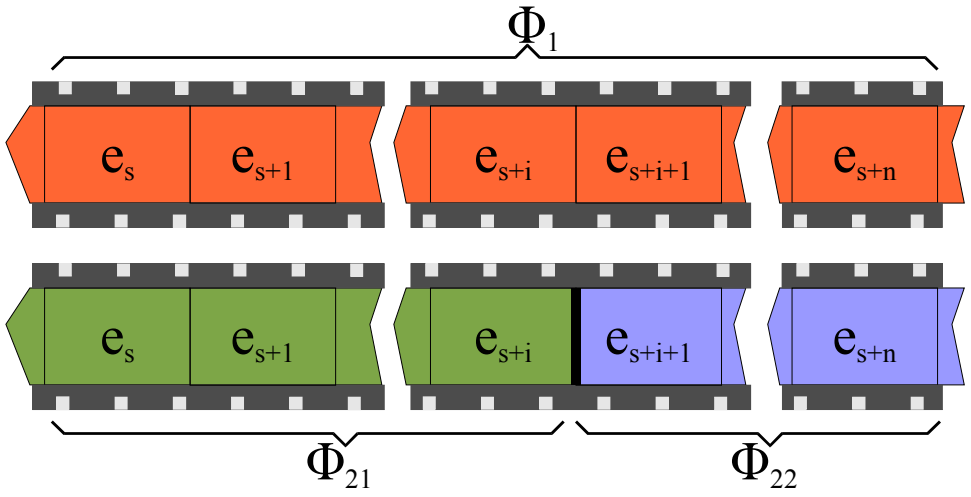


Abbildung 5.6: Schematische Darstellung des BIC

neben der angestrebten Robustheit eine zügige Abarbeitung gewährleistet werden. Ob an der Position i ein inhaltlicher Wechsel stattfindet, kann formal über einen negativen ΔBIC-Wert nach Gleichung 5.11 entschieden werden.

$$\Delta\text{BIC}_i \overset{!}{<} 0 \quad \text{wobei } \Delta\text{BIC}_i \text{ das Minimum repräsentiert} \tag{5.11}$$

Da bei einem Wechsel in der Praxis nicht nur ein negativer Wert auftritt, wird die Grenze der mimischen Aktivität durch den Mittelpunkt zwischen den Nulldurchgängen des ΔBIC-Wertes gewählt [Iur02]. Der ΔBIC-Wert selbst ergibt sich über die Formel 5.12.

$$\begin{aligned}\Delta\text{BIC}_i = &-\frac{n}{2}\log\|\Sigma_1\| + \frac{i}{2}\log\|\Sigma_{21}\| + \frac{n-i}{2}\log\|\Sigma_{22}\| \\ &+ \frac{1}{2}\lambda\left(d + \frac{d(d+1)}{2}\right)\log n\end{aligned} \tag{5.12}$$

Die Kovarianzmatrix Σ_1 in diesem Ausdruck ergibt sich durch die Norm der Merkmalsvektoren $\mathbf{e}_s, \ldots, \mathbf{e}_{s+n}$, die Modelle Σ_{21} und Σ_{22} über die Normen $\mathbf{e}_s, \ldots, \mathbf{e}_{s+i}$ bzw. $\mathbf{e}_{s+i+1}, \ldots, \mathbf{e}_{s+n}$. Die Dimensionalität d bestimmt sich durch die Anzahl der Elemente des Energievektors zu $d = 1$.

Im Einklang mit den theoretischen Grundlagen sollte das Strafgewicht λ den Wert Eins haben. In der Praxis hat es sich jedoch gezeigt, dass dieser Wert die Empfindlichkeit der Segmentgrenzenfindung beeinflusst. In den vorliegenden Anwendungen hat sich ein Wert von $\lambda = 1,1$ als brauchbar erwiesen. Überdies hat sich eine experimentell ermittelte anfängliche Fensterlänge von $n = 15$ als günstig erwiesen. Zusammen mit der Vorsegmentierung bei der Aufzeichnung können die Anfangs- und Endpunkte der mimischen Aktivität nach diesem Ansatz automatisch bestimmt werden.

5.6 Experimente und Ergebnisse der Mimikerkennung

In ersten Versuchsläufen wird die Verwendbarkeit von P3DHMM für die dynamische Mimikerkennung untersucht. Dazu wird zunächst die Datenbank mit vier Emotionen verwendet. Um Lokalisationsfehler auszuschließen werden die zu klassifizierenden Gesichtsbereiche mit Hilfe manuell gesetzter Punkte vor der weiteren Verarbeitung auf eine einheitliche Größe von 196×172 Punkten gebracht. Zur Elimination des Einflusses falscher Segmentgrenzen auf die Erkennung werden im Weiteren ideal vorsegmentierte Bildfolgen vorgegeben. Zur Initialisierung werden Modelle der Dimension $(4 \times 4) \times 4$ erstellt. Insgesamt ergibt sich durch die Wahl dieser Modellarchitektur nach Kapitel 2.7.5 für jede der vier Mimiken aus der ersten Datenbank ein äquivalentes 1DHMM mit je 64 Zuständen.

Die Merkmalsequenzen ergeben sich über die ersten zehn Koeffizienten einer zweidimensionalen DCT-Transformation mit einer Blockgröße von (16×16) Bildpunkten auf Differenzbildern. Bei der Abtastung wurde ein Überlappungsfaktor von 75% berücksichtigt. Zum Training dienen jeweils 3 Sequenzen von sechs Personen. Die Bestimmung der Erkennungsrate, welche in Tabelle 5.1 aufgeführt ist, erfolgt mit dem jeweils verbleibenden Datenmaterial für die Emotionen Überraschung, Freude, Ärger und Ekel.

Accuracy [%]	1 Mixtur	2 Mixturen	3 Mixturen	4 Mixturen
Rang 1	75,00	79,17	**87,50**	87,50
Rang 2	79,17	87,50	95,83	95,83
Rang 3	95,83	100,00	100,00	100,00

Tabelle 5.1: Erkennungsergebnisse der Experimente mit vier Emotionen

Eine Betrachtung der Ergebnisse zeigt das beste Ergebnis mit $87,5\%$ bei Verwendung von kontinuierlichen Produktionswahrscheinlichkeiten bestehend aus drei Mixturen. Durch die weitere Steigerung der Anzahl überlagerter Normalverteilungen konnte keine Verbesserung erreicht werden, was auf die Begrenztheit der Trainingsdaten zurückzuführen ist. Anhand der Ranglisten kann ein zusätzlicher Trend der Erkennungsraten abgelesen werden. So sind über den zweiten Platz beinahe alle Sequenzen richtig klassifiziert worden. Andere Parameterkonstellationen ergaben ebenfalls keine Verbesserung. Die Erkennungsraten von 1DHMM waren diesem Ergebnis mit allen Parameterkonstellationen leicht unterlegen, wobei das Beste eine Zuordnungsrate von $84,5\%$ erreichte.

Nach der erfolgreichen Einführung des P3DHMM-basierten Erkennungssystems für das erste Szenario wird in den folgenden Experimenten versucht das zweite, komplexere Erkennungsproblem zu lösen. Analog zur Vorverarbeitung der Bilddaten in der ersten Datenbank werden die Gesichtsbereiche über die im ersten Bild der Sequenz manuell markierten Augen- und Mundpositionen vorgegeben. Anschließend werden die Bereiche auch hier auf eine einheitliche Größe von 196×172 Bildpunkten skaliert.

Aufgrund des Vorhandenseins von Kopfbewegungen bei spontan geäußerten Gesichtsausdrücken, muss vor der Merkmalsextraktion ein zusätzlicher Schritt zur Bewegungskompensation eingeführt werden. Dies ist unabdingbar, da ansonsten Verfälschungen in den durch

Differenzbildung gewonnenen Merkmalen entstehen, so dass die gesuchten Änderungen aufgrund der mimischen Muskelbewegung undetektierbar und unmessbar werden. Neben einer Vielzahl möglicher Verfahren zur Bewegungskompensation wird hier der Ansatz verfolgt, den Betrag der Bewegungsenergie im Gesichtsbereich zweier benachbarter Bilder durch Translation zu minimieren. Nach dieser Bildstabilisierung können die Merkmale zur Modellierung mit den P3DHMM auf die beschriebene Weise gewonnen werden.

Zu den vier Emotionen des ersten Szenarios kommen in der erweiterten Testumgebung die Emotionen Angst, Neutral und Trauer zum Inventar hinzu. Von einer schrittweisen Erweiterung wurde aufgrund der damit verbundenen, steigenden Anzahl an Versuchen Abstand genommen. Die besten Erkennungsraten unter Verwendung von P3DHMM sind in der Konfusionsmatrix nach Tabelle 5.2 angegeben. Insgesamt standen zu Testzwecken pro Klasse jeweils 17 Sequenzen zur Verfügung. Die durchschnittliche Erkennungsrate ist im oberen linken Feld angegeben.

∅33,61%	Ärger	Ekel	Angst	Freude	Überr.	Trauer	Neutral	Erkannt [%]
Ärger	**5**	6	1	1	2	2	0	29,4
Ekel	1	**7**	1	4	3	0	1	41,2
Angst	1	5	**2**	2	7	0	0	11,8
Freude	0	0	3	**1**	13	0	0	5,9
Überr.	0	2	4	1	**10**	0	0	58,8
Trauer	4	4	4	0	1	**1**	3	5,9
Neutral	0	2	0	0	0	1	**14**	82,4

Tabelle 5.2: Konfusionsmatrix für die Klassifikation mit P3DHMM

Demnach geht mit der Erweiterung der Klassenzahl bei gleichem Modellierungsparadigma ein Einbruch der Erkennungsrate auf $33,61\%$ einher. Im Verhältnis zur Referenz der menschlichen Wahrnehmung leisten P3DHMM somit in etwa nur die Hälfte der Zuordnungsfähigkeit. Zum Teil wurden sogar stark unterzufällige Raten erzielt.

Durch die Verwendung alternativer Merkmale und Klassifikatoren soll im Folgenden versucht werden, die Erkennungsleistung automatischer Systeme auf ein akzeptables Niveau zu verbessern. Hierzu wird zunächst ein 1DHMM System mit Merkmalen auf Basis der globalen Bewegungsbeschreibung eingesetzt. Die Ergebnisse des besten 1DHMM-basierten Erkennungssystems betragen bei 8 Zuständen und 5 Mixturen bestenfalls $38,66\%$. Aufgrund der holistischen Erfassung der Änderungen in benachbarten Bildern scheint die höhere Erkennungsrate hier von der Robustheit der verwendeten Merkmale leicht zu profitieren, wodurch eine Verbesserung von etwa 5 Prozentpunkten gemessen werden konnte.

Zur weiteren Verbesserung der Erkennungsleistung wird die Klassifikation mit Mehrklassen-SVM erprobt. Hierzu werden die verschieden langen Merkmalssequenzen auf eine einheitliche Länge von 15 Bewegungsvektoren umgerechnet und durch Aneinanderreihung zu einem Vektor mit statischer Länge umgeformt. Zwar wird die zeitliche Dynamik des Musters durch diese Längennormalisierung eingeschränkt, in der Regel weicht die Dauer einer spontan geäußerten Mimik aber nicht gravierend von dieser Länge ab. Zur Unter-

scheidung der zu erkennenden Mimiken werden paarweise geschachtelte Mehrklassen-SVM mit polynomialen Kernfunktionen vom Grad 2 trainiert [Has98, Pla98]. Wie aus den Erkennungsergebnissen nach Tabelle 5.3 abgelesen werden kann, ist durch die Verwendung von SVM eine weitere Steigerung der Erkennungsleistung um fast 4 Prozentpunkte auf $42,33\%$ möglich. Andere Diskriminierungsstrategien der SVM brachten keine weitere Verbesserung.

∅42,33%	Ärger	Ekel	Angst	Freude	Überr.	Trauer	Neutral	Erkannt [%]
Ärger	**7**	5	3	3	2	1	6	25,9
Ekel	7	**10**	2	2	0	2	4	37,0
Angst	2	3	**7**	4	5	0	6	25,9
Freude	1	4	4	**14**	2	0	2	51,9
Überr.	5	1	3	3	**12**	1	2	44,4
Trauer	2	5	2	0	1	**4**	13	14,8
Neutral	0	0	0	1	0	0	**26**	96,3

Tabelle 5.3: Konfusionsmatrix für die Klassifikation mit SVM

Trotz der vorliegenden Verbesserung konnte die Perzeptionsleistung des Menschen mit den vorliegenden Methoden noch nicht erreicht werden. Durch die vorgestellten Schritte wurde der Abstand aber bereits verringert. Durch Betrachtung obiger Konfusionsmatrix kann geschlossen werden, dass die spontan geäußerten Emotionen Trauer und Angst bei vielen Sequenzen für eine Erkennung zu schwach ausgeprägt sind, wie in der Literatur ebenfalls beobachtet [Zha98].

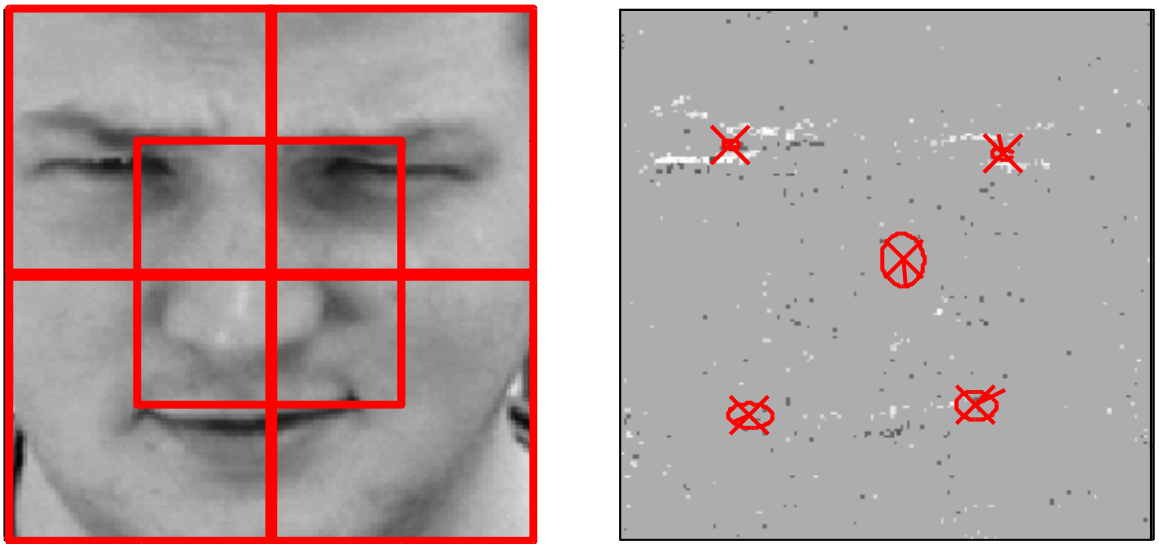


Abbildung 5.7: Unterteilung des Mimikbereichs und resultierende Merkmale

Durch eine Unterteilung des ursprünglich definierten Mimikbereichs in fünf gleichgroße Felder nach Abbildung 5.7 wird für weitere Versuche eine parallele Bestimmung globaler Be-

wegungsmomente möglich. Diese Erweiterung stellt einen Mittelweg zwischen der ganzheitlichen Modellierung und der DCT-Merkmalsberechnung mit Ortsauflösung dar. Hierdurch entsteht zunächst ein hochdimensionaler Merkmalsvektor mit $5 \cdot 7 \cdot 15$ Elementen, welcher über eine Merkmalsselektion nach Schuller wieder auf ein akzeptables Maß von etwa 100 Merkmalen reduziert werden kann [Sch06].

∅61,67%	Ärger	Ekel	Angst	Freude	Überr.	Trauer	Neutral	Erkannt [%]
Ärger	**15**	4	1	1	0	3	2	57,7
Ekel	10	**12**	3	1	2	0	0	35,7
Angst	4	6	**11**	0	2	0	1	45,8
Freude	1	0	1	**19**	1	1	3	70,3
Überr.	1	2	2	1	**17**	2	0	68,0
Trauer	3	2	0	2	0	**13**	5	52,0
Neutral	0	0	0	0	0	2	**24**	92,3

Tabelle 5.4: Ergebnisse mit SVM und Merkmalsselektion

Wie an den Resultaten aus Tabelle 5.4 abgelesen werden kann, geht mit dieser neuartigen Verarbeitung eine beachtliche Leistungssteigerung auf $61,67\%$ einher, was sogar marginal über der mittleren Perzeptionsleistung der Probanden liegt. Ursache hierfür scheint die günstigere Merkmalsextraktion, welche neben einer globalen Repräsentation eine gewisse Ortsauflösung berücksichtigt. Zudem wirkt sich die angewendete Merkmalsselektion leistungssteigernd auf die Performanz aus. Bezüglich der Differenzbild-basierten Basismerkmale scheint die Kompensation durch die Kopfbewegungen noch nicht ausgereizt zu sein. Durch eine feinere Repräsentation der Bewegungen konnte keine Verbesserung des Systems erzielt werden, was auf die bereits erreichte, maximale Trennbarkeit der Daten hindeutet.

Im Besonderen haben sich die beiden Emotionen Trauer und Angst im erweiterten Inventar als schwer erkennbar herausgestellt. Dies liegt zum einen an der im Zusammenhang der Erzeugung bereits diskutierten Schwierigkeit eindeutige spontane Emotionen hervorzurufen, zum anderen ist die vergleichsweise schwache mimische Ausprägung bzw. Intensität bei der Erkennung problematisch. Dies gilt aufgrund der zum Teil starken personenspezifischen Ausprägungen unabhängig vom angesetzten Klassifikationsmechanismus [Mic03].

Zur abschließenden Beurteilung der Erkennung spontaner Emotionen wird das Mimikinventar der zweiten Datenbank auf das der ersten mit vier Klassen reduziert. Dabei wurden etwa 53% unter Verwendung von P3DHMM und ca. 79% bei Mehrklassen-SVM gemessen, was an die Ergebnisse der ersten Datenbank ohne Kopfbewegung heranreicht. Der Grund für das unterlegene Abschneiden der P3DHMM liegt wahrscheinlich an der Empfindlichkeit der Differenzbild basierten zweidimensionalen DCT gegenüber nicht vollständig kompensierten Kopfbewegungen. Außerdem besteht die Gefahr einer Unterbestimmtheit der strukturbedingt hohen Anzahl an Modellparametern. Zuletzt scheint der Vorteil der Mehrklassen-SVM neben robusteren Merkmalen im diskriminativen Training begründet zu liegen.

5.7 Bewertung der Ergebnisse

Aufgrund der kurzen Verfügbarkeit der vorgestellten Datenbank und dem Fehlen frei verfügbarer Referenzdaten ist ein direkter quantitativer Vergleich mit anderen in der Literatur vorgestellten Verfahren nicht möglich. In verschiedenen Arbeiten wird jedoch von ähnlichen Problematiken bezüglich Eindeutigkeit berichtet. Eine ganzheitliche Analyse der zu erkennenden Mimik ohne aufwändige Suche nach einzelnen Stützpunkten scheint aufgrund der Ergebnisse jedoch prinzipiell möglich zu sein, was durch die in der Literatur vergleichbaren Leistungen indiziert wird [Pan03].

Neben einer verbesserten Bewegungskompensation können in weiterführenden Arbeiten robustere Merkmale durch Glättungsmaßnahmen gewonnen werden. Hierzu bieten sich beispielsweise die schon im Zusammenhang der Verhaltenserkennung erfolgreich eingesetzten Kalman-Filter an [Zob04]. Alternativ kann versucht werden, die mimische Muskelaktivität über den optischen Fluss zu erfassen und zu modellieren. Die bei der Merkmalsgeneration zur SVM-Klassifikation entstandene Einschränkung bezüglich Musterlänge kann zukünftig durch gebildete Funktionale der deskriptiven Statistik wieder gelockert werden [Sch06].

Darüber hinaus hat sich die Unterteilung der zu klassifizierenden Emotionen in sieben Klassen aufgrund der individuell unterschiedlichen mimischen Ausprägung der Probanden auch für die als Referenz angesehene Leistung des Menschen als problematisch erwiesen. In diesem Zusammenhang wäre eine detaillierte Unterteilung der Basisemotionen sinnvoll.

Da sich die visuell-basierte Emotionserkennung zur Zeit aufgrund der teilweise schwachen Eindeutigkeit generell als weniger zuverlässig erweist, sollte für robuste Anwendungen eine stärkere Kopplung mit akustischen, aus der Stimme abgeleiteten Merkmalen in Betracht gezogen werden [Sch04].

Kapitel 6

Zusammenfassung

Diese Arbeit befasst sich mit der Implementierung neuartiger und effizienter Systeme zur robusten Detektion von Gesichtern in Einzelbildern und Bildfolgen. Daneben werden Systeme zur Identifikation und Verifikation von Gesichtern vorgestellt und deren Leistung mit Hilfe geeigneter Datenbanken evaluiert. Zudem werden Methoden zur dynamischen Erkennung von Emotionen über die Mimik vorgestellt. Die eingesetzten Verfahren aus dem Bereich der Mustererkennung umfassen im wesentlichen die Gruppe der künstlichen neuronale Netze sowie probabilistische Hidden Markov Modelle und Support Vector Maschinen. Anliegen war es hierbei, die heute noch haptisch dominierte Mensch-Maschine-Kommunikation langfristig für den Menschen natürlicher und komfortabler zu gestalten sowie Lösungen für gesichtsbasierte Sicherheits- und Multimediaanwendungen zu liefern.

Detektion von Gesichtern. In diesem Themenkreis steht die Findung von Gesichtern in beliebigen Farb- und Grauwertbildern, sowie die Verfolgung in Bildsequenzen mit beliebig geartetem Hintergrund im Mittelpunkt. Nach der Definition der Objektklasse *Gesicht*, werden zunächst Methoden zur blockbasierten Findung frontal aufgenommener Gesichter vorgestellt, die mit Hilfe der Hautfarbe oder unter Verwendung erscheinungsbasierter Modelle funktionieren. Diese werden anhand der CMU-Standarddatenbank zur Gesichtsdetektion bezüglich Komplexität und Qualität miteinander verglichen. Daraufhin wird ein neuronaler Ansatz gewählt und zu einem neuartigen Verfahren mit zusätzlicher Schätzung der Kopfdrehung expandiert.

Mit dem Ziel einer möglichst robusten und schnellen Gesichtsverfolgung mit geringem Rechenaufwand in Bildsequenzen kann dieses Verfahren zusammen mit einer auf Hautfarben basierenden Gesichtsfindung und einem speziellen *Tracking*-Algorithmus gekoppelt werden. Das Gesamtsystem kann erfolgreich auf ein typisches Szenario im Bereich der automatischen Protokollierung von Besprechungen sowie zur Überwachung von Passagieren in einem Flugzeug eingesetzt werden.

Weiterhin wird die Gesichtsdetektion zur Bearbeitung von omnidirektionalen Kameraaufnahmen eingesetzt. In diesem Zusammenhang werden Schritte für eine Umwandlung in zweidimensionale Projektionen auf Basis der klassischen Bildbearbeitung vorgestellt. Das implementierte System kann Gesichter auch in geometrisch verzerrten Bildern finden.

Die gewonnenen Erkenntnisse werden erfolgreich auf ein System zur Findung von Gesichtsbereichen, wie Augen und Mund portiert.

Identifikation und Verifikation von Personen. Die gesichtsbasierte Erkennung von Personen wird in zwei Teilprobleme untergliedert: die Erkennung von vornehmlich frontal aufgenommenen Gesichtern und die Zuordnung von Profilen. Zur Lösung der ersten Problematik werden vorverarbeitende Schritte im Bildbereich auf Grundlage von Augen- und Mundposition eingeführt. Nach einer Merkmalsextraktion über Koeffizienten der Diskreten Cosiuns Transformation wird ein ganzheitlicher Ansatz zur erscheinungsbasierten Modellierung, hauptsächlich mit diskreten pseudo zweidimensionalen Hidden Markov Modellen vorgestellt. Unter Verwendung der drei bekannten Datenbanken AT&T (vormals ORL), AR und FERET können die Bedeutungen der einzelnen Modellparameter herausgearbeitet werden. Bei durchgeführten Simulationen werden auf den Datenbanken überragende Identifikationsraten erreicht.

Durch eine Untergliederung des Gesichtes in *Viseme*, bei gleichzeitiger Nutzung von Mehrheitsentscheidungen, kann eine zusätzliche Leistungssteigerung experimentell nachgewiesen werden. Nach der Bewertung verschiedener Verifikationsmechanismen wird ein leistungsfähiges Identifikationssystem entwickelt und ebenfalls auf der FERET-Datenbank evaluiert, wobei eine sehr geringe Fehlerquote von etwa 3% bei einem großen Personenschatz von 1.196 Personen auftritt.

Durch Kopplung dieser Komponenten mit der automatischen Gesichtsdetektion wird ein praxistaugliches Gesamtsystem zur Zugangskontrolle eines Flugzeuges unter der Voraussetzung kooperativen Verhaltens konstruiert. Bei einer günstigen Parametrisierung ist es möglich, eine für diese Aufgabe akzeptable Fehlerquote von kaum mehr als 1% zu erreichen.

Neben der rein probabilistischen Zuordnung von Vorderansichten wird der Spezialfall der Zuordnung von Profilen zu Frontalansichten in die Untersuchungen einbezogen. Alle in diesem Zusammenhang vorgestellten Verfahren werden anhand der Mugshot Datenbank bewertet. Da aufgrund verdeckter Gesichtsbereiche keine perfekten Erkennungsraten zu erwarten sind, wird bei diesem nicht vollständig lösbaren Problem die Perzeptionsleistung des Menschen als Referenz herangezogen, welche in Versuchen mit etwa 72% ermittelt wurde.

Zur Profilbilderkennung unter Kenntnis von Frontalansichten werden zunächst neu entwickelte neuronale Verfahren mit *Multi Layer Perzeptronen* zur bildbasierten Überführung von Vorderansichten in künstliche Profile vorgestellt. Diese synthetisierten Profilansichten werden unter Zuhilfenahme verschiedener Klassifikationsansätze mit einer beachtlichen Erkennungsleistung von bis zu 60% richtig zugeordnet. Aus der Integration der vorgeschalteten Transformationsstufe mit der anschließenden Modellierungsstufe erwuchsen vielversprechende hybride Strukturen, welche unter den gegebenen Bedingungen keinen Beitrag zur Verbesserung der Erkennung zu leisten konnten.

Gesichtsbasierte Emotionserkennung. Als Lösungsansatz für dieses Problem werden selbstorganisierende Methoden zur dynamischen Erkennung der sechs MPEG-4 konformen Basisemotionen nach Ekman angesetzt [Ekm74]. Es soll davon ausgegangen werden, dass

sich die Grundemotionen einer Person im Gesichtsausdruck bzw. ihrer Mimik abgelesen lassen.

Für diesen Zweck werden zunächst zwei Datenbanken aufgenommen: eine mit vier karikativ dargestellten Emotionen wie Ärger, Ekel, Freude und Überraschung, eine zweite mit allen sechs Basisemotionen: Angst, Ärger, Ekel, Freude, Trauer und Überraschung sowie dem neutralen Gemütszustand, welche möglichst natürlich und spontan sein sollen. Als Referenz für die Bewertung der vorgestellten Verfahren dient die Perzeptionsleistung von zwanzig Probanden mit einer mittleren Zuordnungsrate von 61%. Diese verhältnismäßig geringe Leistung lässt sich auf fehlenden Kontext sowie eine Uneindeutigkeit der Gesichtsausdrücke zurückführen. Die Elizitierung besser zu unterscheidender Gemütszustände bleibt im Hinblick auf eine eindeutige Bewertung der Leistung künstlicher Emotionserkennung wünschenswert.

Da die Emotionserkennung typischerweise in Videoströmen erfolgt, wird vor der Erkennung eine zeitliche Segmentierung der Videodaten notwendig. Dies kann durch die Verwendung des aus der Sprachverarbeitung bekannten *Bayesian Information Criterion* zufriedenstellend gelöst werden.

Zur Modellierung der Videosequenzen werden zwei verschiedene, schnelle Merkmalsextraktionen ohne aufwändige Suche von Stützpunkten vorgestellt. Zum einen wird die ganzheitliche Modellierung mit globalen Bewegungsmerkmalen, zum anderen eine dynamische Diskrete Cosinuns Transformation auf Differenzbildern eingesetzt. Zur Klassifikation kommen sowohl eindimensionale, als auch pseudo dreidimensionale Hidden Markov Modelle sowie Support Vector Maschinen zum Einsatz.

In Versuchsreihen mit vier ausgeprägten Mimiken wird bereits eine Zuordnungsrate von 87.5% mit Hilfe von P3DHMM ermittelt. Für sieben spontane Emotionen wird durch den Einsatz von Support Vector Maschinen ein Ergebnis von ca. 62% gemessenen, welches mit der Perzeptionsleistung der Probanden vergleichbar ist. Eine Überlegenheit von P3DHMM konnte im Rahmen der Untersuchungen nicht bestätigt werden.

Anhang A

Datenbanken

A.1 Überblick der gebräuchlichsten Gesichtsdatenbanken

In diesem Anhang wird zunächst ein Überblick über die am gebräuchlichsten, frei zugänglichen Gesichtsdatenbanken gegeben.

Die im Zusammenhang der Gesichtsdetektion am häufigsten verwendeten Datenbanken sind aus der Tabelle A.1 ersichtlich. Dabei variiert die Anzahl der Personen und die Anzahl der Referenzen entsprechend den Angaben.

Bezeichung	Ortsangabe im WWW	Kurzbeschreibung
MIT Test Set [Sun98]	`http://www.cs.cmu.edu/~har`	Zwei Datensätze mit niedriger und hoher Auflösung von Grauwertbildern, welche mehrere Gesichter vor komplexen Hintergründen beinhalten
CMU Test Set [Row98]	`http://www.cs.cmu.edu/~har`	130 Grauwertbilder mit insgesamt 507 Frontalansichten
CMU Profile Face Test Set [Sch00]	`ftp://eyes.ius.cs.cmu.edu/usr20/ftp/testing_face_images.tar.gz`	208 Grauwertbilder von Gesichtern mit Profilansicht
Kodak Data Set [Lou98]	Eastman Kodak Corporation	Farbbilder mit Gesichtern verschiedener Größe, Ansicht und unter verschiedenen Belichtungseinflüssen
BIO ID [Jes01]	`http://www.humanscan.de/support/downloads/face-db.php`	Sammlung von Frontalbildserien verschiedener Personen

Tabelle A.1: Überblick der verbreitetsten Datenbanken zur Gesichtsdetektion

Die Tabelle A.2 enthält Datenbanken, welche für verschiedene Problemstellungen der Gesichtserkennung konzipiert wurden.

Bezeichung	Ortsangabe im WWW	Kurzbeschreibung
MIT Datenbank [Tur91a]	ftp://whitechapel.media.mit.edu/pub/images	Gesichtsdaten von 16 verschiedenen Personen, jeweils mit 27 Aufnahmen unter veränderten Beleuchtungsverhältnissen mit verschiedener Skalierung und Kopforientierung
FERET Datenbank [Phi00]	http://www.nist.gov/humanid/feret	Eine der größten Datenbanken mit einer Vielzahl von Bildern jeweils einer Person mit variablen Gesichtsausdrücken, Posen und zeitlichen Abständen bis hin zu mehreren Monaten
UMIST Datenbank [Gra98]	http://images.ee.umist.ac.uk/danny/database.html	Enthält 564 Bilder von 20 Individuen, jeweils mit verschiedenen Ansichten vom Profil bis zur Frontalansicht
University of Bern	ftp://iamftp.unibe.ch/pub/Images/FaceImages/	Enthält 450 Bilder von 30 Personen. Dabei entfallen jeweils 10 Aufnahmen pro Person auf die Frontalansicht und 5 auf die Profilansicht
Yale Database [Bel97]	http://cvc.yale.edu	Bilddatenbank mit variablen Gesichtsausdrücken, Brillen und verschiedenen Beleuchtungsverhältnissen
AT&T (Olivetti) Datenbank [Sam94]	http://www.uk.research.att.com	400 Bilder von 40 verschiedenen Personen, mit jeweils 10 Beispielen
Harvard Datenbank [Hal95]	ftp://ftp.hrl.harvard.edu/pub/faces/	Ausgeschnittene und maskierte Gesichtsbilder unter eine breiten Spektrum von Belichtungsverhältnissen während der Aufnahme
M2VTS Datenbank [Pig97]	http://poseidon.csd.auth.gr/M2VTS/index.html	Eine multimodale Datenbank, welche eine Vielzahl von Bildsequenzen beinhaltet
BANCA Datenbank [BB03]	http://www.ee.surrey.ac.uk/banca/BancaHomeBody.htm	Multimodale Datenbank mit Gesichtsaufnahmen unter variablen äußeren Rahmenbedingungen
PIE Datenbank [Sim03]	http://www.ri.cmu.edu/projects/project_418.html	Beinhaltet 41.368 Bilder von 68 verschiedenen Personen bei 13 verschiedenen Posen unter 43 verschiedenen Beleuchtungen und mit 4 verschiedenen Gesichtsausdrücken
NIST Mugshot Datenbank [Wat94]	http://www.nist.gov/srd/nistsd18.htm	FBI-Sammlung von Bildpaaren mit Frontal- und Profilansichten von über 1000 Personen
Purdue AR Datenbank [Mar98]	http://rvl1.ecn.purdue.edu/~aleix/aleix_face_DB.html	Insgesamt 3276 Gesichtsbilder mit variablen Gesichtsausdrücken sowie Verdeckungen unter verschiedenen Belichtungen

Tabelle A.2: Überblick der verbreitetsten Datenbanken zur Gesichtserkennung

A.2 MPEG-4 konforme Stützpunkte

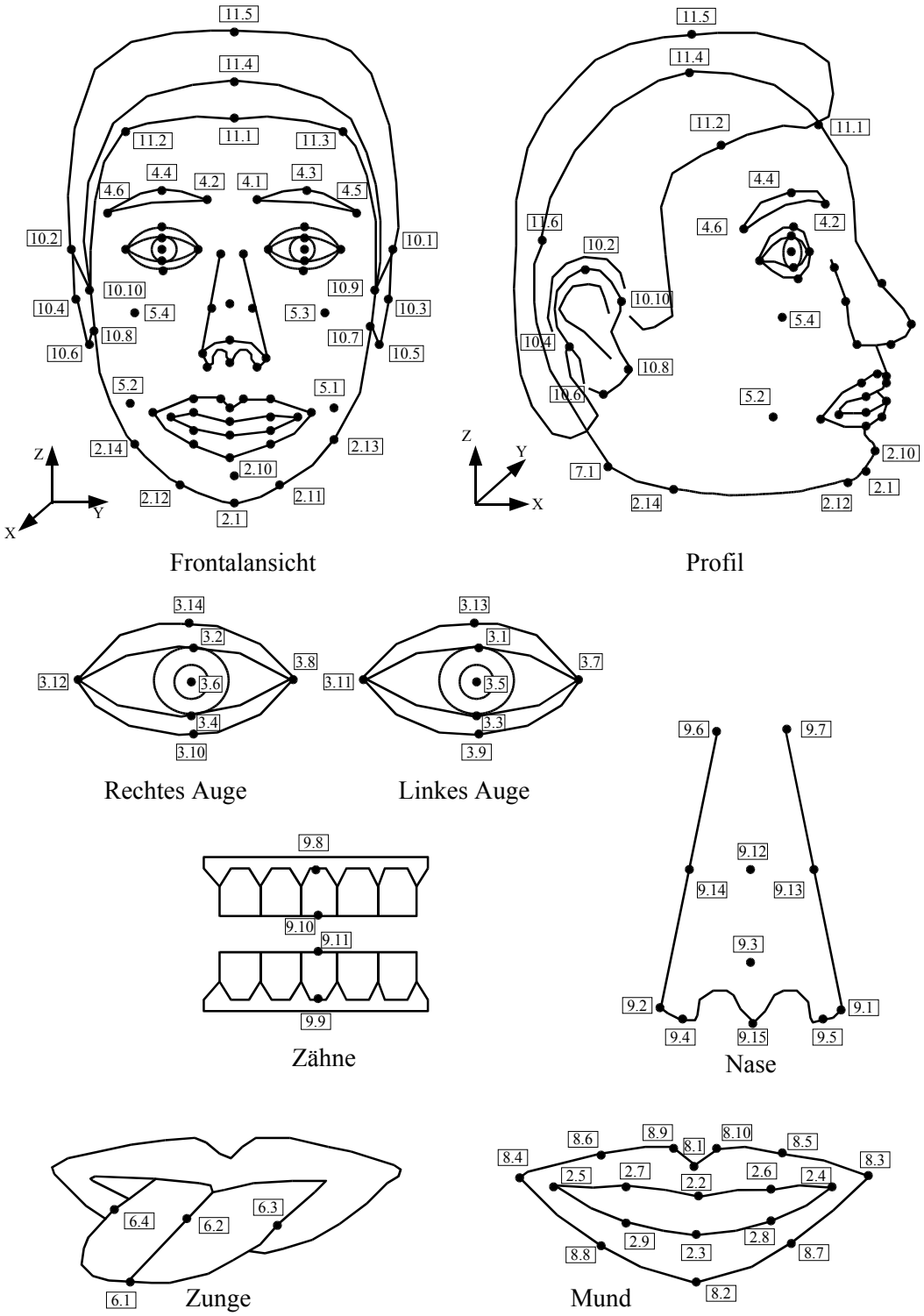
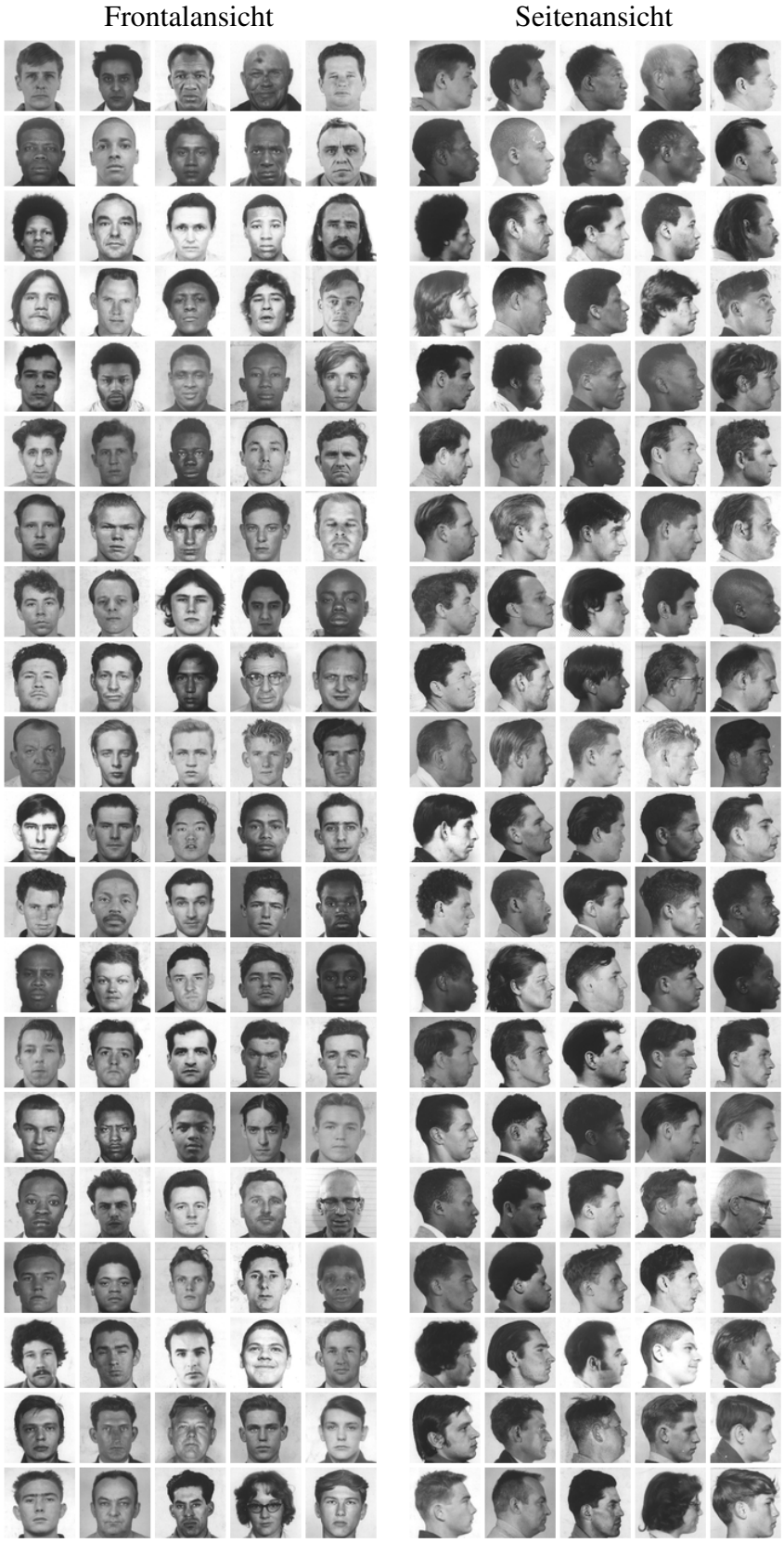


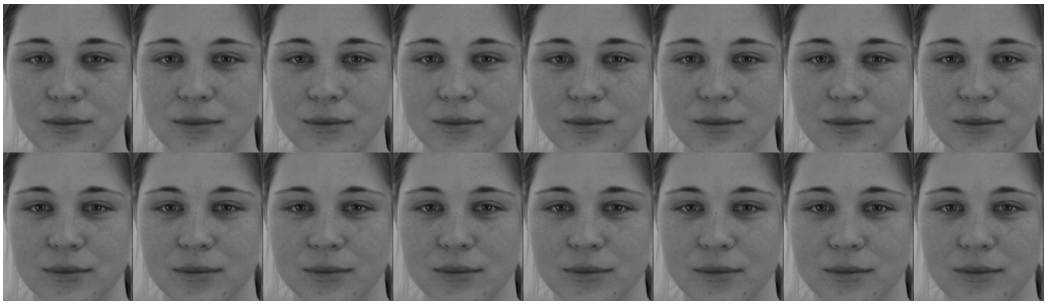
Abbildung A.1: Auflistung MPEG-4 konformer Stützpunkte [Ost98]

A.3 Testkorpus der MUGSHOT-Datenbank

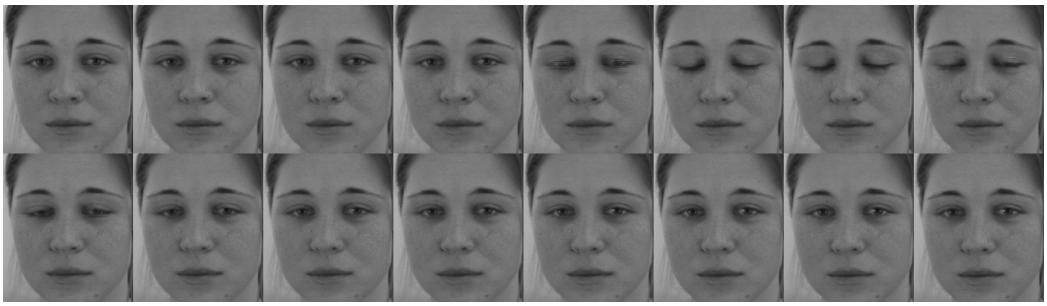


A.4 Auszüge der Mimik-Datenbank

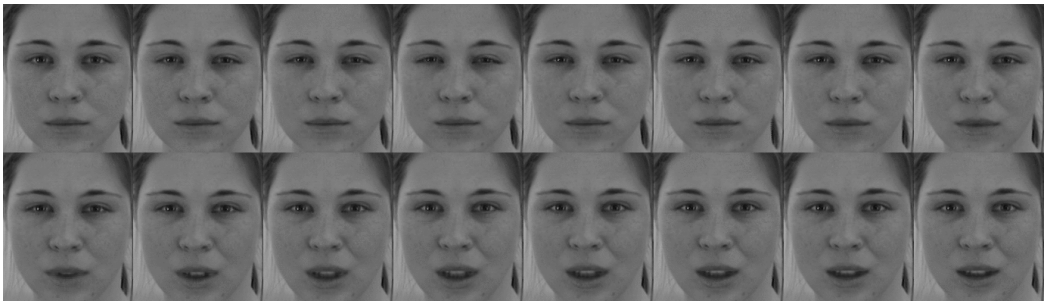
- Neutral:



- Trauer:



- Angst:

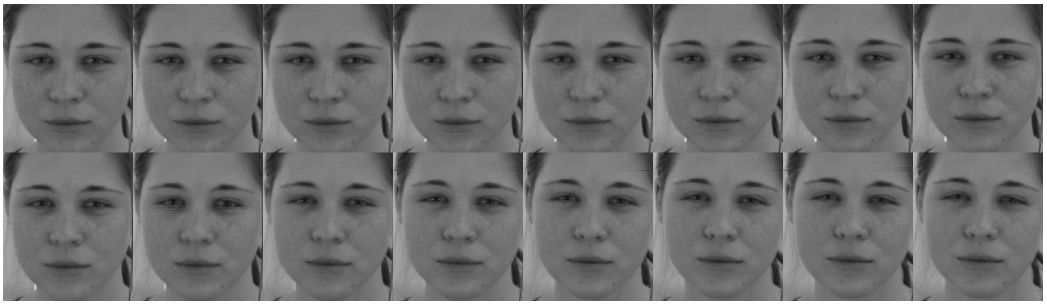


- Freude:

- Ekel:

- Überraschung:

- Ärger:



Anhang B

Mustererkennung

B.1 Berechnung der Hauptachsentransformation

Die Drehmatrix $\mathbf{U}$ ist durch die Eigenvektoren der mittelwertfreien Mustervektoren gegeben:

- $\mathbf{x_1}, \ldots, \mathbf{x_M}$ seien Musterdaten einer gegebenen Verteilung $\mathbf{X}$ in Form von Spaltenvektoren der Dimension $(N \times 1)$

  1. Bestimmung des Erwartungswert-, bzw. Mittelwertvektors $\bar{\mathbf{x}} = \frac{1}{M} \sum_1^M \mathbf{x}_i$
  2. Entfernung des Gleichanteils bei allen Vektoren $\phi_i = \mathbf{x}_i - \bar{\mathbf{x}}$
  3. Formation der Hilfsmatrix $\mathbf{A} = [\phi_\mathbf{1}, \ldots, \phi_\mathbf{M}]$ mit der Dimension $(N \times M)$.
  4. Berechnung der Kovarianzmatrix $\mathbf{C} = \frac{1}{M} \sum_{n=1}^M \phi_n \phi_n^T = \mathbf{A}\mathbf{A}^T$, Dimension $(N \times N)$.
  5. Ermittlung und Sortieren der N Eigenwerte von $\mathbf{C}$: $\lambda_1 > \lambda_2 > \ldots > \lambda_N$.
  6. Bestimmung der Eigenvektoren[1] von $\mathbf{C}$: $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_N$.

- Auf Grund der Tatsache, dass die Kovarianzmatrix $\mathbf{C}$ symmetrisch ist, spannen die Eigenvektoren $\mathbf{u_1}, \ldots, \mathbf{u_N}$ ein orthogonales Basissystem auf. Dadurch können die mittelwertfreien ursprünglichen Vektoren auch als eine Linerkombination der Eigenvektoren angegeben werden.

$$\mathbf{x} - \bar{\mathbf{x}} = y_1\mathbf{u}_1 + y_2\mathbf{u}_2 + \ldots + y_N\mathbf{u}_N = \sum_{i=1}^{N} y_i\mathbf{u}_i \tag{B.1}$$

  Das ursprüngliche Muster $\mathbf{x}$ ist somit in den Vektor $\mathbf{y} = [y_1, \ldots, y_N]^T$ überführt worden.

[1] Die Berechnung der Eigenvektoren kann jedoch insbesondere für große Datenmengen problematisch werden. Zur Lösung dieser Problematik sei auf das Kapitel B.2 verwiesen.

B.2 Berechnung von Eigenvektoren

Eine Matrix $\mathbf{C}$ mit der Dimension $N \times N$ hat einen vom Nullvektor verschiedenen Eigenvektor $\mathbf{u}_i$ mit dem korrespondierenden Eigenwert λ_i, falls

$$\mathbf{C}\mathbf{u}_i = \lambda_i \mathbf{u}_i \tag{B.2}$$

gilt. Aufgrund obigen Ausdrucks sind auch skalare Vielfache von $\mathbf{u}_i$ möglich, was aber letzthin den gleichen Eigenvektor repräsentiert. Zur Erfüllung obiger Forderung muss Gleichung B.3 erfüllt werden, was auf die Berechnung der Nullstellen eines N-gradigen, sogenannten charakteristischen Polynoms hinausläuft.

$$|\mathbf{C} - \lambda_i \mathbf{E}| = 0 \tag{B.3}$$

Die Matrix $\mathbf{E}$ in dieser Gleichung ist eine dimensionsmäßig entsprechende $N \times N$ Einheitsmatrix. Zur Bestimmung der Eigenvektoren wird Gleichung B.2 umgestellt. Der Vektor $\mathbf{0}$ repräsentiert einen Spaltenvektor mit N Nullen.

$$\mathbf{C}\mathbf{u}_i - \lambda \mathbf{u}_i = \mathbf{0} \tag{B.4}$$

Dieses lineare Gleichungssystem mit prinzipiell N verschiedenen Eigenvektoren kann über geeignete numerische Verfahren, wie der Jacoby Methode [Pre92], realisiert werden.

Da in der Praxis aber üblicherweise die Anzahl an sinnvollen, von Null verschiedenen Eigenwerten und somit Eigenvektoren wesentlich kleiner ist als die Dimension N der Matrix, soll zur Beschleunigung der Berechnung ein Näherungsverfahren vorgestellt werden, welches sich insbesondere zur Bestimmung der sogenannten Eigenfaces als praktikabel erwiesen hat [Tur91b].

Hierzu wird die Entstehung der symmetrischen Kovarianzmatrix Matrix $\mathbf{C}$ betrachtet, welche aus dem Produkt $\mathbf{A}\mathbf{A}^T$ gebildet wird. Die Matrix $\mathbf{A}$ $(M \times N)$wurde aus M Mustervektoren zusammengestellt. Da die Anzahl der Vektoren und somit die der Klassen in der Regel wesentlich kleiner als die Dimension der Daten selbst ist $M \ll N$, handelt es sich bei den meisten Eigenwerten in Gleichung B.3 um Werte nahe oder gleich Null. Die korrespondierenden Eigenvektoren nach Gleichungssystem B.4 sind daher nicht von Relevanz und können vernachlässigt werden. Der Berechnungsaufwand für ein derart hochrangiges Gleichungssystem ist somit unpraktikabel. Ziel ist die Dimensionsreduktion von N auf M.

Unter der Annahme, dass die Eigenwerte $\lambda_{M+1}, \ldots, \lambda_N$ gleich Null sind, brauchen nur die Eigenvektoren für eine $M \times M$ Matrix bestimmt werden. Die Eigenvektoren für die ursprüngliche Verteilung ist dann durch eine Linearkombination gegeben. Die Eigenvektoren von $\mathbf{A}^T\mathbf{A}$ seien als $\mathbf{v}_i$ gekennzeichnet, was analog zu Gleichung B.2 durch

$$\mathbf{A}^T\mathbf{A}\mathbf{v}_i = \mu_i \mathbf{v}_i \tag{B.5}$$

ausgerückt werden kann. Durch beidseitige Multiplikation mit $\mathbf{A}$ erhält man Gleichung B.6, aus welcher durch einen Vergleich geschlossen werden kann, dass $\mathbf{A}\mathbf{v}_i = \mathbf{u}_i$ die Eigenvektoren zur ursprünglichen Matrix $\mathbf{C} = \mathbf{A}\mathbf{A}^T$ sein müssen.

$$\mathbf{A}\mathbf{A}^T\mathbf{A}\mathbf{v}_i = \mu_i \mathbf{A}\mathbf{v}_i \tag{B.6}$$

B.3 Künstliche neuronale Netze

B.3.1 Einsatzgebiete neuronaler Netze

Die Funktionalität von NN kann nach Jain [Jai96] in folgende Gruppen unterteilt werden:

1. **Klassifikation von Mustern.** Die Aufgabe der Klassifikation besteht wie schon erläutert in der Zuordnung unbekannter Muster, repräsentiert durch Merkmalvektoren, zu einer von mehreren zuvor definierten Klassen. Zu den bekanntesten Anwendungen gehören die Einzelzeichenerkennung sowie die Erkennung einzelner Phoneme. In dieser Arbeit wird mit Hilfe von NN die Zugehörigkeit zur Klasse Gesicht bzw. nicht Gesicht unterschieden.

2. **Kategorisierung / Clustering**[2]**.** Im Gegensatz zur Klassifikation ist die Klassenzugehörigkeit der Trainingsdaten im Vorfeld nicht bekannt. Durch geeignete Algorithmen wird mit Hilfe der Ähnlichkeit der verwendeten Daten eine Gruppeneinteilung vorgenommen. Mögliche Szenarien hierfür sind beispielsweise die Datenkompression sowie die in dieser Arbeit verwendeten Vektorquantisierer.

3. **Approximation von Funktionen.** Annahme hierbei ist, dass die Eingaben $(x_1, \ldots, x_n)$ über eine unbekannte Funktion μ auf die Ausgaben $(y_1, \ldots, y_n)$ abgebildet wurden. Die Aufgabe ist es nun eine Schätzfunktion $\hat{\mu}$ zu finden, welche die unbekannte, analytisch oftmals nicht anzugebende Funktion möglichst gut annähert. Diese Funktion wird im Rahmen der Synthetisierung von Profilbildern verwendet.

4. **Prädiktion und Vorhersage.** Unter der Kenntnis von gemessenen Zeitreihen, soll mit Hilfe der Prädiktion auf den nächsten, unbekannten Werte einer Größe geschlossen werden. Derartige Netzwerke werden beispielsweise für Bewegungsmodelle beim Tracking verwendet.

5. **Optimierung.** Ziel ist die Findung einer möglichst optimalen Lösung einer gegebenen Aufgabe unter zuvor definierten Bedingungen. Das klassische Problem für diese Aufgabenstellung ist Minimierung des Weges eines reisenden Vertreters, das sogenannte *Traveling Salesman Problem* (TSP).

6. **Inhaltsorientierte Speicher.** Diese Assoziativspeicher ermöglichen es dem Anwender über Inhalte anstatt über Indizes auf bestimmte Adressen bzw. Zellen zuzugreifen. Die besondere Eigenschaft ist die Abrufbarkeit mit unvollständigen oder verrauschten Eingabemustern.

7. **Regelsystem.** Auch zur Steuerung in Regelkreisen können NN eingesetzt werden, beispielsweise zur adaptiven Geschwindigkeitskontrolle.

[2] Dieser englische Ausdruck steht für Gruppenbildung

B.3.2 Backpropagation mit Gradientenabstiegsverfahren

Ziel des Gradientenabstiegsverfahrens ist die iterative Findung geeigneter Gewichte, um eine gegebenen Fehlerfunktion zu minimieren, wie in Abbildung B.1 illustriert.

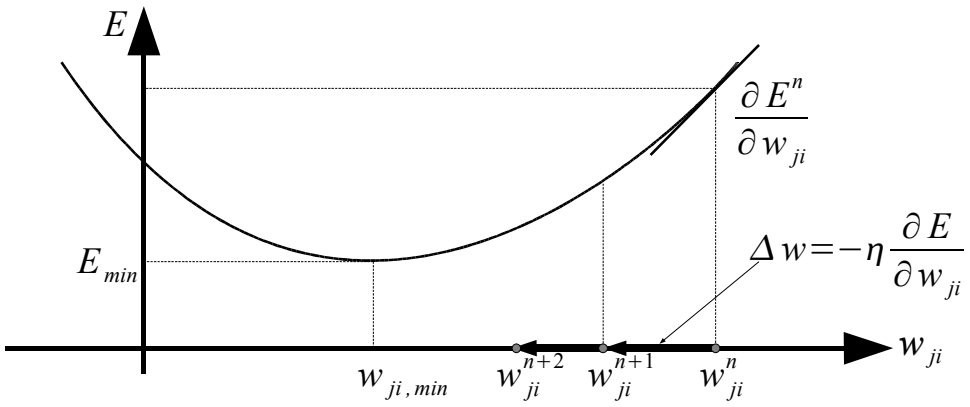


Abbildung B.1: Iterative Neubestimmung der Gewichte beim Backpropagtion-Algorithmus.

Im Beispiel repräsentiert der negative Gradient der Fehlerfunktion $\frac{\partial E}{w_{ji}}$ die lokale Abstiegsrichtung und somit die Suchrichtung für verbesserte Gewichte w_{ji}. Nach der n-ten Trainingsiteration – auch Epoche genannt – können die Gewichte anhand folgender Gleichung neu geschätzt werden:

$$w_{ji}^{n+1} = w_{ji}^{n} + \Delta w_{ji}. \tag{B.7}$$

In dieser Gleichung gibt Δw_{ji} die Proportionalität zum negativen Gradienten $-\frac{\partial E}{w_{ji}}$ an. Mit der Lernrate η kann die Schrittweite auf der Fehlerfunktion für jede Iteration angegeben werden, so dass zusammenfassend Gleichung B.8 formuliert werden kann.

$$\Delta w_{ji} = -\eta \frac{\partial E}{w_{ji}} \tag{B.8}$$

Wie aus der Abbildung B.1 bezüglich der Lernrate abstrahiert werden kann, würde der Algorithmus bei einem zu großen Wert um das Minimum oszillieren, bei einem zu kleinen Wert jedoch nur langsam konvergieren. Sinnvolle Werte für η liegen in der Praxis beispielsweise zwischen $0.05 \leq \eta < 0.5$. Aufgrund der Problematik der Konvergenz werden zunehmend alternative Lernraten verwendet, die im folgenden Kapitel noch ausführlicher diskutiert werden.

Zunächst soll der Gradient bezüglich eines Parameters in der Ausgabeschicht eines Netzes mit Hilfe der zweifach angewendeten Kettenregel bestimmt werden.

$$\frac{\partial E}{w_{ji}} = \underbrace{\frac{\partial E}{\partial o_j}}_{\text{B.10}} \underbrace{\frac{\partial o_j}{\partial a_j}}_{\text{B.11}} \underbrace{\frac{\partial a_j}{\partial w_{ji}}}_{\text{B.12}} \tag{B.9}$$

Hierbei ist die Aktivierung in einem Ausgabeneuron mit dem Index j nach Gleichung 2.9 durch die Aktivierungsfunktion $a_j = \sum_j w_j \cdot x_j$ gegeben. Die Aktivierungen der Neuronen

können durch die vorwärtsgerichtete Fortpflanzung der Eingaben bestimmt werden. Über die Aktivierungsfunktion 2.10 gilt ferner $o_j = F(a_j)$. Die einzelnen partiellen Ableitungen aus Gleichung B.9 können wie folgt aufgelöst und zusammenfassend in Gleichung B.13 geschrieben werden.

$$\frac{\partial E}{\partial o_j} = \frac{\partial \left(\frac{1}{2}\sum_j (t_j - y_j)^2\right)}{\partial o_j} = -(t_j - o_j) \tag{B.10}$$

$$\frac{\partial o_j}{\partial a_j} = f'(a_j) \tag{B.11}$$

$$\frac{\partial a_j}{\partial w_{ji}} = x_i \tag{B.12}$$

$$\frac{\partial E}{\partial w_{ji}} = -(t_j - o_j) f'(a_j) x_i \tag{B.13}$$

Durch Einsetzen von B.13 in Gleichung B.8 können die Gewichtsänderungen folgendermaßen angegeben werden.

$$\begin{aligned} \Delta w_{ji} &= \eta(t_j - o_j) f'(a_j) x_i \\ &= \eta \delta_j x_i \end{aligned} \tag{B.14}$$

Dabei gibt $\delta_j = (t_j - o_j) f'(a_j)$ den so genannten lokalen Fehler an, wodurch der Algorithmus auch über den Namen der Delta-Regel bekannt ist. Da der Fehler $(t_j - o_j)$ nur für die Ausgabeneuronen bestimmbar ist, lässt sich obige Delta-Regel zunächst nur auf einschichtige MPLs anwenden. Für mehrschichtige MLP ist obiger Ausdruck auf die Anwendung in der letzten Schicht beschränkt. Unter Verwendung der nicht-linearen Sigmoidfunktion nach Abbildung 2.8 $F_{BS}(a) = \frac{1}{1+e^{-a}}$ kann die Update Regel wie folgt angegeben werden, wobei $o_j = y_j$ die Ausgänge von Neuron j und x_i die Eingänge von Neuron i sind.

$$\begin{aligned} \Delta w_{ji} &= \eta(t_j - o_j) F(a_j)(1 - F(a_j)) x_i \\ &= \eta(t_j - o_j) o_j (1 - o_j) x_i \end{aligned} \tag{B.15}$$

Der lokale Fehler δ_j in der Ausgabeschicht bei dieser gewählten Nichtlinearität ist somit $\delta_j = (t_j - o_j) o_j (1 - o_j)$.

Für Neuronen in den versteckten Schichten können die Fehler $(t_j - o_j)$ jedoch nicht wie oben angegeben werden, da hier keine expliziten Zielwerte vorhanden sind. Daher wird ein lokaler Fehler benötigt, der mit den vorhandenen Werten bestimmbar ist. Hierzu werden noch einmal die Gleichungen B.8 und B.9 betrachtet und der lokale Fehler reformuliert als $\delta_j = -\frac{\partial E}{\partial o_j}\frac{\partial o_j}{\partial a_j}$ und ein Neuron j in der hintersten verstecken Schicht betrachtet, welches mit einem Ausgabeneuron k verbunden ist.

$$
\begin{aligned}
\delta_j = -\frac{\partial E}{\partial o_j}\frac{\partial o_j}{\partial a_j} &= - \frac{\partial(\frac{1}{2}\sum_k (t_k - o_k)^2)}{\partial o_j} f'(a_j) \\
&= - \left[\sum_k (t_k - o_k)\frac{\partial (t_k - o_k)}{\partial o_j}\right] f'(a_j) \\
&= - \left[\sum_k (t_k - o_k)\frac{\partial (t_k - o_k)}{\partial a_k}\frac{\partial a_k)}{\partial o_j}\right] f'(a_j) \\
&= - \left[\sum_k (t_k - o_k)\frac{\partial (t_k - f(a_k))}{\partial a_k} w_{kj}\right] f'(a_j) \\
&= \left[\sum_k \underbrace{(t_k - o_k) f'(a_k)}\, w_{kj}\right] f'(a_j) \\
&= \left[\sum_k \qquad \delta_k \qquad w_{kj}\right] f'(a_j) \qquad \text{(B.16)}
\end{aligned}
$$

In obigen Gleichungen gilt für die Aktivierungen in der Ausgabeschicht $y_j = o_j$, damit $a_k = \sum_k w_{kj} y_j = \sum_k w_{kj} o_j$ und für die Ableitung $\frac{\partial a_k}{\partial o_j} = \frac{\partial \sum_k w_{kj} o_j}{\partial o_j} = w_{kj}$. Unter Verwendung der Sigmoidfunktion kann dann geschrieben werden $\delta_j = [\sum_k \delta_k w_{kj}][f'(a_j)] = [\sum_k \delta_k w_{kj}][o_j(1 - o_j)]$. Hierin ist δ_k der lokale Fehler, der von hinten durch das Netz propagiert wird. Die Update Regel für die Gewichte in der verstecken Ebene ist somit nach Gleichung B.17 gegeben.

$$\Delta w_{ji} = \eta \left[\sum_k \delta_k w_{kj}\right] [o_j(1 - o_j)] y_i \qquad \text{(B.17)}$$

Für die weiteren verstecken Schichten kann obiges Prinzip rekursiv bis zur Eingabeschicht angewendet werden.

B.3.3 Der RPROP-Algorithmus

Zur Bestimmung eines Vorzeichenwechsels bezüglich des Gradienten wird für jedes Gewicht ein zusätzlicher, individueller Update-Wert Δ_{ji}^n eingeführt, welcher vor dem Training für $n = 0$ vorinitialisiert werden muss und während des Lernprozesses an den Verlauf der Fehlerfunktion nach Gleichung B.18 angepasst wird. Dieser Updatewert repräsentiert die aktuelle Schrittweite der Suche, wobei deren Beträge innerhalb eines gegebenen Bereichs $[\Delta_{ji,min} < \eta^-, \eta^+ < \Delta_{ji,max}]$ bleiben.

$$\Delta_{ji}^n = \begin{cases} \eta^+ * \Delta_{ji}^{n-1} & \text{für } \frac{\partial E^{n-1}}{\partial w_{ji}} * \frac{\partial E^n}{\partial w_{ji}} > 0 \\ \eta^- * \Delta_{ji}^{n-1} & \text{für } \frac{\partial E^{n-1}}{\partial w_{ji}} * \frac{\partial E^n}{\partial w_{ji}} < 0 \\ \Delta_{ji}^{n-1} & \text{sonst} \end{cases} \qquad \text{(B.18)}$$

In obiger Gleichung geben $0 < \eta^- < \eta^+$ die jeweiligen Update-Parameter für ein gleichbleibendes bzw. alternierendes Vorzeichen an. In Anwendungen werden diese beiden Werte

ohne Beschränkung der Allgemeinheit oftmals als $\eta^- = 0.5$ und $\eta^+ = 1.2$ gewählt. Konkret bedeutet dies, dass die letzte Änderung im Falle eines Vorzeichenwechsels zu groß war und, dass ein Minimum zwischen altem und aktuellem Gewicht liegt. Die ursprüngliche Gewichtung muss somit wiederhergestellt werden, die Richtung umgekehrt und die Suchweite angepasst werden. Falls die Gradienten das gleiche Vorzeichen haben, kann die Suchrichtung beibehalten und die Schrittweite leicht erhöht werden. Zusammen mit Gleichung B.18 kann die Formulierung für Gewichtsänderungen Δw_{ji} als Gleichung B.19 geschrieben werden, welche sowohl die Schrittweite, als auch die Suchrichtung beinhaltet.

$$\Delta w_{ji}^n = \begin{cases} -\Delta_{ji}^n & \text{für } \frac{\partial E^n}{\partial w_{ji}} > 0 \\ +\Delta_{ji}^n & \text{für } \frac{\partial E^n}{\partial w_{ji}} < 0 \\ 0 & \text{sonst} \end{cases} \tag{B.19}$$

Analog zum Backpropagationverfahren, kann das eigentliche Gewichts-Update mit der bekannten Gleichung angegeben werden. Die Gewichte werden beim Training nach RPROP üblicherweise erst nach dem Durchlaufen aller Trainingsdaten angepasst.

$$w_{ji}^{n+1} = w_{ji}^n + \Delta w_{ji}^n \tag{B.20}$$

Das obige Prinzip soll anhand von Abbildung B.2 veranschaulicht werden. Da die letzten beiden Gradienten $\frac{\partial E^{n-1}}{\partial w}$ und $\frac{\partial E^n}{\partial w_{ji}}$ der gegebenen Fehlerfunktion das gleiche Vorzeichen haben, wird die Sprungweite Δw_{ji}^n um den Faktor $\eta^+ = 1.2$ erhöht. Die Sprungrichtung wird bei gleichem Vorzeichen der Gradienten $\frac{\partial E^{n-1}}{\partial w_{ji}}$ und $\frac{\partial E^n}{\partial w_{ji}}$ beibehalten, hier also nach links. In der nächsten Iteration haben die Gradienten $\frac{\partial E^n}{\partial w_{ji}}$ und $\frac{\partial E^{n+1}}{\partial w_{ji}}$ nach obigem Beispiel verschiedene Vorzeichen. Die Sprungweite Δw_{ji}^{n+1} wird deshalb halbiert und die Sprungrichtung umgekehrt.

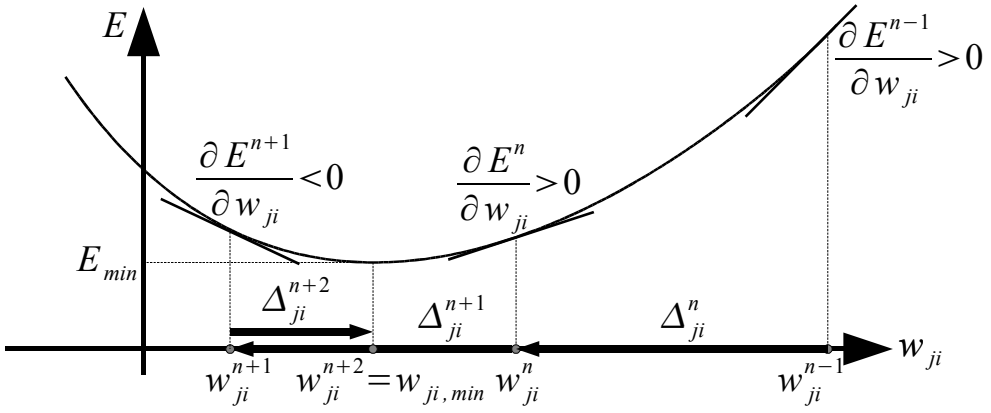


Abbildung B.2: Iterative Anpassung der Sprungweiten beim RPROP-Algorithmus.

Im folgenden Abschnitt ist der RPROP-Algorithmus in Pseudo-Code zusammengefasst.

B.3.4 Der RPROP-Algorithmus in Pseudo-Code

Der folgende Pseudo-Code repräsentiert den RPROP-Trainingsalgorithmus [Rie93].

For All Weights and Biases{
if$(\frac{\partial E^{n-1}}{\partial w_{ji}} * \frac{\partial E^n}{\partial w_{ji}} > 0)$**then**{
$\Delta_{ji}^n = \textbf{minimum}\left(\Delta_{ji}^{n-1} * \eta^+, \Delta_{\text{max}}\right)$
$\Delta w_{ji}^n = -\textbf{sign}\left(\frac{\partial E^n}{\partial w_{ji}}\right) * \Delta_{ji}^n$
$w_{ji}^{n+1} = w_{ji}^n + \Delta w_{ji}^n$
}
else if$(\frac{\partial E^{n-1}}{\partial w_{ji}} * \frac{\partial E^n}{\partial w_{ji}} < 0)$**then**{
$\Delta_{ji}^n = \textbf{maximum}\left(\Delta_{ji}^{n-1} * \eta^-, \Delta_{\text{min}}\right)$
$w_{ji}^{n+1} = w_{ji}^n - \Delta w_{ji}^{n-1}$
$\frac{\partial E^n}{\partial w_{ji}} = 0$
}
else if$(\frac{\partial E^{n-1}}{\partial w_{ji}} * \frac{\partial E^n}{\partial w_{ji}} = 0)$**then**{
$\Delta w_{ji}^n = -\textbf{sign}\left(\frac{\partial E^n}{\partial w_{ji}}\right) * \Delta_{ji}^n$
$w_{ji}^{n+1} = w_{ji}^n + \Delta w_{ji}^n$
}
}

B.4 Hidden Markov Modelle

B.4.1 Die drei fundamentalen Bearbeitungsschritte

Unabhängig von der Wahl der internen Zustandsabfolgen, ihrer Anzahl und den verwendeten Modellierungsparadigmen, kann das Training und die Klassifikation mit HMM in die folgenden drei Teilprobleme aufgespaltet werden [Rab89]:

1. **Berechnung des Wahrscheinlichkeits- oder Ähnlichkeitsmaßes einer Sequenz von Beobachtungen unter Verwendung eines gegebenen HMM**

   Bestimmt werden soll die Wahrscheinlichkeit $P(\mathbf{X}|\lambda)$, welche für die Klassifikation nach Gleichung 2.28 benötigt wird. Gegeben ist hierbei eine Merkmalsequenz $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_T]$ der Länge T und ein HMM über den Parametersatz $\lambda = (\mathbf{A}, \mathbf{b})$.

   Eine triviale Möglichkeit ist die Berechnung durch die Summation der Produktionswahrscheinlichkeiten über alle möglichen Pfade bzw. Zustandssequenzen der Länge T, da die eigentliche Sequenz ja unbekannt ist.

   $$P = (\mathbf{x}|\lambda) = \sum_{Q \in Q^T} P(\mathbf{x}, Q|\lambda) = \sum_{Q \in Q^T} \pi_{q_1} b_{q_1}(\mathbf{x}_1) \cdot \prod_{t=2} T a_{q_{t-1} q_t} b_{q_t}(\mathbf{x}_t) \qquad \text{(B.21)}$$

   Wie anhand obiger Gleichung schnell geschlossen werden kann, ist dieser Ansatz aufgrund der hierbei auftretenden Komplexität unpraktikabel. Nach Rabiner [Rab89] werden zur Berechnung näherungsweise $2T \cdot N^T$ Berechnungsschritte benötigt. Die möglichen Pfade bei der Besetzung der Zustände sind in Abbildung B.3 in Form eines Trellisdiagramms dargestellt.

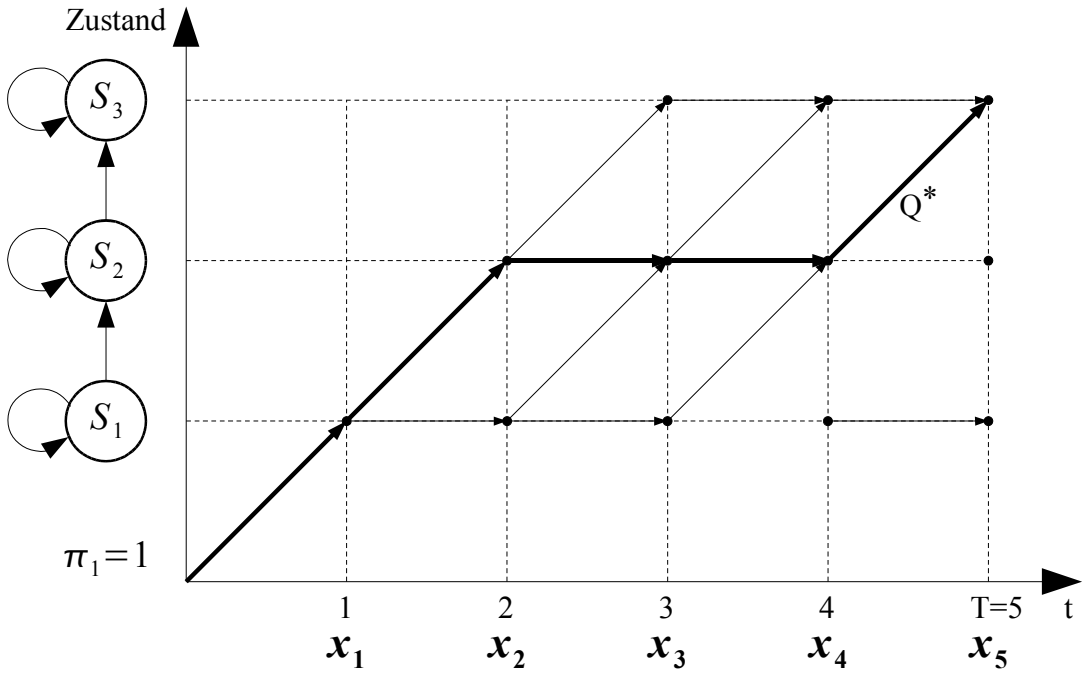


Abbildung B.3: Zuordnung zwischen einem Muster und den Zuständen eines HMM in einem Trellisdiagramm

Eine in der Praxis oftmals eingesetzte Methode zur Findung einer Lösung dieser Problemstellung ist der so genannte Forward-Algorithmus, welcher auf die dynamische Programmierung unter Verwendung von Wertetabellen zurückgreift. Durch die Zusammenfassung von Wahrscheinlichkeiten mehrerer Wege, die zu einem Zustand führen, kann der benötigte Rechenaufwand drastisch auf N^2T Schritte reduziert werden. Der Forward-Algorithmus kann in einer formalen Beschreibung dem Anhang B.4.2 entnommen werden.

2. **Die Findung der besten Zustandsabfolge durch ein Modell**

   Neben der Bestimmung der gesamten Produktionswahrscheinlichkeit, wird die beste Zustandsabfolge insbesondere beim Training von Modellen bei gegebenen Observationen benötigt. Ziel ist quasi die Aufdeckung der vormals als versteckt angenommenen Hintergrundsequenz.

   Als problematisch erweist sich hier jedoch die Tatsache, dass im Gegensatz zu obigem Ansatz keine exakte Lösung existiert. Die Fragestellung welche hieraus erwächst ist daher, welches Optimalitätskriterien angestrebt werden soll. Eine Möglichkeit bestünde in der Wahl der Zustände q_t, welche individuell die Produktionswahrscheinlichkeit einer gegebenen Observation maximiert.

   Das am häufigsten verwendete Qualitätskriterium ist der wahrscheinlichste Gesamtpfad durch die Trellisstruktur. In der Praxis hat sich zur Findung des Pfades der

Viterbi-Algorithmus[3] [For73] als Näherungslösung aufgrund des minimalen Berechnungsaufwandes durchgesetzt. Annahme hierbei ist es, dass nur eine Zustandssequenz einen wesentlichen Beitrag zur Gesamtproduktionswahrscheinlichkeit beiträgt. In diesem Fall braucht von vorhergehendem Problem ausgehend nur noch der Pfad in Betracht gezogen werden, welcher die höchste Produktionswahrscheinlichkeit aufweist, die sogenannte Single-Best-State-Sequence. In Abbildung B.3 ist ein möglicher bester Pfad durch die Zustandsfolge Q^* hervorgehoben. Interessant sind hierbei entgegen der Berechnungen des Forward-Algorithmus jedoch nicht die absoluten Produktionswahrscheinlichkeiten, sondern lediglich die Zustände, welche beim Durchlaufen besucht wurden.

Im Falle von strikt Linearen Modellen nach Kapitel 2.7.1, entwickelt der Viterbi-Algorihmus eine starke Ähnlichkeit zum oben erwähnten DTW-Ansatz, da nur noch Selbstübergänge und Sprünge in den nächsten Zustand erlaubt sind. Eine exakte Formulierung des Viterbi-Algorithmus kann dem Anhang B.4.4 entnommen werden.

3. **Die optimale Anpassung der Modellparameter für ein gegebenes Signal**

   Das komplexeste der drei Problematiken befasst sich mit dem Training bzw. der Findung des optimalen Parametersatzes $\lambda = (\mathbf{A}, \mathbf{b})$ anhand einer gegebenen Menge von Stichproben, hier also Übergangswahrscheinlichkeiten und Ausgabewahrscheinlichkeiten.

   Ähnlich wie bei den vorgestellten NN existiert auch beim Training von HMM keine analytische Methode zur Berechnung der Modellparameter. Prinzipiell könnten auch hier Gradientenverfahren Anwendung finden, in diesem statistischen Kontext wird jedoch hauptsächlich ein effizientes, an HMM angepasstes, ebenfalls iteratives Verfahren, basierend aus einer Kombination der Baum-Welch-Methode und dem EM-Algorithmus, angewendet [Bil97]. Da bei diesem Verfahren die Auftrittswahrscheinlichkeit $P(\mathbf{O}|\lambda)$ der Daten bzgl. der Modelle maximiert wird, spricht man an dieser Stelle von einem Maximum-Likelihood (ML) Training.

   Prinzipiell ist der Baum-Welch-Algorithmus eine Erweiterung aus dem bereits vorgestellten Forward-Algorithmus und dem analog funktionierenden Backward-Algorithmus, welche zusammen genommen den Forward-Backward-Algorithmus bilden. Beide Algorithmen sind in den Anhängen B.4.3 und B.4.5 angegeben.

   Ausgehend von einem vorgegebenen Initialmodell λ wird während eines Lernzyklus ein verbessertes Modell $\bar{\lambda}$ geschätzt, von dem Baum et al. gezeigt haben, dass das neue Modell bezüglich Likelihood in der Regel besser, auf keinen Fall aber schlechter als das Startmodell ist: $P(\mathbf{O}|\lambda) \geq P(\mathbf{O}|\bar{\lambda})$ [Rab89]. Nach mehreren Trainingsiterationen wird sich der Wert für die Ähnlichkeit nicht mehr signifikant ändern. Dieses Konvergenzverhalten kann mit Hilfe einer gegebenen Schranke ϵ als Abbruchkriterium definiert werden. Eine Einschränkung dieses Verfahrens liegt jedoch in der Tatsache, dass

[3] In der Literatur auch als Maximum-Likelihood-Algorithmus

die Parameter nur lokal optimiert und nicht global maximiert werden, was in realen Anwendungen allerdings vernachlässigt werden kann.

Neben diesem klassischen Ansatz zur Parameterschätzung haben sich zudem noch Adaptionsansätze etabliert, welche bereits trainierte Modelle an gegebene Besonderheiten der Merkmale anpassen [Wal00, Bra01, Wal01d]. So ist beispielsweise eine Sprecher- oder Schreiberadaption mit einem begrenztem Rechenaufwand möglich. Außerdem hat es sich gezeigt, dass diskriminative Lernparadigmen, auch als Maximum Mutual Information (MMI) Kriterien bekannt, dem ML-Kriterium oftmals überlegen sind.

B.4.2 Forward-Algorithmus

Bei der Berechnung der Zwischenergebnisse wird jeweils davon ausgegangen, dass ein Folgezustand nur vom vorherigen Zustand abhängt, was einem Markov Prozess erster Ordnung entspricht. Zur Berechnung der Vorwärtswahrscheinlichkeit wird zunächst die Vorwärts-Variable $\alpha_t(i)$ definiert, die Wahrscheinlichkeit für einen Teil der Observationsfolge O, genauer $o_1 o_2 \dots o_t$ für ein gegebenes Modell λ zur Zeit t im Zustand S_i:

$$\alpha_t(i) = P(o_1 o_2 \dots o_t, q_t = S_i|\lambda) \tag{B.22}$$

Im ersten Schritt werden alle $\alpha_1(i)$ initialisiert. Danach werden alle weiteren $\alpha_{t+1}(j)$ nach einer Induktionsvorschrift berechnet. Zum Schluss, wenn alle Zustände durchlaufen sind, wird eine Summierung aller $\alpha_T(i)$ durchgeführt. Diese Summe entspricht der gesuchten Wahrscheinlichkeit $P(O|\lambda)$.

1. Initialisierung:
$$\alpha_1(i) = \pi_i b_i(o_1), \quad 1 \leq i \leq N \tag{B.23}$$

2. Induktion:
$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(j) a_{ij}\right] b_j(o_{t+1}), \quad 1 \leq t \leq T-1, \quad 1 \leq j \leq N \tag{B.24}$$

3. Abbruch:
$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i) \tag{B.25}$$
$$P(O|\lambda) = \alpha_T(N) \text{ wenn nur der Endzustand } N \text{ in Frage kommt} \tag{B.26}$$

Durch diesen Algorithmus kann der Rechenaufwand für die Berechnung von $P(O|\lambda)$ gegenüber einer vollständigen Berechnung drastisch reduziert werden.

B.4.3 Backward-Algorithmus

Der Backward-Algorithmus erweist sich prinzipiell als analog zum bereits vorgestellten Forward-Algorithmus, so dass hier zunächst die Rückwärts-Variable $\beta_t(i)$ eingeführt wird, welche die Wahrscheinlichkeit einer partiellen Observationssequenz $o_{t+1} o_{t+2} \dots o_T$ für ein bestimmtes HMM zum Zeitpunkt t im Zustand S_i beschreibt. Die Größe $\beta_t(i)$ wird hier ebenfalls induktiv ermittelt. Die Wahl von $\beta_T(i)$ zu Beginn kann willkürlich erfolgen.

1. Initialisierung:
$$\beta_T(i) = 1, \quad 1 \leq i \leq N \tag{B.27}$$

2. Induktion:
$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(o_{t+1}) \beta_{t+1}(j), \quad t = T-1, T-2, \ldots 1, \quad 1 \leq i \leq N \tag{B.28}$$

B.4.4 Viterbi-Algorithmus

Der Viterbi-Algorithmus bietet die am häufigsten eingesetzte Lösung zur Findung der besten Zustandssequenz, der *Single Best State Sequence*, und kann mit Methoden der dynamischen Programmiertechniken effizient gelöst werden. Um die beste Zustandsfolge $Q = \{q_1 q_2 \ldots q_T\}$ für eine gegebene Beobachtungsfolge $O = \{o_1 o_2 \ldots o_T\}$ zu finden wird zunächst die Größe $\delta_t(i)$ definiert zu:

$$\delta_t(i) = \max_{q_1, q_2, \ldots, q_{(t-1)}} P[q_1, q_2, \ldots, q_{(t-1)} = i, o_1 o_2 \ldots o_t | \lambda] \tag{B.29}$$

Durch die unten angegebene vollständige Induktionsvorschrift erhält man eine Zustandssequenz mit maximaler Wahrscheinlichkeit, deren Weg über das Array $\psi_t(j)$ verfolgt werden kann.

1. Initialisierung:
$$\delta_1(i) = \pi_i b_i(o_1), \quad 1 \leq i \leq N \tag{B.30}$$
$$\psi_1(i) = 0, \quad 1 \leq i \leq N \tag{B.31}$$

2. Rekursion:
$$\delta_t(j) = \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] b_i(o_t), \quad 2 \leq t \leq T \quad 1 \leq j \leq N \tag{B.32}$$
$$\psi_t(j) = \operatorname*{argmax}_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}], \quad 2 \leq t \leq T \quad 1 \leq j \leq N \tag{B.33}$$

3. Abbruch:
$$P^* = \max_{1 \leq i \leq N} [\delta_T(i)] \tag{B.34}$$
$$q_T^* = \operatorname*{argmax}_{1 \leq i \leq N} [\delta_T(i)] \tag{B.35}$$

4. Backtracking:
$$q_t^* = \psi_{t+1}(q_{t+1}^*), \quad t = T-1, T-2, \ldots, 1 \tag{B.36}$$

B.4.5 Baum-Welch-Methode

Die Baum-Welch-Methode dient zur iterativen Schätzung eines verbesserten Modells $\overline{\lambda}$ aus gegebenem Trainingsmaterial und einem gegebenen Modell λ. Das durch Konvergenz des Ähnlichkeitsmaßes $P(O|\overline{\lambda})$ gefundene Maximum muss nicht unbedingt das globale Maximum sein, es kann sich auch um ein lokales Optimum handeln.

Die Bestimmung der HMM-Parameter kann unter Verwendung des Forward-Backward-Algorithmus und den Verbundwahrscheinlichkeiten $\xi_t(i,j) = \frac{\alpha_t(i)a_{ij}b_i(o_{t+1})\beta_{t+1}(j)}{P(O|\lambda)}$ und $\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j)$ wie folgt formuliert werden:

$$\overline{\overrightarrow{\pi}}_i = \text{erwartete Verweildauer in State } S_1 \text{ zum Zeitpunkt } (t=1) = \gamma_1(i) \tag{B.37}$$

$$\overline{a}_{ij} = \frac{\text{erwartete Anzahl von Sprüngen von } S_i \text{ nach } S_j}{\text{erwartete Sprünge aus } S_i} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \tag{B.38}$$

$$\overline{b}_t(k) = \frac{\text{erwartete Verweildauer in } S_j \text{ unter der Beobachtung } v_k}{\text{erwartete Verweildauer im Zustand } S_j} = \frac{\sum_{\substack{t=1 \\ o_t=v_k}}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)} \tag{B.39}$$

B.5 Boosting-Algorithmus

- Sammlung von Positiv- und Negativbeispielen mit den korrespondierenden Labeln: $(x_1, y_1), \ldots, (x_N, y_N)$, wobei $y_i = 1$ bei einem Ausschnitt mit Gesicht, $y_i = 0$ andernfalls
- Initialisierung der Gewichte $w_{1,i} = \frac{1}{2m}, \frac{1}{2l}$ für $y_i \in [o, 1]$ wobei m und l durch die Anzahl der Positiv- und Negativbeispiele , n durch deren Summe gegeben ist
- Wiederhole für die Iterationsschritte $t = 1, \ldots, T$:
  1. Normalisierung der Gewichte:
     $$w_{t,i} := \frac{w_{t,i}}{\sum_{j=1}^{N} w_{t,j}}$$
     so dass w_t einer Wahrscheinlichkeitsverteilung entpsricht
  2. Training eines trivialen Klassifikators auf Basis von Schwellwerten h_j basierend auf einem einzigen Merkmal j, wobei der entstehende Fehler bezüglich w_t zu
     $$\epsilon_j = \sum_i w_i |h_j(x_i) - y_i)|$$
     angegeben wird

3. Wähle den Klassifikator h_t mit dem geringsten Fehler ϵ_t
4. Erneuerung der Gewichte:

$$w_{t+1,i} = w_{t,i}\beta_t^{1-e_i}$$

wobei bei einer korrekten Klassifikation $e_i = 0$ gilt, sonst $e_i = 1$ and $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$

- Der gesamte resultierende Klassifikator ergibt sich durch:

$$h(x) = \begin{cases} 1 & \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & \text{sonst} \end{cases}$$

wobei gilt $\alpha_t = \log\left(\frac{1}{\beta_t}\right) = \log\left(\frac{1-\epsilon_t}{\epsilon_t}\right)$

B.6 Retinex-Algorithmus

Das Retinex Verfahren ist ein Mitglied der Klasse der Mittelpunkts/Umgebungsfunktionen, bei denen der Mittelpunkt über die Pixelintensität und die Umgebung über Gaußsche Funktionen definiert ist [Hin04]. Das Wort selbst ist aus der Kombination von Retina und Cortex abgeleitet, wodurch die Nähe zum biologischen visuellen Apparat angedeutet werden soll. Der Retinex Algorithmus dient der allgemeinen Bildverbesserung. Dieser wird durch eine nicht lineare räumliche und spektrale Transformation zur Aufbereitung lokaler Kontraste unter Vernachlässigung der im Bild vorhandenen Beleuchtungsintensität erreicht.

Abbildung B.4: Originalaufnahme (links) und aufbereitete Version (rechts) eines Einstiegsszenarios.

Mathematisch ausgedrückt ist eine monochromatische, Single-Scale Retinex definiert durch

$$R(x_1, x_2) = \alpha\left(\log(I(x_1, x_2)) - \log(I(x_1, x_2) * F(x_1, x_2))\beta\right) \tag{B.40}$$

Dabei ist I das Eingangbild, R das Ausgangsbild und α und β sind Skalierungsfaktoren bzw. Offset-Parameter zur Transformation und Steuerung der Ausgabe der Logarithmusfunktion.

Das Gaußsche Filter F ist definiert durch

$$F(x_1, x_2) = \kappa \exp\left[-(x_1^2 + x_2^2)/\sigma^2\right] \tag{B.41}$$

Hierbei ist σ die Standardabweichung des Filters, welche die enthaltene Fläche reguliert. Der Normalisierungsparameter κ dient dazu, den Wert der Fläche unter der Gaußschen Kurve bei 1 zu halten. In einer Näherung kann die Intensität I durch das Produkt der Beleuchtungskomponente $i(x_1, x_2)$ und einer Reflektanzkomponente $\rho(x_1, x_2)$ ausgedrückt werden:

$$I(x_1, x_2) = i(x_1, x_2)\rho(x_1, x_2) \tag{B.42}$$

Unter Vernachlässigung der Parameter α und β kann Gleichung B.40 geschrieben werden als:

$$R(x_1, x_2) = \log\left(i(x_1, x_2)\rho(x_1, x_2)\right) - \log\left((i(x_1, x_2)\rho(x_1, x_2)) * F(x_1, x_2)\right) \tag{B.43}$$

Unter Voraussetzung einer langsamen Änderung des Beleuchtungseinflusses über die Szene kann angenommen werden, dass der Intensitätsverlauf lokal unabhängig, und in einer Näherung konstant ist: $i(x_1, x_2) = I_0$. Daher kann Gleichung B.43 umformuliert werden zu:

$$R(x_1, x_2) = \log\left(\frac{I_0\rho(x_1, x_2)}{I_0\rho(x_1, x_2) * F(x_1, x_2)}\right) \tag{B.44}$$

Obiger Ausdruck ist nach dem Kürzen somit unabhängig von der vorhandenen Intensität I_0. Zur Bestimmung der verbleibenden Funktionen wurden mehrere Ansätze vorgeschlagen. Im Zusammenhang der hier benötigten Bildoptimierung wird nur eine einstufige Transformation nach Funt [Fun04] verwendet.

B.7 Levenshtein Distanz

Im Folgenden ist der Algorithmus zur Messung der Ähnlichkeit zwischen zwei Zeichenketten angegeben. Der Abstand repräsentiert die Anzahl an Auslöschungen, Einfügungen sowie Substitutionen zur Überführung von s nach t. Je größer der Abstand, desto verschiedener die Strings. Im vorliegenden Fall sollen jedoch keine Zeichenketten auf Ebene von Literalen verglichen werden, es erfolgt vielmehr eine symbolische Abstraktion mit Personenindizes aus N-Besten Listen.

1. Setze n auf die Länge der ersten Liste s
   Setze m auf die Länge der zweiten Liste t
   Abbruch bei leeren Listen
   Erstellen einer Distanzmatrix d mit m Reihen und n Spalten

2. Initialisieren der ersten Reihe mit den Werten $0 \ldots n$
   Initialisieren der ersten Spalte mit den Werten $0 \ldots m$

3. $\forall(i = 1 \ldots n)$ /* Für alle Elemente von s */

4. $\forall(j = 1 \ldots m)$ /* Für alle Elemente von t */

5. $C = \begin{cases} 0 & \text{wenn } s[i] == t[j] \text{ /* keine Kosten */} \\ 1 & \text{sonst} \end{cases}$

6. Setzen des Elements $d[i, j]$ auf das Minimum von:

   (a) Das darüber liegende Element um eins erhöht: $d[i - 1, j] + 1$

   (b) Das linke Element um eins erhöht: $d[i, j - 1] + 1$

   (c) Das linke obere Element um eins erhöht plus die Kosten: $d[i - 1, j - 1] + C$

7. Nach dem Ablauf aller Iterationsschritte $3 \ldots 6$ befindet sich die berechnete Distanz im Element $d[n, m]$.

Abkürzungsverzeichnis

1DHMM	Eindimensionales Hidden Markov Modell
2DHMM	Zweidimensionales Hidden Markov Model
3DHMM	Dreidimensionale Hidden Markov Modelle
ADALINE	ADAptives LIneares NEtzwerk
AGMA	Automatische Generierung audiovisueller Metadaten im Kontext von MPEG-7
ASE	Automatische Spracherkennung
BIC	Bayesian Information Criterion
CCD	Bildsensor mit einem Charged Coupled Device
CMOS	Bildsensor mit einem Complementary Metall Oxyde Semiconductor
DCT	Diskrete Cosinus Transformation
DPW	Dynamic Plane Warping
DTW	Dynamic Time Warping
EER	Equal Error Rate
FACS	Facial Action Coding System
FAR	False Acceptance Rate
FGNET	Face and Gesture Network
FRR	False Rejectance Rate
FRVT	Facial Recognition Vendor Test (FRVT)
GMM	Gaußches Mixtur Modell
HMM	Hidden Markov Modell
IR	Infrarot
JPEG	Joint Photographic Expert Group
KLT	Kahunen-Loeve-Transformation
M4	MultiModal Meeting Manager
MIT	Massachusetts Institute for Technology
ML	Maximum-Likelihood
MLP	Multi-Layer Perzeptron
MMI	Maximum Mutual Information
MPEG	Moving Picture Experts Group
NCC	Normalized Color Coordinates
NIST	National Institute of Standards and Technology
NN	Künstliches neuronales Netz
OCR	Optical Character Recognition

ORL	Olivetti Research Laboratories
P2DHMM	Pseudo-zweidimensionales Hidden Markov Model
P3DHMM	Pseudo-dreidimensionale Hidden Markov Modelle
PCA	Principle Components Analysis, deutsch Hauptachsentransformation
RFID	Radio Frequency Identification (Device)
RGB	Bezeichnung eines Farbraumes mit den Komponenten Rot, Grün und Blau
ROC	Receiver Operating Characteristic
SAFEE	Security of Aircraft in the Future European Environment
SVM	Support Vector Machines
TSP	Traveling Salesman Problem
UV	Ultraviolett
VQ	Vektorquantisierer
WDF	Wahrscheinlichkeitsdichtefunktion

Symbolverzeichnis

$\mathbf{A}$	Zustandsübergangswahrscheinlichkeit (HMM)
b	Abstand zwischen Fläche und Ursprung
$b(.)$	allgemeine Produktionswahrscheinlichkeit einer Observation
B	Breite einer Matrix oder eines Bildes
$\mathbf{C}$	Kovarianzmatrix
d_{AA}	Augenabstand in Pixeln
d_{AM}	vertikaler Abstand zwischen Augen und Mund
$d_E(a,b)$	Abstand über Euklidische Distanz
$d_M(a,b)$	Abstand über Distanz nach Mahalanobis
E	Fehlerfunktion beim Training
$F(a)$	Aktivierungsfunktion
$G(\mathbf{x})$	Propagierungsfunktion
h_j	triviale Entscheidungsfunktion
H	Höhe einer Matrix oder eines Bildes
$\mathcal{H}$	allgemeine Hyperebene
i,j,k	Indizes und Laufvariablen
$i(t)$	Intensität der Bewegung
$ii(x,y)$	Integralbild an gegebener Koordinate
$\mathbf{I}_d$	Differenzbild nach Rauschelimination
$I(x,y)$	Intensität eines Grauwerts am Ort (x,y)
$I_{r,g,b}(u,v)$	Intensität eines Farbwerts am Ort (u,v)
$k(.)$	allgemeine Kernelfunktion
k	diskreter Zeitpunkt
K	Anzahl vorgegebener Klassen
l^*	Index der wahrscheinlichsten Klasse
L	Umfang des Inventars
$L_(u,v,\lambda)$	einfallende Luminanz
$m_{x,y}(t)$	Massenschwerpunkt/Bewegung
$\mathcal{M}$	allgemeines Ausgabealphabet
M,N	Dimension eines Vektors oder eines Feldes
$\mathcal{N}$	allgemeine Normalverteilung
o_j	aktuelle Netzausgangswerte
$P(.)$	allgemeine Wahrscheinlichkeit

$q_1, \ldots, q_T$	Zustandsfolge
$\mathcal{Q}$	Zustandsmenge eines Automaten
$\mathbf{r}_i$	Referenzvektor
$\mathbb{R}$	Ordnung eines Raumes
s	beliebiges Information tragendes Signal
$S_{r,g,b}(\lambda)$	spektrale Empfindlichkeit
$\{s_1, \ldots, s_N\}$	Menge diskreter Zustände
t_j	Zielwerte am Ausgang des Netzes
t	(diskrete) Zeit
T	Gesamtdauer
$\mathbf{U}$	Eigenvektormatrix
$\mathbf{w}$	Richtungsvektor einer Ebene (SVM)
$\mathbf{w}$	Gewichtungsvektor eines Neurons (NN)
w_0	Bias-Wert eines Neurons
$\mathbf{x}$	allgemeiner Datenvektor
$\bar{\mathbf{x}}$	Mittelwertvektor
$\tilde{\mathbf{x}}$	Rückprojektion eines Datenvektors
$\mathbf{x}_1, \ldots, \mathbf{x}_M$	Sequenz von Mustervektoren
$\mathbf{y}$	allgemeiner Merkmalsvektor
(x, y)	Koordinate in einem zweidimensionalen Feld
γ	Kopfneigung
Γ	Moment
Δ	Proportionalität zum Gradienten
$\Delta m_{x,y}(t)$	Änderung der Bewegung
η	Lernrate
Θ	allgemeine Entscheidungsschwelle
λ	Wellenlänge des Lichts
λ	allgemeine Bezeichnung für ein Modell (HMM)
λ_i	Eigenwerte bzw. Parameter der i-ten Klasse
Λ	Klasseninventar
ξ	Schlupfvariable
π	Besetzungswahrscheinlichkeit eines Anfangszustandes
$\sigma_{x,y}(t)$	Varianz der Bewegung
Σ	allgemeine Kovarianzmatrix
τ	Entscheidungsschwelle für Konfidenzmaße
Φ	Transformation
φ	Drehwinkel eines Gesichts gegenüber der Frontalansicht
$\mathbf{\Phi}_i$	mittelwertfreier Datenvektor
Ω	Klassenkennzeichnung
$(.)^T$	Transponierte eines Vektors
$\|.\|$	Betrag des Arguments
$(.)^{-1}$	Matrixinvertierung

Literaturverzeichnis

[All01] E. L. Allwein, R. E. Schapire und Y. Singer. "Reducing multiclass to binary: a unifying approach for margin classifiers." *Journal of Machine Learning Research*, 1, Seiten 113–141, 2001. 23

[Ars03] D. Arsić. "Eintwicklung eines Systems zur Detektion von Gesichtern." Studienarbeit am Lehrstuhl für Mensch-Maschine-Kommunikation, Technische Universität München, 2003. 53, 55

[Bar03] M. S. Bartlett, G. Littlewort, I. Fasel und J. R. Movellan. "Real time recognition of facial expressions: Development and applications to human computer interaction." *Computer Vision and Pattern Recognition*, 2003. 1, 114

[BB03] E. Bailly-Bailliere, S. Bengio, F. Bimbot, M. Hamouz, J. M. Josef Kittler, J. Matas, K. Messer, V. Popovici, F. Poree, B. Ruiz und J.-P. Thiran. "The BANCA Database and Evaluation Protocol." In *Fourth International Conference on Audio and Video-based Biometric Person Authentication (AVBA), Springer Verlag, LNCS 2688*, Seiten 625–638, 2003. 130

[Bel97] P. Belhumeur, J. Hespanha und D. Kriegman. "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection." *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 19, Nr. 7, Seiten 711–720, 1997. 130

[Bil97] J. Bilmes. "A Gentle Tutorial on the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models." Technischer Bericht TR-97-021, University of Berkeley, ICSI, 1997. 144

[Bir99] N. Birbaumer und R. Schmidt. *Biologische Psychologie. Vierte, vollständig überarbeitete und ergänzte Auflage*. Springer, Heidelberg, 1999. 6

[Bis95] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995. 12, 16, 18

[Bla02a] J. Blaesing. "Implementierung eines Systems zur Findung von Gesichtern und Gesichtsbereichen zur Personenerkennung." Diplomarbeit am Lehrstuhl für Technische Informatik an der Gerhard-Mercator-Universität Duisburg, 2002. 65

[Bla02b] V. Blanz, S. Romdhani und T. Vetter. "Face Identification across Different Poses and Illuminations with a 3D Morphable Model." In *Proc. of the 5th IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, Seiten 202–207. Washington, D.C., USA, May 2002. 69, 93

[Bla05] D. Blackburn, M. Bone und P. Phillips. "Face Recognition Vendor Test 2000." Online Bericht `http://www.frvt.org`, September 2005. 79, 108

[Bra01] A. Brakensiek, J. Rottland, F. Wallhoff und G. Rigoll. "Adaptation of an Address Reading System to Local Mail Streams." In *6th International Conference on Document Analysis and Recognition (ICDAR)*, Seiten 872–876. Seattle, USA, September 2001. 145

[Bun05] Bundesdruckerei. "ePass - Foto-Mustertafel." `http://www.bundesdruckerei.de/de/behoerde/epass/`, 20.09.2005 2005.

[Bur98] C. J. C. Burges. "A Tutorial on Support Vector Machines for Pattern Recognition." *Data Mining and Knowledge Discovery*, 2, Nr. 2, Seiten 121–167, 1998. 20, 21, 22

[Cha05] Y. Chang, R. Cutler, Z. Li, Z. Zhang, A. Acero und M. Turk. "Automatic Head-size Equalization in Panorama Images for Video Conferencing." In *Proceedings IEEE Intern. Conference on Multimedia and Expo (ICME)*. Amsterdam, The Netherlands, July 2005. 62

[Che98] L. F. Chen, H. Y. M. Liao, C. C. Han und J. C. Lin. "Why a statistics-based face recognition system should base its recognition on the pure face portion: A probabilistic decision-based proof." In *Proc. Symposium on Image, Speech, Signal Processing, and Robotics*, Seiten 225–230. The Chinese University of Hong Kong, September 1998. 70

[Che01] C. Cheng und S.-H. Lai. "An Integrated Approach to 3D Model Reconstruction from Video." In *IEEE Proceedings of the second Int. Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real Time Systems in Conjunction with the Int. Conference on Computer Vision*, Seiten 16–22. Vancouver, Canada, July 2001. 93

[Cor95] C. Cortes und V. Vapnik. "Support-Vector Networks." *Machine Learning*, 20, Nr. 3, Seiten 273–297, 1995. 20

[Dar72] C. Darwin. *The Expression of the Emotions in Man and Animals*. John Murry, London, 1872. 1, 109

[Dem60] W. N. Dember. *The psychology of Perception*. Holt, Rinehart & Winston, 1960. 5

[Eic98a] S. Eickeler, S. Müller und G. Rigoll. "Improved Face Recognition Using Pseudo-2D Hidden Markov Models." In *Workshop on Advances in Facial Image Analysis*

and Recognition Technology in conjunction with 5th European Conference on Computer Vision. Freiburg, Germany, Juni 1998. 73

[Eic98b] S. Eickeler, S. Müller und G. Rigoll. "Person-Independent Continuous Online Recognition of Gestures." In *Proceedings Intern. Conference on Computer Vision and Pattern Recognition (CVPR)*. Santa Barbara, USA, Juni 1998. Demo session. 114

[Eic00a] S. Eickeler, M. Jabs und G. Rigoll. "Comparison of Confidence Measures for Face Recognition." In *IEEE Int. Conference on Automatic Face and Gesture Recognition*, Seiten 257–262. Grenoble, France, März 2000. 85, 86

[Eic00b] S. Eickeler, S. Müller und G. Rigoll. "Recognition of JPEG Compressed Face Images Based on Statistical Methods." *Image and Vision Computing Journal, Special Issue on Facial Image Analysis*, 18, Nr. 4, Seiten 279–287, März 2000. 70, 73, 77

[Eic01] S. Eickeler, F. Wallhoff, U. Iurgel und G. Rigoll. "Content-Based Indexing of Images and Video Using Face Detection and Recognition Methods." In *IEEE Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. Salt Lake City , Utah, Mai 2001. 91

[Eic02] S. Eickeler. *Automatische Bildfolgenanalyse mit statistischen Mustererkennungsverfahren*. Doktorarbeit, Faculty of Electrical Engineering, Gerhard-Mercator-University Duisburg, 2002. 86, 95

[Ekm71] P. Ekman. "Universals and cultural differences in facial expressions of emotion." *In J. Cole (Ed.), Nebraska Symposium on Motivation; Lincoln, NE: University of Nebraska Press.*, 19, Seiten 207–283, 1971. 109

[Ekm74] P. Ekman und W. V. Friesen. "Detecting Deception from the Body or Face." *Journal of Personality and Social Psychology*, 29, Nr. 3, Seiten 288–298, 1974. 109

[Ekm78] P. Ekman und W. V.Friesen. "Facial action coding system: A technique for the measurement of facial movement." *Palo Alto, Calif.: Consulting Psychologists Press.*, 1978. 110, 113

[Fai90] G. Faigin. *Mimikzeichnen leichtgemacht*. Benedikt Taschen Verlag GmbH, 1990. 111

[Fis87] K. Fischer. *Bildkommunikation*. Springer Verlag, 1987. ISBN 3-540-16974-1. 1

[For73] G. D. Forney. "The Viterbi Algorithm." *Proceedings of the IEEE*, 61, Nr. 3, Seiten 268–278, März 1973. 144

[Fro97] T. Fromherz, P. Stucki und M. Bichsel. "A Survey of Face Recognition." MML Technical Report, No 97.01, Dept. of Computer Science, University of Zurich, 1997. 93

[Fun04] B. Funt, F. Ciurea und J. McCann. "Retinex in MATLAB." *Journal of Electronic Imaging*, 13, Nr. 1, Seiten 48–57, January 2004. 7, 82, 149

[Gau05] D. Gaultier. "Security of aircraft in the future European environment (SAFEE)." Project Homepage: `http://europa.eu.int/comm/research/aeronautics/info/news/article_681_en.html`, September 2005. 90

[Ger04] W. Gerstner. "Supervised Learning for Neural Networks: A tutorial with JAVA exercises." EPFL Graduate Volume on Itelligent Systems: `http://diwww.epfl.ch/mantra/tutorial/docs/supervised.pdf`, Dezember 2004. 17, 18

[Gon87] R. C. Gonzalez und P. Wintz. *Digital Image Processing, Second Edition*. Addison Wesley, 1987. 7, 45, 73

[Gor95] G. G. Gordon. "Face Recognition from Frontal and Profile Views." In *The International Workshop on Automatic Face and Gesture Recognition*. Zürich, 1995. 93

[Gra98] D. Graham und N. Allinson. "Characterizing Virtual Eigensignatures for General Purpose Face Recognition." *Face Recognition: From Theory to Applications, Editors: H. Wechsler and P.J. Phillips and V. Bruce and F. Fogelman-Soulie and T.S. Huang*, 163, Seiten 446–456, 1998. 130

[Gro94] M. Groß. *Visual Computing - The Integration of Computer Graphics, Visual Perception and Imaging*. Computer Graphics: Systems and Applications. Springer-Verlag Berlin Heidelberg, 1994. 93

[Gro01] R. Gross, J. Shi und J. Cohn. "Quo Vadis Face Recognition?" Technischer Bericht, Robotics Institute, Carnegie Mellon University, December 2001. 93, 108

[Hab91] P. Haberäcker. *Digitale Bildverarbeitung: Grundlagen und Anwendungen - 4., durchgesehene Auflage*. Carl Hanser Verlag München Wien, 1991. 8

[Hal95] P. Hallinan. *A Deformable Model for Face Recognition Under Arbitrary Lighting Conditions*. Doktorarbeit, Harvard University, 1995. 130

[Has98] T. Hastie und R. Tibshirani. "Classification by Pairwise Coupling." In M. I. Jordan, M. J. Kearns und S. A. Solla, Herausgeber, *Advances in Neural Information Processing Systems*, Band 10. The MIT Press, 1998. 24, 121

[Hau94] G. Hauske. *Systhemtheorie der visuellen Wahrnehmung.*. Teubner Verlag Stuttgart, 1994. 5, 6, 8

[Hea98] M. Hearst, B. Schölkopf, S. T. Dumais, E. Osuna und J. Platt. "Support Vector Machines." *IEEE Intelligent Systems*, Seiten 18–28, Juli 1998. 20, 53, 55

[Hei03] T. Heinrich. “Gesichtsverfolgung und Blickrichtungsschätzung mit Partikelfilter.” Studienarbeit am Lehrstuhl für Mensch-Maschine-Kommunikation, Technische Universität München, 2003. 61

[Hel76] T. Hellbrügge und J. H. von Wimpffen. *Die ersten 365 Tage im Leben eines Kindes*. Knaur, München, 1976. 1

[Hin04] G. D. Hines, Z. ur Rahman, D. J. Jobson und G. A. Woodell. “DSP Implementation of the Retinex Image Enhancement Algorithm.” In Z. ur Rahman, R. A. Schowengerdt und S. E. Reichenbach, Herausgeber, *Visual Information Processing XIII*, Seiten 13–24. SPIE, July 2004. Orlando, FL, USA. 148

[Hje01] E. Hjelmas und B. K. Low. “Face Detection: A Survey.” *Computer Vision and Image Understanding (CVIU)*, 83, Nr. 3, Seiten 236–274, September 2001. 36

[Hul01] F. Hülsken, F. Wallhoff und G. Rigoll. “Facial Expression Recognition with Pseudo-3D Hidden Markov Models.” In *23. DAGM-Symposium, Tagungsband Springer-Verlag*. Munich, Germany, September 2001. 116

[Isa98] M. Isard und A. Blake. “Condensation – conditional density propagation for visual tracking.” *International Journal of Computer Vision (IJCV)*, 29, Nr. 1, Seiten 5–28, 1998. 58, 59

[Iur02] U. Iurgel, S. Werner, A. Kosmala, F. Wallhoff und G. Rigoll. “Audio-Visual Analysis of Multimedia Documents for Automatic Topic Identification.” In *Signal Processing, Pattern Recognition, and Applications, ISBN 0-88986-338-5*, Seiten 550–555. Crete, Greece, Juni 2002. ACTA Press. 118

[Jai96] A. K. Jain, J. Mao und K. M. Mohiuddin. “Artificial Neural Networks: A Tutorial.” *IEEE Computer*, 29, Nr. 3, Seiten 31–44, März 1996. 16, 137

[Jai00] A. K. Jain, R. P. W. Duin und J. Mao. “Statistical Pattern Recognition: A Review.” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 22, Nr. 1, Seiten 4–37, Januar 2000. 9, 10

[Jel97] F. Jelinek. *Statistical methods for speech recognition*. MIT Press, 1997. 24

[Jes01] O. Jesorsky, K. Kirchberg und R. Frischholz. “Robust Face Detection Using the Hausdorff Distance.” In J. Bigun und F. Smeraldi, Herausgeber, *Audio and Video based Person Authentication - AVBPA 2001*, Seiten 90–95. Springer, 2001. 129

[JM99] J. Jones M., Rehg. “Statistical Color Models with Application to Skin Detection.” *Cambridge Research Laboratory, Computer Vision and Pattern Recognition (CVPR99), Ft.Collins, CO*, Seiten 274–280, June 1999. 37

[Jor97] M. I. Jordan und C. M. Bishop. “Neural Networks.” In A. Tucker, Herausgeber, *CRC Handbook of Computer Science*. Boca Ration, FL.: CRC Press, 1997. 13

[Koh05] J. Köhler. "Automatische Generierung audiovisueller Metadaten im Kontext von MPEG-7." Projektblatt: `http://www.imk.fraunhofer.de/sixcms/media.php/130/agma.pdf`, Februar 2005.

[Kuo94] S. S. Kuo und O. E. Agazzi. "Keyword spotting in poorly printed documents using pseudo 2-d hidden markov models." *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 16, Nr. 8, Seiten 842–848, August 1994. 30

[Lan95] A. Lanitis, C. J. Taylor und T. F. Cootes. "An automatic face identification system using flexible appearance models." *Image and Vision Computing*, 13, Nr. 5, Seiten 393–402, June 1995. 69

[Lav00] F. Lavagetto, R. Pockaj und M. Costa. "Smooth Surface Interpolation and Texture Adaptation for MPEG-4 Compliant Calibration of 3D Head Models." *Image and Vision Computing Journal, Special Issue on Facial Image Analysis*, 18, Nr. 4, Seiten 345–354, März 2000. 93

[Lie02] R. Lienhart, A. Kuranov und V. Pisarevsky. "Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection." Technischer Bericht, Microprocessor Research Lab, Intel Labs, May 2002. 51, 54

[Lip87] R. P. Lippmann. "An Introduction to Computing with Neural Nets." *IEEE Trans. Acoustics, Speech, and Signal Proc.*, 4, Seiten 4–22, April 1987. 20

[Lip89] R. P. Lippmann. "Pattern Classification Using Neural Networks." *IEEE Communications Magazines*, 27, Nr. 11, Seiten 47–50, 1989. 12

[Lou98] A. C. Loui, C. N. Judice und S. Liu. "An Image Database for Benchmarking of Automatic Face Detection and Recognition Algorithms." In *Proceedings IEEE Intern. Conference on Image Processing (ICIP)*, Seiten 146–150, 1998. 129

[Lu03] X. Lu. "Image Analysis for Face Recognition." Personal Notes: `http://diwww.epfl.ch/mantra/tutorial/docs/supervised.pdf`, Mai 2003. 69

[Man02] A. Mansfield und J. Wayman. "Best Practices in Testing and Reporting Performance of Biometric Devices." Technischer Bericht, National Physical Laboratory, Teddington, Middlesex, UK, August 2002. 68

[Mar98] A. Martinez und R. Benavente. "The AR Face Database." Technischer Bericht 24, Purdue University, 1998. 81, 82, 130

[Mar02a] A. Martinez. "Recognizing Imprecisely Localized, Partially Occluded and Expression Variant Faces from a Single Sample per Class." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24, Nr. 6, Seiten 748–763, 2002. 70

[Mar02b] B. Martinkauppi. *Face Colour Under Varying Illumination - Analysis and Applications*. Doktorarbeit, University of Oulu, 2002. 39, 41

[Mau95] T. Maurer und C. von der Malsburg. “Learning Feature Transformations to Recognize Faces Rotated in Depth.” In *Proceedings of the International Conference on Artificial Neural Networks*. Paris, October 1995. 93, 99, 108

[Mic03] P. Michel und R. E. Kaliouby. “Real time facial expression recognition in video using support vector machines.” In *Proceedings of the 5th international conference on Multimodal interfaces*, Seiten 258–264. ACM Press, Vancouver, British Columbia, Canada, 2003. 1-58113-621-8. 113, 123

[Mul99a] S. Müller, G. Rigoll, A. Kosmala und D. Mazurenok. “Combining Shape Matrices and HMMs for Hand-Drawn Pictogram Recognition.” In S.-W. Lee, Herausgeber, *Advances in Handwriting Recognition*, Kapitel 9, Seiten 519–528. World Scientific, 1999. 24

[Mul99b] S. Müller, F. Wallhoff, S. Eickeler und G. Rigoll. “Content-based Retrieval of Digital Archives Using Statistical Object Modeling Techniques.” In *Electronic Imaging & the Visual Arts*. Berlin, Germany, November 1999. 2

[Mul00] S. Müller, S. Eickeler und G. Rigoll. “Crane Gesture Recognition using Pseudo 3-D Hidden Markov Models.” In *IEEE Int. Conference on Automatic Face and Gesture Recognition*, Seiten 398–402. Grenoble, France, März 2000. 31

[Mul01a] K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda und B. Schölkopf. “An Introduction to Kernel-Based Learning Algorithms.” *IEEE Transactions on Neural Networks*, 12, Nr. 2, Seiten 181–201, 2001. 20

[Mul01b] S. Müller, F. Wallhoff und G. Rigoll. “Retrieval of Overlapping and Touching Objects Using Hidden Markov Models.” In *IEEE Int. Conference on Image Processing (ICIP)*. Thessaloniki, Greece, Oktober 2001. 24

[Mul02a] S. Müller. *Segmentierung und Klassifizierung von Bildern und Bildsequenzen mit Hidden-Markov-Modellen*. Doktorarbeit, Faculty of Electrical Engineering, Gerhard-Mercator-University Duisburg, 2002. 29, 31

[Mul02b] S. Müller, F. Wallhoff, F. Hülsken und G. Rigoll. “Facial Expression Recognition Using Pseudo 3-D Hidden Markov Models.” In *16th Int. Conference on Pattern Recognition (ICPR)*. Quebec, Canada, August 2002. 32, 116

[Mor04] P. Morguet, C. Narr, H. Lorch, F. Wallhoff und G. Rigoll. “Reconstruction-Free Matching for Fingerprint Sweep Sensors. Tagungsband.” *Proceedings IEEE Intern. Conference on Image Processing (ICIP)*, Oktober 2004. 25

[Mos94] Y. Moses, Y. Adini und S. Ullman. “Face Recognition: the Problem of Compensating for Changes in Illumination Direction.” In *European Conference on Computer Vision (ECCV)*, Seiten 286–296, 1994. 67

[Nef99] A. Nefian. *A hidden Markov model-based Approach for Face Detection and Recognition*. Doktorarbeit, Georgia Institute of Technology, August 1999. 73

[Neu99] C. Neukirchen. *Integration neuronaler Vektorquantisierer in ein Hidden-Markov-Modell-basiertes System zur automatischen Spracherkennung*. Doktorarbeit, Faculty of Electrical Engineering, Gerhard-Mercator-University Duisburg, 1999. 28, 29

[Neu01] C. Neukirchen, J. Rottland, D. Willett und G. Rigoll. "A Continuous Density Interpretation of Discrete HMM Systems and MMI-Neural Networks." In *IEEE Transactions on Speech and Audio Processing*, May 2001. 32, 75

[Nie03] H. Niemann. "Klassifikation von Mustern, 2. überarbeitete Auflage." Internet Publikation: `http://www5.informatik.uni-erlangen.de/MEDIA/nm/klassifikation-von-mustern/m00-www.pdf`, Juli 2003. 10, 20, 23, 24

[O'R92] J. O'Regan. "Mysteries of Visual Perception: The World as an Outside Memory." *Canadian Journal of Psychology*, 46, Seiten 461–488, 1992. 93

[Ost98] J. Ostermann. *Animation of Synthetic Faces in MPEG-4*, Seiten 49–51. Proceedings of the Computer Animation. IEEE Computer Society, June 1998. ISBN 0-8186-8541-7. 43, 69, 131

[Pan03] M. Pantic und L. Rothkrantz. "Towards an Affect-sensitive Multimodal Human-Computer Interaction." *Proceedings of the IEEE*, 91, Nr. 9, Seiten 1370–1390, September 2003. 110, 123

[Pap98] C. P. Papageorgiou, M. Oren und T. Poggio. "A General Framework for Object Detection." In *Proceedings of International Conference on Computer Vision*. Bombay, India, January 1998. 51

[Pat96] D. Patterson. *Künstliche neuronale Netze - Das Lehrbuch*. Prentice Hall, 1996. 12, 18, 103

[Pet02] T. Petermann und A. Sauter. "Biometrische Identifikationssysteme - Sachstandsbericht." Technischer Bericht, Büro für Technikfolgen-Abschätzung beim Deutschen Bundestag (TAB), 2002. 88

[Phi00] P. Phillips, H. Moon, S. Rizvi und P. Rauss. "The FERET Evaluation Methodology for Face-Recognition Algorithms." *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 22, Nr. 10, Seiten 1090–1034, Oktober 2000. 67, 78, 79, 80, 130

[Pig97] S. Pigeon und L. Vandendrope. "The M2VTS Multimodal Face Database." In *Proceedings on First International Conference on Audio- and Video-Based Biometric Person Authentication*, 1997. 130

[Pla98] J. Platt. “Fast Training of Support Vector Machines using Sequential Minimal Optimization.” In B. Schoelkopf, C. Burges und A. Smola, Herausgeber, *Advances in Kernel Methods - Support Vector Learning*. MIT Press., 1998. 121

[Pre92] W. H. Press, B. P. Flannery, S. A. Teukolsky und W. T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing, 2nd Edition*. Cambridge University Press, UK, 1992. 136

[Rab89] L. R. Rabiner. “A Tutorial on HMM and Selected Applications in Speech Recognition.” *Proceedings of the IEEE*, 77, Nr. 2, Seiten 257–286, Februar 1989. 142, 144

[Raj99] Y. Raja, S. McKenna und S. Gong. “Tracking Color Objects Using Adaptive Mixture Models.” *Image and Vision Computing Journal (IVCJ)*, 17, Nr. 3–4, Seiten 225–232, 1999. 38

[Ren05] S. Renals. “The MultiModal Meeting Manager (M4).” Project Homepage `http://www.dcs.shef.ac.uk/spandh/projects/m4/`, September 2005. 56

[Rie93] M. Riedmiller und H. Braun:. “A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm.” *Proceedings IEEE Intern. Conference on Neural Networks (ICNN)*, Seiten 586–591, April 1993. 17, 141

[Rig94] G. Rigoll. *Neuronale Netze - Eine Einführung für Ingenieure, Informatiker und Naturwissenschaftler*. Kontakt und Studium. Expert Verlag, Renningen-Malmsheim, 1994. 13, 16

[Rig96] G. Rigoll. *Neuroinformatik I*. Vorlesungsmanuskript, Gerhard Mercator Universität Duisburg, 1996. 16, 17

[Rig98] G. Rigoll, A. Kosmala und D. Willett. “An Investigation of Context-Dependent and Hybrid Modeling Techniques for Very Large Vocabulary On-Line Cursive Handwriting Recognition.” In *6th Int. Workshop on Frontiers in Handwriting Recognition (IWFHR)*. Taejon, Korea, August 1998. 24

[Rig02] G. Rigoll. “Combination of Hidden Markov Models and Neural Networks for Hybrid Statistical Pattern Recognition.” In H. Bunke und A. Kandel, Herausgeber, *Hybrid Methods in Pattern Recognition*, Band 47 von *Series in Machine Perception and Artificial Intelligence*, Seiten 113–144. World Scientific Publishing, Mai 2002. ISBN 981-02-4832-6. 32

[Rig04a] G. Rigoll. *Manuskript vor Vorlesung Pattern Recognition*. Technische Universität München, Fakultät für Elektro- und Informationstechnik, Lehrstuhl für Mensch-Maschine-Kommunikation, 2004. 9, 11

[Rig04b] G. Rigoll, H. Breit und F. Wallhoff. “Robust Tracking of Persons in Real-World Scenarios Using a Statistical Computer Vision Approach.” *Image and Vision Computing Journal (IVCJ)*, 22, Nr. 7, Seiten 571–582, Mai 2004. 33, 58

[Rot00] J. Rottland und G. Rigoll. "Tied Posteriors: An Approach for Effective Introduction of Context Dependency in Hybrid NN/HMM LVCSR." In *IEEE Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. Istanbul, Turkey, Juni 2000. 32

[Row98] H. Rowley, S. Baluja und T. Kanade. "Neural Network-Based Face Detection." *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 20, Nr. 1, Seiten 23–38, Januar 1998. 47, 48, 55, 129

[Row99] H. A. Rowley. *Neural Network-Based Face Detection*. Doktorarbeit, School of Computer Science, Carnegie Mellon University, Mai 1999. 45

[Rus03a] G. Ruske. *Skriptum zur Vorlesung Datenanalyse und Informationsreduktion*. Technische Universität München, Fakultät für Elektro- und Informationstechnik, Lehrstuhl für Mensch-Maschine-Kommunikation, 2003. 9, 10

[Rus03b] G. Ruske. *Skriptum zur Vorlesung Digitale Verarbeitung von Sprachsignalen*. Technische Universität München, Fakultät für Elektro- und Informationstechnik, Lehrstuhl für Mensch-Maschine-Kommunikation, 2003. 9

[Sam94] F. Samaria. *Face Recognition Using Hidden Markov Models*. Doktorarbeit, University of Cambridge, 1994. 30, 69, 72, 77, 78, 130

[San03] C. Sanderson und K. K. Paliwal. "Fast Features for Face Authentication Under Illumination Direction Changes." *Pattern Recognition Letters*, 24, Seiten 2409–2419, 2003. 83

[San04] C. Sanderson und S. Bengio. "Statistical Transformations of Frontal Models for Non-Frontal Face Verification." In *Proceedings IEEE Intern. Conference on Image Processing (ICIP)*, Seiten 585–588, 2004. 83, 93

[Sch97] R. E. Schapire, Y. Freund, P. Bartlett und W. S. Lee. "Boosting the margin: A new explanation for the effectiveness of voting methods." In *Proceedings of the Fourteenth International Conference on Machine Learning*, 1997. 52

[Sch00] H. Schneiderman und T. Kanade. "A Statistical Method for 3D Object Detection Applied to Faces and Cars." In *Proceedings Intern. Conference on Computer Vision and Pattern Recognition (CVPR)*, Seiten 746–751, 2000. 129

[Sch02] B. Schölkopf und A. Smola. *Learning with Kernels-Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press, Cambridge, MA, 2002. 20, 21, 22

[Sch04] B. Schuller, G. Rigoll und M. Lang. "Speech Emotion Recognition Combining Acoustic Features and Linguistic Information in a Hybrid Support Vector Machine - Belief Network Architecture." In *Proceedings IEEE Intern. Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Band 1, Seiten 577–580, 2004. Montreal, Quebec, Kanada. 24, 123

[Sch06] B. Schuller. *Automatische Emotionserkennung in der sprachlichen und manuellen Interaktion*. Doktorarbeit, Technische Universität München, Lehrstuhl für Mensch-Maschine-Kommunikation, 2006. 121, 123

[Sim03] T. Sim, S. Baker und M. Bsat. "The CMU Pose, Illumination, and Expression Database." *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 25, Nr. 12, Seiten 1615–1618, Dezember 2003. 130

[Sir87] L. Sirovich und M. Kirby. "Low-dimensional Procedure for the Characterization of Human Faces." *Journal of the Optical Society of America*, 4, März 1987. 46

[Ska93] W. Skarbek und A. Koschan. "Colour Image Segmentation – A Survey." Technischer Bericht, Technische Universität Berlin, Fachbereich 13 Informatik, 1993. 37

[Sor00] M. Soriano, S. Huovinen, B. Martinkauppi und M. Laaksonen. "Skin Detection in Video under Changing Illumination Conditions." In *Proc. 15th International Conference on Pattern Recognition, Barcelona, Spain*, Seiten 839–842, 2000. 40

[ST95] E. G. Schukat-Talamazzini. *Automatische Spracherkennung – Grundlagen, statistische Modelle und effiziente Algorithmen*. Künstliche Intelligenz. Vieweg, Braunschweig, 1995. 24

[Sto99] M. Störring, H. J. Andersen und E. Granum. "Skin Colour Detection Under Changing Lighting Conditions." In *Proceedings on 7th Symposium on Intelligent Robotics Systems in Coimbra, Portugal*, Seiten 187–195, Juli 1999. 39

[Sto04] M. Störring. *Computer Vision And Human Skin Colour*. Doktorarbeit, Faculty of Engineering and Science, Aalborg University, 2004. 37, 39

[Ste93] R. Steinbrecher. *Bilderverarbeitung in der Praxis*. R. Oldenbourg Verlag, München, 1993. 5, 6, 9, 62, 64

[Sun98] K.-K. Sung und T. Poggio. "Example-Based Learning for View-Based Human Face Detection." *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 20, Nr. 1, Seiten 39–51, Januar 1998. 44, 49, 54, 129

[Svo99] T. Svoboda. *Central Panoramic Cameras Design, Geometry, Egomotion*. Doktorarbeit, Center for Machine Perception, Faculty of Electrical Engineering, Czech Technical University, September 1999. 62

[Tri99] A. Tritschler und R. Gopinath. "Improved Speaker Segmentation and Segments Clustering Using the Bayesian Information Criterion." In *Proc. EUROSPEECH*, Band 2, Seiten 679–682. Paris, France, 1999. 117

[Tur91a] M. Turk und A. Pentland. "Eigenfaces for Recognition." *Journal of Cognitive Neuroscience*, 3, Nr. 1, Seiten 71–86, 1991. 46, 69, 130

[Tur91b] M. Turk und A. Pentland. "Face Recognition using Eigenfaces." In *Proceedings Intern. Conference on Computer Vision and Pattern Recognition (CVPR)*, Seiten 586–591, Juni 1991. 11, 136

[Val97] D. Valentin, H. Abdi und B. Edelman. "What Represents a Face : A Computational Approach for the Integration of Physiological and Psychological Face Data." In *Perception: The effects of distinctiveness in recognising and classifying faces.*, Band 26, Seiten 525–536, 1997. 93, 95

[Vet98] T. Vetter. "Synthesis of Novel Views from a Single Face Imge." *International Journal of Computer Vision (IJCV)*, 28, Nr. 2, Seiten 103–116, 1998. 108

[Vio01] P. Viola und M. Jones. "Robust Real-time Object Detection." In *Proceedings to the 2nd Intl. Workshop on Statistical And Computational Theories of Vision - Modeling, Learning, Computing, And Sampling in Conjunction with the ICCV*, Seiten 1–25. Vancouver, Canada, July 2001. 51, 55

[Wal00] F. Wallhoff, D. Willett und G. Rigoll. "Frame Discriminative and Confidence-Driven Adaptation for LVCSR." In *IEEE Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Seiten 1835–1838. Istanbul, Turkey, Juni 2000. 96, 145

[Wal01a] F. Wallhoff, S. Eickeler und G. Rigoll. "A Comparison of Discrete and Continuous Output Modeling Techniques for a Pseudo-2D Hidden Markov Model Face Recognition System." In *IEEE Int. Conference on Image Processing (ICIP)*. Thessaloniki, Greece, Oktober 2001. 79

[Wal01b] F. Wallhoff, S. Müller und G. Rigoll. "Hybrid Face Recognition Systems for Profile Views Using The MUGSHOT Database." In *Workshop RAT-FG in conjunction with the IEEE Int. Conference on Computer Vision (ICCV)*. Vancouver, Canada, Juli 2001. 96

[Wal01c] F. Wallhoff und G. Rigoll. "A Novel Hybrid Face Profile Recognition System Using The FERET And MUGSHOT Databases." In *IEEE Int. Conference on Image Processing (ICIP)*. Thessaloniki, Greece, Oktober 2001. 96

[Wal01d] F. Wallhoff, D. Willett und G. Rigoll. "Scaled Likelihood Linear Regression for Hidden Markov Model Adaptation." In *European Conference on Speech Communication and Technology*. Aalborg, Denmark, September 2001. 145

[Wal02] F. Wallhoff. "FGnet 2nd Forsight Report: Observing Interacting People, Martigny, Schweiz." Face and Gesture Recognition Working Group Online-Tagungsbericht: `http://www-prima.inrialpes.fr/FGnet/html/surveys.html`, September 2002. 2

[Wal03a] F. Wallhoff. "FGnet 3rd Forsight Report: Human Machine Interaction, Limassol, Zypern." Face and Gesture Recognition Working Group Online-Tagungsbericht:

http://www-prima.inrialpes.fr/FGnet/html/surveys.html, August 2003. 2

[Wal03b] F. Wallhoff und G. Rigoll. "Synthesis and Recognition of Face Profiles." In R.Westermann, E. Steinbach, T. Ertl, H. Niemann und G. Greiner, Herausgeber, *Tagungsband 8th Intern. Fall Workshop Vision, Modelling, and Visualization (VMV) 2003, München*, Seiten 545–55. Aka Verlag, Berlin, IOS Press, Amsterdam, November, 2003. 99, 103

[Wal04a] F. Wallhoff, M. Zobl und G. Rigoll. "Action Segmentation And Recognition in Meeting Room Scenarios." *Proceedings IEEE Intern. Conference on Image Processing (ICIP)*, Oktober 2004. 61, 114, 117

[Wal04b] F. Wallhoff, M. Zobl, G. Rigoll und I. Potucek. "Face Tracking in Meeting Room Scenarios Using Omnidirectional Views." *Proceedings Intern. Conference on Pattern Recognition (ICPR)*, August 2004. 61, 64

[Wal05] F. Wallhoff. "Homepage of the Facial Expressions and Emotion Database (FEEDTUM)." Database Homepage http://www.mmk.ei.tum.de/~waf/fgnet/feedtum.html, September 2005. 112

[Wat94] C. I. Watson. "Mugshot Identification Data - Fronts and Profiles." *Reference Data of NIST Special Database 18*, Dezember 1994. 94, 95, 130

[Wil98] D. Willett, A. Worm, C. Neukirchen und G. Rigoll. "Confidence Measures for HMM-based Speech Recognition." In *5th International Conference on Spoken Language Processsing (ICSLP)*, Seiten 3241–3244. Sydney, Dezember 1998. 85

[Wis97] L. Wiskott, J.-M. Fellous, N. Krüger und C. von der Malsburg. "Face Recognition by Elastic Bunch Graph Matching." *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 19, Nr. 7, Seiten 775–779, 1997. 69

[Wür94] R. Würz. *Multilayer dynamic link networks for establishing image point correspondences and visual object recognition*. Doktorarbeit, Bochum University im Verlag Harri Deutsch, 1994. 93

[Yal00] I. K. Yalcin, A. T. Kilinc, S. Müller und G. Rigoll. "Gesture Recognition Using Pseudo 3D Hidden Markov Models." In *22. DAGM-Symposium, Tagungsband Springer-Verlag*. Kiel, Germany, September 2000. 31, 116

[Yan99] M. H. Yang und N. Ahuja. "Gaussian Mixture Model for Human Skin Color and Its Application in Image and Video Databases." In *Proc. of the Conf. on Storage and Retrieval for Image and Video Databases (SPIE), San Jose*, Band 3656, Seiten 458–466, Januar 1999. 37

[Yan02] M.-H. Yang, D. Kriegman und N. Ahuja. "Detecting Faces in Images: A Survey." *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 24, Nr. 1, Seiten 34–58, Januar 2002. 35, 55

[Zha98] Z. Zhang, M. Lyons, M. Schuster und S. Akamatsu. "Comparison Between Geometry-Based and Gabor-Wavelets-Based Facial Expression Recognition Using Multi-Layer Perceptron." In *Third IEEE International Conference on Automatic Face and Gesture Recognition*, Seiten 454–459. IEEE Computer Society, April 1998. 114, 121

[Zha03] W. Zhao, R. Chellappa, P. Phillips und A. Rosenfeld. "Face Recognition: A Literature Survey." *ACM Computing Surveys (CSUR)*, 35, Nr. 35, Seiten 339–458, December 2003. 69

[Zob03] M. Zobl, F. Wallhoff und G.Rigoll. "Action Recognition in Meeting Scenarios using Global Motion Features." In J. Ferryman, Herausgeber, *Proceedings of the Fourth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS-ICVS), Graz, Österreich*, Seiten 32–36. University of Reading, UK, März 2003. 117

[Zob04] M. Zobl, A. Laika, F. Wallhoff und G. Rigoll. "Recognition of Partly Occluded Person Actions in Meeting Scenarios." *Proceedings IEEE Intern. Conference on Image Processing (ICIP)*, Oktober 2004. 117, 123