Gestikbasierte Mensch-Maschine-Kommunikation auf der Grundlage visueller Aufmerksamkeit und Adaptivität

Dissertation zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften (Dr.-Ing.)

der Technischen Fakultät der Universität Bielefeld vorgelegt von

Robert Rae
11. September 2000

Gestikbasierte Mensch-Maschine-Kommunikation auf der Grundlage visueller Aufmerksamkeit und Adaptivität

Dissertation zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften (Dr.-Ing.)

vorgelegt von

Robert Rae

an der Technischen Fakultät
der Universität Bielefeld

11. September 2000

Danksagung

Zum Gelingen der vorliegenden Arbeit haben viele Personen beigetragen, denen ich an dieser Stelle danken möchte.

An erster Stelle steht dabei Prof. Helge Ritter, der mich in meiner wissenschaftlichen Laufbahn immer mit Ratschlägen und Esprit unterstützt hat. Danken möchte ich auch Prof. Stefan Posch, der als zweiter Projektverantwortlicher die Arbeiten zu fokussieren wußte und mit konstruktiven Vorschlägen zu dieser Arbeit beigetragen hat.

Mein weiterer Dank gilt den beiden Diplomanden Axel Varchmin und Markus Fislage, ohne die ein solches Projekt unmöglich gewesen wäre. Sie haben mit ihren Ideen und Implementierungen einen großen Einfluß auf die technische Realisierung ausgeübt. Ebenfalls danken möchte ich Daniel Brün, der an vielen technischen Lösungen mitgewirkt hat.

Besonderer Dank gilt meinen Kollegen Nils Jungclaus, der mir bei der Korrektur dieser Arbeit (und mancher Programme) sehr geholfen hat, Christof Dücker, der mich als mein Büronachbar auch in schweren Zeiten immer motiviert hat und nicht zuletzt Ján Jockusch, der mir mit seinen pragmatischen Ideen und Lösungen ebenfalls sehr geholfen hat. Die Zusammenarbeit gefiel mir so gut, daß wir uns entschlossen haben auch die weitere berufliche Zukunft in unserem eigenen Unternehmen gemeinsam zu bestreiten. Darüber hinaus möchte ich auch Daniel Schlüter danken, der mehr als nur mein Projektpartner ist. Hilfreich waren auch die Gespräche und Diskussionen mit Gunther Heidemann.

Weiterer Dank gilt allen anderen Kollegen meiner Arbeitsgruppe, der Technischen Fakultät und des SFBs, die mehr oder weniger an meinen Arbeiten beteiligt waren.

Ein großes Dankeschön auch an meine Mutter, die mir diesen Werdegang ermöglicht hat. Besonderer Dank gilt auch meiner Frau Bettina, die mich immer hervorragend unterstützt hat.

Inhaltsverzeichnis

1 Einleitung

DER Entwicklung von intelligenten und leistungsfähigen Mensch-Maschine-Schnittstellen wird eine wichtige Schlüsselposition für die zukünftige Gestaltung komplexer technischer Systeme zukommen. Die immer schnelleren und kompakteren Rechner, die uns bereits überall im täglichen Leben begegnen, sind in ihrer Performanz und Ausgabeleistung zunehmend besser geworden. Ein Beispiel dafür ist die fortschreitende Entwicklung der Grafikleistung, die heute eine bewegte, fotorealistische Visualisierung von dreidimensionalen Daten selbst auf einem Standard-PC ermöglicht. Allerdings ist trotz dieses rasanten Anstiegs der Rechenleistung die Bedienung vieler Geräte noch immer kompliziert und wenig intuitiv. Während nämlich die Ausgabeleistung der Systeme immer weiter gesteigert wurde hat sich auf der Eingabeseite wenig verändert. So werden zur Steuerung eines Computers hauptsächlich Eingabegeräte verwendet, die sich nur minimal von denen unterscheiden, die bereits eine Dekade zuvor zur Verfügung standen.

Inzwischen haben sich neben der Tastatur weitere Geräte wie die Maus, die Space-Maus oder der Datenhandschuh etabliert, die zumindest die Bedienung von grafischen Applikationen wie beispielsweise CAD-Programmen, erleichtert haben. Diese Geräte besitzen jedoch einen entscheidenden Nachteil: sie schränken die Kommunikationsmöglichkeiten des Benutzers stark ein und binden ihn physikalisch an ein technisches Gerät. Zwar lassen sich über die Tastatur prinzipiell *alle* Funktionen steuern, und besonders in Verbindung mit der Maus können die üblichen Bedienoberflächen gut bedient werden, allerdings sind dreidimensionale Ansteuerungen etwa von 3D-Spielen wenig intuitiv.

Aktuelles Forschungsziel ist deshalb die Entwicklung von flexiblen Mensch-Maschine-Schnittstellen, die diese Probleme überwinden und die natürlichen Kommunikationsfähigkeiten des Menschen wie Sprache und Gestik ausnutzen. Die nicht-verbale Kommunikation durch Bewegungen und Gesten spielt zunehmend eine wichtige Rolle. Der Mensch besitzt ein vielfältiges Repertoire an Körpersprache, die er zusätzlich zur gesprochenen Sprache einsetzt. Neben der Gesichtsmimik geben auch bestimmte Körperhaltungen Auskunft über den Gesprächspartner. Besonders häufig wird aber Handgestik als Kommunikationsmittel eingesetzt.

So haben sich in allen Kulturen typische Gesten entwickelt, die eine bestimmte zugrundeliegende Semantik repräsentieren. Aber auch komplizierte Sprachen, die auf der Grundlage eines Handgestenalphabetes basieren, haben sich im Laufe der Zeit entwickelt. Diese Gestensprache ermöglicht taubstummen Menschen eine genauso komplexe Kommunikation wie mit gesprochenen Worten.

Die computerbasierte Erkennung solcher Gesten ist allerdings zur Zeit noch äußerst schwierig. Zum einen ist die Erkennungsgeschwindigkeit zu gering, zum anderen unterscheiden sich die ausgeführten Handgesten der Personen doch sehr stark[1]. Deshalb beschränken sich die meisten gestikbasierten Mensch-Maschine-Schnittstellen auf einfachere Untermengen von Handgesten. Besonders die Erkennung von Zeigegestik kann die Disambiguierung in bestimmten Kontexten erheblich vereinfachen. So kann etwa eine Äußerung wie „Nimm das da!“ erst durch eine gleichzeitig ausgeführte Zeigegeste interpretiert werden. Der Gesprächspartner wird dadurch angewiesen, ein bestimmtes, durch die Hand referenziertes Objekt zu greifen. Die Auswertung der sprachlichen Äußerung wäre in diesem Fall nicht ausreichend.

Die technische Grundlage für die Entwicklung „sehender Schnittstellen“ sind die Methoden und Techniken der Bildverarbeitung, die auf Basis der in den letzten Jahren kontinuierlich gestiegenen Computerleistungen und verbesserten Aufnahmegeräten erhebliche Möglichkeiten zur bildbasierten Erkennung von Objekten eröffnen. Mittels Spezialhardware können inzwischen komplexe Bildverarbeitungssysteme realisiert werden, die in Echtzeit[2] den Bildinhalt auswerten. Gleichzeitig hat sich auch die zugrundeliegende Algorithmik wesentlich weiterentwickelt. Insbesondere die verbesserte Methodik zur Merkmalsreduktion und Klassifizierung erlaubt die Konstruktion leistungsfähiger Bilderkennungssysteme. Anwendungen solcher Systeme finden sich etwa in der industriellen Qualitätskontrolle, bei der große Mengen an Objekten innerhalb kürzester Zeit zuverlässig klassifiziert werden müssen.

Darüber hinaus wird intensiv an der Architektur von Bildverarbeitungssystemen geforscht. Im Mittelpunkt steht dabei die Definition von Bausteinen und deren Zusammenspiel, um auch komplexe Erkennungsaufgaben zu ermöglichen. Solche Bausteine dienen etwa der Bildvorverarbeitung und normieren die aufgenommenen Bilder auf bestimmte Werte. Andere Bausteine wiederum versuchen, aus der großen Menge an Bildinformation wesentliche Merkmale zu extrahieren. Diese Merkmale dienen dann als Grundlage für die Klassifikation, um bestimmte Objekte oder Bildeigenschaften zu erkennen. Die so gewonnene Information kann anschließend interpretiert werden und die Ausgabe des technischen Systems bestimmen. Die Art und Weise, wie die Bausteine kombiniert werden, bildet dabei die eigentliche Architektur des bildverarbeitenden Systems.

[1] Ähnlich wie bei der Sprache gibt es auch bei Gesten „Dialekte“, die sich während der Durchführung z.B. in unterschiedlichen Trajektorien, Geschwindigkeiten und Handstellungen äußern.

[2] Darunter versteht man die Informationsverarbeitung in einem fest vorgegebenen Takt, der in der Bildverarbeitung i.d.R. der Videobildrate, z.B. bei einer Standard-Videokamera 25 Bildern pro Sekunde (PAL), entspricht.

Ähnlich wie bei der Sprachverarbeitung treten beim Design von Bildverarbeitungssystemen Probleme auf, die bislang nicht grundsätzlich gelöst sind:

- Abhängigkeit der Erkennungsleistung von Umgebungseinflüssen.
- Hohe Störanfälligkeit.
- Anpassung an neue Anforderungen.

Das macht den Einsatz von „klassischen" Bildverarbeitungssystemen in nicht normierten Umgebungen schwierig, wie sie insbesondere bei der visuellen Mensch-Maschine-Interaktion vorkommen. Im Gegensatz zu industriellen Anwendungen bilden hier Aspekte wie Ergonomie, Robustheit und kooperatives Systemverhalten die Grundlage für die Systemarchitektur. So muß das System unter anderem mit unterschiedlichen Benutzern funktionieren, angemessen auf Umweltänderungen reagieren und robuste Erkennungsleistungen garantieren.

Künstliche neuronale Netze bieten hier einen Ansatz, um einige dieser Probleme zu lösen. Dabei wird versucht, Erkenntnisse über Lebewesen in technische Systeme zu übertragen. Ziel ist ein adaptives Systemverhalten, das sich zumindest in gewissen Grenzen an die jeweiligen Anforderungen anpassen läßt. Im Gegensatz zu starr vorgegebenen Algorithmen wird das System damit in die Lage versetzt, intelligent auf seine Umwelt zu reagieren. Dies wird durch entsprechendes Trainieren von neuronalen Netzen etwa auf verschiedene Benutzer und Beleuchtungsbedingungen erreicht. Aufgrund der unscharfen Repräsentation von Information innerhalb des neuronalen Netzes wird somit eine robuste und weniger störanfällige Erkennungsleistung erreicht. Damit ist es möglich, „intelligente" Systeme zu realisieren, die ähnlich wie Lebewesen auf ihre Umwelt reagieren und sich anpassen können.

Betrachtet man nun das Kommunikationsverhalten von Menschen, so ist festzustellen, daß zusätzlich zur visuellen Informationsverarbeitung sehr viel Kontextwissen über die Umwelt in die Verständigung einfließt. Ein technisches System zu implementieren, das eine möglichst natürliche Interaktion zwischen dem Menschen und der Maschine erlaubt, bedeutet also mehr als nur eine reine Erkennung der Gestik zu realisieren. In der vorliegenden Arbeit werden deshalb unterschiedliche Aspekte zum Design gestikbasierter Mensch-Maschine-Schnittstellen beleuchtet. Neben der Entwicklung einzelner Bausteine, etwa zur Erkennung von Hautregionen, werden insbesondere Architekturfragen zur Integration weiterer solcher Bausteine behandelt. Das technische System soll dabei in die Lage versetzt werden, zusätzliche Information wie etwa Objektwissen auszuwerten und entsprechend seine Handlungen anzupassen. Darüber hinaus sollen, ähnlich wie beim Menschen, unterschiedliche Modalitäten ausgewertet werden, die insgesamt ebenfalls die Reaktion des Systems beeinflussen. So spielt neben der Sprache auch die Einbindung von Gestik und Mimik eine starke Rolle in der Verständigung.

Das wird deutlich bei der Referenzierung von Objekten mittels Handgesten: Der Gesprächspartner erkennt eine Zeigegeste und vernimmt gleichzeitig eine Äußerung wie z.B. „Was ist das für ein Gegenstand dort drüben?“. Allein durch die Erkennung des Handwinkels und der Handstellung wird es für die Person schwierig, den referenzierten Ort exakt zu lokalisieren. Eventuell wird auch die Blickrichtung der zeigenden Person in die Auswahl des Objektes mit einbezogen. Typischerweise schaut die Person grob in die Richtung der Zeigegeste und betrachtet die dort in Frage kommenden Objekte. Befindet sich dort nur ein Objekt, so wird dieses gewählt. Sind dort jedoch mehrere Objekte zu sehen, findet wahrscheinlich eine Rücksprache statt.

Zentrale Bedeutung kommt dabei den Begriffen *Aufmerksamkeit* und *Reaktivität* zu. Erst durch die gemeinsame Koordination des Aufmerksamkeitsbereiches, also über die Beobachtung der Gestik (Zeigegeste, Blickrichtung etc.) des Gesprächspartners, und der entsprechenden eigenen Reaktion etwa durch Kopfbewegungen, entwickelt sich eine non-verbale Verständigung. Als Rückmeldung dient dabei zum einen die Sprache (z.B. „Weiter links!“), zum anderen aber auch die körperliche Reaktion des Partners, die sich etwa durch die Auswertung seiner Blickrichtung oder seiner Handgestik ergibt.

Auf der Basis dieses Kommunikationsverhaltens werden in der vorliegenden Arbeit zwei konkrete Gestik-Schnittstellen vorgestellt: Ein System realisiert die Erkennung von Blickrichtungen eines Menschen und kann die Blicktrajektorien in Form von Blickpunkten auf dem Monitor aufzeichnen. Das andere System erlaubt die Referenzierung von Objekten mittels Zeigegesten, die in natürlicher Art und Weise vom Benutzer ausgeführt werden. Beide Systeme kommen ohne zusätzliche Markierungen oder am Benutzer befestigte Geräte aus und unterstützen die natürlichen Kommunikationsmöglichkeiten des Benutzers. Die jeweils vorgeschlagene Architektur orientiert sich an Erkenntnissen über die Wahrnehmung beim Menschen und bietet darüber hinaus Raum für die Integration sowohl zusätzlicher Gesten, als auch neuer Modalitäten, wie etwa Sprache.

1.1 Grundprinzipien

Mit der technischen Realisierung von gestikbasierten Schnittstellen wird es möglich, derartige Systeme in Mensch-Maschine-Kommunikationsszenarios einzusetzen. Ziel ist eine möglichst natürliche Verständigung zwischen künstlichem System und dem Benutzer. Wie aber können die Grundprinzipien derartiger System erforscht werden, und wie sehen einzelne Bausteine dafür praktisch aus? In dieser Arbeit werden zur Beantwortung dieser Fragen Anleihen an natürlichen Vorbildern gemacht. Dabei steht nicht die genaue kognitive Simulation der biologischen Erkenntnisse im Mittelpunkt, sondern vielmehr die Ausnutzung der Prinzipien als Leitfaden für das Design technischer Systeme.

Höhere Lebewesen, die eine gestikbasierte Kommunikation entwickelt haben, besitzen im allgemeinen eine adaptive Gehirnstruktur, die es ihnen erlaubt, flexibel auf Dinge zu reagieren. Neben der kurzfristigen Reaktion auf Ereignisse werden sie damit auch in die Lage versetzt, in völlig neuen Lebensräumen ihr Verhalten zu ändern, etwa um bestimmte Ressourcen an Nahrung besser ausnutzen zu können. Beobachtet man diese Lebewesen, so ist schnell ein weiteres gemeinsames Merkmal zu erkennen: die Aufmerksamkeit. Da die meisten höheren Lebewesen Augen besitzen, die sich wiederum in einem beweglichen Kopf befinden, kann mittels entsprechender Kopf- und Augenbewegungen die visuelle Aufmerksamkeit schnell auf unterschiedliche Orte gelenkt werden. Damit wird trotz der im Vergleich zum Computer langsam arbeitenden „biologischen Hardware“ eine erstaunlich robuste visuelle Wahrnehmung der Umwelt realisiert.

Die Nachbildung und Simulation dieser Grundprinzipien in technischen Systemen, in Verbindung mit der inzwischen kostengünstig verfügbaren hohen Rechenleistung und entsprechender Bilderfassungs- und -verarbeitungshardware rückt die Verwirklichung „sehender“ Computerschnittstellen in Reichweite. Die Mensch-Maschine-Schnittstellen der Zukunft werden neben einer sprachlichen Dialogfähigkeit durch Spracherkennung und -produktion den Benutzer auch visuell wahrnehmen und etwa auf Gestik oder auf visuell dargebotene Instruktionen (z.B. durch Vormachen von Handhabungsabläufen) reagieren. Dabei wird, ähnlich wie bei einem menschlichen Dialogpartner, die Blickrichtung der Maschine dem Menschen erkennbar gemacht. Ziel ist es, eine dynamisch und kontextabhängig wandernde Region „gemeinsamer visueller Aufmerksamkeit“ zu schaffen und aufrechtzuerhalten. Dies unterstützt beispielsweise wirkungsvoll die Referenzierung von Gegenständen oder ermöglicht wichtige Dialogfunktionen wie z.B. den Sprecherwechsel (*turn-taking*).

Eine wichtige Errungenschaft der Evolution ist die Entwicklung neuronaler Strukturen. Die diesem Konzept zugrundeliegende Ausprägung der Adaptivität erlaubt es den Organismen, komplizierte Vorgänge wie etwa das Fliegen oder Laufen zu koordinieren. Darüber hinaus bietet das adaptive Systemverhalten die Möglichkeit, auf Situationen und Umweltveränderungen adäquat zu reagieren und gewisse Vorgänge zu erlernen und zu behalten. Mit künstlichen neuronalen Netzen werden solche biologischen Netze in ihrer Funktionalität nachgebildet. Dafür werden abstrakte mathematische Modelle formuliert, die bestimmte Eigenschaften der biologischen Netze beschreiben. Im Vordergrund steht dabei das Verhalten und nicht die genaue Struktur, die erst durch einen Lernprozeß erzeugt wird. Nachdem dieser Lernprozeß abgeschlossen ist, erzeugt das Netz für eine Eingabe die (hoffentlich) gewünschte Ausgabe.

Prinzipiell eignen sich künstliche neuronale Netze besonders gut für die Anwendung in der Bildverarbeitung und der Robotik, da die zugrundeliegende Komplexität dieser Problemstellungen oftmals sehr groß ist. Das macht eine präzise mathematische Analyse und Modellierung der auftretenden Prozesse unmöglich oder zumindest schwierig. Insbesondere die Lernfähigkeit und Robustheit der neuronalen Netze erlaubt eine flexible Anpassung an Trainingsbeispiele sowie die Reduzierung der Parameter zur Vereinfachung des Systems. Daher bilden sie einen wichtigen Ansatz beim Design der in dieser Arbeit vorgestellten visuellen Mensch-Maschine-Schnittstellen.

Die Evolution biologischer Sehsysteme hat auch eine Reihe bemerkenswerter Verarbeitungsstrategien hervorgebracht, um die für reaktionsschnelles Sehen erforderlichen, sehr hohen Anforderungen an die Verarbeitungsleistung zu bewältigen. Eine davon ist die bei den Sehsystemen der höheren Tiere sehr gut entwickelte *visuelle Aufmerksamkeit*, die eine wirkungsvolle Bündelung der begrenzten Verarbeitungsressourcen auf den jeweils relevanten Teil des Blickfeldes bewirkt. Diese Mechanismen werden zunehmend auch auf technische Systeme übertragen. Besonders vielversprechend erscheint der Einsatz aktiver Sehsysteme, bei denen die Kameras auf bestimmte Bildbereiche ausgerichtet werden. Damit kann die riesige Menge an visuellen Informationen, die vom Bildverarbeitungssystem verarbeitet wird, drastisch reduziert werden.

1.2 Zielsetzung

Gestik bildet neben Sprache eine der Hauptsäulen unserer natürlichen Kommunikation. Nichtsprachliche Mechanismen wie etwa die Koordination visueller Aufmerksamkeit durch Augen- und Kopfbewegungen, die Bezugnahme auf Objekte durch Zeigegesten oder die Demonstration von Handlungsabläufen tragen dabei oft einen erheblichen Anteil des gesamten Kommunikationsgeschehens. Besonders die Interaktion zwischen den Gesprächspartnern, um sich auf einen gemeinsamen Aufmerksamkeitsfokus zu einigen, erfordert eine genaue Beobachtung der Körpergestik.

Die Nachbildung derartiger Fähigkeiten in einem künstlichen System bildet daher eine wichtige Komponente „künstlicher situierter Kommunikatoren“, wie sie im Rahmen des Bielefelder Sonderforschungsbereichs 360 erforscht werden. Als Szenario dient die kooperative Konstruktion eines Flugzeugmodells aus einzelnen Holzbausteinen. Der Zusammenbau der Teile erfolgt durch Anweisungen eines menschlichen Instrukteurs, der durch Sprache und Gestik den künstlichen Konstrukteur anleitet. In dieser Arbeit werden zwei Systeme vorgestellt, die das Zusammenspiel einiger Grundfunktionen „aktiv sehender“ Mensch-Maschine-Schnittstellen unter diesem Aspekt demonstrieren.

Zunächst wird ein adaptives System zur *Erkennung menschlicher Blickrichtungen* vorgestellt. Das System basiert auf adaptiven Bausteinen, die typische Gesichtsmerkmale wie Augen, Nase und Mund in den Kamerabildern detektieren. Die so ermittelte Augenregion dient dann der Auswertung der Blickrichtung mit einem an diese Aufgabe angepaßten neuronalen Netz.

Das zweite System realisiert eine *Zeigegestenerkennung* zur interaktiven Steuerung eines Roboterarms. Es besteht aus einem steuerbaren binokularen Kamerakopf, dessen „Aufmerksamkeitsfokus“ von zwei komplementären Aufmerksamkeitsmechanismen gesteuert wird. Der Erprobungsrahmen dieses Systems ist ein Szenario gestikbasierter Mensch-Maschine-Kommunikation, bei dem das autonome Explorationsverhalten des Systems durch Zeigegesten eines menschlichen Partners unterbrochen und selektiv auf einzelne Objekte fokussiert werden kann. Die erzielten Resultate bei der Erkennung von Zeigegesten unterschiedlicher Benutzer mit dem künstlichen adaptiven Sehsystems erweisen sich als vielversprechend.

Beide Systeme arbeiten auf der Grundlage neuronaler Netze und orientieren sich am Paradigma des „aktiven Sehens“. Sie funktionieren nach dem „Aktion–Reaktion-Prinzip“, werden allerdings auf der Basis zweier unterschiedlicher Aufmerksamkeitsmodellierungen gesteuert. Im Falle der Blickrichtungserkennung handelt es sich um eine zwar datengetriebene, aber mit zusätzlichen Modellwissen versehene Lösung. Diese Herangehensweise bietet sich an, weil Gesichter im allgemeinen wenig variabel in der Anordnung der üblichen Gesichtsmerkmale (Augen, Nase und Mund) sind.

Das ist grundlegend anders im Falle der Handgestenerkennung: Zum einen ist das Erscheinungsbild der Hand je nach Fingerstellung und Blickwinkel stark unterschiedlich, und zum anderen werden großflächigere Bewegungen mit der Hand durchgeführt. Darüber hinaus ist es notwendig, zusätzliches Szenenwissen über die vorhandenen Objekte in die Richtungserkennung mit einzubeziehen, da die Auswertung der Zeigerichtung bislang nur sehr ungenau erfolgen kann. Daher wurde in diesem Fall eine rein datengetriebene Aufmerksamkeitsmodellierung gewählt, die flexibel auf neue Merkmale und Reize erweiterbar ist.

Ziel dieser Arbeit ist es, neben der eigentlichen Realisierung der Systeme, Architekturkonzepte aufzuzeigen und zu analysieren, die sich am Vorbild der Natur orientieren. Zentrale Prinzipien sind dabei Aufmerksamkeit, Adaptivität und aktives Sehen. Nach der Modellierung dieser Eigenschaften werden Bausteine realisiert, die basierend auf einer adäquaten Architektur zu intelligenten Systemen für die natürliche Mensch-Maschine-Interaktion zusammengesetzt werden. Nur durch eine geeignete Modularisierung und eine erweiterbare Architektur kann dieses Ziel erreicht werden. Diese Arbeit liefert daher konkrete Vorschläge zur Integration dieser Bausteine und einen Beitrag zum Design gestikbasierter visueller Mensch-Maschine-Schnittstellen.

Die Evaluierung der im folgenden vorgestellten „sehenden Schnittstellen“ basiert auf der realen Anwendung der Systeme in typischen Anwendungsszenarien. Nur so können die oben genannten Anforderungen wie Robustheit, Erweiterungsfähigkeit und Flexibilität wirklich getestet werden. Konkrete Ergebnisse und entsprechende Versuchsreihen untermauern diese Systemeigenschaften und geben einen Eindruck der Gesamtperformanz im praxisnahen Einsatz. Soweit vergleichbare Lösungen oder Ansätze von anderen Forschern vorhanden bzw. bekannt sind, werden ausführlich die Vor- und Nachteile sowie eventuell bestehende Synergien diskutiert. Zukünftig wird vermutlich erst die Integration dieser unterschiedlichen Ansätze zur Gestenerkennung zu einer marktreifen Entwicklung ergonomischer Computer-Schnittstellen führen. Diese Produktreife ist nicht das Ziel dieser Arbeit, wenn auch bereits Aspekte wie Anwenderfreundlichkeit, Minimierung der Hardwarevoraussetzungen und Effizienzbetrachtungen eine große Rolle beim Design gespielt haben. Im Mittelpunkt steht viel mehr die Erforschung und praktische Erprobung von grundlegenden Voraussetzungen und Bausteinen, die in ihrer Summe in Zukunft einen natürlicheren Umgang mit technischen Systemen für den Menschen ermöglichen.

1.3 Aufbau der Arbeit

Das Kapitel 2 beschäftigt sich zunächst mit den biologischen Grundlagen des menschlichen Sehsystems. Neben den physiologischen Erkenntnissen geht es auch um psychophysische Beobachtungen, wie sie etwa bei Eyetracking-Experimenten gemacht werden. Diese Ergebnisse dienen später zur Erstellung des Aufmerksamkeitsmodells und fließen in die technische Realisierung ein. Anschließend werden die für die Klassifizierung verwendeten künstlichen neuronalen Netze vorgestellt (Kapitel 3). Die Architektur der beiden Netzwerktypen „Multilagen-Perzeptron“ und „Local Linear Map“ werden miteinander verglichen und Vor- und Nachteile im Kontext der Bildverarbeitungsaufgaben diskutiert.

In Kapitel 4 wird eine modellgestützte Aufmerksamkeitssteuerung vorgestellt. Diese Modellierung bildet die Grundlage für ein System zur Erkennung menschlicher Blickrichtungen (GAZE). Neben der Modellbildung werden die einzelnen Bausteine des Systems beschrieben. Diese teilen sich auf in die Gesichtsverfolgung, die Detektion typischer Gesichtsmerkmale sowie die eigentliche Blickrichtungserkennung. Daraus entstandene Ideen und implementierte Bausteine fließen auch in das folgende System zur Zeigegestenerkennung ein.

Die nächsten Kapitel beschreiben die Grundlagen für eine weitere Mensch-Maschine-Schnittstelle, die die Erkennung von Zeigegesten leistet. Hier wird eine datenbasierte Modellierung der visuellen Aufmerksamkeit für das künstliche Sehsystem gewählt (Kapitel 5). Zur schnellen Verfolgung der Hände im Blickfeld der aktiven Kameras wird in das Aufmerksamkeitsmodell zusätzlich eine disparitätsbasierte Objektverfolgung integriert, die in Kapitel 6 beschrieben wird. Schließlich wird in Kapitel 7 das integrierte Gesamtsystem zur Erkennung der Zeigegestik mit der Ansteuerung des Manipulators gezeigt.

Darauf folgt eine Diskussion der gewonnenen Erkenntnisse, die sich aus den vorgestellten Systemen zur Mensch-Maschine-Kommunikation ergeben (Kapitel 8). Insbesondere der

Vergleich beider Systeme zeigt die Vor- und Nachteile der verwendeten Ansätze auf. Daran schließt sich ein Ausblick an, der erläutert, wie weitere Verbesserungen der Mensch-Maschine-Schnittstellen aussehen, die etwa durch die Integration neuer Modalitäten erzielt werden (Kapitel 9). Im Mittelpunkt steht dabei die Frage, inwieweit derartige Systeme in naher Zukunft ihren Weg in die Praxis finden. Abschließend erfolgt in Kapitel 10 eine kurze Zusammenfassung der Arbeit und der daraus folgenden Ergebnisse und Erkenntnisse.

2 Biologische Grundlagen menschlichen Sehens

DIE Natur hat im Laufe der Evolution die visuelle Informationsverarbeitung bei Mensch und Tier in erstaunlichem Maße optimiert. Neben der immer besseren Konstruktion der Sensorik spielt die Entwicklung von speziellen Gehirnstrukturen eine große Rolle bei der visuellen Perzeption. Ausgehend von einer simplen hell/dunkel Wahrnehmung mittels Sehmulden hat sich das hochkomplexe Auge bestehend aus einem Glaskörper und einer dahinter liegenden lichtempfindlichen Retina entwickelt. Damit wird beispielsweise der Mensch in die Lage versetzt, seine Umwelt detailreich und in Farbe wahrzunehmen.

Um die dabei anfallende riesige Menge an Information effizient zu verarbeiten, war auch eine Optimierung der Verarbeitungsstrategie und der dafür zur Verfügung stehenden biologischen „Hardware“ nötig. So hat sich eine ortsvariante Sensorverteilung auf der Retina entwickelt, die in der Mitte einen hochauflösenden Bereich, die Fovea, aufweist. Mit der großen Dichte an Rezeptoren im Mittelpunkt des Auges können kleine Bildbereiche sehr detailliert wahrgenommen werden. Die Verteilung der neuronalen Sensoren korrespondiert wiederum mit der topologieerhaltenden Verschaltung in den visuellen Kortex hinein. Mittels einer nach heutigen Erkenntnissen hierarchisch organisierten Verarbeitung in höheren Gehirnregionen kann die visuelle Information effizient gebündelt und intelligent ausgewertet werden.

Ein weiteres wichtiges Prinzip der visuellen Perzeption basiert auf der Möglichkeit, das Auge schnell und präzise bewegen zu können. Der Mensch kann so die hochauflösende Fovea abrupt auf neue Bildbereiche lenken, um diese detailliert zu betrachten. Interessant ist dabei die Frage, auf welcher Grundlage diese Bewegungen ausgelöst werden und welche Bereiche im Gehirn daran beteiligt sind. Aufschluß darüber geben zum einen anatomische und physiologische Untersuchungen, zum anderen Beobachtungen der Augenbewegungen selbst, die Einblicke in die ablaufenden Denkprozesse erlauben [Pom98]. Deshalb haben sich Forscher zur Aufgabe gesetzt, Augenbewegungen zu untersuchen. Sie unterscheiden dabei zwei unterschiedliche Grundtypen von Bewegungen: (i) die Sakkade und (ii) die weiche Blickfolgebewegung. Sakkaden bezeichnen dabei die schnellen, ruckartigen Augenbewegungen, die zur genauen Betrachtung von Bildbereichen dienen. Die statische Phase zwischen den Sakkaden, also die Ruhephase des Auges, wird *Fixation* genannt. Die weiche Folgebewegung wird dagegen zur Verfolgung von bewegten Objekten im Blickfeld eingesetzt. Heutzutage ist es möglich, diese unterschiedlichen Mechanismen mittels Eyetracking-Experimenten zu beobachten und auszuwerten. Ziel ist es dabei, die zugrundeliegenden Blicksteuerungsmechanismen zu verstehen.

Erkenntnisse aus diesem Forschungsbereich helfen auch bei der Entwicklung technischer Systeme, indem Prinzipien des natürlichen Vorbilds etwa zur Datenreduktion in das Design einfließen. Die Probleme, die beim Aufbau eines künstlichen Bildverarbeitungssystems auftreten, sind eng verwandt mit denen, die die Natur löst: Die riesigen Datenmengen, die bei der Aufnahme von Bildern anfallen, können mit der heute verfügbaren Rechenleistung nur unzureichend verarbeitet werden. Es befinden sich ca. 125 Millionen Stäbchen und ca. 3-5 Millionen Zapfen auf der menschlichen Retina. Im Vergleich dazu erzeugt ein Farb-CCD-Kamerachip bei einer Auflösung von 512×512 Bildpunkten und einer Bildrate von 25 Bildern pro Sekunde eine Datenmenge von ungefähr 18.75 MByte pro Sekunde. Auch bei der raschen Weiterentwicklung der Computer ist die Verarbeitung dieser Datenmengen schwierig. Deshalb wird nach neuen Lösungen gesucht, die sich am natürlichen Vorbild orientieren und die nur in einer interdisziplinären Zusammenarbeit zwischen unterschiedlichen Fachrichtungen erforscht werden können. Daher werden auch in dieser Arbeit zunächst die Erkenntnisse über das menschliche Sehsystem zusammengetragen, um in den späteren technischen Realisierungen Parallelen dazu aufzuzeigen.

2.1 Physiologische Erkenntnisse

In diesem Abschnitt werden die grundlegenden Erkenntnisse aus der Biologie über den anatomischen und neuronalen Aufbau des menschlichen Sehapparates zusammengefaßt. Diese kurze Einführung erleichtert das Verständnis für die Mechanismen des natürlichen Sehens und klärt einige Begrifflichkeiten, die für die späteren Ausführungen der technischen Systeme wichtig sind. Neben dem organischen Aufbau des Auges geht es vor allem um die neuronalen Verschaltungen, die bei der Verarbeitung visueller Perzepte eine Rolle spielen.

2.1.1 Anatomie des Auges

Das menschliche Auge besteht im wesentlichen aus einem Glaskörper, der in einer Augenhöhle innerhalb des Kopfes ruht (Abbildung 2.1). Das emittierte Licht der betrachteten Umgebung wird über die Linse durch den Glaskörpers auf eine dahinter liegende Projektionsschicht abgebildet. Für die Bewegung des Auges sind sechs Augenmuskeln zuständig, die jeweils paarweise antagonistisch an den Augapfel angreifen. Dadurch können horizontale und vertikale Augenbewegungen ausgeführt werden. Zusätzlich ist auch eine Rollbewegung möglich. Diese erfolgt um eine etwas außerhalb des optischen Zentrums des Augapfels liegende Achse. Allerdings sind diese Rollbewegungen in erster Linie an den *vestibulären Reflex* gekoppelt, der dafür sorgt, daß immer ein stabiles Bild auf der Retina erscheint. Die Augen können mit den drei Muskelpaaren bei Winkelgeschwindigkeiten von bis zu $700^o/s$ präzise bewegt werden. Die Koordination der Muskeln wird vom *okulomotorischen System* (Abbildung 2.4) übernommen, auf das in Abschnitt 2.1.3 noch näher eingegangen wird.

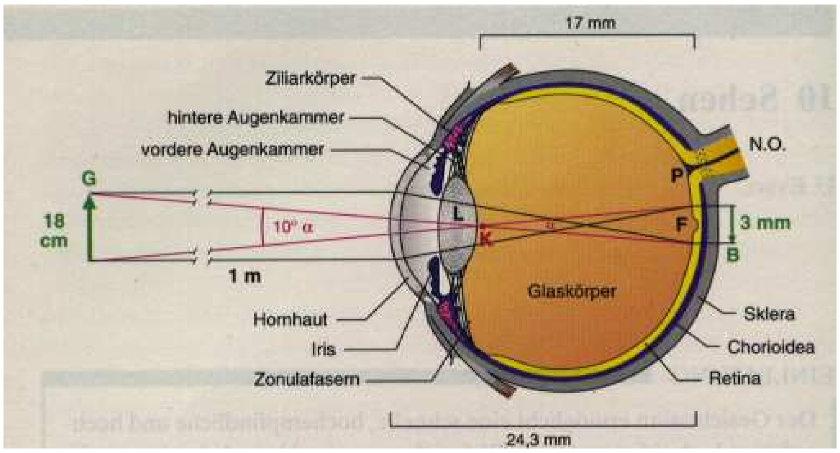


Abbildung 2.1: Schematische Darstellung des menschlichen Auges: Die optische Abbildung erfolgt über die Linse (L) durch den Glaskörper auf die Retina. Der foveale Bereich ist mit F gekennzeichnet, der abgehende Sehnerv mit P. (aus [Sch95]).

Der Augapfel des Menschen besteht zudem noch aus weiteren Komponenten, die für die optische Abbildung sorgen. Die Hornhaut (*Cornea*), die Iris und die Linse sorgen für eine Projektion der Bilder auf die dahinterliegende Netzhaut (*Retina*). Das optische System funktioniert dabei ähnlich wie ein Kameraobjektiv, wobei das Licht durch den Glaskörper geleitet wird und die Iris die Funktion der Blende übernimmt. Die Linse kann mit Hilfe der *Ciliarmuskeln* verformt werden, so daß die betrachteten Objekte immer scharf auf der Netzhaut abgebildet werden. Unschärfen und Schwankungen der Helligkeit werden zusätzlich durch entsprechende Mechanismen auf der Retina ausgeglichen.

2.1.2 Visuelle Informationsverarbeitung auf der Retina

Direkt hinter dem Glaskörper des Auges befindet sich die Retina. Diese sensorische Schicht nimmt die einfallenden Lichtstrahlen auf und verwandelt diese in neuronale Signale. Neben der reinen Erfassung der visuellen Information findet in dieser Schicht auch eine parallele neuronale Vorverarbeitung der Signale statt. Dabei ist das Auflösungsvermögen der Retina nicht uniform verteilt, sondern die Empfindlichkeit nimmt von der *Fovea* — dem Zentrum höchster Auflösung — nach außen hin logarithmisch ab. Das hohe Auflösungsvermögen der Fovea ist also nur auf einen kleinen, etwa drei Grad großen Teil der Retina beschränkt. In der Abbildung 2.1 ist der foveale Bereich mit F gekennzeichnet.

Die Retina besteht in erster Linie aus zwei Typen neuronaler Photorezeptoren: den sog. Stäbchen und den Zapfen. Die besonders lichtempfindlichen Stäbchen sind vornehmlich für das Dämmerungssehen bei sehr wenig Umgebungslicht zuständig. Die Zapfen hingegen ermöglichen die Wahrnehmung von Farben und feinen Details. Die Stäbchen befinden sich insgesamt in der Überzahl, allerdings kommen im Bereich der Fovea nur Zapfen vor. Dadurch wird die schon beschriebene maximale Sehschärfe und Farbwahrnehmung in diesem Bereich erzielt. Das Farbensehen wird durch unterschiedliche Photopigmente innerhalb der Zapfen erreicht, deren Absorption bei einer bestimmten Wellenlänge maximal ist. Aufgrund chemischer Prozesse wird so die Erregung des Photorezeptors beeinflußt. Bei steigendem Lichteinfall nimmt der Dunkelstrom innerhalb des Rezeptors ab, und die Rate der generierten Nervenimpulse sinkt (die sogenannte Feuerrate). Es wird also der einfallende Lichtreiz in Nervenimpulsen kodiert.

Auf die Rezeption der Reize durch die sensorischen Zellen folgen vier weitere zelluläre Schichten[1], die eine retinale Vorverarbeitung der Signale realisieren (Abbildung 2.2): Die *Horizontalzellen*, die *Bipolarzellen*, die *amakrinen Zellen* und die *Ganglienzellen*. Die Bipolarzellen sind über die Horizontalzellen mit den Photorezeptoren verbunden und können mittels *lateraler Inhibition* bereits Kanten extrahiert sowie Änderungen der Bewegungsrichtung auf der Retina erfassen. Dabei erhält eine Bipolarzelle Signale von einer Reihe weiterer Rezeptoren. Diese Menge an Photorezeptoren repräsentiert somit einen bestimmten Bereich der Retina — das *rezeptive Feld* einer Bipolarzelle.

Prinzipiell wird zwischen *On*- und *Off-Zellen* unterschieden. Die On-Zellen besitzen im Zentrum exzitatorische Verbindungen und im Randbereich inhibitorische Verbindungen zu den vorgelagerten Rezeptoren. Bei den Off-Zellen ist die Verschaltung genau umgekehrt. Durch diese Kombination wird die Bipolarzelle kontrastspezifisch und kann auf Lichtkanten reagieren. Diese Zellen werden auch als Komplexzellen bezeichnet, die etwa im primären visuellen

[1] Obwohl sie anatomisch *vor* den Photorezeptoren liegen und zuerst vom Licht durchdrungen werden.

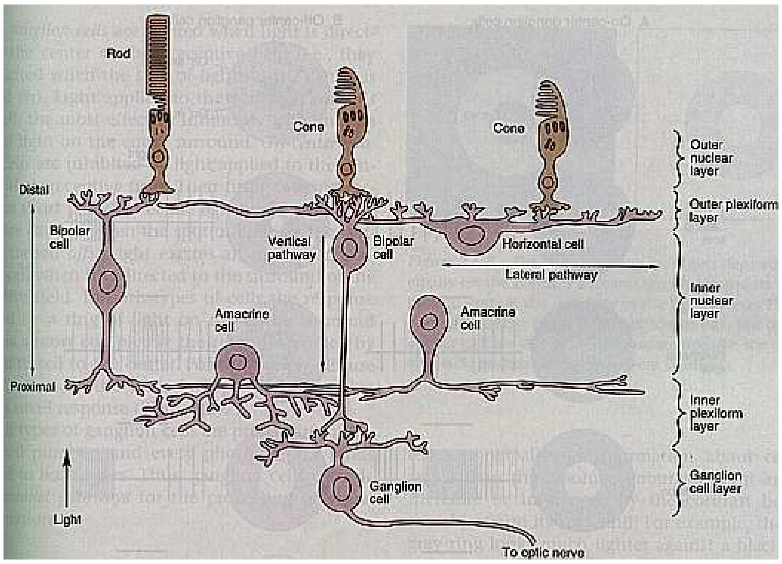


Abbildung 2.2: Der schematische Aufbau der Retina: In der hinteren Schicht befinden sich die Photorezeptoren (Stäbchen und Zapfen). Davor liegen die Horizontalzellen, die die Schnittstelle zu den Bipolarzellen bilden. Die amakrinen Zellen wiederum liegen vor den Ganglienzellen. (aus [KSJ95]).

Kortex nachgewiesen sind [KSJ95] Aufgrund der massiv parallelen Verarbeitung der eingehenden Sensorsignale wird auf dieser Ebene ein Ausgleichsmechanismus gegenüber einzelnen Fehlern bei den Photorezeptoren realisiert. Durch die Überlagerung der Signale aus mehreren Bipolarzellen können bestimmte visuelle Reize wie etwa Größe und Orientierung detektiert werden (Abbildung 2.3).

Die amakrinen Zellen besitzen unter anderem eine hemmende Wirkung auf die Stäbchen, um damit bei genügend Licht ein besseres Farbensehen zu ermöglichen [KSJ95]. Ihnen angeschlossen befinden sich die Ganglienzellen, die die letzte Verarbeitungsschicht der Retina darstellen. Auch die Ganglienzellen besitzen rezeptive Felder, die etwa der Ermittlung der Reizgröße dienen. Die Verschaltungsarten sind dabei ähnlich zu denen der Bipolarzellen (Abbildung 2.3).

Die ausgehenden Reize der rezeptiven Felder in der Retina charakterisieren unterschiedliche optische Eigenschaften, wie Orientierung, Bewegung und Reizlänge. Diese Nervensignale werden in den *visuellen Kortex* weitergeleitet. Wie die weitere Verarbeitung der Signale im menschlichen Gehirn aussieht, wird im nächsten Abschnitt beschrieben. Darüber hinaus wird dort auch das Okularsystems zur motorischen Steuerung der Augen erklärt.

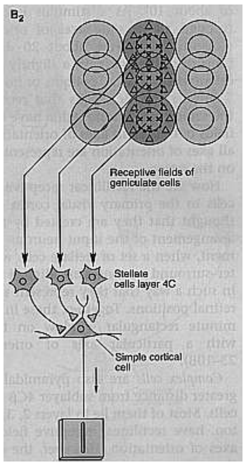


Abbildung 2.3: Schematische Organisation eines rezeptiven Felds zur visuellen Detektion von Reizorientierungen und -längen. (aus [KSJ95]).

2.1.3 Neurophysiologie der Augenbewegungssteuerung

Die Augenbewegungen werden beim Menschen durch das okulomotorische System gesteuert [WW88]. Mittels eines komplexen Integrationsnetzwerkes werden dabei die sechs Augenmuskeln, die das Auge durch Kontraktion bewegen, kontrolliert. An diesem Vorgang sind unterschiedliche Bereiche im Gehirn beteiligt [Rei90], die schematisch in Abbildung 2.4 gezeigt sind. Das im *Thalamus* lokalisierte laterale Geniculatum stellt das Bindeglied zwischen der Retina einerseits und dem *primären Kortex* (V1) sowie dem *superioren Colliculus* (SC) andererseits dar. Der superior Colliculus ist nachweislich an der Sakkadensteuerung beteiligt.

Der modulare Aufbau des okulomotorischen Systems ermöglicht viele parallele Kontrollmechanismen. So ist unter anderem das Gleichgewichtsorgan und der *Vestibularkern* daran beteiligt, das Auge zu steuern[2]. Neben diesen eher reflexartigen Kontrollmechanismen sind auch höhere Gehirnfunktionen an der Augenbewegungssteuerung beteiligt. So wurden in Tierversuchen inzwischen Neuronen detektiert, die bei speziellen komplexen Reizen besonders stark reagieren[3]. Aus diesem Gehirnbereich laufen ebenfalls Nerven zurück zum okularen Motorkern (siehe Abbildung 2.4), was die Annahme verstärkt, daß auch diese Bereiche einen direkten Einfluß auf die Augensteuerung haben. Trotz der hohen Zuverlässigkeit und Geschwindigkeit, mit der die Augenbewegungen stattfinden, sind die neuronalen Verschaltungen adaptiv. Sie können sich also an neue Umgebungsbedingungen oder körperliche Veränderung anpassen [Rei90].

[2] Durch entsprechende *in vitro* Experimente können diese Abläufe sichtbar gemacht werden.
[3] Im Affenkortex wurden etwa Neuronen, die auf die Gesichtserkennung spezialisiert sind, gefunden [Rei90].

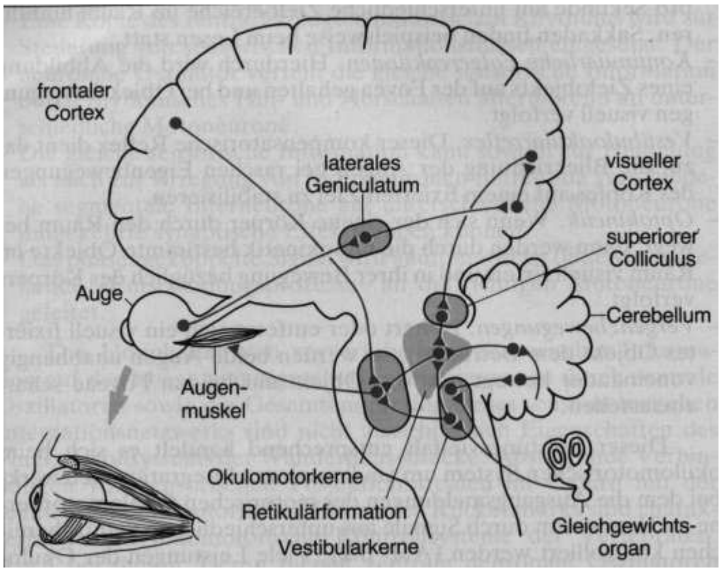


Abbildung 2.4: Schematische Darstellung des menschliche Okularsystem zur Augenbewegungssteuerung (aus [Rei90]).

Wie sehen die Verbindungen der Nervenstränge des visuellen Systems nun im einzelnen aus? Wie schon beschrieben projizieren nahezu alle Ganglienzellen der Retina über den Sehnerv in das CGL (Abbildung 2.5). Hier wird die Information aus beiden Augen zusammengeführt und in sechs verschiedenen Schichten verschaltet. Nur ein kleiner Teil der Nervenstränge (etwa 10%) läuft direkt in den SC (siehe [Wan95]). Im CGL werden die visuellen Nervensignale gepuffert und neu miteinander verknüpft. Die Abbildung der Nerven aus der Retina folgt dabei einem festen Muster. Jeweils drei der insgesamt sechs Schichten repräsentieren die Information vom linken bzw. rechten Auge. Die Signale werden getrennt auf unterschiedlichen *Bahnen* in den CGL weitergleitet, die dort wiederum *Schichten* bilden. Dabei werden zwei Typen von Bahnen/Schichten unterschieden: (i) die *parvozellulären* und (ii) die *magnozellulären Bahnen*. Die Neuronen in der parvozellulären Schicht basieren auf kleineren rezeptiven Feldern auf der Retina als die der magnozellulären Schicht und werden direkt in den visuellen Kortex abgebildet. Die Neuronen der magnozellulären Bahn wiederum haben größere rezeptive Felder und reagieren spezifisch auf bewegte Reize [KSJ95].

Zwei der magnozellulären Bahnen sind darüber hinaus direkt in das SC geführt. Die Signale der anderen vier Bahnen werden zunächst weiterverarbeitet. Dafür wird die Information aus dem CGL retinotop, also gebündelt bezüglich der rezeptiven Feldcharakteristika (wie etwa Position, Orientierung, Form und Farbe), im primären visuellen Kortex abgebildet (siehe Abbildung 2.6). Die im V1 und V2 nachgeschalteten Nervenzellen charakterisieren dabei Ei-

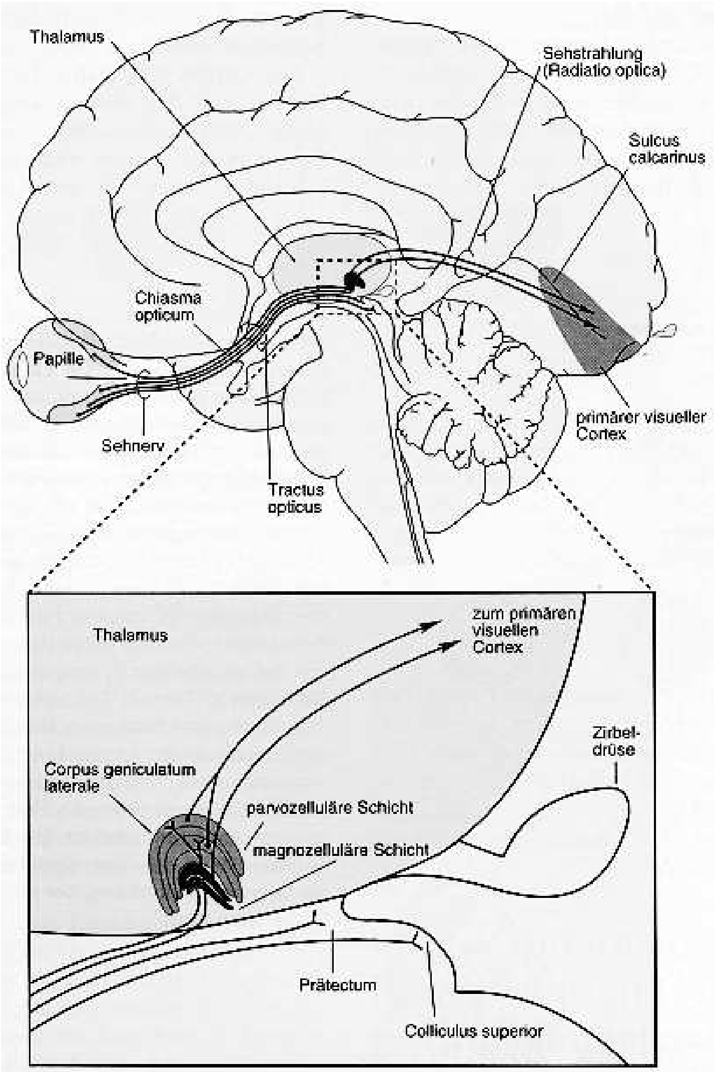


Abbildung 2.5: Die menschliche Sehbahn: Der Sehnerv leitet die Information der Retina in getrennten Bahnen weiter in das CGL. Von dort aus laufen Nerven in den primären Kortex (aus [KSJ95]).

genschaften wie Orientierung, Form und Farbe des optischen Reizes (Abbildung 2.6). Die dort nachgewiesenen *Dominanzbänder* verarbeiten die Nervensignale und haben wiederum Verbindungen zum SC und somit auf die Sakkadensteuerung. Der Aufbau des kortikalen Nervensystems ist in Abbildung 2.7 skizziert.

Der superior Colliculus ist in erster Linie für die Sakkadensteuerung der Augen zuständig [DLS96]. Deshalb wird jede Information, die an der Bewegung der Augen beteiligt ist, in dieses Areal geleitet. Im SC werden die Sakkaden dann durch eine *Bewegungskarte* (*motor map*)

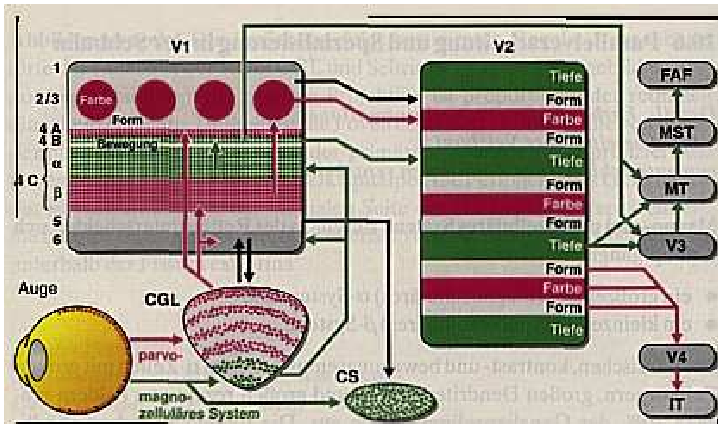


Abbildung 2.6: Organisation der magno- und parvozellulären Bahnen in den Gehirnarealen V1 und V2 im menschlichen Gehirn (aus [Sch95]).

ausgelöst [DLS96]. Auf dieser Karte repräsentiert ein Punkt eine bestimmte Augenbewegung zur Fixation eines Raumpunktes. Die Sakkaden werden primär auf der Grundlage der Signale aus den magnozellulären Bahnen — die Bewegung und Veränderung kodieren — und dem V1 generiert.

In Abbildung 2.7 wird die kortikale Organisation der unterschiedlichen Reizcharakteristika im V1 visualisiert. In den Orientierungssäulen sind diejenigen Neurone gebündelt, die ein orientierungsspezifisches Reizverhalten zeigen. Dieses führt im Querschnitt der Schichten zu einer topologischen Abbildung der Orientierungen. Die Schicht 5 stellt dabei eine Verbindung zum SC her. Dies bekräftigt die Hypothese, daß die Information aus dem V1 direkt die Aufmerksamkeit im SC verändern kann. Somit erhalten auch Reize wie Orientierung, Farbe und Form einen Einfluß auf die Ausführung der Augenbewegung. Dabei werden bestimmte Bewegungsmuster ausgelöst, die in der Bewegungskarte kodiert sind. Es wird angenommen, daß der stärkste Reiz in dieser Karte die Sakkade auslöst und die Muskeln entsprechend gesteuert werden, um die Fovea auf die nächste Blickposition zu lenken. Auf der Basis dieses neuronalen Aufmerksamkeitsapparates sind unterschiedliche Arten von Augenbewegungen beim Menschen realisiert. Wie diese im einzelnen aussehen und welchem Zweck sie dienen, wird im folgenden Abschnitt ausgeführt.

2.2 Unterschiedliche Augenbewegungsarten

Im vorangegangenen Kapitel 2.1 wurden die physiologischen Grundlagen des Sehapparates erklärt. Dabei wurden unterschiedliche Mechanismen des Okularsystems aufgezeigt, die für

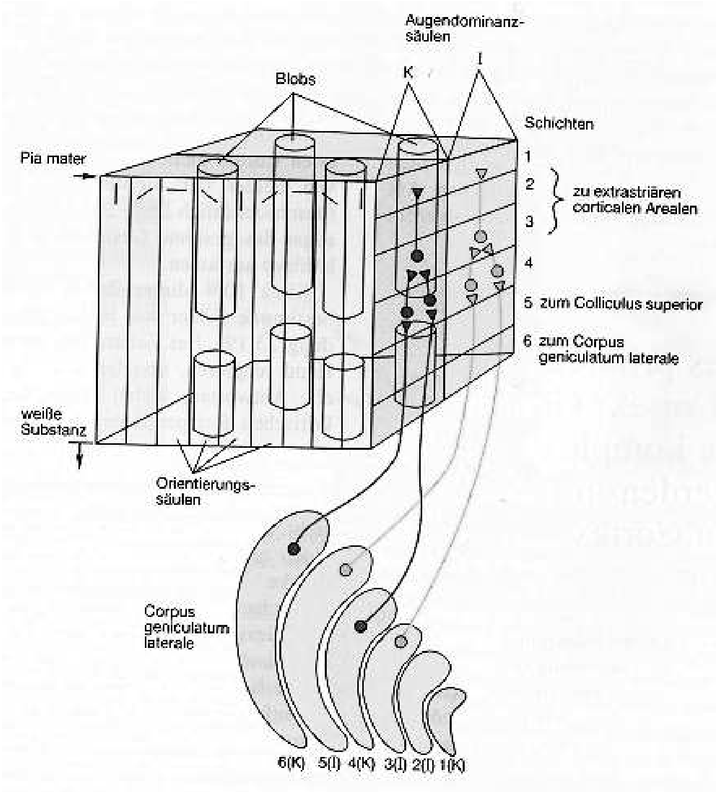


Abbildung 2.7: Organisation der Dominanzsäulen im primären visuellen Kortex V1 (aus [KSJ95]).

die Augenbewegungen zuständig sind. Bei der äußerlichen Betrachtung der Augenbewegungen beim Menschen ist schnell festzustellen, daß es eine Vielzahl unterschiedlicher Bewegungsformen gibt. Prinzipiell wird daher zwischen *weichen* (*smooth*) und *abrupten* (*saccadic*) Augenbewegungen unterschieden. Diese Bezeichnung bezieht sich auf die Bewegung des Augapfels, die zum einen weich und ohne erkennbares Zucken durchgeführt wird, und zum anderen abrupt und mit sehr großen Geschwindigkeiten stattfindet.

Zusätzlich werden die weichen Bewegungen noch in *disjunktive* und *konjunktive* Augenbewegungen unterteilt. Unter den disjunktiven Bewegungen wird die Ausrichtung der Augen auf ein bestimmtes Ziel verstanden. Dabei werden beide Augen in einer Bewegung auf die Entfernung und Position eingestellt (*Vergenzbewegung*). Durch Vergenzbewegungen können also Objekte in den Fixationspunkt beider Augen gebracht und somit in Stereo betrachtet werden. Bei der konjunktiven Bewegung hingegen werden bewegte Objekte in einer kontinuierlichen Blickanpassung fixiert. Da die Blickposition permanent an die neue Objektposition angepaßt werden muß, ergibt sich so eine weiche *Folgebewegung* (*smooth pursuit*). Dieser Mechanismus sorgt dafür, daß bewegte Objekte innerhalb der Fovea gehalten werden.

Ebenfalls zu den konjunktiven Augenbewegungen werden die *optokinetischen Bewegungen* gezählt. Dabei handelt es sich um einen Prozeß zur relativen Objektverfolgung. Diese tritt z.B. beim Durchqueren eines Raumes in Erscheinung: Während wir gehen, haben wir den Eindruck, daß sich die Objekte im Raum auf uns zu bewegen. Trotzdem wissen wir, daß die Objekte statisch im Raum stehen. An dieser Wahrnehmung sind weitere Mechanismen beteiligt, die diese komplizierte Aufgabe lösen, auf die aber hier nicht weiter eingegangen wird. Bei der Eigenbewegung kommt noch ein weiteres Problem hinzu: Der Kopf des Menschen bewegt sich, und trotzdem erhalten wir ein konstantes und stabiles Bild. Um dies zu erreichen, sind vestibulatorische Augenbewegungen nötig, die vom Vestibularorgan gesteuert werden. Dort wird die Beschleunigung des Kopfes gemessen und eine Gegenbewegung der Augenmuskeln ausgelöst, die das Bild auf der Retina stabilisieren.

Besondere Bedeutung kommt den abrupten Augenbewegungen zu. Auch sie werden nochmals unterteilt in *Sakkaden* und *Mikrosakkaden*. Die Mikrosakkaden dienen dazu, die foveale Position nach einer Sakkade nachzusteuern. Darüber hinaus verhindern sie ein Abdriften des Auges. Neben den Mikrosakkaden gibt es noch die *Anti-Sakkaden*, die einen Schutzmechanismus für das Auge darstellen. Sie sorgen dafür, daß bei unerwarteten plötzlichen Ereignissen (z.B. starker Lichteinfall) das Auge durch eine ruckartige, reflexgesteuerte Ausweichbewegung geschützt wird.

Die Modellierung von abrupten Sakkaden in einem Aufmerksamkeitsmodell steht im Mittelpunkt von Kapitel 5. Die Sakkaden bezeichnen Sprünge der Augen auf bestimmte Zielbereiche — die Fixationspunkte. Diese ruckartigen Bewegungen sind sehr schnell und können mit einer Winkelgeschwindigkeit von bis zu $700^o/s$ durchgeführt werden. Während einer Sakkade ist die visuelle Wahrnehmung unterbrochen. Am Ende einer Sakkade ist die Fovea auf das Zielobjekt gerichtet und verharrt dort (Fixation). Die Fixationsdauer, also die Länge des Verweilens des Auges zwischen den Sakkaden auf einem Bildpunkt, variiert in Abhängigkeit vom Stimulus und kann bei komplexen Objekten länger sein als bei uniformen, eher flächigen Stimuli.

Interessant ist in diesem Zusammenhang die Auswahl des nächsten Fixationspunktes auf der Basis der vorhandenen visuellen Stimuli („where-to-look-next“-Problem). Eine wichtige Rolle spielt dabei neben der Auswertung der fovealen Information der visuelle Eindruck in den Randbereichen. Mittels geeigneter Mechanismen können so Objekte, die nur schemenhaft am Randbereich zu erkennen sind, aufgrund der Form, Farbe oder anderen Eigenschaften das Ziel der nächsten Sakkade werden.

Ebenfalls einen Einfluß auf die Sakkadensteuerung haben auch kognitive Prozesse der visuellen Informationsverarbeitung in höheren Ebenen. Das wird deutlich bei der Beobachtung des Auges während unterschiedlicher visueller Aufgaben wie etwa beim Lesen eines Zeitungsartikels oder beim Verfolgen des Verkehrs bei einer Autofahrt. Um die Zusammenhänge zwischen visuellen Stimulus und den menschlichen Augenbewegungen zu untersuchen, werden entsprechende Eyetracking-Untersuchungen durchgeführt. Ein solches Experiment wird im folgenden Kapitel beschrieben und zeigt, welche visuellen Merkmale einen Einfluß auf die Sakkaden des menschlichen Auges haben.

2.3 Psychophysische Erkenntnisse

Die Aufmerksamkeit eines Menschen wird typischerweise durch verschiedene Reize wie etwa Geräusche, Schmerz oder visuelle Perzepte beeinflußt. Besonders unsere Augen sind ständig in Bewegung, obwohl wir dies trotz zum Teil sehr schneller Bewegungen nicht bewußt wahrnehmen. Der visuelle Fokus verschiebt sich in Abhängigkeit von der Aufgabe, die wir lösen möchten (z.B. das Lesen eines Textes), von Reflexen (z.B. zum Augenschutz) oder als Reaktion in Folge von besonders auffälligen Merkmalen im Blickfeld (z.B. Bewegung oder kräftige Farben). Da das menschliche Auge nur einen kleinen Bereich mit hoher Auflösung besitzt (die Fovea), sind Blickbewegungen wichtig für die detaillierte visuelle Betrachtung einer Szene. Die Evolution hat deshalb Mechanismen und Strategien hervorgebracht, um verschiedene visuelle Aufgaben wie etwa die visuelle Suche oder die Objekterkennung zu lösen. Viele Forscher sind daher an zugrundeliegenden Augenbewegungsverhalten und den dabei ablaufenden Prozessen interessiert. Für Experimente werden unter anderem moderne Eyetracking-Geräte eingesetzt, die einen Einblick in die kognitiven Mechanismen der menschlichen visuellen Wahrnehmung erlauben [Pom98],[Yar67].

Verschiedene Experimente haben aufgedeckt, welche Strategien Menschen zur Lösung visueller Aufgaben wählen. Besonders gut wurde dabei die visuelle Suche erforscht. Im Mittelpunkt stehen bei diesen Experimenten die Faktoren, die die Blicktrajektorien der Versuchspersonen beeinflussen. Deshalb wird dem Probanden ein Monitorbild präsentiert, auf dem ein spezieller Stimulus gezeigt wird. Der Proband soll dann eine Aufgabe lösen, etwa indem er sich das Bild einprägen muß, um in einem darauffolgenden Bild einen Unterschied zu detektieren. Während dieses Ablaufs werden mit einem Eyetracker die Fixationen mit Ort und Verweildauer fortlaufend aufgezeichnet. Aus den so gewonnen Blicktrajektorien werden mittels statistischer Methoden Zusammenhänge zwischen Blickverhalten und Stimulus vorhergesagt. Basierend auf diesen Erkenntnissen werden anschließend künstliche Modelle entwickelt, die menschliche Augenbewegungen simulieren können.

2.3.1 Untersuchungen im „Frühstücksszenario"

Ein im Vorfeld der Entwicklung des hier vorgestellten Aufmerksamkeitsmodells durchgeführtes Eyetracking-Experiment zeigt, wie Versuchspersonen Bilder von einem typischen „Frühstücksszenario“ betrachten[4] [FFP+98]. Dabei werden die Blicktrajektorien der Versuchspersonen aufgezeichnet, und mit dem Bildinhalt an den Blickpositionen in Zusammenhang gebracht. Die so gewonnenen qualitativen Aspekte aus den aufgezeichneten Blicktrajektorien stützen die Konstruktion des in Kapitel 5 dargestellten Augenbewegungsmodells.

Das Frühstücksszenario hat sich als besonders geeignet für die Augenbewegungsuntersuchungen herausgestellt, da die verwendeten Objekte (Gläser, Teller, Besteck etc.) systematisch ausgetauscht und an unterschiedlichen Orten plaziert werden können. Ebenfalls konnten problemlos gleiche Objekte in verschiedenen Farben eingesetzt und variiert werden. Zunächst wurde deshalb eine Bilddatenbank mit unterschiedlichen Anordnungen von Objekten aufgenommen. Die so gewonnenen Bilder wurden dann einer Reihe von Probanden vorgeführt und die bei der Betrachtung der Monitorbilder durchgeführten Augenbewegungen aufgezeichnet. Mittels statistischer Auswertungen sind anschließend Aussagen über bevorzugt fixierte Objekte und über die typische Aufmerksamkeitsverteilung der Versuchspersonen möglich.

2.3.2 Eyetracking-Experiment

Die Gestaltung des Eyetracking-Experiments dient der Aufzeichnung empirischer Augenbewegungsdaten bei der visuellen Exploration von realistischen Szenen. Um die Probanden dazu zu bewegen, die präsentierten Bilder zu betrachten und zu erforschen, wird zuvor eine Aufgabe gestellt. Die Probanden müssen dabei in zwei aufeinanderfolgenden ähnlichen Bildern eine Differenz entdecken. Die Abbildung 2.8 zeigt ein solches Bildpaar mit einigen typischen Objekten, die in unserem Experiment verwendet werden.

Die Szene besteht aus alltäglichen Frühstücksobjekten wie etwa Teller, Tassen und Besteck. Das gesamte Bildmaterial wurde aus einer festen Menge solcher Objekte in unterschiedlichen Anordnungen generiert. Allerdings besteht jeweils zwischen dem ersten Bild und dem zweiten Bild, das der Versuchsperson gezeigt wurde genau einer der folgenden Unterschiede:

- Eines der Objekte verschwindet.
- Ein zusätzliches Objekt erscheint.
- Ein Objekt ändert seine Farbe.
- Ein Objekt verändert seine Form.

[4] Diese Arbeiten wurden im Projektseminar „Modellierung menschlicher Augenbewegungen“ zusammen mit Dr. Marc Pomplun durchgeführt.

(a) (b)

Abbildung 2.8: Typische „Frühstücksszenen“: Erstes Bild A eines Durchgangs und zweites Bild B mit einem kleinen Unterschied, den es zu entdecken gilt — Die Tasse hat sich verändert.

Um diese Unterschiede feststellen zu können, müssen die Probanden sich also zunächst die Szene einprägen. Für die Einprägung des Bildes wird nur ein begrenzter Zeitrahmen von 20 Sekunden zur Verfügung gestellt. Damit soll erreicht werden, daß in erster Linie der visuelle Eindruck der Objekte eine Rolle spielt (Farbe, Form etc.), und weniger die Art und Funktion der Teile (Tasse, Teller etc.). Auf diese Art und Weise kann zwar das Objektwissen des Menschen nicht komplett unterdrückt werden, aber eine vollständige Erkennung und Memorisierung der gezeigten Objekte ist durch zeitliche Einschränkungen nicht möglich.

Die 16 Versuchspersonen waren Studenten der Universität Bielefeld, die keine Sehfehler aufwiesen. Die Stimulus-Bilder wurden auf einem Computermonitor mit einer Ortsauflösung von 640×480 Pixeln gezeigt. Die Bilder repräsentierten standardisierte Frühstückssituationen (siehe Abbildung 2.8). Die Größe der Frühstücksobjekte variierte zwischen einem Durchmesser von 2.5 und 15 Grad des Sehwinkels.

Die Augenbewegungsmessungen wurden mit einem Omnitrack 1-System durchgeführt [Sta93]. Dabei handelt es sich um ein monokulares nicht-invasives Eyetracking-System, das mit Bildverarbeitungshard- und -software sowie zwei Infrarot Kameras ausgestattet ist, die an einer Kopfbefestigung montiert sind. Mit Hilfe der ersten Kamera wird die Position der rechten Pupille der Testperson ermittelt. Die zweite Kamera dient zur Bestimmung der Kopfposition. Damit können Kopfbewegungen während der Versuchsphase erkannt und ausgeglichen werden. Aus den erfaßten Augenbildern werden die Fixationen der Personen auf dem Farbmonitor (17” Monitor und Sitzabstand 60cm) im Takt der Bildrate von 60Hz fortlaufend berechnet. Dabei werden nur Fixationen mit einer Fixationsdauer von über 83ms aufgezeichnet[5]. Neben den Bildschirmkoordinaten wird auch die Dauer und der Zeitpunkt der einzelnen Fixationen sowie die Pupillengröße während jeder Fixation registriert. Die absolute Genauigkeit der detektierten Bildschirmkoordinaten des kalibrierten Systems liegt bei ca. 0.7 bis 1.0 Grad des Sehfeldes. Das entspricht etwa 7 bis 10mm auf dem Monitor.

Die Versuchspersonen wurden einzeln getestet. In jedem Durchgang wurde der Person eine Folge von drei Bildern gezeigt. Als erstes ein „Einprägebild" A des Frühstücksszenarios (15 Sekunden lang), als zweites ein Distraktorbild[6] (5 Sekunden), und drittens wiederum ein Frühstücksbild B (maximal 10 Sekunden). In dem letzten Bild mußten die Versuchspersonen den einen Unterschied zur Szene aus Bild A entdecken. Der Durchgang wurde beendet, wenn die Person den Unterschied ausgemacht hatte und dieses mittels eines Druckknopfes signalisierte. Jede der Personen mußte 20 Sequenzen solcher Bilder betrachten. Vor jeder Betrachtung einer Sequenz mußte das System mit einem Kalibrierungsmuster rekalibriert werden.

2.3.3 Ergebnisse

Die Analyse der Augenbewegungen wird im folgenden auf die Betrachtungsphase des „Einprägebildes" A beschränkt. Die Abbildung 2.9 zeigt einen typischen Betrachtungspfad einer Versuchsperson. Die kleinen Quadrate markieren den Fixationspunkt, und die Helligkeit an den Fixationspositionen visualisiert die Fixationsdauer an dieser Stelle. An dieser Darstellung läßt sich leicht erkennen, daß in diesem Fall besonders das Glas und die Tasse die Aufmerksamkeit der Person angezogen haben.

Die Abbildung 2.10(b) zeigt die Anzahl der Fixationen pro Durchgang als eine Funktion über den Verlauf der jeweils 20 Durchgänge. Es ist zu erkennen, daß die Anzahl der Fixationen gegen Ende der 20 Durchgänge leicht abnimmt. Dies kann durch eine Varianzanalyse (ANOVA) belegt werden: Die mittlere Anzahl der Fixationen pro Durchgang ist in den Durchgängen 1 bis 10 (36.15) signifikant höher als in den Durchgängen 11 bis 20 (33.24) ($F(1, 159) = 27.56; MSE = 68.27; p < 0.0001$). Dieser kleine, aber signifikante Unterschied zeigt, daß die Versuchspersonen in der Lage sind, von dem relativ konstanten Aufbau der Frühstücksszenen zu profitieren. Sie lernen also bei der Betrachtung der Stimuli, an welcher Stelle typischerweise welche Objekte auftreten, und müssen sich somit nicht mehr die gesamte Szene einprägen. Damit werden nicht mehr so viele Fixationen während der Einprägung benötigt.

[5] Dies ist eine technische Beschränkung des Eyetracking-Systems. Neuere Geräte können auch kürzere Verweildauern aufzeichnen.

[6] Dieses Bild zeigt einen völlig anderen Kontext und soll den Effekt des „Herausspringens" des Unterschieds beim Betrachten der Bilder A und B verhindern.

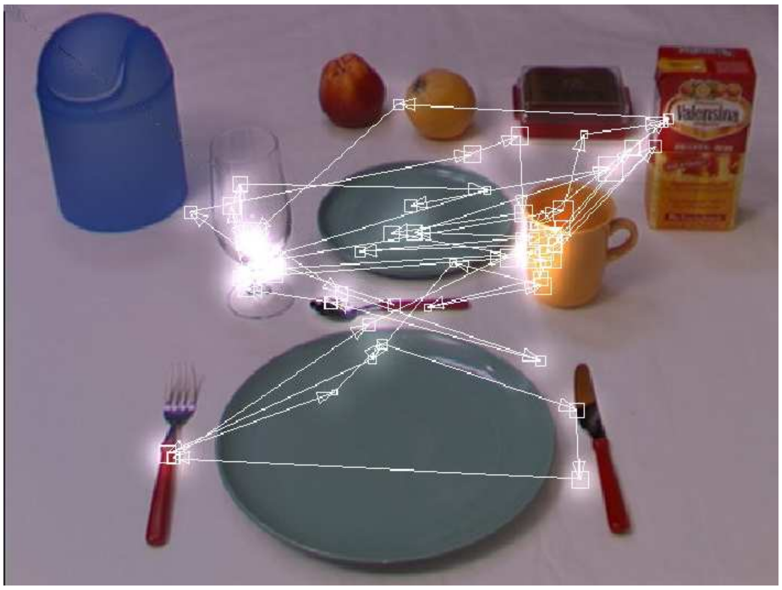
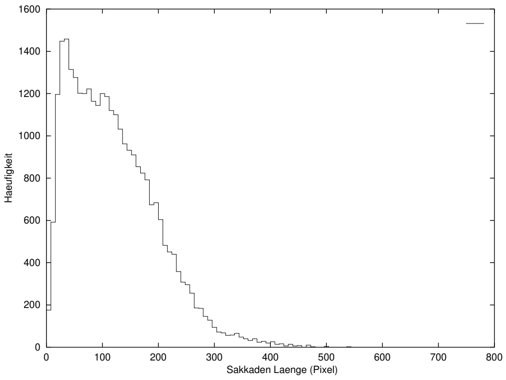
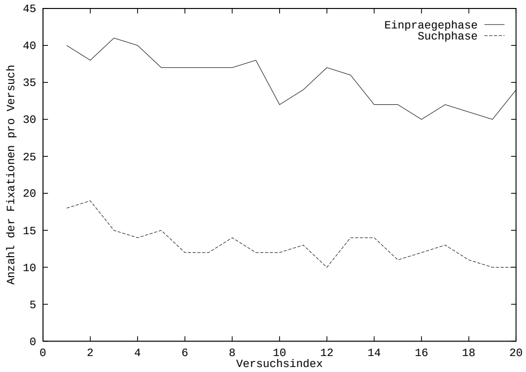


Abbildung 2.9: Beispiel der Blicktrajektorie einer Versuchsperson bei der Betrachtung der Szene. Die Größe des Quadrats visualisiert die Dauer der zugehörigen Fixation. Die Umgebungen der Fixationen wurden aufgehellt, um die Aufmerksamkeitsbereiche hervorzuheben.



(a)



(b)

Abbildung 2.10: (a) Histogramm der Sakkadenlänge für alle Versuchspersonen und alle Einprägungsstimuli; (b) Anzahl der Fixationen während eines Durchgangs gemittelt über alle Personen.

Eine quantitative Auswertung der Blicktrajektorien basiert auf der Umwandlung der Fixationen in Blickpfade, die die Blickbewegungen als Folge der betrachteten Objekte beschreiben.

Dafür wurde jedem Objekt ein elliptisches Gebiet zugeordnet. Diese „Aufmerksamkeitsfläche“ umfaßt das eigentliche Objekt sowie einen schmalen Rand, um Meßungenauigkeiten auszugleichen. Alle Fixationen, die in diesem Bereich landen, werden dem zugehörigen Objekt zugerechnet. Fixationen, die zwischen den Objektbereichen detektiert werden, werden einer separaten Gruppe zugeschlagen. Diese Gruppe wird mit „Orientierung“ bezeichnet, da sie vermutlich der visuellen Orientierung in der Szene dienen. Darüber hinaus wurden die Objekte in unterschiedliche Gruppen von gleicher Farbe und gleicher Form eingeteilt. Damit kann untersucht werden, ob die Versuchspersonen bestimmte Übergänge zwischen spezifischen Kombinationen von Objektfarben- oder formen bevorzugen. Alle verwendeten Frühstücksutensilien und die zugehörigen Attribute sind in Tabelle 2.1 dargestellt.

		Farbe	Form
a	Mülleimer	Hellblau	Komplex
b	Apfel	Rot	Rund
c	Orange	Orange	Rund
d	Butterdose	Rot	Rechtwinkelig
e	Kleiner Teller	Grün	Rund
f	Tasse	Orange/blau	Komplex
g	Glas	Transparent	Komplex
h	Karton	Orange	Rechtwinkelig
i	Löffel	Silber	Langgezogen
j	Messer	Silber	Langgezogen
k	Gabel	Silber	Langgezogen
l	Großer Teller	Grün	Rund

Tabelle 2.1: Die Frühstücksobjekte mit den zugehörigen Farb- und Formattributen.

Die Tabellen 2.2 und 2.3 zeigen die standardisierten Übergangshäufigkeiten in Bezug auf die vorkommenden Farb- und Formkombinationen. Besonders auffällig bei den Farbübergängen ist die Häufigkeit, mit der zwischen Rot und Orange gewechselt wird. Daran kann abgelesen werden, daß diese Objekte auch bevorzugt fixiert werden. Diese Fixationen werden offensichtlich durch den Signalcharakter dieser Farben ausgelöst. Ähnliches kann bei den Formübergängen festgestellt werden: Die komplexeren Objekte wie das Glas und die Tasse werden ebenfalls bevorzugt angeschaut. Das Auge scheint also auch von komplexeren Bildregionen angezogen zu werden, bei denen eine höhere Zahl an Ecken und Kanten auftreten.

Die Ergebnisse aus den Eyetracking-Untersuchungen zeigen einige charakteristische Eigenschaften des menschlichen Blickverhaltens. Schon die Betrachtung dieser einfachen Früh-

	Hellblau	Rot	Orange	Grün	Blau	Transp.	Silber	Orient.
Hellblau	0	195	63	95	1	248	45	88
Rot	211	383	1000	440	85	218	85	252
Orange	100	915	474	925	10	463	375	170
Grün	119	446	779	229	130	729	974	213
Blau	7	73	7	147	0	67	53	13
Transparent	177	248	443	839	51	0	547	125
Silber	26	72	304	940	51	487	632	114
Orientierung	60	260	214	245	15	120	103	0

Tabelle 2.2: Übergangshäufigkeit zwischen den auftretenden Farbattributen (Normalisiert auf einen Maximalwert von 1000).

	Rund	Rechtwinkelig	Komplex	Langgezogen	Orient.
Rund	596	598	974	475	181
Rechtwinkelig	618	364	795	246	178
Komplex	1000	835	858	616	194
Langgezogen	463	214	532	407	74
Orientierung	220	182	186	66	0

Tabelle 2.3: Übergangshäufigkeiten zwischen den auftretenden Formattributen (Normalisiert auf einen Maximalwert von 1000).

stücksszenen erfordern eine hohe Anzahl an Fixationen. Bei der Betrachtung dieser Stimuli zeigen sich trotz der stark unterschiedlichen einzelnen Blicktrajektorien zwischen den Versuchspersonen Gemeinsamkeiten. Obwohl in Abbildung 2.10(b) ein gewisses Lernverhalten zu beobachten ist, spielen auch so primitive Bildattribute wie Farben und Kanten eine Rolle bei der Auswahl des nächsten Fixationspunktes. Es wurden in anderen Arbeiten viele weitere Untersuchungen zu diesem Thema angestellt (siehe auch weiterführende Literatur in [Pom98]), auf deren Grundlage fundierte Aussagen über die dem Blickverhalten zugrundeliegenden Mechanismen gemacht werden können. Innerhalb des Frühstücksszenarios ist jedenfalls deutlich festzustellen, daß besonders hervortretende Merkmale wie Farben und Reflexionen potentielle Ziele für Sakkaden darstellen. Auch der Übergang von runden zu komplexen Gegenständen ist auffällig. Dies ist ein Hinweis darauf, daß auf einer niedrigen Ebene im Sakkadensystem aufgenommene Reize auch reflexartig eine Sakkade auslösen können. Gleiches gilt sicher auch für das Merkmal Bewegung. Jeder weiß aus Erfahrung, daß unser Blick reflexartig auf plötzlich im Blickfeld auftauchende Objekte gelenkt wird. Allerdings sind alle Blicksteuerungsprozesse überlagert durch bewußte, aufgabenabhängige Blickstrategien.

Das Wissen aus dem Experiment fließt in die Modellierung des künstlichen Sakkadenmodells in Kapitel 5 ein, und gibt Hinweise auf die Auswahl der zu extrahierenden Bildmerkmale (Kanten, Farben etc.). Prinzipiell können aber auch die quantitativen Ergebnisse genutzt werden. In einer ersten Version des Aufmerksamkeitsmodells wurden die gewonnen Aufmerksamkeitsbereiche aus den Experimenten genutzt, um das adaptive mehrschichtige Modell zu trainieren und die Auswahl des Fixationspunktes möglichst an die Daten anzupassen [FFP+98]. Um anschließend überprüfen zu können, ob die vom Modell generierten Blicktrajektorien mit denen der Versuchspersonen korrespondieren, wurde von Ahmed Faisal ein Distanzmaß zum Vergleich von Blicktrajektorien entwickelt, das Markov-Pfad Distanzmaß (zu Einzelheiten siehe [FFP+98]). Auf der Grundlage dieses Distanzmaßes konnte gezeigt werden, daß die künstlichen Blicktrajektorien qualitativ eine große Ähnlichkeit mit den natürlichen Blickbewegungen aufweisen. Die Untersuchungen bezogen sich allerdings nur auf statische Stimuli und eine feste Betrachterperspektive. Da in dieser Arbeit dynamischen Szenarien und bewegten Kameras eine zentrale Bedeutung zukommt, wird hier nicht weiter auf diese Vorarbeiten eingegangen.

In die im weiteren realisierten Systeme zur Mensch-Maschine-Interaktion sind neben technischen Nachbildungen auch konkrete Aspekte aus den theoretischen Erkenntnissen über biologische Sehsysteme eingeflossen:

- Es werden aktive Kamerasysteme für die Bildaufnahme verwendet, die ähnliche Bewegungsfreiheitsgrade wie die Augen aufweisen.
- Die Merkmalsextraktion basiert ähnlich wie beim Menschen auf rezeptiven Feldern, wobei besonders die Merkmale Farbe, Kanten, Orientierung und Bewegung berücksichtigt werden.
- Viele Module verwenden adaptive Mechanismen auf der Grundlage künstlicher neuronaler Netze zur besseren Anpassung an die aktuellen Umweltbedingungen.
- Die Systemarchitektur besteht aus kleinen Modulen die interaktiv miteinander kommunizieren. Die Verschaltung dieser Module orientiert sich an der Organisation des okulomotorischen System beim Menschen.

Das Ziel ist die Entwicklung von Systemen, die ähnlich wie die natürlichen Vorbilder robust und zuverlässig funktionieren, ohne diese im Detail zu modellieren[7]. Anders als bei klassischen Ansätzen wird hier kein monolithisches System aufgebaut, in das alle Systemzustände fest einprogrammiert sind. Statt dessen sind die folgenden Systeme darauf ausgelegt, zu

[7] Was schon aufgrund der z.Zt. noch unzulänglichen Hardwareressourcen ohnehin nicht möglich ist!

reagieren und sich auf die aktuellen Arbeitsbedingungen einzustellen. Damit werden sie den hohen Anforderungen bei der Mensch-Maschine-Kommunikation gerecht.

3 Grundlagen der verwendeten Neuronalen Netze

EINE Grundeigenschaft höherentwickelter Lebewesen ist die Lernfähigkeit. Angefangen von einfachen „Schaltkreisen“, die etwa das Laufverhalten von Heuschrecken steuern, bis hin zum menschlichen Gehirn haben sich intelligente „Denkmaschinen“ entwickelt, die auf komplexen zellulären Strukturen basieren. Dazu haben sich Zellen zu komplexen neuronalen Systemen zusammengeschlossen. Die Grundbausteine bilden die Nervenzellen (Neuronen). Sie empfangen Signale und geben neue, von ihnen erzeugte, Signale weiter. Die Neuronen besitzen dendritische Fortsätze, die die Eingangsinformation sammeln. Die Dendriten empfangen dabei Signale von anderen Nervenzellen an spezifischen Schaltstellen, den Synapsen. Die Ausgabesignale des Neurons werden wiederum über das Axon an andere Zellen weitergeleitet. Die Signalverarbeitung basiert auf elektro-chemischen Prozessen und die Übertragung der Information erfolgt mittels Aktionspotentialen. Nach heutigen Annahmen basiert die Verarbeitung der Information innerhalb der Neuronen auf der Summation der ankommenden Signale. Dabei können eingehende Signale auch eine Hemmung und somit eine Verzögern neuer Aktionspotentiale verursachen. Vereinfacht ausgedrückt wird durch das Axon ein Ausgangssignal gesendet, wenn die Summe von erregenden und hemmenden Signalen eine Reizschwelle überschritten haben.

Die Speicherung von Information erfolgt den aktuellen Erkenntnissen der Forschung zufolge in den synaptischen Kontaktstellen. Durch die Öffnung der Ionenkanäle, die sich in der Membran der Zielzelle befinden, kann die Reizschwelle einer Zelle beeinflußt werden. Hierbei kann die einlaufende synaptische Information im Verlauf zu einer Veränderung der Reizschwelle führen und somit ein geändertes Feuerverhalten der Zielzelle auslösen. Dieser komplexe Vorgang, der auf chemischen Botenstoffen basiert, die auf Rezeptoren wirken, sorgt so für die Adaptivität der Zelle. Das wiederum wirkt sich in der Summe auf das Gesamtverhalten des neuronalen Schaltkreises aus.

Besonders im menschlichen Gehirn haben sich große Netzwerke gebildet, die aus vielen Millionen miteinander vernetzten Nervenzellen bestehen. Der Mensch wird damit in die Lage versetzt, eine Vielzahl an sensorischen und motorischen Perzepten gleichzeitig zu verarbeiten. Die Informationsverarbeitung erfolgt dabei parallel in allen Neuronen, die über die dendritischen Fortsätze verknüpft sind. Die Schaltstellen (Synapsen) können dabei die Ein- und Ausgangssignale eines Neurons adaptiv ändern. Das Gehirn ist darüber hinaus hierarchisch in funktionale Bereiche gegliedert. Es gibt etwa Areale, die überwiegend für die visuelle Wahrnehmung oder die Ansteuerung der Muskeln zuständig sind. Der Mensch kann sich dadurch optimal an seine Umwelt anpassen und gelerntes Wissen behalten.

Was der Mensch mit Leichtigkeit beherrscht, ist für den Computer bisher eine anscheinend unlösbare Aufgabe. Viele Probleme lassen sich nicht einfach als „starres“ Programm formalisieren, da sich die Umwelt dynamisch ändert und somit eine permanente Anpassung des Programmablaufs nötig wird. *Künstliche neuronale Netze* (KNN) bieten hier einen Ansatzpunkt. Sie stellen eine mathematische Nachbildung der parallelen Informationsverarbeitung in biologischen Nervensystemen dar. Ähnlich wie beim Vorbild sind Programme, die auf neuronalen Netzen basieren, lernfähig und fehlertolerant. Allerdings sind die künstlichen Netze aufgrund mangelnden theoretischen Wissens nur eine sehr vereinfachte Simulation der biologischen Nervensysteme. Es sind daher verschiedene Netzwerktypen entwickelt worden, die verschiedene Aspekte des natürlichen Vorbilds unterschiedlich genau reproduzieren (siehe auch [Roj93]). Die hier verwendeten Netze können durch Lernen aus Beispielen ihre Klassifikationsgüte verbessern. Dafür wird grob die Struktur des Netzes vorgegeben, und in der anschließenden Lernphase durch das Präsentieren von Beispielen mit den dazugehörigen Erkennungsergebnissen an das jeweilige Problem angepaßt. Dieser Vorgang der internen Organisation der Netzwerkstruktur wird als *überwachtes Lernen* bezeichnet. Ist die Lernphase beendet, ist das Netz je nach Trainingserfolg in der Lage, ähnliche Eingabemuster wie die Trainingsbeispiele zu klassifizieren und den gewünschten Ausgabewert zu erzeugen. Das erlernte Wissen liegt anschließend *implizit* in Form von Gewichtsparametern vor.

Wie kann eine mathematische Modellierung eines neuronalen Netzes aussehen? Allgemein besteht ein künstliches neuronales Netz aus einer Menge von *Knoten*, welche die eingehenden Signale verarbeiten. Das berechnete Ausgangssignal wird über *gewichtete Verbindungen* an die nachfolgenden Knoten weitergeleitet. Die Art und Struktur der Knoten und Verbindungskanten bestimmt den Typ des neuronalen Netzes. Ein neuronales Netz bildet also Eingabevektoren auf Ausgaben ab und verwendet dazu eine Funktion mit zahlreichen Parametern. In der Vergangenheit sind Netzwerke entwickelt worden, die durch ihre Topologie eine große Vielfalt derartiger parametrisierter Funktionen implementieren. Dabei wird im wesentlichen zwischen vorwärtsgerichteten und rekursiven Netzen unterschieden, die je nach Aufgabenstellung ihre Verwendung finden. In dieser Arbeit wird zum einen das Multilagen-Perzeptron (MLP) eingesetzt, und zum anderen die „Local Linear Map“ (LLM), ein spezieller Netztyp auf der Basis selbstorganisierender topologieerhaltender Karten. Beide Netztypen werden mittels überwachten Training an ihre Aufgabe adaptiert.

Im folgenden werden die beiden in dieser Arbeit verwendeten Netzwerktypen MLP und LLM näher beschrieben. Diese Netzwerktypen sind besonders für den Einsatz in der Bildverarbeitung und der Robotik geeignet und haben sich in vielerlei Anwendungen bewährt (siehe etwa [BP94] und [Dre95, LDR95, Hei98]).

3.1 Multilagen-Perzeptron MLP

Das *Multilagen-Perzeptron* (MLP) basiert auf dem von Rosenblatt entwickeltem *Perzeptron-Modell* [Ros58]. Das Perzeptron besitzt zwei Schichten: eine *Eingabe-* und eine *Ausgabeschicht*. Es ist damit in der Lage, Abbildungsrelationen zwischen Mustern zu erlernen, d.h. ein Eingabemuster ruft ein bestimmtes Ausgabemuster hervor. Allgemein wird dabei also die Menge der Eingabemuster auf die Menge der Ausgabemuster abgebildet. Auf der Eingabeseite besitzt das Perzeptron eine Anzahl von Neuronen, die der Dimension des Eingabemusters entspricht. Wird nun ein Eingabemuster angelegt, breitet sich die Aktivität zur Ausgabeschicht aus, wo ein Ausgabesignal entsteht. Ein neuronales Netz, bei dem sich die Ausbreitung der Aktivierung von der Ein- zur Ausgabeseite entwickelt, wird als *vorwärtsgerichtetes Netz* bezeichnet. Die Dimension der Ausgabeschicht kann dabei unabhängig von der der Eingabeschicht gewählt werden.

Die Knoten des Perzeptrons sind über Kanten verbunden, die wiederum mit Gewichtsparametern versehen sind. Diese Gewichtsparameter bestimmen das Ausgabeverhalten des neuronalen Netzes. Das Ziel ist es nun, den Eingabemustern das gewünschte Ausgabemuster zuzuordnen um etwa Objektmerkmalen eine gewünschte vordefinierte Objektklasse zuzuteilen. Das kann durch eine geeignete Wahl der Gewichte erreicht werden, bei denen an der Ausgabeseite genau dasjenige Muster entsteht, das in der Zielmenge dem zugehörigen Eingabemuster entspricht. Damit diese Gewichte korrekt ermittelt werden, wird ein Netz *trainiert*. Dafür wird dem Netz in einer Sequenz von Lernschritten eine Menge an Eingabemustern zusammen mit den dazugehörigen Ausgabemustern gezeigt. Die Abweichung zwischen dem Ein- und Ausgabemuster wird dann zum Adaptieren der Gewichte verwendet. Dieser Vorgang wird für alle Musterpaare öfters wiederholt. Ein Durchgang über alle Trainingspaare wird dabei mit dem Begriff *Epoche* bezeichnet. Der Trainingsprozeß wird abgebrochen, sobald die Abweichung zwischen dem Eingabemuster und dem gewünschten Ausgabemuster für alle Trainingspaare eine bestimmte Grenze unterschritten haben. Für diese *Fehlerminimierung* werden unterschiedliche *Lernverfahren* eingesetzt.

Im Gegensatz zum zweischichtigen Perzeptron kommen beim MLP drei oder mehr Neuronenschichten zum Einsatz. Die erste Schicht dient als Eingabeschicht, während die letzte Schicht die Ausgabe bereitstellt. Die Neuronen der dazwischen befindlichen Schichten werden *innere*

Neuronen (*hidden neurons*) genannt und sind nur für die interne Aktivitätsausbreitung zuständig. Damit kann das einfache Perzeptron-Modell erweitert werden, um komplexere Probleme zu lösen (siehe dazu etwa [Roj93]).

3.1.1 Informationsverarbeitung im MLP

Auch beim MLP ist die Ausbreitung der Aktivitäten vorwärtsgerichtet, und die Information wird von den Eingabeneuronen über die mittleren Schichten zur Ausgabeschicht weitergeleitet (*forward propagation*). Die Eingabeneuronen übertragen das Eingabesignal direkt an die erste Zwischenschicht. Dort wird das Signal von den Neuronen aufgenommen und mit einer *Aktivierungsfunktion* zu einem Ausgabesignal an die nächste Schicht berechnet. Ein Neuron der Zwischenschicht kann Eingaben von mehreren vorgelagerten Neuronen erhalten, daher wird der Netto-Input am einzelnen Neuron berechnet. Unter der Annahme, daß i und j zwei aufeinanderfolgende Neuroneschichten sind, und Schicht i vor Schicht j liegt, berechnet sich der Netto-Input, also die Summe aller Eingaben, net_j eines Neurons in der Schicht j zu:

$$net_j(t) = \sum_{i=1}^{N} w_{ij} \cdot o_i(t) + w_{0j} \cdot \Theta_j(t) \tag{3.1}$$

Es wird zunächst die Summe über alle N Verbindungen, die an Neuron j von den Neuronen der Schicht i ankommen, berechnet. Das eingehende Signal von einem vorgelagerten Neuron i ist dabei das Produkt aus dem Ausgabewert $o_i(t)$ des Neurons i, und dem Verbindungsgewicht w_{ij}. Der Bias stellt ein Verbindungsgewicht zu einem Neuron dar, das ein konstantes Signal der Höhe Eins liefert.

Die *Aktivierung* $a_j(t)$ des Neurons j wird mit einer *nicht-linearen Aktivierungsfunktion* berechnet, die im allgemeinen eine *sigmoide Funktion* darstellt:

$$a_j(t) = \frac{1}{1 + e^{-net_j(t)}} \tag{3.2}$$

Durch die Wahl der nicht-linearen Aktivierungsfunktion kann der Vorteil der zusätzlichen Neuronenschichten ausgenutzt werden. Die sigmoide Aktivierungsfunktion sorgt dafür, daß die möglichen Aktivationswerte auf ein Intervall von $(0, 1)$ beschränkt sind. Somit werden Eingabewerte nahe 0 stärker auseinandergezogen, und besonders große bzw. kleine Eingabewerte werden auf die Werte nahe 1 bzw. 0 abgebildet. Die Ausgabefunktion eines Neurons im MLP ist die Identität, d.h. der Ausgabewert $o_j(t)$ entspricht der Aktivierungsfunktion $a_j(t)$. Die Informationsausbreitung im MLP kann somit sequentiell von den Eingabe- zu den Ausgabeneuronen berechnet werden, um die Netzausgabe zu bestimmen.

3.1.2 Lernen im MLP

Damit das Netz die vorgegebene Menge von Eingabemustern auf die entsprechenden Zielmuster möglichst ohne Fehler abbilden kann, müssen die Gewichtswerte w_{ij} der Netzverbindungen an die jeweilige Aufgabe angepaßt werden. Dafür wurde ein Lernverfahren entwickelt, das den quadratischen Fehler E über alle Trainingspaare als Maß für die Performanz eines Netzwerkes betrachtet:

$$E = \sum_{p=1}^{K} E_p \qquad \text{und} \qquad E_p = \sum_{j=1}^{M} (t_{pj} - o_{pj})^2 \tag{3.3}$$

Die Bezeichnung p indiziert die Musterpaare K und der Index j die Ausgabeneuronen M. Der *Lernfehler* eines Ausgabeneurons j berechnet sich also über die quadratische Differenz zwischen dem vorgegebenen Zielwert t_j und dem Ausgabewert o_j. Der Gesamtfehler E_p für ein zu lernendes Muster p ist die Summe der Lernfehler über alle Ausgabeneuronen. Durch die Veränderung der Verbindungsgewichte w_{ij} kann also der Fehler eines Neurons und somit auch der Gesamtfehler E des Netzwerkes verändert werden.

Ziel des Lernverfahrens ist es nun, diesen Gesamtfehler mit Hilfe eines Gradientenabstiegsverfahrens zu minimieren. Dabei werden die einzelnen Gewichte w_{ij} mittels eines Fehlergradienten verändert. Die Idee liegt darin, die Gewichte proportional zur Steigung der Fehlerfunktion zu verändern. Dafür wird der Quotient der Veränderung des Fehlers und der Gewichtsänderung ermittelt. Die Gewichtsänderung berechnet sich somit zu:

$$\Delta w_{ij} = -\epsilon \cdot \frac{\partial E_p}{\partial w_{ij}} \tag{3.4}$$

Dabei stellt ϵ einen Proportionalfaktor dar –die *Lernrate*. Die Vorzeichenumkehr sorgt dafür, daß die Richtung der Gewichtsänderung zur Minimierung des Gesamtfehlers führt. Die *verallgemeinerte Delta-Regel* beschreibt, wie mit der abgeleiteten nicht-linearen (aber differenzierbaren) Aktivierungsfunktion das Fehlersignal δ_j eines Neurons j berechnet werden kann:

$$\delta_j = (t_j - o_j) \cdot F'(net_j) \tag{3.5}$$

Nach der Ableitung, und unter der Annahme einer Identitätsfunktion als Ausgabefunktion, kann das Fehlersignal eines Ausgabeneurons berechnet werden:

$$\delta_j = (t_j - o_j) \cdot o_j \cdot (1 - o_j) \tag{3.6}$$

Für die Neuronen i der inneren Schicht kann kein korrekter Ausgabewert ermittelt werden. Deshalb muß der Lernfehler adäquat ersetzt werden. Unter Verwendung der Kettenregel kann der Lernfehler der internen Neuronen rekursiv berechnet werden:

$$\delta_i = \left(\sum_{j=1}^{N} w_{ij} \cdot \delta_j \right) \cdot o_i \cdot (1 - o_i) \tag{3.7}$$

Die Berechnung erfolgt dabei *rückwärts*, d.h. zunächst wird der Fehler in der übergeordneten Schicht ermittelt, und anschließend die Fehlersignale der aktuellen Schicht. Die Informationsausbreitung erfolgt also entgegengesetzt zum üblichen Aktivierungsfluß. Deshalb wird das Verfahren auch *Backpropagation* genannt. Das *überwachte* Lernen erfolgt nun in drei Phasen.

In der ersten Phase werden die Trainingsmuster an die Eingabeneuronen angelegt. Anschließend wird die Information der Eingabeschicht durch das Netzwerk propagiert. In der zweiten Phase wird der aktuelle Ausgabewert der Ausgabeneuronen mit dem gewünschten Zielmuster verglichen. Der so ermittelte Lernfehler wird entgegengesetzt durch das Netzwerk zurück geleitet. Dabei werden die Gewichtsänderungen der internen Schichten berechnet und in der dritten Phase entsprechend angepaßt. Dieser Vorgang erfolgt entweder nach jedem Lernschritt (*musterweises Lernen*), oder nach dem Durchlaufen aller Trainingsmuster (*Epochenlernen*). Beim Lernen in Epochen werden die Verbindungsänderungen der einzelnen Schritte zunächst aufsummiert und erst nach Ende einer Epoche angewendet.

Beim Backpropagation-Lernverfahren haben die unterschiedlichen Netzparameter (Knotenanzahl, Verbindungsstruktur etc.) einen erheblichen Einfluß auf die Performanz. Neben der Lernrate ϵ können zusätzliche Parameter eingeführt werden, die die Konvergenz des Lernverfahrens verbessern und beschleunigen können. Dazu zählt etwa die Einführung eines *Impulsterms*, das dafür sorgt, daß kurzfristige Schwankungen in der Gewichtsänderung ausgeglichen werden [PNH86]. So ergibt sich eine Art „Erinnerungseffekt“, und der Lernprozeß erfolgt kontinuierlicher. Eine Beschleunigung des Backprop-Lernens wird zum Beispiel durch den *RProp*-Algorithmus erreicht [RB93]. Dabei wird die Schrittweite der Gewichtsanpassung nicht mehr global sondern lokal bestimmt und automatisch angepaßt. Allerdings ist die richtige Einstellung aller Netzparameter nicht trivial, deshalb sei zu weiteren Einzelheiten des MLP-Netzes auf die weiterführende Literatur verwiesen (siehe zum Beispiel [Roj93]). Das MLP wird im Rahmen dieser Arbeit für die Erkennung von Zeigegesten eingesetzt (Kapitel 5.7). Das MLP hat sich für die Klassifikation anhand der Merkmalsvektoren als besonders geeignet erwiesen.

3.2 Local Linear Map LLM

Das LLM-Netz ist ein weiterer Netzwerktyp, der im Rahmen dieser Arbeit Verwendung findet. Die *Local Linear Map* wurde von Helge Ritter entwickelt [Rit91], und ist eine Erweiterung der *selbstorganisierenden Karten* (*self-organized maps*; siehe auch [Koh84]). Die selbstorganisierenden Karten (SOM) passen ihre interne Struktur durch ein *unüberwachtes* Lernen an die Trainingsmuster an. Im Gegensatz dazu werden die LLM-Netze mit *überwachtem Lernen* trainiert. Die Struktur eines LLM-Netzes ist in Abbildung 3.1 gezeigt.

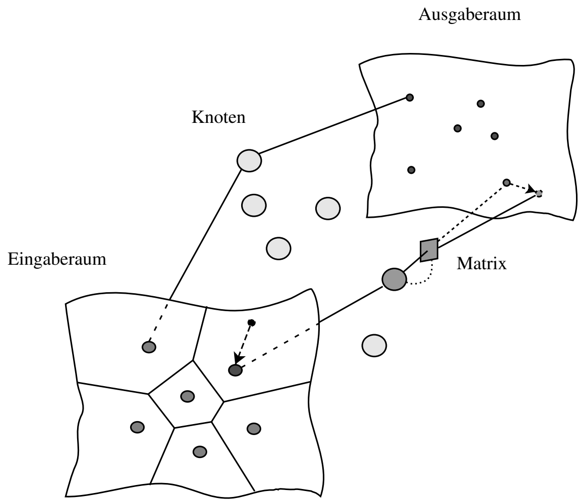


Abbildung 3.1: Darstellung der schematischen Netzstruktur eines LLM-Netzes (aus [Dre95]).

Es wird unterschieden zwischen dem *Eingabe-* und dem *Ausgaberaum*. Dazwischen befindet sich *eine* Schicht an internen Knoten, die über *Referenzvektoren* mit beiden Räumen verbunden sind. Die Lage der Referenzvektoren bestimmt die Zugehörigkeit eines Knoten zu einem Punkt im Ein- bzw. Ausgaberaum. Darüber hinaus wird jedem Knoten eine Matrix (die lokal-lineare Abbildung) zugeordnet. Bei Eingabe eines zu klassifizierenden *Merkmalsvektors* wird derjenige Knoten aus der Gesamtheit der Knoten ausgewählt, dessen eingabeseitiger Referenzvektor der Eingabe am ähnlichsten ist. Dadurch ergibt sich eine Art „Voronoi-Parkettierung“ des Eingaberaums, so daß jeder Knoten einen eigenen Zuständigkeitsbereich besitzt und für diesen Bereich die Abbildung von Eingaben auf Ausgaben bestimmt. Diese Bereiche, die den jeweiligen Knoten umhüllen, sind in der Abbildung 3.1 ebenfalls dargestellt. Die Ausgabe des LLM-Netzes kann dabei auf zweierlei Weise berechnet werden: In einem Fall bestimmt allein der ausgewählte Knoten die Ausgabe („winner-takes-all-Regel“), im anderen Fall wird die Ausgabe aller Knoten mit einer Ausgabefunktion überlagert, und die gewichtete Summe stellt die Ausgabe des Netzes dar.

Die Antwort eines Knotens wird in erster Linie durch den ausgabeseitigen Referenzvektor festgelegt. Dieses gilt aber nur für den Fall, in dem der zugehörige Knoten im Eingaberaum genau getroffen wird. Liegt der Eingabewert neben diesem Punkt, so muß die Abweichung entsprechend im Ausgaberaum berücksichtigt werden (siehe Pfeil in Abbildung 3.1). Hier kommt die dem Knoten zugeordnete Matrix zum Einsatz, die eine lokale lineare Abbildung repräsentiert. Mit Hilfe dieser Abbildung wird eine ausgabeseitige Korrektur des Referenzvektors durchgeführt (siehe Pfeil im Ausgaberaum). Damit wird schließlich der korrekte — dem Merkmalsvektor zugeordnete — Ausgabewert des Netzes bestimmt.

Sowohl die Referenzvektoren als auch die lokal-linearen Abbildungen werden in einem Lernverfahren adaptiert. Dabei wird dem Netz ein Trainingsmuster gezeigt und anschließend der Knoten ausgewählt, dessen Ausgabe dem Merkmalsvektor am nächsten liegt. Anschließend wird der Referenzvektor dieses Knotens in Richtung des Merkmalsvektors verändert. Auch die lokal-lineare Abbildung wird dabei angepaßt. Sie stellt eine Art *Jacobi-Matrix* dar, die differentielle Abweichungen im Eingaberaum durch differentielle Korrekturen im Ausgaberaum ausgleicht. Die Abbildung ist *lokal*, da sie nur im Kontext des jeweils zugeordneten Knotens Anwendung findet. Durch die Kombination der Referenzvektoren mit den lokalen Abbildungen kann das LLM-Netz sehr gut nicht-lineare Funktionen approximieren.

Die jeweilige *Lernschrittweite*, mit der die Referenzvektoren und die Matrix angepaßt werden, wird im Laufe der Lernphase immer kleiner gewählt. Dadurch wird zu Anfang eine grobe Adaptation an die Trainingsbeispiele erzielt, während zum Ende der Lernphase eine Feinanpassung erfolgt. Im Laufe des Lernprozesses verteilen sich die Knoten über den Merkmalsraum. Entwickeln die Knoten dabei einen zu großen Eingabebereich, ist das Netz nur zu einer groben Klassifikation in der Lage. Besser ist eine feinere Verteilung der Knoten. Da das LLM-Netz nur wenige Knoten für eine gute Approximation benötigt, sind das Training und die Ausführung dieses Netztyps wesentlich effizienter als etwa beim MLP-Netz. Gunther Heidemann hat den LLM-Netzwerktyp noch weiterentwickelt; da diese Erweiterungen aber im Rahmen dieser Arbeit keine Anwendung fanden, sei zu Einzelheiten auf [Hei98] verwiesen.

3.2.1 Informationsverarbeitung im LLM

Das LLM-Netzwerk besteht im wesentlichen aus einer internen Schicht, zusammengesetzt aus N Knoten, die im weiteren mit r indiziert werden. Jeder Knoten bezieht denselben Inputvektor $\vec{x}$ der Dimension L. Die Ausgabe eines Knotens ist bestimmt durch den Eingabevektor $\vec{w}_r^{(in)} \in \mathbb{R}^L$, den Ausgabevektor $\vec{w}_r^{(out)} \in \mathbb{R}^M$ und der $M \times L$-Matrix A_r. Dabei ist M die Dimension des Ausgaberaumes. Der zu einem Eingabevektor $\vec{x} \in \mathbb{R}^L$ zugehörige Ausgabevektor $\vec{y_r} \in \mathbb{R}^M$ eines Knotens, ist gegeben durch:

$$\vec{y}_r = \vec{w}_r^{(out)} + A_r \left(\vec{x} - \vec{w}_r^{(in)} \right) \tag{3.8}$$

Dies stellt eine lineare Abbildung mit Berücksichtigung der Jacobi-Matrix A_r im Punkt $x = w_r^{(in)}$ dar. Welche der Abbildungen den Ausgabewert des Netzes bestimmt, wird durch den Abstand $d_r = \|\vec{x} - \vec{w}_r^{(in)}\|$ zwischen dem Merkmalsvektor $\vec{x}$ und dem Eingangsgewichtsvektoren $\vec{w}_r^{(in)}, r = 1, 2, \ldots, N$ festgelegt. Aufgrund der winner-takes-all-Regel bestimmt ihn nur der Knoten s mit dem geringsten Abstand $d_s = \min_r d_r$.

3.2.2 Lernen im LLM

Damit die vom Netz durchgeführte Transformation nun noch sinnvolle Ergebnisse liefert, muß die schon erwähnte Lernphase durchgeführt werden. Dafür werden dem Netz T korrekte Ein-/Ausgabe-Paare $(\vec{x}^{\alpha}, \vec{y}^{\alpha}), \alpha = 1, 2, \ldots, T$ gezeigt. Dann werden die Ein-, und Ausgabegewichte entsprechend einer einfachen Fehlerkorrektur-Regel verändert:

$$\begin{aligned}
\Delta\vec{w}_s^{(in)} &= \epsilon_1 \left(\vec{x}^{(\alpha)} - \vec{w}_s^{(in)} \right) \\
\Delta\vec{w}_s^{(out)} &= \epsilon_2 \left(\vec{y}^{(\alpha)} - \vec{y}^{(net)} \right) + A_s \cdot \Delta\vec{w}_s^{(in)} \\
\Delta A_s &= \epsilon_3 \left(d_s^2 \right)^{-1} \left(\vec{y}^{(\alpha)} - \vec{y}^{(net)} \right) \left(\vec{x}^{(\alpha)} - \vec{w}_s^{(in)} \right)^T
\end{aligned} \tag{3.9}$$

Dabei sind ϵ_1, ϵ_2 und $\epsilon_3 \in [0, 1]$ und bestimmen die *Lernschrittweite*. Der Erfolg des Lernverfahrens hängt zum einen von der Lernschrittweite ab und zum anderen von der Anzahl der Trainingsschritte. Weiterhin bestimmen zusätzliche Faktoren den Erfolg: die Dimension des Netzes, sowie die Anzahl und Art der Trainingsbeispiele. Besitzt das Netz zu wenig Knoten, kann es die Abbildungsfunktion f nicht ausreichend approximieren. Auf der anderen Seite tritt bei zu vielen Knoten der Effekt des „Overfitting“ auf. Dabei lernt das Netz die Trainingsdaten auswendig, und kann anschließend bei unbekannten Mustern nicht generalisieren. Die Knotenzahl sollte i.d.R. also möglichst gering gehalten werden, so daß eine optimale Approximation der Trainings- und Testdaten gewährleistet ist. Eine genaue Untersuchung zur Variation dieser Netzparameter ist im Zusammenhang mit der Klassifikation von Handposturen von Andrea Drees durchgeführt worden [Dre95].

Auch die Dimensionalität des Merkmalsvektors hat einen Einfluß auf die Netzperformanz. Ist der Merkmalsvektor zu niedrigdimensional, können die Daten evtl. nicht differenziert werden. Ist die Dimensionalität zu groß, besteht wiederum die Gefahr des Overfitting. Deshalb wird meist bei der Merkmalsextraktion die Dimensionalität des Merkmalsvektors reduziert, etwa durch eine vorgeschaltete *Karhunen-Loeve-Transformation*. Im folgenden Kapitel 4 werden die LLM-Netze in einem hierarchischen Modell zur Detektion von Gesichtsmerkmalen eingesetzt. Dabei werden mehrere Netze sequentiell hintereinander geschaltet, um auf der Basis von Merkmalsvektoren die Position wichtiger menschlicher Gesichtsmerkmale in digitalen Kamerabildern zu finden.

4 Modellgestützte Aufmerksamkeitssteuerung

DIE persönliche Kommunikation zwischen Menschen basiert neben der Sprache auch auf typischen Körpergesten. Der Gesichtsausdruck und die Blickrichtung eines Gesprächpartners unterstreichen oftmals die sprachlichen Äußerungen und dienen dabei ebenfalls als Informationsträger. Das Gesicht wird während eines Gesprächs sehr oft beobachtet, und die Aufmerksamkeit des Gegenübers wird nicht nur durch sprachliche Anweisungen oder Handgesten sondern auch über die Blickrichtung beeinflußt. In einem Gespräch entsteht so ein gemeinsamer Aufmerksamkeitsfokus, der die Referenzierung von Gegenständen und Orten wesentlich erleichtert.

Die Erkennung von Blickrichtungen ist auch für die Entwicklung natürlicher Computer-Schnittstellen interessant. Die Auswertung von Kopf- und Augenstellungen kann in einem großen Maße dazu beitragen, leichter zu bedienende Mensch-Maschine-Schnittstellen zu realisieren. Existierende „Eyetracking“-Systeme, wie sie etwa für die Untersuchungen in Kapitel 2.3 verwendet werden, sind nicht nur unkomfortabel zu tragen sondern auch sehr teuer, da sie in erster Linie auf Präzision ausgelegt sind [BP94]. In vielen Situationen enthält jedoch bereits die grobe Blickrichtung des Benutzer genügend Information zur Unterstützung des Dialoges. Schon die einfache Blickrichtungsdetektion reicht zur Steuerung grundlegender Grafikfunktionen aus, wie etwa die Translation und Drehung von Objekten oder Vergrößerung der betrachteten Bildregionen. Sie kann beispielsweise genutzt werden, um 3D-Objekte auf einem Computer-Bildschirm abhängig von der Blickrichtung aus mehreren Perspektiven zu betrachten.

Eine weitere interessante Applikation für ein Blickrichtungserkennungssystem ist die Steuerung einer grafischen Bedienoberfläche. Dabei existieren unterschiedliche Ansätze ein solches System mit Rücksicht auf die Fähigkeiten des Benutzers zu realisieren: Die Übertragung von Augenbewegungen zur Steuerung einer allgemeinen Bedienoberfläche [IH94], oder das Design einer speziellen „augenbasierten“ Schnittstelle [Jac95]. Der im folgenden beschriebene Eyetracker erfüllt sicher nicht alle Ansprüche an eine blickbasierte Schnittstelle (siehe etwa

[IH94]) und es bestehen noch viele Probleme, Blickbewegungen als Eingabestrom zur Steuerung zu benutzen (siehe dazu die Diskussion in [Jac95]). Deshalb werden in dieser Arbeit die Leistungen des Blickrichtungserkennungssystems anhand einfacher grafischer Anwendungen demonstriert. In erster Linie geht es um die Entwicklung eines Erkennungsvorgangs zur zuverlässigen Bestimmung von Blickrichtungen aus Kamerabildern. Ergonomische Aspekte der Mensch-Maschine-Schnittstelle werden hier noch nicht voll berücksichtigt.

4.1 Modellbildung

Die Entwicklung eines kamerabasierten Blickrichtungserkennungssystems umfaßt nicht nur die Auswertung der Augen- sondern auch der Kopfrichtung. Es müssen daher verschiedene Teilaufgaben gelöst werden, um die zur Berechnung der Blickrichtung notwendige Information zu extrahieren. Zunächst muß das Gesicht an sich detektiert werden, wobei störende Bildeinflüsse wie Beleuchtung etc. die Erkennung beeinflussen können. Darüber hinaus dürfen die individuellen Eigenschaften unterschiedlicher Personen keine Rolle bei der Ermittlung der Blickposition spielen. In Eyetracking-Experimenten läßt sich nachweisen, daß Versuchspersonen ein menschliches Gesicht auf Anhieb als ein Gesicht erkennen und sofort damit beginnen, typische Gesichtsmerkmale wie Augen oder Mund zu fixieren [Pom98]. Dabei wird die Aufmerksamkeit abwechselnd auf typische auffällige Merkmale gerichtet (Augen etc.). Deshalb wird im folgenden zunächst ein Modell des menschlichen Gesichts entworfen, das die benutzerunabhängige Detektion und Auswertung der Blickrichtung erlaubt.

Als erster Hinweis auf das Gesicht wird das Merkmal *Hautfarbe* ausgewertet. Die Segmentierung hautfarbener Bildpunkte im Kamerabild ermöglicht die Trennung der Gesichtsregion vom Hintergrund. Da allerdings die Hautfarben unterschiedlicher Personen stark voneinander abweichen können, muß ein adaptiver Mechanismus verwendet werden, um das System auf abweichende Hauttypen einzustellen. Nach der Segmentierung des Gesichts steht eine weitere Problematik im Mittelpunkt: Die Detektion der Augenregion.

Bei der Betrachtung eines menschlichen Gesichts fällt auf, daß es zum einen symmetrisch aufgebaut ist und zum anderen durch die Anwesenheit *typischer Gesichtsmerkmale* wie Mund, Nase, Augen und Ohren geprägt ist. Diese Merkmale stehen in fester Geometrie zueinander, bei der zwar die Abstände und Größen variieren, die Anordnung aber ähnlich bleibt. Deshalb ist es möglich, die Suche der Augenregion durch die Detektion von Nase und Mund zu steuern. Sind diese Regionen gefunden, kann die Augenregion extrahiert werden und die eigentliche Auswertung der Blickrichtung erfolgen. Zusätzliche Information wie etwa die Kopfrichtung des Benutzers können dann auch aufgrund der Anordnungen der Gesichtsmerkmale im Kamerabild berechnet werden.

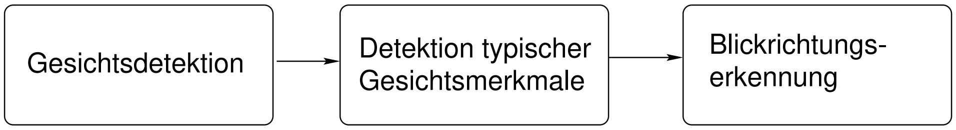


Abbildung 4.1: Die drei Stufen der Blickrichtungserkennung: ausgehend von der Gesichtsregion werden zunächst typische Gesichtsmerkmale bestimmt. Die Augenregion wird anschließend detailliert ausgewertet, um die Blickrichtung zu bestimmen.

Das im folgenden beschriebene System ist in Zusammenarbeit mit Axel Varchmin entstanden [Var97] und verwendet drei Module für den Erkennungsvorgang (Abbildung 4.1). Zunächst identifiziert das erste Modul die Position des Gesichts basierend auf einer *Hautfarbensegmentierung* des Kamerabildes. Innerhalb der Gesichtsregion wird dann nach den auffälligen Gesichtsmerkmalen Nase, Mund und Augen gesucht. Zuständig für diesen Vorgang ist ein zweites Modul, das auf einer *hierarchischen neuronalen Netzwerkarchitektur* als Aufmerksamkeitssteuerung zur passiven Fixation der Gesichtsmerkmale basiert. Passive Fixationen bezeichnen die gezielte Auswertung lokaler Regionen in einem Bild, ohne daß der gesamte Bildinhalt ausgewertet werden muß. Sind markante Punkte, wie etwa die Nase, in einem Bild bereits bekannt, werden sie als Ausgangspunkt genutzt um die Verarbeitung direkt auf die Augenregionen zu fokussieren (die typischerweise oberhalb der Nase liegen). Da diese Gesichtsmerkmale bei verschiedenen Personen sehr ähnlich sind, arbeitet das System unabhängig vom Benutzer schnell und sehr robust. Ausgehend von geschätzten Bildpositionen werden mit Hilfe sequentiell hintereinander geschalteter LLM-Netze die korrekten Gesichtsmerkmalspositionen bestimmt. Anschließend verwendet das dritte Modul die detektierten Gesichtsmerkmale um die Kopfrichtung zu berechnen und die *Blickrichtung* mittels einer detaillierten Analyse der Augenregion zu bestimmen. Auch hier kommt ein neuronales Netz zum Einsatz, um das System durch einen schnellen Online-Trainingsvorgang auf unterschiedliche Benutzer zu adaptieren.

Heutzutage sind die meisten Multimedia-Computer bereits mit einer Videokonferenzkamera und einem Framegrabber ausgerüstet, deshalb verwendet das System nur bildverarbeitende Methoden zur Klassifikation. Das System verfolgt die Kopf- und Blickrichtung mit Hilfe einer mehrstufigen Aufmerksamkeitssteuerung. Andere Arbeiten haben gezeigt, daß diese Aufgabe immer noch eine echte Herausforderung darstellt [KR96], weil sich unter anderem die Lichtbedingungen ändern können, unterschiedliche Benutzer berücksichtigt werden müssen und die Computerleistung bei aufwendigen Bildverarbeitungsmethoden limitiert ist. Andere Autoren, die ebenfalls auf diesem anspruchsvollen Gebiet der Bildverarbeitung forschen, berichten von denselben Problemen [BP94, SW95, GC94, BS95]. Im Vorfeld dieser Arbeit wurde bereits ein System zur Erkennung von Kopfrichtungen realisiert [KR96]. Allerdings ist dieses

System stark benutzerabhängig und nur schwer an neue Beleuchtungssituationen anzupassen. Deshalb wird in dieser Arbeit viel Wert darauf gelegt, diese Einschränkung zu überwinden. Erreicht wird dieses Ziel, indem die Richtungsbestimmung auf der Detektion von Gesichtsmerkmalen beruht und nicht von der holistischen Auswertung der Kopfpostur abhängt.

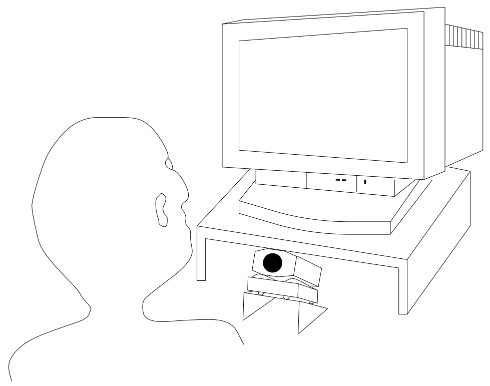


Abbildung 4.2: Der Hardware-Aufbau des Blickrichtungserkennungssystems. Die Kamera steht unterhalb des Monitors, der Benutzer sitzt frontal vor dem Bildschirm.

Der Hardware-Aufbau des Systems besteht aus einer SGI Workstation (Indigo^2 High Impact), einem Framegrabber (INDIGO^2 video) und einer aktiven Kamera (SONY EVI-D31) mit einer eingebauten Trackingfunktion. Es handelt sich dabei um eine dreh- und kippwinkelgesteuerte Videokamera mit integrierter Bildverarbeitungstechnologie. Die Kamera ermöglicht die Verfolgung bewegter Ziele in einem festen Bildausschnitt innerhalb des Blickfeldes. Der Verfolgungsprozeß basiert auf der Auswertung von Farbinformation und arbeitet für Gesichter recht zuverlässig. Die Kamera ist unterhalb des Computermonitors positioniert. Das bietet den Vorteil, daß die Augen sowie die Nasenlöcher des Benutzers deutlich sichtbar sind und nicht allzu leicht verdeckt werden (Abbildung 4.2).

4.2 Gesichtsverfolgung

Der in die Kamera eingebaute Verfolgungsmechanismus erlaubt es, den Kopf des Benutzers innerhalb des Kamerabildes zu halten. Wenn der Kopf sich aus dem aktuellen Blickfeld der Kamera heraus bewegt, wird die Kamera mittels der Motoren sofort auf die neue Position eingestellt. Dieser Vorgang erfolgt automatisch in der Kamera selbst. Allerdings hält der Verfolgungsmechanismus den Kopf des Benutzers nicht konstant im Bildmittelpunkt, was eine weitere präzisere Lokalisierung erfordert. Die exakte Gesichtsregion innerhalb des Bildes wird deshalb mit einem passiven Trackingmodul segmentiert. Innerhalb dieses Moduls wird ein schneller Cluster-Algorithmus verwendet, der die Hautfarben- und Bewegungsinformation zur Berechnung eines Rechtecks heranzieht, das die Gesichtsfläche umschreibt. Im Gegensatz zum später vorgestellten Aufmerksamkeitsmodell GRAVIS (Kapitel 5) fließt diese Information zur Zeit nicht in die Ansteuerung der Kamera zurück; stattdessen schränkt sie nur den auszuwertenden Bildbereich auf die interessante Gesichtsregion ein (vergleiche Kapitel 5.7).

Um die Gesichtsregion zu segmentieren, werden alle hautfarbenen Pixel innerhalb der rechteckigen Bildregion intensiviert. Die dafür benötigte Farb-Klassifizierung akkumuliert die Farbwerte, die zu unterschiedlichen Hautfarbentypen gehören, in einer *allgemeinen Hautfarbenkarte*. Der zugrundeliegende Farbraum ist der rgb-Farbraum, und die typischen Gesichtsfarben werden in der Hautfarbenkarte als Histogrammeinträge gespeichert (zu Details siehe [SW95]). Die Verteilung der Histogrammeinträge basiert auf einer Stichprobe von Bildern unterschiedlicher Benutzer (und somit Hauttypen) und dient als Referenz für die Menge der vom System akzeptierten Hautfarbenwerte. Die in diesen Bildern vorkommenden Hautfarbenwerte werden manuell klassifiziert und in die allgemeine Farbkarte übertragen. Die Histogrammeinträge die zu einer in der Stichprobe vorkommenden Hautfarbe korrespondieren wachsen also an. Die Einträge in dieser allgemeinen Karte werden anschließend auf Werte zwischen 0 und 255 normalisiert.

Während einer Sitzung wird eine Kopie dieser allgemeinen Farbkarte automatisch an die aktuelle Gesichtsfarbe adaptiert. Das bedeutet, daß die Farbwerte des aktuell detektierten Gesichtbereichs ebenfalls in die allgemeine Farbkarte aufgenommen werden. Damit kann sich das System automatisch an die Hautfarbe des Benutzers anpassen (siehe Abbildung 4.3). Dieser Vorgang passiert durch (i) die Abdämpfung aller Histogrammeinträge mit einem festen Faktor und (ii) normalisieren der Farbwerte im Histogramm auf den Wert 255. Das bewirkt, daß die aktuell im Bild vorhandene Hautfarbe besonders hervorgehoben wird. Die Anpassung der Farbkarte findet aber nur statt, wenn die Farbwerte der detektierten Nasenregion bereits in der allgemeinen Farbkarte vorhanden sind. Die Farbkarte spezialisiert sich also auf den aktuellen Hauttyp des Benutzers, wenn die Farbwerte in der Menge der repräsentativen Hauttypen auftauchen, die in der allgemeinen Farbkarte hinterlegt sind.

Die Abbildung 4.3 illustriert eine Beispielsequenz der Bildintensivierung durch die adaptive Farbkarte. Helle Pixel korrespondieren zu hohen Werten in der Farbkarte und repräsentieren hautfarbene Regionen. Der Prozeß der Intensivierung beginnt mit einer unspezifischen allgemeinen Farbkarte — viele Hintergrundpixel werden fälschlicherweise als Hautfarbe klassifiziert. Nach einigen Schritten ist die Farbkarte adaptiert und das Gesicht tritt gegenüber dem Hintergrund klar hervor. Der Adaptionsmechanismus startet erneut, falls ein Problem bei der Detektion einer geeigneten Gesichtsregion auftritt (etwa wenn die Region zu klein ist).

Abbildung 4.3: Adaption der Farbkarte am Anfang einer Bildsequenz. Mit zunehmender Bildanzahl wird die Segmentierung spezifischer und das Gesicht tritt klarer hervor.

4.3 Detektion von typischen Gesichtsmerkmalen

Die menschliche Kopfpostur kann sehr stark unterschiedlich ausfallen. Deshalb werden in dem System Merkmale extrahiert, die über einen großen Bereich variierender Blickwinkel zuverlässig detektiert werden. Geeignete Gesichtsmerkmale sind daher Nase, Mund und Augen [VRR97]. Insgesamt werden fünf spezialisierte neuronale Netze des Typs Local Linear Map (siehe Kapitel 3.2) zur Aufmerksamkeitssteuerung eingesetzt um die korrekten Positionen von Augen und Nase zu bestimmen. Ausgehend von heuristisch geschätzten Merkmalspositionen lenken die Netze die Verarbeitung auf die nächste interessante Bildregion, in der die echte Position des jeweils gesuchten Gesichtsmerkmale vermutet wird. Der Aufmerksamkeitsmechanismus versucht also schrittweise die Position des Gesichtsmerkmals weiter zu verbessern. Die so detektierten Bildregionen werden mit zwei weiteren Netzen verifiziert. Wenn die Bildregion nicht mit dem entsprechenden Gesichtsmerkmal übereinstimmt, wird ein neues Bild aufgenommen und der Erkennungsprozeß beginnt erneut. Die Abbildung 4.4 zeigt die Struktur der gesamten aufmerksamkeitsgesteuerten Gesichtsmerkmalsdetektion. In einem anschließenden Schritt wird die durch diesen hierarchischen Ansatz berechnete Augenregion verwendet, um die Blickrichtung zu bestimmen.

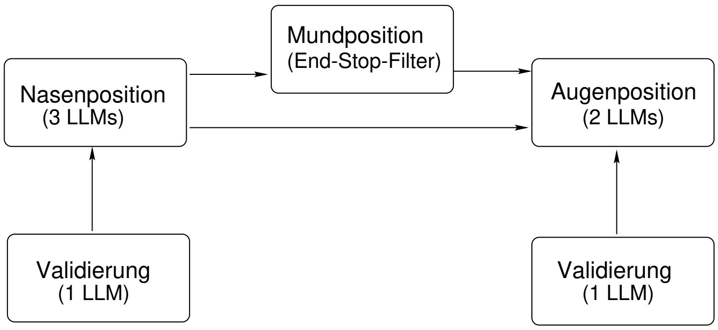


Abbildung 4.4: Der sequentielle Aufbau des Systems zur Detektion typischer Gesichtsmerkmale in den Kamerabildern.

4.3.1 Merkmalsextraktion

Die Extraktion von Merkmalen aus den Eingabebildern sollte einen möglichst niedrigdimensionalen und leicht zu klassifizierenden *Merkmalsvektor* liefern. Dies bezieht sich auf die Robustheit des Klassifikators gegenüber affinen Transformationen, Veränderung der Beleuchtungssituation und Rauschen. In den Vorgängersystemen wurden dafür *lokale Gaborfilter* auf die zu klassifizierende Bildregion angewendet [Dre95, RR98b]. Die Gaborfilter reagieren spezifisch auf generelle Bildmerkmale wie etwa Kontrast und Kantenorientierungen. Das Hauptproblem ist allerdings die Auswahl der Filterparameter, um die Klassifikation optimal zu unterstützen.

In dieser Arbeit wird eine andere Methode für die Merkmalsextraktion entwickelt. Sie basiert auf dem bekannten „Eigengesicht"-Ansatz [TP97, PMST94]. Dabei wird ein Satz von Filtermasken lokal mit der Gesichtsregion gefaltet. Die Filtermasken bestehen aus den aus typischen Gesichtsmerkmalsbildern unterschiedlicher Personen berechneten Eigenvektoren. Die Vorgehensweise zur Gewinnung der Eigenvektoren basiert auf der Karhunen-Loeve-Transformation (KLT). Die KLT bietet die Möglichkeit, diejenigen Merkmale zu bestimmen, deren Fehlen statistisch gesehen den geringsten Informationsverlust bedeutet.

Im vorliegenden Fall können die Bilder aus $l*m$ Pixel als Elemente eines $(l*m)$-dimensionalen Vektorraums verstanden werden. Jedes Pixel eines Bildes spannt eine Achse des Vektorraums auf. Nun sind benachbarte Pixel allerdings hochgradig korreliert, d.h., Bilder von z.B. Nasen sind ähnlich aufgebaut und besitzen keine ungeordneten Schwarz-Weiß-Übergänge. Eine Hauptachsentransformation liefert maximal unkorrelierte Achsen. Der durch die Hauptachsen aufgespannte Raum wird in der Literatur auch *Nasenraum* genannt, falls die Bilder des zugrundeliegenden Datensatzes Nasen enthalten. Im Falle von transformierten Augenbildern heißt der Vektorraum *Augenraum*.

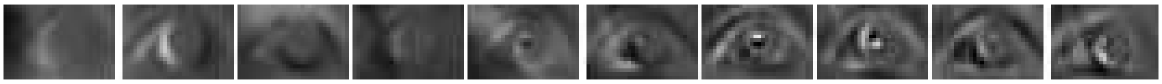


Abbildung 4.5: Die ersten zehn „Eigenaugen“ für die Detektion der Blickrichtung. Die Größe der Filtermasken beträgt 40×20 Pixel.

Da entlang der Richtung der ersten n ($n < l * m$) Basisvektoren, man spricht auch von den ersten n *Eigennasen* (oder *Eigenaugen*), die Varianz der zugrundeliegenden Bilder maximal ist, eignet sich der durch sie aufgespannte n-dimensionale Unterraum sehr gut zur Klassifikation. Bei der resultierenden Datenreduktion geht die statistisch minimale Information verloren. Das Ergebnis nach der Anwendung dieser Eigenvektoren-Filtermasken repräsentiert somit typische Charakteristiken einer Gesichtsregion. Da die Anwendung dieser Masken lokal erfolgt, wird eine robuste und schnelle Merkmalsextraktion erreicht.

Um etwa die Augen zu detektieren, werden aus verschiedenen Bildern mit Augenpartien unterschiedlicher Personen die ersten zehn „Eigenaugen“ berechnet. Die Präzision der Detektion kann durch die Verwendung zweier Sätze von Eigenaugen verbessert werden. Dafür wird in dem einen Satz der Eigenaugen die Position der Augen innerhalb der Bildregion weniger variiert (feiner Satz), in dem anderen stärker (grober Satz).

Die Abbildung 4.5 illustriert Beispiele der für die endgültige Blickrichtungserkennung eingesetzten Eigenaugen-Filtermasken. Ein ähnlicher Extraktionsansatz wird für die Nasendetektion verwendet, bei dem sieben „Eigennasen“ für die lokale Bildfaltung eingesetzt werden. Diese Masken werden aus verschiedenen linear skalierten Gesichtsregionen abgeleitet und auf einem 2×2-Gitter innerhalb der Gesichtsregion angewendet. Dadurch entsteht ein 28-dimensionaler Merkmalsvektor für die Klassifikation. Die Merkmalsextraktion auf der Basis von Eigenvektoren bildet eine geeignete Grundlage für die Klassifikation, da die Filterung bereits spezifische Klasseninformation besonders hervorhebt.

4.3.2 Nasendetektion

Erste Versuche zur Detektion von Gesichtsmerkmalen haben gezeigt, daß die Detektion der Nase relativ einfach und zuverlässig zu implementieren ist. Die Nase ist aufgrund der Kameraposition unterhalb des Bildschirms permanent gut sichtbar. Die Suche nach den Nasenlöchern wird zusätzlich beschränkt auf die mittlere Gesichtsregion. Deshalb wurde ein heuristischer Ansatz gewählt, der in Abbildung 4.6 skizziert ist. Die Nasenlöcher sind im Rot-Kanal der Kamerabilder wesentlich heller als andere Bildregionen des Gesichts. Zusätzlich sind die Helligkeitswerte im Vergleich zu anderen Teilen der Gesichtsregion eher niedrig. Darüber hinaus sind die Nasenlöcher eher im mittleren Gesichtsbereich zu finden, deshalb wird diese Region mit einer Intensitätspyramide höher bewertet. Dadurch treten die Nasenlöcher in diesem Bereich noch besser hervor. Mit diesem Wissen erreicht die Nasendetektion eine Fehlerquote von ungefähr 98% bei einem Satz von 670 Testbildern.

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

Abbildung 4.6: Heuristische Methode zur Nasendetektion: Aus dem Originalbild (a) wird das Rot-Kanalbild verwendet (b). (c): Berechnung des Produkts zwischen dem Rot-Kanal und dem inversen Grauwertkanals. (d): Die Intensitätspyramide wird multipliziert mit dem Cluster-Zentrum um die Nase deutlicher hervorzuheben. (e): Berechnung des Produkts zwischen (d) und der 40×40-Region extrahiert aus (c).

Für die verbleibenden 2% falschen Treffer wird die Nasenposition mit einem *iterativen Korrekturverfahren* detektiert, das auf drei neuronalen Netzen basiert. Dabei handelt es sich um eine Aufmerksamkeitssteuerung, die ausgehend von der geschätzten Nasenposition versucht, durch passive Fixationen sukzessive auf die korrekte Nasenregion zu treffen.

Die Netze vom Typ Local Linear Map aus Kapitel 3.2 sind darauf trainiert, die Position der Nase in unterschiedlich aufgelösten Bildregionen zu ermitteln und die Aufmerksamkeit in Richtung dieser Position zu verschieben. Dafür wird ausgehend vom Mittelpunkt der Gesichtsregion das erste Netz benutzt, um eine grobe Korrektur der Position zu berechnen. Dabei wird der bereits beschriebene 28-dimensionale Merkmalsvektor verwendet. Der neu berechnete Aufsatzpunkt wird vom zweiten Netz genutzt, um die Position weiter in Richtung Nase zu verbessern. Das dritte Netz detektiert dann die endgültige Nasenposition (Abbildung 4.7).

Weitere Tests ergeben bei der Nasendetektion einen mittleren Lokalisierungsfehler von 5.2 Pixel in der x-Richtung und 3.6 Pixel in der y-Richtung für die ganze Netz-Sequenz. Der daraus resultierende normalisierte „Root Mean Square Error“ (NRMSE; zur Berechnung siehe Anhang A) beträgt 0.65 beziehungsweise 0.48. Dieser Test beinhaltet alle vorhergesagten Nasenpositionen, auch diejenigen, die durch das folgende Verifizierschema abgelehnt werden.

Die Überprüfung der vorhergesagten Nasenposition wird durch eine Auswertung der Bildstruktur innerhalb der Nasenregion realisiert. Dafür kommt ein weiteres neuronales Netz zum

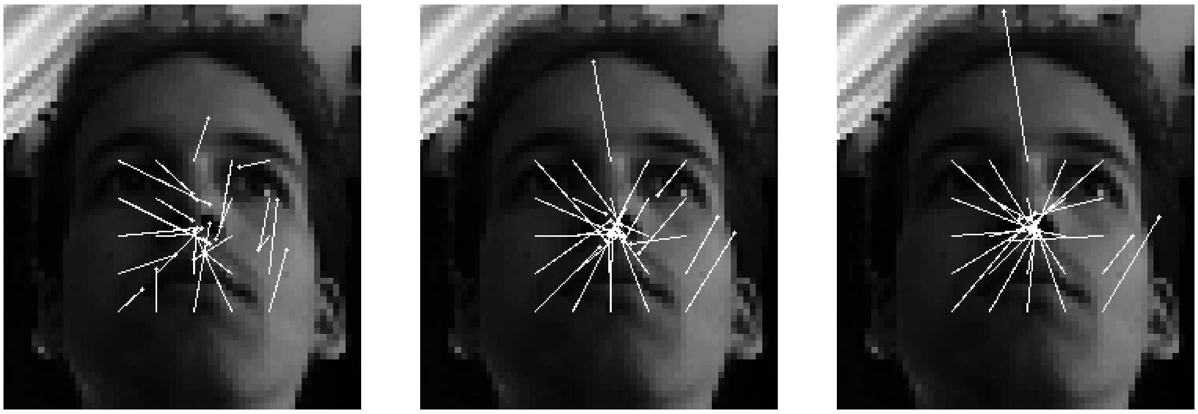


Abbildung 4.7: Korrektur der vorhergesagten Nasenposition, die auf einem Gitter von 5×5 Punkten gestartet wird. Das Aufmerksamkeitsmodell sucht sequentiell die korrekte Nasenposition. Die Startpunkte sind über Linien mit den von dort aus vorhergesagten Nasenpositionen verbunden. Bei der Nutzung des ersten LLM-Netzes ist die Detektion noch ungenau (Links); werden die ersten beiden Netze eingesetzt, liegen die Zielpunkte schon nahe an der echten Nasenposition (Mitte); Die Sequenz aller drei aufeinander folgenden Netze detektiert in fast allen Fällen die korrekte Nasenposition (Rechts).

Einsatz, das speziell auf die Erkennung von Nasen trainiert ist (Abbildung 4.8). Die Eingabemerkmale für dieses Netz sind Skalar-Produkte der vorhergesagten Nasenregion mit 12 „Eigennasen“. Geprüft wird, ob die detektierte Bildregion eine Nase repräsentiert oder nicht. Ein Test dieses Überprüfungsnetzes ergibt eine Klassifikationsrate von 93% korrekt angenommener/abgewiesener Nasenpositionen. Der Testbildsatz enthält 336 Bildregionen, die teilweise Nasen darstellen. Er setzt sich aus Bildern unterschiedlicher Benutzer bei variierenden Beleuchtungssituationen zusammen. Das Netz erkennt also robust und zuverlässig ob das Gesichtsmerkmal „Nase“ korrekt gefunden wurde. Um die Erkennungsrate noch weiter zu steigern, wird darüber hinaus nach der typischen Charakteristik der Nasenlöcher gesucht (Dunkle Nasenlöcher, die umgeben sind von helleren Regionen). Nur wenn beide Module die Region als Nase akzeptieren, wird nach weiteren Gesichtsmerkmalen gesucht. Andernfalls wird das Bild verworfen, und die Bilderkennung startet erneut.

Die Verwendung der Nasenposition als Ausgangspunkt vereinfacht die Erkennung der verbleibenden Gesichtsmerkmale „Mund“ und „Augen“, da die Aufmerksamkeit auf kleinere Unterregionen des aktuellen Bildes eingeschränkt werden kann.

4.3.3 Munddetektion

Die nächsten Merkmale, die Mundwinkel, werden in einer grob aufgelösten Bildregion (16×7) unterhalb der Nase gesucht. Dabei werden einfache „End-Stopping“-Filtermasken mit der

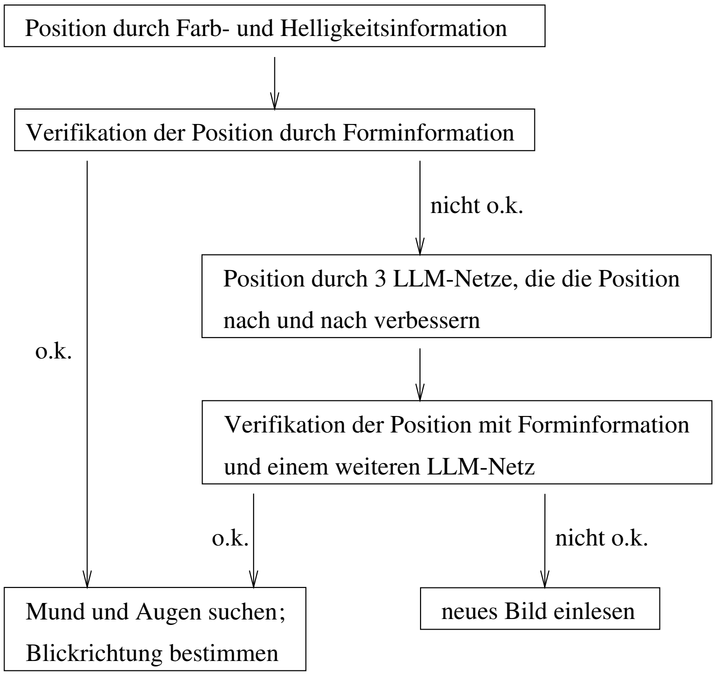


Abbildung 4.8: Ablaufschema der Nasendetektion und die daran anschließende Überprüfung der Bildregion. Erst wenn eine Nase detektiert wird erfolgt die Suche nach Mund und Augen.

Bildregion gefaltet (siehe Abbildung 4.9; vergleiche auch Kapitel 5.2). Tests ergeben, daß die besten Ergebnisse erzielt werden, wenn die Faltung im Grün-Kanal vorgenommen wird. Die zwei höchsten Werte im Ergebnisbild werden dann als die Mundwinkel angenommen.

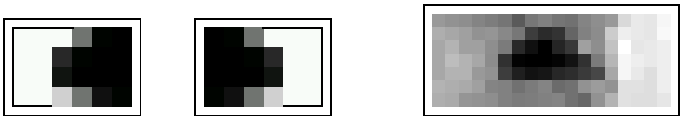


Abbildung 4.9: „End-Stopping"-Filter (Maskengröße 6×4) zur Detektion der Mundwinkel: Rechter und linker Mundwinkel (Links und Mitte). Rechts eine typische Mundregion im Grün-Kanal (Auflösung des Gesichts: 44×36 Pixel).

Die beschriebene Methode arbeitet im Normalfall zuverlässig, jedoch treten Probleme auf, wenn der Mund geöffnet oder verformt ist. Da die Mundposition gemeinsam mit der Nasenposition zur Detektion der Augenpartie herangezogen wird, reicht die Präzision trotzdem aus. Eine Verbesserung der Mundwinkeldetektion kann wenn nötig über einen weiteren Aufmerksamkeitsmechanismus ähnlich zur Nasendetektion erzielt werden.

4.3.4 Augendetektion

Die Positionen der Nase und der Mundwinkel bieten schon eine initiale Schätzung der Augenposition. Deshalb wurde ein weiterer Aufmerksamkeitsansatz gewählt, der ähnlich wie die Nasendetektion funktioniert. Die Bestimmung der Augenposition wird mit zwei neuronalen Netzen (LLMs) durchgeführt, die darauf trainiert sind, in verschieden großen Bildregionen Augen zu detektieren (Abbildung 4.10 (Links)). Das erste Netz ermittelt wiederum eine grobe Schätzung der Position, die durch das zweite — besser spezialisierte — Netz verbessert wird. Die mit dieser Aufmerksamkeitssteuerung erzielten Resultate sind in Abbildung 4.10 (Mitte und Rechts) visualisiert.

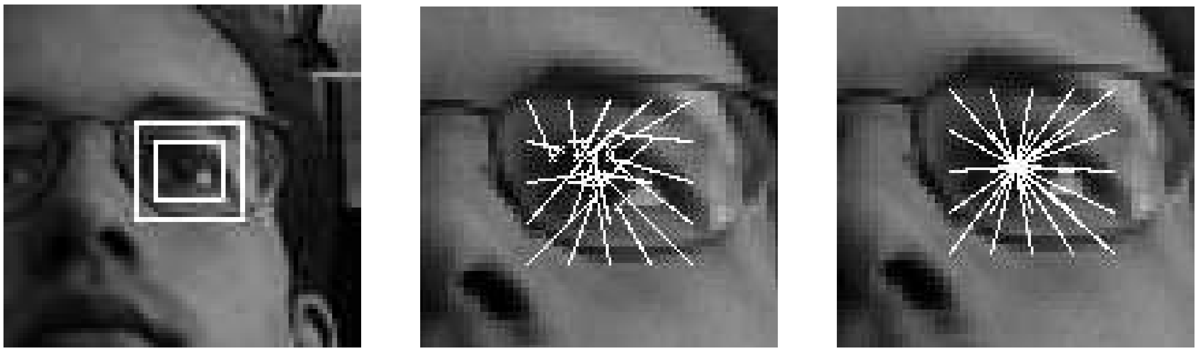


Abbildung 4.10: Links sind die beiden Trainingsregionen für die zwei Augen-Netze dargestellt. Rechts ist die Verbesserung der Augenposition von 5×5 Gitterpunkten durch die Anwendung des Aufmerksamkeitsmodells zur Augendetektion illustriert: Bei Einsatz des erstes LLM-Netzes (Mitte) werden die Augen schon gut getroffen: mit dem darauf folgenden zweiten Netz wird die Detektion noch genauer (Rechts).

Die Hinzunahme des *Entfernungsvektors* zwischen der Nase und der aktuell vorhergesagten Augenposition in den Merkmalsvektor kann den Fehler noch weiter reduzieren: Die Abbildung 4.11 (Links) verdeutlicht den Systemfehler bei der ausschließlichen Verwendung der 2×10 Eigenfilterantworten; Die Abbildung 4.11 (Rechts) hingegen zeigt, daß der Fehler kleiner wird, wenn zusätzlich der Distanzvektor verwendet wird.

Die Tabelle 4.1 vergleicht die Fehler des groben und des feinen LLM-Netzes. Es zeigt sich, daß die Verbesserung 0.2 Pixel in der x- und y-Richtung beträgt, wobei beide Netze auch den Distanzvektor als Merkmal verwenden. Ein weiteres Augen-Verifikationsnetz — ähnlich zur Nasendetektion — kommt anschließend zur Überprüfung der Bildregion zum Einsatz. Der Merkmalsvektor dieses LLM-Netzes besteht aus Skalar-Produkten der vorhergesagten Bildregion (24×12 Pixel) mit 10 „Eigenaugen“-Vektoren. Dieses Netz lehnt fälschlicherweise 22% der korrekten Augen ab und akzeptiert 1% nicht korrekte Augenpositionen. Die hohe Abweisungsrate detektierter Augen liegt an der schwierigen Klassifikation der Augenregion:

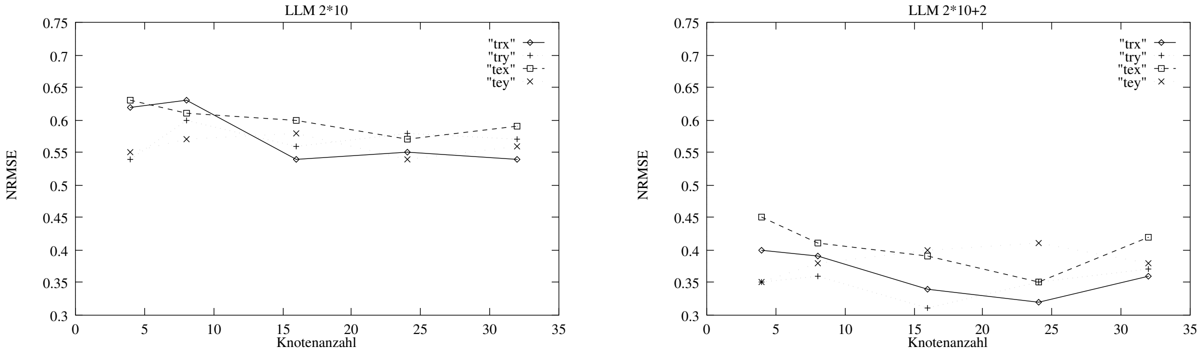


Abbildung 4.11: Fehler in der Detektion der Augenposition bei der Verwendung von 2×10 Eigenaugen-Vektoren (Links); Hinzunahme des Distanzvektors zwischen Nase und aktuell vorhergesagter Augenposition (Rechts). (trx = x-Trainingsfehler, try = y-Trainingsfehler, tex = x-Testfehler, tey = y-Testfehler)

das Bild des Auges variiert je nach Augenstellung. Dafür ist die Abweisung für nicht korrekt detektierte Augenregionen äußerst zuverlässig. Insgesamt bedeutet das eine Erkennungsrate von ca. 88%. Damit wird die Augenposition robust detektiert und kann für die weiteren Verarbeitungsschritte zur Blickrichtungserkennung eingesetzt werden.

LLM-Netz	Durchgang	NRMSE(x)	RMSE(x)	NRMSE(y)	RMSE(y)
Grob	Test	0.39	2.5	0.40	2.3
Grob + Fein	Test	0.36	2.3	0.36	2.1

Tabelle 4.1: Fehlerreduktion durch den zusätzlichen Einsatz des fein-auflösenden Augennetzes. Beide Netze bestehen aus 16 LLM-Knoten.

Zusätzlich wird die Konstellation zwischen Nase, Mundwinkeln und Augen auf ihre Plausibilität geprüft. Es besteht also eine enge Kopplung zwischen dem Gesichtsmodell und den vom Aufmerksamkeitsmodell ermittelten Gesichtsmerkmalspositionen. Die Stellung der Merkmale zueinander kodiert die Kopfstellung. Die Abbildung 4.12 zeigt die Verfolgung von Augen, Nase und Mundwinkeln während einer Folge von Kopfbewegungen vor einem Hintergrund mit hohen Kontrasten. Es ist gut zu sehen, wie die endgültigen Zielpunkte akkurat lokalisiert werden (obere Bildreihe), und das obwohl die Bilder dieses Benutzers nicht im Trainingssatz für die verschiedenen Netze im System enthalten waren. Die untere Bildreihe zeigt einige kritische Aspekte: *(i)* Verdeckungen, wobei die Nase und die Augen trotzdem korrekt detektiert werden; *(ii)* Verdeckungen, wobei der Verifizierungsmechanismus Unstimmigkeiten detektiert (das Bild wird verworfen); *(iii)* rote Objekte stören den heuristischen Ansatz zur Nasendetektion, die Nase wird nicht akzeptiert; *(iv)* wie in Abbildung *iii*, aber diesmal schafft es das Aufmerksamkeitsmodell die Nasenposition durch die Anwendung des sequentiellen Netzschemas zu verbessern, so daß die Nase schließlich akzeptiert wird.

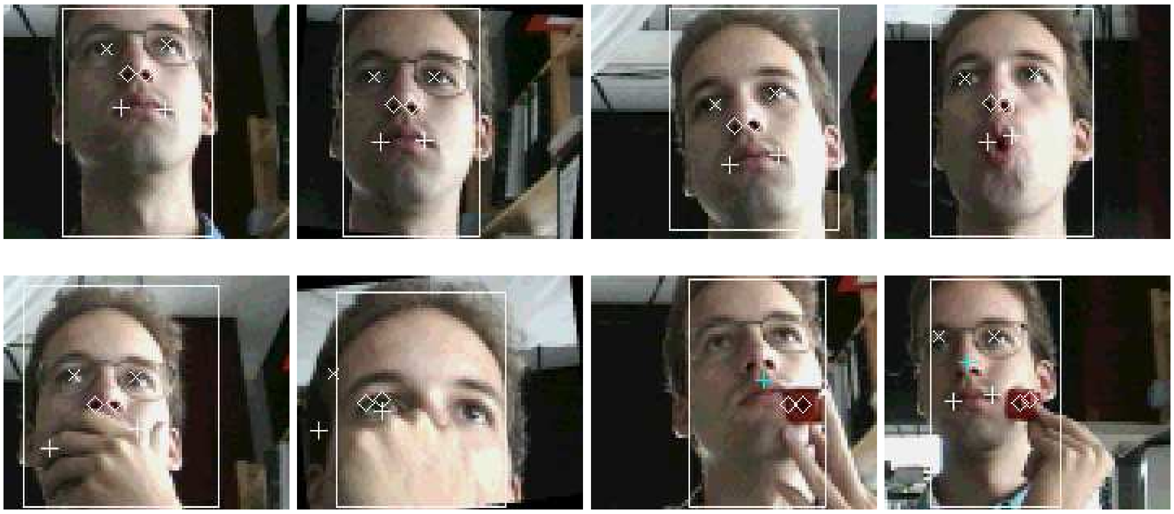


Abbildung 4.12: Obere Reihe: Gesichtsmerkmalsdetektion während einer Bildsequenz (der Benutzer ist nicht im Trainingsdatensatz enthalten). Das Rechteck markiert den Gesichtsausschnitt der durch den Gesichtsverfolgungsansatz bestimmt wurde. Untere Reihe: Das Verhalten des Systems bei Überdeckungen (Links) oder der Anwesenheit von störenden Objekten (Rechts).

4.4 Das System zur Blickrichtungserkennung: GAZE

Zur Bestimmung der Blickrichtung des Benutzers wird schließlich die Kopf und Augenorientierung kombiniert. Durch die modellgestützte Aufmerksamkeitssteuerung wurden bereits die Positionen der Gesichtsmerkmale detektiert und verifiziert. Da somit auch die Augenposition bekannt ist, werden die Augen in einem ersten Verarbeitungsschritt aus dem Bild herausgeschnitten (siehe Abbildung 4.13). Anschließend wird eine Merkmalsextraktion mittels der Anwendung von Eigenaugen-Filtermasken durchgeführt. Zusätzlich wird die Kopfpostur über die beiden Augen-Nasen-Distanzvektoren im Merkmalsvektor repräsentiert. Die gewonnenen Merkmale repräsentieren die Augen und bilden die Eingabe für ein weiteres LLM-Netz, das darauf spezialisiert ist die *Augenrichtung* zu bestimmen. Die Netzausgabe gibt schließlich die berechnete Blickrichtung an. Diese Information kann genutzt werden, um etwa einen Cursor auf dem Computerbildschirm zu steuern.

Vorhergehende Untersuchungen haben gezeigt, daß es relativ schwierig ist die Blickrichtung ohne weitere Restriktionen zu erkennen. Für das hier vorgestellte Blickrichtungserkennungsmodul gelten deshalb noch einige Einschränkungen:

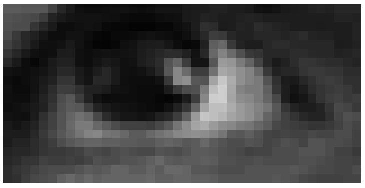


Abbildung 4.13: Aus den Kamerabildern ausgeschnittene typische Augenregionen zur Bestimmung der endgültigen Blickrichtung. Gut zu erkennen ist die Reflexion der Lampe, die zur Verbesserung der Erkennungsleistung verwendet wird.

- Die Erkennung der Blickrichtung ist benutzerabhängig, da die Unterschiede zwischen den Augenregionen (Augenfarbe, Augenstellung etc.) stark ausgeprägt sind.
- Die Erkennung ist beleuchtungsabhängig, weil die Lichteinflüsse als Reflexionen im Auge die Ergebnisse verfälschen.
- Der automatische Kameratrackingmechanismus kann nicht verwendet werden, die Kamera wird nur statisch eingesetzt. Um die aktive Kamera voll ausnutzen zu können, müßte das Trainingsmaterial drastisch erhöht werden[1].
- Um eine genauere Detektion der Blickrichtung zu gewährleisten wird eine Lampe eingesetzt, die eine definierte Reflexion auf das Auge projiziert (Abbildung 4.13).

Andere Autoren — wie etwa [BP94] beschreiben die Maßnahme einer gezielten Beleuchtung ebenfalls als nützlich, um die Ergebnisse zu verbessern. Der Lichtreflex wird in erster Linie dazu genutzt, die Augenmerkmalsregion (40×20 Pixel) exakter auf dem Auge zu plazieren (siehe Abbildung 4.13). Die Merkmalsextraktion mit einem Satz von zehn „Eigenaugen" (siehe Abbildung 4.5) und den zwei Augen-Nase-Distanzvektoren resultiert in einem 14-dimensionalen Merkmalsvektor für ein weiteres LLM-Netz. Dieses Netz (bestehend aus 4 Knoten) ist anhand von 625 Beispielen darauf trainiert die Blickrichtungen eines Benutzers zu klassifizieren. Die Trainingsbilder werden während der Fixation eines 5×5 Gitters auf dem Bildschirm erzeugt. Zum Testen werden leicht unterschiedliche Bildsätze desselben Benutzers während der Fixation des Gitters verwendet (50 Bilder). Der durchschnittliche Systemfehler im Test beträgt 65 Pixel (NRMSE: 0.13) in x-Richtung und 110 Pixel (NRMSE: 0.34) in y-Richtung. Das entspricht einem Winkelfehler von $1.5\,°$ beziehungsweise $2.5\,°$.

Die Tests zeigen weiterhin, daß ein LLM-Netz, das für 3×3 verschiedene Blickrichtungen trainiert wurde, in der Lage ist, auf 5×5 Richtungen zu generaliseren (siehe Abbildung 4.14).

[1] Da beliebig viele Kamerablickwinkel hinzukommen!

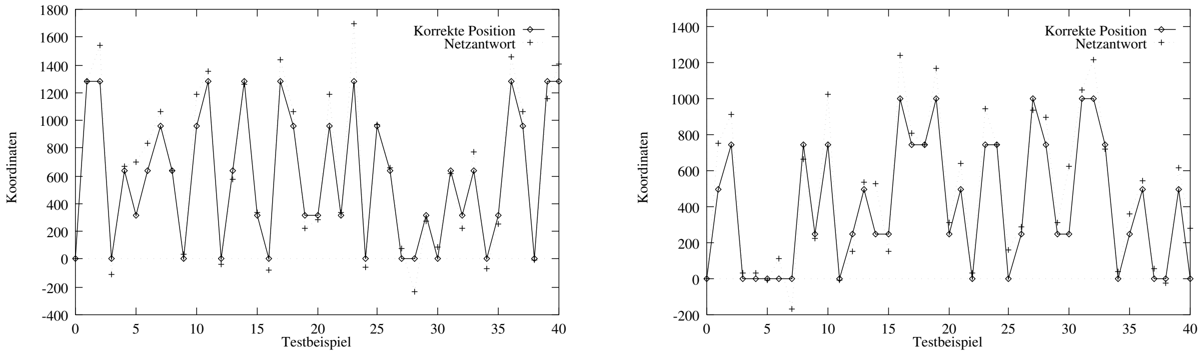


Abbildung 4.14: Der Fehler bei Anwendung eines Blickrichtungsnetzes, das für ein 3×3 Gitter trainiert wurde, auf einem Testsatz von Blickrichtungen, die auf einem 5×5 Gitter aufgenommen wurden (Links: x-Koordinate; Rechts: y-Koordinate).

Deshalb kann das verwendete LLM-Netzwerk leicht auf einen neuen Benutzer oder eine veränderte Beleuchtungssituation angepaßt werden, indem nur die 9 vorgegebenen Positionen auf dem Bildschirm fixiert werden müssen.

In der Abbildung 4.15 wird die Systemausgabe für eine spezielle Anwendung illustriert: Der Benutzer sitzt dabei vor einem Monitor und zeichnet die Konturen eines Hauses mit Hilfe von Blickbewegungen. Die erzielte Erkennungsgenauigkeit ist im Moment noch nicht ausreichend, um kleine Kontrollelemente (wie etwa Menupunkte in einem Fenster) anzusteuern, doch die Detektion des betrachteten Fensters gelingt problemlos. Zusätzlich wurde das System für eine weitere Anwendung getestet, bei der mittels der Blickrichtung ein 3D-Objekt gedreht und verschoben werden kann. Der Benutzer kann somit das Objekt aus verschiedenen Perspektiven betrachten.

Das Blickrichtungserkennungssystem arbeitet mit einer Bildrate von einem Bild pro Sekunde. Das ist zur Zeit noch zu langsam, um das System mit einem guten Eyetracker (etwa dem aus Kapitel 2.3.1) zu vergleichen. Deshalb wurde zu Demonstrationszwecken zusätzlich ein *Echtzeitsystem* implementiert, daß ohne die Gesichtsmerkmalsdetektion auskommt. Dabei muß das Auge des Benutzers jedoch in einer kleinen Bildregion bezüglich der Kamera verharren und Kopfbewegungen sind nicht erlaubt. Daraus resultiert ein sehr schnelles System (25 Bilder pro Sekunde), da nur die kleine Augenregion ausgewertet werden muß. Parallel dazu nimmt natürlich die Präzision der Erkennung ab und es ist für den Benutzer schwierig, das Auge ausreichend ruhig zu halten. Trotzdem demonstriert dieser Test die generellen Vorteile zukünftiger Blickrichtungserkennungssysteme, die mit schnellerer Hardware in der Lage sind bei gleicher Bildrate zusätzlich die Gesichtsmerkmale zu detektieren.

Der im System GAZE verwendete *modellgestützte Aufmerksamkeitsansatz* unterstützt die Blickrichtungserkennung und ermöglicht die Auswertung der Blickrichtung in einer kleinen

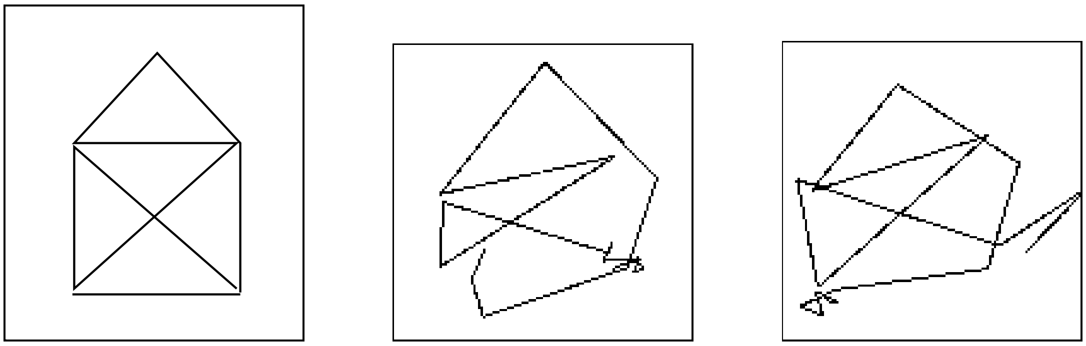


Abbildung 4.15: Zeichnungen eines Hauses, die mit Blickbewegungen angefertigt wurden. Nachdem das Haus fertig gezeichnet ist, verharrt der Blick in der Endposition. Muster (Links) und Kopie gezeichent durch Augenbewegungen (Mitte und Rechts) (Die Bilder bedecken eine Fläche von 1000×1000 Pixel auf dem Monitor).

Augenregion. Das System reduziert durch die Verschiebung des Aufmerksamkeitsfokus die auszuwertende Bildregion drastisch. Die sukzessive Anwendung der LLM-Netze führt dazu, daß alle Gesichtsmerkmale zuverlässig detektiert werden. Das System führt dabei passive Sakkaden (also Sprünge innerhalb des Bildes) aus, die durch ein generelles Gesichtsmodell geführt werden. Die Startpunkte der Fixationen basieren auf heuristisch geschätzten Positionen, die aus dem Modell abgeleitet werden. Dieser Ansatz erlaubt auch bei Teilverdeckungen die übrigen Gesichtsmerkmale zu detektieren. Insgesamt werden im System 6 verschiedene neuronale Netze für die Klassifikation eingesetzt. Drei der LLM-Netze sorgen für eine zuverlässige Detektion der Nasenposition, zwei weitere Netze für die Bestimmung der Augenregion. Darüber hinaus wird auch die Blickrichtung mit einem LLM-Netz berechnet. Obwohl die Ergebnisse der Netze in die jeweils anschließende Systemstufe einfließen, ist das System leicht zu trainieren. Das Training der Netze zur Detektion der Gesichtsmerkmale Nase und Augen kann voneinander getrennt erfolgen, da die Netze nur lokale Bildinformation auswerten. Das gleiche gilt auch für das Netz zur Blickrichtungserkennung. Die Trainingsbilder der Netze können etwa manuell ausgeschnitten und klassifiziert werden, denn das Zusammenspiel der Netze basiert nur auf den zuvor berechneten Aufsatzpunkten.

Allerdings ist eine solche Modellierung nur bei weniger variablen Bildobjekten möglich: Die Grundstruktur von Gesichtern, und insbesondere die Anordnung der Gesichtsmerkmale ist bei fast allen Personen gleich. Auch bei leichten Kopfdrehungen und starker Mimik ändert sich daran nur wenig. Da der hier vorgestellte Ansatz eine ganze Reihe speziell trainierter Netze verwendet, ist diese Vorgehensweise zur Aufmerksamkeitssteuerung nur schwer auf andere Problemstellungen zu übertragen. Zwar können die einzelnen Netze auf neue Merkmale trainiert werden, jedoch ist die Anpassung und Neudimensionierung der Merkmalsextraktion sehr aufwendig. Zusammenfassend läßt sich dennoch feststellen, daß das hier vorgestellte Blickrichtungserkennungssystem GAZE innerhalb seiner Domäne äußerst zuverlässig und robust funktioniert, und einfache Applikationen damit durchaus gesteuert werden können.

5 Datenbasierte Aufmerksamkeitssteuerung

DIE Auswertung von großen Bildmengen ist für technische Systeme aufgrund der limitierten Rechenleistung nach wie vor schwierig. Vor derselben Problematik steht auch die Natur, die beim Menschen die visuelle Aufmerksamkeit als Wahrnehmungsmodalität hervorgebracht hat um die Umwelt effizient wahrzunehmen. Aktive Augenbewegungen sorgen dafür, daß Objekte in das Blickfeld rücken und genauer betrachtet werden können. Besonders bei der Betrachtung natürlicher Szenen wird deutlich, daß unser Auge von bestimmten Bildbereichen besonders angezogen wird. Dies kann etwa eine Reflexion oder eine besondere Farbe sein. Diese Low-Level-Merkmale beeinflussen sehr stark das visuelle Blickverhalten, auch wenn diese Prozesse teilweise durch bewußte Wahrnehmung überlagert werden.

Eine wichtige Grundlage dafür bildet die bereits im Auge beginnende, ortsvariante Verarbeitung des wahrgenommenen Bildes, bei der zentrale Bereiche des Blickfeldes in hoher und periphere Regionen in niedriger Auflösung verarbeitet werden. Die zweite wichtige Komponente bildet ein im *Superior Colliculus* angesiedeltes System zur Blicksteuerung, das mit mehreren topographisch korrespondierenden Merkmalskarten visuelle, aber auch akustische und motorische Information aus verschiedenen kortikalen Arealen integriert und in Blickbewegungskommandos umsetzt. Dabei unterscheidet man zwei Arten von Blickbewegungen: die abrupt erfolgenden Augensprungbewegungen (*Sakkaden*), die im Superior Colliculus ausgelöst werden, und die von einer separaten Struktur geregelten, stetigen Augenfolgebewegungen (*Pursuit*), die der Stabilisierung eines bewegten Objekts in der hochauflösenden Fovearegion dienen.

In Eyetracking-Experimenten ist bei der Beobachtung von Blicksakkaden festzustellen, daß sich interessante Punkte durch besondere Bildmerkmale auszeichnen (Kapitel 2.3.2), oftmals aber neben diesen lokalen Bildmerkmalen auch Blickstrategien einen großen Einfluß auf die Augenbewegung haben [Pom98]. Die gezielte Fixation interessanter Bildbereiche reduziert den Aufwand zur Analyse von komplexen Szenen. Durch eine geschickte Blickstrategie wird etwa das Auffinden von Objekten in einem großen Raum wesentlich beschleunigt. Anstatt jeweils den vollen Bildausschnitt zu evaluieren, wird zunächst eine grobe Vorauswahl in Betracht kommender Bildpunkte getroffen. Wird etwa nur ein rotes Objekt gesucht, reicht es aus, zunächst einmal nach roten Bildbereichen Ausschau zu halten. Diese werden dann nacheinander fovealisiert und in höherer Auflösung analysiert.

5.1 Existierende Ansätze

Die Nachbildung des natürlichen Blickverhaltens ist für Computersehsysteme von erheblichem Interesse und hat in den letzten Jahren zur Etablierung der Forschungsrichtung des „aktiven Sehens“ (*active vision*) geführt [SE93]. Dabei wird Sehen nicht mehr als ein passiver Prozeß der Verarbeitung sensorseitig gelieferter Information aufgefaßt; im Vordergrund steht stattdessen die Erforschung von Mechanismen zur kontextsensitiven Selektion von Information durch *aktive Exploration* der dargebotenen Szene. Dadurch ergibt sich der Vorteil einer erheblichen Reduktion der Informationsbandbreite im ansonsten sehr hochdimensionalen Bildraum [Bal91], was die technische Realisierung — etwa von Objekterkennern — wesentlich vereinfacht. Der Aufwand der Bildverarbeitungsprozesse kann stark reduziert werden, ohne etwas am inhärenten Informationsgehalt des Bildes zu verändern. Viele typische Bildverarbeitungsprobleme, wie zum Beispiel Überwachungsaufgaben, Robotersteuerungen oder Mensch-Maschine-Schnittstellen, lassen sich so besser und robuster realisieren [AS89].

Ein Ansatz zur Implementierung eines solchen Blickverhaltens besteht in der Simulation visueller Aufmerksamkeit. Das künstliche Sehsystem selektiert interessante und relevante Bildbereiche im Eingabebild und verwendet anschließend nur diese Bereiche für eine nähere Betrachtung wie etwa zur Objekterkennung. Eine wichtige Aufgabe für ein solches aktives Sehsystem ist die Lösung des „Where-To-Look-Next“-Problems, also der Auswahl des nächsten Fixationsortes. Im Mittelpunkt steht dabei hauptsächlich die Implementierung einer Strategie für Kamerabewegungen zur effizienten Exploration von statischen und dynamischen Szenen. In der Literatur findet man eine Anzahl unterschiedlicher Systeme zur Simulation visueller Aufmerksamkeit. Einige davon werden hier kurz vorgestellt und diskutiert.

Das Modell von Niebur und Koch [NK96] basiert auf neuen physiologischen Erkenntnissen und Daten vom visuellen System des Affen. Sie verwenden eine sogenannte Aufmerksamkeitskarte zur Steuerung des Aufmerksamkeitsfokus. Die Karte wird berechnet durch die Summation der Ausgabe von verschiedenen Low-Level-Bildmerkmalen, die über Filteroperationen gewonnen werden. So werden unter anderem Farbinformationen aus unterschiedlich aufgelösten Bildern benutzt, um interessante Punkte zu detektieren. Anschließend wird mit einem „Winner-Takes-All“-Netz die visuelle Aufmerksamkeit gesteuert und auffälligste Bildregion selektiert. Die Autoren demonstrieren die Leistungsfähigkeit ihres Systems auf vielen unterschiedlichen Bildtypen.

Das VISIT-Modell von Ahmad [Ahm91] basiert auf physiologischen Experimenten, die sich mit der menschlichen Aufmerksamkeit beschäftigen. VISIT besteht aus einem massiv parallel verknüpften Netzwerk, das zwei visuelle Aufgaben durchführt: (i) Die Berechnung von räumlichen Relationen und (ii) eine auf Bildmerkmalen basierende Objekterkennung. Ein zweites Modell, SWIFT, das ebenfalls einen neuronalen Modellierungsansatz verwendet, minimiert dabei die Anzahl der Fixationen. Das daraus resultierende System ist sehr effizient und flexibel und arbeitet auch auf hochauflösenden Bildern.

Der Ansatz von Fellenz [Fel97a, Fel97b] konzentriert sich auf das Problem der Merkmalsverbindung und verwendet dafür ein mehrschichtiges hierarchisches Prozeßmodell. Die erste Schicht realisiert einen Synchronisationsmechanismus zur Trennung von Figur und Hintergrund. Anschließend folgt ein Prozeß zur Aufmerksamkeitssteuerung, der das segmentierte Objekt in niedriger Bildauflösung im Kamerafokus extrahiert. Diese Daten werden dann in einer nächsten Systemstufe genutzt, um gezielt weiteres Wissen über das Objekt zu sammeln. Die Rückkoppelung zur Kamerasteuerung schließt den Aktion-Wahrnehmungszyklus. Das System von Fellenz ist in der Lage sehr komplexe Bilder zu analysieren.

Kattner [Kat94] hat ein Aufmerksamkeitssystem zur Verbindung von Low- und High-Level-Bildverarbeitung auf der Grundlage der „aktiven Rekonstruktion“ entwickelt. Erste Hinweise auf attraktive Fixationspositionen gehen von einfachen Low-Level-Bildmerkmalen aus, die vom Modell extrahiert und bewertet werden. Die fokussierten Regionen werden klassifiziert und zurückgespielt in ein höheres Modul, das alle gesammelten Merkmale integriert. Das System arbeitet in zwei Richtungen: Aufmerksamkeit wird erzeugt durch einfache Bildmerkmale und durch zusätzliches High-Level-Wissen. Dies resultiert in einem sehr effizienten Fixationsalgorithmus, der bei unterschiedlichen Bildtypen und Erkennungsproblemen sehr gute Ergebnisse liefert.

Der Ansatz von Bollmann, Hoischen und Mertsching [BHM97] integriert dynamische Szeneninformation, wie bewegte Objekte, um ein aktives Kamerasystem zu steuern. Ein präattentives Modul berechnet verschiedene statische Merkmale, wie Orientierungen und Farben, sowie dynamische Bewegungsmerkmale. Prinzipiell arbeitet das System in zwei Modi: dem Hypothesen-Validierungsmodus und dem Verfolgungsmodus. Im Hypothesenmodus steuern die statischen Merkmale in einem Bottom-Up-Ansatz den Erkennungsprozeß. Dabei fließt auch Objektwissen ein. Die dynamischen Merkmale werden hingegen als „Alarmsystem“ verwendet: Sobald das präattentive Modul Bewegungen feststellt, wird die Hypothesenvalidierung unterbrochen und der Verfolgungsmechanismus gestartet.

Die vorgestellten Modelle simulieren jeweils unterschiedliche Aspekte der visuellen Aufmerksamkeit. Ein Vergleich der Systemleistungen ist äußerst schwierig, da zum größten Teil disparate Probleme untersucht werden und ähnlich wie beim Eyetracking-Experiment (siehe Kapitel 2.3.2) *Blickstrategien* verglichen werden müssen. Die Zielstellung reicht von der Auswertung statischer Bilder bis zu Integration dynamischer Information. Bis auf das System von Bollmann et. al. sind die Systeme nicht interaktiv. Ein umfassender Ansatz zur Aufmerksamkeitssteuerung, der neben visueller Information auch andere Modalitäten berücksichtigt, ist bislang nicht realisiert worden.

5.2 Biologische Motivation

Wie in Kapitel 2 beschrieben tragen prinzipiell zwei verschiedene Klassen von Information zur Steuerung von Augenbewegungen bei: (i) Bewegungsinformation, die über die magnozellulären Bahnen ohne weitere Vorverarbeitung in den Colliculus übertragen wird und (ii) die Information aus den parvozellulären Bahnen, die bei der Abbildung auf den primären Kortex vorverarbeitet wird.

Die Verarbeitung sorgt für die Extraktion von Orientierungs-, Farb- und Bewegungsinformation aus den Sensorsignalen. Dabei werden verschiedene neuronale Schichten verwendet, die etwa auf Kanten, Linien und Helligkeitsverläufe spezifisch reagieren. Anschließend wird die Information in die Bewegungskarte im superior Colliculus übertragen. Das Prinzip der mehrschichtigen Verarbeitung von visueller Information ist in das folgende Aufmerksamkeitsmodell übernommen worden.

Die orientierungsspezifische Merkmalsextraktion innerhalb der Schichten erfolgt über Filteroperationen, die die Funktionsweise von Komplexzellen simulieren (siehe Kapitel 2.1.2). Es handelt sich dabei um diskrete Filter, die eine gute Separierung der Kanteninformation gewährleisten. Als Abstraktion der Komplexzellen wird hier die Komet-Filtermaske (siehe Abbildung 5.1) verwendet, die besonders Linienenden hervorhebt („End-Stopping“-Eigenschaft).

Die bewegungsspezifische Verarbeitung der magnozellulären Bahn wird mittels Differenzbildern simuliert [FKC87]. Durch die Berechnung der Differenzen werden Veränderungen im Bild detektiert. Um auch Farbinformation bei der Berechnung der Bewegungskarte zu berücksichtigen, wird die Farbsättigung verwendet. Da die adaptive Anpassung der Photozellen an die aktuellen Beleuchtungsbedingungen nicht linear erfolgt, fließen offensichtlich auch Intensitätsinformationen in den superior Colliculus. Eventuelle Helligkeitsänderungen werden also nicht ausgefiltert. Analog wird im Aufmerksamkeitsmodell eine Schicht zur Repräsentation der Bildintensität verwendet. Diese Modellierung korrespondiert mit der Verknüpfung des V1 mit dem superior Colliculus.

Neben den rein biologischen Erkenntnissen über die prinzipiellen Mechanismen der Augenbewegungen können auch Experimente qualitative Aspekte hervorbringen, die in die Simulation von Aufmerksamkeit einfließen. Wie in Kapitel 2.3.2 bereits beschrieben, begünstigen

Abbildung 5.1: Die diskrete Komet-Filtermaske mit „End-Stopping“-Eigenschaften wird für die Kantenextraktion eingesetzt.

klare Kanten und Ecken eine Fixation. Gesättigte Farben sind ebenso interessant wie starke Lichtintensitäten. Seltener betrachtet werden hingegen glatte Flächen oder der Hintergrund. In Experimenten wurde gezeigt, daß besonders vertikal und horizontal orientierte Kanten vom Menschen bevorzugt wahrgenommen werden. Deshalb werden auch im Modell zwei Arten von Komet-Filtern zur Kantenextraktion verwendet. Um Kanteneckpunkte an beiden Seiten zu detektieren, werden diese jeweils in zwei Richtungen angewandt. Somit werden insgesamt vier Schichten für die Kantenrepräsentation bereitgestellt.

5.3 Modell der Aufmerksamkeitssteuerung

In dieser Arbeit wird ein rein *datengetriebenes mehrschichtiges Aufmerksamkeitsmodell* vorgestellt, das gemeinsam mit Markus Fislage entwickelt wurde [Fis98]. Es benutzt typische Bildmerkmale, um Sakkaden für einen *binokularen Kamerakopf* zu erzeugen. Das künstliche Sehsystem wird damit in die Lage versetzt, interessante Bildregionen auf Basis von Low-Level-Bildmerkmalen zu erkunden. Ziel ist dabei eine möglichst effiziente und vollständige Erkundung der aktuellen Szene. Im Laufe der Exploration werden nach Möglichkeit alle Objekte fixiert. Um dies zu erreichen ist das System mit einer Art Kurzzeitgedächtnis ausgestattet. Da beide Kameras auf das Ziel ausgerichtet werden sollen, um Stereo-Fixationen zu realisieren, wird eine *Disparitätsminimierung* auf den fovealisierten Regionen durchgeführt.

Darüber hinaus reagiert das System auf das Erscheinen von menschlichen Händen und erkennt die *Richtung von Zeigegesten*, die vom Benutzer ausgeführt werden. Ein neuronales Netz dient zur Unterscheidung von Zeigegesten von anderen Handposturen. Ist eine typische Zeigegeste erkannt worden, wird anschließend die Zeigerichtung im Detail ausgewertet. Die akquirierte Information wird im folgenden verwendet, um die Blickrichtung des Kamerakopfes auf das referenzierte Objekt zu lenken.

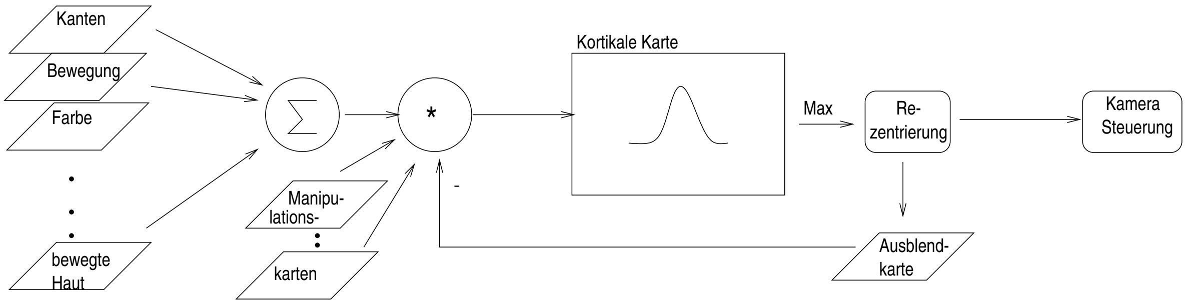


Abbildung 5.2: Der Aufbau des mehrschichtigen Aufmerksamkeitsmodells zur Erzeugung von Kamerasakkaden.

Das Aufmerksamkeitsmodell besteht aus einem Satz von *Merkmalskarten*, die spezielle Bildeigenschaften kodieren [FRR00]. Das sind zum Beispiel im Bild auftretende Kanten oder Intensitätswerte. Diese Karten werden additiv mit einer *Ausblendungskarte* überlagert. Zusätzlich dazu wird eine *Manipulationskarten* eingesetzt, um spezielle Aufmerksamkeitsbereiche zu modellieren. Die Kombination dieser topografischen Karten führt zu einer Aufmerksamkeitskarte (die sogenannte „*kortikale Karte*“, siehe unten), die die räumliche Verteilung der Aufmerksamkeit des Systems simuliert. Die Abbildung 5.2 zeigt den prinzipiellen Aufbau des Systems.

Dabei spielen Gewichtungsfaktoren zwischen den einzelnen Karten eine wichtige Rolle. Diese Gewichtungen werden mittels eines Adaptionsmechanismus so eingestellt, daß alle Merkmale einen gleichmäßigen Einfluß auf die kortikale Karte erhalten. Die Anpassung der Gewichte erfolgt in Schritten und sorgt dafür, daß sich das Modell, etwa bei wechselnden Beleuchtungsbedingungen, an die aktuelle Situation anpaßt. Die Lernschrittweite dieser Adaption bestimmt auch das *Ermüdungsverhalten* des Systems. Anhaltend stark auftretende Merkmale im Bereich der Aufmerksamkeit werden abgeschwächt und somit andere Merkmale aufgewertet, also „interessanter“. Zusätzlich gehen in die Berechnung der kortikalen Karte die Ausblendungskarte und die Manipulationskarten ein, die multiplikativ überlagert werden. Dadurch erhalten sie einen dominanteren Einfluß auf die kortikale Karte als die Merkmalskarten und beeinflussen so die Aufmerksamkeit direkt.

Gegenwärtig werden die folgenden Modellkomponenten im System verwendet:

(i) Merkmalskarten: Insgesamt werden neun Karten eingesetzt, wobei vier Karten unterschiedlich orientierte Kanteninformation und zwei weitere Karten die Sättigung und

Intensität der im Bild auftretenden Farben repräsentieren. In der Bewegungskarte wird durch Differenzbildung dynamische Bildinformation berücksichtigt. In einer speziellen Hautfarbenkarte ist Farbinformation gespeichert, die mittels eines Farbklassifizierers erzeugt wird (siehe auch [Fis98]). Eine weitere Karte wird durch die Multiplikation der Bewegungs- und der Hautfarben-Karte erzeugt. Diese Karte ist besonders entscheidend bei der Detektion von Händen.

(ii) Ausblendungskarte: Damit werden bereits besuchte Bildregionen ausgeblendet und die permanente Beobachtung interessanter Bereiche verhindert. Der Einfluß dieser Karte ist zeitabhängig und repräsentiert ein Sakkadengedächtnis, in dem die vorhergehenden Fixationen gespeichert werden. Die Ausblendung von Bildregionen nimmt mit der Anzahl neuer Fixationen ab, so daß nach einer Weile die zuvor besuchten Bildbereiche zunehmend wieder interessant werden.

(iii) Manipulationskarten: Diese Karten werden verwendet, um die Aufmerksamkeit des Systems direkt zu beeinflussen. Das ist z.B. bei erkannten Zeigegesten nötig, um den Kamerakopf auf das referenzierte Objekt zu lenken. Dafür wird eine Karte erzeugt, die eine kegelförmige Verteilung hoher Aufmerksamkeitswerte in Richtung der zeigenden Hand aufweist (Abschnitt 5.7). Generell ist es möglich mehrere dieser Karten nebeneinander einzusetzen, um weitere Modalitäten zu berücksichtigen. Für die sinnvolle Verrechnung der Karten müssen sich die Bereiche besonderer Aufmerksamkeit jedoch geeignet ergänzen (für die Integration von Sprache und Gestik siehe [MMRR99]).

(iv) Kleine Sakkadensprünge: Augenbewegungsuntersuchungen zeigen, daß Menschen kleine sakkadische Augenbewegungen bevorzugen. Dieses Verhalten wird in der resultierenden kortikalen Karte berücksichtigt. Die Karte repräsentiert eine Gaußsche-Verteilung, die im Mittelpunkt der Karte positioniert ist. Dadurch wird die Zielregion für die nächste Sakkade auf die nähere Umgebung der aktuellen Fixation eingeschränkt.

5.3.1 Berechnung der kortikalen Karte

Die Berechnung der *kortikalen Karte* $C(x, y)$ ist in Abbildung 5.2 skizziert. Die gewichtete additive Überlagerung der n unterschiedlichen Merkmalskarten $f_i(x, y)$ bestimmt zunächst die Ergebniswerte in der kortikalen Karte. Diese wird zusätzlich mit den Manipulationskarten $M_j(x, y)$ moduliert. Die Ausblendungskarte $F(x, y)$ sorgt dafür, daß bereits besuchte Bereiche ausgeblendet werden. Um möglichst kleine Sakkadensprünge zu bevorzugen, wird die berechnete kortikale Karte mit einer Gauß-Verteilung $\Gamma(x, y, \rho) = e^{\frac{-|(x,y)^T - \langle (x,y)^T \rangle|^2}{\rho^2}}$ multipliziert. Die Ausdehnung der Gauß-Verteilung ρ ist beschränkt auf die Hälfte der Größe des Eingabebildes. Damit ist die kortikale Karte $C(x, y)$ wie folgt definiert:

$$C(x,y) \quad = \quad \sum_{i=1}^{n} \theta(w_i \cdot f_i(x,y)) \cdot \frac{\eta - F(x,y)}{\eta} \cdot \Gamma(x,y) \cdot \prod_{m=1}^{l} \left(\frac{M_m(x,y)}{\eta} \right) \qquad (5.1)$$

Dabei ist η das obere Limit für die Einträge in der Karte. Zur Elimination negativer Werte in den Merkmalskarten werden negative Werte auf Null zurückgesetzt. Das Maximum der resultierenden kortikalen Karte spezifiziert einen groben Fixationspunkt, der anschließend noch weiter verfeinert wird (siehe unten). Um auf Änderungen in der zu beobachtenden Szene zu reagieren (Lichteinflüsse etc.) wird zusätzlich eine heuristische Adaptionsmethode der Gewichte w_i eingeführt, die die einzelnen Merkmalskarten unterschiedlich stark gewichtet.

5.3.2 Adaption der Gewichte

Die Gewichte w_i werden langsam geändert und ordnen jedem Merkmal einen zeitlich variierenden Einfluß zu. Grundlage der Adaption ist, daß alle Merkmale, die zur kortikalen Karte beitragen, einen ausgeglichenen Anteil an deren Zusammensetzung haben sollen. Um mit dieser Methode keine Merkmale zu bestrafen, die zwar einen starken Einfluß auf die kortikalen Karte ausüben, jedoch das Interesse noch nicht auf sich lenken, ist es unsinnig diesen Anteil über eine homogene Betrachtung der ganzen Merkmalskarten zu definieren. Um die Stärke eines Merkmals an der Karte auszudrücken, wird daher die Summe über die Werte der Merkmalskarte mit der Gewichtung einer Gaußverteilung um den Mittelpunkt der aktuellen Fixation definiert:

$$S_i = \frac{\sum_{(x,y)} (f_i(x,y) + \gamma) \cdot \Gamma(x,y,\rho)}{\xi_i} \qquad (5.2)$$

Der Parameter γ setzt einen Grenzwert für die Gewichte. Damit wird vermieden, daß die Gewichte bei Abwesenheit eines Merkmals unbeschränkt anwachsen. Die Konstante γ ist also ein Mindestwert für S_i und definiert den maximalen Gewichtswert für w_i. Der Faktor ξ_i bestimmt den Anteil einer Merkmalskarte an der kortikalen Karte und berücksichtigt auch die jeweilige Charakteristik der Merkmalskarten. Das Merkmal Sättigung ist beispielsweise wesentlich homogener verteilt als etwa die extrahierte Kanteninformation. Durch ein geeignetes ξ_i kann somit der Einfluß der Sättigungskarte erhöht werden. Die Werte für die einzelnen ξ_i sind heuristisch bestimmt und werden für die jeweilige Aufgabe gewählt (siehe Abschnitt 5.4).

Aus der Forderung nach ausgeglichenen Anteilen der einzelnen Merkmalskarten folgt:

$$w_i \cdot S_i = \frac{1}{n} \cdot \sum_{k=1}^{n} S_k \qquad (5.3)$$

Daraus resultiert die Lösung für die Gewichte w_i:

$$w_i^{s^*} = \frac{1}{n} \cdot \frac{\sum_{k=1}^{n} S_k}{S_i} \tag{5.4}$$

Damit der Wertebereich der kortikalen Karte auf den der Merkmalskarten normiert wird, ergibt sich die endgültige Berechnung der Gewichte zu:

$$w_i^{s} = \frac{1}{n^2} \cdot \frac{\sum_{k=1}^{n} S_k}{S_i} \tag{5.5}$$

Zwar könnten die Gewicht direkt auf diese Werte angepaßt werden, es zeigt sich jedoch, daß eine schrittweise Veränderung der w_i in Richtung des Soll-Gewichts w_i^s ein besseres Explorationsverhalten zur Folge hat. Dabei wird vermieden, daß ein stark auftretendes Merkmal sofort unterdrückt wird. Durch die Adaption wird ein Ermüdungsverhalten implementiert, so daß das System erst bei einem längeren Vorhandensein eines Merkmals das Interesse verliert. Ein weiterer Vorteil ist, daß sich das System an veränderte Arbeitsbedingungen anpaßt. Damit ergibt sich die endgültige Adaptionsvorschrift mit einer Lernschrittweite ε zu:

$$w_i(t+1) = w_i(t) + \varepsilon \cdot (w_i^s(t) - w_i(t)) \tag{5.6}$$

Mit der Gleichung 5.2 wird das zugrundeliegende Maß für ein auftretendes Merkmal definiert. Der theoretisch maximale Wert für ein Gewicht w_j wird sich einstellen, wenn alle Merkmale $S_i, i \neq j$ maximal und das Merkmal S_j minimal werden. Daher ergibt sich das theoretisch maximale Gewicht w^{max} zu (aus Glg. 5.2, 5.4):

$$\begin{aligned}
S_i^{max} &= \frac{1}{\xi_i} \sum_{(x,y)} ((\eta + \gamma) \cdot \Gamma(x,y,\rho)) \\
S_i^{min} &= \frac{1}{\xi_i} \sum_{(x,y)} (\gamma \cdot \Gamma(x,y,\rho)) \\
w_i^{max} &= \frac{1}{n^2} \cdot \frac{\sum_{(j \neq i)} (S_j^{max}) + S_i^{min}}{S_i^{min}}
\end{aligned} \tag{5.7}$$

Das minimale Gewicht ergibt sich analog:

$$w_i^{min} = \frac{1}{n^2} \cdot \frac{\sum_{(j \neq i)} (S_j^{min}) + S_i^{max}}{S_i^{max}} \tag{5.8}$$

Dadurch sorgt der Adaptionsmechanismus für einen festen Rahmen $w_i(0 < \varepsilon < 1)$ der Gewichte und somit für ein stabiles Adaptionsverhalten des Modells. Auf der Grundlage dieser Adaptionsregeln ergeben sich zwei Vorteile:

- Das System kann sich an variable Faktoren wie Lichtverhältnisse oder starke auftretende Merkmale (wie etwa Texturen) anpassen. Die Adaption sorgt dabei für einen ausgeglichenen Anteil aller Merkmale an der visuellen Aufmerksamkeitssteuerung.

- Das Ermüdungsverhalten des Systems wird durch die Lernschrittweite ε bestimmt. Dadurch werden anhaltend stark auftretende Merkmale in ihrem Anteil an der Aufmerksamkeit unterdrückt und andere Merkmale werden zunehmend interessant.

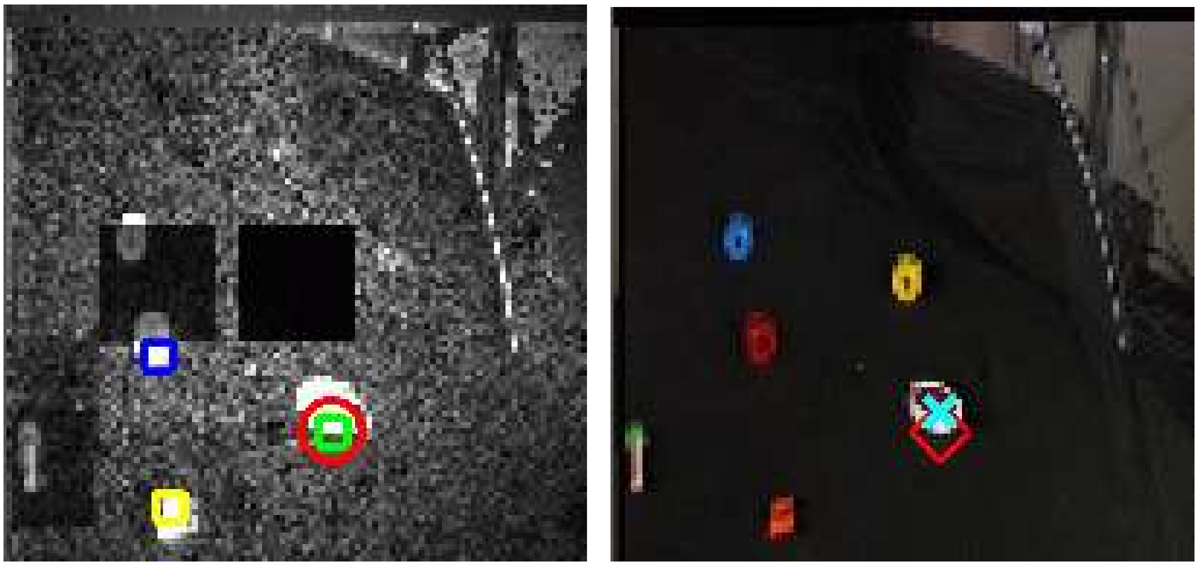


Abbildung 5.3: Visualisierung der kortikalen Karte zur Aufmerksamkeitssteuerung. Helle Punkte korrespondieren mit auffälligen Bildregionen. Gekennzeichnet sind die drei auffälligsten Bereiche. Im rechten Bild ist die dazugehörige Originalszene mit der endgültigen Fixation dargestellt. Ebenfalls zu sehen ist die Korrektur der Fixation (Raute: erster Aufsatzpunkt; Kreuz: korrigierter Fixationspunkt).

Die Werte in der kortikalen Karte stellen einen Interessantheitsgrad für jeden Bildpunkt dar, und die Lage des Maximums in der Karte liefert einen groben Fixationspunkt für die nächste Kamerasakkade (Abbildung 5.3). Dieser Fixationspunkt wird jedoch noch durch weitere Mechanismen nachzentriert (siehe unten), so daß er näher an eventuell auftretenden Objekten liegt. Um auch die andere Kamera auf denselben Bildbereich zu steuern, wird ein einfacher Korrelationsmechanismus eingesetzt (siehe Abschnitt 5.6). Das Modell erzeugt zuverlässige binokulare Kamerasakkaden, die eine visuelle Aufmerksamkeitssteuerung des künstlichen Sehsystems modellieren.

5.4 Bildmerkmale

Die Werte in der Aufmerksamkeitskarte berechnen sich durch die Überlagerung der verschiedenen Merkmalskarten. Die Merkmalscharakteristiken repräsentieren dabei unterschiedliche Aspekte des Bildes, die einen Einfluß auf die Aufmerksamkeit besitzen. Wie aber können diese Bildeigenschaften typischerweise aussehen? Die Art der Merkmale im aktuellen System orientiert sich zum einen an den Ergebnissen des Eyetracking-Experiments (Kapitel 2.3.1) und zum anderen an den biologischen Erkenntnissen über die Vorverarbeitung der optischen Information im menschlichen Sehsystem (Kapitel 2.1.2). Neben statischen Merkmalen wie etwa Farbe spielen auch *dynamische Merkmale* eine Rolle. Insbesondere für die in das Aufmerksamkeitsmodell integrierte Gestenerkennung ist die Detektion von Bewegungen berücksichtigt.

Die genaue Wahl der verwendeten Bildmerkmale ist in Tabelle 5.1 spezifiziert. Der wichtigste Merkmalstyp basiert auf dem Grauwertgradienten im Eingabebild. Die Kanteninformation wird mit der *Komet-Filtermaske* aus Abschnitt 5.2 in verschiedenen Orientierungen aus den Bildern extrahiert. Da dieser Filter End-Stopping-Eigenschaften besitzt, werden insbesondere Ecken und Linienenden aus den Bilddaten gefiltert. Ein Beispiel für ein auf vertikale Dunkel/Hell-Übergänge gefiltertes Bild liefert die Abbildung 5.4(b). Insgesamt werden vier Arten von Grauwertübergängen berechnet.

Beschreibung	**Notation**	ξ_i
Horizontaler Helligkeitsübergang (hell-dunkel)	$f_1(x, y)$	1
Vertikaler Helligkeitsübergang (hell-dunkel)	$f_2(x, y)$	1
Horizontaler Helligkeitsübergang (dunkel-hell)	$f_3(x, y)$	1
Vertikaler Helligkeitsübergang (dunkel-hell)	$f_4(x, y)$	1
Differenzbild	$f_5(x, y)$	2
Sättigung	$f_6(x, y)$	3.5
Intensität	$f_7(x, y)$	0.3
Hautfarbe	$f_8(x, y)$	1.5
Bewegte Hautfarbe	$f_9(x, y)$	4
Ausblendungskarte (Fade-Out)	$F(x, y)$	
Zeigegestenmanipulationskarte	$M_1(x, y)$	

Tabelle 5.1: Die 11 Karten, die einen Einfluß auf die kortikale Karte besitzen. Ebenfalls sind die heuristisch ermittelten Gewichtswerte angegeben. Zur Ausblendungskarte und Manipulationskarte ist im Text näheres zu finden.

Einen weiteren wichtigen Merkmalsraum bildet die Intensität, also die Helligkeitsinformation aus dem Bild. Bei der Betrachtung der Informationsverarbeitung im retinalen System des Menschen zeigt sich, daß Helligkeitsinformation einen Einfluß auf die Aufmerksamkeitssteuerung im superioren Colliculus hat. Da alle Kamerabilder zunächst in den HSI-Raum umgerechnet werden, steht die Komponente Intensität als getrennter Kanal zur Verfügung.

(a) (b) (c)

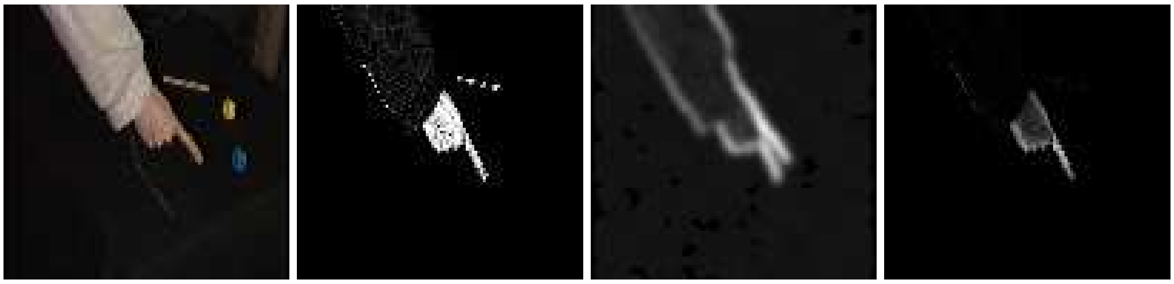
Abbildung 5.4: Beispiel zweier Merkmalskarten aus dem Aufmerksamkeitsmodell mit dem dazugehörigen Original-Kamerabild (a): Visualisierung der Karte für „vertikale Kanten" (dunkel-hell) (b) und der Farbsättigungskarte (c).

Während des Eyetracking-Experiments hat sich herausgestellt, daß auch *Farben* Augenbewegungen beeinflussen. Deshalb wird die Farbsättigung ebenfalls als Merkmal gewählt. Eine berechnete Beispielkarte für die Sättigung ist in Abbildung 5.4(c) zu sehen. Eine blasse Farbe weist eine niedrigeren Sättigungswert auf als kräftige Farben. Besonders bunte Objekte ziehen daher auch die Aufmerksamkeit stärker an.

Die zweite Klasse von Objekten, die in dem Modell berücksichtigt werden soll, sind menschliche Hände. Das primäre Ziel des vorgestellten Aufmerksamkeitsmodells ist die *gestikbasierte Interaktion*. Deshalb repräsentiert eine weitere Merkmalskarte speziell Hautfarbe. Wie schon im System GAZE (Kapitel 4) beschrieben ist Hautfarbe ein guter Hinweis für Gestik. Der Ansatz zur Hautfarbensegmentierung orientiert sich an dem Mechanismus im System GAZE aus Abschnitt 4.2 und basiert ebenfalls auf der Berechnung der zugehörigen Farbwerte im HS-Raum (Hue und Saturation). Da der Kamerakopf im Gegensatz zu GAZE mit Kameras ohne automatischen Weißabgleich und Blendenregelung ausgestattet ist, wurde der Ansatz um eine permanente Adaption der Mittelwerte erweitert. Damit wird sowohl der Sättigungswert wie auch der Farbwinkel, die beide zusammen einen sehr kompakten Raum für Hautfarbe aufspannen, permanent den Beleuchtungsbedingungen angepaßt. Zu näheren Einzelheiten der

Hautfarbensegmentierung sei an dieser Stelle auf [Fis98] verwiesen. Die Repräsentation in der Merkmalskarte zeigt hohe Werte an Stellen, an denen Hautfarbe auftritt (siehe Abbildung 5.5(b)).

Im Zusammenhang mit der Gestenerkennung ist ein anderes Merkmal ebenfalls für das Modell wichtig: *dynamische Bewegungsinformation*. Die Berechnung basiert auf der Differenzbildung aus zwei zeitlich versetzt aufgenommenen Bildern. Da insbesondere hautfarbene Objekte interessieren, wird dafür der Rot-Kanal der RGB-Farbbilder gewählt. Neben der eigentlichen Differenzberechnung wird das Ergebnisbild mittels einer Gaußfilterung zusätzlich geglättet, um Kamerarauschen und lokale Helligkeitsschwankungen zu unterdrücken. Ein Beispiel für eine bewegungsbasierte Merkmalskarte ist in Abbildung 5.5(c) illustriert.



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

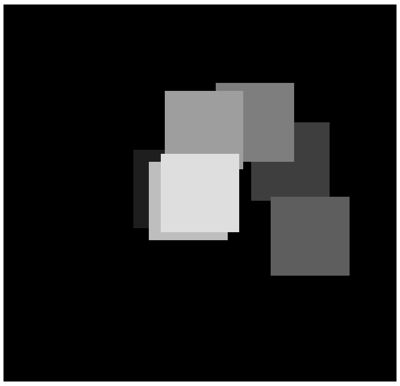
Abbildung 5.5: (a) Original-Handbild zur Berechnung der dynamischen Merkmale mit der Hautfarbensegmentierung (b) und dem Differenzbild (c) während einer Zeigegeste. Die Kombination beider Karten liefert die endgültige Gestenposition (d).

Bei der Berechnung von Differenzbildern tritt ein typisches Problem auf: Es werden Unterschiede an der Stelle detektiert, von der sich das Objekt wegbewegt hat und dort wo es im nächsten Zeitschritt auftaucht. Um etwa eine Hand zu detektieren, ist dieser Ansatz also nur bedingt geeignet. Deshalb wird eine weitere Karte eingeführt, die die Bewegungsinformation und die Hautfarbenkarte kombiniert: Diese Karte repräsentiert bewegte Hautfarbe, indem die Ausgabe aus beiden Karten multipliziert wird. Die Stelle, an der sich die Hand aktuell befindet, tritt dadurch besonders gut hervor (Abbildung 5.5(d)). Mit diesem Ansatz wird die Treffsicherheit einer Fixation beim Auftreten einer Geste erhöht. Mit dieser direkten Steuerung der Aufmerksamkeit wird die später beschriebene Integration der Zeigegestenerkennung in das Modell erreicht.

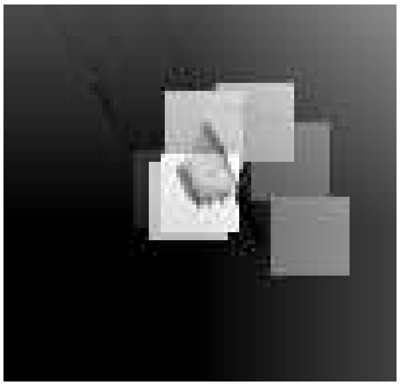
In Tabelle 5.1 sind zusätzlich die heuristisch ermittelten Gewichtungen ξ_i für das jeweilige Merkmal angegeben. Die Kanteninformationen sind jeweils einfach gewichtet. Besonders hohe Gewichte weisen die Sättigungskarte und die Bewegungskarten auf. Damit wird Vorwissen über die zu erfüllende Aufgabe in das Modell integriert. Da im typischen Arbeitsszenario in erster Linie kräftig farbige Objekte und menschliche Hände auftreten, werden die zugehörigen Merkmale höher bewertet.

5.5 Inhibition bereits explorierter Bildbereiche

Eine besondere Herausforderung für die Aufmerksamkeitssteuerung des aktiven Sehsystems ist die *Gedächtnisfunktion*. Das System soll bereits analysierte Bildregionen zunächst meiden und stattdessen neue Bereiche besuchen. Außerdem soll sich die Aufmerksamkeit des Systems auf den vorgegebenen Arbeitsbereich beschränken. Deshalb ist in das Aufmerksamkeitsmodell die Ausblendungskarte (Fade-Out-Karte) integriert. Mit dieser Karte werden bestimmte Bereiche in der Aufmerksamkeitskarte inhibiert. Der Mechanismus basiert dabei auf der Akkumulation rechteckiger Bereiche, die bereits besuchte Bildregionen markieren (Abbildung 5.6(a)). Die Orte der Ausblendungsrechtecke werden aus den Kamerabewegungen gewonnen. Orte, die vor längerer Zeit besucht wurden, werden aufgrund einer abflachenden Inhibition langsam wieder interessant.



(a)



(b)

Abbildung 5.6: (a) Typischer Zustand der Ausblendungskarte: Die hellen Bildbereiche bezeichnen zuvor besuchte Bildregionen. Vor längerer Zeit beobachtete Bereiche werden langsam zurückgesetzt. (b) Zusätzlich fließen Informationen über den Arbeitsbereich und Handgesten in die Berechnung ein.

Damit der Kamerakopf nicht außerhalb des Arbeitsbereichs agiert, fließt in die Berechnung der Ausblendungskarte zusätzlich Information zur Kamerageometrie ein. Dabei werden die Werte der Ausblendungskarte am äußeren Bereich erhöht, wenn der Kopf an seine Gelenkwinkelgrenzen stößt. Er wird dadurch von den kritischen Randbereichen weggeführt. Diese Bereiche sind für das Aufmerksamkeitsmodell also prinzipiell uninteressant (Abbildung 5.6(b) oberer Bereich). Zusätzlich wird bei dem Auftreten einer Geste die Hautfarbenkarte von der Ausblendungskarte abgezogen. Das sorgt dafür, daß dynamische Bildbereich nicht durch die Inhibition von einer neuen Fixation ausgeschlossen werden (Abbildung 5.6(b)). Das Modell kann so in jeder Situation auf eine Handgeste reagieren.

Die Fade-Out-Karte geht in die Berechnung der kortikalen Karte inhibitorisch, also mit negativen Vorzeichen ein. Sie dient als Rückkopplung und sorgt so für ein *dynamisches Blickverhalten*. Das Aufmerksamkeitsmodell exploriert also permanent die Szene und betrachtet neue Bereiche. Nach einer Weile werden die bereits betrachteten Regionen wieder interessant und somit erneut besucht.

5.6 Nachzentrierung zur Stereo-Fixation

Die von der Aufmerksamkeitskarte erzeugten Sakkaden fixieren typischerweise die Ränder von Objekten. Dies liegt in erster Linie am starken Anteil der Kantenstrukturinformation in den Merkmalskarten. Damit die Sakkaden näher an den Objektmittelpunkt rücken, wird ein zusätzlicher Mechanismus benötigt. Im Aufmerksamkeitsmodell ist daher eine *Nachzentrierung* implementiert, die innerhalb der fokussierten Bildregion (Region of Interest, RoI) den Schwerpunkt der darin berechneten Kanten auswertet. Die so berechnete Bildposition bestimmt dann die Anpassung des Kameravergenzwinkels. Bei kleineren Objekten, die noch vollständig in die RoI passen (30×20 Pixel), arbeitet der Algorithmus zuverlässig. Der so erzeugte Aufsatzpunkt kann etwa eine anschließende Objekterkennung wesentlich vereinfachen. Bei größeren Objekten, bei denen die Kanten nicht komplett in die RoI passen, wird der Fixationspunkt zumindest in Richtung des Objekts verschoben. Ein Problem ist darüber hinaus, daß bei der Anwesenheit von zwei Objekten der Fixationspunkt eventuell in die falsche Richtung verschoben wird (zu Details dieser Problematik siehe [Fis98]). Auch für die Fixation von Gesten wird dieser Nachzentrierungsalgorithmus verwendet. In diesem Fall wird allerdings statt der Kanteninformation die Hautfarbensegmentierung als Zentrierungsmerkmal verwendet.

Das Aufmerksamkeitsmodell verwendet in der aktuellen Implementation ausschließlich monokulare Bildinformation um die nächste Sakkade zu berechnen. Die linke Kamera übernimmt die Funktion des „Führungsauges“. Damit auch die räumliche Lage der Objekte ausgewertet werden kann ist es aber notwendig, Objekte mit beiden Kameras zu fixieren. Ein weiterer implementierter Mechanismus sorgt deshalb dafür, daß die rechte Kamera ebenfalls auf den aktuellen Fixationspunkt ausgerichtet wird. Die *Stereo-Fixation* wird erreicht, wenn beide Kameras auf dieselbe räumliche Position ausgerichtet sind und denselben Bildausschnitt im Bildmittelpunkt aufnehmen.

Der Ansatz nutzt eine Eigenschaft des binokularen Kamerakopfs: Da beide Kameras auf einer gemeinsamen Stereobasis montiert sind (zu Einzelheiten siehe Kapitel 7.1.1), ist nur noch ein Anpassen der Vergenzwinkel nötig. Das wiederum bedeutet für die Disparitätsminimierung zwischen den beiden Kamerabilder, daß die fokussierte Bildregion der linken (Führungs-)Kamera im Bild der rechten Kamera wiedergefunden werden muß. Beide Kameras sind auf der gleichen Dreh- und Kippachse befestigt, deshalb kann die Suche auf einen schmalen horizontalen Streifen im rechten Kamerabild beschränkt werden.

Zunächst wird um den fokussierten Fixationspunkt der linken Kamera eine RoI ausgeschnitten. Diese Bildregion wird mit allen Bildpositionen im rechten Kamerabild innerhalb des horizontalen Suchstreifens korreliert. Die Distanz zwischen den Bildregionen ist dabei definiert als die Summe der Differenzen zwischen den drei Farbvektoren der Region im RGB-Farbraum. Ziel ist es, das Minimum der Regionendistanz innerhalb des horizontalen Suchstreifens zu finden. Die so ermittelte Bildposition wird dann genutzt, um die Vergenz der rechten Kamera zu berechnen und somit beide Kameras auf denselben Bildausschnitt zu richten.

Nicht immer kann davon ausgegangen werden, daß die Bildregion auch in der rechten Kamera gefunden wird (Überdeckungen, Reflexionen etc.), deshalb wird zusätzlich überprüft, ob das Minimum eindeutig ist. Erst dann wird die Korrekturbewegung ausgeführt. Sollte in den Bildern keine Übereinstimmung gefunden werden, so wird die rechte Kamera auf dieselbe Bildposition wie die linke Kamera gesteuert. So wird verhindert, daß die Kameras zu stark auseinander driften. Zur genauen Berechnung der Disparitätskorrektur sei an dieser Stelle auf [Fis98] verwiesen.

Sind beide Kameras exakt auf ein Objekt ausgerichtet, so kann aufgrund der Kamerakopf-Kinematik die dreidimensionale Position bestimmt werden (Kapitel 7.4.1). Das Aufmerksamkeitsmodell ist damit in der Lage, Objekte nicht nur zweidimensional auszuwerten, sondern auch die räumliche Objektposition festzustellen, um damit etwa einen Roboter anzusteuern, der das Objekt greifen soll (siehe Kapitel 7.4). Die dreidimensionale Ortsinformation ist aber auch hilfreich bei der Auswertung der ebenfalls in das Aufmerksamkeitsmodell integrierten Handgestenerkennung, die im folgenden Kapitel näher beschrieben wird.

5.7 Integration von Modellwissen

Die Aufmerksamkeitsverteilung im künstlichen Sehsystem wird auf der Grundlage der bereits beschriebenen Merkmalskarten berechnet. Wie in den vorherigen Abschnitten erläutert, werden neben den allgemeinen Bildmerkmalen wie Kanten- oder Farbinformation auch auf spezielle Gestikinformation zugeschnittene Merkmalskarten verwendet. Durch die Repräsentation von Hautfarbe und Bewegung werden von dem Modell bevorzugt menschliche Hände im Blickfeld fixiert. Dadurch ist eine wichtige Voraussetzung für die Gestenerkennung bereits integriert: Die Position und der relevante Bildausschnitt mit der Hand werden automatisch extrahiert.

Deshalb ist es möglich, mittels eines vergleichsweise einfach strukturierten Klassifikators Zeigegesten zu erkennen. Im Gegensatz zu anderen Arbeiten zur Handgestenerkennung [NR98] ist das hier verwendete Modul ausschließlich auf die Auswertung von Zeigegesten spezialisiert und weist andersartige Handgesten lediglich einer Rückweisungsklasse zu. Die Erkennung arbeitet auf vorverarbeiteten farbklassifizierten Bildern, um die für Zeigegesten charakteristische Handhaltung zu erkennen (siehe Abbildung 5.7). Im aktuellen System wird die Auswertung monokular vorgenommen, so daß bislang nur der *Handkippwinkel* genau bestimmt werden kann.

Im Gegensatz zum System GAZE ist die Integration dieses High-Level-Wissens nicht modellgestützt. Die Bestimmung der Handregion geschieht in diesem Fall rein datenbasiert. Die Kontrolle der Blickbewegung über so allgemeine Merkmale wie Bewegung und Hautfarbe ist für eine Handgestenerkennung wesentlich besser geeignet. Das liegt zum einen an der Variabilität der Handpostur und zum anderen an der Großflächigkeit der Geste. Die Hand besitzt also nicht so eindeutige Merkmale wie ein Gesicht (Augen etc.). Punktuell berechnete Merkmale könnten sich allenfalls auf Fingerspitzen beziehen, die aber auch gewissen Verdeckungen unterliegen. Die Vorgehensweise zur Integration von Modellwissen über Zeigegesten ist deshalb wie folgt:

- Eine Hand, die in das Bild greift, wird vom Aufmerksamkeitsmodell detektiert.
- Der fokussierte Bildausschnitt wird extrahiert und die Hautfarbe wird heraussegmentiert.
- Der Bildausschnitt wird analysiert, um die vorliegende Geste zu klassifizieren.
- Falls eine Zeigegeste vorliegt wird die Aufmerksamkeit des Modells in Richtung des referenzierten Objekts gelenkt.

Erscheint also eine Hand im Blickfeld, so wird aufgrund der Merkmale Hautfarbe und Bewegung eine Fixation ausgelöst. Anschließend wird die fixierte Bildregion segmentiert, um hautfarbene Bereiche zu isolieren. Dabei kommt derselbe Hautfarbenklassifikator wie bei der Merkmalskartenberechnung zum Einsatz (Abschnitt 5.4). Zunächst wird auf einer niedrigen Auflösung segmentiert, um zu erkennen ob eine Zeigegeste vorliegt.

Im Bildausschnitt werden daher zwei Histogramme berechnet, die sich durch die Projektion der binarisierten hautfarbenen Pixel entlang der zwei Hauptrichtungen der Hand ergeben (Pfeile in Abbildung 5.7). Als Aufsatzpunkt zur Berechnung der Hauptrichtungen wird zunächst

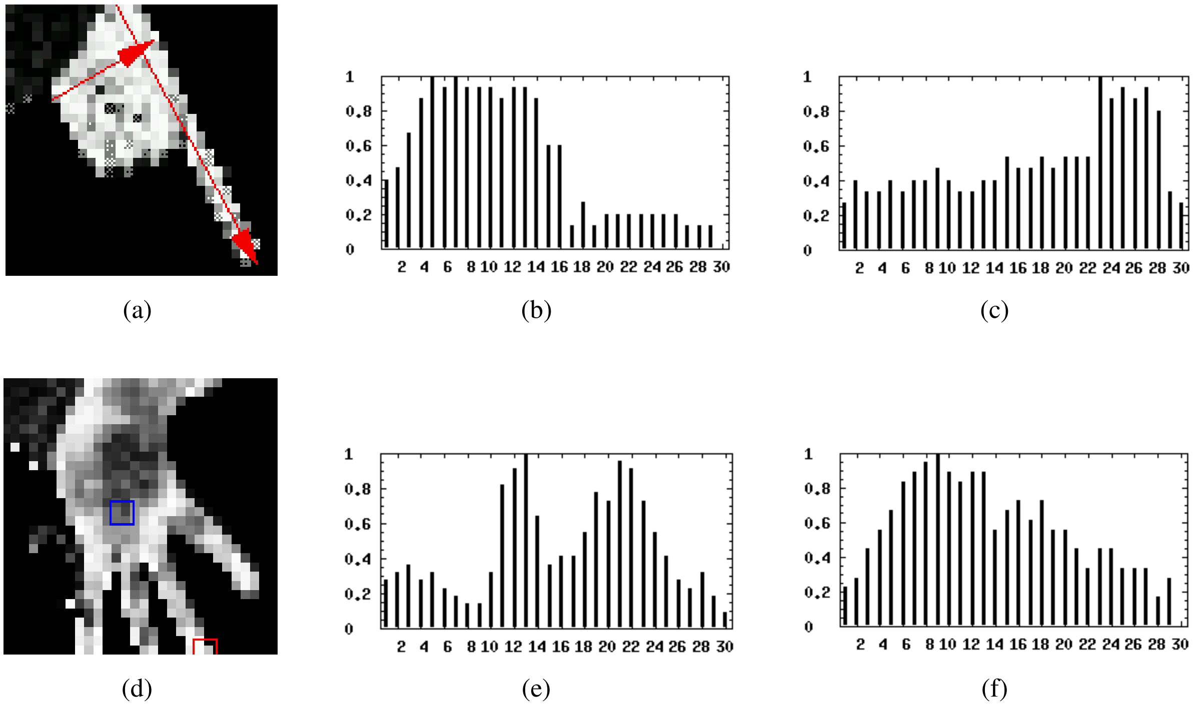
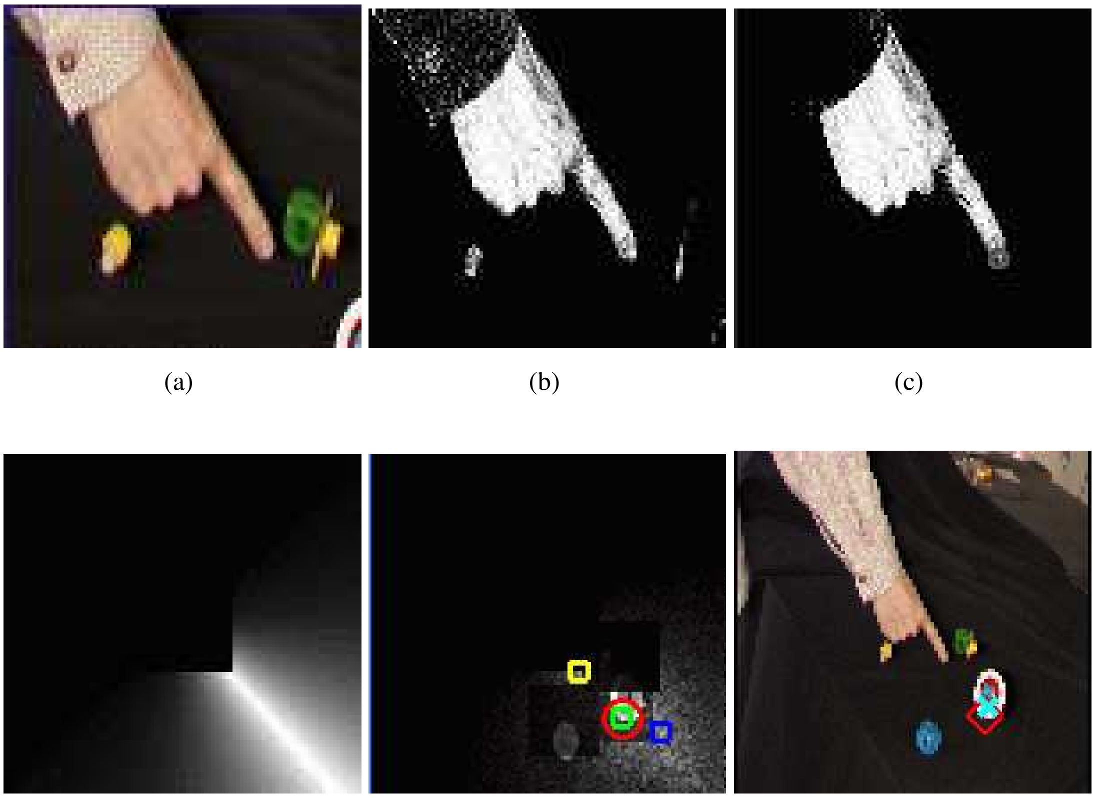


Abbildung 5.7: Die entwickelte Datenrepräsentation für die Zeigegestenerkennung. (a) Die segmentierte Hand mit den berechneten Hauptrichtungen sowie die auf die erste (b) und zweite (c) Hauptrichtung projizierten Histogramme. Dieselbe Merkmalsdarstellung wird für eine andere Handpostur gezeigt und verdeutlicht den starken Unterschied der Histogramme (d-f).

der Schwerpunkt des vorsegmentierten Bildausschnittes angenommen. Zusätzlich wird auf dem separierten Handcluster der Punkt gesucht, der am weitesten vom Clusterschwerpunkt entfernt liegt.

Dieses ist bei einer zeigenden Hand mit hoher Wahrscheinlichkeit die Fingerspitze. Bezüglich dieser neuen Position wird der gegenüberliegende Punkt auf der Hand gesucht, der den Abstand maximiert. Diese zwei Punkte definieren einen Vektor, auf den die Pixelverteilung der Hand projiziert wird. Ebenso wie auf diesen Vektor wird die Pixelverteilung auf den dazu orthogonalen varianzmaximierenden Vektor projiziert. Die Projektion auf die beiden Vektoren wird auf die vordefinierte Histogrammlänge normiert. Diese besondere Art der Merkmalsextraktion berücksichtigt die Posturunterschiede unterschiedlicher Gesten. So variiert die Histogrammausgabe je nach Finger- und Handstellung (siehe Abbildung 5.7 unten). Die normalisierten Merkmalsvektoren stellen eine *handgrößenunabhängige Formrepräsentation* dar und führen bei unterschiedlichen Zeigegesten zu ähnlichen Ergebnissen.

Die Werte der zwei Histogramme (jeweils 30) definieren den Merkmalsvektor für ein 60-4-1 MLP-Netzwerk (siehe Kapitel 3). Die Dimensionierung des Netzwerkes wurde heuristisch ermittelt, indem verschiedene Architekturen und Dimensionierungen getestet wurden. Bei der gewählten Netzstruktur ergibt sich die beste Generalisierungs- und Klassifikationsgüte. Die Antwort des Ausgabeneurons ist eine binarisierte Klassifikation in Zeigegeste oder andere Geste. Das künstliche neuronale Netz ist mittels des *RProp*-Verfahrens (siehe Kapitel 3.1.2) darauf trainiert, Zeigegesten von Nicht-Zeigegesten zu unterscheiden. Dafür wurden 400 Bilder mit je 50% Zeigegesten bzw. Nicht-Zeigegesten als Trainingsmenge verwendet. Die Trainingsmenge umfaßte dabei Bilder unterschiedlicher Benutzer unter variierenden Beleuchtungsbedingungen.



(a) (b) (c)

(d) (e) (f)

Abbildung 5.8: Die aufeinanderfolgenden Stufen der Zeigegestenerkennung: Die fixierte Hand in hoher Auflösung (a) und die anschließende Hautfarben-Segmentierung (b); Nach der Elimination von Rauschen (c) wird die zugehörige Manipulationskarte berechnet (d). Die Manipulationskarte wird mit der kortikaler Karte (e) multipliziert, um die Aufmerksamkeit auf das Rad zu lenken (f).

Neben dem neuronalen Klassifikator werden noch einige weitere Parameter überprüft, um sicherzustellen, daß tatsächlich eine Zeigegeste vorliegt. Unter anderem wird getestet, ob überhaupt genügend hautfarbene Pixel vorliegen (zu Einzelheiten siehe auch [Fis98]).

Im Falle einer erkannten Zeigegeste wird die Zeigerichtung auf einer höheren Bildauflösung fortgesetzt. Zunächst wird die bereits fokussierte Bildregion in einer höheren Auflösung erneut eingelesen. Anschließend wird die Hautfarbensegmentierung verwendet um die Hand zu extrahieren. Daraufhin wird der Schwerpunkt des segmentierten Objekts berechnet. Der von diesem Schwerpunkt am weitesten entfernte Punkt wird als die Spitze des Zeigefingers oder eines anderen abgestreckten Fingers interpretiert. Deshalb wird die Verbindungslinie von der Fingerspitze zum Handmittelpunkt ermittelt und als Zeigerichtung angenommen. Daraus errechnet sich der Handkippwinkel, der schließlich in eine kegelförmig aktivierte *Manipulationskarte* übersetzt wird (Abbildung 5.8). Der Kegel dient dabei als zweidimensionale Repräsentation der Geste. Die Werte des Kegels spannen einen Richtungswinkel auf, der in der Mitte am höchsten bewertet ist. An den Rändern ist er hingegen schwächer ausgeprägt.

Diese Manipulationskarte M_1 beeinflußt die kortikale Karte derart, daß die Bildregion in Richtung der Zeigegeste bevorzugt fixiert wird. Objekte, die exakt in Richtung des Kegels zu finden sind, werden dabei am stärksten bewertet und somit am interessantesten (Abbildung 5.8(e)). Liegt das Objekt nicht genau im Winkel, wird es dennoch durch die Streuung im Aufmerksamkeitskegel erfaßt. Sind mehrere Objekte im Aufmerksamkeitskegel vorhanden, entscheidet sich das Aufmerksamkeitsmodell für das zur Geste am nächsten positionierte Objekt, da es kurze Sakkaden bevorzugt (Abschnitt 5.3). Als Folge wird die nächste Fixation von der Zeigegeste gelenkt, und der referenzierte Gegenstand durch das aktive Sehsystem beobachtet (Abbildung 5.8(f)).

Eine Zeigegeste besteht jedoch typischerweise aus mehreren Phasen: Die Hand wird zur Geste geformt, sie bewegt sich zum Zeigeort und erst dann erlangt die Zeigegeste die beabsichtigte Bedeutung. Insbesondere die dynamische Phase, in der sich die Hand schnell bewegt, kann vom Sakkaden-Modell nicht observiert werden. Die Verarbeitungsgeschwindigkeit ist zu gering, und die Ausblendung bereits untersuchter Bildregionen verhindert eine Folge von Sakkaden, die permanent auf dem sich bewegenden Ziel landet. Daher ist ein anderer Ansatz zur Beobachtung und Analyse dieser Bewegungsphase nötig. Im folgenden Kapitel wird ein Ansatz zur schnellen und flüssigen Verfolgung bewegter Objekte vorgestellt, der anschließend mit dem Aufmerksamkeitsmodell zu einem kompletten interaktiven Gestenerkenner integriert wird.

6 Kamerabewegung zur schnellen Objektverfolgung

DIE Verfolgung schnell bewegter Objekte erfolgt beim Menschen durch *weiche Augenfolgebewegungen*. Beide Augen richten sich dabei auf das Ziel und fixieren es anschließend permanent. Die Nachbildung dieses Verfolgungsverhaltens ist auch im Rahmen der Handgestenerkennung des zuvor vorgestellten Aufmerksamkeitsmodells interessant: Nach der Detektion einer Hand ist die schnelle Verfolgung eine Voraussetzung zur Erkennung dynamischer Zeigegesten. Dafür reicht aber die Verarbeitungsgeschwindigkeit des Sakkaden-Modells aufgrund der aufwendigen Merkmalsberechnungen und Klassifikationsvorgänge nicht aus. Deshalb wird in diesem Kapitel die Konzeption und Implementierung eines *echtzeit-basierten* aktiven Sehsystems zur Verfolgung von Objekten beschrieben, das in das entwickelte Aufmerksamkeitsmodell integriert wird.

Das hier beschriebene System ist in der Lage, Objekte mit unterschiedlicher Form und Größe im Raum zu verfolgen [RR98a]. Es wird wiederum der Stereo-Kamerakopf (siehe Kapitel 5.3 und 7.1) verwendet. Die Bildverarbeitung erfolgt auf einem Spezialrechner, der eine Echtzeitbildverarbeitung erlaubt. Dadurch kann das Kamerasystem ein bewegtes Objekt mittels einer weichen Kameranachführung, auch vor einem komplexen Hintergrund, kontinuierlich verfolgen. Die Kontrolle der Vergenzbewegung basiert dabei auf einer *Nulldisparitätsfilterung* (*zero-disparity filtering* ZDF). Der Dreh- und Kippwinkel des Kamerakopfes wird durch die Berechnung des Objektschwerpunktes im Bild bestimmt. Das Verfahren basiert im wesentlichen auf der Extraktion von senkrechten Kanten im Bild. Es findet keine Objekterkennung statt, daher funktioniert das System bei beliebigen Objekten die allerdings Kantenstrukturen aufweisen müssen. Neben künstlichen Objekten wie etwa einem Roboter kann das aktive Sehsystem auch menschliche Hände im Rahmen eines Konstruktionsszenarios zuverlässig verfolgen. Dabei dienen die Kanten der Finger und des Handballens als Grundlage für die Nulldisparitätsfilterung.

6.1 Biologische und technische Motivation

Im Sehsystem des Menschen findet man primär zwei Arten von Augenbewegungen: (i) die Sakkaden, und (ii) die weichen Augenbewegungen (siehe Kapitel 2). Diese Mechanismen werden im menschlichen Gehirn offensichtlich von getrennten Bereichen gesteuert, und das komplexe Zusammenspiel von reflexartigen und geplanten Augenbewegungen sorgt für eine robuste Wahrnehmung. Es liegt die Hypothese nahe, daß es sich bei der jeweiligen Augenbewegungsart um eine notwendige Anpassung an spezielle visuelle Aufgaben handelt. Mittels Augenbewegungsuntersuchungen und aufgrund der vorliegenden physiologischen Erkenntnisse konnten Teilbausteine des Systems zur weichen Objektverfolgung beim Menschen identifiziert werden. Die Grundlage bildet ein Regelungssystem, das aus verschiedenen sensorischen Informationen wie etwa die Abweichung der Objektposition vom Mittelpunkt der Fovea die Nachführbewegungen der Augen bestimmt. Eine Rückkopplung sorgt dafür, daß das Objekt stabil im fovealen Bereich des Auges gehalten wird [WW88].

Typische Bildverarbeitungssysteme wie etwa Überwachungssysteme, Roboter-Steuerungen und Mensch-Maschine-Schnittstellen benötigen ebenfalls Steuerungsmechanismen zur Beobachtung schnell bewegter Objekte. Sie arbeiten oftmals in dynamischen Umgebungen, in denen sich Objekte bewegen, sich die Beleuchtungsbedingungen ändern oder neue Dinge im Blickfeld erscheinen. Die Verfolgung von Objekten mit einer statischen Kamera ist allerdings schwierig. Die kontinuierliche und schritthaltende Neuberechnung der Objektposition erfordert eine hohe Rechenleistung, da permanent das gesamte Bild nach dem Zielobjekt durchsucht werden muß. Darüber hinaus erscheint das Ziel im Kamerabild wegen der Bewegung „verschmiert“, während der Hintergrund statisch bleibt und daher hohe Kontraste aufweist. Speziell bei diesem Problem bietet ein aktives Kamerasystem Vorteile, da die Kameras auf dem Zielobjekt verharren und somit das Objekt relativ konstant im Zentrum verbleibt. Dadurch hebt sich das Objekt klar vom Hintergrund ab. Es konnte gezeigt werden, daß ein *aktiver Beobachter* weitere Vorteile bei solch schwierigen Bildverarbeitungsaufgaben aufweist [SE93]. Das Paradigma des *aktiven Sehens* wird deshalb genutzt, um technische Systeme zu entwickeln, die mittels kontrollierbarer Kameras die hohe Informationsbandbreite im hochdimensionalen Bildraum reduzieren können [Bal91].

Eine große Anzahl von Ansätzen zum Design binokularer Verfolgungssysteme arbeitet auf der Grundlage von Disparitätsmessung (siehe etwa [AME93, Kro89]). Die meisten dieser Systeme sind allerdings auf einzelne Bildmerkmale spezialisiert. So gibt es Systeme, die lokale Bildbewegung auswerten [DHKS95], Kantenpunkte als Merkmal verwenden [RM96] oder Farbmerkmale benutzen [Sch94]. Coombs, Brown und Ballard haben einen grundlegend neuen Ansatz vorgestellt, der eine binokulare Verfolgungsbewegung realisiert. Diese Methode kommt ohne eine Identifikation des zu verfolgenden Objekts aus [CB93]. Sie verwenden für die Detektion von Tiefenänderungen *Cepstral-Filter*, die allerdings eine hohe Anforderung an die Rechenleistung stellen. Daher verwenden sie Spezialhardware, um ein Echtzeit-System zu realisieren. Dieses System wurde von Rougeaux et. al. vereinfacht, in dem sie den *virtuellen Horopter* eingeführt haben [RKK+94]. Im folgenden Abschnitt wird die Methode der „Nulldisparitätsfilterung“ näher beschrieben.

6.2 Ein Ansatz auf der Grundlage von Nulldisparität

Die Kameranachführung mit Hilfe der Disparitätsminimierung basiert auf einer kontinuierlichen Vergenzsteuerung beider Kameras. Das Ziel ist dabei die Fixation des Objekts im optischen Zentrum beider Kameras. Der Mechanismus des Nulldisparitätsfilters (ZDF) basiert daher auf einer einfachen Grundidee: Angenommen, das Zielobjekt befindet sich bereits im Fixationspunkt, also im optischen Zentrum beider Kameras. Dann liegt das Ziel auf einem *geometrischen Horopter* und weist keine Disparität auf. Der geometrische Horopter beschreibt den Raum, in dem Objekte auf identische Positionen im linken und rechten Bild abgebildet werden. In der horizontalen Ebene verbindet er die beiden Zentren der Kameras mit dem anfixierten Blickpunkt und wird auch *Vieth-Müller-Kreis* genannt (siehe Abbildung 6.1). Im Gegensatz dazu weisen alle Objekte außerhalb des Horopters eine mehr oder weniger große Disparität auf. Werden nun alle Bildregionen ohne Disparität ausgefiltert (etwa durch eine UND-Verknüpfung beider Bilder), so können alle Objekte, die sich in einer anderen Tiefe als das Zielobjekt befinden, leicht eliminiert werden.

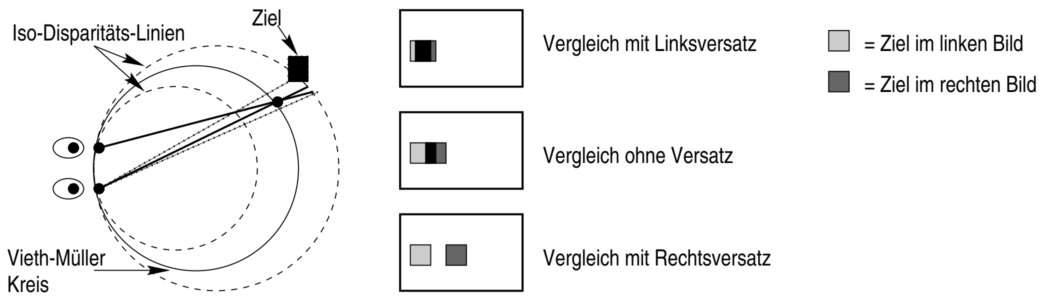


Abbildung 6.1: Tiefenänderung des sich bewegenden Zielobjektes werden mittels Iso-Disparitätslinien detektiert.

Im Idealfall weist die Ausgabe des Nulldisparitätsfilters nur noch die Bildregionen des Zielobjektes auf. Die Anpassung des Dreh- und Kippwinkels des Kamerakopfes kann anschließend auf der Basis des Regionenschwerpunktes berechnet werden. Doch wie kann eine Tiefenänderung, wenn das Objekt den theoretischen Horopter verläßt, festgestellt werden? Auch das ist unter Verwendung von Iso-Disparitätslinien — wie in Abbildung 6.1 visualisiert — relativ einfach: Es wird nicht nur die Nulldisparitätsfilterausgabe für ein exaktes Übereinanderlegen des rechten und linken Bildes berechnet, sondern auch das rechte Bild um einige Pixel horizontal verschoben. Wie in Abbildung 6.1 ersichtlich, entspricht dies in etwa einer kleinen *virtuellen* Rotation der rechten Kamera und somit einer Fixation auf alle drei Disparitätslinien. Mittels einer Verschiebung des Bildes in die eine oder andere Richtung wird also getestet, ob sich das Objekt von der Kamera weg oder auf die Kamera zu bewegt hat. Dabei wird für die drei Disparitätslinien nacheinander geprüft, auf welcher Tiefe sich das Objekt befindet. Dies kann an der maximalen Ausgabe der drei ZDF-Operationen abgelesen werden, die wiederum näherungsweise mit der aktuellen Iso-Disparitätslinie des Objektes korrespondiert. Anschließend kann der Vergenzwinkel beider Kameras an die neue Objekttiefe angepaßt werden. Sollte das Objekt zu schnelle Tiefenänderungen durchgeführt haben, so wird der Vergenzwinkel iterativ an die neue Tiefenebene des Objekts herangeführt. Probleme gibt es nur, wenn das Objekt innerhalb einer Bildaufnahme eine extreme Tiefenänderung aufweist. Dann ist eine Korrespondenz nicht mehr gewährleistet und das Ziel wird verloren. Deshalb ist die Verarbeitungsgeschwindigkeit des Systems besonders kritisch.

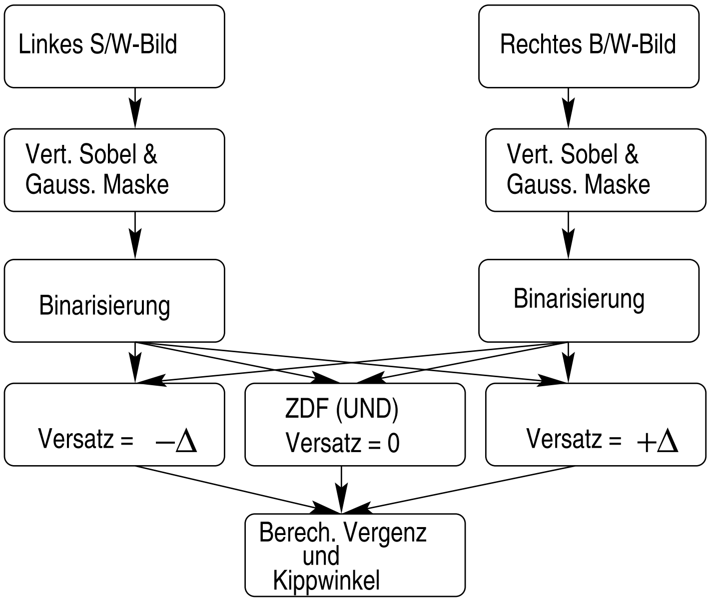
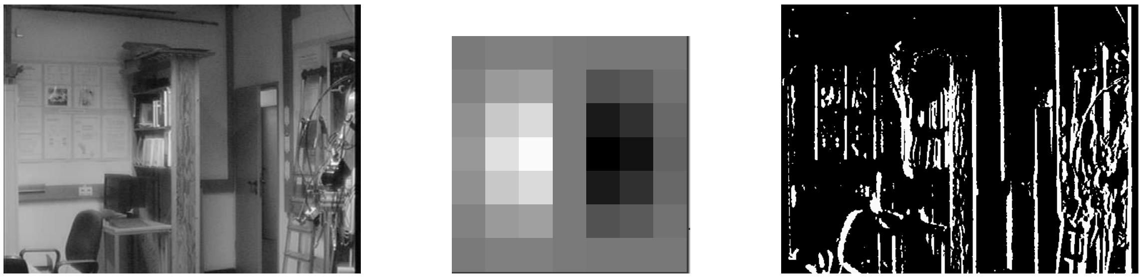


Abbildung 6.2: Die Verarbeitungsschritte des ZDF-Verfolgungsmechanismus zur Berechnung der Kameraposition.

In Abbildung 6.2 ist der verwendete Bildverarbeitungsalgorithmus dargestellt. Im aktuellen System werden vertikale Kanten verwendet, um die Objektdisparitäten zu berechnen (siehe Abbildung 6.3). Vertikale Kanten sind besonders gut für diese Aufgabe geeignet, da sich Disparität in erster Linie durch vertikale Bildverschiebungen auszeichnet Horizontale Kanten werden übereinander geschoben beinhalten keine Disparitätsinformation. Um etwas breitere vertikale Kanten für die Disparitätsminimierung zu erhalten, wird eine spezielle Filteroperation verwendet. Diese basiert auf der Überlagerung eines Sobel-Kantenfilters und einem Gauß-Filter in einer 7×7 Filtermaske, die in Abbildung 6.3(b) dargestellt ist[1]. Das ursprüngliche Schwarz/Weiß-Kamerabild wird zunächst mit dieser Maske gefaltet und das Ergebnisbild anschließend durch eine Schwellwertoperation binarisiert. Das Resultat ist ein binäres Bild mit stark ausgeprägten vertikalen Kanten (Abbildung 6.3(c)).



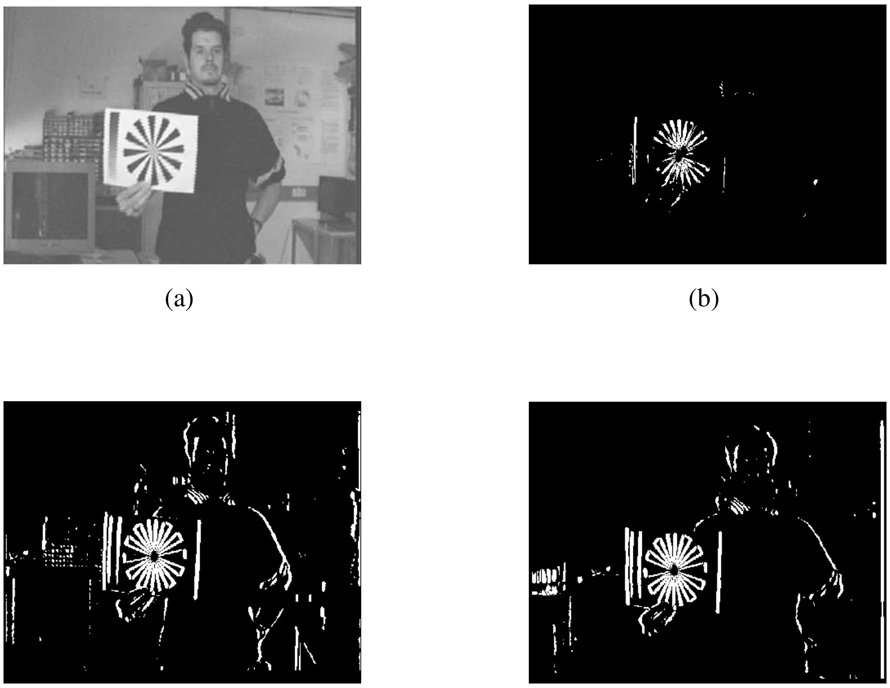
(a) (b) (c)

Abbildung 6.3: Visualisierung der Vorverarbeitung: Das Original-Kamerabild (a) wird mit der Sobel/Gauß-Filtermaske (b) gefaltet. Daneben das binarisierte Bild mit den extrahierten vertikalen Kanten (c).

Werden beide Kamerabilder in dieser Wiese vorverarbeitet, ist die Berechnung der ZDF-Ausgabe mit Hilfe einer einfachen UND-Verknüpfung des rechten und linken Bildes zu realisieren. Diese wird für alle drei Versatzschritte berechnet, und derjenige Versatz bestimmt die Adaptation der Vergenzwinkel, bei dem die meisten Pixel nach der UND-Verknüpfung im Ergebnisbild übrig bleiben. Die Abbildung 6.4(b) zeigt die Ausgabe des ZDF nach der Kombination der binären Stereo-Kantenbilder (Abbildung 6.4(c) und 6.4(d)) mit einem erfaßten Zielobjekt (Abbildung 6.4(a)).

Besonders abrupte Tiefenänderungen des Zielobjektes können nur durch eine schnelle Kameranachführung ausgeglichen werden. Daher ist eine besonders effiziente Bildverarbeitung nötig. Dies wird erreicht, indem alle Bildverarbeitungsschritte auf einem Datacube - Spezialrechner Max-Video 200 implementiert sind (siehe Abschnitt 7.1). Dadurch wird eine schritthaltende Verfolgung unterschiedlicher Objekte mit einer Bildrate von ungefähr 3 Bildern pro Sekunde erreicht. Zusammen mit der Geschwindigkeit der Kameramotoren ist damit

[1] Grundlagen der Bildverarbeitungstechniken und die Beschreibung der typischen Filteroperationen sind z.B. in [SHB93] zu finden.



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

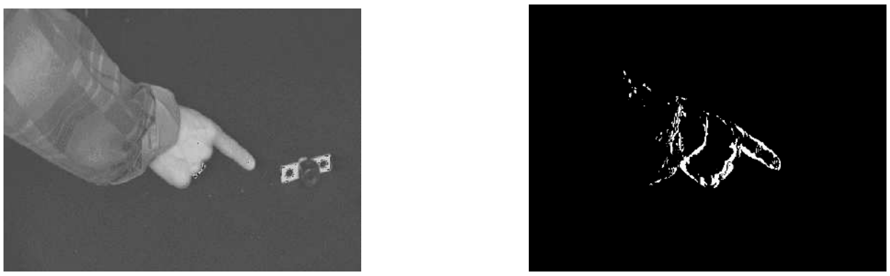
Abbildung 6.4: Die vier Verarbeitungsschritte des ZDF: (a) Originalbild (linke Kamera); (b) Mittels des Nulldisparität-Filter ermittelte Objektkanten, die sich auf der Iso-Disparitätslinie befinden; (c) Linkes und (d) rechtes Kantenbild .

eine ausreichende Performanz für die Verfolgung von Händen und Gesichtern gegeben (siehe nächster Abschnitt). Allerdings treten Probleme auf, wenn die Struktur des Objektes keine vertikalen Kanten aufweist oder wenn der Hintergrund stark texturiert oder verrauscht ist. Die Kameras werden dann unmittelbar von diesen stärker auffallenden Bildregionen angezogen, und das Ziel wird verloren. Ein weiterer Nachteil des ZDF-Verfahrens ist die Voraussetzung, daß sich das Objekt zu Beginn bereits im Fixationspunkt beider Kameras befinden muß. Deshalb wird das System in das Sakkadenmodell integriert, das zuverlässig 3D-Fixationen generiert und eine Erkennungskomponente für Handposturen besitzt. Die erzeugten Fixationen stellen sicher, daß sich das Objekt exakt im Fokus beider Kameras befindet und daß nur Objekte verfolgt werden, die auch die gewünschte Postur aufweisen. Das Sakkadenmodell bietet somit eine hervorragende Basis für ein intelligentes Handverfolgungssystem.

6.3 Verfolgung von Händen

Der Ansatz des Nulldisparitätsfilterung ermöglicht ein robustes Verfolgen von Objekten unter typischen Einsatzbedingungen. Insbesondere für die Klassifikation von Zeigegesten ist die Verfolgung menschlicher Hände unumgänglich und erfordert die Integration vom Aufmerksamkeitsmodell und dem Verfolgungsansatz. Die initiale dreidimensionale Fixation des zu verfolgenden Objektes erfolgt durch das Sakkaden-Modell aus Kapitel 5.3. Damit ist die Voraussetzung für den Einsatz des ZDF-Verfolgungsmechanismus gegeben. Aufgrund der Struktur der menschlichen Hand ist der kantenbasierte ZDF-Ansatz gut geeignet (siehe Abbildung 6.5a). Besonders die Fingerstrukturen treten gut hervor. Auch im Konstruktionsszenario, bei dem bevorzugt Zeigegesten auftreten, funktioniert das Verfahren zuverlässig. Dies liegt wiederum an der Postur bei Zeigegesten: Der Zeigefinger ist ausgestreckt, und die Hand tritt ebenfalls gut hervor (siehe Abbildung 6.5b). Mit dem System wird eine schritthaltende Verfolgung erreicht, die natürliche Handbewegungen robust erfassen kann.

(a)



(b)

Abbildung 6.5: Typische Beispiele für die Verfolgung einer Hand: (a) Originalbild der Hand (linke Kamera) und Ausgabe nach Nulldisparität-Filterung. (b) Die Ausgabe bei einer Zeigegeste.

Für die eigentliche Erkennung von Zeigegesten wird ein zusätzlicher Mechanismus verwendet, der bei einem längeren Verharren der Hand an einem Ort die Handposturerkennung auslöst und die Kontrolle wieder an das Sakkadensystem übergibt. Die genaue Integration der Teilsysteme wird in Kapitel 7.2 diskutiert.

6.4 Erweiterung zu punkt-basierten Merkmalen

Um die Probleme des kantenbasierten Ansatzes des ZDF-Algorithmus zu umgehen, wird im folgenden eine Verbesserung der Bildvorverarbeitung diskutiert. Anstatt die unspezifischen Merkmale der vertikalen Kanten zu verwenden, wird ein Ansatz vorgestellt, der *interessante Punkte* (IP) als Eingabe für den ZDF-Algorithmus verwendet [RR98a]. Dabei existieren viele Möglichkeiten, solche interessanten Punkte in einem Bild zu finden. Typische Punktmerkmale wie etwa Kantenpunkte sind relativ leicht zu detektieren. Diese werden etwa für die Verfolgung von Objekten mittels affiner Strukturen erfolgreich eingesetzt [RM96]. Andere komplexere Merkmale sind etwa Symmetriepunkte oder Farbmerkmale [HNMR96]. Noch spezialisiertere Merkmale werden etwa in Mensch-Maschine-Schnittstellen zur Verfolgung von menschlichen Gesten verwendet [NR97, VRR97]. Damit diese Punktmerkmale für den ZDF-Mechanismus eingesetzt werden können, gilt für die interessanten Punkte: IPs können prinzipiell überall im Bild auftreten, sind aber bevorzugt auf dem Zielobjekt lokalisiert. Ein Modell, welches das Verhalten solcher Punktmerkmale unter variierenden Transformationen des darunter liegenden Objekts beschreibt, sollte folgendes berücksichtigen:

- Eine (feste) Anzahl an IPs befindet sich im Bild.
- IPs können sichtbar bzw. unsichtbar sein.
- IPs können neu erscheinen und auch verschwinden.
- IPs sind für einen kurzen Zeitabschnitt Δt robust am Objekt positioniert, das heißt sie verharren an der gleichen Stelle des (bewegten) Objekts.
- Die Position der IPs kann sich auch bei Stillstand des Objekts um ein paar Pixel verschieben.

Unter Verwendung dieser Regeln können Objektbewegungen im Raum simuliert werden. Damit kann ein virtuelles Verfolgungssystem entwickelt werden, um etwa unterschiedliche Nulldisparität-Meßfunktionen zu simulieren. Eine Methode zur Berechnung der Disparität zweier Wolken p und q von Punktmerkmalen basiert auf der simplen Korrelation der Punktpositionen:

$$C = \frac{\sum_i (p_i - \overline{p})(q_i - \overline{q})}{\sqrt{\sum_i (p_i - \overline{p})^2} \sqrt{\sum_i (q_i - \overline{q})^2}} \tag{6.1}$$

Wobei p und q jeweils ein Satz der Punktmerkmale (x-Positionen) aus dem linken bzw. rechten Bild repräsentieren. Bei jedem Schritt wird die Nulldisparität berechnet, indem dieses Maß anstatt der zuvor verwendeten UND-Operation eingesetzt wird.

Auch wenn die IPs nicht stabil auf dem Objekt positioniert sind (etwa wegen Rauschen oder Überdeckungen), erzeugt die Integration vieler Punkte Unabhängigkeit von der Objektstruktur und führt zu einem robusten Verfolgungsverhalten. Auch Punktmerkmale aus unterschiedlichen Merkmalsextraktoren können überlagert werden. Dafür wird die Korrelation der Punktwolken berechnet, und genau wie beim kantenbasierten Verfahren bestimmt der maximale Wert die Vergenzbewegung. Zukünftig ist geplant, diese Methode in das Verfolgungsmodul zu integrieren. Der Einsatz neuronaler Netze soll dann helfen, die Rotationsachsen und die Struktur des bewegten Objektes zu klassifizieren. Zur Zeit sind allerdings die Anforderungen an die Rechenleistung zur Extraktion und Klassifikation der Punktmerkmale zu hoch, und somit ist die Methode noch nicht praktikabel. Deshalb wird im folgenden Gesamtsystem noch der klassische kantenbasierte ZDF-Mechanismus zur Handverfolgung eingesetzt.

7 Systemarchitektur zur dynamischen Zeigegestenerkennung: GRAVIS

DIE Systemintegration ist ein wichtiger Aspekt bei der Entwicklung von intelligenten Mensch-Maschine-Schnittstellen. Die in den vorangegangenen Kapiteln beschriebenen Module lösen klar umrissene Teilprobleme, wie etwa die Detektion interessanter Bereiche, die Verfolgung von Objekten oder die visuelle Steuerung eines Roboters. Erst die Symbiose aus diesen Einzellösungen führt aber zu einem robusten und praxistauglichen Gesamtsystem. Dieses Kapitel beschäftigt sich daher mit den Konzepten und technischen Grundlagen für die Entwicklung des Systems GRAVIS (Gesture Recognition with an Active Vision System). GRAVIS realisiert die Detektion von Bauteilen und Händen, die Erkennung von Zeigegesten und die Formulierung von Greifanweisungen für einen Roboter. Das Aufmerksamkeitsmodell von GRAVIS arbeitet auf der Grundlage neuronaler Netze und orientiert sich am Paradigma des „aktiven Sehens“. Der Vergleich mit dem zuvor vorgestellten Blickrichtungserkennungssystem GAZE verdeutlicht die unterschiedlichen Aufmerksamkeitsmodelle. Bei der Blickrichtungserkennung im System GAZE handelte es sich um eine zwar datengetriebene, aber mit zusätzlichen Modellwissen versehene Lösung. Diese Herangehensweise ist im System GAZE sinnvoll, da Gesichter im allgemeinen wenig variabel in der Anordnung der üblichen Gesichtsmerkmale (Augen, Nase und Mund) sind. Das ist jedoch grundlegend anders im Falle der Handgestenerkennung: Zum einen ist das Erscheinungsbild der Hand je nach Fingerstellung und Blickwinkel stark unterschiedlich, zum anderen werden großflächigere Bewegungen mit der Hand durchgeführt. Darüber hinaus ist es notwendig, zusätzliches Objektwissen in die Richtungserkennung mit einzubeziehen, da die Auswertung der Zeigerichtung nur sehr ungenau erfolgen kann. Daher wird hier eine rein datengetriebene Aufmerksamkeitsmodellierung gewählt, die flexibel auf neue Merkmale und Reize erweiterbar ist.

Die dafür notwendigen visuellen Verhaltensbausteine sind in einem verteilten System realisiert, das von einer zentralen Zustandsmaschine gesteuert wird. Durch die Koordination der Teilmodule kann das Gesamtsystem in einem kontinuierlichen Ablauf zwischen den einzelnen

Systemzuständen — eigenständig und der notwendigen Aufgabe entsprechend — umschalten und das visuelle Verhalten dem jeweiligen Kontext anpassen.

Um ein solches System aufzubauen, sind neben der Entwicklung der eigentlichen Bildverarbeitungsmodule weitere technische Anforderungen und Voraussetzungen zu berücksichtigen. Deshalb beginnt dieses Kapitel mit einer detaillierten Spezifikation der zum Einsatz kommenden Hardware. Besonders bei der Interaktion zwischen Mensch und Maschine ist die Verarbeitungsgeschwindigkeit des technischen Systems aufgrund ergonomischer Gesichtspunkte ein entscheidender Faktor. Um eine geeignete Ablaufgeschwindigkeit zu erreichen wurde ein verteiltes System realisiert, bei dem die einzelnen Module auf unterschiedlichen Rechnern ausgeführt werden. Die für die Interprozesskommunikation benötigte Software und die zugrundeliegenden Konzepte werden ebenfalls beschrieben.

Das Kapitel beginnt mit der Beschreibung der eingesetzten Hardware gefolgt von einer Darstellung der Gesamtarchitektur von GRAVIS . Es schließt mit einer Präsentation der erzielten Erkennungsleistung und dem Einsatz des Systems im Rahmen einer SFB-Gesamtdemonstration. Dabei wurde GRAVIS als gestikbasierte Mensch-Maschine Schnittstelle in das Konstruktionsszenario des SFB 360 integriert und ermöglicht die Referenzierung von Objekten durch Zeigegesten. Neben den in dieser Arbeit besprochenen Teilmodulen von GRAVIS sind Module aus weiteren Projekten im Rahmen der SFB-Kooperation wesentlicher Bestandteil dieses Demonstrators. Dazu zählt unter anderem ein Modul von Gunther Heidemann zur Erkennung von BAUFIX-Objekten [HR99], ein Modul zur Robotersteuerung von Christof Dücker [Düc95] und ein Modul zur Kontrolle der drei-fingerigen Hydraulikhand von Ján Jockusch [Joc00]. Die einzelnen SFB-Module kommunizieren untereinander über definierte Schnittstellen und ermöglichen somit das Design einer robusten und durch Zeigegesten geführten Greifsteuerung des Roboterarms.

7.1 Technische Grundlage

Damit das Ziel einer ergonomischen visuellen Mensch-Maschine-Schnittstelle erreicht werden kann, ist neben dem Design einer effizienten Software die hardwaretechnische Basis von entscheidender Bedeutung. Im System GRAVIS stellt ein binokularer Kamerakopf die technische Ausprägung des aktiven Sehsystems dar. Um die bildverarbeitenden Prozesse zu beschleunigen, wird darüber hinaus ein spezieller Bildverarbeitungsrechner verwendet, mit dem die Kamerabilder zunächst digitalisiert und vorverarbeitet werden. Die so erzeugten Bilder werden mittels der ebenfalls im SFB 360 von Nils Jungclaus entwickelten Kommunikationssoftware DACS [Jun98] an die verteilt laufenden Teilmodule von GRAVIS verschickt. Die Teilmodule kommunizieren untereinander über DACS und laufen als autonome Prozesse (z.T. auf unterschiedlichen Rechnern). Der hardware-seitige Aufbau und die software-basierte Kommunikation wird im folgenden erläutert.

7.1.1 Eingesetzte Hardware

Im Mittelpunkt von GRAVIS steht ein mechanischer Kamerakopf, dessen Kameras über den angeschlossenen Steuerungsrechner in Abhängigkeit von der Bildinformation gesteuert werden. Es handelt sich um einen mit 10 Freiheitsgraden ausgestatteten binokularen Kamerakopf (siehe Abbildung 7.1). Entworfen wurde er von Henrik I. Christensen und seinen Mitarbeitern an der Universität Aalborg in Dänemark [Chr92]. Der *Aalborg-University-Kamerakopf* (AUC-Head) basiert auf einer verbesserten Version aus dem Jahr 1994 [Chr94] und wird über einen VME-Bus mit einer Sun Workstation angesteuert (zu Details siehe [Kub95]).

Abbildung 7.1: Der Bielefelder AUC-Kamerakopf, ausgerüstet mit zwei Farbkameras und motorisierten Linsen.

Die gesamte Konstruktion des Kamerakopfes ist sehr funktionell gestaltet und orientiert sich am biologischen Vorbild: dem menschlichen Sehsystem. Es stellt einen Kompromiß in Geschwindigkeit, Präzision und Preis dar [Chr94]. Die zur Verfügung stehenden extrinsischen Kopfparameter sind: Vergenz ($\theta_{links}, \theta_{rechts}$; Lateral-mediale Drehung der Kameras), Drehwinkel (Θ; Drehung des unteren Halsmotors) und Kippwinkel (Φ; Drehung des oberen Halsmotors). Die intrinsischen Freiheitsgrade der beiden Kameras sind: Fokus (F), Apertur (A) und Zoom (λ).

Der AUC-Kamerakopf kann somit Schwenk-, Kipp- und Vergenz- Bewegungen ausführen. Dabei ist die Kippbewegung der beiden Kameras durch eine zentrale Achse gekoppelt. Auf die unabhängige Kippbewegung der einzelnen Kameras wurde verzichtet, um eine stabile Stereo-Bildverarbeitung zu ermöglichen. Die erreichten Geschwindigkeiten und die Präzision der Achsen sind der Tabelle 7.1 zu entnehmen.

Achse	Präzision	Verstelldauer (Real)
Vergenz–Motor links	$1/100^o$	100^o/Sek (80^o/Sek)
Vergenz–Motor rechts	$1/100^o$	100^o/Sek (80^o/Sek)
Dreh–Halsmotor	$1/100^o$	10^o/Sek (8^o/Sek)
Kipp–Halsmotor	$1/100^o$	10^o/Sek (8^o/Sek)

Tabelle 7.1: Technische Daten der Kopf-Motoren.

Die motorisierten Linsen sind jeweils mit Fokus-, Zoom- und Blendenmotoren ausgerüstet. Die technischen Daten sind in der Tabelle 7.2 aufgelistet.

Freiheitsgrad	Präzision	Geschwindigkeit (Min–Max)	Bereich
Fokus	≈ 180 Werte	7 Sek	1 m – ∞
Zoom	≈ 180 Werte	6 Sek	8 – 80 mm
Blende	Kein Sensor (Zeit)	4 Sek	F1.4

Tabelle 7.2: Technische Daten der motorisierten Linsen.

Für die Bildaufnahme werden zwei 3-Chip-CCD Farbkameras mit einer Chipgröße von $1/3''$ eingesetzt. Die beiden Kameras — die Augen des künstlichen Sehsystems — sind fest auf den Vergenzmotoren des Kamerakopfes installiert. Die software-seitige Ansteuerung der gesamten Kamerakopf-Hardware wird durch ein auf dem Steuerrechner laufendes Serverprogramm AVSERV netzwerkweit bereitgestellt (zu Einzelheiten siehe [Kub95]).

Besondere Bedeutung in einem aktiven Sehsystem kommt auch der Bildverarbeitungshardware zu. Durch eine sehr hohe Bilddurchsatzrate wird es möglich, eine *Echtzeitkontrolle* des aktiven Sehsystems zu implementieren. Dies ist besonders bei hochauflösenden Farbbildern und dem damit verbundenen Rechenaufwand mit Standardrechnern nur schwer zu realisieren. Das für diese Arbeit benutzte Echtzeitbildverarbeitungssystem der Fa. Datacube basiert auf der Max-Video 200-Bildverarbeitungskarte. Mit der dazugehörigen Digi-Color VME-Bus-Karte ist es möglich, Farbbilder von einer Kamera zu digitalisieren und in das Max-Video 200 zu übertragen.

Das Datacube-System ist ein hochspezialisierter Bildverarbeitungsrechner in Pipeline-Architektur, das heißt, ein Bild wird in einem festen Takt eingelesen, durchläuft anschließend mehrere Spezial-Recheneinheiten, und das Ergebnis steht in einem konstanten Zeittakt zur weiteren Verarbeitung zur Verfügung. Folglich ist eine Echtzeitverarbeitung von Kamerabildern in Videorate möglich (bei PAL mit 25 Bilder pro Sekunde)[1]. Die Steuerung dieses Rechners erfolgt über denselben VME-Bus, der auch für den Kamerakopf eingesetzt wird. Dadurch ist eine direkte zeitlich synchronisierte Kopplung der beiden Systeme möglich.

Im Rahmen dieser Arbeit wurden insbesondere die Algorithmen für den ZDF-Ansatz und die Auflösungsreduzierung der Kamerabilder auf der Max-Video 200-Karte implementiert. Zum Einsatz kommt dabei die Imageflow-Software, die eine Programmierung der unterschiedlichen Bildverarbeitungsalgorithmen in Pipeline-Alternating-Threads (PAT) erlaubt. Dabei wird der Rechner für einen bestimmten Algorithmus, wie etwa die Kantendetektion, vorkonfiguriert und der Systemzustand zwischengespeichert. Dies wird für alle weiteren Verarbeitungsschritte durchgeführt. Anschließend kann schnell zwischen den vorkonfigurierten Systemzuständen umgeschaltet werden (innerhalb eines Bildzyklus), und die aufgenommenen Bilder werden dem jeweiligen PAT entsprechend verarbeitet. Damit lassen sich komplexe Algorithmen effizient durch die Hintereinanderschaltung von einzelnen PATs realisieren.

Beim ZDF-Verfolgungsmechanismus (siehe Kapitel 6.2) sind das zwei PATs für die Bildaufnahme (rechtes bzw. linkes Bild), zwei PATs für die Kantendetektion sowie drei PATs für die UND-Verknüpfung der Stereobilder mit drei unterschiedlichen Offsets. Alle sieben PATs werdennach jeder Bildaufnahme hintereinander ausgeführt und führen zur erwünschten Berechnung der Objektposition aus den Bilddaten in einer Rate von ungefähr drei Bildern pro Sekunde (25Hz Bildrate und Ausführung von 7 PATs pro Bildauswertung). Die Bildkoordinaten des so segmentierten Objekts werden anschließend verwendet, um die Kameramotoren entsprechend anzusteuern. In ähnlicher Weise wird auch die Auflösungsreduktion der Bilder für das System GRAVIS durchgeführt. Dabei werden die reduzierten Bilddaten an die zuständigen Programme (Aufmerksamkeitsmodell, Visualisierung etc.) übertragen.

7.1.2 Softwaretechnische Realisierung

Das System GRAVIS besteht aus einer Reihe von Teilmodulen: dem Aufmerksamkeitsmodell, dem ZDF-Verfolgungsansatz sowie der Hand-Auge-Koordination (siehe Abschnitt 7.4.1 weiter unten). Für jedes Teilmodul sind spezielle Schnittstellen definiert, über die die Interprozeß-Kommunikation stattfindet.

Für die Implementation der Teilmodule kommt das von Helge Ritter entwickelte objektorientierte Software-Paket NEO zum Einsatz[2]. Dieses über mehrere Jahre entstandene Werkzeug

[1] Der Nachteil dieses Systems besteht allerdings in der sehr komplizierten Programmierung.
[2] Dieses Werkzeug basiert auf NST (Neural Simulation Tool); daher Nst graphical EditOr (Neo) [Rit95].

stellt sowohl für die Konstruktion von neuronalen Netzen als auch für die Bildverarbeitung viele vorgefertigte Module zur Verfügung. Damit ist der Programmierer — zumindest in der frühen Entwicklungsphase von Algorithmen — von der komplizierten Programmierung (wie etwa der Datacube-Hardware) befreit. NEO stellt eine grafische Oberfläche als Frontend für die Programmierung zur Verfügung. Damit ist es möglich, durch einfache Operationen grafisch Objekte in Datenfluß-Schaltplänen zu verschalten. Auf dieser Grundlage wurden das Aufmerksamkeitsmodell, die Hand-Auge-Koordination sowie verschiedene Visualisierungs- und Kontrollmodule implementiert. Die Teilmodule stellen unabhängige Neo-Schaltkreise dar, die auf unterschiedlichen Rechnern gestartet werden können.

Die Kommunikation zwischen den Teilmodulen basiert auf DACS (Distributed Application's Communication System). Dabei handelt es sich um eine von Nils Jungclaus im Rahmen des SFB 360 entwickelte Software-Schicht für eine thread-basierte Interprozesskommunikation [Jun98]. Über frei definierbare Strukturen können auch komplexe Datenobjekte asynchron über Rechnergrenzen hinweg transferiert werden. Besonderes Augenmerk liegt auf der Unabhängigkeit von Betriebssystemen und Rechnerarchitekturen. Damit ist DACS auch für die Kommunikation mit Spezialhardware, wie etwa dem Datacube, geeignet. Im System GRAVIS basiert fast der gesamte Datenaustausch zwischen den Teilmodulen auf DACS [JRR98]. Zum einen werden die Kamerabilder an das Aufmerksamkeitsmodell geliefert, zum anderen wird die Synchronisation zwischen den Teilmodulen über prozeßübergreifende Funktionsaufrufe realisiert. Beispielsweise verschickt das Aufmerksamkeitsmodul eine Nachricht an das Verfolgungsmodul, wenn eine Hand detektiert wurde, um damit das Verfolgungsverhalten auszulösen. Während der Verfolgung wird dann das Aufmerksamkeitsmodell blockiert, bis das Verfolgungsmodul die Kontrolle wieder abgibt.

Eine zentrale Stellung im System GRAVIS nimmt das Teilmodul AVIS ein. Hierbei handelt es sich um das Steuerprogramm, das die Verteilung der Hardwareressourcen übernimmt. Dazu zählt etwa die Steuerung des Kamerakopfs und die Versendung der Kamerabilder. Das Programm läuft auf dem Steuerungsrechner für den Kamerakopf und den Datacube-Rechner und gewährleistet eine synchronisierte Ansteuerung der beiden Geräte, was insbesondere bei der Handverfolgung wichtig ist, da hier Bildaufnahme und Kontrolle der Kopfmotoren in einer Echtzeitschleife durchlaufen werden. Die Steuerung der Kameras geschieht dabei zwischen den für den ZDF-Algorithmus nötigen PATs. Alle anderen Teilmodule bekommen die benötigten Bilder entsprechend über eine DACS Schnittstelle per Funktionsaufruf zugeschickt. Die Bildaufnahme kann über die DACS-Funktionsaufrufe synchronisiert werden. In Abbildung 7.2 ist der zeitliche Ablauf der Bildaufnahme für das Aufmerksamkeitsmodell dargestellt. Bei jeder Berechnung einer neuen Sakkade wird die Aufnahme durch das Versenden einer Nachricht gestartet. AVIS nimmt diese Anfrage entgegen, startet auf dem Datacube die entsprechenden PATs und verschickt anschließend die Ergebnisse an die Applikation. Somit ist

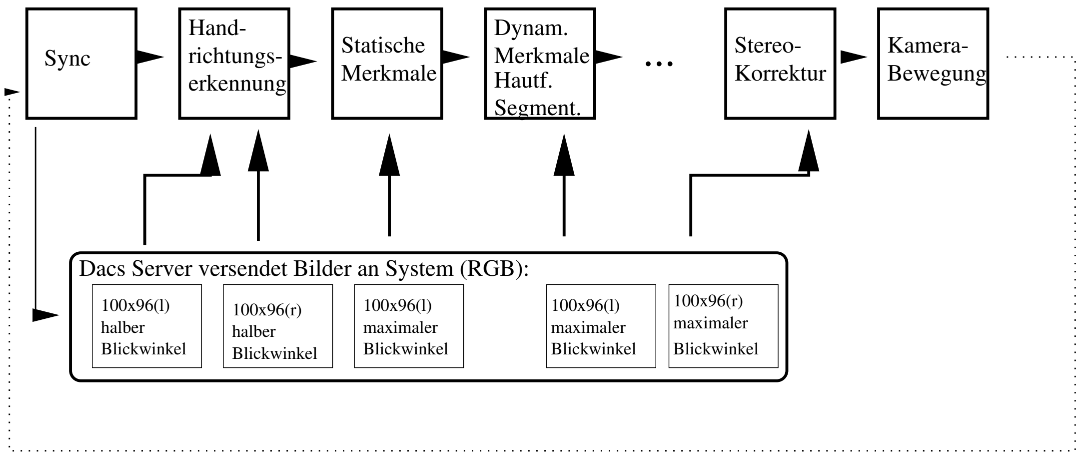


Abbildung 7.2: Synchronisation und zeitliche Optimierung der Bildübertragung an das Aufmerksamkeitsmodell mittels DACS-Funktionsaufrufen (aus [Fis98]).

sichergestellt, daß jedem Teilmodul der aktuellen Situation entsprechend die benötigten Ressourcen zur Verfügung stehen. Im nächsten Abschnitt wird das Zusammenspiel der einzelnen Teilmodule und deren Koordination über eine Zustandsmaschine in AVIS beschrieben.

7.2 Verteilte Implementierung von GRAVIS

Mit dieser Arbeit werden mehrere wichtige Komponenten aktiven visuellen Verhaltens, die in vergangenen Jahren als Einzelmodule realisiert wurden, in ihrem Zusammenspiel untersucht. Dabei bildet die gestikbasierte Mensch-Maschine-Interaktion den Integrationsrahmen. Bisher ist ein aus drei Teilmodulen bestehendes System realisiert, das die folgenden Verhaltensmuster zeigt (Abbildung 7.3):

- *Exploration* der Umgebung und der darin auftretenden Gegenstände durch sakkadische Kamerabewegungen.
- *Detektion* von Händen und Verfolgung von deutlichen Handbewegungen.
- *Erkennung* von Zeigegesten und Auswertung der Zeigerichtung mit anschließender Fixation der Objekte im Zeigefokus.

Die Verhaltensmuster werden über die bereits vorgestellten Module realisiert. Die Exploration basiert auf dem Aufmerksamkeitsmodell und sorgt dafür, daß de in der Szene vorhandenen

Objekte, nacheinander anfixiert werden. Umgekehrt ist es auch möglich, daß von außen ein Raumpunkt vorgegeben wird und der Kamerakopf diesen Punkt fixiert, um eine gewisse Bildregion näher zu inspizieren. Somit sammelt das Gesamtsystem fortlaufend Information über die Szene. Dabei werden bevorzugt Bildregionen extrahiert, die einen hohen Interessantheitsgrad aufweisen (siehe Kapitel 5.3). Die Kamerabewegungen erfolgen dabei im Bildtakt des Aufmerksamkeitsmodells, also mit circa einer Sakkade pro Sekunde. Diese Verarbeitungsgeschwindigkeit reicht zur Exploration einer statischen Szene sicherlich aus. Auch wenn Objekte verschoben werden oder neue hinzukommen, ist durch die fortlaufende Neuberechnung der Aufmerksamkeitsverteilung gewährleistet, daß alle wichtigen Bereiche der Szene erneut betrachtet werden.

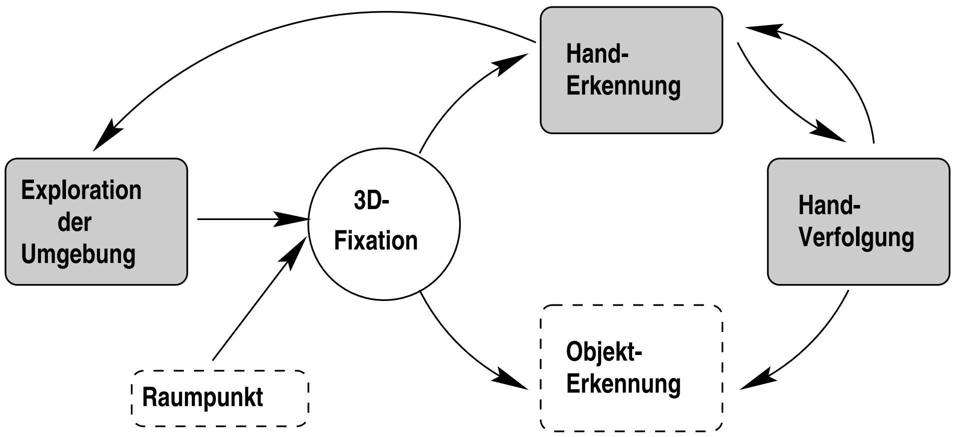


Abbildung 7.3: Die drei Systemzustände: Exploration, Handerkennung und Handverfolgung. Ebenfalls schematisch dargestellt: der Informationsweg zum Objekterkenner, der die jeweils fixierte Region auswerten kann, sowie der Eingabekanal zum Fixieren eines Raumpunktes.

Zur kontinuierlichen Verfolgung schneller Objekte reicht die Verarbeitungsgeschwindigkeit allerdings nicht aus. Eine Hand wird zwar vom Aufmerksamkeitsmodell mittels einer abrupten Sakkade mit beiden Kameras anfixiert, die flüssige Kameranachführung ist jedoch nicht möglich. Zur Beobachtung von dynamischen Handbewegungen ist daher ein zusätzliches Modul notwendig. Daher wird nun der ZDF-Verfolgungsmechanismus (Kapitel 6.2) abhängig von einer weiteren Analyse des Bildausschnittes gestartet: wird in diesem Bildausschnitt initial eine Zeigegeste erkannt, wird das als der Anfang einer dynamischen Geste interpretiert. Der Verfolgungsmechanismus sorgt dann für eine schnelle und flüssige Beobachtung der Hand während der Ausführung der Geste. Der Benutzer des Systems kann nun in seiner natürlichen Bewegungsgeschwindigkeit die Geste ausführen. Am Ende der Gestentrajektorie verharrt die Hand typischerweise in der Zeigegeste am Zielort. Dieses Verharren wird aufgrund der ebenfalls zum Stillstand kommenden Kamerabewegungen registriert.

Daraufhin wird wiederum die aktuell anfixierte Bildregion ausgewertet, in der sich die Hand befindet. Wird nun erneut eine Zeigegeste detektiert, wird durch das Handerkennungsmodul die Zeigerichtung bestimmt (Kapitel 5.7). Diese Richtung wird übersetzt in einen Intensitätskegel, mit dem in der Aufmerksamkeitskarte das entsprechende Objekt hervorgehoben wird. Die Kameras werden anschließend durch eine Sakkade auf das referenzierte Objekt ausgerichtet. Die Objekterkennung kann dann detailliert auf den extrahierten Bildregionen erfolgen.

Anschließend beginnt die Exploration der nun entstandenen Szene erneut. Die Abläufe sind im zentralen Modul AVIS implementiert. Hierbei handelt es sich um eine Zustandsmaschine, die über entsprechende Nachrichten die Koordination der anderen Module übernimmt. Dabei werden die einzelnen Zustände abhängig vom aktuellen Zustand umgeschaltet, dadurch können gewisse Ablaufsequenzen vorgegeben werden. Die möglichen Übergänge sind in Abbildung 7.3 dargestellt.

Aus der modularen Aufteilung der einzelnen Fähigkeiten des Systems entstehen viele Vorteile:

- Die Komponenten können auf mehrere Rechner verlagert werden.
- Einzelne Komponenten können bei Ausfall problemlos neu gestartet werden, ohne das restliche System zu stören.
- Mittels einfacherer Kontrollmodule kann der Zustand des Gesamtsystems direkt gesteuert werden (etwa durch das Absetzen einer Nachricht zur Fixation eines Raumpunktes).
- Die einzelnen Komponenten werden unabhängig voneinander an die aktuelle Situation angepaßt (z.B. durch ein Nachtrainieren der neuronalen Netze).

Insgesamt läßt sich feststellen, daß die Kombination dieser Systemkomponenten zu einem robusten Gesamtsystem führt. Insbesondere bei der Verwendung so komplizierter Systeme wie Kamerakopf und Datacube kommt es manchmal zu Systemabstürzen. Durch das unabhängige Neustarten der betroffenen Komponente ist ein schnelles Wiederherstellen der Gesamtfunktion möglich. Die Verteilung der Komponenten auf unterschiedliche Rechner vermeidet Performanz-Engpässe. So ist es besonders wichtig, das Aufmerksamkeitsmodell auf einem separaten Rechner laufen zu lassen. Ansonsten würde die aufwendige Berechnung der Bildmerkmale eventuell die Zustandsmaschine blockieren. Das zentrale Schaltmodul AVIS bleibt so funktionsfähig und kann Anforderungen von anderen Modulen abarbeiten.

Die Architektur des Systems erleichtert auch die Integration weiterer Module. So werden die anfixierten Bildregionen über das Netz an andere Module etwa zu einer detaillierten Analyse weitergeleitet. Ebenso ist eine weitergehende Analyse der Gestentrajektorie während der Verfolgung realisierbar. Damit kann ein System zur Erkennung dynamischer Handlungsgesten wie zum Beispiel Greifbewegungen aufgebaut werden.

Generell liegt der Vorteil in der situationsabhängigen Zusammenstellung der Module. Die Bausteine ermöglichen konkrete Problemlösungen, die in unterschiedlichen Kontexten zum Einsatz kommen. So wird etwa das Zeigegestenerkennungsmodul zu zwei unterschiedlichen Zeitpunkten verwendet: (i) Nach einer Sakkade zur Überprüfung ob die Handverfolgung gestartet werden soll und (ii) nach Beendigung der dynamischen Verfolgung zur Erkennung der eigentlichen Zeigerichtung. Je nach Bedarf beteiligen sich also die hochspezialisierten Module an der Lösung eines Problems.

Die Koordinierung der einzelnen Abläufe übernimmt die Zustandsmaschine im AVIS Modul. Da die Nachrichtenstruktur sehr einfach aufgebaut ist, sind sehr leicht Experimente mit unterschiedlichen Architekturen möglich. Module können sogar in mehrfacher Instanz verwendet werden: Es existiert etwa ein Modul zur Anzeige der jeweils fokussierten Bildregion. Dasselbe Modul dient in einem anderen Kontext zur Visualisierung von Robotergreifbewegungen (siehe dazu Kapitel 7.4). Mit Hilfe der schnellen Konfigurierbarkeit des Datacube kann dieses Programmierparadigma auch auf die Spezialalgorithmen auf dem Datacube übertragen werden. Deshalb werden etwa je nach Situation unterschiedlich ausgeprägte Varianten des gleichen Trackingalgorithmus von der Zustandsmaschine aus gestartet.

Da in GRAVIS die Systemzustände sequentiell durchlaufen werden, können zeitliche Abstimmungsprobleme vermieden werden. Die Aufteilung der Ressourcen ist ohnehin unkritisch, weil nur ein Kamerakopf zur Verfügung steht: Die Ressource steht zu einem Zeitpunkt nur genau einem Modul zur Verfügung. Die Integration von Aufmerksamkeitsmodell und Handverfolgung ist recht einfach, da die Anfangs- und Endposition des Kamerakopfes jeweils als Startpunkt für das entsprechende Modul dient. Komplizierter sind Vorgänge, die eine plötzliche Änderung der Kameraparameter erfordern. So kann sich etwa die detaillierte Analyse eines Objekts durch das Heranzoomen der Bildregion vereinfachen. Werden dann anschließend neue Anforderungen von anderen Modulen gestellt, muß die ursprünglichen Kamerakonfiguration zurückgespielt werden. Dies wird wiederum zentral in der Zustandsmaschine verwaltet, die in der Lage ist, Zustände des Kamerakopfes zu speichern und bei Bedarf wieder herzustellen. Jedes Teilmodul kann somit unabhängig die jeweils benötigten Kameraparameter einstellen, ohne daß nachfolgende Module beeinflußt werden.

Die vorgestellte Architektur des Systems erlaubt also eine hohe Flexibilität und Integrierbarkeit. Besonders die Bildverarbeitungsmodule werden einzeln konfiguriert, etwa um sie an eine neue Beleuchtungssituation anzupassen. Die Verknüpfung der Module ist ebenfalls leicht: Es werden einfach neue Teilfunktionen in die Zustandsmaschine eingehängt werden. Die Abfolge der Module kann entweder fest vorgegeben werden, oder sie kann durch Benutzerschnittstellen zur Laufzeit verändert werden. Über diesen Mechanismus kann der Benutzer beispielsweise ein Bild eines bestimmten Objekts abrufen, indem er diese Anforderung über die entsprechende Nachricht absetzt. Die Zustandsmaschine sammelt diese Anforderungen und arbeitet sie in der Reihenfolge des Auftretens ab. Die so erzeugten Bilder werden mit einem Zeitstempel an das entsprechende Modul zurückgesandt.

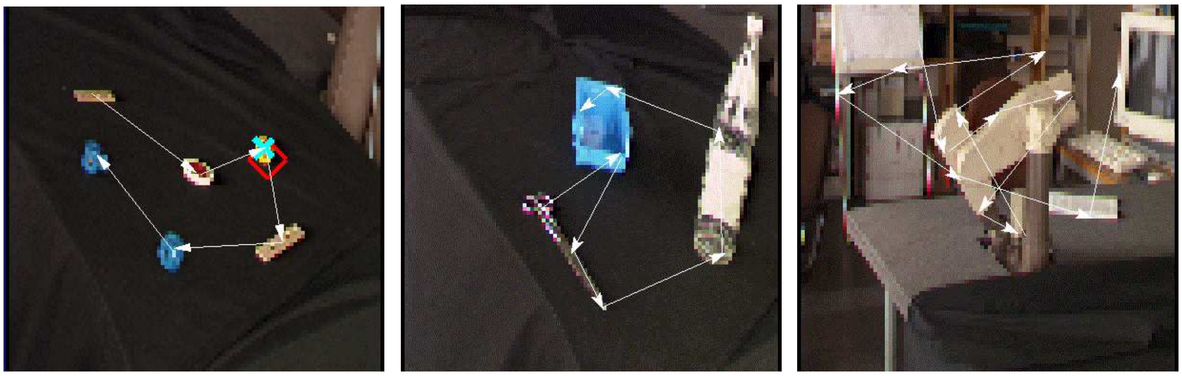
Derzeit erfolgt die Abarbeitung der Anfragen stets sequentiell, es kann aber auch sinnvoll sein, eine Priorisierung der Anfragen einzubauen. Besonders bei der Integration eines Roboters kann es sinnvoll sein, etwa die Greifbewegung des Roboters zu überwachen. Während dieses zeitkritischen Vorgangs ist die ungeteilte Aufmerksamkeit des Kamerakopfes erforderlich. Anfragen an die Zustandsmaschine, andere Bereich zu inspizieren, sollten also während dieser Phase ignoriert werden. Das gilt auch während der dynamischen Handverfolgung. Dies kann erreicht werden, indem die Teilmodule unterschiedliche Priorisierungsstufen erhalten, die in kritischen Situationen eine bevorzugte Abarbeitung vor anderen Anforderungen erlaubt. Diese Priorisierung ist im aktuellen System vorläufig nur unvollständig implementiert, die in der Handverfolgung nicht von außen unterbrochen werden kann. Damit wird vermieden, daß der Kamerakopf die Hand verliert. Erst nach Abschluß der Verfolgung übernimmt wieder das Aufmerksamkeitsmodell die Kontrolle.

7.3 Performanz des Gesamtsystems

Im folgenden werden die Aspekte der gestikbasierten Interaktion und die Erkennungsleistung des künstlichen Sehsystems diskutiert. Zunächst wird das Verhalten des Explorationsmoduls illustriert und einige qualitative Resultate der Erkundung dynamischer und statischer Szenen sowie die Leistungsfähigkeit des Verfolgungsmoduls präsentiert. Anschließend wird näher auf die Handgesten- und Zeigerichtungserkennung eingegangen. Das Zusammenspiel der drei Komponenten resultiert schließlich in einem robusten Erkennungsverhalten des Gesamtsystems, das abschließend vorgestellt wird und den Einsatz als Mensch–Maschine Schnittstelle zeigt.

7.3.1 Statische Szene

Das Explorationsverhalten des Aufmerksamkeitsmoduls wird innerhalb einer statischen Anordnung von Gegenständen anhand von zwei typischen Blicktrajektorien vorgestellt. Die Abbildung 7.4(a) zeigt das Blickverhalten des künstlichen Systems in einem Szenario, das aus einer Ansammlung von verschiedenen kleinen Holzspielzeugteilen besteht. Alle Objekte werden nacheinander betrachtet, ohne daß ein Objekt zweimal fixiert wird. Die Abbildung 7.4(b) zeigt eine Szene mit größeren Gegenständen. Dabei wird deutlich, daß jetzt durchaus ein Objekt mit zwei Sakkadensprüngen oder mehr bedacht wird. Die Berechnungszeit einer Sakkade beträgt im Aufmerksamkeitsmodell inklusive der Bildaufnahme circa eine Sekunde.



(a) (b) (c)

Abbildung 7.4: Die Kamerasakkaden während der Exploration unterschiedlich komplexer statischer Szenen. (a),(b) Blicktrajektorien des Systems im typischen Arbeitsbereich. (c) Aufmerksamkeitsgesteuerte Erkundung des Labors.

Insgesamt generiert das Modell zuverlässige Stereo-Fixationen auf diskreten Objekten, und die daraus resultierenden Blicktrajektorien sind menschlichen Blicktrajektorien sehr ähnlich. Als Illustration dient in Abbildung 7.4(c) die Erkundung eines Labors: der Kamerakopf betrachtet nacheinander besonders auffällige Bildregionen (vergleiche dazu Kapitel 2.3.2).

Aufgrund der Systemarchitektur des visuellen Sakkadengedächtnisses nimmt die Vollständigkeit der Exploration mit der Anzahl der Objekte ab. Dieses Verhalten ist in Abbildung 7.5 illustriert: Bei 5 Objekten genügen noch 5 Sakkaden, um alle Objekte zu besuchen, bei 7 Objekten sind es schon 10, und bei mehr als 7 Objekten kann nicht garantiert werden, daß alle Objekte erreicht werden. Das Problem liegt in der Ausblendungskarte, dem visuellen Kurzzeitgedächtnis des Systems. Da die Ausblendung zeitabhängig erfolgt, werden bereits besuchte Regionen nach kurzer Zeit wieder interessant, und verhindern so die Exploration unbekannter Bildregionen (zu Einzelheiten siehe [Fis98]). An dieser Stelle sind Verbesserungen im Modell vorzunehmen. Ein Ansatz könnte der Aufbau eines Mosaikbilds als Langzeitgedächtnis darstellen. Dabei werden die Einzelansichten während der Erkundung mit entsprechenden Bildtransformationen zu einem großen Bild zusammengefügt. Der Ausblick (Kapitel 9) geht genauer auf diese Erweiterung ein.

7.3.2 Dynamische Szene

Untersuchungen des Fixationsverhaltens im Aufmerksamkeitsmodell zeigen, daß bewegte Objekte aufgrund der Bewegungskarte zeitweilig verfolgt werden. Die Objekte werden in diesem

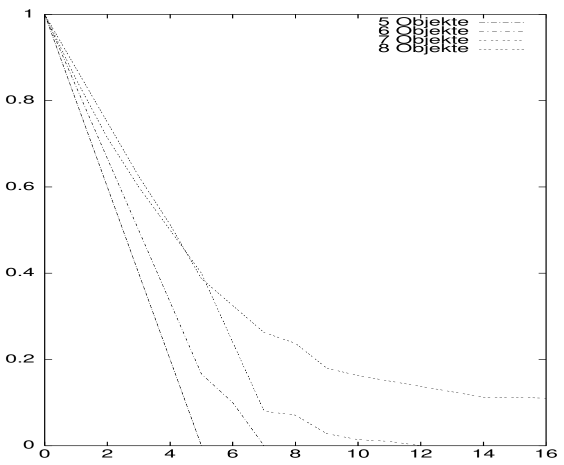


Abbildung 7.5: Anzahl der während der Exploration nicht erreichten Objekte in Testszenen (Ordinate: Anteil noch zu observierender Objekte, Abzisse: Anzahl der Fixationen).

Test auf eine rotierende Scheibe gelegt. Die Sakkaden zielen in diesem Fall nur dann auf das Objekt, wenn das Objekt schnell genug aus dem inhibitierten Bereich in der Aufmerksamkeitskarte heraustritt.Bei kleinen Bewegungen hingegen verliert der Kamerakopf das Objekt, da es ausgeblendet wird (zu Einzelheiten siehe [Fis98]).

Aufgrund der im Aufmerksamkeitsmodell implementierten bewegungsselektiven Karte und der stark gewichteten Karte zur Detektion bewegter hautfarbener Bildbereiche reagiert das System besonders aufmerksam auf Bewegungen hautfarbener Objekte. Daher fixiert das System bevorzugt bewegte Hände. Falls eine Hand erkannt wird, wird entschieden, ob eine Zeigegeste vorliegt. Ist das der Fall, wird der Verfolgungsmechanismus gestartet. Der Aufmerksamkeitsmechanismus selbst wird jedoch nicht zum Verfolgen der Hand eingesetzt. Das hat zwei Gründe: Zum einen die derzeit hierfür noch zu geringe Geschwindigkeit der Verarbeitung, und zum anderen das Ermüdungsverhalten des Modells (siehe Abschnitt 5.5). Dieses würde ein dauerhaftes Verfolgen aufgrund des nachlassenden Interesses verhindern.

7.3.3 Handverfolgung

Wurde innerhalb des fixierten Bildbereichs eine Zeigegeste erkannt, so wird zunächst das Verfolgungsmodul aktiviert. Insbesondere wurde durch die vorangegangene Fixation des Aufmerksamkeitsmoduls gewährleistet, daß sich das Objekt genau im Mittelpunkt beider Kameras befindet und somit die Voraussetzungen für den Einsatz des Nulldisparitätsfilters erfüllt sind. Der disparitätsbasierte Ansatz funktioniert zuverlässig und ist in der Lage, Hände mit einer

Bildrate von $3\frac{4}{7}$Hz zu verfolgen. Die Verfolgung einer Hand vor einem Hintergrund mit komplexen Gegenständen ist dabei flüssig und robust.

Die maximale Ausgabe des ZDF-Filters bestimmt die Vergenz der Kameras, der Schwerpunkt der Bildregion die Nachführung des Dreh- und Kippwinkels. Der ZDF-Mechanismus basiert auf binarisierten Kantenbildern und zeigt prinzipbedingt Probleme bei stark strukturierten Hintergründen, die sich aufgrund der kantenbezogenen Bildauswertung in fehlerhaften Kameranachführungen niederschlagen. Deshalb ist in Zukunft eine Modifikation des Nulldisparitätsfilters in Bezug auf die zu verwendenden Bildmerkmale sinnvoll.

7.3.4 Zeigegesten und -richtungserkennung

Um die Qualität der Handgestenerkennung zu bewerten, werden die Erkennungsergebnisse für vorsegmentierte Zeigegesten untersucht. Insgesamt werden in dem Testsatz 120 Bilder mit statischen Handgesten von 3 Benutzern benutzt, die zuvor manuell klassifiziert wurden. Auf dieser Testmenge ergibt sich mit dem Handgestenerkennungsmodul eine Erkennungsrate von 98% korrekt identifizierter Zeigegesten. Dieser Test beinhaltet allerdings noch nicht das dynamische Fixationsverhalten und die Segmentierungsleistung des Gesamtsystems. Deshalb wurde ein weiterer Versuch durchgeführt, der zeigen soll, wie gut das Modell Handgesten detektiert. Bei dem Test führen drei Benutzer 80 spontane Zeigegesten während der Exploration durch das Aufmerksamkeitsmodell aus. Dabei werden 75 Handgesten korrekt von beiden Kameras im ersten Versuch detektiert und fixiert, die restlichen Gesten erst im zweiten oder dritten Versuch. 76% dieser korrekt fixierten Zeigegesten wurden richtig erkannt. In einem weiteren Test mit alternativen Gesten der Benutzer soll gezeigt werden, daß die Handgestenerkennung falsche Handposturen abweist. Dabei werden 13% von 80 alternativen Handgesten, die keiner echten Zeigegesten entsprachen, fälschlicherweise als Zeigegeste akzeptiert. Ist eine echte Zeiggeste erkannt worden, folgt die eigentliche Erkennung der Zeigerichtung. Die Vektorapproximation der Zeigerichtung resultierte im Experiment in einem durchschnittlichen Fehler von ca. 5^o bei einer maximalen Abweichung von 15^o. Weitere Ergebnisse sind in [Fis98] dokumentiert.

7.3.5 Gestikinteraktion

Abschließend wird das Verhalten des Gesamtsystems bei gestikbasierter Interaktion mit einem menschlichen Benutzer illustriert. Dafür wird dem aktiven Sehsystem während einer Testsequenz eine Anzahl von Zeigegesten präsentiert, die im Abstand von ca. 1m vor der Kamera und ca. 25cm von den Gegenständen entfernt ausgeführt werden. Dabei werden jeweils 20 Zeigegesten innerhalb einer Szene mit unterschiedlicher Anzahl von Objekten ausgeführt [Fis98].

Szene	Korrekt identifiziert
3 Gegenstände	75 %
5 Gegenstände	60 %
7 Gegenstände	40 %

Tabelle 7.3: Erkennungsleistung des Gesamtsystems bei Zeigegesten verschiedener Benutzer bei unterschiedlich komplexen Szenen.

Bei drei Objekten in der Szene liegt die Gesamterkennungsleistung bei 75% korrekt referenzierter Objekte (Tab. 7.3). Wird jedoch die Anzahl der Gegenstände erhöht, so wird gleichzeitig die Erkennungsleistung geringer. Das resultiert aus der Ungenauigkeit der 2D Zeigerichtungsauswertung. Der Fehler liegt hier im Mittel bei etwa 5^o. Deshalb hängt die Erkennung zum einen von der Anzahl und Dichte der auftretenden Gegenstände und zum anderen vom Abstand der Zeigegeste zum Objekt ab. Zukünftig könnte eine dreidimensionale Erkennung der Zeigegesten verbesserte Ergebnisse bringen.

7.4 Integration in einem SFB 360 Demonstrator

Die vorliegende Arbeit ist im Rahmen des Sonderforschungsbereichs 360 an der Universität Bielefeld entstanden. Ziel dieses SFBs ist die Erforschung „künstlicher situierter Kommunikatoren“ in Bezug auf linguistische und kognitive Merkmale, die bei der menschlichen Kommunikation beobachtet werden. Neben der empirischen Beobachtung solcher Prozesse steht besonders die Übertragung und Nutzbarmachung kognitiver Prinzipien auf künstliche Robotersysteme im Mittelpunkt. Dabei sind Aspekte der situativen Abhängigkeit, der Robustheit, Flexibilität, Wissensabhängigkeit und die integrative Funktion kommunikativer Prozesse zu berücksichtigen. Als Forschungsszenario dient eine Montagesituation, in der ein Mensch mit einem künstlichen Manipulator kooperiert. Die während der Konstruktion geäußerten Anweisungen werden mittels Sprach- und Bildverarbeitung erkannt und in konkrete Anweisungen an einen Roboter umgesetzt, der die Bauteile zusammenbaut. Ein wichtiges Paradigma, das auch in dieser Arbeit eine zentrale Stellung einnimmt, ist das an Vorbildern der Natur orientierte Design eines integrierten technischen Systems, . Ein weiterer wichtiger Aspekt ist die Realisierung von flexiblen und anpassungsfähigen Problemlösungen, um ein robustes und zuverlässiges Gesamtsystem zu erhalten. Anwendungsfelder solcher künstlichen Systeme liegen im Bereich künftiger Mensch-Maschine-Schnittstellen, etwa für die Realisierung „sehender Interfaces“ zu komplexen Softwaresystemen, oder im Bereich der intuitiven und ergonomischen Kooperation mit Robotern.

Der SFB besteht aus einer Reihe von Einzelprojekten, die unter anderem im Rahmen eines realen Demonstrationssystems kooperieren. Eine interessante Fragestellung ist daher, wie die Integration der erforschten Methoden in einem technischen System erfolgt. Die Fähigkeit des künstlichen Robotersystems kann so erweitert werden, um mit einem Menschen über Sprache und Gestik zusammenzuarbeiten und aus einzelnen Bauteilen[3] ein Aggregat zu fertigen. Der Benutzer kann dann den komplizierten Bauvorgang durch das künstliche System (etwa beim Zusammenbau eines Flugzeuges) mit Sprach- und Gestikanweisungen aktiv unterstützen (siehe Abbildung 7.6).

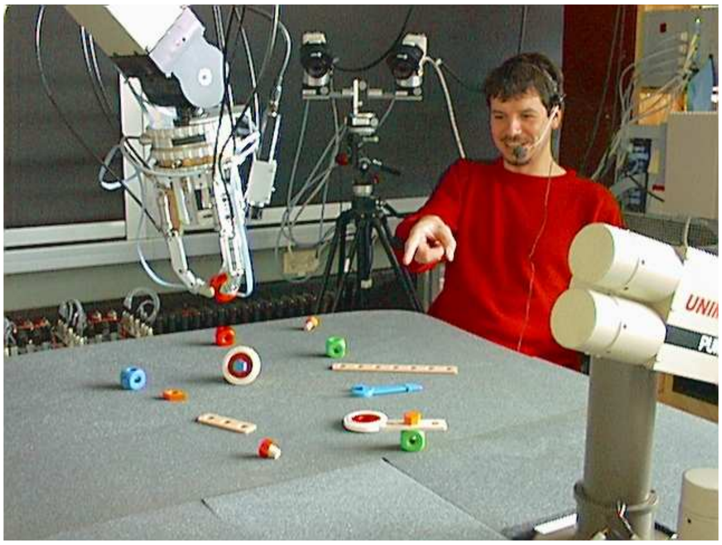


Abbildung 7.6: Der prinzipielle Aufbau des SFB 360 Demonstrators: Ein Roboter ausgestattet mit einer dreifingerigen Hydraulikhand, der binokulare Kamerakopf und der menschliche Benutzer, der über Sprache und Gestik mit dem Robotersystem kommuniziert.

Zu Demonstrationszwecken sind im Laufe des SFBs einige Systeme entstanden, die den jeweils erreichten Stand der Entwicklung präsentieren. Jedes dieser Systeme umfaßt primär einzelne Komponenten zum Sprachverstehen oder zur Roboterkooperation. Als eines dieser Projekte ist ein Demonstrationssystem entstanden, bei dem das aktive Sehsystem GRAVIS mit einem menschlichen Konstrukteur interagiert und die notwendigen Information für ein Robotersystem akquiriert um einen Greifvorgang erfolgreich auszuführen. Das gemeinsam mit anderen Teilprojekten entwickelte System demonstriert die Interaktion zwischen dem Robotergreifsystem und einem menschlichen Benutzer auf der Grundlage von Zeigegesten. Ziel

[3] Hierbei handelt es sich um Holzbauteile, die über Schraub- und Steckverbindungen zusammengebaut werden können (Baufix-Spielzeugteile).

dieses Demonstrators ist es, zu zeigen, daß der menschliche Benutzer Objekte in dem Szenario über Zeigegesten referenzieren kann. Der Roboter soll diese Objekte dann erkennen und selbstständig aufnehmen.

Alle Systemkomponenten sind eigenständige Programmteile, die auf unterschiedlichen Rechnern arbeiten. Die Programme kommunizieren über definierte DACS-Schnittstellen [JRR98]. Insgesamt besteht das System aus vier Teilkomponenten:

- Einem Puma-Roboter, der über ein Steuermodul von Christof Dücker kraftgeregelt in kartesischen Weltkoordinaten angesteuert wird [Düc95].
- Einem neuronalen Objekterkenner, der in zwei Ausprägungen eingesetzt wird: zum einen zur Erkennung der Bauteile in der Gesamtszene (Overhead-Kamera) und zum anderen zur Detektion der Bauteile im Greifbereich (Handkamera) [Hei98].
- Einer drei-fingerigen Hydraulik-Hand, die über ein intelligentes Steuermodul von Ján Jockusch angesteuert wird [Joc00].
- Dem System GRAVIS, das mit Hilfe des Kamerakopfes die Szene exploriert, und bei auftretenden Zeigegesten des Benutzers das referenzierte Objekt detektiert.

Zusätzlich sorgt ein weiteres Modul für die Umrechnung zwischen Kamerakopf- und Roboterpositionen und gewährleistet so die Benutzung eines gemeinsamen Koordinatensystems. Alle Teilkomponenten tauschen so über ein gemeinsames Weltkoordinatensystem Information aus. Die Transformation ist nicht fest einprogrammiert sondern wird mit einem adaptiven Lernverfahren der aktuellen Situation angepaßt.

7.4.1 Hand-Auge-Koordination

GRAVIS produziert aufgrund der Disparitätsminimierung während der Szenenexploration Vergenzbewegungen die zu einer Stereo-Fixation der Objekte führen. Der Aufbau des binokularen Kamerakopfes ermöglicht die Berechnung der dreidimensionalen Fixationsposition: sind beide Kameras auf das gleiche Ziel ausgerichtet, ist mit der Vorwärtskinematik die Berechnung der dreidimensionalen Blickkoordinaten über die aktuelle Motorstellung möglich [Kub95]. Die Zusammenhänge der Kinematik sind in der Abbildung 7.7 dargestellt.

Damit sich die Teilkomponenten des Gesamtsystems untereinander verständigen können, ist ein gemeinsames Weltkoordinatensystem erforderlich. Als Ausgangspunkt dieses Koordinatensystems wird die Basis des Roboters gewählt, da dieser fest am Boden montiert ist. Das Ziel besteht darin, die Umrechnung von Kamerakopfkoordinaten in Roboterkoordinaten zu

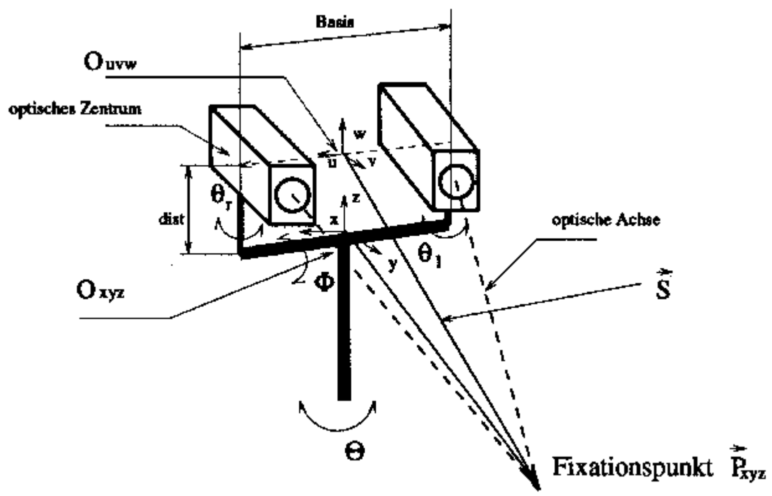


Abbildung 7.7: Dargestellt sind die extrinsischen Freiheitsgrade des Kamerakopfes. Ein Fixationspunkt $\vec{P}$ ist angegeben, in dem sich die beiden optischen Achsen schneiden.

realisieren. Beide Geräte sind nicht immer in der gleichen Relation zueinander aufgebaut. Ján Jockusch hat deshalb einen Algorithmus entwickelt, der die Transformation zwischen den beiden Koordinatensystemen erlernt. Das Verfahren basiert auf einer modifizierten Version des Backprop-Algorithmus und rechnet den dreidimensionalen Kamerakopfvektor in einen Vektor im Weltkoordinatensystem um.

Der Trainingsvorgang des Algorithmus wurde automatisiert, indem der Roboter vorgegebene Raumpunkte abfährt und der Kamerakopf diese Position fixiert. Die korrekte Stereo-Fixation der jeweiligen Position durch den Kamerakopf wird mit einen gelben Würfel erreicht, der sich in der Greifhand des Roboters befindet (siehe Abbildung 7.8). Der Roboter fährt nacheinander Punkte eines $3 \times 3 \times 3$-Gitters ab, das den quadratischen Arbeitsbereich abdeckt. Die vom Kamerakopf angefahrenen Positionen werden aufgezeichnet und als Trainingspunkte für den adaptiven Algorithmus verwendet. Das adaptive Verfahren erreicht bei einmaligen Durchfahren dieser 27 Positionen einen Fehler von circa 2.5 cm Abweichung zwischen Kamera- und Roboterkoordinaten. Die Präzision dieses Verfahrens reicht also aus, um den Greifer grob über der Zielregion zu plazieren. Die Nahannäherung kann dann beispielsweise mit Hilfe der Handkamera und der Greifsensorik durchgeführt werden. Sollte sich die relative Lage zwischen Kamerakopf und Roboter oder andere Einflußfaktoren (Zoomeinstellung etc.) ändern, kann die Abbildung durch erneutes Training an diese Situation jederzeit angepaßt werden. Die Koordinatentransformation realisiert eine komfortable Schnittstelle zwischen GRAVIS und dem

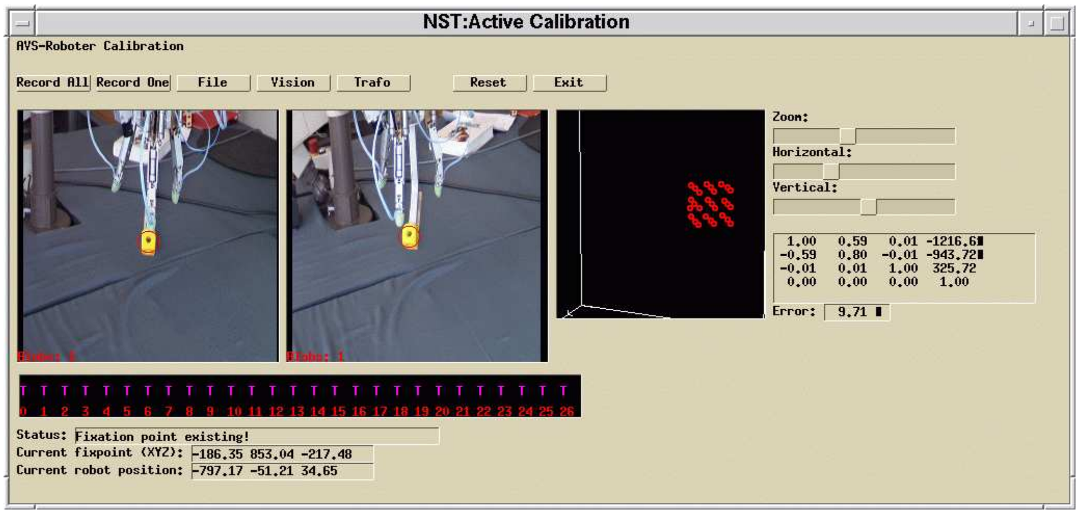


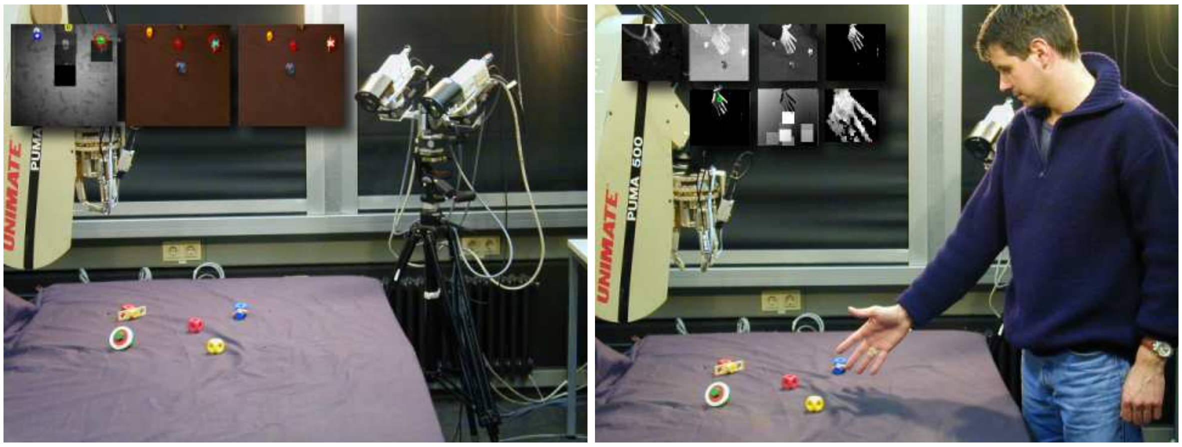
Abbildung 7.8: Die Koordinatentransformation aus der Sicht des Kamerakopfes: Die Stereo-Fixation hat den am Roboterendefektor montierten gelben Würfel erfaßt. Rechts ist die Abbildung der 27 fixierten Kalibrierungsgitterpunkte als transformierte Roboterkoordinaten visualisiert.

Manipulator. Hat der menschliche Benutzer ein Objekt mit einer Handgeste ausgewählt, kann GRAVIS die dreidimensionale Raumposition bereitstellen und per Koordinatentransformation an den Roboter übergeben. Der Roboter ist dann in der Lage die ermittelte Raumposition anzufahren um den Greifvorgang einzuleiten.

7.4.2 Ablauf der Demonstration

Die Ausgangssituation der Demonstration ist die folgende: Der Roboter befindet sich in der Ruheposition und der Kamerakopf startet mit der Szenenexploration durch GRAVIS, während der neuronale Objekterkenner mit einer Overheadkamera die Objekte klassifiziert. Der Kamerakopf, gesteuert durch das Aufmerksamkeitsmodul, beobachtet sukzessive alle in der Szene vorkommenden Bauteile (Abbildung 7.9(a)).

Greift nun ein Mensch in die Szene ein (Abbildung 7.9(b)), fixiert GRAVIS die in der Szene auftauchende Hand. Der Kamerakopf wird auf die Hand ausgerichtet, die sich durch Bewegung und Hautfarbe im Aufmerksamkeitsmodell auszeichnet. Wenn der Benutzer eine Zeigegeste ausführt, wird diese durch das Handerkennungsmodul erkannt, und der schnelle Verfolgungsmechanismus wird gestartet (Abbildung 7.10(a)). Am Ende der Gestentrajektorie wird erneut die Zeigegeste überprüft und ausgewertet (Abbildung 7.10(b)). Anschließend fixiert der Kamerakopf das referenzierte Objekt (Abbildung 7.11(a)).

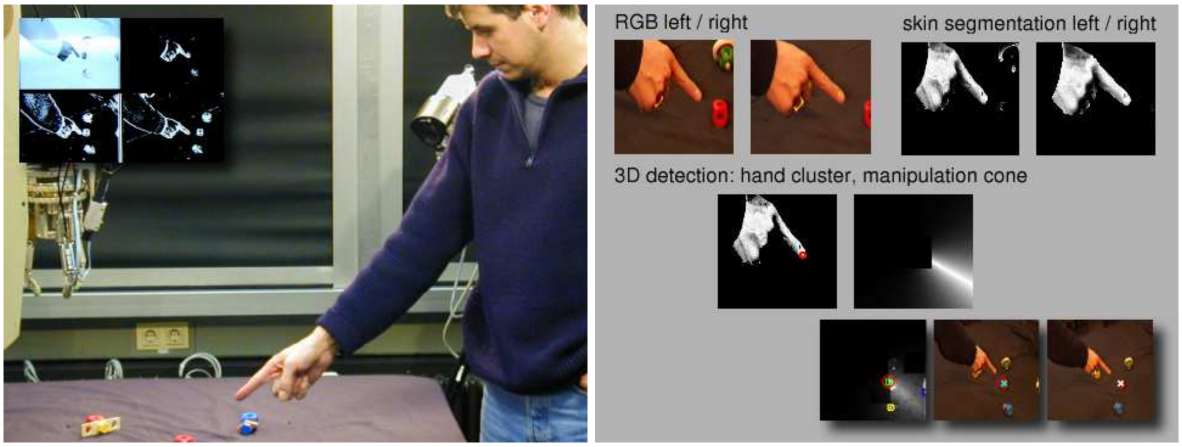


(a) (b)

Abbildung 7.9: (a) Der binokulare Kamerakopf exploriert die Szene. Analog zu Abbildung 5.3 ist links oben die aktuelle Aufmerksamkeitsverteilung eingeblendet. (b) Ein Benutzer greift in die Szene, der Kamerakopf fixiert die Hand. Eingeblendet ist die Zeigegestenerkennung aus Kapitel 5.7. (Abbildungen 7.9 bis 7.12 aus [Joc00])

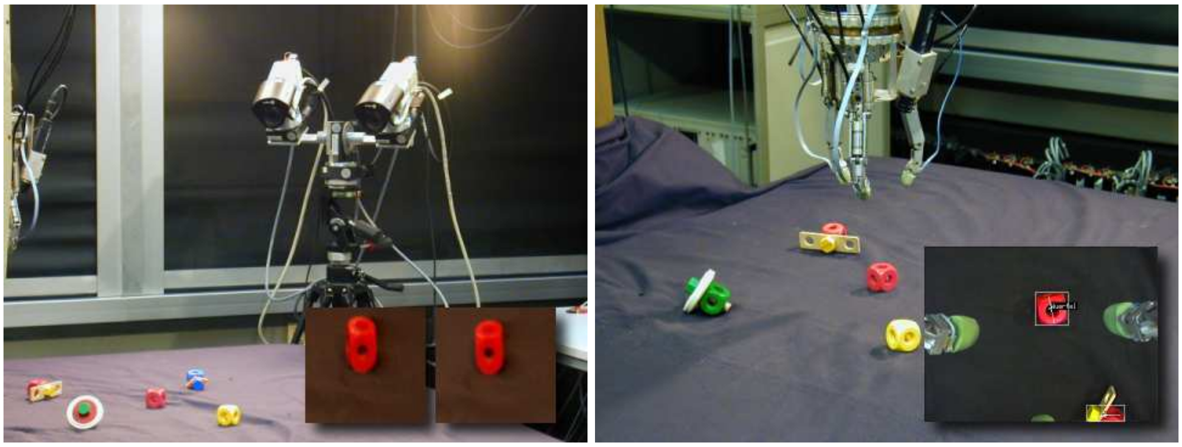
Die Koordinatentransformation rechnet die Kameraposition um und gibt sie an das Robotersystem weiter, und der Roboter fährt die Position direkt an. Da die Fixation in manchen Fällen nicht exakt auf dem Objekt ruht und die Abbildung zwischen den Koordinatensystemen ebenfalls nicht exakt ist, wird die Handkamera verwendet, um den Bereich unterhalb des Manipulators auszuwerten (Abbildung 7.11(b)). Dabei wird das neuronale Objekterkennungssystem von Gunther Heidemann verwendet [Hei98], das Hypothesen über alle vorhandenen Objekte erstellt. Dieses System zeichnet sich dadurch aus, daß es weitgehend automatisiert auf neue Objekte und Umgebungsbedingungen angepaßt werden kann. Deshalb wird es im Demonstrator in zwei Ausprägungen eingesetzt: Zum einen ist der Erkenner darauf trainiert, durch die Handkamera auftretende Objekte auszuwerten. Zum anderen wird ein zweites System zur Gesamtszenenerkennung über eine Overhead-Kamera eingesetzt. Mit Hilfe dieser Kamera werden alle in der Szene vorkommenden Bauteile und Bauteilaggregate ermittelt (zu weiteren Einzelheiten siehe [Hei98]).

Das dem Bildmittelpunkt am nächsten liegende Objekt wird als das referenzierte Objekt angenommen, und die Position sowie die Objekthypothese dem Robotersystem mitgeteilt. Anschließend wird eine Korrekturbewegung vom Roboter ausgeführt, um die Hydraulikhand direkt über dem Objekt zu positionieren. Die dann folgende Annäherungsphase wird kraftgeregelt durchgeführt und der Roboter nähert sich vorsichtig dem Objekt. Ist das Objekt im Greifabstand, wird ein kraftgeregelter Greifvorgang durchgeführt. Dieser erfolgt ohne eine



(a) (b)

Abbildung 7.10: (a) Eine Zeigegeste wurde erkannt, und GRAVIS schaltet um in den Verfolgungsmodus. Eingeblendet ist die Ausgabe des ZDF-Algorithmus. Am Ende der Gestentrajektorie wird die Zeigegeste ausgewertet. In Abbildung (b) sind die wichtigsten Stufen der Erkennung dargestellt: Die segmentierte Hand, die Richtungserkennung sowie der generierte Aufmerksamkeitskegel, der direkt die Aufmerksamkeitsverteilung moduliert (untere Reihe).

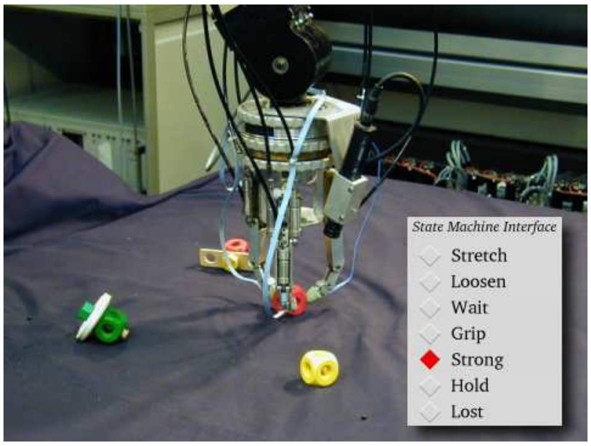


(a) (b)

Abbildung 7.11: (a) Das referenzierte Objekt wird vom Kamerakopf fixiert. Eingeblendet ist die fovealisierte Bildregion mit dem Objekt. (b) Der Roboter positioniert sich über dem Objekt. Eingeblendet ist die Objekterkennung über die Handkamera.

Positionsregelung, die Finger nähern sich aus der gespreizten Postur dem Objekt. Treffen die Finger auf Widerstand, so wird die Druckkraft erhöht. Da das Objekt typischerweise von den drei Fingern umschlossen wird, ergibt sich so ein kräftiger Griff. Die Steuerung des Greifvorgangs erfolgt adaptiv und wird über eine Greif-Zustandsmaschine kontrolliert (Abbildung 7.12(a)). Um den einzelnen Fingern ein Kraftgefühl zu verleihen, wurden von Ján Jockusch spezielle Kraftsensoren für die Fingerkuppen entwickelt (zu Einzelheiten siehe [Joc00]).

Nachdem der Roboter das Objekt ergriffen hat, präsentiert er das Bauteil dem Benutzer (Abbildung 7.12(b)). Anschließend wird es wieder in der Szene abgelegt und der Kamerakopf beginnt erneut mit der Exploration der neuen Szene. Das System ist nun wieder bereit, neue Zeigegesten zu erkennen.



(a)

(b)

Abbildung 7.12: (a) Die Hand hat das Objekt kraftgeregelt gegriffen. Dargestellt sind die Zustände der Greif-Zustandsmaschine. (b) Ein Systemzyklus ist zu Ende, der Roboter legt das Objekt ab und die Exploration der neuen Szene beginnt.

In das beschriebene System zur Objektreferenzierung fließt bisher ausschließlich Information aus der visuellen Objekterkennung, der Zeigegestenerkennung sowie der Greifsensorik ein. Sprachverstehen ist im aktuellen Demonstrationssystem bisher noch nicht integriert. Erst die Verknüpfung von Sprache und Gestik ist aber die Grundlage von intelligenter Kommunikation. In [MMRR99] ist ein kognitionswissenschaftlicher Ansatz zur Integration dieser beiden Aspekte in einem Agentensystem zur natürlichsprachlichen Robotersteuerung beschrieben (näheres siehe Kapitel 9). Ein weiterer Schritt der Integration ist die Verwendung der fovealisierten Bilder während der Exploration für eine detailliertere Objekterkennung. Der neuronale Objekterkenner bietet dafür durch eine weitere Spezialisierung einen vielversprechenden Ansatz. Aufgrund der offenen Architektur von GRAVIS sind unterschiedliche Erweiterungsmöglichkeiten denkbar. Insbesondere die Integration des Robotersystems erfordert weitere Fähigkeiten in Bezug auf Verfolgung des Greifvorgangs und die Erkennung und Interpretation der Objekte.

8 Diskussion

AKTIVE Sehsysteme bilden die Basis für die in dieser Arbeit vorgestellten intelligenten visuellen Mensch-Maschine-Schnittstellen. Eine weitere Gemeinsamkeit bilden die Aspekte „Aufmerksamkeit“ und „Adaptivität“. Die vorgestellten Systeme GAZE und GRAVIS nutzen die Eigenschaften natürlichen visuellen Verhaltens für die robuste und zuverlässige Erkennung menschlicher Gestik. Insbesondere die Modellierung von Aufmerksamkeit orientiert sich an biologischen Vorbildern. Gemeinsam mit den eingesetzten neuronalen Netzen wird so ein anpassungsfähiges und robustes Systemverhalten erzielt. Das menschliche Blickverhalten ist sehr komplex, jedoch lassen sich in Eyetracking-Experimenten einige typische Grundeigenschaften beobachten. Dazu zählt unter anderem die bevorzugte Fixation von interessanten Bildregionen, die sich etwa durch besondere Signalfarben oder starke Kontraste auszeichnen. Neben diesem eher reflexartigen Verhalten, werden Augenbewegungen aber auch durch den jeweiligen Kontext beeinflußt. Dabei fließt Vorwissen und eine gewisse Erwartung in die Auswahl des nächsten Fixationspunktes. In dieser Arbeit wurden beide Ansätze in technischen Bildverarbeitungssystemen umgesetzt.

Das System GAZE basiert zunächst auf einem adaptiven Ansatz zur Detektion des Gesichts. Ausgehend von der Hautfarbensegmentierung mittels Farbhistogrammen werden im ersten Schritt hautfarbene Bildbereiche ermittelt. Um auf Beleuchtungsänderungen und unterschiedliche Hautfarbentypen reagieren zu können, ist das System in der Lage sich automatisch daran anzupassen. Die Grundlage dafür bildet eine allgemeine Farbkarte, die typische Hautfarben repräsentiert und verhindert, daß sich das System an andere Farben anpaßt. Das System ermittelt zunächst nur grob die Gesichtsregion, nach wenigen Anpassungsschritten ist es aber auf die aktuelle Situation spezialisiert und das Gesicht tritt klar hervor. Die Farbinformation wird kombiniert mit der Segmentierung von dynamischen Bildbereichen durch Differenzbilder. Erfahrungsgemäß ist der Kopf des Benutzers immer leicht in Bewegung und kann somit noch zuverlässiger detektiert werden. Bei großflächigeren Bewegungen der Person hält ein in der Kamera eingebauter Trackingmechanismus das Gesicht im Blickfeld.

Der so detektierte Gesichtsbereich wird anschließend mit passiven Fixationen auf relevante Regionen genauer analysiert. Ähnlich wie der Mensch ein Bild von einem Gesicht betrachtet, versucht das künstliche System nun typische Gesichtsmerkmale zu detektieren. Zunächst wird die Nasenposition gesucht, da die Nase aus der Kameraperspektive am besten zu erkennen ist. Dafür wird ein heuristischer Ansatz gewählt, der die Nasenlöcher im Rot-Kanal herausfiltert. Die Trefferquote dieses Ansatzes liegt bei einem Versuch mit 670 Bildern bei ca. 98%. Um die Nasenposition zu überprüfen, findet eine effiziente Verifizierung über einen Formtest statt, der lediglich in 0.5% der Fälle falsch klassifiziert. Wenn keine Nase im Bild ist, wird über einen dreistufigen LLM-Ansatz die Aufmerksamkeit des Systems in Richtung der vermuteten Nasenregion gelenkt. Der dafür verwendete Ansatz zur Merkmalsextraktion mit „Eigennasen“ führt zu einer äußerst robusten Detektion der Nasenposition. Als Trainingsmaterial dienen 296 Bilder von insgesamt neun Personen bei fünf verschiedenen Beleuchtungen. Die durchschnittliche Abweichung der detektierten Nasenposition liegt schließlich bei ungefähr zwei Pixeln in x- und y-Richtung.

Als nächstes werden die Mundwinkel mit Hilfe von „End-Stopp“-Filtern im inversen Grün-Kanal gesucht. Die nächste passive Fixation wird beschränkt auf die Region unterhalb der Nase. Dadurch wird die Bestimmung der Mundwinkelpositionen vereinfacht und zusätzlich beschleunigt. Probleme treten jedoch auf, wenn sich der Mund zu sehr verformt (etwa beim Sprechen).

Mit den ermittelten Positionen von Nase und Mund wird auch der Suchbereich der Augenposition drastisch eingeschränkt. Aufgrund der Nase-Mund-Konstellation werden die Augenmittelpunkte zunächst grob geschätzt. Ausgehend von dieser Positionsschätzung werden wiederum zwei LLM-Netze hintereinandergeschaltet, um die Augenposition korrekt zu ermitteln. Der für die Anpassung der LLMs verwendete Trainingsbildsatz umfaßt 378 Bilder. Die Abweichung von der echten Augenposition liegt mit diesem Ansatz bei ca. zwei Pixeln. Insgesamt funktioniert die Detektion der Gesichtsmerkmale unabhängig vom Benutzer und der eingesetzten Beleuchtung. Auch die Augenposition wird mit einem spezialisierten LLM verifiziert. Die Fehlerrate der Überprüfung setzt sich aus fälschlicherweise akzeptierten bzw. zu unrecht abgelehnten Bildregionen zusammen und beträgt ca. 10%. Da das System nicht holistisch den kompletten Kopf auswertet (wie etwa bei [RR98b]), sondern passive Fixationen relevanter Gesichtsregionen verwendet, erkennt es auch bei Brillenträgern und teilüberdeckten Gesichtern sehr zuverlässig die korrekte Augenposition.

Die Modellierung dieser passiven Bildfixationen zur Erkennung von Gesichtsmerkmalen wird genutzt, um die Blickrichtungserkennung durchzuführen. Damit kann die generelle Einsatzmöglichkeit des Systems evaluiert werden: Die Voraussetzung zur Erkennung von Augenbewegungen ist die korrekte Identifikation der Augenposition. Es zeigt sich, daß aus der relativen 2D-Lage der Augen zur Nase die Kopfrichtung sehr genau berechnet werden kann. Die eigentliche Augenstellung wird von einem neuronalen Netz klassifiziert, das als Merkmale die Skalarprodukte von der Augenregion mit speziell berechneten Eigenaugen erhält. Jedoch muß dieses Netz auf neue Benutzer, Beleuchtungsbedingungen und Kameraparameter angepaßt werden. Deshalb sind die Beleuchtung und die Kameraparameter entsprechend eingeschränkt worden. Ein gut auf die aktuelle Situation adaptiertes Netz ermittelt die Blickrichtung des Benutzers mit einer Abweichung von ca. 65 Pixeln in x- und 110 Pixeln in y-Richtung auf einem Monitor mit einer Auflösung von 1280×1000 Pixeln (der Sitzabstand beträgt ungefähr 60cm). Bei einem unbekannten Benutzer, der nicht im Trainingssatz berücksichtigt wurde, steigt der Fehler jedoch an (267 Pixel in x- bzw. 178 Pixel in y-Richtung). Diese Schwäche des Systems kann durch ein schnelles Verfahren zum Nachtrainieren ausgeglichen werden. Dabei wird das System GAZE mit gezielten Fixationen des Benutzers auf einem vorgegebenen 3×3 Gitter an die neue Situation angepaßt. Dadurch wird der Fehler mit durchschnittlich 208 Pixel in x- bzw. 171 Pixel in y-Richtung signifikant reduziert.

Bis zum Stadium der eigentlichen Blickrichtungserkennung ist das System GAZE in einem weiten Rahmen benutzer- und beleuchtungsunabhängig. Das Schema zur passiven Verschiebung der Aufmerksamkeit innerhalb des Kamerabildes auf relevante Gesichtsbereiche wie Nase, Mund und Augen arbeitet zuverlässig und stellt eine robuste Grundlage für die Blickrichtungserkennung dar. Das Netz zur endgültigen Blickrichtungsbestimmung kann problemlos nachtrainiert werden, um das System an neue Situationen anzupassen. Die Mechanismen Adaptivität und Aufmerksamkeit ermöglichen somit die zuverlässige Erkennung von menschlichen Augenbewegungen.

Aufmerksamkeit ist bislang noch nicht so allgemein modellierbar, daß die Systeme domänenunabhängig einsetzbar sind. Deshalb müssen spezielle Lösungen für spezielle Probleme entwickelt werden. In dieser Arbeit wird für zwei relevante Domänen gezeigt, daß die selektive Aufmerksamkeit zur visuellen Steuerung von Bildverarbeitungsprozessen erfolgreich eingesetzt werden kann und eine geeignete Grundlage für die effiziente und robuste Erkennung von Gestik darstellt.

Im System GAZE wird die Aufmerksamkeit des Systems durch vortrainiertes und modelliertes Wissen beeinflußt. Der hierarchische Ansatz der LLM-Netze zur Verbesserung der Merkmalsposition ist auf bestimmte Gesichtsmerkmale spezialisiert. Darüber hinaus wird eine gewisse Vorauswahl erfolgversprechender Gesichtsregionen getroffen, indem etwa die Annahme, daß die Augen oberhalb der Nase und die Augenmittelpunkte etwa auf einer Linie mit den Mundwinkeln liegen, zur Einschränkung der Suche verwendet wird. Bei typischen Gesichtern ist diese Annahme erfüllt, denn die Anordnung von Augen, Nase und Mund ist relativ ähnlich. Bei einem System, das nur auf der Domäne „Gesicht“ führt dieser Ansatz wie beim System GAZE zur robusten Erkennung der Merkmalspositionen. Anders ist die Situation bei der Erkennung von Handgesten. Diese finden in einer viel höheren Variabilität statt. So kann die Hand nahezu beliebig viele Posturen annehmen, so daß sich das Aussehen signifikant ändert. Auch die Hände verschiedener Benutzer unterscheiden sich stark in Größe und Hautfarbe.

Hinzu kommt, daß es sich bei dynamischen Handgesten um großflächige Bewegungen handelt, die mit einem statischen Kamerasystem nur sehr schwer zu erfassen sind. Da bei Zeigegesten zusätzlich auch 3D-Information eine Rolle spielt, muß zusätzlich mit einem Stereoansatz gearbeitet werden. Deshalb wurde im System GRAVIS ein anderer Ansatz zur Modellierung von Aufmerksamkeit gewählt, der neben der Erkennung von Handgesten auch die Exploration von Bildszenen ermöglicht.

In diesem System kommt ein aktives Stereo-Sehsystem zum Einsatz, das auf einer Aufmerksamkeitssteuerung basiert, und die gestikgestützte visuelle Mensch-Maschine-Kommunikation erlaubt. Dafür wird ein aktiver Stereo-Kamerakopf eingesetzt, der mittels eines adaptiven mehrschichtigen Verarbeitungsmodells auf interessante Bildbereiche gesteuert wird. Jede Schicht des Modells repräsentiert unterschiedliche Bildeigenschaften (Kanten, Sättigung, Intensität, Bewegung, usw.) in einer topografischen Karte. Die Überlagerung dieser Karten resultiert in einer *kortikalen Karte*, deren Maximum den interessantesten Bildbereich markiert. Dieser Punkt wird auf das nächstliegende Objekt korrigiert, so daß die Kameras den entsprechenden Gegenstand binokular fixieren. Durch die Ausblendung bereits besuchter Regionen entwickelt sich ein dynamisches Explorationsverhalten, das gleichzeitig auf bestimmte Ereignisse wie bewegte Objekte, Personen oder Handgesten reagiert.

Aufbauend auf diesem Aufmerksamkeitsmodul wurde eine weitere Systemkomponente entwickelt, die permanent die fixierte Bildregion auswertet und Zeigegesten erkennt. Beim Auftreten einer Zeigegeste kommt das dritte Modul zur Verfolgung von Händen zum Einsatz. Dieses Verfolgungsmodul minimiert die Disparität des verfolgten Objektes binokular und hält es im Zentrum beider Kameras. Die Verfolgung wird mit einer hohen Bildfrequenz durchgeführt und verliert auch schnelle Handbewegungen nicht aus dem Fokus. Verharrt die Hand im Sichtbereich der Kameras, wird die Geste erneut ausgewertet und die Zeigerichtung in einer höheren Bildauflösung bestimmt. Die Zeigerichtung modifiziert anschließend die Aufmerksamkeitsverteilung in der kortikalen Karte. Dadurch wird die Blickrichtung des künstlichen Sehsystems manipuliert, so daß dieses die Bildregion ansteuert und somit das referenzierte Objekt fixiert. Da ein Stereo-Kamerakopf eingesetzt wird und binokulare Bildinformationen ausgewertet werden, ist anschließend aufgrund der Kamerageometrie auch die 3D Objektlage bekannt.

Im Gegensatz zum System GAZE ist der Ansatz zur Aufmerksamkeitssteuerung in GRAVIS rein datengetrieben. Die Integration von Vorwissen, etwa zur Detektion von Händen (Bewegung und Hautfarbe) fließt direkt in die Aufmerksamkeitskarte ein. Dadurch werden bevorzugt Regionen fixiert, in denen Hände auftauchen. Diese Fixation kann aber auch auf einer anderen Bildregion landen, wenn dort ein anderes wichtigeres Merkmal vorliegt und somit die Aufmerksamkeitsverteilung an diesem Ort höher ist. Die Fixationen sind also nicht immer zielgerichtet, vielmehr versucht das System situationsbezogen zu reagieren. Die Integration von

Weltwissen etwa über Hände muß also in den Merkmalskarten repräsentiert werden und bedarf zusätzlicher Klassifikation durch ein weiteres Modul. Deshalb arbeitet das System GRAVIS auch mit unterschiedlichen Systemzuständen: Wird eine Hand über Bewegung und Hautfarbe fixiert, findet eine Handposturerkennung statt. Liegt in diesem Moment eine Zeigegeste vor, wird der Aufmerksamkeitsmechanismus vorübergehend abgeschaltet, und der Verfolgungsmechanismus übernimmt die Steuerung des Kamerakopfes. Nur so kann eine zielorientierte Kamerabewegung realisiert und eine Erkennung von Zeigegesten in den Aufmerksamkeitszyklus integriert werden, ohne daß das System Metawissen über typische Handlungsabläufe besitzt. Das ist beim System GAZE deutlich anders. Das System arbeitet nach einem festen Erkennungsschema und ist darauf trainiert, die Gesichtsmerkmale nach einem bestimmten Gesichtsmodell zu detektieren.

Beide Ansätze haben ihre Stärken und Schwächen: GAZE arbeitet sehr zuverlässig, und klassifiziert permanent und schnell die Augenrichtung des jeweiligen Benutzers. Das System richtet seine Aufmerksamkeit nur auf die speziellen Gesichtsmerkmale und extrahiert robust die Augenrichtung. Die Systemarchitektur ist in sich abgeschlossen, und die Auswertung zusätzlicher Merkmale ist nur schwer zu integrieren. Die Anpassung des Systems an neue Situationen, etwa zur Erkennung von Handgesten ist noch aufwendiger, da eine große Anzahl an neuronalen Netzen neu trainiert werden muß. Dies betrifft insbesondere die spezielle Merkmalsextraktion mit den Eigenvektoren. Das System GRAVIS hingegen basiert auf einer rein datengetriebenen Aufmerksamkeitssteuerung. Die Klassifikation arbeitet davon unabhängig und wird situationsbezogen zugeschaltet. Dadurch ist die Systemarchitektur beliebig erweiterbar, und eine Anpassung auf neue Aufgaben ist ebenfalls möglich. Zum einen können die Merkmals- bzw. Manipulationskarten an andere Situationen angepaßt werden, etwa um Sprachinformationen zu berücksichtigen, zum anderen können weitere Module zur Erkennung von neuen Objekten problemlos integriert werden. Es kann jedoch nicht garantiert werden, daß das System einem festen Handlungsablauf folgt. Ist zum Beispiel während der Phase der Exploration eine Bildregion interessanter als eine zufällig im Bild auftauchende Hand, so wird auch die Handgestenerkennung nicht sofort aktiviert. Der Erfolg der Klassifikation hängt also stark von dem implementierten Aufmerksamkeitsalgorithmus ab.

Beide Ansätze sind in ihrer Domäne erfolgreich und besitzen Stärken, die sie von anderen Ansätzen abheben. Die Gemeinsamkeiten beider Systeme liegen in der Verwendung von aktiven Kamerasystemen und der Modellierung von Aufmerksamkeit zu deren Ansteuerung. Neuronale Netze bilden sowohl in GAZE als auch in GRAVIS die entscheidenden Klassifikatoren zur Erkennung von Gestik. Die zugrundeliegende Adaptivität erlaubt daher eine Anpassung beider Systeme an neue Situationen und Benutzer. Das sind wichtige Voraussetzungen für die Realisierung intelligenter Mensch-Maschine-Schnittstellen und bilden die Basis für die Entwicklung zukünftiger kommerzieller Systeme. Die hier vorgestellten Ansätze erfüllen dabei hohe Anforderungen an Effizienz, Zuverlässigkeit und Anpassungsfähigkeit. Das Hauptziel, die Modellierung visueller Mechanismen natürlicher Sehsysteme in künstlichen Systemen zur Erkennung von Gestik, wurde erfüllt und die Systeme GAZE und GRAVIS zeigen prototypisch die Eigenschaften zukünftiger Bildverarbeitungssysteme zur Mensch-Maschine-Interaktion.

9 Ausblick

DIE künftige Entwicklung von Systemen zur Mensch-Maschine-Interaktion wird Forschungsergebnisse aus vielen Teilbereichen einbeziehen müssen, um die offenen Probleme befriedigend zu lösen (siehe auch [Wex98]). Durch die Kombination unterschiedlicher Ansätze, etwa aus der Biologie, der Sprachwissenschaft und der Informatik kann das Ziel intelligenter und robuster Systeme oftmals schneller erreicht werden. Schon jetzt sind viele Teilprobleme gelöst, und es fehlt nur die geschickte Integration dieser Funktionalität. Die vorliegende Arbeit hat zwei konkrete technische Systeme gezeigt, die beide auf der Grundlage von Adaptivität und Aufmerksamkeit die Erkennung von menschlichen Blick- beziehungsweise Zeigegesten ermöglichen. Insbesondere das System GRAVIS bietet ein offenes Rahmenwerk für zukünftige Erweiterungen, die die Möglichkeiten des Systems wesentlich ausbauen sollen. So ist der Einbau weiterer Modalitäten wie etwa Sprache in das Aufmerksamkeitsmodell möglich, indem weitere Manipulationskarten hinzugefügt werden (siehe dazu [MMRR99]). Gleichzeitig können aber auch weitere Module neben das Aufmerksamkeitsmodell gestellt werden. Diese Module können zusätzliche Funktionalität bereitstellen, um zum Beispiel eine verbesserte Objekterkennung zu realisieren. Die Ablaufsteuerung der einzelnen Module erfolgt aber weiterhin über eine zentrale Zustandsmaschine.

Rechenintensive Verfahren können auch direkt auf den von GRAVIS fixierten Bildregionen arbeiten, wodurch die Verarbeitung auf interessante Bildbereiche eingeschränkt wird. Beispielsweise das System zur „Perzeptiven Gruppierung“, das von Daniel Schlüter und Stefan Posch im Sonderforschungsbereich entwickelt wurde [SP98], kann so beschleunigt werden. Der Ansatz kombiniert Konturapproximationen und konturbasierte Gruppierungen mit segmentierten Regionen zur Verbesserung der Bildinterpretation. Erfolgt die Gruppierung auf dem gesamten Bild, dauert die Verarbeitung aufgrund der hohen Datenmenge relativ lange. Werden hingegen die fixierten Bildregionen sequentiell ausgewertet, stehen innerhalb kurzer Zeit erste Gruppierungshypothesen zur Verfügung.

Darüber hinaus ist auch die Kombination von GAZE und GRAVIS zu erzielen: Der Kamerakopf beobachtet die Person und reagiert erst aufgrund von sprachlichen Anweisungen wie „Nimm den gelben Würfel!“. Während der Äußerung läuft das System GAZE und wertet permanent die Blickrichtung aus. Diese wird dann genutzt, um die darauf folgende Handgestenerkennung in GRAVIS robuster zu gestalten. In die Erkennung soll darüber hinaus die erkannte Blickrichtung mit einbezogen werden. Das ist zum einen über eine zusätzliche Aufmerksamkeitskarte oder eine spezielle Manipulationskarte möglich. Auch eine andere Integration von GAZE und GRAVIS ist denkbar: Der Benutzer betrachtet nicht mehr die Originalszene, sondern ein Abbild der Szene auf einem Monitor. Das Monitorbild stellt die Szene aus der Perspektive des Kamerakopfes dar. Da auf dem Monitor die Blickbewegungen mittels einer weiteren Kamera zuverlässig vorhergesagt werden, können die von GAZE ermittelten Blickregionen direkt als Merkmalskarte in das Aufmerksamkeitsmodell von GRAVIS einfließen. Der Kamerakopf wird also bevorzugt Bildbereiche betrachten, die auch der Mensch inspiziert. Damit ist eine direkte Kopplung der Aufmerksamkeit zwischen dem Kamerakopf und dem Benutzer hergestellt. Der Vorteil dieser Methode liegt in der leichteren Blickrichtungserkennung, die mit einem beweglichen Kamerakopf und einem davor stehenden Benutzer schwierig zu realisieren ist. Der Nachteil besteht darin, daß der Benutzer nur einen begrenzten Ausschnitt (Kameraperspektive) der Szene zu sehen bekommt und auch die Handgestenerkennung keine Anwendung findet[1]. Der Ansatz ist aber in typischen Anwendungsfällen der Telerobotik etwa zur Ferndiagnose von Maschinen durchaus vielversprechend.

Verbesserungen einzelner Systemkomponenten von GAZE und GRAVIS sollen die Zuverlässigkeit und Robustheit der Systeme noch weiter erhöhen. So kann die Blickrichtungserkennung von GAZE verbessert werden, indem Ergebnisse aus den vorherigen Bildern berücksichtigt werden. Dafür ist aber eine hohe Verarbeitungsgeschwindigkeit erforderlich, die in Zukunft durch schnellere Rechner auch sicherlich erreicht wird. Dann können die im vorigen Bild ermittelten Merkmalspositionen als initiale Ausgangsposition zur Detektion im neuen Bild dienen. Dadurch wird das passive Tracking des Gesichts überflüssig und die Zuverlässigkeit des Systems deutlich ansteigen. Die Implementierung des schnellen Blickerkennungssystems, ohne die Detektion der Gesichtsmerkmale, zeigt, daß durch die Filterung der Blickpositionen die Ergebnisse weiter verbessert werden. Die Filterung dient der Eliminierung von „Ausreißern“, wie sie etwa beim Augenblinzeln vorkommen.

Eine weitere Verbesserung könnte eine Entfernungsdetektion bringen. Das System sollte erkennen, ob der Benutzer nah am Monitor sitzt oder sich weiter entfernt aufhält. Dann müssen die Blickpositionen genauer oder weniger genau ermittelt werden. Deshalb könnten für unterschiedliche Entfernungen spezialisierte Netze zum Einsatz kommen. Darüber hinaus ist auch eine Erweiterung hin zu einer echten dreidimensionalen Erkennung der Blickrichtung sinnvoll. Dafür ist (i) die Erkennung von beiden Augenstellungen und (ii) eine bessere Bestimmung der Kopfhaltung nötig. Der zweite Punkt kann etwa über eine Schätzung des Gesichtsmodells auf der Grundlage der detektierten Gesichtsmerkmale erreicht werden. Dies könnte wie bei

[1] Zumindest nicht die im System GRAVIS implementierte.

Ballard und Stockmann [BS95] über eine Messung der 3D-Abstände der Gesichtsmerkmale geschehen.

Mit dem System GRAVIS ist es aktuell möglich, 2D Zeigerichtungen zu bestimmen (Kippwinkel der Hand), was für eine grobe Referenzierung der Objekte ausreicht. Besser wäre eine echte dreidimensionale Auswertung der Gestik, die auch Drehbewegungen der Hand erkennt. Dafür müßten in Zukunft beide Kamerabilder ausgewertet werden. Ansätze zu einer Verbesserung der zugrundeliegenden Handgestenerkennung durch die Detektion von Fingerspitzen und der Fusion der Daten in einem adaptiven Modell werden von Claudia Nölker untersucht [NR98].

Auch an der Integration von Szenenwissen („Wo liegen Objekte?“) und Zeigerichtung sollen Verbesserungen vorgenommen werden. Zum einen kann z.B. Tiefeninformation, die während der Exploration akquiriert wird, ausgenutzt werden und zum anderen benötigt man besonders in dynamischen Szenen ein Langzeitgedächtnis. Hierzu werden bereits Ansätze untersucht (etwa [JvdHRS98]), die ein assoziatives Gedächtnis modellieren. Damit kann das gesammelte Objektwissen über einen längeren Zeitraum gesammelt und effektiver abgefragt werden. Einen anderen Ansatz, der ebenfalls eine Repräsentation der dreidimensionalen Szene modelliert, stellt die Berechnung von Mosaikbildern dar (siehe [IAB+96, IAB+96, SS97]). Dabei werden die Einzelansichten einer Kamera in ein planares Mosaikbild transformiert, das anschließend ein Panorama der Gesamtszene enthält. Das „verkleben“ der Bilder geschieht typischerweise über projektive Transformationen. Die dreidimensionale Umgebung wird also in ein planares Koordinatensystem abgebildet. Die Information in dem Mosaikbild kann anschließend zur Steuerung des aktiven Kamerakopfes genutzt werden. Dadurch werden beispielsweise „weiße Flecken“ in der Panoramakarte gefüllt oder herangezoomte Details integriert.

Das Ziel dieser Arbeit, eine offene Architektur für die Erforschung von Mensch-Maschine-Kommunikation zu entwickeln, wurde mit GRAVIS erreicht. Die Exploration durch das Aufmerksamkeitsmodell kann zum einen die Objekterkennung durch Detailansichten verbessern, und zum anderen die Erstellung eines Weltmodells über die Speicherung der Einzelansichten in einer Art Gedächtnis ermöglichen. Die zugrundeliegende Modellierung von Aufmerksamkeit sorgt dafür, daß in allen darauf aufbauenden Prozessen nur signifikante Bildregionen ausgewertet werden müssen. Die Anpassungsfähigkeit des Systems erweitert zusätzlich das Einsatzspektrum unter realen Bedingungen.

Insgesamt hat sich gezeigt, daß der Einsatz aktiver Sehsysteme in gestikbasierten Mensch-Maschine-Schnittstellen sinnvoll ist, und die Verarbeitung durch eine starke Einschränkung der Bandbreite der Bildinformation auf wesentliche Bildregionen erheblich effizienter und flexibler wird. Die so gewonnene Effizienz ermöglicht eine Anpassung an menschliches Kommunikationsverhalten und stellt einen Schritt in Richtung ergonomischer Interaktion zwischen

Mensch und Computer dar. Bis zur tatsächlichen Produktreife dieser Systeme ist es aber noch ein weiter Weg, und es bestehen noch eine Menge offene Fragen, die es zu klären gilt. Die in dieser Arbeit vorgestellten Systeme zur Mensch-Maschine-Interaktion auf der Grundlage von „Adaptivität“ und „Aufmerksamkeit“ sind ein erfolgversprechender Schritt auf diesem Weg.

10 Zusammenfassung

IM Mittelpunkt dieser Arbeit steht die Erforschung von intelligenten gestikbasierten Mensch-Maschine-Schnittstellen auf der Basis von visueller Aufmerksamkeit und Adaptivität. Aufbauend auf den Erkenntnissen über das visuelle Verhalten des Menschen werden zwei konkrete technische Systeme GAZE und GRAVIS implementiert und untersucht.

Das System GAZE realisiert die Erkennung von Blickbewegungen auf der Grundlage einer modellgestützten Aufmerksamkeitssteuerung. GAZE verwendet einen hierarchischen Erkennungsansatz, der mit Hilfe von hintereinander geschalteten LLM-Netzen typische Gesichtsmerkmale detektiert. Dafür wird zunächst mit einem adaptiven Hautfarbenklassifikator die grobe Segmentierung der Gesichtsregion durchgeführt. Ausgehend von heuristisch geschätzten Positionen wird innerhalb dieser Gesichtsregion nach den Gesichtsmerkmalen Nase, Mund und Augen gesucht. Dabei kommen spezialisierte neuronale Netze zum Einsatz, die besondere Eigenvektoren („Eigennasen" und „Eigenaugen") als Merkmale verwenden und auf die Detektion der zugehörigen Gesichtsregion trainiert sind. Mit diesem Ansatz wird die Augenregion zuverlässig detektiert, die in einer anschließenden Auswertung zur Erkennung der aktuellen Blickposition dient. Diese wird durch ein spezialisiertes neuronales Netz mit einer Genauigkeit von 1.5^o in der vertikalen bzw. 2.5^o in der horizontalen Blickrichtung und einer Bildrate von ca. einem Bild pro Sekunde berechnet.

Das System GRAVIS ist auf die Erkennung von Zeigegesten zur Greifsteuerung eines Roboters in einem Konstruktionsszenario ausgerichtet. Die Grundlage dafür bildet ein aktiver Stereokamerakopf, dessen „Aufmerksamkeitsfokus" durch zwei komplementäre Aufmerksamkeitsmechanismen gesteuert wird. Der eine Mechanismus steuert rein datengetrieben die Ausführung von Sakkaden und berechnet auf der Basis von unterschiedlichen topografischen Merkmalskarten ein „Interessantheitsmaß". Bei dem anderen Mechanismus handelt es sich um einen disparitätsbasierten Tiefenfilter, mit dem die Verfolgung von Objekten in Echtzeit möglich ist. Eine weitere Systemkomponente bildet das neuronale Handgestenerkennungsmodul, das Handstellungen des Benutzers erkennt und Zeigegesten auswertet. Die so gewonnene Information wird in das Sakkadenmodell zurückgeführt, um den nächsten Fixationspunkt zu bestimmen. Das System ermöglicht die gestikbasierte Mensch-Maschine-Kommunikation, bei der das Kamerasystem autonom die Umgebung exploriert und bei auftretenden Zeigegesten des menschlichen Partners seine Aufmerksamkeit auf das referenzierte Objekt lenkt. Die vorgestellten Resultate demonstrieren die Robustheit und Zuverlässigkeit des künstlichen adaptiven Sehsystems bei der Kooperation mit unterschiedlichen Benutzern.

Dem Ziel der Realisierung intelligenter Mensch-Maschine-Schnittstellen mit aktiven Kamerasystemen kommt diese Arbeit mit den Systemen GAZE und GRAVIS ein Stück näher:

- Der im System GAZE eingesetzte adaptive Hautfarbenklassifikator paßt sich automatisch an neue Benutzer und Beleuchtungsbedingungen an.
- Der modellgestützte Aufmerksamkeitsansatz in GAZE detektiert mit Hilfe einer hierarchischen Sequenz von neuronalen Netzen typische Gesichtsmerkmale, die die Grundlage der anschließenden Blickrichtungserkennung bilden.
- Eine Besonderheit des Systems GRAVIS ist die offene Architektur auf Basis einer Zustandsmaschine, die um weitere spezialisierte Module erweitert werden kann.
- Die Nulldisparitätsfilterung basiert auf einem simplen, aber robusten Prinzip und erlaubt die zuverlässige Verfolgung beliebiger Objekte in Echtzeit.
- Das datengetriebene Aufmerksamkeitsmodell im System GRAVIS verwendet eine Reihe von Merkmalskarten zur Berechnung interessanter Fixationspunkte für das künstliche Sehsystem. Die Sakkaden dienen zur effizienten Exploration der in der Szene vorhandenen Objekte.
- Durch eine Manipulationskarte ist die Aufmerksamkeit des Systems direkt durch Zeigegesten zu beeinflussen. GRAVIS ist damit in der Lage das referenzierte Objekt zu fixieren und ermöglicht so beispielsweise die Steuerung eines Greifvorgangs durch einen Roboter.

Dennoch sind viele weitere Forschungsarbeiten nötig, um das spannende und herausfordernde Thema der intuitiven Bedienbarkeit von Maschinen über Gesten befriedigend zu lösen. Durch die rasante Entwicklung der Hardware werden die aufwendigen Verfahren sicherlich wesentlich beschleunigt. Damit erschließen sich Möglichkeiten zur Fusion weiterer Information und neuer Modalitäten. Trotzdem ist bei visuellen Ansätzen die Informationsreduktion weiterhin ein wichtiger Aspekt. Die vorgestellten Systeme GAZE und GRAVIS werden der Problematik mit Hilfe von Aufmerksamkeit und Adaptivität erfolgreich gerecht. Die Arbeit bildet daher die Grundlage intelligenter aktiver Sehsysteme zur natürlichen Mensch-Maschine-Interaktion, wie sie in einigen Jahren mit großer Wahrscheinlichkeit zur Selbstverständlichkeit werden.

A Fehlermaße

Der RMSE und NRMSE

Das Klassifizierungsverhalten eines neuronalen Netzes für eine Menge von Testdaten $(x_i, y_i) \in \Omega = R^{(N \times M)}$, $i = 1, \ldots, n$ bestehend aus n Wertepaaren läßt sich durch das Fehlermaß der *normierten Wurzel des mittleren quadratischen Fehlers* (NRMSE) beschreiben. Dieser ergibt sich aus dem Vergleich der bei einer Eingabe von x_i ermittelten Netzantwort $y^{(net)}(x_i)$ mit dem gewünschten Sollwert $y^{(soll)}(x_i) := y_i$. Dafür wird zunächst die Wurzel des mittleren quadratischen Fehlers (RMSE) berechnet:

$$\text{RMSE} = \sqrt{\frac{1}{n}\left(\sum_{i=1}^{n}(y^{(net)} - y^{(soll)})^2\right)} \tag{A.1}$$

Zur Normierung dieses Fehlers wird anschließend die Wurzel aus der Varianz σ_Ω des Datensatzes bestimmt:

$$\sigma_\Omega = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\left(\frac{1}{n}\sum_{i=1}^{n} y_i\right) - y^{(soll)}(x_i)\right)} \tag{A.2}$$

Damit ergibt sich der NRMSE zu:

$$NRMSE(\Omega) = \frac{\text{RMSE}}{\sigma_\Omega} \tag{A.3}$$

Würde das Netz also immer mit dem Mittelwert des Testdatensatzes antworten, ergäbe sich ein NRMSE von 1. Ein Wert kleiner als 1 bedeutet dagegen, daß ein gutes Ergebnis nahe der Sollwerte erzielt wurde.

Literaturverzeichnis

[Ahm91] Subutai Ahmad. *VISIT: An efficient computational model of human visual attention*. Dissertation, ICSI, Univ. of California, Berkley, 1991.

[AME93] Tomas Uhlin Atsuto Maki und Jan-Olof Eklundh. Phase-Based Disparity Estimation in Binocular Tracking. In K. Heia et al., Hrsg., *Proc. 8th Scandinavian Conference on Image Analysis*, Norwegian Society for Image Processing and Pattern Recognition, Seiten 1145–1152, Tromsø, Norway, 1993.

[AS89] J. Aloimonos und D. Shulman. *Integration of Visual Modules*. Academic Press, Inc., San Diego, USA, 1989.

[Bal91] D. H. Ballard. Animate vision. *Artificial Intelligence*, 48(1):57–86, 1991.

[BHM97] Maik Bollmann, Rheiner Hoischen und Bärbel Mertsching. Interaction of Static and Dynamic Scene - Features Guiding Visual Attention. In E. Paulus und F. Wahl, Hrsg., *DAGM Mustererkennung 1997*, Informatik aktuell, Seiten 483–490, Berlin, 1997. Springer-Verlag.

[BP94] S. Baluja und D. Pomerleau. Non-Intrusive Gaze Tracking Using Artificial Neural Networks. In J.D. Cowan, G. Tesauro und J. Alspector, Hrsg., *Advances in Neural Information Processing Systems (NIPS) 6*. Morgan Kaufmann Publishers, 1994.

[BS95] P. Ballard und G. C. Stockman. Controlling a Computer via Facial Aspect. *IEEE Transactions on Systems, Man, and Cybernetics*, 25(4):669–677, apr 1995.

[CB93] D. Coombs und C. Brown. Real-Time Binocular Smooth Pursuit. *International Journal of Computer Vision*, 11(2):147–164, 1993.

[Chr92] H. I. Christensen. The AUC robot camera head. In *SPIE Applications of Artificial Intelligence X: Machine vision and Robotics, Proceeding of SPIE, The*

International Society for Optical Engineering, 22–24 April, Orlando, Florida, Jgg. 1708, Seiten 26–33, 1992.

[Chr94] H. I. Christensen. Hardware for the the AUC Camerahead (2nd Generation). Persönliche Kommunikation, 1994.

[DHKS95] K. Daniilidis, M. Hansen, Ch. Krauss und G. Sommer. Auf dem Weg zum künstlichen aktiven Sehen: Modellfreie Bewegungsverfolgung durch Kameranachführung. In G. Sagerer, S. Posch und F. Kummert, Hrsg., *Mustererkennung 95, 17. DAGM-Symposium Bielefeld*, Informatik aktuell, Seiten 277–284, Berlin, 1995. Springer-Verlag.

[DLS96] Jennifer M. Groh David L. Sparks. The Superior Colliculus: A Window for Viewing Issues in Integrative Neuroscience. *MIT Press, Cognitive Neuroscience*, 1996.

[Dre95] A. Drees. *Visuelle Erkennung von Handstellungen mit neuronalen Netzen*. Dissertation, Universität Bielefeld, Technische Fakultät, September 1995.

[Düc95] Christof Dücker. Parametrisierte Bewegungsprimitive für ein Roboter-Kraft-/Momenten-Sensor-Handsystem. Diplomarbeit, Universität Bielefeld, Technische Fakultät, September 1995.

[Fel97a] Winfried A. Fellenz. Ein Neuromorphes System für die Datengetriebene Szenenanal yse. *Forschungsberichte VDI, Reihe 10:Informatik/Kommunikationstechnik Nr. 479*, 1997.

[Fel97b] Winfried A. Fellenz. Modeling the Parallel Development of Multiple Featuremaps and Topography in Visual Cortex. In R. Mira, Moreno-Diaz und J. Cabestany, Hrsg., *Biological and Artificial Computation: From Neuroscience to Techno logy, Proc. IWANN'97*, Jgg. 1240 of *LNCS*. Springer, Berlin, 1997.

[FFP+98] Ahmed A. Faisal, Markus Fislage, Marc Pomplun, Robert Rae und Helge Ritter. Observation of Human Eye Movements to Simulate Visual Exploration of Complex Scenes. Report, Universität Bielefeld, SFB 360 Situierte Künstliche Kommunikatoren, 1998. In Vorbereitung.

[Fis98] Markus Fislage. Visuelle Aufmerksamkeitssteuerung bei gestikbasierter Mensch-Maschine Interaktion. Diplomarbeit, Universität Bielefeld, Technische Fakultät, Juli 1998.

[FKC87] Gonzalez R.C. Fu K.S. und Lee C.S.G. *Robotics - Control, Sensing, Vision and Intelligence*. McGraw-Hill International Editors, 1987.

[FRR00] Markus Fislage, Robert Rae und Helge Ritter. Using Visual Attention to Recognize Human Pointing Gestures in Assembly Tasks. In *6th European Conference on Computer Vision*, ECCV 2000, Dublin, Ireland, 26. June - 01. July, 2000. Eingereicht.

[GC94] A. Gee und R. Cipolla. Determining the gaze of faces in images. *Image and Vision Computing*, 12(10):639–647, Dezember 1994.

[Hei98] Gunther Heidemann. *Ein flexibel einsetzbares Objekterkennungssystem auf der Basis n euronaler Netze*. Dissertation, Universität Bielefeld, Technische Fakultät, 1998. Infix, DISKI 190.

[HNMR96] G. Heidemann, T. Nattkemper, G. Menkhaus und H. Ritter. Blicksteuerung durch präattentive Fokussierungspunkte. In B. Mertsching, Hrsg., *Proceedings in Artificial Intelligence*, Seiten 109–116, 1996.

[HR99] G. Heidemann und H. Ritter. Combining multiple neural nets for visual feature selection and classification. In *Proceedings of Ninth Int. Conf. on Artificial Neural Networks ICANN 99*, Seiten 365–370. Springer-Verlag, 1999.

[IAB+96] M. Irani, P. Anandan, J. Bergen, R. Kumar, und S. Hs u. Mosiac Representations of Video Sequences and Their Applications. *Signal Processing: Image Communication*, 7(4-6):529–552, November 1996. Special issue on Coding Techniques for Low Bit-rate Video.

[IH94] H.O. Istance und P. Howarth. Keeping an eye on your interface: The potential for eye-based control of graphical user interfaces. In *Proceedings of HCI'94*. University Press, August 1994.

[Jac95] R. J.K. Jacob. Eye Tracking in Advanced Interface Design. In W. Barfield und T.A. Furness, Hrsg., *Virtual Environments and Advanced Interface Design*, Seiten 258–288. Oxford University Press, third edition, 1995.

[Joc00] Ján Jockusch. *Exploration based on Neural Networks with Applications in Manipulator Control*. Dissertation, Universität Bielefeld, Technische Fakultät, 2000.

[JRR98] Nils Jungclaus, Robert Rae und Helge Ritter. An Integrated System for Advanced Human-Computer Interaction. In *UCSB-Workshop on Signals and Images (SIPL)*, Seiten 93–97, University of California, Santa Barbara, USA, 1998, 1998.

[Jun98] Nils Jungclaus. *Integration verteilter Systeme zur Mensch-Maschine-Kommunikation*. Dissertation, Universität Bielefeld, Technische Fakultät, 1998.

[JvdHRS98] Nils Jungclaus, Markus von der Heyde, Helge Ritter und Gerhard Sagerer. An Architecture for Distributed Visual Memory. *Zeitschrift für Naturforschung C: A Journal of Biosciences*, 53:550–559, 1998. Special Issue: Natural Organisms, Artificial Organisms, and Their Brains.

[Kat94] H. Kattner. Using Attention as a link between low-level and high-level Vision. Technical Report TUM-I9439, Mathematisches Institut und Institut für Informatik, TU München, 1994.

[Koh84] T. Kohonen. Self-Organization and Associative Memory. In *Springer Series in Information Sciences 8*. Springer Verlag Heidelberg, 1984.

[KR96] Robert Kubisch und Helge Ritter. Erkennung menschlicher Kopfhaltungen mittels künstlicher neuronaler Netze. In B. Jähne et al., Hrsg., *Mustererkennung 1996*, Informatik Aktuell, Seiten 109–117, 18. DAGM Symposium, Heidelberg 11. – 13. Sept., Deutschland, 1996. Springer-Verlag, Berlin, Deutschland.

[Kro89] E. P. Krotkov. *Active computer vision by cooperative focus and stereo*. Springer-Verlag, New York, USA, 1989.

[KSJ95] E.R. Kandel, J.H. Schwartz und T.M. Jessel. *Essentials of Neural Science and Behaviour*. Appleton and Lange, 1995.

[Kub95] Robert Kubisch. Aktives Sehen mittels eines binokularen Kamerakopfes: Ein Ansatz auf der Grundlage neuronaler Netze. Diplomarbeit, Universität Bielefeld, Technische Fakultät, September 1995.

[LDR95] E. Littmann, A. Drees und H. Ritter. Visual gesture-based robot guidance with a modular neural system. In *Neural Information Processing Systems 7*. Morgan Kaufman Publishers, San Mateo, CA, 1995.

[MMRR99] Dieter Metzing, Jan Thorsten Milde, Robert Rae und Helge Ritter. Kommunizierende Agenten: gestische und natürlichsprachliche Interaktion. *Kognitionswissenschaften*, 1999. In Druck.

[NK96] Ernst Niebur und Christof Koch. Control of Selective Visual Attention: Modelling the ”Where” Pathway. In D. Touretzky, M. Mozer und M. Hasselmo, Hrsg., *Advances in Neural Information Processing Systems 8 NIPS*95*. Bradford MIT Press, 1996.

[NR97] C. Nölker und H. Ritter. Detektion von Fingerspitzen in Videobildern. In E. Paulus und F. Wahl, Hrsg., *DAGM Mustererkennung 1997*, Informatik aktuell, Seiten 97–104, Berlin, 1997. Springer-Verlag.

[NR98] C. Nölker und H. Ritter. GREFIT: Visuelle Erkennung kontinuierlicher Handposturen. In J. Dassow und R. Kruse, Hrsg., *Informatik '98: Informatik zwischen Bild und Sprache*, Seiten 213–222, Magdeburg, 1998. Springer Verlag. Reihe "Informatik aktuell".

[PMST94] A. Pentland, B. Moghaddam, T. Starner und M. Turk. View-based and modular eigenspaces for face recognition. In *Proc. IEEE Computer Soc. Conf. on Computer Vision and Patt. Recog.*, Seiten 84–91, 1994.

[PNH86] D. Plaut, S. Nowlan und G. Hinton. Experiments on Learning by Back Propagation. Technical Report CMU-CS-86-126, Carnegie Mellon University, Pittsburgh, 1986.

[Pom98] Marc Pomplun. *Analysis and Models of Eye Movements in Comparative Visual Search*. Dissertation, Universität Bielefeld, Technische Fakultät, 1998. Cuvillier Verlag, Göttingen.

[RB93] M. Riedmiller und H. Braun. A direct adaptive method for faster backpropagation learning: The rprop algrithm. In *ICNN-93 Conference Proceedings*, 1993.

[Rei90] Heinrich Reichert. *Neurobiologie*. Thieme, 1990.

[Rit91] H. Ritter. Learning with the Self-Organizing Map. *Artificial Neural Networks*, 1:379–384, 1991.

[Rit95] H. Ritter. *Neo- the NST graphical editor*, Juni 1995.

[RKK+94] S. Rougeaux, N. Kita, Y. Kuniyoshi, S. Sakane und F. Chavand. Binocular Tracking Based on Virtual Horopters. In *Proc. of the IEEE Int. Conf. on Intelligent Robots and Systems IROS '94, held at Munich, Germany, 23-16 sep, 1994*, Seiten 2052–2057, September 1994.

[RM96] I. D. Reid und D. W. Murray. Active tracking of foveated feature clusters using affine structure. *International Journal of Computer Vision*, 18(1):41–60, 1996.

[Roj93] R. Rojas. *Theorie der neuronalen Netze*. Springer-Verlag, Berlin, Heidelberg, Deutschland, 1993.

[Ros58] F. Rosenblatt. The perceptron: a probalistic model for information storage and organization in the brain? *Psychological Review*, 65:386–408, 1958.

[RR98a] Robert Rae und Helge Ritter. 3D Real-time Tracking of Points of Interest based on Zero-disparity Filtering. In S. Posch und H. Ritter, Hrsg., *Workshop Dynamische Perzeption*, Proceedings in Artificial Intelligence, Seiten 105–111, Workshop der GI-Fachgruppe 1.0.4 Bildverstehen in Bielefeld, 18. – 19. Jun., Germany, 1998. INFIX-Verlag, Sankt Augustin, Deutschland.

[RR98b] Robert Rae und Helge Ritter. Recognition of Human Head Orientation based on Artificial Neural Nets. *IEEE Transactions on Neural Networks*, 09(02):257–265, März 1998.

[Sch94] R. Schuster. Color Object Tracking with Adaptive Modeling. In *Proceedings of the 12th IAPR Int. Conf. on Pattern Recognition, held at Jerusalem, Israel, 9-13 oct, 1994*, Jgg. 1, Seite 443. IEEE Computer Society Press, Oktober 1994.

[Sch95] Schmidth. *Neuro- und Sinnesphysiologie*. Springer, 1995.

[SE93] M. Swain und M. A. Stricker (Eds.). Promising Directions in Active Vision. *International Journal of Computer Vision*, 11(2):109–126, 1993.

[SHB93] Milan Sonka, Vaclav Hlavac und Roger Boyle. *Image Processing, Analysis and Machine Vision*. Chapman and Hall, London, UK, 1993.

[SP98] Daniel Schlüter und Stefan Posch. Combining contour and region information for perceptual grouping. In P. Levi, R.-J. Ahlers, F. May und M. Schanz, Hrsg., *DAGM Mustererkennung 1998*, Informatik aktuell, Seiten 393–401, Berlin, 1998. Springer-Verlag.

[SS97] Heung-Yeung Shum und Richard Szeliski. Panoramic Image Mosaics. Technical Report MSR-TR-97-23, Microsoft Research, ONe Microsoft Way, Redmond, WA 98052, 1997.

[Sta93] D.M. Stampe. Heuristic filtering and reliable calibration methods for video-based pupil-tracking systems. *Behavior Research Methods, Instruments, and Computers*, 25:137–142, 1993.

[SW95] B. Schiele und A. Waibel. Gaze Tracking Based on Face-Color. In *Int. Workshop on Automatic Face- and Gesture-Recognition*, Juni 1995.

[TP97] M. Turk und A. Pentland. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1997.

[Var97] Axel Varchmin. Bildbasierte Blickrichtungserkennung mittels adaptiver Methoden. Diplomarbeit, Universität Bielefeld, Technische Fakultät, August 1997.

[VRR97] Axel Christian Varchmin, Robert Rae und Helge Ritter. Facial feature detection using neural networks. In W. Gerstner, A. Germond, M. Hasler und J. Nicoud, Hrsg., *Artificial Neural Networks – ICANN'97*, Jgg. LNCS 1327 of *Lecture Notes in Computer Science*, Seiten 955–960, 7th Int. Conf. on Artificial Neural Networks, Lausanne, 08. – 10. Oct., Switzerland, 1997. Springer-Verlag, Berlin, Deutschland.

[Wan95] Brian A. Wandell. *Foundations of Vision*. Sinauer, 1995.

[Wex98] Alan Wexelblat. Research Challenges in Gesture: Open Issues and Unsolved Problems. In I. Wachsmuth und M. Fröhlich, Hrsg., *Gesture and Sign Language in Human-Computer Interaction*, Jgg. LNAI 1371 of *Lecture Notes in Artificial Intelligence*, Seiten 1–11, Int. Gesture Workshop, Bielefeld, 17. – 19. Sept. 1997, Germany, 1998. Springer-Verlag, Berlin, Deutschland.

[WW88] Jonathan D. Wirtschafter und Alan S. Weingarden. Neuropsychology and Central Pathways in Oculomotor Control: Physiology and Anatomiy of Saccadic and Pursuit Eye Movements. In Cris W. Johnston und Francis J. Pirozzolo, Hrsg., *Neuropsychology of Eye Movements*, Seiten 5–29. Lawrence Erlbaum Associates, 1988.

[Yar67] A.L. Yarbus. *Eye movements and vision*. Plenum Press, New York, 1967.