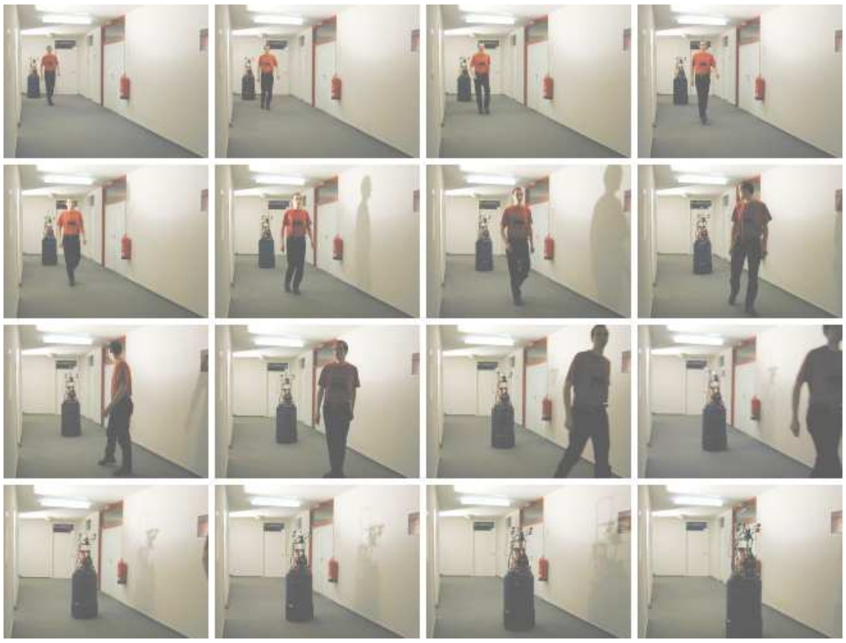
Ein autonomes Robotersystem zum Folgen einer Person



Diplomarbeit
am Lehrstuhl für Kognitive Systeme
des Instituts für Informatik und Praktische Mathematik
der Christian-Albrechts-Universität zu Kiel
Prof. Dr. Gerald Sommer

vorgelegt von
Andreas Bunten

Betreuung:
Dipl. Inform. Bodo Rosenhahn
Dipl. Inform. Volker Krüger

Januar 2002

Zusammenfassung

In der vorliegenden Arbeit wird die Implementierung eines Systems untersucht, welches einen autonomen B21-Roboter befähigt mit Hilfe eines Stereokamerasystems in Echtzeit eine Person zu fixieren und ihr in konstant bleibendem Abstand zu folgen. Es werden dabei alle dem Roboter zur Verfügung stehenden Freiheitsgrade verwendet: Rotation und Translation der Basis, Schwenken und Neigen des gesamten Kamerasystems und Schwenken der einzelnen Kameras. Das kontinuierliche Fixieren einer Person erfordert eine *Kompensation* der Rotation einzelner Gelenke durch andere Gelenke. Die Benutzung aller Freiheitsgrade ergibt eine Redundanz, welche verschiedene Bewegungsstrategien erlaubt. Ziel ist ein robustes System, das ohne exakte Kalibrierung auskommt.

Dem System liegt ein verhaltensbasiertes Konzept zugrunde, welches jedes Gelenk durch ein einzelnes Verhaltensmodul kontrolliert. Die dreifache Kompensation ergibt sich aus dem gemeinsamen Wirken der einzelnen Verhaltensmodule. Art und Parametrisierung der Koordination der Module haben dabei einen bedeutenden Einfluss auf das Gesamtverhalten des Systems.

Es werden sukzessiv alle Komponenten des Systems experimentell untersucht und alle wichtigen Parameter in Versuchen bestimmt. Eigenschaften und Qualität der Kompensation werden bestimmt. Die schrittweise Realisierung des konkreten Gesamtsystems wird so dokumentiert. Das verhaltensbasierte System erweist sich dabei mehrfacher Hinsicht als robust und die Qualität der Kompensation sinkt auch bei Verwendung aller zur Verfügung stehender Freiheitsgrade nicht. Das Gesamtsystem löst die gestellte Aufgabe erfolgreich.

Inhaltsverzeichnis

1 Einleitung

Die Implementierung eines verhaltensbasierten Systems zur Steuerung eines Roboters soll in dieser Arbeit dokumentiert und experimentell untersucht werden. Das System befähigt einen autonomen Roboter, eine Person mit Hilfe eines Stereokamerasystems kontinuierlich zu fixieren und ihr in konstantem Abstand zu folgen. Dies geschieht in einer beliebigen Umgebung und unter Einsatz aller Freiheitsgrade des Roboters.

Im ersten Abschnitt wird die Aufgabenstellung und der Hardwareaufbau skizziert. Nach einer kurzen Vorstellung ähnlicher Arbeiten wird die gestellte Aufgabe motiviert. Der zweite Abschnitt befasst sich mit den Grundlagen der verhaltensbasierten Robotik und dem in dieser Arbeit verfolgten Ansatz. Die konkrete Realisierung des Gesamtsystems wird im darauf folgenden Abschnitt dokumentiert. Der vierte Abschnitt beschreibt die durchgeführten Experimente und deren Ergebnisse. Es folgt eine Diskussion der Arbeit mit einem Ausblick auf mögliche Erweiterungen. Ergänzende Ausführungen zu den verwendeten Algorithmen, zur Entfernungsschätzung und zur Regelung finden sich in den Anhängen.

1.1 Aufgabenstellung

Der verwendete Roboter soll eine Person fixieren und ihr folgen können. Dabei werden alle Freiheitsgrade des Roboters verwendet. Um kontinuierlich fixieren zu können, müssen Rotationen einzelner Gelenke durch die Rotation anderer ausgeglichen werden. Dieser Ausgleich wird im Folgenden *Kompensation* genannt. Die Rotationsachsen von drei Gelenken verlaufen parallel und haben keinen Schnittpunkt. Durch sie kann jeweils eine Rotation der Kameras bewirkt werden. Diese Redundanz erlaubt verschiedene Strategien beim Fixieren und Verfolgen einer Person. Verschiedene Umsetzungen dieser dreifachen Kompensation sollen durch ein verhaltensbasiertes System implementiert und untersucht werden.

Das System soll weiterhin echtzeitfähig und robust sein: Die Person soll sich in normalem Tempo bewegen dürfen. Die Verarbeitung der Daten soll nur an Bord des Roboters stattfinden. Ferner wird die Umgebung nicht für die Aufgabe modifiziert. Eine genaue Kalibrierung des Systems soll nicht nötig sein. Zur Bildverarbeitung und zur Kontrolle der Motoren wird keine spezielle Hardware verwendet.

In den Experimenten werden sukzessiv alle Teile des Systems untersucht, um eine Grundlage für die anschließenden Tests des Gesamtsystems zu erhalten.

1.2 Hardwareaufbau

Es wird ein autonomer B21-Roboter von Real World Interfaces [33] eingesetzt (vgl. Abb. 1). Dieser besitzt zwei CCD-Farbkameras und wird von zwei an Bord befindlichen Rechnern gesteuert. Neben der Translation des gesamten Roboters sind verschiedene Möglichkeiten zur Rotation gegeben. Die Rotation der Basis kann den gesamten Roboter auf der Stelle drehen und legt so die Fahrtrichtung fest. Die Pan-Tilt-Einheit im Aufbau des Roboters ermöglicht ein Neigen und Schwenken der Kameras. Dabei wird das gesamte Stereokamerasystem bewegt.

Die beiden CCD-Kameras können wiederum einzeln durch Vergenzmotoren geschwenkt werden. Die Achsen der Vergenzmotoren, des Pan-Gelenks und der Rotation der Basis verlaufen parallel zueinander, sind aber nicht identisch.

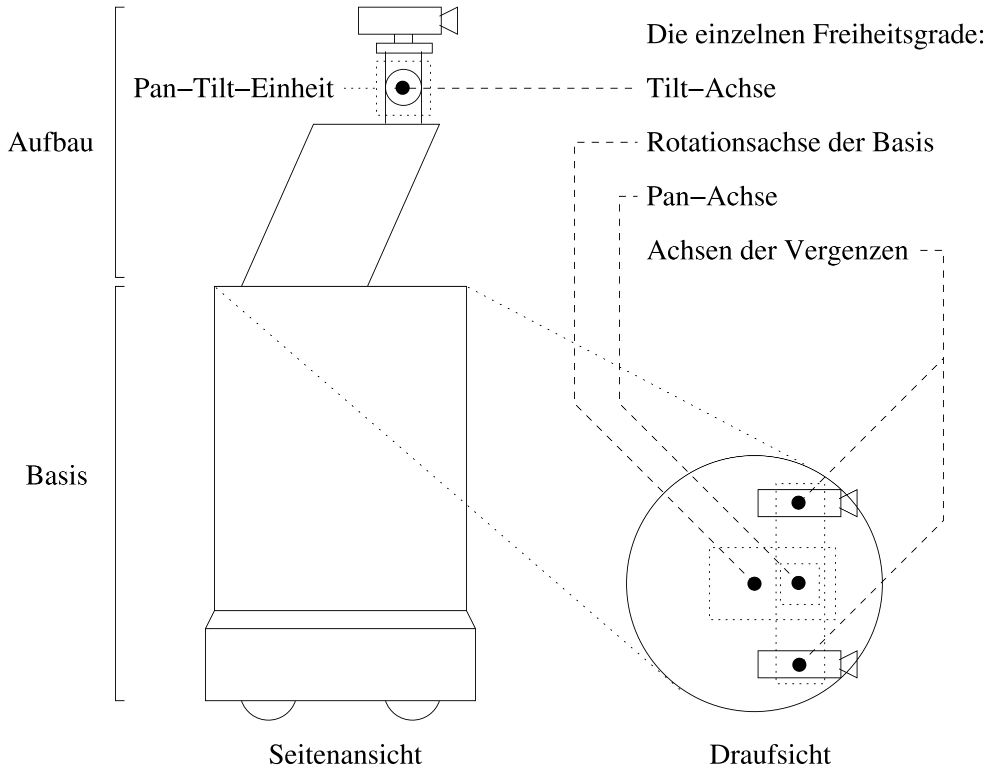


Abbildung 1: *Schematische Darstellung der Freiheitsgrade zur Rotation des B21-Roboters. Die Achse zur Neigung des Stereokamerasystems (*Tilt*) ist links in der Seitenansicht eingezeichnet. Rechts sind in Draufsicht die Achsen zur Rotation des gesamten Roboters (*Basis*), zum Schwenken des Stereokamerasystems (*Pan*) und zur Rotation der einzelnen Kameras (*Vergenzen*) eingezeichnet.*

Zusätzlich zum Stereokamerasystem besitzt der Roboter noch Ultraschall- und Infrarot-Sensoren. Weiterhin können über den Rumpf verteilte Taster eine Kollision feststellen. Eine genauere Beschreibung des Roboters erfolgt in Abschnitt 3.1.

1.3 Einordnung

Das implementierte Gesamtsystem fixiert eine Person und folgt dieser in konstant bleibendem Abstand. Diese Aufgabe ist bereits oft und auf vielfältige Weise gelöst worden. Im Unterschied zur vorliegenden Arbeit werden dabei aber im Allgemeinen Ultraschall- oder Lasersensoren benutzt (siehe z.B. [15], [3]). Drei Arbeiten, in denen ein autonomer Roboter eine Person anhand visueller Daten verfolgt, sollen kurz vorgestellt werden:

Waldherr realisiert im Rahmen einer Diplomarbeit [40] auf einem B21-Roboter ein System zum Folgen einer Person. Es wird ein monookulares Kamerasystem ohne den Freiheitsgrad der

Vergenz verwendet. Zur Entfernungsschätzung und Navigation wurden zusätzlich Laser- und Ultraschallmessungen herangezogen. Die Steuerung ist nicht verhaltensbasiert. Die Schwerpunkte lagen im Bereich der Navigation und der Gestenerkennung im Rahmen der Erteilung von Befehlen an den Robotern.

Feyrer und Zell [21] detektieren die Person auf verschiedene Weisen. Dies gibt dem Verfahren zusätzliche Stabilität. Es wird ein Stereokamerasystem verwendet, aber die Kameras können in diesem Aufbau nicht einzeln bewegt werden. Die Entfernung wird durch Lasermessungen ermittelt. Die externen Parameter der Kamera werden explizit kalibriert.

Ku und Tsai [39] beschreiben ein autonomes Fahrzeug, welches einer Person durch ein Gebäude folgt. Die Person muss dazu ein spezielles Testmuster auf dem Rücken tragen. Es wird dabei lediglich das Fahrzeug selbst rotiert. Es existieren keine weiteren Freiheitsgrade. Schwerpunkte der Arbeit sind die Navigation und die „weiche" Steuerung des Fahrzeuges.

Im Unterschied zu den oben genannten Arbeiten soll in der vorliegende Arbeit ein verhaltensbasiertes System drei parallel laufende Rotationsachsen regeln. Die sich ergebende Redundanz ermöglicht verschiedene Bewegungsstrategien, welche experimentell untersucht und verglichen werden. Die Vor- und Nachteile der Verwendung von mehr als den minimal nötigen Freiheitsgraden sollen dabei bestimmt werden. Die Entfernungsschätzung basiert allein auf visuellen Daten. Ultraschall und Tastsensoren werden nur zur zusätzlichen Absicherung und Kollisionsvermeidung verwendet. Eine explizite Kalibrierung des Systems ist nicht nötig.

1.4 Motivation und Anwendungen

Es ist eine Vielzahl von Anwendungen denkbar, bei denen ein Roboter ein Objekt fixieren und diesem folgen muss. Die Einweisung eines Roboters in einen neuen Arbeitsplatz oder die Überführung der Maschine dorthin vereinfacht sich durch selbstständiges Folgen. In [39] wird vorgeschlagen, den Roboter Angestellten oder Patienten eines Krankenhauses folgen zu lassen oder in ähnlicher Weise die Golftasche über den Golfplatz tragen zu lassen. Ein robotischer Butler ist denkbar.

Diesen einfachen Anwendungen stehen komplexere Aufgaben zur Interaktion mit Personen gegenüber. Ein dem Menschen nachempfundenes Verhalten (u.a. das Fixieren des Objektes, dem momentan die Aufmerksamkeit gilt) fördert einen intuitiven Umgang mit dem Roboter. In [15] wird ein robotischer Museums-Führer vorgestellt. Als Variante ist vorstellbar, dass der Roboter einzelnen Gästen oder Gruppen folgt, anstatt selber zu führen. Zusammen mit einer Gesichtsposenbestimmung (wie zum Beispiel in [27] beschrieben) können Erklärungen zu Exponaten gegeben werden, die längere Zeit von einem Gast betrachtet werden.

Eine weitere Anwendung ist das Fixieren und Folgen von anderen Robotern oder Gegenständen anstatt von Personen. Verschiedene Möglichkeiten zu koordinierten Handlungen sind vorstellbar, unter anderem beim Roboter-Fußball[34].

2 Bewegungsstrategien

Die Freiheitsgrade eines Roboters können mit den Bewegungsmöglichkeiten eines Menschen verglichen werden. Für den B21-Roboter bedeutet das: Die Vergenzmotoren der Kameras ahmen einen Teil der Bewegungen der Augen nach. Pan und Tilt schwenken und neigen das gesamte Stereokamerasystem ähnlich wie Nacken und Hals den Kopf. Eine Rotation der Basis ist mit der Drehung des menschlichen Rumpfes vergleichbar.

Man stelle sich einen Menschen vor, der vor einem sich bewegendem Objekt steht und dieses fixiert: Dazu werden zunächst nur die Augen benutzt. Bei einer größeren Bewegung des Objektes korrigiert die Person die Stellung ihres Kopfes und kompensiert diese Bewegung mit den Augen. Wird der Kopf um mehr als einen gewissen Winkel hinaus gedreht, oder bekommt das Objekt ein erhöhtes Maß an Aufmerksamkeit (zum Beispiel weil die Person dem Objekt folgen möchte), wird der Oberkörper oder auch der gesamte Rumpf nachgeführt. Auch diese Bewegungen müssen durch Hals und Augen kompensiert werden, um das Ziel weiter zu fixieren. Wird dem Objekt gefolgt, so kann es auch fixiert werden, während Hindernissen ausgewichen wird. Einem Menschen stehen eine Vielzahl von Bewegungsmöglichkeiten zur Verfügung, die in der Regel auf anderen Grundvoraussetzungen beruhen, als die Bewegungsformen, über die ein Roboter verfügt. Obwohl dies den direkten Vergleich erschwert, können menschliche Verhaltensweisen zur Strategiefindung bei der Steuerung eines Roboters herangezogen werden.

Im Folgenden sollen verschiedene Ansätze der Robotersteuerung vorgestellt werden. Zuerst wird ein knapper Überblick zu wissensbasierten Systemen gegeben, um dann die verhaltensbasierte Robotik vorstellen zu können. Eine Implementierung - die *Subsumption Architecture* - wird genauer betrachtet. Anschließend kann der in dieser Arbeit verfolgte Ansatz beschrieben und eingeordnet werden.

2.1 Wissensbasierte Robotik

In einem wissensbasierten robotischen System werden im allgemeinen verschiedene sensorische Daten auf ein einheitliches, abstraktes Modell der Umwelt abgebildet (sogenannte *Sensor Fusion*). Eine Planungseinheit legt die nachfolgenden Handlungen fest und gibt diese wiederum an eine Ausführungseinheit, welche die symbolischen Handlungen in Steuersignale für die Motoren umsetzt. Die Planungseinheit soll damit der zentrale Sitz der *Intelligenz* sein. Daraus folgt, dass die Wahl des Modells und die Art der symbolischen Darstellung von zentraler Bedeutung für diese Herangehensweise sind. Dabei hat sich aber die rein symbolische, allgemeine Darstellung der Bedeutung eines Objektes der realen Welt als kaum lösbares Problem erwiesen. Insbesondere ist die Abbildung zweidimensionaler Kamerabilder auf dreidimensionale Modelle der Umgebung sehr schwierig.

Der Entwurf wissensbasierter Systeme verläuft im allgemeinem in *Top-Down*-Vorgehensweise. Die Zerlegung des Problems erfolgt funktional: Es werden Module erstellt, die jeweils gewisse Teilaufgaben lösen. Die Wahrnehmung liefert eine abstrakte, symbolische Beschreibung der Welt (oder eines bestimmten Aspektes davon) und die Motorkontrolle setzt symbolische Anweisungen in reale Aktionen um. Dazwischen verarbeitet eine zentrale Einheit diese abstrakten

Informationen, welche - in der Tradition der klassischen KI - Pläne generiert. In Abb. 2 ist zu sehen, wie die Module nacheinander durchlaufen werden. Im Falle des Versagens eines einzelnen Moduls versagt auch das gesamte System, wenn nicht spezielle Vorkehrungen getroffen wurden. Die Reaktionszeit eines solchen Systems ist relativ lang, sobald die Abbildung der Daten auf das Modell nicht trivial ist und die Planungseinheit einen großen Hypothesenraum durchsuchen muss.

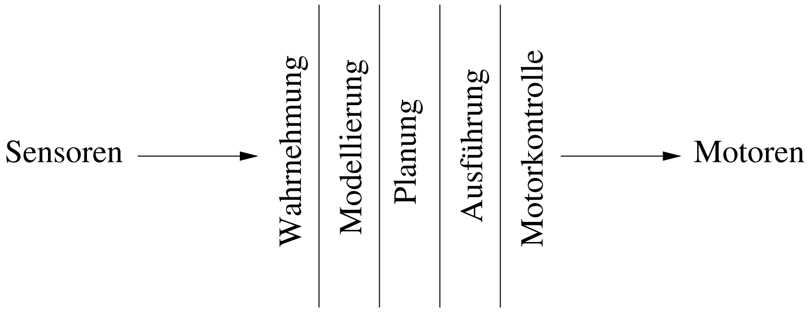


Abbildung 2: *Beispiel einer funktionalen Zerlegung eines Problems gemäß Brooks [9].*

In [30] stellt Nilsson den wissensbasiert gesteuerten Roboter *Shakey* vor, der einige relativ komplexe Probleme selbstständig lösen kann. Allerdings wird die Umgebung stark dem Modell des Roboters angepasst und ist außerdem fast völlig statisch. Der nächste Schritt, die Abkehr vom idealisierten Weltmodell, stellt sich als schwierig heraus. Eine dynamische Umwelt, die nicht speziell für den Roboter hergerichtet wurde und in der unvorhergesehene Ereignisse stattfinden, lässt sich nur schwer in einem Modell erfassen. So kann ein wissensbasierter Roboter zwar klassische Probleme der künstlichen Intelligenz lösen; Aufgaben, wie die Navigation zwischen zwei Orten durch eine nicht vorhersehbare, dynamische Umwelt bleiben aber kompliziert.

2.2 Verhaltensbasierte Robotik

Die verhaltensbasierte Robotik entstand durch die Integration von Ideen aus der Verhaltenslehre und der Neurophysiologie. Ziel ist *Robustheit* und *Anpassungsfähigkeit* in einer nicht vollständig modellierten, sich schnell ändernden Umwelt. Folgende vier Begriffe werden u.a. von Sommer [38] und Arkin [6] für die zentralen Anforderungen an ein verhaltensbasiertes System genannt:

Situiertheit: Der Roboter erhält echte, aktuelle Sensordaten und keine Abstraktion davon. Diese Daten haben unmittelbar Einfluss auf den Zustand des Systems.

Körperlichkeit: Der Roboter kann durch seine physische Präsenz und Wahrnehmung direkt Einfluss auf die Umwelt in für ihn wahrnehmbarer Weise nehmen und kann nur Verhalten zeigen, welche seiner Körperlichkeit entsprechen.

Kompetenz: Anstatt nur an den Menschen angelehnte Intelligenz zu betrachten, wird die Erringung gewisser *Kompetenzen* angestrebt. Diese stellen eine Verallgemeinerung von Intelligenz dar und können auch bei Tieren und z.B. Kleinkindern beobachtet werden[1].

Emergenz: Aus dem Zusammenwirken verschiedener Komponenten des Systems mit einer dynamischen, nicht vollständig modellierbaren Umwelt ergibt sich das Verhalten des Systems. Das Verhalten ist nicht alleine aus den Komponenten des Systems vorhersagbar und komplexer als die Summe der einzelnen Komponenten.

Ein konkretes, verhaltensbasiertes System kann als Gegensatz zu einem rein wissensbasiertem System laut Bogoni [7] wie folgt charakterisiert werden:

- Das gewünschte Gesamtverhalten des Roboters wird in einzelne Module von Wahrnehmung und zugehöriger Handlung zerlegt: sog. *Verhaltensmodule*.
- Die Verhaltensmodule koppeln Sensordaten unmittelbar mit motorischen Reflexen.
- Eine abstrakte Repräsentation von Wissen wird vermieden, da der Aufbau von Weltmodellen aufwändig und fehlerträchtig ist.
- Das System ist durch das Hinzufügen von Verhaltensmodulen erweiterbar.

Aus dem Zusammenwirken einzelner *Verhaltensweisen* ergibt sich das von außen wahrnehmbare *Gesamtverhalten* des Roboters. Implementierte Verhaltensweisen werden im Folgendem *Verhaltensmodule* genannt. Implementierungen von verhaltensbasierten Systemen lassen sich bezüglich ihrer Motivation beim Entwurf wie folgt unterteilen:

Biologisch motiviert: Die Untersuchung von Tieren in der Verhaltensforschung dient als Vorbild für die Strategie zur Lösung eines Problems. Die Reaktionen des Roboters können in Versuchen mit den Reaktionen eines Tieres verglichen werden, um die Zerlegung zu verifizieren oder zu verbessern. Arbib und House [4] untersuchen das Navigationsverhalten einer Kröte und bilden dieses nach.

Durch die Situation aktiviertes Verhalten: Der Roboter wählt anhand der Beobachtung der aktuellen Situation eine Verhaltensweise aus. Wird ein Wechsel der Situation detektiert, wird erneut eine Verhaltensweise gewählt. Hierbei ist ein sehr genaues Wissen über die möglichen Situationen, in denen sich der Roboter befinden kann, nötig. Agre und Chapman [1] steuern ein Computerspiel mit Hilfe situationsaktivierter Verhaltensweisen. Die von Schoppers [36] vorgestellte Variante der *Universal Plans* versucht, jeder möglichen Situation mit einer vorbereiteten Aktion zu begegnen.

Rein experimentell ermitteltes Design: Der Roboter wird mit einer kleinen Zahl von Verhaltensweisen (den *Grundfähigkeiten*) ausgestattet, welche im geplanten Einsatzort ausgiebig getestet werden. Sobald das Gesamtverhalten des Roboters genau den Vorstellungen entspricht, werden schrittweise weitere Verhaltensweisen hinzugefügt und auch ebenfalls ausgiebig getestet. Die Entwicklung erfolgt damit in *Bottom-Up*-Vorgehensweise. Eine

[1] Brooks [12] schlägt insektenähnliche Kompetenz als erstes Ziel vor. Darauf aufbauend soll erst eine abstraktere, Pläne generierende Kompetenz eingeschätzt und untersucht werden.

Variante dieses Ansatzes ist die *Subsumption Architecture*, auf die in Abschnitt 2.2.1 eingegangen wird.

Produzieren verschiedene Verhaltensmodule zur gleichen Zeit verschiedene Stellwerte für den gleichen Motor, so muss dieser Konflikt in irgendeiner Form aufgelöst werden. Die Vorgehensweise bei der Koordination der Verhaltensweisen stellt daher ein weiteres Kriterium zur Unterscheidung verschiedener Ansätze dar:

Kooperative Verhaltensweisen: Die Ausgaben mehrerer Verhaltensweisen werden gleichzeitig berücksichtigt. Arkin [5] realisiert dies durch die Generierung von Vektorfeldern zur Navigation. Eine Kooperation findet durch Addition der Vektoren der einzelnen Verhaltensweisen statt.

Konkurrierende Verhaltensweisen: Stehen die Verhaltensweisen in Konkurrenz, so ist immer nur genau eine aktiv. Die Auswahl dieser Verhaltensweise kann wiederum auf verschiedene Weisen erfolgen: Eine *Hierarchie* kann vorgeben, welche Verhaltensweisen im Falle eines Konfliktes Vorrang vor anderen haben. Es kann eine *Abstimmung* zwischen den Verhaltensweisen realisiert werden. Alternativ kann über eine allgemeine *Auswahlfunktion* genau ein Verhalten gewählt werden. Maes [28] benutzt dynamische *Aktivierungs-Level* für jede Verhaltensweise.

Als Rahmen für die Umsetzung eines verhaltensbasierten Ansatzes wird im allgemeinen eine enge Kopplung von Wahrnehmung und Handlung im sog. *Wahrnehmungs-Handlungs-Zyklus* verwendet. Die Kompetenz des Systems manifestiert sich im inneren Zustand der Verhaltensmodule. Ist das Verhalten des Roboters immer eine direkte, reflexartige Reaktion auf dessen Wahrnehmung der Umwelt, so wird das System als *reaktiv* bezeichnet. Nicht alle Aufgaben lassen sich durch solche Reflexe lösen. Eine Vorgehensweise zur Lösung weiterer Probleme, ist die Verbindung zwischen wissens- und verhaltensbasierten Komponenten. Dabei stoßen *Bottom-Up* und *Top-Down*-Vorgehensweisen aufeinander, und müssen sinnvoll verbunden werden.

Das im Rahmen dieser Arbeit implementierte System besitzt zwar adaptive Komponenten, ist aber prinzipiell reaktiv. Im Folgenden soll daher ein spezieller, rein reaktiver Ansatz genauer betrachtet werden, um die Eigenschaften des verwendeten Systems näher beschreiben zu können.

2.2.1 Subsumption Architecture

Die von Brooks [11] vorgestellte *Subsumption Architecture* ist ein Beispiel für verhaltensbasierte Systeme. Sie lässt sich wie folgt charakterisieren:

- Es werden reale, autonome Roboter konstruiert, die in einer Umwelt agieren, welche nicht für sie angepasst wurde. Simulationen werden abgelehnt, da diese niemals die reale Welt in ihrer vollen Komplexität darstellen.
- Die Systeme werden inkrementell im Sinne des experimentell ermittelten Designs konstruiert. Dies soll eine Art Evolution simulieren und entspricht einer *Bottom-Up*-Vorgehensweise.

- Es entstehen rein reaktive Systeme, deren Verhaltenskompetenz z.B. mit der von Insekten vergleichbar ist.
- Es soll keine globale, abstrakte Repräsentation von Wissen geben und ebenso keine aufwendige Kalibrierung.
- Robustheit gegenüber Ausfällen von Sensoren oder einzelnen Verhaltensmodulen soll erreicht werden.
- Die einzelnen Verhaltensmodule sollen wenig intern kommunizieren. Stattdessen wird *die Welt als ihr eigenes Modell* benutzt. Dies bedeutet, dass ein Verhaltensmodul durch seine Aktionen Spuren in der Umwelt hinterlässt, andere Verhaltensmodule diese Auswirkungen detektieren und darauf reagieren.
- Ideen aus Verhaltenslehre und Neurophysiologie werden genutzt, aber technische Aspekte stehen bei der Implementierung und der Zerlegung des Problems im Vordergrund.

Im Gegensatz zur funktionalen Zerlegung eines Problems wird im verhaltensbasierten Ansatz gemäß Brooks eine Zerlegung nach Aktionen angestrebt. Es werden selbstständig agierende Verhaltensmodule erzeugt, die jeweils Zugriff auf sämtliche Sensordaten haben und alle Motoren ansteuern können. Eine verhaltensbasierte Zerlegung ist schematisch in Abb. 3 dargestellt. Die Verhaltensmodule sind nicht direkt voneinander abhängig.

Plane Veränderungen der Welt
Identifiziere Objekte
Beobachte Änderungen
Sensoren → Baue Karten auf → Motoren
Erkunde Umgebung
Bewege dich
Weiche Objekten aus

Abbildung 3: *Verhaltensbasierte Zerlegung eines Problems gemäß Brooks [9] als Gegensatz zur funktionalen Zerlegung in Abb. 2.*

Bei der Entwicklung sollen zunächst einfache Verhaltensmodule implementiert und systematisch getestet werden. Funktionieren diese zur Zufriedenheit, können schrittweise neue Verhaltensmodule hinzugefügt werden. Das Hinzufügen neuer Verhaltensmodule soll keinen Einfluss auf die bestehenden Module haben. Die Handlungskompetenz des Roboters wird in einem anhaltenden Zyklus von *Erweiterung und Test* ausgebaut. Gemäß der in Abschnitt 2.2 definierten Einteilung führt dieses Vorgehen zu einem rein experimentell ermittelten Design.

Die einzelnen Verhaltensmodule konkurrieren miteinander. Werden widersprechende Ausgaben produziert, muss eine als gültig ausgewählt werden. Trotz der prinzipiellen Unabhängigkeit sind die Verhaltensmodule daher in einer Hierarchie angeordnet. Ein Verhaltensmodul kann auf ein weiter unten in der Hierarchie angesiedeltes Modul folgende Mechanismen anwenden, um einen Konflikt aufzulösen:

Unterdrückung: Ein Verhaltensmodul kann die Ausgabe eines untergeordneten Verhaltensmoduls unterdrücken. Die unterdrückte Ausgabe wird verworfen (*Inhibition*).

Überdeckung: Anstatt nur die Ausgabe eines untergeordneten Verhaltensmoduls zu unterdrücken, wird anstatt der verworfenen eine neue Ausgabe erzeugt (*Subsumption*).

Signale: Verhaltensmodule können sich einfache, asynchrone Signale schicken. Zur Übertragung findet keine Synchronisation statt. Wird ein Signal verschickt, ist nicht klar, wann es vom Empfänger bemerkt wird oder ob es vor dem Bemerken durch ein nachfolgendes Signal überschrieben wird. Eine typische Anwendung ist das Zurücksetzen eines Verhaltensmoduls in dessen Startzustand mit Hilfe eines Reset-Signals.

Konflikte zwischen Verhaltensmodulen werden durch die genannten Mechanismen vermieden. Durch die Auflösung von Konflikten gemäß der Hierarchie, steht die Position eines Verhaltensmoduls innerhalb der Hierarchie, auch für dessen Handlungskompetenz. Die Koordination gemäß dieser strengen Hierarchie wird von Brooks [10] durch dynamische Aktivierungs-Level erweitert.

Jedes Verhaltensmodul hat ein eigenes Ziel und verfolgt dieses selbstständig. So lassen sich auch verschiedene, konkurrierende Aufgaben in dieser Architektur implementieren. Ein komplexes Gesamtverhalten muss daher nicht aus einer komplexen, zentralen Kontrolle entstehen. Es entsteht erst aus der gemeinsamen Interaktion der einzelnen Verhaltensmodule mit einer nicht vorhersehbaren, dynamischen Umwelt. In konkreten Implementierungen der Subsumption Architecture bestehen die einzelnen Verhaltensmodule im Allgemeinen aus kleinen, deterministischen Automaten mit weniger als 10 Zuständen.

Robustheit entsteht durch die Kapselung von Wahrnehmung und Aktionen. Einfache, reaktive Verhaltensmodule sind nicht auf das Funktionieren der in der Hierarchie höher angesiedelten Verhaltensmodule angewiesen. Reflexe zum Selbstschutz des Roboters und seiner Umgebung sind leicht zu implementieren. Verhaltensmodule mit höherer Handlungskompetenz wiederum können sich auf verschiedene einfache Verhaltensmodule stützen, die verschiedene Sensordaten auswerten oder zumindest auf unterschiedliche Weise Informationen aus diesen Daten extrahieren. Fällt ein Sensor aus oder versagt ein Verfahren in einem einfachen Verhaltensmodul, wird der Roboter nicht handlungsunfähig[2].

2.2.2 Diskussion der Subsumption Architecture

Die Vorgehensweise in der Subsumption Architecture bringt spezifische Probleme mit sich:

Es gibt keine klare Methodik, wie eine verhaltensbasierte Zerlegung eines Problems tatsächlich anzugehen ist. Ein Problem kann auf sehr unterschiedliche Weise erfolgreich modelliert werden; häufig entsprechen erfolgreiche Zerlegungen nicht der intuitiven Vorstellung. Brooks rät

[2] Dies wird von Arkin [6] als *Behavior Fusion* im Gegensatz zur oben genannten Sensor Fusion bezeichnet.

zum *clever engineering*[3]. Colombetti et al [16] versuchen, eine strengere Methodik in die Modellierung zu bringen. Wilson und King [42] verwenden Algorithmen zur Evolution bei der Modifikation von verhaltensbasierten Systemen in der Simulation.

Das strenge Verfolgen des Leitsatzes *die Welt ist ihr eigenes Modell* führt bei der Implementierung von Connell [18] zu eigentümlichen Konstruktionen. Es wird ein Roboter vorgestellt, der in einer Büroumgebung Dosen einsammelt. Hierbei wird lediglich eine Unterdrückung zur internen Koordination zwischen den einzelnen Verhaltensmodulen des Roboters benutzt. Ein Verhaltensmodul bedingt einen Greifreflex, der durch das Brechen einer Lichtschranke zwischen den Greiferbacken mit einer Verzögerung von 2 Sekunden ausgelöst wird. Der Roboter greift also auch bei zufälliger Unterbrechung der Lichtschranke zu. Durch Kontextwissen könnte eine zusätzliche Bedingung geschaffen werden, die eine Fehlauslösung verhindert (z.B. kann gefordert werden, dass vor dem Greifen immer eine Annäherung an das Ziel stattgefunden haben muss). Das vorgestellte System ist erfolgreich, aber fest einprogrammierten Zeitspannen (z.B. die 2 Sekunden Verzögerung vor dem Greifen) lassen das System unflexibel und schwer erweiterbar erscheinen.

Durch das Weglassen von globalem Wissen müssen manche Berechnungen mehrmals in verschiedenen Verhaltensmodulen durchgeführt werden. Dies stellt einen zusätzlichen Aufwand für die Entwicklung und für die Ressourcen des Roboters dar.

Der vorgeschlagene schrittweise Aufbau der Verhaltenskompetenz sieht vor, dass erst die Grundfähigkeiten durch einfache Verhaltensmodule implementiert und sehr ausführlich in einer nicht speziell für den Roboter hergerichteten Umgebung getestet werden. Für die Implementierung neuer Module sollen dann die bestehenden nicht mehr geändert werden. Dabei wird vom Entwickler ein sehr großes Maß an Voraussicht verlangt, das der Praxis sehr fern ist. Müssen grundlegende Verhaltensmodule dennoch modifiziert werden, so beginnt der Zyklus von Test und Anpassung aller darauf aufbauenden Module von neuem.

Es ist nicht klar ersichtlich, wieviel *Intelligenz* mittels rein reaktiver Verhaltensweisen (vgl. Subsumption Architecture) möglich ist und ob auf diese Weise komplexe Aufgaben gelöst werden können, die eine längere Abfolge von Handlungen und Planung benötigen. Ein sinnvoller Wechsel zwischen mehreren komplexen Aufgaben sollte möglich sein; bei der Subsumption Architecture kann allerdings im allgemeinen kein sinnvolles Zusammenwirken der Verhaltensweisen garantiert werden. Hartley und Pipitone [24] führen aus, dass die Subsumption Architecture für Anwendungen, wie z.B. das Steuern eines Flugzeuges ausscheidet, da ein Erfolg nur wahrscheinlich und nicht garantiert ist. Wünschenswert ist auch eine Adaption einzelner Verhaltensweisen und deren Koordination (z.B. durch das Erlernen, in welchen Situationen Überdeckung oder Unterdrückung bestimmter Verhaltensweisen zum gewünschten Ziel führt). Dies ist im ursprünglichen Ansatz von Brooks nicht vorgesehen.

[3]Pauli [31] bezeichnet dies in einem Überblick als *ad-hocery*.

2.3 Verhaltensbasierte Steuerung in dieser Arbeit

Die Kontrolle des Roboters erfolgt in der vorliegenden Arbeit durch ein verhaltensbasiertes System, da schnell auf eine nicht vorhersehbare, dynamische Umwelt reagiert werden muss. Es wird untersucht, auf welche Weise die Regelung der einzelnen Motoren durch separate Verhaltensmodule vereinfacht werden kann. Die Vorgehensweise ähnelt der *Subsumption Architecture*, erweitert allerdings deren Rahmen: Es wird sowohl Wissen über die Umwelt eingesetzt, als auch die Koordination der Verhaltensweisen untereinander für die Versuche in Abschnitt 4.4 variiert. Die verhaltensbasierte Zerlegung des Problems bleibt dabei konstant und soll im Folgenden vorgestellt werden. Die Kernpunkte der Implementierung können wie folgt umrissen werden:

- Die implementierten Verhaltensmodule produzieren auf der Basis aktueller Sensordaten Befehle zur Steuerung der Motoren, ohne sich gegenseitig zu synchronisieren.
- Die Verhaltensmodule stehen untereinander in Konkurrenz. Stehen die Ausgaben verschiedener Module im Konflikt, wird dieser anhand einer Hierarchie aufgelöst. Verhaltensmodule können die Ausgaben von in der Hierarchie tiefer stehenden Modulen unterdrücken oder überdecken. Außerdem ist das Versenden asynchroner Signale möglich.
- Die Implementierung erfolgt in *Bottom-Up*-Vorgehensweise. Einfache Verhaltensmodule werden realisiert und ausgiebig getestet, bevor weitere hinzugefügt werden.
- Das System kommt ohne exakte Kalibrierung aus und ist robust bezüglich des Ausfalls einzelner Verhaltensmodule.
- Es existiert keine explizite, symbolische Darstellung von Wissen über die Welt. Es werden jedoch die Messwerte der Sensoren gespeichert, um Verhaltensweisen zur Laufzeit zu adaptieren und zu kontrollieren.

Das Speichern von Messwerten und die Überprüfung von Zwangsbedingungen setzt implizites Wissen über die Welt voraus. Es wird z.B. davon ausgegangen, dass die detektierte Person sich zwischen zwei aufeinander folgend aufgenommenen Bildern nicht beliebig weit bewegen kann (vgl. Abschnitt 2.3.1).

In Abb. 4 sind sechs implementierte Verhaltensmodule dargestellt. Die Verhaltensmodule *Richte Basis aus*, *Richte Pan aus*, *Richte Vergenzen aus* und *Richte Tilt aus* kontrollieren jeweils genau ein Rotationsgelenk. Zusammen ergeben sie die Verhaltensweise *Fixiere Person*. Das Modul *Halte Abstand* schätzt die Entfernung zur Person und führt falls nötig eine Translation durch. *Kontrolliere Zwangsbedingungen* überprüft die Konsistenz der Mess- und Stellwerte und versetzt gegebenenfalls einzelne Verhaltensmodule in den Startzustand. Zusammen realisieren sie die Verhaltensweise *Folge Person in konstantem Abstand*. Die Möglichkeiten zur Unterdrückung (*Inhibition*) und Überdeckung (*Subsumption*) der Ausgaben von Verhaltensmodulen wird durch Pfeile mit der Beschriftung **I** bzw. **S** bezeichnet. Gestrichelte Linien stellen Signalflüsse dar. Das Signal besteht entweder in einer Aufforderung zur Kompensation oder zur Rückkehr in den Startzustand.

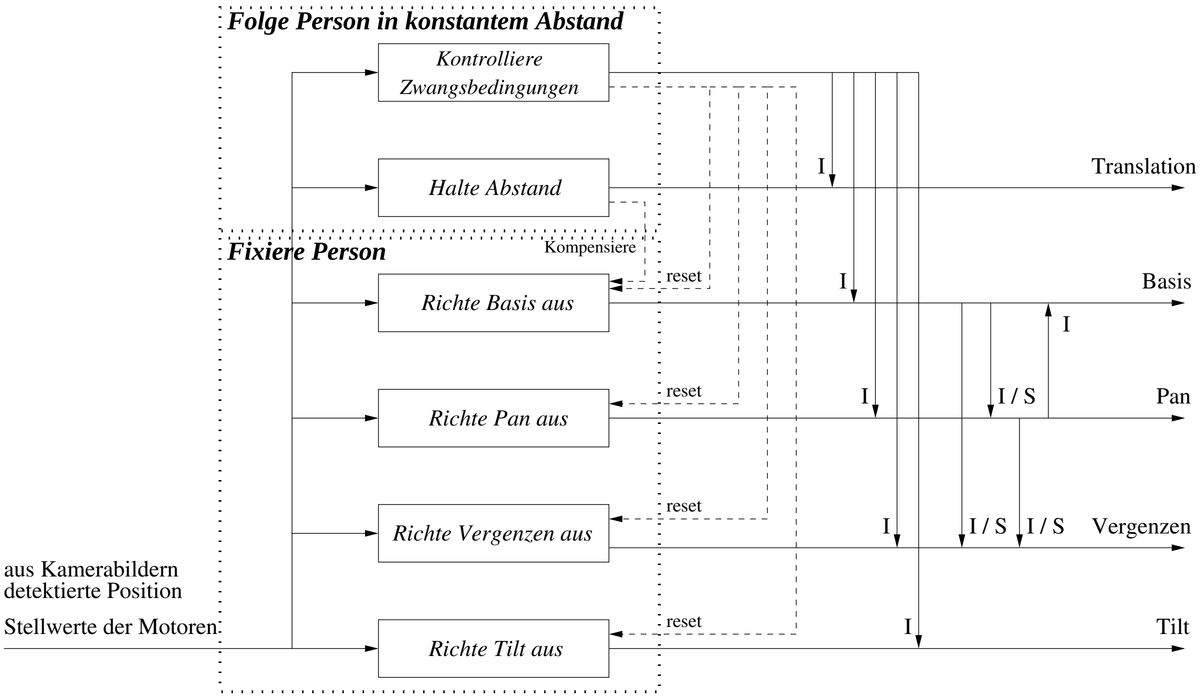


Abbildung 4: *Sechs implementierte Verhaltensmodule sind als Kästen dargestellt. Die verschiedenen getesteten Möglichkeiten zur gegenseitigen Unterdrückung (**I**nhibition) und Überdeckung (**S**ubsumption) sind als Pfeile eingezeichnet. Zwischen den Modulen* Richte Basis aus, Richte Pan aus *und* Richte Vergenzen aus *sind alle in den Versuchen getesteten Variationen zusammen eingezeichnet. Die in den gepunkteten Kästen zusammengefassten Verhaltensmodule stellen die beiden realisierten Verhaltensweisen dar. Signalflüsse sind als gestrichelte Linien angedeutet.*

Alle Verhaltensmodule sind als deterministische, zeitabhängige Automaten implementiert. Diese realisieren die Mechanismen zur Koordination der Module und stellen sicher, dass die notwendigen Wartezeiten bei der Ansteuerung der Motoren eingehalten werden. In Abschnitt 3.1.5 wird auf die Notwendigkeit solcher Wartezeiten eingegangen und in Abschnitt 4.1 werden diese Wartezeiten experimentell bestimmt. Alle Verhaltensmodule, die Motoren ansteuern, benutzen geschlossene Regelkreise. In Abschnitt 4.3 werden verschiedene Regler mit variierender Parametrisierung experimentell untersucht. Für die Regelkreise werden Mess- und Stellwerte jeweils in einem lokalem Gedächtnis gespeichert.

2.3.1 Implementierte Verhaltensmodule

Als Wahrnehmung der Umwelt stehen allen Verhaltensmodulen die Bilddaten und die Messwerte der Ultraschallsensoren zur Verfügung. Letztere dienen nur der Vermeidung von Kollisionen. Zur Eigenwahrnehmung des Systems werden die Stellwerte der Motoren genutzt. Die Bilddaten sind direkt (d.h. ohne Vorverarbeitung) zugreifbar; für die gegebene Aufgabenstellung nutzen die Verhaltensmodule allerdings nur die Position der in den Kamerabildern detektierten Person. Auf die Personendetektion wird in Abschnitt 3.2.3 genauer eingegangen. Im Folgenden sollen die einzelnen Verhaltensmodule kurz vorgestellt werden.

Richte Tilt aus

Das Verhaltensmodul *Richte Tilt aus* bildet das arithmetische Mittel der detektierten vertikalen Position im linken und rechten Kamerabild. Der Automat wechselt in den Zustand *Kompensieren*, wenn die gemittelte Position um einen gewissen Schwellenwert vom Sollwert - der Bildmitte - abweicht. Ein Regelkreis korrigiert diesen Wert mit Hilfe des Tilt-Motors. Liegt die gemittelte Position innerhalb einer definierten Umgebung der Bildmitte, wird wieder in den *Ruhezustand* gesprungen und die Regelung wird ausgesetzt. Diese Umgebung wird durch einen zweiten Schwellenwert definiert. Die beiden Schwellenwerte sind Parameter, welche das Gesamtverhalten des Systems stark beeinflussen. Sind sie zu groß, reagiert das Verhaltensmodul eventuell zu spät und das Objekt wird aus dem Blickfeld verloren. Sind sie zu klein, wird das System instabil und es finden unnötige Bewegungen statt.

Richte Vergenzen aus

Auch hier werden die in den Kamerabildern detektierten Positionen als Eingabe verwendet. Die Regelung setzt ein, wenn die horizontalen Positionen einen definierten Abstand von der Bildmitte überschreiten. Je nach Parametrisierung des Verhaltens werden verschiedene Regelkreise verwendet. Das dadurch ausgelöste Nachführen der Vergenzen wird nur durch Unterdrückung (bzw. Überdeckung) oder durch das Signal zur Rückkehr in den Startzustand unterbrochen.

Richte Pan aus

Im Verhaltensmodul *Richte Pan aus* wird aus den Vergenzwinkeln der Wert Ψ berechnet, der die Ausrichtung des Roboterkopfes zum Objekt angibt. Ähnlich wie bei *Richte Tilt aus* wird zwischen einem *Ruhezustand* und einem Zustand *Kompensieren* durch Abfrage verschiedener Schwellenwerte bezüglich Ψ gewechselt. Die Interaktion mit den anderen Verhaltensmodulen ist auf vier verschiedene Weisen realisiert. Über einen Parameter kann zur Laufzeit eine andere Art der Koordination gewählt werden:

Nur Pan: Der Freiheitsgrad Pan wird ohne Koordination mit anderen Verhaltensmodulen geregelt.

Überdeckung der Vergenzen: Der Freiheitsgrad Pan wird geregelt; die Vergenzen unter Berücksichtigung des neuen Panwinkels nachgesteuert. Das Verhaltensmodul *Richte Vergenzen aus* kann zum gleichen Zeitpunkt keine Ausgaben produzieren.

Überdeckung der Vergenzen und Unterdrückung der Rotation der Basis: Wie bei der Überdeckung der Vergenzen werden die Freiheitsgrade Pan und Vergenzen geregelt. Die Verhaltensmodule *Richte Vergenzen aus* und *Richte Basis aus* werden am Erzeugen neuer Ausgaben gehindert.

Unterdrückung der Rotation der Basis: Der Freiheitsgrad Pan wird geregelt; gleichzeitig wird das Verhaltensmodul *Richte Basis aus* am Generieren neuer Ausgaben gehindert.

Richte Basis aus

Aus den Vergenzwinkeln wird der Wert Ψ berechnet, der die Ausrichtung des Roboterkopfes zum Objekt angibt. Sobald Ψ und der Pan-Winkel einen definierten Schwellenwert überschreiten, wird zwischen den Zuständen *Ruhezustand* und *Kompensieren* gewechselt. Die Interaktion mit anderen Verhaltensmodulen ist ähnlich wie bei *Richte Pan aus* unterschiedlich realisiert. Die verschiedenen Varianten der Koordination können durch einen Parameter zur Laufzeit gewechselt werden:

Nur Basis: Die Rotation der Basis wird ohne Koordination mit anderen Verhaltensmodulen geregelt.

Überdeckung der Vergenzen: Die Rotation der Basis wird geregelt und die Vergenzen werden unter Berücksichtigung der Bewegung der Basis gesteuert. Das Verhaltensmodul *Richte Vergenzen aus* kann zeitgleich keine Ausgaben produzieren.

Überdeckung von Pan: Die Rotation der Basis wird geregelt und die Pan-Einheit unter Berücksichtigung der Bewegung der Basis gesteuert. Das Verhaltensmodul *Richte Pan aus* wird zeitgleich am Generieren neuer Ausgaben gehindert.

Überdeckung der Vergenzen und Unterdrückung von Pan: Wie bei der Überdeckung der Vergenzen werden die Rotation der Basis und der Vergenzen geregelt. *Richte Vergenzen aus* und *Richte Pan aus* werden am Erzeugen neuer Ausgaben gehindert.

Überdeckung der Vergenzen und von Pan: Die Rotation der Basis, der Vergenzen und der Pan-Einheit werden geregelt. Die Verhaltensmodule *Richte Vergenzen aus* und *Richte Pan aus* werden am Erzeugen neuer Ausgaben gehindert.

Unterdrückung von Pan: Die Rotation der Basis wird geregelt, während das Verhalten *Richte Pan aus* am Generieren neuer Ausgaben gehindert wird.

Halte Abstand

Aus den Vergenzwinkeln wird der Wert Ψ und die Entfernung des Roboters zur Person berechnet. Bei Überschreitung der gewünschten Entfernung wird eine Translation durchgeführt. Die Translation wird in drei Zuständen modelliert, um auf verschiedene Entfernungen mit angemessener Translationsgeschwindigkeiten reagieren zu können. Vor einer Translation wird die Ausrichtung der Basis überprüft und gegebenenfalls ein Signal zur Kompensation an *Richte Basis aus* gesendet. Ein Wechsel zwischen den Zuständen findet mit einer gewissen Trägheit statt. So führen einzelne, fehlerhafte Schätzungen der Entfernung nicht zur Änderung der Translationsgeschwindigkeit.

Kontrolliere Zwangsbedingungen

Das Verhaltensmodul *Kontrolliere Zwangsbedingungen* verwendet die im Kamerabild detektierte Position und die Motorstellwerte als Eingabewerte. Es wird überprüft, ob der momentane

Zustand inkonsistent ist, um dann (sofern möglich) korrigierend einzugreifen. Mess- und Stellwerte werden zusätzlich gespeichert, um zeitliche Inkonsistenzen zu finden. Eine sich ergebene Zwangsbedingung ist z.B., dass die linke und rechte Vergenz ein gemeinsames Ziel fixieren müssen[4].

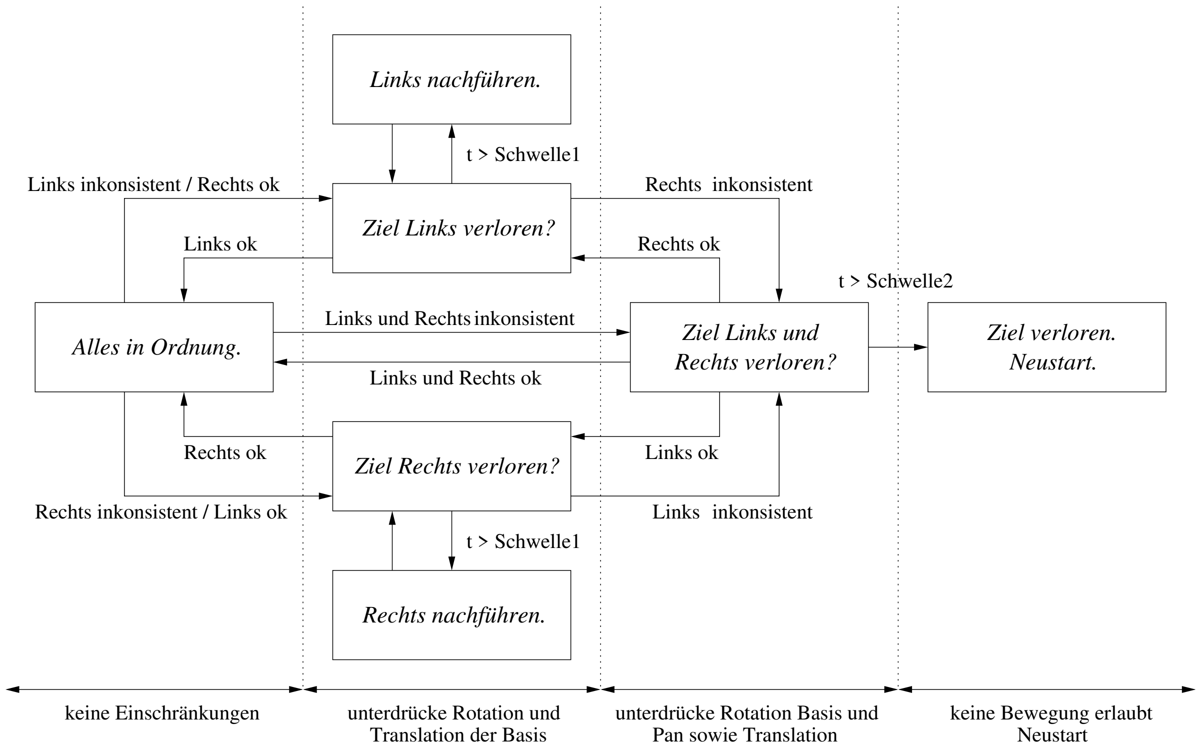


Abbildung 5: *Schematische Darstellung des Automaten zum Verhaltensmodul* „Kontrolliere Zwangsbedingungen". *Eine zeitliche Komponente wird durch den Zähler t realisiert, der beim Verweilen in einem Zustand inkrementiert wird. Bei Zustandswechseln wird t auf Null gesetzt. Drei Bedingungen für Zustandswechsel fragen den aktuellen Stand des Zählers t ab. Die gestrichelten Linien deuten an, welche Konsequenzen das Verweilen in einem Zustand für andere Verhalten hat.*

Das Verhaltensmodul wird durch einen deterministischen Automaten realisiert, der in Abb. 5 dargestellt ist. Der Automat besitzt sieben Zustände und benutzt für Entscheidungen über den Wechsel zwischen den Zuständen die Prädikate *Links* und *Rechts*, die jeweils den Wert *ok* oder *inkonsistent* annehmen können. Sie zeigen an, ob Zweifel an der Korrektheit der jeweils detektierten Position bestehen. Zusätzlich wird eine Zeit-Variable t mitgeführt, welche beim Verbleiben im selben Zustand inkrementiert und beim Wechsel in einen anderen Zustand auf Null gesetzt wird. Die Zustände *Ziel Links verloren?*, *Ziel Rechts verloren?* und *Ziel Links und Rechts verloren?* werden betreten, wenn Zweifel an der Korrektheit der detektierten Position im linken, rechten oder in beiden Kamerabildern existieren. Nach einer kurzen Zeitspanne wird versucht, das Ziel wiederzufinden: Verweilt der Automat länger als *Schwelle1* Runden im Zustand *Ziel Links verloren?*, wird in den Zustand *Links nachführen.* gewechselt. Dort wird die

[4]Divergieren die Vergenzen, fixieren sie auch kein gemeinsames Objekt und die Zwangsbedingung ist verletzt.

linke Vergenz so bewegt, dass ein von der rechten Vergenz fixiertes Ziel auch im Sichtbereich der linken Vergenz erscheint. Bestehen an den Werten der linken und rechten Kamerabilder länger als *Schwelle2* Runden Zweifel, wurde die Person aus dem Sichtbereich verloren: Die Vergenzmotoren werden in die Ausgangsstellung gebracht.

Sobald Zweifel an der Korrektheit der Daten bestehen, soll der Roboter *vorsichtiger* agieren, um sich und umstehende Personen nicht zu gefährden. Vorsichtiger bedeutet hier, dass bei zunehmenden Zweifeln weniger Aktionen möglich sind. Solange Zweifel an den Werten einer Kamera bestehen, wird auf Translation und Rotation der Basis verzichtet. Werden die Werte beider Kameras in Frage gestellt, werden nur noch Bewegungen der Vergenzen zugelassen. Wurde die Person aus dem Sichtbereich verloren, werden alle anderen Verhaltensmodule an der Generierung von Ausgaben gehindert und durch ein Signal in den Startzustand gebracht.

Die Prädikate *Links* und *Rechts*, die Zweifel an den detektierten Punkten im linken bzw. rechten Kamerabild ausdrücken, werden anhand dreier Kategorien von Zwangsbedingungen belegt. Es gehen dabei aktuelle und gespeicherte Mess- und Stellwerte ein:

Geometrische Zwänge: Ohne Berücksichtigung älterer Zustände ergeben sich statische, geometrische Bedingungen, die immer erfüllt sein müssen:

- Da der Kopf des Roboters sich nicht seitwärts neigen lässt, müssen die detektierten Ziele in beiden Kamerabildern auf etwa der gleichen Höhe liegen. Ist dies nicht der Fall, ist ein Teil des Ziels eventuell nur für eine Kamera vollständig sichtbar, oder es werden zwei verschiedene Ziele fixiert.
- Die Größe der detektierten, farblichen Zusammenhangskomponenten in den beiden Bildern darf sich nicht zu stark in der Anzahl der Pixel sowie in der horizontalen und vertikalen Ausdehnung unterscheiden.
- Die geschätzte Entfernung zur Person darf bei Navigation in Innenräumen eine gewisse Grenze nicht überschreiten. Diese Grenze wird in Metern vorgegeben und kann je nach Situation angepasst werden.

Zeitliche Kontinuität: Bei der Betrachtung einer zeitlichen Folge von Werten wird eine gewisse Kontinuität erwartet. Insbesondere wird folgendes überprüft:

- Die Position der detektierten Person kann durch schnelle Bewegungen springen. Mehrmaliges, aufeinander folgendes Springen deutet aber im allgemeinen auf eine Fehldetektion hin.
- Die Entfernung des Ziels kann sich nicht um mehr als wenige Meter pro Sekunde ändern.
- Ändert sich die Größe der farblichen Zusammenhangskomponente stark, kann dies auf plötzliche Beleuchtungsänderungen hinweisen. Existiert gleichzeitig eine Diskrepanz zwischen den Werten der linken und rechten Kamera, ist ein Fehler wahrscheinlich.

Zwänge des Systems: Die Software ist - wie in Abschnitt 3.2 beschrieben - in unterschiedliche Komponenten aufgeteilt. Wird ein Fehler detektiert, wird dieser entweder behoben, oder der Fehlerzustand kenntlich gemacht.

Die Zwänge des Systems sind z.B. bei den *Tracker*-Prozessen zur Verarbeitung der Bilder anwendbar. Tritt dort ein Fehler auf, der zum Stoppen der Bildverarbeitung führt, so wird dies erkannt, da ein entsprechender Zeitstempel nicht aktualisiert wird. Die Verarbeitung der Bilder der beiden Kameras, wird durch unabhängige Prozesse durchgeführt. Im Fehlerfall ist gegebenenfalls nur eine Kamera betroffen. Der Prozess kann neu gestartet werden und die entsprechende Vergenz wird nach kurzer Zeit auf das von der anderen Kamera fixierte Ziel ausgerichtet.

3 Realisierung des Systems

Die Software zur Steuerung des Roboters besteht aus einer Reihe von autarken Bausteinen, die auf verschiedenen Wegen miteinander kommunizieren. Einige dieser Bausteine steuern die Hardware, andere stellen Kontrollinstanzen dar. Im Folgenden sollen zuerst die Hardware, dann die einzelnen Softwarebausteine und das daraus erstellte Gesamtsystem vorgestellt werden.

3.1 Die Hardware

Der verwendete Roboter ist autonom und hat zwei Rechner an Bord, welche durch ein lokales Netzwerk verbunden sind. Eine Funkethernet-Brücke stellt die Verbindung zu einer Unix-Workstation[5] her. Dies ist auf Abb. 6 schematisch dargestellt. Das lokale Netz an Bord des Roboters wird mit 10 MBit/s betrieben, die Verbindung über Funk bietet nur eine Bandbreite von 2 MBit/s.

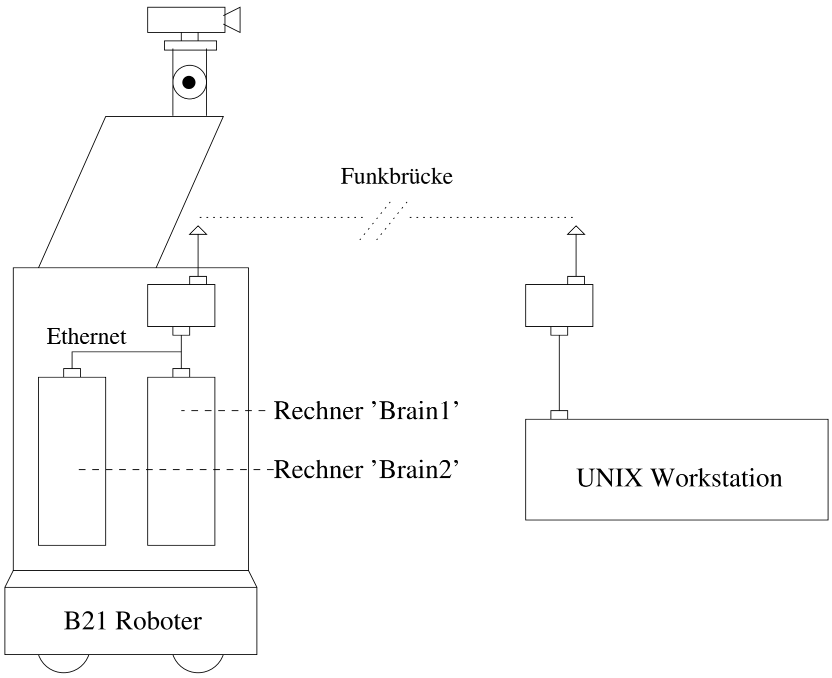


Abbildung 6: *Eine schematische Darstellung der einzelnen Hardwarekomponenten. Die beiden an Bord befindlichen Rechner sind über ein lokales Netzwerk verbunden. Eine Funkbrücke stellt die Verbindung zu einer Unix-Workstation her.*

Die Workstation wird zur Visualisierung und Entwicklung benötigt. Zur Steuerung des Roboters werden lediglich die beiden an Bord befindlichen Rechner verwendet. Ein- und Ausgaben zur Laufzeit werden durch Bedienelemente und einen Sprachgenerator am Roboter durchgeführt. Die nächsten Abschnitte beschreiben den Roboter und die damit verbundene Hardware genauer.

[5] Dabei handelt es sich um eine Ultra Sparc I Workstation von SUN ohne spezielle Grafikhardware.

3.1.1 Der Roboter

Es wird ein B21-Roboter von Real World Interfaces [32] und [33] verwendet (vgl. Abb. 7)[6]. Der Roboter hat einen Durchmesser von 52 cm und eine Höhe von 106 cm. Die gesamte Höhe mit Aufbau beträgt 182 cm. Das Gewicht des Roboters beträgt etwa 140 kg. Der Roboter besitzt ein 4-Achsen-Fahrwerk, welches ihm ermöglicht, sich auf der Stelle zu drehen. Die maximale Translationsgeschwindigkeit beträgt 1 Meter pro Sekunde. Eine Rotation kann mit maximal 160 Grad pro Sekunde erfolgen. Die Kameras auf dem Aufbau können zusätzlich durch die Pan-Tilt-Einheit gedreht und geneigt werden. Durch die Vergenzmotoren können die Kameras einzeln rotiert werden.

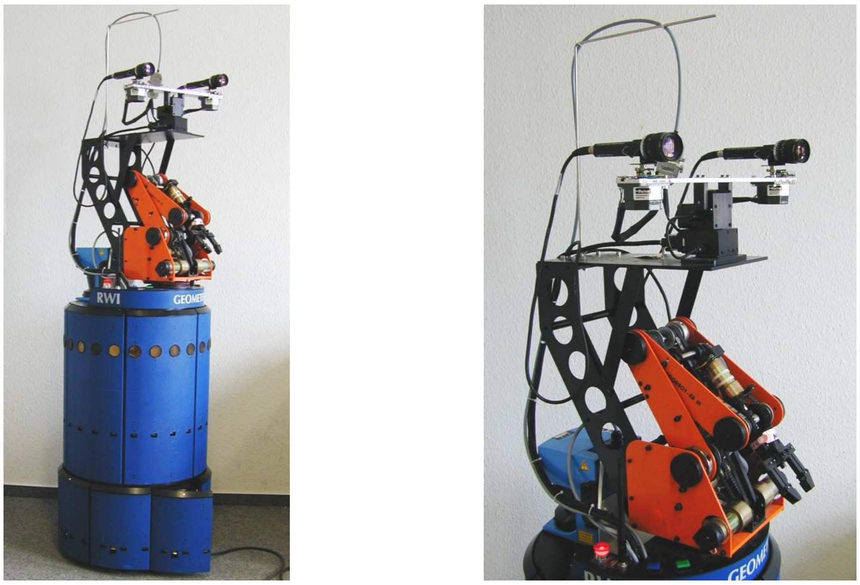


Abbildung 7: *Der B21-Roboter von Real World Interfaces. Rechts ist der Aufbau vergrößert abgebildet.*

3.1.2 Sensoren und Mechanik

Der B21-Roboter besitzt neben den beiden Kameras noch 48 Ultraschall-Sensoren, welche in zwei Ringen um den Rumpf des Roboters angeordnet sind. Die Ultraschall-Sensoren werden in dieser Arbeit nur benutzt, um Hindernisse zu detektieren und Kollisionen zu vermeiden. Weiterhin stehen 24 Infrarot-Sensoren zur Verfügung, welche am Roboter einen Kreis in Bodennähe bilden. Über den gesamten Rumpf sind 56 Tastsensoren verteilt, welche eine bereits erfolgte Kollision anzeigen können. Dieser Fall führt zu einem sofortigen Abschalten aller Motoren. Vier verschieden farbige Schalter stehen als Eingabequelle auf dem Aufbau des Roboters zur Verfügung. Zur Kommunikation mit dem Benutzer wird ein einfacher Sprachgenerator an Bord des Roboters verwendet.

[6] Die Schnittstelle zum Roboter wird vom Hersteller in [33] beschrieben. Darauf aufsetzenden Treiber sind in [20] sowie [25] genauer dokumentiert.

3.1.3 Interne Hardware zur Steuerung

Im Roboter werden die zwei Rechner *Brain1* und *Brain2* zur Steuerung verwendet. Beide bestehen aus handelsüblicher PC-Hardware[7], welche allgemein verfügbar ist. Die Rechner sind durch ein lokales Netzwerk verbunden. Als Betriebssystem wird auf beiden Rechnern Linux[8] verwendet. Die Controller der Vergenzen und der Pan-Tilt-Einheit werden jeweils getrennt über serielle Schnittstellen von *Brain1* angesprochen. In *Brain2* stehen zwei Framegrabberkarten[9] für die Digitalisierung der Kamerabilder zur Verfügung.

3.1.4 Die Kameras

Der Roboter ist mit zwei CCD-Farbkameras[10] ausgestattet. Sie sind auf jeweils einem Vergenzmotor montiert und können so einzeln geschwenkt werden. Die Kameras sind in Abb. 9 zu sehen. Die Bilder der Kameras werden über ein synchronisierendes Genlock-Interface durch jeweils eine Framegrabberkarte digitalisiert.

Die Kameras liefern Bilder in PAL-Auflösung (768×576 Pixel) und haben keinen Zoom. Bei der Verarbeitung der Bilder werden in dieser Arbeit allerdings Höhe und Breite der Kamerabilder durch Skalierung halbiert (384×288 Pixel). Die Brennweite lässt sich nur manuell regeln und ist bei allen Experimenten auf eine optimale Schärfe bei einem Abstand von 2 Metern eingestellt. Das Sichtfeld beträgt etwa 30 Grad. Ein Farbabgleich der Kamerabilder kann über Dipschalter konfiguriert werden. In den Experimenten werden zwei verschiedene Einstellungen benutzt, je nachdem ob künstliches oder natürliches Licht in der Umgebung überwiegt.

Es werden unweigerlich auch Bilder bei der Durchführung schneller Bewegungen aufgenommen. Dabei tritt der sogenannte *Lattenzauneffekt* durch das zeitversetzte Auslesen der geraden und ungeraden Bildzeilen des CCD-Chips der Kamera auf (vgl. Abb. 8).
Bei dem in Abschnitt 3.2.3 beschriebenen, farbbasierten Verfahren ist es nicht relevant, ob Kanten und klare Ränder von Objekten bestehen bleiben, solange die gesuchten farblichen Zusammenhangskomponenten nicht zerteilt werden. Bei anderen - zum Beispiel nicht erscheinungsbasierten - Verfahren, kann dies zu Problemen führen. Die Beschleunigung bei Bewegungen wird so gering gehalten, wie es die zeitlichen Anforderungen der Problemstellung erlauben.

3.1.5 Die Vergenzen

Die *Vergenzen* bestehen aus zwei Schrittmotoren auf denen jeweils eine Kamera montiert ist. Die Verbindungslinie zwischen den optischen Zentren der beiden Kameras - die sogenannte *Baseline* - hat eine Länge von 29 cm (vgl. Abb. 9). Im Gegensatz zur Pan-Tilt-Einheit können durch die Vergenzmotoren beide Kameras unabhängig voneinander rotiert werden. Diese Vergenz-Einheiten wurden am Lehrstuhl für Kognitive Systeme in Kiel gebaut.

[7] Die Rechner sind mit Intel Pentium III 800 MHz Prozessoren ausgestattet.
[8] Es wird der Linux Kernel 2.0.29 mit einer RedHat Distribution verwendet.
[9] Die in [29] beschriebenen Meteor Framegrabberkarten von Matrox werden verwendet.
[10] Es handelt sich dabei um Sony CCD Color Video Camera Module XC-999/999P.

Abbildung 8: *Im linken Bild sind alle vertikalen Kanten stark verwischt, da jede zweite Zeile verschoben ist („Lattenzauneffekt"). Im rechten Bild standen die Kameras still.*

Die Motoren sind jeweils an Controller (siehe auch Hardwarebeschreibung [17]) angeschlossen, welche wiederum seriell von einem Rechner an Bord des Roboters gesteuert werden. Die Motoren selbst haben keine Positionsgeber. Die vom Controller vorgegebenen Positionsänderungen werden lediglich akkumuliert. Der jeweilige Motor kann daher keine Anfragen zur Position beantworten. Die Software zur Steuerung der Vergenzen rechnet die vorgegebenen Winkel in Schritte[11] des Motors um. Die Umrechnung zwischen Schritten und Winkeln gehört neben den mechanischen Ungenauigkeiten des Motors zu den möglichen Fehlerquellen, welche in 4.1.1 experimentell untersucht werden, da sie die Qualität der Entfernungsschätzung wesentlich bestimmen.

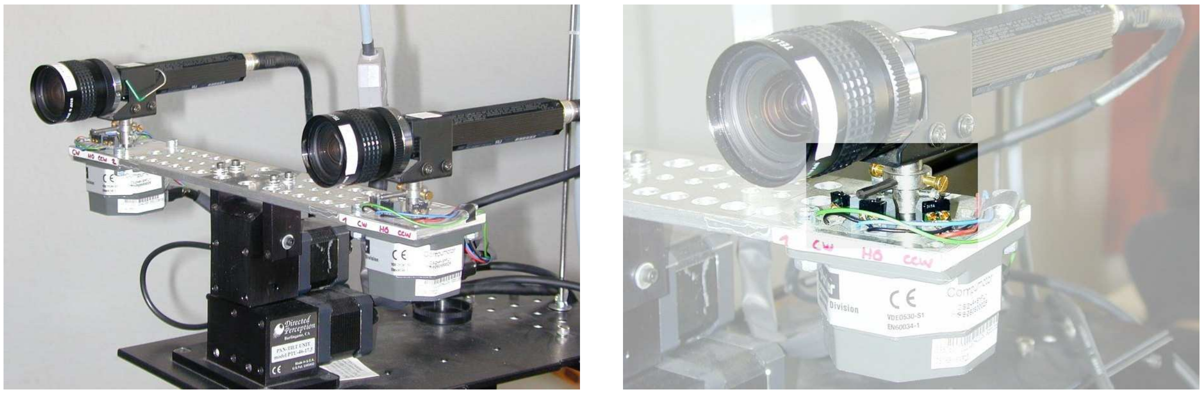


Abbildung 9: *Die Vergenzeinheit ist links als Ganzes zusammen mit der Pan-Tilt-Einheit zu sehen. Rechts sind die den Bewegungsbereich definierenden Schalter hervorgehoben.*

Es können jederzeit Bewegungsbefehle an den Controller geschickt werden. Geschieht dies in schnellerer Folge als diese abgearbeitet werden können, speichert der Controller die nicht abgearbeiteten Befehle in einer Warteschlange. Bewegungsbefehle werden in weniger als 50

[11] Eine Motordrehung entspricht 25000 Schritten.

Millisekunden angenommen und eine Unterbrechung der Bewegung ist zwar möglich, aber erfordert danach eine Positionsabfrage. Bis das Ergebnis einer Positionsabfrage vorliegt, vergehen 600 bis 1500 Millisekunden. Um möglichst wenige Positionsabfragen machen zu müssen und trotzdem schnell die Bewegungsrichtung ändern zu können, führen die Vergenzen kurze, schnelle Bewegungen durch. Diese sollen in einem Takt veranlasst werden, der sowohl immer eine Vollendung der Bewegung, als auch eine schnelle Reaktion erlaubt. Die Regelung der Vergenzen benötigt weiterhin eine zeitliche Übereinstimmung zwischen den Stellwinkel der Vergenzen und den Kamerabildern. Ist diese Übereinstimmung nicht gegeben (z.B. da der zeitliche Takt in dem Befehle an den Controller geschickt werden zu knapp ist), äußert sich dies durch ein Schwingen des Reglers. In Abschnitt 4.1.1 wird auf das experimentelle Ermitteln des Taktes eingegangen.

Dem Motor kann eine Beschleunigung und eine maximale Geschwindigkeit vorgegeben werden. Die maximale Geschwindigkeit kann bis zu 7.96 Umdr.$/s$ betragen. Die maximale Beschleunigung beträgt entsprechend 7.96 Umdr.$/s^2$. Bei den kurzen, schnellen Bewegungen der Vergenzen wird die maximale Geschwindigkeit so groß gewählt, dass sie aufgrund des kurzen Weges nicht erreicht werden kann. Der relevante Parameter bei den Experimenten ist damit lediglich die Beschleunigung.

Als *Nullstellung der Vergenzen* wird in dieser Arbeit die Motorstellung bezeichnet, bei der die optischen Achsen der Kameras senkrecht zur Baseline stehen. Diese Nullstellung und die äußeren Ränder des Bewegungsbereichs der Vergenzen werden jeweils durch Tastschalter markiert (vgl. Abb. 9). Der Schalter für die Nullstellung wird bei der Initialisierung der Vergenzen benutzt. Die Kameras werden langsam von einer Seite an den Schalter heran bewegt bis dieser aktiviert wird. Diese Stellung wird dann vom Controller als Nullstellung registriert. Dies ist zwar sehr gut reproduzierbar, aber durch die Ungenauigkeiten der Montage der Schalter stehen die optischen Achsen nicht genau senkrecht zur Baseline. Die *genaue Nullstellung* wird durch die in Anhang B beschriebene Kalibrierung für die Untersuchung der Entfernungsschätzung bestimmt. Die dabei erreichte Präzision wird in Abschnitt 4.2 untersucht.

3.1.6 Die Pan-Tilt-Einheit

Die Pan-Tilt-Einheit[12] kann das gesamte Stereokamerasystem (also beide Kameras inklusive Vergenzmotoren) schwenken und neigen. Sie ist in Abb. 9 links unter den Vergenzen sichtbar. Der Controller und die Motoren sind in [19] näher beschrieben.

Der Pan-Tilt-Einheit können unter anderem die Beschleunigung und die maximale Geschwindigkeit vorgegeben werden. Die maximalen Werte richten sich stark nach der montierten Nutzlast. Ein Test mit der Beschleunigung von 115 Grad$/s^2$ ergab hier bereits einen schnellen Verlust der Kalibrierung (vgl. Abb. 10). Die bereits montierte Nutzlast liegt an der oberen Grenze der Belastbarkeit der Motoren der Pan-Tilt-Einheit. Es können daher nur langsame Bewegungen mit niedriger Beschleunigung ausgeführt werden, wenn eine ständige Rekalibrierung vermieden werden soll.

[12]Es handelt sich um eine Directed Perceptions Pan-Tilt-Unit PTU-46-17.5.

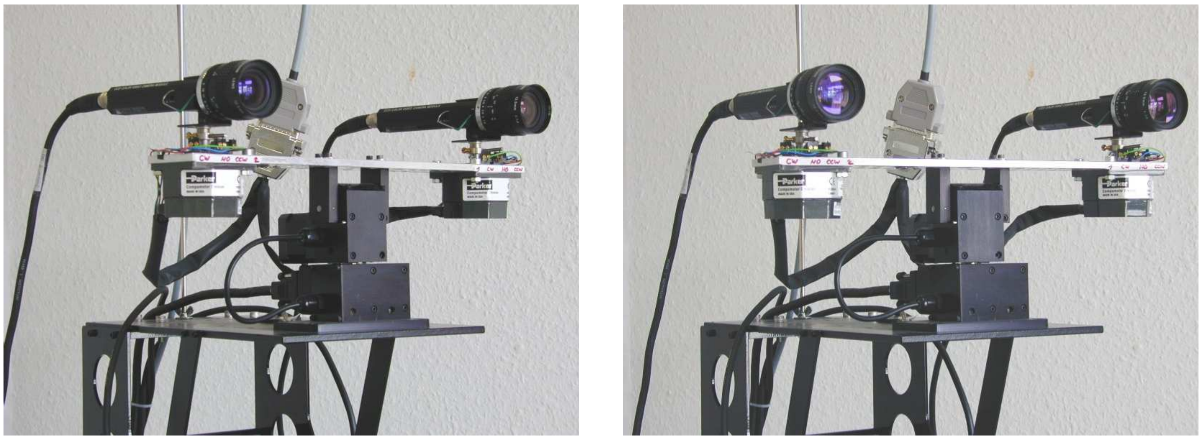


Abbildung 10: *Die Pan-Tilt-Einheit ist links vor und rechts nach lediglich zwei schnellen Bewegungen bei einer Beschleunigung von 115 Grad/s^2 zu sehen. In beiden Fällen meldet der Controller, dass sich die die Pan-Tilt-Einheit in Nullstellung befindet.*

3.2 Softwarebausteine

Das System ist in verschiedene Komponenten aufgetrennt und auf die zwei Rechner des autonomen Roboters verteilt. Die Aufteilung stellt eine Kapselung von Problemlösungen dar und sichert damit eine gewisse Wiederverwertbarkeit einzelner Softwarekomponenten. Weiterhin ist so eine Entkoppelung der Robotersteuerung und der Bildverarbeitung möglich, was insgesamt die begrenzten Ressourcen des Roboters schont. Zusätzliche Robustheit wird gewonnen, da Fehler einzelner Komponenten, wie der Ausfall eines Tracker-Prozesses, selbstständig durch kontrollierende Prozesse aufgefangen werden können. Intern findet die Kommunikation zwischen den Komponenten über das Netzwerk und über gemeinsamen Speicher statt. Dabei wird an zeitkrittischen Stellen wie der Bildverarbeitung, der Aufwand für die Kommunikation durch Verwendung von möglichst wenig Zugriffssperren minimiert.

In Abb. 11 sind die einzelnen Komponenten schematisch dargestellt. Die UNIX-Workstation war nur für die Entwicklung und das Auswerten der Versuche notwendig. Alle für das Funktionieren des Systems notwendigen Komponenten arbeiten auf den Rechnern *Brain1* und *Brain2* des autonomen B21-Roboters. Durch Prozesse auf *Brain1* wird die Hardware gesteuert, während auf *Brain2* der Prozess *B21Ctrl* die Steuerung dieser Hardware koordiniert. Für jede Kamera läuft eine Instanz der Tracker-Software, welche die Kamerabilder verarbeitet und die gewonnenen Informationen weitergibt. Im *Controller*-Prozess sind die einzelnen Verhaltensmodule realisiert, welche direkt auf die gewonnen Daten der Tracker-Prozesse und die Steuerung der Hardware mittels des Prozesses *B21Ctrl* zugreifen können. Auf die von den Tracker-Prozessen bearbeiteten Kamerabilder können auch vom System unabhängige Anwendungen zugreifen und z.B. zur Kontrolle an die UNIX-Workstation übertragen, oder anderweitig weiter analysieren. Der auf *Brain1* laufende *TCX Server* wurde vom Hersteller mit dem B21-Roboter geliefert. Die Tracker-Prozesse basieren auf der Arbeit von Krüger [26]. Die restliche Software wurde im Rahmen der vorliegenden Arbeit erstellt. Im Folgenden werden die zentralen Komponenten näher vorgestellt.

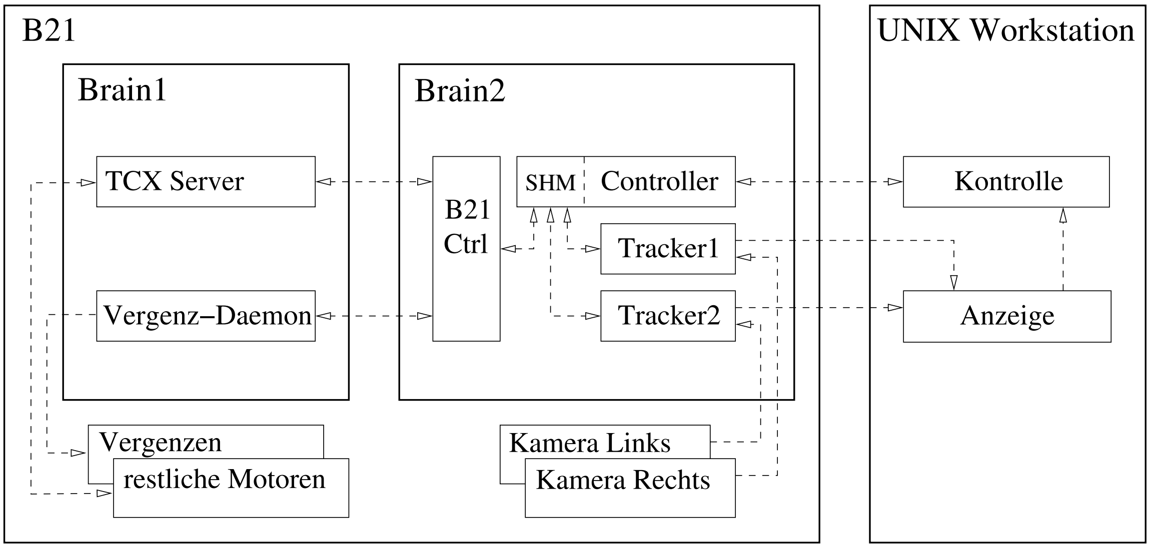


Abbildung 11: *Die schematische Darstellung zeigt die Verteilung der Software-Komponenten auf den Rechner und deren Kommunikationsstrukturen. Auf* Brain2 *ist der zur Kommunikation genutzte, gemeinsame Speicher (*SHM*) zu erkennen.*

3.2.1 Der Controller-Prozess

Der *Controller*-Prozess startet die restlichen Komponenten und stellt die zentrale Komponente des Systems dar. Im Falle eines internen Fehlers kann er einzelne Komponenten zur Laufzeit erneut starten und initialisieren[13].

Im *Controller*-Prozess sind die in Abschnitt 2 beschriebenen Verhaltensmodule implementiert. Sie können alle 50 Millisekunden agieren, wobei den Mechanismen zur Kommunikation zwischen den Modulen (Überdeckung, Unterdrückung und Signale) Rechnung getragen wird. Ein kürzerer zeitlicher Takt ist nicht notwendig, da die Bildverarbeitung nicht in schnellerer Folge Kamerabilder auswerten kann. Die einzelnen Verhaltensmodule sind als deterministische Automaten mit maximal 7 Zuständen realisiert.

3.2.2 Der Prozess B21Ctrl

Der Prozess *B21Ctrl* steuert die Hardware des Roboters. Dazu wird eine Netzwerkverbindung zu den Komponenten auf dem Rechner *Brain1* gehalten. *B21Ctrl* wiederum wird über eine einheitliche Schnittstelle auf dem Rechner *Brain2* kontrolliert. Diese Aufteilung erleichtert eine Übertragung der zentralen Komponenten auf andere Roboter. Der Prozess *B21Ctrl* übernimmt unter anderem die Kontrolle der Ultraschall- und Tastsensoren und stoppt alle Motoren im Falle einer Kollision oder der Detektion eines sehr nahen Hindernisses. Dies ist als eine Art Reflex implementiert und bedarf keines ausdrücklichen Befehls.

[13]Ein Laufzeitfehler in einer anderen Komponente wird dabei durch das Fehlen von aktuellen Zeitstempeln im gemeinsamen Speicher detektiert.

3.2.3 Die Tracker-Prozesse

Für jede Kamera läuft eine unabhängige Instanz des Tracker-Prozesses, welche jeweils die Bilder verarbeitet und die Position der detektierten Person im jeweiligen Kamerabild über den gemeinsamen Speicher den anderen Softwarekomponenten zur Verfügung stellt. Die Tracker-Prozesse basieren auf einer Software zur Steuerung eines aktiven Kamerasystems für Telekonferenzen [26]. Diese stellt ein visuelles, verhaltensbasiertes System dar, welches als Wahrnehmungs-Handlungs-Zyklus realisiert ist. Mit Hilfe des Systems kann in einer Diskussionsrunde immer die jeweils sprechende Person fixiert und möglichst groß aufgenommen werden. Der in dieser Arbeit verwendete Tracker basiert auf einer Portierung auf das Betriebssystem Linux [13].

Funktionsweise des Tracking-Algorithmus

Das Detektieren von Personen geschieht über ein schnelles, rein farbbasiertes Verfahren. Dabei wird ein Farbbereich spezifiziert, der z.B. die typischen in einem Gesicht zu findenden Hautfarbtöne abdecken. Der Tracker-Prozess sucht in den Kamerabildern nach Pixeln dieser Färbung und bildet auf ihnen Zusammenhangskomponenten. Bei der Initialisierung wird eine Zusammenhangskomponente als die gesuchte Person gewählt. Dieser Zusammenhangskomponente wird in den Kamerabildern über die Zeit gefolgt. Dabei wird beachtet, dass sich weder Position noch Größe stark zwischen zwei aufeinander folgenden Bildern ändern können. Statt der Gesichtsfarbe kann auch die Farbe eines T-Shirts oder anderer Bekleidungsteile benutzt werden. Mit diesem schnellen Verfahren können mehr als 20 Bilder pro Sekunde und pro Kamera ohne Verwendung von spezieller Hardware nur durch die an Bord befindlichen Rechner verarbeitet werden.

Die Bilder werden in den HSI-Farbraum überführt, um dort Bereiche von Bildpunkten mit ähnlicher Farbe zu suchen. Wird eine Farbzusammenhangskomponente als die gesuchte Person erkannt, so braucht in den folgenden Bildern lediglich in einer Umgebung der aktuellen Position und nicht im gesamten Bild gesucht zu werden. Der Tracker ist als deterministischer Automat realisiert und wechselt zwischen verschiedenen Zuständen, je nachdem, ob eine *Initialisierung* nötig ist, ob im *gesamten Bild* eine Zusammenhangskomponente gewählt werden muss oder ob in einem *kleineren Fenster* die Komponente lediglich wiedergefunden werden muss.

Im gemeinsamen Speicher wird die Position, die Anzahl der Bildpunkte und die maximale Ausdehnung der ausgewählten Farbzusammenhangskomponenten in vertikaler und horizontaler Richtung abgelegt. Die Verhaltensmodule im *Controller*-Prozess können so die zeitliche Konsistenz der gewonnen Daten überprüfen und gegebenenfalls durch Befehle steuernd eingreifen. Ein Eingriff wäre zum Beispiel das Erzwingen eines Wechsels in einen bestimmten Zustand oder die Anpassung des gesuchten Farbbereiches.

4 Experimente

Das System wird stufenweise durch die Experimente untersucht. Abb. 12 zeigt, wie die Ergebnisse der einzelnen Versuche aufeinander aufbauen. Ziel ist eine Eingrenzung der möglichen Fehlerquellen und damit eine Anpassung geeigneter Parameter (unter anderem der Hardware) für ein lauffähiges Gesamtsystem, welches in Echtzeit[14] auf seine Umgebung reagieren kann.

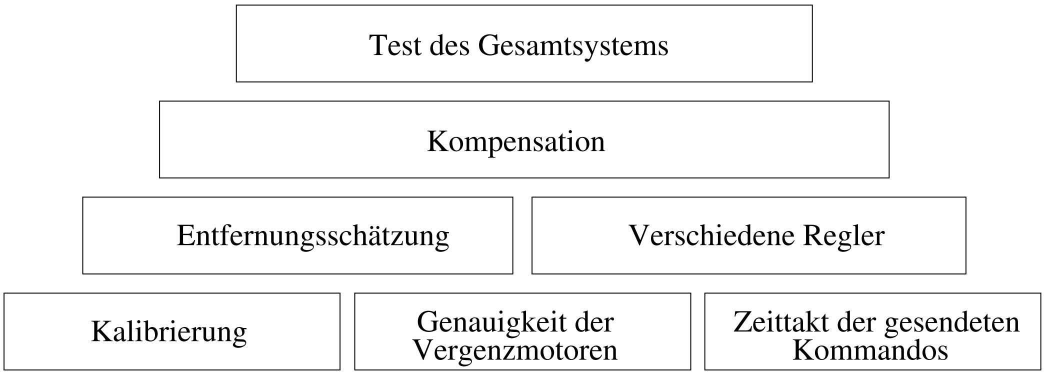


Abbildung 12: *In Experimenten wird das entwickelte System systematisch untersucht. Die Abbildung zeigt, wie die Ergebnisse der einzelnen Versuche aufeinander aufbauen.*

Als Basis für alle anderen Experimente wird die Genauigkeit der Vergenzsteuerung untersucht. Ebenso grundlegend ist die Bestimmung der Rotation der beiden Kameras zueinander, welche im Folgenden als Kalibrierung bezeichnet wird[15]. Darauf aufbauend können erst sinnvolle Aussagen über die Qualität der Entfernungsschätzung gemacht werden. Diese Kalibrierung (also die Bestimmung der Orientierung der beiden Kameras zueinander) ist nur für die Untersuchung der Entfernungsschätzung nötig. Zum Betrieb des Systems wird sie nicht gebraucht.

Weiterhin werden Experimente zur Taktfrequenz der Kommandos an die einzelnen Komponenten der Hardware durchgeführt. Dabei muss ein Kompromiss zwischen Geschwindigkeit und Stabilität gefunden werden (vgl. Abschnitt 3.1.5). In Abhängigkeit von der Genauigkeit der Vergenzen können verschiedene Regler zur Kontrolle der einzelnen Motoren betrachtet werden. Auf die Entfernungsschätzung und die Auswahl der Regler aufbauend wird schließlich die Kompensation von zwei bzw. von drei Achsen untersucht. Zuletzt wird die Fähigkeit des Gesamtsystems zur Lösung der gestellten Aufgabe betrachtet.

4.1 Basisversuche

Die Versuche der untersten Ebene von Abb. 12 werden als Basisversuche bezeichnet, da erst von ihren Ergebnissen ausgehend das restliche System sinnvoll untersucht werden kann. Im

[14] Gemeint ist die in 1.1 definierte Echtzeit. Das System soll einem Menschen folgen können, der sich für ihn normal bewegt und dabei keine besondere Rücksicht auf den Roboter nimmt.

[15] In Anhang B wird diese Art der Kalibrierung genauer vorgestellt und ausgeführt, warum es keiner Bestimmung der internen und externen Kameraparameter im Sinne einer *üblichen* Kalibrierung bedarf.

Folgenden wird die Genauigkeit der Vergenzen und der Zeittakt, mit dem Befehle an die einzelnen Motoren geschickt werden können, untersucht. Die Kalibrierung wird zusammen mit der Entfernungsschätzung in Abschnitt 4.2 betrachtet.

4.1.1 Die Genauigkeit der Vergenzeinheit

Wird ein Objekt mit beiden Kameras des Stereosystems fixiert, so kann anhand der Stellwinkel und dem bekannten Abstand der optischen Zentren der Kameras die Entfernung zum Objekt berechnet werden (vgl. Anhang A). Ist bereits die Steuerung der Vergenzen ungenau, kann nur schwer eine Aussage über die Qualität der Entfernungsschätzung gemacht werden. Daher ist eine Untersuchung der Fehlerentwicklung zwischen der angenommenen Odometrie und den tatsächlichen Stellwinkeln der Vergenzen wichtig. Außerdem müssen die Vergenzen in einer angemessenen Geschwindigkeit zu steuern sein um die schnellen, kurzen Bewegungen machen zu können, die nötig sind um das Objekt nicht aus dem Sichtbereich zu verlieren. Ein Kompromiss zwischen Geschwindigkeit und Genauigkeit unterliegt hierbei den Beschränkungen der Hardware. Ziel ist die Findung von Parametern, die eine ausreichende Geschwindigkeit bei handhabbarem Fehler erlauben.

Aufbau und Ablauf

Im Versuch wird ein äußerer Taster (vgl. Abb. 9) der Vergenz *eingemessen*, indem sich die Vergenz diesem mit kleinen Schritten nähert. Beim Aktivieren des Schalters wird eine Positionsabfrage durchgeführt.

Das Einmessen des Tasters findet am Anfang des Versuches und nach einer Anzahl kurzer, zufälliger, schneller Bewegungen statt. Ungenauigkeiten der Motormechanik und akkumulierte Rundungsfehler schlagen sich in einer Differenz der zu Beginn und am Ende gemessenen Positionen nieder. Die Beschleunigung der Vergenzen wurde bei den einzelnen Versuchsdurchläufen variiert.

Messwerte

Die Vergenz bewegt sich in Schritten von 0.057 Grad an den äußeren Schalter heran. Die zufälligen Bewegungen finden in einem Bereich von 17.2 Grad statt. Bei jeder Bewegung kann sich die Vergenz sowohl vor, als auch zurück bewegen. Die Weite der Bewegung ist ebenfalls zufällig, aber durch den Bereich begrenzt.

Es werden 32 Durchgänge mit 1000 bzw. 10000 Bewegungen zwischen den beiden Messungen durchgeführt. Die Beschleunigung wurde bei den einzelnen Versuchen in einem Bereich von 115 bis 5156 Grad$/s^2$ variiert. Das entspricht 0.3 bis 14.3 Umdr.$/s^2$. Die Differenz zwischen den beiden Messungen bleibt immer unter der Schrittweite von 0.057 Grad. Daher kann über die tatsächlich entstandenen Fehler lediglich ausgesagt werden, dass sie kleiner als die Schrittweite sind.

Es kann festgestellt werden, dass die Motoren der Vergenzen für ihre Last (die jeweilige Kamera) ausreichend stark ausgelegt sind. Auch bei hohen Beschleunigungen bleibt der Schrittmotor exakt. Aufgrund der Genauigkeit der Bewegung muss die Beschleunigung nicht beschränkt werden. Lediglich das Schwanken des Aufbaus des B21 bei hoher Beschleunigung spricht gegen die Verwendung der maximalen Beschleunigung, da die Qualität der Kamerabilder langsam abnimmt[16].

4.1.2 Der Zeittakt der Vergenzsteuerung

Stammen die Stellwinkel der Vergenzmotoren und die aus den Kamerabildern gewonnen Positionen des Objektes nicht vom gleichen Zeitpunkt, so wird die Regelung instabil. Dies ist zum Beispiel der Fall, wenn für die Regelung die Zielwerte einer Vergenzbewegung zusammen mit Bildern verwendet werden, die vor dem Erreichen der Zielposition aufgenommen wurden. Das System wird instabil und hört unter Umständen nicht auf zu Schwingen. Für ein Funktionieren der Regelung muss daher nach dem Erteilen eines Bewegungsbefehls gewartet werden, bis die in den Kamerabildern detektierten Positionen des Objektes vom Zeitpunkt der Beendigung der Vergenzbewegung stammen. Außerdem dürfen sich die Bewegungsbefehle nicht im Hardware-Controller, wie in Abschnitt 3.1.5 beschrieben, aufstauen.

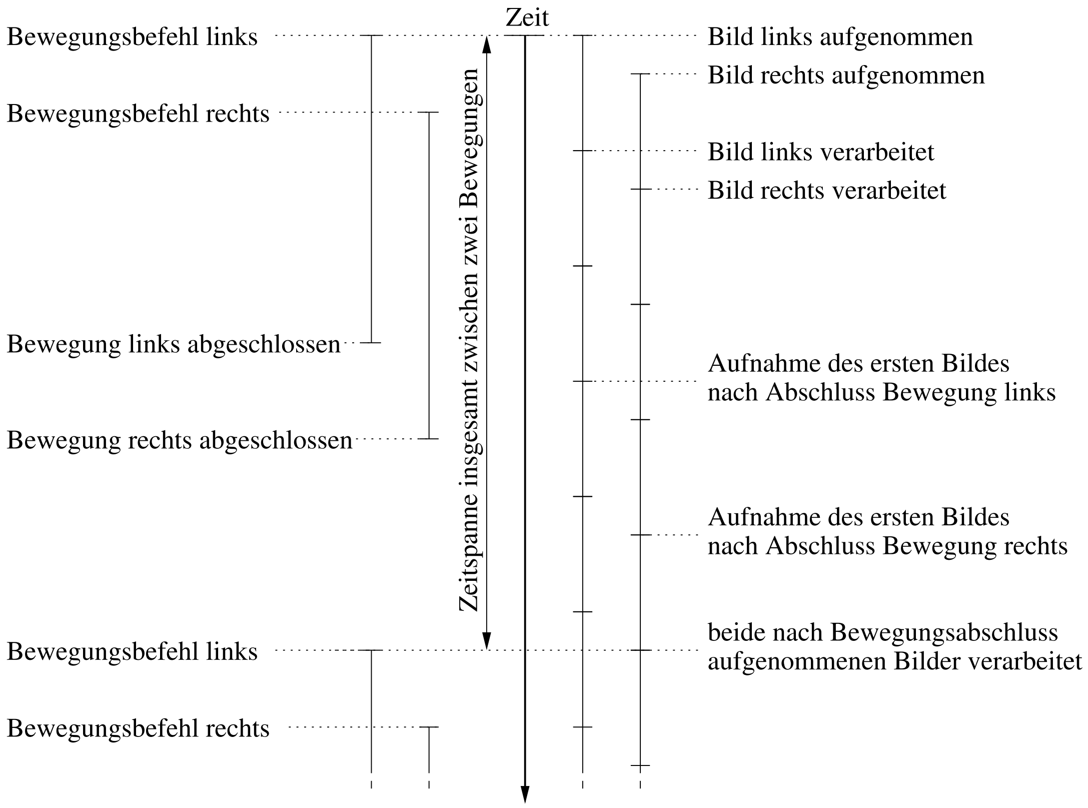


Abbildung 13: *Zeitlicher Ablauf der Vergenzbewegungen.*

In Abb. 13 ist die Wartezeit zwischen zwei Vergenzbewegungen dargestellt. Die Befehle zur

[16]Siehe Abb. 8 als Beispiel für eine gestörte Aufnahme.

Bewegung werden an die linke und rechte Vergenz gegeben[17]. Nach einer gewissen Zeitspanne haben beide Vergenzen ihr Ziel erreicht. Dann muss gewartet werden, bis die aus den Kamerabildern detektierten Positionen vom Zeitpunkt nach der Vollendung der Bewegung stammen.

Diese Zeitspanne wird im folgenden Experiment ermittelt. Um sicher zu gehen, dass diese Wartezeit ebenfalls dazu ausreicht, dass keine Warteschlangen im Hardware-Controller entstehen, wird ein zweites Experiment durchgeführt. Es werden viele Bewegungsbefehle in Folge abgeschickt, um den Zeitpunkt des Aufbaus einer Warteschlange im Controller zu bestimmen. Dazu wird nach der Durchführung von 50 Bewegungen in schneller Folge die Verzögerung beim Eintreffen am Ziel der letzten Bewegung gemessen. Bei insgesamt 122 Versuchsläufen wurden die Schrittweiten und die Beschleunigung variiert. Eine Warteschlange im Controller bildete sich nur bei Wartezeiten unterhalb von 300 Millisekunden zwischen den einzelnen Bewegungen.

Aufbau und Ablauf

In diesem Versuch soll die zeitliche Spanne zwischen dem Erteilen eines Bewegungsbefehls an die Vergenzen und der Verfügbarkeit von Positionsdaten des detektierten Objektes nach Vollzug der Bewegung ermittelt werden.

Der Ablauf einer Messung sieht wie folgt aus: Die Vergenzen werden in die durch einen Taster gekennzeichnete Ausgangsposition gebracht. Eine Marke wird im Bild detektiert und deren Position gespeichert. Danach wird die Vergenz um eine definierte Schrittweite bewegt. Nun wird die Messung der Zeitspanne gestartet und der Befehl zur Rückkehr in die Ausgangsposition gegeben. Die Beendigung der Bewegung wird durch den Tastschalter signalisiert. Die Zeit wird bis zu dem Punkt gemessen, an dem die Marke wieder auf der gespeicherten Position detektiert wird. Bei den einzelnen Versuchsläufen werden die Schrittweite und die Beschleunigung variiert.

Die Abfrage der Taster für die Feststellung des Erreichens des Zieles kostet Zeit und kann die Messung geringfügig verfälschen. Das Abfragen der Taster und die Kontrolle der detektierten Positionen fanden in einem Rhythmus von 10 Millisekunden statt. Dies stellt damit die maximale Auflösung der Ergebnisse dar.

Messwerte

Es werden Schrittweiten von 5.73, 11.46 und 17.19 Grad bei Beschleunigungen von 1.6 bis 7.96 Umdr.$/s^2$ untersucht. Größere Schrittweiten sind bei der Regelung nicht sinnvoll, da ein Objekt in einem Meter Entfernung bei einer Bewegung um 17 Grad bereits über das gesamte Kamerabild verschoben wird. In Tab. 1 sind die Ergebnisse aufgeführt.
Die Werte entsprechen den intuitiven Erwartungen und können als Basis zur Feinabstimmung des Systems dienen. Aus den Messwerten fallen nebenbei noch weitere Ergebnisse ab: Die

[17] Der Controller der Vergenzen kann diese Befehle nur nacheinander annehmen. Da diese Befehle über serielle Leitungen übermittelt und weitergegeben werden, ist die Zeitspanne zwischen dem Einsetzen der Bewegung des linken und des rechten Vergenzmotors von Bedeutung.

Schrittweite in Grad	*Wartezeit in ms*	*Beschleunigung in Umdrehungen$/s^2$*				
		1.6	3.18	4.77	6.37	7.96
5.72	Mittel	489.2	445	440.7	414.7	381.75
	Maximum	560	510	500	460	441
11.46	Mittel	567.4	521.4	463.6	447.2	462
	Maximum	620	540	520	496	510
17.19	Mittel	618	568.8	498.6	489	490.4
	Maximum	660	640	590	543	520

Tabelle 1: *Die Wartespanne zwischen der Erteilung von Bewegungsbefehlen an die Vergenzen und dem Erreichen des Zieles wird gemessen. Der Tabelle liegen 161 Messungen zugrunde.*

Verzögerung bei Bewegungen zwischen linker und rechter Vergenz beträgt im Schnitt 30 Millisekunden. Die Spanne zwischen Empfang des Signals des Tasters und dem Detektieren der aktuellen Position liegt im Schnitt bei 60 Millisekunden.

Bei einer Beschleunigung von 7.96 Umdr.$/s^2$ und einer Schrittweite von 5.73 Grad reicht in allen Versuchsläufen eine Wartezeit von 441 Millisekunden aus, damit nach dem Erteilen von Bewegungsbefehlen an die Vergenzen, die aktuellen detektierten Positionen vom gleichen Zeitpunkt wie die Vergenzwinkel stammen. Eine höhere Schrittweite von 17.19 Grad kann dabei mit einer geringfügig längeren Wartezeit von 520 Millisekunden erkauft werden. Kürzere Wartezeiten sind möglich, wenn gelegentliche Fehler in Kauf genommen werden. Es muss zwischen der zur Verfolgung des Objektes nötigen Geschwindigkeit und der Stabilität der Regelung abgewogen werden. Eine Variation der Wartezeiten auf Basis dieser Ergebnisse findet daher nochmal in Abschnitt 4.4 bei der Untersuchung der Kompensation statt.

4.1.3 Der Zeittakt bei der Steuerung von Pan-Tilt und der Basis

Die Pan-Tilt-Einheit und die Basis werden wie die Vergenzen geregelt, allerdings ist dort eine hohe Geschwindigkeit weniger entscheidend, da nur kompensatorische Bewegungen ausgeführt werden sollen. Das Fixieren des Objektes findet primär mit den Vergenzen statt. Wiederum ist wichtig, dass bei der Regelung die Stellwerte der Motoren bekannt sind. Ähnlich wie im vorherigen Abschnitt wird dazu die Zeitspanne zwischen Erteilung des Befehls und dem Abschluss der Bewegung untersucht.

Aufbau und Ablauf

Im Gegensatz zum Versuch im vorherigen Abschnitt wird die Beendigung der Bewegung hier nicht durch das Auslösen eines Tasters signalisiert. Es kann lediglich die Zeitspanne bis zum Wiederfinden der vorher detektierten, visuellen Marke an der gleichen Position in den Kamerabildern gemessen werden. Bei den Versuchsläufen wurden die Schrittweite und die Beschleunigung variiert.

Messwerte

Die Ergebnisse sind in den Tab. 2 und 3 verzeichnet. Erwartungsgemäß bewirkt eine hohe Beschleunigung erst ab einer bestimmten Schrittweite eine deutliche Verringerung der Wartezeit. Bei der Durchführung der Versuche fällt auf, dass eine kleine Schrittweite in Kombination mit großer Beschleunigung starke Odometrie-Fehler erzeugt. Da die Genauigkeit sowohl der Pan-Tilt-Einheit als auch der Basis keinen besonderen Einfluss auf die Lösung der gestellten Aufgabe haben, sollen sie hier nicht weiter untersucht werden[18]. Die gewonnen Werte dienen als Basis für die Feinabstimmung der Regelung und des Gesamtsystems.

Schrittweite in Grad	*Wartezeit in ms*	*Beschleunigung in Grad/s^2*				
		100	150	200	250	300
5	Mittel	1002.8	847.4	736.6	728.4	704
	Maximum	1079	960	859	800	760
10	Mittel	1160	1176	1019.2	939.4	832
	Maximum	1200	1210	1081	979	1000
20	Mittel	1663.8	1468	1295	1204	1080
	Maximum	1760	1520	1400	1340	1240
30	Mittel	1995.6	1764	1580	1464	1231.8
	Maximum	2120	1900	1660	1620	1280

Tabelle 2: *Die Wartespanne zwischen der Erteilung von Bewegungsbefehlen an die Basis und dem Erreichen des Zieles wird gemessen. Der Tabelle liegen 103 Messungen zugrunde.*

Schrittweite in Grad	*Wartezeit in ms*	*Beschleunigung in Grad/s^2*		
		34.38	51.57	68.75
5.72	Mittel	496	474.3	488
	Maximum	540	440	580
11.46	Mittel	722	702.4	583.1
	Maximum	800	860	720
17.19	Mittel	870	846	732
	Maximum	1020	960	900

Tabelle 3: *Die Wartespanne zwischen der Erteilung von Bewegungsbefehlen an die Pan-Tilt-Einheit und dem Erreichen des Zieles wird gemessen. Der Tabelle liegen 100 Messungen zugrunde.*

[18] Dies gilt natürlich nur, solange keine Rekalibrierung nötig ist. Bei hoher Beschleunigung der Pan-Tilt-Einheit ergeben sich Probleme (vgl. Abb. 10).

Aus den Basisversuchen gewonnene Parameter

Aus den Versuchen können geeignete Parameter für die Steuerung der Hardware abgeleitet werden. Die Vergenzen können auch bei hoher Beschleunigung genau gesteuert werden. Dabei kommt die Regelung in den zugrunde liegenden Versuchsläufen mit einer Wartezeit von 441 bzw. 520 Millisekunden zwischen den einzelnen Bewegungsbefehlen aus; je nachdem, ob eine größere Schrittweite vorteilhaft erscheint. Die Pan-Tilt-Einheit und die Basis werden langsamer rotiert, da sie nur kompensatorische Bewegungen durchführen sollen und bei größerer Beschleunigung rekalibriert werden müssen.

4.2 Entfernungsschätzung und Kalibrierung

Zur Untersuchung der Entfernungsschätzung durch die Vergenzen muss bekannt sein, in welchem Winkel die Kameras (bzw. deren optischen Achsen) zueinander stehen. Wie in Abschnitt 3.1.5 beschrieben, ist dieser Winkel grob bekannt, da die Nullstellung[19] durch Tastschalter markiert wird. Mit Hilfe des in Anhang A vorgestellten Verfahrens wird die genaue Stellung der Vergenzen ermittelt. Erst dann kann sinnvoll untersucht werden, wie genau die Entfernungsschätzung ist. Bei den folgenden Versuchen wird jeweils ein rotes Objekt als Landmarke auf eine definierte Position gestellt und dessen Schwerpunkt in beiden Kamerabildern bestimmt.

4.2.1 Fehler bei der Kalibrierung

Die Kalibrierung ist durch die unbekannte Lage der optischen Zentren der Kameras und durch Ungenauigkeiten beim Vermessen und beim Aufbau des Versuches immer mit einem Fehler behaftet. Um diesen Fehler einschätzen zu können, wird die Kalibrierung hier 72 mal mit verschiedenen Parametern durchgeführt. Die Entfernung des Objektes wird zwischen 25 cm und 150 cm variiert. Die Länge der Baseline beträgt entweder 4.5 cm oder 24.5 cm. Weiterhin wird der Versuch dabei mehrmals neu aufgebaut.

Bei einer Baseline von 4.5 cm beträgt die Varianz der Kalibrierungen 0.47 Grad. Eine Länge von 24.5 cm ergibt eine Varianz von 0.21 Grad. Bei der Schätzung der Entfernung eines einen Meter entfernten Objektes bedeutet ein Fehler von 0.21 Grad eine Fehlschätzung von 1.56 cm.

4.2.2 Die Genauigkeit der Entfernungsschätzung

Die zu fixierende Landmarke kann bei der Kalibrierung in verschiedenen Abständen zum Kamerasystem aufgebaut werden. Der Einfluss dieses Abstandes auf die Schätzung verschiedener Entfernungen zur Laufzeit des Systems wird im folgenden Experiment untersucht. Das Gesamtsystem soll in einer Entfernung von ein bis drei Metern die genauesten Schätzungen liefern, da dies etwa der Abstand zur verfolgenden Person sein soll. Die dafür geeigneten Parameter werden experimentell bestimmt.

[19] Als Nullstellung der Vergenzen wird die Motorstellung bezeichnet, bei der die optischen Achsen der Kameras genau senkrecht zur Baseline verlaufen.

Abbildung 14: *Links ist der gesamte Aufbau des Versuchs zu sehen. Im rechten Bild fixieren die Kameras den als Landmarke dienenden roten Quader.*

Aufbau und Ablauf

Es wird auf einer Fläche von 3 mal 4 Metern ein Raster aufgezeichnet, um die Marke an eine genau definierte Position setzen zu können. Die Vergenz-Einheit ist für diesen Versuch vom B21 getrennt worden und steht am Ursprung des Rasters (vgl. Abb. 14).

Das System wird fünf mal kalibriert. Die Marke steht dabei jeweils einmal in einer Entfernung von 40, 80, 120, 160 und 200 cm zum Kamerasystem. Danach wird die Marke in 3 Durchgängen jeweils an 24 Messpunkten von beiden Kameras fixiert. Aus den gewonnen Stellwinkeln der Vergenzmotoren ergeben sich damit $5 * 3 * 24 = 120$ Entfernungsschätzungen.

Messwerte

In Abb. 15 werden die relativen Schätzungsfehler bei verschiedenen Entfernungen der Landmarke vom Kamerasystem abgetragen. Die Kalibrierung auf 40 cm erzeugt einen deutlich höheren Fehler bei großen Entfernungen. Die Kalibrierung auf 200 cm ist dort am besten. Bei kurzen Entfernungen lassen sich nur wenige deutliche Unterschiede zwischen den verschiedenen Kalibrierungen erkennen.

In Abb. 16 wurden die Mittelwerte über die relativen Fehler von Entfernungsbereichen von je 50 cm abgetragen. Deutlicher als in Abb. 15 ist zu erkennen, dass die Schätzung bei steigender Entfernung zwar schlechter wird, sich aber eine Kalibrierung auf eine große Distanz deutlich positiv bemerkbar macht. In der Abbildung sind die Graphen im Bereich der kurzen Entfernungen vergrößert dargestellt. Hier kehrt sich das Verhältnis um. Bei der Schätzung von sehr kurzen Entfernungen schneidet die Kalibrierung auf 40 cm am besten ab. Es lässt sich folgende Schlussfolgerung ziehen: In dem Bereich, der bei der Bestimmung der Relation der Vergenzen zueinander benutzt wurde, fällt der Fehler bei der Entfernungsschätzung am geringsten aus.

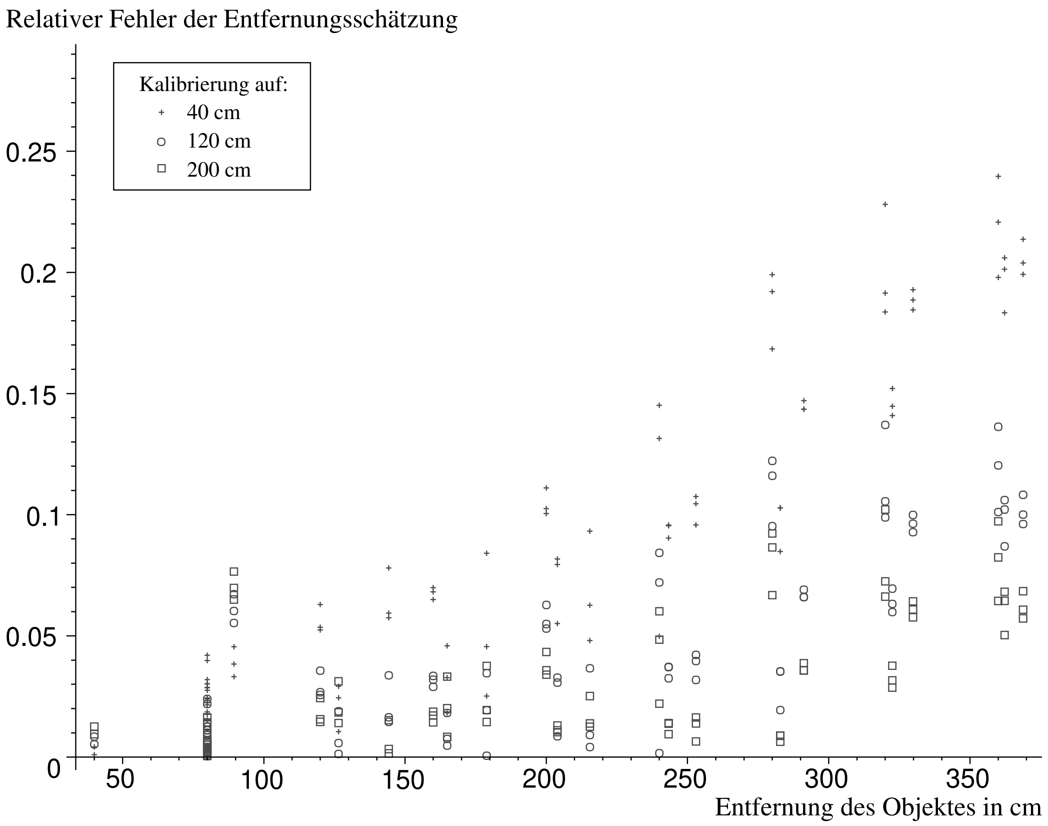


Abbildung 15: *Der relative Fehler der Schätzung bei verschiedenen Entfernungen ist abgetragen. Durch Kreuze, Kästchen und Kreise sind die jeweils verschiedenen, den Schätzungen zugrunde liegenden, Kalibrierungen dargestellt.*

Die gewonnenen Parameter

Der Fehler der Entfernungsschätzung bleibt bei jeder Kalibrierung unter 20 Prozent, was für die Anwendung in dieser Arbeit ausreichend ist. Falls eine Kalibrierung der Entfernungsschätzung vorgenommen werden soll, so bietet es sich an bei dieser Kalibrierung die Entfernung zu benutzen, die auch später am genauesten geschätzt werden soll.

In dieser Arbeit ist die Schätzung der Entfernungen zwischen einem und drei Metern am interessantesten. Bei einer Kalibrierung der Entfernungsschätzung bei zwei Metern blieb der Fehler in allen Versuchen unter 4 Prozent.

4.3 Regelung

Die Regelung hat hohe Bedeutung im realisiertem System, da mit ihrer Hilfe fehlerbehaftete Messwerte und schnelle Bewegungen der zu verfolgenden Person ausgleichen werden sollen. Dazu wird der Zyklus, in dem die Regelläufe stattfinden, möglichst klein gewählt. In Abschnitt 4.1.2 wurde untersucht, wie knapp dieser Zyklus gewählt werden kann. Um wirklich sicher zu gehen, dass Vergenzwinkel und die aus den Bildern gewonnenen Daten vom gleichen Zeitpunkt stammen, muss zum Teil mehr als eine halbe Sekunde gewartet werden. Bei einem knapperen

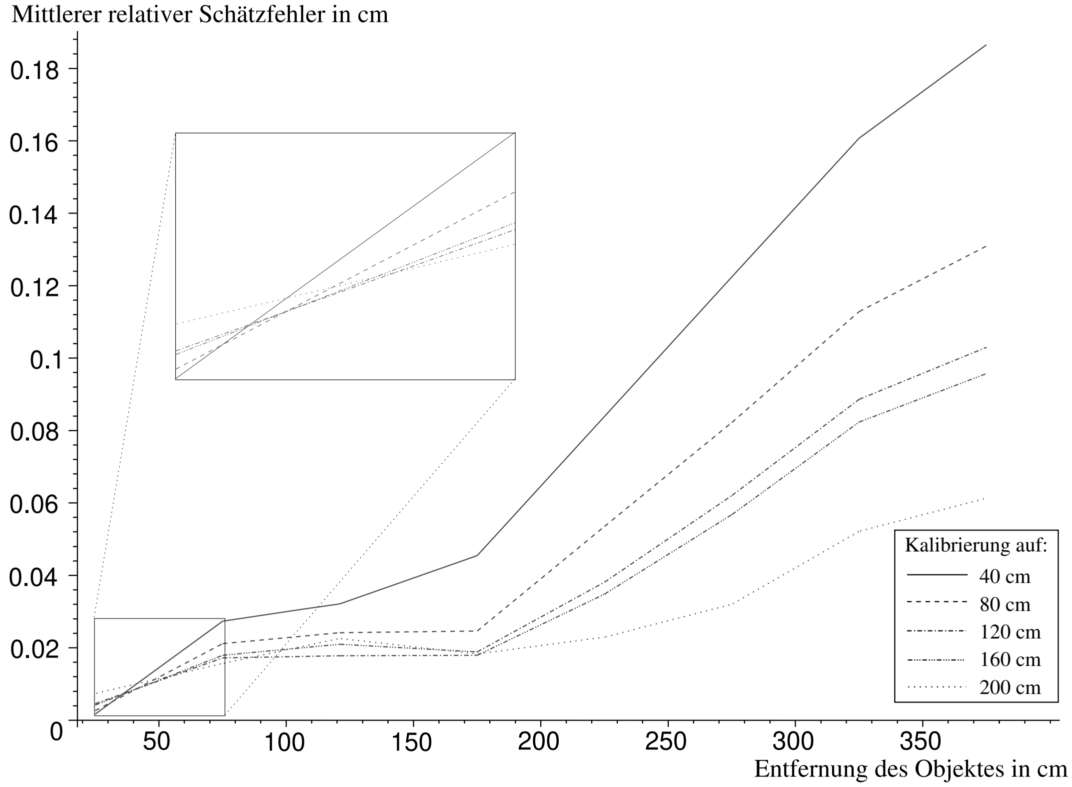


Abbildung 16: *Der mittlere relative Fehler (für Bereiche von* 50 *cm Breite) bei der Schätzung der jeweiligen Entfernung ist abgetragen. Die einzelnen Graphen stehen für verschiedene Kalibrierungen. Der Bereich der kurzen Entfernungen ist zusätzlich in Vergrößerung dargestellt.*

Zyklus müssen vereinzelte Fehler in Kauf genommen werden, die größere Anforderungen an die Regelung stellen.

In diesem Versuch werden P- und PI-Regler untersucht (vgl. Anhang D). Die Größe und die Gewichtung des Gedächtnis des Reglers werden variiert. Der P-Regler wird dabei als Spezialfall des PI-Reglers ohne Gedächtnis behandelt. Die Regler besitzen die Werte der Vergenzmotoren als Stellgrößen und die Koordinaten der in den Kamerabildern detektierten Person als Rückführgrößen. Der Sollwert ist immer die Bildmitte. Im Anhang D werden die verschiedenen Regler genauer vorgestellt. Ein wichtiges Kriterium zur Beurteilung der Qualität des Reglers ist seine Fähigkeit schnellen Bewegung folgen zu können.

Aufbau und Ablauf

Im Versuch wird eine einen Meter entfernte, feststehende Landmarke durch die Vergenzen fixiert. Das Pan-Gelenk vollführt eine langsame Bewegung um 30 Grad nach rechts, um dann (nach kurzer Wartezeit) eine schnelle Bewegung zurück in die Ausgangsposition zu vollziehen. Da das Pan-Gelenk in diesem Versuch nicht geregelt wird, kann in dieser Weise eine reproduzierbare Bewegung der Marke simuliert werden. Während des Versuches wird der zu untersuchende Regler benutzt, um die Marke mit Hilfe der Vergenzmotoren in der Bildmitte zu halten. Die Stellwinkel der Vergenzen, die geschätzte Entfernung und die detektierte Position der Landmarke in den Kamerabildern werden alle 50 Millisekunden gespeichert.

Bei den einzelnen Versuchsläufen wird die Größe des Gedächtnis und dessen Skalierung variiert. Die Regelung findet in diskreten *Runden* statt und die Größe des Gedächtnis eines PI-Reglers wird somit ebenfalls in Runden angegeben. Mit Hilfe der Skalierung wird den zeitlich weiter zurückliegenden Werten eine geringere Gewichtung als den aktuellen Werten zugewiesen[20].

Die Darstellung der Messwerte

In Abb. 17 sind die Winkel der Vergenzen über die Zeit abgetragen. Die linke und rechte Vergenz sind jeweils als einzelner Graph eingetragen. In späteren Versuchen kommen noch der Pan-Winkel und die Rotation der Basis hinzu. Ein Winkel von 0 Grad steht für die Ruhestellung und eine Ausrichtung des Gelenkes in Fahrtrichtung des Roboters[21]. Ein positiver Wert zeigt bei allen Achsen eine Rotation im Uhrzeigersinn an. Wie in Abschnitt 3.2 beschrieben, werden die Verhaltensmodule in einem Takt von 50 Millisekunden aktiviert. Dieser Zeitraum wird im Folgenden als eine Runde bezeichnet. Die Zeit wird auf der horizontalen Achse in Runden gezählt. Der Abstand der beiden Vergenzgraphen steht für den Winkel, den die beiden Vergenzen zueinander einnehmen. Fallen z.B. beide Vergenzwinkel ab, aber die Graphen bleiben dabei in konstantem Abstand zueinander, bedeutet das: Ein Objekt wird fixiert, welches sich in konstant bleibendem Abstand (vom Roboter aus gesehen) nach links bewegt.

Die Graphen der Vergenzwinkel scheinen sich sprunghaft zu ändern. Tatsächlich wird aber immer nur das Ziel der Bewegung zum Zeitpunkt der Erteilung des Befehls notiert, da nur dieser exakt messbar ist. Die realen Bewegungen des Roboters sind also weicher. Die langsame Bewegung des Pan-Gelenkes startet in Runde 50. Eine schnelle Rückbewegung findet ab Runde 100 statt. Die Zeitpunkte der Erteilung der Bewegungsbefehle sind in den Diagrammen jeweils durch eine gepunktete Linie markiert. Die tatsächlichen Bewegungen setzen darauf folgend ein. Das System reagiert sobald diese Bewegungen in den Kamerabildern detektiert werden und wiederum später sind die Auswirkungen auf die Entfernungsschätzung zu erkennen.

Die geschätzte Entfernung wird in Abb. 18 in Zentimetern für die jeweiligen Runden angegeben. Im Zuge der Bewegungen - speziell bei der zweiten, schnelleren - schwanken die Schätzungen der Entfernung. Ist ein Wert größer als 4 Meter oder von 0 Metern verzeichnet, erfolgt zu dem Zeitpunkt keine sinnvolle Schätzung.

[20]Konkret wird in der Implementierung ein Faktor angegeben, durch den die zurückliegenden Werte geteilt werden. Der Faktor wird dabei immer wieder mit sich selbst multipliziert. Mit einem Skalierungs-Faktor von 2 wird damit der aktuelle Wert nicht skaliert und der letzte Wert wird durch 2 geteilt. Der vorletzte Wert wird durch 4 geteilt, der davor durch 8 usw. Ein Faktor von 1 bedeutet, dass keine unterschiedliche Gewichtung vorgenommen wird.

[21]Damit sich die Ausrichtung des Gelenkes wirklich mit der Fahrtrichtung deckt, müssen sich natürlich auch alle anderen Gelenke in Ruhestellung befinden.

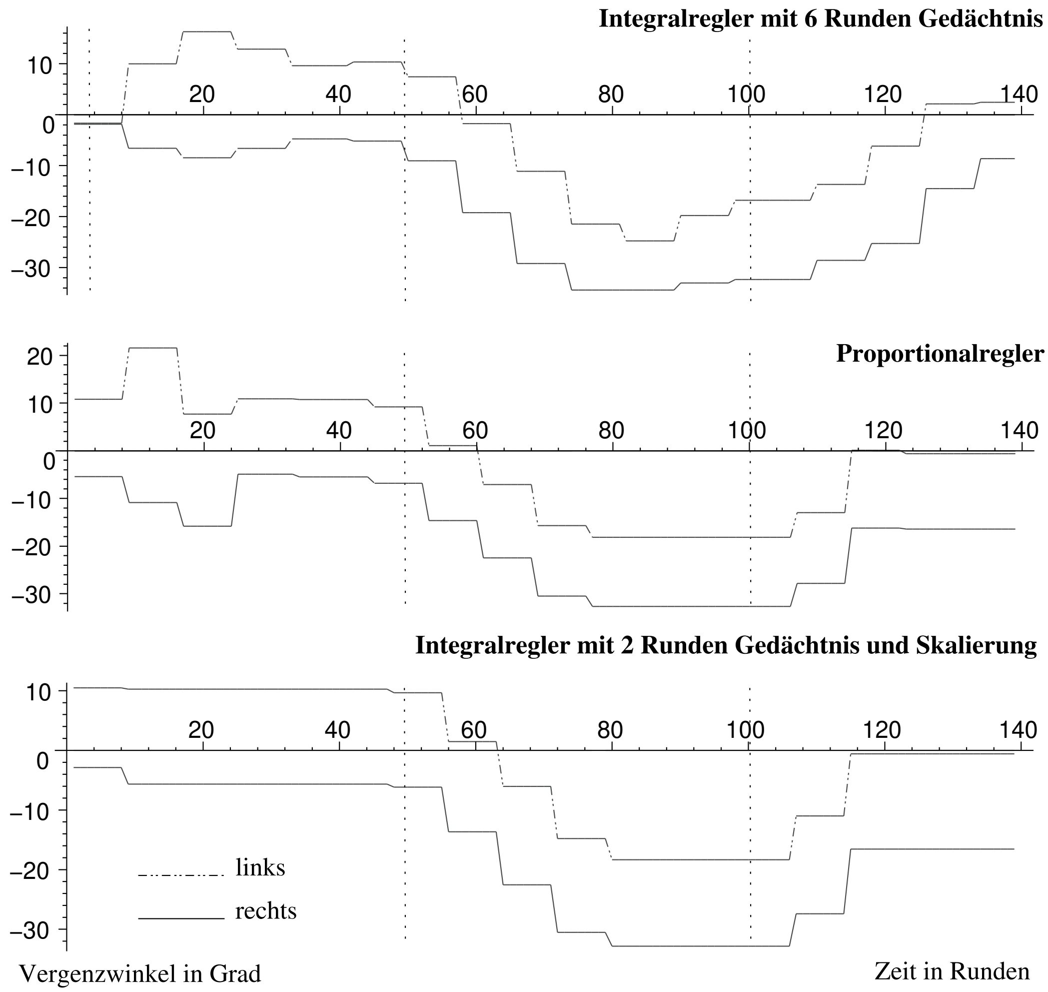


Abbildung 17: *Die generierten Vergenzwinkel von drei exemplarisch ausgewählten Reglern sind hier über die Versuchszeit abgetragen. Durch gepunktete Linien sind das Einsetzen der langsamen Bewegung in Runde* 50 *und der schnellen Rückbewegung ab Runde* 100 *angedeutet.*

Messwerte

Der Versuch wurde 32 mal mit einem Gedächtnis von 0 bis 8 Runden[22] und den Faktoren 1, 1.5 und 2 zur Skalierung durchgeführt. In Abb. 17 sind exemplarisch die im Laufe des Versuches generierten Winkel von drei verschiedenen Reglern abgetragen. Es werden ein P-Regler ohne Gedächtnis, ein PI-Regler mit einem Gedächtnis von 6 Runden und ein PI-Regler mit einem Gedächtnis von 2 Runden und einem Skalierungsfaktor von 2 dargestellt. Abb. 18 zeigt die dazugehörigen Entfernungsschätzungen.

Wird ein großes Gedächtnis benutzt, so wird der Regler erwartungsgemäß träge. Der erste dargestellte Regler mit 6 Runden zurückreichendem Gedächtnis lässt die Stellwerte der Vergenzen während des ganzen Versuches leicht schwingen. Dies ist in Abb. 17 an dem sich kontinuierlich ändernden Abstand der Graphen zueinander erkennbar. Wie in Abb. 18 zu sehen schwankt

[22] Ein Regler mit Gedächtnis von 0 Runden stellt einen einfachen Proportionalregler dar.

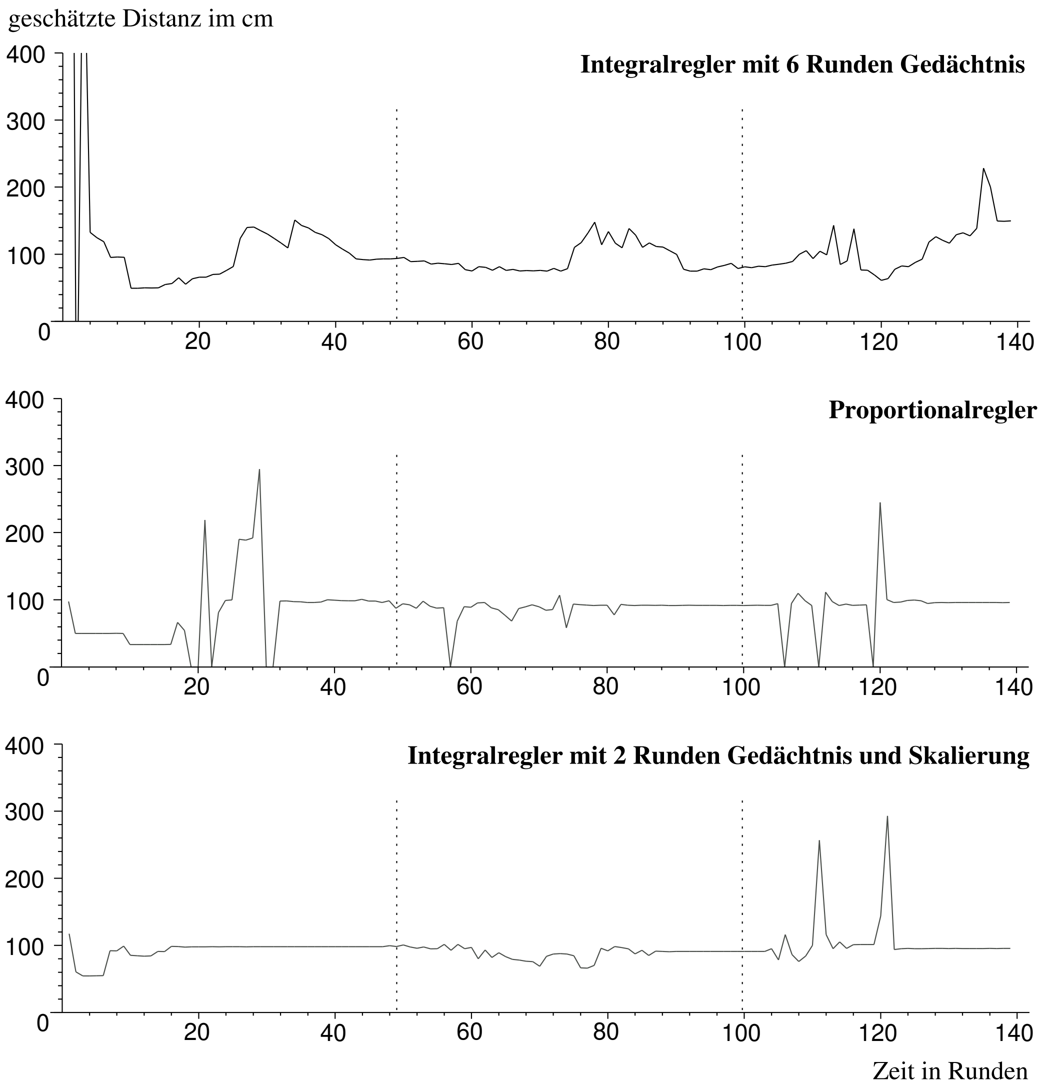


Abbildung 18: *Die im Laufe des Experiments geschätzte Entfernung ist jeweils für drei exemplarisch ausgewählte Regler dargestellt. Durch gepunktete Linien sind das Einsetzen der langsamen Bewegung in Runde* 50 *und der schnellen Rückbewegung ab Runde* 100 *angedeutet.*

dieser auch erheblich - wenn auch nicht abrupt. Die stark schwankenden Werte in den ersten Runden zeigen, dass anfänglich keine sinnvolle Entfernungsschätzung stattfindet. Vor allem die zweite, schnellere Bewegung wird nur sehr langsam verfolgt. Das Gedächtnis beeinträchtigt die Reaktionsfähigkeit des Reglers erheblich.

Der Proportionalregler lässt die Vergenzen in Abb. 17 am Anfang schwingen, aber regelt dann recht stabil. Die Entfernungsschätzung schwankt etwas im Laufe des Versuches. Dies ist z.B. beim Schwingen am Anfang des Versuches und im Laufe der beiden Bewegungen zu beobachten. Das Schwanken ist sehr abrupt, wie das Fehlen jeglichen Gedächtnisses erwarten lässt. Der Regler reagiert schnell auf die Bewegungen.

Beim PI-Regler mit 2 Runden Gedächtnis und einem Skalierungsfaktor von 2 hat das Gedächtnis den geringsten Einfluss aller im Experiment getesteten Regler[23]. In Abb. 17 ist lediglich ein sehr kurzes Einschwingen am Anfang zu bemerken. Die geschätzte Distanz zur Marke bleibt in Abb. 18 recht stabil. Nach dem Einschwingen am Anfang stabilisiert sich die Schätzung und schwankt nur leicht im Laufe der langsamen Bewegung ab Runde 50, was auf den geringen Einfluss des Gedächtnisses zurückzuführen ist. Der Ausgleich der starken Pan-Bewegung nach Runde 100 führt zu abruptem Schwanken der geschätzten Distanz. Die Entfernungsschätzung liefert den ganzen Versuchslauf über sinnvolle Werte. Wie beim Proportionalregler ohne Gedächtnis ist die Reaktionsfähigkeit gut.

Die gewonnenen Parameter

PI-Regler mit größerem Gedächtnis erweisen sich im Versuch klar als zu träge und ungeeignet. Gar kein Gedächtnis bei der Regelung zu beachten, führt beim reinen P-Regler zu abrupten Fehlern der Entfernungsschätzung. Unter den PI-Reglern mit kurzem Gedächtnis sind diejenigen mit niedriger Gewichtung am erfolgreichsten. In allen folgenden Versuchen werden nur noch diese Regler verwendet.

4.4 Kompensation

Die Kompensation ergibt sich aus den Reaktionen mehrerer Verhaltensmodule auf deren Wahrnehmung der Umwelt. Dabei existieren verschiedene (in Abschnitt 2.3 vorgestellte) Möglichkeiten zur Koordination der Module und eine Reihe von Parametern, die variiert werden können. Das sich ergebene Gesamtverhalten des Roboters ist schwer aus der Betrachtung der einzelnen Verhaltensmodule absehbar. Dies macht die Untersuchung der Kompensation besonders wichtig und interessant. Im Folgenden wird erst die doppelte Kompensation und darauf aufbauend die dreifache Kompensation untersucht. Bei der doppelten Kompensation steht noch die Findung geeigneter Parameterkombinationen im Vordergrund. Bei der Untersuchung der dreifache Kompensation werden die unterschiedlichen, sich ergebenden Gesamtverhalten verglichen. Als Kriterien werden bei doppelter und dreifacher Kompensation herangezogen:

Qualität der Entfernungsschätzung: Wie zuverlässig wird die Entfernung geschätzt, und wie anhaltend sind die Fehlschätzungen?

Reaktionsgeschwindigkeit: Wie schnell kann das System den Bewegungen folgen? Können unvorhergesehene Änderungen der Bewegungsrichtung verfolgt werden?

Die subjektive *Ruhe* des Systems: Wieviele unnötige Bewegungen finden statt, und wie weit werden die einzelnen Freiheitsgrade ausgelenkt?

4.4.1 Doppelte Kompensation

Die Untersuchung der doppelten Kompensation *und* der dreifachen Kompensation begründet sich in der Vielzahl der Parameter die Einfluss nehmen. Die vorhergehenden Versuche geben

[23] Natürlich bis auf den gedächtnislosen P-Regler.

zwar klare Anhaltspunkte zur Wahl einiger Parameter, aber machen keine Aussage, wie die Parameter sich gegenseitig beeinflussen. Es werden hier sinnvolle Kombinationen von Parametern bestimmt, die in den folgenden Versuchen Verwendung finden werden. Weiterhin kann dadurch die Effizienz der dreifachen und der doppelten Kompensation verglichen werden.

In den Experimenten zur doppelten Kompensation wird wie im vorhergehenden Versuch eine der Achsen des Roboters benutzt, um eine reproduzierbare Bewegung zu simulieren. Es wird hier das Zusammenspiel von Pan und Vergenz untersucht, wobei die Rotation der Basis für die besagte Simulation zur Verfügung steht[24].

Aufbau und Ablauf

Es finden 32 Versuchsläufe statt, in denen eine ein Meter entfernte, feststehende Landmarke durch die Vergenzen fixiert wird. Die Stellwinkel der Vergenzen und der Pan-Einheit sowie die geschätzte Entfernung zur Marke werden rundenweise gespeichert; eine Runde dauert 50 Millisekunden. In den Diagrammen werden entweder die Stellwinkel oder die geschätzte Entfernung zur Marke über den Zeitraum von 250 Runden (also 12.5 Sekunden) abgetragen. In den Versuchsläufen wird die Basis in Runde 40 mit einer Beschleunigung von 200 Grad$/s^2$ und einer maximalen Geschwindigkeit von 30 Grad$/s$ um 40 Grad rotiert. Direkt nach Beendigung der Bewegung wird sie in Runde 100 mit etwas höherer Geschwindigkeit (50 Grad$/s$) wieder in die Ausgangsstellung gebracht. Die erste Rotation stellt sich als eine Bewegung mit einer Geschwindigkeit von $0.524\ m/s$ der tatsächlich feststehenden Marke dar. Die direkt folgende Rückbewegung erfolgt mit $0.872\ m/s$. Die Zeitpunkte der Erteilung der Bewegungsbefehle ist in den Abbildungen durch eine gestrichelte Linie markiert: Bei *(i)* wird der Befehl zur ersten Bewegung gegeben. Bei *(ii)* wird die Rückbewegung initiiert; (vgl. z.B. Abb. 19). Sowohl das Pan-Gelenk als auch die Vergenzen werden über die jeweiligen Verhaltensmodule gesteuert, um der detektierten Bewegung in den Kamerabildern zu folgen und die Landmarke kontinuierlich zu fixieren. Es ergibt sich also eine Verzögerung zwischen der Erteilung des Befehls und der Reaktion des Systems.

Wahl der Parameter

In den 32 Versuchsläufen werden folgende Parameter variiert:

Beschleunigung: Die Vergenzen werden bei allen Läufen mit 7.96 Umdrehungen$/s^2$ beschleunigt. Die Beschleunigung der Pan-Einheit variiert zwischen 34.38 und 114.59 Grad$/s^2$. Die tatsächlich erreichte Geschwindigkeit ergibt sich aus der gewählten Schrittweite.

Schrittweite: Die maximale Größe einer einzelnen Bewegung der Vergenzen und der Pan-Einheit werden unabhängig voneinander festgelegt. Sie liegen beide jeweils zwischen 5.73 und 11.46 Grad.

[24] Eine vollständige Reproduzierbarkeit ist allerdings nicht gegeben, da schon das Rauschen in den Kamerabildern den Zeitpunkt des Überschreitens der Schwellenwerte bei der Regelung und damit den Zeitpunkt des Einsetzens der Kompensation beeinflussen kann.

Wartezeit zwischen Bewegungen: Aus den gewählten Beschleunigungen und Schrittweiten ergeben sich aus den Basisversuchen in Abschnitt 4.1 Wartezeiten zwischen 600 und 1100 Millisekunden zwischen den einzelnen Bewegungsbefehlen.

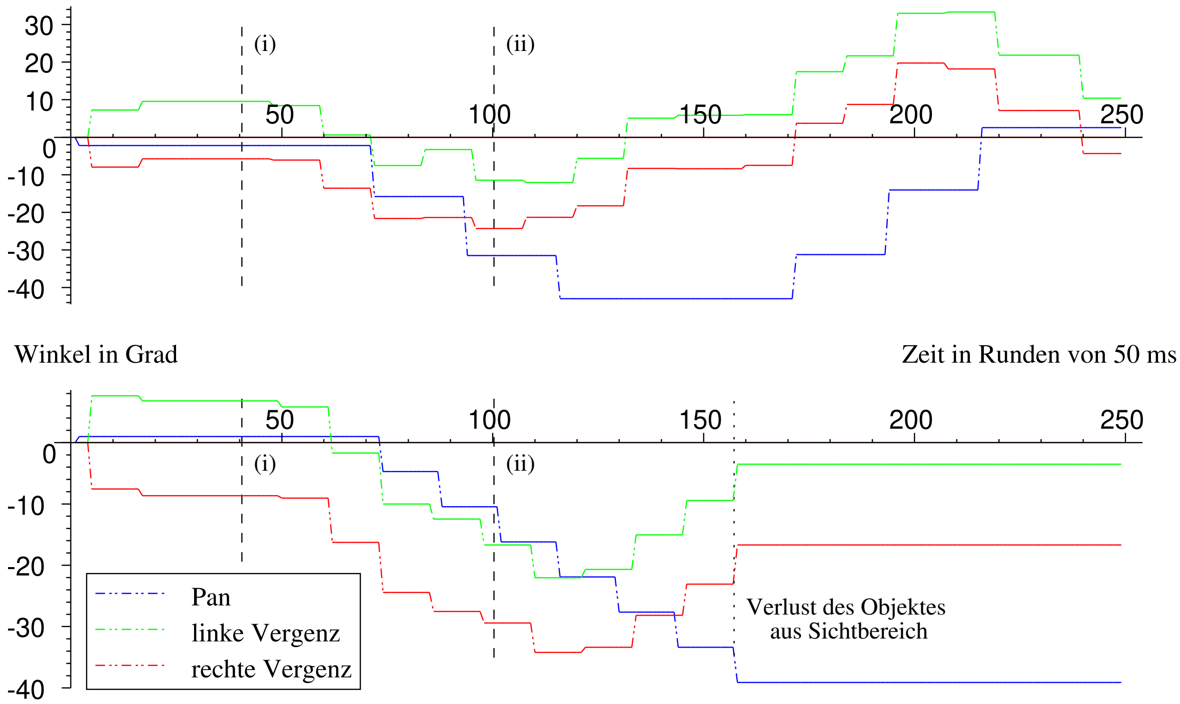
Schwellenwerte zum Ein- und Ausstieg der Kompensation: Wie in Abschnitt 2.3.1 beschrieben, setzt die Kompensation in den einzelnen Verhaltensmodulen erst ein, wenn die Abweichung vom Sollwert einen definierten Schwellenwert übersteigt. Sie setzt wieder aus, wenn ein zweiter Schwellenwert unterschritten wird. Diese Werte werden für das Verhaltensmodul *Richte Pan aus* zwischen 2.86 und 20.05 Grad variiert.

Koordination der Verhaltensmodule: Die verschiedenen Möglichkeiten zur Koordination der Verhaltensmodule werden in Abschnitt 2.3.1 vorgestellt. Hier wird abwechselnd *Nur Pan* - es findet keine Koordination der beiden Module statt - und *Überdeckung der Vergenzen* verwendet.

Die Ergebnisse sind wie bei den Versuchen zur Regelung dargestellt. Die Diagramme der Winkel enthalten zusätzlich den Winkel der Pan-Einheit. Ein positiver Wert steht wieder für eine Rotation im Uhrzeigersinn und der Wert 0 für eine Ausrichtung in Fahrtrichtung. Die Entfernung wird nur geschätzt, wenn nicht gerade eine Vergenzbewegung stattfindet. Daher ändert sich die geschätzte Entfernung sprunghaft in diskreten Intervallen. Unter den Diagrammen sind die relevanten Parameter und die gewählte Methode zur Koordination der Verhaltensmodule notiert. In den Experimenten werden jeweils einzelne Parameter variiert. Im Folgenden werden exemplarisch einige sich ergebende Effekte bei der Wahl bestimmter Parameterkombinationen vorgestellt.

Variation der maximalen Schrittweite

Die Beschleunigung der Vergenzen wird in allen Versuchsläufen gleich gewählt, da eine hohe Beschleunigung möglich ist ohne messbar an Genauigkeit zu verlieren; vgl. Abschnitt 4.1.1. Eine relativ hohe Beschleunigung der Pan-Einheit führt schnell zu Problemen mit der Kalibrierung und die entsprechenden Versuchsläufe waren deutlich weniger erfolgreich. Die Wahl der Schrittweite ist wiederum direkt an die Größe der Wartezeit zwischen Bewegungsbefehlen gekoppelt. In Abb. 19 sind beispielhaft die Winkel der einzelnen Freiheitsgrade bei Versuchsläufen mit großer und mit kleiner Schrittweite abgebildet.
Alle anderen Parameter wurden nicht zwischen den Läufen variiert. Bei großer maximaler Schrittweite reagiert das Verhaltensmodul *Richte Pan aus* mit starken Bewegungen und lenkt die Pan-Einheit erst spät wieder zurück. Die Vergenzen müssen ebenso stark die Bewegung der Pan-Einheit ausgleichen. Durch die einzelnen starken Bewegungen der Pan-Einheit, scheinen die Vergenzen zwischen den Runden 80 und 120 leicht zu schwingen, was am schwankenden Abstand der Graphen der beiden Vergenzen zu erkennen ist. Die Qualität der Entfernungsschätzung nimmt zu dem Zeitpunkt ab. Die Bewegung wird vollständig verfolgt und alle Freiheitsgrade befinden sich am Ende wieder in der Ausgangslage. Im unteren Diagramm sind nur viel kleinere Bewegungen der Pan-Einheit zulässig, die aber in schnellerer Folge veranlasst werden können. Die Pan-Einheit zeigt ebenfalls ab Runde 70 eine Reaktion, aber kann



Keine Koordination der Verhaltensmodule

Kompensation:	Einstieg	Ausstieg
	11.46 Grad	5.73 Grad

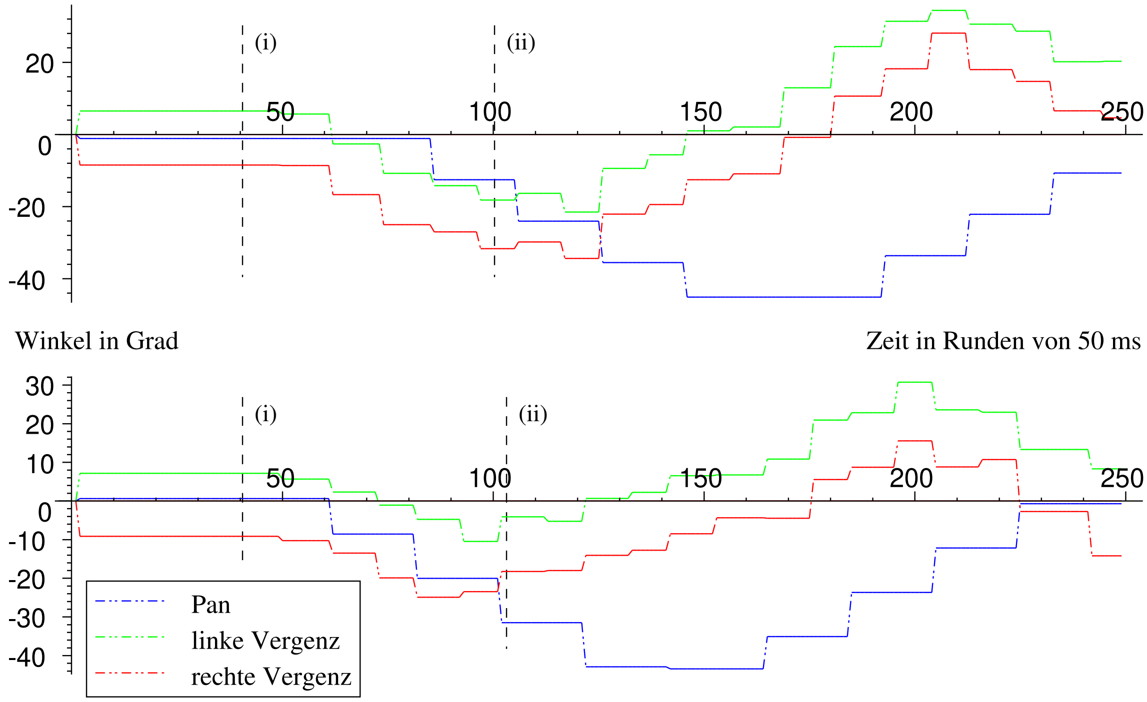
	Beschleunigung		Schrittweite	Wartezeit
Vergenz	7.96 Umdr./ s^2		11.46 Grad	600 ms
Pan	34.38 Grad / s^2	**oben:**	17.19 Grad	1100 ms
		unten:	5.73 Grad	700 ms

Abbildung 19: *Die Stellwinkel der Vergenzen und der Pan-Einheit sind für zwei Versuche zur doppelten Kompensation über die Dauer von* 12.5 *Sekunden abgetragen. Alle relevanten Parameter sind den Tabellen unter den Diagrammen zu entnehmen. Im oben abgebildeten Versuchslauf kann die Pan-Einheit große Bewegungen machen, aber muss zwischen diesen Bewegungen lange warten. Im unteren Diagramm ist die maximale Schrittweite kleiner und damit auch die Wartezeit zwischen einzelnen Bewegungen. Im unten abgebildeten Versuchslauf wird das Objekt ca. bei Runde* 160 *verloren, was durch eine gepunktete Linie angedeutet wird. Die zu verfolgenden Bewegungen werden bei* (i) *und* (ii) *initiiert.*

der Bewegung nicht so schnell folgen. Die schnellere Rückbewegung wird von der Pan-Einheit verpasst und die Marke wird aus dem Blickbereich verloren. (In Abb. 19 ist der Zeitpunkt durch eine gepunktete Linie markiert.) Eine große Schrittweite scheint zwar teilweise zum Schwingen der Vergenzen zu führen, aber ist die Schrittweite zu klein, lassen schnelle Bewegungen das zu verfolgende Objekt aus dem Blickbereich entkommen. Es wird im Folgenden ein Kompromiss (11.46 Grad) gewählt.

Variation der Schwellenwerte

Die Kompensation durch die Pan-Einheit setzt ein, sobald der erste Schwellenwert überschritten wird. Bei Unterschreitung des zweiten (in der Regel kleineren) Schwellenwerts, setzt die Kompensation wieder aus. Diese Schwellenwerte bestimmen mit welcher Verzögerung kompensiert wird. Sind beide niedrig, muss damit gerechnet werden, dass viele unnötige Bewegungen von der Pan-Einheit ausgeführt werden. Sind die Werte zu hoch, setzt die Kompensation eventuell zu spät ein und das Objekt wird aus dem Blickbereich der Kameras verloren. In Abb. 20 sind die Winkel der benutzten Freiheitsgrade zweier Versuchsläufe abgetragen.



Modul zur Steuerung des Pan kann Ausgabe des Moduls zur Steuerung der Vergenzen überdecken

Kompensation:	Einstieg	Ausstieg		Beschleunigung	Schrittweite	Wartezeit
oben:	20.05 Grad	11.46 Grad	Vergenz	7.96 Umdr./ s^2	11.46 Grad	600 ms
unten:	5.73 Grad	2.86 Grad	Pan	34.38 Grad / s^2	11.46 Grad	1000 ms

Abbildung 20: *Die Stellwinkel der Vergenzen und der Pan-Einheit sind für zwei Versuche zur doppelten Kompensation über die Dauer von* 12.5 *Sekunden abgetragen. Alle relevanten Parameter sind den Tabellen unter den Diagrammen zu entnehmen. Die Schwellenwerte für das Ein- bzw. Aussetzen der Kompensation durch die Pan-Einheit werden im oberen Diagramm hoch angesetzt und im unteren niedrig. Im oberen Diagramm reagiert das System entsprechend träge und langsam. Im unteren Diagramm wird dagegen wird stark ausgelenkt. Die zu verfolgenden Bewegungen werden bei* (i) *und* (ii) *initiiert.*

Die verwendeten Parameter unterscheiden sich nur in der Höhe der Schwellenwerte. Im oberen Diagramm sind die Schwellenwerte hoch und entsprechend reagiert die Pan-Einheit erst nach der Runde 80 und erreicht ihre maximale Auslenkung ebenfalls spät. Sowohl die Pan-Einheit als auch die Vergenzen erreichen nicht mehr die Ausgangsstellung bis zum Versuchende. Es fällt noch auf, dass die Auslenkung der Vergenzen zu den positiven Winkeln gering ist. Im unteren Diagramm sind die Schwellenwerte niedrig und die Pan-Einheit kompensiert bereits in Runde

60. Sowohl Pan als auch die Vergenzen erreichen vor Versuchende ihre Ausgangsstellung. Die Auslenkung zu den positiven Winkeln ist bei den Vergenzen etwas stärker als im oberen Diagramm. Insgesamt lassen die niedrigen Schwellenwerte das System nicht unruhig erscheinen. Einzelnes, leichtes Schwingen der Vergenzen ist in beiden Diagrammen zu finden und hat seinen Ursprung in einer nicht idealen Überdeckung der Vergenzen bei der Kompensation durch die Pan-Einheit.

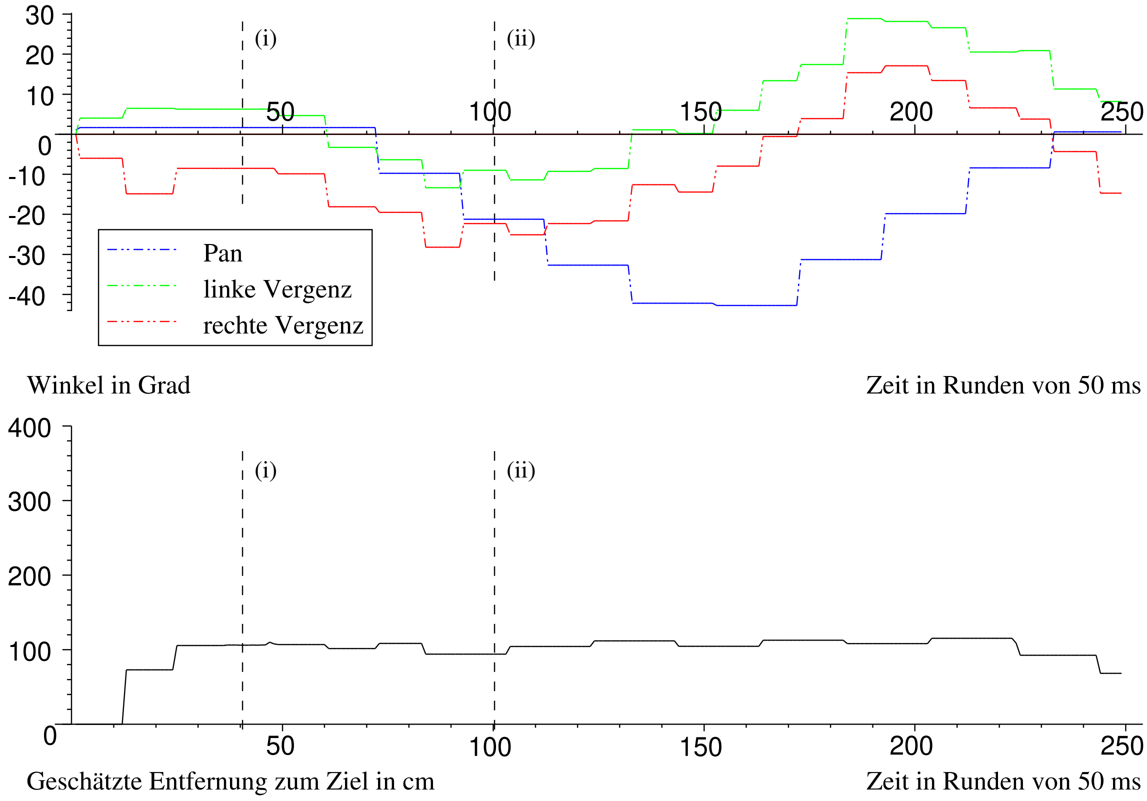
Koordination der Verhaltensmodule durch Überdeckung

Wird als Koordination der Verhaltensmodule die Überdeckung der Ausgabe des Verhaltensmoduls zur Kontrolle der Vergenzen durch das Modul Pan-Einheit gewählt, so ergeben sich weitere Variationsmöglichkeiten. Es kann versucht werden, die ganze Pan-Bewegung durch die Vergenzen in einem Schritt auszugleichen.

Kann aber das Modul zur Steuerung der Vergenzen wieder Ausgaben produzieren, bevor die Pan-Bewegung abgeschlossen ist, werden die Vergenzen gegebenenfalls zurück in die Richtung geregelt, aus der sie gerade kamen um dann (bei Vollendung der Pan-Bewegung) wieder das zu verfolgende Objekt in der ursprünglichen Richtung zu verfolgen. Können die Vergenzen während der gesamten Pan-Bewegung keine eigenständige Bewegung durchführen, so wird dieser Effekt des *Gegensteuerns* zwar vermieden, aber es kann z.B. nicht auf eine plötzliche Richtungsänderung des zu verfolgenden Objektes reagiert werden. Alternativ kann versucht werden, immer einen Teil der Pan-Bewegung zu überdecken. Idealerweise wird genau die Strecke überdeckt, welche von der Pan-Einheit bis zum Zeitpunkt des Vollendens eines neuen Bewegungsbefehls an die Vergenzen bereits zurückgelegt wurde. Durch Tests erwies sich folgende einfache Vorgehensweise als ausreichend stabil: Ist die geplante Pan-Bewegung größer als die halbe maximale Schrittweite, so wird nur die Hälfte der geplanten Bewegung kompensiert. Ist die geplante Bewegung kleiner, wird sie vollständig kompensiert. In Abb. 21 sind die gemessenen Winkel der verwendeten Freiheitsgrade und die zeitgleich geschätzte Entfernung zur verwendeten Landmarke eines entsprechenden Versuchslaufs zu sehen. In der Runde 100 tritt der Effekt des *Gegensteuerns* auf. Insgesamt wird der Bewegung schnell gefolgt und es finden sonst keine unnötigen Bewegungen statt. Alle Freiheitsgrade erreichen bis zum Versuchsende ihre Ausgangslage. Die Entfernungsschätzung liefert während des gesamten Versuches sinnvolle Werte.

Keine Koordination der Verhaltensmodule

Werden keine Mechanismen zur gegenseitigen Koordination der Verhaltensmodule benutzt, ergeben sich weniger Variationsmöglichkeiten. Bei dem in Abb. 22 abgebildeten Versuchslauf ist kaum ein Schwingen der Vergenzen zu erkennen. Die Pan-Einheit reagiert sehr früh und alle Freiheitsgrade sind am Ende wieder in der Ausgangslage. Die Entfernung wird insgesamt sehr gut und konstant geschätzt. Die leichten Schwankungen in der Schätzung stammen aus den in Abschnitt 4.2 untersuchten Fehlerquellen. Ein Einfluss der Bewegung des zu verfolgenden Objektes auf die Entfernungsschätzung ist nicht zu entdecken.



Modul zur Steuerung des Pan kann Ausgabe des Moduls zur Steuerung der Vergenzen überdecken

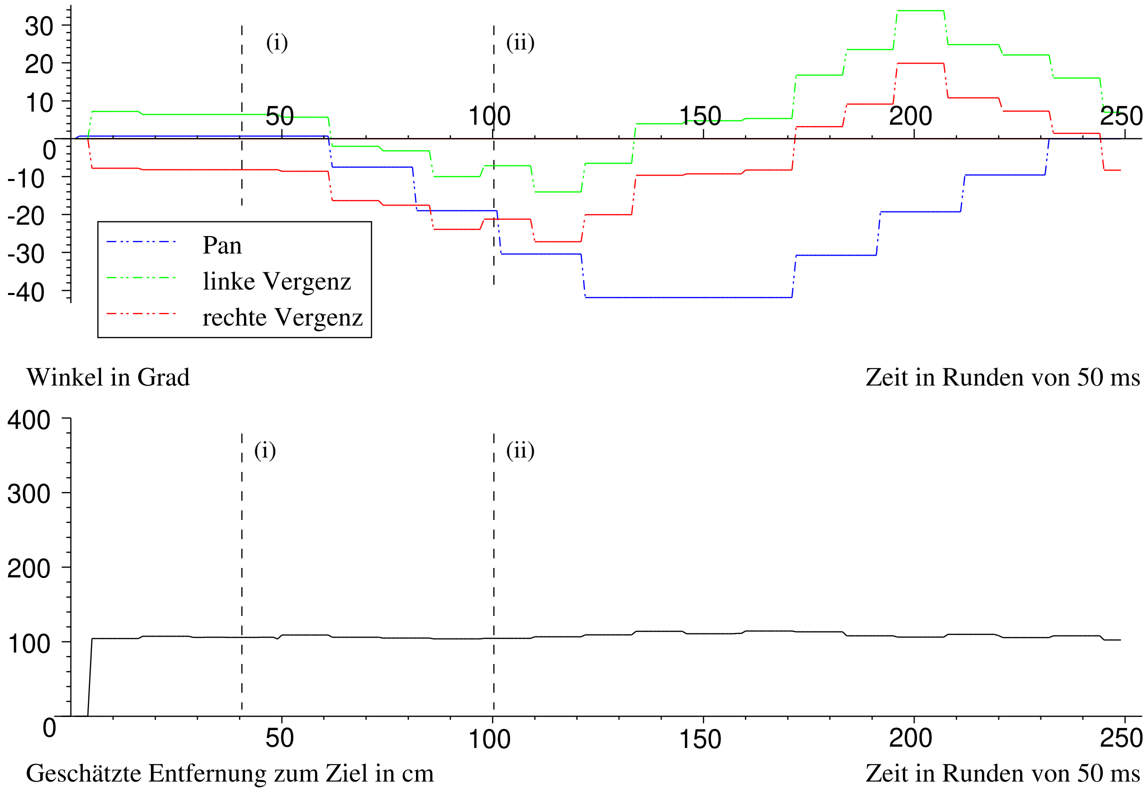
Kompensation:	Einstieg	Ausstieg
	11.46 Grad	5.73 Grad

	Beschleunigung	Schrittweite	Wartezeit
Vergenz	7.96 Umdr./ s^2	11.46 Grad	600 ms
Pan	34.38 Grad / s^2	11.46 Grad	1000 ms

Abbildung 21: *Die Stellwinkel der Vergenzen und der Pan-Einheit sowie die geschätzte Entfernung zur fixierten Landmarke sind für einen Versuchslauf zur doppelten Kompensation über die Dauer von* 12.5 *Sekunden abgetragen. Alle relevanten Parameter sind den Tabellen unter den Diagrammen zu entnehmen. Es findet eine Überdeckung der Ausgabe des Moduls zur Kontrolle der Vergenzen durch das Modul der Pan-Einheit statt. Bewegungen der Pan-Einheit werden nur ab einer definierten Größe vollständig durch die Vergenzen kompensiert, um den Effekt des* Gegensteuerns *zu vermeiden. Die zu verfolgenden Bewegungen werden bei* (i) *und* (ii) *initiiert.*

Die gewonnenen Parameter

An konkreten Parametern hat sich aus den Experimenten ergeben: Eine niedrige Beschleunigung (34.38 Grad$/s$) der Pan-Einheit ist notwendig, um die Kalibrierung nicht zu verlieren. Die Vergenzen können problemlos mit sehr viel höherer Beschleunigung (7.96 Umdr.$/s^2$) verwendet werden. Wird die maximal zulässige Schrittweite bei Rotation der Pan-Einheit zu sehr beschränkt, kann Bewegungen nicht unbedingt gefolgt werden. Wird die maximale Schrittweiten groß gewählt, ist das System zwar erfolgreich, aber auch unruhig, da die Wartezeit zwischen einzelnen Bewegungsbefehlen ebenso ansteigt. Es wird im folgenden ein Kompromiss von 11.45 Grad verwendet. Die Schwellenwerte zum Ein- bzw. Aussetzen der Kompensation



Keine Koordination der Verhaltensmodule

Kompensation:	Einstieg	Ausstieg
	5.73 Grad	2.86 Grad

	Beschleunigung	Schrittweite	Wartezeit
Vergenz	7.96 Umdr./ s^2	11.46 Grad	600 ms
Pan	34.38 Grad / s^2	11.46 Grad	1000 ms

Abbildung 22: *Die Stellwinkel der Vergenzen und der Pan-Einheit sowie die geschätzte Entfernung zur fixierten Landmarke sind für einen Versuchslauf zur doppelten Kompensation über die Dauer von* 12.5 *Sekunden abgetragen. Alle relevanten Parameter sind den Tabellen unter den Diagrammen zu entnehmen. Die zu verfolgenden Bewegungen werden bei* (i) *und* (ii) *initiiert. Es findet keine Koordination der Verhaltensmodule statt.*

können niedrig gewählt werden (5.73 bzw. 2.86 Grad), ohne dass das Verhalten des Systems sehr unruhig wirkt. Die schnelle Rückbewegung in den Versuchsläufen führt bei hohen Schwellenwerten manchmal zum Verlust der Marke aus dem Sichtbereich der Kameras.

Sind obige Parameter sinnvoll gewählt, können die Verhaltensmodule verschieden koordiniert werden. Überdeckt das Modul *Richte Pan aus* die Ausgabe des Verhaltensmoduls zur Steuerung der Vergenzen, muss definiert werden, wie lange die Ausgaben von *Richte Vergenzen aus* verworfen werden und wie stark die Vergenzen durch *Richte Pan aus* ausgelenkt werden sollen. Eine kurzzeitige Sperrung der Ausgaben zusammen mit einer teilweisen Auslenkung der Vergenzen hat sich in den Versuchen als sinnvoll erwiesen. In dieser Weise ist ein erfolgreiches Gesamtverhalten möglich. Ohne Koordination der beiden Verhaltensmodule ist das System ebenfalls erfolgreich. Ohne die Eingriffe des Moduls *Richte Pan aus* in die Steuerung der Vergenzen ist trotz der gegenläufigen Bewegungen eine sehr konstante Entfernungsschätzung möglich. Die im Anfang des Abschnitts 4.4 genannten Anforderungen (*zuverlässige Entfernungsschätzung*, *Reaktionsgeschwindigkeit* und *subjektive Ruhe des Systems*) werden durch beide Methoden der Koordination erfüllt.

4.4.2 Dreifache Kompensation

Dienen die vorherigen Experiment noch teilweise der Abstimmung der Parameter, werden in den Experimenten zur dreifachen Kompensation verschiedene Möglichkeiten zur Koordination der Verhaltensmodule untersucht. Die verschiedenen sich im Gesamtverhalten ergebenden Bewegungsstrategien werden verglichen. Es ist zu klären, ob das in dieser Arbeit konstruierte, verhaltensbasierte System ein sinnvolles Gesamtverhalten bei Verwendung aller verfügbaren Freiheitsgrade zeigen kann.

Aufbau und Ablauf

In diesem Experiment werden alle Freiheitsgrade zur Rotation durch die jeweiligen Verhaltensmodule geregelt. Um eine reproduzierbare Bewegung zu erzeugen wird eine Modelleisenbahn verwendet. Wie in Abb. 23 schematisch zu sehen, wird der Roboter neben den Gleisen positioniert und es wird im Laufe des Versuches keine Translation durchgeführt.

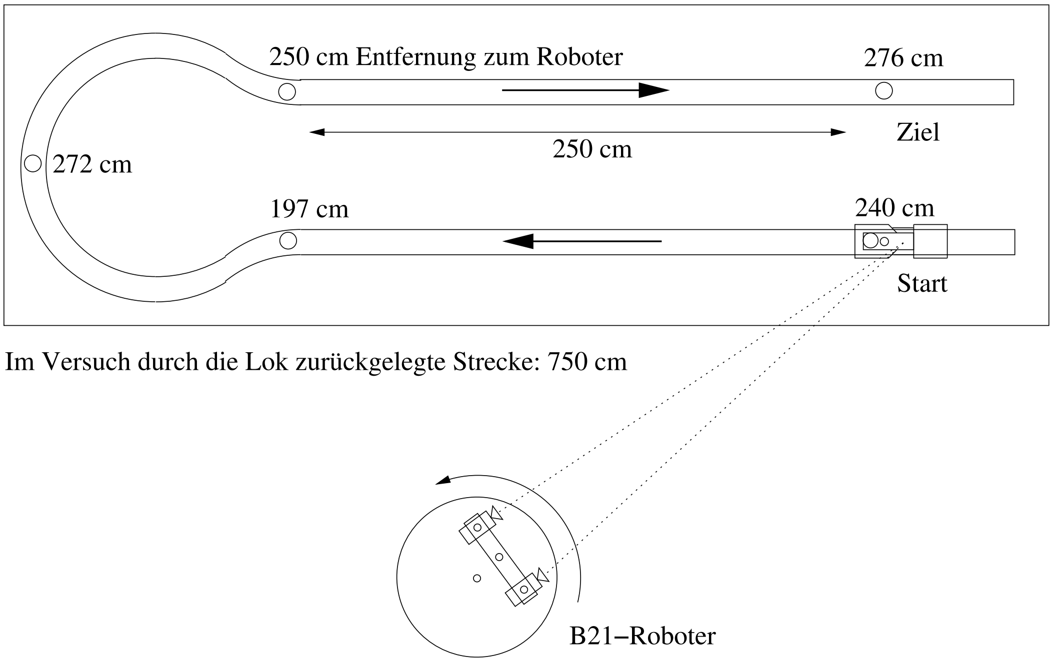


Abbildung 23: *Schematisch ist der Aufbau des Versuchs zur dreifachen Kompensation in Draufsicht abgebildet.*

Die Vergenzen fixieren nun die in der Startposition stehende Lok. Dazu sind Basis und die Tilt-Einheit geeignet ausgerichtet und die Pan-Einheit ist in Nullstellung. Bei der Fahrt der Lok zur Zielposition werden die Vergenzen, Pan, Tilt, und die Basis durch ihre jeweiligen Verhaltensmodule geregelt. Wie beim Experiment zur doppelten Kompensation werden die Stellwinkel der verwendeten Freiheitsgrade und die geschätzte Entfernung zur Marke rundenweise gespeichert; eine Runde dauert 50 Millisekunden. In den Diagrammen werden die Stellwinkel und die geschätzte Entfernung zur Marke über die Dauer der Fahrt der Lok abgetragen. Im Gegensatz zum vorherigen Versuch ist die Entfernung nicht konstant. Um die Qualität der Entfernungsschätzung mit Hilfe einer Grundwahrheit beurteilen zu können, wird die Entfernung der Lok zum Kamerasystem vor dem Versuch für die jeweiligen Gleisabschnitte von Hand ausgemessen. In Abb. 23 ist diese Entfernung für fünf markante Punkte eingezeichnet. In den Diagrammen der einzelnen Versuchsläufe sind die Messpunkte als kleine Kreise über dem Graphen der geschätzten Entfernung zu sehen.

Die Gleise der Modelleisenbahn befinden sich in einer Höhe von etwa 80 cm vom Boden, das Stereokamerasystem ist in einer Höhe von 160 cm montiert. Zum Fixieren der Lok muss daher kontinuierlich die Neigung des Kamerakopfes durch das Verhaltensmodul *Richte Tilt aus* geregelt werden. Aufgrund der nicht idealen Odometrie des Kamerakopfes bewirkt eine Neigung des Kopfes eine horizontale Verschiebung des Objektes im Kamerabild. Dies erschwert insgesamt die Verfolgung des Objektes; siehe auch Anhang C. In Abb. 24 ist links der Roboter beim Fixieren der Lok zu sehen. Im rechten Bild sind die Strecke und die Lok der Modelleisenbahn abgebildet.

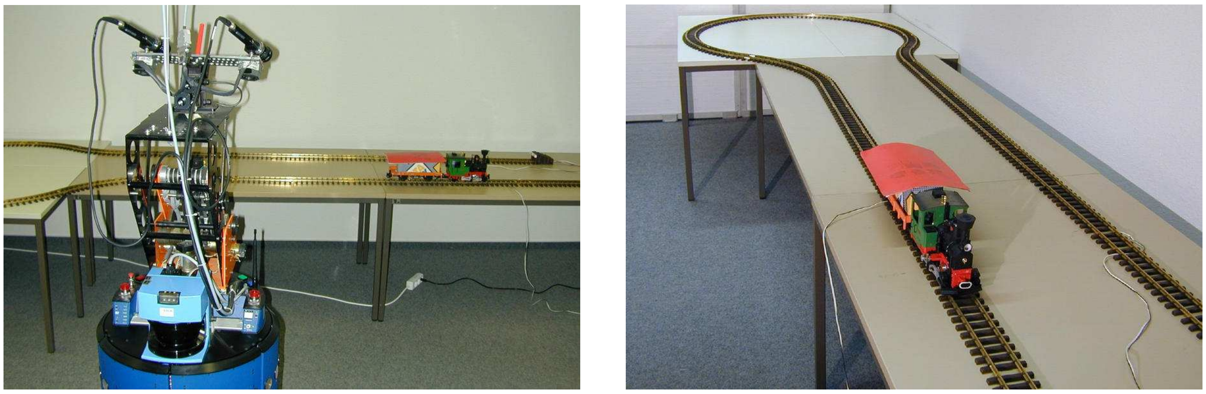


Abbildung 24: *Links ist der Roboter beim fixieren der Lok zu sehen. Rechts ist die Strecke mit der Lok abgebildet.*

In den Diagrammen werden jeweils die Winkel der benutzten Freiheitsgrade - außer der Neigung durch den Freiheitsgrad Tilt - und die dabei geschätzte Entfernung wie in den vorherigen Versuchen dargestellt. Die Regelung der Neigung wird in keinem der Versuche durch die anderen Verhaltensmodule beeinflusst. Der genaue Zeitpunkt des Starts der Lok ist von einer analogen Schaltung abhängig und die Geschwindigkeit wird mit dem Trafo der Modelleisenbahn gesteuert. Die in den Diagrammen zur Entfernungsschätzung eingetragenen Messpunkte zur Referenz mussten daher manuell an die Kurve angepasst werden. Weiterhin muss mit Ungenauigkeiten bei den Messpunkten selber gerechnet werden, da sich die Position des B21-Roboters bei Rotationen der Basis auf dem 4-Achsen-Fahrwerk etwas verschiebt. Aus den genannten Gründen unterscheiden sich die Diagramme ebenso in der Gesamtlänge der Versuche. Da alle anderen Einfluss nehmenden Parameter (wie Aufbau und Beleuchtung der Strecke oder die Position des Roboters) zwischen den Läufen gleich geblieben sind, sind die Ergebnisse trotzdem vergleichbar.

Wahl der Parameter

Es finden 24 Versuchsläufe statt. Hierbei werden die verschiedenen Möglichkeiten zur Koordination der beteiligten Verhaltensmodule variiert. In den Versuchsläufen werden zunächst die in Abschnitt $2.3.1$ vorgestellten Möglichkeiten zur Koordination der Verhaltensmodule getestet. An einzelnen Beispielen können die Auswirkungen auf das Gesamtverhalten erläutert werden. Erfolgreiche Versuchsläufe werden daraufhin bei höherer Geschwindigkeit des zu verfolgenden Objektes wiederholt. Anschließend wird zum Vergleich der gleiche Versuch bei doppelter Kompensation wiederholt, um die Effektivität beim Einsatz von drei überlappenden Freiheitsgraden einschätzen zu können. Im Folgenden werden beispielhaft einige sich ergebenden Effekte und Bewegungsstrategien vorgestellt.

Keine Koordination der Verhaltensmodule

Die Lok bewegt sich etwa mit einer Geschwindigkeit von ca. $0.75\ m/s$. Bei fast allen Läufen kann die sie während des gesamten Versuches fixiert und die Entfernung geschätzt werden. Diese Kriterien stellen gegenläufige Anforderungen an das System und es muss ein geeigneter Kompromiss gefunden werden. Die Möglichkeiten zur Koordination der Verhaltensmodule lassen sich danach einteilen, wie stark die Module sich gegenseitig beeinflussen. In den Versuchen zur doppelten Kompensation war ein Extrem (das Weglassen jeglicher Koordination) sehr erfolgreich.

Beim in Abb. 25 dargestellten Versuch findet keine Koordination der vier beteiligten Verhaltensmodule statt. Die Lok wird während des gesamten Versuchs erfolgreich fixiert. Die Auslenkung der einzelnen Freiheitsgrade erscheint sinnvoll: Bis zum Zeitpunkt *(i)* wird das zu verfolgende Objekt allein mit Hilfe der Vergenzen fixiert. Dann wird die Pan-Einheit ausgelenkt und die Vergenzen kompensieren die neue Bewegung des Objektes in den Kamerabildern. Am Zeitpunkt *(ii)* wird der Schwellenwert der Basis erreicht und die Pan-Einheit führt eine letzte Bewegung aus. Während die Basis weiter rotiert, gleichen die Vergenzen die Bewegung aus und fixieren weiterhin die Lok. Zum Zeitpunkt *(iii)* erreicht die Basis ihre weiteste Auslenkung und eine Rückbewegung der Pan-Einheit setzt ein. Diese Rückbewegung ist etwas unstet und endet nicht an der x-Achse. Zum Zeitpunkt *(iv)* verhält die Pan-Einheit und die Basis rotiert wieder in Richtung Ausgangsposition.

Am Versuchslauf fällt auf: Wenn die Pan-Einheit und Basis sich gleichzeitig bewegen, fangen die Vergenzen an zu Schwingen; siehe Zeitpunkte *(ii)* und *(iii)*. Dies ist auch in einem starken Schwanken der Entfernungsschätzung zu erkennen. Da keine Koordination stattfindet, müssen die Vergenzen in dem Augenblick drei unabhängige, unbekannte Bewegungen auf einmal ausgleichen: die Rotation der Basis, der Pan-Einheit und die Eigenbewegung der Lok. Kennzeichnend für diese Bewegungsstrategie ist auch die starken Auslenkungen der einzelnen Gelenke: Die Basis nach links (-80 Grad) bei gleichzeitiger Ausrichtung der Pan-Einheit nach rechts

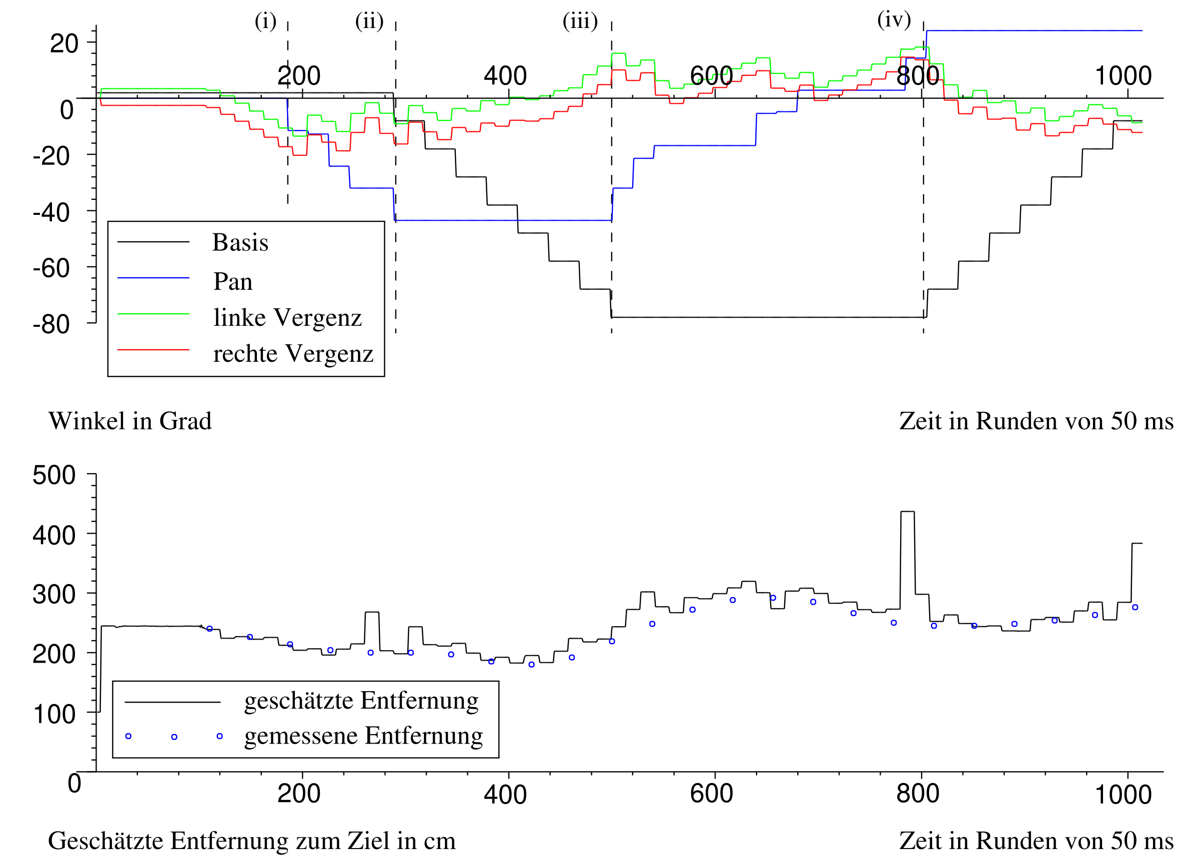


Abbildung 25: *Die Stellwinkel der einzelnen Freiheitsgrade und die geschätzte sowie gemessene Entfernung zum Objekt sind über die Dauer des Versuches abgetragen. Durch gestrichelte Linien wird der Versuchslauf in verschiedene Abschnitte unterteilt. Es findet keine Koordinierung der Verhaltensmodule statt.*

(+20 Grad) zum Zeitpunkt *(iv)*. Das sich ergebende Gesamtverhalten erscheint für das Fehlen jeglicher Koordination erstaunlich sinnvoll. Dies ist in der geeigneten Wahl der einzelnen Parameter begründet.

Extreme Koordination aller beteiligten Verhaltensmodule

Das gegensätzliche Extrem - eine große Kontrolle der Verhaltensmodule übereinander - ist in Abb. 26 dargestellt. Hier überdeckt das Modul *Richte Pan aus* die Ausgaben des Moduls zur Steuerung der Vergenzen und kann die Ausgaben des Moduls zur Rotation der Basis unterdrücken. Gleichzeitig überdeckt *Richte Basis aus* ebenfalls die Ausgaben des Moduls zur Steuerung der Vergenzen und kann die Ausgaben des Moduls zur Steuerung der Pan-Einheit unterdrücken. Das Ablauf ist dem in Abb. 25 ähnlich: Zum Zeitpunkt *(i)* setzt die Bewegung der Basis ein. Ab hier schwingen die Vergenzen erheblich bis zum Zeitpunkt *(ii)* - allerdings ohne dass die Entfernungsschätzung sehr darunter leidet. Zum Zeitpunkt *(iii)* bewegt sich die Basis wieder in Richtung Ausgangsposition und die Pan-Einheit verharrt in ihrer Auslenkung.

Die Verhaltensmodule *Richte Pan aus* und *Richte Basis aus* scheinen sich zeitweise bei der

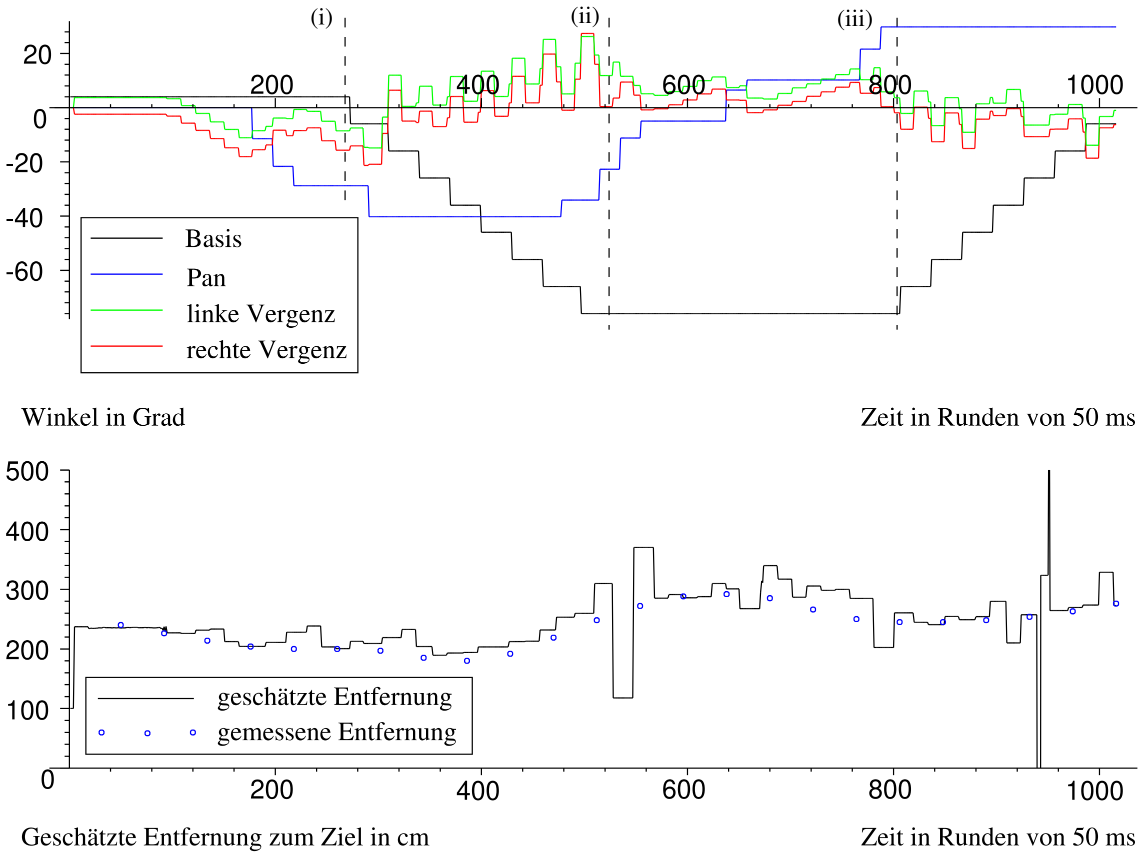


Abbildung 26: *Die Stellwinkel der einzelnen Freiheitsgrade und die geschätzte sowie gemessene Entfernung zum Objekt sind über die Dauer des Versuches abgetragen. Durch gestrichelte Linien wird der Versuchslauf in verschiedene Abschnitte unterteilt. Die Pan-Einheit und die Basis können gegenseitig ihre Ausgaben unterdrücken und die Ausgabe der Vergenzen überdecken.*

Überdeckung der Vergenzen abzuwechseln und lösen so das starke Schwingen aus. Die Entfernungsschätzung wird nur zu den Zeitpunkten *(ii)* und *(iii)* unbrauchbar: Hier bewegen sich die Pan-Einheit und Basis trotz Koordination gleichzeitig. Positiv fällt auf, dass die Basis und Pan-Einheit keine unnötig weiten Bewegungen machen. Das System wirkt insgesamt aber durch die Konkurrenzsituation bzgl. der Überdeckung des Moduls *Richte Vergenzen aus* sehr unruhig.

Sequentielle Überdeckung der Ausgaben von Verhaltensmodulen

Anstatt den Verhaltensmodulen für Pan und Basis jeweils die Möglichkeit zur Überdeckung der Vergenzen zu geben, wird in Abb. 27 beispielhaft ein Versuchslauf dargestellt, in dem eine sequentielle Überdeckung getestet wird: Das Modul *Richte Basis aus* überdeckt die Ausgabe das Modul *Richte Pan aus*. Dieses wiederum überdeckt die Ausgabe des Moduls *Richte Vergenzen aus*. Wie in den vorher dargestellten Läufen setzt nach einer anfänglichen Verfolgung des Objektes nur durch die Vergenzen ein Nachziehen der Pan-Einheit zum Zeitpunkt *(i)* ein. Zum Zeitpunkt *(ii)* beginnt die Rotatition der Basis, und die Pan-Einheit wird wieder in die

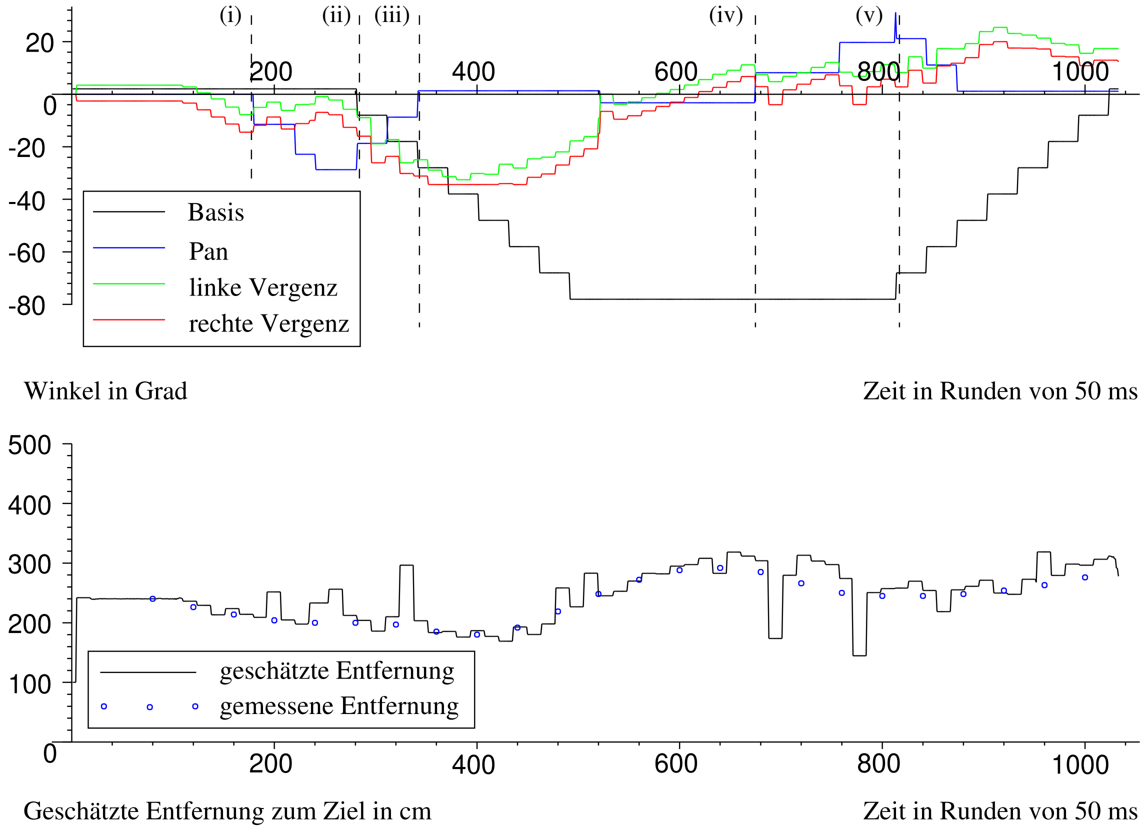


Abbildung 27: *Die Stellwinkel der einzelnen Freiheitsgrade und die geschätzte sowie gemessene Entfernung zum Objekt sind über die Dauer des Versuches abgetragen. Durch gestrichelte Linien wird der Versuchslauf in verschiedene Abschnitte unterteilt. Die Pan-Einheit dann die Ausgabe der Vergenzen überdeckt und die Basis überdeckt die Ausgabe der Pan-Einheit.*

Ausgangslage gebracht, wo sie ab Zeitpunkt *(iii)* verharrt. Dabei müssen die Vergenzen sehr stark ausgelenkt werden, da die Basis alleine der Bewegung der Lok nicht schnell genug folgen kann. Während die Basis längere Zeit in ihrer maximalen Auslenkung verweilt, setzt zum Zeitpunkt *(iv)* wieder eine Rotation der Pan-Einheit ein. Ein Maximum wird bei *(v)* erreicht und die Rotation der Basis setzt wieder ein. Die dadurch erneut zustande kommende Überdeckung des Moduls *Richte Pan aus* sorgt erneut für eine Rückkehr der Pan-Einheit in die Ausgangslage.

Das starke Schwingen der Vergenzen tritt hier tatsächlich nicht wieder auf. Es zeigt sich nur vereinzeltes Schwingen, z.B. bei Runde 800. Die sich durch die sequentielle Überdeckung ergebende Hierarchie verhindert aber eine sinnvolle Ausnutzung der Freiheitsgrade. Sobald das Modul *Richte Basis aus* anfängt zu kompensieren, wird die Pan-Einheit wieder in die Ausgangslage gebracht. Dies sorgt zwar für eine sinnvolle Lage am Ende (Pan-Einheit und Basis sind nicht in entgegengesetzten Richtungen ausgelenkt), aber führt auch fast zum Verlust des Objektes aus dem Sichtbereich nach Zeitpunkt *(iii)*.

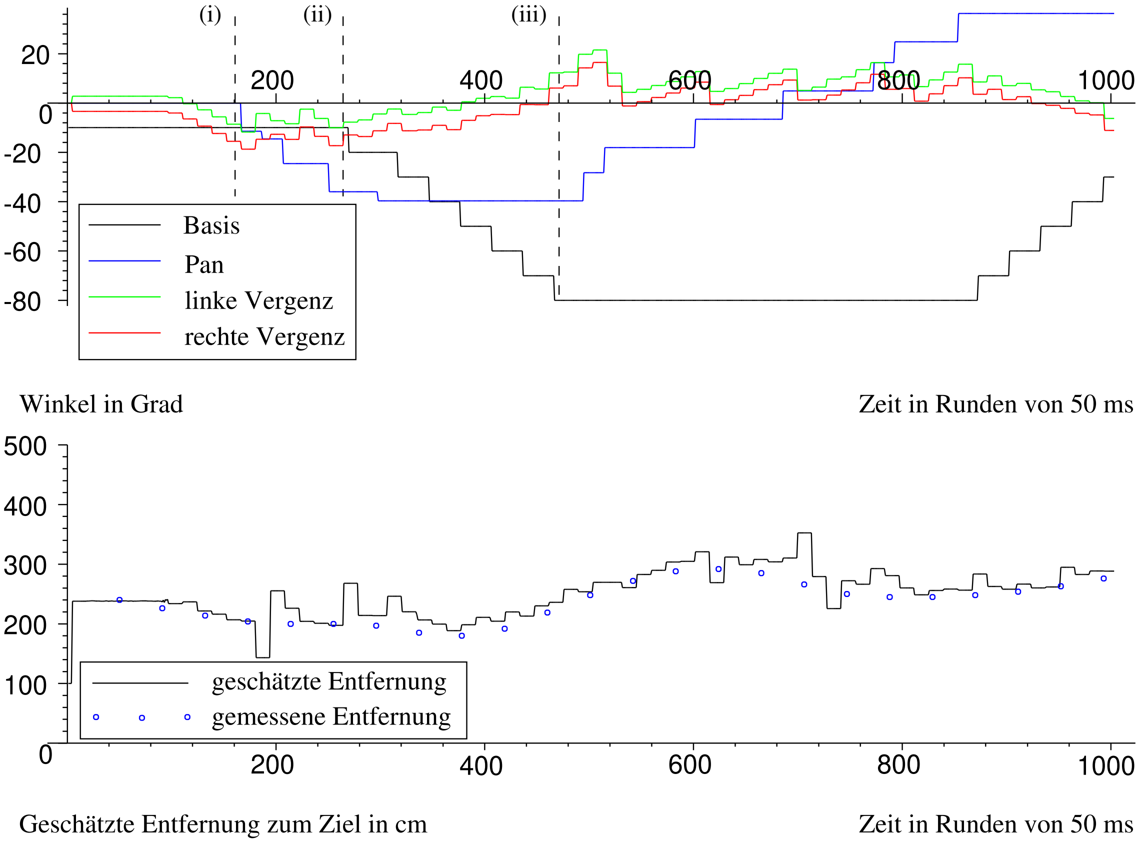


Abbildung 28: *Die Stellwinkel der einzelnen Freiheitsgrade und die geschätzte sowie gemessene Entfernung zum Objekt sind über die Dauer des Versuches abgetragen. Durch gestrichelte Linien wird der Versuchslauf in verschiedene Abschnitte unterteilt. Die Pan-Einheit und die Basis können gegenseitig ihre Ausgaben unterdrücken.*

Gegenseitige Unterdrückung der Ausgaben von Verhaltensmodulen

Es bietet sich an die Koordination weiter zur Verringern und lediglich ungünstige Fälle wie die gleichzeitige Bewegung der Pan-Einheit und der Basis auszuschließen. In Abb. 28 ist ein Versuchslauf dargestellt, bei dem die Verhaltensmodule *Richte Pan aus* und *Richte Basis aus* sich können gegenseitig ihre Ausgaben unterdrücken. Zum Zeitpunkt *(i)* setzt wie in fast allen Läufen eine Rotation der Pan-Einheit ein, nachdem lediglich die Vergenzen das Objekt verfolgten. Ab Zeitpunkt *(ii)* bewegen sich Basis und Pan-Einheit. Tatsächlich finden die Bewegungen in kurzen zeitlichen Abständen abwechselnd statt, da beide Module die Ausgabe des jeweilig anderen kurzzeitig blockieren kann. Die Vergenzen fangen hierbei nicht an zu schwingen. Zum Zeitpunkt *(iii)* hat die Basis ihre maximale Auslenkung erreicht und die Pan-Einheit schwenkt langsam zurück.

Die Vergenzen sind lediglich vereinzelnd am Schwingen und die Entfernungsschätzung ist insgesamt sehr zuverlässig. Es treten nur geringfügige Abweichungen auf. Die Auslenkung der Vergenzen ist während des gesamten Versuchslauf gering und die Auslenkungen der Basis und der Pan-Einheit ergänzen sich gut.

Messwerte bei erhöhter Geschwindigkeit der Lok

Einzelne Versuchsläufe werden bei erhöhter Geschwindigkeit (ca. $1\ m/s$) wiederholt. Dies ist eine niedrigere Geschwindigkeit, als sie z.B. ein Fußgänger zeigen würde, aber das Problem des Fixierens wird hier erheblich durch die starke Neigung des Kamerakopfes zum Objekt erschwert. Eine kontinuierliche Anpassung der Neigung ist nötig, aber diese kann (ebenso wie die Pan-Einheit) nicht mit der gleichen Geschwindigkeit wie die Vergenzen gesteuert werden. Weiterhin ist das Objekt (die Lok der Modelleisenbahn) sehr viel kleiner als eine Person, was die Aufgabe erheblich erschwert.

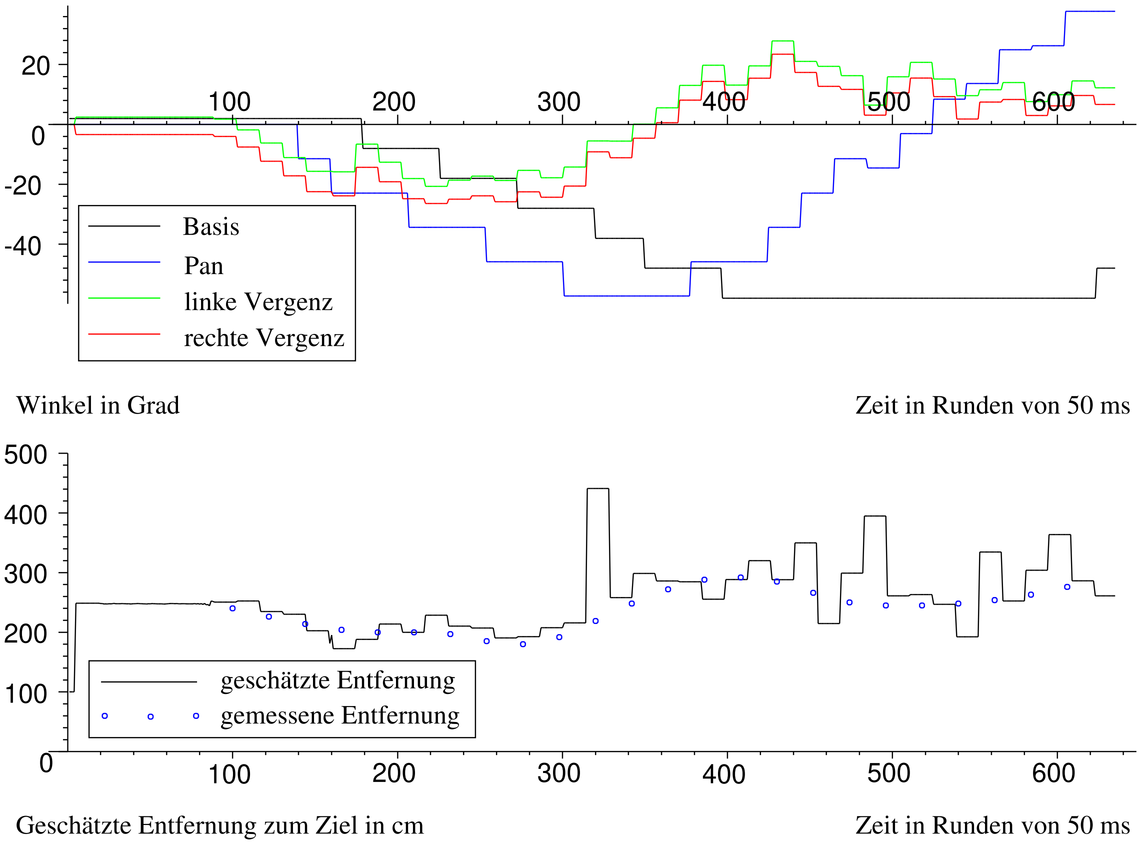


Abbildung 29: *Die Stellwinkel der einzelnen Freiheitsgrade und die geschätzte sowie gemessene Entfernung zum Objekt sind über die Dauer des Versuches abgetragen. Die Pan-Einheit und die Basis können sich sich gegenseitig unterdrücken. Das Objekt bewegt sich mit erhöhter Geschwindigkeit.*

Bei dem in Abb. 29 dargestellten Versuch unterdrücken die Verhaltensmodule *Richte Pan aus* und *Richte Basis aus* gegenseitig ihre Ausgaben. Die Lok wird auch hier während des gesamten Versuches erfolgreich fixiert. Das System wirkt erwartungsgemäß unruhiger und es ist eine stärkere Auslenkung der Vergenzen nötig, um die Lok nicht aus dem Sichtbereich zu verlieren. Die Entfernungsschätzung zeigt kurzzeitige Störungen, aber bleibt insgesamt zuverlässig. Die viel geringere Auslenkung der Basis fällt auf. Statt dessen ist die Auslenkung der Pan-Einheit und der Vergenzen größer, als bei den Versuchsläufen bei niedrigerer Geschwindigkeit. Dies ist in den größeren Schwellenwerten und den ebenso größeren Wartezeiten des Verhaltensmoduls *Richte Basis aus* begründet. Das bisher beobachtbare Gesamtverhalten (Auslenkung der Pan-Einheit, dann Unterstützung durch die Basis und die Rückkehr der Pan-Einheit in die Ausgangslage) ist auch hier zu finden.

Um einen Vergleich mit der doppelten Kompensation machen zu können, wird in weiteren Versuchsläufen der Freiheitsgrad der Basis nicht mehr benutzt. In Abb. 30 sind die Messwerte eines solchen Laufs bei erhöhter Geschwindigkeit abgetragen. Die Pan-Einheit muss hierbei beim Start auf die Lok ausgerichtet werden, anstatt (wie bei den bisherigen Versuchsläufen) in der Ausgangslage zu starten. Die Vergenzen zeigen auch hier leichtes Schwingen, welches sich in Fehlern bei der Schätzung der Entfernung niederschlägt. Zwischen den Runden 450 und 500 wird über einen längeren Zeitraum fehlerhaft geschätzt. Insgesamt kann die Lok aber während des gesamten Versuchs erfolgreich fixiert werden.

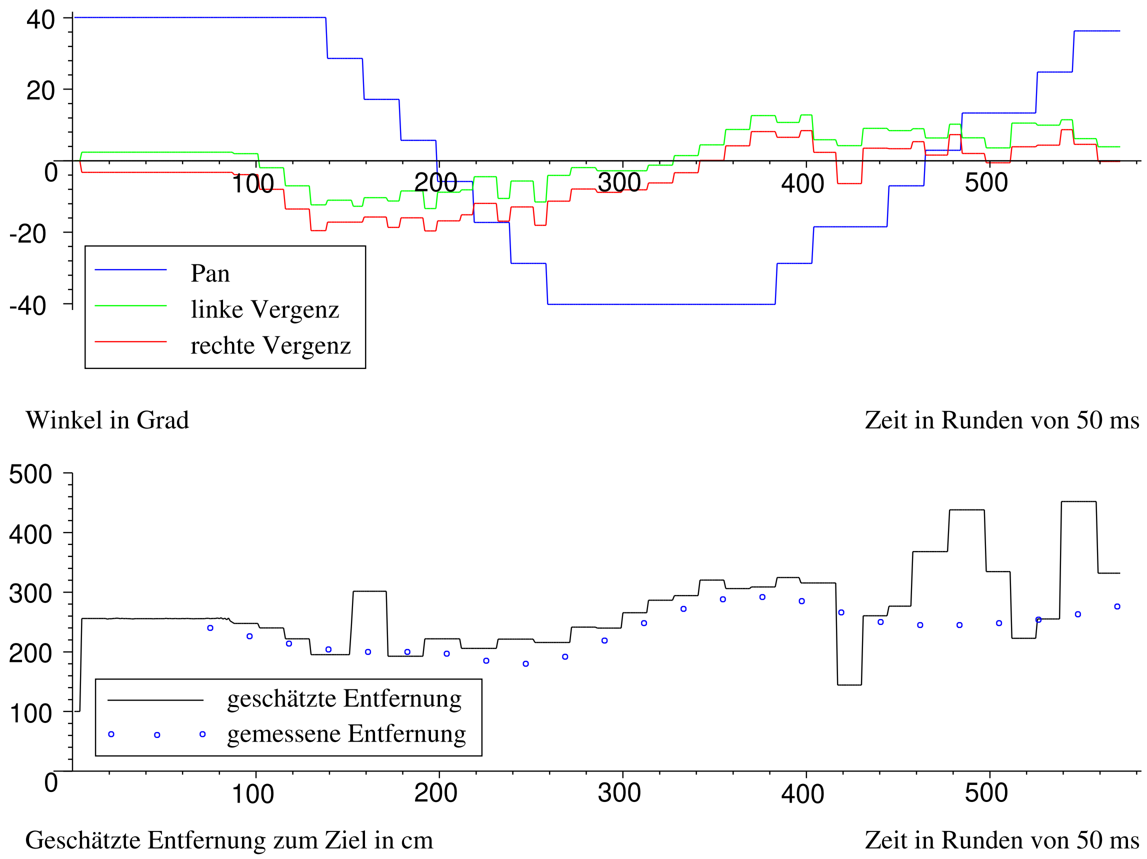


Abbildung 30: *Die Stellwinkel der einzelnen Freiheitsgrade und die geschätzte sowie gemessene Entfernung zum Objekt sind über die Dauer des Versuches abgetragen. Zum Test wird ein Freiheitsgrad (Rotation der Basis) nicht verwendet. Die Pan-Einheit überdeckt dabei die Ausgaben der Vergenzen. Das Objekt bewegt sich mit erhöhter Geschwindigkeit.*

Ergebnisse

Es kann festgestellt werden, dass eine Steuerung der Freiheitsgrade durch einzelne Verhaltensmodule auch bei der Verwendung von vier Achsen erfolgreich ist. Eine Anpassung der Koordination ist nötig. Eine starke Einflussnahme der Verhaltensmodule aufeinander stellt sich als eher hinderlich heraus. Ebenso scheint das Weglassen jeglicher Koordination ein unruhiges - wenn auch überraschend stabiles - System zu ergeben. Das Erzeugen einer klaren Hierarchie unter den Modulen durch sequentielle Überdeckung erweist sich ebenfalls als ungünstig. Die Möglichkeiten der Freiheitsgrade können nicht vollständig genutzt werden und das Objekt wird fast aus dem Sichtbereich verloren. Eine minimale Koordination zum Ausschluss ungünstiger Zustände (wie das gleichzeitige Einsetzen der Bewegung von Pan-Einheit und Basis) wird durch die gegenseitige Unterdrückung der Module zur Steuerung der Pan-Einheit und der Basis realisiert. Diese Art der Koordination stellt sich insgesamt als am erfolgreichsten heraus. Auch bei erhöhter Geschwindigkeit der zu verfolgenden Lok wird die Aufgabe gut bewältigt. Um eine Vergleichsmöglichkeit zur doppelten Kompensation zu besitzen wird die Modelleisenbahn (ebenso bei erhöhter Geschwindigkeit) nur mit den Freiheitsgraden der Basis, der Vergenzen und Tilt verfolgt. Die doppelte Kompensation erweist sich trotz der Vereinfachung des Regelungsproblems nicht als überlegen.

Die sequentielle Überdeckung zeigt als einzige der vorgestellten Versuchsläufe eine Bewegungsstrategie, bei der eine gegensätzliche Ausrichtung der Pan-Einheit und der Basis verhindert wird. Diese Eigenschaft erscheint zwar sinnvoll, aber die Nachteile der nicht ausgeschöpften Bewegungsmöglichkeiten überwiegen. Bei der gegenseitigen Unterdrückung der Ausgaben der Module zur Steuerung der Pan-Einheit und der Basis bewegten sich alle Gelenke gleichzeitig, ohne dass Störungen in Form von starkem Schwingen der Vergenzen auftreten.

Die am Anfang von Abschnitt 4.4 gestellten Anforderungen werden von der gewählten Koordination der Verhaltensmodule erfüllt: Die Entfernungsschätzung ist nicht immer fehlerfrei, aber die Fehlschätzungen sind weder anhaltend noch groß. Das System reagiert schnell genug um die mit erhöhter Geschwindigkeit fahrende Lok kontinuierlich zu fixieren. Die Bewegungen wirken dabei insgesamt ruhig und es werden keine unnötigen Bewegungen durchgeführt.

Der Versuch zeigt weiterhin, dass eine kontinuierliche, fehlerfreie Entfernungsschätzung, wie sie sich im vorherigen Experiment (z.B. in Abb. 22) zeigt, nicht möglich ist. Die bei der Verwendung von vier Freiheitsgraden zur Rotation auftretenden Störungen lassen alle Schätzungen weniger kontinuierlich erscheinen. Eine Mittelung über mehrere Schätzungen der Entfernung ist zur Steuerung des Roboters nötig um nicht auf einzelne Fehlschätzungen mit einer Translation des Roboters zu reagieren. Ein derartiger Mechanismus ist im Verhaltensmodul *Halte Abstand* implementiert.

4.5 Test des Gesamtsystems

Die Experimente unter Verwendung des gesamten Systems und aller Verhaltensmodule bilden den Abschluss der systematischen Untersuchung des in der vorliegenden Arbeit implementierten Systems. Im Folgenden wird die eigentliche Aufgabe des Systems (das Fixieren und

Folgen von Personen) in der Büroumgebung des Instituts getestet. Danach wird der Ausfall der Pan-Tilt-Einheit simuliert um die Robustheit des Systems gegen Fehler in der Hardware zu untersuchen.

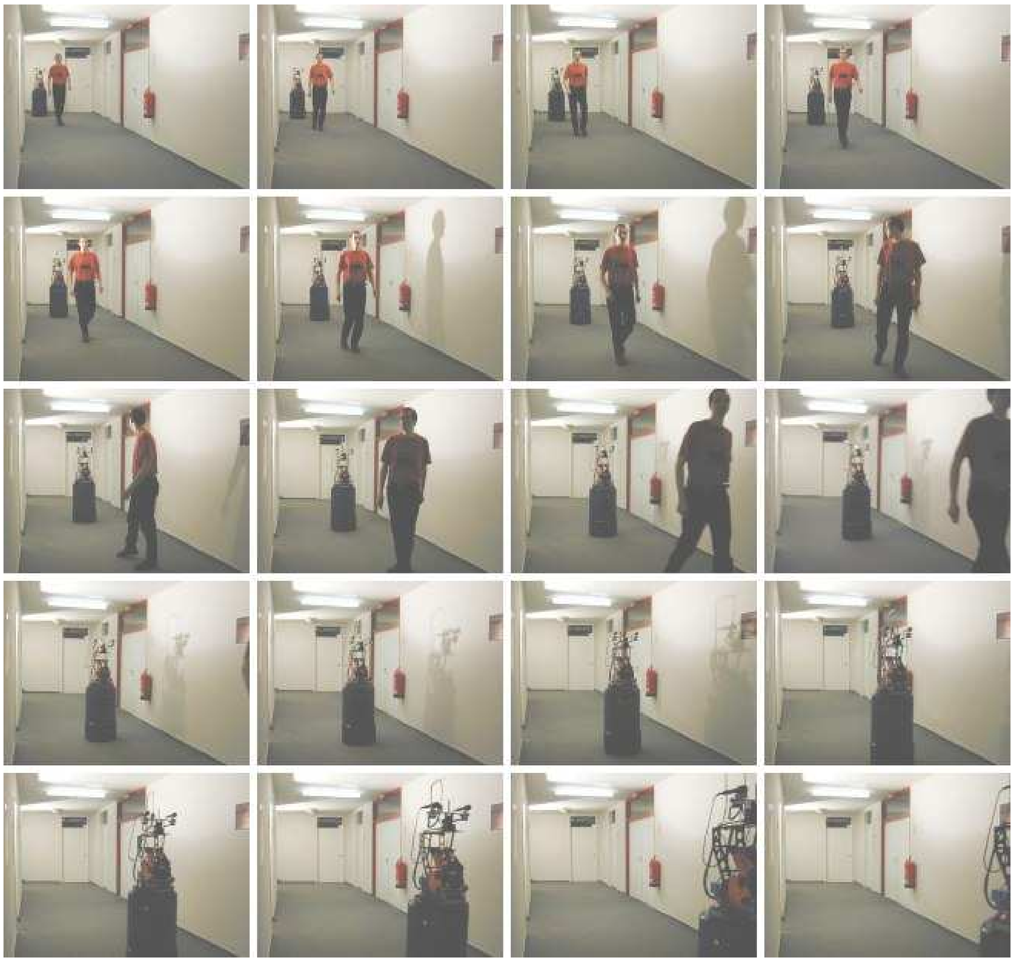


Abbildung 31: *Im Test des Gesamtsystems folgt der B21-Roboter einer Person durch einen geraden Gang.*

4.5.1 Das Folgen einer Person durch Gänge

Eine Untersuchung der eigentlichen Aufgabe des Roboters unter realen Bedingungen ist nur schwer reproduzierbar durchzuführen. Die zu verfolgende Person müsste für jeden Versuchslauf exakt die gleiche Strecke in jeweils der gleichen Geschwindigkeit zurücklegen. Daher werden exemplarisch zwei Versuchsläufe mit Bildsequenzen und ohne Werte dokumentiert.

Vor und bei den Versuchsläufen findet keine Kalibrierung der Entfernungsschätzung statt, da lediglich ein vorgegebener Abstand eingehalten werden soll. Maßangaben bezüglich der Entfernung zur Person sind nicht nötig. Es werden die in den vorhergehenden Versuchen bestimmten Parameter verwendet. Es findet nur minimale Koordination der Verhaltensmodule statt: Die Module *Richte Pan aus* und *Richte Basis aus* können gegenseitig ihre Ausgaben unterdrücken.

Abbildung 32: *Im Test des Gesamtsystems folgt der B21-Roboter einer Person um eine Flurbiegung.*

In den Versuchsläufen geht die Person einen Gang entlang (vgl. Abb. 31), geht um Flurecken (vgl. Abb. 32) oder durchquert Türen. Die Person wird anhand der Farbe des T-Shirts detektiert. Der Roboter soll einen Abstand von ca. 1.5 Meter halten.

Die Person kann sich beliebig bewegen und z.B. auch umdrehen oder Ausfallbewegungen vollziehen. Geht sie schneller als einen Meter pro Sekunde, bleibt das System hinter der Person zurück, da dies die maximale Translationsgeschwindigkeit des B21-Roboters darstellt. Der Abstand zur Person vergrößert sich in einzelnen Versuchsläufen zeitweilig auf bis zu 2.5 Meter, da das Verhaltensmodul *Halte Abstand* bisher nur drei Abstufungen der Geschwindigkeit vorsieht und ein Wechsel der Geschwindigkeit nur stattfindet, wenn die Entfernung über einen definierten Zeitraum Konstanz zeigt.

Das System löst die gestellte Aufgabe insgesamt sehr gut. Das Verhaltensmodul *Kontrolliere Zwangsbedingungen* verhindert zuverlässig die Bewegungen des Roboters, sobald die Person nicht mehr mit beiden Kameras fixiert wird. Ein Fehler im Sinne eines Wechsels der Aufmerksamkeit auf eine zweite, den Weg kreuzende Person mit den gleichen farblichen Merkmalen kann in dieser Weise allerdings nicht festgestellt werden. Eine Unterscheidung einer plötzlichen Bewegung von einer zweiten Bewegung einer ähnlichen Farbzusammenhangskomponente kann nicht auf Grund der verwendeten Daten vorgenommen werden. Im Abschnitt 5 werden Möglichkeiten zur robusteren Detektion von Personen aufgezeigt.

4.5.2 Simulation des Ausfalls der Pan-Tilt-Einheit

Im letzten Experiment wird das Verhalten des Roboters bei Ausfall von Komponenten der Hardware untersucht. In den Versuchsläufen folgt der B21-Roboter einer Person, welche eine definierte Strecke durch eine Tür und um eine Gangbiegung innerhalb einer Büroumgebung zurück-legt (vgl. Abb. 33). Durch das Abschalten der Stromversorgung der Pan-Tilt-Einheit wird ein Defekt der Hardware simuliert.

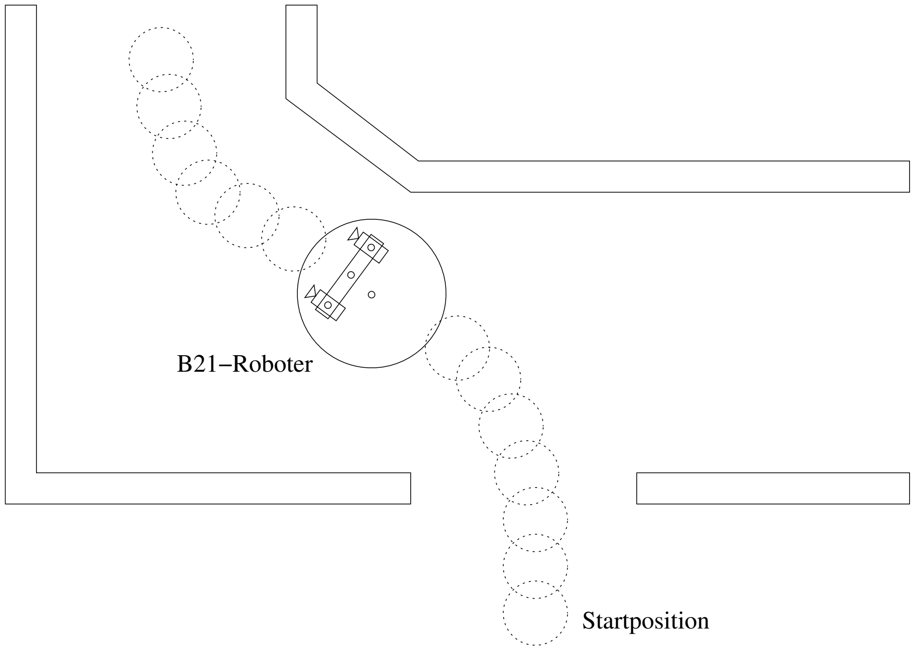


Abbildung 33: *Die zurückgelegte Strecke beim Test des Ausfalls der Pan-Tilt-Einheit ist schematisch in Draufsicht abgebildet. Durch gepunktete Kreise wird die Route des B21-Roboters durch die Tür und durch den Gang angedeutet.*

Es werden wieder die in den vorherigen Experimenten bestimmten Parameter benutzt. Zur Koordination der Verhaltensmodule können lediglich die Module *Richte Pan aus* und *Richte Basis aus* ihre Ausgaben gegenseitig unterdrücken. In Abb. 34 sind die Winkel zweier Läufe über die Zeit des Versuchs abgetragen. Im unteren Diagramm wird direkt nach Start des Systems die Stromversorgung der Pan-Tilt-Einheit abgeschaltet. Zum Vergleich ist im oberen Diagramm ein Versuchslauf mit intakter Hardware dargestellt.

In beiden Versuchsläufen wird die Person erfolgreich verfolgt. In den Diagrammen ist jeweils eine Ausrichtung der Basis zu den negativen Winkeln zu beobachten. Dies ist im Weg der Person (durch die Tür und nach links in den Gang) begründet. Im unteren Diagramm findet diese Ausrichtung kontinuierlicher und stärker statt. Die kompensatorischen Bewegungen der Vergenzen zum Ausgleich von Pan-Bewegungen fehlen im unteren Diagramm. Insgesamt findet die Rotation des Roboters bei dem simulierten Ausfall der Pan-Tilt-Einheit langsamer statt, da die Schwellenwerte zum Einsetzen der Kompensation durch die Basis bei einem Ausfall der Pan-Tilt-Einheit zu hoch liegen. Weiterhin müssen die Vergenzen stärker ausgelenkt werden um die Person nicht aus dem Sichtbereich der Kameras zu verlieren.

Insgesamt ist das System auch bei Ausfall der Pan-Tilt-Einheit noch einsatzfähig und kann einer Person folgen. Ein erfolgreiches Gesamtverhalten des Systems ergibt sich aus dem Zusammenwirken der restlichen Verhaltensmodule. Der Ausfall der Pan-Tilt-Einheit wird nicht

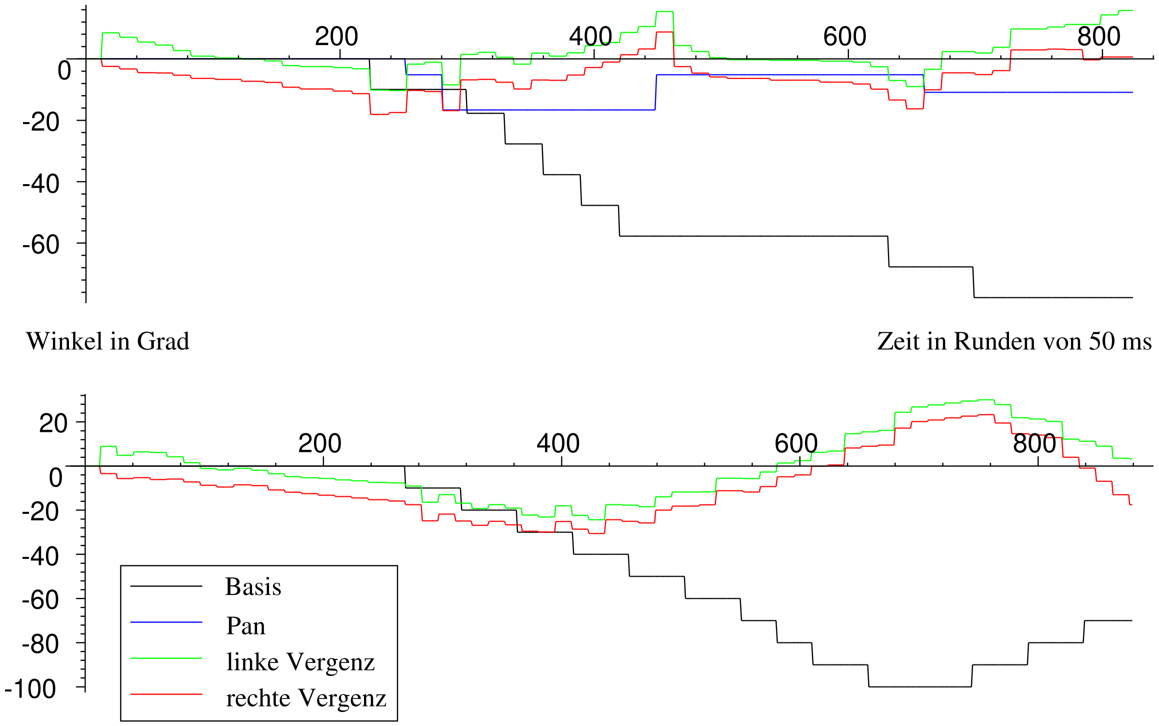


Abbildung 34: *Es sind die Stellwinkel der einzelnen Freiheitsgrade bei zwei Läufen des Experiments zum Ausfall der Pan-Tilt-Einheit dargestellt. Im oberen Diagramm folgt der B21-Roboter einer Person durch eine Tür um eine Biegung. Das untere Diagramm stellt den gleichen Ablauf dar, wobei aber die Stromversorgung für die Pan-Tilt-Einheit abgeschaltet ist.*

in der Software behandelt: Das Verhaltensmodul *Richte Pan aus* erteilt während des gesamten Versuchslauf Bewegungsbefehle, die mangels Stromversorgung nicht ausgeführt werden. Da die anderen Verhaltensmodule nur auf in der Umgebung wahrnehmbare Aktionen der Pan-Tilt-Einheit reagieren, hat dies keinen Einfluss auf das Verhalten des Roboters. Auch hier ist die Art der Koordination der Verhaltensmodule entscheidend: Wird zur Koordination eine Überdeckung der Ausgaben anderer Module durch das Verhaltensmodul *Richte Pan aus* gewählt, so würden kompensatorische Bewegungen veranlasst, die zum Verlust der Person aus dem Sichtbereich führen können. Um also eine Robustheit gegenüber Ausfällen der Hardware gewährleisten zu können sollte die gegenseitige Einflussnahme der Verhaltensmodule minimiert werden.

5 Diskussion und Ausblick

In diesem Abschnitt werden die erzielten Ergebnisse zusammengefasst und das realisierte System bewertet. Daraufhin werden im Ausblick Möglichkeiten zur Erweiterung des Systems aufgezeigt.

5.1 Diskussion des realisierten Systems

Ein B21-Roboter von Real World Interfaces wird befähigt eine Person visuell zu detektieren und mit einem Stereokamerasystem zu fixieren. Dabei werden die Freiheitsgrade der Vergenzen, der Pan-Tilt-Einheit und die Rotation der Basis genutzt. Für eine sinnvolle Kompensation von Rotationen entlang einzelner Freiheitsgrade ergeben sich durch diese Redundanz mehrere Möglichkeiten. Entfernt sich die Person, soll ihr in konstant bleibendem Abstand gefolgt werden. Die Aufgabe ist in Echtzeit zu lösen und es werden nur die Ressourcen genutzt, die der Roboter zur Verfügung stellt. Zur Detektion der Person in den Kamerabildern wird eine modifizierte Version der Tracker-Software aus [26] verwendet. Alle notwendigen Parameter wurden systematisch durch Experimente bestimmt.

Die Realisierung erfolgt durch ein verhaltensbasiertes System: Verhaltensmodule regeln die Bewegungen entlang einzelner Freiheitsgrade und weitere Module initiieren eine Translation zum Halten eines definierten Abstandes und zur Selbstkontrolle des Systems zur Vermeidung von Fehlern. Aus dem Zusammenwirken der Verhaltensmodulen mit der dynamischen, nicht modellierten Umwelt ergeben sich die beiden Verhaltensweisen des Fixieren und Folgens. Stehen die Verhaltensmodule im Konflikt, können sie durch die Mechanismen der Unterdrückung und der Überdeckung koordiniert werden. Das Gesamtverhalten des Systems hängt stark von der Art der Verwendung dieser Mechanismen und deren Parameter ab. Es ergeben sich unterschiedliche Bewegungsstrategien, die experimentell untersucht werden (vgl. Abschnitt 4.4.2).

Zur Steuerung des Roboters werden gebräuchliche Komponenten verwendet. Dies sind unter anderem die beiden an Bord des Roboters befindlichen Rechner und die Controller und Motoren der Vergenzen und der Pan-Tilt-Einheit. Die Ansteuerung über serielle Schnittstellen ist sehr viel langsamer, als z.B. über einen Feldbus oder über spezielle Hardware[25]. Dieses Problem wird in Abschnitt 4.1.2 untersucht und durch Anpassungen der Software ausgeglichen. Die einzelnen Verhaltensmodule werden nicht synchronisiert, so dass der jeweils für den zu regelnden Motor günstigste Takt benutzt werden kann, ohne die spezifischen Wartezeiten zu unterschreiten. Die gestellte Aufgabe kann in dieser Weise auch mit Standard-Hardware in Echtzeit gelöst werden.

5.1.1 Der verhaltensbasierte Ansatz

Einzelne Verhaltensmodule regeln jeweils die Motoren zur Rotation entlang eines Freiheitsgrades. Das Fixieren und die Kompensation von Bewegung wird also nicht durch eine zentrale

[25] Bradshaw et al. [8] benutzen z.B. spezielle Hardware zur Steuerung einer Pan-Tilt-Einheit. Den Motoren können dabei mit einem Takt von 500 Hz Befehle erteilt werden. Die in dieser Arbeit verwendete serielle Lösung zur Steuerung der Vergenzen erlaubt maximal zwei Bewegungsbefehle pro Sekunde.

Instanz gesteuert, sondern ergibt sich aus den Aktionen voneinander weitgehend unabhängiger Teile des Systems. Als Rahmen für die Realisierung werden Wahrnehmung und Handlung in einem engen Zyklus gekoppelt. Innerhalb dessen reagieren die einzelnen Verhaltensmodule asynchron auf ihre Wahrnehmung der Umwelt. Für einen solchen verhaltensbasierten Ansatz werden in Abschnitt 2.2 Anforderungen in der Form von vier Begriffen gestellt. Diese können im vorgestellten System wiedergefunden werden:

Situiertheit: Die einzelnen Verhaltensmodule reagieren auf die Position der in den Kamerabildern detektieren Person. Produziert ein Verhaltensmodul Ausgaben, die keine wahrnehmbaren Auswirkungen in der Umwelt haben, so haben diese Ausgaben auch keinen Einfluss auf das restliche System (vgl. Simulation des Ausfalls der Pan-Einheit in Abschnitt 4.5).

Körperlichkeit: Die Verhaltensmodule nehmen direkt Einfluss auf die Umwelt durch die Regelkreise zur Kontrolle der Motoren, welche die Bewegungen entlang der einzelnen Freiheitsgrade bewirken. Änderungen der Stellwerte schlagen sich direkt in der Wahrnehmung der Umwelt - der Position der detektierten Person in den Kamerabildern - nieder. Es werden alle zur Verfügung stehenden Freiheitsgrade benutzt, was eine gewisse Redundanz der Bewegungsmöglichkeiten ergibt.

Emergenz: Die Verhaltensmodule sind jeweils sehr einfach und können allein die gestellte Aufgabe nicht sinnvoll lösen. Die Verhaltensweisen des Fixierens und des Folgens ergeben sich aus dem Zusammenwirken der Verhaltensmodule und deren Reaktion auf ihre Wahrnehmung der Umwelt.

Kompetenz: Die Aufgabe des Fixierens und Folgens wird gelöst und das System zeigt sich dabei sehr robust, auch gegenüber nicht explizit modellierten Problemen, wie dem Ausfall eines der Gelenke (vgl. Abschnitt 5.1.3).

Die Motivation des Entwurfs ist nach der in Abschnitt 2.2 vorgestellten Unterscheidung experimentell ermittelt. Sämtliche relevanten Parameter und dadurch auch die Bewegungsstrategie bei der Kompensation ergeben sich durch eine stufenweise Untersuchung des Systems und seiner Einzelteile. Die Verhaltensweisen (und auch die einzelnen Verhaltensmodule) stehen in Konkurrenz zueinander. Konflikte werden durch die aus der *Subsumption Architecture* [9] bekannten Mechanismen der Unterdrückung (*Inhibition*) und Überdeckung (*Subsumption*) aufgelöst. Ebenso an die *Subsumption Architecture* angelehnt ist die *Bottom-Up*-Vorgehensweise bei der Konstruktion des Systems und der Verzicht auf globale Repräsentation der Umgebung. Es findet eine Abstimmung der Detektion der Person in den beiden Kamerabildern statt, um eine Entfernungsschätzung durchzuführen. Aber weder eine explizite 3D-Rekonstruktion noch eine andere modellhafte Darstellung der Welt wird benutzt. Weiterhin wird bis auf die Mechanismen zur Auflösung von Konflikten, wenig zwischen den Verhaltensmodulen kommuniziert.

Es stellt sich die Frage, ob die in Abschnitt 2.2.2 zur *Subsumption Architecture* geäußerte Kritik auch auf diese Arbeit zutrifft. Die Verfolgung des Leitsatzes *die Welt ist ihr eigenes Modell* führt hier nicht zu eigentümlichen Konstruktionen, wie bei Connell [18], sondern macht die

Lösung der Aufgabe stabiler als bei einem vergleichbaren, wissensbasierten System. Allerdings sind die in der vorliegenden Arbeit implementierten Verhaltensweisen im Vergleich mit den in [18] implementierten viel grundlegender. Der Verzicht auf globales Wissen und ein Modell der Umgebung führt hier nicht zu Problemen. Ähnlich wie bei Hartley und Pipitone [24] existieren Verhaltensmodule zur Selbstkontrolle des Systems; eine Garantie für ein erfolgreiches Gesamtverhalten kann dennoch nicht gegeben werden. Wieviel mehr Handlungskompetenz das System durch Erweiterung um Verhaltensmodule im gleichen Stil dazu gewinnen kann, ist ungewiss. Wie eine Erweiterung aussehen könnte, wird in Abschnitt 5.2 skizziert. Die in Abschnitt 1.3 zur Referenz vorgestellten Arbeiten legen ihre Schwerpunkte in andere Bereiche und verfolgen nicht direkt verhaltensbasierte Ansätze.

5.1.2 Die Kompensationsaufgabe

Die verhaltensbasierte Modellierung der Kompensation erweist sich in den Experimenten als erfolgreich. Eine *ideale Reaktion* des Systems kann zwar nicht garantiert werden, aber nach einer Anpassung der kritischen Parameter zeigt sich das Verhalten des Roboters als sehr zuverlässig. Die Entfernung wird auch in schwierigen Situationen ausreichend gut geschätzt, die einzelnen Motoren führen kaum unnötige Bewegungen aus und das System wirkt subjektiv ruhig und nicht hektisch. Durch die ungünstige Odometrie des Stereokamerasystems entstehen beim naiven Kompensieren von Bewegungen Fehler, die aber durch die toleranten Regelkreise ausgeglichen werden.

In den Experimenten erweist sich die komplexere, dreifache Kompensation als nicht weniger erfolgreich, als die doppelte Kompensation. In den Arbeiten von Waldherr [40], Feyrer und Zell [21] sowie von Ku und Tsai [39] werden die Kameras selber nicht durch Vergenzmotoren bewegt. Der Schwerpunkt liegt jeweils entweder auf der Detektion der Person oder auf der ruhigen Steuerung des Roboters. Keine der Arbeiten hat sich explizit mit dem Problem der Kompensation beschäftigt. Eine Benutzung dieses zusätzlichen Freiheitsgrades erweist sich in dieser Arbeit als vorteilhaft, durch den Gewinn zusätzlicher Robustheit. Die Kontrolle des Roboters wird auch bei Benutzung aller verfügbaren Freiheitsgrade durch den verhaltensbasierten Ansatz kaum aufwendiger. Die sich ergebenden Bewegungsstrategien legen tatsächlich den in Abschnitt 2 vorgestellten Vergleich mit menschlichen Bewegungsmustern nahe: Beim Fixieren eines sich bewegenden Objektes werden zuerst die Augen nachgeführt. Es folgt eine Bewegung durch Nacken und Hals. Bewegt sich das Objekt weiter, oder soll es verfolgt werden, muss der gesamte Rumpf gedreht werden.

5.1.3 Robustheit des Systems

Das realisierte System ist in mehrfacher Hinsicht robust. So ist z.B. eine genaue Kalibrierung des Roboters nicht notwendig. Die internen Parameter der Kameras, wie z.B. radiale und tangentiale Linsenverzerrung oder der exakte Schnittpunkt der optischen Achse mit der Bildebene haben zwar Einfluss auf die Steuerung des Roboters, aber dieser wird durch die Regelung ausgeglichen. Die genaue Lage der Koordinatensysteme der Kameras bzgl. des Roboters wird nicht explizit bestimmt. Eine zumindest ungenaue Bestimmung des Zusammenhangs zwischen Bewegungen der einzelnen Motoren und der Verschiebung von Objekten in den Kamerabildern muss stattfinden. Eine Verfeinerung dieser ungenauen Startwerte kann durch adaptive Regler geschehen (vgl. Anhang D). Die Bestimmung der exakte Lage der einzelnen Gelenke beim Start des Systems wird ebenso nicht ermittelt. Befindet sich z.B. die Pan-Einheit im Startzustand, so ist sie nicht exakt in Fahrtrichtung, sondern um etwa 2 Grad rotiert ausgerichtet. Auch diese Ungenauigkeit wird durch die Regelkreise ausgeglichen: Folgt der Roboter z.B. einer Person über einen langen, geraden Flur, so fährt das System nicht in einer geraden Linie. Die Route des Roboters weicht leicht vom idealen Weg ab, da die Ausrichtung der Pan-Einheit in Fahrtrichtung einen kleinen Fehler beinhaltet. Der Weg des Roboters schwankt leicht um diese ideale Route, aber die Person wird nicht aus dem Sichtbereich verloren.

Auf die Basisversuche aufbauend werden verschiedene Regelkreise und die Entfernungsschätzung untersucht. Um die Entfernungsschätzung beurteilen zu können, wird zuvor die genaue Ausrichtung der optischen Achsen der beiden Kameras ermittelt. Dieser Vorgang wird als *Kalibrierung der Entfernungsschätzung* bezeichnet. Die Versuche ergeben, dass die zur Kalibrierung verwendete Entfernung zur Laufzeit am genauesten geschätzt werden kann. Dieses Ergebnis ermöglicht es, gänzlich auf eine Kalibrierung zu verzichten. Die Person positioniert sich dazu beim Start des Systems einmal zur Instruktion im gewünschten Abstand zum Roboter. Da diese Entfernung zur Laufzeit am genauesten geschätzt wird, kann der tatsächliche Abstand des Roboters relativ zum so gelernten Abstand geregelt werden. Eine genauere Bestimmung des Abstandes ist nicht notwendig.

Das Gesamtsystem ist tolerant bezüglich des Ausfalls einzelner Komponenten der Software. Durch einen zentralen Prozess werden Teile des System neu gestartet und initialisiert, sobald sie nicht mehr ansprechbar sind. In Abschnitt 4.5 wird der Versuch zur Simulation des Ausfalls der Pan-Einheit beschrieben. In der Software wird so ein Ausfall nicht explizit modelliert. Durch die Redundanz der Bewegungsmöglichkeiten und durch das Zusammenwirken der Verhaltensmodule über Änderungen am wahrnehmbaren Zustand der Welt wird die Aufgabe auch so erfolgreich gelöst. Das Verhaltensmodul zur Steuerung der Pan-Einheit produziert ununterbrochen Ausgaben, die aber keinen Effekt auf die Umwelt haben und daher für die anderen Module nicht relevant sind.

Weitere Robustheit bei der Lösung der gestellten Aufgabe wird durch ein Verhaltensmodul zur Selbstkontrolle des Roboters erreicht. Durch die Überprüfung von Zwangsbedingungen können effektiv Fehlerzustände erkannt werden. Ist dieser Fehler z.B. an der Detektion der Person in einem der Kamerabilder fest zu machen, wird eine Korrektur vorgenommen um wieder einen stabilen Zustand zu erreichen.

5.2 Ausblick auf mögliche Erweiterungen

Die eigentliche Detektion der zu folgenden Person erfolgt allein durch ein schnelles, farbbasiertes Verfahren. Starke Helligkeitsänderungen oder den Sichtbereich kreuzende, ähnlich gefärbte Objekte können zu Problemen führen. Eine sinnvolle Erweiterung stellen alternative, z.B. nicht nur erscheinungsbasierte, Verfahren zur Detektion dar. Feyrer und Zell [21] benutzen z.B. neben Farbinformationen Konturen und die Detektion von Merkmalen des Gesichts der Person.

Ein zusätzliches Verfahren zur Detektion passt gut in das verhaltensbasierte Konzept, da so nicht mehrere Verhaltensmodule von einem Algorithmus zur Detektion abhängen. Eine derartige Erweiterung bietet sich an, da die Softwarekomponenten stark entkoppelt sind und die sehr effiziente Detektion durch die Tracker-Software [26] nur verhältnismäßig wenig Rechenzeit benötigt. Die Kombination mit weiteren Verfahren würde es ermöglichen, die farbbasierte Detektion zur Laufzeit z.B. an veränderte Beleuchtungsverhältnisse anzupassen.

Mit Ausnahme der adaptiven Regler werden alle Parameter durch Experimente bestimmt. Einige Parameter können nicht für jede Situation ideal gewählt werden, wie die oben erwähnte Anpassung an die Beleuchtungsverhältnisse. Eine Adaption von Verhaltensweisen ist daher wünschenswert. Die Bestimmung sinnvoller Wertebereiche durch Experimente ist dennoch unerlässlich, da nicht alle Parameter zur gleichen Zeit angepasst werden können und es sinnvolle Startwerte geben muss. Die Definition einer sinnvollen Zielfunktion ist sonst kaum möglich. Neben der Adaption ist das Erlernen von Verhaltensweisen denkbar; z.B. durch einen *Reinforcement*-Ansatz. Durch das Einbringen von im voraus erlangten Wissen gelang es Hailu [23], das sonst mit recht langen Trainingsphasen verbundene Reinforcement-Lernen zu beschleunigen. Es bietet sich an, die Koordination der einzelnen Verhaltensmodule zur Kompensation zu erlernen. Die Definition einer Zielfunktion ist dabei nicht trivial, da es eine größere Zahl von Variationsmöglichkeiten gibt. Ahrns [2] benutzt zur Navigation und Hindernisvermeidung ebenfalls Reinforcement-Lernansätze. Eine Erweiterung der Kompetenz und Robustheit des Systems durch Lernansätze und Adaption ist ein wünschenswerter Schritt.

Naheliegend ist eine Erweiterung um neue Verhaltensweisen, um die Handlungskompetenz des Systems zu erhöhen. Werden z.B. Verhaltensweisen zur Navigation oder Selbstlokalisation hinzugefügt, so muss das rein reaktive Konzept überdacht werden. Wie eine Verbindung des vollständig in *Bottom-Up*-Vorgehensweise erstellten Ansatzes mit einer weniger reaktiven Top-Down-Vorgehensweise realisiert werden kann, wird dabei die relevante Frage sein.

A Entfernungsschätzung

Fixieren beide Kameras ein Objekt, bilden die optischen Achsen der Kameras zusammen mit der Baseline[26] ein Dreieck (vgl. Abb. 35). Die Seitenhalbierende e schneidet die Achse des Pan-Gelenks und das fixierte Objekt. Die Länge der Verbindungsgrade EC wird als die zu schätzende Entfernung betrachtet. Der Winkel Ψ gibt die Ausrichtung des gesamten Roboterkopfes zum fixierten Objekt an.

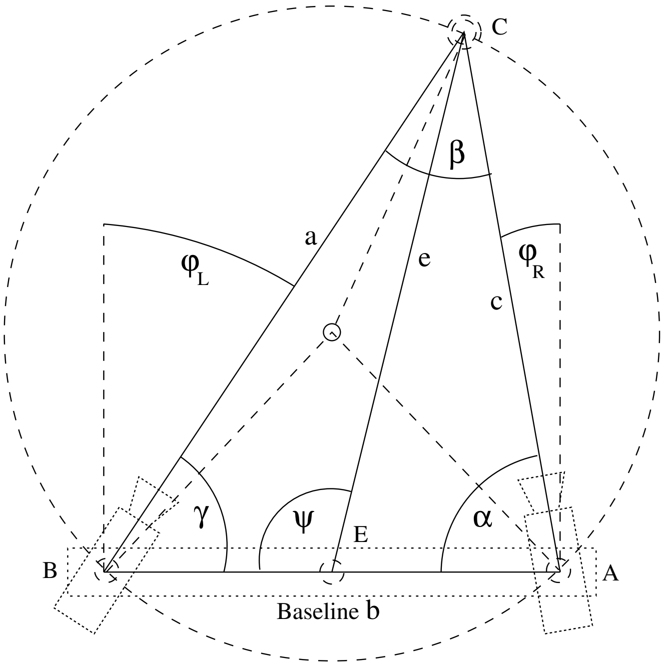


Abbildung 35: *Schematische Darstellung des Kopfes des Roboters und die benutzten geometrischen Größen bei der Entfernungsschätzung.*

Da sich zwei Dreieckswinkel aus den Stellwinkeln der Vergenzmotoren ergeben und die Länge der Baseline bekannt ist, können die restlichen Größen bestimmt werden. Da $\beta = 180 - (\alpha + \gamma)$ ist, folgt mit dem Sinussatz im Dreieck ABC:

$$a = \frac{\sin(\alpha) b}{\sin(\beta)}$$

Mit dem Cosinussatz im Dreieck CEB folgt:

$$e = \sqrt{(\frac{b}{2})^2 + a^2 - 2\frac{b}{2} a \cos(\gamma)} = \sqrt{\frac{b^2}{4} + a^2 - ba \cos(\gamma)}$$

Dabei sind:

$$\alpha = \varphi_R + \frac{\pi}{2} \qquad \gamma = -\varphi_L + \frac{\pi}{2}$$

[26]Die Verbindungslinie zwischen den optischen Zentren der beiden Kameras.

Die Variablen φ_L und φ_R stehen für die Stellwinkel des Motors der linken und der rechten Vergenz. Tatsächlich stimmen die Formeln nicht ganz mit der Realität überein, da die Ausgangsstellung der Vergenzen (die Ausrichtung in Fahrtrichtung des Roboters) nicht genau bekannt ist. Um die Entfernungsschätzung experimentell untersuchen zu können wird das in Anhang B vorgestellte Verfahren zum Bestimmen der genauen Ausgangsstellung benutzt.

A.1 Erweiterung des intuitiven Verfahrens

Bei einem tatsächlichen Lauf des Systems wird immer versucht die zu verfolgende Person zu fixieren, aber sie befindet sich im Regelfall nicht in der Bildmitte. Die Entfernung muss daher auch geschätzt werden können, wenn sich das Ziel lediglich im Blickbereich beider Kameras befindet. Für die Regelung der Vergenzen wird ein Zusammenhang zwischen der Position der Person in den 2D-Kamerabildern zu Motorstellwerten hergestellt. Der Regelungsschritt zum Fixieren wird im Moment der Schätzung nicht durchgeführt, sondern nur als Offset zu den aktuellen Stellwerten der Motoren verrechnet. Ist der Regelungsschritt ungenau, schlägt sich dies auch in der Schätzungen der Entfernung nieder.

A.2 Probleme bei der Schätzung

Ein weiteres Problem bei der Schätzung der Entfernung ist die langsame Ansteuerung der Hardware. Stammen die Kamerabilder, in denen die Person detektiert wird, nicht vom gleichen Zeitpunkt wie die Motorstellwerte der Vergenzen, kommt ein zusätzlicher Fehler zur Schätzung hinzu. Das Problem wird in Abschnitt 3.1.5 beschrieben und in Abschnitt 4.1.2 experimentell untersucht.

Die Person wird durch die Lokalisation einer Farbzusammenhangskomponente detektiert. Ist die Beleuchtung nicht homogen und kommt das Licht z.B. nur von einer Seite, dann liegt der Schwerpunkt der Zusammenhangskomponente in den beiden Kamerabildern an unterschiedlichen Positionen. Der Schnittpunkt der optischen Achsen - und damit die geschätzte Entfernung - liegt daher vor oder hinter der Person.

B Kalibrierung

In der Robotik gibt es einige Größen und Beziehungen zwischen verschiedenen Komponenten, die zum Funktionieren des jeweiligen Systems genau bestimmt werden. Im Folgenden soll ein kurzer allgemeiner Überblick gegeben werden um dann die in dieser Arbeit verwendete Kalibrierung vorzustellen. Ein großer Teil der im Rahmen einer Kalibrierung zu bestimmenden Werte fällt in eine dieser drei Kategorien:

Die externen Kameraparameter: Die Lage der optischen Zentren (die Ursprünge der Kamera-Koordinatensysteme) relativ zum Koordinatensystem des Roboters.

Die internen Kameraparameter: Die Parameter des optischen Systems der Kamera: z.B. der Schnittpunkt der optischen Achse mit der Bildebene, die Verzerrung des Kamerabilds oder die tangentiale und radiale Linsenverzerrung.

Die Beziehungen zwischen einzelnen Gelenken: Die Position (Rotation und Translation) und Ausgangsstellung der einzelnen Rotationsachsen zueinander.

Eine Kalibrierung ist manchmal aufwendig und kostet immer Zeit. Ist das System auf eine Kalibrierung angewiesen und wird diese durch laengere Verwendung z.B. eines bestimmten Gelenkes langsam verloren, muss zur Laufzeit rekalibriert werden. Von einer genauer Kalibrierung unabhängig zu sein ist daher ein Ziel der vorliegenden Arbeit.

Für die Regelung ist die Kenntniss der Beziehung zwischen Bewegungen der einzelnen Achsen und der Verschiebung des fixierten Objektes in den 2D-Kamerabildern nötig. Diese Beziehung wird in mehreren Versuchen über adaptive Regler grob bestimmt und zur Laufzeit des Systems angepasst. Bewegt sich das fixierte Objekt in den 2D-Kamerabildern, so ist niemals klar, ob dies nur auf eine Rotation einer oder mehrerer Achsen des Roboters zurückzuführen ist, oder ob sich das Objekt zusätzlich selber bewegt hat. Zur Laufzeit des Gesamtsystems werden aus diesem Grund die in dieser Weise grob bestimmten Werte benutzt, ohne dass eine Anpassung zur Laufzeit stattfindet. Diese Werte stehen in direktem Zusammenhang mit den externen Parametern des Kamerasystems. Die internen und exterenen Kameraparameter werden in der vorliegenden Arbeit allerdings niemals explizit bestimmt.

Um die Entfernungsschätzung qualitativ untersuchen zu können wird die Beziehung zwischen den beiden Kameras des Stereosystems bestimmt. Der Abstand der beiden Rotationsachsen zueinander ist leicht auszumessen. Die Rotation der Kameras ist weniger leicht zu messen, da die Ausgangsstellung der Kameras beim Start der Hardware sehr ungenau durch Tastschalter festgelegt wird (vgl. Abschnitt 3.1.5). Daher wird durch eine Kalibrierung die tatsächliche Ausgangsstellung (eine Ausrichtung in Fahrtrichtung des Roboters) bestimmt. Die auf diese Kalibrierung aufbauende Untersuchung der Entfernungsschätzung wird in Abschnitt 4.2 beschrieben. Zur Laufzeit des Systems ist eine solche Kalibrierung nicht nötig, da nur eine relative Entfernung zum fixierten Objekt bestimmt werden muss. Es muss kein Maß in einer bestimmten Einheit angegeben werden. Im Folgenden wird die Kalibrierung der Beziehung zwischen den beiden Kameras des Stereosystems genauer vorgestellt.

B.1 Kalibrierung der Lage der beiden Kameras zueinander

Bei der Kalibrierung wird die gleiche Rechnung wie bei der Entfernungsschätzung benutzt. Es wird lediglich in *entgegengesetzter Richtung* gerechnet. Die Entfernung des Objektes zum kamerasystem ist bekannt und die Winkel der Vergenzen werden berechnet. Diese so bestimmten Winkel werden mit den Stellwinkeln der Motoren verglichen. Die Differenz stellt den Offset der nominellen Ausgangsstellung der Vergenzmotoren beim Start der Hardware zur *idealen Ausgangsstellung* dar.

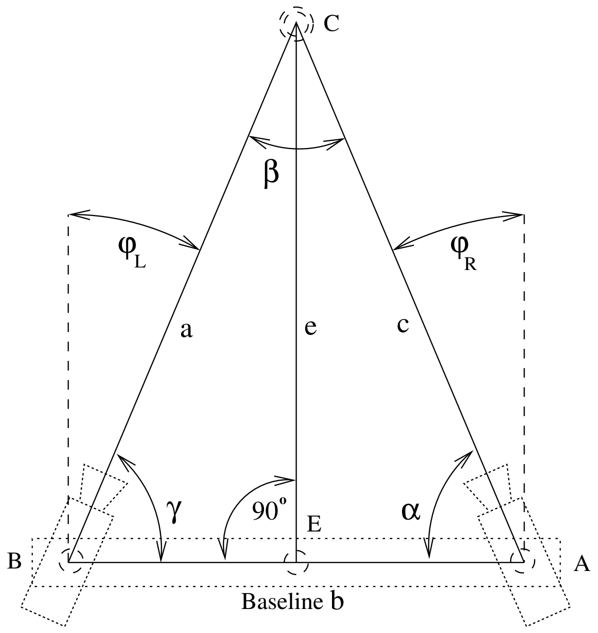


Abbildung 36: *Schematische Darstellung des Kopfes des Roboters und die benutzten geometrischen Größen bei der Kalibrierung.*

In Abb. 36 sind alle verwendeten Größen eingezeichnet. Da die Position des zu fixierenden Objektes frei gewählt werden kann, lässt sich die Rechnung durch den rechten Winkel der Geraden e zur Baseline b etwas vereinfachen:

$$\alpha = \gamma = \arctan(\frac{e}{(\frac{b}{2})})$$

Wird ein Objekt fixiert, kann entweder eine Entfernungsschätzung oder eine Kalibrierung vorgenommen werden. Je nachdem ob die Position des Objektes als bekannt vorausgesetzt wird, oder ob eine Kalibrierung vorgenommen werden soll. In Abschnitt 4.2 wird daher jeweils die Lage des zu fixierenden Objektes bestimmt um alle Werte des Versuches jeweils zur Kalibrierung und zur Entfernungsschätzung verwenden zu können.

C Kompensation

Die in der vorliegenden Arbeit verwendete Kompensation kann als *naiv* bezeichnet werden, da durch die einfache Rechnung immer ein gewisser Fehler entsteht. Im Folgenden soll diese naive Kompensation vorgestellt und der dabei entstehende Fehler abgeschätzt werden.

C.1 Naive Kompensation

In Abb. 37 ist ein einfaches Beispiel der doppelten Kompensation zu sehen. Eine Rotation entlang der Pan-Achse wird durch die Vergenzen ausgeglichen. Dabei wird lediglich der Winkel um den die Pan-Achse rotiert wird vom Stellwinkel der Vergenz abgezogen. Es entsteht ein Fehler in Form einer parallelen Verschiebung der optischen Achse der Kamera.

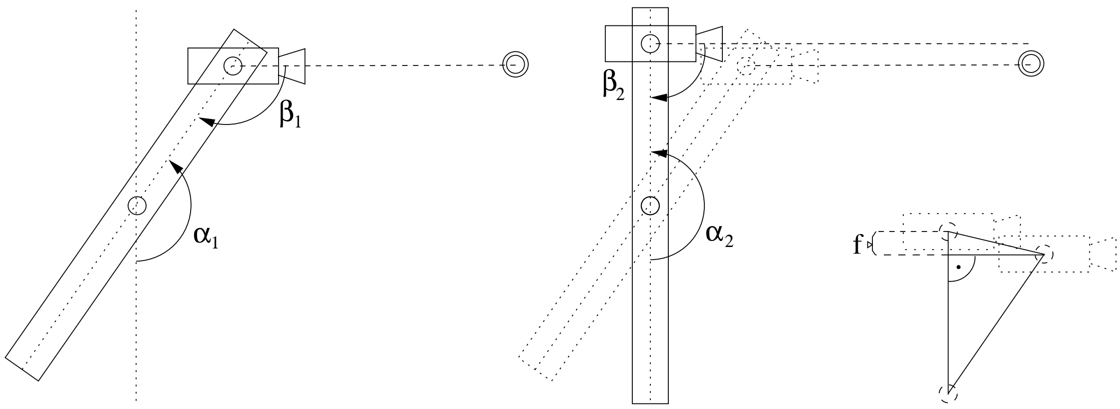


Abbildung 37: *Eine Bewegung der Pan-Achse wird durch die Vergenz naiv kompensiert. Der Stellwinkel α_1 wird zu α_2 und β_1 zu β_2 angepasst. Der Fehler f besteht aus der Verschiebung der optischen Achse der Kamera, welche jeweils als gestrichelte Linie eingezeichnet.*

Bei einer *genauen Kompensation* wäre die neue Lage der optischen Achse nicht parallel zur alten Lage, da die Entfernung zum fixierten Objekt berücksichtigt werden müsste. Der bei der naiven Kompensation entstehende Fehler soll genauer betrachtet werden.

C.2 Fehlerabschätzung

Der entstehende Fehler bei der naiven Kompensation hängt von den Abständen der einzelnen Rotationsachsen zueinander ab und wird im Folgenden durch die Größe der parallelen Verschiebung der optischen Achse der jeweiligen Kamera angegeben. Der Fehler in Form einer Verschiebung des fixierten Objektes in den 2D-Kamerabildern hängt wiederum von der konkreten Entfernung des Objektes vom Roboter ab. Um so näher das Objekt, desto größer der Fehler im Bild. Die naive Kompensation deckt sich mit der genauen Kompensation, wenn das fixierte Objekt unendlich weit entfernt ist.

Die einfachste Abschätzung ist die Addition der Abstände der parallelen Rotationsachsen multipliziert mit dem Faktor 2. Im Falle des B21-Roboters wären dies $15\ cm$ Abstand zwischen den

Rotationsachsen der Pan-Einheit und der Basis und $14.5\ cm$ Abstand zwischen der Achse der Pan-Einheit und der Achse der jeweiligen Vergenz. Es ergibt sich insgesamt ein maximaler Fehler von $59\ cm$. Der Fehler ist völlig unakzeptabel und ebenso unrealistisch: Diesem *worst case* liegen Rotationen von Basis und Pan-Einheit von jeweils 180 Grad zugrunde. Die Pan-Einheit und die Vergenzen könen sich auf dem B21-Roboter gar nicht soweit drehen und die Basis wird durch das in der vorliegenden Arbeit realisierte System ebenso nur um kleinere Winkel rotiert. Der begrenzende Einfluss der maximalen Auslenkung und der maximalen Schrittweite der jeweiligen Achse muss daher berücksichtigt werden.

An dem in Abb. 38 dargestellten Beispiel soll der maximale Fehler hergeleitet werden. Es findet eine Rotation der Pan-Achse um α Grad statt. Der neue Vergenzwinkel ergibt sich durch naive Kompensation: $\beta_2 = \beta_1 + \alpha$.

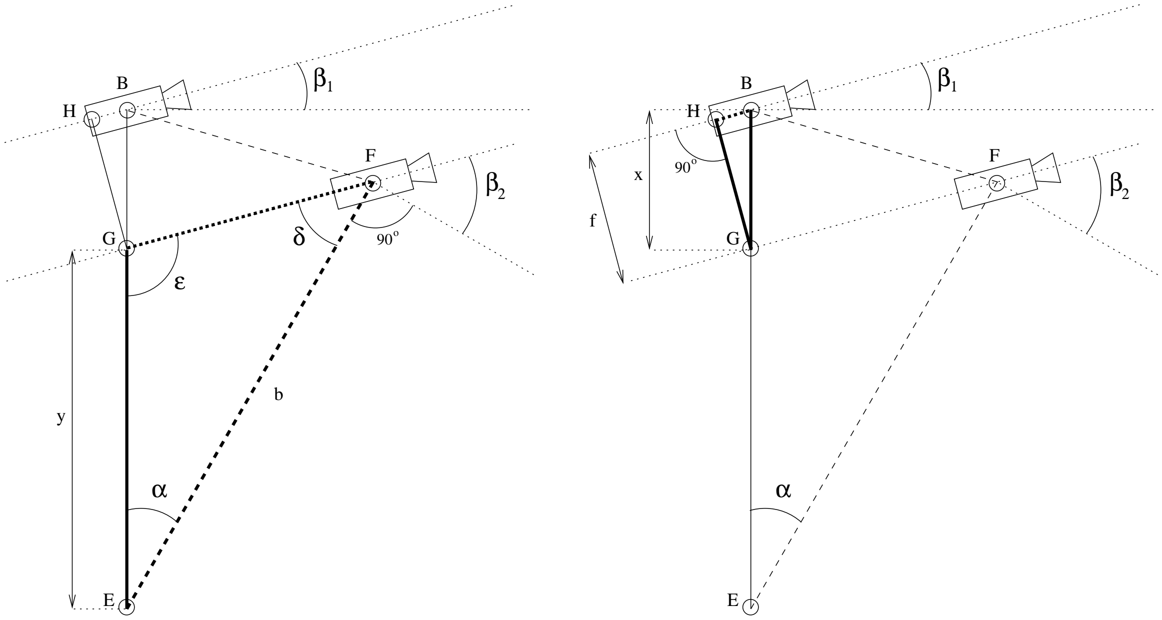


Abbildung 38: *Eine Bewegung der Pan-Achse wird durch die Vergenz naiv kompensiert. Der Fehler f besteht aus der Verschiebung der optischen Achse der Kamera. Die Herleitung des Fehlers findet in zwei Schritten im links bzw. rechts hervorgehobenen Dreieck statt.*

Die in Abb. 38 verwendeten Symbole:

E	Rotationsachse der Pan-Einheit
B	Rotationsachse der Vergenz vor Kompensation
F	Rotationsachse der Vergenz nach Kompensation
α	Rotationswinkel der Pan-Einheit
β_1	Stellwinkel der Vergenz vor Kompensation relativ zur Ausgangsstellung
β_2	Stellwinkel der Vergenz nach Kompensation relativ zur Ausgangsstellung
f	Die parallele Verschiebung der optischen Achse bei der Kompensation
b	Abstand der Rotationsachse der Vergenz zur Achse der Pan-Einheit
$G, H, x, \epsilon, \delta$	Hilfsgrößen

Der Fehler wird in zwei Schritten ermittelt: erst wird die Seite y im Dreieck EFG bestimmt, und dann der eigentliche Fehler f im Dreieck GBH. Die Dreiecke sind links bzw. rechts in Abb. 38 hervorgehoben. Im Dreieck EFG sind die Winkel α und $\delta = 90 - \beta_2$ bekannt. Weiterhin ist der Abstand der Rotationsachsen b gegeben. Mit der konstanten Winkelsumme im Dreieck und einer Anwendung des Sinussatzes ergibt sich:

$$y = \frac{b \sin(\delta)}{\sin(\epsilon)} = \frac{b \sin(90 - \beta_2)}{\sin(90 - \alpha + \beta_2)}$$

Im Dreieck GBH ergibt sich parallel:

$$f = x \sin(90 - \beta_1)$$

Bei den Versuchen zur dreifachen Kompensation wird eine maximale Schrittweite von 11.46 Grad verwendet. Die Auslenkung der Vergenzen ist stets kleiner als 25 Grad. Mit diesen Werten ergibt sich ein maximaler Fehler von $0.96\ cm$. Durch Addition des parallel hergeleiteten maximalen Fehlers bei Rotation der Basis (hier wird von einer maximalen Schrittweite von 20 Grad ausgegangen) ergibt sich insgesamt eine parallele Verschiebung der optischen Achse von $2.31\ cm$.

Dieser Fehler ist gut handhabbar und die naive Kompensation damit verwendbar. Eine Kompensation unter berücksichtigung der Entfernung zum fixierten Objekt bringt hingegen einige Nachteile: Im Ansatz der vorliegenden Arbeit werden die einzelnen Rotationsachsen unabhängig voneinander kontrolliert. Weiterhin hängt die Regelung der Achsen von der Qualität der Entfernungsschätzung ab.

D Grundlagen zur Regelung

Im Folgenden werden allgemein Regelsysteme und verschiedene Kombinationen von Reglern eingeführt, um dann die Verbindung zur Regelung in der vorliegenden Arbeit zu ziehen. Die Definitionen sind stark an [22] angelehnt.

D.1 Regelungssysteme

Die Idee der Regelung ist es, einem dynamischen System durch eine Stellgröße ein bestimmtes Verhalten gegen den Einfluss einer unvollständig bekannten Störgröße aufzuprägen. Dabei wird das System ständig beobachtet und die so gewonnene Information zur Anpassung der Stellgröße genutzt. In Abb. 39 ist ein schematischer Regelkreis zu sehen.

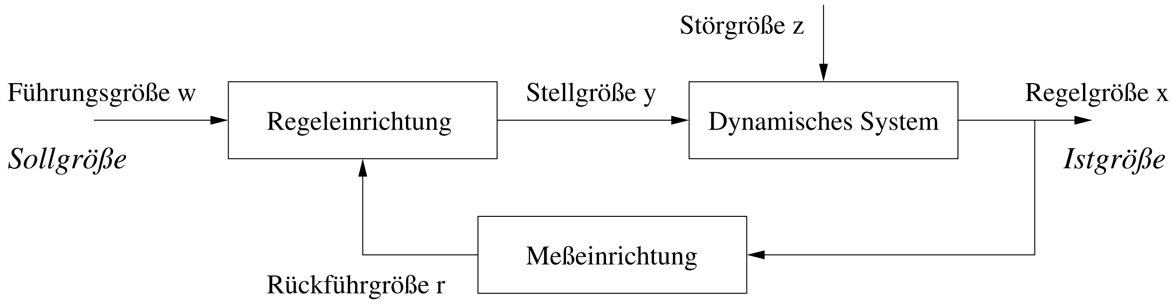


Abbildung 39: *Vereinfachtes Schema eines Regelkreises.*

Ist das Verhalten des dynamischen Systems in Form einer *Systemgleichung* bekannt, so kann aus deren Umkehrung und dem *Soll-Ist-Vergleich* von Führungsgröße und Rückführgröße der neue Wert der Stellgröße ermittelt werden[27].

Bei der Anpassung der Stellgröße muss im allgemeinen auf das dynamische Verhalten der Regelung geachtet werden um folgenden Anforderungen zu genügen:

1. Nach Änderung der Stör- oder Führgröße durchläuft die Regelgröße oft einen Einschwingvorgang. Klingt dieses Schwingen bei fortschreitender Zeit ab, so ist die Regelung *stabil*.

2. Die *stationäre Genauigkeit* bezeichnet den Abstand des Regelgröße von der Führungsgröße nach Abklingen des Einschwingens. Dieser soll hinreichend klein sein.

Sind die Grundforderungen nach Stabilität und einer gewissen stationären Genauigkeit erfüllt, so lässt sich ergänzen:

3. Als Reaktion auf eine Störgrößenänderung sollte keine starke Oszillation stattfinden. Es soll eine hinreichende Dämpfung vorhanden sein.

[27]Die Systemgleichung kann auch heuristisch oder experimentell ermittelt werden. Beispiele für solche Fälle sind in [22] und [37] zu finden.

4. Der Regelkreis soll genügend schnell sein: Bei einen Änderung der Störgröße soll die Regelgröße dem Sollwert in hinreichend kurzer Zeit nahekommen.

Diese qualitativen Forderungen müssen im konkreten Fall überprüft und gegeneinander abgewogen werden. Aufgrund ihrer Gegensätzlichkeit ist immer ein Kompromiss nötig.

D.2 Reglertypen

Diese Anforderungen werden meist durch einen Regler erfüllt, der eine Kombination aus den drei Grundtypen darstellt. Zu deren Erläuterung seien definiert:

w	Führungsgröße / Sollgröße
r	Rückführgröße
x	Regelgröße / Istgröße
$v(t)$	Änderung der Stellgröße zum Zeitpunkt t
$e(t) = w(t) - r(t)$	Regelabweichung zum Zeitpunkt t
κ	konstanter Koeffizient

Die Systemgleichung gibt wieder, wie sich aus den Zeitverläufen von Stell- und Führungsgröße die Regelgröße ergibt. Im Folgenden wird von dem einfachsten Fall ausgegangen, in dem die Systemgleichung durch einen konstanten Koeffizienten κ ausgedrückt werden kann.

Proportionalregler

Der P-Regler ist die einfachste Variante. Er reagiert mit einem zur Regelabweichung proportionalen Betrag.

$$v(t) = \kappa e(t)$$

Ist der Faktor κ zu groß gewählt, so wird der Regelkreis leicht instabil und gerät ins Schwingen. Außerdem wird der Sollwert im allgemeinen nicht exakt erreicht. Im diskreten Fall ist dies allerdings möglich.

Integralregler

Beim I-Regler wird über die zurückliegenden Regelabweichungen integriert. Es wird eine Historie von Werten berücksichtigt.

$$v(t) = \kappa \int_0^t e(t)dt$$

Die stationäre Genauigkeit ist wesentlich besser als bei einem reinen P-Regler, aber ein I-Regler ist auch im allgemeinen langsamer.

Folgende Modifikationen lassen sich noch anbringen: Die Intervallgröße kann beschränkt werden. Es werden also nicht alle vorhergehenden Werte berücksichtigt, sondern nur eine definierte Anzahl. Eine Gewichtung je nach Alter ist möglich. Gerade eben noch aktuelle Werte bekommen so mehr Einfluss als länger zurückliegende.

Differentialregler

Ein D-Regler setzt die Änderungsgeschwindigkeit der Regelabweichung in einen Stellwert um. Ändert sich die Regelabweichung stark, wird ebenso stark reagiert. Bleibt die Abweichung konstant, wird der Stellwert nicht geändert. Ein Differentialregler ist damit nur in Kombination mit einem der vorherigen Regler sinnvoll einzusetzen.

$$v(t) = \kappa \frac{de(t)}{dt}$$

Auch hier kann das Differential über einen begrenzten Bereich gebildet werden, und eine Gewichtung ist ebenso möglich.

Ein D-Regler wird oft zur schnellen Annäherung an den Sollwert (also zur Verbesserung der Dynamik) benutzt (siehe Forderung 4. im Abschnitt D.1). Bei starkem Systemrauschen oder Störschwingungen kann es allerdings dazu kommen, dass diese verstärkt werden.

Mischformen

Die einzelnen Regler können durch Reihenschaltung kombiniert werden. Dabei können die einzelnen Anteile wiederum gewichtet werden. Laut [22] werden meistens PI- oder PID-Regler eingesetzt. In [35] und [14] wurden reine Proportionalregler verwendet, da aufgrund der diskreten Werte und der genauen Hardwareansteuerung kein höherer Aufwand nötig schien.

D.3 Regelung in dieser Arbeit

Die beiden in Abschnitt 3.1.5 beschriebenen Vergenzen werden einzelnd geregelt, um ein bestimmtes Objekt fortlaufend zu fixieren. Der Regelkreis erhält jeweils den Stellwinkel des Vergenzmotors und die detektierte 2D-Position des Objektes im Kamerabild.

Sollgröße w und Rückführgröße r sind also Bildkoordinaten und die Stellgröße der Winkel des Vergenzmotors. Aufgrund des geringen Blickwinkels der verwendeten Kameras[28], kann der lineare Zusammenhang zwischen dem horizontalen Versatz des Schwerpunktes des detektierten Objektes im Kamerabild und dem Vergenzwinkel durch einen konstanten Faktor ausgedrückt werden.

Die Positionen im Bild sind zwar subpixelgenau, aber dennoch diskret. Die Systemgleichung ist zwar vollständig bekannt und einfach, aber die Störgröße z ist völlig unvorhersehbar. Durch einen knapp gewählten Zeitrahmen bei der Aufnahme der Bilder und dem Bewegen der Vergenzen sind die aufgenommenen Bilder nicht unbedingt vom gleichen Zeitpunkt, wie die aktuellen Vergenzwinkel. Dieser Fehler kann sich als eine Art Störschwingen bemerkbar machen.

[28] Die in Abschnitt 3.1.4 beschriebenen Kameras verzerren das Bild kaum. Im Gegensatz dazu wurden zum Beispiel in [41] Weitwinkel-Objektive benutzt, die den linearen Zusammenhang etwas verzerren.

Bemerkung zum adaptiven Regler

Es wird zunächst ein adaptiver Regler verwendet, welcher den Wert κ anhand der Fehler der letzten Schritte anpasst. Die Wirkung der Adaption kann nicht einfach am Erfolg des letzten Regelungsschritts gemessen werden, da nie eine erneute Eigenbewegung des fixierten Objektes ausgeschlossen werden kann. Der Nutzen beschränkt sich darauf, den optimalen Wert zu finden und diesen möglichst zu erhalten. Dieser Wert läßt sich unter Versuchsbedingungen leicht messen. Zur Laufzeit der Anwendung muss der Wert nicht angepasst werden. Das System wäre lediglich flexibler bezüglich der verwendeten Hardware.

Weiterhin wird eine Adaption des Verhaltens in Betracht gezogen. Um Fehler oder Erfolg der Adaptionen des Verhaltens oder der Regelung zuordnen zu können muss eine komplexe Zielfunktion definiert werden. Da die Adaption des Verhaltens interessanter scheint, wird ein vorher experimentell ermittelter Wert κ für die Regelung verwendet.

Bemerkung zur alleinigen Verwendung von Regelungssystemen

Als alternativen Ansatz könnte man die gesamte Steuerung von Vergenzen, Pan und Basis als Regelungsaufgabe ansehen. Es wären dann drei Stellgrößen und eventuell mehrere Regelungsgrößen in einem System zu vereinen. Dagegen spricht das relativ aufwendige Regelungssystem und der Wunsch verschiedene Verhalten zu implementieren. Letzteres bedeutet zum Beispiel, dass der Roboter in einem Fall die Basis schnell ausrichten soll, um das anvisierte Ziel auch zu erreichen, und in einem anderen Fall die Basis explizit nicht ausgerichtet sein soll, um ein Hindernis zu umfahren.

Abbildungsverzeichnis

Tabellenverzeichnis

Literatur

[1] AGRE, P. ; CHAPMAN, D.: Pengi: An implementation of a theory of activity. In: *Proceeding of Sixth National Conference on Artificial Intelligence, AAAI-87, Seattle, WA* (1987), S. 268–272

[2] AHRNS, I.: *Ultraschallbasierte Navigation und adaptive Hindernisvermeidung eines autonomen mobilen Roboters*, Lehrstuhl für Kognitive Systeme des Instituts für Informatik der Christian-Albrechts-Universität zu Kiel, Diplomarbeit, 1996

[3] AHRNS, I. ; BRUSKE, J. ; HAILU, G. ; SOMMER, G.: Neural fuzzy techniques in sonar-based collision avoidance. In: JAIN, L.C. (Hrsg.) ; FUKUDA, T. (Hrsg.): *Soft Computing for Intelligent Robotic Systems*. Physica-Verlag (Springer), 1998 (Studies in Fuzziness and Soft Computing), S. 185–214

[4] ARBIB, M. ; HOUSE, D.: Depth and detours: An essay on visually guided behavior. In: ARBIB, M. (Hrsg.) ; HANSON, A. R. (Hrsg.): *Vision, Brain, and Cooperative Computation*. MIT Press, Cambridge MA., 1987

[5] ARKIN, R. C.: Motor Schema-Based Mobile Robot Navigation. In: *International Journal of Robotics Research* 8 (1989), Nr. 4, S. 92–112

[6] ARKIN, R. C.: *Behavior-Based Robotics*. MIT Press, Cambridge MA., 1998 (Intelligent Robotics and Autonomous Agents series)

[7] BOGONI, L.: Subsumption architecture and discrete event systems: A comparison / Department of Computer Science, University of Pennsylvania, Philadelphia. 1993. – Forschungsbericht

[8] BRADSHAW, K.J. ; MCLAUCHLAN, P.F. ; REID, I.D. ; MURRAY, D.W.: Saccade and pursuit on an active head/eye platform. In: *Image and Vision Computing* 12(3) (1994), S. 155–163

[9] BROOKS, R.: A robust layered control system for a mobile robot. In: *IEEE Journal of Robotics and Automation* RA-2 (1986), Nr. 1, S. 14–23

[10] BROOKS, R.: *Integrated Systems Based on Behaviors*. 1991. – Special issue of SIGART on Integrated Intelligent Systems

[11] BROOKS, R.: Intelligence Without Representation. In: *Artificial Intelligence* 47 (1991), Nr. 1–3, S. 139–159

[12] BROOKS, R.: From earwigs to humans. In: *Robotics and Autonomous Systems* 20 (1997), S. 291–304

[13] BRUNS, S.: *Bestimmung der Gesichtspose unter Anwendung von Gaborwaveletnetzen*, Lehrstuhl für Kognitive Systeme des Instituts für Informatik der Christian-Albrechts-Universität zu Kiel, Diplomarbeit, 2000

[14] BUNTEN, A.: *Autonome Ausrichtung eines Stereokamerakopfes auf einen Manipulatorarbeitsbereich*, Lehrstuhl für Kognitive Systeme des Instituts für Informatik der Christian-Albrechts-Universität zu Kiel, Studienarbeit, 2000

[15] BURGARD, W. ; CREMERS, A. ; FOX, D. ; HAHNEL, D. ; LAKEMEYER, D. ; SCHULZA, D. ; STEINER, W. ; THRUN, S.: Experiences with an Interactive Museum Tour-Guide Robot. In: *Artificial Intelligence* 114 (1999), Nr. 1-2, S. 3–55

[16] COLOMBETTI, M. ; DORIGO, M. ; BORGHI, G.: Behavior Analysis and Training - A Methodology for Behavior Engineering. In: *IEEE Transactions on Systems, Man, and Cybernetics - Part B* 26 (1996), June, Nr. 3, S. 365–380

[17] Compumoter Division: *Software Reference Guide Microstepper Products*. 1997. – Hardwarebeschreibung der Vergenz- Motoren und Controler am B21

[18] CONNELL, J.: A Behavior-Based Arm Controller. In: *IEEE Transactions on Robotics and Automation* 5 (1989), Nr. 6, S. 784–91

[19] Directed Perception Inc.: *Computer Controlled Pan-Tilt Unit Users's Manual*. 1994. – Hardwarebeschreibung der Pan-Tilt-Einheit am B21

[20] FEDOR, Christopher: *TCX - An interprocess communication system for building robotic architectures*, Carnegie Mellon University, School of Computer Science, Robotics Institute, Diplomarbeit, 1994

[21] FEYRER, S. ; ZELL, A. *Tracking and Persuing Persons with a Mobile Robot*. 1999

[22] FÖLLINGER, O.: *Regelungstechnik*. Hüthig Buch Verlag Heidelberg, 1994

[23] HAILU, G.: *Towards Real Learning Robots*, Inst. f. Informatik u. Prakt. Math. der Christian-Albrechts-U niversität zu Kiel, Diss., 1999

[24] HARTLEY, R. ; PIPITONE, F.: Experiments with the subsumption architecture. In: *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, 1991, S. 1652–1658

[25] HATZACK, Wolfgang: *RHINO Software Documentation V1.3*. RHINO Team, Universität Bonn, 1995

[26] KRÜGER, V. ; HERPERS, R. ; DANIILIDIS, K. ; SOMMER, G.: Teleconferencing using an attentive camera system. In: *AVBPA*, 1999, S. 142–147

[27] KRÜGER, V. ; MAHLMEISTER, U. ; SOMMER, G.: Attentive Face Detection and Recognition. In: *P. Levi (Ed.), 20. DAGM Symposium Mustererkennung, Stuttgart, Germany*, Springer-Verlag Berlin/Heidelberg, 1998

[28] MAES, P.: Adaptive action selection. In: *Proceedings Thirteenth Annual Conference of the Cognitive Science Society* (1991), S. 108–113

[29] Matrox Electronic Systems: *Matrox Meteor Hardware Reference*. 1997. – Hardwarebeschreibung der Meteor Framegrabber-Karten von Matrox

[30] NILSSON, N. J.: Shakey the Robot, SRI International Technical Note 323, 1984 / SRI International. 1984. – Forschungsbericht

[31] PAULI, J.: *Learning-Based Robot Vision*. Springer-Verlag Berlin/Heidelberg, 2001

[32] Real World Interface Inc.: *System Software and RAI-1 Documentation*. 1996. – Beschreibung der Treiber und Schnittstellen des B21-Roboters

[33] Real World Interface Inc.: *BeeSoft User's Guide and Software Reference*. 1997. – Beschreibung der Softwareschnittstelle des B21-Roboters

[34] ROBOCUP FEDERATION, The. *RoboCup Official Site*. http://www.robocup.org

[35] SCHMIDT, A.: *Manipulatorregelung durch fortwährende visuelle Rückkopplung*, Lehrstuhl für Kognitive Systeme des Instituts für Informatik der Christian-Albrechts-Universität zu Kiel, Diplomarbeit, 1998

[36] SCHOPPERS, M. J.: Universal Plans for Reactive Robots in Unpredictable Environments. In: *Proceedings of the 10th International Joint Conference on Artificial Intelligence (IJCAI 87)*. Milan, Italy, 1987, S. 1039–1046

[37] SOLTYSIAK, R.: *Wissensbasierte Prozeßregelung*. Oldenbourg-Verlag, 1989

[38] SOMMER, G.: The global algebraic frame of the perception–action cycle. In: JÄHNE, B. (Hrsg.) ; HAUSSECKER, H. (Hrsg.) ; GEISSLER, P. (Hrsg.): *Handbook of Computer Vision and Applications* Bd. 3. Academic Press, San Diego, 1999, S. 221–264

[39] TSAI, W. ; KU, C: Smooth Vision-Based Autonomous Land Vehicle Navigation in Indoor Environments by Person Following Using Sequential Pattern Recognition. In: *Journal of Robotic Systems* 16 (1999), Nr. 5, S. 249–262

[40] WALDHERR, S.: *Gesture Recognition on a Mobile Robot*, School of Computer Science at the Carnegie Mellon University / Lehrstuhl für Kognitive Systeme des Instituts für Informatik der Christian-Albrechts-Universität zu Kiel, Diplomarbeit, 1998

[41] WALDHERR, S. ; THRUN, S. ; ROMERO, R. ; MARGARITIS, D.: Template-based recognition of pose and motion gestures on a mobile robot. In: *In Proceedings of Annual AAAI Conference* (1998)

[42] WILSON, M. ; KING, C.: Evolving hierarchical robot behaviours. In: *Robotics and Autonomous Systems* 22 (1997), Nr. 3-4, S. 215–230

Danksagung

An dieser Stelle möchte ich all den Personen danken, die zum Gelingen dieser Arbeit beigetragen haben.

Ich danke Bodo Rosenhahn und Volker Krüger für ihre konstruktive und geduldige Betreuung dieser Diplomarbeit. Ferner danke ich dem Lehrstuhl für kognitive Systeme für hilfreiche Diskussionen und eine angenehme Atmosphäre. Insbesondere gilt mein Dank auch den technischen Mitarbeitern des Lehrstuhls, die bei technischen Problemen mit viel Geduld umgehend geholfen haben. Mein Dank gilt desweiteren Josef Pauli für anregende Diskussionen und konstruktive Hinweise zur Literatursuche.

Für die Korrektur meiner zahllosen Rechtschreib- und Grammatikfehler sowie für konstruktive Kritik bedanke ich mich bei Antje, Bioern, Christian, Georg, Petra, Sven und Timm.

Erklärung

Hiermit versichere ich, daß ich die vorliegende Arbeit selbständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe.

Kiel, den 20. Dezember 2001

- -

Andreas Bunten