Mensch-Roboter-Interaktion in gestenbasierten Zweipersonenspielen

NAO spielt Schere, Stein, Papier.

Masterarbeit

im Studiengang Informatik
der technischen Fakultät
der Friedrich-Alexander Universität Erlangen-Nürnberg
in Kooperation mit der Otto-Friedrich Universität Bamberg

Verfasser: André Kowollik
Matrikelnummer: 21626540
Datum: 07.01.2014

Betreuer Universität Erlangen-Nürnberg: Prof. Dr. Lutz Schröder
Betreuer Universität Bamberg: Prof. Dr. Ute Schmid

Abstrakt

Serviceroboter, die in der Zukunft Menschen in ihrem alltäglichen Leben unterstützen sollen, sei es als Haushaltshilfe, Pflegedienst, Tutor oder Spielgefährte, müssen in das komplexe Lebensumfeld der Menschen nahtlos integriert werden. In dieser Masterarbeit wird daher die Mensch-Roboter-Interaktion unter den folgenden Aspekten betrachtet:

- **Technik**: Ist es möglich, eine Mensch-Mensch-Interaktion auf einen technisch mittelmäßigen ausgestatteten Roboter zu übertragen?
- **Interaktionsmöglichkeit**: Wird von Menschen die Interaktion mit einem Roboter, per Sprache und Gestik, tatsächlich als intuitiv und natürlich angesehen?
- **Verhalten**: Was für eine Verhaltensweise muss ein Roboter in Situationen aufweisen, in denen mit Menschen interagiert wird?
- **Akzeptanz**: Welche Auswirkung hat die Verhaltensweise auf die Akzeptanz eines Roboters?

Das Spiel "Schere, Stein, Papier" ist eine Situation, in der normalerweise eine zwischenmenschliche Interaktion stattfindet. Diese Interaktion wurde auf den Roboter (NAO) übertragen, indem eine simple Spracherkennung implementiert, eine Gestenerkennung auf Basis des Frameworks OpenCV und verschiedene Spielstrategien entwickelt wurden. Die Strategien stellen dabei vereinfachte Verhaltensmuster von NAO dar. Zur Bewertung der Mensch-Roboter-Interaktion und der Verhaltensmuster wurde eine Nutzerstudie durchgeführt.

Als Ergebnis kann gesagt werden, dass eine Mensch-Mensch-Interaktion generell auf einen Roboter mit mittelmäßiger Ausstattung übertragen werden kann. Damit die Gestenerkennung aber auch in der Praxis eingesetzt werden kann, müsste dazu eine andere Kameratechnik eingebaut werden. Die Interaktion per Gestik und Sprache wurde von den meisten Testpersonen als natürlich und intuitiv eingeschätzt, und sie würden diese Art der Interaktion auch verwenden, wenn es einige Randbedingungen zu beachten gäbe. Die Verhaltensweise eines Roboters muss an die Situation und an das soziale Umfeld angepasst sein. Diese These wurde durch die Nutzerstudie bekräftigt. Aus ihr lässt sich auch ableiten, dass die situativ richtige Verhaltensweise einen positiven Einfluss auf die Akzeptanz von Robotern hat.

Abstract

Service Robots which will support humans in the future with their everyday life, be it as a family care worker, nursing service, teacher or playmate, have to be seamlessly integrated into the human environment. In this thesis, therefore, the human-robot-interaction is considered under the following aspects:

- **Technology:** Is it possible to transfer a human-human interaction to a technically mediocre equipped robot?
- **Interaction:** Will people actually experience the interaction with a robot by voice and gesture as intuitive and natural?
- **Behavior:** What behavior should have a robot in a situation, in which he interacts with a human being?
- **Acceptance:** What impact has a behavior on the acceptance of a robot?

The game "Rock, Paper, Scissors" is a situation in which always a human-human-interaction occurs. That interaction was transferred to the robot NAO by using a simple voice recognition and by implementing a gesture recognition based on the OpenCV framework as well as various game strategies. Those strategies represent simple behavior patterns for the robot, in that particular situation. To evaluate the human-robot interaction and the game strategies, a user study was conducted.

As a result, it is safe to say that a human-human interaction can be applied to a robot with mediocre equipment generally. For using the developed gesture recognition in practice, a different camera technique should be installed. The interaction by gestures and voice was considered by most participants of the user study as natural and intuitive. They would also use this type of interaction, even if specific (unnatural) constraints have to be considered. The behaviors of a robot must be adapted to the situation and to the social environment. This hypothesis was confirmed by the user study. From the user study can also be derived that a situational correct behavior has a positive influence on the acceptance of robots.

Danksagung

An dieser Stelle möchte ich mich bei all denjenigen bedanken, die mich bei der Anfertigung dieser Master-Thesis unterstützten.

Ein besonderer Dank geht an Frau Prof. Dr. Ute Schmid der Otto-Friedrich-Universität Bamberg, für ihre Betreuung und hilfreichen Anregungen während der gesamten Zeit.

Ein besonderer Dank geht auch an Herrn Prof. Dr. Lutz Schröder der Friedrich-Alexander-Universität Erlangen-Nürnberg. Ohne seine Bereitschaft, die Masterarbeit ebenfalls zu betreuen, wäre diese nicht zustande gekommen.

Danken möchte ich auch Herrn Peter Kamionka, welcher durch seine Anregungen und Vorschläge zur Steigerung der sprachlichen Qualität der Inhalte maßgeblich dazu beigetragen hat.

Nicht zuletzt möchte ich mich bei meinen Eltern bedanken, die mir das Masterstudium ermöglichten und mich die komplette Zeit moralisch unterstützten.

Inhaltsverzeichnis

Darstellungsverzeichnis

Kapitel 1

Einleitung

1.1. Motivation

Im Jahr 2014 feiern die Industrieroboter ihren 60. Geburtstag. 1954 meldete George Devol ein Patent für einen programmierbaren Manipulator an. Seit dem ist die Technik immer besser und ausgeklügelter geworden. Heutzutage sind die mechanischen Helfer aus der Industrie nicht mehr wegzudenken. Sie erledigen ständig wiederkehrende Arbeitsschritte unermüdlich, mit hoher Präzision und gleichbleibender Leistung. Das macht sich nicht nur in der Qualität der Produkte bemerkbar, sondern auch in den Preisen. Wenn die Roboter in der Industrie einen derart großen Nutzen haben, ist eine Frage natürlich naheliegend: Warum existieren für den gewerblichen und privaten Bereich bisher nur so wenige Roboter? Eine vermeintliche Antwort darauf lautet in vielen Fällen, dass die Technologie noch nicht weit genug dafür fortgeschritten sei. Eine Machbarkeitsstudie des Bundesministeriums für Bildung und Forschung [1] gibt dieser Aussage aber nur zum Teil recht. Im Zeitraum von Dezember 2009 bis November 2010 haben die Fraunhofer-Institute für Produktionstechnik und Automatisierung IPA sowie für System- und Innovationsforschung ISI, Servicerobotik - Anwendungen entwickelt, und anhand von Szenarien die technische und wirtschaftliche Machbarkeit analysiert. Dabei kamen sie zu dem Ergebnis, dass die Technik für die meisten Anwendungen generell ausreichen würde. Wirtschaftlich gesehen ist es jedoch aufgrund hoher Kosten nicht rentabel [1 p. 7]. Zusätzlicher Forschungsbedarf wurde außerdem in den Bereichen der Wahrnehmung, Navigation und Manipulation identifiziert, was meist in Zusammenhang mit insuffizient gelösten Softwareproblemen steht [1 p. 7]. Weiterhin stehen der Verbreitung von Servicerobotern, neben den technischen Anforderungen, auch Aspekte, wie Flexibilität, Sicherheit, Akzeptanz sowie hohe Entwicklungs- und Systemkosten im Wege [1 p. 11].

Die Mensch-Roboter-Interaktion steht in der Studie des BMBF zwar nicht im Vordergrund, spielt aber eine zentrale Rolle, wenn Serviceroboter nahtlos in das Umfeld von Menschen integriert werden sollen. Für den erfolgreichen Einsatz ist nicht nur die verwendete Technik verantwortlich, sondern auch die Interaktionsmöglichkeit, Verhaltensweise und die Akzeptanz der Roboter. Daher soll in dieser Masterarbeit die Mensch-Roboter-Interaktion anhand der Aspekte Technik, Interaktionsmöglichkeit, Verhaltensweise und Akzeptanz, betrachtet werden.

1.2. Zielsetzung dieser Arbeit

Das Ziel der Masterarbeit ist es, zu prüfen, wie gut eine Mensch-Mensch-Interaktion, anhand eines gestenbasierten Zweipersonenspiel (Schere, Stein, Papier), auf einen technisch mittelmäßig ausgestatteten, humanoiden Roboter (NAO) übertragbar ist. Dazu soll eine Gestenerkennung entwickelt werden, die ohne Hilfsmittel, wie zum Beispiel einem Farbhandschuh, auskommt. Weiterhin sind Spielstrategien zu implementieren, die unterschiedliche Verhaltensmuster repräsentieren. Anschließend sollen die Interaktionen mit NAO und die verschiedenen Spielstrategien, sowie deren Einfluss auf die Akzeptanz des Roboters, unter Zuhilfenahme einer Nutzerstudie, bewertet werden.

1.3. Struktur der Master-Thesis

Die Struktur der vorliegenden Arbeit ist folgendermaßen gestaltet: Kapitel 2 handelt über die Mensch-Roboter-Interaktion. Als Erstes wird in 2.1 der Begriff Serviceroboter definiert. Anschließend werden in Unterkapitel 2.2 verschiedene Robotertypen vorgestellt und mögliche Einsatzgebiete aufgezeigt. 2.3 geht auf die Problematik der Akzeptanz ein und 2.4 widmet sich der Interaktion. Im Abschnitt 2.4.1 geht es um den Trend zur innovativen Steuerung von Computern. 2.4.2 beschäftigt sich mit speziellen Aspekten der Mensch-Roboter-Interaktion, und den Unterschieden zur Mensch-Computer-Interaktion. 2.4.3 handelt über die Gestenerkennung, welche eine Grundlage dieser Arbeit darstellt. Anschließend wird in Unterkapitel 2.5 über Spiele gesprochen, und

2.5.1 begründet die Tatsache, dass Spiele als Interaktion angesehen werden können. Darauf hin wird das Spielprinzip von "Schere, Stein, Papier" in 2.5.2 vorgestellt. Abschnitt 2.5.3 hingegen zeigt auf, was der Begriff Spiel in der Spieltheorie bedeutet, und in 2.5.4, wie die Begriffe Spiel und Spielstrategie im Kontext dieser Arbeit gesehen werden.

Kapitel 3 handelt über den eigenen Ansatz zur Problemlösung. Dieser teilt sich in zwei Themenbereiche. Der erste Bereich umfasst die Mensch-Roboter-Interaktion mit dem verwendeten Roboter NAO. Dabei wird auf die Randbedingungen für den menschlichen Spieler in 3.1.1 und für den Einsatz der Gestenerkennung in 3.1.2 eingegangen. Das darauf folgende Thema befasst sich mit der Natürlichkeit von NAO. Dazu zählen die verschiedenen Spielstrategien in Abschnitt 3.1.3, als auch, wie NAO spricht. sich bewegt (3.1.4) und wie er die Gesten für das Schere-Stein-Papier-Spiel formt (3.1.5). Der zweite Themenbereich beschäftigt sich mit der, in dieser Arbeit entwickelten, Gestenerkennung (3.2). Dazu wird erst geklärt, welche Schritte dafür durchlaufen werden müssen (3.2.1). Anschließend werden, in den Abschnitten 3.2.2 bis 3.2.9, die einzelnen Schritte der Gestensteuerung näher betrachtet.

Kapitel 4 handelt über die durchgeführte Nutzerstudie (4.1) und legt die Ergebnisse (4.2) dar. Anschließend wird in Kapitel 5 eine Schlussfolgerung gezogen und weitere Verwendungsmöglichkeiten dieser Arbeit diskutiert.

Kapitel 2

Mensch-Roboter-Interaktion

2.1. Definition Serviceroboter

Das Wort Serviceroboter ist bisher schon öfters gefallen, und bevor in die Thematik tiefer eingestiegen wird, soll dieser Begriff noch definiert werden.

Laut International Federation of Robotics (IFR) ist ein Serviceroboter ein Roboter, der nützliche Aufgaben für den Menschen oder dessen Ausrüstung ausführt, wobei die automatisierte Fertigung davon ausgeschlossen ist. Wann ein Roboter als Service- bzw. Industrieroboter klassifiziert wird, ist abhängig von seiner vorgesehenen Anwendung. [2]

2.2. Robotertypen und Einsatzgebiete

Generell können Serviceroboter in zwei Kategorien aufgeteilt werden: für die private Nutzung und für die kommerzielle Nutzung.

Schenkt man den Zahlen auf der IFR-Webseite Glauben, so sind seit 1998 bis heute über 126.000 Serviceroboter im gewerblichen Bereich im Einsatz. Wie viele tatsächlich noch funktionieren, kann jedoch nur schwer nachgewiesen werden. Die durchschnittliche Lebensdauer eines Industrieroboters beträgt ca. 8 Jahre, die eines Unterwasserroboter vergleichsweise 10 Jahre. [3]

Zu den gewerblichen Einsatzgebieten, in denen bereits Serviceroboter eingesetzt werden, zählen das Militär, die Agrarwirtschaft, die Medizin, die professionelle Reinigung sowie Sicherheits- und Rettungsdienste. Im privaten Bereich werden bisher nicht "so viele" Serviceroboter eingesetzt. Hauptsächlich werden hier Putz- und Staubsaugerroboter sowie Mähroboter und Spielzeuge verkauft [1 p. 11].

Dass die Serviceroboter ein großes (wirtschaftliches) Potenzial besitzen, kann auch daran erkannt werden, wie intensiv sich die Forschung mit dieser Thematik beschäftigt. Es werden Roboter entwickelt, die komplexere Aufgaben übernehmen sollen, als zum Beispiel die einfache und spezielle Aufgabe des Staubsaugens. Dazu gehört Rubi the Robot Tutor, Asimo oder Care-o-Bot. Grundlagenforschung wird insbesondere in den Bereichen des maschinellen Lernens (Child Bot 2 [4], iCub) oder der Mensch-Roboter-Interaktion (Flobi, Paro) betrieben. Einige von den genannten Beispielen werden nachfolgend vorgestellt.

Asimo ("Advanced Step in Innovative Mobility") von Honda wurde erstmals am 31. Oktober 2000 der Öffentlichkeit vorgestellt. Der fortschrittlichste Roboter der Welt ist 130 cm groß, wiegt 54 Kilo und hat 57 Freiheitsgrade. Er besitzt die Fähigkeit mit seinen zwei Beinen zu gehen, zu rennen, zu hüpfen und sogar Treppen zu steigen. Zusätzlich kann er Personen und Gesten erkennen. Er versteht sprachliche Kommandos und kann auf Geräusche in der Nähe reagieren. Sein erster, richtiger Einsatz außerhalb von Bühnenpräsentationen überforderte ihn aber schon. Er sollte als Museumsführer in Tokio den Besuchern Fragen beantworten. Immer, wenn ein Besucher eine Frage stellen möchte, soll er dazu die Hand heben. Diese waren jedoch mehr damit beschäftigt, den Roboter zu fotografieren, als ihm Fragen zu stellen. Asimo verwechselte das Fotografieren mit der Geste für eine Frage. Demnach fing sich der Roboter in einer Schleife und fragte ständig: "Wer will Asimo eine Frage stellen?". Die Software von Asimo ist nach dem Top-Down-Ansatz aufgebaut. Sämtliche Tätigkeiten des Roboters müssen von den Entwicklern vorab einprogrammiert werden. Daraus folgt, dass Asimo keine Möglichkeit hat, etwas dazuzulernen. Also auch nicht, ob die Erkennung einer Geste richtig oder falsch war.

Den gegenteiligen und moderneren Ansatz, den Bottom-up-Ansatz, verwendet der Roboter CB2 (Child Bot 2). CB2 wurde in Osaka entwickelt, ist 130 cm groß, wiegt 32 Kilo und verfügt über die Intelligenz und Bewegungen eines Zweijährigen. Er strampelt mit den Füßen, gibt nicht verständliche Laute von sich, und erfasst die Umgebung mit seinen beweglichen Augen. CB2 besitzt eine Silikonhaut, unter der sich viele Sensoren und Prozessoren befinden, die allerlei Daten verarbeiten. So kann er eine Berührung bemerken und diese "Empfindung" abspeichern. Inzwischen hat der Roboter schon gelernt, Emotionen von Menschen zu erkennen und in Kategorien wie "traurig" oder

"fröhlich" einzuordnen, sowie unter Hilfestellung zu laufen. Es wird geschätzt, dass es ihm in ungefähr zwei Jahren sogar möglich sein soll, einfache Sätze zu sprechen.

Flobi hingegen ist nur ein Roboterkopf, der von der Universität Bielefeld entwickelt wurde. Der Kopf eines Roboters ist nach Ingo Lütkebohle et al. die "prominenteste" Schnittstelle zur Interaktion zwischen Mensch und Roboter [5]. Flobi dient also hauptsächlich dazu, die Mensch-Roboter-Interaktion zu erforschen. Dazu kann Flobi sechs grundlegende menschliche Emotionen (neutral, fröhlich, traurig, ängstlich, verärgert und überrascht) zeigen. Besonders großen Wert wurde darauf gelegt, dass der Kopf nicht zu menschlich, aber auch nicht zu technisch wirkt. Damit soll das sogenannte "Uncanny Valley" (siehe Abschnitt 2.3) umgangen werden. Weiterhin wurde darauf geachtet, dass keine Löcher in der Maske vorhanden sind und die Technik sichtbar ist. Das soll verhindern, dass Abweichungen von einem normalen Erscheinungsbild ein unwohles Gefühl herbeiführen [5 p. 3].

Mit dem heutigen Stand der Forschung sind wir aber noch lange nicht da angekommen, wo wir gerne hin möchten: Nämlich zu voll funktionierenden (humanoiden) Servicerobotern, wie sie in fast jedem Science-Fiction-Film vorkommen. Das Forschungsgebiet ist sehr umfassend, denn nicht nur die technischen Probleme wollen gelöst, sondern auch psychologische Fragen müssen geklärt werden. Die Akzeptanz von Robotern ist eine davon.

2.3. Akzeptanz von Servicerobotern

Wenn Menschen von einem humanoiden Roboter angesprochen werden, zum Beispiel auf einer Technologiemesse, so können ganz unterschiedliche Reaktionen beobachten werden. Kleine Kinder fangen vielleicht an zu weinen, ältere Personen halten sich zurück und wissen nicht genau, wie sie sich verhalten sollen. Andere sind voller Neugierde und versuchen mit dem Roboter zu interagieren. An den unterschiedlichen Reaktionen kann erkannt werden, dass noch Skepsis gegenüber Servicerobotern besteht, ganz im Gegensatz zu Computern. Solange diese Skepsis noch vorhanden ist, werden die Menschen Serviceroboter auch nicht so einfach akzeptieren. Die meisten Menschen sind gegenüber neuen Dingen erst einmal misstrauisch eingestellt, da sie diese noch nicht gut einschätzen können. Erst nach einer gewissen Zeit entscheiden sie sich dafür oder

dagegen. Individuen tendieren dazu, Dinge anhand von gewissen Merkmalen zu vergleichen und zu messen. Dies ist bei Servicerobotern genau so. Ein Merkmal, das zur Bewertung meistens hergenommen wird, ist die Nutzbarkeit.

In der Studie des VDE "Senioren pro Serviceroboter" steht die Mehrheit der Befragten (Techniker, Senioren, Pflegekräfte) Servicerobotern zwar positiv gegenüber. Aber die Meinung schwankt besonders bei den Senioren. So lehnen 40 % die Servicerobotik im Alltag spontan ab. Der Grund scheint hier der Zweifel an der Nützlichkeit und Bedienbarkeit zu sein. 60 % empfinden Serviceroboter sogar als "unheimlich". Generell würde die Mehrheit der Befragten jedoch einen Roboter im Haushalt anstelle des Altersheims bevorzugen. D. h. der Wunsch möglichst lange selbstständig bleiben zu können, fördert die Akzeptanz von Robotersystemen im eigenen Heim. Demnach ist der Nutzen des Roboters, wie Mobilität, Orientierung, Unabhängigkeit, Selbstständigkeit und Schutz der Intimsphäre, ein starker Faktor. [6]

Ist der Nutzen also für die eigene Person ausreichend hoch, so sinkt das Misstrauen und die Akzeptanz steigt. Denn der eigene Vorteil aus einer Sache nimmt einen hohen Stellenwert ein und verringert die akzeptanzhemmenden Argumente.

Neben dem Nutzen ist auch das Aussehen des Roboters ein weiteres Merkmal, das einen Einfluss auf die Akzeptanz ausübt. Wie bereits gesagt, neigen Menschen dazu, Dinge mit Anderen zu vergleichen. Als Beispiel sei eine körperlich eingeschränkte Person genannt. Instinktiv vergleichen wir diese Person mit einem Bild des Menschen, welches als "normal" bzw. "gesund" gilt. Es wird sofort bemerkt, dass etwas nicht stimmt. Dadurch entsteht dann, insbesondere wenn man nicht oft in Kontakt mit behinderten Menschen kommt, ein "befremdliches" Gefühl. Dasselbe gilt für einen Serviceroboter, der für sich beansprucht, humanoid zu wirken. Er wird automatisch an einer Person gemessen. Sobald Unstimmigkeiten auftreten, sei es in der Sprache, im Aussehen, in der Bewegung oder im Verhalten, wird das sofort bemerkt. Denn die Erwartungen, die wir an den Roboter stellen, befinden sich auf dem Niveau eines Menschen. Diese werden aber von ihm nicht erfüllt und das wirkt sich umgehend negativ auf die Akzeptanz aus. Im Zusammenhang mit dem Aussehen eines (humanoiden) Roboters wird daher oft das sogenannte Uncanny Valley genannt.

Das Uncanny Valley, zu Deutsch "unheimliches Tal", ist ein Begriff, der von dem japanischen Robotikforscher Masahiro Mori geprägt wurde. Er stellte fest, dass je menschlicher ein Roboter wahrgenommen wird, umso vertrauter wirkt er auf uns.

Allerdings nur bis zu einem gewissen Punkt. Dann sinkt die Vertrautheit und wir empfinden den Roboter eher als "unheimlich". Erst mit anschließender Zunahme der Menschlichkeit steigt auch die Vertrautheit wieder. Am höchsten ist die Vertrautheit, wenn der Roboter nicht mehr von einem echten Menschen zu unterscheiden ist. Mori erklärt dieses Phänomen anhand einer Handprothese. Viele Prothesen sind auf den ersten Blick nicht von einer echten Hand zu unterscheiden. Bei einer Berührung wird aber sofort bemerkt, dass sie nicht echt ist. Es kommt ein Gefühl des "Unbehagens" bzw. der "Befremdlichkeit" auf. Das ist dass unheimliche Tal. Mori schlägt den Roboterentwicklern als Lösung vor, einen nicht zu menschenähnlichen Roboter zu bauen. Das würde das Abrutschen in das Uncanny Valley verhindern. Als ein gutes Beispiel, solle man sich an Brillen orientieren. Sie stellen zwar keine echten Augen dar, aber sie sind passend und verleihen den Augen einen gewissen Charme. [7]

Das bedeutet, dass eine humanoide Gestalt eines Roboters zwar die Akzeptanz erhöht, aber auch, dass es ins Negative abfallen kann. Es ist also wichtig, bei der Entwicklung von Robotern darauf zu achten, ein gutes Gleichgewicht zu erreichen.

Die vorgeschlagene Lösung von Mori kann sehr gut an aktuellen, computeranimierten Filmen verdeutlicht werden. Prinzipiell gilt die Theorie des Uncanny Valley nämlich nicht nur für Roboter, sondern für alle künstlich erschaffene Figuren. Wer schon einmal den Kinofilm "Final Fantasy - Die Mächte in dir" angesehen hat, wird das "Uncanny Phänomen" vermutlich selbst erlebt haben. Die auf realistisch gestalteten, computeranimierten Charaktere wirken alles andere als menschlich. Damit die Zuschauer mit den Figuren jedoch sympathisieren können, werden in vielen neueren computeranimierten Filmen (insbesondere Kinderfilme) selten möglichst realistische Menschen verwendet. Ein humanoider Körper ist zwar meistens vorhanden, aber comichafte Eigenschaften werden standardmäßig hinzugefügt. Das fängt bei überproportional großen Augen an und hört bei katzenhaften Ohren auf. Dadurch ist der Charakter humanoid genug, sodass die Zuschauer eine gewisse Vertrautheit empfinden, aber auch weit genug davon entfernt, um nicht in das Uncanny Valley zu fallen.

Aber nicht nur der Nutzen oder das Aussehen wirkt sich auf die Akzeptanz eines Roboters aus, sondern auch sein Verhalten.

Menschen haben unterschiedliche Ziele oder Erwartungen, die abhängig von der Situation sind, in der sie sich gerade befinden. In einer Gesprächssituation könnte das Ziel sein, möglichst interessante Informationen auszutauschen oder das Gespräch am Laufen zu halten. Während des Spielens sind die Ziele, spaß zu haben und zu gewinnen.

Alle Verhaltensmuster, die nicht diesem Ziel entsprechen, werden negativ aufgefasst und umgekehrt. Während des Schreibens der Masterarbeit ist es ein Ziel, ungestört arbeiten zu können. Ist dabei eine Person im Raum, die Lärm verbreitet, so wird dieses Verhalten als störend empfunden, und somit negativ bewertet. Ist die Person jedoch ruhig, so ist es dem gesetzten Ziel dienlich, und wird positiv bewertet. Menschen sind außerdem Individualisten, und das Bewerten von Verhaltensmustern ist von vielen Faktoren abhängig, wie zum Beispiel auch von der Erziehung, von der eigenen Einstellung, Laune oder des Wohlbefindens. In einer ernsten Situation wird meistens kein Verhalten toleriert, welches der Situation nicht entspricht. So ist ein fröhliches, spaßiges oder "gute Laune"-Verhalten auf einer Beerdigung einfach unangemessen.

Das Verhalten hat somit einen direkten Einfluss (über die Bewertung) auf die Akzeptanz. Daher muss auch ein Serviceroboter, der im menschlichen Umfeld agiert, und von den Menschen akzeptiert werden soll, ein an die Situation angepasstes Verhalten besitzen.

2.4. Interaktion

Der Begriff (soziale) Interaktion [1] wird laut Duden als eine "Wechselbeziehung zwischen Personen u. Gruppen" [8] definiert. Das heißt, es muss eine Handlung zwischen zwei Akteuren stattfinden, die aufeinander bezogen sind. Dies lässt sich anhand eines einfachen Beispiels erklären:

Während einer Vorlesung hat ein Student eine Frage. Um dies dem Professor mitzuteilen, meldet er sich. Dazu hebt er die Hand und wartet darauf, dass der Professor ihn bemerkt. Der Professor nimmt den auf sich aufmerksam machenden Studenten wahr. Indem er auf ihn deutet und den Satz "Ja, bitte?!" ausspricht, signalisiert der Professor, dass er bereit ist, den Studenten anzuhören. Der Student stellt darauf hin eine Frage. Anschließend antwortet der Professor auf die Frage und der Student gibt mit einem

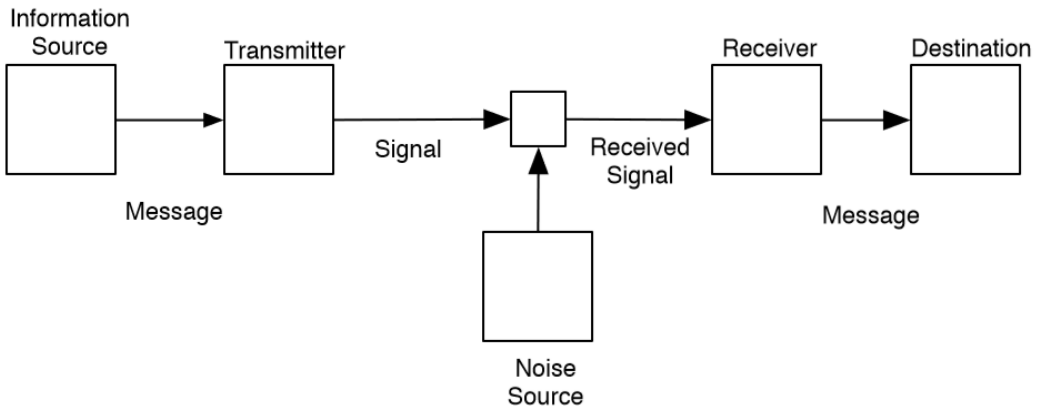
[1] Als Synonym zu Interaktion kann auch der Begriff Kommunikation verwendet werden, was in dieser Arbeit auch so gehandhabt wird.

Kopfnicken und einem "Ach so ist das!" zu erkennen, dass er die Erklärung verstanden und angenommen hat.

Die Handlungen "Frage stellen" und "Frage beantworten" sind aufeinander bezogene Handlungen zwischen den beiden Akteuren Student und Professor. Wird die Wechselbeziehung als Bedingung aufgefasst, um eine Interaktion als solche zu definieren, ist sie mit den oben genannten Handlungen erfüllt.

Das beschriebene Beispiel soll aber nicht nur die Begriffsbestimmung von Interaktion veranschaulichen, sondern auch aufzeigen, dass die zwischenmenschliche Kommunikation verschiedene Kanäle für das Übertragen und Empfangen von Informationen verwendet. Im Allgemeinen können die Kanäle in Empfangs- und Sendekanäle unterteilt werden. Als Sendekanäle zählen die gesprochene Sprache, die Mimik und die Gestik. Empfangskanäle sind dementsprechend das Sehen, das Fühlen und das Hören. Somit werden vom Sender also, per Sprache oder Geste, Informationen an den Kommunikationspartner übertragen, welcher via Hören und Sehen die Informationen empfängt. Anschließend wird diese interpretiert und adäquat darauf reagiert.

Diese Ansicht des Kommunikationsprozesses lässt sich auch leicht abstrahiert in den technischen Bereich übertragen und anhand eines Sende- und Empfängermodells (Darst. 1) darstellen, was bereits 1949 mit dem Shannon-Weaver-Modell [9] getan wurde.



Darstellung 1: Kommunikationsprozess nach Shannon-Weaver. [9]

Die Interaktion mit Computern kann mit diesem Modell ebenfalls dargestellt werden. Der Benutzer (Informationsquelle) überträgt seine Instruktion (Nachricht) mittels eines Sendegeräts (Tastatur, Maus) über den Kommunikationskanal (Kabel) an den Empfänger (Kabeleingang), um anschließend am Ziel (Prozessor) weiterverarbeitet zu

werden. Damit ist auch ein, insbesondere in den letzten Jahren, immer wichtiger gewordenes Thema angesprochen: die Mensch-Computer-Interaktion.

2.4.1. Mensch-Computer-Interaktion

Die Mensch-Computer-Interaktion ist ein Teilgebiet der Informatik und eine Unterkategorie der Mensch-Maschine-Interaktion. Sie beschäftigt sich hauptsächlich mit den Fragen, wie ein System oder eine Software optimal gestaltet und bedient werden kann. Der Fokus liegt dabei auf der Benutzerschnittstelle, also alle Handlungen oder Möglichkeiten, mit denen ein Benutzer mit einem interaktiven System interagieren kann. Betrachtet man nun die Entwicklungen im Bereich HCI (Human-Computer-Interaction) der letzten Jahre, so fällt sofort ein gewisser Trend auf. Dieser führt weg von den bisher etablierten Tastaturen, Mäusen oder Gamepads. Stattdessen legen die Forscher und Entwickler Wert auf eine möglichst natürliche Steuerung. Dazu orientieren sie sich hauptsächlich an den Kommunikationskanälen der Menschen. Sprache und Gestik ist voranging zu nennen, obwohl auch die Forschung an einer Gedankensteuerung weiter voranschreitet. Nachfolgend sollen einige natürliche Benutzerschnittstellen zur Interaktion mit interaktiven Systemen vorgestellt werden.

Gegenstandsbasierte Benutzerschnittstelle (Tangible User Interface)

Das TUI basiert auf reelle Objekte, welche mit virtuellen Informationen verknüpft sind, und diese manipulieren können. Das können runde Scheiben sein, die auf einen Tisch gelegt werden und mittels Drehung die Lautstärke einer Audiodatei regeln oder ein viereckiger Würfel, der die eigentliche Audiodatei repräsentiert und aufruft. Somit dienen die Objekte sowohl der Eingabe als auch der Ausgabe. Denn sie repräsentieren einen jeweiligen Zustand der Maschine, unabhängig davon, ob sie eingeschaltet ist oder nicht. Ein TUI vereinfacht also ein kompliziert zu steuerndes System, indem bereits erlernte Fähigkeiten zur Objektmanipulation aus der realen Welt angewendet werden können. Das in der Entwicklung befindliche "Sandscape" von Ishii et al. [10] verdeutlicht diesen Ansatz sehr gut. Sandscape ist im Prinzip ein Sandkasten, in den auch Gebäudemodelle und andere Objekte physikalisch eingesetzt werden können und der mittels Laser ausgemessen wird. Anschließend generiert ein Computer in Echtzeit eine 3D-Umgebung der Landschaft und berechnet dazu Schatten, Höhen oder

Wasserverläufe. Diese werden wiederum über installierte Lichter auf den Sand gestrahlt. Folglich ermöglicht Sandscape Landschaftsdesignern eine intuitive und kooperative Möglichkeit Projekte zu planen oder zu besprechen.

Gehirn-Computer-Schnittstelle (Brain Computer Interface)

Eine Gehirn-Computer-Schnittstelle ermöglicht prinzipiell die Steuerung eines Computers oder eines Roboters mittels den Gedanken. Dazu wird (aktuell noch) eine Elektrode in das Gehirn eingesetzt, welche die Muster der elektrischen Nervensignale erkennt und zum Beispiel in gedachte Bewegungen übersetzt.

Gestenbasierte Benutzerschnittstelle (Gestural Interface)

Das Gestural-Interface ist ein Ansatz zur natürlichen Steuerung von technischen Geräten. Dazu werden Gesten von Menschen erkannt und interpretiert. Anschließend lassen sich daraus Befehle ableiten. Ein gutes Beispiel hierfür ist Kinect von Microsoft.

Erschienen ist die Kinect Technologie 2010 für die Xbox360, inzwischen gibt es die Kamera auch für den PC. Kinect besteht aus einer Leiste, in der verschiedene Sensoren und Kameratypen eingebaut sind. So besitzt sie eine normale RGB-Kamera, einen Infrarot-Projektor, eine Infrarotkamera und mehrere Mikrofone. Kinect erkennt die Position und die Gesten des Benutzers, indem der gesamte Raum mittels des Infrarot-Projektors ausgeleuchtet wird. Dazu wird ein fest definiertes Punkteraster in den Raum projiziert, anhand dessen die Entfernung von Objekten berechnet werden kann. Dazu empfängt die Infrarotkamera das reflektierte Licht und liefert die Daten an einen Mikroprozessor. Dieser berechnet anhand des empfangenen Punktmusters und des Referenzmusters ein Tiefenbild. Das Verfahren kommt auch bei so genannten Time-of-Flight Kameras zum Einsatz. Anschließend wird ein Skelettmodell berechnet, bei dem im ersten Schritt versucht wird einzelne Körperteile (Schulter, Unterarm, Hand etc.) zu identifizieren. Hierfür wird eine große Trainingsdatenbank mit unterschiedlichen Posen verwendet. Aufgrund der Körperteile können nun Rückschlüsse auf einzelne Gelenke getroffen werden. Diese werden dann mit Kanten verbunden und ein Skelett daraus gebildet, was wiederum eine Klassifizierung der Posen anhand der Gelenkstellung bzw. des Skeletts zulässt.

Die Gestenerkennung ermöglicht somit eine berührungslose Bedienung von technischen Geräten. Das ist zum Beispiel in der Medizin sehr vorteilhaft, da steril gearbeitet werden muss. Wenn für die Steuerung von Geräten keine Berührung notwendig ist, werden dadurch auch weniger Bakterien verbreitet. Ein anderes Einsatzgebiet könnten Reklamemonitore in Schaufenster sein, bei denen der Benutzer wie in einem Katalog blättert.

Sprachbasierte Benutzerschnittstelle (Voice User Interface)

Die Sprachsteuerung ist ebenso, wie die gestenbasierte Benutzerschnittstelle, eine natürliche Art und Weise ein System zu steuern. Dazu spricht der Mensch zu einer Maschine und erteilt ihr Befehle. Die Instruktionen können sowohl einzeln als auch in einem kompletten Satz ausgesprochen werden.

Als Beispiel soll hier die von Apple ent-wickelte Sprachsteuerung Siri dienen, die erstmals 2011 vorgestellt wurde. Der Benutzer kann einem IPhone sprachlich Befehle erteilen oder Fragen stellen. Es ist zum Beispiel möglich, nach dem Wetter in München zu fragen und prompt liefert Siri die Antwort. Der große Vorteil der Sprachsteuerung liegt darin, dass auch komplexere Aufgaben durchgeführt werden können, ohne dabei das Smartphone mit der Hand zu bedienen. Das macht besonders in Situationen Sinn, in denen es nicht möglich ist die Aufmerksamkeit auf das Handy zu richten, wie beispielsweise beim Autofahren. Oder auch für blinde Personen, die die modernen Touchscreen - Smartphones nur schlecht bedienen können.

Für die Implementation einer Spracherkennung gibt es mehrere Arten. Ist die Sprachsteuerung bzw. Spracherkennung als Software mit eigenem Wortschatz lokal installiert, dann wird meistens der komplette Arbeitsprozess grammatikbasierend auf dem Endgerät verarbeitet. Dazu wird in dem mitgelieferten Wortschatz nach einzelnen Befehlen gesucht, die mit der gesprochenen Version übereinstimmt. Dies hat den Vorteil, dass die Software nur wenig Ressourcen benötigt, um zu funktionieren. Werden jedoch spezielle Wörter oder Formulierungen verwendet, ist eine hohe Fehlerquote der Erkennung sicher. Bei einer serverbasierten Lösung, so wie es bei Siri der Fall ist, werden die vom Benutzer gesagten Wörter oder Sätze aufgenommen und an einen Server von Apple geschickt und dort verarbeitet. Das hat den Vorteil, dass wesentlich mehr Schlüsselwörter für die Erkennung zur Verfügung stehen. Ein Update der Algorithmen ist auch durchführbar, ohne dass der Benutzer die Software aktualisieren muss. Der Nachteil ist jedoch auch die Grundvoraussetzung für den Dienst: eine aktive Internetverbindung. Das Übermitteln der Daten könnte, je nach Datentarif, etwas kosten. Dazu kommt, dass sämtliche

Sprachbefehle oder Sätze bei Apple verarbeitet werden und man somit hinsichtlich des Datenschutzes auf das Unternehmen vertrauen muss.

Leider ist die Sprachsteuerung noch nicht ausgereift genug, um wirklich produktiv für eine Vielzahl von Systemen genutzt zu werden. Zusätzliche Stimmen oder Hintergrundgeräusche können die Erkennung erschweren. Eine einheitliche Erkennung von Wörtern ist ebenfalls nicht zu realisieren, da jeder Mensch eine etwas andere Aussprache besitzt. Für kleinere und leichte Aufgaben, wie Befehle geben, ist die Sprachsteuerung nach dem aktuellen Stand der Technik jedoch ganz gut geeignet.

Für die Zukunft ist es sicherlich der richtige Weg, die Steuerung von interaktiven Systemen so natürlich wie möglich zu gestalten. Denn die Technologisierung des menschlichen Lebensraums schreitet von Jahr zu Jahr weiter voran. Dabei werden die Systeme auch zunehmend mit mehr Funktionalität ausgestattet, was die Bedienung komplexer werden lässt. Viele Leute sind zum Beispiel nach dem Kauf eines neuen Autos erst einmal mit den ganzen Funktionen überfordert. Es gilt nicht nur den Radio oder die Klimaanlage zu steuern, sondern auch das eingebaute Navigationsgerät, den Einparkassistenten, die elektrischen Fensterheber oder die programmierbaren Sitzeinstellungen. Viele Einstellungen sind zentral über den Bordcomputer vorzunehmen. Dafür darf der Benutzer dann erst einmal ein dickes Handbuch durchlesen, ist im Nachhinein aber trotzdem nicht schlauer als vorher, welcher Knopf für welche Funktion gedrückt werden muss. Während der Autofahrt muss für Einstellungen die Aufmerksamkeit auf das Display des Bordsystems gerichtet werden, was die Sicherheit gefährdet. Und zu guter Letzt vergisst ein Benutzer sehr schnell, wie etwas bedient wird, wenn er es nicht ständig wiederholt. Deshalb bauen viele Autohersteller inzwischen eine Sprachsteuerung ein, was den oben genannten Trend zu den natürlichen Benutzerschnittstellen bestätigt.

2.4.2. Spezielle Aspekte in der Mensch-Roboter-Interaktion

Die Mensch-Roboter-Interaktion (MRI) kann mit der Mensch-Compter-Interaktion (MCI) durchaus verglichen werden. Beide Gebiete beschäftigen sich mit dem Verstehen, Konzipieren und Evaluieren von Mensch-Maschine Interaktionen. Die MRI unterscheidet sich aber in einigen wesentlichen Punkten von der MCI.

In erster Linie sind Roboter physische Maschinen, die in der realen Welt situiert sind, und in dieser agieren können. Hierdurch ist ein kooperatives Handeln mit dem

Menschen ein gewünschtes und realistisches Szenario, und ein elementarer Unterschied zur Mensch-Computer-Interaktion. Auf diesem Gebiet ist noch einiges an Forschung nötig, wird aber seit ein paar Jahren immer intensiver betrieben. Es müssen dabei nicht nur die technischen Fragestellungen geklärt werden, sondern auch soziologische, wie die Rollenverteilung während einer Mensch-Roboter-Kooperation und welchen Einfluss die physische Form des Roboters auf die Zusammenarbeit ausübt.

Arbeiten zwei oder mehrere Menschen zusammen, dann werden normalerweise verschiedene Rollen eingenommen. Beim Aufbau von Möbeln zum Beispiel übernimmt eine Person die Führungsrolle und gibt Instruktionen, wie einzelne Teile verbunden werden müssen. Die restlichen Mitwirkenden stellen die entsprechenden Komponenten bereit und setzen sie zusammen. Daraus ergeben sich instruktive und unterstützende Rollen.

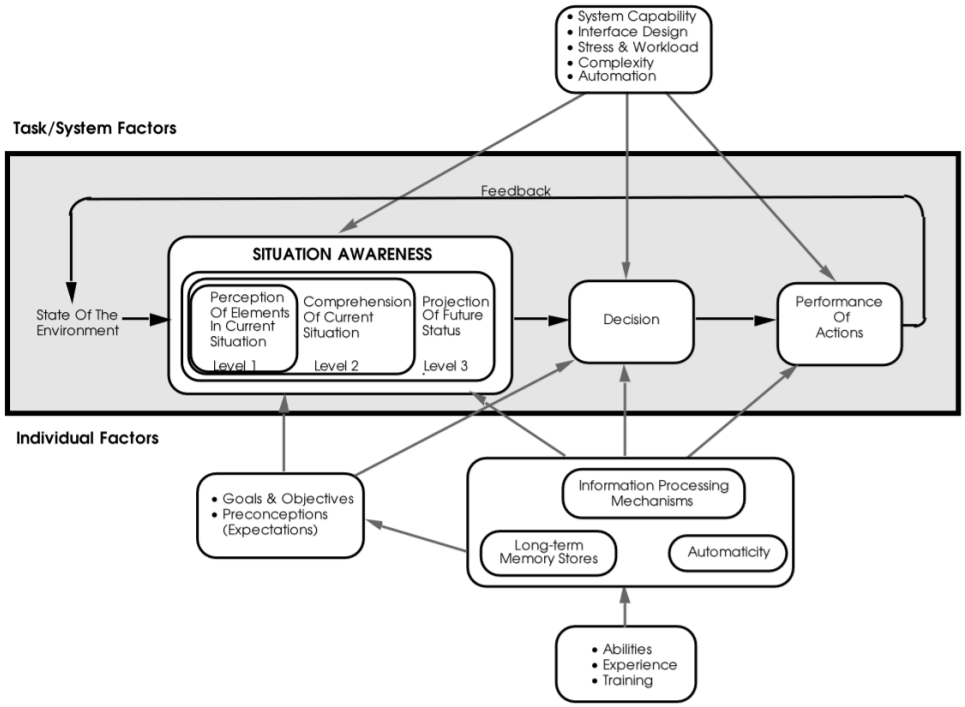
In der Studie von Giuliani und Knoll [11] wurde dazu bereits untersucht, wie Menschen auf die vom Roboter eingenommenen Rollen reagieren. Sie kamen zu dem Ergebnis, dass keine Rolle bevorzugt wird. Stattdessen nahmen die Testpersonen immer den Gegenpart ein und passten ihr Verhalten an das des Roboters an.

In der Studie Reliance on Robots [12] wird versucht zu klären, welchen Einfluss die physische Gestalt des Roboters und dessen (sozialer) Status auf die MRI ausübt. Dazu wurde untersucht, in welchem Ausmaß die Menschen ihre Verantwortlichkeit an den Roboter abgeben und sich auf ihn verlassen. Das Resultat war, dass die Teilnehmer eine intensivere Zusammenarbeit mit einer menschlichen Person aufwiesen und sich mehr auf diese verließen, sowie deren Rat weniger ignorierten, als gegenüber einem Roboter. Die Verantwortlichkeit wurde bei einem menschlichen Kollegen ebenfalls weniger übertragen. Einen weiteren interessanten Schluss zog die Studie in der Hinsicht, dass das Verantwortlichkeitsgefühl des Menschen am stärksten ausgeprägt ist, wenn der Roboter eher maschinell aussieht und den Status eines Untergebenen besitzt. Die physische Existenz ermöglicht jedoch nicht nur die kooperative Zusammenarbeit, sondern beeinflusst auch, wie Menschen Roboter wahrnehmen. So wurde in der Studie Effect of a Robot on User Perceptions [13] festgestellt, dass ein Roboter anziehender, glaubwürdiger und informativer wahrgenommen wird, als ein virtueller Agent. Auch die Interaktion mit dem Roboter wurde als angenehmer empfunden. Dies ist nützlich, um das Vertrauen zu den Maschinen besser aufbauen zu können.

Der nächste wichtige Unterschied ist, dass Roboter sich ihrer Benutzer bewusst sein müssen. Ein Computer, Smartphone oder andere technische Geräte erhalten durch ihre Nutzer Befehle, und werden so gesteuert. Dabei bemerken die Geräte den User jedoch nicht, was auch gar nicht notwendig ist. Es reicht vollkommen aus, wenn sie die erhaltenen Instruktionen ausführen. Für eine Mensch-Roboter-Interaktion muss sich der Roboter aber seiner Benutzer bewusst sein. Dies fängt bei einem einfachen Dialog an. Der Roboter muss erkennen, dass ein Mensch mit ihm spricht. Besonders, wenn sich mehrere Personen in einem Raum befinden, und eine Unterhaltung führen, ist es wichtig für den Roboter zu wissen, wem er seine Aufmerksamkeit zuwenden muss. Es könnten auch mehrere Leute mit dem Roboter reden, dann muss er ebenfalls unterscheiden können, welche Stimme zu welcher Person gehört. Anschließend muss er sein Verhalten dem Gegenüber anpassen. Bei Kindern beispielsweise sollte ein anderes sprachliches Niveau verwendet werden, als bei erwachsenen Personen.

Die Sicherheit ist ein Themenfeld, dass das Bewusstsein über den Benutzer genauso erfordert. Um die Privatsphäre einzelner Personen zu schützen, darf der Roboter vertrauliche Informationen nicht an andere Nutzer weitergeben. Es kann auch unterschieden werden, wer zum Beispiel über administrative Rechte verfügt und dem Roboter Befehle erteilen darf. Es sollte für den Roboter also erkennbar sein, welche Person mit ihm interagiert.

Der dritte spezielle Aspekt der MRI ist, dass Roboter in einer dynamischen Umwelt situiert sind und damit zurecht kommen müssen. Ein mechanischer Museumsführer, der die Besucher herumführt, kann mittels einer Markierung am Boden durch das Museum navigieren. Wäre das Gebäude leer, so würden bei der Navigierung keinerlei Probleme auftreten. Da sich die Umwelt aber dynamisch ändert, indem Besucher die Bahn des Roboters kreuzen oder eventuell Gegenstände auf der Linie stehen, muss der Roboter sich anpassen können und eine andere Route berechnen. Daraus folgt, dass die Mensch-Roboter-Interaktion sich mit der Thematik beschäftigen muss, wie ein Roboter am Besten auf Änderungen seiner Umwelt reagiert. Eine Lösung wäre, die Fähigkeit des Situationsbewusstseins auf Roboter zu übertragen. Grob gesagt, beschreibt dieser Begriff das Wissen, was um einen herum passiert. Endsley [14] unterscheidet in seinem Modell (Darst. 2) drei Level von Situationsbewusstsein.



Darstellung 2: Modell des Situationsbewusstseins nach Endsley. [14]

Das erste Level ist fundamental und beinhaltet das Wahrnehmen von Objekten. Das zweite Level ist das Verstehen der Bedeutung dieser Objekte, und die Entscheidung, ob die Informationen relevant für das Erreichen der eigenen Ziele sind. Das höchste Level des Situtationsbewusstsein ist die Fähigkeit, aus der aktuellen Situation heraus auf zukünftige Geschehnisse zu schließen, um dann rechtzeitig eine Entscheidung treffen zu können. [14 pp. 3–4]

So gesehen kann das Situationsbewusstsein also als ein internes Zustandsmodell der Umwelt verstanden werden. Aufgrund des Zustandsmodells kann dann der Roboter Entscheidungen treffen, was er in der jeweiligen Situation tun soll. Daher wird in Endsley's Modell das Situationsbewusstsein, als eigenständiger Prozess vor der eigentlichen Entscheidungsfindung dargestellt.

In dem Museumsbeispiel muss der Roboter zu erst also Personen und Objekte erkennen und wahrnehmen. Anschließend muss es ihm klar sein, dass die Personen seine Bahn entweder kreuzen oder blockieren könnten. Ableitend aus der Bewegungsrichtung und Geschwindigkeit muss er dann vorhersagen, ob und wann eine Person seinen Weg versperrt bzw. kreuzt. Danach sollte sich der Roboter entscheiden, ob er sich langsamer beweget und somit der Person den Vortritt gibt, oder ob er ihr ausweicht.

Die drei beispielhaften genannten Themenfelder sind die, die die Mensch-Roboter-Interaktion von der Mensch-Computer-Interaktion bzw. Mensch-Maschine-Interaktion abgrenzen. Andere (aber nicht MRI spezifische) Forschungsfelder und Herausforderungen betreffen Autonomität, Lokalisierung, Sprachverstehen und das Computersehen. Natürlich spielt auch die Frage, welche Interaktionsmöglichkeit die Geeignetste ist, eine wichtige Rolle. Hier haben sich jedoch die Sprache, die Gestik und der Touchscreen etabliert. In erster Linie ist die Art und die Häufigkeit der Interaktionen von dem Einsatzgebiet des Roboters abhängig. Ein Staubsaugerroboter, der autonom seine einzige Aufgabe ausführt, benötigt verständlicherweise keine ausgeprägte Kommunikationsmöglichkeit. Ein einfacher Ein / Aus Knopf wäre theoretisch ausreichend. Je mehr der Serviceroboter aber mit Menschen zusammenarbeitet und komplexere, sowie vielfältigere Aufgaben übernimmt, umso notwendiger und wichtiger wird auch eine gute Interaktionsmöglichkeit. Besonders im Hinblick auf die Zielgruppe, also keine Fachkräfte der Robotik oder Informatik, ist der Einsatz von Sprach- und Gestensteuerung als natürliche Kommunikationsform durchaus als sinnvoll zu betrachten. Ein Touchscreen sollte aber nur bei Robotern mit "primitiven" Aufgaben, wie Gegenstände von A nach B bringen, als primäre Steuerung verwendet werden. Sonst ergeben sich gleich mehrere Probleme. Erstens müssen sämtliche komplexen Steuerungsmöglichkeiten mittels einer grafischen Benutzeroberfläche in intuitiver Weise dargestellt werden, was schon bei manchen Computerprogrammen eine schwere Aufgabe ist. Man denke hier nur daran, dem Roboter per Touchscreen neue Aktionen beizubringen. Zweitens ist diese Form der Steuerung für blinde Menschen kaum bedienbar. Und drittens kann in Aktionen des Roboters nicht sofort eingegriffen werden, sondern nur mit einer zeitlichen Latenz. Schließlich muss davon ausgegangen werden, dass der Touchscreen entweder am Roboter direkt befestigt ist, oder die Kommunikation von anderen Geräten aus stattfindet, wie von einem Smartphone. Folglich ergibt sich immer eine Verzögerung, bis der Mensch entweder am Roboter ist oder das Handy aus der Tasche gezogen hat. In einer Situation, in der in die Aktionen des Roboters eingegriffen werden muss, was unvermeidbar sein wird, würde jeder Mensch instinktiv "Halt" oder "Stop" rufen. Daher ist die Sprache nicht nur eine natürliche und intuitive Form der Steuerung, sondern bietet auch zeitliche und dadurch womöglich sicherheitstechnische Vorteile. Der Roboter kann in "Echtzeit" auf die Stimme reagieren. Allerdings darf auch nicht die Gestik vernachlässigt werden. Sie unterstützt nicht nur das Gesprochene, sondern dient z. B. auch für

stumme Menschen als vordringliches Kommunikationsmittel. Daher soll die Gestik im nächsten Kapitel näher betrachtet werden.

2.4.3. Gestik

Die Gestik ist eine Form der Körpersprache und somit eine Unterform der nonverbalen Kommunikation. Die verbale Kommunikation drückt das "Was" gemeint ist aus. Die nonverbale Kommunikation hingegen das "Wie" etwas gemeint ist. Dabei dienen Gesten der Informationsübertragung und der Regulation des Gesprächsverlaufs. Sie können unbewusst und bewusst ausgeführt werden. Ekman und Friesen [15] teilen die Gesten in verschiedene Kategorien auf.

Embleme

Gesten, die ein oder zwei Wörter symbolisch ersetzen können, werden als Embleme bezeichnet. Normalerweise werden diese bewusst ausgeführt und häufig eingesetzt, wenn eine verbale Verständigung nicht möglich ist. Das kann sein, wenn die Entfernung zum Empfänger zu groß ist, oder die Umgebungsgeräusche zu laut sind. Die verbale Bedeutung des Emblems ist meistens allen Personen einer Gruppe oder Kultur bekannt. So ist das typische "Okay" Zeichen, mit einem gestreckten, nach oben zeigenden Daumen und sonst geschlossener Hand, ein Emblem, das, zum Beispiel in den USA, überall bekannt ist. In Japan hingegen, wird die gleiche verbale Bedeutung mit einem aus Daumen und Indexfinger geformten Kreis repräsentiert.

Illustratoren

Illustrative Gesten veranschaulichen das Gesagte und sind damit direkt an die gesprochene Sprache gebunden. Demgemäß ist eine Zeigegeste (Deixis) mit Wörtern, wie "hier", "dort", "da" oder "ich" und "du" verknüpft, und verweist entweder auf tatsächlich vorhandene oder imaginäre Objekte, Orte oder Personen.

Affektdarstellung

Affektäußerungen werden am häufigsten mit Bewegungen der Gesichtsmuskeln erzeugt. Sie zeigen Emotionen, wie Traurigkeit, Verärgerung, Fröhlichkeit, Überraschtheit oder Angst an.

Regulatoren

Gesten, die die Interaktion mit einer anderen Person regulieren, fallen in die Kategorie der Regulatoren. Sie kommunizieren nonverbal, dass der Gegenüber mit dem Gesprochenen fortfahren darf, sich beeilen, etwas ausführlicher beschreiben oder wiederholen soll. Ein typisches Beispiel ist das Kopfnicken. Es ist die symbolische Darstellung des verbalen Ausdrucks "mm-hmm" und bedeutet, dass der Zuhörer das Gesagte verstanden hat, bzw. immer noch dem Sprecher seine Aufmerksamkeit zuwendet.

Adaptoren

Adaptoren sind Bewegungen, die unbewusst durchgeführt werden, und dabei keine Informationen übermitteln. Meistens werden sie aus der Gewohnheit heraus erzeugt. Das Lippenlecken, wenn man nervös ist, ist eine Geste, die in diese Kategorie fällt. Aber auch ein sich an der Nase kratzen oder Veränderungen der Körperhaltung.

Nehaniv [16] stellt fest, dass ein Roboter die Aktivitäten eines Menschen und dessen Intention, zumindest partiell, verstehen muss, um entsprechend darauf reagieren zu können. Damit aus der Geste eines Menschen dessen Absicht abgeleitet werden kann, ist es hilfreich eine Klassifizierung der beobachteten Geste vorzunehmen. Denn Gesten sind, ohne Betrachtung des Kontexts, in dem sie ausgeführt werden, oftmals doppeldeutig. Die Kategorisierung hilft dabei, diesen Kontext herzustellen und ermöglicht somit eine leichtere Erkennung.

Für die Mensch-Roboter-Interaktion sind Gesten also wichtig, da sie eine Möglichkeit zur Steuerung des Roboters, sowie eine wichtige Informationsquelle für diesen selbst darstellen. Die wichtigsten Gesten fallen in die Kategorien Embleme und Illustratoren. Embleme können für die Steuerung des Roboters verwendet werden.

Eine Halt-Geste lässt die Bewegung des Roboters stoppen, ein Winken kann die Aufmerksamkeit auf sich lenken und per Herbeiwinken lautet der Befehl "Komm zu mir!". Die Illustratoren, vor allem die Zeigegesten, sind wichtige Informationsquellen und untermauern das Gesagte. Damit lässt sich beispielsweise die Bestimmung und Lokalisation von Objekten wesentlich einfacher bewerkstelligen, als den Ort verbal genau zu bestimmen. Der Satz "Die blaue Tasse da oben." verbunden mit dem Deuten auf die Tasse, ist für den Menschen leichter und natürlicher zu formulieren, als "Die blaue Tasse in dem zweiten Schrank von links, auf dem dritten Regal von unten.".

2.5. Spiele

Fußball, UNO, Schach, World of Warcraft, Theater und Klavier haben alle etwas gemeinsam: Sie werden gespielt. Spielen ist eine Tätigkeit, welche sehr früh im Kindesalter (egal ob Mensch oder Tier) begonnen wird und selbst im hohen Alter noch getan wird. Im Kindesalter dient das Spiel vor allem der Entwicklung von kognitiven und motorischen Fähigkeiten.[2]

So erfährt Spielen, in Abhängigkeit zur kindlichen Persönlichkeitsentwicklung, einen Wandel. Begonnen wird mit dem Funktionsspiel, wodurch sensomotorische Koordinationsleistungen gelernt und geübt werden. Das Funktionsspiel geht später in das Konstruktionsspiel über, bei dem lebenswichtige Kompetenzen entwickelt werden, die in komplexen Gesellschaften unbedingt benötigt werden. Dazu zählen das Organisieren, Bewerten und die innere Regulation des Handelns. Nach dem Konstruktionsspiel folgt das Rollenspiel, das zur Erfahrungsbewältigung dient und automatisch die Sozialität des Spiels und die Kommunikation fördert. Als letzte Form wird das Regelspiel genannt, das einen enormen Einfluss auf die Persönlichkeitsentwicklung hat. Spiele, die die kindliche Bewegungsfähigkeit fordern, fördern die gesamte körperliche Elastizität und Flexibilität, während Spiele, die die Denkleistung beanspruchen, die kognitive Kombinationsfähigkeit und die Strukturierung der Handlungsplanung unterstützen. [17]

Das Spielen wird zwar hauptsächlich des Vergnügens wegen ausgeübt, aber auch zur Entspannung, als Zeitvertreib oder als Beruf. Spiele werden in Gemeinschaft oder Einzeln ausgeführt, und lassen sich prinzipiell unter verschiedenen Gesichtspunkten

[2] Der Ansatz der kindlichen Entwicklung (respektive des Lernens) wird auch in der Robotik angewandt, wie bereits in Kapitel 2 mit dem Child Bot 2 vorgestellt.

gliedern. Dabei sollte jedoch von einer zu scharfen Abgrenzung abgesehen werden, denn viele Spiele bestehen aus einer Kombination dieser Gesichtspunkte. So gibt es kooperative, wettkampforientierte, bewegungsorientierte, geräteabhängige oder ortsabhängige Spiele. Fußball kann beispielsweise zu den kompetitiven Spielen zählen, wobei ein kooperatives Zusammenspiel innerhalb der Mannschaft stattfindet, und gleichzeitig eine Interpretation als Laufspiel zulässt.

2.5.1. Interaktion in Spielen

In vielen Gesellschaftsspielen findet (soziale) Interaktion statt. Ein sehr gutes Beispiel sind die in der letzten Dekade immer beliebter gewordenen Massively Multiplayer Online Role-Play Games (MMORPG). In diesen Computerspielen tummeln sich Tausende von Spielern gleichzeitig auf einem Server. Das Ziel eines jeden Spielers ist es vorrangig seinen Charakter (in dessen Rolle er schlüpft, daher Role-Play) auf die maximale Stufe zu heben. Dazu werden von Nichtspielercharakteren (Non Player Character, kurz NPC) Aufträge, sogenannte Quest, vergeben. Die meisten Quest können solo erledigt werden. Anschließend gibt es eine Belohnung in Form von Erfahrungspunkten, Ausrüstung und / oder den dort verwendeten Währungsmitteln. MMORPG sind jedoch auf das gemeinsame Spielen ausgelegt. Das kann sowohl kooperativ, als auch kompetitiv sein. Deshalb kommen die Spieler im Verlauf ihrer Abenteuer immer wieder über Aufgaben und Orte, die alleine nicht zu bewältigen sind und mehrere Spieler benötigen. Diese Gruppenspielinhalte bieten gegenüber den normalen Einzelspieler-Inhalten bessere Belohnungen, die in den meisten Fällen auch notwendig sind, um konkurrenzfähig zu bleiben. Dazu schließen sich die Spieler in Gemeinschaften zusammen, die im kleinen Rahmen Gruppen, und im größeren Rahmen sogenannte Gilden oder Clans sind. Die notwendige Kommunikation zwischen den Spielern findet hauptsächlich über den eingebauten Chat, also textuell, statt. In den letzten paar Jahren hat sich aber auch verstärkt die verbale Kommunikation über Voice Chats durchgesetzt. Sogar die nonverbale Kommunikation ist durch die Spielfiguren möglich, indem diese Gesten ausführen. Die Interaktion zwischen den Spielern besteht vor allem im kooperativen Spielen, um besagte Gruppeninhalte zu lösen, oder im kompetitiven Umfeld, indem sie sich gegenseitig die Köpfe einschlagen. Die materiellen Belohnungen sind entweder zufallsgeneriert, und somit nicht für jeden Charakter brauchbar, oder der Gegenstand ist nur in limitierter Anzahl vorhanden. Folglich sind die Spieler dazu angehalten, regen Handel zu betreiben.

Wie gesehen werden kann, finden in modernen Computerspielen auf vielfältige Art und Weise Kommunikation und Interaktion statt. Aber auch in konservativen Spielen, wie Brettspielen, ist soziale Interaktion vorhanden. Bei Mensch-Ärgere-Dich-Nicht erfolgt oftmals auf einen Wurf der eigenen Spielfigur die Rache, indem bei der nächsten Gelegenheit gezielt die Figur des Widersachers geschmissen wird. Auch in anderen Spielen, wie Schach oder UNO finden wechselseitige Handlungen statt. Selbst wenn dies nicht der Fall sein sollte, so ist das gemeinschaftliche Spielen an sich schon eine Interaktion. Denn "Mitspielen" ist eine auf "Spielen" bezogene Handlung, womit die Bedingung der Wechselseitigkeit erfüllt ist. Daher kann also behauptet werden, dass in allen Spielen, die mindestens zwei Spieler erfordern, zwischenmenschliche Interaktion vorkommt.

2.5.2. Das Spiel Schere, Stein, Papier

"Schere, Stein, Papier" oder auch "Schnick, Schnack, Schnuck", "Ching, Chang Chong", Knobeln oder "Jan, Ken, Pon" ist ein gestenbasiertes Zwei- oder Mehrpersonenspiel. Die Regeln sind einfach und schnell erlernbar. Daher ist es ein optimales Kinderspiel, kann aber auch, ähnlich wie der Münzwurf , zur friedlichen Konfliktlösung beitragen.

Gespielt wird Schere, Stein, Papier (nachfolgend SSP), indem alle Spieler die geschlossene Hand synchron vor sich auf und ab bewegen. Dabei wird entweder bis drei gezählt oder der Name des Spiels ausgesprochen. Bei der Aussprache des letzten Wortes (Drei, Papier, Schnuck, Chong, Pon) formen alle Spieler gleichzeitig eines der drei verwendeten Symbole. Dazu wird mit der Hand die Form einer Schere, eines Steines oder eines Blattes gezeigt. Anschließend wird mit den Regeln ausgewertet, wer gewonnen hat: Schere schlägt (zerschneidet) das Blatt, Blatt schlägt (bedeckt) den Stein und der Stein schlägt (stumpft) die Schere. Falls die Symbole übereinstimmen, gibt es ein Unentschieden, und die Runde wird wiederholt. Eine Spielrunde besteht aus so vielen Spielzügen, bis ein Spieler zwei Siege hat. Ein Spiel besteht aus drei Spielrunden und der Sieger des Spiels ist der Spieler, der zwei Spiele gewonnen hat.

2.5.3. Das Spiel in der Spieltheorie

Die Spieltheorie definiert ein Spiel als mathematisches Modell einer Entscheidungssituation. Für Wessler [18] gehört zu einem Spiel:

1. Zwei oder mehrere Spieler.
2. Eine Ausgangs-Spielposition.
3. Eine Menge von Strategien pro Spieler.
4. Eine Reihenfolge, in der die Spieler die Strategien wählen dürfen.
5. Eine allen bekannte Übersicht von Spielpositionen, resultierend aus der Strategiewahl.
6. Eine Regel für das Spielende.
7. Einen eindeutig messbaren Nutzen für jeden Spieler am Ende des Spiels.

In [18 p. 19] wird anhand von Schere, Stein, Papier aufgezeigt, in welcher Weise diese Regeln erfüllt werden:

1. Es wird mit zwei oder mehreren Spielern gespielt.
2. Die Ausgangssituation ist "leer".
3. Zu jeder Zeit ist die Auswahl zwischen den drei Symbolen (Schere, Stein, Papier) gegeben.
4. Die Strategien (Symbole) müssen gleichzeitig angezeigt werden.
5. Es können aus den drei Symbolen neun unterschiedliche Kombinationen entstehen.
6. Die Spielrunde ist beendet, sobald ein Gewinner / Verlierer feststeht.
7. Am Ende des Spiels ist klar, wer gewonnen und wer verloren hat, was zunächst den "messbaren Nutzen" darstellt.

Es gibt Spiele mit vollständiger Information, bei denen allen Spielern die Spielregeln, Entscheidungsmöglichkeiten und Auszahlungen vollständig bekannt sind. Auf SSP bezogen ist jedem Spieler bekannt, wie das Spiel abläuft, welche Symbole (Entscheidungsmöglichkeiten) es gibt, und welches Symbol gegen welches andere Symbol gewinnt, respektive verliert. Daher ist SSP ein Spiel mit vollständiger Information. Ein Spiel mit perfekter Information wird als solches definiert, wenn zu jedem Zeitpunkt jeder Spieler über das gesamte Spielgeschehen informiert ist. Schach zum Beispiel gehört in diese Kategorie. Denn zu jedem Zeitpunkt ist klar, welche Spielzüge vorher

stattgefunden haben und wie die aktuelle Lage ist. Bei SSP hingegen findet ein gleichzeitiges Zeigen der Symbole statt, womit der Zug des Gegners nicht vor dem eigenen Zug bekannt ist, und folglich ein Spiel mit imperfekter Information vorliegt.

Eine Strategie in der Spieltheorie, ist eine Entscheidung für einen oder mehrere Spielzüge. Das heißt, sie kann als Verhalten des Spielers in jeder Spielsituation aufgefasst werden. So besteht die Gesamtheit der Strategien bei Schere, Stein, Papier aus den drei bekannten Symbolen. Von einer reinen Strategie ist die Rede, wenn die Entscheidung für eine Strategie ganz bewusst getroffen wurde. Würde die Auswahl der Symbole per Zufall geschehen, beispielsweise mithilfe eines Würfels, so wird dies eine gemischte Strategie genannt.

Schere, Stein, Papier wird in der Spieltheorie auch als Nullsummenspiel bezeichnet. Dabei gewinnt der eine Spieler das, was der andere Spieler verliert - und umgekehrt.

An einer Auszahlungsmatrix (Darst. 3) lässt sich ein Nullsummenspiel sehr gut verdeutlichen.

Strategie	Schere	Stein	Papier
Schere	0	-1	1
Stein	1	0	-1
Papier	-1	1	0

Darstellung 3: Auszahlungsmatrix Nullsummenspiel

Der Gewinn wird in der Matrix mit dem Wert '1' dargestellt, der Verlust mit '-1'. Die Betrachtung der Tabelle ist von der Zeile aus zu vollziehen. Wenn der (Zeilen) -Spieler die Strategie Stein wählt, und der (Spalten) -Spieler Papier, so ist die Auszahlung für den Zeilenspieler -1 und für den Spaltenspieler 1. Ein wichtiger Faktor in SSP ist das zeitgleiche Anzeigen der gewählten Strategie. Wird das Symbol auch nur einen Bruchteil früher gezeigt, so ist es dem Kontrahenten theoretisch möglich, eine passende Gegenstrategie zu wählen.

Daher sollte ein jeder Schere, Stein, Papier-Spieler versuchen, das Verhalten des Gegners so gut wie möglich einzuschätzen. Wenn erst einmal die Strategie des anderen bekannt ist, ist es möglich zu 100 % zu gewinnen[3].

Menschen können nicht zufällig entscheiden, sondern haben immer gewisse Tendenzen. Eine Studie [19] hat ergeben, dass Menschen dazu neigen unterbewusst eine gesehene Geste zu kopieren. Aber auch Vorlieben, wie nicht zweimal das gleiche Symbol hintereinander zu spielen, oder Angewohnheiten, wie in der ersten Runde immer Schere zu zeigen, können gegen den Gegner verwendet werden. Die richtige Taktik ist, um das Ausnutzen von psychologischen Faktoren zu verhindern, eine gemischte Strategie zu spielen, also die Symbole zufällig (zu gleichen Wahrscheinlichkeiten) auszuwählen. Dies könnte mit einem Würfel bewerkstelligt werden. Die Symbole werden vor einem Spiel, für eine Anzahl von Spielzügen ausgewürfelt, anschließend aufgeschrieben und auswendig gelernt. Bei dem nächsten Spiel wird dann die imaginäre Liste abgearbeitet und die aufgeschriebenen Symbole gespielt.

2.5.4. Spiel und Spielstrategie im Kontext dieser Arbeit

Das Spiel wird im Kontext dieser Arbeit und im weiteren Sinne, als Stellvertreter für eine Vielzahl an Situationen verstanden, in denen ein Roboter mit Menschen interagiert. Dabei stellen die Spielstrategien Verhaltensweisen des Roboters dar, die unter Umständen auch in anderen Situationen Verwendung finden könnten.

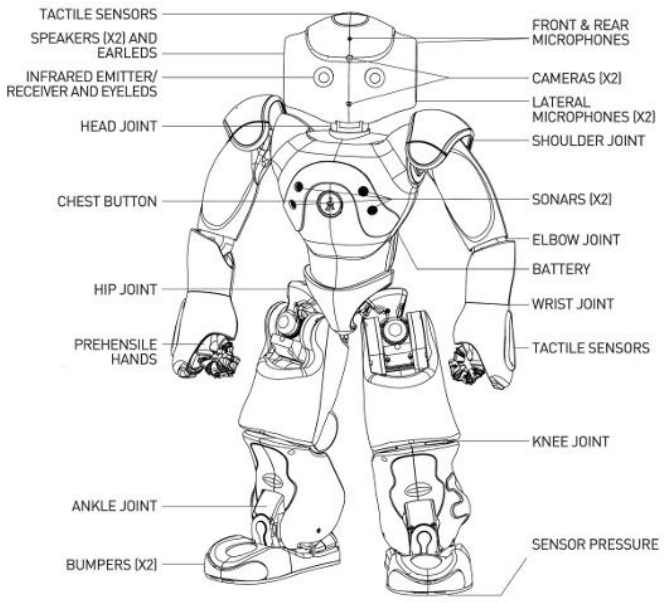
[3] In einem Experiment wurde ein Roboterarm gebaut, der mithilfe einer Hochgeschwindigkeitskamera einen Menschen zu 100 % bei SSP besiegen kann. Die Technologie soll als Beispiel für eine Mensch-Roboter-Kooperation dienen, die eine Kontrolle im Millisekundenbereich zulässt. Siehe auch http://www.k2.t.u-tokyo.ac.jp/fusion/Janken/index-e.html (last viewed: 26.12.13)

Kapitel 3

Eigener Ansatz

3.1. Mensch-Roboter-Interaktion mit NAO

NAO ist ein 58 cm großer, humanoider Roboter mit 25 Freiheitsgraden, der von dem französischen Unternehmen Aldebaran-Robotics entwickelt wurde. Seit 2007 ist NAO der Nachfolger von Sonys Aibo als Standardplattform des RoboCups. Es existieren unterschiedliche Ausführungen des Roboters, daher nachfolgend einige technische Merkmale, der in dieser Arbeit verwendeten Version:



Darstellung 4: Technische Ausstattung des NAO Roboters.4

NAO hat schon Algorithmen zum maschinellen Sehen, oder für die Spracherkennung bzw. Sprachausgabe vorinstalliert. So kann er Gesichter erkennen, sie verfolgen oder

[4] Quelle: Aldebaran-Robotics
Available from: https://community.aldebaran-robotics.com/doc/1-14/
(last viewed: 30.12.13)

gar einen roten Ball finden. Für eigene Module besteht die Möglichkeit, auf das Open Source Framework OpenCV zurückzugreifen. Die Spracherkennung ermöglicht die Benutzung von "command words", sprich das Erkennen einzelner Wörter, sowie den Einsatz von word spotting, also das Erkennen mehrerer einzelner Wörter in einem Satz.

3.1.1. Randbedingungen für den menschlichen Spieler

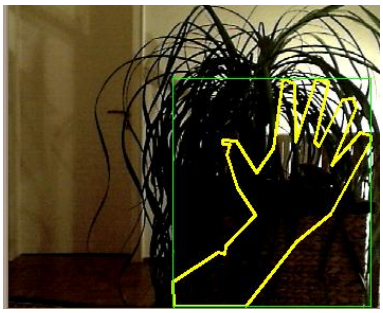
Damit die Interaktion mit NAO möglichst fehlerfrei funktioniert, muss der Nutzer gewisse Bedingungen einhalten.

Für die Spracherkennung müssen die Befehle auf Englisch ausgesprochen werden. NAO kann zwar theoretisch Deutsch verstehen, wurde aber auf dem verwendeten Roboter nicht installiert.

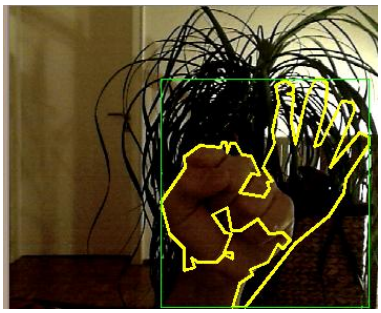
Generell ist es besser, wenn der Spieler sich nicht direkt in NAO's Sichtfeld befindet, da es sonst zu Fehlerkennungen kommen kann. Diese Bedingung ergibt sich durch die Segmentierung des Bildes mittels Background Subtraction (3.2.3).

Background Subtraction

Direkt, nach dem NAO gefragt hat, ob man mit ihm SSP spielen möchte, nimmt er, aufgrund der sehr klein eingestellten Lernrate (3.2.3), quasi ein statisches Hintergrundbild für die Hintergrund-Vordergrund-Segmentierung auf. Während dieses kurzen Zeitraums (eins bis zwei Sekunden) sollten sich keine beweglichen Objekte oder Personen vor seiner Kamera befinden. Ist dies doch der Fall, so wird das Objekt / die Person als Hintergrundobjekt angesehen. Bewegt sich das Objekt / die Person anschließend, so entsteht eine Lücke im segmentierten Bild (siehe Darst. 5), was zu Fehlerkennungen in diesem Bereich führt, (Darst. 6) solange das Hintergrundmodell nicht aktualisiert wurde.



Darstellung 5:
Eine Lücke im Hintergrundmodell.



Darstellung 6:
Fehlerkennung im Bereich der Lücke.

Eine weitere Randbedingung ist, dass der Spieler zum Starten der ersten Spielrunde seine Hand mit gespreizten Fingern dem Roboter zeigen muss. Dies ist wichtig für die (automatische) Selektion der Pixelverteilung (3.2.8), welche der Camshift-Algorithmus benötigt (3.2.7).

3.1.2. Randbedingungen für den Einsatz der Gestenerkennung

Die Umgebung, in der die Gestenerkennung eingesetzt wird, muss einige Voraussetzungen erfüllen, damit eine fehlerfreie Erkennung möglich wird. Die wichtigste Bedingung ist ein gleichbleibendes Lichtverhältnis. Gibt es häufige und starke Änderungen, so stören diese den Background Subtraction Algorithmus und es kommt zu Fehlerkennungen. Auch sollte auf Schattenwurf durch den Spieler geachtet werden. Ist der Schatten mit im Bild, so wird dieser als ein Vordergrundobjekt wahrgenommen und eine einwandfreie Erkennung der Hand ist nicht mehr gewährleistet. Dass gleiche gilt bei reflektierenden Objekten. Wird in der Spiegelung eine Bewegung erkannt, so wird diese als Vordergrundobjekt angesehen, was die anschließende Konturenerkennung verfälscht.

3.1.3. Verhaltensmuster von NAO

Die nachfolgenden Strategien repräsentieren verschiedene Verhaltensweisen von NAO, in der Situation des Spielens.

Fix Strategy

Ist die fixe Strategie eingestellt, so wird zufällig eines der drei Symbole (Schere, Stein, Papier) ausgesucht. Dieses Symbol wird dann (fix) für alle folgenden Spielrunden verwendet, bis der Nutzer das Spiel neu startet.

Würde dieses Verhalten auf reale Situationen übertragen, so kann es durchaus positiv bewertet werden, da der Roboter berechenbar ist und somit verlässlich wirkt. Vom Prinzip her, entspricht es dann dem Konzept eines Industrieroboters. Dieser führt seine programmierten Aufgaben immer mit denselben Bewegungen aus, und garantiert daher eine gleichbleibende Qualität seiner Arbeit (und der Produkte).

Im Rahmen der Nutzerstudie wird angenommen, dass die fixe Strategie mit am schlechtesten abschneidet. In allen drei zu evaluierenden Kategorien, bei der Natürlichkeit, Menschlichkeit und des Spaßfaktors, werden negative Bewertungen erwartet.

Random Strategy

Die Zufallsstrategie wählt für jeden Spielzug eines der drei Symbole mithilfe eines Zufallsgenerators aus.

Auf eine andere Situation, als das Spielen, angewandt, kann dieses unberechenbare Verhalten sogar positiv bewertet werden. In der Studie von M. Salem et al. [20] wurde festgestellt, dass ein Roboter positiver von Menschen wahrgenommen wird, wenn er zu seinem Gesprochenen partiell richtige Geste zeigt, als wenn er gar keine Gesten verwenden würde. Ein gewisser Grad an Unberechenbarkeit eines humanoiden Roboters lässt ihn menschlicher wirken, und beeinflusst somit die Mensch-Roboter-Interaktion im positiven Sinne. [20 p. 40]

In Bezug auf die Nutzerstudie wird erwartet, dass diese Strategie, aufgrund der Natürlichkeit und der Menschlichkeit in einer SSP-Spielsituation, mit am besten abschneidet. Insbesondere der Spaßfaktor dürfte hoch bewertet werden.

Robot wins

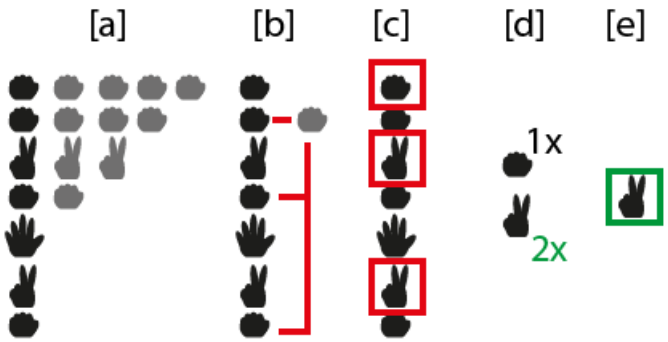
Die Strategie Robot wins (100% of the time) betrügt beim Spielen, indem der Roboter, aufgrund seiner schnellen Wahrnehmung vor seiner Entscheidung für ein Symbol, erfährt, was der menschliche Spieler für eine Geste geformt hat. Anschließend wählt er sein Symbol so, dass dieses gegen den Spieler gewinnt.

In einer realen Situation ist dieses Verhalten nicht nur nützlich, sondern manchmal auch zwingend erforderlich. Zum Beispiel dann, wenn mit einem Menschen zusammengearbeitet werden muss. Mithilfe einer Hochgeschwindigkeitskamera kann der Roboter die Bewegungen des Menschen extrem schnell erfassen, dessen Intention bestimmen und entsprechend darauf reagieren.

Pattern Analysis

Die Strategie Pattern Analysis ist nach dem Vorbild auf der Webseite der New York Times [21] entstanden. Der Algorithmus geht davon aus, dass Menschen nicht zufällig entscheiden können, sondern gewisse Entscheidungsmuster aufweisen, die dann dafür genutzt werden können, den nächsten Zug vorauszusagen.

Für die Strategie speichert NAO jedes gezeigte Symbol des Spielers in einer Datei ab. Nachdem mindestens fünf Symbole abgespeichert wurden, beginnt der Roboter damit, diese über die Zeit wachsende Liste nach Verhaltensmuster zu durchsuchen. Zuerst nimmt er die letzten vier gespielten Symbole des Menschen und sucht nach ihnen in der Liste. Wird er nicht fündig, so wird das älteste Zeichen aus dem zu suchenden Muster entfernt. Ist er auch mit diesem Muster nicht fündig, so wiederholt er die Maßnahme so lange, bis nur noch nach einem Symbol gesucht wird (Darst.7, [a]). Spätestens jetzt wird der Roboter einen oder mehrere Treffer finden (Darst.7, [b]). Anschließend wird das auf das Muster nachfolgende, gespielte Symbol als das wahrscheinlichste Symbol des Spielers für die nächste Runde angenommen (Darst.7, [c]). Bei mehreren Treffern, werden alle Symbole gezählt, und dass mit der höchsten Summe wird, als mögliche Entscheidung des Spielers für die nächste Runde, akzeptiert (Darst.7, [d], [e]). Darauf hin bestimmt der Roboter sein Zeichen so, dass es gegen das angenommene Symbol des Gegenspielers gewinnt.



Darstellung 7: Ablauf der Pattern Analysis.

Das Interessante an dieser Strategie ist, dass sich NAO nach einer gewissen Zeit (ca. 15 Spielzüge) tatsächlich langsam, aber sicher, an die Verhaltensweise des Spielers gewöhnt. Somit kann er immer öfters vorhersagen, welches Symbol der Spieler in der nächsten Runde zeigen wird. Natürlich gewinnt der Roboter dadurch nicht jede Runde, insbesondere dann nicht, wenn der Spieler sich der Strategie des Roboters bewusst ist,

und gewollt aus seinem Verhalten ausbricht. Aber es kann schon sehr frustrierend sein, wenn NAO wesentlich öfters gewinnt, als er verliert.

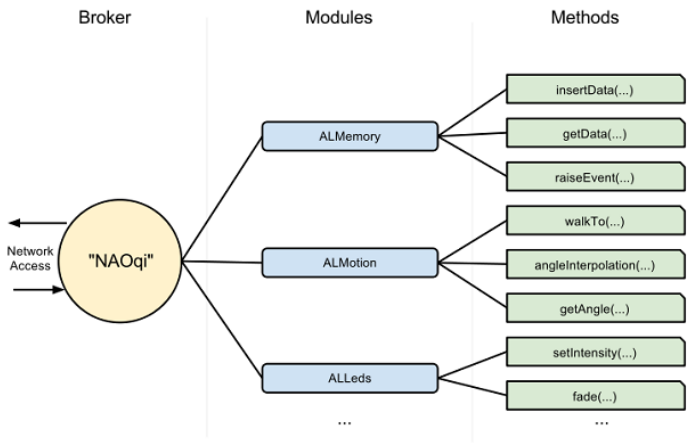
Wird dieses Verhalten auf eine reale Situation übertragen, so könnten Angewohnheiten eines Menschen gespeichert und, wenn nötig, mit in Betracht gezogen werden. Beispielsweise hat eine ältere Person die Angewohnheit, seine Schlüssel zu vergessen. Dann könnte der Roboter in dieser Situation darauf reagieren, indem er seinen Besitzer daran erinnert, an die Schlüssel zu denken.

In der Nutzerstudie wird erwartet, dass diese Strategie mit der Zufallsstrategie konkurriert oder sogar besser bewertet wird. Das liegt vor allem an der Möglichkeit des Roboters, auf ein unnatürliches Verhalten des Spielers zu reagieren. Spielt der Mensch immer Stein, was bei der Random-Strategie nicht beachtet werden würde, so kann der Roboter mit Pattern Analysis darauf reagieren und entsprechend Papier zeigen.

3.1.4. NAO's Sprache und Bewegung

Broker, Proxy und Module

Die Software, die auf NAO läuft und sich um die Steuerung des Roboters kümmert, heißt NAOqi. Der NAOqi-Prozess, der nach dem Einschalten von NAO gestartet wird, lädt einen sogenannten Broker (Main Broker). Ein Broker lädt Bibliotheken, die verschiedene Module enthalten, und deren Methoden durch den Broker für andere Module zur Verfügung gestellt werden (Look-Up-Service). Als weitere Aufgabe bietet er einen Netzwerkzugriff an, damit die Module, die bei ihm registriert sind, auch remote aufgerufen werden können. Diese Zusammenhänge werden in Darstellung 8 dargestellt.



Darstellung 8: Zusammenhang Broker, Module und Methoden.5

Die einzelnen Module stellen Methoden für die Funktionen des Roboters bereit. Ein Modul kann lokal oder remote ausgeführt werden. Eine lokale Ausführung bedeutet, dass das Modul direkt auf NAO ausgeführt und im selben Prozess wie alle anderen Module gestartet wird. Dadurch ist eine Kommunikation der Methoden untereinander, nur mittels einen Broker möglich. Der Vorteil von lokalen Modulen ist, dass sie eine sehr gute Geschwindigkeit bieten. Der Nachteil ist, dass wenn sie abstürzen, auch den Broker, mit dem sie verbunden sind, zum Abstürzen bringen. Ist dieser Broker gerade der Main Broker, wird der Roboter sämtliche Bewegungen einstellen und eventuell umfallen. Module, die ihren eigenen Broker verwenden, wie die Remote-Module, können dieses Problem nicht auslösen.

Remote-Module laufen auf einem PC und kommunizieren mit dem Roboter über ein Netzwerk. Dazu wird ein zweiter Broker benötigt, an dem die Remote-Module angeheftet werden. Dieser zweite Broker wird dann mit dem Main Broker, der auf NAO läuft, verknüpft und ermöglicht so das Aufrufen der lokalen Module bzw. Methoden.

Ein Proxy ist ein Objekt, welches ein Modul repräsentiert und dessen Methoden zur Verfügung stellt.

[5] Quelle: Aldebaran- Robotics
Available from: https://community.aldebaran-robotics.com/doc/1-14/dev/naoqi/index.html#naoqi-overview
(last viewed: 30.12.13)

Spracherkennung und Sprachausgabe

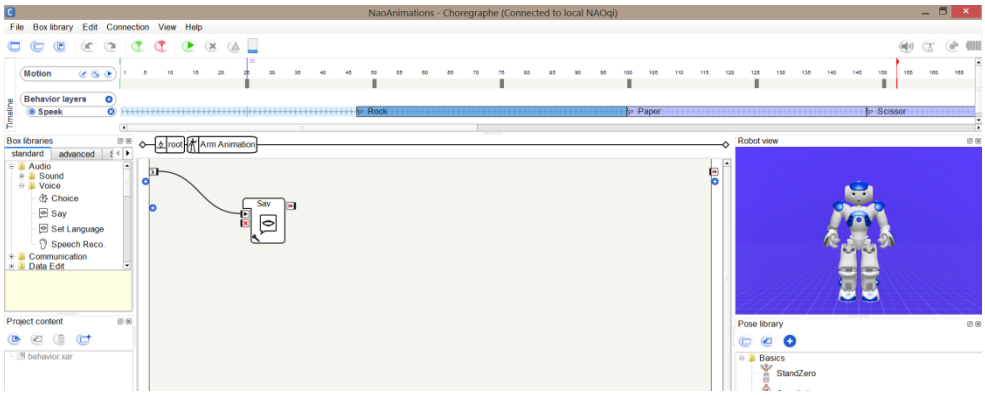
Für die Spracherkennung und Sprachausgabe werden die in NAO integrierten Module ALSpeechRecogniton und ALTextToSpeech verwendet. Das Modul für die Sprachausgabe wandelt einen eingegeben Text in Sprache um. Dazu muss erst ein Proxyobjekt erstellt werden, welches die Methode zur Text-in-Sprache-Umwandlung zur Verfügung stellt.

Die Sprachausgabe wird beim Spielen mit NAO dazu verwendet, dem Spieler verbal Informationen zu den Geschehnissen zukommen zu lassen. Das sind Informationen zu der aktuellen Spielrunde, die gespielte Geste von NAO, die erkannte Geste des Spielers, den Gewinner der Runde und den aktuellen Punktestand. Außerdem zeigt er mittels Sprachausgabe auch seine Freude und Enttäuschung, wenn er gewinnt oder verliert.

Die Spracherkennung bietet die Option, entweder auf einzelne Wörter zu achten (command words) oder auf mehrere Wörter, die in einem Satz gefunden werden (word spotting). Sobald NAO ein Wort erkannt hat, wird dieses Wort in sein Gedächtnismodul (ALMemory) abgelegt. ALMemory ist ein Array vom Typ ALValue. ALValue ist ein universeller Container für Datentypen, wie String oder Images. Auf die Werte im Gedächtnismodul kann anschließend von anderen Modulen zugegriffen werden. So ist es schließlich möglich eine Abfrage zu entwickeln, um, wie in dieser Arbeit, den Spielfluss zu steuern. NAO stellt nach dem Starten des "Schere, Stein, Papier" - Programms die Frage, ob der Spieler mit ihm überhaupt SSP spielen möchte. Dieser muss dann mit "Yes" oder "No" antworten. Wird mit "Yes" geantwortet, so startet das Spiel, wird hingegen mit "No" geantwortet, beendet sich das Programm. Ebenso kann am Ende einer jeden Spielrunde entschieden werden, ob eine weitere Runde gestartet, oder ob das Programm beendet werden soll.

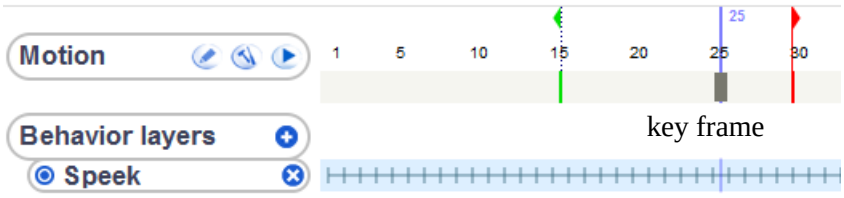
Bewegungen: Choregraphe

Die Bewegungsanimationen von NAO wurden mithilfe der beigelegten Software Choregraphe (Darst. 9) erstellt. Choregraphe stellt eine grafische Benutzeroberfläche bereit, mit der Animationen oder ganze Verhaltensweisen erstellt und mit einem simulierten Roboter getestet werden können. Dazu werden abgespeicherte Animationen per Drag & Drop in die Mitte des Programms gezogen und dort mit "Signalkabeln" verbunden. So ist es auch möglich, mehrere Animationen oder Funktionen des Roboters miteinander zu verbinden und ein Verhalten zu kreieren.



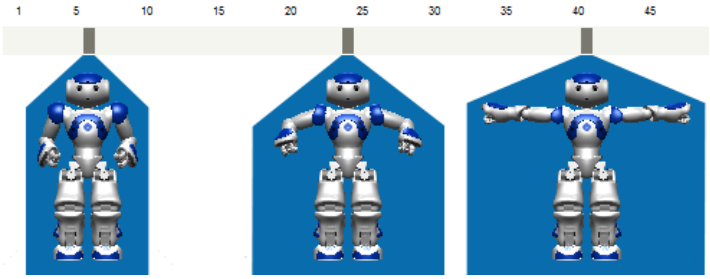
Darstellung 9: Der Editor Choregraphe von Aldebaran-Robotics.

Die Arm- und Fingerbewegungen für das SSP-Spiel wurden manuell über die ein gebaute Zeitleiste (timeline) erstellt. Die grafische Darstellung des Zeitstrahls (Darst. 10) besteht aus einzelnen Bildern (frames) und Schlüsselbildern (key frames). Die key frames bilden einzelne Zustände von NAO ab. In der Darstellung 10 können zwei Marker gesehen werden. Der grüne Marker stellt die Startposition und der rote Marker die Endeposition einer Animation dar.



Darstellung 10: Aufbau der timeline.

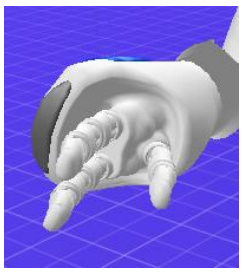
Innerhalb dieser Spanne werden alle Posen von NAO der Reihe nach abgespielt (Darst.11). Die Software berechnet automatisch die Bewegung, die nötig ist, um von der aktuellen Haltung in die neue Pose zu gelangen.



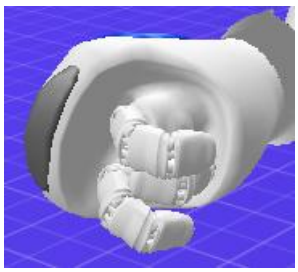
Darstellung 11: Einzelne key frames mit den gespeicherten Posen.6

3.1.5. "NAO Style" der Gesten für das SSP-Spiel

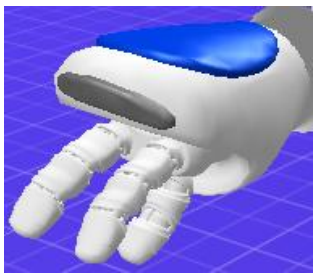
Die Hand von NAO besteht nur aus drei Fingern. Diese können auch nur simultan geöffnet oder geschlossen werden. Somit ist die Art und die Anzahl der Gesten, die damit geformt werden können, sehr eingeschränkt. Für das Spiel sind wenigstens drei Symbole erforderlich. Die nachfolgenden Darstellungen zeigen, wie NAO die Gesten für Schere (Darst. 12), Stein (Darst. 13) und Papier (Darst. 14) formt.



Darstellung 12: Symbol für Schere.



Darstellung 13: Symbol für Stein.



Darstellung 14: Symbol für Papier.

[6] Quelle: Aldebaran- Robotics
Available from:
https://community.aldebaran-robotics.com/doc/1-14/software/choregraphe/panels/timeline_panel.html
(last viewed: 30.12.13)

3.2. Handerkennung für NAO, als intuitive und robuste Kommunikationsmöglichkeit

3.2.1. Maschinelles Sehen

Das Binokularsehen ermöglicht uns Menschen eine räumliche Wahrnehmung unserer Umgebung. Wir können ungefähr abschätzen, wie weit ein Objekt entfernt ist, wo es im Raum positioniert ist, und ob es vor oder hinter einem anderen Objekt steht. Wir können Gegenstände anhand von ihren speziellen Merkmalen, wie etwa Form oder Farbe, eindeutig identifizieren und voneinander unterscheiden. Eine Hand erkennen wir beispielsweise schon allein durch das Wissen, an welcher Position diese am Körper zu finden ist. Um die bis dahin bestehende Vermutung, es handle sich um eine Hand, zu bestätigen, orientieren wir uns an ihrer speziellen Form. Die Finger sind charakteristisch und maßgebend dafür. Eine Form ist aber nicht nur ein wichtiger Faktor, um etwas zu identifizieren, sondern auch um dem Identifizierten eine Bedeutung zu geben. Das beste Beispiel im Kontext dieser Arbeit sind Gesten, die von einer Hand geformt werden und in der Gebärdensprache bestimmte Bedeutungen haben. Damit ein Roboter Gesten überhaupt verstehen kann, sind verschiedene Algorithmen notwendig, die das Lokalisieren, Identifizieren und Erkennen der Zeichen übernehmen. In dieser Umsetzung der Handerkennung müssen folgende aufeinander aufbauende Ebenen durchlaufen werden, damit am Ende eine Geste erkannt werden kann.

1. Hintergrund - Vordergrund Segmentierung
2. Konturerkennung
3. Berechnung der konvexen Hülle und konkaven Einbuchtungen
4. Region of Interest
5. Camshift-Algorithmus für das Verfolgen der Hand
6. Fingererkennung

Zur Entwicklung der Gestenerkennung ist die unter der BSD-Lizenz stehende Programmbibliothek OpenCV, benutzt worden. Sie umfasst eine große Menge an Algorithmen für das Computersehen aus der Forschung. Darunter auch die in dieser Arbeit verwendeten Verfahren:

- **Background Subtraction:** BackgroundSubtractor MOG2 [22]
- **Konturenfindung:** FindContours [23]
- **Konvexität:** ConvexHull [24] und ConvexityDefects
- **Objektverfolgung:** CAMSHIFT [25]

Der erste Teil der Handerkennung beschäftigt sich mit der Segmentierung. Dazu wird ein Bild in Bereiche zerlegt und somit das Lokalisieren der Handform vereinfacht. Für die Zerlegung gibt es verschiedene Verfahren, von denen einige nachfolgend kurz erläutert werden sollen.

Die pixelorientierten Verfahren segmentieren ein Bild, indem jeder einzelne Bildpunkt, unabhängig von seiner Umgebung oder seiner Nachbarn, überprüft, und anhand von Merkmalen einem Bereich zugeordnet wird. Eine einfache Umsetzung ist das Thresholding. Bei dieser Methode wird jedes Pixel und dessen Intensität mit einem Schwellenwert verglichen. Liegt die Intensität unterhalb des Schwellenwertes, so wird dem Pixel eine 0 zugewiesen. Liegt die Intensität oberhalb, so erhält der Pixel eine 1. Das daraus resultierende Binärbild enthält zwar nicht zwingend zusammenhängende Segmente, aber nach dem Homogenitätskriterium (Schwellenwert) gefundene Pixel. Somit kann je nach Bild und Objekt[7] (grob) entschieden werden, ob ein Pixel zu einem gesuchten Objekt oder dessen Umgebung gehört. Zu beachten ist jedoch, dass das Ergebnis nicht aussagt, ob das gefundene Objekt tatsächlich sinnvoll ist.

Die kantenorientierten Verfahren versuchen, entlang von Linien oder Objektübergängen Kanten zu entdecken. Eine Kante wird als solche angesehen, wenn zwei benachbarte Pixel starke Unterschiede aufweisen, wie zum Beispiel in der Intensität oder der Farbe. Je nach verwendetem Algorithmus werden alle offenen und geschlossenen Kanten eines Bildes geliefert. Der Canny Edge Algorithmus gehört zu den Methoden, die beide Arten von Kanten liefert. Für die Objekterkennung ist das jedoch suboptimal, da Objekte sich

[7] Wenn in dieser Arbeit von einem Objekt gesprochen wird, ist damit nicht nur ein lebloses Objekt gemeint, sondern durchaus auch Tiere, Personen oder einzelne Körperteile.

hauptsächlich durch geschlossene Konturen definieren und auch so erst erkannt werden können. Daher müssen die offenen Kantenzüge anschließend noch mit Kantenverfolgungsalgorithmen geschlossen werden. Ein Beispiel, welches geschlossene Kanten liefert, ist die findContours ()-Funktion von OpenCV.

Die modellbasierten Verfahren setzen voraus, dass Wissen über die Form des gesuchten Objekts vorhanden ist. Generell benutzen die bisher vorgestellten Methoden nur lokale Informationen (einzelne Pixel oder Bereiche etc.). Sobald ein Objekt aber nicht komplett abgebildet ist, wird dieses auch nicht mehr erkannt. Der modellbasierte Ansatz nutzt die Fähigkeit von Algorithmen, wie die Hough-Transformation, die Linien und andere geometrische Figuren erkennen können, um den fehlenden Teil zu ergänzen. Anschließend werden Matching-Algorithmen eingesetzt, um die erkannte Figur mit einem Modell zu vergleichen. Das Template-Matching ist ein solcher Algorithmus.

Texturorientierte Verfahren helfen bei der Segmentierung von nicht einfarbigen Objekten. So besitzen die Oberflächen eines Objektes oftmals ein Muster bzw. eine einheitliche Struktur. Ein gutes Beispiel dafür ist ein Schachbrett. Texturorientierte Verfahren sehen die Textur sozusagen als Homogenitätskriterium an, um das Bild danach zu segmentieren.

3.2.2. Segmentierung durch Hautfarbenerkennung

Für den ersten Ansatz ist aufgrund der Einfachheit und Geschwindigkeit die Wahl auf ein pixelorientiertes Verfahren gefallen. Die Hautfarbe ist ein charakteristisches Merkmal für eine Hand, daher ist die Idee also naheliegend das Bild nach diesen Pixeln zu segmentieren. Als Schwellenwert wird dafür ein Wertebereich gewählt. Fällt ein Pixel in diesen Bereich, so wird er als Hautfarbe klassifiziert und mit einer 1 markiert, andernfalls mit 0. Das entstandene Binärbild sollte dann die Form der Hand als einen mit weißen Pixeln gefüllten Bereich darstellen. In der Theorie klingt das soweit vielversprechend, bei den ersten Tests konnte dieser Ansatz jedoch nicht ganz überzeugen. Generell steht man bei Verwendung dieser Methode vor folgenden Problemen:

1. Die richtige Auswahl des Wertebereichs.
2. Verfälschte Eingabebilder bedingt durch die Kamera

3. Objekte mit ähnlichen Farbwerten
4. Wechselnde Beleuchtung

Problem 1: die richtige Auswahl des Wertebereichs

Die Auswahl des richtigen Wertebereichs spielt eine zentrale Rolle. Es ist jedoch schwer, insbesondere in Hinsicht auf den praktischen Einsatz der Handerkennung, einen allgemeingültigen Wertebereich festzulegen. Denn die Hautfarbe ist von Mensch zu Mensch verschieden. So ist die Farbe von Europäern eher weiß, die von Asiaten bewegt sich im gelblichen Bereich und die von Farbigen ist eher dunkel. Natürlich kann nun argumentiert werden, man müsse einfach nur vorher festlegen, welcher Hautfarbentyp erkannt werden soll. Aber der Teint ist nicht zu jeder Zeit gleichbleibend! Die Menge an Melanin, das für die Pigmentierung ausschlaggebend ist, wird durch die Sonne vermehrt gebildet. Dies führt normalerweise zu einer dunkleren Hautfarbe im Sommer. Weiterhin ändert sich der Farbton in Abhängigkeit des Gesundheitszustandes. Bei Erkältung kann die Haut blass werden, bei Fieber eher ins Rötliche gehen. Somit müsste mit saisonal oder gesundheitlich bedingten Schwankungen der Hautfarbe eine Änderung des Wertebereichs einhergehen. Daher ist eine manuelle Festlegung von Wertebereichen wenig praktikabel. Eine automatisierte Selektion wäre dafür eine Lösung, wirft aber eine weitere Frage auf: "Wie kann der Roboter bzw. der Algorithmus für die Farberkennung wissen, von welchem Teil des Bildes die Farbwerte ausgelesen werden sollen, wenn er das entsprechende Objekt nicht kennt?". Insbesondere bei humanoiden Robotern, die eine manuelle Markierung der Bereiche aufgrund fehlender Eingabemöglichkeiten (Touch, Maus, Tastatur etc.) nicht zulassen, ist das eine berechtigte Frage. Ein Lösungsansatz hierfür wird in 3.2.8 vorgeschlagen.

Problem 2: Verfälschte Eingabebilder bedingt durch die Kamera.

Die Verfälschung von Eingabebildern bedingt durch die Kamera ist ein weiteres Problem, welches gelöst werden muss. Die meisten Softwareprogramme, welche die Kamera steuern, besitzen Einstellungen für die Belichtung, für den Weißabgleich, für eine Gesichtsverfolgung und für einen Zoom. Diese Einstellungen werden automatisch anhand der Umgebung geregelt. Damit soll ein besseres Bild erreicht werden, was prinzipiell auch stimmt. Allerdings werden die Bilder, die bei der Bildverarbeitung analysiert werden sollen, dadurch verfälscht. Das kann je nach verwendeten Algorith-

men zu schweren Problemen führen und diese sogar nutzlos werden lassen. Das folgende Beispiel soll die Problematik näher erläutern.

Gegeben ist ein Raum, der durch eine normale Deckenlampe beleuchtet wird. Für die Kamera ist das möglicherweise zu wenig Licht, also stellt sie die Belichtung automatisch höher. Dadurch wird nicht nur das gesamte Bild aufgehellt, sondern auch die Hautfarbe. Ist der Wertebereich zu klein gewählt worden, dann kann es sein, dass die Hand nicht mehr oder nur noch teilweise erkannt wird.

Die Lösung für das Problem ist jedoch denkbar einfach: Die automatischen Einstellungen werden, insofern möglich, einfach deaktiviert. Dadurch treten keine Schwierigkeiten mehr bei den verwendeten Bildverarbeitungsalgorithmen auf. Einen nicht akzeptablen Nachteil für mobile Roboter hat die Lösung aber trotzdem: Die Kamera muss jedes Mal erneut kalibriert werden, sobald sich die Lichtverhältnisse ändern.

Problem 3: Objekte mit ähnlichen Farbwerten, wie die Haut.

Das nächste Problem, welches gelöst werden muss, besteht schon aufgrund der Natur von pixelorientierten Algorithmen. Nachdem lediglich Pixel für Pixel überprüft wird, ob ein Schwellenwert überschritten wurde oder nicht, spielt es für den Algorithmus keine Rolle, ob der betrachtete Pixel tatsächlich zum gesuchten Objekt gehört oder nicht. Daher werden auch sämtliche Holzmöbel, Kleidungsstücke oder Objekte mit hautfarbenen Töne erkannt.

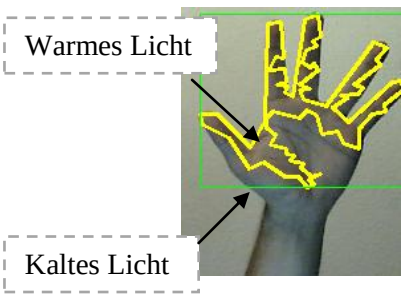
Zur Lösung dieses Problems können statische Gegenstände, die im Hintergrund stehen, mittels Vordergrund-Hintergrund Segmentierung (3.2.3) entfernt werden. Gesichter können mit Hilfe von Gesichtserkennungsalgorithmen erkannt und ignoriert werden. Und andere Objekte sind aufgrund ihrer Größe leicht herauszufiltern.

Problem 4: wechselnde Beleuchtung.

Ähnlichfarbige Objekte sind zwar ein Problem, eine wechselnde Beleuchtung aber ein ungleich Größeres. Bei der Betrachtung der Hand unter verschiedenen Lichtquellen fällt sofort auf, wie sich die Hautfarbe durch diese ändert. Unter dem Tageslicht erscheint

der Teint fast weiß, in der Nacht ist er sehr dunkel. Auch verschiedene künstliche Lichtquellen (warm oder kalt) führen zu einer Änderung der Hautfarbe.

Es gibt den Hinweis, dass die Wahl eines geeigneten Farbraums eine Möglichkeit ist, um dieses Problem anzugehen. So gibt es auf der einen Seite die technisch-physikalischen Farbmodelle, bei denen Farben aus anderen Farben gemischt werden, wie RGB (Red, Green, Blue) oder CMYK (Cyan, Magenta, Yellow, [Key] Black). Ändert sich hier ein Wert in einem der Farbkanäle, so verschiebt sich automatisch die Helligkeit, Sättigung und der Farbton. Auf der anderen Seite gibt es wahrnehmungsorientierte Farbmodelle, die Farben durch Helligkeit, Sättigung und Farbton beschreiben. Dazu zählt zum Beispiel HSV (Hue, Saturation, Value) oder YCbCr (Blue-Yellow Chrominance, Red-Green Chrominance). Hier sind die Farbkanäle, im Gegensatz zu den technisch-pyhsikalischen Farbmodellen, unabhängig voneinander und können einzeln bearbeitet werden. Um also die Auswirkungen der Lichtveränderung zu vermindern, kann der Helligkeitskanal komplett entfernt werden. Übrig bleibt dann nur noch der Farbton und die Sättigung, was infolgedessen eine weniger feine Farbabstufung ergibt. Da davon ausgegangen wird, dass die Hautfarbe sich hauptsächlich in der Helligkeit verändert und nicht so sehr im eigentlichen Farbton, ist dieser Ansatz durchaus sinnvoll um die Robustheit der Farberkennung zu erhöhen. Bei den durchgeführten Tests und leichten Lichtveränderungen konnte dies auch festgestellt werden. So ist die Robustheit beim Drehen der Hand tatsächlich leicht verbessert. Bei starken Lichtänderungen (Lichtart ändert sich von warm auf kalt) hingegen, lieferte der Algorithmus immer noch kein überzeugendes Ergebnis. Darstellung 15 verdeutlicht die Auswirkung unterschiedlicher Lichtarten. Zur Veranschaulichung wurde die Hand mit einem warmen Licht von oben und mit einem kalten Licht von unten bestrahlt. Darstellung 16 zeigt die erkannte Hautfarbe in weißen Pixeln an



Darstellung 15:
Fehlerkennung aufgrund verschiedener Lichtarten.

Darstellung 16:
Erkannte Hautfarbe wird in weißen Pixeln angezeigt.

Für die Gestenerkennung wird jedoch zwingend eine gute Form der Hand benötigt, damit diese in späteren Schritten analysiert werden kann. Dies führte am Ende zu der Entscheidung, den Ansatz der Hautfarbenerkennung, zumindest für den Schritt der Segmentierung, komplett zu verwerfen und stattdessen das Ergebnis der Hintergrund-Vordergrund Segmentierung als Grundlage für die Gestenerkennung zu verwenden.

3.2.3. Segmentierung mittels Trennung des Hintergrunds vom Vordergrund

Der Themenbereich Hintergrundsubtraktion (Background Subtraction) beschäftigt sich mit der Trennung des Hintergrunds vom Vordergrund. In vielen Anwendungen des Computersehens, wie der Videoüberwachung, ist der Hintergrund unwichtig. Tatsächlich führt er eher zu einer häufigeren Fehlerkennung von Objekten, wie bereits im Abschnitt der Hauterkennung angesprochen. Die eigentlich interessanten Objekte, zum Beispiel Fahrzeuge oder Personen, sind beweglich, und daher nicht immer im Bild vorhanden. Der Hintergrund kann grundlegend auf zwei verschiedene Arten gefiltert werden. Entweder hardwaretechnisch, wie Kinect es tut, indem die Kamera diesen Schritt mithilfe von Distanzinformationen bereits erledigt, oder softwaretechnisch. Für NAO, der nur eine einfache Kamera eingebaut hat, ist die zuletzt genannte Variante nötig.

Eine recht simple Methode, um bewegte Objekte von Statischen zu unterscheiden, ist das sogenannte frame difference - Verfahren [26] . Es werden fortlaufend zwei direkt aufeinander folgende Bilder (frames), Pixel für Pixel voneinander subtrahiert.

Sei Δ das Ergebnis der Subtraktion, $frame_t$ und $frame_{t-1}$ die benötigten Bilder zum Zeitpunkt t, sowie das jeweilige Pixel an der Position (z, s), dann lautet die Formel für die Differenz:

$$D(z,s) = |frame_t(z,s) - frame_{t-1}(z,s)|$$

Das resultierende Differenzbild D wird anschließend in ein Binärbild B umgewandelt, indem jeder Bildpunkt mit einem Schwellenwert verglichen wird. Dafür gilt:

$$B(z,s) = \begin{cases} 0, & \Delta(z,s) < Schwellenwert \\ 1, & sonst \end{cases}$$

Liegt der Wert unterhalb des Schwellenwerts, wird der Wert des Pixels auf 0 gesetzt, ansonsten auf 1.

Somit beinhaltet das Binärbild Konturen der sich bewegenden Objekte, aber keine ganze Fläche. Das liegt in erster Linie daran, dass bei beiden Bildern, $frame_t$ und $frame_{t-1}$, die Vordergrundobjekte mit im Bild sind. Dadurch ist die Veränderung der Pixel, innerhalb der von der Kontur umspannten Fläche, nicht hoch genug, um den Schwellenwert zu überschreiten. Dies kann unter Umständen für die Weiterverarbeitung der Objekte schwerwiegende Folgen haben. So können einzelne Objekte durchtrennt und in mehrere kleine, falsch erkannte, Fragmente aufgeteilt werden. Für die Objekt- oder Handerkennung fatal. Auch Objekte, die plötzlich die Bewegung einstellen, wie die Hand beim Zeigen einer Geste, werden sofort als Hintergrund klassifiziert, sobald keine Differenzen der Pixel mehr wahrgenommen werden. Für die Erkennung von Bewegungsgesten (Wischen von links nach rechts o.ä.) mag diese Methode ausreichend sein, aber für Handzeichen, die einige Sekunden ruhig gehalten werden, ist sie nicht verwendbar. Alternativ könnte statt eines vorhergehenden Bilds $frame_{t-1}$ ein statisches Bild des Hintergrunds, ohne Vordergrundobjekte, genommen werden. Damit wäre auch das Problem der Flächenerkennung und des plötzlichen Bewegungsstopps gelöst. Nachdem das Objekt gar nicht im Hintergrundbildmodell existiert, sind die Abweichungen der Pixel an jeder Stelle hoch genug, um den Schwellenwert zu überschreiten. Allerdings reagiert diese Variante extrem auf Änderungen der Umgebung. So führen Lichtveränderungen, verschobene Objekte oder Bäume, deren Blätter sich im Wind bewegen, zu Fehlerkennungen.

Wegen der Schwächen, der vorangegangen Verfahren, sind adaptive Background Subtraction - Verfahren entwickelt worden. Sie passen das Hintergrundmodell langsam an Änderungen der Umgebung an. Eines davon ist der MOG2-Algorithmus (Mixture of Gaussian), der in OpenCV implementiert ist. Er basiert auf dem Verfahren, das in [27] vorgestellt wird. Aufgrund der Breite dieser Masterarbeit soll jedoch auf eine vollständige mathematische Beschreibung dieses Algorithmus (und nachfolgender Algorithmen) verzichtet werden. Stattdessen wird versucht, einen groben Überblick der Funktionsweise zu geben. Zum genauen Verständnis der Arbeitsweise sei auf die jeweiligen Forschungspapiere verwiesen.

Die Funktionsweise des MOG-Algorithmus [27] lässt sich, wie folgt, beschreiben:

Sei I eine Videosequenz, dann ist zu jedem Zeitpunkt t die Historie der Werte eines Pixels (x_0, y_0) bekannt. Das lässt sich mathematisch ausdrücken, mit [27 p. 3]:

$$\{X1, \ldots, Xt\} = \{I(x_0, y_0, i) : 1 \leq i \leq t\}$$

Die jüngste Historie eines jeden Pixel, wird mit einer Mischung aus K Gaußverteilungen modelliert. Die Wahrscheinlichkeit, den aktuellen Wert eines Pixels zu beobachten, kann mit folgender Formel berechnet werden [27 p. 3]:

$$P(X_t) = \sum_{i=1}^{K} \omega_{i,t} * \eta(X_t, \mu_{i,t}, \Sigma_{i,t})$$

Dabei ist K die Anzahl an Gaußverteilungen (normalerweise $K \in \{3,5\}$). Die Gaußsche Wahrscheinlichkeitsdichtefunktion η lautet [27 p. 3]:

$$\eta(X_t, \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}} | \sum |^{\frac{1}{2}}} e^{-\frac{1}{2}(X_t - \mu_t)^T \Sigma^{-1} (X_t - \mu_t)}$$

Jeder neuer Wert eines Pixel wird mit den K Gaußverteilungen solange verglichen, bis ein Treffer gefunden wurde. Ein Treffer wird als solcher definiert, wenn der Wert des Pixels sich innerhalb der Standardabweichung von 2,5 der Verteilung befindet. Wird kein Treffer gefunden, so wird die am wenigsten wahrscheinliche Verteilung, mit der Verteilung des aktuellen Wertes und dessen Mittelwert ersetzt. Anschließend wird die Gewichtung der vorherigen Verteilungen angepasst, wobei der Wert der Lernrate für das Modell mit einen Treffer auf 1, und für das Modell ohne Treffer auf 0 gesetzt wird.

Die Anpassung der Gewichtung erfolgt mit der Formel [27 p. 4]:

$$\omega_{k,t} = (1 - \alpha)\omega_{k,t} - 1 + \alpha(M_{k,t})$$

Somit können beobachtete Veränderungen der Szene, z. B. Lichtveränderungen, in das Hintergrundmodell übernommen werden.

Zur Klassifizierung von Pixeln in Vorder- und Hintergrundpixel, sortiert man zuerst die Gaußschen Funktionen nach dem Wert $\frac{\omega}{\sigma}$, der eine Liste ergibt, in der die wahrscheinlichsten Hintergrundverteilungen an der Spitze bleiben, während die unwahrscheinlichsten Hintergrundverteilungen unten angesiedelt sind. Dort werden sie eventuell von anderen, neueren Verteilungen ersetzt. Anschließend werden die ersten B oberen

Verteilungen als das Hintergrundmodell angesehen, die einen Schwellenwert T überschreiten.

Der MOG2-Algorithmus, der in dieser Arbeit verwendet wird, verbessert den MOG-Algorithmus dahin gehend, dass nicht nur die Parameter automatisch angepasst werden, sondern auch die Anzahl der K Gaußverteilungen. Daraus resultiert eine verringerte Verarbeitungszeit und eine leicht verbesserte Segmentierung [25 p. 4].

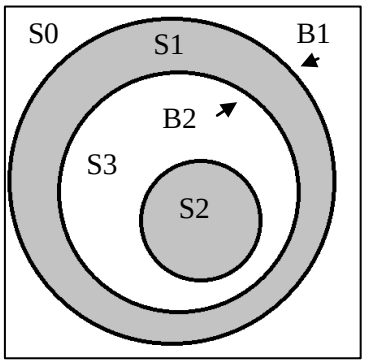
Die automatische Anpassung des Hintergrundmodells ist ein Vorteil, um Lichtveränderungen entgegenzuwirken. Allerdings darf der Lernfaktor dabei nicht zu klein eingestellt werden. Ansonsten werden die Vordergrundobjekte zu schnell in den Hintergrund übernommen, sobald die Hand ein wenig ruhig gehalten wurde. Dies hatte im Test schließlich beim Zeigen von Gesten zu schlechten Erkennungsraten geführt. Mit einer sehr kleinen Lernrate, leidet jedoch auch die Robustheit gegenüber Lichtveränderungen. Diese stören jetzt ähnlich, wie bei nicht adaptiven Verfahren. Im Gegensatz zu diesen, kumulieren sich die Fehler aber über die Zeit nicht, sondern werden nach einer längeren Dauer ohne Lichtschwankungen, in das Hintergrundmodell mit übernommen. Das hat zu der Entscheidung geführt, das MOG2-Verfahren den nicht-adaptiven Methoden vorzuziehen und für die Segmentierung zu verwenden. Das daraus resultierende Binärbild dient als eine Grundlage für die nachfolgenden Schritte, wie die Konturerkennung.

3.2.4. Konturerkennung

Konturen sind im Allgemeinen nützlich für Formanalysen oder Objektdetektion und Objekterkennung. Die Hintergrundsubtraktion liefert mit dem Binärbild der Vordergrundobjekte eine gute Basis. Mit Hilfe der kantenorientierten Verfahren sollen diese Objekte zu geschlossene Regionen transformiert werden. Dies geschieht, indem eine Kontur um einzelne Objekte gezogen wird. Daher müssen zunächst die Konturen mittels eines Kantendetektors entdeckt und anschließend per Kantenverfolgungsalgorithmus festgelegt und geschlossen werden. Daraus resultiert dann eine zusammenhängende Liste mit Punkten. Das hier verwendete Verfahren, findContours(), basiert auf dem Algorithmus von Suzuki [23]:

Die Funktionsweise wird anhand der Darstellung 18 erklärt. Dazu soll vorher noch verdeutlicht werden, wie die Begriffe outer border und hole border zu verstehen sind (Darst. 17). Ebenso sollen die Bedingungen für einen Kantenverfolgungsstartpunkt erläutert werden.

Ein outer border (B1) ist ein Rand, der vollständig von einer Pixelregion mit den Werten 0 umgeben ist und zwischen dieser und einer Pixelregion mit den Werten 1 liegt. Ein hole border (B2) ist ein Rand, der von einer 1-Pixelregion vollständig umschlossen ist und zwischen dieser und einer 0-Pixelregion liegt. S0 ist der Hintergrund (0-Pixelregion), während S1 und S2 Regionen mit Pixeln, deren Wert 1 beträgt, darstellen. S3 hingegen wird als hole bezeichnet.



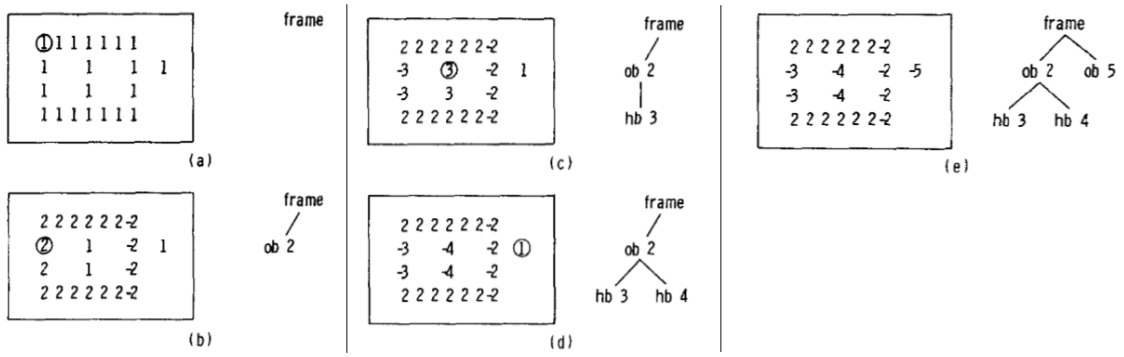
Darstellung 17: Verdeutlichung der Begriffe outer border und hole border. [23 p. 33]

Bedingungen für einen Kantenverfolgungsstartpunkt

Ein gefundenes Pixel muss entweder zu einem outer border oder zu einem hole border gehören, damit eine Kantenverfolgung gestartet werden kann. Das bedeutet konkret ausgedrückt:

- Ein Pixel (i, j) gehört zu einem outer border, wenn gilt:
  - Pixel (i, j-1) = 0 und Pixel (i, j) = 1.
- Ein Pixel (i, j) gehört zu einem hole border, wenn gilt:
  - Pixel (i, j) = 1 und Pixel (i, j+1) = 0.

Funktionsweise des Algorithmus



Darstellung 18: Beispiel der Funktionsweise des findContours()-Algorithmus. [nach:23 p. 37]

Ein binäres Bild wird Zeile für Zeile gescannt. Bei Darstellung 18 (a) findet der Scan an der Stelle ① einen Pixel der die Bedingungen eines Kantenverfolgungsstartpunktes erfüllt. Diesem Rand wird nun in Darstellung 18 (b) eine sequenzielle, einzigartige Nummer zugewiesen. Da für den frame schon die 1 vergeben wurde, ist die nächste Nummer die 2. Als nächstes wird versucht, den parent border des gerade neu gefunden Rands zu bestimmen. Da beim scannen die eindeutige Nummer des zuletzt gefundenen Rands gespeichert wird, kann überprüft werden, ob dieser der parent border ist, oder einer, der denselben Elternteil teilt, wie der gerade gefundene Rand. Da bisher jedoch noch kein Rand existiert, ist der zuletzt gefundene Rand und der aktuelle Rand vom selben Typ, nämlich ein outer border. Aus der Darstellung 19 kann nun entnommen werden, welchen Elternteil der neu gefundene Rand besitzt.

Type of the border B' with the sequential number LNBD / Type of B	Outer border	Hole border
Outer border	The parent border of the border B'	The border B'
Hole border	The border B'	The parent border of the border B'

Darstellung 19: Entscheidungstabelle für den übergeordneten Rand. [23 p. 36]

In diesem Fall ist B und B' ein outer border, sodass der Elternteil von B' auch der Elternteil von B sein muss. Der parent border ist dementsprechend der frame. Anschlie-

ßend wird dem Rand gefolgt und dieser mit Pixel markiert. Die Werte der Pixel ändern sich entweder auf 2 oder -2. Die Regel hierfür lautet:

- Ist der aktuell befolgte Rand zwischen einer 0-Pixelregion, welche die Pixel (p,q+1) enthalten, und einer 1-Pixelregion, welche die Pixel (p,q) enthalten, so ändere den Wert der Pixel (p,q) auf negativ.
- Ansonsten, setze die Werte der Pixel (p, q) auf positiv, außer (p, q) befinden sich auf einen bereits gefolgten Rand.

Anschließend setzt der Algorithmus das Scannen an der Stelle fort, an der er vorher aufgehört hat. Daher geht es mit der Kantenverfolgung weiter bei ②. Die Schritte werden nun solange ausgeführt, bis die rechte untere Ecke des Bildes erreicht wird. Als Ergebnis sind nun alle gefundenen Ränder markiert und die Topologien in einer Baumstruktur abgebildet.

Der Vorteil des Algorithmus liegt darin, dass die Topologien der Ränder mit abgespeichert werden. Dies ermöglicht es, zum Beispiel, nur die outer borders oder nur die hole borders vom Algorithmus liefern zu lassen. Im Falle der Handerkennung wird nur der äußere Rand geliefert.

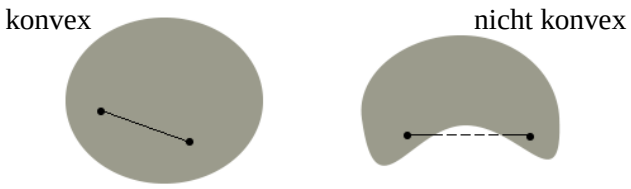
Um in diesem Stadium der Handerkennung die Komplexität für die weiteren Verarbeitungsschritte zu verringern, ist es möglich, anhand der gefundenen Konturen einige nicht relevante Objekte zu eliminieren. Dazu wird mit der Funktion contourArea() die Größe der Kontur berechnet. Es wird angenommen, dass die Hand in den meisten Fällen näher an der Kamera ist, als andere Objekte. Daher bleibt nach dem Entfernen aller Konturen, bis auf die Größte, nur die Kontur der Hand übrig. Die Kontur wird als Nächstes dazu verwendet, die konvexe Hülle zu berechnen.

3.2.5. Berechnung der konvexen Hülle und konkaven Einbuchtungen

Die Symbole Schere, Stein und Papier werden hauptsächlich durch das Zeigen einer bestimmten Anzahl von Fingern geformt. Daher ist es notwendig, diese an der Hand zu lokalisieren. Die hier verwendete Möglichkeit besteht darin, die konvexe Hülle der Hand zu berechnen. Anschließend kann anhand der konkaven Einbuchtungen, die von

den gespreizten Fingern erzeugt wurden, grob die Position und die Anzahl der Finger bestimmt werden.

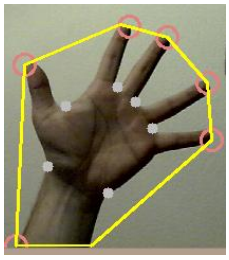
Der Algorithmus zur Konturfindung gibt, wie bereits in Abschnitt 3.2.4 erwähnt, eine Liste mit Punkten zurück. Aus dieser Menge von Punkten kann die konvexe Hülle berechnet werden. In der Mathematik ist eine Menge von Punkten konvex, wenn die Strecke, zwischen zwei beliebigen Punkten der Menge, ganz in der Menge liegt (Darst. 20).



Darstellung 20: Konvexität

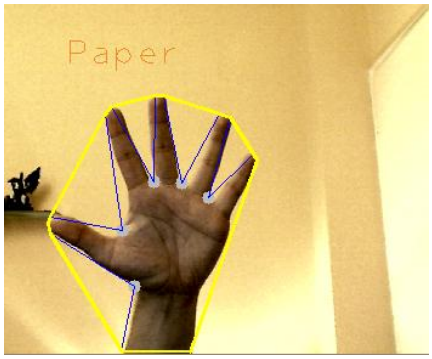
Einfach ausgedrückt bedeutet dies, dass nur die äußersten Punkte, einer Menge von Punkten, mit einer geraden Linie verbunden werden müssen, um eine konvexe Menge zu erhalten. Stellt man sich Nägel auf einem Brett vor, so reicht es, wenn ein Band um die Nägel gespannt wird, um die konvexe Hülle zu erhalten. OpenCV stellt für diese Berechnung den Algorithmus von Sklansky [24] bereit. Dieser liefert die äußersten Punkte der Handkontur, die mit einer Linie verbunden werden können.

Mit der konvexen Hülle ist es nun möglich, die konkaven Einbuchtungen zu berechnen. Konkave Einbuchtungen sind "Einschnitte" in der konvexe Hülle. Sie unterbrechen damit die Verbindung zweier Punkte innerhalb der konvexen Menge. Stellt man sich einen Ballon vor, ist dessen glatte und relativ runde Oberfläche konvex. Drückt man jedoch mit dem Finger darauf, entsteht an dieser Stelle eine "Delle", die konkave Einbuchtung. An der Darstellung 21 lässt sich die konvexe Hülle der Hand sehr gut erkennen.



Darstellung 21: Die konvexe Hülle der Hand. Hier dargestellt als gelbe Linie.

Die Zwischenräume der Finger stellen einen "Einschnitt" dar, wodurch eine konkave Einbuchtung (Darst. 22) entsteht. Der depth point (Darst. 22) ist der am Weitesten von der konvexen Hülle entfernte Punkt innerhalb der Einbuchtung. Er wird in dieser Arbeit für die Autoselektion der Hautfarbe und für die Bestimmung der Anzahl der Finger verwendet.

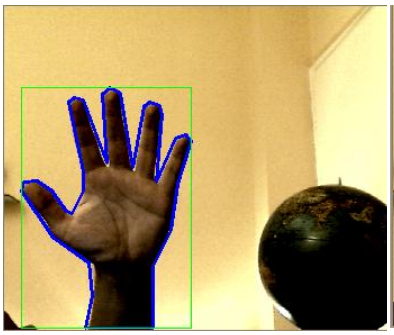


Darstellung 22: Konkave Einbuchtungen (blaue Linien) und deren depth point.

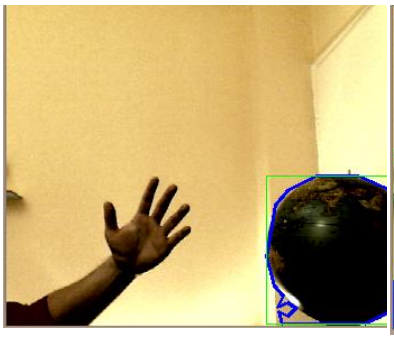
3.2.6. Region of Interest

Wie die Abschnitte 3.2.3 und 3.2.4 zur Background Subtraction und Konturenfindung gezeigt haben, werden alle sich bewegende Objekte erkannt und deren Kontur ermittelt.

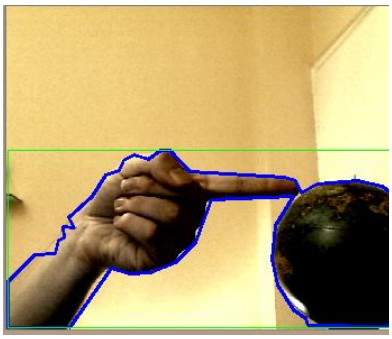
Anschließend wird die größte Kontur herausgefiltert. Ist das größte Objekt die Hand, so wird sie richtig erkannt (Darst. 23). Sind jedoch noch andere, größere Objekte oder Personen im Bild, so werden diese anstatt der Hand erkannt (Darst. 24). Es kommt auch zur falschen Konturbildung, wenn zwar die Hand soweit richtig erkannt wird, diese aber mit einem Objekt in Berührung kommt. Dadurch "verschmelzen" beide zu einem großen Objekt, und die daraus entstehende Kontur ist nicht mehr für die Gestenerkennung verwendbar (Darst. 25). Die Verschmelzung passiert nicht, weil die Kugel selbst eine Kontur besitzt, sondern weil die Hand und die Kugel als ein Objekt aufgefasst wird.



Darstellung 23:
Die Hand ist das größte Objekt im Bild.



Darstellung 24:
Die Weltkugel ist das größte Objekt im Bild.



Darstellung 25:
Verschmelzen der Konturen zweier Objekte.

Um diese Fehler möglichst zu vermeiden, und auch sonstige Störungen im Bild weitestgehend zu eliminieren, wird um die Hand eine Region of Interest (ROI) gelegt. Eine ROI stellt einen Ausschnitt aus dem Gesamtbild dar und kann mit einer Schablone verglichen werden. So werden alle Bereiche von der Schablone verdeckt, bis auf den "ausgeschnittenen" Bereich. Die Schablone, oder auch Maske, wird erstellt, indem auf einem schwarzen Bild eine weiße, geometrische Form (hier eine Ellipse) gezeichnet wird. Anschließend ist dieses Binärbild, bitweise mit dem UND-Operator, mit dem Binärbild der Hintergrundsubtraktion, logisch zu verknüpfen. Folglich bleiben nur die Pixel sichtbar, die den Wert 1 besitzen.

Damit ist aber nur eine statische Maske erzeugt worden. Da bei einer Gestenerkennung für einen Roboter nicht darauf geachtet werden kann, wohin die Hand genau gehalten wird, ist die ROI so noch nicht einsetzbar. Daher ist der nächste Schritt, die Maske der Hand folgen zu lassen. Dafür wird der Tracking-Algorithmus Camshift (3.2.7) eingesetzt. Weiterhin hat sich die Maske an die Handgröße anzupassen. Denn je näher die Hand an die Kamera gehalten wird, umso größer ist sie, und je weiter weg, umso kleiner. Dafür ist ebenfalls der Camshift-Algorithmus dienlich. Beim Verfolgen der Hand legt der Algorithmus ein Rechteck um diese, das bei einer Größenänderung oder Rotation automatisch angepasst wird. Erzeugt man die Kreismaske nun auf dem Mittelpunkt des Rechtecks und macht den Umfang des Kreises von den Maßen des Rechtecks abhängig, so verschiebt sich nicht nur die Schablone mit der Bewegung der Hand, sondern passt sich auch automatisch der Größe an. Allerdings muss dann noch verhindert werden, dass die Maske nicht außerhalb des Bildes "gezeichnet" wird. Sonst wird nicht mehr in die Bildmatrix geschrieben, sondern in einen anderen Speicherbereich, was zu einem Fehler führt. Dies kann leicht verhindert werden, indem eine Abfrage stattfindet, die

prüft, ob der x- und y-Wert der oberen linken und der unteren rechten Ecke nicht die minimale und maximale Bildgröße überschreitet.

Die Region of Interest (Darst. 26) ermöglicht somit, dass sich bewegte Objekte im Kamerabild befinden dürfen, solange diese nicht die Hand "berühren" oder sich direkt hinter ihr befinden. Auch Rauschen (Darst. 27), dass durch Lichtveränderungen an einigen Stellen im Bild auftauchen kann, wird so ignoriert.

Darstellung 26:
Maskierte Hand mittels runder ROI.

Darstellung 27:
Rauschen durch Lichtveränderungen, dass ohne ROI zu sehen ist.

3.2.7. Camshift

Der Camshift[8]-Algorithmus [25] ist ein Verfahren zur Verfolgung eines Objekts in einer Videosequenz. Er basiert auf dem Mean-Shift-Algorithmus, bietet aber einige Verbesserungen gegenüber dem Original. Die Funktionsweise des Mean-Shift-Algorithmus soll zuvor dennoch an einem Beispiel kurz erklärt werden:

Gegeben sei eine Pixelmenge, sowie ein kleines Fenster (Ellipse, Rechteck, etc.). Das Ziel des Algorithmus ist es nun, das Fenster an die Stelle zu schieben, an der die Pixeldichte am Höchsten ist. Das Fenster startet bei einer initialen Position und der Mittelpunkt des Fensters ist bekannt. Anschließend wird der Mittelwert der Pixel berechnet, die innerhalb des Fensters liegen.. Werden nun die Positionen des Mittelpunkts und des Mittelwerts verglichen, so wird ziemlich sicher festgestellt, dass sie nicht übereinstimmen. Das Fenster wird dann an die Position des Mittelwerts der Pixel verschoben und die Berechnung findet erneut statt. Die Positionen werden, aufgrund neuer Pixel wieder

[8] Continously Adaptive Mean Shift

nicht übereinstimmen und der Algorithmus verschiebt das Fenster wiederum auf die Position des Mittelwerts. Dies wird solange wiederholt, bis der Mittelpunkt des Fensters mit der Position des Mittelwerts der Pixel konvergiert. Sobald dies der Fall ist, ist das Fenster an der Stelle, an der die Pixeldichte am höchsten ist.

So funktioniert also auch die Verfolgung der Hand. Es wird manuell die Pixeldichte (per Auswahl mit der Maus) festgelegt, die verfolgt werden soll. Bewegt sich diese Pixeldichte nun, verfolgt der Algorithmus sie in der Art, wie in dem Beispiel oben beschrieben.

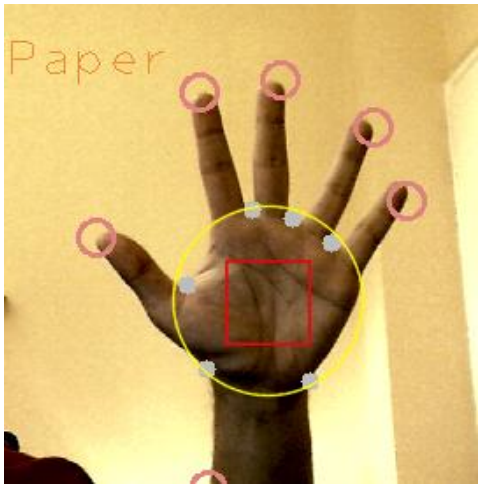
Der Mean-Shift-Algorithmus hat allerdings einen entscheidenden Nachteil. Die Größe und Ausrichtung des Fensters ist fix! Wird das zu verfolgende Objekt größer, so ist die Lokalisierung schlechter, da nicht mehr alle zugehörigen Pixel des Objekts im Suchfenster des Algorithmus sind. Wird das Objekt hingegen kleiner, so kann es sein, dass, aufgrund von Rauschen oder eines überladenen Hintergrundes, die Verfolgung fehlschlägt. An diesem Punkt setzt der Camshift-Algorithmus an. Erst wird der Mean-Shift-Algorithmus angewendet und anschließend die Größe und die Rotation des Suchfensters angepasst.

Für die in dieser Arbeit entwickelte Gestenerkennung dient der Camshift-Algorithmus vor allem dazu, die Region of Interest, sowohl an die Position, als auch an die Größe der Hand anzupassen. Ein Problem, das gelöst werden musste, ist die Selektion der Pixelverteilung. Normalerweise würde die Auswahl manuell stattfinden, zum Beispiel mit der Maus. Bei NAO ist dies jedoch, wegen des Fehlens des Displays und entsprechender Eingabemöglichkeiten, nicht möglich. Daher wurde als "workaround" eine Autoselektion entwickelt, deren Funktionsweise nachfolgend erläutert werden soll.

3.2.8. Autoselektion der Pixelverteilung für den Camshift-Algorithmus

Da eine Hand durch den Camshift-Algorithmus getrackt werden soll, muss ein Teil dieser markiert werden, um daraus eine Pixelverteilung zu erhalten. Somit stellte sich die Frage, wie denn eine Auswahl automatisch auf eine Handfläche positioniert werden könnte. Die Lösung dazu sind die konkaven Einbuchtungen. Davon ausgehend, dass die Einbuchtungen zwischen den gespreizten Fingern und am Handgelenk gebildet werden, kann dies ausgenutzt werden, um in die Mitte ein Auswahlrechteck zu setzen. Dazu

wird ein minimal umschließender Kreis (Darst. 28) der Einbuchtungen berechnet. Anschließend kann an dessen Mittelpunkt ein Rechteck (Darst. 28) platziert werden, das zur Selektion der Pixel dient. Hieraus ergibt sich auch die Randbedingung (3.1.1), dass der Spieler erst seine Hand mit gespreizten Fingern dem Roboter zeigen muss, bevor er das Spiel (und die Handerkennung) startet.



Darstellung 28: Automatische Selektion einer Pixelregion.

3.2.9. Fingererkennung

Zur Erkennung der "Schere, Stein, Papier" - Gesten, ist es ausreichend die gezeigten Finger zu bestimmen. Für das Stein-Symbol dürfen keine Finger, für das Scheren-Symbol müssen zwei Finger, und für das Blatt-Symbol müssen fünf Finger gezählt werden. Zur Bestimmung der Anzahl der Finger kann sich an den konkaven Einbuchtungen (3.2.5) orientiert werden. Bei einer gespreizten Hand sind mindestens vier konkave Einbuchtungen vorhanden. Allerdings existieren oftmals auch noch weitere Einbuchtungen, zum Beispiel am Handgelenk. Daher muss anhand eines Merkmals entschieden werden, welche Einbuchtung tatsächlich zu einem Finger gehört.

Bei der Betrachtung der Hand fällt auf, dass der angenommene Winkel zwischen zwei Fingern nicht mehr als 90° beträgt. Deshalb wird dieser Winkel als Kriterium verwendet, um die konkaven Einbuchtungen der Finger zu definieren.

Aus den Berechnungen der Konvexität stehen neben den depth points, auch die Start- und Endpunkte des Einschnittes zur Verfügung. Anhand dieser kann der Winkel des Fingerzwischenraums berechnet werden.

Dazu werden zu erst die Distanzen zwischen des Startpunkts und des Tiefenpunkts, sowie des Tiefenpunkts und des Endpunkts berechnet, wofür die Standardformel angewandt wird.

Die Distanz zwischen zwei Punkten $P_1(x_1, y_1)\ und\ P_2(x_2, y_2)$ ist definiert als:

$$Distanz\ d \coloneqq \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Mit den Distanzen lassen sich nun das Skalarprodukt (in der euklidischen Ebene mit kartesischen Koordinaten) und anschließend der Winkel berechnen. Das Skalarprodukt ist durch folgende Formel ermittelbar:

$$\vec{a} = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} und\ \vec{b} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

$$\vec{a} * \vec{b} = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} * \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = a_1 b_1 + a_2 b_2$$

Der Winkel ist nun mittels des Skalarprodukts und den Distanzen Startpunkt - depth point d_a sowie depth point - Endpunkt d_b mit folgender Formel berechenbar:

$$\text{Winkel} = \frac{\left(\cos^{-1} \frac{Skalarprodukt}{d_a * d_b}\right) * 180}{\pi}$$

Im Allgemeinen ist die Anzahl der Einbuchtungen um eins weniger, als Finger gezeigt werden. Erst ab zwei Fingern entsteht die erste konkave Einbuchtung. Bei fünf Fingern hingegen existieren nur vier davon. Daher muss hier ein kleiner Trick aushelfen. Die Einbuchtungen, die sich unterhalb der 90°-Grenze befinden, werden in ein eigenes Array abgespeichert. Auf die Anzahl der darin gespeicherten Punkte wird immer der Wert eins addiert. So ergibt eine Einbuchtung tatsächlich zwei Finger und vier Einbuchtungen fünf Finger. Wird jedoch keine Einbuchtung, mit einem Winkel unter 90° gefunden, dann kann die Fingeranzahl direkt als 0 ausgegeben werden.

Kapitel 4

Evaluation

4.1. Nutzerstudie

Die Nutzerstudie zu dieser Arbeit wurde unter Laborbedingungen durchgeführt. Dazu wurden zehn Testpersonen eingeladen, die einzeln mit dem Roboter NAO Schere, Stein, Papier spielen sollten. Für die Evaluation ist ein Fragebogen ausgeteilt worden, der in vier Kapitel unterteilt ist. Alle Personen wurden mündlich befragt, ob sie mit dem Filmen des Versuchs einverstanden sind. Die vorhandenen Videos (siehe CD) sind nur von Teilnehmern, die ihr Einverständnis dazugegeben haben. Die Strategie, die der Roboter in einem Durchlauf verfolgt, wurde bereits im Voraus für jede einzelne Person festgelegt. Dazu wurde die Reihenfolge durch Permutation zufällig festgelegt. Die Benutzer hatten zu keinem Zeitpunkt Kenntnis davon, welche Strategie gerade gespielt wird. Um die Strategien bewerten zu können, sind Kärtchen mit Symbolen (♠♣♥♦) ausgegeben worden, die die Strategien und deren Reihenfolge repräsentieren. Die Strategie Pattern Analysis konnte aufgrund von einer zu geringen Anzahl von Spielrunden ihr volles Potenzial nicht komplett entfalten. Deshalb ist eine korrekte Bewertung nur schwer möglich gewesen, was beim Lesen der Evaluation bedacht werden muss.

Ablauf der Nutzerstudie

Alle Teilnehmer haben vorab per E-Mail ein Dokument zur Einweisung zugesandt bekommen. Dieses war zusätzlich noch einmal im Labor ausgelegt. Nach einer kurzen, mündlichen Einführung in die Steuerung des Roboters, wurden ein bis zwei Trainingsspiele durchgeführt. Anschließend sind insgesamt vier[9] Durchläufe mit je drei Spielen SSP gespielt worden. Nach jedem Durchlauf wurden die Teilnehmer dazu aufgefordert, in Abhängigkeit des aktuellen Symbols, die Fragen des ersten Teils des Fragebogens zu

[9] Die geringe Anzahl an Durchläufe und Spiele wurde im Vorfeld der Studie abgesprochen und genehmigt. Dies lag in erster Linie an der begrenzten Bearbeitungszeit der Masterarbeit.

beantworten. Nach dem letzten Durchlauf sind die Testpersonen darauf hingewiesen worden, die Teile zwei bis vier des Fragebogens zu beantworten. Im Anschluss daran ist es den Probanden freigestellt worden, eigene Kommentare zu dem Versuch zu geben.

Aufbau des Fragebogens

Der Fragebogen ist in vier Kapitel untergegliedert**.** Kapitel eins versucht die Strategien nach verschiedenen Kriterien zu bewerten. Diese Kriterien sind nachfolgend aufgelistet:

- Bewertung der Strategien nach der Natürlichkeit im Vergleich zu einem Menschen.
- Bewertung der Strategien anhand des Unterschieds, wie im Vergleich zum Roboter ein menschlicher Gegenspieler agieren würde.
- Bewertung der Strategien nach dem Spaßfaktor.
- Bewertung der Strategien nach der "Erkennbarkeit", also welche Strategie gerade gespielt wurde.

Kapitel zwei behandelt das Thema Interaktion. Hier sollte festgestellt werden, wie intuitiv und natürlich die Teilnehmer die Steuerung des Roboters per Geste und Sprache empfanden. Ob sie diese Variante erneut wählen, oder eine andere Eingabemethode bevorzugen würden. Wie die Bewegungsgeschwindigkeit von NAO wirkte, und auf welche Signale (Geste, Sprache) die Testpersonen geachtet haben, um die gespielten Symbole von NAO herauszufinden.

Kapitel drei ist etwas allgemeiner gehalten. Die Fragen drehen sich darum, ob die Probanden generell mit NAO ein Zweipersonenspiel in ihrer Freizeit spielen, und ob sie einen Roboter oder einen Computer zum Spielen bevorzugen würden. Außerdem sollten die Teilnehmer das Design von NAO bewerten, und sich überlegen in welche Kategorie (Spielgefährte, Arbeitsmaschine) sie einen humanoiden Roboter einordnen würden. Als letzter Punkt sollte geklärt werden, ob Kinder daran Spaß hätten mit NAO SSP zu spielen.

Das vierte Kapitel dient zur Erhebung von demografischen Daten. Konkret erhoben wurde, ob die Person Informatiker/in ist, sowie das Alter und das Geschlecht.

Aufgetretene Probleme

Bei einem Versuch kam es zu einer größeren Störung der Algorithmen durch plötzlich auftretende Lichtveränderungen. Aufgrund längerer Behebungsversuche und der daraus resultierenden Verkürzung der Zeit konnte eine Strategie nicht bewertet werden.

Weiterhin vergaß ein Teilnehmer den zweiten bis vierten Teil des Fragebogens auszufüllen. Daher konnten von ihm, bis auf das Geschlecht, keine demografischen Daten erhoben werden. Ebenso sind die Daten zur Interaktion und den allgemeinen Fragen nicht auswertbar gewesen.

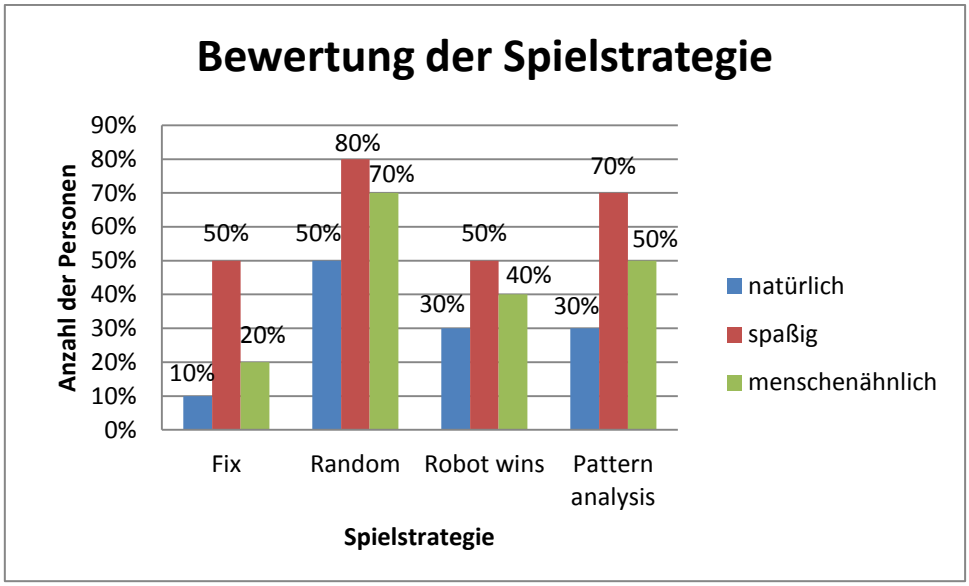
4.2. Auswertung der Nutzerstudie

Demografische Daten

Insgesamt haben zehn Personen an der Nutzerstudie teilgenommen. Davon waren sieben weiblich (70 %) und drei männlich (30 %). Von den neun[10] Personen betrug das minimale Alter 22 Jahre und das maximale Alter 46 Jahre. Das Durchschnittsalter lag bei 30,33 Jahren. Informatiker waren mit 78 % (7 Personen) überdurchschnittlich viel vertreten. Die restlichen 22 % (2 Personen) waren aus einer anderen Fachrichtung.

[10] Siehe "Aufgetretene Probleme".

Bewertung der Spielstrategien



Darstellung 29: Bewertung der Spielstrategien.

Darstellung 29 zeigt die Bewertung der einzelnen Strategien (x-Achse) hinsichtlich ihrer Natürlichkeit, Menschenähnlichkeit und des Spaßfaktors. Auf der y-Achse ist die Anzahl der Personen in Prozent dargestellt, die die Strategien nach den jeweiligen Kriterien bewertet haben. Die Personen musste bei jeder Strategie zu jedem Kriterium eine Bewertung abgeben. Deshalb beträgt die Gesamtsumme der Personen, in Bezug auf ein Merkmal und über alle Strategien kumuliert, nicht genau 100 %.

Als Erstes soll die Natürlichkeit betrachtet werden. Wie zu erwarten, ist die Zufallsstrategie von 50 % der Personen als die natürlichste und Fix von 10 % als die unnatürlichste Strategie bewertet worden. Werden beide Strategien auf den Menschen übertragen, so machen eine feste Wahl und ein wiederholtes Spielen eines einzigen Symbols keinen Sinn. So würde sich kein Mensch verhalten! Im Gegensatz dazu meinen die meisten Menschen ihre Entscheidung zufällig zu wählen, weshalb die Zufallsstrategie am natürlichsten wirkt. Erstaunlich hoch (30 %) ist der Wert bei der Strategie, mit der der Roboter immer gewinnt. Hier kann angenommen werden, dass sechs Spielrunden noch zu wenig sind, um das ständige Verlieren als "nicht zufällig" einzustufen.

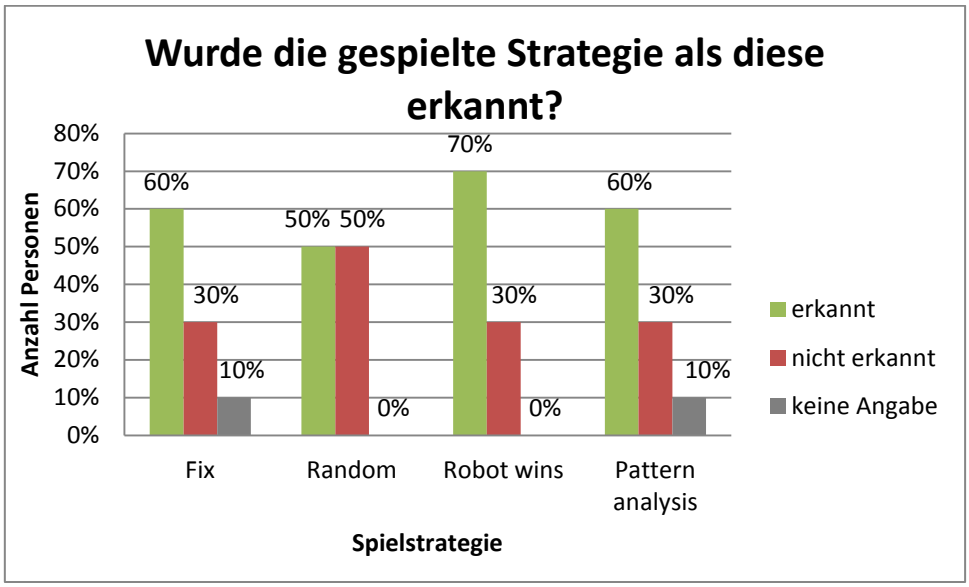
Das Kriterium Menschenähnlichkeit ist von der Rangfolge ähnlich verteilt, wie die Natürlichkeit der Strategien. So ist die Zufallsstrategie von 70 % der Teilnehmer

als menschenähnlich bewertet worden. Die fixe Strategie hingegen ist nur von 20 % als menschenähnlich befunden worden. Dies macht auch Sinn, denn je menschenähnlicher ein Verhalten ist, umso natürlich wirkt es auf die Menschen.[11]

Auch vom Spaßfaktor her ist die Zufallsstrategie von den meisten Probanden (80 %) am Besten bewertet worden. Das Pattern Analysis mit 70 % fast genauso viel Spaß gemacht hat, dürfte auf die bereits erwähnte zu geringe Rundenanzahl zurückzuführen sein. Wie in Kapitel (3.1.3) erläutert, gewinnt die Strategie nach längerer Zeit der Benutzung immer öfter. Somit sollte sich der Spaßfaktor eigentlich eher im Bereich von Robot wins ansiedeln. Interessant ist auch, dass 50 % der Testpersonen die fixe Strategie als spaßig empfunden haben. Auch hier ist anzunehmen, dass die geringe Anzahl an Spielen einen Einfluss auf die Bewertung hatte. Auf kurzer Dauer gesehen, kann es nämlich durchaus interessant sein, dieses Verhaltensmuster systematisch zu testen oder jede Runde sicher zu gewinnen.

[11] Aufgrund der fehlenden Stimme bei der Pattern Analysis Strategie, kann diese Aussage leider nicht stichfest gemacht werden.

Erkennen der Strategien



Darstellung 30: Bewertung der Interaktion mittels Sprache und Gestik.

Das Diagramm in Darstellung 30 zeigt, welche tatsächlich gespielte Strategie auch als diese wahrgenommen wurde. Die Mehrheit der Teilnehmer hat die Strategien richtig erkannt. Lediglich bei der Zufallsstrategie waren sich die Probanden nicht sicher, ob es zufällig ausgewählte Symbole (50 %) sind, ob eine fixe Strategie (20 %) verwendet wurde, ob die Entscheidung aufgrund vorangegangener Züge (30 %) gewählt wurde oder ob mittels einer schnelle Wahrnehmung der eigenen Wahl durch den Roboter die Entscheidung gefällt wurde. Verwunderlich ist besonders, dass bei der gespielten Zufallsstrategie gedacht wurde, es handle sich um eine fixe Strategie. Anzunehmen ist hier, dass die Probanden der Meinung waren, eine fixe Strategie sei eine festgelegte Reihenfolge von unterschiedlichen Symbolen, die entsprechend abgearbeitet wird.

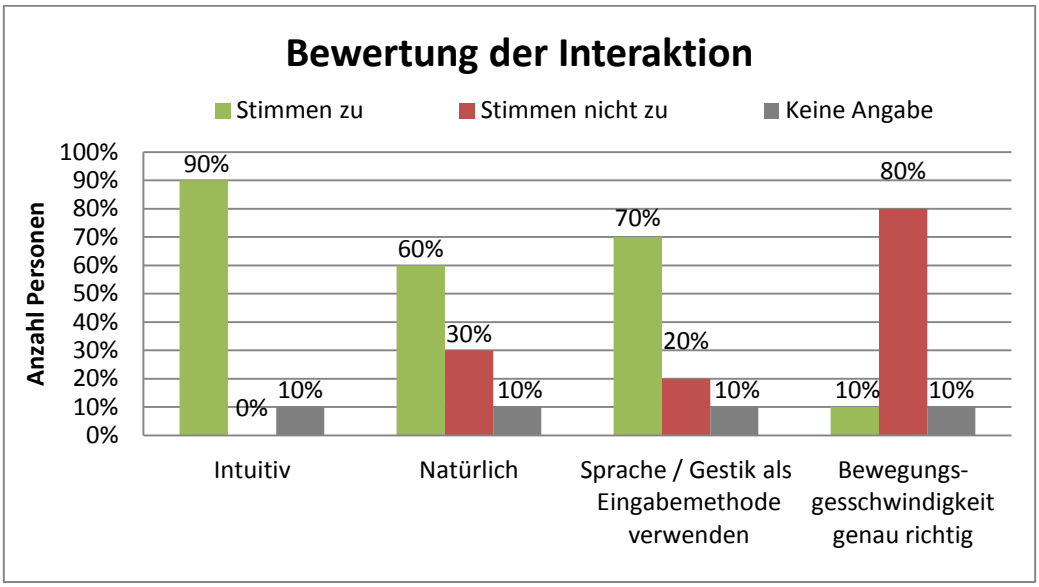
Konfusionsmatrix tatsächlich gespielter und vermuteter Strategien

		Tatsächlich gespielte Strategie			
		Fix Strategy	**Random** Strategy	**Robot** wins	**Pattern** Analysis
Vermutete Strategie	**Fix** Strategy	6	3	5	3
	Random Strategy	2	5	2	3
	Robot wins	2	3	7	2
	Pattern Analysis	3	3	5	6

Darstellung 31: Matrix der gespielten und vermuteten Strategien

Die Konfusionsmatrix zeigt im Detail, welche Strategien tatsächlich gespielt und welche von den Teilnehmern vermutet wurden. Die Werte stellen die Anzahl der Personen dar. Bei den falsch geschätzten Strategien ist die Verteilung relativ ausgeglichen. Zwei stärkere Verwechslungen gab es jedoch mit der Robot wins Strategie. Fix Strategy und Pattern Analysis wurden von jeweils 50 % der Teilnehmer vermutet, obwohl tatsächlich Robot wins gespielt wurde. Der Wert von Pattern Analysis wäre, bei einer höheren Anzahl an Spielrunden pro Person (> 15 Spielrunden), aufgrund seiner Funktionsweise nachvollziehbar. Der Wert der fixen Strategie lässt aber nur Mutmaßungen zu. Es ist anzunehmen, dass die Probanden der Meinung waren, dass der Roboter irgendeine spezielle Strategie benutzt, mit der ziemlich oft gewonnen werden kann. Eine andere Möglichkeit wäre, dass das Betrügen als feste Strategie interpretiert wurde. Also in dem Sinn, dass der Roboter nicht willkürlich ein Symbol ausgewählt hat, sondern ganz bewusst schummelt.

Bewertung der Interaktionsmöglichkeiten (Gestik und Sprache)

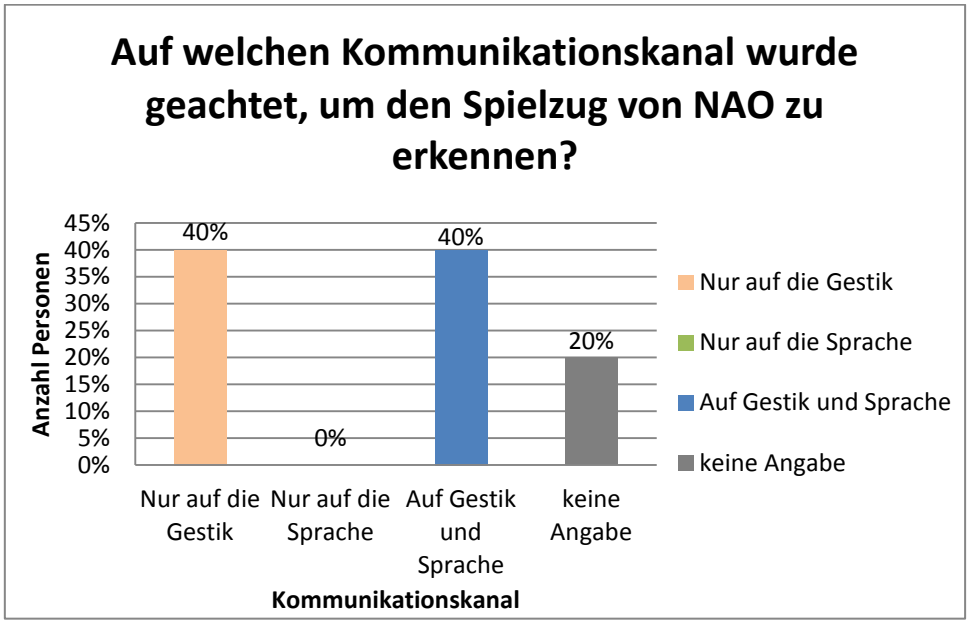


Darstellung 32: Bewertung der Interaktion mit NAO.

Damit die Gestenerkennung korrekt eingesetzt werden kann, sind einige Randbedingungen nötig. Diese wurden bereits im Abschnitt 3.1.2 erläutert. Trotz dieser vorgeschriebenen Bedingungen wird in Darstellung 32 eindeutig dargestellt, dass 90 % der Teilnehmer die Interaktion mit Sprache und Gestik als intuitiv empfunden haben. Allerdings leidet verständlicherweise die Natürlichkeit (60 %) unter den gegebenen Einschränkungen. Dennoch würden 70 % der Teilnehmer diese Art der Interaktion gegenüber einer Alternative (NAO-Marks [12], Touchscreen) bevorzugen. Die eingestellte Bewegungsgeschwindigkeit des Roboters war für 80 % zu langsam (60 % eher zu langsam / 20 % deutlich zu langsam). Nicht nur die gebremste Bewegung fühlte sich unnatürlich an, sondern es entstand auch oftmals der Eindruck, er würde vor der Entscheidung für eines der Symbole abwarten, was der Gegner zeigt.

[12] NAO-Marks sind spezielle, ausdruckbare Zeichen, die der Roboter von sich aus interpretieren kann.

Auf welchen Kommunikationskanal wurde hauptsächlich geachtet, um die Spielzüge von NAO zu erfahren?



Darstellung 33: Welcher Kommunikationskanal wurde für die Spielzugerkennung beachtet?

Das Resultat dieser Frage weicht teilweise von den Erwartungswerten ab. Es wurde angenommen, dass die Sprache einen höheren Stellenwert besitzt als die Gestik. In Darstellung 33 wird jedoch das Gegenteil verdeutlicht. Zwar wurde nicht komplett auf die Sprache verzichtet (40 %), beweist jedoch, dass die Gestik (80 %) eine wichtige Rolle in der Interaktion spielt. Dieses Ergebnis deckt sich auch mit der Erkenntnis der Studie von M. Salem et al. [20]. Dabei reagierten die Testpersonen positiver auf den Roboter, wenn dieser partiell richtige Gesten zu dem Gesagten machte, als gar keine.

Eigene Beobachtungen

An dieser Stelle sollen noch einige Beobachtungen kurz angesprochen werden, die an den Teilnehmern aufgefallen sind. Anfänglich sind alle Testpersonen verunsichert, wie sie mit dem Roboter umgehen sollen. Dazu zählte die Positionierung der Hand, und ob bzw. wie schnell sie diese bewegen dürfen. Die Unsicherheit rührte von den Randbedingungen her, auf welche im Vorfeld hingewiesen wurden. Mit der Zeit wurden die Probanden jedoch immer "sicherer", was sich an einem flüssigeren Bewegungsablauf bemerkbar machte.

Sehr interessant war auch die Tatsache, dass die Bewegungsgeschwindigkeit an die des Roboters angepasst wurde. Sobald NAO keine Bewegungen mehr ausführte, sind die menschlichen Spieler entweder bei der Anzahl der typischen "auf und ab Bewegungen" durcheinander gekommen oder haben die Bewegungen für die Runde komplett eingestellt.

Zur Positionierung der Hand orientierten sich die Testpersonen generell an NAOs Augen als Fixpunkt. Tatsächlich ist die verwendete Kamera jedoch an der Stirn verbaut und leicht schräg nach oben ausgerichtet, was den Teilnehmern auch bewusst war. Dies deutet implizit darauf hin, dass dem Roboter aufgrund seiner humanoiden Form auch menschliche Eigenschaften attestiert wurden. Das Sehen ist beim Menschen normalerweise mit den Augen assoziiert. Da NAO im Ansatz über einen humanoiden Kopf mit Augen verfügt, wurde ihm die Eigenschaft des "Sehens mit den Augen" zugeschrieben.

Gesamtfazit der Evaluation

Die Nutzerstudie hat gezeigt, dass verschiedene Verhaltensmuster in einer Situation unterschiedlich aufgefasst und deswegen positiv und negativ bewertet werden. Ein Verhalten, das nicht für die Situation angemessen ist, erhält schlechte Bewertungen. Das kann am Beispiel der fixen Strategie nachvollzogen werden. Da dieses "sinnlose" Verhalten weder menschlich, und daher für die Menschen auch nicht natürlich ist, sind diese Merkmale entsprechend negativ bewertet worden. Der Spaßfaktor macht dabei deutlich, wenngleich er durch die geringe Anzahl an Spielen höher ist, als er sein sollte, dass für ein solches Verhalten kein Verständnis vorhanden ist. Daraus resultiert eine verringerte Benutzung des Roboters, was folglich auch eine Senkung der Akzeptanz des Serviceroboters bedeutet. Dasselbe ist bei der Robot wins Strategie beobachtbar. Die Strategie Random hingegen wurde in allen Kategorien, am besten bewertet. Die hohe Menschenähnlichkeit führt dazu, dass das Verhalten als natürlich wahrgenommen wird. Ein natürliches Verhalten (herausfordernd, aber Fair Play) macht wesentlich mehr Spaß, als die berechenbare fixe Strategie und die schummelnde Robot wins Strategie. Daher kann hier davon ausgegangen werden, dass der Roboter öfters benutzt wird. Somit steigt also seine Akzeptanz. Es kann festgehalten werden, dass es wichtig ist, ein an die Situation angepasstes Verhalten vorzuweisen. Denn je nachdem, wie die Menschen das Verhalten des Roboters bewerten, steigt oder sinkt die Akzeptanz von Maschinen.

Die Interaktion mittels Sprache und Gestik wurde von den Teilnehmern insgesamt sehr gut bewertet. Trotz der Randbedingungen (3.1.1), die dafür eingehalten werden mussten, würden viele Teilnehmer diese Art der Interaktion bevorzugen. Daraus kann geschlussfolgert werden, dass die in der Robotikforschung etablierte Interaktionsmöglichkeit der Sprach- und Gestensteuerung der richtige Weg ist.

Kapitel 5

Schlussfolgerung und Ausblick

5.1. Fazit

An dieser Stelle soll noch einmal abschließend auf die Fragen eingegangen werden, die sich am Anfang der Masterarbeit gestellt haben. Es sollte, anhand der Aspekte Technik, Verhalten, Interaktionsmöglichkeiten und Akzeptanz, die Mensch-Roboter-Interaktion betrachtet werden. Die Frage, ob es generell möglich ist, auf einen technisch mittelmäßigen ausgestatteten Roboter eine Mensch-Mensch-Interaktion zu übertragen, kann mit "Ja" beantwortet werden. Die technische Umsetzung erfordert jedoch einige Randbedingungen, insbesondere für den erfolgreichen Einsatz der Gestenerkennung, die einen praktischen Einsatz nicht erlauben. Dazu müsste allerdings eine bessere Kameratechnologie in NAO eingebaut werden. Dennoch würde NAO nicht als "High End"-Roboter gelten, wenn eine Technologie, wie Kinect, eingebaut werden würde. Daher ist das "Ja" also durchaus gerechtfertigt.

Die Interaktionsmöglichkeiten mittels Sprache und Gestik sind bei den Testpersonen der Nutzerstudie gut angekommen, trotz der teilweise unnatürlichen Position die sie einnehmen mussten. In diesem Fall ist der Trend zur natürlichen und intuitiven Interaktion also der richtige Weg für die Zukunft.

In Sachen Verhalten und Akzeptanz hat sich die These als richtig erwiesen, dass ein Roboter ein an die Situation angepasstes Verhalten aufweisen muss. Die Nutzerstudie bestätigte dies mit den einzelnen Bewertungen der Strategien. So ist eine "dumme" oder eine betrügerische Verhaltensweise (in der Situation des Spielens) keine gute, und bezogen auf den Betrug eine moralisch verwerfliche, Idee. Allerdings kann in der richtigen Situation, zum Beispiel während der Kooperation mit Menschen, ein solches Verhalten durchaus sinnvoll sein. Generell kann auch gesagt werden, dass die Akzeptanz mit einem richtigen oder falschen Verhalten sinkt bzw. steigt. Dies wurde aus der Bewertung des Spaßfaktors der einzelnen Strategien gefolgert.

5.2. Ausblick

Mit dieser Masterarbeit und der darin entwickelten Gestenerkennung für NAO ist ein Grundstein, für eventuelle nachfolgende Projekte, gelegt worden. Ein interessanter Aspekt wäre, die Gestenerkennung so zu verbessern, dass ein direktes Stehen vor der Kamera möglich ist. In Verbindung mit dem EU-Projekt HUMAVIPS[13] hat Aldebaran-Robotics einen Kopf für NAO entwickelt, mit dem Stereoskopiesehen möglich ist, was die Verwendung von Background Subtraction Verfahren obsolet macht.

Ein anderes, denkbares Folgeprojekt, könnte die Gestenerkennung dazu verwenden, Gebärden zu erkennen. Aber auch die Erkennung von Gesichtsausdrücken und somit menschlichen Emotionen wäre machbar. Das könnte sich dieser Studie anschließen und untersuchen, wie verschiedene Verhaltensweisen von NAO, als Reaktion auf die menschliche Mimik, von Menschen empfunden wird.

[13] http://humavips.inrialpes.fr/ (last viewed: 25.12.13)

Anhang A

Inhalt der CD

Die beigefügte CD enthält folgende Ordnerstruktur:

- NAO
  - Gestenerkennung
- Nutzerstudie
  - Ergebnisse
  - Videos

Im Ordner Gestenerkennung befindet sich der C++ Quellcode für die entwickelte Gestenerkennung. Eine Anleitung zur Benutzung des Codes ist ebenfalls vorhanden. Im Ordner Ergebnisse sind die Fragebögen und die ausführlichen Ergebnisse der Studie zu finden. Der Ordner Videos enthält die bei der Nutzerstudie aufgenommenen Videos.

Im Wurzelverzeichnis befindet sich zusätzlich die Masterarbeit als PDF.

Literaturverzeichnis

[1] HÄGELE, Martin, Nikolaus BLÜMLEIN, and Oliver KLEINE. *Wirtschaftlichkeitsanalysen neuartiger Servicerobotik-Anwendungen und ihre Bedeutung für die Robotik-Entwicklung. BMBF*. Hannover, Stuttgart [u.a.]: Technische Informationsbibliothek u. Universitätsbibliothek, 2011.

[2] *Service Robots - IFR International Federation of Robotics* [last viewed 3 Januar 2014]. Available from: http://www.ifr.org/service-robots/.

[3] *Statistics - IFR International Federation of Robotics* [last viewed 3 Januar 2014]. Available from: http://www.ifr.org/service-robots/statistics/.

[4] MINATO, T., Y. YOSHIKAWA, T. NODA, S. IKEMOTO, H. ISHIGURO, and M. ASADA. CB2: A child robot with biomimetic body for cognitive developmental robotics. *Humanoid Robots, 2007 7th IEEE-RAS International Conference on,* 2007, pp. 557-562.

[5] LUTKEBOHLE, I., F. HEGEL, S. SCHULZ, M. HACKEL, B. WREDE, S. WACHSMUTH, and G. SAGERER. The bielefeld anthropomorphic robot head "Flobi". *Robotics and Automation (ICRA), 2010 IEEE International Conference on,* 2010, pp. 3384-3391.

[6] *VDE-Studie: Senioren pro Serviceroboter* [last viewed 3 Januar 2014]. Available from: http://www.vde.com/de/Verband/Pressecenter/Pressemeldungen/Fach-und-Wirtschaftspresse/2011/Seiten/2011-15.aspx.

[7] MORI, M., K. F. MACDORMAN, and N. KAGEKI. The Uncanny Valley [online]. *Robotics & Automation Magazine, IEEE.* 2012, **19**(2), 98-100. Available from: 10.1109/MRA.2012.2192811.

[8] DUDENVERLAG. *Interaktion* [online]. 3 Januar 2014, 12:00 [last viewed 3 Januar 2014]. Available from: http://www.duden.de/rechtschreibung/Interaktion.

[9] SHANNON, Claude E. and Warren WEAVER. *The mathematical theory of communication.* Urbana: Univ. of Illinois Pr, 1971.

[10] ISHII, H., C. RATTI, B. PIPER, Y. WANG, A. BIDERMAN, and E. BEN-JOSEPH. Bringing clay and sand into digital design - continous tangible user interfaces. *BT Technology Journal.* 2004, **22**(4), 287-299.

[11] GIULIANI, M., and A. KNOLL. Evaluating supportive and instructive robot roles in human-robot interaction. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).* 2011, **7072 LNAI**, 193-203.

[12] HINDS, Pamela J., Teresa L. ROBERTS, and Hank JONES. Whose job is it anyway? A study of human-robot interaction in a collaborative task. *Human-Computer Interaction.* 2004, **19**(1), 151-181.

[13] KIDD, C. D., and C. BREAZEAL. Effect of a robot on user perceptions. *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on,* 2004, pp. 3559-3564.

[14] ENDSLEY, Mica R. Theoretical underpinnings of situation awareness: A critical review. *Situation awareness analysis and measurement.* 2000, 3-32.

[15] EKMAN, Paul, and W. V. FRIESEN. The Repertoire of Nonverbal Behavior: Categories, Origins, Usage, and Coding. *Semiotica: Journal of the International Association for Semiotic Studies/Revue de l'Association Internationale*, **1**, 49-98.

[16] NEHANIV, C. L. Classifying types of gesture and inferring intent. *AISB'05 Convention: Social Intelligence and Interaction in Animals, Robots and Agents - Proceedings of the Symposium on Robot Companions: Hard Problems and Open Challenges in Robot-Human Interaction.* 2005, 74-81.

[17] MOGEL, Hans. *Psychologie des Kinderspiels. Von den frühesten Spielen bis zum Computerspiel Die Bedeutung des Spiels als Lebensform des Kindes, seine Funktion und Wirksamkeit für die kindliche Entwicklung.* 3., aktualisierte und erweiterte Auflage. Berlin Heidelberg: Springer Berlin Heidelberg, 2008. 9783540466444.

[18] WESSLER, Markus. *Entscheidungstheorie. Von der klassischen Spieltheorie zur Anwendung kooperativer Konzepte.* Wiesbaden: Springer Gabler, 2012. 978-3-8349-3734-6.

[19] COOK, R., G. BIRD, G. LÜNSER, S. HUCK, and C. HEYES. Automatic imitation in a strategic context: Players of rock-paper-scissors imitate opponents' gestures. *Proceedings of the Royal Society B: Biological Sciences.* 2012, **279**(1729), 780-786.

[20] SALEM, Maha, Friederike EYSSEL, Katharina ROHLFING, Stefan KOPP, and Frank JOUBLIN. Effects of gesture on the perception of psychological anthropomorphism: a case study with a humanoid robot. *Social Robotics:* Springer, 2011, pp. 31-41.

[21] *Rock-Paper-Scissors: You vs. the Computer* [last viewed 3 Januar 2014]. Available from: http://www.nytimes.com/interactive/science/rock-paper-scissors.html?_r=1&.

[22] ZIVKOVIC, Z. Improved adaptive Gaussian mixture model for background subtraction. *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on,* 2004, pp. 28-31.

[23] SUZUKI, S., and K. BE. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics and Image Processing.* 1985, **30**(1), 32-46.

[24] SKLANSKY, J. Finding the convex hull of a simple polygon. *Pattern Recognition Letters.* 1982, **1**(2), 79-83.

[25] BRADSKI, G. R. Real time face and object tracking as a component of a perceptual user interface. *Applications of Computer Vision, 1998. WACV '98. Proceedings., Fourth IEEE Workshop on,* 1998, pp. 214-219.

[26] SEYLER, A. J. Real-time recording of television frame difference areas [online]. *Proceedings of the IEEE.* 1963, **51**(3), 478-480. Available from: 10.1109/PROC.1963.1864.

[27] STAUFFER, Chris, and W.E.L. GRIMSON. Adaptive background mixture models for real-time tracking. *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on*, 1999.

Zusammenfassung der Ergebnisse

Für die Masterarbeit wurde eine Gestenerkennung auf Basis des OpenCV Frameworks entwickelt. Diese wurde anschließend erfolgreich auf den Roboter NAO von Aldebaran-Robotics installiert, und während der Nutzerstudie verwendet. Die Nutzerstudie diente dem Zweck, verschiedene Verhaltensweisen des Roboters und die Interaktion mittels Sprache und Gestik, während des Spielens von Schere, Stein, Papier, zu bewerten. Dabei hat es sich herausgestellt, dass ein an die Situation (hier: das Spiel) angepasstes Verhalten von NAO eine positive Auswirkung auf die Akzeptanz des Roboters hat. Die Interaktionsmöglichkeiten sind von den Teilnehmern der Studie durchgängig als intuitiv, wenngleich auch (leicht) unnatürlich, bewertet worden. Sie würden diese Art der Interaktion, trotz Randbedingungen, einer alternativen Eingabemethode (Touchscreen, NAO-Marks) vorziehen.

Prüfungsrechtliche Erklärung

Ich versichere, dass ich die Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe und dass die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.

_____

Nürnberg, 07. Januar 2014