



Bau eines preiswerten Tiefenbildsensors für die mobile Robotik

ausgeführt zum Zweck der Erlangung des
akademischen Grades eines Diplom-Ingenieurs

unter der Leitung von

Ao. Univ. Prof. Dipl.-Ing. Dr. techn. Markus Vincze
Dipl.-Ing. Walter Wohlkinger

E376

Institut für Automatisierungs- und
Regelungstechnik

eingereicht an der Technischen Universität Wien
Fakultät für Elektrotechnik und Informationstechnik

von

Stefan Kitzler, *BSc.*

Matr.-Nr.: 0126602

Wachtberg 8, A-3365 Allhartsberg

Wien, im März 2010

Danksagung

An dieser Stelle möchte ich mich bei all jenen bedanken, die mich während meines Studiums unterstützt haben. Ein ganz besonderer Dank gilt meinen Eltern Aloisia und Josef, die mir durch ihre Unterstützung mein Studium ermöglicht haben.

Ein spezieller Dank geht auch an alle Personen, die mir im Zuge meiner Diplomarbeit mit Rat und Tat zur Seite standen. Ich bedanke mich bei allen Mitarbeitern der Vision for Robotics Forschungsgruppe des Institutes für Automatisierungs- und Regelungstechnik. Insbesondere bei Prof. Markus Vincze und meinem Betreuer Walter Wohlking.

Kurzfassung

Der Mensch hat seine Sinne um die Umwelt wahr zu nehmen. In der Technik treten Sensoren an Stelle der Sinnesorgane. Zur Erkennung und Unterscheidung verschiedener Objekte und um sich zu Orientieren, verwendet der Mensch vorwiegend seinen Sehsinn. Autonome Roboter benötigen ähnliche Fähigkeiten zur Orientierung und zur berührungslosen Erkennung von Objekten. Eine Möglichkeit dafür stellen optische Tiefenbildsensoren dar, mit deren Hilfe es möglich ist, dreidimensionale Bilder der Umgebung aufzunehmen.

Auch wenn am Markt schon viele verschiedene Tiefenbildsensoren existieren, wird hier ein weiterer Sensor präsentiert. Der Sensor eignet sich für mobile Roboter im Homecare-Bereich wo er zu Erkennung unterschiedlicher Objekte in der unmittelbaren Umgebung dient, womit es möglich wird, diese Objekte durch einen Roboterarm zu greifen.

Der hier beschriebene Sensor verwendet das Lichtschnittverfahren zur Erzeugung von Tiefenbildern und liefert dabei auch noch farbige Texturen, mit denen sich die 3D-Daten ergänzt werden können. Zusätzlich ermöglicht der verwendete Aufbau auch noch den Einsatz des Sensors als Stereokamera. Der Sensor selbst liefert lediglich Kamerabilder und Positionsinformationen. Die Verarbeitung dieser Daten zu dreidimensionalen Tiefenbildern erfolgt an einem PC oder Notebook.

Mit einem Preis von rund € 500 ist der Sensor preiswerter als vergleichbare Systeme. Für geringe Hardwareanforderung an den Computer wird ausschließlich die FireWire-Schnittstelle verwendet, worüber nicht nur die Ansteuerung erfolgt, sondern gleichzeitig auch Datenübertragung und Energieversorgung. Für den Einsatz an mobilen Robotern weist der Sensor neben einem geringen Energiebedarf auch noch ein geringes Gewicht und eine geringe Größe auf.

Das zur 3D-Datengewinnung eingesetzte Lichtschnittverfahren arbeitet nach der Methode der Linienprojektion mit Hilfe eines Linienlasers. Die zur Aufnahme mehrerer Lichtschnitte erforderliche Schwenkbewegung wird im Sensor selbst erzeugt. Mit Hilfe einer Schwenkeinheit kann der Sensor, angetrieben durch einen RC-Servomotor, eine 90° Drehbewegung ausführen.

Zur Verringerung des Einflusses von Kamera- beziehungsweise Laserschatten werden zwei Kameras verwendet. Dieser Aufbau macht schließlich auch den Einsatz als Stereokamera möglich.

Abschließend wurden die Eigenschaften des Sensors getestet und Erweiterungen und Verbesserungsvorschläge für eine weitere Version erarbeitet.

Abstract

Humans use different senses to perceive the world. In technical applications we use sensors instead of the senses. To recognize various objects and to differentiate them among each other and to orientate themselves, humans usually use their eyes. Autonomous robots require similar skills to orientate them self and also for contactless identification of objects. One possibility to reach this, is to use optical range sensors which can capture the three-dimensional structure of the world from the sensors viewpoint.

Even though there are already different types of optical range sensors available, another new sensor is shown here. The sensor is designed for use at mobile robots in homecare. Using this sensor the robot is be able to recognize different objects in the near surrounding area, so that the robot can grasp the objects.

The sensor described here is a laser range sensor with an additional feasibility to capture coloured textures to expand the three-dimensional data. Furthermore the used construction permits also the use of the sensor as stereoscopic camera. The sensor himself provides only the camera images and position informations. The calculation of three-dimensional images is done by a PC or a notebook.

With total costs of about € 500 the sensor is cheaper than comparable systems. For slight hardware requirements for the computer the firewire interface is used exclusively. One firewire port takes over control, data transfer and power supply. To can be used on mobile robots the sensor also is low in weight, size and power consumption.

To capture three-dimensional data the laser range sensor works on the method of projecting light stripes with a linelaser. For grabbing different light stripes, which is needed to get a complete three-dimensional image, the sensor can be rotated. By the use of a pan unit, containing a RC servo, the sensor can be turned in a range of 90°.

To reduce the influence of shadow from camera or laser, two cameras are used. Due to this configuration the use as stereoscopic camera gets possible. For higher quality images monochrome cameras are used.

Finally, the properties of the sensor have been tested and extensions and suggestions for a more elaborated version have been made.

Inhaltsverzeichnis

Danksagung	i
Kurzfassung	ii
Abstract	iii
1 Einleitung	1
1.1 Der Begriff Roboter	1
1.2 Der Begriff Robotik	2
1.3 Die Anfänge der Robotik	2
1.4 Die Vorteile von Robotern	3
1.5 Die Mobile Robotik	4
1.6 Die Sensoren für mobile Roboter	5
1.7 Die Motivation	6
1.8 Das Projekt robots@home	7
1.9 Das bestehende System	8
2 Grundlagen	9
2.1 Die Kamera	9
2.1.1 Sensortypen	9
2.1.2 Monochrom- und Farbkameras	10
2.1.3 Kameramodell	13
2.1.4 Verzeichnung	14
2.1.5 Kamerakalibrierung	15
2.2 Die Triangulation	16
2.3 Das Lichtschnittverfahren	17
2.3.1 Überblick	17
2.3.2 Lichtschnittverfahren mit Punktlasern	18
2.3.3 Lichtschnittverfahren mit Linienlasern	18
2.3.4 Verwendung von Mustern und Moiré-Effekt	19
2.4 Das Stereoverfahren	20
2.4.1 Triangulation	20
2.4.2 Epipolargeometrie	22
2.4.3 Rektifizierung	22
2.4.4 Korrespondenzsuche	24
2.5 Der Vergleich der Verfahren	24
3 Der Sensor	26
3.1 Das Einsatzgebiet	26

3.2	Die Anforderungen	26
3.3	Die Schwenkeinheit	29
3.4	Der Laser- und Kameraschatten	30
3.5	Das Stereosehen mit zwei Kameras	32
3.6	Die Aufnahme einer Textur	33
3.7	Der grundlegende Aufbau	34
4	Die Komponenten	35
4.1	Die Kameras	35
4.2	Der Linienlaser	37
4.3	Die Schwenkeinheit	38
4.3.1	Die elektrischen Komponenten	38
4.3.2	Die mechanischen Komponenten	42
4.4	Die Spannungsversorgung des Lasers	44
4.5	Das Gehäuse und die Befestigung	45
4.5.1	Für den Prototyp	45
4.5.2	Für die Endversion	46
5	Die Software	47
5.1	Die Programmiersprache und das Zielsystem	47
5.2	Die Ansteuerung der Schwenkeinheit	48
5.3	Das Auslesen der Kamerabilder	49
5.4	Die Probleme mit der Schwenkeinheit	51
5.5	Die Portabilitätsschwierigkeiten	52
6	Experimente und Ergebnisse	53
6.1	Die Spannungsversorgung	53
6.2	Der Linienlaser	55
6.3	Die Schwenkeinheit	56
6.3.1	Versuchsaufbau	56
6.3.2	Versuchsablauf und Auswertung	57
6.3.3	Ergebnisse	59
6.3.4	Fazit	63
7	Erkenntnisse und Verbesserungen	64
7.1	Die Spannungsversorgung	64
7.2	Die Schwenkeinheit	64
7.3	Die Kameras	66
7.4	Das Gehäuse und der Rahmen	67
8	Resümee	69
	Literaturverzeichnis	70

Abbildungsverzeichnis

1.1	Die ersten Roboter	3
1.2	Beispiele für den Einsatz von Robotern	4
1.3	Beispiele für mobile Roboter	5
1.4	Der bestehende Tiefenbildsensor	8
2.1	Schematische Darstellung zweier Typen von Photosensoren	10
2.2	Die vier Möglichkeiten der Farbfilteranordnung beim Bayer-Filter	11
2.3	Umwandlung eines RAW-Bildes in ein Farbbild	12
2.4	Schematische Darstellung einer Lochkamera	13
2.5	Radiale Verzeichnung	14
2.6	Tangentiale Verzeichnung	15
2.7	Das Triangulations-Prinzip	17
2.8	Funktionsprinzip des Lichtschnittverfahrens	19
2.9	Triangulation beim Stereoverfahren	21
2.10	Zusammenhang von Tiefenauflösung und Entfernung	21
2.11	Begriffe der Epipolargeometrie	22
2.12	Ziel der Rektifizierung	23
2.13	Korrektur der Bilder beim Stereoverfahren	23
3.1	Die geplante Befestigungsposition des Sensor am Roboterarm	28
3.2	Sichtbereich beim Schwenken des Laser	30
3.3	Laser- und Kameraschatten	31
3.4	Die Auswirkung von Schatten auf die Erkennung von Objekten	32
3.5	Möglichkeiten zur Verringerung des Schattens	32
3.6	Schematische Darstellung des Aufbaus	34
4.1	Unibrain Fire-i OEM Kameramodul	36
4.2	Der Linienlaser: Picotronic LFL650-5-12(9x20)90	38
4.3	Verschiedene Drehwinkelsensoren	39
4.4	Das Funktionsprinzip eines Schrittmotors	40
4.5	RC-Servo mit USB-Steuermodul	41
4.6	Das Lager für die Schwenkeinheit	42
4.7	Unterschiedliche Arten von Hebel zur Kraftübertragung beim RC-Servo	43
4.8	Die Schwenkeinheit	43
4.9	Die Schaltung zur Spannungsversorgung des Lasers	44
4.10	Befestigungselemente	45
4.11	Prototyp und Endversion des Sensors	46
5.1	Methoden der C++-Klasse ServoControl	49

5.2	Methoden der C++-Klasse DC1394_Grabber	50
6.1	Position des FireWire-Anschlusses am Sensor:	53
6.2	Die Schaltung zur Spannungsversorgung des Lasers	54
6.3	Vergleich zweier Linienlaser bei unterschiedlichen Lichtverhältnissen	56
6.4	Versuchsaufbau zum Test der Schwenkeinheit	57
6.5	Ermittlung der Pixelabweichung	59
6.6	Die beiden getestete Servos mit der Steuerung	60
6.7	Soll-Ist-Vergleich bei der Schwenkeinheit	61
6.8	Positionsabweichung der Schwenkeinheit vom Sollwert	62
6.9	Positionsabweichung der Schwenkeinheit vom genäherten, linearen Verlauf . .	62
7.1	Schwenkeinheit ohne zusätzlicher Lagerung	66
7.2	Unterschiedliche Ausführungen der Fire-i™ Kamera	67
7.3	Rahmen und Gehäuse ohne Berücksichtigung von Verbesserungen	68

Tabellenverzeichnis

2.1	Liste von Farbmodellen	11
2.2	Vergleich der benötigten Komponenten	24
2.3	Vergleich zwischen Lichtschnitt- und Stereoverfahren	25
4.1	Auswahl von in Frage kommenden Kameras	36
4.2	Liste mit in Betracht gezogenen Linienlasern	37
4.3	Liste mit in Betracht gezogenen RC-Servo-Steuerungen	41
6.1	Vergleichsdaten der beiden Laser-Modelle	55
6.2	Technische Daten der beiden verwendeten RC-Servos	60

1

1 Einleitung

Das Ziel dieser Diplomarbeit ist die Entwicklung eines Tiefenbildsensors für Forschungs- und Entwicklungszwecke im Bereich der mobilen Robotik. Dieses einleitende Kapitel beschäftigt sich mit der Definition, der Geschichte und den Vorteilen von Robotern beziehungsweise der Robotik. Eine besondere Bedeutung kommt dabei dem Gebiet der mobilen Robotik und der dabei verwendeten Sensoren zu. Die Beschreibung eines laufenden Forschungsprojekts aus dem Bereich der mobilen Robotik und eines bereits bestehenden Tiefenbildsensors führt schließlich zur Motivation dieser Arbeit.

1.1 Der Begriff Roboter

Der Begriff Roboter stammt aus dem Theaterstück „*R.U.R. – Rossumovi Univerzální Roboti*“ vom tschechischen Schriftsteller Karel Čapek [1], welches 1921 erstmals aufgeführt wurde. *Roboter* leitet sich aus dem slawischen Wort *robotá* ab, was im Deutschen *Arbeit* bedeutet. In *R.U.R.* sind die Roboter menschenähnliche Maschinen, die den Menschen die Arbeit abnehmen.¹

Heute wird der Begriff Roboter als Überbegriff für eine Vielzahl von Maschinen verwendet. Eine exakte, technische Definition des Begriffs zur Abgrenzung von anderen, automatisierten Maschinen ist schwierig. In seinem Vorwort zu „*Springer Handbook of Robotics*“ schreibt Bernard Roth, Professor für Maschinenbau an der Stanford Universität, dass als Roboter meist solche Maschinen bezeichnet werden, die eine zuvor vom Menschen auszuführende Tätigkeit übernehmen. Nach einiger Zeit, wenn es bereits normal ist, dass Maschinen diese Tätigkeiten erledigen, werden die Roboter allerdings wieder zu Maschinen degradiert [2].

Trotz aller Schwierigkeiten bei der Definition des Begriffs Roboter wurden von einigen Organisationen Definitionen erstellt. Der VDI – Verein Deutscher Ingenieure – verwendet beispielsweise die folgende Definition [3]:

Industrieroboter sind universell einsetzbare Bewegungsautomaten mit mehreren Achsen, deren Bewegungen hinsichtlich Bewegungsfolge und Wegen bzw. Winkeln frei (d.h. ohne mechanischen Eingriff) programmierbar und gegebenenfalls sensorgeführt sind. Sie sind mit Greifern, Werkzeugen oder anderen Fertigungsmitteln ausrüstbar und können Handhabungs- und/oder Fertigungsaufgaben ausführen.

¹ Solche Maschinen würden heutzutage wegen der Ähnlichkeit zum Menschen als humanoide Roboter klassifiziert werden.

Auf eine ähnliche Definition des Begriffs hat sich die RIA – Robotic Industries Association – festgelegt [4]:

Roboter, Industrieroboter: automatisch gesteuerter, frei programmierbarer Mehrzweck-Manipulator, der in drei oder mehr Achsen programmierbar ist und zur Verwendung in der Automatisierungstechnik entweder an einem festen Ort oder beweglich angeordnet sein kann.

Beide Definitionen sind allerdings auf Industrieroboter ausgerichtet und beschreiben moderne, autonom arbeitende Roboter unzureichend. In dieser Arbeit werden aus diesem Grund als Roboter mobile, meist autonom agierende Maschinen bezeichnet.

1.2 Der Begriff Robotik

Der Begriff Robotik wurde erstmals 1942 von Isaac Asimov in dessen Kurzgeschichte „*Run-around*“ verwendet [5]. Und das schon mit der heute verwendeten Bedeutung des Begriffs, wonach Robotik jene Wissenschaft ist, die sich mit der Entwicklung von Robotern befasst. Sie umfasst den mechanischen und elektronischen Aufbau der Roboter sowie deren Steuerung und Programmierung. Die Robotik umfasst somit Teilgebiete aus dem Maschinenbau, der Elektronik und der Informatik.

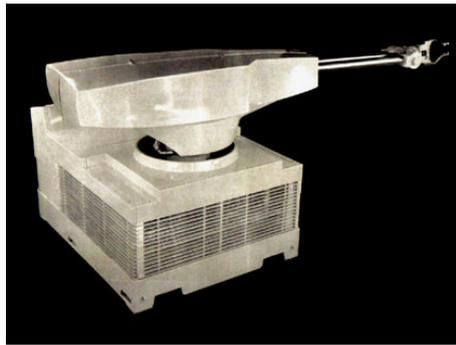
Eine umfangreichere Beschreibung des Begriffes Robotik findet sich im Duden – Wörterbuch der New Economy [6]:

Die Robotik ist eine sich rasant entwickelnde wissenschaftliche Disziplin, die Künstliche Intelligenz, Computerwissenschaften, Mechanik, Nanotechnologie, Psychologie, Anatomie, Design, Biologie, Neurophysiologie, Linguistik und viele weitere wissenschaftliche Felder involviert. Ihre interdisziplinäre Forschung will Maschinen konstruieren, die intelligente, lernfähige, mechanische Einheiten mit lebensnahem Verhalten vereinen. Die Roboter sollen den Menschen sowohl in der alltäglichen Umgebung als Haushaltshilfe oder am Arbeitsplatz unterstützen als auch als nanotechnologisch konstruierter Kleinstroboter körperliche Disfunktionen (Chipchirurgie) ausgleichen.

1.3 Die Anfänge der Robotik

Die Geschichte der Robotik beginnt mit der Entwicklung des ersten Industrieroboters. Am 10. Dezember 1954 reichte der amerikanische Erfinder George Charles Devol, Jr. Pläne einer Maschine zur „*programmierten Handhabung von Gegenständen*“ zum Patent ein. Am 13. Juni 1961 wurde ihm schließlich das Patent auf seine Erfindung erteilt [7].

Im Jahre 1956 gründete Devol mit dem Techniker und Unternehmer Joseph F. Engelberger die Firma Unimation. Gemeinsam konnten sie 1959 einen funktionstüchtigen Prototypen fertigstellen. Im Jahre 1961 wurde der erste Industrieroboter unter dem Namen *Unimate* (Abbildung 1.1a) bei General Motors zum Transport von Gussteilen eingesetzt.



Quelle: Wikipedia

(a) Unimate



© Phil Hollenback (phrenologist), Flickr

(b) Shakey

Abbildung 1.1: Die ersten Roboter: (a) Der Unimate wurde 1961 als erster Industrieroboter bei General Motors eingesetzt. (b) Shakey gilt als der erste mobile Roboter. Er wurde in der Zeit von 1966 bis 1972 am Stanford Research Institute entwickelt.

Im Dezember desselben Jahres verfasste Heinrich Arnold Ernst am Massachusetts Institute of Technology mit seiner Doktorarbeit „*MH-1, a computer-operated mechanical hand*“ die erste wissenschaftliche Arbeit aus dem Bereich der Robotik [8].

Von 1966 bis 1972 wurde am Stanford Research Institute mit „*Shakey*“ (Abbildung 1.1b) der erste intelligente, mobile Roboter entwickelt, der in der Lage war, Aktionen selbst zu planen und auszuführen.

1970 fand in Chicago das erste internationale Symposium der industriellen Robotik (ISIR – International Symposium on Industrial Robotics) statt.

In Japan schlossen sich 1971 erstmals mehrere Unternehmen zur Japan Industrial Robot Association – kurz JIRA – zusammen. 1974 wurde in Amerika die Robotic Industries Association – kurz RIA – gegründet.

1981 wurde das erste wissenschaftliche Buch über Roboter von Richard P. Paul mit dem Titel „*Robot Manipulators – Mathematics, Programming and Control*“ verfasst [9].

Wegen der verstärkten Forschungsarbeit im Bereich der Robotik und der damit verbundenen steigenden Zahl an Forschungsarbeiten wurden erste Journale gegründet. So wurden 1982 das „*International Journal of Robotics Research*“ und 1985 das „*IEEE Journal of Robotics and Automation*“ gegründet.

1.4 Die Vorteile von Robotern

Der Grund für die steigende Anzahl von Robotern liegt in den vielen Vorteilen, die Roboter gegenüber anderen Maschinen und gegenüber dem Menschen haben.

Ein wichtiger Grund für den Einsatz von Robotern in der Industrie liegt in der Wirtschaftlichkeit. Durch ihre schnelle und genaue Arbeitsweise sind Roboter und Maschinen den Menschen oft überlegen. Im Vergleich zu herkömmlichen Maschinen besitzen Roboter den Vorteil, dass sie durch eine Umprogrammierung einfacher für neue Arbeiten angepasst werden können.

Ein weiterer Grund für den Einsatz von Robotern liegt in der erzielbaren Arbeitserleichterung. Roboter können den Menschen schwierige Arbeiten abnehmen und ihn somit entlasten (Abbildung 1.2a). Weit wichtiger als die Arbeitserleichterung ist allerdings die Sicherheit.

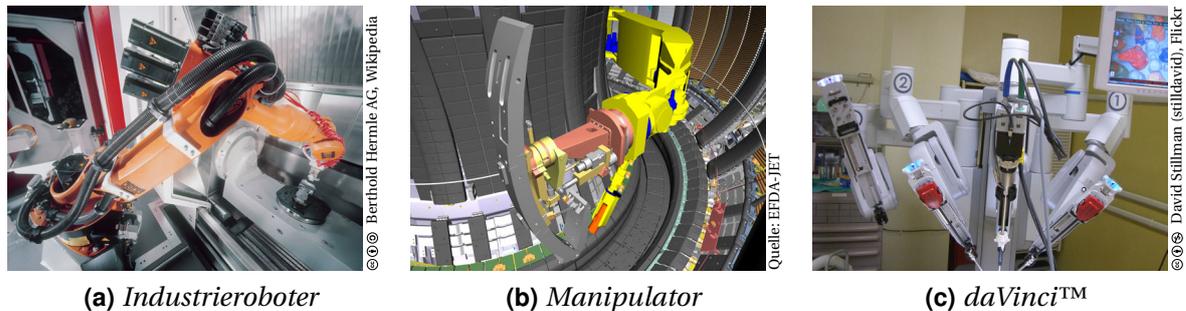


Abbildung 1.2: Beispiele für den Einsatz von Robotern: (a) Industrieroboter werden für unterschiedlichste Aufgaben eingesetzt. Sie werden beispielsweise verwendet zum Schweißen, zum Lackieren, zum Transportieren und, wie in diesem Beispiel, zum Bestücken von Bearbeitungszentren mit schweren Teilen. (b) In gefährlichen Umgebungen werden spezielle Roboter, die so genannten Manipulatoren, eingesetzt um wichtige Arbeiten ferngesteuert zu erledigen. Das Beispiel zeigt eine Simulation eines Manipulators, der am Kernfusionsforschungsreaktor des JET-Projektes zum Einsatz kommt. (c) Patienten profitieren vom Einsatz von Operationsrobotern wie dem daVinci™ durch kleinere Operationswunden.

Roboter können riskante Arbeiten übernehmen und auch in für den Menschen gefährlichen Umgebungen arbeiten. So werden beispielsweise Roboter in Atomkraftwerken in radioaktiver Umgebung eingesetzt, und in Zukunft werden sie auch bei der Kernfusion zum Einsatz kommen (Abbildung 1.2b) [10].

In der Medizin haben Operationsroboter wie der *daVinci*™ [11] für den Patienten den Vorteil, dass bei der Operation kleine Einschnitte reichen und daher die Wunden schneller verheilen, als dies bei herkömmlichen Methoden der Fall ist (Abbildung 1.2c). Dabei ergeben sich auch wirtschaftliche Vorteile, da die Patienten schneller wieder ihrer Beschäftigung nachgehen können.

Auch die Wissenschaft profitiert durch den Einsatz von Robotern. So lassen sich Vulkane, Planeten oder der Meeresgrund mit Hilfe von Robotern leichter und mit geringerem Risiko untersuchen.

Es gibt noch eine Vielzahl von Bereichen, die vom Robotereinsatz profitieren oder noch profitieren werden.

1.5 Die Mobile Robotik

Auch in der Robotik ist Mobilität von großer Bedeutung. Erst durch die Ortsunabhängigkeit von Robotern lassen sich so manche Einsatzbereiche erschließen. Für die Forschung ergeben sich durch die Ortsunabhängigkeit jedoch auch neue Probleme, die es zu lösen gilt. Im Folgenden werden einige wichtige Aspekte der mobilen Robotik beschrieben.

Von großer Bedeutung in der mobilen Robotik ist die Energieversorgung. Da keine Kabelverbindung zum Roboter vorhanden ist, muss die Energieversorgung des Roboters und all seiner Komponenten direkt am Roboter erfolgen. Die Energie kann dabei beispielsweise direkt aus Akkumulatoren entnommen werden. Sie kann aber auch mit Hilfe von Solarzellen oder Verbrennungsmotoren gewonnen werden. Da die Energieversorgung am Roboter Platz benötigt, muss in der mobilen Robotik zusätzlich auf effiziente Komponenten Wert gelegt werden, um die Einsatzdauer der Roboter zu erhöhen.

Mobile Roboter benötigen selbstverständlich auch einen Antrieb um eine Fortbewegung zu ermöglichen. Die Art des Antriebes richtet sich dabei in erster Linie nach dem Einsatzgebiet. Manche Roboter fahren auf Rädern oder Ketten, andere können fliegen, schwimmen oder tauchen, wieder andere gehen auf Beinen. Bei der Wahl des Antriebs ist auch der Regelungsaufwand zu berücksichtigen. Ein Roboter, der auf drei Rädern fährt lässt sich leichter regeln als einer, der auf zwei Beinen geht.

Bei den mobilen Robotern sind Sensoren noch wichtiger als bei ortsfesten Robotern. Während bei den ortsfesten Robotern neben den internen Sensoren, die für die Lageregelung des Roboters unbedingt erforderlich sind, auf externe Sensoren verzichtet werden kann, so sind diese bei mobilen Robotern zwingend erforderlich. Im Falle eines Industrieroboters, der immer gleiche Bauteile an der gleichen Position aufnimmt und transportiert, sind zum Beispiel keine Sensoren zur Positionserkennung dieser Bauteile erforderlich.² Da sich mobile Roboter allerdings meist in einer veränderlichen und oft auch unbekanntem Umgebung bewegen, sind Sensoren, die dem Roboter eine Erfassung der Umgebung ermöglichen, zwingend erforderlich.

Die Steuerung der mobilen Roboter muss – verglichen zu ortsfesten Robotern – mehr Aufgaben übernehmen. Die Sensordaten müssen verarbeitet werden und der Roboter muss darauf entsprechend reagieren. Vielfach werden Algorithmen zur Wegfindung, zur Objekterkennung, zur Kollisionsvermeidung und Ähnlichem eingesetzt.

In Abbildung 1.3 ist eine kleine Auswahl verschiedener mobiler Roboter für unterschiedliche Aufgaben dargestellt. Anhand dieser Beispiele lässt sich erkennen, wie weit gefächert das Gebiet der mobilen Robotik ist.

1.6 Die Sensoren für mobile Roboter

Sensoren sind für Roboter unumgänglich, denn selbst der einfachste Roboter benötigt Sensoren zur Regelung von beispielsweise Arm oder Greifer. Mobile Roboter benötigen auch Sensoren, die es ihnen ermöglichen, die Umgebung wahrzunehmen. Eine hohe Autarkie und

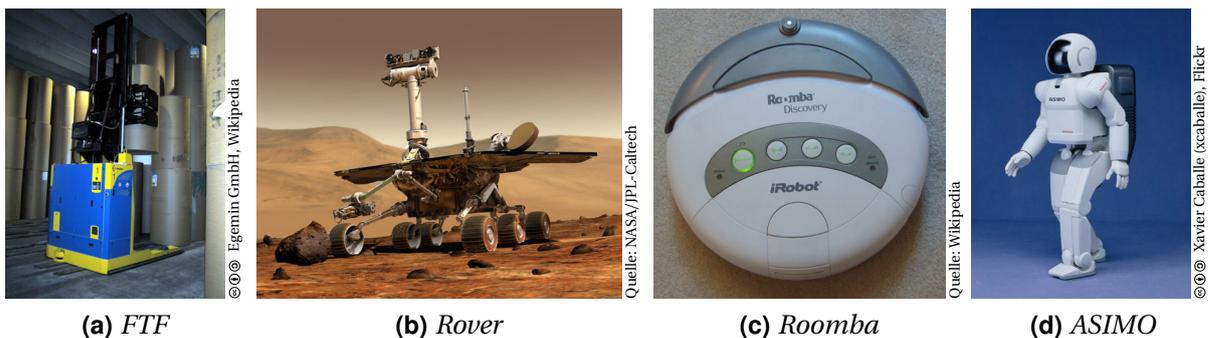


Abbildung 1.3: Beispiele für mobile Roboter: (a) Fahrerlose Transportfahrzeuge – kurz FTF – werden in der Industrie für Transportaufgaben eingesetzt. (b) Der Mars Exploration Rover MER-A „Spirit“ wurde zur Erforschung der Marsoberfläche gebaut. Er verwendet unter anderem ein optisches Sensorsystem, das nach dem Stereoverfahren arbeitet [12]. (c) Der Staubsaugerroboter Roomba der Firma iRobot zählt zu den erfolgreichsten Vertretern seiner Art. (d) Der ASIMO von Honda ist ein rund 120cm großer menschenähnlicher Roboter.

² Auch wenn Sensoren nicht zwingend nötig sind, so werden sie doch vielfach eingesetzt. Einfache Sensoren haben kaum Einfluss auf die Produktionskosten der Roboter, verbessern jedoch die Flexibilität der Roboter.

Intelligenz ist nur erreichbar, wenn dem Roboter genügend Informationen zur Verfügung gestellt werden können. Diese Informationen erhält ein Roboter aus eben diesen Sensoren. Es gibt unterschiedliche Sensortypen, von denen im Folgenden einige oft verwendete Sensortypen erwähnt werden.

Kraftsensoren und taktile Sensoren werden besonders dort benötigt, wo teilweise empfindliche Gegenstände sicher gegriffen werden müssen. Aus diesem Grund werden sie häufig in Greifern eingesetzt. Mit Hilfe der Sensordaten kann die Kraft des Greifers so dosiert werden, dass der Gegenstand gefahrlos bewegt werden kann und gleichzeitig nicht durch den Greifer beschädigt wird. Mit Hilfe dieser Sensoren können aber auch kleinere Objekte erkannt werden [13, 14]. Bei zweibeinigen Robotern werden Kraftsensoren auch in der Fußsohle verwendet, um den Roboter besser regeln zu können [15, 16].

Drehgeber, Beschleunigungssensoren, Gyroskope und GPS werden zur absoluten oder relativen Positionsbestimmung von mobilen Robotern eingesetzt.

Ultraschallsensoren werden beispielsweise dazu verwendet, um Hindernisse zu erkennen. Gerade für diesen Einsatzbereich gibt es viele preiswerte Sensoren. Für genauere Anwendungen sind diese Sensoren allerdings nur schlecht geeignet, da sie eine geringe Tiefenauflösung besitzen.

Optische Sensoren arbeiten genauer als Ultraschallsensoren und eignen sich daher auch zur dreidimensionalen Erfassung von Gegenständen. Meist arbeiten sie nach dem, in dieser Arbeit beschriebenen Triangulationsprinzip. Dabei arbeiten sie entweder aktiv wie zum Beispiel ein Laserscanner oder passiv wie eine Stereokamera. Eine detaillierte Beschreibung des Funktionsprinzips solcher Sensoren findet sich in den Kapiteln 2.3 und 2.4. Es gibt aber auch optische Tiefenbildsensoren, bei denen die gesamte aufzunehmende Szene auf einmal beleuchtet wird und die Tiefenbildinformationen aus Laufzeitmessungen ermittelt werden [17–19].

Meist werden gleich mehrere der oben aufgelisteten Sensortypen parallel eingesetzt. Dabei werden oft auch zwei oder mehrere Sensortypen für dieselbe Aufgabe verwendet, mit dem Ziel, die Vorteile der unterschiedlichen Typen zu kombinieren. In solchen Fällen muss die Software des Roboters dazu in der Lage sein, teils unterschiedliche Sensordaten zu kombinieren und ungenaue Daten herauszufiltern.

1.7 Die Motivation

Das Ziel dieser Arbeit ist die Entwicklung eines preiswerten Sensorsystems zur Aufnahme von Tiefenbildern. In diesem Kapitel werden die Hauptmotivationsgründe für diese Entwicklung beschrieben.

Wie die Entwicklungsgeschichte der Robotik zeigt, war die Robotik lange von den Industrierobotern dominiert. Neben den Industrierobotern, die schnell zum Standard wurden, wurden meist nur ganz spezielle Roboter verwendet, von denen es oft nur einen einzelnen gab. Die Ansprüche, die die Industrie an die Roboter stellt unterscheiden sich jedoch stark von denen, die an Haushaltsroboter gestellt werden.

In der Industrie wird auf Robustheit, Langlebigkeit und Präzision Wert gelegt, was sich auch in den zum Einsatz kommenden Komponenten widerspiegelt. Diese Komponenten

sind für die mobile Anwendung jedoch oft zu schwer oder zu groß. Ein weiterer Nachteil dieser Komponenten, der eine rasche Verbreitung von Robotern im Haushalt verhindern könnte, ist der relativ hohe Preis, der für die besseren Eigenschaften zu zahlen ist.

Das Problem des hohen Preises ließe sich durch Serienfertigung großer Stückzahlen passender Komponenten reduzieren. Im Forschungsbereich, indem meist nur ein Prototyp gebaut beziehungsweise erprobt wird, fällt der Preis meist hoch aus. Dies ist insbesondere dann ein Problem, wenn mehrere verschiedene Komponenten miteinander verglichen werden sollen.

Eine weitere Schwierigkeit liegt in der Ansteuerung der Komponenten. In der Industrie werden dafür verschiedene Protokolle und Verfahren wie beispielsweise CAN-Bus, Interbus oder Profibus verwendet. Zur Ansteuerung der Komponenten müssen die PCs daher mit den jeweiligen Schnittstellen ausgerüstet werden. Zudem müssen meist auch noch Software-Lizenzen zur Programmierung dieser Schnittstellen angekauft werden. Weisen die verfügbaren Komponenten dann auch noch unterschiedliche Schnittstellen auf, so können auch hier hohe Kosten entstehen.

Im Forschungsbereich ist es daher wünschenswert, wenn die einzelnen Komponenten eines mobilen Roboters vergleichsweise leicht und klein sind und mit handelsüblichen PCs angesteuert werden können. Da solche Komponenten oft noch nicht existieren, müssen sie neu entwickelt werden. Der Tiefenbildsensor, dessen Entwicklung Inhalt dieser Arbeit ist, ist eine solche neu zu entwickelnde Komponente. Er soll genau diese Kriterien erfüllen und daher besonders für Test- und Vergleichszwecke in der Forschung geeignet sein.

1.8 Das Projekt robots@home

Ein Ziel vieler Wissenschaftler im Bereich der Robotik ist es, Robotern den Einzug in unsere Haushalte zu ermöglichen. Roboter sollen diverse, oft unliebsame Arbeiten übernehmen. Bisher kommen im Haushalt Roboter nur zum Wischen oder Staubsaugen zum Einsatz, wobei dabei die meisten Roboter kaum Intelligenz aufweisen und die Wohnung zufällig abfahren.³ Für eine weitere Integration von Robotern in den Haushalten sind noch einige Verbesserungen erforderlich.

Das mit Mitteln aus dem sechsten Forschungsrahmenprogramm der Europäischen Kommission unterstützte Projekt *robots@home* soll eine offene, mobile und preiswerte Plattform für zukünftige Entwicklungen zur Verfügung stellen.

Projektziel: Ausgestattet mit rein optischen Sensoren soll sich der Roboter unmittelbar nach dem Auspacken in einer, ihm unbekanntem Wohnung oder einem Büro zurechtfinden. Dabei soll der Roboter mit so wenig Unterstützung von Seiten des Benutzers als möglich auskommen. Anhand von bereits gespeicherten Informationen über typische Möbel soll der Roboter in der Lage sein, die Möbel in der neuen Umgebung zu klassifizieren. Der Roboter soll in der Lage sein, selbstständig seinen Weg zu finden, ohne dabei an Hindernisse anzustoßen.

³ Mittlerweile gibt es mit den Modellen VC-RE70V und VC-RE72V aus der Serie Hauzen des koreanischen Konzerns Samsung Staubsaugerroboter, die zusätzlich mit einer Kamera und einer Wegplanung ausgestattet sind.

Das Projekt, das im Mai 2007 gestartet wurde, soll bis April 2010 abgeschlossen sein. Mehr Informationen über das Projekt finden sich im Internet unter <http://robots-at-home.acin.tuwien.ac.at>.

1.9 Das bestehende System

Am Institut für Automatisierungs- und Regelungstechnik der TU Wien existiert bereits ein Tiefenbildsensor, der dem, im Zuge dieser Diplomarbeit entwickelten Sensor als Grundlage diente. Er wurde im Zuge einer Diplomarbeit im Jahr 2004 erstellt [20] (Abbildung 1.4).

Als Kamera kommt bei diesem System ein MAPP 2500 von der, inzwischen zum Unternehmen SICK gehörenden, Firma IVP zum Einsatz. Diese Kamera hat den Vorteil, dass sie einen integrierten Mikroprozessor besitzt, der eine interne Bildverarbeitung ermöglicht und somit den angeschlossenen Computer entlastet. Angeschlossen wird die Kamera jedoch über eine HSSI-Schnittstelle⁴, wofür für PCs kaum preiswerte Schnittstellenkarten zu finden sind.

Geschwenkt wird der Sensor mit Hilfe einer Schwenk-/Neigeeinheit der PowerCube-Serie von der Firma AMTEC, die inzwischen zur SCHUNK GmbH & Co. KG gehört. Angesteuert wird diese Schwenkeinheit über eine CAN-Schnittstelle, wofür ebenfalls eine eigene Schnittstellenkarte benötigt wird. Im Vergleich zu HSSI bietet CAN zumindest den Vorteil, dass auch CAN-USB Adapter erhältlich sind, wodurch eine Ansteuerung mittels Notebook möglich wird.

Die unterschiedlichen Schnittstellen und der Umstand, dass die entsprechenden Komponenten nur noch schwer erhältlich sind, sind von großem Nachteil. Bei einem eventuell auftretenden Defekt an einer Komponente können erhebliche Kosten für deren Austausch entstehen.



Abbildung 1.4: Der bestehende Tiefenbildsensor: Die wichtigsten Bestandteile des am Institut existierenden Tiefenbildsensors sind eine „MAPP 2500“ Kamera von IVP und eine Schwenk-/Neigeeinheit „PowerCube“ von AMTEC.

⁴ HSSI steht für *High Speed Serial Interface*. Diese Schnittstelle wurde eigentlich zur Hochgeschwindigkeitsübertragung in Telefon- und Datennetzen entwickelt.

2

2 Grundlagen

Dieses Kapitel bietet einen Überblick der verwendeten Verfahren zur Tiefenbildgenerierung. Den Beginn macht ein Kapitel über die technischen und mathematischen Grundlagen der für diese Verfahren benötigten Kameras. Im Anschluss daran erfolgt eine kurze Beschreibung der Triangulation, die bei beiden Verfahren benötigt wird. Nach einer Beschreibung von Lichtschnitt- und Stereoverfahren werden diese miteinander verglichen.

2.1 Die Kamera

2.1.1 Sensortypen

Der Sensor ist das Herzstück einer jeden digitalen Kamera. Er nimmt die optische Strahlung des einfallenden Lichts auf und wandelt sie in elektrische Signale um. Dabei ändert sich das Signal mit der Helligkeit des Lichtes. Es kommen heute hauptsächlich zwei unterschiedliche Sensortypen zum Einsatz [21].

CCD-Sensor

Der erste Sensortyp ist der so genannte CCD-Sensor. CCD steht für *Charge Coupled Device*. Dieser elektronische Bauteil wurde 1969 von Willard Boyle und George E. Smith in den Bell Laboratorien entwickelt [22]. Ein CCD-Sensor besteht aus mehreren Zellen, die elektrische Ladung speichern können. Diese Ladung kann durch ein Taktsignal von Zelle zu Zelle weitergeschoben werden. Somit können die Ladungswerte der einzelnen Zellen sequentiell ermittelt werden. Werden diese Zellen mit lichtempfindlichen Photodioden kombiniert oder selbst lichtempfindlich gemacht, so entsteht ein zeilen- oder matrixförmiges Sensorelement [23–25]. Jede CCD-Zelle entspricht dabei einem Bildelement (Pixel¹).

Die Matrix bei zweidimensionalen Sensoren wird dabei meist durch mehrere vertikale und einem horizontalen CCD aufgebaut (Abbildung 2.1a auf der nächsten Seite). Dieser Aufbau hat den Vorteil, dass nur eine Verstärkerschaltung benötigt wird. Zusätzlich ist der Anteil der fotoempfindlichen Sensorfläche an der Gesamtfläche sehr groß. Da aber für einen kompletten Auslesevorgang alle Ladungen bis zur Verstärkerschaltung verschoben werden müssen, ist die Auslesegeschwindigkeit im Vergleich zum Active Pixel Sensor geringer.

¹ Pixel ist eine Abkürzung aus dem englischen Wörtern *picture* für Bild und *element* für Element.

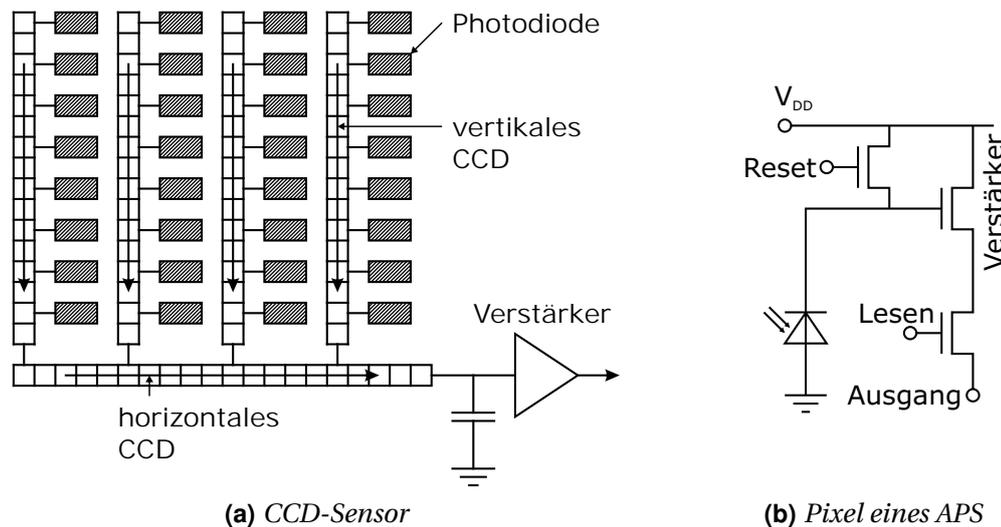


Abbildung 2.1: Schematische Darstellung zweier Typen von Photosensoren: (a) Beim CCD-Sensor werden in mehreren Zellen elektrische Ladungen gespeichert. In Kombination mit Photodioden entsteht ein lichtempfindlicher Flächensensor. (b) Beim Active Pixel Sensor – auch als CMOS-Sensor bezeichnet – wird für jedes einzelne Pixel ein Verstärker verwendet. Bei einer schlechteren Flächenausnutzung ermöglicht er höhere Bildraten bei geringerem Energieverbrauch.

Active Pixel Sensor

Der *Active Pixel Sensor* – kurz APS – ist ein weiterer Sensortyp. Er wird oft auch als CMOS-Sensor bezeichnet, da er in CMOS²-Technologie gefertigt wird. Der gravierende Unterschied zum CCD besteht darin, dass für jedes Pixel einen eigenen Verstärker verwendet wird [26–28] (Abbildung 2.1b). Somit sind neben der Fotodiode noch mehrere Transistoren erforderlich, wodurch sich die Flächenausnutzung verschlechtert. Von Vorteil sind allerdings der geringere Energieverbrauch und die höhere Bildrate. Allerdings ist das Rauschverhalten von Active Pixel Sensoren schlechter beziehungsweise kann es nur durch aufwendige Methoden verbessert werden.

Hauptgründe für das schlechtere Rauschverhalten sind die geringere Flächenausnutzung und die hohe Anzahl an Verstärkern. Durch die geringere Flächenausnutzung gelangt weniger Licht auf die fotoempfindlichen Zellen, was aber durch den Einsatz von Mikrolinsen reduziert werden kann [28]. Wegen Produktionsungenauigkeiten arbeiten die Verstärker nicht exakt gleich, was vor allem bei kurzer Belichtung zu starken Unterschieden bei den einzelnen Pixeln führt. Durch aufwendigere Verstärkerschaltungen wird versucht, diesen Einfluss zu verringern [29, 30].

2.1.2 Monochrom- und Farbkameras

Sowohl der CCD-Sensor als auch der Active Pixel Sensor arbeiten mit fotoempfindlichen Elementen. Diese Elemente wandeln die eintreffende optische Strahlung in elektrische Ladungen um, wobei die Höhe der Ladung mit der Menge des Lichtes, also mit der Helligkeit zusammenhängt. Da die fotoempfindlichen Elemente im gesamten Bereich des sichtbaren Lichtes und auch darüber hinaus empfindlich sind, können diese nur eine Helligkeitsinfor-

² CMOS steht für *Complementary Metal Oxide Semiconductor*.

mation liefern. Es sind somit nur monochrome Bilder möglich. Für Farbbilder sind zusätzliche Maßnahmen erforderlich.

Bei teuren Kameras wird das Bild durch Prismen in die drei Farbkomponenten Rot, Grün und Blau zerlegt und von drei Fotosensoren aufgenommen. Bei den meisten anderen Kameras wird ein so genannter Bayer-Filter verwendet, um Farbinformationen zu erhalten.

Der Bayer-Filter

Der Bayer-Filter ist nach seinem Erfinder Bryce E. Bayer benannt, der im Jahre 1975 ein Patent für einen Farbfilter einreichte [31]. Dabei werden die einzelnen Sensorpixel mit Farbfiltern versehen und zwar 50% mit grünen Filtern und jeweils 25% mit roten beziehungsweise blauen Filtern. Diese Aufteilung beruht darauf, dass das menschliche Auge empfindlicher für grüne Farben ist als für rote oder blaue. Dabei gibt es vier mögliche Anordnungen der Farbfilter, die in Abbildung 2.2 dargestellt sind. Die tatsächlichen Farben der einzelnen Pixel werden durch Interpolation ermittelt. Diese Interpolation erfolgt meist in der Kamera, kann aber auch erst später am Computer erfolgen.

Im Vergleich zu Monochromkameras stehen bei Farbkameras mit Bayer-Filter nur ein Viertel der Pixel für Rot- und Blau-Kanal beziehungsweise nur die Hälfte für den Grün-Kanal zur Verfügung. Diese geringere effektive Sensorgröße führt zu schlechteren, beziehungsweise ungenaueren Bildern, woran auch eine Interpolation nichts mehr ändern kann. Eine Verwendung einer Farbkamera als Monochromkamera bringt auch keine Verbesserung.

Farbmodelle und deren Speicherbedarf

Farbbilder benötigen zumindest drei Werte je Pixel um Farben korrekt wiedergeben zu können. Bei 8-Bit je Kanal ergeben sich somit 24 Bit je Pixel bei Farbbildern. Durch verschiedene Farbmodelle kann die Anzahl der Bit je Pixel verringert und dadurch die Datenrate erhöht werden. In Tabelle 2.1 sind einige Farbmodelle aufgelistet.

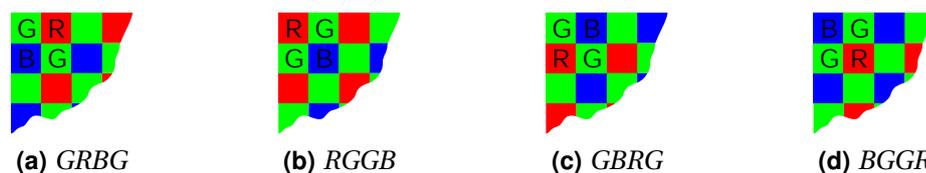


Abbildung 2.2: Die vier Möglichkeiten der Farbfilteranordnung beim Bayer-Filter. Die Bezeichnung ergibt sich aus den ersten 2x2 Feldern.

Name	Bit/Pixel	Beschreibung
YUV 4:4:4	24	YUV-Farbmodell mit allen drei Werten je Pixel
RGB-24	24	RGB-Farbmodell
YUV 4:2:2	16	YUV-Farbmodell: Y-Wert für jedes Pixel, U- und V-Wert für jedes zweite.
YUV 4:1:1	12	YUV-Farbmodell: Y-Wert für jedes Pixel, U- und V-Wert für jedes vierte.
Y-8	8	Monochrom

Tabelle 2.1: Liste von Farbmodellen und deren Speicherbedarf

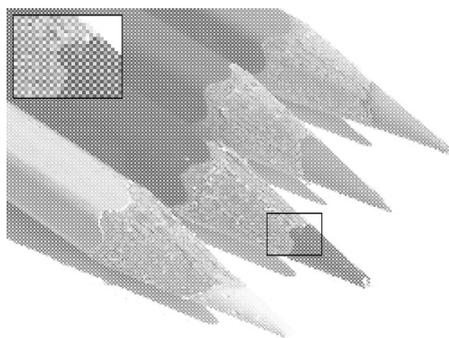
Das RGB-Farbmodell wird zur Darstellung von Bildern am Computer verwendet und besitzt die drei Farbkanäle rot (R), grün (G) und blau (B). Viele Algorithmen zur Verarbeitung von Farbbildern arbeiten mit diesem Farbmodell. Bilder im RGB-Farbmodell haben somit den Vorteil, dass sie gleich direkt verwendet werden können, während Bilder in anderen Farbmodellen erst umgewandelt werden müssen. Beim YUV-Farbmodell trägt der Y-Kanal die Helligkeitsinformationen (Luminanz) des Bildes, U- und V-Kanal tragen den Farbanteil (Chrominanz). Der Vorteil dieses Formates liegt darin, dass der Y-Kanal das gesamte monochrome Bild trägt, da monochrome Bilder ja nur Helligkeitsinformationen enthalten. Dadurch ist das YUV-Farbmodell kompatibel mit dem monochromen Y-Format.

Farbkameras im Vergleich mit RAW-Kameras

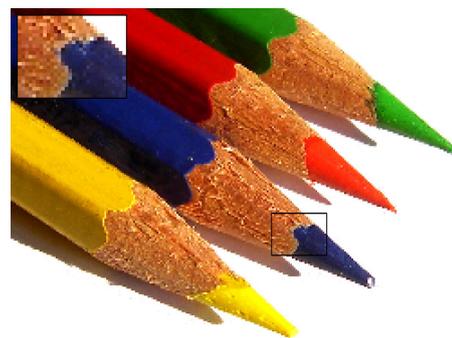
Bei den Kameras wird zwischen reinen Farbkameras und so genannten RAW-Farbkameras unterschieden. Bei beiden Kameras werden dieselben Bildsensoren mit denselben Bayer-Filtern verwendet. Der Unterschied liegt lediglich im übertragenen Signal.

Bei den reinen Farbkameras wird das Bild bereits in der Kamera durch einen Mikroprozessor in ein Farbbild umgewandelt und als solches übertragen. RAW-Kameras liefern ein Signal im monochromen Y-8-Format beziehungsweise im Y-16-Format. Die Umwandlung in ein Farbbild hat durch den PC zu erfolgen. Das monochrome Signal darf allerdings nicht darüber hinwegtäuschen, dass die Kamera durch den Bayer-Filter nicht dazu in der Lage ist, monochrome Bilder zu liefern. Wegen dem Filter weist eine monochrome Darstellung der Bilder ein Gittermuster auf (Abbildung 2.3a).

Durch den, für monochrome Bilder geringeren Speicherbedarf, können bei RAW-Kameras mehr Bilder je Zeiteinheit übertragen werden – die Übertragungsgeschwindigkeit ist größer. Im Gegenzug dafür steigt allerdings der erforderliche Rechenaufwand zur Umwandlung des Bildes in ein Farbbild bei der RAW-Kamera. Ein weiterer Vorteil von RAW-Kameras liegt in deren Flexibilität. Bei andern Farbkameras ist der Algorithmus zur Berechnung des Farbbildes vorgegeben, während bei RAW-Kameras diese Umrechnung noch beeinflusst werden kann, wodurch sich Bildqualität und Rauschen eventuell verbessern lassen [32].



(a) Kamerabild im RAW-Format



(b) Kamerabild nach der Interpolation

Abbildung 2.3: Umwandlung eines RAW-Bildes in ein Farbbild: Bei RAW-Kameras wird das Bild monochrom im RAW-Format übertragen. Wird dieses Bild ohne zusätzliche Maßnahmen betrachtet, so erscheint es als ein Monochrombild mit Gittermuster (a). Erst nach der Anwendung eines Bayer-Algorithmus ist das eigentliche Farbbild zu sehen (b).

2.1.3 Kameramodell

Zur mathematischen Beschreibung des Abbildungsvorganges der realen, dreidimensionalen Welt in ein zweidimensionales Bild wird ein idealisiertes Kameramodell benötigt. Als solches Modell wird die einfachste Kamera verwendet, die so genannte Lochkamera, die im Englischen als *pinhole camera* bezeichnet wird. Eine Lochkamera ist eine dunkle Zelle mit einem kleinen Loch an einer Seite (Abbildung 2.4). Das durch dieses Loch einfallende Licht erzeugt an der gegenüberliegenden Seite ein Bild, das von einem Bildsensor aufgenommen werden kann.

Der Abstand der Hauptebene zur Bildebene wird als Brennweite f bezeichnet. Unter Anwendung des Kongruenzsatzes für Dreiecke ergibt sich zwischen den tatsächlichen und den abgebildeten Größen nach Abbildung 2.4 der Zusammenhang

$$-y = f \frac{Y}{Z}. \quad (2.1)$$

Für eine weitere Vereinfachung des Modells kann die Bildebene vor die Hauptebene gesetzt werden, wodurch sich lediglich das Vorzeichen der Gleichung ändert.

Bislang wird für das Kameramodell lediglich ein Parameter – die Brennweite f – benötigt. Da es bei Bildern allerdings nicht üblich ist, Abstände von der Mitte aus, sondern vielmehr von einer Ecke aus zu messen, wird noch ein zusätzlicher Offset benötigt. Dieser Offset gibt die Verschiebung zwischen dem Bild- und dem Kamerakoordinatensystem in x- und y-Richtung an und wird als p_x beziehungsweise p_y geschrieben. Somit lässt sich die Projektion formal schreiben als

$$x = f \left(\frac{X}{Z} \right) + p_x, \quad y = \left(f \frac{Y}{Z} \right) + p_y. \quad (2.2)$$

Diese beiden Gleichungen lassen sich mit Hilfe von Vektoren und Matrizen in eine kompaktere Form bringen. Zur leichteren Berechnung von beispielsweise Translation und Rotation wird meist mit so genannten homogenen Koordinaten gearbeitet. Dabei wird in einem um eine Dimension erhöhten Raum gearbeitet. In diesen homogenen Koordinaten

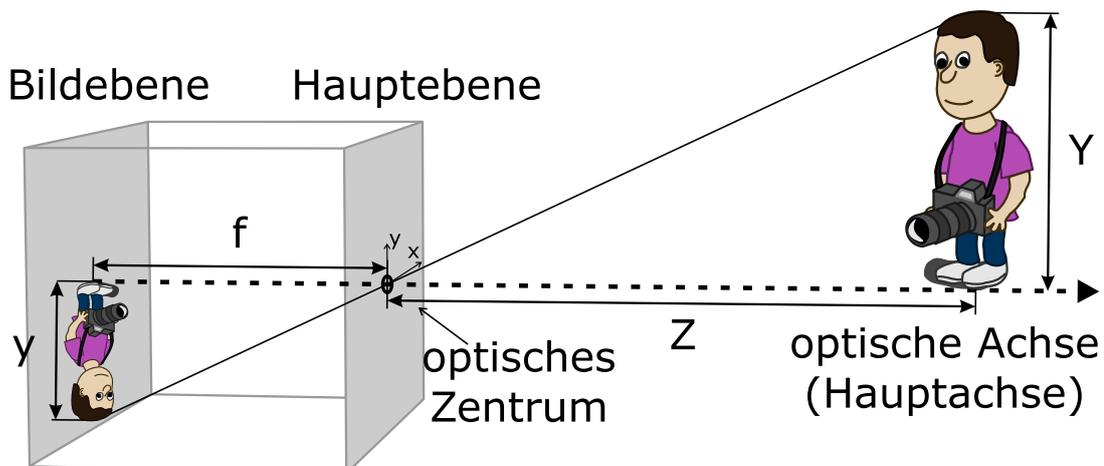


Abbildung 2.4: Schematische Darstellung einer Lochkamera

ergibt sich Gleichung 2.2 zu:

$$\begin{pmatrix} x \\ y \\ Z \end{pmatrix} = \begin{bmatrix} \mathbf{K} & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}, \quad \text{mit } \mathbf{K} = \begin{bmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix}. \quad (2.3)$$

Dabei enthält die Kalibrierungsmatrix \mathbf{K} die drei bisherigen Parameter.

Gleichung 2.3 setzt voraus, dass es sich bei den Bildkoordinaten um euklidische Koordinaten mit gleich skalierten Achsen handelt. Bei Bildsensoren ist es allerdings einfacher, als Koordinaten die Zeilen und Spalten des Sensors zu wählen. Da die einzelnen Pixel nicht unbedingt quadratisch sind, wird der Parameter f durch die Parameter α_x und α_y ersetzt. Dabei sind $\alpha_x = f m_x$ und $\alpha_y = f m_y$ die Brennweiten in Pixel-Dimension. Die Parameter m_x und m_y geben die Anzahl der Pixel je Längeneinheit an. Mit den neuen Offset-Parametern $x_0 = m_x p_x$ und $y_0 = m_y p_y$ ergibt sich die Kalibrierungsmatrix somit zu:

$$\mathbf{K} = \begin{bmatrix} \alpha_x & 0 & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (2.4)$$

2.1.4 Verzeichnung

Für eine technische Anwendung weist die Lochkamera jedoch einen entscheidenden Nachteil auf. Durch das sehr kleine Loch kann auch nur sehr wenig Licht eintreten, was bedeutet, dass eine sehr lange Belichtungszeit erforderlich ist. Diese Zeit ist jedoch für eine Videokamera, die 30 oder mehr Bilder je Sekunde aufnehmen soll, zu lange. Aus diesem Grund arbeiten Kameras mit Linsen, die das einfallende Licht bündeln. Somit gelangt mehr Licht in die Kamera und genügend kurze Belichtungszeiten sind realisierbar.

Die Verwendung von Linsen führt allerdings zu einem Problem. Aufgrund von Fertigungsungenauigkeiten und Materialfehlern arbeiten Linsen nicht ideal. Besonders bei günstigeren Linsen tritt das Phänomen der radialen Verzeichnung verstärkt auf. Dadurch wird das Bild kissen- oder tonnenförmig verzerrt und gerade Linien erscheinen im Bild gekrümmt (Abbildung 2.5).

Gelingt es, den durch die Verzeichnung auftretende Fehler mathematisch zu beschreiben, so kann der Fehler aus dem Bild herausgerechnet, beziehungsweise korrigiert werden. Solche dafür notwendigen Modelle zur Beschreibung der Verzeichnung wurden bereits in

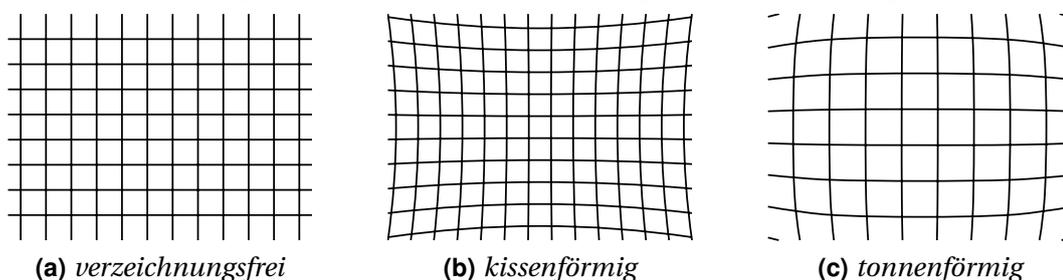


Abbildung 2.5: Radiale Verzeichnung: Durch die radiale Verzeichnung werden ursprünglich gerade, parallele Linien (a) gekrümmt. Die Verzeichnung kann dabei kissen- (b) oder tonnenförmig (c) ausfallen.

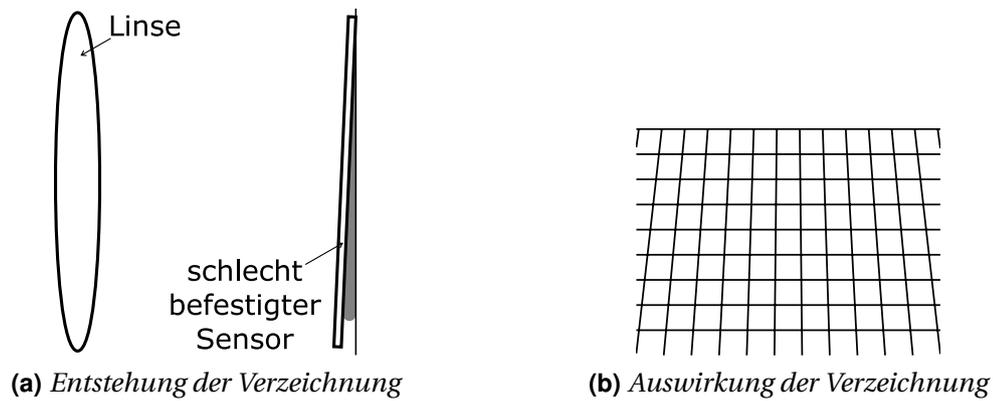


Abbildung 2.6: Tangentiale Verzeichnung: Wird der Bildsensor nicht exakt parallel zur Linse montiert, so kommt es zur tangentialen Verzeichnung (a). Dabei erscheinen einst parallele Linien nicht mehr parallel (b).

den 70er-Jahren entwickelt [33, 34]. Die radiale Verzeichnung lässt sich durch eine Taylorreihe beschreiben, die bereits nach dem dritten Term abgebrochen werden kann. Somit sind für die Korrektur der radialen Verzeichnung drei weitere Parameter zu bestimmen.

Neben der radialen Verzeichnung existiert auch noch eine tangentielle Verzeichnung. Diese kommt zustande, wenn der Bildsensor nicht exakt parallel zur Linse befestigt ist. Bei der tangentialen Verzeichnung bleiben die Linien gerade, verlaufen aber nicht mehr parallel (Abbildung 2.6). Zur Korrektur der tangentialen Verzeichnung sind zwei weitere Parameter erforderlich.

Mehr zur Korrektur der Verzeichnung findet sich in „*Learning OpenCV: Computer Vision with the OpenCV Library*“ [35, S. 370ff].

2.1.5 Kamerakalibrierung

Verfahren zur Tiefenbildgenerierung benötigen für korrekte Ergebnisse verzerrungsfreie Bilder. Mit Hilfe der oben beschriebenen Parameter können die Bilder entzerrt werden. Zusätzlich zu diesen Parametern werden bei der Tiefenbildgenerierung noch Informationen über die Position der Kamera benötigt. Die Ermittlung all dieser Parameter wird als Kamerakalibrierung bezeichnet.

Neben den bisher genannten Kameraparametern, die auch als intrinsische Parameter bezeichnet werden, existieren auch noch so genannte extrinsische Parameter. Die intrinsischen Parameter beschreiben die geometrischen Zusammenhänge im Kamerainneren. Dem entsprechend beschreiben die extrinsischen Parameter die äußeren Zusammenhänge. Konkret beschreiben diese Parameter eine Koordinatentransformation zwischen dem Kamera-Koordinatensystem und einem beliebigen Bezugssystem. Sie bestehen also aus je drei Translations- und Rotationswerten.

Zur Ermittlung der Parameter wird ein Gleichungssystem benötigt. Im Zuge der Kamerakalibrierung werden dazu Bilder von einem bekannten Kalibrierobjekt aufgenommen, in denen anschließend nach markanten Punkten gesucht wird. Sind genug Punkte gefunden, so kann unter Zuhilfenahme der bekannten Informationen über das Kalibrierobjekt das Gleichungssystem gelöst werden. Bei Systemen mit mehr als einer Kamera, wie dem in dieser Arbeit beschriebenen, wird die Kalibrierung der Kameras oft auch gemeinsam durchgeführt.

Dadurch lassen sich auch andere Parameter, wie zum Beispiel der Abstand der Kameras zueinander, ermitteln.

Als Kalibrierobjekt sind verschiedene Objekte denkbar. Am häufigsten wird aber ein zweidimensionales Objekt mit einem Schachbrett-Muster verwendet [36–38]. Es gibt aber auch Methoden, die mit eindimensionalen Objekten arbeiten [39]. Und besonders bei Systemen zur Tiefenbildgenerierung kommen gerne auch dreidimensionale Objekte zum Einsatz.

2.2 Die Triangulation

Triangulation dient im Allgemeinen dazu, aus bekannten Längen oder Winkeln, unter Ausnutzung spezieller Eigenschaften von Dreiecken – wie dem Kongruenzsatz – verbleibende Unbekannte zu ermitteln. Im Falle der optischen Tiefenbildverfahren dient dies dazu, aus den vorhandenen, zweidimensionalen Bildinformationen dreidimensionale Informationen zu gewinnen.

Gemäß dem Kameramodell ergibt sich der Zusammenhang zwischen einem dreidimensionalen Punkt \mathbf{X} und dessen zweidimensionalem Abbild \mathbf{x} zu

$$\mathbf{x} = \mathbf{P}\mathbf{X} \quad (2.5)$$

wobei die 3×4 Parametermatrix \mathbf{P} sowohl die intrinsischen als auch die extrinsischen Parameter enthält. Schreiben wir nun $\mathbf{x} = w(x, y, 1)^\top$ mit einem Skalierungsfaktor w für die Bildkoordinaten und \mathbf{p}_i für die i -te Zeile von \mathbf{P} , so kann Gleichung 2.5 geschrieben werden als

$$wx = \mathbf{p}_1\mathbf{X}, \quad wy = \mathbf{p}_2\mathbf{X}, \quad w = \mathbf{p}_3\mathbf{X}. \quad (2.6)$$

Durch Elimination des Skalierungsfaktor w aus diesen Gleichungen ergibt sich das Gleichungssystem

$$\begin{aligned} xp_3\mathbf{X} &= \mathbf{p}_1\mathbf{X} \\ yp_3\mathbf{X} &= \mathbf{p}_2\mathbf{X}. \end{aligned} \quad (2.7)$$

Mit den Daten aus zwei unterschiedlichen Bildern, sei es von verschiedenen Kameras oder einfach aus verschiedenen Positionen, lässt sich dieses Gleichungssystem auf ein System mit vier linearen Gleichungen erweitern. Mit einer passenden 4×4 Matrix \mathbf{A} lässt sich das Gleichungssystem schreiben als

$$\mathbf{A}\mathbf{X} = \mathbf{0}, \quad (2.8)$$

wofür nur noch eine von $\mathbf{0}$ verschiedene Lösung für \mathbf{X} gefunden werden muss.

Zum Lösen der Gleichung 2.8 existieren unterschiedliche Methoden. Eine populäre Methode ist die des *linearen Eigenwertes*. Bei dieser Methode wird nach einem Vektor \mathbf{X} gesucht, der $\|\mathbf{A}\mathbf{X}\|$ unter der Bedingung $\|\mathbf{X}\| = 1$ minimiert. Die Lösung entspricht dabei dem Eigenvektor zum kleinsten Eigenwert von $\mathbf{A}^\top\mathbf{A}$.

Diese Methode ist allerdings für ungenaue Daten, wie sie zum Beispiel durch veräuschte Bilder entstehen nicht besonders gut geeignet, da sich in diesem Fall nicht immer eine exakte Lösung finden lässt. Hartley entwickelte deshalb eine Methode mit der das Triangulationsproblem in ein Minimierungsproblem übergeführt wird [40]. In dessen Arbeit findet sich auch ein Vergleich zwischen der Methode des linearen Eigenwertes und seiner Methode.

2.3 Das Lichtschnittverfahren

An Verfahren zur Tiefenbilderzeugung wird bereits seit den 70er-Jahren geforscht. Einen Überblick über die Forschungsergebnisse gab Jarvis bereits im Jahre 1983 [41]. Und 1988 veröffentlichte Besl eine Übersicht von Verfahren und kommerzielle Sensoren [42]. Ein Verfahren, an dem schon ebenso lange geforscht wird ist das Lichtschnittverfahren, das in diesem Kapitel näher beschreiben wird.

2.3.1 Überblick

Beim Lichtschnittverfahren handelt es sich um ein aktives, optisches Verfahren. Aktiv bedeutet in diesem Fall, dass eine zusätzliche Lichtquelle benötigt wird. Die Funktionsweise beruht auf dem Prinzip der Triangulation. Dabei wird aus dem bekannten Abstand von Lichtquelle und Photosensor über den Winkel der Abstand zum reflektierenden Punkt gemessen (Abbildung 2.7).

Dafür projiziert eine Lichtquelle Punkte, Linien oder allgemein regelmäßige Muster auf die Oberflächen der zu vermessenden Objekte. Die Triangulation wird umso genauer, je genauer die Position der Punkte auf der Oberfläche bestimmt werden kann. Deshalb wird als Lichtquelle meist ein Laser verwendet, da dieser die Projektion scharfkantiger Muster ermöglicht. Es ergeben sich dadurch auch noch weitere Vorteile, wie zum Beispiel eine gute Filtermöglichkeit des Lichtes aufgrund der konstanten Wellenlänge. Außerdem lassen sich infrarote Laser verwenden und durch die Halbleitertechnologie sind kurz Lichtimpulse möglich. Durch Verwendung von weißem Laserlicht, dass aus rotem, grünem und blauem Licht zusammengesetzt wird, ist auch die Aufnahme von farbigen 3D-Modellen möglich [43].

Zur Detektion der reflektierten Strahlen wird in den meisten Fällen eine Kamera verwendet. Bei Verwendung von Punktlasern sind aber auch einfache Zeilensensoren möglich. Durch verschiedene Methoden lässt sich die Auflösung des Detektors bei der Lokalisierung des reflektierten Lichtes auf weniger als ein Pixel verbessern [44].

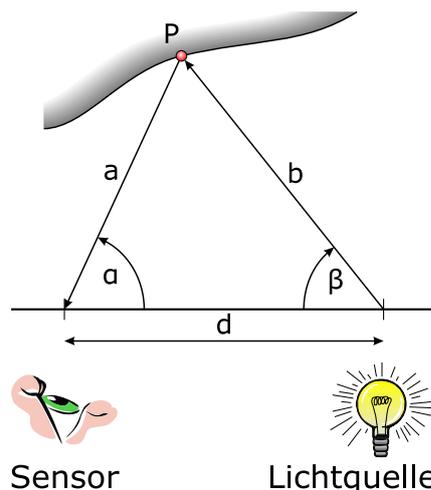


Abbildung 2.7: Das Triangulations-Prinzip: Bei bekanntem Abstand d von Lichtquelle und Sensor und bekanntem Winkel β wird der Winkel α vom Sensor ermittelt. Mit Hilfe der beiden Winkel und der Länge d lässt sich die Distanz zum angepeilten Punkt ermitteln.

Durch Triangulation lassen sich allerdings immer nur die Abstände zu jenen Punkten ermitteln, die durch das projizierte Muster ausgeleuchtet werden. Besonders bei der Verwendung von nur einem einzigen Punkt ist klar, dass dadurch noch kein dreidimensionales Bild des gesamten Objektes möglich ist. Aus diesem Grund muss das Muster bewegt und das Objekt somit abgetastet werden. Genauer gesagt muss sich das Muster relativ zum Objekt bewegen, wodurch drei Möglichkeiten zur Bewegung entstehen.

Die erste Möglichkeit besteht darin, lediglich das Muster zu bewegen. Diese Methode wird meist bei der Verwendung von Punktlasern angewandt, wobei in diesem Fall der Laserstrahl durch bewegliche Spiegel abgelenkt und somit bewegt wird. In diesem Fall verändern sich die Winkel für die Triangulation ständig, wodurch die Berechnung etwas aufwendiger wird.

Bei der zweiten Möglichkeit wird der gesamte Sensor bewegt, wobei der Winkel des Laserstrahls gleich bleibt. Der Nachteil dieser Methode liegt allerdings im erhöhten Kraftaufwand, der für das Bewegen des Sensors notwendig ist. Als Vorteil ergibt sich allerdings für den Sensor ein erhöhter Sichtbereich, wie später in Kapitel 3.3 bei der Beschreibung der Schwenkeinheit noch erläutert wird.

Die dritte Möglichkeit ist ähnlich der zweiten, mit dem Unterschied, dass nun das Objekt bewegt wird. Auf den ersten Blick erscheint diese Methode gerade bei schweren Objekten als sehr umständlich, zumal der Kraftaufwand jetzt nochmals größer ist. Es gibt aber Fälle, in denen diese Methode dennoch Sinn macht. Besonders in der Industrie – bei der Vermessung von Objekten auf Förderbändern – wird diese Methode gerne eingesetzt, da hier die Bewegung ja bereits durch das Förderband erfolgt.

Durch die Abtastung entstehen – zumindest bei der Verwendung von Linien als Muster – mehrere Schnitte, womit sich die Bezeichnung Lichtschnittverfahren erklärt. Erst nachdem die einzelnen Schnitte zusammengesetzt wurden ergibt sich eine dreidimensionale Darstellung der Objekte.

2.3.2 Lichtschnittverfahren mit Punktlasern

Am Anfang der Entwicklung von Scannern nach dem Lichtschnittverfahren diente als Lichtquelle meist ein Punktlaser. Als Detektor dient anstelle einer Kamera ein Zeilensensor beziehungsweise früher eine LEP (Lateral Effect Photodiode). Zur Abtastung muss der Laserstrahl nun in zwei Richtungen bewegt werden, sofern nicht das Objekt selbst bewegt wird, was bei der Beschreibung aller Verfahren vorausgesetzt wird. Dabei wird der Laserstrahl üblicherweise durch Spiegel entsprechend abgelenkt. Bei so genannten synchronen Scannern wird der Strahl des reflektierten Lichts synchron abgelenkt (Abbildung 2.8a auf der nächsten Seite) wodurch der Scanner kleiner wird [45, 46].

2.3.3 Lichtschnittverfahren mit Linienlasern

Durch die Verfügbarkeit von Optiken, die einen Laserstrahl zu einer Ebene aufweiten können, wurden verstärkt Systeme mit einem solchen Linienlaser entwickelt. Die Verwendung eines Linienlasers bringt auch den Vorteil mit sich, dass der Laserstrahl nun nur mehr in eine

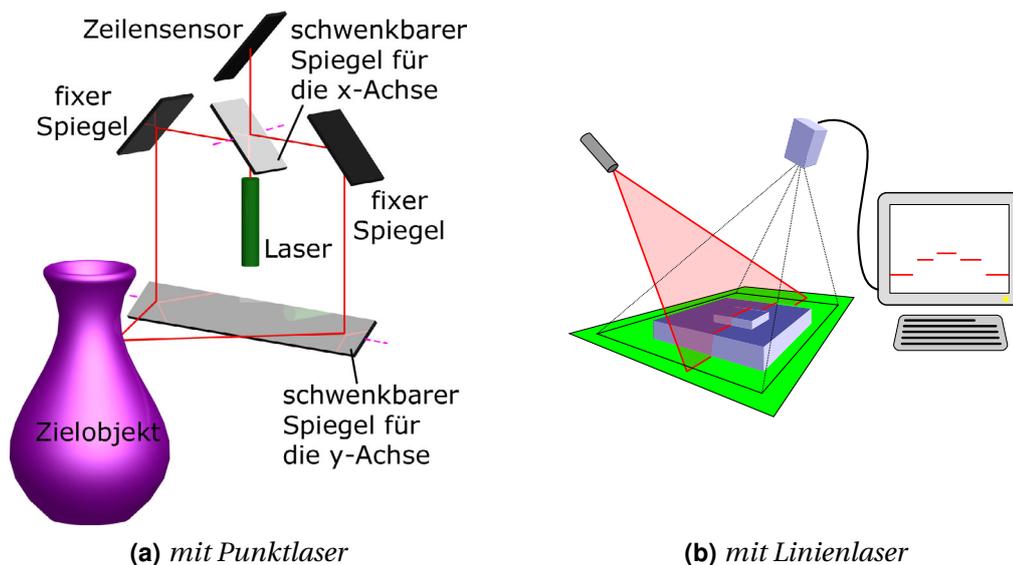


Abbildung 2.8: Funktionsprinzip des Lichtschnittverfahrens: (a) Bei der Verwendung eines Punktlasers muss das Zielobjekt in x - und y -Richtung abgetastet werden. Dafür wird der Laserstrahl durch Spiegel in zwei Ebenen abgelenkt. Ein Zeilensensor erfasst den reflektierten Strahl und gibt die Position an die nachfolgende Berechnung weiter. (b) Bei der Verwendung eines Linienlasers nimmt eine Kamera einen kompletten Lichtschnitt auf, der vom Computer weiterverarbeitet wird. Die Abtastung hat nur mehr in einer Richtung zu erfolgen und kann daher auch schneller bewerkstelligt werden.

Richtung abgelenkt werden muss. Dadurch entfallen zumindest jene mechanischen Komponenten die für die Ablenkung in die zweite Richtung notwendig wären, was vor allem eine Gewichts- und Platzersparnis bewirkt.

Im Gegenzug dafür muss aber der Zeilensensor durch einen zweidimensionalen Photosensor einer Kamera ersetzt werden. Die Kamera nimmt nun das Bild der projizierten Linie auf, aus dem vom Computer durch Triangulation das Profil des einen Lichtschnittes errechnet wird. Durch Bewegen des Lasers oder des gesamten Systems können mehrere Profile aufgenommen und zu einem Gesamtbild kombiniert werden. In Abbildung 2.8b ist der prinzipielle Aufbau eines solchen Laserscanners dargestellt.

2.3.4 Verwendung von Mustern und Moiré-Effekt

Es gibt für Laser auch optische Aufsätze, die mehrere parallele Linien oder andere Muster erzeugen. Werden gleich mehrere parallele Linien auf die zu scannende Objekte projiziert, so kann das mechanische Bewegen entfallen. Ein weiterer Vorteil in diesem Fall ist, dass eine einzige Aufnahme genügt. Da die Anzahl der Linien, die eine solche Optik erzeugen kann, aber nicht sonderlich hoch ist – sie liegt im Normalfall bei 10–20 Linien – lässt sich so jedoch kein besonders detailliertes Tiefenbild erzeugen.

Für bessere Ergebnisse wird daher gerne ein Muster mit Hilfe von Projektoren auf die Objekte projiziert. Dieses Muster kann ein einfaches Linienmuster sein, es kommen aber auch verschiedene veränderliche Muster zum Einsatz [47–50]. Für Objekte mit geringen Tiefenunterschieden wird zusätzlich der Moiré-Effekt ausgenutzt [51, 52]. Als Detektor wird wie schon beim einfachen Linienlaser eine Kamera verwendet, weshalb der Aufbau vom Prinzip her gleich bleibt, lediglich die für die Bewegung erforderlichen Komponenten entfallen gänzlich.

2.4 Das Stereoverfahren

Ein weiteres Verfahren zur Aufnahme von Tiefenbildern ist das Stereoverfahren. Beim Stereoverfahren handelt es sich im Gegensatz zum Lichtschnittverfahren um ein passives, optisches Verfahren. Das heißt, es werden nur Kameras, ohne eine zusätzlichen Beleuchtung verwendet³.

Die Tiefeninformationen werden bei diesem Verfahren aus zwei Kamerabildern ermittelt. Dazu werden Bilder von zwei Kameras synchron aufgenommen⁴, um ähnlich wie bei uns Menschen die Umgebung mit zwei „Augen“ wahrzunehmen.

Nach der Aufnahme der beiden Bilder erfolgt die Generierung des Tiefenbildes in folgenden Schritten:

1. *Entzerren*: Mathematische Korrektur der radialen und tangentialen Verzeichnung (vgl. Kapitel 2.1.4).
2. *Rektifizierung*: Winkel- und Lageabweichungen der Kameras werden mathematisch berichtigt und die Bildzeilen werden ausgerichtet.
3. *Korrespondenzsuche*: In beiden Bildern wird nach gleichen Punkten gesucht und der Lageunterschied der Punkte gespeichert.
4. *Tiefenbildgenerierung*: Aus den zuvor ermittelten Abweichungen wird das Tiefenbild durch Triangulation generiert.

In den nun folgenden Kapiteln werden diese Schritte genauer beschrieben, wobei die Reihenfolge aber aus Gründen der Verständlichkeit geändert wird.

2.4.1 Triangulation

Im Falle des Stereoverfahrens sind die mathematischen Zusammenhänge bei der Triangulation sehr einfach, sofern die Anordnung der Kameras in perfekter Art und Weise erfolgt. Eine solche perfekte Anordnung setzt voraus, dass die optischen Achsen der beiden Kameras exakt parallel verlaufen und dass die Bildebenen exakt koplanar sind. Zusätzlich muss der Offset zwischen Bild- und Kamerakoordinatensystem (vgl. Kapitel 2.1.3) beider Kameras exakt gleich groß sein. Als weitere Bedingung müssen die einzelnen Bildzeilen der Kameras exakt auf einer Linie liegen.

Unter diesen Voraussetzungen lässt sich die Triangulation für einen Punkt P aus der realen Welt, der auf den Bildsensoren beider Kameras abgebildet wird, nach Abbildung 2.9 auf der nächsten Seite berechnen. Werden die horizontalen Koordinaten des abgebildeten Punktes mit x^l für das linke respektive x^r für das rechte Bild bezeichnet, so kann der Unterschied beider Punkte als einfache Differenz $d = x^l - x^r$ angeschrieben werden. Im Zusammenhang mit Stereoverfahren wird diese Differenz auch als Disparität oder Deviation

³ Für bessere Aufnahmen bei schlechter Beleuchtung können selbstverständlich auch zusätzliche Lichter verwendet werden, um das Ergebnis zu verbessern.

⁴ Es gibt auch Verfahren, die mit mehr als zwei Kameras arbeiten und solche mit nur einer Kamera [53]. Bei Verfahren mit nur einer Kamera muss die Kamera bewegt werden, um eine zweite Ansicht zu erhalten, wodurch diese nicht für bewegliche Objekte geeignet sind.

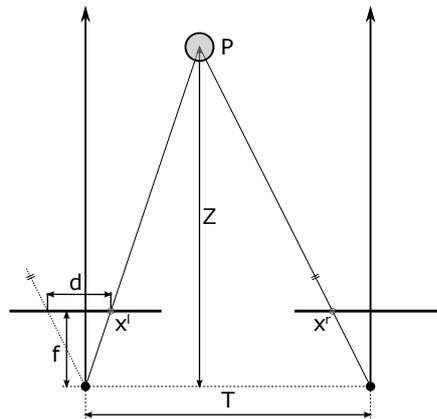


Abbildung 2.9: Triangulation beim Stereoverfahren: Bei koplanar angeordneten Bildsensoren mit parallelen, optischen Achsen kann die Tiefe eines Punktes bei bekanntem Kameraabstand T und bekannter Brennweite f aus der Differenz d der Bildpunkte berechnet werden.

bezeichnet. Nach Abbildung 2.9 lässt sich unter Anwendung des Kongruenzsatzes für Dreiecke mit dem Kameraabstand T und der Brennweite f die gesuchte Tiefe Z berechnen:

$$Z = \frac{Tf}{d}. \quad (2.9)$$

Da die Tiefe verkehrt proportional zur Disparität ist, ergibt sich ein wichtiger Zusammenhang. Bei einer sehr kleinen Disparität ändert sich die Tiefe bei kleinen Schwankungen der Disparität sehr stark, während bei einer großen Disparität eine Schwankung kaum einen Einfluss auf die Tiefe hat. Dadurch ist die Tiefenauflösung von Stereokamerasystemen entfernungsabhängig und umso besser, je näher das Objekt an den Kameras ist. Dieser Zusammenhang ist in Abbildung 2.10 dargestellt.

Mit Hilfe der hier beschriebenen Triangulation kann ein Tiefenbild aus einzelnen, ermittelten Disparitäten ermittelt werden. Deshalb werden im vorhergehenden Schritt bei der Suche nach Übereinstimmungen für alle gefundenen Punkte die Disparität ermittelt und in einer so genannten *Disparity-Map* gespeichert.

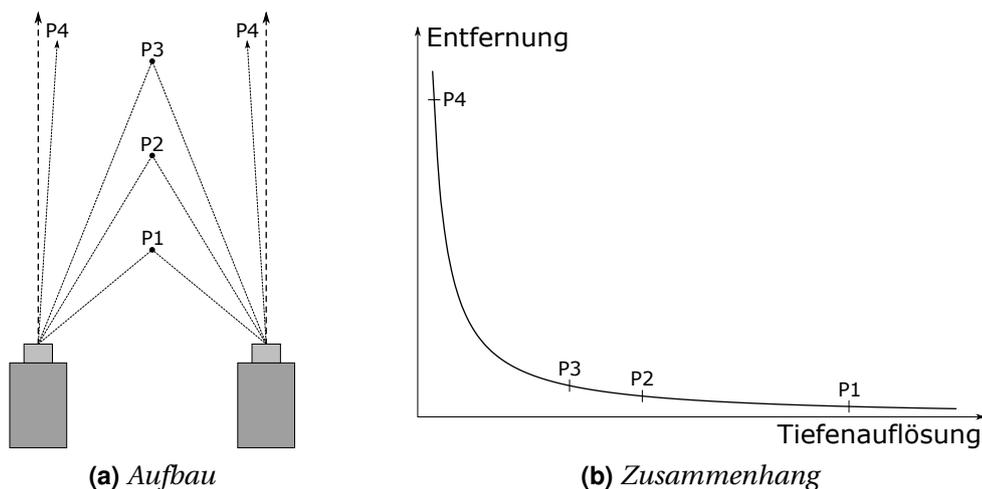


Abbildung 2.10: Zusammenhang von Tiefenauflösung und Entfernung: Da sich die Tiefe verkehrt proportional zur Disparität – dem räumliche Unterschied der beiden Kamera-Bildpunkte – verhält, ist die Tiefenauflösung dort besser, wo eine kleine Tiefenänderung bereits eine große Änderung der Disparität hervorruft. Diese Änderung ist umso größer, je näher ein Punkt an den Kameras ist.

Der Umstand, dass sich die hier beschriebene perfekte Anordnung der Kameras nie erreichen lässt, ist der Grund für den Arbeitsschritt der *Rektifizierung*, der im Anschluss an eine Einführung in die Epipolargeometrie erläutert wird.

2.4.2 Epipolargeometrie

Die Epipolargeometrie kombiniert zwei Lochkamera-Modelle und erweitert diese um zusätzliche Aspekte. Ein vor den beiden Kameras liegender Punkt spannt mit den beiden optischen Zentren eine Ebene – die *Epipolarebene* – auf. Die Verbindung der beiden optischen Zentren erzeugt einen neuen Punkt – den *Epipol* – in beiden Bildebenen, der das Abbild des optischen Zentrums der jeweils anderen Kamera darstellt. Der Epipol wiederum bildet mit dem Bildpunkt die *Epipolarlinie*. In Abbildung 2.11 sind diese Zusammenhänge dargestellt.

Das besondere an der Epipolarlinie ist, dass die Bilder aller Punkte, die in derselben Epipolarebene liegen, auch auf der selben Epipolarlinie liegen. Somit kann für einen, im linken Bild gefundenen Punkt, die Entsprechung im rechten Bild nur auf der Epipolarlinie liegen. Mit dieser Einschränkung lässt sich die Suche nach übereinstimmenden Bildpunkten erheblich erleichtern und auch verbessern, da falsche Übereinstimmungen schwerer möglich sind. Dieser Vorteil kann jedoch nur dann genutzt werden, wenn die Epipolarlinien bekannt sind.

Für den im vorherigen Kapitel beschriebenen Idealfall einer perfekten Anordnung der Kameras nehmen die Epipolarlinien eine besondere Form an, sie sind allesamt parallele Linien und liegen in den Bildern auf gleicher Höhe. Das heißt, dass die Epipolarlinien mit den Bildzeilen identisch sind. Somit erklärt sich, weshalb eine Rektifizierung der Bilder erforderlich ist.

2.4.3 Rektifizierung

Zum leichteren Finden von Übereinstimmungen in den beiden Bildern und zur einfacheren Triangulation sollten die Bilder einem idealen Modell der Kameraanordnung angepasst werden. Die Bildebenen sollen also auf mathematischem Weg, wie in Abbildung 2.12 auf der nächsten Seite dargestellt, durch Rotation und Translation in die Idealform gebracht werden. Ziel ist es, dass die Bildebenen koplanar sind und dass die einzelnen Bildzeilen auf gleicher Höhe liegen.

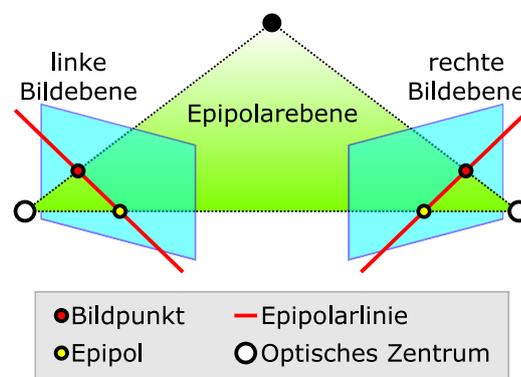


Abbildung 2.11: Begriffe der Epipolargeometrie

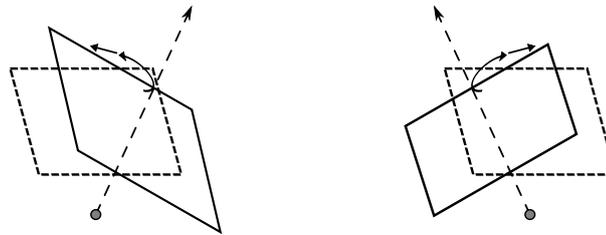


Abbildung 2.12: Ziel der Rektifizierung: Durch das Rektifizieren der Bildern sollen die Bildebenen auf mathematischem Weg koplanar ausgerichtet werden. Durch Rotation und Translation wird die ursprüngliche Bildebene (durchgezogen dargestellt) in eine ideale Ebene (strichliert dargestellt) überführt.

Für die Rektifizierung der Bilder existieren verschiedene Algorithmen. Die quelloffene Programmbibliothek *OpenCV*⁵ hat dabei zwei Algorithmen implementiert. Der erste Algorithmus basiert auf der Arbeit von Hartley [54]. Hartley's Algorithmus arbeitet mit unkalierten Kamerabildern, was aber den Nachteil mit sich bringt, dass das Ergebnis ein Tiefenbild ohne exakte Größeninformationen ist. Das bedeutet, das Tiefenbild gibt die Form der Objekte zwar korrekt wieder, die tatsächliche Größe ist aber nicht bekannt.

Der zweite implementierte Algorithmus basiert auf den von Bouguet entwickelten und in dessen *MATLAB*[®]-Toolbox [55] verwendeten Algorithmus. Dieser benötigt zwar kalibrierte Bilder, erzeugt dafür aber auch Ergebnisse mit bekannten Abmessungen. Für Fälle, in denen eine Kalibrierung möglich ist, ist dieser Algorithmus meist die bessere Wahl. Bouguet's Algorithmus nutzt die durch die Kalibrierung bekannten extrinsischen Parameter um eine Rotation und Translation der beiden Bildebenen durchzuführen. Dabei wird die Rotation gleichmäßig auf beide Ebenen aufgeteilt, damit die notwendigen Änderungen so gering als möglich sind. In Abbildung 2.13 ist der Weg der Bilder von der Aufnahme bis zum fertig korrigierten Bild dargestellt.

Neben diesen beiden bereits erwähnten Algorithmen gibt es aber auch noch weitere. So wurde bereits 1988 eine Methode zur Rektifizierung von Bildern bei Systemen mit zwei oder drei Kameras vorgestellt [56].

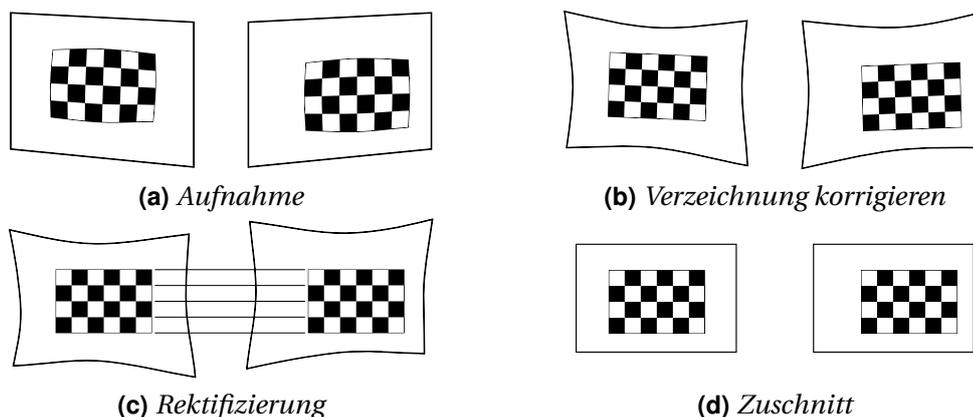


Abbildung 2.13: Korrektur der Bilder beim Stereoverfahren: In den aufgenommenen Bildern (a) sind die Linien des aufgenommenen Schachbrettmusters nicht gerade. Nach Korrektur der Verzeichnung (b) sind diese Linien gerade, das Bild ist allerdings nicht mehr rechteckig. Durch die Rektifizierung der Bilder (c) sind die Zeilen des Schachbrettmusters nun auf einer Linie. Nach dem Zuschneiden des Bildes (d) ist das Bild wieder rechteckig.

⁵ OpenCV ist erhältlich bei SourceForge unter <http://sourceforge.net/projects/opencvlibrary/>.

2.4.4 Korrespondenzsuche

Da für einen Punkt dessen Tiefe nur berechnet werden kann, wenn dieser Punkt in beiden Bildern bekannt ist, müssen für gute Tiefenbilder möglichst viele solche Punkte gefunden werden. Zur Suche nach übereinstimmenden Punkten in den beiden Bildern existieren verschiedene Methoden. Durch die zuvor erfolgte Rektifizierung der Bilder können diese Methoden wichtige Einschränkungen machen. So kann eine Übereinstimmung nur in derselben Bildzeile vorkommen. Zusätzlich gilt noch, dass Punkte im rechten Bild rechts von den zugehörigen Punkten des linken Bildes liegen müssen beziehungsweise auch umgekehrt.

Konolige entwickelte eine Methode, die blockweise nach Übereinstimmungen sucht [57]. Das bedeutet, dass markante Punkte in kleinen Bildblöcken gesucht werden, für die dann im anderen Bild eine Übereinstimmung gesucht wird. Durch die Suche nach markanten Punkten ist die Anzahl der gefundenen Punkte zwar nicht so groß, im Gegenzug dafür werden aber auch weniger falsche Übereinstimmungen gefunden. Die Arbeit gliedert sich dabei in drei Schritte. Im ersten Schritt wird das Bild vorgefiltert um die Suche nach markanten Punkten zu erleichtern. Im nächsten Schritt wird nun entlang der einzelnen Bildzeilen nach übereinstimmenden Blöcken gesucht. Abschließend werden im dritten Schritt, so weit als möglich, falsche Übereinstimmungen wieder entfernt.

Neben solchen blockweisen Methoden gibt es auch noch Methoden, bei denen Pixelweise nach Übereinstimmungen gesucht wird. Im Jahre 1999 beschrieb Birchfield eine solche Methode [58]. Bei einer Pixelweisen Suche nach Übereinstimmungen kommt es jedoch leicht zu falschen Übereinstimmungen. Der Vorteil liegt aber darin, dass mehr Punkte gefunden werden und dadurch die Tiefeninformation sehr gut erhalten bleibt. Im Idealfall werden bei solchen Verfahren sogar alle Punkte gefunden, die tatsächlich in beiden Bildern zu sehen sind. In seiner Arbeit zeigt Birchfield Methoden, um falsche Übereinstimmungen zu vermeiden.

2.5 Der Vergleich der Verfahren

In der Tabelle 2.2 sind die erforderlichen Komponenten für die Minimalconfiguration von Sensoren nach beiden Verfahren aufgelistet. Beim Lichtschnittverfahren sei nochmals darauf hingewiesen, dass zur Aufnahme mehrere Ebenen – wie es für ein dreidimensionales Bild erforderlich ist – unbedingt ein Antrieb nötig ist. Dabei muss der Antrieb allerdings nicht unbedingt Teil des Sensors sein, es kann auch wie in Kapitel 2.3.1 beschrieben das Objekt bewegt werden.

Lichtschnittverfahren	Stereoverfahren
<ul style="list-style-type: none"> • 1 Kamera bzw. Lichtsensor • 1 Lichtquelle z. B. Linienlaser + Antrieb für mehrere Ebenen	<ul style="list-style-type: none"> • 2 Kameras

Tabelle 2.2: Benötigte Komponenten bei der Minimalausführung beider Tiefenbildsensoren im Vergleich.

In Tabelle 2.3 auf der nächsten Seite werden das Lichtschnittverfahren und das Stereoverfahren anhand von vier wichtigen Eigenschaften verglichen. Dabei werden lediglich Unterschiede hervorgehoben, Gemeinsamkeiten sind nicht aufgelistet. Beide Verfahren haben

Eigenschaft	LS	Stereo	Erklärung
Arbeitsweise	aktiv	passiv	Das Lichtschnittverfahren arbeitet deshalb <i>aktiv</i> , da eine Lichtquelle benötigt wird.
Sichtbereich	eine Ebene	der der Kameras	Beim Lichtschnittverfahren kann nur die Lichtebe- ne erfasst werden, beim Stereoverfahren der gesam- te Sichtbereich der Kameras.
Genauigkeit	höher	geringer	Die Genauigkeit ist beim Lichtschnittverfahren des- halb höher, da der Lichtschnitt die zu messenden Punkte vorgibt, während beim Stereoverfahren eine fehleranfällige Suche erforderlich ist.
optimale Lichtverhält- nisse	wenig Licht	viel Licht	Bei wenig Umgebungslicht hebt sich das Laserlicht besser von der Umgebung ab und kann so besser er- kannt werden. Das Stereoverfahren benötigt jedoch gut ausgeleuchtete Bilder und somit mehr Umge- bungslicht.

Tabelle 2.3: Vergleich zwischen Lichtschnitt- und Stereoverfahren. (Die Abkürzung LS in der Tabellen-
überschrift steht dabei für das Lichtschnittverfahren.)

vor allem eines gemeinsam, es sind beides optische Verfahren. Somit haben beide unter an-
derem auch damit Schwierigkeiten, farblose, transparente Objekte wie beispielsweise Gläser
zu erkennen.

3

3 Der Sensor

Ein Tiefenbildsensor kann in unterschiedlichen Ausführungen für unterschiedliche Anwendungen entwickelt werden. Er kann preiswert oder kostspielig sein und Ergebnisse von geringer bis hoher Qualität liefern. Dieses Kapitel beschreibt und erklärt die Einschränkungen, die der Entwicklung des Sensors zu Grunde liegen. Zusätzlich werden Probleme, die bei solchen Tiefenbildsensoren auftreten, mit deren Lösung beschrieben. Mit Ende dieses Kapitels sind der prinzipielle Aufbau und die Funktionsweise des Sensors festgelegt.

3.1 Das Einsatzgebiet

Der Sensor soll – befestigt an einem mobilen Roboter – zur Erfassung von Tischszenen im Haushalt zum Einsatz kommen.

Das Einsatzgebiet des Sensors befindet sich also im Bereich der mobilen Robotik. Dieser Bereich stellt, wie bereits in Kapitel 1.5 beschrieben, besondere Anforderungen an die verwendeten Komponenten. Aufgrund der Größe des Forschungsbereichs der mobilen Robotik muss das Einsatzgebiet noch weiter eingegrenzt werden, umfasst der Bereich doch alle mobilen Roboter, vom einfachen Roboterstaubsauger bis hin zum hochentwickelten Forschungsroboter zur Erkundung des Weltalls.

Der Sensor soll im so genannten Homecare-Bereich zum Einsatz kommen. Ausgerüstet mit dem Sensor soll ein Roboter den Menschen im Haushalt unterstützend zur Seite stehen. So soll der Roboter unter anderem dazu in der Lage sein, Kaffee zu servieren, oder einfach nur dem Benutzer die passende Fernbedienung zu bringen. Der Sensor soll dabei dem Roboter eine Erfassung der nahen Umgebung ermöglichen, damit die benötigten Gegenstände, wie die Kaffeetasse oder die Fernbedienung, lokalisiert werden können. Es soll, wenn sich der Roboter direkt vor einem Tisch befindet, ein Tiefenbild der Tischplatte mit den darauf befindlichen Gegenständen erstellt werden.

3.2 Die Anforderungen

Aus dem Einsatzgebiet und aus anderen, beispielsweise wirtschaftlichen Überlegungen ergeben sich die Anforderungen, die bei der Entwicklung des Sensors eingehalten werden müssen. Diese Anforderungen dienen auch dazu, der Entwicklung Grenzen zu setzen und das Ziel im Wesentlichen festzulegen. Im Folgenden finden sich die festgelegten Anforderungen an den Sensor:

Die Arbeitsweise: Der Sensor soll nach dem Lichtschnittverfahren arbeiten.

Am Institut für Automatisierungs- und Regelungstechnik ist bereits ein Sensor im Einsatz, der nach dem Lichtschnittverfahren arbeitet. Um diesen alten Sensor zu ersetzen, soll auch der neue nach demselben Prinzip arbeiten.

Die Aufnahme von Texturen: Der Sensor soll farbige Texturen aufnehmen können.

Das Lichtschnittverfahren liefert nur 3D-Daten ohne Farb- oder Helligkeitsinformationen. Werden vom Sensor jedoch auch noch farbige Texturen aufgenommen, so lassen sich die 3D-Daten um diese Farbinformationen erweitern. Somit kann der Roboter beispielsweise auch zwischen einer roten und einer blauen Kaffeetasse unterscheiden.

Das Gewicht: Für das Gesamtgewicht des Sensors ist ein Grenzwert von 0,5kg festgelegt, der nicht überschritten werden soll.

Im Bereich der mobilen Robotik spielt das Gewicht eine entscheidende Rolle. Ein höheres Gewicht erhöht die zum Bewegen eines Roboters erforderliche Energie. Da die mobilen Roboter auf eine interne Energieversorgung angewiesen sind, verringert sich durch den erhöhten Energiebedarf die mögliche Einsatzdauer des Roboters.

Der festgelegte Grenzwert begründet sich auch damit, dass der Sensor an einem am Institut vorhandenen Roboterarm befestigt werden soll. Durch den Sensor wird die Tragfähigkeit des Arms weiter reduziert, was unvermeidbar, aber dennoch nicht wünschenswert ist.

Die Größe: Der Sensor soll bei der Hauptabmessung einen Wert von 20cm nicht überschreiten. Für die Abmessungen in den beiden anderen Richtungen gelten Grenzwerte von 10cm.

Beim Lichtschnittverfahren müssen die Kamera und der Laser in einem Abstand voneinander montiert werden, um eine Triangulation zu ermöglichen. Bei einem größeren Abstand wirkt sich ein Tiefenunterschied im Kamerabild stärker aus. Dadurch kann die Auflösung des Verfahrens erhöht werden. Gegen einen großen Abstand spricht jedoch das geplante Einsatzgebiet. Der Sensor darf im Vergleich zum Roboter nicht groß geraten, da die Bewegungsmöglichkeit des Arms auf keinen Fall eingeschränkt werden soll.

Der Sichtbereich: Der erforderliche Sichtbereich hat eine Tiefe von 0,6 bis 1,6 Metern und einen Durchmesser von 1,5 Metern, wobei diese Grenzwerte zur Auswahl der Kameraoptik dienen.

Der Durchmesser des Sichtbereiches ergibt sich aus der Forderung, wonach der Sensor Tischszenen aufnehmen soll. Für einen üblichen Tisch sollte daher ein Durchmesser von 1,5 Metern ausreichend sein.

Entscheidend für die Tiefe des Sichtbereiches ist die geplante Befestigung des Sensors am Roboter. Wie in Abbildung 3.1 auf der nächsten Seite gezeigt wird, soll der Sensor an einem Greifarm befestigt werden. Da die geplante Befestigungsposition rund 60cm von der Hand am Ende des Greifarms entfernt ist, ist es nicht notwendig, den Sichtbereich früher beginnen zu lassen, womit die Untergrenze der Tiefe bei diesen 60cm liegt. Die Obergrenze wird um einen Meter höher angesetzt, da das für eine Tischszene ausreichen sollte und der Sensor nötigenfalls auch durch den Roboterarm näher an die zu erfassenden Objekte gebracht werden kann.

Die Energieversorgung und der Datenanschluss: Sowohl Energieversorgung als auch Datenanschluss sollen über einen einzigen FireWire-Anschluss erfolgen.

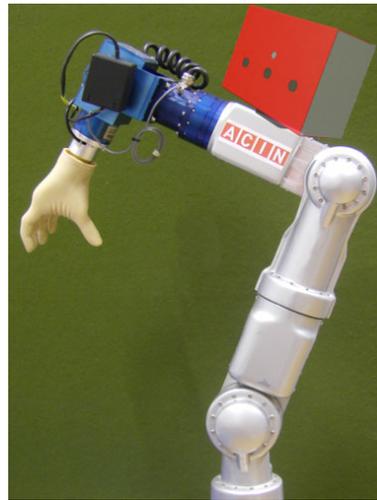


Abbildung 3.1: Die geplante Befestigungsposition des Sensor am Roboterarm

Wie schon bei der Motivation in Kapitel 1.7 erläutert wurde, verursacht die Verwendung untypischer oder gar unterschiedlicher Schnittstellen zum Teil erhebliche Zusatzkosten. Gerade bei der Erforschung neuer Komponenten ist es wichtig, dass diese über Standardschnittstellen verfügen, damit keine zusätzlichen Investitionen für Schnittstellenkarten für PC oder Notebook getätigt werden müssen. Bei Kameras, die ja das Kernstück des Sensors bilden, kommen in der Industrie heute die Schnittstellen CameraLink, FireWire und Gigabit-Ethernet zum Einsatz. Zusätzlich wird im Consumer-Bereich meist noch USB verwendet. Dabei wird bei CameraLink eine eigene Schnittstellenkarte, der so genannte Frame-Grabber, benötigt, während die restlichen Schnittstellen bei modernen PCs und Notebooks schon zum Standard gehören. Aus diesem Grund kommt CameraLink nicht in Frage.

Neben einer Daten-Schnittstelle wird auch eine Energieversorgung benötigt. Um den Verkabelungsaufwand so gering als möglich zu halten, ist es gewünscht, dass die Energieversorgung auch über den Datenanschluss erfolgt. USB und FireWire bieten genau diese Möglichkeit. Bei Ethernet besteht mit Power over Ethernet – kurz PoE – zwar auch die Möglichkeit einer gleichzeitigen Energieversorgung, es werden dafür jedoch eigene Geräte erforderlich.

FireWire bietet im Vergleich zur noch verbleibenden USB-Schnittstelle einige Vorteile. So wurde FireWire speziell für den Video-Bereich entwickelt [59] und ist daher in diesem Bereich USB überlegen. Bei FireWire können beispielsweise mehrere Kameras synchronisiert werden, was beispielsweise beim Stereoverfahren von großem Vorteil ist. Zusätzlich ist die Energieversorgung über FireWire besser möglich als per USB, so sind über FireWire – bei Spannungen von 8–33V und einer maximalen Stromstärke von 1,5A – bis zu 48W an Leistung möglich, während bei USB nur 2,5 Watt bei 5V mit maximal 500mA möglich sind.

Die Eigenbewegung: Die für das Lichtschnittverfahren erforderliche Bewegung hat im Sensor selbst zu erfolgen.

Beim Lichtschnittverfahren wird das Tiefenbild durch die Aufnahme mehrerer Lichtschnitte erzeugt. Dafür muss die Lichtebene bewegt werden. Bei der angedachten Verwendung an einem Roboterarm wäre es grundsätzlich möglich, diese Bewegung vom Roboterarm ausführen zu lassen. In diesem Fall müsste die Bewegung des Roboters bei der Berechnung der Tiefenbilder durch Koordinatentransformationen berücksich-

tigt werden. Eine Schwierigkeit dabei besteht aber in einer eventuell eingeschränkten Bewegungsmöglichkeit des Roboterarms. Da der Roboterarm zum Bewegen von Gegenständen verwendet werden soll, kann es sein, dass ein in der Hand befindlicher Gegenstand seine Bewegung einschränkt. Hält der Roboterarm beispielsweise eine volle Kaffeetasse in seiner Hand, so darf er sich nicht schnell bewegen und muss die Tasse waagrecht halten.

Was zusätzlich gegen die Verwendung des Roboterarms zur Erzeugung mehrere Lichtschnitte spricht ist, dass dadurch das Einsatzgebiet des Sensors eingeschränkt wird. Wird die erforderliche Bewegung vom Roboterarm ausgeführt, so kann der Sensor nicht ohne den Roboterarm arbeiten.

Aus Platzgründen kommt für die Bewegung nur eine Schwenkbewegung in Frage. Würde sich der Sensor linear bewegen, so müsste er so groß sein, wie die zu erfassende Szene. Das bedeutet im konkreten Fall, dass der Sensor so groß sein müsste, wie ein Tisch breit ist.

Der Preis: Als Preisobergrenze ist ein Betrag von € 500 angesetzt.

Eine allgemeine Einschränkung für diese Diplomarbeit war, einen preiswerten Sensor zu entwickeln, weshalb die Preisobergrenze bei € 500 gesetzt wurde. Mit der Information der Kamerapreise erfolgt eine Abschätzung der Kosten für die anderen Komponenten entsprechend deren technischer Komplexität.

Die Festlegung einer so niedrigen Preisobergrenze erfolgte unter anderem deshalb, weil gezeigt werden soll, was mit bescheidenen, finanziellen Mitteln möglich ist. Bei Tiefenbildsensoren für den industriellen Einsatz versuchen sich die Hersteller gegenseitig mit immer genaueren Modellen zu übertrumpfen. Dabei sind diese Sensoren nicht unbedingt preiswert, so liegt beispielsweise der Preis für die günstigste 3D-Kamera der Firma SICK IVP aus der Serie Ranger bei € 3000¹ ohne Software und Zubehör.

Eine weitere Begründung für die niedrige Preisgrenze liegt in einer eventuellen Verwendung des Sensors bei zukünftigen Haushaltsrobotern. In diesem Segment werden sich Roboter wohl erst so richtig durchsetzen können, wenn die Kosten in einem angemessenen Bereich liegen.

Mit den hier aufgelisteten Anforderungen kann ein Grundkonzept für den Sensor erstellt werden, aus dem sich – unter Einbeziehung weiterer Überlegungen – schlussendlich der Aufbau des Sensors festlegen lässt.

3.3 Die Schwenkeinheit

Für das Lichtschnittverfahren muss der Linienlaser, wie aus den Anforderungen hervorgeht, geschwenkt werden, damit mehrere Schnitte aufgenommen werden können. Ein Schnittprofil für sich gibt lediglich Auskunft über die Entfernungen zu den, auf diesem Profil liegenden Punkte. Erst durch die Kombination mehrere solcher Schnittprofile entsteht ein dreidimensionales Tiefenbild. Für die Schwenkung kommen zwei Möglichkeiten in Frage, die nun näher erläutert werden.

¹ Stand: Jänner 2010

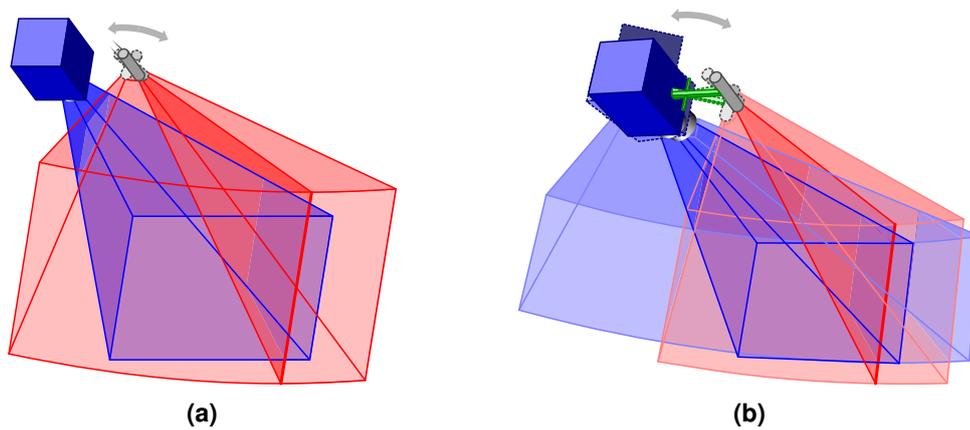


Abbildung 3.2: Sichtbereich beim Schwenken des Laser: (a) Wird nur der Laser geschwenkt, so entspricht der Sichtbereich des Sensors jenem der Kamera. (b) Werden Kamera und Laser gemeinsam geschwenkt, so ergibt sich ein größerer Sichtbereich.

Die beiden Möglichkeiten unterscheiden sich darin, dass in einem Fall nur der Laser geschwenkt wird, während im anderen Fall Laser und Kamera gemeinsam geschwenkt werden. Die beiden unterschiedlichen Methoden ermöglichen unterschiedliche Sichtbereiche, wie in Abbildung 3.2 dargestellt.

Steht die Kamera still, während der Laser geschwenkt wird, so tritt ab einem bestimmten Winkel die Laserebene aus dem Sichtbereich der Kamera aus. Dadurch ist der Sichtbereich des Sensors gleich jenem der Kamera. Dieses Verhalten ist in Abbildung 3.2a illustriert. Der Vorteil dieses Verfahrens liegt jedoch darin, dass eine geringere Masse bewegt werden muss, wodurch die Schwenkeinheit weniger Drehmoment und damit auch weniger Energie benötigt.

Wird nun im Gegensatz dazu die Kamera mit dem Laser um eine gemeinsam Achse geschwenkt, so ist dafür ein höheres Drehmoment erforderlich. Im Gegenzug dafür erhöht sich allerdings der Sichtbereich des Sensors, was in Abbildung 3.2b dargestellt ist. Zusätzlich ist dadurch auch sichergestellt, dass die Laserlinie den Sichtbereich der Kamera nicht wieder verlässt.

Nachdem nun zwei Möglichkeiten zur Auswahl standen, galt es eine davon auszuwählen. Die Wahl fiel auf die Methode, bei der die Kamera mit geschwenkt wird und somit auf den größeren Sichtbereich. Da das erlaubte Maximalgewicht des Sensors ohnehin sehr gering ist, ist der fürs Schwenken der Kamera zusätzlich erforderliche Energiebedarf zu vernachlässigen.

3.4 Der Laser- und Kameraschatten

Schatten sind ein Phänomen, das bei allen optischen Methoden auftritt. Dieses Kapitel beschreibt, inwieweit Schatten beim Lichtschnittverfahren ein Problem darstellen und was unter Laser- beziehungsweise Kameraschatten zu verstehen ist. Dabei werden auch Vorschläge gemacht um den Einfluss des Schattens abzuschwächen.

Im Allgemeinen wird als Schatten ein Bereich bezeichnet, der von der vorhandenen Lichtquelle nicht ausgeleuchtet wird, da ein lichtundurchlässiges Objekt dieses verhindert.

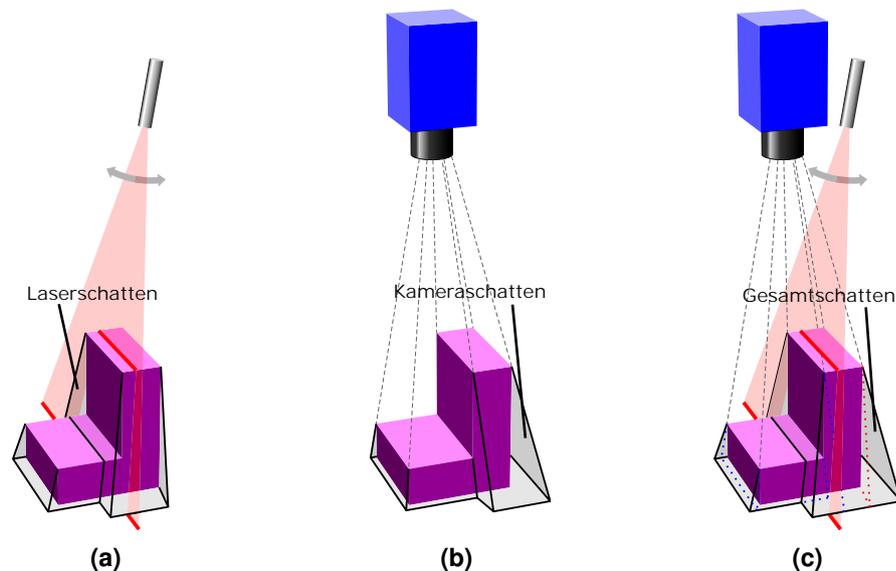


Abbildung 3.3: Laser- und Kameraschatten: (a) Jener Bereich, der vom Laser nicht ausgeleuchtet werden kann, wird als Laserschatten bezeichnet. (b) Als Kameraschatten wird jener Bereich bezeichnet, der von der Kamera nicht erfasst werden kann, da er von einem Objekt verdeckt wird. (c) Das Lichtschnittverfahren arbeitet nur in jenen Bereichen, die weder im Laser- noch im Kameraschatten liegen. Somit entsteht ein größerer Gesamtschatten.

Dementsprechend ist der Laserschatten der Bereich, der vom Laserlicht nicht erreicht werden kann. Der Laserschatten führt beim Lichtschnittverfahren dazu, dass Bereiche vom Laser trotz Schwenken nicht ausgeleuchtet werden (Abbildung 3.3a).

Der Kameraschatten ist – anders als beim Laserschatten – kein Bereich, der nicht ausgeleuchtet wird, sondern jener Bereich, von dem kein Licht zur Kamera gelangen kann, obwohl dieser grundsätzlich im Sichtbereich der Kamera liegt. Einfach ausgedrückt ist es jener Bereich, der am Kamerabild nicht zu sehen ist, da er von einem Objekt verdeckt wird (Abbildung 3.3b).

Nachdem sich Schatten nicht vermeiden lassen, stellen sie für sich betrachtet kein Problem dar. Beim Lichtschnittverfahren macht sich jedoch die Kombination von Laser- und Kameraschatten negativ bemerkbar. Das Lichtschnittverfahren kann nur solche Punkte erfassen, die vom Laser beleuchtet und gleichzeitig auch im Sichtbereich der Kamera liegen werden. Das bedeutet, dass nur die Bereiche erfassbar sind, die in keinem der beiden Schatten fallen. Nachdem für die Triangulation ein Abstand zwischen Laser und Kamera sein muss, sind deren Schattenbereiche unterschiedlich. Somit entsteht ein Gesamtschatten aus der Kombination der beiden einzelnen Schatten, der größer ist als die beiden einzelnen Schatten (Abbildung 3.3c).

Der vergrößerte Gesamtschatten hat Auswirkung auf die Erfassung von Objekten. Die Kamera kann nur sichtbare Oberflächen erfassen, weshalb der im Schatten liegende Bereich nur geschätzt werden kann. Das einfache Objekt aus Abbildung 3.4a auf der nächsten Seite könnte auch die Form des in Abbildung 3.4b dargestellten haben und trotzdem das gleiche Kamerabild liefern. Kommt nun noch der Laserschatten hinzu und ist – wie beim Lichtschnittverfahren – nur der Bereich erfassbar, der in keinen der beiden Schatten fällt, so lassen sich noch weniger Details des Objektes erfassen. Das Objekt könnte jetzt auch die Form des in Abbildung 3.4c dargestellten Objektes aufweisen.

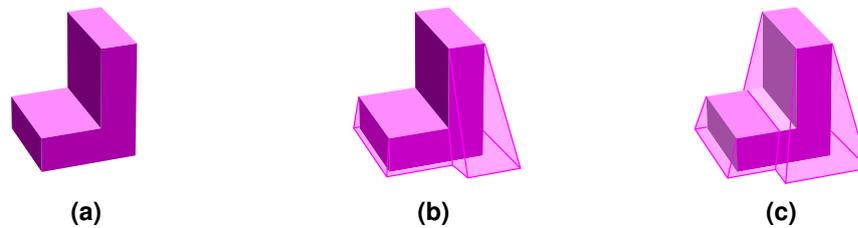


Abbildung 3.4: Die Auswirkung von Schatten auf die Erkennung von Objekten: Das eigentliche Objekt (a) erscheint durch den Kamerashatten erweitert (b). Durch den Laserschatten erscheint das Objekt zusätzlich erweitert (c).

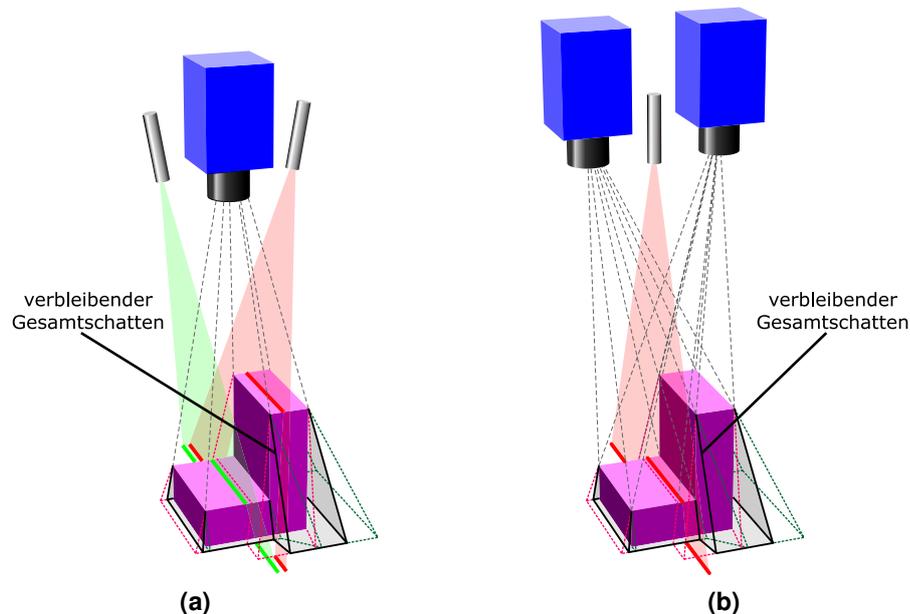


Abbildung 3.5: Möglichkeiten zur Verringerung des Schattens: Zwei Linienlaser (a) verringern den Gesamtschatten auf die Größe des Kamerashattens. Auch bei der Verwendung von zwei Kameras (b) wird der Schatten ebenso verringert.

Werden diese Bereiche nun durch einen zusätzlichen Laser ausgeleuchtet, oder wird eine zusätzliche Kamera installiert, so lässt sich der Gesamtschatten wieder verringern. Voraussetzung dafür ist, dass der zusätzliche Laser oder die zusätzliche Kamera richtig positioniert wird. Eine Verbesserung ist nur möglich, wenn durch die zusätzliche Komponente ein zusätzlicher Bereich ausgeleuchtet oder aufgenommen wird. Das bedeutet, ein zusätzlicher Laser muss auf der gegenüberliegenden Seite der Kamera positioniert werden. Für eine zusätzliche Kamera gilt dasselbe in Bezug auf den Laser. Diese beiden Möglichkeiten zur Verringerung des Gesamtschattens sind in Abbildung 3.5 dargestellt.

3.5 Das Stereosehen mit zwei Kameras

Zur Vergrößerung des Sichtbereiches eines, nach dem Lichtschnittverfahren arbeitenden Sensors, kann die Grundkonfiguration aus Kamera und Laser erweitert werden. Für die Größe des erweiterten Sichtbereichs ist es, bei gleicher räumlicher Anordnung, unerheblich, ob nun die Grundkonfiguration um eine zusätzliche Kamera oder einen zusätzlichen Laser erweitert wird. Die Wahl hat aber Auswirkung auf Preis und Gewicht des Sensors. Ein Linienlaser ist normalerweise günstiger und leichter als eine Kamera, weshalb einem zusätzlichen

Laser den Vorzug zu geben wäre. Dennoch wurden die Konfiguration mit zwei Kameras anstatt mit zwei Lasern gewählt.

Durch die beiden Kameras ist es möglich, den Sensor nicht nur nach dem Lichtschnittverfahren, sondern auch nach dem Stereoverfahren arbeiten zu lassen. Die Entscheidung für diese Konfiguration wurde getroffen, nachdem die Preise für die Kameras bekannt wurden. Nachdem diese günstiger waren als erwartet, war die Idee, einen Sensor zu bauen, der nach beiden Verfahren arbeitet, zu verlockend, zumal das Gewicht der Kamera vergleichsweise gering ist. Somit ist es möglich, den Sensor für unterschiedliche Einsatzzwecke zu verwenden. Außerdem können mit ein und demselben Sensor zwei verschiedene Methoden der Tiefenbildaufnahme entwickelt und erprobt werden.

Zwei Kameras bieten also die Möglichkeit, das Stereoverfahren und das Lichtschnittverfahren zu verwenden und verbessern gleichzeitig noch den Sichtbereich des Sensors beim Lichtschnittverfahren.

3.6 Die Aufnahme einer Textur

Tiefenbildverfahren liefern nur die räumlichen 3D-Daten der Umgebung ohne jeglicher Farb- oder auch nur Helligkeitsinformationen. Werden diese Informationen jedoch benötigt, so muss die so genannte Textur aus den Kamerabildern ermittelt werden.

Sowohl Lichtschnitt- als auch Stereoverfahren liefern lediglich dreidimensionalen Ortsinformationen über den Sichtbereich. Sie erzeugen eine so genannte Punktwolke, in der für jeden Punkt die Koordinaten gespeichert sind. Farb- oder Grauwertinformationen müssen zusätzlich aus einem (oder aus mehreren) Kamerabild entnommen werden, um sie der dreidimensionalen Darstellung der Punktwolke zu überlagern. In diesem Zusammenhang werden die Farb- oder Grauwertinformationen als Textur bezeichnet. Da beide Tiefenbildverfahren mit mindestens einer Kamera arbeiten, sollte die Aufnahme der Textur leicht möglich sein.

Grundsätzlich ist es nicht sonderlich aufwendig, aus den ohnedies schon vorhandenen Kamerabildern eine Textur zu erzeugen. Doch besonders dann, wenn farbige Texturen verwendet werden sollen, ergeben sich aufs Neue einige Schwierigkeiten.

Zum Ermitteln der 3D-Daten werden keine Farbbilder benötigt, können aber selbstverständlich dennoch verwendet werden, wobei sie eventuell zuerst in Grauwertbilder umgewandelt werden müssen. Für farbige Texturen könnten daher einfach Farbkameras anstelle von Monochromkameras verwendet werden. Farbkameras liefern allerdings, wie schon in Kapitel 2.1.2 beschrieben wurde – im Vergleich zu Monochromkameras – Bilder von schlechterer Qualität. Die Bildqualität hat jedoch einen hohen Einfluss auf die Qualität der daraus gewonnenen 3D-Daten. Werden nun anstelle von Monochromkameras Farbkameras verwendet, so muss für gleiche Qualität der 3D-Daten noch zusätzlich in hochwertigere Kameras investiert werden.

Eine Möglichkeit, gute Qualität und geringen Preis zu vereinen, besteht darin, für die eigentliche Tiefenbildgewinnung Monochromkameras zu verwenden und eine zusätzliche Farbkamera zur Aufnahme der Texturen einzusetzen. Ob diese Methode tatsächlich preiswerter ist, hängt von den Kosten der Kameras ab. Wie später noch bei der Auswahl der Komponenten in Tabelle 4.1 auf Seite 36 zu sehen ist, liegen die Kosten für eine Farb- und zwei

Monochromkameras der Firma Unibrain bei € 317. Für zwei Farbkameras des nächst günstigeren Typs betragen die Kosten bereits € 616, was einen Unterschied von € 299 bedeutet.

Ein Nachteil bei der Verwendung von drei anstatt zwei Kameras liegt im höheren Platzbedarf. Da die Kameras jedoch nicht einmal 6×6cm groß sind, hält sich auch der zusätzliche Platzbedarf in Grenzen. In diesem Fall ist unbedingt den geringeren Kosten den Vorzug zu geben.

3.7 Der grundlegende Aufbau

Nachdem nun in den vorangegangenen Kapiteln mehrere Möglichkeiten für den Aufbau beschrieben wurden, soll nun in diesem Kapitel der grundlegende Aufbau des Sensors zusammenfassend beschrieben werden.

Damit der Sensor nach dem Stereoverfahren arbeiten kann werden zunächst einmal zwei Kameras benötigt. Aus Kosten- und Qualitätsgründen werden dafür Monochromkameras verwendet. Für das Lichtschnittverfahren wird zusätzlich noch ein Linienlaser benötigt. Außerdem muss der Sensor dafür schwenkbar sein. Es wird also eine Schwenkeinheit benötigt, die in der Lage ist, den gesamten Aufbau zu schwenken. Zur zusätzlichen Aufnahme von farbigen Texturen wird eine dritte Kamera – eine Farbkamera – benötigt.

Der gesamte Aufbau ist in Abbildung 3.6 schematisch dargestellt. Genauer festgelegt werden kann der Aufbau erst nach der erfolgten Festlegung auf die einzelnen Komponenten. Mehr dazu gibt es im folgenden Kapitel.

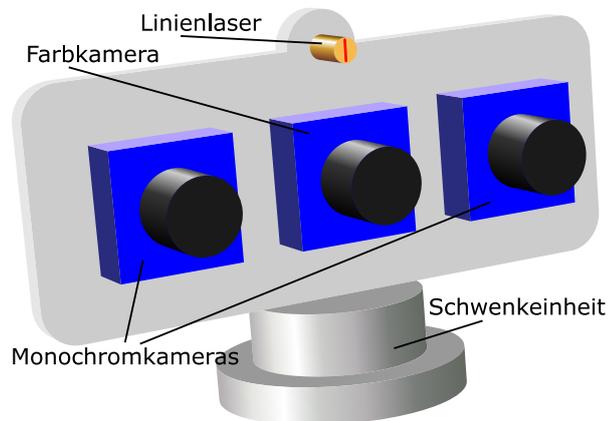


Abbildung 3.6: Schematische Darstellung des Aufbaus

4 Die Komponenten

4.1 Die Kameras

Wie schon beim grundlegenden Aufbau in Kapitel 3.7 beschrieben, werden für den Sensor zwei Monochromkameras und eine Farbkamera benötigt. Bei den Anforderungen in Kapitel 3.2 ist außerdem noch festgelegt, dass der Sensor mit einem einzigen FireWire-Anschluss das Auslangen finden soll. Dadurch wird die Liste der möglichen Kameras schon einmal auf jene, mit einem solchen FireWire-Anschluss eingeschränkt.

Übliche FireWire-Kameras lassen sich in zwei Gruppen einteilen. In der ersten Gruppe sind die FireWire-Webcams, während in der anderen Gruppe die Industriekameras zu finden sind. Webcams sind für den privaten Einsatz am PC konzipiert und haben im Vergleich zu den Industriekameras einen niedrigeren Preis. Diese Webcams besitzen meist ein Kunststoffgehäuse, und lassen sich entweder mit einer Klemme am Monitor befestigen oder haben einen Standfuß, um sie am Tisch zu platzieren. Möglichkeiten, die Kamera mit Schrauben sicher zu befestigen, sind dabei leider nicht vorhanden, wodurch sich diese Kameras nur schlecht für den Sensor eignen. Denn gerade eine gute und sichere Befestigungsmöglichkeit ist für den Sensor sehr wichtig, da ansonst die Qualität der Daten schlechter wird. Das liegt an den verwackelten Bildern, die bei schlechter Befestigung auftreten würden.

Ein weiterer Nachteil von Webcams ist, dass diese in den meisten Fällen nur einen FireWire-Anschluss besitzen. So ist es aber nicht möglich, mehrere Kameras hintereinander, in einer so genannten Daisy-Chain, zu schalten. Dazu wäre ein Hub – ein Verteiler – erforderlich. Hinzu kommt noch, dass sich bei den Webcams die Optik meist nicht tauschen lässt, wodurch der Sichtbereich der Kamera fix vorgegeben ist. Das führt zu einer geringeren Auswahl bei den Kameras, da nur jene in Betracht gezogen werden können, die den geforderten Sichtbereich entsprechen.

Industriekameras hingegen lassen sich problemlos befestigen, sind häufig mit zwei FireWire-Anschlüssen erhältlich und arbeiten fast ausnahmslos mit Wechselobjektiven. Eine Eigenschaft aller Industriekameras ist deren robuste Bauart. Sie besitzen meist metallene Gehäuse, die zum einen sehr widerstandsfähig sind und zum anderen auch für eine gute Befestigungsmöglichkeit verantwortlich sind. Die metallenen Gehäuse sind leider aber auch schwerer als die bei den Webcams üblichen Kunststoffgehäuse. Neben diesem höheren Gewicht ist auch noch der Preis von Nachteil. Im Vergleich mit den Webcams liegt der Preis der Industriekameras deutlich höher, was sich wohl auch mit der qualitativ besseren Bauweise begründen lässt.

Aus Gewichts- und Platzgründen sollte auf einen Hub verzichtet werden, wodurch zwei FireWire-Anschlüsse bei den Kameras sehr wichtig sind. Um den Sensor einfach für andere

Anwendungen mit anderen Sichtbereichen adaptieren zu können, sollte sich das Objektiv wechseln lassen. Mit diesen beiden Forderungen und der Forderung nach einer guten Befestigungsmöglichkeit scheidet die Webcams bereits aus. Somit bleiben nur noch die Industriekameras übrig. Glücklicherweise wurde bei der Suche nach Kameras noch eine weitere Möglichkeit gefunden.

Die Firma Unibrain bietet – neben fertigen Webcams – auch noch OEM¹-Kameramodule an. Solch ein Kameramodul ist lediglich das Innenleben einer Webcam, also nur die Leiterplatte mit den darauf befindlichen elektronischen Bauteilen und der Objektivbefestigung. Dieses Modul bietet auch die Möglichkeit, das Objektiv zu wechseln und verfügt über zwei FireWire-Anschlüsse. Tabelle 4.1 zeigt einen Vergleich dieser Kameramodule mit den preiswertesten Industriekameras.

Hersteller	Bezeichnung	Auflösung	FPS	Farbe	Gewicht	Preis
<i>Unibrain</i>	<i>Fire-i OEM</i>	<i>640×480</i>	<i>30</i>	<i>Farbe</i>	<i>35g</i>	<i>€ 79</i>
<i>Unibrain</i>	<i>Fire-i OEM</i>	<i>640×480</i>	<i>30</i>	<i>Monochrom</i>	<i>35g</i>	<i>€ 119</i>
PHYTEC	FireWire-CAM-002	640×480	30	Farbe	165g	€ 308
PHYTEC	FireWire-CAM-001	640×480	30	Monochrom	165g	€ 338
AVT	Guppy F-036B	752×480	60	Monochrom	50g	€ 490
AVT	Guppy F-036C	752×480	60	Farbe	50g	€ 490
Unibrain	Fire-i Pro XGAc	1024×768	36	Farbe	38g	€ 540
Unibrain	Fire-i Pro XGAb	1024×768	36	Monochrom	38g	€ 540
Sony	XCD-V50CR	640×480	60	Farbe	120g	€ 557
PHYTEC	FireWire-CAM-011H	1024×768	30	Monochrom	180g	€ 590
PHYTEC	FireWire-CAM-012H	1024×768	15	Farbe	180g	€ 590

Tabelle 4.1: Auswahl von in Frage kommenden Kameras. Die gewählten Module sind kursiv hervorgehoben. (Die Abkürzung FPS steht für Frames Per Second und gibt die Bildrate an.)

Tabelle 4.1 zeigt einen Gewichtsvorteil des gewählten Kameramoduls gegenüber den Industriekameras. Ein weit aus größerer Vorteil ist allerdings beim Preis gegeben. Selbst bei den günstigsten Industriekameras liegt der Preis für drei Kameras schon weit über dem gesetzten Maximalkosten von € 500 für den gesamten Sensor. Somit gibt es aus Kostengründen keine Alternative zu den gewählten Kameramodulen. Abbildung 4.1 zeigt ein Bild eines *Unibrain Fire-i OEM* Kameramoduls.



Quelle: Unibrain S.A.

Abbildung 4.1: *Unibrain Fire-i OEM* Kameramodul: Insgesamt drei Stück dieses Kameramoduls werden für den Sensor verwendet.

¹ OEM steht für Original Equipment Manufacturer. OEM-Produkte sind funktionsfähige Baugruppen, die von verschiedenen Firmen in deren eigene Produkte integriert werden können.

4.2 Der Linienlaser

Als Laser kommt nur ein Halbleiterlaser in Frage, wie er zum Beispiel auch bei Laser-Pointer, bei CD-Spielern oder auch bei Lichtshows verwendet wird. Bei Halbleiterlaser – auch Diodenlaser genannt – sind sehr kleine Abmessungen möglich. Zusätzlich benötigen Halbleiterlaser nur geringe elektrische Leistungen, was für die Anwendung beim Sensor von großem Vorteil ist.

Ein Linienlaser ist im Grunde nur ein Punktlaser mit einer aufgesetzten Optik. Für einen Linienlaser ist die Optik eine Zylinderlinse, die den Strahl in eine Richtung aufweitet. Für andere Anwendungsfälle gibt es auch Optiken, die Kreuze, Kreise und andere Muster erzeugen.

Auch für die Wahl des Linienlasers gibt es wieder die allgemeinen Kriterien wie Größe, Preis und Gewicht. Neben diesen Kriterien gibt es allerdings auch einige laserspezifische Auswahlkriterien. Es gibt Laser mit unterschiedlichen Wellenlängen und mit verschiedenen Lichtleistungen. Auch die Höhe der benötigten beziehungsweise der erlaubten Versorgungsspannung ist unterschiedlich. Speziell beim Linienlaser spielt auch der Öffnungswinkel der Strahlaufweitung eine Rolle.

Laserlicht hat grundsätzlich ein sehr enges Frequenzspektrum – also eine bestimmte Farbe – während im weißen Sonnenlicht alle Farben vertreten sind. Aus diesem Grund gibt es Laser auch in unterschiedlichen Farben beziehungsweise mit unterschiedlichen Wellenlängen. Für die Anwendung beim Sensor ist die Farbe nur soweit von Bedeutung, als sie im für die Kamera sichtbaren Frequenzbereich liegen muss. Die CCD-Sensoren der Kameras sind für den Bereich des sichtbaren Lichts und auch noch etwas darüber hinaus empfindlich.

Die Leistung des Lasers ist in zweierlei Hinsicht von Bedeutung. Zum einen bedeutet eine höhere Leistung auch eine erhöhte Lichtintensität, wodurch die projizierte Laserlinie in heller Umgebung besser wahrnehmbar wird. Zum anderen bedeutet eine höhere Lichtleistung auch einen erhöhten Energiebedarf, was eventuell zu Problemen mit der Energieversorgung führen könnte. Der Energiebedarf der in Betracht gezogenen Laser liegt, wie auch in Tabelle 4.2 gezeigt wird, im Bereich von einigen, wenigen Milliwatt, während die Kameras rund 1W benötigen. Aus diesem Grund ist der Energiebedarf des Lasers nicht entscheidend.

Mehr von Bedeutung ist die Art und Höhe der Versorgungsspannung. Da der Sensor als einzigen Anschluss einen FireWire-Anschluss besitzen soll, kann die Versorgung des Lasers auch nur über diesen Anschluss erfolgen. Bei FireWire sind Gleichspannungen im Bereich

Bezeichnung	Spannung	∠	Leistung	Wellenlänge	Preis
<i>LFL650-5-12(9x20)90</i>	2,5–12,5V DC	90°	5mW	650nm	€ 12,90
LFL650-5-12(9x20)60	2,5–12,5V DC	60°	5mW	650nm	€ 12,90
LFL650-5-4.5(15x68)90	2,4–4,5V DC	90°	5mW	650nm	€ 19,90
LFL650-5-4.5(15x68)60	2,4–4,5V DC	60°	5mW	650nm	€ 19,90
LI650-2.5-3(5)(8x26)	2,7–6V DC	90°	2,5mW	650nm	€ 26,50
LH650-2.5-3(5)(12x32)	3 oder 5V DC	10°–110°	2,5mW	650nm	€ 44,40
LH635-4-3(5)(12x32)	3 oder 5V DC	10°–110°	4mW	635nm	€ 48,00
LH670-2.5-3(5)(12x32)	3 oder 5V DC	10°–110°	2,5mW	670nm	€ 48,00

Tabelle 4.2: Liste mit in Betracht gezogenen Linienlasern der Firma Picotronic. Der gewählte Laser ist kursiv hervorgehoben.



Abbildung 4.2: Der Linienlaser: Picotronic LFL650-5-12(9x20)90

von 8–33V möglich. Die üblichen Spannungswerte von Halbleiterlasern liegen unterhalb dieses Bereiches, wodurch die geforderte Höhe der Versorgungsspannung leicht erreichbar ist. Entscheidend im Bezug auf die Versorgungsspannung ist vielmehr die erlaubte Spannungsschwankung. Um Schwierigkeiten zu vermeiden, sollte die Versorgungsspannung Schwankungen im Bereich von rund 1V zulassen. Anderenfalls wäre der Schaltungsaufwand für die in Kapitel 4.4 beschriebene Spannungsversorgung bedeutend höher.

Ein entscheidendes Kriterium ist der Öffnungswinkel des Linienlasers. Für den geforderten Sichtbereich von 1,5m Durchmesser bei einer Tiefe von 0,6m ist ein Öffnungswinkel von rund 103° erforderlich. In Tabelle 4.2 auf der vorherigen Seite ist eine Auswahl von Linienlasern mit den entscheidenden Daten aufgelistet.

Der geforderte Öffnungswinkel von mindestens 103° wird erst von den teureren Modellen erreicht. Aus Kostengründen wurde der *Picotronic LFL650-5-12(9x20)90* (Abbildung 4.2) gewählt, auch wenn dieser nur einen Öffnungswinkel von 90° erreicht. In diesem Fall ist die Laserlinie bei einer Tiefe von 0,6m nur 1,2m statt der geforderten 1,5m lang. Die geforderte Länge wird erst bei einer Tiefe von 0,75m erreicht. Ein Punkt, der auch für den gewählten Laser spricht, ist die große erlaubte Schwankungsbreite bei der Versorgungsspannung. Die Laser mit einem größeren Öffnungswinkel stellen größere Anforderungen an die Versorgungsspannung.

4.3 Die Schwenkeinheit

Zur Aufnahme der 3D-Daten nach dem Lichtschnittverfahren wird, wie bereits beschrieben, eine Schwenkeinheit benötigt. Dies muss dazu in der Lage sein, den Sensor zu drehen und zu positionieren. Gerade das Positionieren ist sehr wichtig, denn dadurch kann für jede geforderte Position ein Lichtschnitt aufgenommen werden. Ohne den zugehörigen Winkel können aus den einzelnen Schnitten keine korrekten 3D-Daten errechnet werden.

Da bei den Recherchen keine Schwenkeinheit gefunden werden konnte, die bezüglich Größe und Gewicht den Anforderungen entspricht, musste die Schwenkeinheit selbst konstruiert und gebaut werden. Die Schwenkeinheit kann als eigene elektromechanische Baugruppe betrachtet werden. Aus diesem Grund waren bei der Planung sowohl elektrische als auch mechanische Aspekte und Komponenten zu beachten. In den beiden folgenden Unterkapiteln werden diese unterschiedlichen Aspekte beschrieben.

4.3.1 Die elektrischen Komponenten

Die wohl wichtigste Komponente der Schwenkeinheit ist der Antrieb. Dafür kommen verschiedene Typen von Elektromotoren in Frage. Doch ganz gleich, welcher Motorentyp ver-

wendet wird, ohne eine dazu passende, geeignete Steuerung kann die Schwenkeinheit nicht funktionieren.

Die Steuerung muss entsprechend der allgemeinen Anforderungen dazu in der Lage sein, den Motor so anzusteuern, dass dieser die vom PC vorgegebene Position in angemessener Zeit anfährt. Sie muss aber auch dazu in der Lage sein, den PC mit Informationen über die aktuelle Position der Schwenkeinheit zu versorgen.

Es gibt unterschiedliche Typen und Ausführungen von Elektromotoren, wobei es für die Auswahl ganz entscheidend ist, ob es auch eine geeignete Ansteuerung gibt. Im Folgenden werden drei in Betracht gezogene Motorentypen und deren Eignung als Antrieb für die Schwenkeinheit näher beschrieben.

Der gewöhnliche Gleichstrommotor

Als erster Motorentyp wurde der gewöhnliche Gleichstrommotor betrachtet. Durch seinen einfachen Aufbau und die weite Verbreitung ist er preiswerter als die beiden hier noch betrachteten Alternativen. Doch leider gibt es für den Gleichstrommotor keine brauchbare Ansteuerung, die FireWire unterstützt. Ein Problem mit der Ansteuerung begründet sich auch damit, dass zur Positionierung eine Regelung erforderlich ist. Sollte eine Ansteuerung selbst entwickelt werden, so muss die entsprechende Regelung ebenfalls entworfen und in die Ansteuerung integriert werden. Diese Regelung benötigt, wie jede andere Regelung auch, einen Messwertaufnehmer, also einen Sensor.

Für die Schwenkeinheit ist als Sensor ein Drehwinkelsensor erforderlich. Die einfachste und günstigste Ausführung eines Drehwinkelsensors ist ein Drehpotentiometer (Abbildung 4.3a). Dieser analoge Sensor arbeitet allerdings weniger exakt, als die nachstehenden digitalen Sensoren, dafür aber kontinuierlich und nicht schrittweise. Zur Messwertaufnahme bei einer digitalen Regelung ist zusätzlich auch eine Analog-Digital-Wandlung erforderlich. Die einfachste Art der digitalen Drehwinkelsensoren ist der Inkremental-Drehgeber (Abbildung 4.3b). Bei diesem wird die Drehung meist optisch durch eine Abtastscheibe und zwei Lichtschranken gemessen. Es gibt aber auch Sensoren, bei denen die Drehung durch Magnete oder mit Hilfe von Schleifkontakten ermittelt wird. Inkrementale Drehgeber können

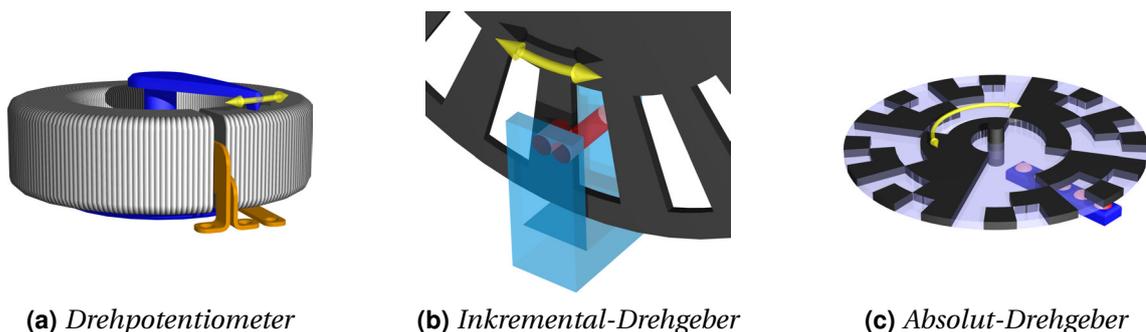


Abbildung 4.3: Verschiedene Drehwinkelsensoren: (a) Der Drehpotentiometer arbeitet als stetig einstellbarer Spannungsteiler. Aus dem Spannungsteilverhältnis kann der Drehwinkel ermittelt werden. (b) Beim inkrementalen Drehgeber wird der Winkel aus der Abtastung einer drehenden Lochscheibe ermittelt. Zur Ermittlung der Drehrichtung und zur Auflösungserhöhung um den Faktor 4 erfolgt die Abtastung mit zwei, um eine viertel Teilung versetzte, Lichtschranken. (c) Beim Absolut-Drehgeber kann die absolute Winkelstellung durch Abtastung der binär codierten Impulsscheibe ermittelt werden.

jedoch nur den zurückgelegten Winkel messen und nicht die absolute Position. Dazu ist eine zusätzliche Referenz erforderlich, die beispielsweise mit einem Schalter realisiert werden kann. Für den geplanten Einsatzfall ließe sich diese Referenz auch durch eine am Sensorsystem angebrachte Markierung, die von der Kamera erfasst werden kann, verwirklichen. Eine solche Referenz ist beim Absolut-Drehgeber (Abbildung 4.3c) nicht erforderlich. Dieser im Vergleich zum inkrementalen Drehgeber teurere Sensor kann die absolute Position bestimmen. Dafür wird eine binär codierte Impulsscheibe verwendet, die von Photodetektoren abgetastet wird. Dabei ist je Bit ein Detektor erforderlich.

Grundsätzlich ließe sich der Antrieb für die Schwenkeinheit mit einem gewöhnlichen Gleichstrommotor, einem geeigneten Drehwinkelsensor und einer selbst erstellten Ansteuerung sehr gut verwirklichen. Der Selbstbau hätte auch den Vorteil, dass alle Anforderungen, insbesondere der FireWire-Anschluss, bestmöglich berücksichtigt werden können. Diese Variante wäre somit die beste Wahl, gäbe es keine fertige Ansteuerung für andere Motorentypen.

Der Schrittmotor

Der zweite betrachtete Motorentyp ist der Schrittmotor. Dieser Motor besitzt mehrere Spulenstränge, die in abgestimmter Reihenfolge von Strom durchflossen werden. Durch diese schrittweise Abfolge der Strangströme dreht sich auch der gezahnte Rotor schrittförmig weiter. In Abbildung 4.4 ist das Prinzip eines einfachen Schrittmotors schematisch dargestellt. Die Ansteuerung des Motors mit digitalen Schaltungen ist durch die einfache Signalform der einzelnen Strangströme – es sind einfache Rechteck-Signale – sehr leicht zu bewerkstelligen. Durch die schrittweise Bewegungsform ist der zurückgelegte Drehwinkel bekannt, weshalb kein Sensor zur Positionsbestimmung erforderlich ist. Lediglich eine Referenz wie beim inkrementalen Drehgeber wird benötigt, damit die Ausgangsposition erkannt werden kann.

Schrittmotoren werden im Hobby-Bereich häufig eingesetzt, weshalb es dafür auch viele einfache Ansteuerungen gibt. Diese weisen jedoch als Schnittstelle nur eine serielle RS232-Schnittstelle oder eine USB-Schnittstelle auf (z.B. PhidgetStepper Unipolar 4-Motor oder PhidgetStepper Bipolar 1-motor von Phidget, Stepper-Bee von pc-control.co.uk). Für eine FireWire-Schnittstelle konnte zumindest keine Steuerung gefunden werden, die den Anforderungen entspricht. Es gibt einige wenige industrielle Schrittmotorsteuerungen, die sich per FireWire ansteuern lassen, diese sind aber zu groß und teuer. Bei den Steuerungen

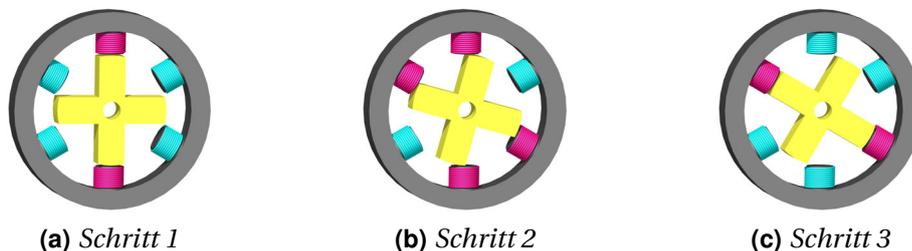


Abbildung 4.4: Das Funktionsprinzip eines Schrittmotors: (a) Der gezahnte Rotor – in diesem einfachen Beispiel besitzt er lediglich vier Zähne – richtet sich durch das Magnetfeld der Strom durchflossenen Spulen aus (die Strom durchflossenen Spulen haben die Farbe *Magenta*). (b) Werden zusätzliche Spulen aktiviert, so dreht sich der Rotor in die entsprechende Richtung. (c) Werden nun die zu Beginn aktivierten Spulen abgeschaltet, so dreht sich der Rotor noch einen Schritt weiter und richtet sich wieder nach den aktiven Spulen aus.

mit USB-Schnittstelle wird für den Motor auch noch eine zusätzliche Spannungsversorgung benötigt. Das bedeutet allerdings, dass bei Verwendung einer solchen Steuerung das Sensorsystem neben der FireWire-Schnittstelle noch eine USB-Schnittstelle und eine Stromversorgung benötigt, was nicht den Anforderungen entspricht.

Der RC-Servo

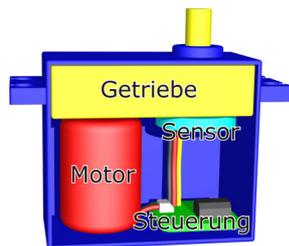
Der dritte und letztendlich auch gewählte Motorentyp ist der so genannte RC-Servo. Ein RC-Servo wird als Positionierantrieb im Modellbau bei ferngesteuerten² Modellen verwendet. Wie die schematische Darstellung eines RC-Servos in Abbildung 4.5a zeigt, handelt es sich dabei eigentlich nicht um einen eigenen Motorentyp. Vielmehr besteht ein RC-Servo aus einem Gleichstrommotor mit Getriebe und integrierter Lageregelung. Die Ansteuerung erfolgt mit einem pulsweitenmodulierten Signal. Dabei gibt das Tastverhältnis von Impulsdauer zu Periodendauer den gewünschten Winkel an. Die integrierte Regelung steuert den Gleichstrommotor so an, dass sich am Sensor der gewünschte Winkel einstellt. Das Sensorsignal ist allerdings nur intern vorhanden, weshalb sich ein Positionsfehler, wie er entstehen könnte, wenn der Servo blockiert wird, nicht erkennen lässt. Im normalen Betriebszustand entspricht die tatsächliche Position nach Ablauf einer gewissen Ausregelzeit jedoch der vorgegebenen Position.

Auch im Falle des RC-Servos konnte keine Ansteuerung mit FireWire-Anschluss gefunden werden. Es konnte jedoch eine kleine Auswahl von Steuerungen mit USB-Anschluss gefunden werden, die in Tabelle 4.3 aufgelistet ist.

Hersteller	Bezeichnung	Stromversorgung	Winkel	Preis
<i>Phidgets</i>	<i>PhidgetServo 1-Motor</i>	USB	-23° – 232,99°	€ 27,42
Parallax	Servo Controller (USB)	extern	—	€ 39,73
Pololu	USB 16-Servo Controller	extern	-180° – 180°	€ 99,00

Tabelle 4.3: Liste mit in Betracht gezogenen RC-Servo-Steuerungen. Die gewählte Steuerung ist kursiv hervorgehoben.

Bei der Ansteuerung des RC-Servos fiel die Wahl auf die *PhidgetServo 1-Motor*, da diese als einzige Steuerung die Stromversorgung des RC-Servos per USB zulässt. Als RC-Servo wurde der von Phidget empfohlene *HS-322-HD* von Hitec gewählt. In Abbildung 4.5b ist die gewählte Steuerung zu sehen.



(a) RC-Servo



(b) PhidgetServo 1-Motor

Abbildung 4.5: RC-Servo mit USB-Steuermodul: (a) Die schematische Darstellung eines RC-Servos zeigt dessen vier Grundkomponenten. (b) Mit dem verwendeten PhidgetServo 1-Motor lässt sich ein RC-Servo über USB ansteuern.

² Die englische Bezeichnung für ferngesteuert lautet *remote control*, kurz RC. Daher auch die Bezeichnung RC-Servo.

Die Alternative zur Steuerung per USB wäre ein Selbstbau einer Steuerung gewesen, die sich per FireWire kontrollieren lässt. Da allerdings nicht gesagt ist, dass der RC-Servo ein geeigneter Antrieb ist, ist es fürs erste besser den RC-Servo mit einer existierenden Steuerung zu überprüfen und erst danach einen möglichen Eigenbau zu planen.

4.3.2 Die mechanischen Komponenten

Nach der Festlegung der Antriebsart galt es, den Antrieb mit zusätzlichen, mechanischen Komponenten zu einer Schwenkeinheit zu kombinieren. Entscheidend dabei sind die Art der Lagerung sowie die der Kraftübertragung.

Die erste Entscheidung, die zu treffen war, betraf die Befestigungsposition des RC-Servos. Eine Möglichkeit wäre, den Servo an der feststehenden Befestigungsplatte, die später zur Befestigung des Sensors am Roboterarm dienen soll, zu montieren. In diesem Fall würde die Gesamthöhe des Sensors von der Höhe der Kameras und der Höhe des Servos bestimmt. Aus Platzgründen ist es daher besser, den Servo am drehbaren Teil des Sensors, der auch die Kamera trägt, zu befestigen. Dadurch wird die Gesamthöhe durch die größte der beiden Höhen von Kamera und Servo bestimmt. Da der Servo niedriger ist als die Kameraplatine, wird die Gesamthöhe des Sensors nicht durch den Servo beeinträchtigt.

Ein sehr wichtiger Punkt bei der Konstruktion der Schwenkeinheit ist die Lagerung. Da der Sensor vorwiegend in horizontaler Richtung geschwenkt werden soll, ist eine Lagerung erforderlich, die hauptsächlich Druckkräfte in axialer Richtung aufnehmen kann. Da der Teil des Roboterarms, der den Sensor trägt, allerdings nicht ausschließlich waagrecht ist, muss die Lagerung auch radiale Kräfte aufnehmen, damit die Schwenkeinheit auch bei einer solchen Schiefelage leichtläufig bleibt. Für den Fall, dass der Sensor eventuell am Kopf stehend montiert werden soll, muss die Lagerung auch noch Zugkräfte in Achsrichtung aufnehmen. Das einzige fertige Lager, das Radialkräfte und axiale Zug- und Druckkräfte aufnehmen kann und zugleich eine geringe Baugröße aufweist, das gefunden werden konnte, ist ein Axial-Schrägnadellager von INA der Baureihe ZAXFM. Im Konkreten war das Lager ZAXFM0835 (Abbildung 4.6a) von der Baugröße geeignet. Eine Anfrage beim Wiener Vertriebspartner von INA, der Firma Haberkorn Ulmer GmbH vom Jänner 2008 ergab allerdings einen Netto-Preis von € 97,20 für das Lager. Bei diesem Preis wäre das Lager eine der teuersten Komponenten des gesamten Sensors, weshalb eine alternative Lösung erforderlich war. Aus diesem Grund wurde der Entschluss gefasst, selbst eine geeignete Lagerung zu entwerfen. Um dabei die Kosten gering zu halten, wurde das Hauptaugenmerk auf die Verwendung von Standardkomponenten gelegt.

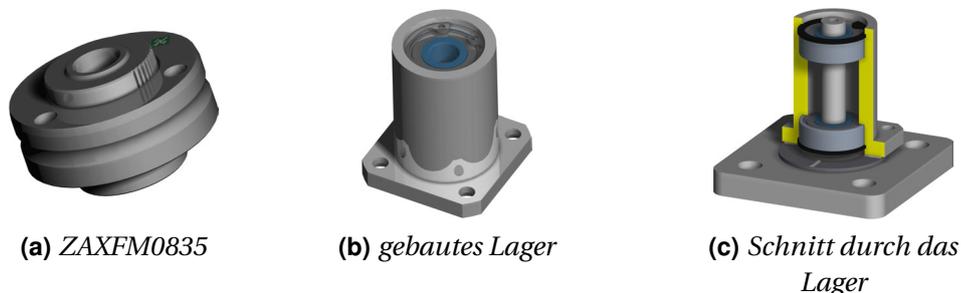


Abbildung 4.6: Das Lager für die Schwenkeinheit: (a) Das fertige Lager ZAXFM0835 ist leider zu teuer. (b) Aus diesem Grund wurde als Ersatz ein eigenes Lager gebaut. (c) In der Schnittdarstellung lassen sich die verbauten Standardlager erkennen.

In Abbildung 4.6b auf der vorherigen Seite ist die Lagerung dargestellt. Das Lager besteht im Wesentlichen aus einem Gehäuse, indem zwei Standard-Rillenkugellager vom Typ 608 aus der Reihe 60 nach DIN 625-1 sitzen. Zur Aufnahme von axialen Druckkräften befindet sich ein Axial-Nadelkranz vom Typ AXK2035 nach DIN 5405-2. Im Gegensatz zum fertigen Lager ZAXFM0835 ist das selbstgebaute nicht in der Lage, axiale Zugkräfte aufzunehmen.

Der letzte Punkt bei der Konstruktion der Schwenkeinheit ist die Festlegung der Kraftübertragung. Im Modellbau erfolgt die Kraftübertragung vom RC-Servo mit Seilzügen oder Schubstangen. Dazu werden verschiedene Hebel an die Servoachse aufgesetzt und angeschraubt. Die Abbildung 4.7 zeigt solche unterschiedliche Hebel. Damit diese Hebel nicht durchrutschen können, sorgt eine Außenverzahnung an der Servoachse mit der dazu passenden Innenverzahnung für einen Formschluss. Seilzüge oder Schubstangen sind für die Schwenkeinheit nicht geeignet, da damit keine gleichbleibenden Übersetzungen möglich sind. Damit der mögliche Schwenkbereich des Servos mit rund 180° auf den, für den Sensor nötigen Wert von 90° reduziert werden kann, ist eine Übersetzung erforderlich.

Für die Kraftübertragung wurden geradzahnte Stirnräder mit Evolventenverzahnung nach DIN 867 mit einem Modul nach DIN 780 von 1mm gewählt. Dabei hat das Zahnrad am Servo 25 Zähne und das angetriebene hat 50 Zähne. Somit ergibt sich eine Übersetzung von 2:1. Der Grund für die Wahl von Zahnrädern zur Kraftübertragung liegt darin, dass Zahnräder in anderen Anwendungen gut erprobt sind und sehr günstig zu bekommen sind. Ein Problem mit den Zahnrädern ergab sich allerdings bei der Befestigung am RC-Servo. Die Außenverzahnung an der Servoachse ist nicht genormt. Daher gibt es keine Werkzeuge, mit denen sich eine solche Verzahnung herstellen ließe. Die Lösung des Problems bestand darin, das Zahnrad auf eine, dem Servo beiliegende Scheibe zu schrauben, die sich problemlos am Servo befestigen lässt. In Abbildung 4.8 ist die gesamte Schwenkeinheit dargestellt.



Abbildung 4.7: *Unterschiedliche Arten von Hebel zur Kraftübertragung beim RC-Servo.*

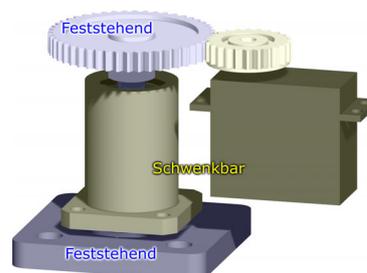


Abbildung 4.8: *Die Schwenkeinheit: Die als feststehend markierten Teile sind durch die Bodenplatte fix mit dem Roboterarm verbunden. Die schwenkbaren Teile sind mit dem Rahmen, der Kameras und Laser trägt, verbunden.*

4.4 Die Spannungsversorgung des Lasers

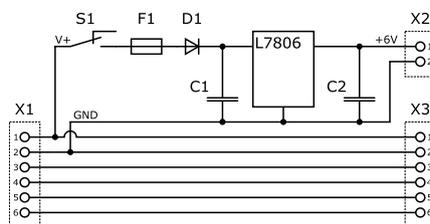
Wie bereits erwähnt, benötigt der Linienlaser eine eigene Spannungsversorgung, die gemäß den Anforderungen per FireWire bereitgestellt werden muss. Somit wurde eine Schaltung benötigt, die über einen FireWire-Anschluss verfügt, die Stromversorgung des Linienlasers aus der des FireWire-Bus bereitstellen kann. Die integrierte Stromversorgung bei FireWire ist mit einer erlaubten Spannung von 8–33V DC bei 1,5A Stromstärke und maximal 48W Leistung genormt. Der gewählte Linienlaser benötigt eine Versorgungsspannung von 2,5–12,5V DC (siehe auch Tabelle 4.2 auf Seite 37). Somit liegt der Spannungsbereich des Lasers unter dem vom FireWire-Bus, was für die Erzeugung der Versorgungsspannung des Lasers von Vorteil ist.

Zur Erzeugung einer stabilen Spannung werden in der Elektronik meist Spannungsregler eingesetzt. Da der benötigte Spannungswert nicht besonders exakt sein muss, ist der Einsatz eines Festspannungsreglers mit positiver Ausgangsspannung der Serie L7800 von STMicroelectronics möglich. Diese als integrierte Schaltung aufgebaute Regler beinhalten bereits alle wichtigen Komponenten. So sind neben der obligatorischen Referenzspannungsquelle auch noch diverse Schutzschaltungen integriert [60].

Festspannungsregler der Serie L7800 werden für unterschiedliche Spannungswerte im Bereich von 5–24V angeboten. Aufgrund besserer Verfügbarkeit wurde ein Spannungsregler L7806 im TO-220 Gehäuse verwendet. Der L7806 liefert eine Spannung von 6V mit einer maximalen Abweichung von $\pm 0,35V$ [60].

Neben dem Festspannungsregler besteht die Schaltung der Spannungsversorgung noch aus einigen zusätzlichen Bauteilen. Trotz aller bereits integrierten Schutzschaltungen sind die Spannungsregler noch zusätzlich abzusichern. Durch die internen Regelvorgänge können die Spannungsregler zu schwingen beginnen und dadurch schädliche, hochfrequente Signale erzeugen. Um diese Schwingungen zu verhindern, sollten solche Festspannungsregler immer sowohl eingangs- als auch ausgangsseitig mit Kondensatoren als Tiefpassfilter versehen werden.

Die weiteren auf der Schaltung angebrachten Schutzmaßnahmen dienen nicht zum Schutz der Schaltung selbst, sondern eher zum Schutz der anderen, am FireWire-Bus angeschlossenen Geräte. Die angebrachte Feinsicherung soll bei einem Defekt der Schaltung eine Überlastung des FireWire-Bus verhindern. Als zusätzlicher Verpolungsschutz dient eine einfache Diode. Um den Laser nach Bedarf ein- oder auszuschalten, wurde auch noch ein zusätzlicher Mikroschalter angebracht. Der Schaltplan sowie ein Bild der gebauten Schaltung sind in Abbildung 4.9 dargestellt.



(a) Schaltplan



(b) Schaltung

Abbildung 4.9: Die Schaltung zur Spannungsversorgung des Lasers: Mit Hilfe eines Festspannungsreglers (L7806) erzeugt die Schaltung eine Gleichspannung von 6V aus der, per FireWire zur Verfügung gestellten Versorgungsspannung.

4.5 Das Gehäuse und die Befestigung

Nachdem in diesem Kapitel bereits alle einzelnen Komponenten des Sensors beschrieben wurden, können die vorhandenen Komponenten zum fertigen Sensor kombiniert werden. Was dazu aber noch fehlt, ist ein Rahmen, der die einzelnen Komponenten trägt. Dieser Rahmen soll selbstverständlich ein möglichst geringes Gewicht aufweisen und dennoch ausreichend stabil sein. Eine Größenvorgabe ist nicht erforderlich, da die Größe durch die anderen Komponenten vorgegeben wird.

Um die Kosten für diesen Rahmen gering zu halten, musste eine entsprechende Fertigungsart gewählt werden. Als Vorbild dafür dienten Elektrogeräte aus der Unterhaltungs- und Informationselektronik. Das Innenleben von PCs, DVD-Abspielgeräten und ähnlichen sitzt meist auf einem Rahmen aus gebogenem Blech. Durch moderne computergesteuerte Werkzeugmaschinen wie Laserschneidanlagen, Stanznibbelmaschinen und Abkantpressen lassen sich verschiedene Blechteile auch in kleiner Stückzahl kostengünstig fertigen. Ein Vorteil in der Verwendung von Blechen liegt im geringen Gewicht. Dieses geringe Gewicht ist durch die Verwendung von dünnwandigen Blechen erreichbar. Zur Erhöhung der Steifigkeit des Rahmens können Abkantungen³ dienen.

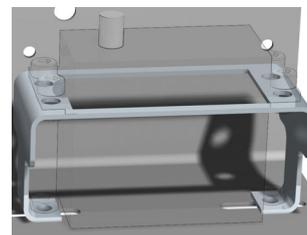
Der Rahmen soll nun alle Komponenten des Sensors tragen. Das beinhaltet die drei Kameras, den Laser und die Schwenkeinheit. Diese Komponenten sollen sich durch Schrauben am Rahmen direkt oder mit Hilfe kleiner zusätzlicher Bauteile befestigen lassen. So wird beispielsweise für den runden Laser eine eigene Befestigung benötigt (Abbildung 4.10a).

4.5.1 Für den Prototyp

Der im Zuge dieser Diplomarbeit erstellte Sensor ist lediglich ein Prototyp und weist nicht alle Eigenschaften auf, die der Sensor später einmal aufweisen soll. So besitzt der Prototyp kein Gehäuse und die elektrischen Schaltungen können nicht befestigt werden. Dafür lässt sich beim Prototyp der RC-Servo noch verschieben, wodurch es noch möglich ist, andere Zahnräder am Servo zu befestigen und damit andere Übersetzungsverhältnisse zu erproben. Damit sich der Servo verschieben lässt, wurde ebenfalls aus Blech eine eigene Befestigung gebaut, in der der Servo montiert wird (Abbildung 4.10b). Diese Befestigung wird wiederum durch zwei Langlöcher an den Rahmen geschraubt. Durch diese Langlöcher ist es nun möglich, den Servo zu verschieben.



(a) Befestigung für den Laser



(b) Befestigung des RC-Servos

Abbildung 4.10: Verschiedene zusätzliche Befestigungselemente: Der Laser wird einfach zwischen zwei Backen eingeklemmt (a). Beim Prototyp wird der RC-Servo in einem gebogenen Blechteil befestigt, der durch Langlöcher verschiebbar am Rahmen befestigt wird (b).

³ Ein Abkantung entsteht, wenn ein flaches Stück eines Blechteiles entlang einer Biegekante gebogen wird.

4.5.2 Für die Endversion

Auch wenn kein serienreifes Endprodukt gefertigt wurde, so wurden zumindest Konstruktionspläne dafür erstellt. Im Vergleich zum Prototypen sind dabei folgende Änderungen vorgesehen:

- Der Rahmen trägt nun alle Komponenten: Die Kameras, den Linienlaser in seiner Befestigung, die elektrischen Schaltungen, den RC-Servo und die Lagerung der Schwenkeinheit.
- Während beim Prototyp noch Muttern zur Befestigung der Komponenten verwendet wurden, sollen beim Endprodukt im Rahmen Tiefzuggewinde⁴ platziert werden.
- Die Endversion soll mit einem lackierten Gehäuse aus Blech versehen werden.

In Abbildung 4.11 ist der Prototyp mit Entwürfen der Endversion zu sehen.

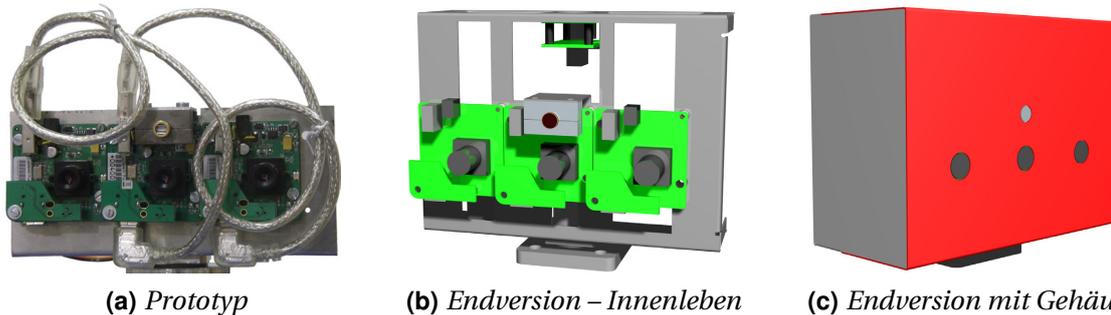


Abbildung 4.11: Prototyp und Endversion des Sensors: (a) Am Prototypen zeigt sich deutlich, dass die Kabel für die Kameras viel Platz benötigen. (b) Damit die Kabel bei der Endversion unter dem Gehäuse Platz finden, muss die Endversion entsprechend hoch gebaut werden. Der Entwurf zeigt auch Befestigungsmöglichkeiten für Spannungsversorgung und Motorsteuerung. (c) Das Innenleben des Sensors soll durch ein lackiertes Gehäuse aus Blech geschützt werden.

⁴ Da in dünnwandigen Blechen nicht genug Gewindegänge geschnitten werden können, werden die Gewinde häufig in Löcher geschnitten, die durch ein Umformung in deren Tiefe verlängert wurden.

5

5 Die Software

Grundsätzlich benötigt jeder Tiefenbildsensor eine Software, die aus Bildern und Lageinformationen 3D-Daten generiert. Eine solche Software könnte in den Sensor direkt integriert werden, wenn der Sensor mit einem entsprechenden Mikroprozessor ausgestattet wird. Beim gebauten Sensor soll die Erstellung der 3D-Daten jedoch an einem handelsüblichen PC oder Notebook erfolgen. Für diesen Fall existieren bereits unterschiedliche Softwarelösungen, die am Institut für Automatisierungs- und Regelungstechnik zum Teil bereits im Einsatz sind. Die Software muss jedoch zuerst an die jeweilige Hardware angepasst werden.

Als Teil der Diplomarbeit, die sich hauptsächlich mit der Hardware des Sensors befasst, galt es jedoch auch, Funktionen zur Ansteuerung der Hardware zur Verfügung zu stellen. Dabei ging es darum, eine Ansteuerung der Schwenkeinheit und der Kameras durch einfache Klassen zu ermöglichen. Das folgende Kapitel gibt Auskunft über die verwendete Programmiersprache, den Umfang der Klassen und über benötigte Programmbibliotheken. Am Abschluss des Kapitels wird noch über aufgetretene Schwierigkeiten bei der Ansteuerung der Komponenten berichtet.

5.1 Die Programmiersprache und das Zielsystem

Der Sensor soll, wie auch bei den Anforderungen in Kapitel 3.2 festgelegt, über einen einzigen FireWire-Anschluss an einen PC anschließbar sein. Dies wurde festgelegt, da der Sensor ohne Schwierigkeiten an einen Standard-PC anzuschließen sein soll, um auf teurer Hardware verzichten zu können. Aus diesem Grund soll auch auf eine spezielle Software verzichtet werden können, weshalb eine Programmiersprache zu wählen war, die eine einfache Portierung der Software ermöglicht.

Nachdem die Software zur Ansteuerung der Hardware benötigt wird, ist bei der Wahl der Programmiersprache immer etwas Vorsicht geboten. Nicht jede Hardware funktioniert mit jedem Betriebssystem und jeder Software. Wichtig ist, dass für das Betriebssystem Treiber vorhanden sind, die zusätzlich noch passende Programmierschnittstellen bieten. Da die Kameras mit einer FireWire-Schnittstelle über einen Standardanschluss verfügen, ist es möglich, die Kameras über freie Programmbibliotheken anzusprechen und somit auf den Kauf von spezieller Software zu verzichten. Die Motor-Ansteuerung der Schwenkeinheit erfolgt allerdings nicht über eine Standardkomponente, was Schwierigkeiten bereiten kann. Glücklicherweise bietet der Hersteller kostenlos Treiber für Windows, Mac OS X und Linux an. Dieser Treiber bietet eine Unterstützung für eine Vielzahl von Programmiersprachen. Darunter sind C/C++, C#, Java, Python, Visual Basic und noch einige andere, wodurch die Auswahl an in Frage kommenden Programmiersprachen kaum eingeschränkt wird.

Da jedoch auch noch eine leichte Portierung möglich sein soll, war eine Programmiersprache zu wählen, die auf möglichst vielen Betriebssystemen weit verbreitet ist. Somit standen im Grunde nur noch C/C++ oder Java zur Auswahl, wobei C/C++ gewählt wurde.¹

Als Betriebssystem kam OpenSUSE 10.2 mit einem Linux-Kernel in der Version 2.6.18.2 zum Einsatz. Grundsätzlich sollte die Software auch mit Mac OS X und Windows laufen, jedoch gibt es bei Windows noch einige Schwierigkeiten, wie in Kapitel 5.5 erklärt wird.

5.2 Die Ansteuerung der Schwenkeinheit

Für den, bei der Schwenkeinheit verwendeten Controller, den „PhidgetServo 1-Motor“ von Phidgets, gibt es vom Hersteller Treiber für Linux, Mac OS X und Windows. Eingesetzt wurde der Treiber in der Version 2.1.4 vom 5. August 2008. Voraussetzung für eine Installation bei Linux ist lediglich ein Kernel der Version 2.6 oder höher. Die Installation erfolgt wie üblich mit *Root*-Rechten² und dem *Make*-Befehl³. Damit neben dem Root-Benutzer auch noch andere Benutzer das erforderliche Schreibrecht auf der USB-Schnittstelle erhalten, muss das Schreibrecht gemäß den Anweisungen der, dem Treiber beiliegenden, README-Datei erteilt werden.

Die Programmierschnittstelle, kurz API (für englisch application programming interface), bietet alle Funktionen zur Ansteuerung des Servos. So kann der Servo aktiviert und deaktiviert werden, es kann eine Position vorgegeben und die aktuelle Position abgefragt werden. Bei der Abfrage der aktuellen Position ist allerdings zu beachten, dass nur die vom Servocontroller vorgegebene Sollposition zurückgemeldet werden kann. Die tatsächliche Position lässt sich nicht ermitteln, da ein RC-Servo keine Positionsrückmeldung bietet (siehe auch Kapitel 4.3.1).

Die geschriebene Klasse *ServoControl* verwendet diese Funktionen um alle für die Ansteuerung der Schwenkeinheit erforderlichen Funktionen zu erfüllen. Sie baut automatisch eine Verbindung zum angeschlossenen Servo auf und stellt Funktionen

- zur Aktivierung und Deaktivierung des Servos,
- zum Festlegen der Übersetzung der Schwenkeinheit,
- zur Festlegung der Endwerte und des Nullpunktes des Schwenkes,
- zur Positionierung und zur Positionsrückmeldung
- und zum schrittweisen Bewegen

zur Verfügung.

Das Festlegen der Übersetzung dient dazu, dass bei Positionierung und Positionsrückmeldung die tatsächlichen Winkelwerte verwendet werden können. Das schrittweise Bewegen hat den Vorteil, dass zur Aufnahme eines Bildes nach dem Lichtschnittverfahren nur die

¹ Es gibt noch andere Programmiersprachen, die auf vielen Betriebssystemen funktionieren. Die meisten benötigen aber eine eigene Software zur Ausführung der Programme und sind auch nicht so stark verbreitet.

² Der Root-Account ist unter UNIX und den meisten Derivaten, wie Linux und Mac OS X, der Benutzer mit allen Zugriffsrechten. Bei Windows entspricht dies dem Administrator-Account.

³ Make ist ein kleines Hilfsprogramm das Kommandos aus einer Datei, dem so genannten *Makefile* der Reihe nach ausführt. Bei der Softwareentwicklung wird es gerne dazu verwendet, die einzelnen Teile in richtiger Reihenfolge zu kompilieren.

Endwerte festgelegt werden müssen und dann die Schwenkeinheit schrittweise von der Start- zur Endposition bewegt werden kann. Somit müssen nicht ständig Positionen vorgegeben werden, die dann eventuell gar nicht erreichbar sind. Die Vorgabe von beliebigen Positionen bereitet aber auch der API Probleme, wie in Kapitel 5.4 noch erläutert wird.

Die Klasse wurde auch mit Mac OS X 10.4 und Windows XP getestet und hat – dank der guten Treiber von Seiten des Herstellers Phidgets – einwandfrei funktioniert. Somit ist für diese Klasse die Portabilität, zumindest für diese drei Betriebssysteme gegeben. In Abbildung 5.1 sind die wichtigsten Methoden der C++-Klasse aufgelistet.

5.3 Das Auslesen der Kamerabilder

Ohne Kamerabilder kann kein Tiefenbild erzeugt werden. Aus diesem Grund wird auch eine Klasse benötigt, die Bilder von den Kameras auslesen und zur Weiterverarbeitung aufbereiten kann. Da die Bilder nach dem Auslesen von der, für die Tiefenbildgewinnung zuständigen Software verwendet werden müssen, sollten die Bilder gleich in einer passenden Form geladen werden. Eine umfangreiche, gern verwendete und zugleich quelloffene Programm-bibliothek zur Bildverarbeitung in C/C++ ist *OpenCV*⁴. OpenCV bietet eine Vielzahl an Funktionen, wie sie für die Tiefenbildgewinnung benötigt werden. Zusätzlich ermöglicht OpenCV es auch, Bilder von einer Kamera einzulesen, weshalb OpenCV auch gleich direkt verwendet werden kann.

Beim Austesten der OpenCV-Funktionen *cvCaptureFromCAM* und *cvQueryFrame* wurde festgestellt, dass mit Hilfe dieser Funktionen immer nur Farbbilder geladen werden. Das hat jedoch zum Nachteil, dass bei den Monochromkameras der Speicherbedarf der Bilder verdreifacht wird. Auch für die Farbkamera ist dies von Nachteil, da die verwendete Farbkamera die Bilder ausschließlich im RAW-Format – also „monochrom“ – überträgt. Um den Speicherbedarf zu verringern, müssen die Bilder in monochrome Bilder umgewandelt werden. Für farbige Bilder ist die Funktion *cvCvtColor* zu verwenden. Diese kann allerdings nur auf Monochrombilder den Bayer-Filter anwenden und nicht auf Farbbilder. Somit müssen

ServoControl	
void	enableServo (bool enable) <i>aktiviert oder deaktiviert den Servomotor</i>
bool	isEnabled () <i>liefert den Aktivierungszustand der Servomotors</i>
bool	setPosition (double position) <i>bewegt den Servomotor zur angegebenen Position</i>
double	getPosition () <i>liefert die eingestellte Position</i>
bool	stepLeft (unsigned int steps) <i>bewegt den Servomotor die angegebene Anzahl an Schritte nach links</i>
bool	stepRight (unsigned int steps) <i>bewegt den Servomotor die angegebene Anzahl an Schritte nach rechts</i>
... verschieden Zugriffsfunktionen für die Klassen-Attribute ...	

Abbildung 5.1: Übersicht der wichtigsten Methoden der C++-Klasse ServoControl zur Ansteuerung der Schwenkeinheit.

⁴ *OpenCV* ist erhältlich bei SourceForge unter <http://sourceforge.net/projects/opencvlibrary/> und wurde in der Version 1.0.0 verwendet.

die, als Farbbilder aufgenommenen, aber trotzdem monochromen Bilder, zuerst in ein Monochrombild umgewandelt werden und dann in ein echtes Farbbild.

Um diese umständliche Prozedur zu umgehen, wurde bei der erstellten Klasse ein anderer Ansatz gewählt. Das Auslesen der Bilder erfolgt hier mit Hilfe der Bibliothek *libdc1394*⁵ die zur Steuerung von Kameras nach der Spezifikation „1394-based Digital Camera Specification“ (kurz DCAM) verwendet wird. Diese Bibliothek bietet nicht nur den Vorteil, dass die Anwendung des Bayer-Filters einfacher möglich ist, sondern erlaubt auch noch verschiedene Kameraeinstellungen zu ändern. So können nicht nur Framerate, Helligkeit, Kontrast, Sättigung und Farbton angepasst werden, sondern auch noch Belichtung, Weißabgleich, Verschlusszeit und mehr geändert werden. Die erstellte Klasse bietet neben der Unterstützung dieser Kameraeinstellungen auch zwei Funktionen zum Auslesen der Kamerabilder. Die Funktion *grabFrame* dient dabei zum Auslesen eines Monochrom- oder Farbbildes. Zum Auslesen von RAW-Farbbildern existiert die Funktion *grabBayerFrame*. Eine Übersicht der wichtigsten Methoden der Klasse *DC1394_Grabber* findet sich in Abbildung 5.2.

Wegen unterschiedlicher Datentypen der Bilder bei *libdc1394* und OpenCV wurde auch noch die Funktion *grabIplImage* erstellt, die ein zu OpenCV kompatibles Bild liefert. Dabei erkennt die Funktion anhand des Vorhandenseins der Weißabgleichfunktion eine Farbkamera und erzeugt automatisch ein Farbbild. Andernfalls wird ein monochromes Bild erzeugt.

Die Bibliothek *libdc1394* befindet sich für Windows noch im Entwicklungszustand und konnte deshalb nicht für Windows getestet werden. In Linux funktioniert die Bibliothek jedoch ohne Probleme. Mehr zu den Schwierigkeiten mit Windows gibt es unten in Kapitel 5.5.

DC1394_Grabber	
	DC1394_Grabber (uint64_t id, typeOfID type) <i>Konstruktor: Erstellt ein Objekt und initialisiert die Kamera mit der angegebenen ID</i>
bool	init (uint64_t id, typeOfID type) <i>initialisiert die Kamera mit der angegebenen ID</i>
uint64_t	getGUID () <i>liefert die GUID der Kamera</i>
dc1394video_frame_t *	grabFrame () <i>holt ein Bild von der Kamera</i>
dc1394video_frame_t *	grabBayerFrame () <i>holt ein Farbbild von der Kamera unter Anwendung eines Bayer-Filters</i>
IplImage *	grabIplImage () <i>holt ein Bild im OpenCV-kompatiblen Format</i>
	<i>... verschiedene Zugriffsfunktionen für die Kamera-Einstellungen (Framerate, Helligkeit, ...) ...</i>

Abbildung 5.2: Übersicht der wichtigsten Methoden der C++-Klasse *DC1394_Grabber* zum Auslesen der Kamerabilder.

⁵ *libdc1394* ist ebenfalls bei SourceForge erhältlich unter <http://sourceforge.net/projects/libdc1394/>. Die eingesetzte Version war 2.0.2.

5.4 Die Probleme mit der Schwenkeinheit

Nach dem Handbuch zum „PhidgetServo 1-Motor“ bietet der Controller eine Auflösungsge- nauigkeit von $0,1^\circ$ [61]. Daraus lässt sich schließen, dass mit einer schrittweisen Erhöhung des Winkelwertes um diese $0,1^\circ$ alle möglichen Positionen abgedeckt werden. Mit einem Ver- suchsprogramm, das eine solche Bewegung von der unteren zu oberen Grenze ausführt, wur- de jedoch eine ungleichmäßige Bewegung festgestellt. Dieses unerwartete Verhalten galt es deshalb zu analysieren.

Gemäß der C-API-Dokumentation [62] liefert die Funktion *CPhidgetServo_getPosition* die Servoposition als Zahlenwert vom Typ *Double*. Zum Testen dieser Funktion wurden ver- schiedene Werte mit der Funktion *CPhidgetServo_setPosition* vorgegeben und die Rückga- bewerte der get-Funktion ausgewertet. Die Ergebnisse waren immer Festkommazahlen mit zwei Stellen nach dem Dezimalkomma. Nachdem die zweite Stelle nach dem Komma unter- schiedliche Werte aufwies, war klar, dass der Unterschied kein ganzzahliges Vielfache von $0,1$ sein kann.

Zur Ermittlung der tatsächlichen Auflösung der Motorsteuerung wurde ein Testpro- gramm erstellt, das ausgehend von der Minimalposition die vorgegebene Position in $0,02^\circ$ - Schritten⁶ erhöht. Wurde beim darauf folgenden Auslesen der Position eine Veränderung zum vorherigen Wert festgestellt, so wurde dieser neue Wert in eine Datei geschrieben. Da- durch wurden alle einstellbaren Werte ermittelt. Dabei wurden 2.701 Werte im Bereich von $-22,81^\circ$ bis $231,91^\circ$ erfasst. Eine Auswertung der Ergebnisse zeigte, dass der Unterschied zweier, aufeinanderfolgenden Werte $0,1^\circ$ oder $0,09^\circ$ beträgt. Mit diesem Ergebnis erklärt sich eine ungleichmäßige Bewegung des Motors dadurch, dass bei schrittweiser Erhöhung der Vorga- bewerte um $0,1$ manche Werte übersprungen werden. Eine Möglichkeit zur Behebung dieses Problems besteht darin, nicht einfach die Vorgabewerte zu erhöhen, sondern einen neuen Vorgabewert aus der aktuellen Position durch Addition von $0,1^\circ$ zu Erzeugen. Diese Metho- de funktioniert einwandfrei, wenn die Positionserhöhung nur in $0,1^\circ$ -Schritten erfolgen soll. Werden allerdings größere Schritte vorgegeben, so kann diese Methode wieder zu Fehlern führen.

Die gewählte Lösungsmethode arbeitet derart, dass die Schritte von 1 bis 2701 durch- nummeriert werden. Durch diese Nummerierung kann eine Bewegung in beliebigen, gleich großen Schritten⁷ erfolgen. Die Zuordnung eines Winkelwertes zur Schrittnummer könnte anhand der ermittelten Werte durch Nachschlagen in einer Tabelle erfolgen. Aufgrund des annähernd linearen Zusammenhangs der Werte – sie sollten sich ja immer um $0,1^\circ$ unter- scheiden – wurde jedoch eine andere Methode gewählt.

Mit Hilfe des freien Programms *Gnuplot* wurde die Geradengleichung $y = kx + d$ mit den ermittelten Werten angenähert. Wobei x die Schrittnummer und y der zugehörige Win- kelwert sind. Dabei wurden die Werte

$$k = 0,0943396243510375,$$

$$d = -22,9056610802583$$

⁶ Die Schrittweite von $0,02^\circ$ wurde deshalb gewählt, damit je $0,1^\circ$ mehrere Positionen vorgegeben wurden. Bei einer Schrittweite von $0,05^\circ$ konnten nicht alle Werte ermittelt werden.

⁷ Die Schritte sind nicht exakt gleich groß, da die $0,09^\circ$ -Schritte nicht regelmäßig vorkommen.

ermittelt.⁸

Werden diese Werte in die Geradengleichung eingesetzt, so ergeben sich bei mathematischer Rundung auf zwei Nachkommastellen exakt die ermittelten Winkelwerte für die einzelnen Schritte. Da die Phidget-API allerdings die vorgegebenen Winkelwerte immer abrundet und immer den nächst kleineren Wert einstellt, ist noch dafür zu sorgen, dass die Rundung richtig funktioniert. Das lässt sich durch Addition von 0,005 zu d erreichen.

Mit Hilfe der Parameter für die Geradengleichung kann eine Umrechnung zwischen Position und Schrittnummer erfolgen. Anders als bei der Phidget-API wird bei der erstellten Klasse *ServoControl* zu einem vorgegebenen Winkelwert immer der am nächsten liegende Wert eingestellt.

5.5 Die Portabilitätsschwierigkeiten

Wie oben bereits erwähnt, ist die Windows-Version der Bibliothek *libdc1394* noch in Entwicklung. Es ist zwar bereits eine Alpha-Version der Bibliothek erhältlich, diese kommt jedoch ohne Installationsanleitung. Da auch sonst wenig Information über die Installation der Bibliothek unter Windows gefunden wurden, ist es nicht gelungen, die Bibliothek zum Laufen zu bringen. Somit kann die erstellte Klasse zur Aufnahme der Bilder nicht unter Windows verwendet werden.

Solang *libdc1394* noch nicht unter Windows funktioniert, muss eine andere Methode zur Aufnahme der Bilder unter Windows verwendet werden. Eine Möglichkeit ist es, die OpenCV-Funktion *cvCaptureFromCAM* zu verwenden. Diese arbeitet unter Windows mit *Video for Windows*⁹ oder *Matrox Imaging Library*¹⁰. Dabei ergibt sich allerdings wieder das oben beschriebene Problem mit der Umwandlung der RAW-Bilder in Farbbilder.

Neben der Möglichkeit, Funktionen der OpenCV-Bibliothek zu nutzen, können auch andere Bibliotheken genutzt werden. Dabei muss wie im Fall der *libdc1394* das aufgenommene Bild in das Format von OpenCV umgewandelt werden, damit es von OpenCV weiterverwendet werden kann. Die Dokumentation von OpenCV nennt dafür als Beispiele die Bibliothek *videoInput*¹¹ und den *CMU 1394 Digital Camera Driver*¹².

Eine Schwierigkeit bei Windows als Betriebssystem besteht auch darin, dass für die Hardware nicht immer geeignete Treiber, die auch mit den freien Bibliotheken zusammenarbeiten, verfügbar sind. Für FireWire-Kameras nach der Spezifikation DCAM (Version 1.30) sollte allerdings das Problem durch den *CMU 1394 Digital Camera Driver* zu lösen sein.

⁸ Im Zuge der Auswertung der Testergebnisse der Schwenkeinheit (Kapitel 6.3) wurde festgestellt, dass sich die Nachkommastellen der Winkel im Abstand von 10° wiederholen. Da in einem Bereich von 10° genau 106 Positionen liegen, wäre die Steigung k exakt $\frac{10}{106} = 0,09433962264150943396$. Da dieser Wert aber fast dem verwendeten entspricht, war keine Änderung erforderlich, da im verfügbaren Bereich wegen der Rundung ohnehin kein Fehler auftritt.

⁹ *Video for Windows* ist eine API, die mittlerweile von *DirectShow*, einer Komponente von *Microsoft DirectX* beziehungsweise von *Microsoft Windows SDK*, ersetzt wurde.

¹⁰ *Matrox Imaging Library* ist eine kostenpflichtige, sehr umfangreiche Programmbibliothek zur industriellen Bildverarbeitung der Firma *Matrox Electronic Systems Ltd.* aus Kanada.

¹¹ Die freie Bibliothek *videoInput* ist erhältlich unter <http://muonics.net/school/spring05/videoInput/>.

¹² *CMU 1394 Digital Camera Driver* ist erhältlich unter <http://www.cs.cmu.edu/~iwan/1394/>.

6

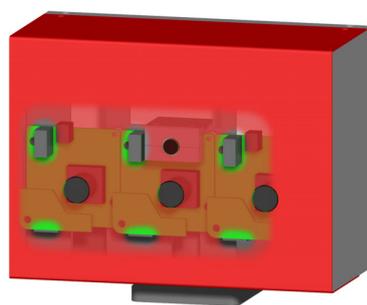
6 Experimente und Ergebnisse

Nachdem für den Sensor teilweise unerprobte Konzepte und Komponenten verwendet werden, ist es besonders wichtig, deren Funktionsweise und Eignung zu überprüfen. In diesem Kapitel werden die selbst erstellten Komponenten wie die Spannungsversorgung und die Schwenkeinheit getestet. Daneben wird auch noch der zugekaufte Linienlaser auf seine Eignung hin überprüft. Aus den dabei ermittelten Ergebnissen lassen sich Verbesserungsmöglichkeiten am Sensor herleiten.

6.1 Die Spannungsversorgung

Die erste Komponente des Sensors, deren Überprüfung hier beschrieben wird, ist die Spannungsversorgung. Diese Spannungsversorgung sollte zwei Aufgaben übernehmen. Die erste Aufgabe ist die Versorgung des Linienlasers mit der notwendigen elektrischen Energie, wobei die Energieversorgung von der FireWire-Schnittstelle übernommen werden soll.

Die zweite Aufgabe der Spannungsversorgung ist ein Weiterleiten oder Durchschleifen des FireWire-Signals. Da alle Komponenten mit einem FireWire-Anschluss in Serie miteinander verbunden werden müssen, darf nur eine Komponente auf das Durchschleifen verzichten. Die Kameras ermöglichen selbstverständlich ein Durchschleifen, weshalb es bei der Spannungsversorgung nicht notwendig wäre. Durch das für die Endversion vorgesehene Gehäuse sind die Anschlüsse der Kameras jedoch nicht mehr von außen zugänglich (Abbildung 6.1a). Das bedeutet, dass ein anderer Anschluss verwendet werden muss. Kann die Spannungsversorgung ebenfalls das FireWire-Signal weiterleiten, so kann diese auch den



(a) Anschlüsse der Kameras



(b) Anschluss an der Rückseite

Abbildung 6.1: Position des FireWire-Anschlusses am Sensor: (a) Die sechs FireWire-Anschlüsse der Kameras liegen hinter dem Gehäuse verborgen und können daher nicht als Anschluss für den Sensor verwendet werden. (b) Ermöglicht die Spannungsversorgung auch ein Durchschleifen des FireWire-Signals, so kann deren Anschluss als Anschluss des Sensors dienen.

Anschluss für den gesamten Sensor bereitstellen. So ist es beispielsweise möglich, den Anschluss an der Rückseite des Sensors zu legen (Abbildung 6.1b).

Beide Aufgaben wurden beim Entwurf der Schaltung berücksichtigt. Dabei wurde allerdings bewusst auf aufwendige Abschirmmaßnahmen verzichtet, da die Fertigung der Leiterplatte in der universitätseigenen Werkstatt erfolgte und die Bestückung per Hand durchgeführt wurde. Aufwendige Abschirmmaßnahmen hätten allerdings höhere Ansprüche an die Leiterplatte und deren Bestückung gestellt. So müssten dafür etwa die vier Signalleitungen parallel verlaufen und möglichst die gleiche Länge aufweisen, was aber wegen der von der Werkstatt geforderten Mindestleiterbahnenstärke nicht möglich war. Als weitere Abschirmmaßnahme hätten Kondensatoren eingesetzt werden müssen, die aus Platzgründen besser als SMD-Bauteile¹ ausgeführt werden sollten. Eine Leiterplatte, die die Anforderungen an die Abschirmung besser erfüllt, hätte nicht in der Werkstatt gefertigt werden können. Für die Produktion hätte eine Firma beauftragt werden müssen. Eine Online-Preiskalkulation bei Beta LAYOUT GmbH hat einen Preis von € 31,42 ergeben.² In Abbildung 6.2 ist die gebaute Schaltung mit dem zugehörigen Schaltplan dargestellt.

Nach der Fertigstellung der Schaltung wurde diese sogleich einem Funktionstest unterzogen. Bei diesem Test sollte ermittelt werden, ob die Schaltung auch beide Aufgaben erfüllen kann. Zum Testen wurde zunächst einmal der Laser an die Schaltung angeschlossen und diese dann per FireWire mit dem Computer verbunden. Da der Linienlaser leuchtete und sich auch durch den an der Schaltung abgebrachten Schalter ein- und ausschalten ließ, war der erste Test bestanden.

Zum Testen, ob auch das Durchschleifen des FireWire-Signals funktioniert, wurde zunächst einmal die Farbkamera direkt an den Computer angeschlossen und deren Funktion unter Linux mit dem Programm *Coriander* getestet, was einwandfrei funktionierte. Nachdem sichergestellt war, dass die Kamera auch tatsächlich funktioniert, wurde die Kamera mit einem kurzen FireWire-Kabel an die Spannungsversorgung und diese wiederum mit dem anderen Kabel an den Computer angeschlossen. In dieser Konfiguration konnten keine Bilder mehr von der Kamera empfangen werden. Eine an der Kamera angebrachte Sta-

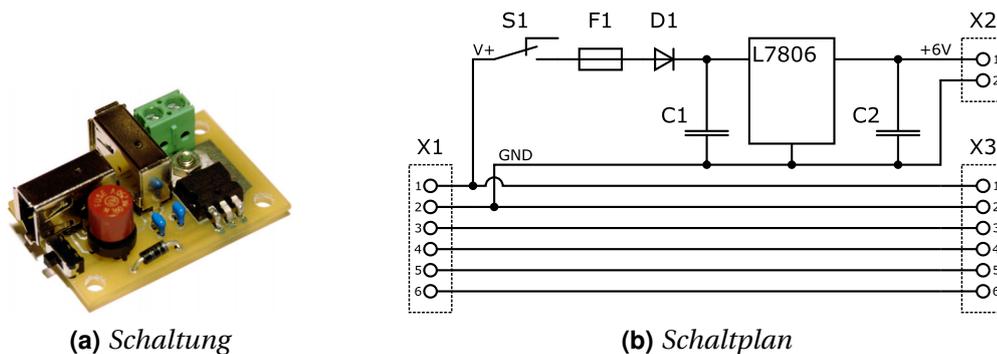


Abbildung 6.2: Die Schaltung zur Spannungsversorgung des Lasers: (a) Zur Spannungsversorgung des Lasers wurde eine eigene Schaltung gebaut. (b) Der Schaltplan zeigt, dass das Durchschleifen des FireWire-Signals ohne Abschirmmaßnahmen erfolgt.

¹ SMD steht für Surface-Mounted Device. SMD-Bauteile sind besonders kleine Bauteile, die direkt auf die Oberfläche der Leiterplatte gelötet werden, während bei den größeren, bedrahteten Bauteilen der Draht durch eine Bohrung geführt und an der Rückseite verlötet wird.

² Stand: 7. Oktober 2009. Der Preis basiert auf einer Mindestabnahmemenge von 10cm² was bei der verwendeten Platinengröße sechs Stück entspricht.

tus-LED leuchtete zwar, was allerdings nur eine korrekte Spannungsversorgung signalisiert. Die Spannungsversorgung der Kamera und des Lasers funktionieren, die Kamera liefert allerdings kein Bild, was bedeutete, dass das Durchschleifen nicht korrekt funktioniert, da das Signal wegen der schlechten Abschirmung zu stark gestört wird.

Da die Spannungsversorgung kein Durchschleifen des FireWire-Signals erlaubt, kann sie nur als letztes Glied in der Signalkette verwendet werden. Die Hauptaufgabe – die Versorgung des Lasers – wird jedoch korrekt erfüllt. Um den FireWire-Anschluss des Sensors an eine leicht zugängliche Position zu legen, kann alternativ eine FireWire-Einbaubuchse verwendet werden.

6.2 Der Linienlaser

Der Linienlaser wird für das Lichtschnittverfahren benötigt. Er muss eine Laserlinie projizieren, die von der Kamera erfasst und anschließend von der Software ausgewertet wird. Je besser die Laserlinie in den Kamerabildern – auch bei hellem Umgebungslicht – erkennbar ist, umso bessere Ergebnisse können erzielt werden.

Aus Kostengründen wurde mit dem *Picotronic LFL650-5-12(9x20)90* ein vergleichsweise schwacher Laser gewählt. Der Linienlaser liefert nur ein Zehntel der Leistung des Lasers vom existierenden Tiefenbildsensor [20], wie auch Tabelle 6.1 zu entnehmen ist. In der Tabelle findet sich eine Gegenüberstellung der wichtigsten technischen Daten der Linienlaser.

Eigenschaft	Picotronic	Lasiris
Leistung	5mW	50mW
Öffnungswinkel	90°	70°
Wellenlänge	650nm	685nm
Größe	∅9×20mm	∅19×82mm

Tabelle 6.1: Vergleichsdaten der beiden Laser-Modelle

Zum Vergleich der Laser wurde dieselbe Szene mit den beiden unterschiedlichen Lasern aufgenommen. Und zwar je einmal bei eingeschalteter und bei ausgeschalteter Beleuchtung. Da der Laser von Lasiris größer ist als der Picotronic, konnte der Laser nicht einfach an dessen Stelle in der Laserbefestigung eingespannt werden. Für den Versuch wurde deshalb der Sensor so positioniert, dass die Laserebenen der beiden Laser annähernd übereinstimmen. So konnten die Bilder unter gleichen Lichtbedingungen aufgenommen werden, da keine Umbauarbeiten notwendig waren, sondern die Laser einfach abwechselnd an- und ausgeschaltet wurden. Die aufgenommenen Bilder sind in Abbildung 6.3 auf der nächsten Seite zusammengefasst.

Wie erwartet, sind beide Laserlinien in der hellen Umgebung schwieriger erkennbar als in der dunklen Umgebung. Entgegen den Erwartungen ist allerdings die Laserlinie des schwächeren Picotronic Laser, selbst in der helleren Umgebung, besser erkennbar, als die des Lasiris. Der Grund dafür ist, dass die Laserlinie beim Picotronic breiter ist als beim Lasiris. Wegen dieser breiteren Laserlinie ist die Erkennbarkeit auch trotz der geringeren Leistung gleich gut gewährleistet.

Für die Berechnung der Tiefenbilder muss die Mitte der Laserlinie ermittelt werden. Mit Hilfe verschiedener Methoden [44] ist es möglich, die Linienmitte mit einer Subpixel-

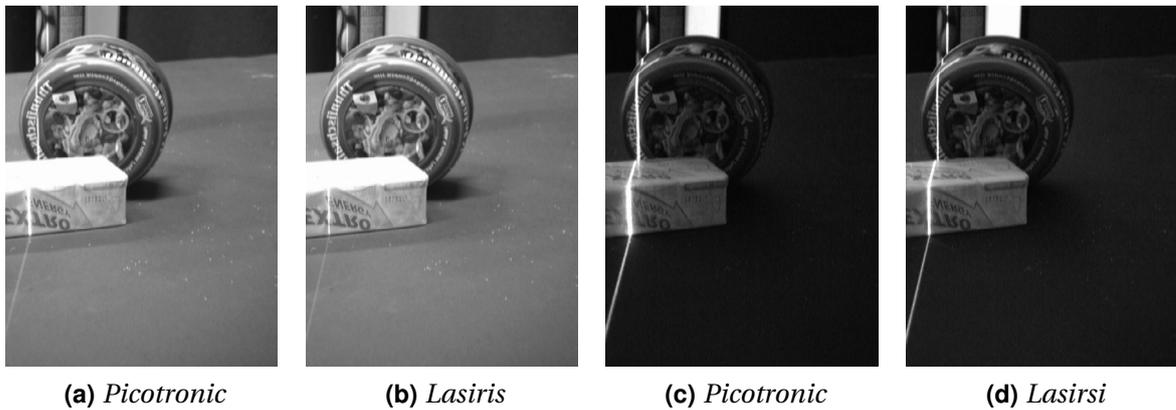


Abbildung 6.3: Vergleich zweier Linienlaser bei unterschiedlichen Lichtverhältnissen: Trotz geringerer Leistung ist die Laserlinie vom Picotronic sogar besser erkennbar, als die des Lasiris. Die Bilder (a) und (b) wurden bei künstlicher Beleuchtung erstellt, die Bilder (c) und (d) bei ausgeschaltetem Licht am Abend. Alle Bilder wurden mit einer automatischen Belichtung aufgenommen.

Genauigkeit zu ermitteln. Somit hat eine breitere Laserlinie auch keinen negativen Einfluss auf die Genauigkeit des Tiefenbildes.

6.3 Die Schwenkeinheit

Die Genauigkeit der Schwenkeinheit ist besonders beim Lichtschnittverfahren sehr wichtig, wird hier doch die Winkelposition direkt für die Berechnung des Tiefenbildes herangezogen. Wegen der Schwierigkeiten bei der Suche nach einer geeigneten Ansteuerung der Schwenkeinheit musste als beste Alternative ein RC-Servo für den Antrieb gewählt werden. Da dieser jedoch keine Positionsrückmeldung bietet, ist es wichtig, dass dieser sehr genau den Vorgaben der Steuerung folgt. Diese Genauigkeit galt es in einem Versuch zu ermitteln. Die Vorgehensweise und die Testergebnisse werden in den folgenden Punkten näher beschrieben.

6.3.1 Versuchsaufbau

Zur Ermittlung der Genauigkeit der Schwenkeinheit muss die tatsächliche Winkelposition mit der vorgegebenen verglichen werden. Wegen der großen Anzahl an möglichen Positionen – es sind 2.701 verschiedene Position einstellbar – ist eine automatische Messdatenerfassung und -auswertung wünschenswert. Eine Möglichkeit zur automatischen Messdatenerfassung wäre es, einen Winkelsensor am Sensor zu befestigen und damit die Position zu ermitteln. Da ein solcher Sensor allerdings nicht verfügbar war, und die Befestigung auch nicht so einfach möglich gewesen wäre, wurde nach einer anderen Möglichkeit gesucht.

Bei einem optischen Sensor bietet sich eine optische Messmethode zur Ermittlung der Position an. Das gewählte Prinzip funktioniert derart, dass der Sensor auf eine zusätzliche Schwenkeinheit montiert wird, deren Position sich ermitteln lässt. Bewegt sich diese zusätzliche Schwenkeinheit gegengleich zum Sensor, so sollten die Kameras immer in die gleiche Richtung weisen. Eine Abweichung der eingestellten Position resultiert somit auch in einer Verschiebung in den Kamerabildern. Bei bekannten Abmessungen der Versuchsanordnung und der, von der Kamera erfassten, Objekte kann die Winkelabweichung zwischen zwei Bildern ermittelt werden. Durch Berechnung der Winkelfehler für alle Positionen kann die Positioniergenauigkeit der Schwenkeinheit ermittelt werden.

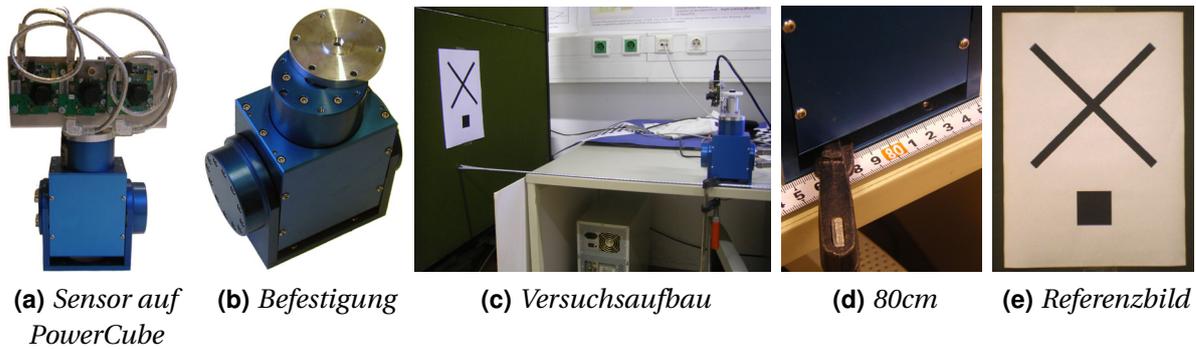


Abbildung 6.4: Versuchsaufbau zum Test der Schwenkeinheit: (a) Zum Testen der Schwenkeinheit wird der Sensor auf einer Schwenkeinheit vom Typ PowerCube befestigt, die den Sensor gegengleich zu dessen Bewegung dreht. (b) Die Befestigung des Sensors erfolgt dabei mit einer eigens gefertigten Scheibe, die exakt auf den PowerCube passt. (c) Für den Versuch wurde der Sensor vor einer Wand positioniert. (d) Für die Auswertung der Daten wurde der Abstand zur Wand auf 80cm festgelegt. (e) Die mittige Kamera erfasste das an der Wand befestigte Blatt Papier. Das Quadrat mit der Seitenlänge von 40mm dient als Größenreferenz und das Kreuz zur Positionsbestimmung.

Als zusätzliche Schwenkeinheit wird ein AMTEC PowerCube Schwenk-/Neigeeinheit, wie sie auch beim bestehenden System [20] zum Einsatz kommt, verwendet. Diese Schwenkeinheit kann – dank des integrierten optischen Winkeldecoders – die Position mit einer Genauigkeit von 555 Schritten je Grad bestimmen [63]. Angesteuert wird das PowerCube Modul über CAN-Bus. Dazu wurde ein CAN-USB Adapter der Firma *esd electronic system design gmbh* vom Typ *CAN-USB-Mini* verwendet. Die Energieversorgung für die Schwenk-/Neigeeinheit erfolgte dabei allerdings nicht per USB, sondern mit Hilfe eines externen Netzgerätes.

Zur Befestigung des Sensors an der Schwenkeinheit wurde eine eigene Befestigungsscheibe angefertigt, die genau auf die Abmessungen der Schwenkeinheit abgestimmt wurde, damit die Drehachse des Sensors mit der des PowerCube übereinstimmt. Dieser Aufbau wurde in einer Entfernung von 80cm zu einer Wand auf dem Tisch befestigt, wobei die Entfernung von der Wand zur Drehachse gemessen wurde. An der Wand wurde ein Blatt Papier aufgeklebt, auf dem ein Quadrat mit 40mm Seitenlänge als Referenz und ein, um 45° gedrehtes Kreuz abgebildet waren. Der Versuchsaufbau ist in Abbildung 6.4 dargestellt.

6.3.2 Versuchsablauf und Auswertung

Der Versuch umfasst mehrere Schritte von der Datenaufnahme bis zum Endergebnis. Im konkreten wurden folgende Schritte ausgeführt:

1. Aufnahme der Bilder und Speicherung der Positionsdaten:

Ein selbsterstelltes Programm steuert die beiden Schwenkeinheiten und speichert für jede Position die Winkelwerte und ein Bild. Dabei wurden die Daten für alle Positionen für schrittweise Erhöhung der Vorgabewerte und in einem zweiten Versuch auch für die Verringerung der Werte aufgenommen.

2. Wahl eines Referenzbildes:

Aus allen aufgenommenen Bildern wurde jenes Bild gewählt, bei dem der Mittelpunkt des Kreuzes am nächsten zur Bildmitte ist. Zur Ermittlung des Mittelpunktes wurde der selbe Algorithmus verwendet, der auch bei der nachfolgenden Ermittlung der Abweichungen verwendet wurde.

3. Ermittlung der Pixelabweichung zwischen dem Referenzbild und aller aufgenommenen Bilder:

Mit Hilfe von Bildverarbeitungsfunktionen der OpenCV-Programmbibliothek wird der Mittelpunkt des Kreuzes der Bilder ermittelt und mit dem Mittelpunkt des Referenzbildes verglichen.

4. Umrechnung von Pixelabweichung auf Winkelabweichung:

Aus dem Referenzbild wird die Größe des Quadrates in Pixel bestimmt und daraus mit Hilfe des Abstandes zum Bild (80cm) der Umrechnungsfaktor zwischen Pixel und Winkel ermittelt.

$$\frac{\text{Winkel}}{\text{Pixel}} = \frac{\arctan\left(\frac{\text{Abstand zum Bild}}{\text{Seitenlänge des Quadrates}}\right)}{\text{Seitenlänge des Quadrates in Pixel}}$$

5. Berechnung des Winkelfehlers:

Die Position, die der Winkelencoder des PowerCube-Moduls liefert, stimmt nicht exakt mit der vorgegebenen Position überein. Deshalb ergab sich der Gesamtfehler aus der Abweichung zwischen Vorgabewert und der tatsächlichen Position des PowerCube sowie aus der zuvor ermittelten Winkelabweichung aus den Bildern. Die Berechnung dieser Fehler erfolgte mit einem Tabellenkalkulationsprogramm.

6. Ermittlung eines konstanten Verschiebungsfaktors für mittelwertfreie Fehler:

Die bis zu diesem Punkt ermittelten Fehler beziehen sich immer auf das eine Referenzbild. Es wird also angenommen, dass die zugehörige Position fehlerfrei ist. Zur Korrektur dieser falschen Annahme wurde mit Hilfe des Programms *Gnuplot* der Mittelwert der berechneten Fehler ermittelt. Für diese Berechnung wurden die Werte für beide Drehrichtungen herangezogen. Anschließend wurden alle Fehlerwerte um den, dabei ermittelten Wert korrigiert. Somit sind im Endergebnis die Fehler gleichmäßig um die Nulllinie verteilt.

Ermittlung der Pixelabweichung

Zur Ermittlung der Pixelabweichung wurde zuerst ein Ansatz basierend auf *Optical Flow* Methoden gewählt, der aber wegen unzureichender Ergebnisse durch einen weiteren Ansatz ersetzt werden musste.

Der Optical Flow – zu deutsch *Optischer Fluss* – repräsentiert die Bewegung der einzelnen Bildpunkte zwischen Bildern einer Bildsequenz. In OpenCV dient die Funktion *cvCalcOpticalFlowPyrLK* zur Berechnung des Optical Flow für zuvor im ersten Bild mit der Funktion *cvGoodFeaturesToTrack* gefundenen markanten Bildpunkte. Mit der Funktion *cvFindCornerSubPix* ließe sich die Position der Punkte zusätzlich noch auf Subpixel, als auf Bruchteile eines Pixel, genau bestimmen, wodurch die Pixelabweichung ebenfalls auf Subpixel genau ermittelt werden könnte. Leider findet die Funktion *cvGoodFeaturesToTrack* nicht wie ursprünglich angenommen nur Eckpunkte, sondern auch Punkte auf Kanten, wie zum Beispiel am Blattrand oder an den Konturen der aufgedruckten Objekte. Da es für solche Punkte nicht korrekt möglich ist, im zweiten Bild den korrespondierenden Punkt zu finden, liefert diese Methode für solche Punkte falsche Pixelabweichungen. Einige dieser falschen Werte könnten zwar herausgefiltert werden, wenn die Bewegung nicht in Richtung der Drehbewegung erfolgt. Es ist jedoch nicht möglich, die falschen Werte exakt herauszufiltern und die richtigen Werte zu belassen. Es wird immer Fälle geben, wo noch falsche Werte verbleiben,

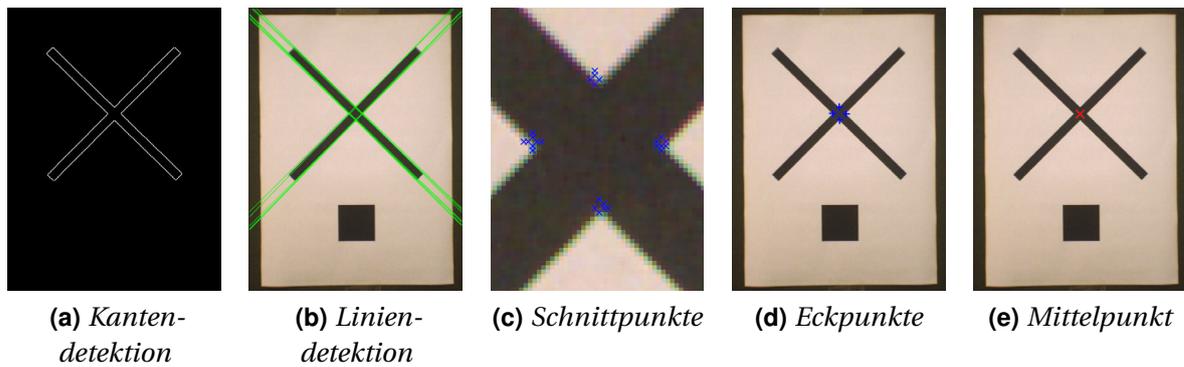


Abbildung 6.5: Ermittlung der Pixelabweichung: (a) Im ersten Schritt wird auf das Bild ein Algorithmus zur Kantendetektion nach der Canny Methode angewendet. (b) Nach der Kantendetektion werden mit Hilfe eine Hough-Transformation Linien gesucht. (c) Die Berechnung aller Schnittpunkte zwischen den 45° und den 135° Linien ergibt viele Punkte (Bild zehnfach vergrößert). (d) Aus den entsprechenden Schnittpunkten werden Mittelwerte für die vier Eckpunkte ermittelt. (e) Der Mittelwert der vier Eckpunkte ist der Mittelpunkt des Kreuzes.

die das Ergebnis verfälschen. In anderen Fällen wäre es denkbar, dass auch alle richtigen Werte als vermeintlich falsch herausgefiltert werden, wodurch gar kein Ergebnis für diese Fälle zustande kommen kann.

Für brauchbare Ergebnisse musste daher ein anderer Ansatz gewählt werden. Dieser Ansatz basiert auf *Liniendetektion* und die Berechnung der Pixelabweichung erfolgt aus der Verschiebung der Schnittpunkte der Linien. Dabei wird der Mittelpunkt des Kreuzes bestimmt und mit dem des Referenzbildes verglichen. Die Bestimmung erfolgt in folgenden Schritten:

1. Kantendetektion mittels Canny Methode mit *cvCanny* (Abbildung 6.5a).
2. Suche nach Linien mit Hilfe der Hough-Transformation mit der Funktion *cvHoughLines2* (Abbildung 6.5b).
3. Filterung der, für das gedrehte Kreuz in Frage kommenden, Linien anhand des Winkels. Nur Linien mit annähernd 45° beziehungsweise 135° kommen in Frage.
4. Berechnung der Schnittpunkte zwischen den 45° und den 135° Linien (Abbildung 6.5c).
5. Ermittlung des Wertebereichs aller berechneten Schnittpunkte und Zuordnung der Punkte zu vier Bereichen (oben, unten, links und rechts).
6. Mittelwertbildung für die vier Bereiche zur Bestimmung der vier Schnittpunkte (Abbildung 6.5d).
7. Mittelwertbildung aus den Schnittpunkten zur Bestimmung der Mitte (Abbildung 6.5e).

Dabei wird für den Fall, dass wegen einer zu großen Bildabweichung keine vier Schnittpunkte gefunden werden können, der Mittelpunkt aus den verbleibenden Punkten trotzdem berechnet, dem Ergebnis aber auch die Zahl der verwendeten Schnittpunkte hinzugefügt, um bei der späteren Auswertung diese Ergebnisse als falsch verwerfen zu können.

6.3.3 Ergebnisse

Der oben beschriebene Versuchsablauf wurde bei Rechtsdrehung und auch bei Linksdrehung ausgeführt. Da die ersten Ergebnisse bezüglich der Positioniergenauigkeit nicht zufriedenstellend waren, wurde zum Vergleich der Versuch noch einmal mit einem digitalen Servo

durchgeführt, um Vergleichswerte zwischen analogen und digitalen Servos zu erhalten. In Tabelle 6.2 sind einige technische Daten der beiden Servos (Abbildung 6.6) aufgelistet.

Eigenschaft	HS-322HD	HS-5245MG Digi
Arbeitsweise	analog	digital
Geschwindigkeit [sec/60°]	0,19	0,15
Drehmoment [Ncm]	30	44
Haltemoment [Ncm]	—	88
Stromaufnahme [mA/60°]	160	250
Blockierstrom [mA]	700	—

Tabelle 6.2: Die wichtigsten technischen Daten der beiden verwendeten RC-Servos.

RC-Servos besitzen eine interne Positionsregelung (vgl. Kapitel 4.3.1), die durch ein einfaches Vergleichen von Soll- und Ist-Position die Lage regelt. Diese Regelung funktioniert sowohl bei analogen als auch bei digitalen Servos nach einem simplen Prinzip. Aus dem pulsweitenmodulierten Steuersignal ergibt sich durch Integration der Sollwert. Die Regelung bewegt den Motor solange in die entsprechende Richtung, bis sich am internen Lageregler – ein einfacher Drehpotentiometer – der korrekte Wert einstellt. Dabei wird der Motor jedoch nicht kontinuierlich sondern pulsweise bewegt. Der Unterschied zwischen analogen und digitalen Servos liegt in der Häufigkeit der möglichen Motorimpulse. Während bei analogen Servos alle 20ms ein Motorimpuls möglich ist, ist bei digitalen Servos alle 400µs ein Impuls möglich. Somit erfolgt die Reaktion beim digitalen Servo schneller und genauer. Durch die höhere Pulszahl ist auch ein größeres Drehmoment möglich. Der Nachteil der Digitalservos liegt in einem höheren Strombedarf, was gerade bei der Ansteuerung über USB von Nachteil ist, da hier maximal 500mA an Stromaufnahme möglich sind. Durch diese Begrenzung der erlaubten Stromaufnahme sind nur wenige am Markt erhältliche Digitalservos für den Einsatz an der USB-Schnittstelle geeignet.

Der in Abbildung 6.7 auf der nächsten Seite dargestellte Soll-Ist-Vergleich sollte im Idealfall der Sollkurve folgen. Das würde bedeuten, dass die Schwenkeinheit exakt die eingestellten Positionen einnimmt. Die Messung wurde über den gesamten Winkelbereich der Servosteuerung „PhidgetServo 1-Motor“ (Abbildung 6.6c) durchgeführt. Wie erwartet kann der Servo bei kleinen Winkelwerten nicht die gewünschte Position anfahren, da das Drehmoment zu gering ist. Warum jedoch auch bei hohen Werten (ab 109° für den analogen Servo bzw. 98° für den digitalen) die Servos den Vorgaben nicht mehr folgen können, ist nicht ersichtlich, zumal der erwartete Endwert bei rund 116,5° liegt. Allem Anschein nach erreichen



Abbildung 6.6: Die beiden getestete Servos mit der Steuerung: (a) Der analoge Standardservo HS-322HD von Hitec zeichnet sich durch seinen günstigen Preis aus. (b) Der kleinere Hitec HS-5245MG Digi ist einer der wenigen digitalen RC-Servos, der mit den 450mA, die das Steuermodul liefern kann, arbeitet. (c) Für die Ansteuerung der RC-Servos wird das USB-Modul „PhidgetServo 1-Motor“ verwendet.

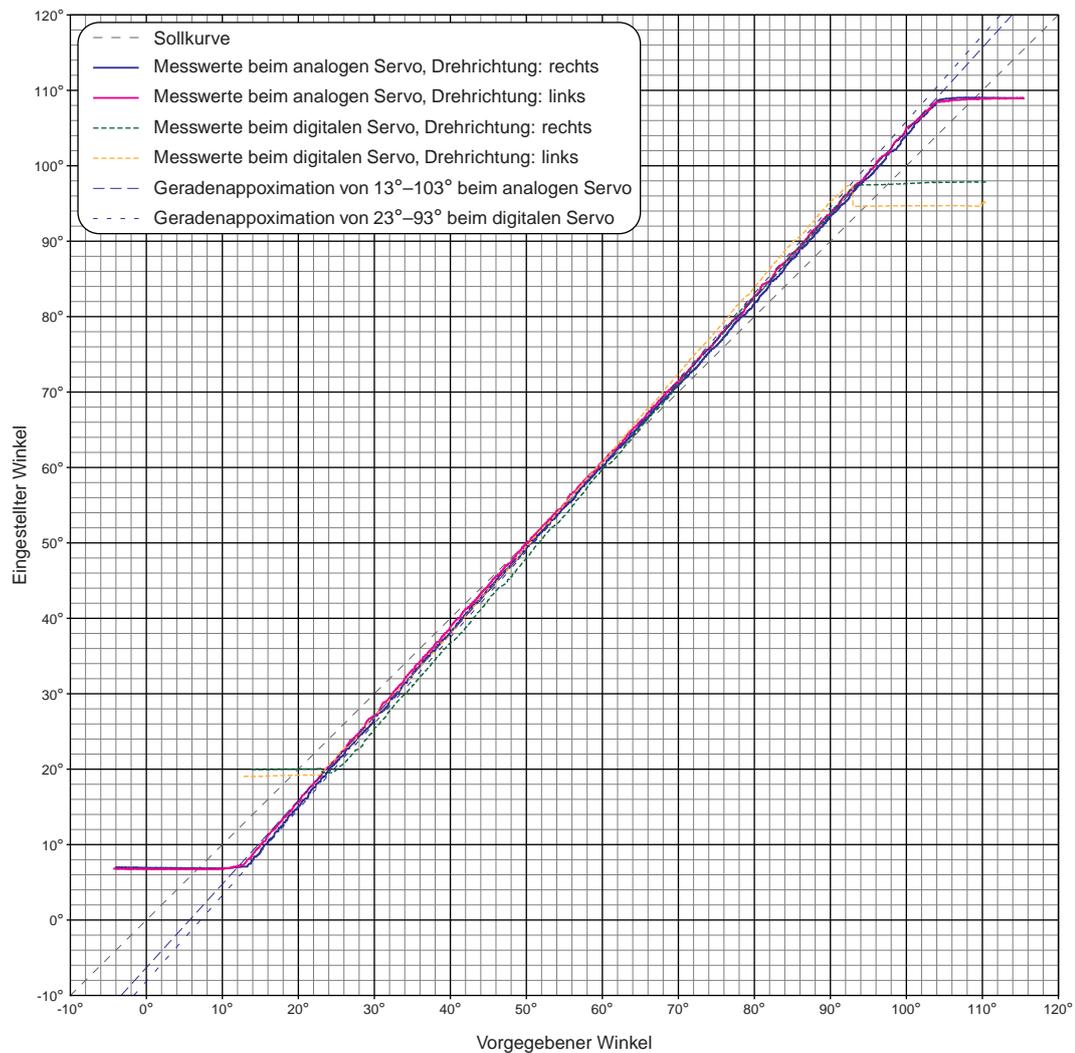


Abbildung 6.7: Soll-Ist-Vergleich bei der Schwenkeinheit: Eigentlich sollten die Kurven der Messwerte der Sollkurve folgen. Die aufgenommene Kurve weist zwar in einem großen Bereich einen annähernd linearen Zusammenhang auf, der aber steiler verläuft als die Sollkurve. Die Messkurven zeigen somit einen Linearitätsfehler.

die Servos ihre interne Endposition jedoch schon bei geringeren Werten.

Dass nicht der gesamte Winkelbereich von $-11,5^\circ$ – $116,5^\circ$ möglich ist, war von Anfang an zu erwarten. Aus diesem Grund ist es nicht weiter problematisch, dass die Endposition schon früher erreicht wird. Zumindest beim analogen Servo ist bei der verwendeten Übersetzung von 2:1 immer noch der geforderte Winkelbereich von 90° möglich.

Problematischer hingegen ist es, dass die Positionen im Bereich zwischen den Endpositionen stark von der Vorgabe abweichen. Besonders auffällig ist es, dass der Zusammenhang zwischen Soll- und Ist-Position zwar linear zu sein scheint, jedoch im Vergleich zur Sollkurve einen steileren Verlauf aufweist. Dieser Linearitätsfehler führt dazu, dass die Abweichung zu den Sollwerten mehrere Grad beträgt (Abbildung 6.8). Allem Anschein nach, ist die Umrechnung zwischen Pulsweite und Winkel bei der Servoansteuerung nicht richtig, was auch die zu niedrige Endposition erklären würde. Dabei wurden mit dem Hitec HS-322HD (analog) und dem HS-5245MG (digital) nur Servos verwendet, die auch vom Hersteller getestet

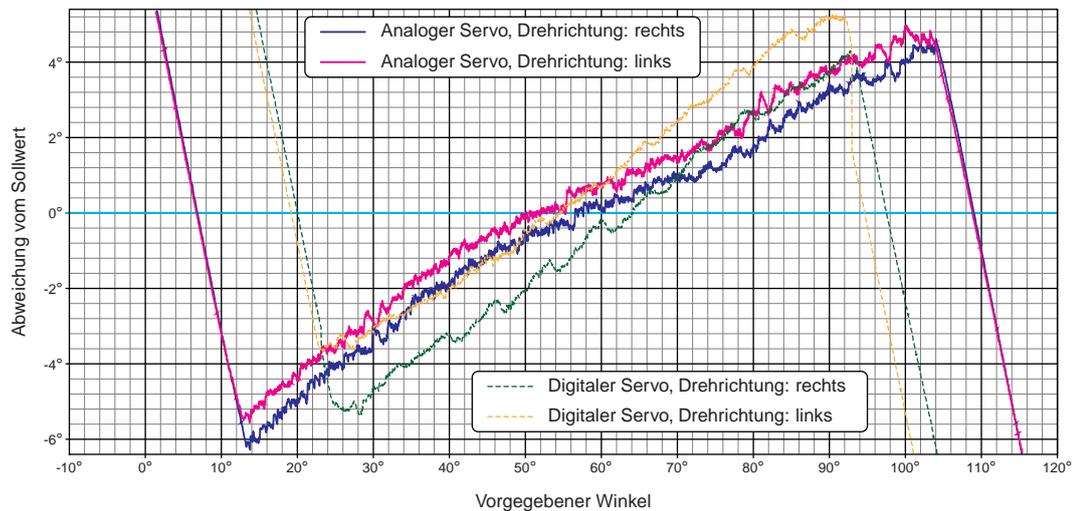


Abbildung 6.8: Positionsabweichung der Schwenkeinheit vom Sollwert: Die Ist-Position weicht bei beiden Servo-Typen stark vom Sollwert ab. Der Grund dafür ist der Linearitätsfehler aus Abbildung 6.7.

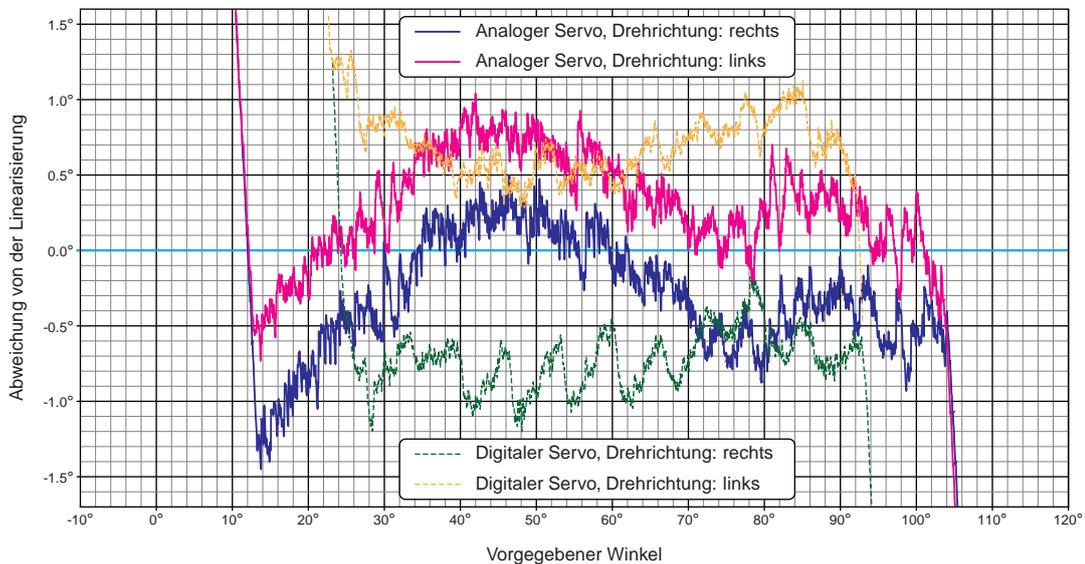


Abbildung 6.9: Positionsabweichung der Schwenkeinheit vom genäherten, linearen Verlauf: Wird die Positionsabweichung nicht zur Sollkurve berechnet, sondern zur angenäherten Geraden, so fallen die Fehler deutlich geringer aus. Beim analogen Servo sind dabei, bei gleichbleibender Drehrichtung, Abweichungen in der Größenordnung von 2° möglich. Beim digitalen Servo beträgt die Abweichung nur rund 1° . Dafür ist beim digitalen Servo der Unterschied zwischen den beiden Drehrichtungen deutlich größer. Während beim analogen Servo dieser Unterschied rund $0,5^\circ$ beträgt, so liegt er beim digitalen Servo bei über 1° .

wurden.³

Zur mathematischen Korrektur des Linearitätsfehlers wurden die Messkurven in den Bereichen laut Abbildung 6.7 einer Geraden angenähert. Diese Approximation erfolgt mit Hilfe des Programms *Gnuplot* und dessen *Fit*-Funktion. Wird der Fehler nicht als Abweichung zur Sollkurve, sondern zur approximierten Gerade gerechnet, so ergibt sich der Fehler-

³ Mit der Version 2.1.6.20090917 der Phidget Programm-Bibliothek wurde eine Unterstützung verschiedener Servotypen eingeführt, die eventuell den Zusammenhang zwischen Pulsweite und Winkel genauer wiedergibt [61, 64].

verlauf aus Abbildung 6.9. Diese Vorgehensweise ist auch durchaus zulässig, da die Korrektur des Linearitätsfehlers auch in der Software möglich wäre. Dazu müssten lediglich die Werte aus Kapitel 5.4 angepasst werden.

Was in Abbildung 6.9 auch besonders stark auffällt, ist der Unterschied zwischen den Drehrichtungen. Beim digitalen Servo variieren die Fehler bei gleichbleibender Drehrichtung um rund 1° , während diese Abweichung beim analogen Servo beinahe 2° beträgt. Beim Vergleich von Rechts- und Linkslauf zeigen sich allerdings umgekehrte Verhältnisse. Beim digitalen Servo unterscheiden sich die Messergebnisse für die beiden Drehrichtungen um mehr als 1° , während sich die Ergebnisse beim analogen Servo um nur rund $0,5^\circ$ unterscheiden. Ein möglicher Grund dafür könnte die Befestigung des Servos sein. Die Befestigung des Servos erfolgt in einer extra angefertigten Vorrichtung (Abbildung 4.10b auf Seite 45), die den Servo mit dem Sensorgehäuse verbindet. Diese Vorrichtung ist allerdings nur an der Unterseite mit dem Gehäuse verbunden, wodurch sich eine große Hebelwirkung ergibt. Der Abstand zwischen den Befestigungspunkten und dem Kraftangriffspunkt beträgt rund 45mm. Beim Versuch, den Sensor per Hand zu bewegen, lässt sich deutlich ein Verdrehen der Servobefestigung erkennen. Eine Verbesserung würde eine andere Befestigungsmöglichkeit des Servos, wie sie auch schon für die Finalversion vorgeschlagen wurde (vgl. Kapitel 4.5.2), mit sich bringen.

6.3.4 Fazit

Wegen der fehlenden Positionsrückmeldung des RC-Servos wurde von Beginn an nur eine mäßige Genauigkeit der Schwenkeinheit erwartet. Die Tests haben gezeigt, dass selbst nach Korrektur des Linearitätsfehlers eine Ungenauigkeit von rund $\pm 1,5^\circ$ bleibt. Selbst bei nur einer Drehrichtung und der Verwendung eines digitalen Servos bleibt eine Ungenauigkeit von $\pm 0,5^\circ$.

Die Schwenkeinheit ist daher eine der Komponenten, die beim Bau einer verbesserten Version des Sensors, geändert werden sollte. Mehr zu den Verbesserungsvorschlägen gibt es im folgenden Kapitel.

7

7 Erkenntnisse und Verbesserungen

Bei dem gebauten Sensor handelt es sich lediglich um einen Prototypen. Die Endversion des Sensors soll sich zumindest dadurch vom Prototypen unterscheiden, dass sie ein geschlossenes Gehäuse besitzt, das die Komponenten vor äußeren Einflüssen schützt. Im Allgemeinen werden Prototypen zur Erprobung verschiedener Eigenschaften gebaut, wobei die Testergebnisse in die Entwicklung der Endversion einfließen. Aus diesem Grund finden sich in diesem Kapitel abschließend Verbesserungsvorschläge für die bereits am Prototypen festgestellten Mängel. Dabei werden für fast alle Komponenten mögliche Änderungen vorgeschlagen.

7.1 Die Spannungsversorgung

Die Spannungsversorgung des Linienlasers hat, wie in Kapitel 6.1 bereits beschrieben, das Problem, dass das FireWire-Signal nicht korrekt durchgeschliffen wird. Auch wenn das nicht zwingend notwendig ist, so sollte dies bei der Entwicklung eines verbesserten Sensors berücksichtigt werden.

Was für eine Verbesserung jedenfalls berücksichtigt werden sollte, ist eine Möglichkeit, die Spannungsversorgung des Lasers softwaremäßig schalten zu können. Somit könnte ganz einfach softwaremäßig zwischen den beiden Betriebsarten Stereokamera und Lichtschnittverfahren gewechselt werden. Ohne diese Möglichkeit muss der Laser weiterhin manuell aus beziehungsweise eingeschaltet werden.

Um beide Verbesserungen umzusetzen, muss die Schaltung erweitert werden. Während zum Durchschleifen des FireWire-Signals lediglich eine ausreichende Schirmung erforderlich ist, so sind zum Schalten der Spannungsversorgung schon mehr Maßnahmen zu treffen. Damit ein FireWire-Signal zum Schalten der Spannungsversorgung verwendet werden kann, muss ein Mikrocontroller mit einer zusätzlichen FireWire-Schnittstelle verbaut werden. Dieser Aufwand mag für das Schalten der Spannungsversorgung als zu groß angesehen werden, in Kombination mit weiteren Änderungen am Sensor kann sich der Aufwand jedoch schnell lohnen. So kann diese Änderung beispielsweise mit einer neuen Steuerung der Schwenkeinheit kombiniert werden.

7.2 Die Schwenkeinheit

Wird für die Spannungsversorgung ein Mikrocontroller mit FireWire-Schnittstelle verwendet, so kann dieser gleichzeitig auch zur Ansteuerung einer neuen, verbesserten Schwenkeinheit

verwendet werden. Die Schwenkeinheit stellt beim gebauten Sensor die größte Schwachstelle dar. Wie in Kapitel 6.3 ausgiebig erläutert wurde, arbeitet die Schwenkeinheit zu ungenau. Vor allem beim Lichtschnittverfahren, wo die Position der Schwenkeinheit benötigt wird, sind keine guten Ergebnisse im Bezug auf die Tiefenbilder zu erwarten. Aus diesem Grund ist eine Überarbeitung der Schwenkeinheit unbedingt erforderlich.

Neben der Ungenauigkeit, hervorgerufen durch die fehlende Positionsrückmeldung, hat die derzeitige Schwenkeinheit auch noch den Nachteil, dass sie nicht über FireWire steuerbar ist. Auch dieser Nachteil ließe sich durch den Bau einer eigenen Ansteuerung mit FireWire-Schnittstelle und integrierter Spannungsversorgung beheben.

Ein RC-Servo als Antrieb ist für eine neue Schwenkeinheit eine schlechte Wahl. Die neu zu bauende Schaltung muss gleichzeitig auch den Antrieb regeln können, was mit einem Servo wegen der integrierten Regelung kaum möglich ist. Der Neubau einer Schwenkeinheit bietet die Möglichkeit, andere Antriebe wie Schrittmotor oder Gleichstrommotor zu verwenden (vgl. Kapitel 4.3).

Neben einem neuen Antrieb ist auch zwingend ein Drehwinkelsensor erforderlich. Dieser wird zum einen dazu benötigt, um den Antrieb regeln zu können. Zum anderen liefert der Drehwinkelsensor auch die aktuelle Position, wodurch die Genauigkeit der Tiefenbilder erhöht werden kann. Als Drehwinkelsensoren kommen unterschiedliche Typen in Frage, wie sie in Abbildung 4.3 auf Seite 39 dargestellt sind. Bei der Wahl sind wie üblich der Preis, die Genauigkeit und auch die Auswertemöglichkeit zu berücksichtigen. Drehpotentiometer sind günstiger, arbeiten allerdings nicht so exakt wie digitale Drehwinkelsensoren und benötigen – als analoge Sensoren – eine Analog-Digital-Wandlung bei der Verwendung in einer digitalen Regelung. Die Wahl richtet sich jedoch auch stark nach den verwendeten Mikrocontrollern und deren Anschlussmöglichkeiten, weshalb hier keine Empfehlung abgegeben werden kann.

Durch den neuen Antrieb ändern sich auch die mechanischen Komponenten der Schwenkeinheit. Die Kraftübertragung zwischen Antrieb und den feststehenden Teilen des Sensors muss an den neuen Antrieb angepasst werden. Dabei sind Zahnräder weiterhin zu bevorzugen. Es ist allerdings auch denkbar, dass der Antrieb direkt das Verbindungsstück zwischen dem feststehenden und dem schwenkbaren Teil des Sensors darstellt. Dabei könnte das Gehäuse des Antriebs mit dem bisher schwenkbaren Teil des Sensors verbunden werden und die Antriebswelle müsste dann entsprechend mit dem feststehenden Teil verbunden werden (Abbildung 7.1 auf der nächsten Seite). Bei dieser Methode ist keine weitere Lagerung notwendig, was zu einer Kostenreduktion führt.

Ein weiteres Vorteil, den eine neue Schwenkeinheit mit FireWire-Schnittstelle bieten würde, ist, dass dadurch der USB-Anschluss wegfallen würde, was der Anforderung nach nur einen einzigen Anschluss (vgl. Kapitel 3.2) entsprechen würde.

Für eine genauere Schwenkeinheit, die auch besser den Anforderungen genügt, sind daher folgende Maßnahmen zu treffen:

- Bau einer Ansteuerung mit FireWire-Anschluss
- Verwendung eines anderen Antriebes
- Den Antrieb direkt zum Schwenken verwenden
- Einsatz eines Drehwinkelsensors zur exakten Positionsbestimmung

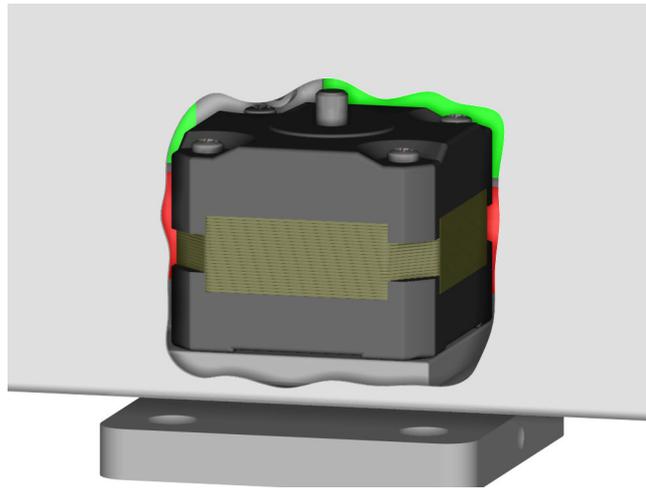


Abbildung 7.1: Schwenkeinheit ohne zusätzlicher Lagerung: Wird ein Antrieb verwendet, der direkt eine Verbindung zwischen dem drehenden und dem feststehenden Teil des Sensors herstellen kann, so ist keine zusätzliche Lagerung der Schwenkeinheit erforderlich, da diese vom Antrieb bereitgestellt wird. Die Darstellung zeigt einen handelsüblichen Schrittmotor.

7.3 Die Kameras

Die verwendeten Unibrain Fire-i OEM Kameramodule zeichnen sich durch den konkurrenzlos niedrigen Preis aus. Die Qualität der Bilder ist erwartungsgemäß nicht so gut wie bei vergleichbaren Industriekameras, die allerdings ein vielfaches der Unibrain Kameramodule kosten (vgl. Tabelle 4.1 auf Seite 36). Die Qualität der Kamerabilder ist dennoch ausreichend, was aber doch störend wirkt sind die Objektive, beziehungsweise deren Befestigungsmöglichkeit. Das Objektivgehäuse besteht aus Kunststoff und wird einfach in eine Kunststofffassung am Kameramodul geschraubt. Diese Schraubverbindung weist allerdings ein vergleichsweise hohes Spiel auf. Das bedeutet, die Objektive sitzen nicht fest in der Fassung und bewegen sich bei Berührung, was den Nachteil hat, dass sich die Parameter der Kamera kalibrierung ändern und eine erfolgte Kalibrierung somit unbrauchbar wird.

Zur Behebung des Problems mit den nicht feststehenden Objektiven kann das Objektiv festgeklebt werden. Auch ein Dichtungsband kann dabei helfen, den Gewindedurchmesser zu erweitern und dadurch das Spiel zu verringern. Diese Methoden haben allerdings den Nachteil, dass sich das Objektiv nur mehr schwer bis gar nicht wechseln beziehungsweise die Brennweite verstellen lässt. Eine andere Möglichkeit ist es, den von Unibrain vertriebenen C-Mount Adapter zu verwenden. Dieser ermöglicht es, Objektive anderer Hersteller mit dem genormten C-Mount Anschluss zu verwenden. Die meisten Industriekameras verwenden solche Objektive, die auch am Institut vielfach eingesetzt werden. So kann auch zwischen vielen verschiedenen Objektiven mit unterschiedlichen Parametern und in unterschiedlicher Qualität gewählt werden.

Neben leichten Qualitätsmängel weisen die Kameras jedoch einen entscheidenden Nachteil auf. Die beiden gegenüberliegenden FireWire-Anschlüsse (siehe Abbildung 7.2a) bewirken, dass die überstehenden Stecker der Kabel den Platzbedarf je Kamera mehr als verdoppeln. Dies führt dazu, dass das Gehäuse für den Sensor sehr hoch sein muss, damit die Kabel darin Platz finden. Dafür bietet Unibrain jedoch auch Lösungen. Zum einen gibt es die Version „Remote CCD“ der Kameramodule (Abbildung 7.2b), bei denen der CCD-Chip und das Objektivgehäuse auf einer eigenen kleinen Platine sitzen und über ein flexibles Kabel

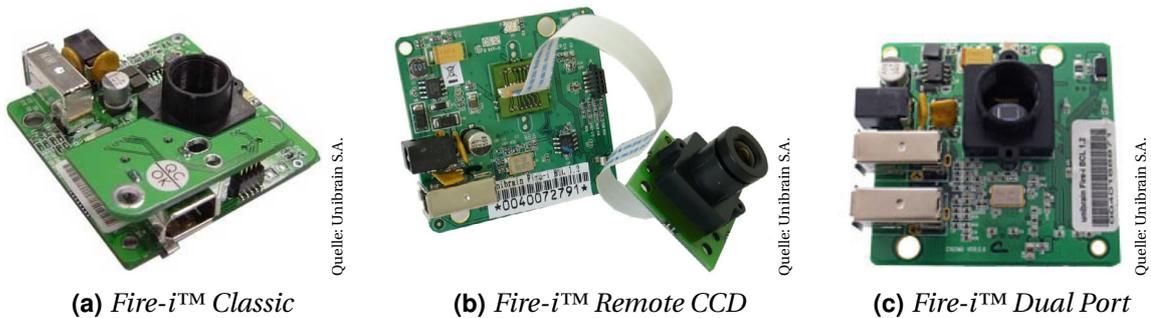


Abbildung 7.2: Unterschiedliche Ausführungen der Fire-*i*TM Kamera: (a) Die beim Sensor verwendeten Kameramodule von Unibrain aus der Modellreihe Fire-*i*TM nehmen wegen der, nach beiden Richtungen weggehenden Kabel, viel Platz ein. (b) Eine Möglichkeit Platz zu sparen wäre die Verwendung der Remote CCD Modelle. In diesem Fall könnten die Kameramodule hintereinander montiert werden, während die kleineren CCD Module den Platz der bisher verwendeten Kameras einnehmen. (Im Bild ist der zweite FireWire-Anschluss nicht dargestellt.) (c) Auch mit der Dual Port Ausführung kann Platz gespart werden, da die Kabel nur in eine Richtung weggehen.

mit dem Kameramodul verbunden werden. Eine zweite Möglichkeit bietet die neuere „Dual Port“ Version der Kamera (Abbildung 7.2c). Bei dieser Ausführung sitzen die beiden FireWire-Anschlüsse an der selben Seite des Kameramoduls. Diese Version erlaubt jedoch nicht den Einsatz des C-Mount Adapters.

Bei der Verwendung der „Remote CCD“ Version ist es möglich, die Kameralinsen vorne am Sensor zu befestigen und den Rest platzsparend dahinter. Da die Schwenkeinheit aber auch einen Platz benötigt, wird sich dadurch die Tiefe des Sensors etwas erhöhen. Für die „Dual Port“ Version sind vom Aufbau her nur geringe Änderungen am Sensor nötig. Die Höhe des Sensors kann jedoch verringert werden, da bei den Kameras die Kabel nur mehr an einer Seite angeschlossen werden müssen.

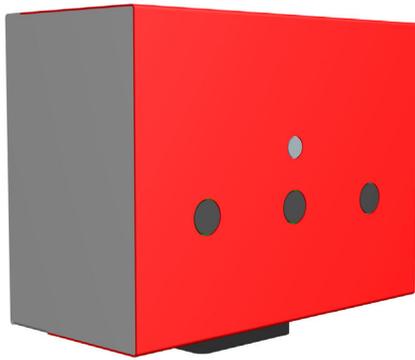
Der Preisunterschied zwischen der Classic-Variante und der „Remote CCD“ Version ist gering. Bei den Farbkameras beträgt der Aufpreis € 20 und bei der Monochromkamera nur € 10. Die Dual Port Ausführung wird allerdings nur auf Bestellung produziert, weshalb dafür der Preisunterschied – vor allem bei der Abnahme von nur wenigen Stücken – verhältnismäßig groß sein wird.

7.4 Das Gehäuse und der Rahmen

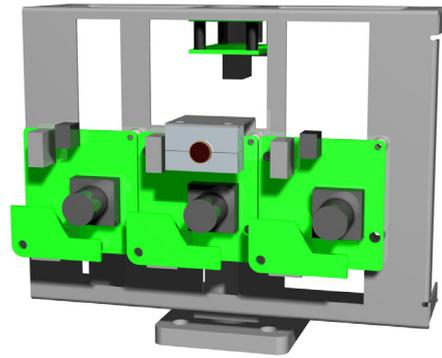
Der gebaute Prototyp besitzt kein geschlossenes Gehäuse, sondern nur einen Rahmen, auf dem Kameras, Laser und Schwenkeinheit befestigt sind. Wie schon in Kapitel 4.5.2 beschrieben, soll die Endversion des Sensors jedoch ein geschlossenes Gehäuse besitzen. Aus Kostengründen ist geplant, diese Gehäuse, ebenso wie den Rahmen, aus Blech zu fertigen.

Wie in Kapitel 6.3.3 bereits festgestellt wurde, muss beim Rahmen bei Verwendung des RC-Servos unbedingt darauf geachtet werden, dass der Servo stabil befestigt ist und sich trotz Gegenkraft nicht bewegt. Ansonsten führt diese Bewegung zu noch ungenaueren Positionswerten bei der Schwenkeinheit.

Die genaue Festlegung von Rahmen und Gehäuse ist erst möglich, wenn Detailzeichnungen der verwendeten Komponenten mit den entscheidenden Anschlussmaße vorliegen.



(a) Entwurf mit dem Gehäuse



(b) Entwurf mit abgenommenen Gehäuse

Abbildung 7.3: Rahmen und Gehäuse ohne Berücksichtigung von Verbesserungen: (a) Das entworfene, geschlossenes Gehäuse schützt die Komponenten vor äußeren Einflüssen und verleiht dem Sensor ein professionelles Aussehen. (b) Ohne dem Gehäuse zeigt sich der hohe Platzbedarf wegen der FireWire-Kabel. Nach unten wird wegen der Verwendung einseitig gewinkelter Stecker weniger Platz benötigt.

Aus diesem Grund ist eine Änderung der Konstruktion auch erst sinnvoll, wenn alle vorgesehenen Änderungen genau durchgeplant sind. Die Entwürfe aus Abbildung 7.3 sind für die, beim Prototyp verwendeten Komponenten. Bei diesen Entwürfen wurde unter anderem der Platzbedarf für die Kabel mit berücksichtigt.

8

8 Resümee

Die Forschung arbeitet fieberhaft an der Integration von Robotern in das Alltagsleben der Menschen. Mit Ausnahme einiger „Haushaltsroboter“ für kleine Arbeiten wie Saugen, Wischen oder Rasenmähen existieren noch keine marktreifen Roboter mit dem Potential für gravierende Änderungen unseres Alltages. Der Weg zu intelligenten, mobilen Haushaltshilfen ist noch ein weiter. Ein nötiger Schritt dahin ist die Entwicklung zuverlässiger Sensoren, die dem Roboter eine selbstständige Wahrnehmung ermöglichen.

In dieser Diplomarbeit wurde ein preiswerter Ansatz für einen Tiefenbildsensor vorgestellt, mit dessen Hilfe dreidimensionale Bilder erstellt werden können. Bei der Dimensionierung des Sensors wurde besonders auf die Einsatzfähigkeit in der mobilen Robotik geachtet. Durch die vorwiegende Verwendung bereits existierender Komponenten wurde sichergestellt, dass der Preis für den Sensor auch als Einzelstück gering ist. Somit kann der Sensor sehr gut im Forschungsbereich zur Erprobung von Bildverarbeitungsverfahren eingesetzt werden. Basierend auf den dabei gewonnen Ergebnissen lässt sich das Konzept später noch weiter verbessern.

Die im Zuge der Diplomarbeit aufgetretenen Probleme zeigen sehr deutlich, wie kompliziert es oft ist, verschiedene, vorhandene Komponenten zu einem Gesamtsystem zusammen zu fügen. Vielfach mag es einfacher sein, schlecht passende Komponenten durch Eigenentwicklungen zu ersetzen, stünden dem nicht meist sehr hohe Kosten gegenüber. Die Tatsache, dass es bei dieser Arbeit gelungen ist, einen, nach Lichtschnittverfahren und Stereokamera-Prinzip arbeitenden, preiswerten Tiefenbildsensor zu bauen gibt Hoffnung auf mehr. Bei entsprechend hohen Stückzahlen lässt sich sicher ein noch preiswerterer und zugleich besserer Sensor bauen, für den dann auch viele Bauteile selbst entwickelt werden können. Besonders die Preise der verwendeten Kameramodule erscheinen sehr hoch, wenn zum Vergleich die Preise von verbreiteten Consumer-Produkten wie Webcams oder Mobiltelefone herangezogen werden. Bei hohen Stückzahlen sind hohe Preisnachlässe bei den Kameras zu erwarten. Dabei könnten durchaus auch Eigenentwicklungen von Vorteil sein.

Auch wenn der hier gebaute Tiefenbildsensor wegen der ungenauen Schwenkeinheit beim Lichtschnittverfahren Mängel aufweist, so wurde gezeigt, dass es auch mit sehr geringem Budget möglich ist, einen Tiefenbildsensor zu entwickeln.

Literaturverzeichnis

- [1] K. Čapek: *R.U.R. – Rossumovi Univerzální Roboti: kolektivní drama o vstupní komedii a 3 dejstvích*. Aventinum, Praha, 1920.
- [2] B. Siciliano and O. Khatib (editors): *Springer Handbook of Robotics*. Springer, Berlin, Heidelberg, 2008.
- [3] VDI 2860: *Montage- und Handhabungstechnik; Handhabungsfunktionen, Handhabungseinrichtungen; Begriffe, Definitionen, Symbole*. Gesellschaft Produktionstechnik VDI 2860:1990.
- [4] ISO 10218-1: *Industrieroboter – Sicherheitanforderungen – Teil 1: Roboter*. DIN EN ISO 10218-1:2009-7 (Deutsche Fassung).
- [5] I. Asimov: *Runaround*. Astounding Science Fiction, 29(1), March 1942.
- [6] P. Wippermann, A. Steinle und N. Bolz: *Duden – Wörterbuch der New Economy*. Bibliographisches Institut, Mannheim, Januar 2001.
- [7] G. C. Devol Jr.: *Programmed article transfer*, US 2988237, June 1961.
- [8] H. A. Ernst: *MH-1, a computer-operated mechanical hand*. PhD thesis, Massachusetts Institute of Technology, 1962.
- [9] R. P. Paul: *Robot manipulators : mathematics, programming, and control : the computer control of robot manipulators*. MIT Press, Cambridge, Mass., 1981.
- [10] S. Sanders: *Remote operations for fusion using teleoperation*. Industrial Robot: An International Journal, 33(3): 174–177, 2006.
- [11] G. S. Guthart and J. Salisbury, J. K.: *The Intuitive™ telesurgery system: overview and applications*. In Proc. IEEE International Conference on Robotics and Automation ICRA '00, volume 1, pages 618–621, 2000.
- [12] C. C. Liebe, J. W. Alexander, and L. Scherr: *Sun Imaging through the Martian Atmosphere*. In Proc. IEEE Aerospace Conference, pages 1–11, 2008.
- [13] A. M. Okamura and M. R. Cutkosky: *Feature Detection for Haptic Exploration with Robotic Fingers*. The International Journal of Robotics Research, 20(12): 925–938, December 2001.
- [14] K. S. Roberts: *Robot active touch exploration: constraints and strategies*. In Proc. IEEE International Conference on Robotics and Automation, pages 980–985 vol.2, 1990.
- [15] J.-H. Kim, J.-Y. Kim, and J.-H. Oh: *Adjustment of home posture of a biped humanoid robot using an inertial sensor and force torque sensors*. In Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS 2007, pages 2223–2229, November 2007.

- [16] A. Sol: *Walking Humanoid Robot – Concepts of Control*. Diplomarbeit, Technische Universität Wien, September 2006.
- [17] R. Lange and P. Seitz: *Solid-state time-of-flight range camera*. IEEE Journal of Quantum Electronics, 37(3): 390–397, March 2001.
- [18] T. Oggier, M. Lehmann, R. Kaufmann, M. Schweizer, M. Richter, P. Metzler, G. Lang, F. Lustenberger, and N. Blanc: *An all-solid-state optical range camera for 3D real-time imaging with sub-centimeter depth resolution (SwissRanger™)*. In L. Mazuray, P. J. Rogers, and R. Wartmann (editors): Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, volume 5249, pages 534–545. SPIE, February 2004.
- [19] D. Anderson, H. Herman, and A. Kelly: *Experimental Characterization of Commercial Flash Ladar Devices*. In G. Sen Gupta, S. C. Mukhopadhyay, and C. Messom (editors): International Conference of Sensing and Technology, pages 17–22, Palmerston North, New Zealand, November 2005.
- [20] M. Nössing: *Tiefenbildaufnahme nach dem Lichtschnittverfahren mittels Dreh-/Schwenkeinheit*. Diplomarbeit, Technische Universität Wien, November 2004.
- [21] B. Ackland and A. Dickinson: *Camera on a chip*. In Proc. IEEE International Solid-State Circuits Conference - Digest of Technical Papers. 42nd ISSCC, pages 22–25, 412, 1996.
- [22] W. S. Boyle and G. E. Smith: *Charge coupled semiconductor devices*. Bell System Technical Journal, 49: 587–593, 1970.
- [23] G. F. Amelio, J. Bertram, W. J., and M. F. Tompsett: *Charge-coupled imaging devices: Design considerations*. IEEE Transactions on Electron Devices, 18(11): 986–992, 1971.
- [24] M. F. Tompsett, G. F. Amelio, J. Bertram, W. J., R. R. Buckley, W. J. McNamara, J. Mikkelsen, J. C., and D. A. Sealer: *Charge-coupled imaging devices: Experimental results*. IEEE Transactions on Electron Devices, 18(11): 992–996, 1971.
- [25] D. F. Barbe: *Imaging devices using the charge-coupled concept*. Proceedings of the IEEE, 63(1): 38–67, 1975.
- [26] E. R. Fossum: *CMOS image sensors: electronic camera on a chip*. In Proc. International Electron Devices Meeting, pages 17–25, 1995.
- [27] E. R. Fossum: *CMOS image sensors: electronic camera-on-a-chip*. IEEE Transactions on Electron Devices, 44(10): 1689–1698, 1997.
- [28] A. El Gamal and H. Eltoukhy: *CMOS image sensors*. IEEE Circuits and Devices Magazine, 21(3): 6–20, 2005.
- [29] S. Ohba, M. Nakai, H. Ando, S. Hanamura, S. Shimda, K. Satoh, K. Takahashi, M. Kubo, and T. Fujita: *MOS Area Sensor: Part II – Low-Noise MOS Area Sensor with Antiblooming Photodiodes*. IEEE Journal of Solid-State Circuits, 15(4): 747–752, 1980.
- [30] L. G. McIlrath: *A low-power low-noise ultrawide-dynamic-range CMOS imager with pixel-parallel A/D conversion*. IEEE Journal of Solid-State Circuits, 36(5): 846–853, 2001.
- [31] B. E. Bayer: *Color imaging array*, US 3971065, July 1976.
- [32] P. Vora and C. Herley: *Trade-offs between color saturation and noise sensitivity in image sensors*. In Proc. International Conference on Image Processing ICIP 98, volume 1, pages 196–200 vol.1, 1998.

- [33] D. C. Brown: *Close-Range Camera Calibration*. Photogrammetric Engineering, 31: 855–866, August 1971.
- [34] J. G. Fryer and D. C. Brown: *Lens distortion for close-range photogrammetry*. Photogrammetric Engineering and Remote Sensing, 52: 51–58, 1986.
- [35] G. Bradski and A. Kaehler: *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly Media, Inc., 1st edition, October 2008.
- [36] Z. Zhang: *Flexible camera calibration by viewing a plane from unknown orientations*. In Proc. Seventh IEEE International Conference on Computer Vision, volume 1, pages 666–673, 1999.
- [37] Z. Zhang: *A flexible new technique for camera calibration*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(11): 1330–1334, 2000.
- [38] P. F. Sturm and S. J. Maybank: *On plane-based camera calibration: A general algorithm, singularities, applications*. In Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, volume 1, pages 432–437, 1999.
- [39] Z. Zhang: *Camera calibration with one-dimensional objects*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 26(7): 892–899, 2004.
- [40] R. I. Hartley: *Lines and Points in Three Views and the Trifocal Tensor*. International Journal of Computer Vision, 22(2): 125–140, March 1997.
- [41] R. A. Jarvis: *A Perspective on Range Finding Techniques for Computer Vision*. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-5(2): 122–139, 1983.
- [42] P. J. Besl: *Active, optical range imaging sensors*. Machine Vision and Applications, 1(2): 1–63, June 1988.
- [43] R. Baribeau, M. Rioux, and G. Godin: *Color reflectance modeling using a polychromatic laser range sensor*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 14(2): 263–269, 1992.
- [44] R. B. Fisher and D. K. Naidu: *A Comparison of Algorithms for Subpixel Peak Detection*. In Image technology: advances in image processing, multimedia and machine vision, pages 385–404. Springer, 1996.
- [45] M. Rioux: *Laser range finder based on synchronized scanners*. Applied Optics, 23(21): 3837–3844, November 1984.
- [46] M. Rioux, F. Blais, J. Beraldin, and P. Boulanger: *Range imaging sensors development at NRC Laboratories*. In Proc. Workshop on Interpretation of 3D Scenes, pages 154–160, 1989.
- [47] F. M. Wahl: *A Coded Light Approach for Depth Map Acquisition*. In Mustererkennung 1986, 8. DAGM-Symposium, pages 12–17, London, UK, 1986. Springer-Verlag.
- [48] A. J. P. Haasteren, van and H. J. Frankena: *Real-time displacement measurement using a multicamera phase-stepping speckle interferometer*. Applied Optics, 33(19): 4137–4142, July 1994.

- [49] G. Sansoni, S. Corini, S. Lazzari, R. Rodella, and F. Docchio: *Three-dimensional imaging based on Gray-code light projection: characterization of the measuring algorithm and development of a measuring system for industrial applications*. Applied Optics, 36(19): 4463–4472, July 1997.
- [50] M. Sjö Dahl and P. Synnergren: *Measurement of Shape by Using Projected Random Patterns and Temporal Digital Speckle Photography*. Applied Optics, 38(10): 1990–1997, April 1999.
- [51] H. Takasaki: *Moiré Topography*. Applied Optics, 9(6): 1467–1472, June 1970.
- [52] A. Asundi: *Computer aided moiré methods*. Optics and Lasers in Engineering, 18: 213–238, 1993.
- [53] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse: *MonoSLAM: Real-Time Single Camera SLAM*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 29(6): 1052–1067, 2007.
- [54] R. I. Hartley: *Theory and Practice of Projective Rectification*. International Journal of Computer Vision, 35(2): 115–127, November 1999.
- [55] J.-Y. Bouguet: *Camera Calibration Toolbox for Matlab*. Website, June 2008. http://www.vision.caltech.edu/bouguetj/calib_doc/, visited on 15th May 2009.
- [56] N. Ayache and C. Hansen: *Rectification of images for binocular and trinocular stereo-vision*. In Proc. th International Conference on Pattern Recognition, volume 1, pages 11–16, November 1988.
- [57] K. Konolige: *Small Vision Systems: Hardware and Implementation*. In Proc. International Symposium on Robotics Research, pages 111–116, Hayama, Japan, 1997.
- [58] S. Birchfield and C. Tomasi: *Depth Discontinuities by Pixel-to-Pixel Stereo*. International Journal of Computer Vision, 35(3): 1073–1080, December 1999.
- [59] A. Paskins: *The IEEE 1394 bus*. In Proc. IEE Half-Day Colloquium on New High Capacity Digital Media and Their Applications (Digest No: 1997/114), pages 4/1–4/6, 1997.
- [60] STMicroelectronics: *L78xx/L78xxC – Positive voltage regulators*, 19. edition, 2008. <http://www.st.com/stonline/products/literature/ds/2143/l7806c.pdf>
- [61] Phidgets Inc.: *Phidget21 C API Documentation*, 2.1.6. edition, September 2009. <http://www.phidgets.com/documentation/web/cdoc/index.html>, visited on 5th October 2009.
- [62] Phidgets Inc.: *1000 - PhidgetServo 1-Motor*, September 2009. <http://www.phidgets.com/documentation/Phidgets/1000.pdf>, visited on 5th October 2009.
- [63] amtec GmbH: *Powercube Manual (Deutsch)*, 1999. Version 2.01.
- [64] Phidgets Inc.: *Changelog*. <http://www.phidgets.com/changelog.php>, visited on 24th October 2009.