

Gegenstandserkennung und kategoriebasierter Transport anhand von kameraunterstützten NXT-Robotern

Studienarbeit

für die Prüfung zum
Bachelor of Engineering

von

Sebastian Hüther & Lorenzo Toso

5. Mai 2015

Bearbeitungszeitraum: 2 Semester
Matrikelnummer: 8853105 & 1906813
Kurs: TINF12B3
Studienfach: Informationstechnik
Ausbildungsfirma: Karlsruher Institut für Technologie
Betreuer: Gertrud Nieder

Inhaltsverzeichnis

Eidesstattliche Erklärung	IV
Abkürzungsverzeichnis	1
1 Einleitung	3
2 Problemstellung	4
3 Materialien und Methoden	5
4 Hardwareumsetzung	6
4.1 Entwurf des NXT-Roboters	6
4.2 Steuerung des Roboters	8
4.3 Wahl des Kameramoduls	9
5 Softwareumsetzung	10
5.1 Wahl der Bildverarbeitungsbibliothek	10
5.2 Raumerkennung	11
5.3 Algorithmen zur Objekterkennung	11
5.4 Mono-Kamerabasierte Entfernungsschätzung	12
5.5 Zielzonenerkennung	12
5.6 Hauptschleife	12
6 Tests des Robotersystems	13
6.1 Tests im gesicherten Rahmen	13

Inhaltsverzeichnis	III
6.2 Realtests	13
7 Zusammenfassung und Ausblick	14
Literaturverzeichnis	V

Eidesstattliche Erklärung

Gemäß § 5 (3) der „Studien- und Prüfungsordnung DHBW Technik“ vom 22. September 2011.
Ich habe die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen
Quellen und Hilfsmittel verwendet.

Ort, Datum

Unterschrift

Abkürzungsverzeichnis

Abbildungsverzeichnis

1 Einleitung

Im Rahmen der Studienarbeit des fünften und sechsten Semesters der Prüfung zum Bachelor of Engineering, stellt diese Arbeit eine Dokumentation zur Entwicklung eines kameragestützten Roboters dar. Ziel der Arbeit ist es mit Hilfe eines Android-Smartphones und eines LEGO Mindstorm NXT-Kits einen Roboter zu entwerfen, der Gegenstände in einem Raum erkennt, anfährt und in eine vordefinierte Zielzone transportiert. Hierfür werden diverse Methoden der Bildverarbeitung eingesetzt, welche unter der Verwendung der OpenCV-Library [1] implementiert werden.

Das nachfolgende Kapitel beschreibt die Problemstellung und erklärt eine Grundproblematik der Zusammenarbeit der beiden Hardwaremodule. Kapitel drei beinhaltet genaue Daten zu den Hardwaremodulen und deren Zusammenspiel, sowie die notwendigen Grundlagen der Bildverarbeitung, die für spätere Methoden genutzt werden. Die beiden darauf folgenden Kapitel gehen gesondert auf den genauen Aufbau des Roboters samt Konstruktionsplänen, sowie Details zur Softwareimplementierung der Objekterkennung und der Roboteransteuerung ein. Kapitel 6 beschreibt durchgeführte Tests sowohl unter speziell präparierten Bedingungen, als auch Realbedingungen. Das abschließende Kapitel ist ein letztes Fazit, welches einen Überblick über die gesamte Arbeit bildet.

2 Problemstellung

Ziel dieser Studienarbeit ist die praktische Anwendung gelernter Kenntnisse in Hard- und Software. Durch die Benutzung zweier verschiedener getrennter Module sind sowohl Kenntnisse in der Programmierung, als auch Kenntnisse in Prozessautomatisierung und in der Entwicklung verteilter Systeme erforderlich.

Konkrete Aufgabenstellung ist es einen kameragestützten Roboter zu entwickeln, der autonom Gegenstände in einem Raum erkennt und in definierte Zielzonen transportiert. Der Prozess kann hierbei in drei Teilprozesse unterteilt werden.

Erstens muss der Roboter mit Hilfe von bekannten Verfahren der Bildverarbeitung Objekte auf Grund ihrer physikalischen Beschaffenheit, beispielsweise ihrer Größe, ihrer Form und ihrer Farbe, erkennen und zielgerichtet anfahren.

Zweitens soll sich der Roboter gegenüber des Gegenstands optimal positionieren und diesen mit Hilfe eines mechanischen Greifarms aufnehmen.

Zuletzt muss der Roboter den aufgenommenen Gegenstand kategorisieren, nach der entsprechenden Zielzone suchen diese Anfahren und den Gegenstand ablegen. Die Erkennung der Zielzone kann dabei durch Markierungen an Wänden und Böden des Raumes erfolgen.

Als Bewertungskriterien dienen hierbei beispielsweise ob der Roboter alle Gegenstände erfolgreich erkennt, diese korrekt kategorisiert und in korrekte Zielzonen bewegt, sowie die Zeit in der dies geschieht.

3 Materialien und Methoden

NXT-Roboter

Android-Smartphone

Bluetooth-Verbindung

Some more Stuff

4 Hardwareumsetzung

4.1 Entwurf des NXT-Roboters

Die hardwareseitigen Voraussetzungen an den Roboter bestanden im Hauptsächlichsten aus der freien Bewegung im Raum und dem Aufnehmen, Mitführen und Ablegen von kleinen Gegenständen in einem vordefinierten Bereich.

Nach kurzer Recherche[2] und Durchsicht von Bauanleitungen für verschiedenste Anwendungsbereiche wurde sich für den Standardaufbau aus der zum Bauset zugehörigen LEGO NXT Bauanleitung entschieden.

Sie wurde lediglich um den Schall- und den Abstandssensor erleichtert; eine Halterung für das Smartphone wurde hinzugefügt.

1.hier Bild des Roboters einfügen

4.1.1 Sensoren

Tastsensor

Der berührungsempfindliche Sensor vorne dient zum Detektieren von Gegenständen im Bereich des Greifarms, woraufhin dieser geschlossen werden kann.

Rotationssensoren

Die Rotationssensoren in den Servomotoren erlauben es dem NXT-Roboter, die Geschwindigkeit der Motoren abhängig des Widerstands (des Untergrunds) zu regulieren. So werden unter anderem präzises Abbremsen und Fehlerminimierung bei der Positionsbestimmung ermöglicht.

2.Farbsensor?

4.1.2 Aktoren

Antriebsmotoren

Die beiden Servomotoren links und rechts des NXT-Roboters bilden den differentialen Antrieb und ermöglichen freie Fortbewegung.

Greifarmmotor

Der dritte Motor im vorderen Teil des Roboters dient zum Öffnen und Schließen des Greifarms und so zur Mitführung von Gegenständen.

4.2 Steuerung des Roboters

Die Steuerung des Roboters durch das Smartphone erfolgt via Bluetooth. Das Kommunikationsprotokoll und damit die nötigen Befehle zum Regeln der Aktoren und Auslesen der Sensoren wurde von LEGO dokumentiert und ist online erhältlich[3].

Auf dem NXT selbst wird hierbei kein Programm ausgeführt, um alle Logik zentral in der NXT-App auf dem Smartphone zu halten.

Zunächst muss eine Bluetooth-Verbindung erstellt werden, wozu beide Geräte aktiviertes Bluetooth aufweisen, der Roboter zusätzlich sichtbar für das Smartphone sein müssen.

Bei Erstverbindung muss der gesuchte NXT ausgewählt werden, danach ist die Bluetooth-Adresse bekannt und die App kann ohne Benutzerinteraktion eine Verbindung mit dem NXT-Roboter aufnehmen.

Kommt eine Verbindung zustande, können seriell Byte für Byte die Kommandos an den NXT übertragen, eventuelle Antworten empfangen werden.

App-seitig übernimmt ein gesonderter Thread in der Klasse NxtTalker nach Zustandekommen einer Verbindung das Management der Daten.

Zum Bewegen der Motoren muss zunächst per Befehl pro Aktor eine Geschwindigkeit (und Parameter wie Regulierung) übergeben, zum Stoppen können alle Motoren auf Geschwindigkeit '0' gesetzt werden.

Zwei Motoren können synchronisiert werden, sodass diese gleichzeitig starten. Ansonsten würde der Zeitversatz zwischen dem Absetzen der zwei 'setze Geschwindigkeit'-Befehle bewirken, dass der Roboter vor dem geradeaus fahren kurz nur das erste Rad ansteuert und in eine Richtung abdriftet.

4.3 Wahl des Kameramoduls

5 Softwareumsetzung

Mit Hilfe des in Kapitel 4.3 beschriebenen Kameramoduls müssen verschiedene Aufgaben aus dem Bereich der Bildverarbeitung bewältigt werden.

5.1 Wahl der Bildverarbeitungsbibliothek

Die Umsetzung der zu bewältigenden Aufgaben kann durch die Wahl einer geeigneten Bildverarbeitungsbibliothek deutlich vereinfacht werden. Wichtige Kriterien für die Wahl der Bibliothek sind unter anderem Funktionsumfang, Dokumentation und Aktivität der Community.

5.1.1 LibCCV

LibCCV ist eine open-source Bildverarbeitungsbibliothek, die viele bekannte Algorithmen implementiert. LibCCV steht unter einer BSD-Clause-3-Lizenz und kann somit für eine Studienarbeit problemlos unbegrenzt verwendet werden. Die Bibliothek ist größtenteils in C++ verfasst und somit potenziell auf einem Android-Smartphone verwendet werden. Die Verwendung auf dem Smartphone wird jedoch nicht offiziell unterstützt und kann potenziell weitere Schwierigkeiten mit sich bringen.

5.1.2 Imagemagick

Kleines Projekt, Outdated

5.1.3 OpenCV

Guter Android Port

Sehr große Library

Sehr bekannt

Gute Dokumentation

5.2 Raumerkennung

5.2.1 Kameragestützt

5.2.2 Ultraschallsensor

5.2.3 Kombination unterschiedlicher Sensordaten

5.3 Algorithmen zur Objekterkennung

5.3.1 Farbbasierte Objekterkennung

Konvertierung in HSV-Format

Filtern nach Saturation

Filtern nach Intensity

FindContours() - von OpenCV benutzt: Suzuki, S. and Abe, K., Topological Structural Analysis of Digitized Binary Images by Border Following. CVGIP 30 1, pp 32-46 (1985)

5.3.2 Kantenerkennung

5.3.3 Fokussierung eines Objekts

5.4 Mono-Kamerabasierte Entfernungsschätzung

5.5 Zielzonenerkennung

5.6 Hauptschleife

Arbeitszustände

1. Objekt suchen
2. Objekt ansteuern
3. Objekt aufnehmen
4. Objekt kategorisieren
5. Zielbereich suchen
6. Zielbereich ansteuern
7. Objekt ablegen

[INSERT ZUSTANDSÜBERGANGSDIAGRAMM]

6 Tests des Robotersystems

6.1 Tests im gesicherten Rahmen

6.2 Realtests

7 Zusammenfassung und Ausblick

Literaturverzeichnis

- [1] G. Bradski. Open cv. *Dr. Dobb's Journal of Software Tools*, 2000.
- [2] Freie nxt bauanleitungen.
- [3] LEGO GROUP et al. Lego mindstorms nxt communication protocol. *LEGO MindStorms NXT Bluetooth Developer Kit*, 2006.