

# Aufsammeln von Cola-Dosen mit einem mobilen autonomen Roboter

Oliver Bittel, Michael Fritz, Thorsten Rosin, Frank Schnürle

Fachbereich Informatik, Fachhochschule Konstanz  
{bittel,mfritz,rosin,franksch}@fh-konstanz.de

## 1 Einleitung

Mobile Roboter, die in einer dynamischen Umwelt selbständig agieren können, spielen in vielen Bereichen eine wichtige Rolle. Im Rahmen eines studentischen Projekts des Fachbereichs Informatik wurde im SS und WS 2000 ein Roboter realisiert, der in einer beliebigen Anordnung von Räumen und Gängen Cola-Dosen sucht, aufsammelt und an einem bestimmten Ablageplatz abliefern. Dabei mussten folgende Probleme gelöst werden, die bei mobilen Robotern typischerweise auftreten:

- Lokalisierung: wie weiss der Roboter, wo er sich befindet
- Navigation: wie kommt der Roboter kollisionsfrei von einem Ort A zu einem Ort B
- Objektsuche: wie lassen sich Cola-Dosen finden und aufnehmen

Das Projekt konnte dabei aufsetzen auf ein Bildverarbeitungsmodul zur farbbasierten Objekterkennung [Sch00] und auf ein Lokalisierungsmodul aus [Gut00], [Hoe00], das u.a. erfolgreich im Roboterfussball eingesetzt wird.

## 2 Ausstattung des mobilen Roboters

### 2.1 Hardware

Für das Projekt stand ein Pioneer-2 Roboter der Firma Activmedia zur Verfügung (siehe Abbildung 1). Der etwa im Durchmesser 50 cm große Roboter wird über zwei einzeln angetriebene Räder bewegt, so dass auch Drehungen auf der Stelle möglich sind. Mit einem Greifarm kann der Roboter Gegenstände aufnehmen und transportieren. Ein Laserscanner der Firma Sick, der die Umwelt mit einem Blickwinkel von 180 Grad bei einer Auflösung von 1 Grad abtastet, gestattet Abstandsmessungen zu Hindernissen mit einer Genauigkeit von etwa 1 cm. Die ebenfalls für die Hinderniserkennung verwendbaren Ultraschallsensoren wurden aufgrund der schlechteren Genauigkeit in diesem Projekt nicht eingesetzt. Eine schwenk-, neig- und zoombare Kamera dient zur Objekterkennung. Ausserdem ist der im Roboter installierte Linux-PC über ein Funk-Ethernet mit 11 Mbps (siehe Kiste mit Antenne in Abbildung 1) mit dem Hochschulnetz verbunden, so dass



Abbildung 1: Pioneer-2 Roboter

Visualisierung von Sensorik und

Motorik und Eingriffsmöglichkeiten in die Robotersteuerung von außen in einfacher Weise gewährleistet sind.

## 2.2 Software

Saphira [Sap98] ist eine Entwicklungs- und Steuerungsumgebung für Roboteranwendungen, die vom Hersteller des Roboters bereits mitgeliefert wird. Verschiedene Steuerungsaufgaben wie z.B. Lokalisierung, Navigation und Kamerasteuerung lassen sich in C/C++ realisieren und können dann als übersetzte Module in Saphira in einem Zeitscheibenverfahren gestartet werden. Darüber hinaus gestattet Saphira einen einfachen Zugriff auf die Standardsensorik (Ultraschall, Positionsbestimmung über Odometrie) und –Aktorik (Antriebsmotoren, Greifarm). Für eigene Sensoren (Kamera, Laser) und Aktoren (Schwenken, Neigen und Zoomen der Kamera) wurden eigene Module erstellt und in Saphira eingebunden.

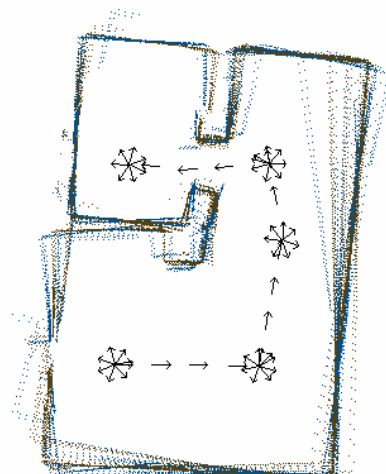
## 3 Lokalisierung

Wenn auf Leitlinien oder Landmarken in der Umwelt verzichtet werden soll, aber dennoch eine möglichst genaue Selbstlokalisierung des Roboters gewünscht wird, muss ein Laser-basiertes Lokalisierungsverfahren eingesetzt werden [GBFK98]. In unserem Projekt wurde ein vom amtierenden Weltmeister im Roboterfußball entwickeltes Lokalisierungs-Verfahren verwendet [Gut00]. Es arbeitet in 2 Phasen:

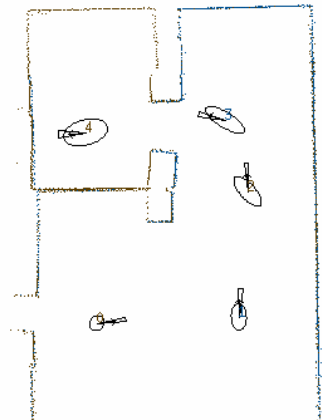
1. Explorationsphase: Der Roboter wird per Joystick durch die Umwelt geführt. Dabei werden Laser-Scans der Umwelt (180-Grad-Aufnahmen; siehe Abbildung 2) aufgenommen und die ungefähre Position der Aufnahme über Odometrie (Radumdrehungen) gemessen. Die Scans werden in eine gemeinsame Karte eingetragen. Da die Odometrie ungenau ist, enthält die Karte fehlerhaft positionierte Scans (siehe Abbildung 3). Durch paarweises Verdrehen und Verschieben der Scans unter Minimierung eines globalen Fehlers lässt sich die Karte nachträglich verbessern (siehe Abbildung 4).



**Abbildung 2:** 180-Grad Scan eines Raums



**Abbildung 3:** Während einer Explorationsfahrt aufgenommene Scans.



**Abbildung 4:** Korrigierte Umweltkarte

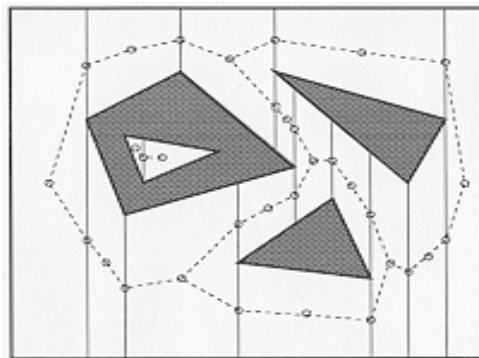
2. Lokalisierungsphase: Die in der Explorationsphase ermittelte Umweltkarte lässt sich nun für die Lokalisierung einsetzen. Dazu werden während der Fahrt laufend Scans aufgenommen und mit der Karte abgeglichen. Daraus wird die jeweils aktuelle Position berechnet.

## 4 Navigation

Um von einem Ort A zu einem Ort B kollisionsfrei zu gelangen, wird zunächst in einem Wegplaner eine Folge von Punkten berechnet, die dann unscharf abgefahren werden. Da der Wegplaner nur die statische Umwelt kennt, muss während der Fahrt noch dynamischen Hindernissen (z.B. plötzlich auftauchende Personen) ausgewichen werden.

### 4.1 Wegplaner

Der Wegplaner zerlegt zunächst den freien Raum der in der Explorationsphase ermittelten Umweltkarte in Trapeze (siehe auch [BKOS97]). Die Mittelpunkte nebeneinander liegender Trapezflächen werden verbunden und ergeben dann eine topologische Wegekarte (siehe Abbildung 5). In der Wegekarte können schließlich mit dem Algorithmus von Dijkstra kürzeste Wege zwischen einem Start- und einem Zielort bestimmt werden.



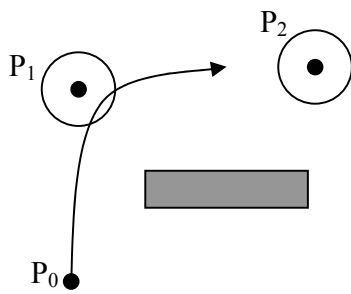
**Abbildung 5:** Zerlegung der Freifläche einer Umwelt in Trapeze. Die Mittelpunkte nebeneinander liegender Trapezflächen werden über den Mittelpunkt der senkrecht liegenden Trapezseiten verbunden. Die Punkte mit ihren Verbindungslinien bilden dann eine topologische Wegekarte.

### 4.2 Unscharfe Wegverfolgung

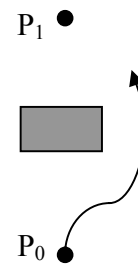
Würden die Punkte, die vom Wegplaner berechnet wurden, direkt angefahren werden, würde sich eine sehr eckige Fahrweise ergeben (man stelle sich das Abfahren eines Rechtecks vor). Daher wird mit einem Fuzzy-System (siehe auch [SRK93]) an jeder Position ein Kompromiss zwischen dem nächsten und übernächsten anzufahrenden Punkt berechnet, wobei der nächste Punkt lediglich in einer bestimmten Entfernung erreicht werden muss. Dadurch ergibt sich ein runderes Abfahren der Punktefolge, wobei die Punkte nur noch „gestreift“ werden. (siehe Abbildung 6).

### 4.3 Unscharfe Hindernisvermeidung.

Um einem plötzlich auftretenden Hindernis auszuweichen, wird durch ein weiteres Fuzzy-System ein Kompromiss zwischen Anfahren des nächsten Zielpunktes und Ausweichen des Hindernisses gebildet (siehe Abbildung 7).



**Abbildung 6:** Um von  $P_0$  über  $P_1$  nach  $P_2$  zu gelangen, muss  $P_1$  nur „gestreift“ werden“.



**Abbildung 7:** Es wird ein Kompromiss gebildet zwischen Zielpunkt  $P_1$  und Hindernisvermeidung.

## 5 Suchen und Sammeln der Coladosen

Mittels des Wegplaners werden der Reihe nach alle größeren Freiflächen der Umwelt angefahren und nach Cola-Dosen abgesucht. Hat der Roboter eine dieser Freiflächen erreicht, schwenkt er seine Kamera, dreht sich um die eigene Achse und schwenkt erneut seine Kamera. Dadurch erhält der Roboter einen 360-Grad-Blick seiner näheren Umgebung. Die Kameraaufnahmen werden nach Cola-Dosen farbasiert durchsucht. Konturen werden dabei nicht berücksichtigt. Die Position einer gefundenen Dose wird aufgrund der Dosen-Position innerhalb des Kamerabildes abgeschätzt. Der Roboter fährt dann auf die Coladose zu, wobei seine Richtung aufgrund neuer Bildinformation laufend korrigiert wird. Sobald die Cola-Dose mit dem Greifarm erreicht ist, wird die Dose aufgenommen und an einem Sammelplatz abgelegt. Die Suche nach weiteren Dosen wird fortgesetzt. Der korrekte Ablauf der einzelnen Teilaufgaben wird mit einem Zustandsautomaten gesteuert.

## 6 Literaturverzeichnis

- [BKOS97] M. de Berg, M van Kreveld, M. Overmars und O. Schwarzkopf, *Computational Geometry: Algorithms and Applications*; Springer, 1997.
- [FRS01] M. Fritz, T. Rosin und F. Schnürle, Aufsammeln von Cola-Dosen mit einem mobilen autonomen Roboter, Projektbericht, FH Konstanz, 2001.
- [GBFK98] J. Gutmann, W. Burgard, D. Fox und K. Konolige, *An Experimental Comparison of Localisation Methods*, International Conference on Intelligent Robots and Systems, Canada, 1998.
- [Gut00] J. Gutmann, W. Hatzack, I. Hermann, B. Nebel, F. Rittinger, A. Topor und T. Weigel, *The CS Freiburg Team: Playing Robotic Soccer Based on a Explicit World Model*, AI Magazine, 21(1), 2000.
- [Hoe00] C. Hölzl, *Laserbasierte Selbstlokalisierung und Wegeverfolgung eines autonomen mobilen Roboters*, Diplomarbeit, FH Konstanz, 2000.
- [SRK93] A. Saffiotti, E. Ruspini und K. Konolige, *Blending Reactivity and Goal-Directedness in a Fuzzy Controller*, IEEE Neural Nets and Fuzzy Control, San Fransisco, 1993.
- [Sap98] *Saphira Software Manual*, [www.activmedia.com](http://www.activmedia.com), 1998.
- [Sch00] S. Schablowski, *Entwicklung einer Bildverarbeitungsklassenbibliothek zur farbasierten Erkennung von Objekten*, Diplomarbeit, FH Konstanz, 2000.