

# Computer Vision für autonome mobile Roboter

Felix v. Hundelshausen

Freie Universität Berlin, Institut für Mathematik und Informatik

Takustr. 9, 14195 Berlin, Germany

Tel. ++49-(0)30-838-75-170, Fax. ++49-(0)30-838-75-109

e-mail: hundelsh@inf.fu-berlin.de

## 1 Einleitung



Abbildung 1: Das MidSize Spielfeld. (WM 2002 in Fukuoka, Japan)

„Computersehen“ ist das Thema der Dissertation, die im Rahmen der MidSize-Liga des RoboCup-Projekts geschrieben wird. Hier spielen vier gegen vier Roboter auf einem ca. 10x5m großen Feld Fußball ( Abbildung 1). Zur Wahrnehmung tragen die Roboter eine Kamera, die nach oben gerichtet in einen konvexen Spiegel schaut (Abbildung 2).

Dieser reflektiert das Licht aus allen Richtungen in die Kamera und vermittelt so eine Rundumsicht (Abbildung 3) der Umgebung.

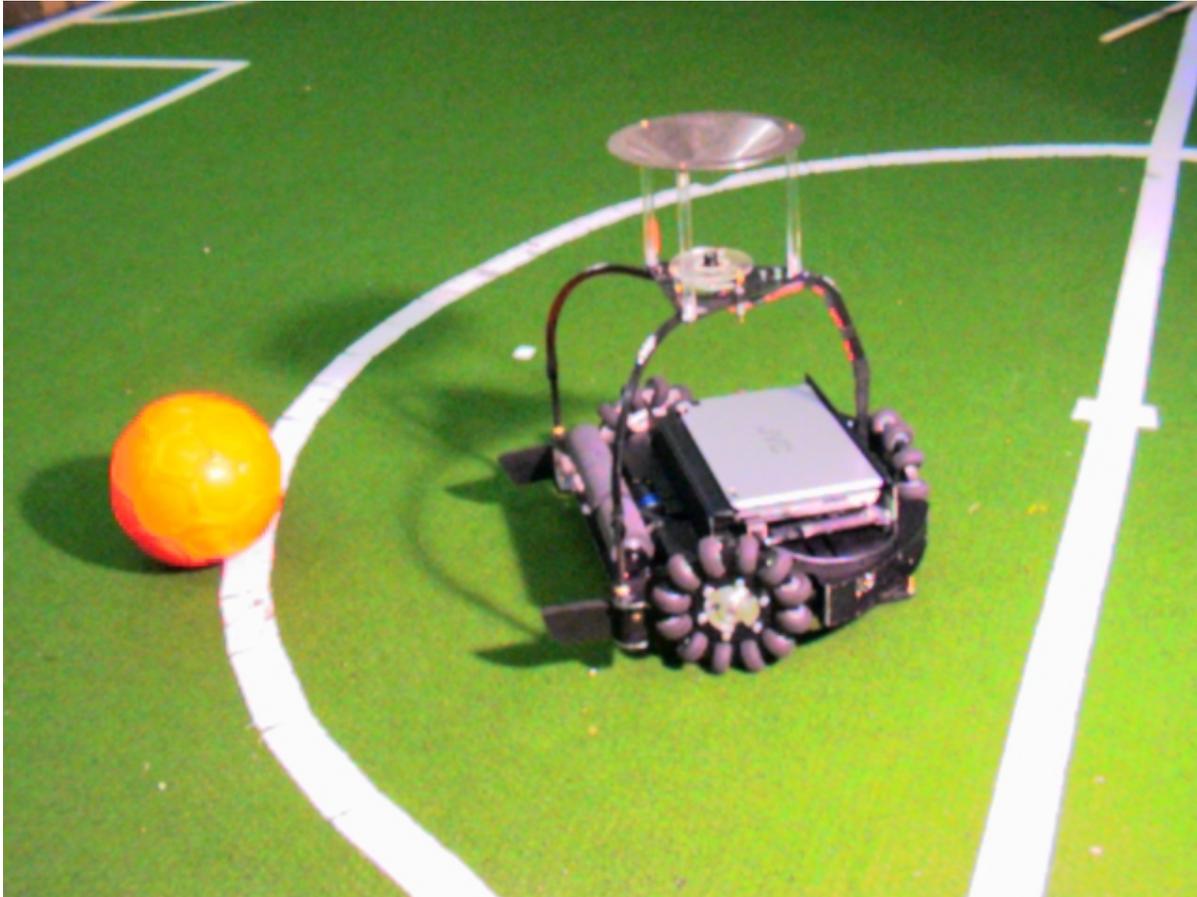


Abbildung 2: Unserer MidSize-Roboter

Gegenüber der realen Welt ist die Umgebung vereinfacht. Die Objekte sind durch Farben gekennzeichnet. Das Feld ist grün, die Tore sind gelb und blau angestrichen, die Roboter sind schwarz. Pfosten stehen an den Ecken des Spielfeldes. Sie dienen als Landmarken für die Lokalisierung der Roboter. Die Positionsbestimmung ist eine der wichtigsten Aufgaben und die meisten Systeme suchen nach den Toren und Pfosten und bestimmen damit die Roboterposition. Hierbei kommen vor allem Verfahren, die auf Triangulierung beruhen und Monte Carlo Verfahren [8] zum Einsatz. Letztere haben in den letzten Jahren eine große Popularität erreicht. Sie sind extrem robust, weil sie anstatt einer einzigen möglichen Position eine ganze Wahrscheinlichkeitsverteilung von Positionen betrachten und in der Lage sind, diese über die Zeit zu propagieren.

Wesentliches Ziel des Dissertationsvorhabens war es, die Lokalisierung über die Erkennung der Markierungslinien zu erreichen. Diese Aufgabe zerfällt in zwei Teile: Die Linien effizient aus den Bildern zu extrahieren, und die eigentliche Lokalisierung mit Hilfe der

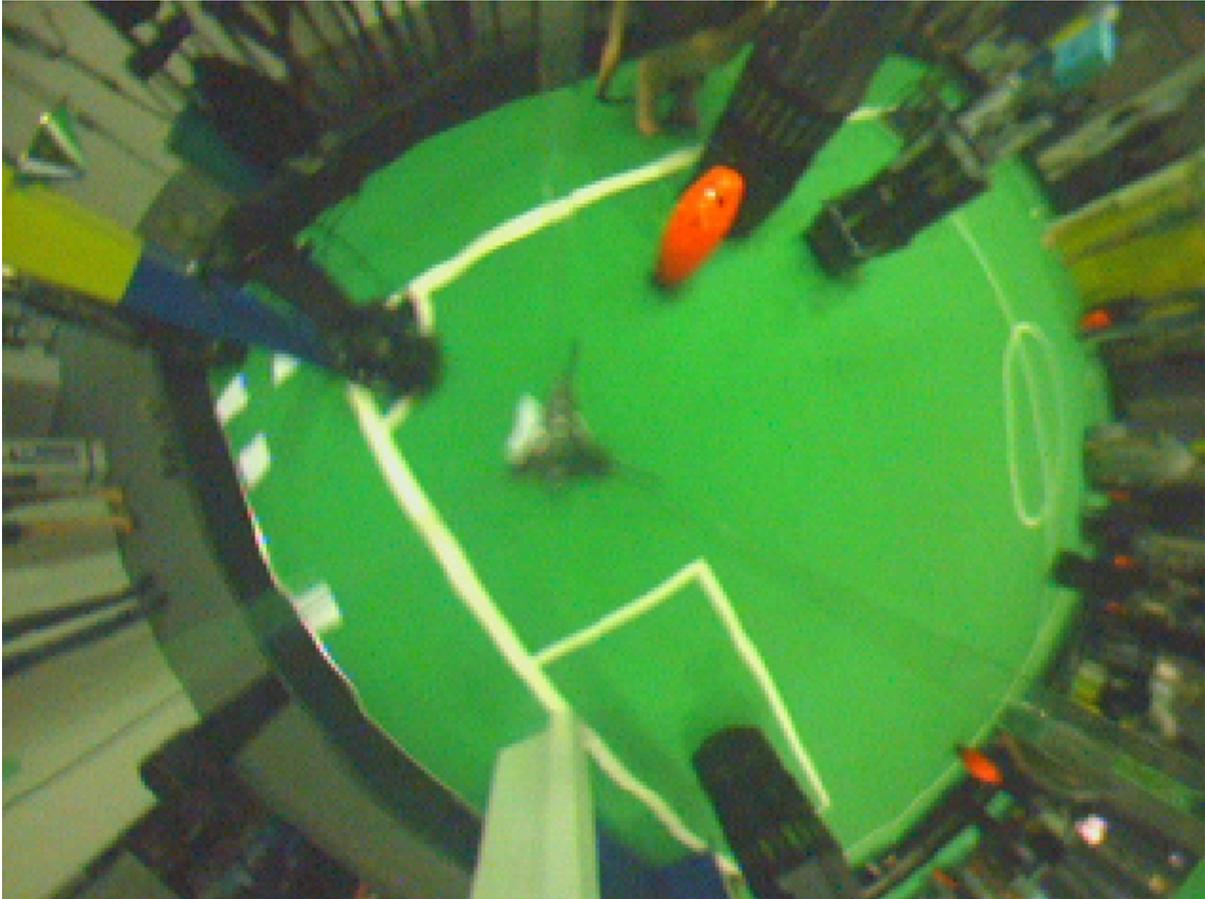


Abbildung 3: Rundumsicht des Roboters.

Linien zu bewerkstelligen. Die Lokalisierung wiederum kann durch Matchen der Daten mit einem Modell der Linien oder durch Erkennung eindeutiger Strukturen wie z.B den Mittelkreis erfolgen. Sowohl für die Extraktion der Linien als auch für die Erkennung von Substrukturen wird jeweils ein neues Verfahren vorgestellt.

## 2 Bestehende Verfahren zur Extraktion der Linien

Es gibt vier grundsätzliche Verfahren, die Linien zu extrahieren, die in Abbildung 4 dargestellt sind. Die einfachste Methode besteht darin einen Schwellwert für die Intensität zu verwenden und so die hellen Linienpixel zu extrahieren. Um die Datenmenge zu reduzieren kann dann noch „Thinning“ [2] angewendet werden. Problematisch hierbei ist, dass Punkte detektiert werden können, die nicht von Linien stammen (z.B. weisse Wände wie beim Tor). Aufgrund unterschiedlicher Beleuchtungsverhältnisse ist es auch oft nicht möglich einen einheitliche Schwellwert zu finden.

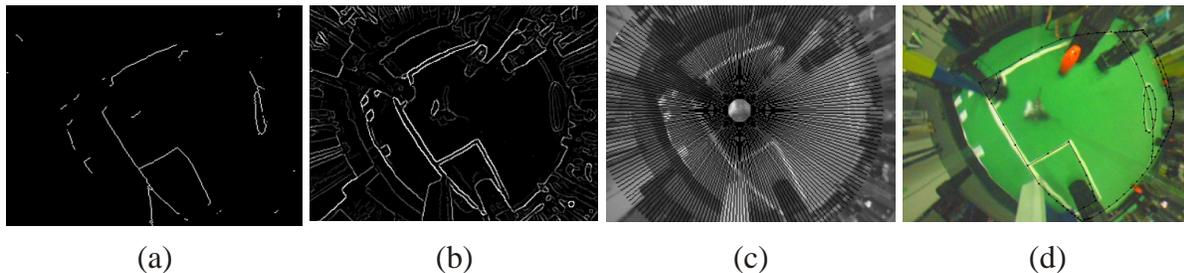


Abbildung 4: (a)Schwellwert und „Thinning“ (b) Filter-Operatoren (c) Radial-Methode (d)Verwendung eines Modells

Anstatt einen einfachen Schwellwert zu verwenden könnten man über die Anwendung von Filtern nachdenken, wie z.B. den Canny Edge-Detektor [1], den Roberts Cross Edge Detektor[6], den Sobel Kanten Detector [5], den Kompass-Operator [7], Gabor-Filter[4] oder auf Wavelets basierende Lösungen [3]. Die Anwendung eines Filters auf das ganze Bild ist jedoch extrem rechenintensiv.

Eine weitere Möglichkeit ist die Anwendung der Radial-Methode, bei der strahlenförmig vom Mittelpunkt des Bildes aus nach Linien gesucht wird. Parallel zu den Strahlen verlaufende Linien werden jedoch schlecht erkannt.

Letztendlich kann man noch ein Modell verwenden, das den Verlauf der Linien im Bild vorhersagt. Hierzu muß die Position des Roboters jedoch bereits bekannt sein.

### 3 Ein neues Verfahren zur Extraktion der Linien

Es wurde ein neues Verfahren zur Extraktion der Linien entwickelt, das von den Regionen zwischen den Linien ausgeht. Hierzu wird das bekannte „Region Growing“-Prinzip erweitert, um die Regionen effizient über eine Bildfolge hinweg verfolgen zu können. Aufeinanderfolgende Bilder sind meistens ähnlich und die Idee ist bereits gewonnene Information zu verwenden um die Verarbeitung späterer Bilder zu beschleunigen. Wenn Bilder größere gleichfarbige Flächen beinhalten, so kommt es oft vor, dass sich die Flächen aufeinanderfolgender Bilder überdecken. Diese Überlappung wird ausgenutzt indem Regionen zunächst auf die Schnittmenge schrumpfen und sich dann wieder ausdehnen. Das Verfahren ist im Detail in [9] beschrieben. Das Prinzip wird in Abbildung 5 dargestellt. Das Verfahren repräsentiert zu jedem Zeitpunkt den Rand der Regionen als eine Folge von Punkten. Nach einer Glättung kann nun orthogonal zu dem Verlauf des Randes nach Linien gesucht werden (siehe Abbildung 6).

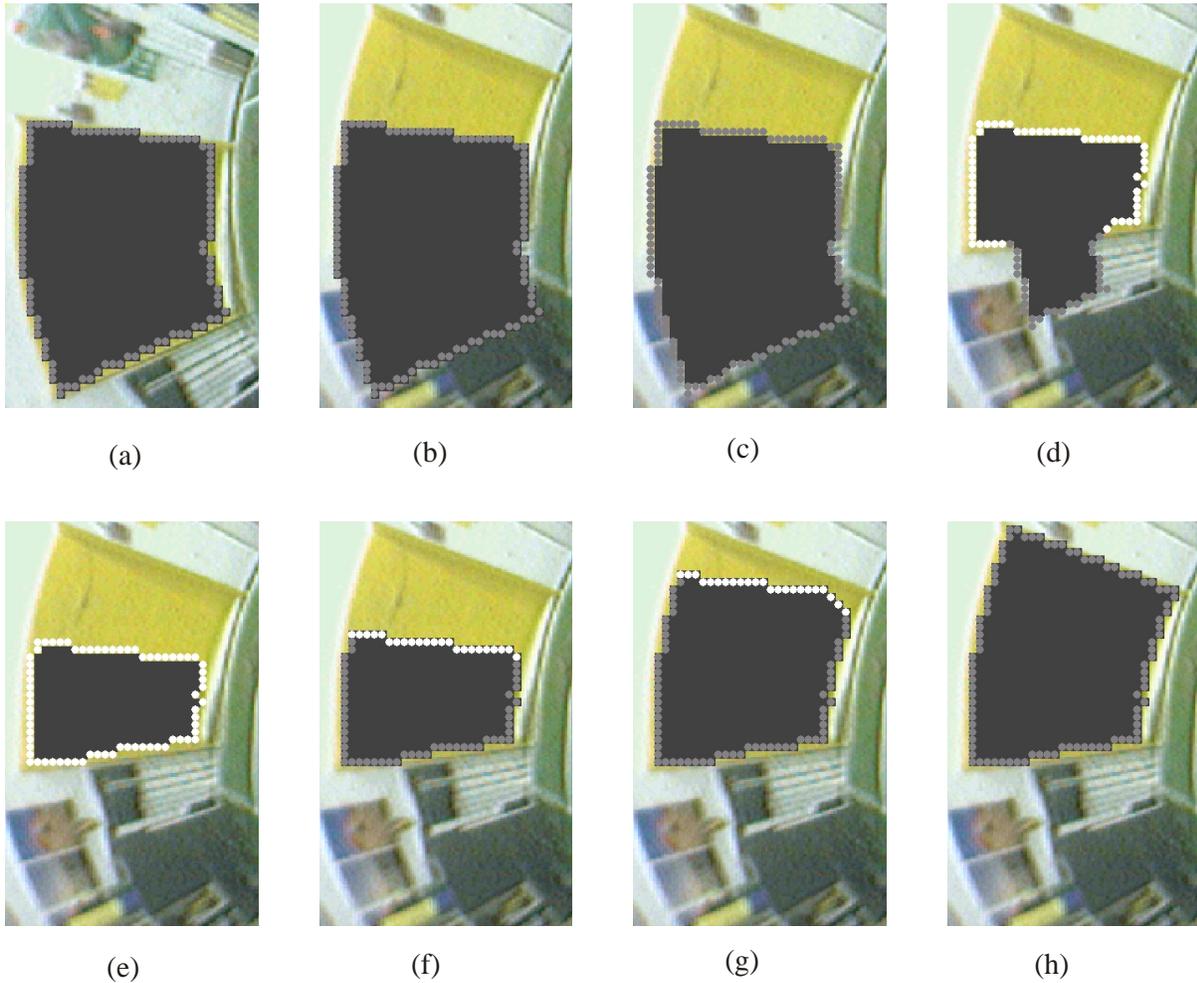
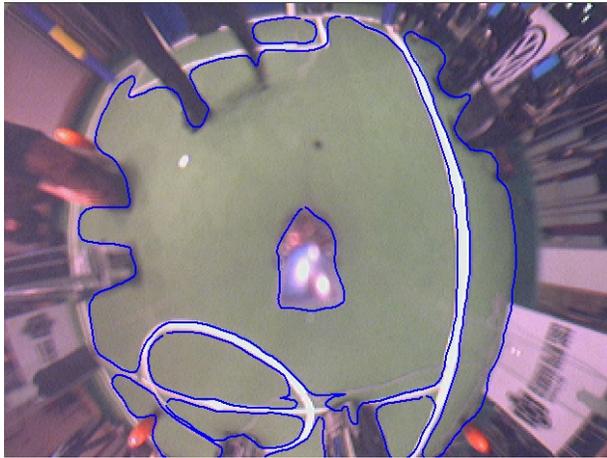


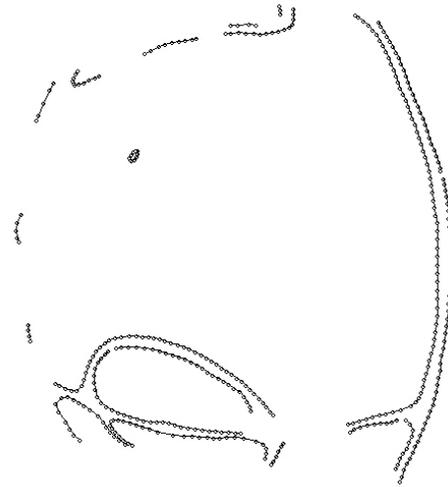
Abbildung 5: Erweiterung des Region-Growing Prinzips

## 4 Lokalisierung mit den Markierungslinien

Unabhängig von der Art der Lokalisierung muß zunächst die Verzerrung aus den Linienkonturen herausgerechnet werden. Hier spielt die Kalibrierung der sogenannten „Abstandsfunktion“ eine entscheidende Rolle und ein Ziel ist es, diese Funktion automatisch zu kalibrieren. Die Abstandsfunktion beschreibt den Zusammenhang zwischen Entfernungen von Objekten zum Roboter und dem entsprechendem Pixelabstand im Bild. Wir gehen im folgenden davon aus, dass die Abstandsfunktion kalibriert ist. Dann kann jeder Punkt der Linien transformiert werden und die Verzerrung ist herausgerechnet. Die Linien sind dann in lokalen Weltkoordinaten repräsentiert, d. h. Entfernungen sind in Zentimetern repräsentiert und ein Vergleich mit einem geometrischen Modell der Linien ist möglich. Bevor ein solcher Vergleich stattfinden kann, muß jedoch zunächst eine Hypothese für die Roboterposition vorliegen. Transformiert man die Linien aus dem loka-



(a)



(b)

Abbildung 6: (a) Anwendung eines Detektors orthogonal zu dem Linienverlauf (b) Die gefundenen Konturen werden als Menge von Linienzügen representiert, von denen jeder aus einer Folge von Punkten besteht.

len Koordinatensystem dieser Hypothese heraus, dann sollten die Linien mit dem Modell übereinstimmen, falls die Hypothese richtig ist. Um den Grad der Übereinstimmung zu messen ist es wichtig, eine Abstandsfunktion zwischen wahrgenommenen transformierten und Modelllinien zu definieren. Es wurde folgender Ansatz verwendet: Zu jedem Punkt der wahrgenommenen Linien wird der euklidisch nächste Punkt des Modells ermittelt und der Abstand berechnet. Überschreitet dieser einen Schwellwert von 40cm, dann wird der Punkt als Ausreißer betrachtet und nur die 40 cm als Abstand veranschlagt. Der Gesamtfehler ist dann der mittlere Abstand. Der Vorteil dieses einfachen Abstandsmaßes ist, dass eine Lookup-Tabelle erzeugt werden kann und die Berechnung sehr effizient durchgeführt werden kann. Die Lookup-Tabelle entspricht einer zweidimensionalen Diskretisierung des Spielfeldes. Jede Zelle speichert den Abstand zur nächsten Linie. So ist es dann mit einem Pentium III, 800 Mhz möglich, über 1500 Positionshypothesen bei wahrgenommenen Linienskizzen von 400 Punkten in einer halben Sekunde auszuwerten. Auf diese Weise kann man eine globale Suche durchführen, indem man die Hypothesen gleichmäßig über das Feld verteilt und die beste herausgreift. Ein fundamentales Problem ist jedoch, daß die Methode nicht gut funktioniert, wenn die Entfernungskalibrierung nicht genau eingestellt wurde, oder wenn zu viele Ausreißer vorhanden sind.

Deshalb wurde noch ein zweiter Ansatz verfolgt, der versucht in den Linienkonturen Merkmale zu erkennen. Hierbei werden die Linien zunächst geglättet und dann für jeden Punkt ein Krümmungsmaß berechnet. Lokale Maxima der Krümmungen ergeben Hypothesen für Eck-Merkmale. Hier können je nach positiver oder negativer Krümmung konvexe und konkave Ecken unterschieden werden. Die konkaven Ecken treten nur an den Ecken des

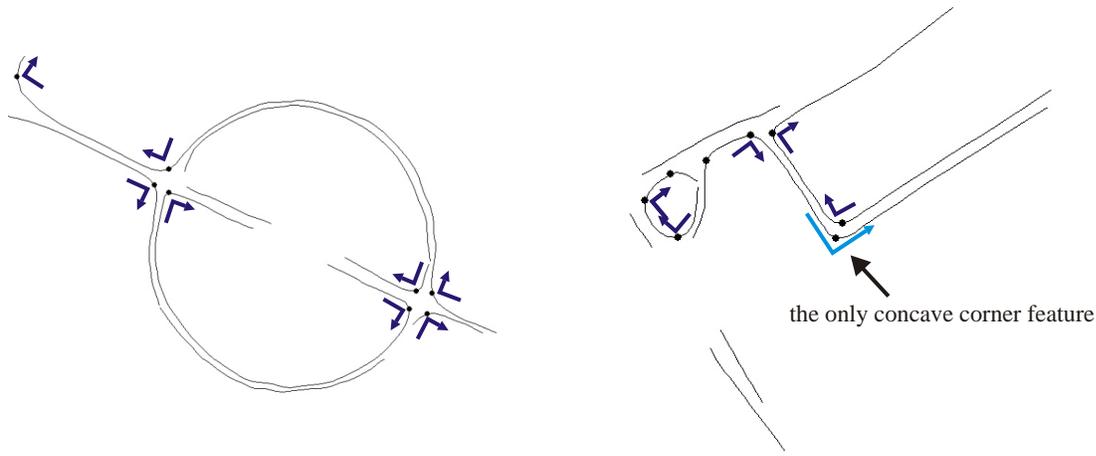


Abbildung 7: Konvexe Ecken sind durch kleine dunkle Ecken dargestellt. Konkave durch größer Ecken.

Strafraums auf, und ihre Erkennung liefert einen sehr starken Hinweis auf die Position des Roboters.

Nach der Eckerkennung werden die Linien an den Krümmungsmaxima aufgetrennt und für jedes Segment wird mittels einer einfachen Heuristik ermittelt, ob es gerade oder gekrümmt ist. Gekrümmte Stücke werden dann gruppiert und der Mittelkreis kann, falls sichtbar, erkannt werden. Aus den geraden Stücken können andere Merkmale wie z.B. Rechtecke ermittelt werden. Da nur der Strafraum durch ein Rechteck begrenzt wird, liefert auch dieses Merkmal einen guten Hinweis auf die Roboterposition. Auch aus den Ecken können weitere Merkmale gebildet werden. So bilden zwei Ecken an einigen Stellen ein T-Stück.

## Literatur

- [1] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 8:679–698, 1986. 4
- [2] E. R. Davies. *Machine Vision: theory, algorithms, practicalities*. Academic Press, 1990. 3
- [3] S. Mallat and S. Zhong. Characterization of signals from multiscale edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 14(7):710–732, 1992. 4
- [4] R. Mehrotra, K. R. Namudura, and N. Ranganathan. Gabor filter-based edge detection. *Pattern Recognition*, 25:1479–1494, 1992. 4

- [5] K. K. Pingle. Visual perception by computer. In A. Grasselli, editor, *Automatic Interpretation and Classification of Images*, pages 277–284. Academic Press, New York, 1969. 4
- [6] L. G. Roberts. Machine perception of three-dimensional solids. In J. T. Tippet et al., editor, *Optical and Electro-Optical Information Processing*, pages 159–197. MIT Press, Cambridge, 1965. 4
- [7] Mark A. Ruzon and Carlo Tomasi. Color edge detection with the compass operator. *IEEE Conference on Computer Vision and Pattern Recognition*, 2:160–166, 1999. 4
- [8] W. Burgard S. Thrun, D. Fox and F. Dellaert. Robust monte carlo localization for mobile robots. *Artificial Intelligence*, 2001. 2
- [9] F. v. Hundelshausen and R. Rojas. Tracking regions and edges by shrinking and growing. In *Proceedings of the RoboCup 2003 International Symposium, Padova, Italy*, 2003. 4