Technische Hochschule
Brandenburg
University of
Applied Sciences
Fachbereich
Informatik und Medien

Entwicklung des Suchroboters „Findus“

Franziska Gohlicke, Frederic Falck und Jannik Zschiesche
Projekt: Künstliche Intelligenz • Studiengang Informatik • Fachbereich Informatik und Medien • 28.01.2025

Aufgabenstellung

Das Projekt hatte das Ziel einen selbstfahrenden LEGO-Suchroboter zu entwickeln, welcher in einem vorgegebenen Gitternetz navigiert. Der Roboter ist darauf ausgelegt, effizient und autonom nach bestimmten Gegenständen (blauen Bällen) am Rand zu suchen. Dabei sind verschiedene Kriterien zu beachten:

- Das Gitternetz kann mehrere Kreuzungen enthalten, die gesperrt sind, wodurch alternative Wege gefunden werden müssen.
- Der Roboter entwickelt dynamisch die kürzeste Route zu den definierten Randpunkten des Gitternetzes und passt seine Navigation an Hindernisse an.
- An den Randpunkten des Suchfeldes kann der Roboter überprüfen, ob dort ein gesuchter Gegenstand liegt.

Um die Suchergebnisse interaktiv darzustellen, wird zusätzlich ein NAO-Roboter(Abb. 5) in das System integriert. Falls der Suchroboter einen Gegenstand findet, kann er die Information an den NAO-Roboter weitergeben, welcher die Entdeckung anschließend per Sprachausgabe kommuniziert.

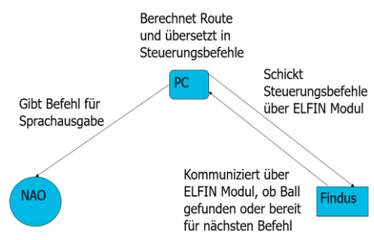

Abb. 1: Gesamtarchitektur

Vision

Das übergeordnete Ziel dieses Projekts ist die Entwicklung eines Roboters, der insbesondere älteren oder sehbehinderten Menschen im Alltag helfen kann. Der Roboter könnte beispielsweise die Wohnung nach Gegenständen wie einer Brille oder Fernbedienung absuchen. Gesperrte Kreuzungen stellen Gegenstände wie eine Kommode oder einen Stuhl dar.
Durch die Sprachfunktion des NAO-Roboters kann dem Nutzer mitgeteilt werden, wo sich der gesuchte Gegenstand befindet. Dies würde älteren und sehbehinderten Menschen mehr Komfort bieten und ihren Alltag erheblich erleichtern.

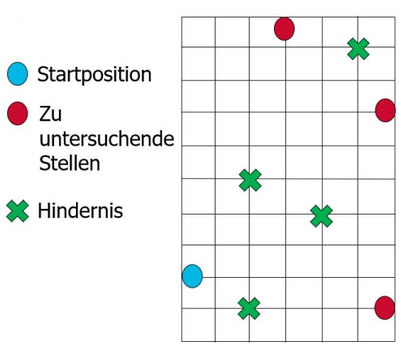

Abb. 2: mögliche Karte der Wohnung

N (Türseite)
[70, 71, 72, 73, 74, 75, 76]
[63, 64, 65, 66, 67, 68, 69]
[56, 57, 58, 59, 60, 61, 62]
[49, 50, 51, 52, 53, 54, 55]
[42, 43, 44, 45, 46, 47, 48]
W [35, 36, 37, 38, 39, 40, 41] O
[28, 29, 30, 31, 32, 33, 34]
[21, 22, 23, 24, 25, 26, 27]
[14, 15, 16, 17, 18, 19, 20]
[7, 8, 9, 10, 11, 12, 13]
[0, 1, 2, 3, 4, 5, 6]
S (Fensterseite)

Abb. 3: Repräsentation der Karte

Routenplanung

Die Routenplanung wird mit Breitensuche auf einem PC in Python durchgeführt. Dabei wird eine Karte des Gitternetzes erstellt (Abb. 3), welche einen optimalen Weg berechnet und in Steuerungsbefehle für den Roboter übersetzt. Der Algorithmus ermöglicht eine effiziente und dynamische Navigation, wobei verschiedene Parameter manuell eingegeben werden müssen:

- **Startpunkt:** Der Ausgangspunkt des Roboters innerhalb des Gitternetzes. (Abb. 2: Startposition)
- **Abzusuchende Randpunkte:** Eine Liste von Zielpunkten an den Rändern des Suchbereichs, welche überprüft werden sollen. (Abb. 2: Zu untersuchende Stellen)
- **Gesperrte Kreuzungen:** Die Anzahl und Position der blockierten Kreuzungen, die nicht durchquert werden können. (Abb. 2: Hindernis)

Komponenten (Hardware)

Das Chassis des Suchroboters besteht aus LEGO-Teilen und LEGO-Komponenten. Außerdem besitzt es zwei Räder (Abb. 4: B), zwei Servomotoren, einen Akku, ein AKSEN-Board (Abb. 4: C), ein ELFIN Modul (Abb. 4: A), zwei Tast-Sensoren (Abb. 4: D), einen Infrarot-Sender und –Empfänger (Abb. 4: F) sowie vier Optokoppler (Abb. 4: E).
Beide Räder besitzen einen separat angesteuerten Motor, wodurch beispielsweise eine Drehung auf der Stelle ermöglicht wird. Die Taster werden verwendet, um festzustellen, ob ein Prüfobjekt erreicht wurde. Anschließend testen der Infrarotsender und Infrarotempfänger, ob sich ein Objekt auf dem Prüfobjekt befindet. Der Suchroboter verfügt an der Unterseite, sowohl vorne als auch hinten, über jeweils zwei nach unten gerichtete Optokoppler. Diese können schwarze und weiße Oberflächen erkennen und ermöglichen dadurch Vorwärts-, Rückwärts- und Kurvenfahrten entlang der schwarzen Linien. Auf das AKSEN-Board wird das Steuerungsprogramm (in C geschrieben) für den Suchroboter geflasht.

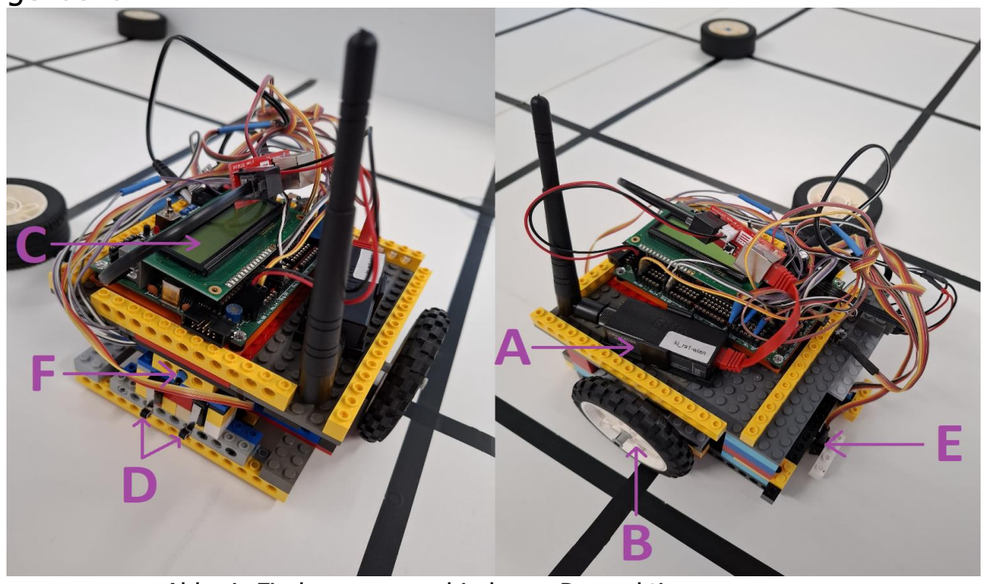

Abb. 4: Findus aus verschiedenen Perspektiven

Programmablauf

Zu Beginn stellt das Programm eine Verbindung zu NAO und Findus über das ELFIN-Modul her. Anschließend wird die Route nach der Benutzereingabe berechnet (siehe „Routenplanung“). In der Folge übersetzt das Programm den optimalen Pfad in Steuerungsbefehle für Findus. Danach werden die Befehle schrittweise an Findus gesendet. Nach der Ausführung eines Befehls sendet Findus ein Signal zurück, woraufhin das Programm den nächsten Befehl übermittelt (Ping-Pong Prinzip).
Falls der Ball gefunden wird, sendet Findus ein entsprechendes Signal, und das Programm weist den NAO an, eine Sprachausgabe zu machen, um den Fund zu vermelden. Sobald entweder der Ball gefunden oder die letzte zu überprüfende Position erreicht wurde, berechnet das Programm den Rückweg zur Startposition und schickt Findus (wieder mit Ping-Pong Prinzip) auf die Heimreise.

Abb. 5: NAO-Roboter

Betreuung: Prof. Dr. Emanuel Kitzelmann • Technische Hochschule Brandenburg • Dipl.-Inform. Ingo Boersch • Technische Hochschule Brandenburg