



Hochschule für Angewandte Wissenschaften Hamburg  
*Hamburg University of Applied Sciences*

## **Ausarbeitung: Seminar**

Torben Becker

Umgebungsidentifikation und Kartierung durch RatSLAM  
für autonome Fahrzeuge

## Inhaltsverzeichnis

<b>1</b>	<b>Einführung</b>	<b>2</b>
1.1	Einleitung . . . . .	2
1.2	Zielsetzung . . . . .	2
1.3	Gliederung . . . . .	3
<b>2</b>	<b>Vorbereitung</b>	<b>4</b>
2.1	Projekt 1: Visuelles Odometriesystem . . . . .	4
2.1.1	Aufbau und Funktion . . . . .	4
2.1.2	Probleme und Risiken . . . . .	6
2.1.3	Fazit . . . . .	7
2.2	Projekt 2: RatSLAM . . . . .	7
2.2.1	Risiken . . . . .	8
<b>3</b>	<b>Masterthesis</b>	<b>10</b>
3.1	Konzept . . . . .	10
3.1.1	CampusBot . . . . .	10
3.1.2	Carolo-Cup . . . . .	12
3.2	Chancen und Risiken . . . . .	14
<b>4</b>	<b>Weiteres Vorgehen</b>	<b>15</b>

**Zusammenfassung**

In dieser Ausarbeitung wird das grundlegende Thema für die Masterthesis erörtert. Zuvor wird auf die vorbereitenden Arbeiten für die Masterthesis eingegangen. Hierbei wird dargelegt, welche Chancen, aber auch Risiken, sich mit diesen Arbeiten ergeben und welchen Zweck diese erfüllen. Am Schluss wird auf die Masterthesis eingegangen, die sich mit einer selbstständigen Umgebungsidentifikation und Kartierung durch RatSLAM beschäftigt.

# 1 Einführung

## 1.1 Einleitung

Da das Themengebiet SLAM (Siciliano und Khatib (2008)) ein sehr weitläufiges Gebiet ist mit vielen unterschiedlichen Techniken und Möglichkeiten der Datenverarbeitung, besteht die Hauptaufgabe in Anwendung 1 (Becker (2012a)) darin, ein Verständnis für dieses Themengebiet zu entwickeln. Im späteren Verlauf ist der Algorithmus RatSLAM ausgewählt worden, um mit diesem Tests durchzuführen und zu überprüfen, wie gut dieser Algorithmus auf den Plattformen des Forschungsprojekts FAUST läuft. Es ist ein Verständnis für die Funktion und das Datenmanagement entwickelt worden. Außerdem sind theoretische Überlegungen durchgeführt worden, welche Chancen und Risiken RatSLAM für das Forschungsprojekt bietet.

Zum besseren Verständnis von RatSLAM und zur Überprüfung der Einsatzfähigkeit ist in Anwendung 2 (Becker (2012b)) eine Recherche von praktischen Einsätzen und Implementierungen vorgenommen worden. Bei der Bewertung der Tests ist ein besonderes Augenmerk auf bestimmte Aspekte gelegt worden. Es interessierten vor allem die Echtzeitfähigkeit und der Ressourcenbedarf. Die Erkenntnis aus den Arbeiten ist, dass eine Echtzeitfähigkeit möglich, aber nicht erprobt worden ist. Dafür ist allerdings der Ressourcenbedarf gemessen an der kartografierten Umgebungsgröße sehr sparsam und auch die Berechnungszeit für sehr große Umgebungen fällt sehr klein aus. Die meisten Arbeiten bieten aber auch einen kleinen negativen Aspekt, da bei den meisten Arbeiten die Hardware-Spezifikationen der Testsysteme nicht dargelegt worden sind.

## 1.2 Zielsetzung

RatSLAM ist ein Algorithmus, der eine Karte seiner gefahrenen Strecke aufzeichnet, in dem nicht wie in anderen SLAM Verfahren die gesamte Umgebung vermessen wird, sondern

nur die zurückgelegte Distanz und die Rotation des Roboters im Raum. RatSLAM schafft es dabei, sich innerhalb seiner aufgezeichneten Karte zu lokalisieren. Dies bedeutet, dass RatSLAM sofort seine Position innerhalb der Karte erkennt sowie Schleifen erkennen und schließen kann. Ein Streckenabschnitt wird Schleife genannt, wenn ein neu erkundeter Streckenabschnitt in einen bereits bekannten Streckenabschnitt mündet. Durch diese Lokalisierung wird es RatSLAM ermöglicht sogar zum Teil einen im Uhrzeigersinn aufgezeichneten Kurs auch gegen den Uhrzeigersinn zu erkennen und richtig zu interpretieren.

Es soll in der Masterthesis ein System entwickelt werden, das vollkommen selbstständig seine Umgebung erkundet und von dieser Umgebung eine Karte erstellt. Dabei soll einer der Forschungsschwerpunkte die Behandlung von dynamischen Objekten darstellen. Es soll untersucht werden, wie RatSLAM auf Veränderungen der Umgebung reagiert. Bei dieser Untersuchung soll ebenfalls überprüft werden, ob eine Verbesserung zur Behandlung von dynamischen Objekten möglich ist und in welchem Rahmen diese gestaltet werden können. Eine weitere Forschungsaufgabe befasst sich mit der Beschleunigung von RatSLAM für kleine, nicht leistungsstarke System-On-Chip Plattformen wie dem Pico-PC auf den Carolo-Cup Plattformen. Hierbei geht es um die Frage, welche Maßnahmen ergriffen werden müssen, um RatSLAM grundlegend zu beschleunigen und welche fortführenden Maßnahmen existieren, um eine weitere Beschleunigung zu ermöglichen. Das übergeordnete Ziel ist dann, im simuliertem Straßenverkehr schwierige Stellen, wie z.B. fehlende Fahrbahnmarkierungen, zu überbrücken und den Spurführungsalgorithmus in einem kontrollierten Zustand zu halten. Dabei bedeutet kontrollierter Zustand, dass der Spurführungsalgorithmus zu jedem Zeitpunkt eine vorhersagbare Reaktion besitzt oder durch RatSLAM eine vorhersagbare Reaktion vorgegeben bekommt.

### 1.3 Gliederung

In dem nächsten Kapitel wird auf die vorbereitenden Arbeiten, Projekt 1 und Projekt 2, für die Masterthesis eingegangen. Dabei wird zum einen die theoretische Funktion dargestellt, zum anderen aber auch Risiken oder Hindernisse bzw. Aufgaben, die bei der Entwicklung auftreten können.

Das dritte Kapitel geht auf die Masterthesis ein und behandelt mögliche Konzepte für die Masterarbeit. Dabei werden zwei verschiedene Forschungsplattformen behandelt und welche Chancen, aber auch Risiken, diese Masterarbeit bei Abschluss bieten kann.

Abschließend wird das weitere Vorgehen, sprich der Zeitplan, dargelegt und welche Aufgaben möglicherweise sich für weiterführende Arbeiten ergeben.

## 2 Vorbereitung

Um mit der Masterthesis beginnen zu können, müssen zuerst bestimmte Bedingungen erfüllt werden. Zunächst muss ein Messsystem für die Translation bzw. Bewegung und die Rotation eines Roboters entwickelt werden, speziell für den CampusBot. Dieser Roboter verfügt nicht über ein solches System. Danach muss der Algorithmus für RatSLAM implementiert und evaluiert werden, um als letzten Schritt mit der Masterthesis zu beginnen.

### 2.1 Projekt 1: Visuelles Odometriesystem

Bei bestimmten Antriebsarten von Robotern ist es schwer bis unmöglich eine Erfassung der Odometrie mittels Sensorik durchzuführen. Dabei ist die gängige Sensorik ein Inkremental- oder Hallsensor an der Achse oder an den Rädern um die gefahrene Distanz zu bestimmen. Mittels Lenkwinkelsensoren oder einer Einmessung und Abfrage einer Look-Up-Table kann der Lenkwinkel bestimmt und somit auch eine Rotation des Roboters erfasst werden.

Die Forschungsplattform CampusBot verfügt über einen sogenannten Panzerantrieb. Dies bedeutet, dass pro Seite drei Räder montiert sind, die ebenfalls nur pro Seite angesprochen werden. Es drehen sich entweder alle Räder der linken oder rechten Seite. Dieses Prinzip des Antriebes erlaubt es dem CampusBot, dass sich dieser besser in einem schwergängigen Gelände bewegen kann, macht aber auch die Odometrieerfassung bedeutend schwerer, da nicht eindeutig zwischen Fortbewegung und Drehung unterschieden werden kann.

Aus diesem Grund ist ein alternatives System zur Erfassung der Odometrie gesucht und implementiert worden. Dieses System bzw. Verfahren (Campbell u. a. (2005)) basiert auf einer Kamera und der Berechnung des optischen Flusses. Für die Bildverarbeitung wird durchgehend auf die freie Bibliothek OpenCV (OpenCV) gesetzt. Durch den Einsatz von OpenCV reduziert sich zum einen der Aufwand für die Entwicklung, zum anderen sind die in OpenCV implementierten Algorithmen so weit wie möglich optimiert.

#### 2.1.1 Aufbau und Funktion

In Abbildung 1 ist eine schematische Skizzierung der Funktionsweise dargestellt. Das Kästchen System ist eine Zusammenfassung für das restliche System. Darunter fällt das Scheduling von Tasks, Speichermanagement, Treiberverwaltung und die Ausführung anderer Tasks, die bestimmte Aufgaben erledigen. Das System liefert an den Controller das aktuelle Bild der Kamera. Entspricht es nicht bestimmten Voraussetzungen, wird es zunächst an bestimmte Bedingungen angepasst. Der wichtigste Punkt ist, dass das Bild ein Graustufen-Bild sein muss. Es wäre auch möglich eine Skalierung des Bildes vorzunehmen, wenn es zu groß ist oder die Berechnung des Verfahrens zu lange dauert. Allerdings geht damit oft auch ein Verlust von wichtigen Bildinformationen einher.

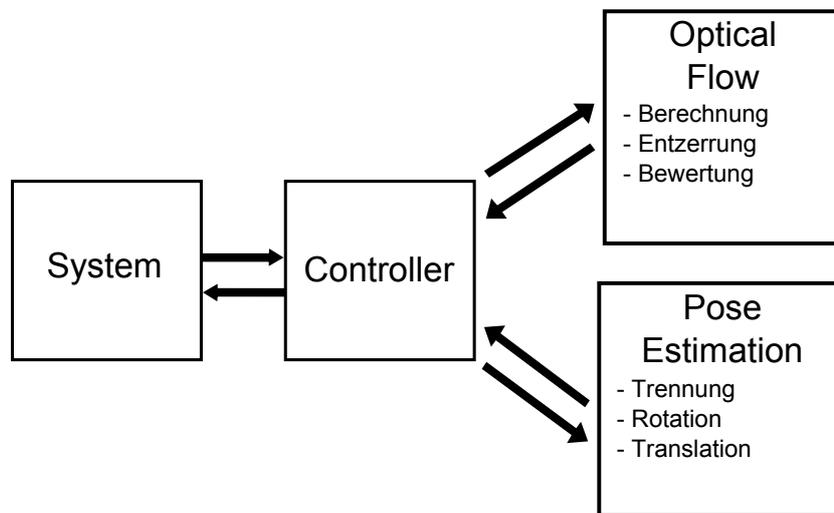


Abbildung 1: Schematische Skizzierung der Funktionsweise

In dem Kästchen Optical Flow wird der optische Fluss berechnet. Der optische Fluss ist eine Schätzung, um wie viel Pixel sich ein bestimmter Pixel in einem Bild bewegt hat. Dafür gibt es zwei Methoden: Lucas-Kanade (Lucas und Kanade (1981)) und Horn-Schunck (Horn und Schunck (1981)) bzw. Farneback (Farneback (2003)). Die spezielle Implementierung der Lucas-Kanade Methode von Jean-Yves Bouguet (Bouguet (2001)) wird zur visuellen Odometrieerfassung benutzt, da diese nicht so rechenintensiv wie die Horn-Schunck oder Farneback Methode ist. Die verwendete Methode setzt Graustufen-Bilder voraus, da in diesen Bildern nach markanten Punkten gesucht wird. Markante Punkte sind sehr eindeutige Übergänge von schwarz auf weiß oder weiß auf schwarz in der horizontalen Ebene. Punkte, die auf einer vertikalen Linie liegen, können in einer voranschreitenden Bewegung des Bildes nicht verfolgt werden. Aus diesem Grund werden solche Stellen nicht berücksichtigt.

Sind die markanten Punkte bzw. Merkmale gefunden, müssen diese Punkte korrigiert werden, um die Linsenverzeichnung zu entfernen. Um diesen Schritt durchzuführen, gibt es zwei Möglichkeiten. Die eine Möglichkeit besteht darin, ein entwickeltes Verfahren von Nico Manske (Manske (2008)) zu verwenden. Die andere Möglichkeit ist auf Funktionen von OpenCV zu setzen. Das Verfahren von Nico Manske ist eine Vorwärtstransformation bei dem ausgehend vom Quellbildpunkt ein Zielbildpunkt berechnet wird.

Als letzter Schritt zur Bestimmung des optischen Flusses werden die gefundenen Punkte bewertet, um vorhandene Störfaktoren wie Lichtreflexionen oder Menschen weitestgehend herauszufiltern. Dabei wird davon ausgegangen, dass ein Roboter sich gleichmäßig bewegt.

Aus diesem Grund wird über sieben Bilder hinweg geschaut, ob sich das gefundene Merkmal um mehr als 30 Grad in der horizontalen Ebene verschoben hat. Ist dies der Fall, wird der Punkt entfernt und nicht näher betrachtet. Die Liste mit gültigen Merkmalen wird zurück an den Controller gegeben, der diese wiederum an das Kästchen Pose Estimation übergibt.

In der Pose Estimation ist der erste Schritt eine Auftrennung des Bildes in drei Bereiche: Bodenregion, Horizont und Himmelsregion. Die Himmelsregion befindet sich im oberen Bereich des Bildes und wird zur Berechnung der Rotation benutzt. In der Regel befindet sich in der Himmelsregion eine sehr weit entfernte Sicht des Roboters auf die Umgebung. Durch diesen Umstand ist die Translation in diesem Bereich so gering, dass diese vernachlässigt werden kann und direkt die Rotation berechnet wird. Der Horizont befindet sich ungefähr mittig im Bild. In diesem Bereich werden sämtliche Punkte verworfen, da sich die Rotation und Translation zu stark vermischen und keine Möglichkeit besteht, die beiden Bewegungen klar voneinander zu trennen. Die Bodenregion ist im unteren Bereich des Bildes definiert. Diese Region dient der Berechnung der Translation. Dafür werden die Punkte mittels einer projektiven Transformation in ein Weltkoordinatensystem des Roboters transformiert und dann die berechnete Rotation abgezogen, so dass die reine Translation übrig bleibt. Die berechnete Rotation und Translation werden zurück an den Controller übergeben. Der Controller übergibt die Werte an das System.

### 2.1.2 Probleme und Risiken

Da das visuelle Odometriesystem im Wesentlichen auf der Bildverarbeitung basiert, ist es wichtig, dass die Bilder gute und viele Merkmale bieten. In Abbildung 2 ist ein Beispielbild zu sehen, das mit der Carolo-Cup Plattform aufgenommen worden ist. Dabei ist zu erkennen, dass sich der optische Fluss im Wesentlichen auf die Start- und Endpunkte der mittleren Fahrbahnmarkierung bezieht. Alle gefundenen Punkte, die sich auf dem Fahrzeug direkt befinden, können verworfen werden, da hier keine Bestimmung möglich ist. Die restliche Umgebung bietet keine weiteren Möglichkeiten für ein Finden und Verfolgen von Merkmalen.

Wenn nur wenig Merkmale bzw. Punkte gefunden werden, ist es möglich, dass sich die Anzahl schnell verkleinern kann. Da die Art der Linsenverzeichnungskorrektur eine Vorwärtstransformation ist, gibt es Quellbildpunkte, zu denen keine Zielbildpunkte berechnet werden können. Es gibt keine Möglichkeit diesen Umstand zu verhindern, ohne den Berechnungsaufwand enorm zu vergrößern.

Obwohl die optimierte Variante zur Berechnung des optischen Flusses eingesetzt wird, ist der Berechnungsaufwand noch immer enorm. Wie schon erwähnt, gibt es keine optimale Lösung für dieses Problem. Um die Berechnung zu beschleunigen, kann das Bild verkleinert werden, allerdings gehen damit auch wichtige Bildinformationen verloren, die zu einem Verlust von Merkmalen führen bzw. das Auffinden von Merkmalen erschweren oder unmöglich machen.

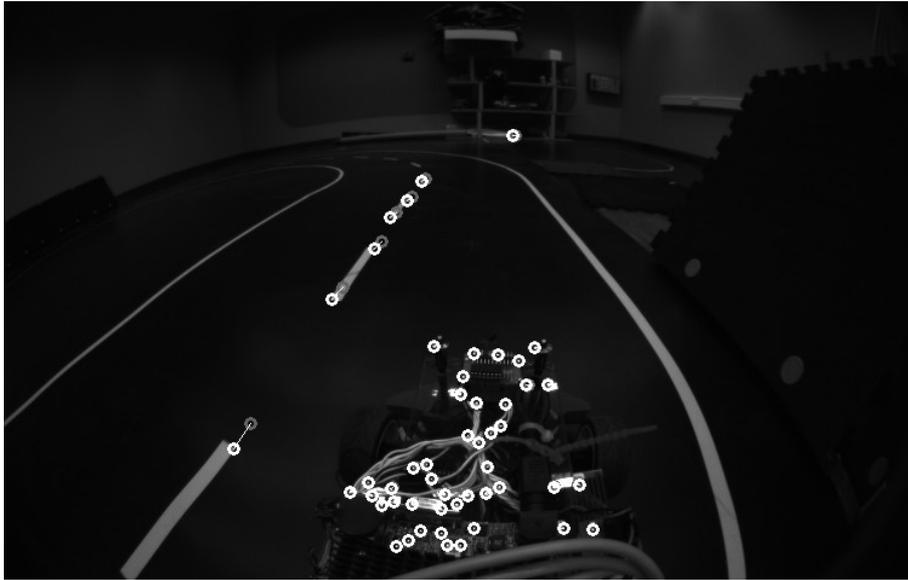


Abbildung 2: Problem der Erkennung

### 2.1.3 Fazit

Das visuelle Odometriesystem ist zuerst auf der Fahrzeugplattform Carolo-Cup getestet worden. Dabei hat sich herausgestellt, dass dieses System von Grund auf nicht lauffähig ist. Die Berechnungszeit (210 Millisekunden) für den optischen Fluss übertrifft die zulässige Zykluszeit (33,3 Millisekunden) um das sechs Fache. Wird das Bild runter skaliert, um die Berechnungszeit zu senken, wird die benötigte Zeit trotzdem noch über dem zulässigen Höchstmaß liegen. Darüber hinaus werden wichtige Bildinformationen entfernt. Die wenigen Merkmale, die auffindbar sind, werden abermals dezimiert.

Die Einsatzfähigkeit auf dem CampusBot ist noch nicht getestet worden. Allerdings sollte die Zeit für die Berechnung kein Hindernis darstellen, da die PC-Komponente bei dem CampusBot über wesentlich mehr Leistung verfügt. Allerdings besteht noch keine Erkenntnis darüber, wie gut die Beschaffenheit von Fluren sich für diese Methode eignet. Sollte die Beschaffenheit in einem Bild zu wenig Merkmale aufweisen, so besteht die Möglichkeit, das Verfahren für die Berechnung des optischen Flusses zu wechseln auf die Methode von Farneback. Dies wird allerdings den Aufwand für die Berechnung wieder erhöhen, da bei dieser Methode für jeden Bildpunkt in dem aktuellen Bild der optische Fluss berechnet wird und nicht nur für ein paar markante Punkte.

## 2.2 Projekt 2: RatSLAM

In Projekt 2 soll das SLAM Verfahren RatSLAM implementiert werden. Die Implementierung wird zuerst für die Plattform CampusBot erfolgen, da hier bereits alle Voraussetzungen

durch das visuelle Odometrie-System (Unterabschnitt 2.1) zum Start von Projekt 2 erfüllt sein werden. Anschließend soll die Portierung von RatSLAM auf die Fahrzeugplattform Carolo-Cup erfolgen. Um diesen Schritt zu ermöglichen, muss RatSLAM beschleunigt werden, indem zum einen Sensoren benutzt werden, um die von RatSLAM geforderten Messwerte zu erhalten, zum anderen muss aber auch geschaut werden, ob der Algorithmus von RatSLAM beschleunigt oder durch bessere Programmierung optimiert werden kann.

Neben einer generellen Lauffähigkeit ist auch wichtig, wie hoch die Berechnungszeit auf den beiden unterschiedlichen Plattformen für RatSLAM ist. Es gibt drei mögliche Messszenarios. Das erste Szenario ist eine Messung der Berechnungszeit, wenn RatSLAM noch keine Umgebung erfasst hat. Dies bedeutet, dass keinerlei Daten existieren. Das zweite Szenario stellt eine Messung der Berechnungszeit dar, wenn Daten gesammelt worden sind. Im dritten Szenario wird eine erneute Messung durchgeführt, wenn die Datenmenge weiter angestiegen ist. Daraus lässt sich eventuell ein Rückschluss auf die Berechnungszeit und Größe der Umgebung ziehen.

Ein weiterer wichtiger Punkt ist auch die Qualität der Karten und die Positionierung des Roboters innerhalb dieser Karten. Es ist keine exakte Raumvermessung notwendig, allerdings sollte der Roboter in der Lage sein eine Karte zu erstellen, die die Bewegung des Roboters exakt wieder gibt und sich innerhalb der erstellten Bewegungskarte exakt wiederfindet und nicht stark davon abweicht. Es ist nicht förderlich, wenn die Position in der Karte z.B. 20 Zentimeter abweicht von der realen Welt und aus diesem Grund gegen eine Wand fährt, obwohl eine Tür passiert werden sollte. Es ist auch wichtig, dass die Relokalisierung schnell und genau arbeitet. Dabei ist mit Relokalisierung das Finden der Position in der Karte gemeint, wenn der Roboter bekanntes Terrain verlässt und danach wieder auf eine bekannte Umgebung trifft. Hierbei soll er schnell merken, dass er diese Umgebung schon kennt und nicht anfangen eine neue Umgebung aufzubauen.

### 2.2.1 Risiken

Eines der am schwierigsten einzuschätzenden Faktoren ist die Berechnungszeit des RatSLAM Verfahrens. Es ist darin begründet, dass zwar eine Menge praktische Tests durchgeführt worden sind, auch mit Angabe der Berechnungszeit, allerdings später nicht auf die Rechenleistung des Computers, auf dem RatSLAM ausgeführt wurde, eingegangen worden ist. Dieser Umstand erschwert eine Abschätzung, wie lange RatSLAM für die Berechnung benötigt.

Ein weiterer Umstand, der Einschränkungen oder weitere Aufgaben mit sich bringen kann, ist die Tatsache, dass RatSLAM auf der Verarbeitung von Bildinformationen basiert. Dabei ist die Helligkeit der Umgebung wichtig für die Wiedererkennung. Wird eine Umgebung in einem sehr hellen Licht kartografiert, so kommt es vor, dass während der Dämmerung die gleiche Umgebung nicht mehr als bekannt identifiziert werden kann und eine neue Teilkarte erstellt wird. Kleinere Lichtunterschiede verursachen nicht dieses Problem. Ein weiteres

Risiko stellt eine dynamische Umgebung dar. Dies bedeutet, dass sich die Umgebung im Laufe der Zeit verändert, da Menschen durch das Bild laufen oder Gegenstände verschwinden bzw. auftauchen können. Die praktischen Tests haben ergeben, dass dies in einem geringen Umfang keine Konflikte verursacht. Wie das Verhalten von RatSLAM aber in größeren dynamischen Umgebungen ist, dazu ist keine Aussage getroffen worden.

## 3 Masterthesis

In der Masterthesis soll ein Roboter um die Funktion einer autonomen Umgebungskartierung erweitert werden. Autonome Umgebungskartierung bedeutet, dass der Roboter eine Unterscheidung zwischen bekanntem und unbekanntem Terrain durchführt und wenn ein Terrain unbekannt ist, dieses erkundet und selbstständig in die Karte integriert. Dabei gibt es zwei Möglichkeiten eine Umgebung zu erkunden. Die eine Möglichkeit besteht in der Beschränkung der Umgebung, wie z.B. bei einer Straße die Fahrbahnmarkierungen oder in dem der Roboter anfängt, den Raum zu erkunden und mit Blick auf die Karte versucht, neue Umgebungen zu finden.

Dabei kommt Sensorik zum Einsatz, um Kollisionen mit Personen oder Gegenständen zu vermeiden. Die Sensorik kann aber auch für die Raumerkundung genutzt werden, in dem sie offene Türen oder neue Gänge entdeckt. Es soll auch möglich sein, die erstellte Karte über WLAN an ein Tablet oder Computer zu schicken, sodass sich eine Person die Karte anschauen kann. Ein weiterer Gedankengang wäre dann eine Interaktion zwischen Roboter und Mensch zur Orientierung in Umgebungen.

### 3.1 Konzept

#### 3.1.1 CampusBot

In dem Konzept für die Plattform CampusBot geht es um das höhere Ziel einer selbstständigen Erstellung einer Karte des Campus Berliner Tor der Hochschule für Angewandte Wissenschaften Hamburg (Abbildung 3). Da dies allerdings ein enormes Unterfangen darstellt, werden anfänglich nur ein Flur und eventuell ein paar angrenzende Räume erkundet. Im Wesentlichen soll mit dem CampusBot ergründet werden, wie die Behandlung von dynamischen Umgebungen durch RatSLAM funktioniert und wo sich die Grenzen des Algorithmus befinden. Es soll außerdem überprüft werden, inwieweit Verbesserungen zur Verarbeitung von dynamischen Umgebungen möglich sind. Zu einem späteren Zeitpunkt kann dies noch auf mehrere Stockwerke bzw. den gesamten Campus erweitert werden.

Um autonom eine Umgebungskartierung durchzuführen, muss kontinuierlich zusammen mit der bereits erstellten Karte die Umgebung erkundet werden, sodass das System bzw. der Roboter weiß, welche Umgebung er kennt und wo sich eventuell noch unbekanntes Terrain befindet. Werden neue und unbekannte Umgebungen entdeckt, werden diese in die Karte integriert. Dabei ist es besonders wichtig, dass eventuelle Schleifen, also Verbindungen zwischen zwei bekannten Punkten, richtig miteinander verknüpft werden.

Um neue, bisher nicht erkundete Räume zu entdecken und Kollisionen zu vermeiden, wird ein Laser-Scanner verwendet, der eine Reichweite von vier Metern besitzt und bis zu zwölf Abtastungen pro Sekunden durchführt. Geht man davon aus, dass der CampusBot in der Regel in der Mitte eines Flurs fährt, so bleibt zwischen den Seiten des CampusBots und

der Flurwand weniger als zwei Meter Platz und somit kann zwischen einer Wand und einer Tür unterschieden werden. Ist es ein neu entdeckter Raum, der bisher nicht in der Karte verzeichnet ist, erkundet der CampusBot diesen Raum.



Abbildung 3: Karte des Campus Berliner Tor (AStA)

Darüber hinaus soll der Laser-Scanner auch eingesetzt werden, um Hindernisse und Personen zu erkennen und gegebenenfalls ausweichen oder stoppen und warten, bis der Weg wieder frei ist. Hierbei besteht eine Schwierigkeit in der Unterscheidung, ob der Flur zu Ende ist oder sich nur ein Hindernis im Weg befindet. Ein einfacher Ansatz wäre zu schauen, ob sich neben dem Hindernis genug freier Raum für eine Durchfahrt befindet. Ist dies der Fall, kann in den meisten Fällen davon ausgegangen werden, dass der Flur hinter dem Hindernis weiter verläuft. Eine weitere Aufgabe stellt die Unterscheidung zwischen Wand und geschlossener Tür dar. Im schlimmsten Falle wird eine geschlossene Tür als eine Wand identifiziert und nicht weiter beachtet werden. Im besten Falle funktioniert die Unterscheidung auf Antrieb und es ist keine spezielle Heuristik notwendig, um zwischen Wänden und Türen unterscheiden zu müssen.

Um zu überprüfen, wie gut RatSLAM mit dynamischen Umgebungen verfährt, wird zunächst ein Flur kartiert. Anschließend wird der Flur mit einem Hindernis ausgestattet, sodass der CampusBot immer noch den Flur entlang fahren kann. Bei diesem Versuch soll überprüft werden, ob RatSLAM eine neue Strecke erkennt oder weiter auf der bereits be-

kannten Strecke fährt. Ist dieser Versuch gelungen, sollen weitere Hindernisse in den Flur gestellt werden und überprüft werden, wie RatSLAM mit den neuen Hindernissen umgeht. Mittels diesem Versuch kann festgestellt werden, inwieweit sich die Umgebung, in der sich RatSLAM bewegt, verändern kann ohne das eine Beeinflussung der Lokalisierung stattfindet. Ist die Grenze überschritten, in der RatSLAM eine sichere Lokalisierung gewährleistet, soll ergründet werden, was zur Auslösung der Fehllokalisierung geführt hat und überprüft werden, ob es Techniken gibt, um weiterhin eine sichere Lokalisation zu gewährleisten.

Um die Kartendaten darstellen zu können, soll RatSLAM um eine Komponente erweitert werden, die die Kartendaten per WLAN an ein Tablet oder einen Computer überträgt. Ebenfalls soll die aktuelle Position des CampusBots innerhalb dieser Karte dargestellt werden. Für einen späteren Zeitpunkt wäre auch in diesem Fall eine Aufwertung der Karte mit weiteren Informationen möglich. Dabei können solche Informationen zum Beispiel die Position von Laboren oder Räumen der Mitarbeiter enthalten.

### 3.1.2 Carolo-Cup

In dem Konzept für die Forschungsplattform Carolo-Cup geht es vor allem um die Korrektur des Spurführungsalgorithmus in kritischen Situationen. In Abbildung 4 ist die aktuelle Karte des Wettbewerbs Carolo-Cup der Technischen Universität Braunschweig ohne Hindernisse dargestellt. Kritische Situationen können dadurch entstehen, dass z.B. bis zu einem Meter der Fahrbahnmarkierung von maximal zwei Fahrbahnmarkierungen fehlen kann. Kommt ein Spurführungsalgorithmus in solch eine Situation, ist es schwer, die Fahrbahnführung eindeutig zu bestimmen. In dieser Situation soll RatSLAM mit der erstellten Karte eingreifen und den Spurführungsalgorithmus daran hindern, in ein unkontrolliertes Verhalten auszuarten, indem z.B. das Fahrzeug in einer Linkskurve nach rechts ausbricht. Darüber hinaus kann die Karte auch mit weiteren Informationen aufgewertet werden. Kreuzungen und Standorte von möglichen Hindernissen werden in der Karte vermerkt, sodass das System weiß, in welchen Regionen eine besondere Vorsicht gegeben sein muss.

Zunächst muss RatSLAM von der Plattform CampusBot auf die Plattform Carolo-Cup portiert werden. Dabei gibt es einen enormen Unterschied zwischen der Rechenleistung. Während der CampusBot mit einem Intel Core i7 und 16GB Arbeitsspeicher ausgestattet ist, besitzt der PC der Carolo-Cup Plattform einen Intel Atom Z530 mit 2GB Arbeitsspeicher. Eine Beschleunigung des Algorithmus soll zum einen durch die Verwendung von Sensoren erreicht werden, die das visuelle Odometrie System (Unterabschnitt 2.1) ersetzen. Dabei wird für die Distanzmessung ein Hall-Sensor und für die Rotationserfassung des Fahrzeuges ein Drehraten-Sensor verwendet. Durch den Einsatz dieser Sensoren ist es möglich, eine genauere Karte zu erstellen, als es mit dem visuellem Odometrie System möglich ist, da hier keine Bildtransformationen vorgenommen werden, die Ungenauigkeiten hervor rufen. Um den Algorithmus RatSLAM selbst zu beschleunigen, wird versucht, den Algorithmus für ein Multicore System zu implementieren. Eine mögliche Kombination wäre, dass in einem

Thread die Vorverarbeitung, sprich Auswertung der Sensorinformationen, etc., stattfindet und in einem anderen Thread die Aktualisierung des neuronalen Netzes.

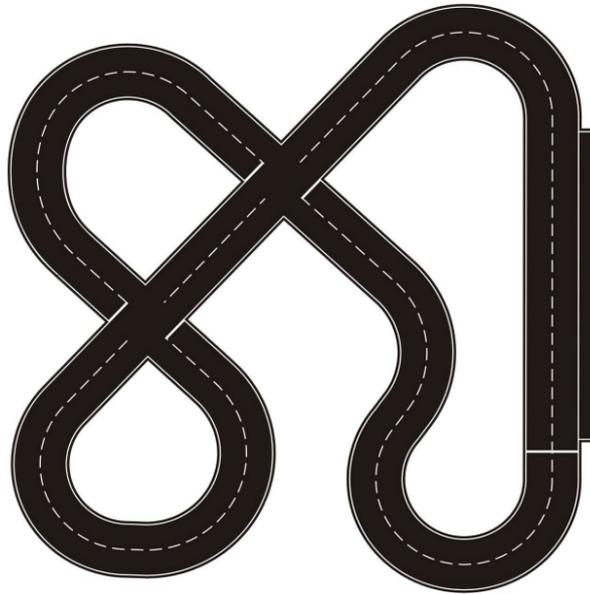


Abbildung 4: Parcours während des Carolo-Cup (Carolo-Cup)

Bisherige Implementierungen von SLAM Verfahren haben große Probleme mit der Relokalisierung des Fahrzeuges gehabt, nachdem diese die Strecke verlassen haben oder eine so große Diskrepanz zwischen den in der Karte vermerkten Strecken und den tatsächlich gemessenen Strecken entstanden ist, dass das SLAM Verfahren das Fahrzeug nicht korrekt positioniert hat. Dieses Problem wird durch eine verbesserte Translationsmessung bewältigt.

Die Karte kann entweder mittels des Spurführungsalgorithmus oder per manueller Steuerung erstellt werden. Zu bevorzugen ist die erste Variante, da die erstellte Karte besser zu dem Fahrverhalten des Spurführungsalgorithmus passt. Wird das Fahrzeug manuell durch den Parcours geführt, da der Spurführungsalgorithmus an bestimmten Stellen ein unkontrolliertes Verhalten aufweist, erstellt RatSLAM eine perfekte Karte, allerdings kann diese teilweise von dem eigentlichen Fahrverhalten abweichen. Diese leichten Abweichungen sollten kein Hindernis für eine einwandfreie Lokalisierung darstellen.

Wie schon weiter oben erwähnt, wäre es wünschenswert die Karte um weitere Informationen wie z.B. Kreuzungen oder stationäre Hindernisse zu erweitern. Für beide Fälle sind bereits Algorithmen entwickelt, die Kreuzungen und Hindernisse erkennen. Die Schwierigkeit bei der Hinderniserkennung besteht darin, eine klare Unterscheidung zwischen einem fahrenden Hindernis (ein anderes Auto auf dem Parcours) und einem statischen Hindernis (feststehend auf dem Parcours) zu treffen. Darüber hinaus muss auch geschaut werden, wie

diese Informationen mit der Karte verknüpft werden können, ohne einen zu tiefen Eingriff in RatSLAM selbst herbeizuführen.

### 3.2 Chancen und Risiken

In bisherigen praktischen Einsätzen hat RatSLAM bewiesen, dass eine große Umgebung fehlerfrei und schnell bearbeitet werden kann. Dabei ist eine Strecke von 55 Kilometern mit 50 Schleifen kein Hindernis und das System bleibt stabil berechenbar ohne einen explosionsartigen Anstieg der Berechnungszeit. Die Datenmengen sind ebenfalls nicht explosionsartig angestiegen, sondern sind kontinuierlich mit der Größe der Umgebung gewachsen. Diese Erkenntnisse lassen die Schlussfolgerung zu, dass die Kartierung des Campus Berliner Tor der Hochschule für Angewandte Wissenschaften Hamburg theoretisch einwandfrei funktionieren sollte.

Es ist auch gezeigt worden, dass die Relokalisierung sehr schnell und präzise arbeitet. Dies bedeutet, sollte der Roboter auf eine unbekannte Umgebung treffen und stößt anschließend wieder auf ein bekanntes Stück, werden mögliche Schleifen richtig verbunden und der Roboter befindet sich sowohl in der Karte als auch in der Realität an der gleichen Stelle. Dieser Fall trifft auch zu, sollte RatSLAM eine Umgebung fälschlicherweise als unbekannt deklarieren und in einer kurzen Entfernung merken, dass sich der Roboter wieder in einer bekannten Umgebung befindet, entstehen im schlimmsten Falle kleine Ausbuchtungen, die aber häufig keine weitere Rolle spielen.

Das größte Risiko liegt in der Unterschätzung des Arbeitsaufwandes bzw. der Zeit. Es ist zurzeit schwer abzuschätzen, wie viel Entwicklungszeit in die verschiedenen Konzepte für eine grundlegende Funktionalität benötigt wird. Entsprechend müssten die gesteckten Ziele bei massiver Überschreitung der Zeit oder des Arbeitsaufwandes verringert werden. So würde zwar auch der Funktionsumfang vermindert werden, aber es gäbe eine lauffähige Grundversion, auf der in späteren Arbeiten aufgebaut werden kann.

Ein weiteres Risiko stellt eine fehlerhaft arbeitende Sensorik dar. Produzieren Sensoren Werte, die außerhalb des gültigen Bereichs liegen oder nicht zur aktuellen Umgebung passen, müssen die Ursachen sehr aufwendig herausgefunden werden, indem viele verschiedene Tests durchgeführt werden oder wirklich jeder einzelne Schritt zwischen Empfang und Analyse der Sensorik zurück verfolgt wird und geschaut wird, ob alles korrekt funktioniert. Darüber hinaus können auch geringe Abweichungen bereits eine große Auswirkung auf das Endergebnis haben. Ändern sich die Parameter, die für die Linsenverzeichnungskorrektur zuständig sind, so kann dies bereits enorm veränderte Ergebnisse in der Odometrie hervorrufen ohne dass dieses Fehlverhalten schnell erkannt werden kann.

## 4 Weiteres Vorgehen

Da Projekt 1 noch nicht fertiggestellt ist, muss dies zuerst vollendet werden. Dies ist notwendig, da die Entwicklung für RatSLAM zuerst auf der Plattform CampusBot stattfinden soll. Läuft RatSLAM in vielen Punkten hervorragend auf dem CampusBot, soll eine Portierung für die Forschungsplattform Carolo-Cup angestrebt werden. Dabei wird die meiste Zeit nicht in die direkte Portierung für RatSLAM fließen, sondern an das Abgreifen der notwendigen Sensorwerte und die Umrechnung. Wenn damit auch Projekt 2 abgeschlossen ist, kann mit der Masterthesis begonnen werden.

Die Richtung von weiterführenden Aufgaben geht vor allem in die intensivere Interaktion zwischen Mensch und Maschine, im Speziellen zwischen Mensch und der Forschungsplattform CampusBot. Da der CampusBot einmal eine komplette Karte des Campus Berliner Tor der HAW Hamburg besitzen soll, sind die Möglichkeiten der Interaktion weit gestreut. Von einem Führen der entsprechenden Person bis hin zum Zeigen des Weges auf dem Smartphone, Tablet oder Computer über Serviceleistungen, wie z.B. Holen von Getränken oder Speisen aus der Mensa. Um dieses Ziel zu erreichen, muss allerdings noch viel Arbeit investiert werden.

Abschließend bleibt zu sagen, dass mit RatSLAM, wenn es so gut funktioniert wie gehofft, ein aufregender und spannender Algorithmus zur Kartografie und Orientierung in Umgebungen entstanden ist. Es ist nicht nur ein komplett konträrer Ansatz, vor allem in der Datenverwaltung, zu vorherigen Algorithmen, sondern verknüpft die Welt der Informatik weiter mit biologisch motivierten Ansätzen.

## Literatur

- [AStA ] ASTA: *HAW Hamburg / Lageplan HAW*. – URL <http://www.asta.haw-hamburg.de/files/imagepicker/1/LageplanHAW.jpg>
- [Becker 2012a] BECKER, Torben: Biologisch motivierter SLAM Algorithmus, Anwendung 1 / Hochschule fuer Angewandte Wissenschaften (HAW) Hamburg, Hamburg. 2012. – Forschungsbericht
- [Becker 2012b] BECKER, Torben: RatSLAM, Anwendung 2 / Hochschule fuer Angewandte Wissenschaften (HAW) Hamburg, Hamburg. 2012. – Forschungsbericht
- [Bouguet 2001] BOUGUET, Jean-Yves: Pyramidal Implementation of the Affine Lucas Kanade Feature Tracker Description of the algorithm / Intel Corporation, Microprocessor Research Labs. 2001. – Forschungsbericht
- [Campbell u. a. 2005] CAMPBELL, J. ; SUKTHANKAR, R. ; NOURBAKHSI, I. ; PAHWA, A.: A Robust Visual Odometry and Precipice Detection System Using Consumer-grade Monocular Vision / Carnegie Mellon University, Intel Research Pittsburgh, NASA Ames Research Center. 2005. – Forschungsbericht
- [Carolo-Cup ] CAROLO-CUP: *Regelwerk 2013, S. 19*. – URL <http://www.carolo-cup.de/>
- [Farneback 2003] FARNEBACK, Gunnar: Two-Frame Motion Estimation Based on Polynomial Expansion / Computer Vision Laboratory, Linköping University, Linköping, Schweden. 2003. – Forschungsbericht
- [Horn und Schunck 1981] HORN, Berthold ; SCHUNCK, Brian: Determining Optical Flow / Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge. 1981. – Forschungsbericht
- [Lucas und Kanade 1981] LUCAS, Bruce ; KANADE, Takeo: An Iterative Image Registration Technique with an Application to Stereo Vision / Computer Science Department, Carnegie-Mellon University, Pittsburgh. 1981. – Forschungsbericht
- [Manske 2008] MANSKE, Nico: Kamerabasierte Praezisionsnavigation mobiler Systeme im Indoor-Bereich, Kapitel 4.2 / Hochschule fuer Angewandte Wissenschaften (HAW) Hamburg, Hamburg. 2008. – Forschungsbericht
- [OpenCV ] OPENCV, Open Source Computer Vision L.: *Website*. – URL <http://www.opencv.org/>
- [Siciliano und Khatib 2008] SICILIANO, Bruno ; KHATIB, Oussama: *Springer Handbook of Robotics*. Springer, 2008

## Abbildungsverzeichnis

1	Schematische Skizzierung der Funktionsweise . . . . .	5
2	Problem der Erkennung . . . . .	7
3	Karte des Campus Berliner Tor (AStA) . . . . .	11
4	Parcour während des Carolo-Cup (Carolo-Cup) . . . . .	13