



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

## Anwendung 2 - SoSe 2011

Ben Struss

Autonome 3D-Kartierung von halbstatistischen  
Indoor Umgebungen

## Inhaltsverzeichnis

<b>1 Einführung</b>	<b>3</b>
1.1 Vorstellung der Plattform . . . . .	4
1.2 Zielsetzung . . . . .	4
<b>2 Related Work</b>	<b>5</b>
2.1 3D-Scanmatching . . . . .	5
2.2 SLAM-Verfahren . . . . .	7
2.3 Kartierung . . . . .	9
2.4 Exploration . . . . .	10
<b>3 Fazit und Abgrenzung</b>	<b>11</b>
<b>4 Ausblick</b>	<b>12</b>
<b>Literatur</b>	<b>13</b>

## 1 Einführung

Die Erkundung, Erfassung und Interpretation unbekannter Umgebungen ist etwas, das wir Menschen ständig intuitiv durchführen. Sei es bei einem Stadtbummel oder nur der Einkauf in einem, uns fremden, Supermarkt. Diese Fähigkeit ist eine der wichtigsten Voraussetzungen innerhalb dieser Umgebung unsere Ziele zu erreichen. Nur wenn geographische Informationen vorliegen ist es z.B. möglich einen Weg zu planen. Zusätzlich müssen permanent mögliche Hindernisse erkannt und ihr Einfluss auf das Vorrankommen bewertet werden. Gegebenenfalls führt dies zu einer Änderung der Lauf- oder Fahrtrichtung inklusive erneuter Planung der Strecke auf Basis der gewonnenen Erkenntnisse.

Die gleichen Anforderungen gelten im wesentlichen auch für autonome Haushaltsroboter. Es wird erwartet, dass sich ein Roboter automatisch den Gegebenheiten anpasst, die Umgebung beobachtet und auf Änderungen reagiert. Gerade bei Robotern, die den Menschen Arbeiten abnehmen und das Leben erleichtern sollen, kann nicht erwartet werden, dass ständig manuelle Anpassungen vorgenommen werden müssen.

In dieser Arbeit soll ein Verfahren entwickelt werden, welches einem Serviceroboter die autonome Erkundung und Kartierung seiner Umgebung mit Hilfe einer 3D-Tiefenkamera ermöglicht. Im weiteren Verlauf dieses Kapitels wird kurz die Roboterplattform vorgestellt, sowie die Zielsetzung und die Anforderungen an das Verfahren konkretisiert. In Kapitel 2 werden verwandte Arbeiten vorgestellt und in Kapitel 3 wird die Abgrenzung des zu entwickelnden Verfahrens zur related Work vorgenommen. Kapitel 4 gibt schließlich noch einen Ausblick auf das weitere Vorgehen.

## 1.1 Vorstellung der Plattform

Das Roboterlabor der HAW-Hamburg verfügt seit einiger Zeit über die SCITOS G5 Roboterplattform der Firma Metralabs<sup>1</sup>, welche für den Einsatz als Indoor-Serviceroboter zusätzlich mit einem Gelenkarm des Herstellers SCHUNK<sup>2</sup> ausgerüstet wurde. Dieser bietet insgesamt fünf Freiheitsgrade in der Bewegung sowie einen Greifer und ermöglicht es dem Roboter damit Gegenstände zu transportieren und zu manipulieren. Sensorseitig ist die Plattform ab Werk mit 24 Ultraschallsensoren zur Distanzmessung, einem umlaufenden Bumper zur Kollisionserkennung sowie einem 2D Laserscanner der Firma Leuze Electronis<sup>3</sup> ausgestattet. Als Odometriequelle fungieren zwei Umdrehungssensoren, die pro Radumdrehung jeweils 460 Ticks auslösen. Zusätzlich wird die Plattform im Rahmen dieser Arbeit um eine Microsoft KINECT<sup>4</sup> erweitert, welche als 3D Tiefenbildkamera, sowie als Farbkamera dient. Abbildung 1 zeigt den Roboter mit der angedeuteten Position der Kinect direkt auf dem Greifer.



## 1.2 Zielsetzung

In dieser Arbeit soll ein Verfahren entwickelt werden, welchen den in Kapitel 1.1 vorgestellten Roboter in die Lage versetzt eine Indoor-Umgebung autonom zu erkunden. Es wird angestrebt, dass der Roboter explizit die Möglichkeiten der am Arm montierten 3D-Kamera nutzt um auch schwer zugängliche Bereiche zu kartieren und so eine möglichst vollständige Karte der Räumlichkeiten zu erstellen. Das erwünschte Verhalten lässt sich mit der Metapher „*neugieriger Roboter*“ beschreiben.

Abbildung 1: SCITOS G5

Im weiteren Verlauf soll der Roboter in der Lage sein sich Anhand der Karte zu lokalisieren und Veränderungen der Umwelt in die Karte zu integrieren. Zusätzlich müssen die erfassten Informationen detailliert genug sein, damit es höheren Schichten möglich ist, Aufgaben zu planen und durchzuführen.

<sup>1</sup><http://www.metralabs.com/>

<sup>2</sup><http://www.schunk.de/>

<sup>3</sup><http://www.leuze.de/>, Modell ROTOSCAN RS4-2E

<sup>4</sup><http://www.xbox.com/de-DE/Kinect/>

## 2 Related Work

Um die in Kapitel 1.2 gestellten Anforderungen erfüllen zu können, müssen eine Reihe von Techniken und Ansätzen kombiniert und weiterentwickelt werden. In diesem Kapitel werden diese Gebiete kurz erläutert, sowie relevante und grundlegende Arbeiten der Bereiche vorgestellt.

### 2.1 3D-Scanmatching

Um lokal konsistente Karten erstellen zu können, ist es wichtig, dass für jeden Scan der Umgebung der genaue Punkt der Aufnahme bekannt ist. Nur so lassen sich mehrere Scans später nahtlos zusammenfügen, ohne dass Doppelkonturen oder andere Artefakte entstehen. Da jedoch sowohl die Odometrie des Roboters relativ ungenau ist (z.B. Schlupf der Räder, unebene Untergründe), als auch der Arm gewisse Toleranzen aufweist und so schon sehr kleine Fehler in der Aufnahmeposition zu Abweichungen von mehreren Zentimetern führen können, ist es notwendig den tatsächlichen relativen Versatz zwischen den Scans zu bestimmen. Es wird also typischerweise von einem bekannten Punkt aus gestartet oder der unbekannte Startpunkt als Nullpunkt angenommen und die erste Aufnahme gemacht. Nun muss jeder weitere Scan mit seinem Vorgänger verglichen und so die Translation sowie Rotation festgestellt werden. Durch Addition der relativen Veränderung auf die Koordinate der vorigen Aufnahme erhält man die Position des aktuellen Scans. Dies garantiert eine lokal konsistente Darstellung, führt aber auch dazu, dass sich eventuelle Fehler akkumulieren können.

Das bekannteste und verbreitetste Verfahren zum Scanmatching ist der *Iterative Closest Point*-Ansatz (ICP) der für 3D-Daten erstmals in (Besl und McKay, 1992) vorgestellt wurde. In ihrer Veröffentlichung beschreiben Besl und McKay wie die Ausrichtung einer Daten-Punktwolke (der neue Scan) zu einem bestehenden Modell (entweder ebenso eine Punktwolke oder auch geometrisch und mathematisch definierte Oberflächen) iterativ bestimmt werden kann. Da dieser Algorithmus nur die Konvergenz gegen ein lokales Minimum garantiert, ist es hilfreich mit einer möglichst guten Schätzung der Transformation zu starten.

Kern des Verfahrens ist die Minimierung der mittleren quadratischen Distanzen zwischen korrespondierenden Punkten (euklidische Distanz). Da die tatsächlichen Korrespondenzen apriori nicht bekannt sind, wird jedem Datenpunkt der euklidisch dichteste Punkt (diskret oder dichtester Punkt auf der beschreibenden Funktion, Ebene) des Modells zugeordnet und der jeweilige Distanzvektor bestimmt. Aus diesen Distanzvektoren werden nun die Rotation und Translation bestimmt und auf die Datenpunkte angewendet. Solange das quadratische Mittel der Distanzvektoren größer als ein definierter Schwellwert ist, wird die nächste Iteration durchgeführt. Durch die Transformation verschieben sich mit jedem Schritt die Korrespondenzen bis das gesamte Verfahren konvergiert.

Im Laufe der Zeit wurden viele Anpassungen und Optimierungen am klassischen ICP vorgenommen. Eine Übersicht über die ersten 10 Jahre bietet die Arbeit von (Rusinkiewicz und Levoy, 2001). So wurden unter anderem an der Auswahl der zu matchenden Punkte, der Gewichtung von Distanzen, der Filterung von Ausreißern und der Integration von Farb- und/oder Intensitätswerten in die Distanzmetrik geforscht. Eine Verbesserung, die sich explizit die teilweise planare Beschaffenheit natürlicher Umgebungen zunutze macht wird in (Segal u. a., 2009) unter dem Titel *Generalized-ICP* vorgestellt. Hier wird für jeden Punkt in Modell und Daten die Normale aus den umgebenden Punkten berechnet und der Einfluss auf die Transformation entsprechend gewichtet. So haben Abweichungen entlang der Normalen einen starken Effekt, während Verschiebungen der Punkte in Richtung der Ebene kaum eine Veränderung bewirken. Dies wird unter dem Begriff Ebene-zu-Ebene ICP zusammengefasst und bewirkt besonders in urbanen Umfeldern mit vielen ebenen Flächen Vorteile in der Geschwindigkeit der Konvergenz sowie der resultierenden absoluten Genauigkeit.

Mit dem Aufkommen von RGBD-Sensoren<sup>5</sup> wie der Microsoft KINECT, welche auf einer Technik von Primesense<sup>6</sup> basiert, wurde das ICP Verfahren in (Henry u. a., 2010) und (Engelhard u. a., 2011) um einen vorgelagerten Schritt ergänzt. Zunächst werden lokale Merkmale (SIFT (Lowe, 1999) in (Henry u. a., 2010), bzw. SURF (Bay u. a., 2006) in (Engelhard u. a., 2011)) aus dem RGB-Teil des aktuellen Scans extrahiert und mit den Merkmalen des Modells, bzw. des vorherigen Scans verglichen. Die gefunden Übereinstimmungen werden nun mittels der jeweiligen Bildkoordinaten und Tiefeninformationen in 3D-Punkte umgewandelt und es wird unter Verwendung des RANSAC-Algorithmus versucht eine gültige Transformation zwischen Scan und Modell zu finden. Diese Transformation kommt dem tatsächlichen Ergebnis typischerweise schon sehr nahe und wird als Startpunkt für das klassische ICP-Verfahren verwendet. Da der Versatz nun bereits deutlich kleiner ist, konvergiert das Verfahren schneller und führt mit größerer Wahrscheinlichkeit zum globalen Minimum. Gerade bei sehr hoch aufgelösten Punktwolken kann dieser Schritt viel Rechenzeit sparen.

---

<sup>5</sup>RGB + Depth, jeder Pixel enthält zusätzlich eine Entfernungsangabe

<sup>6</sup><http://www.primesense.com/>

## 2.2 SLAM-Verfahren

SLAM steht für *Simultaneous Localization and Mapping* und beschreibt Verfahren, die Roboter in die Lage versetzen eine unbekannte Umgebung gleichzeitig zu erkunden (mapping) und sich in ihr zu lokalisieren (localization). Während es bei den Scanmatching-Verfahren in 2.1 vor allem um eine lokal konsistente Darstellung der unmittelbaren Umgebung ging, versuchen SLAM-Ansätze die globale Konsistenz der erstellten Karte sicherzustellen. Die ersten bekannten Lösungen für das SLAM Problem basierten auf Extended Kalman Filtern (EKF) und wurden gegen Ende der 1980er Jahre veröffentlicht. Da diese Methoden jedoch ein ungünstiges Laufzeitverhalten von  $O(n^2)$  für  $n$  beobachtete Merkmale in der Umgebung haben, sind sie heutzutage etwas aus der Mode gekommen. (Thrun und Leonard, 2008)

In den letzten 10 Jahren sind vor allem auf Partikelfiltern basierende Verfahren in den Fokus der Forschung gerückt. Eines der bekanntesten von ihnen ist FastSLAM (Montemerlo und Thrun, 2003). Anstelle von einer Karte die alle Unsicherheiten und Fehlerwahrscheinlichkeiten beinhalten muss, besitzt bei diesem Verfahren jedes Partikel eine eigene Schätzung der aktuellen Position und der bisher zurückgelegten Trajektorie. Wird eine neue Beobachtung der Umgebung vorgenommen, so werden zunächst auf Basis der Odometriedaten alle Partikel um die gemessene Strecke weiterbewegt und für jedes Partikel eine zufällige, normalverteilte Abweichung in der Position aufaddiert. Durch dieses „Rauschen“ werden vorhandene Fehler in der Odometrie und andere Ungenauigkeiten explizit berücksichtigt. Nun wird die neue Beobachtung auf jeden Partikel angewendet und mittels einer Bewertungsfunktion festgestellt, wie gut sich die Beobachtung mit den bisherigen Beobachtungen des jeweiligen Partikels deckt. Partikel bei denen der Bewegungsvektor zufällig sehr gut stimmt, erzielen hier typischerweise höhere Werte. Diese Werte werden nun in einer *Resampling-Phase* als Wahrscheinlichkeit ein bestimmtes Partikel auszuwählen verwendet. Durch dieses „ziehen mit zurücklegen“ werden die am höchsten bewerteten Partikel mit erhöhter Wahrscheinlichkeit mehrfach ausgewählt, während schlecht bewertete Partikel aussortiert werden. Der große Vorteil dieser Gruppe von Verfahren ist, dass der Rechenaufwand beinahe ausschließlich mit der Zahl der Partikel linear skaliert. Um eine möglichst hohe Präzision zu erreichen, können also solange Partikel hinzugefügt werden, wie die Leistung der Hardware es zulässt. In Abbildung 2 sind drei Partikel mit ihren jeweiligen Positions- und Trajektoenschätzungen sowie der daraus resultierenden Karte abgebildet. Beim zweiten und dritten Partikel sind am rechten, oberen Rand gut die aus der fehlerhaften Positionsschätzung entstehenden Doppelkonturen zu erkennen. Von diesen drei Partikeln wird mit hoher Wahrscheinlichkeit der linke verwendet werden.

Zwei wesentliche Verbesserungen werden in (Grisetti u. a., 2007) vorgestellt. Auch dieser Ansatz basiert auf einem (Rao-Blackwellized) Partikelfilter, jedoch wird für die Fortschreibung der Partikel nicht die verhältnismäßig ungenaue Odometrie verwendet, sondern auf präzisere Sensorik wie Laserscanner und Scanmatchingverfahren zurückgegriffen. Dies ermöglicht es, die Verteilung der Partikel, wie in Abbildung 3 gezeigt, den Unsicherheiten des Scanmatchings

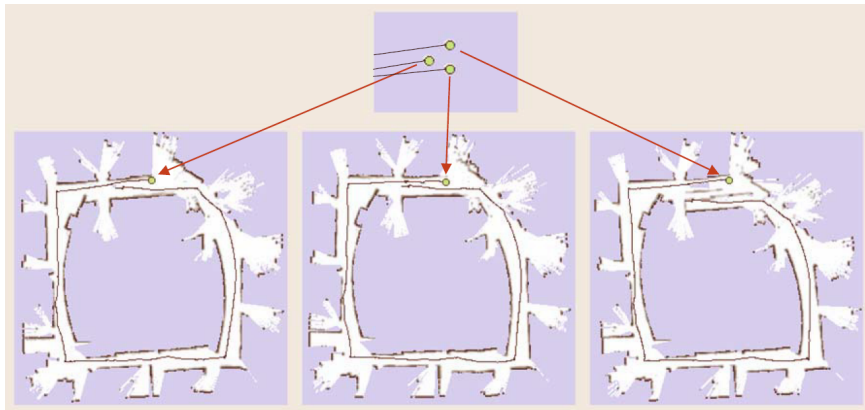


Abbildung 2: Grid-Based FastSLAM (Thrun und Leonard, 2008)

anzupassen. Bei a) ist die Unsicherheit in alle Richtungen sehr niedrig, so dass alle Partikel extrem dicht beieinander liegen. Im Fall b) kann zwar die Position in der Breite des Korridors sehr genau festgelegt werden, jedoch sind in Längsrichtung keine begrenzenden Strukturen sichtbar, so dass die Partikel in dieser Richtung entsprechend um den Schätzwert der Odometrie herum normalverteilt angeordnet werden. Sollten wie im Fall c) keinerlei Strukturen für ein Scanmatching zur Verfügung stehen, so wird, wie bei FastSLAM auch, komplett auf die Odometrie zurückgegriffen. Da dieses Vorgehen in den meisten strukturierten Umgebungen zu einer deutlich geringeren Streuung der Partikel führt, wurde zusätzlich die Häufigkeit des Resamplings reduziert und von der Streuung der Partikel abhängig gemacht. Dies verringert massiv das Risiko, dass durch zu häufiges probabilistisches Resampling die besten Schätzungen verloren gehen könnten.

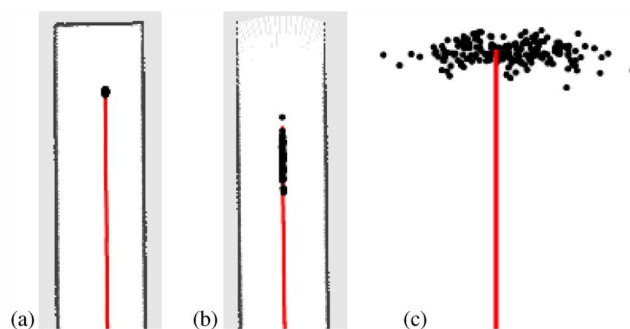


Abbildung 3: Partikelverteilung in verschiedenen Umgebungssituationen (Grisetti u. a., 2007)



### 2.3 Kartierung

Die Karte der Umgebung ist die wichtigste Grundlage für die Navigation des Roboters und damit wesentlicher Bestandteil der Aufgabenplanung. Ergebnis der vorrangegangenen SLAM- und Scanmatchingschritte sind eine Reihe von Punktwolken und ihre jeweiligen Aufnahmepositionen. Um die Handhabung zu vereinfachen, werden die Scans in ein gemeinsames Koordinatensystem gebracht und die Punktwolken zusammengefügt, was jedoch zu vielen Redundanzen führt, da einzelne Punkte in mehreren überlappenden Aufnahmen enthalten sind. Des Weiteren ist es für Navigationsaufgaben von Nachteil, dass keine Informationen über geschlossene Flächen und freien Raum vorliegen. Die Einteilung der Karte in Voxel<sup>7</sup> wie in (Roth-Tabak und Jain, 1989) vorgestellt, löst diese Probleme. Hierfür wird die Umgebung in ein festes Voxelraster eingeteilt und bei der Integration eines neuen Scans mit einem dem Raytracing ähnlichen Verfahren berechnet, welche Voxel frei und welche belegt sind.

Auf dieses Prinzip aufbauen wurde in (Wurm u. a., 2010) *Octomap* vorgestellt, welches die Voxel nicht mehr als 3D-Array speichert sondern in einem Octree<sup>8</sup> verwaltet. Hierdurch ist es möglich den Speicherbedarf drastisch zu reduzieren indem große Bereiche gleicher Belegung (z.B. freier Raum) auf einer hohen Knotenebene zusammengefasst und die Kindknoten abgeschnitten werden. Auch ist der Baum und damit die Größe des zu kartierenden Gebietes jederzeit erweiterbar. Es genügt dazu einen neuen Wurzelknoten zu erstellen und den bestehenden Baum darunterzuhängen. Ein weiterer Vorteil der Baumstruktur und auch in *Octomap* integriert ist, dass sich durch Beschränkung der Traversierungstiefe sehr einfach verschiedene Auflösungen der Karte abfragen lassen (Abbildung 4). Gerade der Bereich Navigation und Wegfindung kann hiervon profitieren, da typischerweise sehr große Datenmengen verarbeitet werden müssen, es aber selten auf den Zentimeter genau ankommt.

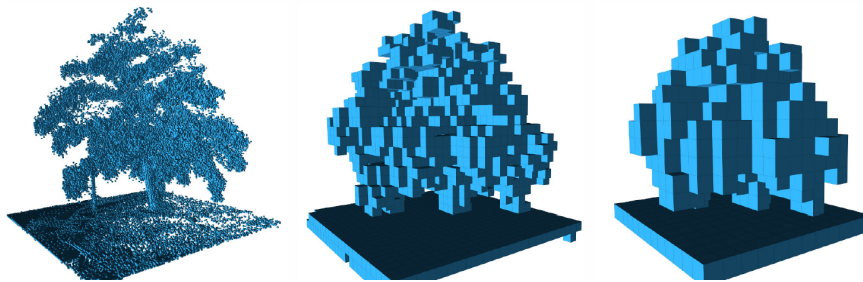


Abbildung 4: „OctoMap“: Abfrage verschiedenen Auflösungen, (Wurm u. a., 2010)

Ein Ansatz abseits der Voxelgrids mit dem Fokus auf semantischen Karten wird in (Rusu u. a., 2007) präsentiert. Basierend auf den Punktwolken wird zunächst ein Resampling zur „Glättung“

<sup>7</sup>Volumetrische „3D“-Pixel

<sup>8</sup>Ein dreidimensionaler Binärbaum

von Oberflächen und der Unterdrückung von Ausreißern und anschließend eine Segmentierung in geometrische Formen wie Flächen und Zylinder vorgenommen. Im folgenden werden mithilfe stark spezialisierter Algorithmen und Regeln Objekte von Interesse innerhalb der Karte identifiziert. So werden zum Beispiel Tische als ebene Flächen in einer Höhe zwischen 60 und 100cm beschrieben. Zusätzlich wurde die Möglichkeit integriert, neue Objekte anhand von vorhandenen 3D-Modellen z.B. aus Googles 3D Warehouse<sup>9</sup> zu erlernen.

Für den Umgang mit dynamischen oder halbstatistischen Objekten in Karten machen Biber und Duckett in (Biber und Duckett, 2005) den Vorschlag mit mehreren räumlich und zeitlich lokalen Karten zu arbeiten. Ausgehend von einer initialen statischen Karte werden bei Änderungen an der Umgebung neue lokale Karten angelegt, die mit der Zeit in das „Langzeitgedächtnis“ übergehen. Dort werden eine Reihe von Karten unterschiedlichen Alters gespeichert und mittels folgender, passender Beobachtungen verbessert. Für die Lokalisierung innerhalb der Umgebung wird immer die Variante der lokalen Karte genommen, die die aktuelle Messung am besten erklärt. Vorteile des Verfahrens sind, dass keine Interpolation zwischen Zuständen nötig ist, was unter Umständen zu Werten führen konnte, die nie tatsächlich beobachtet wurden und dass vorübergehende Änderungen wie z.B. abgestellte Kartons in einem Gang nur geringen Einfluss haben und nach der Rückkehr zur vorherigen Situation nahtlos mit den alten, etablierten Karten gearbeitet werden kann.

## 2.4 Exploration

Um autonom eine fremde Umgebung erkunden zu können, muss ein Roboter zwangsläufig über eine Karte verfügen die neben freiem und belegtem Raum auch unbekannte Bereiche explizit abbildet. Für die Exploration relevante sind dabei alle Stellen an denen freier in unbekanntem Raum übergeht, da hier das Potential für weitere Erkenntnisse gegeben ist. Yamauchi nennt dies in (Yamauchi, 1997) einen *Frontier-Based* Ansatz. In einer 2D Gridmap werden diese Grenzen zwischen freien und unbekanntem Bereichen gesucht und nach zusammenhängender Größe gefiltert, so dass nur Grenzgebiete erhalten bleiben, die signifikanten Informationsgewinn versprechen. In (Fournier u. a., 2007) wird dieses Verfahren erfolgreich auf eine „2,5D“ Quadmap angewendet, die aus dem eigentlichen erstellten 3D Octree erzeugt wird. Zur Bewertung der möglichen Explorationspunkte werden hier mithilfe eines Sensormodells und Raytracing-Techniken für jeden Kandidaten überprüft, wieviel unbekannter Raum durch einen Scan erschlossen werden kann. Zusätzlich wird noch die Entfernung vom aktuellen Aufenthaltsort sowie die Lage in Bezug zur Fahrtrichtung in die Planung mit einbezogen.

Klassische Frontier-Based Verfahren lassen sich nur schlecht auf echte 3D Umgebungen übertragen, da der Rechenaufwand um ein Vielfaches ansteigt und autonome, mobile Plattformen damit häufig überfordert wären. Vor wenigen Monaten wurde in (Shen u. a., 2011a)

---

<sup>9</sup><http://sketchup.google.com/3dwarehouse/>

eine Lösung dieses Problems vorgeschlagen, die auf die Simulation molekular-dynamischer Prozesse zurückgeht. Es werden eine Reihe von Partikeln ausgesandt, die bestimmten Anziehungskräften ausgesetzt sind und solange von Wänden und anderen bekannten Strukturen reflektiert werden, bis sie auf unbekannte Bereiche treffen. Leider ist die finale Publikation noch nicht verfügbar, jedoch zeigen die veröffentlichten Experimente sehr gute Ergebnisse in der Exploration (Abbildung 5). Die orangen Sphären zeigen Punkte, an denen die Partikel auf unbekannte Bereiche gestoßen sind. Zusätzlich wird noch eine Blickrichtung festgelegt, da der verwendete Tiefenbildsensor nur ein relativ geringes Sichtfeld besitzt. Der zweite Teil der Abbildung zeigt den Zustand nachdem die Punkte der ersten Hälfte abgearbeitet wurden.

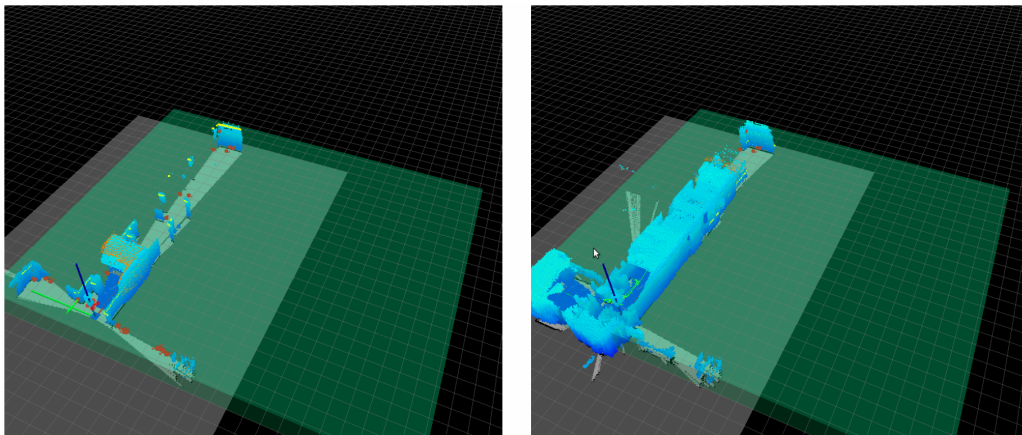


Abbildung 5: Zielpunkte für die autonome Exploration, (Shen u. a., 2011a)

### 3 Fazit und Abgrenzung

Die Robotik generell ist eines der aktivsten und facettenreichsten Forschungsgebiete überhaupt und hat durch immer günstigere Hardware, Verfahren zum rapid Prototyping und stromsparendere sowie leistungsfähigere mobile Hardware in den letzten Jahren noch einmal deutlich zugelegt. Diese Vielfalt ist natürlich sehr zu begrüßen, die stark unterschiedlichen Einsatzszenarien führen jedoch sehr häufig zu stark angepassten Speziallösungen, die sich teilweise nur schwer generalisieren und auf andere Gebiete übertragen lassen. Dennoch gibt es gerade im 3D-Bereich bemerkenswerte Fortschritte zu verzeichnen. Mit der Microsoft KINECT ist seit Ende 2010 ein sehr günstiger Sensor auf den Markt gekommen, der in vielen Bereichen deutliche Vorteile gegenüber vergleichbaren Techniken bietet. Die KINECT hat mit 640x480 Pixeln eine beinahe 16-fach höhere Auflösung als bisher verfügbare Time-of-Flight Kameras und kostet dabei nur einen Bruchteil. Gegenüber klassischer Stereoskopie mit kalibrierten Kameras zeichnet sie sich durch die aktive Projektion des Sensormusters aus, was es ermöglicht

auch unstrukturierte Oberflächen zuverlässig zu vermessen und bei schlechter oder sogar komplett ohne Umgebungsbeleuchtung zu funktionieren. Schließlich bietet sie im Vergleich zu schwenkbaren Laserscannern den Vorteil bis zu 30 Bilder/s zu liefern und somit Stop-and-Go Scannen überflüssig macht. Nachteile sind dagegen die Verhältnismäßig geringe Reichweite von maximal 10 Metern bei stark eingeschränkter Präzision oder ca. 3,5m bei etwa 2cm Tiefenauflösung, sowie der relativ kleine Sichtbereich mit diagonal ca. 60°.

Die große Mehrzahl der vorgestellten Arbeiten und Verfahren sind mit dem hohen Datenaufkommen der KINECT schlicht überfordert und müssen daher an die geänderten Rahmenbedingung angepasst und für diesen Einsatz optimiert werden. Am ehesten mit dem geplanten Vorhaben vergleichbar sind die Arbeiten von Henry et al. (Henry u. a., 2010) sowie Engelhard et al. (Engelhard u. a., 2011). In beiden Fällen wurde die Kamera jedoch von einer Person durch die Räumlichkeiten getragen und keinerlei Aufwand hinsichtlich Navigation und Exploration getrieben. Auch werden keine volumetrischen Karten erzeugt, sondern im Fall der ersten Arbeit *Surfels*<sup>10</sup> verwendet, während sich die zweite Arbeit auf farbige Punktwolken beschränkt. Beides ist keine geeignete Grundlage für autonome Navigation und Exploration. In dieser Hinsicht sind die Ansätze in den beiden Arbeiten von Shen et al. (Shen u. a., 2011a,b) besser geeignet. Es kommt hier jedoch zusätzlich noch ein 360° Laserscanner zum Einsatz und auf Grund der präsentierten Ergebnisse ist fraglich, ob das vorgestellte Explorationsverfahren auch in komplexeren und detaillierten Umgebungen die Anforderungen der geplanten Arbeit an die Präzision und Vollständigkeit der Karte erfüllen kann. Ein weiterer Punkt der bisher kaum angegangen wurde ist der Umgang mit nicht statischen 3D-Karten. Erste vielversprechende Ansätze für 2D Darstellungen wurden vorgestellt, es gilt jedoch noch zu evaluieren, ob sich diese auch auf dreidimensionale Anwendung übertragen lassen.

## 4 Ausblick

Die Planung für die weitere Arbeit sieht vor zunächst ein robustes Kartierungsverfahren auf Basis der KINECT und der Odometrie sowie evtl. des Laserscanners zu entwickeln und diese Karte für Navigationsaufgaben zu verwenden. Hierzu möchte ich zunächst auf *Octomap* zurückgreifen um Praxiserfahrung mit volumetrischen Karten zu sammeln. Darauf aufbauend sollen Verfahren zur autonomen Exploration implementiert und untersucht, sowie eine mögliche Erweiterung von *Octomap* um dynamische Karten nach Vorbild von (Biber und Duckett, 2005) überprüft und getestet werden.

---

<sup>10</sup>Surfels sind zweidimensionale Flächen, deren Position im Raum, Ausrichtung, Größe und Farbe bekannt sind

## Literatur

- [Bay u. a. 2006] BAY, Herbert ; TUYTELAARS, Tinne ; VAN GOOL, L.: SURF: Speeded Up Robust Features. In: *9th European Conference on Computer Vision*. Graz Austria, May 2006
- [Besl und McKay 1992] BESL, P.J. ; MCKAY, H.D.: A method for registration of 3-D shapes. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 14 (1992), feb, Nr. 2, S. 239 –256. – ISSN 0162-8828
- [Biber und Duckett 2005] BIBER, Peter ; DUCKETT, Tom: Dynamic Maps for Long-Term Operation of Mobile Service Robots. In: *Robotics: Science and Systems*, 2005, S. 17–24
- [Engelhard u. a. 2011] ENGELHARD, Nikolas ; ENDRES, Felix ; HESS, Juergen ; STURM, Juergen ; BURGARD, Wolfram: Real-time 3D visual SLAM with a hand-held RGB-D camera. In: *Proc. of the RGB-D Workshop on 3D Perception in Robotics at the European Robotics Forum, Vasteras, Sweden, 2011.*, 2011
- [Fournier u. a. 2007] FOURNIER, J. ; RICARD, B. ; LAURENDEAU, D.: Mapping and Exploration of Complex Environments Using Persistent 3D Model. In: *Computer and Robot Vision, 2007. CRV '07. Fourth Canadian Conference on*, may 2007, S. 403 –410
- [Grisetti u. a. 2007] GRISSETTI, G. ; STACHNISS, C. ; BURGARD, W.: Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters. In: *Robotics, IEEE Transactions on* 23 (2007), feb., Nr. 1, S. 34 –46. – ISSN 1552-3098
- [Henry u. a. 2010] HENRY, Peter ; KRAININ, Michael ; HERBST, Evan ; REN, Xiaofeng ; FOX, Dieter: *RGB-D Mapping: Using Depth Cameras for Dense 3D Modeling of Indoor Environments*. 2010. – URL [http://www.cs.washington.edu/ai/Mobile\\_Robotics/postscripts/3d-mapping-iser-10-final.pdf](http://www.cs.washington.edu/ai/Mobile_Robotics/postscripts/3d-mapping-iser-10-final.pdf)
- [Lowe 1999] LOWE, David G.: Object recognition from local scale-invariant features, 1999, S. 1150–1157
- [Montemerlo und Thrun 2003] MONTEMERLO, M. ; THRUN, S.: Simultaneous localization and mapping with unknown data association using FastSLAM. In: *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on* Bd. 2, 2003, S. 1985 – 1991 vol.2. – ISSN 1050-4729
- [Roth-Tabak und Jain 1989] ROTH-TABAK, Y. ; JAIN, R.: Building an environment model using depth information. In: *Computer* 22 (1989), jun, Nr. 6, S. 85 –90. – ISSN 0018-9162
- [Rusinkiewicz und Levoy 2001] RUSINKIEWICZ, S. ; LEVOY, M.: Efficient variants of the ICP algorithm. In: *3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference on*, 2001, S. 145 –152

- [Rusu u. a. 2007] RUSU, R.B. ; BLOW, N. ; MARTON, Z. ; SOOS, A. ; BEETZ, M.: Towards 3D object maps for autonomous household robots. In: *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, 29 2007-nov. 2 2007, S. 3191 –3198
- [Segal u. a. 2009] SEGAL, A. ; HAEHNEL, D. ; THRUN, S.: Generalized-ICP. In: *Proceedings of Robotics: Science and Systems*, URL <http://www.roboticsproceedings.org/rss05/p21.pdf>, Juni 2009
- [Shen u. a. 2011a] SHEN, Shaojie ; MICHAEL, Nathan ; KUMAR, Vijay: 3D Indoor Exploration with a Computationally Constrained MAV. In: *Proc. of the RSS 2011 Workshop on 3D Exploration, Mapping, and Surveillance with Aerial Robots, 2011.*, 2011
- [Shen u. a. 2011b] SHEN, Shaojie ; MICHAEL, Nathan ; KUMAR, Vijay: Autonomous multi-floor indoor navigation with a computationally constrained MAV. In: *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, may 2011, S. 20 –25. – ISSN 1050-4729
- [Thrun und Leonard 2008] THRUN, Sebastian ; LEONARD, John J.: Simultaneous Localization and Mapping. In: SICILIANO, Bruno (Hrsg.) ; KHATIB, Oussama (Hrsg.): *Springer Handbook of Robotics*. Springer Berlin Heidelberg, 2008, S. 871–889. – URL [http://dx.doi.org/10.1007/978-3-540-30301-5\\_38](http://dx.doi.org/10.1007/978-3-540-30301-5_38). – 10.1007/978-3-540-30301-5\_38. – ISBN 978-3-540-30301-5
- [Wurm u. a. 2010] WURM, K.M. ; HORNUNG, A. ; BENNEWITZ, M. ; STACHNISS, C. ; BURGARD, W.: OctoMap: A Probabilistic, Flexible, and Compact 3D Map Representation for Robotic Systems. In: *Proc. of the ICRA 2010 Workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation*. Anchorage, USA, Mai 2010. – URL <http://ais.informatik.uni-freiburg.de/publications/papers/wurm10octomap.pdf>
- [Yamauchi 1997] YAMAUCHI, B.: A frontier-based approach for autonomous exploration. In: *Computational Intelligence in Robotics and Automation, 1997. CIRA'97., Proceedings., 1997 IEEE International Symposium on*, jul 1997, S. 146 –151