Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Masterarbeit

Ben Struss

Freihand 3D-Kartierung merkmalsarmer Indoor-Umgebungen

Fakultät Technik und Informatik
Department Informatik

Faculty of Engineering and Computer Science
Department of Computer Science

Ben Struss

Freihand 3D-Kartierung merkmalsarmer Indoor-Umgebungen

Masterarbeit eingereicht im Rahmen der Masterarbeitprüfung
im Studiengang Master Informatik
am Department Informatik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr.-Ing. Andreas Meisel
Zweitgutachter: Prof. Dr. Wolfgang Fohl

Abgegeben am 29. Oktober 2015

Ben Struss

Thema der Masterarbeit

Freihand 3D-Kartierung merkmalsarmer Indoor-Umgebungen

Stichworte

3D-Kartierung, Kinect v2, merkmalsarme Umgebungen, projizierte Merkmale

Kurzzusammenfassung

In dieser Arbeit wird ein Verfahren vorgestellt, das die 3D-Kartierung von merkmalsarmen Innenräumen ermöglicht. Es werden einfache projizierte Merkmale verwendet, um eine stabile Positionsbestimmung einer handgeführten Kinect v2 im Raum zu ermöglichen. Hierzu wurden verschiedene Merkmalsdetektoren untersucht und Techniken zur Vorfilterung entwickelt. Das Problem der Korrespondenzbestimmung wird mittels einer neuartigen Indexstruktur auf Basis von geometrischen Eigenschaften der Punktnachbarschaften gelöst.

Ben Struss

Title of the paper

Hand-held 3D mapping of featureless indoor environments

Keywords

3d mapping, Kinect v2, featureless environments, projected features

Abstract

In this masters thesis a technique for 3d mapping of featureless indoor environments is presented. It is based on simple projected features, which are tracked by a hand-held Kinect v2 sensor to produce a stable estimate of the current position. Different feature-detectors were tested and pre-filters have been developed and implemented. The correspondence-problem is being solved by using a novel indexing structure based on the geometric properties of a features neighbourhood.

Inhaltsverzeichnis

1. Einleitung

Virtuelle Welten sind ein Thema, das die Menschen schon lange fasziniert. Seit moderne Computer immer realistischere Grafiken darstellen können, sind sie eine Möglichkeit, der Realität vorübergehend zu entfliehen und in imaginäre oder auch weit entfernte Welten einzutauchen. Wir stehen derzeit kurz vor der breiten Verfügbarkeit von 3D Brillen, die durch die freie Kopfbewegung eine viel tiefere Immersion erlauben, als es auf normalen Bildschirmen möglich wäre. Die Samsung Gear VR ist bereits auf dem Markt und die Rift von Oculus sowie die HTC Vive werden in Kürze erscheinen und dem Thema *Virtual Reality* einen starken Schub nach vorne verschaffen.

Abseits der Spielindustrie werden Angebote für virtuelle Rundgänge bereits von einer ganzen Reihe Museen und vergleichbaren Einrichtungen angeboten, und auch die Immobilienbranche erkennt zunehmend das Potential, das das Angebot einer virtuellen Wohnungsbesichtigung bieten kann. Bisher bestehen solche Touren jedoch fast ausschließlich aus Aneinanderreihung mehrerer 360° Panoramen, zwischen denen hin- und zurückgesprungen werden kann. Eine wirklich freie Bewegung des Besuchers ist daher nicht möglich und man ist vollständig von den gewählten Perspektiven des Fotografen abhängig.

Um dem Anwender diese Freiheit gewähren zu können, müssten vollständige 3D-Modelle der Umgebungen angefertigt werden. Dies ist jedoch trotz fortschreitender Entwicklung in diesem Bereich immer noch mit hohen Kosten und/oder Zeiteinsatz verbunden. Den Bedarf an teurer Sensorik und speziell geschultem Personal gilt es deutlich zu senken, wenn die Technik in naher Zukunft auch den Weg in den breiten Markt finden soll.

1.1. Ziel der Arbeit

Im Rahmen dieser Arbeit sollen die Grundlagen für ein 3D-Kartierungs-Verfahren von Innenräumen entwickelt und untersucht werden. Das System sollte von einer einzelnen Person ohne spezielle Vorbildung bedienbar sein und ohne besondere Kalibrierung vor Ort auskommen. Es muss auch in Rohbauten und Räumen funktionieren, die nur wenige bis keine Anhaltspunkte an Böden, Wänden oder Decken aufweisen. Die daraus entstehenden 3D-Modelle sollten in sich konsistent und möglichst direkt im korrekten metrischen Maßstab sein.

1.2. Gliederung

Im folgenden Kapitel 2 wird zunächst eine grundlegende Einführung in die 3D-Kartierung, gebräuchliche Sensorik und etablierten Verfahren gegeben und das Problemfeld der merkmalsarmen Umgebungen erläutert. Darauf aufbauend, werden in Kapitel 3 die Zielsetzung der Arbeit und die Anforderungen präzisiert und das entworfene Verfahren ausführlich diskutiert. Anschließend folgt noch im gleichen Kapitel die Vorstellung des entwickelten Software-Frameworks und ein Überblick über die verwendeten Bibliotheken.

Die experimentelle Bewertung der Verfahrensbestandteile beginnt in Kapitel 4 mit der Erkennung von Merkmalen. Dort werden verschiedene Detektoren vorgestellt und im Hinblick auf Stabilität, Geschwindigkeit und Präzision bewertet. Darauf folgend befasst sich Kapitel 5 mit dem Problem der Feststellung von Punkt-Korrespondenz. Es werden die implementierten und neu entwickelten Verfahren vorgestellt und ebenfalls hinsichtlich Stabilität, Geschwindigkeit und Konsistenz untersucht. Abschließend wird in Kapitel 6 ein Fazit aus dem Erreichten gezogen und ein Ausblick auf mögliche weitere Bereiche der Entwicklung gegeben.

Die Dokumentation der Projektionsvorrichtung findet sich im Anhang A. Dort werden die verwendeten Teile, die Konstruktion und Funktionsweise kurz dargestellt.

2. Grundlagen

Dieses Kapitel beschreibt zunächst die Grundlagen der 3D-Kartierung. Es werden unterschiedliche Verfahren mit ihren speziellen Anwendungsbereichen vorgestellt und ein Überblick über die dazu verwendete Sensorik gegeben. Im zweiten Teil werden die Besonderheiten von merkmalsarmen Umgebungen dargelegt und Ansätze diskutiert, wie damit umgegangen werden kann.

2.1. 3D-Kartierung

Kartierung allgemein beschreibt den Vorgang, aus Beobachtungen ein möglichst präzises Abbild der realen Umwelt zu erschaffen. Während es bei zweidimensionalen Karten zumeist ausreicht, die Eckpunkte und Kantenformen von Objekten zu erfassen, werden für eine echte 3D Darstellung sehr viel mehr Daten benötigt. Diese bestehen in der Rohform aus vielen einzelnen Punkten, deren Position auf den Oberflächen bestimmt werden. Um eine möglichst vollständige Karte eines begrenzten Umfeldes zu erhalten ist es dabei fast immer nötig, diese aus mehreren Einzelmessungen zusammenzusetzen. Eine so entstehende Karte sollte in sich konsistent und maßstabsgetreu sein. Der genaue Maßstab an sich ist zum Zeitpunkt der Erfassung jedoch irrelevant, da sich dieser nachträglich durch Skalierung beliebig festlegen lässt.

Die größte Herausforderung bei der Erstellung von Karten aus mehreren Messungen ist die präzise Ermittlung der jeweiligen Sensorposition, so dass die Aufnahmen fehlerfrei zu einem Gesamtbild zusammengesetzt werden können. Aus diesem Umstand ergeben sich zwei grundlegende Kategorien der Kartierung: Die mit externer Lokalisierung und die ohne. In ersterem Fall wird die Lage und Position des Sensors durch zusätzliche Hilfsmittel relativ zu einem oder mehreren ortsfesten Bezugspunkten in der Umgebung erfasst. Dies findet beispielsweise in der Präzisionsvermessung großvolumiger Bauteile, wo eine Messsonde von einem stationären Lasertracker erfasst wird und bis auf wenige Mikrometer genaue Messungen erlaubt.[1] Aber auch klassisch schwenkbare Laserscanner auf einem Stativ, die mit einer präzisen Bestimmung des Drehwinkels arbeiten und ansonsten nicht räumlich verschoben werden, gehören zu dieser Kategorie. Sie werden häufig für die Architekturvermessung und zunehmend auch von Behörden zur Erfassung von Unfall- oder Tatorten eingesetzt.

Lässt der Anwendungsfall oder die Umgebung keine externe Lokalisierung zu, so muss die aktuelle Position anhand der bisher generierten Umgebungskarte ermittelt werden. Alle Ansätze, die nach diesem Prinzip funktionieren, werden daher unter dem Begriff *SLAM* für *Simultaneous Localization and Mapping* zusammengefasst. (Thrun und Leonard, 2008) Dies ist ein sehr aktiver Bereich der Forschung und es haben sich eine ganze Reihe von Verfahren

[1] Leica Laser Tracker System: http://www.leica-geosystems.de/de/

etabliert, von denen einige im übernächsten Abschnitt kurz vorgestellt werden. Da sich diese jedoch teilweise stark an den Fähigkeiten und Eigenschaften der Sensorik orientieren müssen, folgt zunächst eine Einführung in die gebräuchliche 3D Sensorik.

Sensorik

Sensoren, die zur Kartierung eingesetzt werden, messen immer direkt oder indirekt Entfernungen. Speziell im Bereich der 3D Erfassung haben sich in den letzten Jahrzehnten eine Reihe von aktiven oder passiven optischen Sensoren durchgesetzt, die alle gewisse Vor- und Nachteile mit sich bringen. Diese werden in Tabelle 1 kurz dargestellt.

Verfahren	Vorteile	Nachteile
Passive Verfahren		
Stereo-Vision	Hohe Frameraten, hohe Auflösungen, Genauigkeit und Reichweite fast beliebig skalierbar (Abhängig von Auflösung und Stereobasis), günstig	Rechenintensiv, keine Daten für einfarbige Flächen, benötigt ausreichende externe Beleuchtung
Aktive Verfahren		
Time of Flight Cameras	Hohe Framerate, gute Genauigkeit, kompakt	Niedrige Auflösung, kleiner Blickwinkel, begrenzte Reichweite, teuer
Schwenkbare Laserscanner	Hohe Genauigkeit, große Reichweite möglich, großes Sichtfeld	Sehr langsam (mehrere Sekunden/Scan), teuer
Rotierende Laserscanner[2]	Hohe Genauigkeit, große Reichweite möglich, 360 Grad Sichtfeld, hohe Scanfrequenzen	Relativ groß und schwer, nur wenige Höhenlinien, sehr teuer
Structured Light	Beinahe beliebige Genauigkeit[3], hohe Auflösung, verhältnismäßig günstig, Reichweite nur durch Projektor begrenzt	Relativ komplexer Aufbau, abhängig von Umgebungslichtbedingungen, je nach Verfahren mehrere Aufnahmen pro Messung nötig

Tabelle 1: Überblick über gebräuchliche 3D-Sensorik

Klassische, passive Stereo-Vision Verfahren mit zwei normalen Kameras lassen sich relativ einfach und kostengünstig realisieren, sind jedoch zwingend auf ausreichend Merkmale in der

[2] Beispiel: Velodyne HDL-32/64E (http://www.velodynelidar.com/products.html)
[3] Begrenzt durch die Wellenlänge des Lichtes, bis unter ein Mikrometer

Umgebung angewiesen. (Fisher und Konolige, 2008) Messwerte lassen sich nur für Punkte ermitteln, die einander in beiden Bildern sicher zugeordnet werden können. Für künstliche Indoor-Umgebungen mit vielen einfarbigen Fläche und wenig Struktur sind sie daher nur schlecht geeignet.

Laserscanner messen die Entfernung entweder über Laufzeiten (*Time-of-Flight*) eines Lichtpulses oder anhand des Phasenversatzes zwischen dem gesendeten und empfangenen Signal. Dies erfolgt immer nur für genau einen Punkt, auf den der Strahl gerade trifft. Um ein vollständigeres Bild der Umgebung zu erhalten, wird der Laserstrahl typischerweise über einen rotierenden Spiegel umgelenkt und so Messungen im Bereich von bis zu 360 Grad ermöglicht. Da sich die Punkte dabei immer noch alle in einer Ebene befinden, kann der Scanner für die 3D-Kartierung noch gekippt werden, um so zeilenweise auch verschiedene Höhen zu bestimmen. Auch wurden Scanner entwickelt, welche über bis zu 64 Strahlen einen vertikalen Winkel von etwa 30-40 Grad erfassen können. Generell finden Laserscanner sehr viel Verwendung im Bereich der Robotik und autonomen Autos, aber zum Beispiel auch in der präzisen Vermessung von Architektur.

Structured Light ist im Grunde eine Art aktives Stereo-Vision. Anstelle der zweiten Kamera wird ein Projektor eingesetzt, der ein bekanntes Muster auf die Szene projiziert. Je nach Entfernung zum Objekt führt dies zu einer Verschiebung des Musters, welche mit der normalen Kamera beobachtet wird. Aus dem Grad des Versatzes lässt sich dann mittels Triangulation die Distanz bestimmen. Der große Vorteil dieser Verfahren im Gegensatz zum passiven Ansatz ist, dass auf den zu erfassenden Objekten selber keinerlei erkennbare Strukturen vorhanden sein müssen. Auch lässt sich mit jeder Aufnahme ein relativ großer Ausschnitt erfassen. Beschränkender Faktor sind hierbei die Bildwinkel von Kamera und Projektor. Einer der bekanntesten Sensoren, der nach diesem Prinzip funktioniert, ist die Kinect von Microsoft. Im Jahr 2010 war diese der erste Sensor, der 3D Aufnahmen mit hoher Bildrate und relativ hoher Auflösung (320 x 240 Pixel im Tiefenbild) bei niedrigem Preis ermöglichte.

Time of Flight (ToF) Kameras basieren auf den gleichen Funktionsprinzipien wie Laserscanner, denn auch sie messen Laufzeiten oder Phasenverschiebungen von kontrolliert ausgesendetem Licht. Allerdings wird nicht nur ein Punkt gemessen, sondern ein zweidimensionaler optischer Sensor mit höherer Auflösung eingesetzt. Hierdurch kann mit jeder Aufnahme direkt einen größerer Bildausschnitt erfasst werden und es lassen sich Raten von 30-50 Bildern pro Sekunde erreichen. Dies führt jedoch auch zu einer deutlich geringeren Reichweite von nur etwa 5-10 m, da der komplette Bildbereich von der Lichtquelle ausgeleuchtet werden muss und nicht nur ein einzelner Punkt. Typischerweise kommen dabei sehr schmalbandig abstrahlende Infrarot Leuchtdioden oder Laser zum Einsatz. Durch entsprechende Filter vor der Kamera wird somit eine hohe Toleranz gegenüber Umgebungslicht erreicht.

Bis zum Ende des Jahres 2013 waren *ToF* Kameras mit Auflösungen von bis zu etwa

200 x 200 Pixeln zu Preisen von mehreren tausend Euro erhältlich.[4] Zusammen mit der Spielkonsole Xbox One wurde im November 2013 die zweite Version der Kinect von Microsoft herausgebracht, welche nun ebenfalls auf einen *ToF* Sensor setzt. Dieser Chip bietet eine deutlich gesteigerte Auflösung von 512 x 424 Pixeln bei 30 Bildern pro Sekunde und einer Reichweite von maximal 8 Meter. (Bamji u. a., 2015; Fankhauser u. a., 2015) Dabei kostet die gesamte Einheit aus Distanzsensor, Farbkamera (1920x1080 Pixel) und Mikrofonarray weniger als 200 Euro.

Solche Kombinationen von Farb- und Tiefenbildkamera werden auch als *RGB-D* (RGB + Depth) Sensoren bezeichnet. Diese haben besonders bei der mobilen Verwendung auf Robotern oder freihändig geführt Vorteile, da kombinierte Farb- und Tiefenwerte die Objekterkennung wesentlich unterstützen und zusätzliche Informationen in die so erstellten Karten einfließen können. Auch entfällt bei integrierten Einheiten die mitunter sehr aufwendige extrinsische Kalibrierung der Sensoren zueinander.

Verfahren

In den vergangenen Jahrzehnten wurde eine riesige Anzahl verschiedener Kartierungs-Verfahren entwickelt und vorgestellt. Viele davon mit dem Fokus auf sehr spezielle Umgebungs- und Sensorkonstellationen. Im folgenden Abschnitt werden drei Systeme beschrieben, deren Schwerpunkt auf der freihändigen Kartierung in Innenräumen liegt. Für einen größeren Überblick, insbesondere auch mit dem Fokus der Robotik-Anwendung, sei der Übersichtsartikel von (Thrun und Leonard, 2008) empfohlen.

In (Tomono, 2009) wird eine handgeführte Stereokamera verwendet. Auf beiden Bildern einer Aufnahme wird dabei jeweils der Canny-Kantendetektor (Canny, 1986) ausgeführt und anhand der gefundenen Kanten die Stereo-Korrespondenzen ermittelt. Zur Ermittlung der Kamerabewegung werden die so gewonnenen 3D Punkte nun mittels einer Variante des *Iterative Closest Point* (*ICP*) Verfahrens auf die Punkte der vorherigen Aufnahme abgebildet. Dieses Verfahren wurde erstmals in (Besl und McKay, 1992) vorgestellt und seitdem in vielen verschiedenen Varianten weiterentwickelt. (Segal u. a., 2009; Rusinkiewicz und Levoy, 2001) Mit seiner Hilfe kann die Transformation bestimmt werden, die eine gemessene Punktwolke möglichst gut in eine Referenzpunktwolke einpasst. In jeder Iteration der Verfahrens wird den Punkten der Messung jeweils ein korrespondierender Punkt in der Referenz zugeordnet (typischerweise der dichteste Punkte) und daraus die Transformation bestimmt, die die Distanzen zwischen den Korrespondenzen minimiert. Danach wird die Messung mit diesem Ergebnis transformiert und das Verfahren nähert sich einem Minimum. Damit es dabei auch zum globalen Minimum konvergiert, müssen die Punktwolken bereits zu Beginn grob korrekt zueinander ausgerichtet sein.

[4] Beispielsweise die SwissRanger SR4000 von Mesa Imaging oder der CamCube von PMD.

Im Falle von (Tomono, 2009) wird aufgrund der hohen Bildrate der Stereokamera davon ausgegangen, dass der Versatz zwischen zwei aufeinanderfolgenden Aufnahmen stets klein genug ist, so dass keine lokalen Minima getroffen werden. Um sich akkumulierende Fehler (*Drift*) zu vermeiden, die entstehen, wenn immer nur zum jeweils vorherigen Bild verglichen wird, werden in regelmäßigen Abständen Keyframes gesetzt, die jeweils bis zum nächsten Keyframe die Basis für die Positionskorrektur bilden. Mit den kombinierten Einzelbewegungen seit dem letzten Keyframe können die aktuellen Punkte bis zum Keyframe zurückgerechnet und an diesem ausgerichtet werden. Dies führt zu leicht erhöhtem Rechenaufwand, verbessert das Ergebnis jedoch erheblich.

Kurz nach Erscheinen des ersten Kinect Sensors wurde in (Newcombe u. a., 2011) das Verfahren *KinectFusion* vorgestellt. Dieses basiert auf dichten[5] Tiefenbildern von Time-of-Flight oder Structured Light Sensoren, welche mit hohen Bildraten erfasst werden. Das besondere an *KinectFusion* ist die sehr gute Abbildung von Oberflächen (Abbbildung 1). Dies wird durch die Integration jedes Tiefenbildes in eine globale Voxelstruktur ermöglicht. Jeder Voxel enthält dabei ein Gewicht und den Abstand zur nächsten Oberfläche. Die genaue Position der Oberfläche lässt sich dann mittels *Raycasting*[6] anhand des Nulldurchgangs entlang des Strahls ermitteln. (Voxel hinter der Oberfläche erhalten negative Distanzen.) Die Sensorbewegung zwischen zwei Aufnahmen wird ebenfalls mittels einer *ICP* Variante bestimmt. Dabei erfolgt der Vergleich jedoch nicht nur gegen die vorherige Aufnahme, sondern jederzeit gegen das komplette bisher erstellte Modell. Für verhältnismäßig kleine Szenen (1-3 m Kantenlänge), um die der Sensor herum bewegt werden kann, wird dadurch Drift beinahe vollständig vermieden. Diese aufwendige Verarbeitung kann dank hoch optimierter Algorithmen, die auf Grafikprozessoren (GPU) massiv parallel ausgeführt werden, mit bis zu 30 Bildern pro Sekunde betrieben werden. Die entstehende Karte kann (in Abhängigkeit vom vorhandenen Grafikspeicher) aus bis zu 512^3 Voxeln[7] bestehen. Im Falle eines Tracking-Verlustes durch zu schnelle Bewegung oder eines bildfüllenden Hindernisses besteht jedoch keine Möglichkeit eine erneute globale Lokalisierung[8] durchzuführen, so dass die Rekonstruktion nur fortgesetzt werden kann, wenn der Sensor ausreichend genau zur letzten erfolgreichen Messposition gebracht wird.

Zwei sehr ähnliche Verfahren, welche beide auf kombinierte *RGB-D* Sensoren setzen, wurden in (Endres u. a., 2012) und (Henry u. a., 2010) vorgestellt. Die Bestimmung der Sensorbewegung erfolgt hierbei in einem mehrstufigen Verfahren, welches sowohl Farb- als auch Tiefenbild verwendet. Zunächst werden aus den Farbbildern lokale Merkmale (SIFT (Lowe, 2004)) extrahiert und mit einer oder mehreren vorherigen Aufnahmen verglichen. Hieraus ergeben sich eine Reihe von Korrespondenzen, welche nun mittels des Tiefenbildes und der

[5] Englisch „*dense*“: Ein Bild/Matrix welches keine oder nur sehr wenige Lücken aufweist.

[6] Für jeden darzustellenden Pixel im Ausgabebild wird ein Strahl von der Kameraposition aus in die Karte „geschossen“, bis eine Fläche getroffen wird.

[7] Ein Voxel ist eine Einheit in einer 3D Grid-Struktur.

[8] Bei einer globalen Lokalisierung wird versucht die aktuelle Position in der Karte ohne eine initiale Schätzung zu bestimmen.

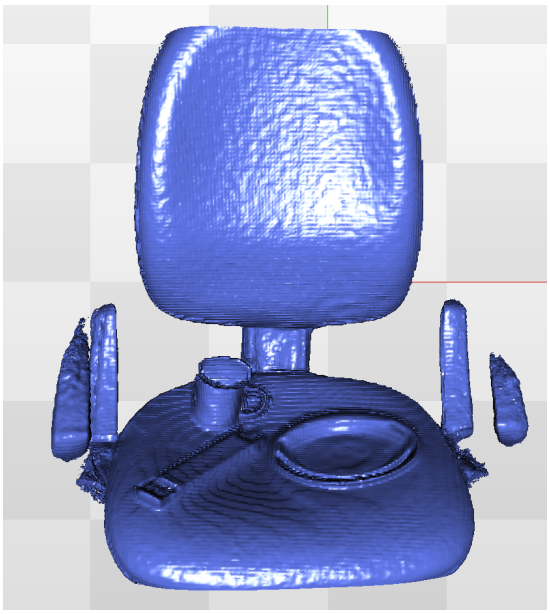

Abbildung 1: 3D Oberflächenrekonstruktion eines Stuhls mit KinectFusion

Sensor-Kalibrierung in 3D Koordinaten umgerechnet werden. Die Korrespondenzen werden nun im 3D Raum durch den *RANSAC*[9] Algorithmus gefiltert, um fehlerhafte Zuordnungen auszuschließen. Dabei werden jeweils drei Korrespondenzen auf Kompatibilität geprüft und eine Transformation bestimmt. Die Transformation, welche auf die restlichen Punkte angewendet die meisten passenden Korrespondenzen aufweist, wird dann als initiale Schätzung für einen folgenden *ICP* Schritt verwendet.

Durch die Verwendung von stark distinkten SIFT Merkmalen können auch größere Sprünge oder die erneute Beobachtung bereits zuvor kartierter Bereiche erkannt und behandelt werden. Wird ein sogenannter *Loop Closure*[10] erkannt, können nun die vorherigen Transformationen so optimiert werden, dass sich ein global konsistentes Modell ergibt. Gerade bei länger Anwendung in Innenräumen, unter Anwesenheit von Fluren und Durchgängen, erhöht sich so die Qualität der entstehenden Karte massiv.

[9] Random Sample Consensus

[10] Beschreibt den Zustand, wenn nach einem längeren Pfad, ein bereits bekannter Bereich aus einer anderen Richtung betreten wird.

Darstellungsformen

In Abhängigkeit vom Anwendungsfall sind verschiedene Darstellungsformen gebräuchlich. Die einfachste Form stellen rohe Punktwolken dar, in denen aus jedem Messwert direkt ein 3D Punkt erzeugt wird. Abhängig von Größe der Szene, Auflösung und Filterung der Daten können hier schnell mehrere Millionen Punkte zusammenkommen. Da sich diese Darstellung nur schlecht für die direkte Verwendung in der Praxis eignet, sind die Punktwolken zumeist nur ein Zwischenschritt zu einer kompakteren Darstellung.

Eine besonders für die bessere Darstellbarkeit und Ansicht durch Menschen verbreitete Technik ist die Rekonstruktion von Oberflächen aus den Punktwolken. Hierdurch wird Rauschen minimiert und Ausreißer eliminiert. Die Darstellung ist wesentlich sauberer und kleine Löcher in den Flächen werden geschlossen. Einen umfangreichen Überblick über verschiedene Techniken bietet (Berger u. a., 2014). Derartig generierte Oberflächen lassen sich sehr gut visualisieren, da die Kontur im Gegensatz zu reinen Punkten korrekt beleuchtet werden kann und somit wesentlich plastischer wirkt.

Soll die Karte für (teil-)autonome Roboter verwendet werden, so ist neben der Darstellung der vorhandenen Hindernisse vor allem auch die explizite Erfassung von freien Bereichen notwendig. Hierfür werden volumetrische Karten erzeugt, die den Raum in dreidimensionale Würfel aufteilen und für diese jeweils zwischen den Zuständen *frei*, *belegt* und *unbekannt* unterscheiden. Da dies bei großen Karten und feiner Auflösung zu sehr hohem Speicherbedarf führt, wurden effizientere, auf 3D Bäumen basierende, Verfahren vorgestellt. Als Beispiel sei hier OctoMap von (Wurm u. a., 2010) genannt (Abbildung 2).

2.2. Merkmalsarme Umgebungen

Um eine aus Rotation und Translation bestehende 3D Transformation zwischen zwei Koordinatensystemen zu bestimmen, werden mindestens drei Korrespondenzen zwischen den Systemen benötigt. Dies können Punkte sein, die in beiden Messungen eindeutig identifizierbar sind, oder auch Flächen oder andere hinreichend genau lokalisierte Objekte. Wichtig ist dabei, dass die Punkte nicht in einer Linie oder Flächen nicht parallel zueinander liegen. Eine *merkmalsarme Umgebung* liegt genau dann vor, sobald es Messungen gibt/geben kann, in denen sich diese Mindestzahl an Merkmalen nicht bestimmen lässt.

Dabei hängt es von der verwendeten Sensorik ab, ob eine Umgebung als merkmalsarm im Hinblick auf den jeweiligen Sensor einzustufen ist. Dies betrifft sowohl grundsätzliche Funktionsweisen, als auch quantitative Parameter wie Blickwinkel und Reichweite. Für Laserscanner oder *Time-of-Flight* Kameras ist eine glatte Wand frei von Merkmalen, auch wenn sie möglicherweise eine deutliche Texturierung aufweist. Im Gegenzug wird eine Stereokamera keine

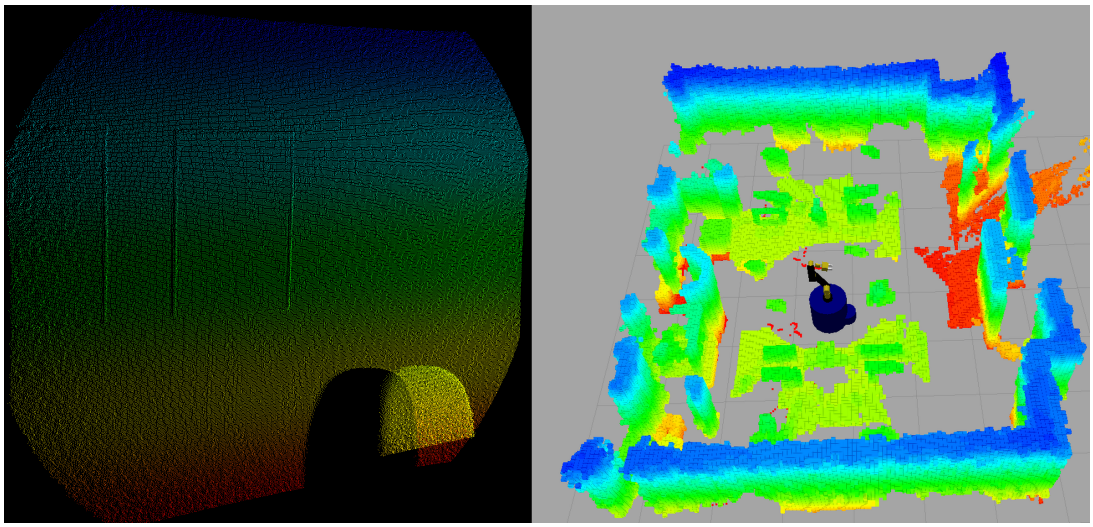

Abbildung 2: Links: Reine Punktwolke einer Stuhllehne vor einer Wand. Rechts: OctoMap eines Raumes. (Die Kantenlänge der Voxel beträgt 5 cm)

Anhaltspunkte auf einer geschwungenen einfarbigen Fläche erkennen. Abbildung 3 zeigt eine sowohl im Farb- als auch Tiefenbild merkmalsarme Wand.

Laserscanner haben häufig den Vorteil eines großen Blickwinkels von mindestens 180° und Reichweiten von 10-60 m. In Innenräumen sind daher fast immer ausreichend Wände und Ecken für eine robuste Positionsbestimmung vorhanden. Problematisch bleiben beispielsweise sehr lange Korridore, bei denen das Ende nicht mehr in Sensorreichweite liegt. Bei *Structured Light* oder *Time-of-Flight* Sensoren dagegen ist das Blickfeld deutlich kleiner und liegt eher in Bereichen zwischen 40° und 70° horizontal. Nimmt man einen leeren Raum an, der eine glatte Wand von 5 m Länge aufweist, so müssen selbst bei einem 70° Blickfeld etwa 3,6 m Abstand zu dieser Wand eingehalten werden, um zu jedem Zeitpunkt mindestens eine Raumecke im Bild zu haben. Bei den 60° der Kinect v1 erhöht sich dieser Abstand bereits auf mindestens 4,5 m und liegt damit an der Grenze, die für zuverlässige Messwerte angegeben ist.

Im Robotik-Umfeld wird in solchen Fällen häufig auf die Odometrie oder Beschleunigungssensoren zurückgegriffen, um die Position entlang der unsicheren Richtung möglichst gut abzuschätzen (beispielsweise in (Grisetti u. a., 2007; Hahnel u. a., 2003)). Bei handgeführten Sensoren stehen diese externen Sensoren zumeist nicht zur Verfügung. Um die neue Messung trotzdem möglichst genau verwenden zu können, kann wie in (Endres u. a., 2012) die Annahme getroffen werden, dass die Bewegung des Sensors konstant erfolgt. Bei hohen Bildraten und den daraus resultierenden kurzen Abständen zwischen den Aufnahmen, ist dies häufig eine gute Näherung der tatsächlichen Bewegung. Hierzu wird die Transformation zwischen der vorletzten und der letzten Messung extrahiert und daraus die aktuelle Position abgeleitet. Ge-

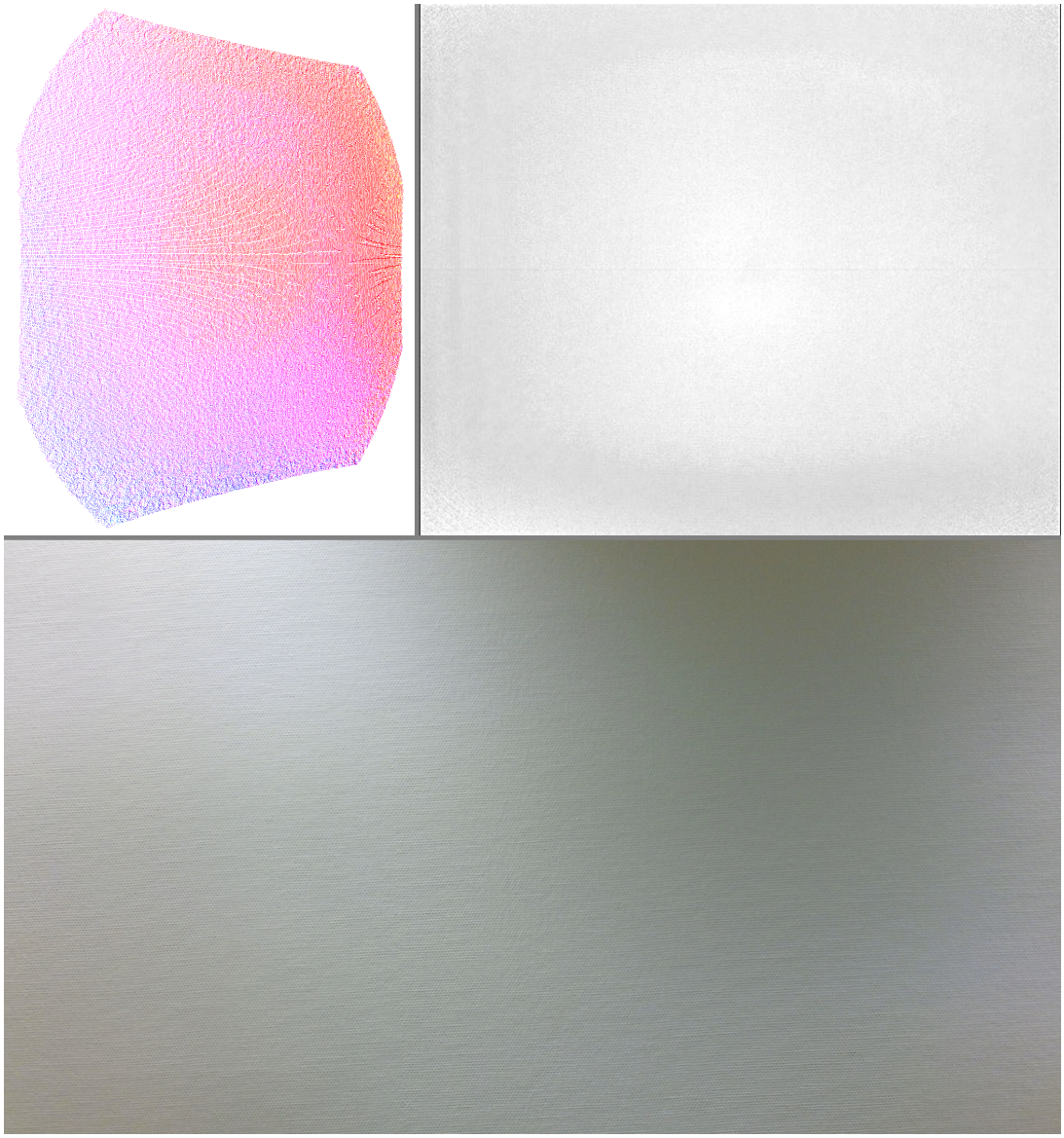

Abbildung 3: Eine glatte Wand ohne erkennbare Merkmale als Tiefenbild (Punktwolke), Infrarotbild und Farbbild.

nerell lassen sich hierdurch aber nur sehr kurze Aussetzer im Tracking überbrücken und die Genauigkeit der Karte leidet.

Ist bereits vor Beginn der Kartierung abzusehen, dass die Umgebung problematische Bereiche enthält, so können künstliche Merkmale an den entsprechenden Stellen platziert werden. In (Meyer-Delius u. a., 2011) werden spezielle reflektive Marker an Wänden angebracht, die während der Kartierung in den Intensitätswerten des Laserscanners erkannt werden. Durch geschickte Verteilung der Marker ist es dabei möglich die aktuelle Position mit hoher Wahrscheinlichkeit zu bestimmen. Für kamerabasierte Systeme wurden in (Schweiger u. a., 2009) spezielle Marker entwickelt, die zu bestmöglicher Erkennung durch *SIFT* oder *SURF* Detektoren führen. Diese können auf ansonsten merkmalsarme Flächen geklebt werden und somit die Ermittlung der Kamerapose stabilisieren.

3. Konzeptüberlegung

Im folgenden Kapitel werden zunächst die Rahmenbedingungen, die Zielsetzung und die Anforderungen an die zu entwickelnde Lösung definiert. Darauf aufbauend folgt die grobe Vorstellung des Ansatzes und eine Zerlegung in die zu realisierenden Teilschritte. Abschließend wird auf die dazu entwickelte Software mit den relevanten Komponenten und Bibliotheken eingegangen.

3.1. Zielsetzung

Die allgemeinen Anforderungen an das zu entwickelnde Verfahren wurden bereits in Kapitel 1.1 grob beschrieben. Im folgenden werden diese ergänzt, konkretisiert und in die beiden Kategorien *notwendig* und *optional* eingeteilt.

Notwendige Eigenschaften

- Merkmalsarme Innenräume müssen zuverlässig kartierbar sein. Die maximale Größe wird dabei nur durch den Aktionsradius und die Reichweite des verwendeten Sensors eingeschränkt.
- Die benötigte Hardware sollte nicht mehr als 500 bis 1000 Euro kosten.
- Für die Kartierung soll nicht mehr als eine Person benötigt werden.
- Der Sensor muss freihändig im Raum bewegt werden können.
- Pro Kartierung darf keine besondere Kalibrierung notwendig sein.

- Der Raum soll nicht besonders für die Kartierung verändert werden müssen.
- Die entstehende Karte muss *dicht* und vollständig sein.
- Die metrische Darstellung soll bis auf wenige Zentimeter genau und reproduzierbar sein.

Optionale Eigenschaften

- Der aktuelle Stand der Kartierung sollte sich live verfolgen lassen.
- Dem Benutzer sollten keine besonderen Vorgaben bezüglich des Ablaufes der Kartierung gemacht werden.
- Es sollte möglich sein, die Kartierung zu unterbrechen und später fortzusetzen.
- Das Ergebnis sollte außer der 3D Karte auch Farbinformationen beinhalten.

3.2. Vorgehen

Als Sensor wird die in Kapitel 2.1 bereits kurz vorgestellte Kinect v2 von Microsoft verwendet. Sie erfüllt die Anforderungen in Bezug auf freihändigen Einsatz, günstigen Preis, hinreichende Präzision, Erfassung dichter Punktwolken und die Integration einer kalibrierten Farbkamera. Als problematisch könnte sich die fehlende feste zeitliche Synchronisation zwischen Farb- und Tiefenbild sowie die Kontrolle über die Belichtungsparameter der Farbkamera erweisen. Bei zu wenig Umgebungslicht drosselt die Kinect v2 die Frequenz der Farbbilder auf 15 Bilder pro Sekunde.

Bedingt durch die zwei getrennten Kameras für Farb- und Tiefenbild werden im Umgang mit der Kinect v2 drei verschiedene Koordinatensysteme verwendet. Im Farbbild mit 1920 x 1080 Pixeln gibt es genau wie im Tiefenbild mit 512 x 424 Pixeln 2D Pixel-Koordinaten (x, y). Zusätzlich gibt es noch das metrische 3D Koordinatensystem, dessen Ursprung genau in der Infrarotkamera liegt.[11] Da sich diese Koordinaten jedoch aufgrund des horizontalen Versatzes und der unterschiedlichen intrinsischen Parameter von Farb- und Infrarotkamera nicht einfach linear ineinander umrechnen lassen, stellt der Microsoft Treiber einen *CoordinateMapper* für die Konversion zur Verfügung. Dieser ist abhängig vom derzeit angeschlossenen Sensor und wird mit der Werkskalibrierung initialisiert.[12]

Mit etwa 70° x 60° (horizontal/vertikal) ist das Blickfeld des Tiefenbildsensors eines der größten aller derzeit erhältlichen Time-of-Flight Sensoren. Unter der Annahme, dass bei der Kartierung

[11] Microsoft Kinect Coordinate mapping: https://msdn.microsoft.com/de-de/library/dn785530.aspx

[12] Bei der Verwendung von gespeicherten Daten muss daher unbedingt darauf geachtet werden, dass der selbe Sensor wieder angeschlossen wird, da sonst mit unpassenden Parametern gerechnet wird.

jederzeit ein Abstand von mindestens 1,5 m zu Wänden eingehalten werden kann, ergibt sich daraus ein sichtbarer Bildausschnitt von mindestens 2,1 x 1,7 m^2. Anhand dieser Größe kann nun bestimmt werden, ab wann eine Umgebung als merkmalsarm und somit nicht ohne Hilfsmittel präzise kartierbar eingestuft werden muss. Dies ist der Fall, sobald es mindestens einen entsprechenden Bildausschnitt gibt, in dem weniger als drei Merkmale eindeutig identifizierbar sind. Gerade in Innenräumen mit glatten, unstrukturierten Wänden gibt es sehr häufig Bereiche, auf die diese Bedingung zutrifft.

Erzeugung künstlicher Merkmale

Um den Raum präzise kartierbar zu machen, müssen also durch den Sensor erkennbare Merkmale an allen problematischen Stellen hinzugefügt werden. Bezogen auf die Kinect v2 könnten das sowohl räumliche als auch optische Marker sein. Die Verwendung optischer Marker ist dabei die bevorzugte Variante, da diese im Tiefenbild nicht (oder nur kaum) sichtbar sind und somit keinen Einfluss auf die entstehende 3D Karte nehmen. Doch auch die Anbringung von Markern, wie sie beispielsweise in (Schweiger u. a., 2009) vorgestellt wurde, ist mit erheblichem Aufwand verbunden. Bereits ein Raum mit 5 x 5 m^2 und 2,5 m Raumhöhe hat eine reine Wandfläche von 70 m^2 und würde daher mindestens etwa 20-30 optimal verteilte Marker benötigen.

Da dies im Widerspruch zu den Anforderungen steht und je nach Umgebung möglicherweise auch gar nicht umsetzbar ist, wird in dieser Arbeit ein Verfahren entwickelt und vorgestellt, welches mit einfachen projizierten Merkmalen eine stabile Kartierung einer ansonsten merkmalsarmen Umgebung ermöglicht. Als Marker werden Lichtpunkte verwendet, die sich leicht mit einer handelsüblichen Spiegelkugel und einem Spot-Scheinwerfer erzeugen lassen. Ein Beispiel der Projektion ist in Abbildung 4 dargestellt. Der Aufbau der Projektionsvorrichtung findet sich im Anhang A.1.

Abbildung 4: Projektion von Merkmalen im Flur und einer Raumecke

3.3. Kartierungsverfahren

Der generelle Ablauf der Kartierung entspricht im Wesentlichen den vergleichbaren RGB-D Kartierungsverfahren (Endres u. a., 2012; Henry u. a., 2010) aus Kapitel 2.1. Der schematische Aufbau wird in Abbildung 5 dargestellt. Im Folgenden werden die einzelnen Schritte vorgestellt und auf die Besonderheiten in Bezug auf die Verwendung der Kinect v2 und den Einsatz von nicht unterscheidbaren Merkmalen eingegangen.

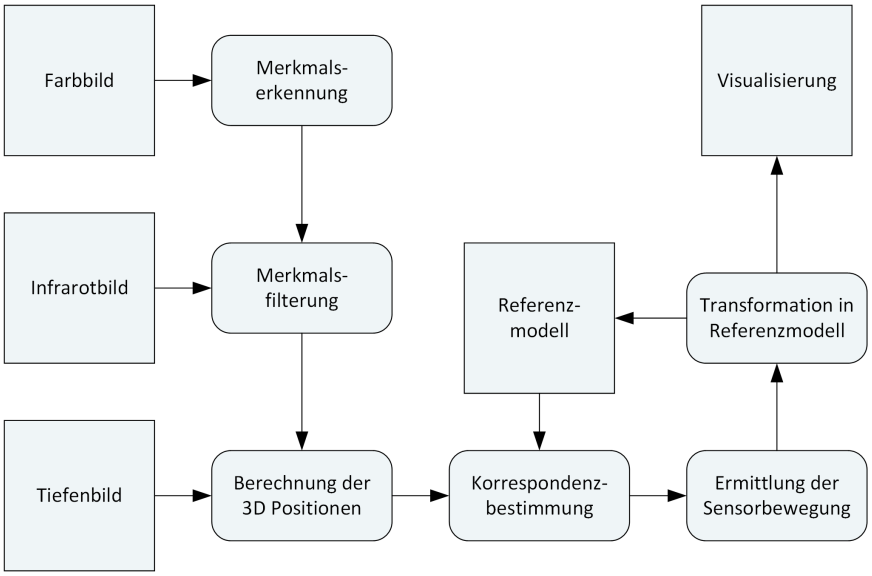

Abbildung 5: Schematischer Ablauf des Kartierungsverfahrens

Detektion der Merkmale

Im ersten Schritt geht es darum, die projizierten Merkmale in dem Farbbild zu erkennen. Hierzu werden mehrere Detektoren getestet und bezüglich ihrer Erkennungsrate, der Geschwindigkeit und Stabilität untersucht. Zusätzlich wird geprüft, ob sich durch eine Vorverarbeitung der Bilder das Verhalten weiter verbessern lässt. In diesem Schritt werden ausschließlich die 2D Positionen der Merkmale im Farbbild erfasst und keine weiteren Deskriptoren extrahiert. Da keine Anforderungen an die Beschaffenheit der Merkmale oder des Untergrunds definiert werden, muss davon ausgegangen werden, dass sich die künstlichen Merkmale optisch nicht voneinander unterscheiden lassen (siehe Abbildung 6).

Abbildung 6: Zwei optisch nicht unterscheidbare projizierte Merkmale

Filterung der Merkmale

Gute Merkmale für die Kartierung müssen stabil sein. Das heißt, sie sollten aus möglichst jedem Blickwinkel an der gleichen Stelle gefunden werden und die Messwerte an der Stelle sollten nicht stark variieren. Dies betrifft bei kombinierten RGB-D Sensoren vor allem Merkmale, die auf Kanten von Gegenständen gefunden werden, da hier bereits durch kleinen Versatz in der Umrechnung von Farb- zu Tiefenbild große Abweichungen in der resultierenden 3D Koordinate entstehen können. Außerdem ist es für Ansätze, die mit nicht unterscheidbaren Merkmalen arbeiten, wichtig, dass nicht zu viele Punkte dicht beieinander auftreten, da dies die eindeutige Zuordnung erschwert oder gar unmöglich macht. Daher werden die in Schritt 1 gefundenen Merkmale in mehreren Stufen gefiltert.

Dabei wird sich zunächst eine besondere Eigenschaft des Kinect v2 Infrarot-Bildes zunutze gemacht, um möglichst viele Punkte auszuschließen, die nicht durch die aktive Projektion entstehen. Die Kinect v2 unterdrückt im Infrarot-Bild jegliches Umgebungslicht, so dass eine Aufnahme entsteht, die nur durch die integrierte modulierte Lichtquelle beleuchtet ist. Die mit „Fremdlicht“ projizierten Merkmale sind im Infrarot-Bild daher nicht sichtbar. Um nun Merkmale auszuschließen, welche aus natürlichen Strukturen resultieren, wird auf das IR-Bild zunächst ein Canny-Kantendetektor angewendet und durch anschließende Dilatation und Erosion Lücken geschlossen und die Kanten vergrößert. Die einzelnen Zwischenergebnisse können der Abbildung 7 entnommen werden.

Das so entstandene Bild wird nun als Maske verwendet, indem die Positionen der Merkmale aus dem Farbbild ins Tiefen-/Infrarot-Bild umgerechnet und alle Merkmale entfernt werden, die in markierte Bereiche der Maske fallen. Um nun noch evtl. verbliebene mehrdeutige Merkmale zu entfernen, wird aus den 2D Positionen zunächst ein Index erzeugt und für jedes Merkmal der dichteste Nachbar bestimmt. Alle Merkmale, deren Distanz zum dichtesten Nachbarn einen

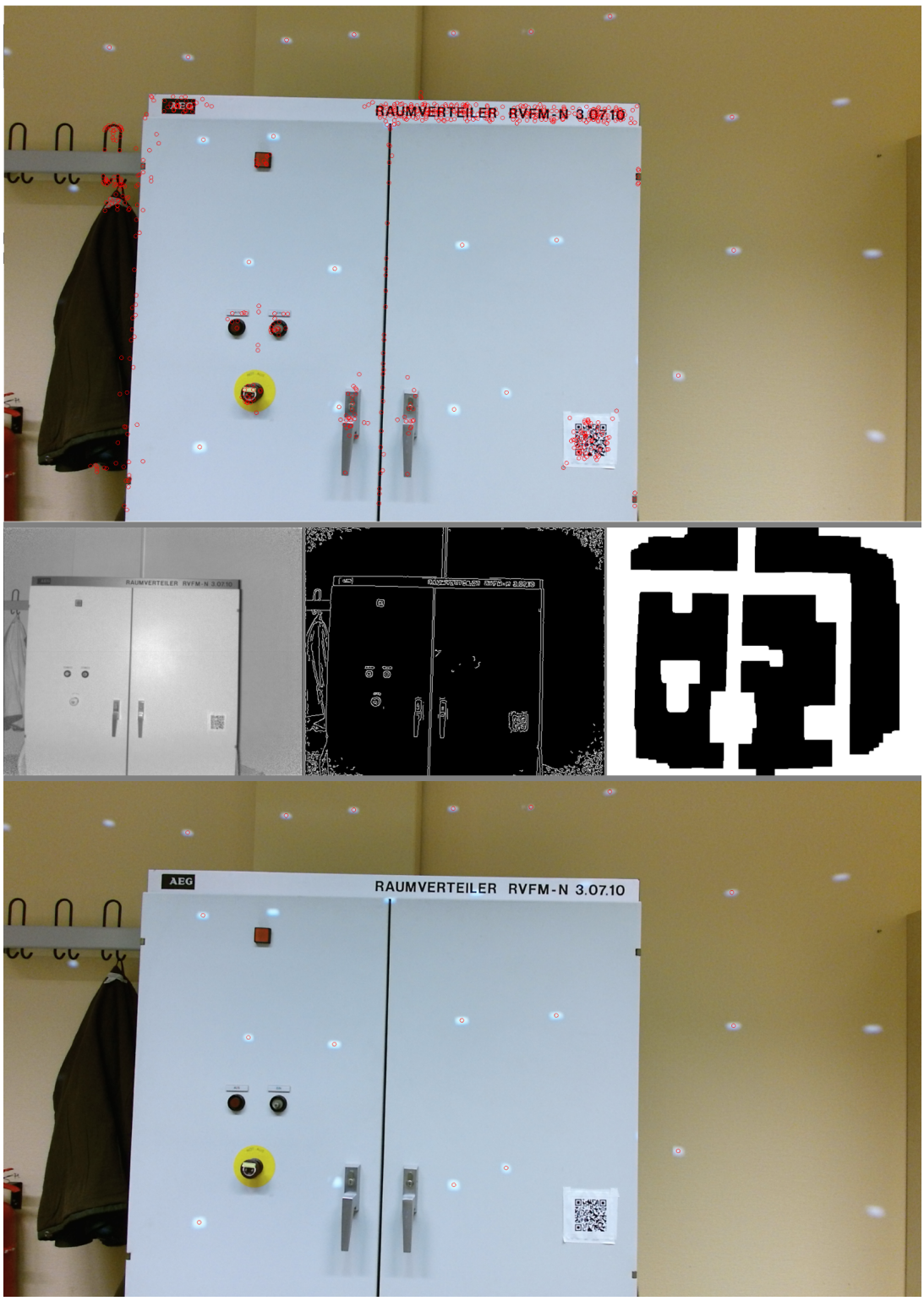

Abbildung 7: Oben: Ungefiltertes Bild mit allen Merkmalen. Mitte: IR-Bild ohne sichtbare Merkmale, nach Canny Detektor und nach Dilatation/Erosion. Unten: Gefilterte Merkmale nach Maskierung.

festgelegten Grenzwert in Pixeln unterschreiten, werden anschließend entfernt. Dieser Grenzwert ist abhängig von den Abständen der Punkte des projizierten Musters und der erwarteten Entfernung der Kamera bei der Aufnahme. Er sollte so hoch wie möglich gesetzt werden, so dass gerade noch keine gewünschten Merkmale verschwinden (Abbildung 8).

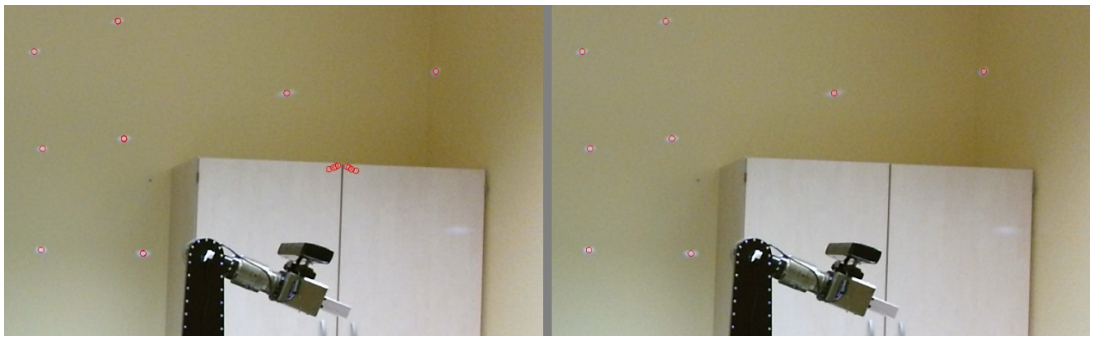

Abbildung 8: Links: Ungefilterte Aufnahme mit sporadisch auftretenden Merkmalen. Rechts: Gefilterte Aufnahme.

Bestimmung der Korrespondenzen

Im nächsten Schritt müssen für die verbliebenen Merkmale die korrespondierenden Punkte im Gesamtmodell gefunden werden. Da diese Zuordnung unabhängig vom Blickwinkel der Kamera sein muss, werden zunächst die 3D Koordinaten der Merkmale relativ zur aktuellen (noch unbekannten) Kameraposition berechnet. Die Referenzpunkte aus allen vorherigen Aufnahmen befinden sich in einem *Weltkoordinatensystem*, welches seinen Ursprung in der Sensorposition der ersten Aufnahme hat. Da die Merkmale für sich genommen nicht eindeutig identifizierbar sind können im Folgenden nur die geometrischen Information für die Zuordnung verwendet werden.

Für den einfacheren Fall wird angenommen, dass sich die Position des Sensors nur leicht verändert hat. Je langsamer die Führung per Hand und je höher die Bildrate der Kamera ist, desto wahrscheinlich ist es, dass dieser Fall zutrifft. Dabei würden auch die korrespondierenden Merkmale jeweils noch entsprechend dicht beieinander liegen. Da die Referenzpunkte jedoch nicht mehr im Koordinatensystem der vorherigen Aufnahme vorliegen, werden diese nun zunächst dorthin zurück transformiert. Für jedes aktuelle Merkmal wird dann in den transformierten Referenzpunkten nach dem dichtesten Nachbarn gesucht. Ist dieser nicht weiter entfernt als ein bestimmter Grenzwert, wird das Paar aus aktueller Merkmalsposition und Referenzpunkt als potentielle Korrespondenz aufgenommen.

Das Problem mit dem kontinuierlichen Tracking auf Basis der jeweils letzten bekannten Sensorposition ist jedoch, dass es mit jedem übersprungenen oder fehlerhaften Bild schwieriger wird,

die aktuelle Position zu bestimmen, da sich der Sensor immer weiter von der letzten Bezugsposition entfernt. So etwas kann beispielsweise durch eine zu schnelle Bewegung entstehen, bei der das Bild kurzzeitig verwischt oder wenn aus anderen Gründen eine Messung fehlerhaft ist und übersprungen wird. Wird eine solche Situation erkannt, in der die Zuordnung über die dichtesten Nachbarn fehlschlägt, muss eine sogenannte globale Lokalisierung durchgeführt werden, bei der die Position ohne vorherige Näherungsschätzung bestimmt wird.

Zu diesem Zweck wurde eine Indexstruktur entwickelt, welche Punkte über die Relationen zu ihren Nachbarn identifizierbar macht. Dabei werden für jede Merkmalsposition Informationen über die benachbarten Punkte gesammelt und im Index nach Punkten mit einer vergleichbaren Umgebungsstruktur gesucht. Bei hinreichender Übereinstimmung werden diese Korrespondenzen dann in die Kandidatenliste mit aufgenommen. Details des Verfahrens werden in Kapitel 5.3 beschrieben.

Ermittlung der Sensorbewegung

Zur Berechnung einer möglichen Bewegung (Transformation) des Sensors werden immer drei Korrespondenzen benötigt. Da bei den gefundenen Kandidatenpaaren auch Fehlzuordnungen enthalten sein können, werden nach dem *RANSAC* Verfahren aus allen Kandidaten jeweils drei Stück zufällig ausgewählt und untersucht. Seien die Punkte des aktuellen Bildes als Spaltenvektoren $\vec{a_1}, \vec{a_2}, \vec{a_3}$ und die korrespondierenden Referenzpunkte als $\vec{b_1}, \vec{b_2}, \vec{b_3}$ bezeichnet, so werden nun zunächst die Abweichungen der Distanzen zwischen je zwei Korrespondenzen bestimmt:

$$d_1 = ||\vec{a_1} - \vec{a_2}| - |\vec{b_1} - \vec{b_2}|| \quad (1)$$

$$d_2 = ||\vec{a_1} - \vec{a_3}| - |\vec{b_1} - \vec{b_3}|| \quad (2)$$

$$d_3 = ||\vec{a_2} - \vec{a_3}| - |\vec{b_2} - \vec{b_3}|| \quad (3)$$

Sobald mindestens eine dieser Abweichungen d_i größer als einige Zentimeter[13] ist, so kann davon ausgegangen werden, dass mindestens eine der Korrespondenzen fehlerhaft ist und die aktuelle Kombination verworfen werden kann. Die beiden, durch die gewählten Punkte beschriebenen, Dreiecke hätten dann nicht die gleichen Kantenlängen und ließen sich allein durch Translation und Rotation nicht aufeinander abbilden.

Sind die Abweichungen klein genug, so dass von einer validen Zuordnung ausgegangen werden kann, so werden nun die optimale Translation und Rotation bestimmt. Dies geschieht mit Hilfe der Singulärwertzerlegung (SVD) nach (Fisher und Konolige, 2008). Zunächst müssen

[13] Abhängig von der Genauigkeit des Sensors. Für die Kinect v2 sind 3-5 cm ein geeigneter Grenzwert.

jeweils die Durchschnittswerte der Punkte ermittelt werden, um den translationalen Anteil entfernen zu können:

$$\vec{\mu_a} = \frac{1}{3}(\vec{a_1} + \vec{a_2} + \vec{a_3}) \tag{4}$$

$$\vec{\mu_b} = \frac{1}{3}(\vec{b_1} + \vec{b_2} + \vec{b_3}) \tag{5}$$

Aus den bereinigten Punktvektoren wird nun eine Matrix für die Singulärwertzerlegung generiert, diese ausgeführt und die 3 x 3 Rotationsmatrix R bestimmt:

$$H = \sum_{i=1}^{3} (\vec{a_i} - \vec{\mu_a})(\vec{b_i} - \vec{\mu_b})^\top \tag{6}$$

$$[U, S, V] = SVD(H) \tag{7}$$

$$R = VU^\top \tag{8}$$

Dabei kann es einen Sonderfall geben, wenn alle Punkte annähernd in einer Ebene liegen. In diesem Fall enthält das Ergebnis nicht nur eine Rotation sondern auch eine Reflexion, die hier unerwünscht ist. Tritt dies ein, so ist die Determinante der Matrix R negativ. Als Korrektur kann die dritte Spalte von V mit -1 multipliziert und R neu berechnet werden. Abschließend wird noch der Translationsvektor $\vec{t}$ bestimmt:

$$\vec{t} = \vec{\mu_b} - R\vec{\mu_a} \tag{9}$$

Soll nun ein Punkt $\vec{a_i}$ aus der aktuellen Messung in das Referenzkoordinatensystem transformiert werden, wird zuerst die Rotation und anschließend die Translation angewendet:

$$a\vec{t}_i = R\vec{a_i} + \vec{t} \tag{10}$$

Anhand der bestimmten Transformation werden nun alle Merkmale des aktuellen Bildes in das Referenzkoordinatensystem umgerechnet und per Bestimmung der jeweils nächsten Nachbarn geprüft, wie passend die Transformation insgesamt ist. Dafür werden alle die Merkmale gezählt, deren nächster Nachbar nicht weiter als 5 cm entfernt ist. Nun wird die nächste zufällige

Dreierkombination aus Korrespondenzen getestet, bis entweder alle möglichen Kombinationen überprüft oder ein Schwellwert erreicht wurde.

Die Transformation, welche die höchste Anzahl übereinstimmender Merkmale hatte, wird nun gewählt und noch einmal verfeinert. Dazu wird für alle nun passenden Korrespondenzen noch einmal eine optimale Transformation bestimmt. In diesem Fall liefert die Singulärwertzerlegung eine Optimierung nach den niedrigsten quadratischen Distanzen zwischen den korrespondierenden Punkten. Letztlich wird die Transformation als gültig angesehen, wenn mindestens die Hälfte der Merkmale des aktuellen Bildes korrekt auf das Referenzmodell abgebildet werden konnten.

Integration der Messung und Darstellung

Liegt eine gültige Transformation für die aktuelle Messung vor, so werden nun die Merkmale in das Weltkoordinatensystem transformiert und in ein globales Modell eingefügt. Dieses enthält alle bisher beobachtete Merkmalspositionen und bildet die Grundlage für die globale Lokalisierung und bietet Stabilität, wenn bereits kartierte Bereiche erneut vermessen werden.

Für die live Visualisierung des aktuellen Standes werden aus dem Tiefenbild alle 8 x 8 Pixel ein Messwert ausgewählt, in 3D Koordinaten umgerechnet und in das Weltsystem transformiert, so dass eine Punktwolke mit 1/64 der Gesamtauflösung entsteht. Diese kann nun zusammen mit der aktuellen Transformation an ein externes Programm zur Darstellung gesendet werden.

Auf die Implementierung klassischer SLAM-Verfahren zur nachträglichen globalen Optimierung und die explizite Behandlung von *Loop Closures* wird in dieser Arbeit verzichtet, da diese Techniken bereits gut erforscht sind und eine Implementierung für die Erkenntnisse dieser Arbeit keinen Mehrwert bieten. Da jedoch alle Transformationen und Einzelaufnahmen zwischengespeichert werden und Techniken zur globalen Erkennung von Merkmalen entwickelt wurden, sind die grundlegenden Anforderungen für eine Erweiterung um solche Verfahren gegeben.

3.4. Software / Analyseframework

Umgesetzt wurde das vorgestellte Verfahren in einem Software-Framework, welches die ausführliche Untersuchung und experimentelle Analyse ermöglicht. Dafür mussten eine Reihe von Anforderungen erfüllt werden:

- Volle Integration der Kinect v2 mit allen bereitgestellten Bildformaten (Farbe, Tiefe und Infrarot)
- Abstrahierte Unterstützung beliebiger Merkmalsdetektoren
- Feingranulare Steuerung und Parametrisierung der einzelnen Kartierungsschritte
- Visualisierung der gefundenen und getrackten Merkmale
- Anzeige von Statistiken und Status der aktuellen Kartierung
- Möglichkeit, die (Roh-)Daten des Sensors zu speichern und solche abgespeicherten Sequenzen wieder abzuspielen
- Ausgabe der rekonstruierten Umgebungskarte
- Generierung ausführlicher statistischer Auswertungen des Kartierungsvorgangs

Das komplette Framework wurde in Microsoft .Net und C# umgesetzt und kann grob in sechs funktionelle Bereiche gegliedert werden, welche im Folgenden vorgestellt werden.

Frontend

Die Benutzeroberfläche (Abbildung 9) wurde in WPF[14] realisiert und definiert den Rahmen des Verfahrens. Auf der rechten Seite wird jeweils das aktuelle Live-Bild des Sensors (Farbe, Tiefe oder Infrarot wählbar) angezeigt, während sich am linken Rand die Einstellungen finden. Dort werden auch sämtliche Statistiken angezeigt und die Parameter des Trackings lassen sich anpassen. Da die Kinect v2 aufgrund ihrer Spielkonsolen-Herkunft alle Bilder horizontal gespiegelt ausgibt, wurde die Möglichkeit integriert, das dargestellte Bild erneut zu spiegeln. Da sämtliche Koordinaten und Umrechnungen zuvor auf den gespiegelten Daten basieren, kann dies nicht bereits bei der Erfassung geschehen.

[14] Windows Presentation Foundation

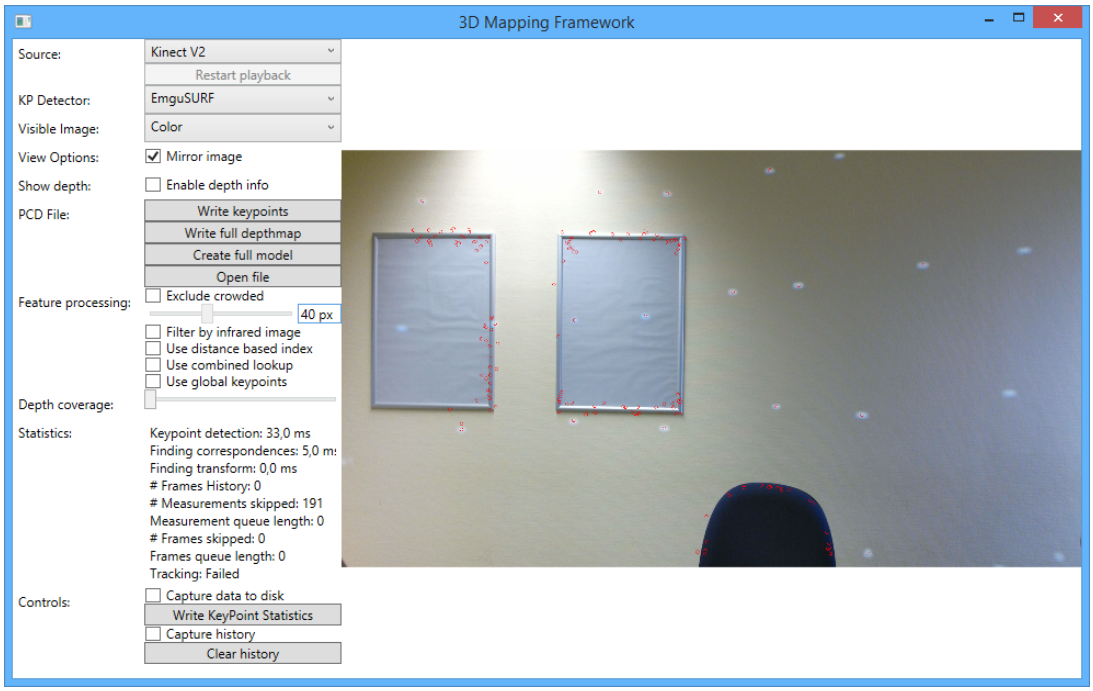

Abbildung 9: Das Frontend des 3D-Kartierungs Frameworks

Datenerfassung

Dieser Teil befasst sich mit der Abholung der Daten vom angeschlossenen Sensor und der Bereitstellung in einem einheitlichen Format zur Weiterverarbeitung. Entgegen der ursprünglichen Planung, auch die Datenquelle völlig modular zu gestalten, mussten im Zuge der Entwicklung einige besondere Aspekte der Kinect v2 berücksichtigt werden, so dass die zukünftige Integration anderer Sensoren zwar noch möglich, aber sehr komplex sein wird. Die abstrahierte Datenquelle bietet dabei insbesondere Informationen über die erfassten Daten und Methoden, um Koordinaten zwischen den verschiedenen Bezugssystemen (siehe Abschnitt 3.2) umzurechnen.

Als Quellen wurden die Kinect v2, basierend auf dem Microsoft *Kinect for Windows SDK*[15] in der Version 2.0, sowie ein *DiskReader* implementiert. Während es bei der direkten Kinect v2 Implementierung darum geht, die Rohdaten vom Treiber entgegen zu nehmen und in nutzbare Formate umzuwandeln, kann der *DiskReader* aufgezeichnete Messdaten mit konfigurierbarer Geschwindigkeit abspielen. Dies ist die Grundlage für reproduzierbare Messreihen mit unter-

[15] Verfügbar bei Microsoft unter: http://www.microsoft.com/en-us/download/details.aspx?id=44561

schiedlichen Konfigurationen. Die Ausgabe beider Quellen sind Messungs-Objekte, welche neben dem obligatorischen Farb-, Tiefen- und Infrarotbild auch noch einen Zeitstempel und Daten für die Koordinatenumrechnung enthalten. Diese werden für die weitere Verarbeitung in einen Puffer geschrieben. Um die Visualisierung und andere Abläufe nicht zu blockieren, läuft dieser Prozess in getrennten Threads.

Merkmalsextraktion und Filterung

Ebenfalls in einem eigenen Thread geschieht die Extraktion der Merkmale und die anschließende Filterung. Die Schnittstelle der Merkmalsdetektoren ist ebenfalls abstrahiert und so lassen sich einfach weitere Detektoren in das Framework integrieren. Die derzeit implementierten Detektoren basieren dabei auf der freien und quelloffenen Bildverarbeitungsbibliothek *OpenCV* (Bradski, 2000) und dem Wrapper *Emgu CV* (Huang, 2010), der die Bibliothek unter .Net verwendbar macht.

Auch der Canny-Kantendetektor für die Filterung der Punkte und die Methoden zur Dilatation und Erosion der Maske stammen aus der *OpenCV* Bibliothek. Des Weiteren wird die dortige Implementierung von *k-d-Bäumen*[16] für die Unterstützung aller Umkreis- und nächste Nachbarn-Suchen in der Merkmalsfilterung, sowie Korrespondenz- und Transformationsbestimmung verwendet.

Sind die Merkmale extrahiert und entsprechend der Einstellungen gefiltert, so werden die Positionen der Merkmale in das Farbbild eingezeichnet und das Messungs-Objekt zusammen mit der Merkmalsliste in den nächsten Puffer geschrieben.

Ermittlung der Korrespondenzen und Transformation

Im letzten Schritt wird in einem weiteren separaten Thread die Messung aus dem Puffer geholt und auf Basis der gefundenen Merkmale versucht, die Merkmalskorrespondenzen und die bestmögliche Transformation zu bestimmen. Für die Berechnung der Transformation mittels Singulärwertzerlegung, sowie alle anderen Vektor- und Matrixoperationen kommt die *Math.NET Numerics*[17] Bibliothek zum Einsatz. Wurde eine gültige Transformation ermittelt, so werden nun die Ergebnisse und Korrespondenzvektoren in das Farbbild eingezeichnet und das Ergebnis im Frontend dargestellt.

[16] k-dimensionale Suchbäume, die den Suchraum partitionieren und besonders für die Suche nächster Nachbarn geeignet sind

[17] http://numerics.mathdotnet.com/

Statistiken und Hilfsfunktionen

Dieser Teil beinhaltet alle Methoden, welche nicht ständig automatisch durchlaufen, sondern vom Anwender selber aufgerufen werden. Darunter fallen Funktionen wie die Ausgabe von einzelnen Punktwolken, die Generierung des Gesamtmodells oder das Erstellen von statistischen Auswertungen aus dem aktuellen Kartierungsvorgang. Auch hier wird im Wesentlichen die *Math.NET* Bibliothek für die anfallenden Koordinaten-Transformationen sowie für die Bestimmung statistischer Kenngrößen verwendet.

Live-Visualisierung

Zur Darstellung des aktuellen Fortschritts der Kartierung in 3D wurde eine externe Visualisierung auf Basis der *Unity 3D*[18] Engine entwickelt. Die Übermittlung der Daten aus dem Kartierungs-Framework erfolgt mittels einfacher UDP-Netzwerkkommunikation[19] und es können drei verschiedene Nachrichtentypen übertragen werden:

- eine Punktwolke, welche bereits in das globale Koordinatensystem transformiert wurde,
- eine Transformation, welche die aktuelle Kameraposition beschreibt,
- ein Reset-Befehl um die Karte zurückzusetzen.

Innerhalb der Visualisierung (Abbildung 10) wird die aktuelle Position durch einen Pfeil kenntlich gemacht und die Punkte entlang der y-Achse[20] mit einem Farbverlauf versehen, welcher sich einmal pro Meter wiederholt. Zusammen ermöglicht dies eine gute Einschätzung der Qualität der aktuellen Kartierung und gibt eine gute Übersicht über die räumlichen Verhältnisse.

[18] https://unity3d.com/, verwendet wurde die kostenfreie *Personal Edition*

[19] Verwendet wurde diese Bibliothek von Amir Hesami: http://www.codeproject.com/Tips/420551/Send-Receive-classes-through-UDP

[20] Wird der Sensor bei der ersten Aufnahme komplett waagerecht gehalten, entspricht die y-Achse genau der Höhe im Raum.

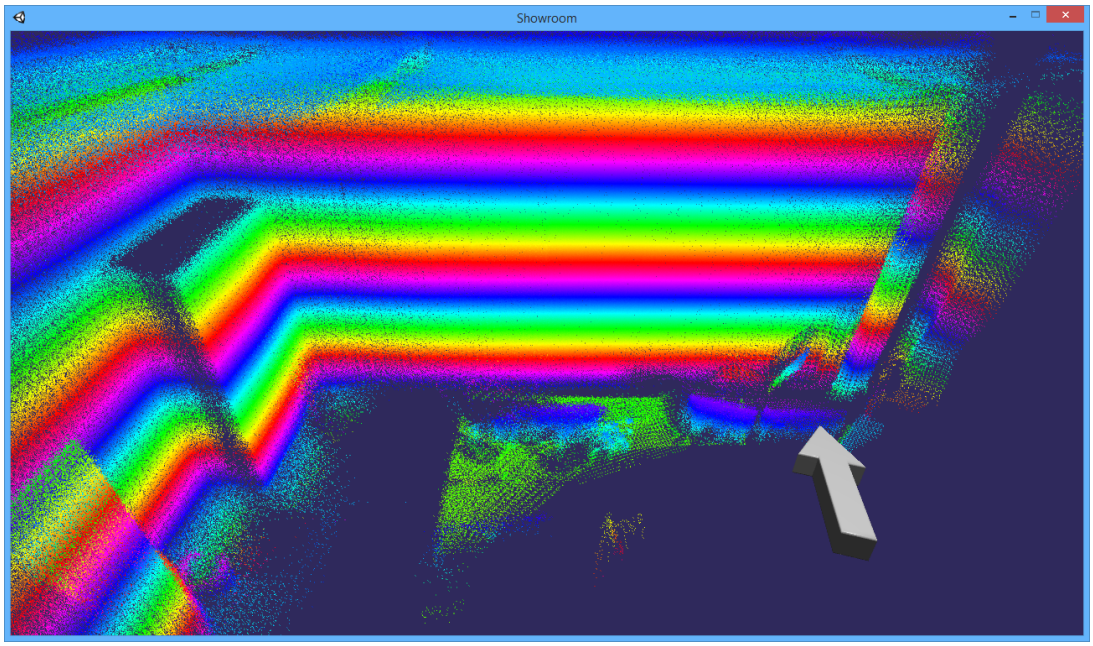

Abbildung 10: Darstellung der Live-Visualisierung

4. Merkmalserkennung

Dieses Kapitel befasst sich mit den Bereichen der Merkmalserkennung und Stabilitätsanalyse. Zunächst werden verschiedene Detektoren experimentell auf ihre generelle Eignung hin untersucht und die Stabilität der Erkennung sichergestellt. Im zweiten Teil werden Optimierungen der Parameter sowie der Einfluss möglicher Bildvorverarbeitungen evaluiert und abschließend erfolgt die Bewertung der Stabilität im dynamischen Einsatz sowie der zu erwartenden Präzision des Sensors. Alle folgenden Messwerte wurden dabei auf einem Intel Core i7-3770 (4x 3,40 GHz) mit 8 GB Arbeitsspeicher ermittelt.

4.1. Detektoren

Merkmalsdetektoren werden eingesetzt, um in Bildern möglichst prägnante Punkte oder Bereiche zu finden. Die Ansätze unterscheiden sich jedoch sehr stark darin, welche Strukturen jeweils als prägnant oder relevant erkannt werden. Bereits die Tatsache, dass ein bestimmter Detektor an einer Stelle ein Merkmal findet, trifft eine Aussage über die dortige Beschaffenheit. In Abbildung 11 werden die im Folgenden getesteten Detektoren auf einem Muster für die Ka-

merakalibrierung angewendet. Dies verschafft einen ersten Eindruck von der Funktionsweise der Detektoren und welche Art von Punkten typischerweise erkannt werden.

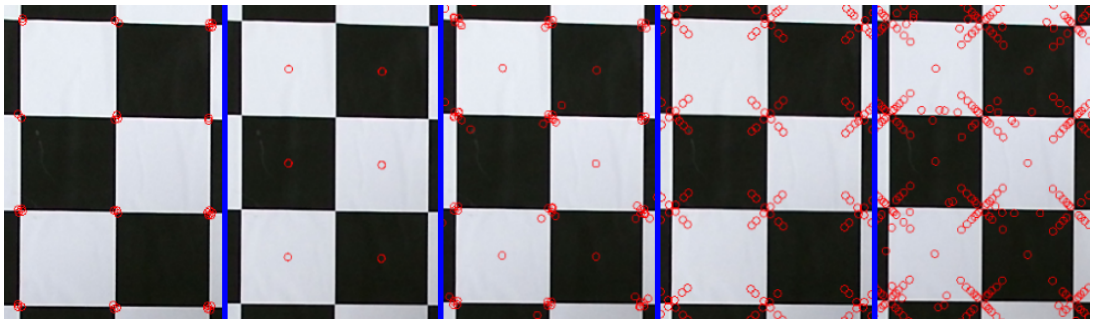

Abbildung 11: Vergleich von Merkmalsdetektoren. Von links nach rechts: FAST, MSER, SIFT, Star, SURF

Untersucht werden diese Detektoren:

- FAST (Rosten und Drummond, 2006, 2005): Ist ein auf Geschwindigkeit optimierter Kanten- bzw. Eckendetektor.
- MSER (Maximally Stable Extremal Regions) (Matas u. a., 2004): Findet Regionen, die sich über einen großen Bereich von Grenzwerten von ihrer Umgebung abgrenzen. Die ausgegebenen Punkte bezeichnen jeweils die Mittelpunkte dieser stabilen Regionen.
- SIFT (Scale Invariant Feature Transform) (Lowe, 1999, 2004): Ist einer der bekanntesten Detektoren der vergangenen Jahre und erkennt Punkte an lokalen Extremwerten über verschiedene Skalierungen des Bildes.
- Star: Basiert auf dem Detektor *CenSurE* von (Agrawal u. a., 2008) und ist technisch eng mit SIFT und SURF verwandt.
- SURF (Speeded Up Robust Features) (Bay u. a., 2006): Wurde als schnellere Alternative zu SIFT entwickelt und approximiert einige aufwendige Operationen des Originals durch schnellere Verfahren.

Als Testszene für die grundlegende Erkennungsrate und Stabilität des Detektors wurden 100 Aufnahmen frontal vor einer Wand in 190 cm Entfernung aufgenommen. Darauf wird für sieben ausgewählte Punkte (Abbildung 12) geprüft, in wie vielen Aufnahmen diese jeweils gefunden wurden. Der Sensor wurde dabei auf einem Stativ positioniert und es gab auch keine sonstigen Änderungen der Umgebung. Als einziger Einfluss auf die Erkennung bleibt das unvermeidliche Sensorrauschen. Die Ergebnisse dieser Versuchsreihe werden in Tabelle 2 dargestellt.

Abbildung 12: Testszene für die Merkmalsdetektoren (zentraler Bildausschnitt)

Detektor \ Punkt	1	2	3	4	5	6	7	Durchschnitt
FAST	0	0	0	0	0	0	0	0
MSER	28	17	3	0	9	17	0	10,6
SIFT	100	100	100	100	99	100	100	99,9
Star	86	76	99	0	99	94	0	64,9
SURF	100	100	100	100	100	100	100	100

Tabelle 2: Anzahl der gefundenen Punkte in statischer Testszene (Abbildung 12)

FAST als eckpunktbasierter Detektor erkennt erwartungsgemäß keines der projizierten Merkmale. Auch MSER findet die meisten Punkte nur sporadisch und die beiden mit dem geringsten Kontrast (Punkt 4 und 7) gar nicht. Mit diesen hat auch der Star Detektor die größten Probleme, liefert aber ansonsten ein deutlich besseres Ergebnis als seine Vorgänger. Jeweils (fast) perfekte Erkennungsquoten bieten die SURF und SIFT Detektoren

4.2. Stabilität der Erkennung

Für die Verfahren SURF und SIFT wird nun die Stabilität unter verschiedenen Winkeln und Distanzen geprüft. Hierzu wurden jeweils 100 Messungen unter den Winkeln 0° (frontal), 15°, 30°, 45°, 60° und 75° (alle bei ca. 200 cm Distanz), sowie zusätzlich unter 0° Winkel bei 100 cm und 330 cm Entfernung zur Wand durchgeführt. (Abbildung 13)

SIFT								
Position \ Punkt	1	2	3	4	5	6	7	Durchschnitt
0°	100	100	100	100	99	100	100	99,9
15°	92	84	63	100	100	100	79	88,3
30°	100	100	100	96	81	100	99	96,6
45°	100	81	100	46	100	100	81	86,9
60°	41	100	100	100	100	100	14	79,3
75°	44	100	100	39	93	70	17	66,1
0°/100 cm	100	100	100	100	100	100	100	100
0°/330 cm	100	100	98	100	83	64	98	91,9
SURF								
Position \ Punkt	1	2	3	4	5	6	7	Durchschnitt
0°	100	100	100	100	100	100	100	100
15°	100	100	100	100	100	100	100	100
30°	100	100	100	100	100	100	99	99,9
45°	100	100	100	100	100	100	100	100
60°	100	100	100	100	100	100	100	100
75°	100	100	100	100	100	99	100	99,9
0°/100 cm	100	100	100	100	100	100	100	100
0°/330 cm	100	100	100	100	100	100	100	100

Tabelle 3: Anzahl der gefundenen Punkte in statischen Testszenen (Abbildung 13)

Wie der Tabelle 3 zu entnehmen ist, lässt mit zunehmender Deformation durch sehr flache Blickwinkel auf die Wand die Zuverlässigkeit von SIFT stark nach. Auch wenn die Punkte aufgrund größerer Distanz sehr klein werden, leidet die Erkennungsrate. Der SURF Detektor dagegen bleibt unter allen getesteten Bedingungen robust und stabil.

Abbildung 13: Testszenen für Winkel- und Distanzabhängigkeit. Von links nach rechts und oben nach unten: 0°, 15°, 30°, 45°, 60°, 75° alle bei ca. 200 cm Distanz, sowie 0° bei 100 cm und 330 cm Entfernung.

4.3. Optimierung der Geschwindigkeit

Pro Bild benötigt die SURF CPU-Implementierung aus OpenCV etwa 170 ms. Da dies die maximale Bildrate auf nur ca. 6 Hz begrenzt, wurde untersucht, ob sich die Geschwindigkeit durch eine Verkleinerung des Bildes auf 50% oder die Begrenzung auf nur zwei Oktaven[21] verbessern lässt, ohne die Erkennungsraten negativ zu beeinflussen. Es werden hierbei neben der Standardszene (0° und 190 cm) nur die drei schwierigsten anderen getestet.

SURF - Bild skaliert auf 50%								
Position \ Punkt	1	2	3	4	5	6	7	Durchschnitt
0°	100	100	100	100	100	100	100	100
75°	0	100	100	100	0	100	100	71,4
0°/100 cm	100	100	100	100	100	100	100	100
0°/330 cm	100	0	0	0	0	100	0	28,6
SURF - Zwei Oktaven								
Position \ Punkt	1	2	3	4	5	6	7	Durchschnitt
0°	100	100	100	100	100	100	100	100
75°	100	100	81	0	100	84	0	66,4
0°/100 cm	0	0	0	0	0	0	0	0
0°/330 cm	100	100	100	100	100	100	100	100

Tabelle 4: Anzahl der gefundenen Punkte in statischen Testszenen (Abbildung 13)

Eine vorherige Verkleinerung des Bildes auf 50% führt zu einer durchschnittlichen Laufzeit von 45 ms, verschlechtert dabei jedoch auch die Detektion von kleinen Merkmalen (Tabelle 4). Bei 330 cm Entfernung werden nur noch zwei der sieben Testpunkte zuverlässig erkannt. Gegenteilig verhält es sich bei der Reduktion auf nur noch zwei Oktaven. In diesem Fall werden die Punkte aus kurzer Distanz nicht mehr erkannt, da sie für die fein aufgelösten Stufen des Detektors zu groß sind. Die Beschleunigung auf 110 ms pro Bild rechtfertigt dabei nicht die Einschnitte in der Erkennungsrate. Alle weiteren Untersuchungen werden daher mit dem Standard-SURF Detektor vorgenommen.

[21] Skalierungsstufen des Bildes zwischen denen die lokalen Extremwerte gesucht werden. Standardeinstellung sind vier Oktaven.

4.4. 3D-Stabilität

Bisher wurde sichergestellt, dass die Merkmale im Farbbild grundsätzlich gefunden werden. Im Folgenden soll nun untersucht werden, wie stabil diese Merkmale in ihren 3D-Beziehungen zueinander sind. Da sich die 2D und 3D Koordinaten der Punkte relativ zum Sensor durch die unterschiedlichen Ausrichtungen verändern, sind diese nicht direkt miteinander vergleichbar. Daher werden für die drei Punkte 1, 3 und 6 jeweils die 3D Distanzen d_{13}, d_{16} und d_{36} als Mittelwerte, sowie die Standardabweichungen σ_{13}, σ_{16} und σ_{36} über alle Messungen einer Testsequenz bestimmt und verglichen.

Neben den bereits vorgestellten acht statischen Einstellungen aus Kapitel 4.2 werden auch die drei folgenden dynamischen Sequenzen für die Untersuchung verwendet:

1. Langsamer horizontaler Schwenk (639 Aufnahmen): Der Sensor wurde auf einem Stativ aufgestellt und horizontal mehrfach in mäßigem Tempo hin und her geschwenkt. Der Schwenkbereich entspricht etwa 100°, was ein Blickfeld von knapp 180° ergibt. (Abbildung 14 oben)
2. Schneller horizontaler Schwenk (252 Aufnahmen): Wie Punkt 1, jedoch schnellere horizontale Drehung des Sensors.
3. Freihand-Bewegung (580 Aufnahmen): Der Sensor wurde freihändig in einem Bereich von ca. 160° horizontal und 40-60° vertikal bewegt. Dabei wurden Abschnitte mehrfach und in unterschiedlichen Geschwindigkeiten erfasst. Der Abstand zu den Wänden betrug zwischen 2-4 m. (Abbildung 14 unten)

Szene	d_{13}	σ_{13}	d_{16}	σ_{16}	d_{36}	σ_{36}
0°	833 mm	2,6 mm	1030 mm	1,4 mm	566 mm	2 mm
15°	832 mm	1,4 mm	1028 mm	1,9 mm	572 mm	2,3 mm
30°	838 mm	1,4 mm	1036 mm	1,2 mm	566 mm	0,4 mm
45°	837 mm	1,4 mm	1034 mm	1,2 mm	574 mm	0,5 mm
60°	843 mm	2,7 mm	1038 mm	2 mm	570 mm	1,6 mm
75°	847 mm	4,9 mm	1043 mm	2,7 mm	572 mm	0,5 mm
0°/100 cm	825 mm	1,5 mm	1025 mm	1,5 mm	577 mm	0,5 mm
0°/330 cm	836 mm	0,5 mm	1028 mm	0,6 mm	570 mm	0,3 mm
Schwenk (1)	835 mm	3,7 mm	1037 mm	4,6 mm	577 mm	3,5 mm
Schwenk schnell (2)	835 mm	3,9 mm	1036 mm	7 mm	576 mm	3,5 mm
Freihand (3)	832 mm	5,7 mm	1032 mm	6,1 mm	575 mm	4,8 mm

Tabelle 5: Durchschnittliche Distanzen und entsprechende Standardabweichungen zwischen den Punkten 1, 3 und 6

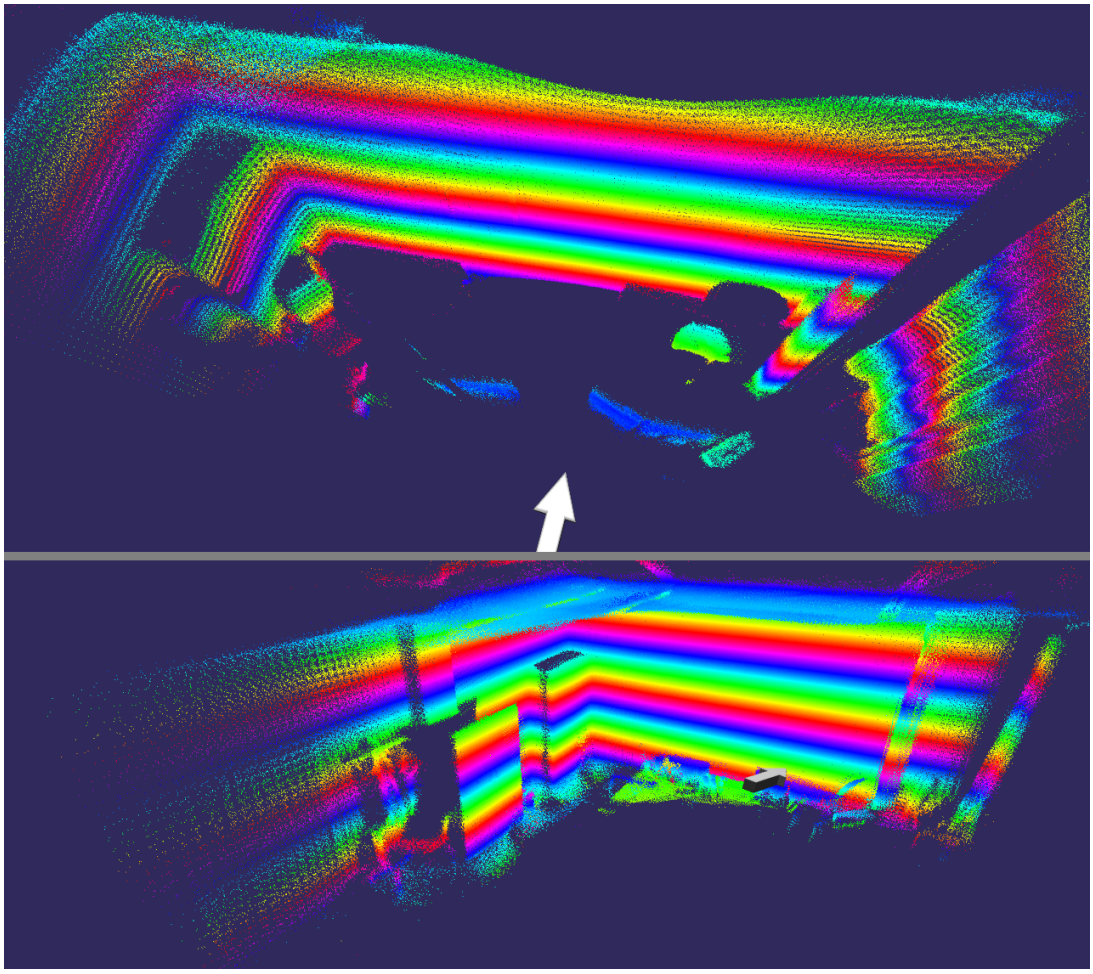

Abbildung 14: Abgedeckte Bereiche der dynamischen Testszenen. Oben: Schwenkszenen (1, 2) Unten: Freihand-Bewegung (3)

Innerhalb der statischen Szenen beträgt die maximale Abweichung zwischen den Durchschnittswerten 22 mm für d_{13}, 18 mm für d_{16} und 11 mm für d_{36} (Tabelle 5). Werden die beiden eher ungewöhnlichen Situationen mit 75° Winkel und 100 cm Distanz ausgeschlossen, verringern sich die Unterschiede auf 11 mm, 10 mm und 8 mm. Die Standardabweichung ist dabei im Wesentlichen durch das Rauschen des Tiefenbildsensors verursacht und mit fast durchgehend unter 3 mm sehr gering. Die 2D Positionen der Merkmale weisen in den statischen Testfällen eine Standardabweichung von durchgehend unter 0,2 Pixeln auf und sind somit weitgehend zu vernachlässigen.

In dynamisch bewegten Szenen ist ein Anstieg der Standardabweichung auf 3,5-7 mm zu beobachten, wobei die absoluten Distanzen dicht beieinander und auch im Rahmen der Ergebnisse der statischen Einstellungen liegen. Die höchsten gemessenen Abweichungen vom Durchschnitt betragen bis zu 40 mm für die Freihand- und knapp 20 mm für die Schwenkszenen. Jeweils mindestens 80% der Werte liegen jedoch im Bereich +/-10 mm um den Mittelwert herum.

5. Feststellung von Korrespondenzen

Um eine neue Aufnahme in das bestehende Modell integrieren zu können, muss zunächst eine passende Transformation bestimmt werden. Diese wiederum lässt sich nur anhand von mindestens drei eindeutigen Merkmalszuordnungen zwischen aktueller Aufnahme und dem Referenzmodell berechnen. In diesem Kapitel werden die Verfahren zur Bestimmung der Korrespondenzen vorgestellt und untersucht.

Nach der Extraktion der Merkmale besteht die aktuelle Messung nur aus einer Ansammlung von 3D Koordinaten. Diese gilt es nun den Punkten des Referenzmodells eindeutig zuzuordnen. Da abgesehen von den Positionen im Raum keine weiteren Informationen über die Punkte vorliegen, müssen für die Zuordnung daher ausschließlich geometrische Eigenschaften verwendet werden.

5.1. Suche der nächsten Nachbarn

Die einfachste geometrische Eigenschaft ist die reine Distanz zweier Punkte zueinander. Sie ist unabhängig vom Bezugssystem und reduziert eine beliebige Dimensionalität auf einen einzigen Wert. Beobachtet man nun einen ortsfesten Punkt mit einem Sensor, so ist die Position des Punktes im Koordinatensystems von der Haltung des Sensors abhängig. Wird der Sensor bewegt, so verschiebt sich der Punkt innerhalb des lokalen Koordinatensystems. Könnte die Bewegung des Sensors dabei in beliebig viele kleine Schritte zerlegt werden, so würde auch die relative Verschiebung des Punktes pro Schritt beliebig klein.

Dank dieses Umstandes lassen sich Punkte über mehrere Aufnahmen hinweg verhältnismäßig einfach verfolgen, indem der Punkt der aktuellen Aufnahme an gleicher Stelle in die vorhergegangene Messung übertragen wird und von dort aus der dichteste Punkt gesucht wird. Ob es sich dabei tatsächlich um denselben Punkt handelt, hängt von zwei Faktoren ab:

1. Keiner der beiden Punkte darf aus einer Instabilität des Detektors entstanden sein, sondern muss in beiden Aufnahmen gefunden worden sein.
2. Die Bewegung des Sensors zwischen den beiden Aufnahmen darf nicht zu groß gewesen sein.

Ab wann genau die Bewegung für eine erfolgreiche Zuordnung zu groß wird, hängt im Wesentlichen von der Distanz der Merkmale zueinander ab. Solange die Verschiebung nicht größer wird als die halbe Distanz zwischen zwei Merkmalen, wird die korrekte Korrespondenz garantiert gefunden. Abbildung 15 zeigt drei unterschiedlich starke Verschiebungen und ihren Einfluss auf die Zuordnungen. Die roten Punkte P_1 entsprechen dabei den Merkmalen des aktuellen Bildes und die Pfeile zeigen auf die jeweils gefundene Korrespondenz aus der vorherigen Aufnahme. Im Fall von nur wenigen Fehlern lässt sich mit Verfahren wie RANSAC trotzdem noch eine korrekte Transformation bestimmen. Sobald jedoch mehr als etwa die Hälfte aller Zuordnungen fehlerhaft sind und im schlimmsten Fall auch noch in ähnliche Richtungen zeigen, ist eine zuverlässige Ermittlung der tatsächlichen Verschiebung nicht mehr möglich.

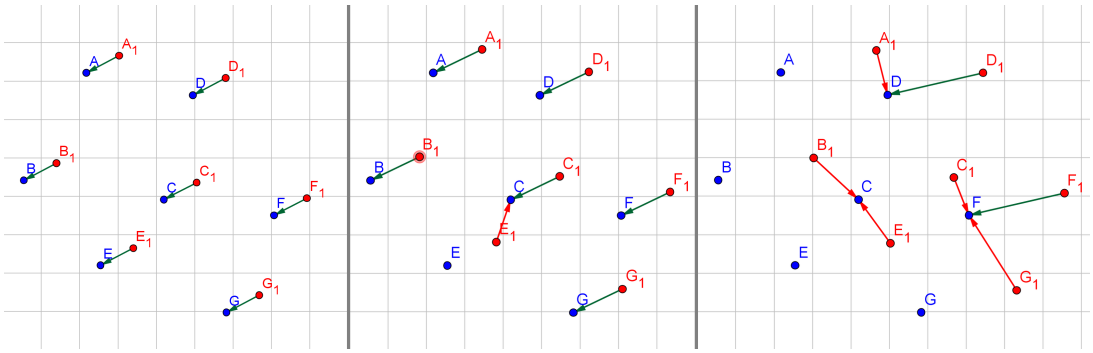

Abbildung 15: Vergleich der Zuordnungen mittels dichtester Nachbarn bei unterschiedlich starken Verschiebungen

Zu weiteren fehlerhaften Zuordnungen kann es durch Punkte kommen, die erstmals am Rand der neuen Aufnahme auftauchen und zu denen es noch keine korrekte Zuordnung geben kann. Um dies so gut wie möglich zu vermeiden, wurde eine Obergrenze für die zulässige Distanz der Korrespondenzen definiert. Der genaue Wert hängt auch wieder von der durchschnittlichen Distanz der projizierten Merkmale voneinander ab und sollte kleiner als diese Distanz

abzüglich der erwarteten durchschnittlichen Sensorbewegung sein. Für die konkrete Projektionsvorrichtung dieser Arbeit hat sich ein Wert von 40 cm als praktikabel erwiesen und liegt allen weiteren Messungen zugrunde.

Mit diesem relativ einfachen Verfahren ist es bereits möglich, in der ersten Testszene (Schwenk langsam aus Kap. 4.4) ein relativ stabiles Tracking zu erreichen. Abbildung 16 zeigt pro Messung, ob eine gültige Transformation ermittelt werden konnte. Die größere Lücke zwischen den Aufnahmen 537 bis 555 entsteht dabei durch einen Bereich, in dem es zu wenige Merkmale gibt, um die Bewegung zuverlässig zu bestimmen.

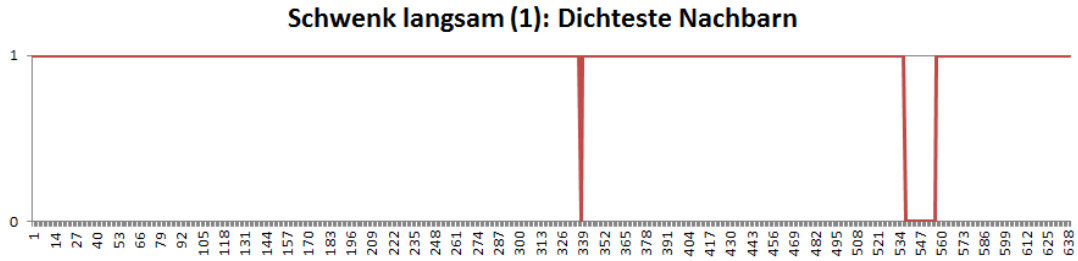

Abbildung 16: Verlauf des Trackings

Dass sich die Kartierung nach einer so großen Lücke wieder fortsetzen ließ, ist jedoch nur dem Umstand geschuldet, dass der Sensor auf dem Stativ ausschließlich horizontal hin und her bewegt wurde und daher beim nächsten Schwenk zurück fast genau an der letzten bekannten Position vorbei kommt. Dies stellt auch das größte Problem eines Ansatzes dar, welcher nur bei kleinen Abweichungen um die letzte Position herum die passenden Korrespondenzen findet. Großer Vorteil der Einfachheit ist jedoch, dass die Implementierung sich sehr gut durch die Verwendung von Indexstrukturen unterstützen lässt, so dass die Feststellung der Korrespondenzen pro Messung im Schnitt unter einer Millisekunde benötigt.

5.2. Globales Referenzmodell

Ebenfalls ein großes Problem bei Verfahren, die zu jedem Zeitpunkt nur den Versatz zu ihrem direkten Vorgänger bestimmen[22], ist sich akkumulierender Drift. Dieser entsteht durch eigentlich kleine Abweichungen (zum Beispiel durch Sensorrausch), die jedoch dazu führen, dass sich mit längerer Dauer die Fehler zu einer großen Abweichung summieren können. Deutlich wird dies insbesondere, wenn zu einem späteren Zeitpunkt ein bereits bekannter Bereich erneut beobachtet wird. Im oberen Teil der Abbildung 17 (entstanden aus der ersten Testszene mit einfacher nächste Nachbarn Suche) ist gut zu erkennen, wie bei jedem Schwenk hin und her der Fehler in der Rotation zunimmt und sich Doppelkonturen der glatten Wand bilden.

[22] In der Robotik auch als *Dead Reckoning* bezeichnet. Deutsch: Koppelnavigation

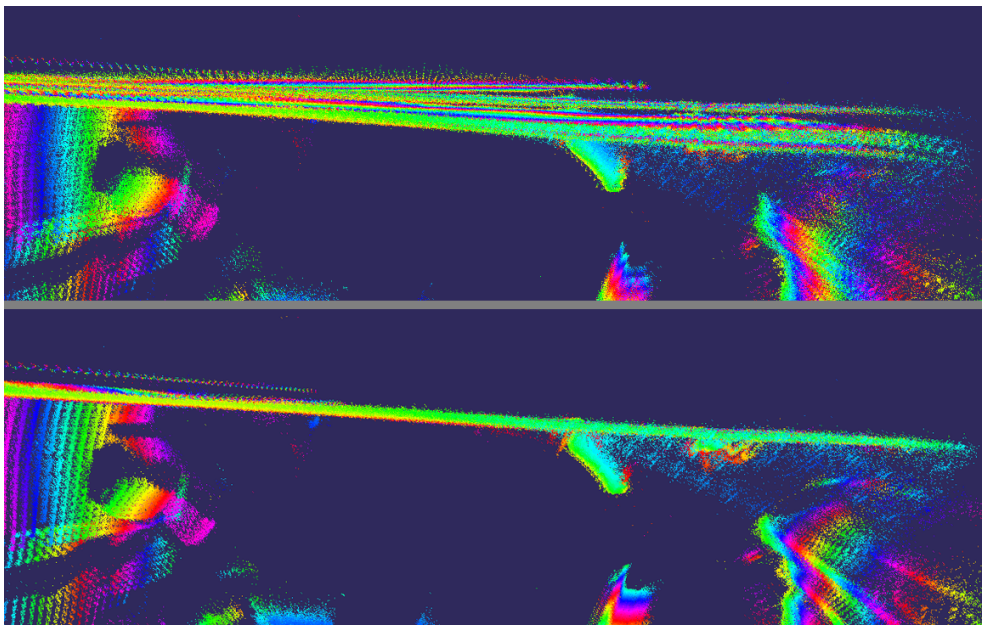

Abbildung 17: Sicht von oben auf eine glatte Wand. Oben: Tracking von Messung zu Messung mit akkumulierendem Fehler. Unten: Tracking mit globalem Referenzmodell.

Zur Behebung dieses Problems wurde ein globales Referenzmodell entwickelt, welches die erkannten Merkmale aller bisher erfolgreich getrackten Messungen zusammenfasst. Als Ursprung dieses Modells wird dabei die Position der ersten Aufnahme gewählt. Um nun eine neue Messung nach erfolgter Bestimmung der Transformation in das Modell einzufügen, werden zunächst die Merkmalspositionen in das Koordinatensystem des Modells umgerechnet. Würden diese Punkte alle direkt in das Gesamtmodell übernommen, so würden sich aufgrund von kleinsten Abweichungen in kürzester Zeit große Ansammlungen von Punkten auf einem Haufen bilden und das Problem des Drifts über die Zeit wäre wieder da.

Für jedes Merkmal, welches in das Modell eingefügt werden soll, wird daher zunächst im Umkreis von 20 cm um die neue Position[23] nach bereits existierenden Punkten gesucht. Nur wenn dabei keine bestehenden Punkte gefunden werden, wird das neue Merkmal mit in das globale Modell aufgenommen. Durch diese Filterung wird sichergestellt, dass räumliche Zuordnungen weiterhin möglichst eindeutig möglich sind und nicht mehrere Punkte innerhalb der Toleranzen des verwendeten Sensors liegen.

Auf eine nachträgliche Interpolation der globalen Merkmalsposition zum Beispiel als Mittelwert aller zugeordneten späteren Messungen wird bewusst verzichtet, da auch hierdurch wieder

[23] Dieser Mindestabstand muss kleiner sein als die kleinste erwartete Distanz zwischen zwei beliebigen projizierten Merkmalen.

(langsamer) Drift ermöglicht wird und auch bereits erfolgte Messungen aufgrund der Veränderungen im Referenzsystem entsprechend angepasst werden müssten. Der dazu nötige Aufwand rechtfertigt nicht die eventuell erzielbaren Verbesserungen in den Merkmalspositionen.

Um einzelne Fehlerkennungen oder instabile Merkmale auch wieder aus dem Modell entfernen zu können, führt jeder Punkt einen internen Zähler mit sich. Wird ein Merkmal in einer Messung erkannt und neu eingefügt oder einem bestehenden Punkt zugeordnet, so wird der Zähler dieses Punktes um zwei erhöht. Sind alle neuen Merkmale in das Modell integriert, wird das komplette Modell mittels der inversen Transformation in das lokale Koordinatensystem des Sensors gebracht und bei sämtlichen Punkten, die nach dieser Umwandlung im Blickfeld des Sensors liegen, der Zähler um eins verringert. Abschließend werden alle Punkte gelöscht, bei denen der Zähler unter Null gefallen ist. Einmalig erkannte Merkmale verschwinden also nach der zweiten Aufnahme ohne Erkennung wieder aus dem Modell.

Müssen nun die Korrespondenzen für eine neue Messung ermittelt werden, so werden immer alle bekannten Merkmale als Referenz verwendet. Da sich die Koordinaten der Merkmale allerdings nun im globalen System befinden, werden diese bei Bedarf (zum Beispiel für die Zuordnung über die nächsten Nachbarn) noch an die letzte bekannte Position transformiert. Die auf diese Art entstandene zweite Aufnahme der Abbildung 17 zeigt dabei den Gewinn an Stabilität des Trackings.

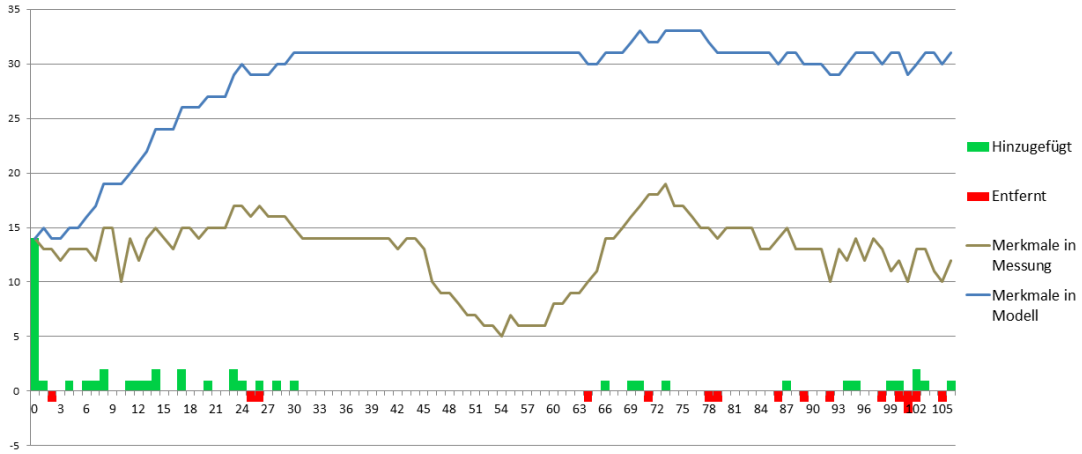

Abbildung 18: Entwicklung der Punkte im globalen Modell in den ersten 107 Messungen des langsamen Schwenkes.

Abbildung 18 zeigt die interne Entwicklung des globalen Modells zu Beginn der langsamen Schwenkszene. Für die Messung 0 werden zunächst alle erkannten Merkmale aufgenommen. Ab hier bis zur Messung 30 dauert der erste Schwenk nach rechts über die Wand an. In diesem Bereich werden von den 10-17 beobachteten Merkmalen pro Messung vereinzelt ein bis zwei

neue Merkmale in den Index aufgenommen, sobald sie erstmals im Blickfeld auftauchen. Auch drei instabile Merkmale werden zwischendurch wieder entfernt.

Zwischen Aufnahme 30 und 45 wird der Sensor nicht bewegt, so dass auch im Modell keinerlei Änderungen mehr stattfinden. Von Bild 45-65 findet eine Bewegung in einen stark strukturierten Bereich statt, in dem keine Merkmale als stabil gelten und komplett gefiltert werden. Daher sinkt die Zahl der erkannten Merkmale pro Messung, auf das Modell hat dies jedoch keinen Einfluss. Ab der Messung 65 bis 107 findet nun der Schwenk zurück zur Startposition statt. Die Fluktuation im Modell ist in diesem Bereich minimal, da im Wesentlichen nur bereits bekannte Punkte erneut beobachtet werden. Ab der Aufnahme 30 kann somit von einem stabilen Modell gesprochen werden, da die Schwankungsbreite bei nur noch +/- 2 Punkte um einen Mittelwert von 31 Merkmalen liegt.

5.3. Distanz-basierte globale Indizierung

Gegen Ende des Abschnitts 5.1 wurde bereits festgestellt, dass die reine Bild zu Bild Zuordnung über die nächsten Nachbarn an ihre Grenzen stößt, sobald größere Sprünge auftreten. Besonders bei der Freihandkartierung ist es dabei mitunter schwierig, genau die letzte bekannte Position zu finden um von dort das Tracking fortsetzen zu können. Im folgenden Teil wird daher eine Indexstruktur und das zugehörige Verfahren vorgestellt, um auch komplett unabhängig von der Kameraposition korrespondierende Merkmale bestimmen zu können.

Ein einzelner Punkt alleine ist zunächst nichts anderes als eine Koordinate im Raum. Um ihn von anderen Punkten unterscheidbar zu machen, wird daher immer der angrenzende Bereich mit einbezogen und dessen Charakteristik bestimmt. Im Fall von SIFT Merkmalen sind das beispielsweise die Gradientenrichtungen der Umgebung. Da für die Kartierung hier jedoch keine dichten Informationen über die direkte Umgebung eines Punktes vorliegen, muss die Identifizierung über die relative Lage des Punktes zu seinen Nachbarn erfolgen. Da sich Winkel zu einzelnen Nachbarn jedoch nicht unabhängig von der Position der Aufnahme bestimmen lassen, bleibt einzig die Entfernung als nutzbare Information.

Indizierung der Merkmale

Der Index basiert nun auf der Annahme, dass sich ein Punkt über die Distanzen zu seinen drei nächsten Nachbarn hinreichend genau identifizieren lässt. In der Praxis ergeben sich daraus jedoch Probleme, sobald ein Merkmal zu viel oder zu wenig in der Messung erkannt wird. Da dieser Fall nicht grundsätzlich ausgeschlossen werden kann, werden tatsächlich die vier nächsten Nachbarn für jeden Punkt bestimmt und daraus die vier möglichen Kombinationen aus jeweils drei Distanzen für die Indizierung erstellt.

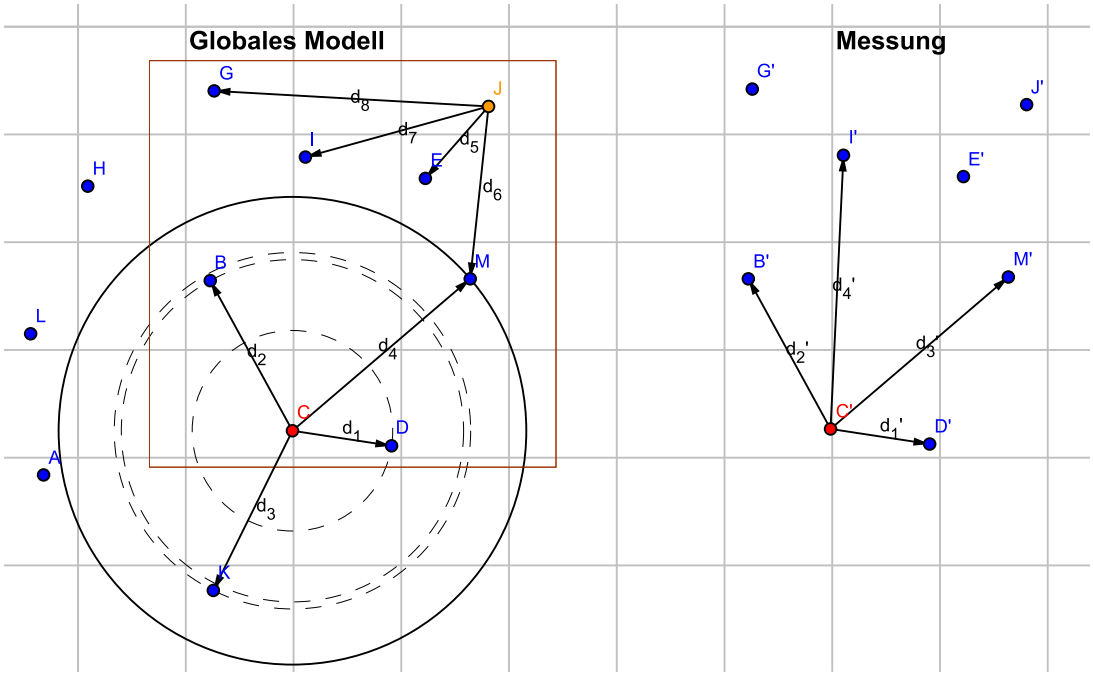

Abbildung 19: Links: Gewählte Distanzen für die Indexerstellung. Rechts: Distanzen für die Suche im Index.

Auf der linken Seite der Abbildung 19 wird dieser Auswahlprozess dargestellt. Der Punkt C hat die vier nächsten Nachbarn D, B, K, M mit den jeweiligen Distanzen d_1, d_2, d_3, d_4. Daraus werden die vier Indexeinträge $[d_1, d_2, d_3], [d_1, d_2, d_4], [d_1, d_3, d_4], [d_2, d_3, d_4]$ generiert. Wichtig dabei ist, die Distanzen immer gleich zu sortieren, damit die Abweichung als einfache euklidische Distanz bestimmt werden kann. Diese Einträge werden nun zusammen mit den Einträgen für alle anderen Punkte in einen k-d-Baum für die effiziente Suche eingefügt.

Suche im Index

Um für einen aktuellen Punkt im Index nach korrespondierenden Referenzpunkten suchen zu können, müssen zunächst wieder die Distanzen zu den Nachbarn des Punkten bestimmt werden. Genau wie bei der Indizierung werden auch hier die vier nächsten Nachbarn einbezogen, um Robustheit gegen Fehlerkennungen zu gewinnen. In der rechten Seite der Beispielabbildung 19 ist der Inhalt der aktuellen Messung abgebildet (der Rahmen zeigt den sichtbaren Ausschnitt im Modell). Für den Punkt C' sind die nächsten Nachbarn D', B', M', I'. Im Vergleich zur Indizierung K durch I ersetzt, da dieser nicht mehr im Sichtfeld des Sen-

sors liegt. Daraus resultieren für die eigentliche Anfrage an den Index die vier Kombinationen $[d'_1, d'_2, d'_3]$, $[d'_1, d'_2, d'_4]$, $[d'_1, d'_3, d'_4]$, $[d'_2, d'_3, d'_4]$.

Für jede dieser Kombinationen wird nun im k-d-Baum nach allen Vektoren gesucht, die einen euklidischen Abstand von d_{max} nicht überschreiten. Dieser hängt von der erwarteten Genauigkeit des Sensors ab und berechnet sich aus der maximalen zulässigen Abweichung pro Einzeldistanz (t_{max}) wie folgt:

$$d_{max} = \sqrt{3(t_{max})^2} \tag{11}$$

Alle Referenzpunkte, die darüber gefunden werden, werden dann in einer Kandidatenliste gesammelt. Im skizzierten Beispiel würde der Suchvektor $[d'_1, d'_2, d'_3]$ mindestens zu dem indizierten Vektor $[d_1, d_2, d_4]$ passen und den Punkt C ergeben[24]. Zusätzlich würde auch der Suchvektor $[d'_1, d'_2, d'_4]$ dicht genug am indizierten Vektor $[d_5, d_6, d_8]$ des Punktes J liegen. Die Kandidatenliste nach der Indexsuche für den Punkt C' enthält also die beiden Punkte C, J.

Im nächsten Schritt wird jeder der Kandidaten einzeln betrachtet und zunächst alle passenden Kombinationen der Einzeldistanzen als Tupel gebildet. Für den Kandidaten C sind das die Tupel $(d'_1, d_1), (d'_2, d_2), (d'_2, d_3), (d'_3, d_4)$. Nun werden für jede Zweierkombination aus Tupeln die Distanzen zwischen den zugehörigen Nachbarn aus Modell und Messung bestimmt. Für die beiden Tupel (d'_1, d_1) und (d'_2, d_2) sind das die Distanzen $d_m = |D' - B'|$ und $d_r = |D - B|$ (Grüne Vektoren in Abbildung 20). Liegt nun die Differenz dieser beiden Distanzen unterhalb der Toleranz für Strecken t_{max}, so wird das Korrespondenz-Triple $((C', C), (D', D), (B', B))$ in eine Liste von Transformationskandidaten aufgenommen. Beim Kandidaten J ergeben sich die Tupel $(d'_1, d_5), (d'_2, d_6), (d'_4, d_8)$. Da dort jedoch die Distanzen zwischen den jeweiligen Kombinationen aus Nachbarpunkten zu groß sind, entstehen daraus keine weiteren Transformationskandidaten (Abb. 20: $\vec{B'D'}$ im Vergleich zu $\vec{MF}$).

Im letzten Schritt wird nun für jedes Triple in den Transformationskandidaten die Transformation bestimmt und alle Merkmale der Messung damit in das Referenzkoordinatensystem transformiert. Anschließend wird geprüft, wie viele der Merkmale dicht genug an einem Referenzpunkt liegen. Sobald mindestens 50% aller Merkmale der Messung diese Bedingung erfüllen, wird die Korrespondenz zwischen Messpunkt und Referenzpunkt als Ergebnis zurückgegeben. Da die Transformationen hier prinzipbedingt aus Korrespondenzen berechnet werden, welche dicht beieinander liegen, muss eine größere Toleranz angewendet werden als bei der späteren Filterung in 3.3. Für diese Implementierung wurde der Grenzwert von 5 cm auf 10 cm verdoppelt.

[24] Aufgrund der sehr ähnlichen Länge von d_2 und d_3 wird auch der Vektor $[d_1, d_3, d_4]$ gefunden. Jeder Kandidat wird jedoch nur einmal in die Liste eingefügt.

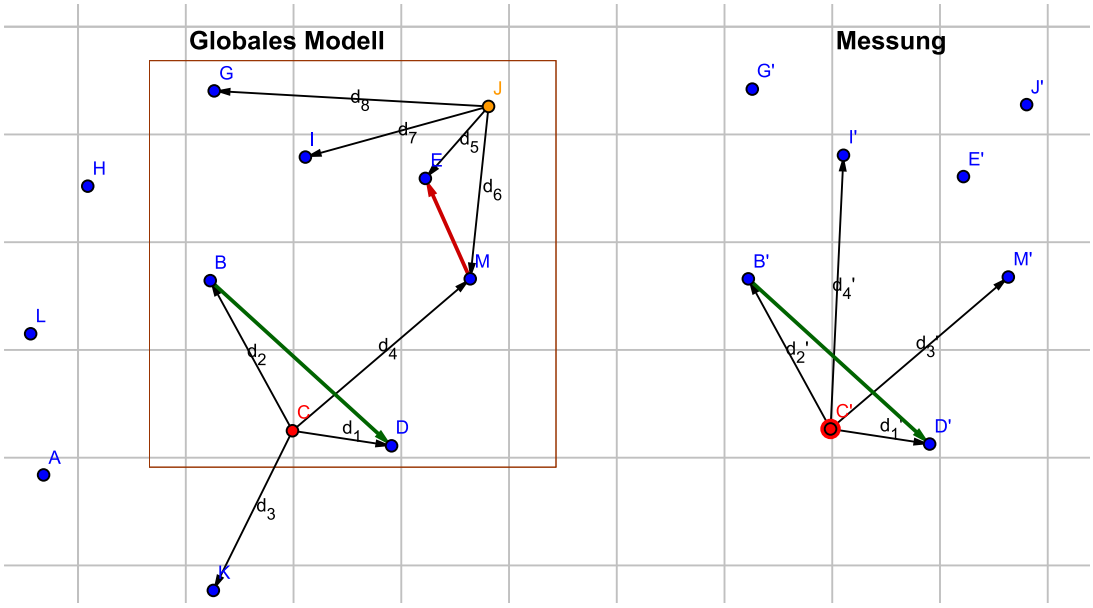

Abbildung 20: Vergleich der Distanzen zwischen den Nachbarn mit ähnlichen Distanzen.

Analyse

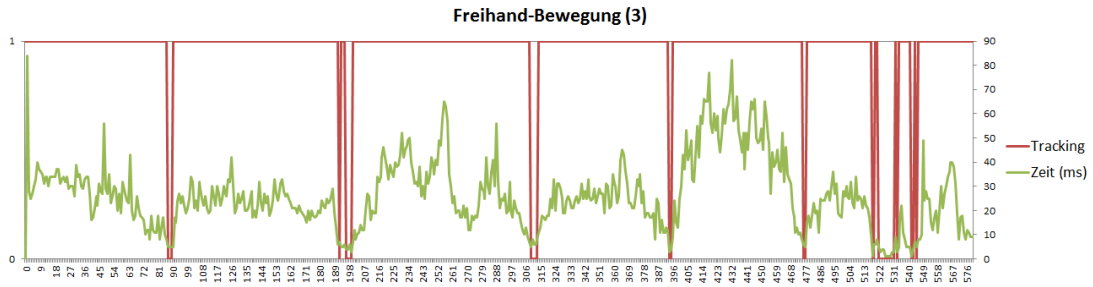

Abbildung 21: Verlauf des Trackings (1=erfolgreich) und der Dauer der Suche im Distanzindex in der handgeführten Testszene

Für die Analyse der Geschwindigkeit und Effizienz des Verfahrens wurde die handgeführte Testszene (3. aus Kap. 4.4) verwendet, da diese den größten Bereich abdeckt und somit auch die meisten Merkmale aufweist. Bei einem Index mit etwa 50 Referenzpunkten benötigt die Suche nach Referenzen im Schnitt etwa 30 ms mit Höchstwerten im Bereich 60-70 ms (Abbildung 21). Dabei wird auf Basis der gefundenen Korrespondenzen für 94% der Messungen auch eine gültige Transformation gefunden.

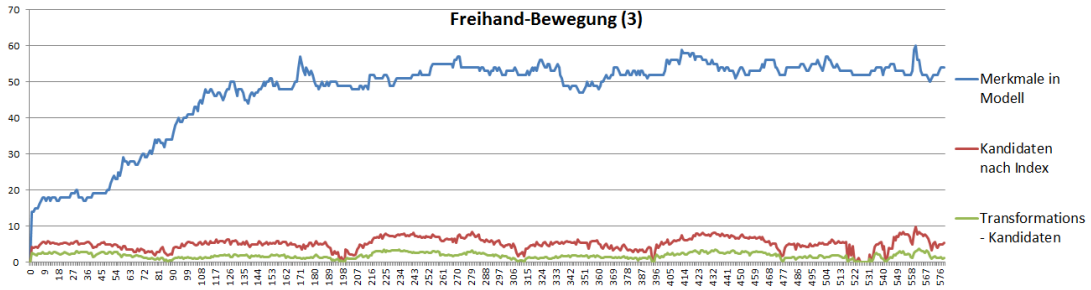

Abbildung 22: Durchschnittliche Anzahl der Kandidaten nach den beiden Filterstufen des Distanzindex in der handgeführten Testszene

Neben der Geschwindigkeit ist vor allem auch die Selektivität eines Index entscheidend. Wäre der Index zu wenig selektiv, so würde sich der Aufwand nicht lohnen, da direkt der komplette Suchraum getestet werden könnte. In Abbildung 22 wurde daher zunächst untersucht, wie viele Kandidaten es im Schnitt pro Merkmal nach der distanzbasierten Suche im Index gibt (Kandidaten nach Index). Ins Verhältnis zur Gesamtzahl der Referenzmerkmale gesetzt ergibt sich so eine Quote von 12%. Es werden also im Schnitt 88% der Punkte bereits durch die erste Stufe ausgeschlossen.

Der zweite relevante Wert ist die Anzahl an Transformationskandidaten, die pro Merkmal gefunden werden. Dieser Wert gibt an, für wie viele Referenzpunkte aus dem Index auch mindestens zwei passende Nachbarn mit ähnlicher Distanz zueinander gefunden wurden. Über alle Messungen in Abbildung 22 hinweg, beträgt der Durchschnitt 1,8 Kandidaten pro Merkmal, für die tatsächlich Transformationen berechnet werden müssen. Perfekt wäre hier ein Wert von 1, wenn ausschließlich korrekte Korrespondenzen die Filterstufen passieren würden. Viel problematischer als eine zu hohe Zahl Kandidaten wäre jedoch, wenn eigentlich passende Referenzpunkte nicht gefunden würden.[25]

Die Anzahl der letztlich, nach Prüfung der Transformation, gefundenen Korrespondenzen wird in Abbildung 23 im Verhältnis zu den gesuchten Merkmalen betrachtet. Zusätzlich wurde erhoben, wie viele fehlerhafte Korrespondenzen gefunden wurden. Als fehlerhaft zählt eine Korrespondenz, wenn die beiden beteiligten Punkte nach einer erfolgreich gefundenen Transformation weiter als 10 cm voneinander entfernt liegen. Im Durchschnitt beträgt die Quote der fehlerhaften Zuordnungen pro Messung nur etwa 3%. Für 77% der Merkmale wird hingegen die jeweils korrekte Korrespondenz ermittelt.

Dieser Wert kann im dynamischen Einsatz die 100% jedoch nie erreichen, da es für Punkte im Randbereich der aktuellen Aufnahme immer sein kann, dass nicht mehr mindestens drei

[25] In der Statistik als *Recall* bezeichnet. Beschreibt das Verhältnis aus gefundenen Lösungen zur Anzahl der insgesamt tatsächlich vorhandenen Lösungen.

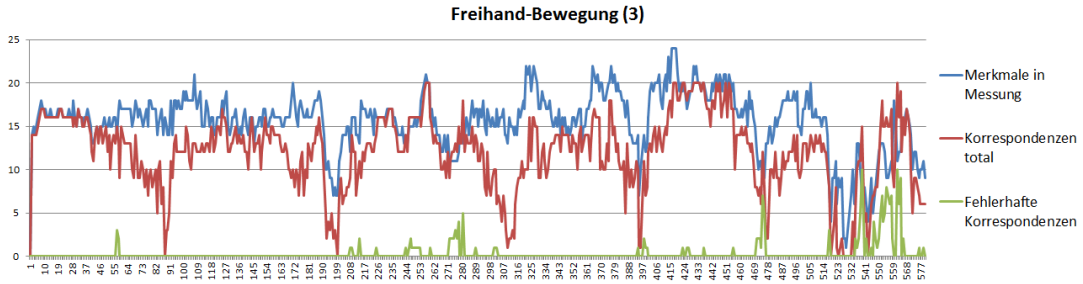

Abbildung 23: Verhältnis der Merkmale einer Messung zu den gefundenen Korrespondenzen des Distanzindex und den davon fehlerhaften in der handgeführten Testszene

der vier nächsten Nachbarn sichtbar sind (beispielsweise Punkt B aus der Abbildung 20). Ein generelles Problem für dieses Verfahren ist, wenn das projizierte Muster zu regelmäßig ist und viele Punkte sich daher bezüglich ihrer Lage zueinander stark ähneln. Einen solchen Fall zeigt die Abbildung 24, in der zwei Reihen Merkmale fast parallel verlaufen.

5.4. Kombinierte Korrespondenzsuche

Beide Verfahren zur Korrespondenzsuche haben ihre Vor- und Nachteile. Die Suche über die nächsten Nachbarn ist sehr schnell und liefert bei kleinen Bewegungen äußerst stabile Ergebnisse, versagt jedoch, sobald ein Sprung zu groß wird und einmal das Tracking verloren wurde. Im Gegensatz dazu ist der Distanzindex völlig unabhängig von der aktuellen Position und hat nur wenige Fehlerkennungen, ist aber verhältnismäßig langsam und hat häufiger Probleme mit vereinzelten Aussetzern.

Um die Vorteile beider Techniken zu kombinieren wird nun für jede neue Messung zunächst geprüft, ob sich mittels der nächsten Nachbarn Korrespondenzen finden und ob diese auch zu einer gültigen Transformation führen. Ist dies nicht der Fall, so wird der distanzbasierte Index verwendet, um unabhängig von der letzten Position nach den Korrespondenzen zu suchen. Dieser Vorgang entspricht einer erneuten globalen Lokalisierung. Wurde auf dieser Basis erfolgreich eine Transformation bestimmt, wird automatisch ab der nächsten Messung wieder der schnellere Ansatz der nächsten Nachbarn verwendet.

Die Stabilität der drei Verfahren (Nächste Nachbarn, Distanzindex und kombinierte Suche) wurde an den beiden anspruchsvolleren Testszenen 2 und 3 (schneller Schwenk und Freihand-Bewegung) verglichen und die Ergebnisse sind in Abbildung 25 dargestellt. Beim schnellen Schwenk haben zu Beginn zwischen Messung 12 und 20 alle Verfahren Probleme mit dem Tracking. Ab dort ist der kombinierte Ansatz jedoch ohne weitere Aussetzer stabil, während

Abbildung 24: Fehlerhafte Korrespondenzbestimmung des Distanzindex

beide Einzeltechniken zwischendurch immer wieder Schwierigkeiten haben. Ähnlich ist die Lage bei der Freihand-Bewegung, wobei das Tracking über die nächsten Nachbarn ab Messung 520 komplett abbricht, wogegen sich die index-basierten Verfahren von dem Einbruch erholen.

Um die Anforderungen weiter zu erhöhen und Daten aus noch schnellerer Bewegung zu simulieren, wurden beide Szenen erneut untersucht, bei der Wiedergabe jedoch jede dritte Messung ausgelassen. Hiervon besonders betroffen ist der Ansatz der nächsten Nachbarn, welcher auf geringen Versatz zwischen den Messungen angewiesen ist. Aber auch für den Distanzindex steigt die Schwierigkeit in neuen Bereichen, da pro Messung mehr neue Merkmale hinzukommen und die bereits indizierten schneller in die Randbereiche rutschen, wo die Erkennungsrate sinkt.

Das Ergebnis dieser Untersuchung (Abbildung 26) spiegelt dann auch die Erwartungen wider. Beim schnellen Schwenk verliert der Ansatz der nächsten Nachbarn sehr häufig das Tracking auch für längere Bereiche. Der Distanzindex hat zu Beginn ebenfalls einen Einbruch in einem Bereich, in dem noch viele neue Merkmale hinzukommen. Danach gibt es jedoch nur noch sporadische Aussetzer. Beide Techniken kombiniert liefern auch hier, bis auf zwei Einbrüche, ein durchgehend stabiles Ergebnis.

Im Fall der Freihand-Bewegung ergeben sich im Vergleich zur Messung ohne übersprungene Messung zunächst kaum größere Unterschiede. Die Bewegung ist immer noch langsam

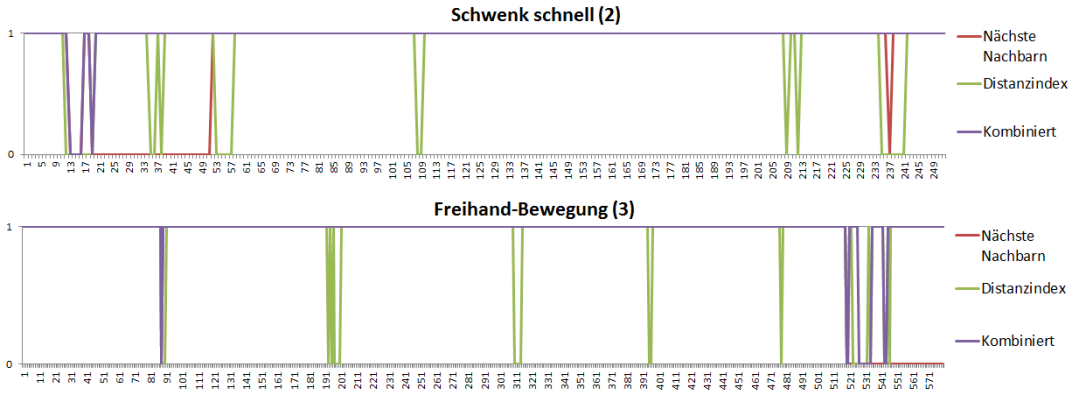

Abbildung 25: Stabilität des Trackings unter Verwendung der verschiedenen Verfahren (1=erfolgreich)

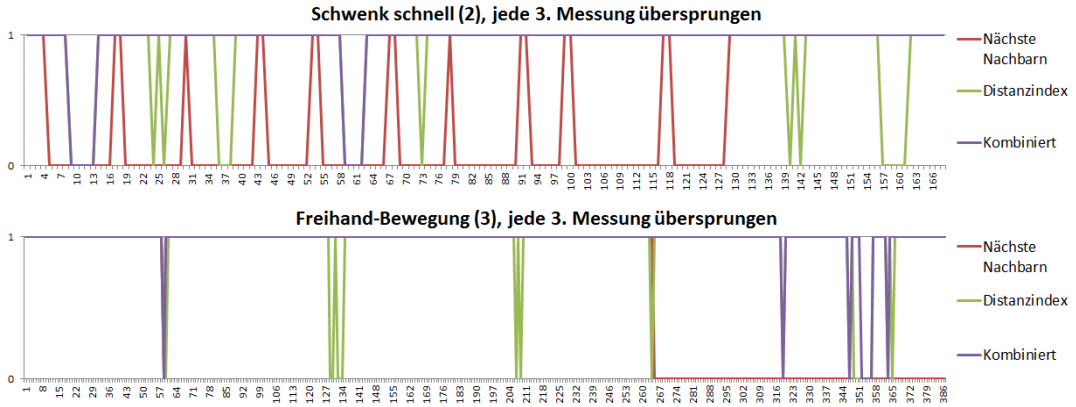

Abbildung 26: Stabilität des Trackings bei übersprungenen Messungen (1=erfolgreich)

genug, so dass in den meisten Fällen auch die Lokalisierung mittels der nächsten Nachbarn erfolgreich ist. Jedoch erfolgt der Abriss des Trackings für diesen Ansatz schon deutlich früher bei Messung 265. Die kombinierte Verwendung der Verfahren ist auch in dieser Testszene in Stabilität und Kontinuität den Einzelverfahren überlegen.

6. Fazit und Ausblick

6.1. Fazit

Im Rahmen dieser Arbeit wurde ein Verfahren entwickelt und präsentiert, welches die Kartierung von Innenräumen mit Hilfe einfachster projizierter Merkmale ermöglicht. Dieses orientiert sich im grundsätzlichen Aufbau an bereits etablierten 3D-Kartierungsverfahren. Die einzelnen Komponenten wurden jedoch der Aufgabenstellung angepasst und Zwischenschritte hinzugefügt (3.2). Als Basis wurde ein Software-Framework entwickelt, das die flexible Evaluation unterschiedlicher Konfigurationen ermöglicht, die aktuellen Abläufe visualisiert und teil-modular aufgebaut ist. Parallel dazu wurde eine Projektionsvorrichtung entworfen und gebaut (A.1), die die Grundlage für alle folgenden Versuche war.

Im nächsten Schritt musste ein Detektor gefunden werden, der die projizierten Merkmale zuverlässig und stabil erkennt (4). Versuche der Optimierung wurden in 4.3 unternommen, führten jedoch zu inakzeptablen Verschlechterungen der Detektionsleistung. Auf Basis besonderer Fähigkeiten des eingesetzten Kinect v2 Sensors wurde dann eine Filtertechnik vorgestellt, die Merkmale in problematischen Bereichen des Bildes fast vollständig entfernt (3.3). Um die Leistungsfähigkeit des Sensors in Verbindung mit den Merkmalen besser einschätzen zu können, wurde in Kapitel 4.4 die zu erwartende Genauigkeit bei stationärer und dynamischer Verwendung untersucht.

Als stabile Basis für Kartierung ohne Drift-Effekte musste nun ein globales Referenzmodell implementiert werden, welches auch die Voraussetzung für globale Lokalisierung war (5.2). Zur Feststellung der Korrespondenzen von einzeln nicht unterscheidbaren Merkmalen wurde zunächst das einfache Verfahren der nächsten Nachbarn umgesetzt (5.1) und für robuste Positionserkennung, auch ohne vorherige Schätzung, ein neues Indizierungsverfahren auf Basis der Distanzen zu Nachbarpunkten entwickelt (5.3). Abschließend wurden diese beiden Verfahren kombiniert und auf Stabilität auch unter deutlich erschwerten Bedingungen untersucht (5.4).

Das vorgestellte Verfahren erfüllt somit die meisten der gestellten Anforderungen. Die verwendete Spezialhardware ist für unter 300 Euro zu beschaffen und ermöglicht die freihändige

Kartierung durch eine Einzelperson, die den Sensor durch den Raum bewegt. Je nach Beschaffenheit des Raumes können zu viele bestehende Merkmale jedoch die Kartierung behindern, da nach der Filterung nicht mehr ausreichend Kandidaten erhalten bleiben. In diesem Fall müsste der Raum vor der Kartierung doch etwas angepasst und „merkmalsärmer“ gemacht werden. Die finale Genauigkeit der Karte ließ sich in der Punktwolkendarstellung nicht absolut bestimmen, die Ergebnisse aus Kapitel 4.4 legen jedoch nahe, dass Distanzen bis auf wenige Zentimeter genau bestimmt werden können.

Für die Live-Verfolgung des Kartierungsfortschritts wurde eine separate Lösung entwickelt, welche die Daten über eine Netzwerkverbindung erhält und somit aus Performance-Gründen auch auf ein externes Gerät ausgelagert werden kann. Derzeit sind in diesen Daten keine Farbinformationen enthalten, ließen sich aber verhältnismäßig einfach nachrüsten, da die Grundlagen vorhanden sind. Dies ist im Wesentlichen eine Frage der Verarbeitungsgeschwindigkeit.

Der größte Vorteil des Verfahrens dürfte die beinahe beliebige Skalierbarkeit und Flexibilität hinsichtlich der kartierbaren Umgebungen sein. Da keine speziellen Anforderungen an die Anordnung der Merkmale gestellt werden, können auch problemlos mehrere Projektoren in Räumen mit komplexeren Grundrissen aufgestellt werden, um alle Wände zu bedecken. Auch lässt sich die Anzahl der projizierten Merkmale einfach über die Größe der verwendeten Spiegelkugeln variieren, und bei viel Umgebungslicht kann der Strahler gegen eine kräftigeres Modell ausgetauscht werden.

Problematisch insgesamt sind die sehr begrenzten Möglichkeiten des Microsoft Kinect-SDK für die Kinect v2. So lässt sich beispielsweise keinerlei Einfluss auf die Belichtungssteuerung der Kameras nehmen, was dazu führt, dass der Sensor bei wenig Umgebungslicht die Geschwindigkeit auf 15 Farbbilder pro Sekunde drosselt und die Synchronizität zwischen Farb- und Tiefenbild recht weit auseinander gehen kann. Dies führt dann mitunter zu hohen Ungenauigkeiten in Schwenkbewegungen und kann das Ergebnis stark beeinträchtigen.

6.2. Ausblick

Für die weitere Bearbeitung des Themas gibt es eine Reihe interessanter Punkte, die das theoretische Verfahren an sich oder die praktische Einsetzbarkeit betreffen. Ein grundlegender Punkt für beide Bereiche ist die Geschwindigkeitsoptimierung der Merkmalserkennung. Bei derzeit nur etwa 6 Hz auf einem starken Desktop-PC stellt dieser Punkt eine deutliche Einschränkung dar. Hier könnten, je nach Hardwareausstattung, SURF-Implementierungen für Grafikprozessoren getestet werden oder eine Optimierung des Detektors an sich auf diesen speziellen Anwendungsfall hin erfolgen.

Für den praktischen Einsatz könnte zusätzliche Präzision gewonnen werden, indem, zusätzlich zur Transformationsbestimmung durch die korrespondierenden Merkmale, noch eine Optimierung anhand der kompletten Punktwolken vorgenommen wird. Hier würden sich beispielsweise

verschiedene *ICP*-Varianten (Kapitel 2.1) anbieten. Darauf aufbauend könnten für die bessere Ergebnisdarstellung auch noch die Farbinformationen integriert und eine Oberflächenrekonstruktion angewendet werden. Dies wäre für die meisten Anwender intuitiv besser Verständlich als die derzeitige farbcodierte Punktwolke.

Verfahrenstechnisch wäre die Integration von *SLAM*-Techniken für eine globale Modelloptimierung und bessere Unterdrückung von Ausreißern interessant. Besonders die Fähigkeit einen expliziten *Loop Closure* zu vollführen, wenn nach einer Schleife erneut bekannte Bereiche kartiert werden, würde die globale Konsistenz der Karte deutlich erhöhen. Die Grundlagen für die Erkennung eines solchen Loop Closures sind durch den Distanzindex schon gegeben, die rückwirkende Korrektur der Zwischenposen ist jedoch kein triviales Unterfangen.

Gänzlich neue Möglichkeiten könnten sich ergeben, wenn Microsoft per Treiber-, Firmware- oder SDK-Update auch das ungefilterte Infrarot-Bild inklusive Fremdbeleuchtung zugänglich machen würde. Durch die Verlagerung der Projektion in das schmalbandig gefilterte Infrarot-Spektrum käme man wahrscheinlich mit deutlich weniger Projektionsleistung aus und die fehleranfällige Transformation der Koordinaten zwischen Farb- und Tiefenbild könnte entfallen. Als positiver Nebeneffekt würden die Merkmale dann auch nicht mehr im Farbbild sichtbar sein, so dass eine rekonstruierte Karte mit Farbinformationen ebenfalls sauberer aussähe.

Literatur

[Agrawal u. a. 2008] AGRAWAL, Motilal ; KONOLIGE, Kurt ; BLAS, MortenRufus: CenSurE: Center Surround Extremas for Realtime Feature Detection and Matching. In: FORSYTH, David (Hrsg.) ; TORR, Philip (Hrsg.) ; ZISSERMAN, Andrew (Hrsg.): *Computer Vision - ECCV 2008* Bd. 5305. Springer Berlin Heidelberg, 2008, S. 102–115. – URL http://dx.doi.org/10.1007/978-3-540-88693-8_8. – ISBN 978-3-540-88692-1

[Bamji u. a. 2015] BAMJI, C.S. ; O'CONNOR, P. ; ELKHATIB, T. ; MEHTA, S. ; THOMPSON, B. ; PRATHER, L.A. ; SNOW, D. ; AKKAYA, O.C. ; DANIEL, A. ; PAYNE, A.D. ; PERRY, T. ; FENTON, M. ; CHAN, Vei-Han: A 0.13 um CMOS System-on-Chip for a 512 x 424 Time-of-Flight Image Sensor With Multi-Frequency Photo-Demodulation up to 130 MHz and 2 GS/s ADC. In: *Solid-State Circuits, IEEE Journal of* 50 (2015), Jan, Nr. 1, S. 303–319. – ISSN 0018-9200

[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, Mai 2006

[Berger u. a. 2014] BERGER, Matthew ; TAGLIASACCHI, Andrea ; SEVERSKY, Lee M. ; ALLIEZ, Pierre ; LEVINE, Joshua A. ; SHARF, Andrei ; SILVA, Claudio: State of the Art in Surface Reconstruction from Point Clouds. In: *Eurographics STAR (Proc. of EG'14)* (2014)

[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), Februar, Nr. 2, S. 239 –256. – ISSN 0162-8828

[Bradski 2000] BRADSKI, G.: The OpenCV Library. In: *Dr. Dobb's Journal of Software Tools* (2000). – URL http://opencv.org

[Canny 1986] CANNY, John: A Computational Approach to Edge Detection. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* PAMI-8 (1986), Nov, Nr. 6, S. 679–698. – ISSN 0162-8828

[Endres u. a. 2012] ENDRES, F. ; HESS, J. ; ENGELHARD, N. ; STURM, J. ; CREMERS, D. ; BURGARD, W.: An evaluation of the RGB-D SLAM system. In: *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, May 2012, S. 1691–1696. – ISSN 1050-4729

[Fankhauser u. a. 2015] FANKHAUSER, Péter ; BLOESCH, Michael ; RODRIGUEZ, Diego ; KAESTNER, Ralf ; HUTTER, Marco ; SIEGWART, Roland: Kinect v2 for Mobile Robot Navigation: Evaluation and Modeling. In: *International Conference on Advanced Robotics (ICAR)* IEEE (Veranst.), 2015, S. 388–394

[Fisher und Konolige 2008] FISHER, Robert B. ; KONOLIGE, Kurt: Range Sensors. In: SICILIANO, Bruno (Hrsg.) ; KHATIB, Oussama (Hrsg.): *Springer Handbook of Robotics*. Springer, 2008, S. 521–542. – URL http://dx.doi.org/10.1007/978-3-540-30301-5_23. – ISBN 978-3-540-23957-4

[Grisetti u. a. 2007] GRISETTI, G. ; STACHNISS, C. ; BURGARD, W.: Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters. In: *Robotics, IEEE Transactions on* 23 (2007), Februar, Nr. 1, S. 34 –46. – ISSN 1552-3098

[Hahnel u. a. 2003] HAHNEL, D. ; BURGARD, W. ; FOX, D. ; THRUN, S.: An efficient fastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements. In: *Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on* Bd. 1, Oct 2003, S. 206–211 vol.1

[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

[Huang 2010] HUANG, Canming: *Emgu CV*. http://www.emgu.com. 2010. – URL http://www.emgu.com

[Lowe 1999] LOWE, David G.: Object recognition from local scale-invariant features, 1999, S. 1150–1157

[Lowe 2004] LOWE, David G.: Distinctive Image Features from Scale-Invariant Keypoints. In: *Int. J. Comput. Vision* 60 (2004), November, Nr. 2, S. 91–110. – URL http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94. – ISSN 0920-5691

[Matas u. a. 2004] MATAS, J ; CHUM, O ; URBAN, M ; PAJDLA, T: Robust wide-baseline stereo from maximally stable extremal regions. In: *Image and Vision Computing* 22 (2004), Nr. 10, S. 761 – 767. – URL http://www.sciencedirect.com/science/article/pii/S0262885604000435. – British Machine Vision Computing 2002. – ISSN 0262-8856

[Meyer-Delius u. a. 2011] MEYER-DELIUS, D. ; BEINHOFER, M. ; KLEINER, A. ; BURGARD, W.: Using artificial landmarks to reduce the ambiguity in the environment of a mobile robot. In: *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, May 2011, S. 5173–5178. – ISSN 1050-4729

[Newcombe u. a. 2011] NEWCOMBE, Richard A. ; IZADI, Shahram ; HILLIGES, Otmar ; MOLYNEAUX, David ; KIM, David ; DAVISON, Andrew J. ; KOHI, Pushmeet ; SHOTTON, Jamie ; HODGES, Steve ; FITZGIBBON, Andrew: KinectFusion: Real-time dense surface mapping and tracking. In: *Mixed and Augmented Reality (ISMAR), 2011 10th IEEE International Symposium on*, Oct 2011, S. 127–136

[Rosten und Drummond 2005] ROSTEN, Edward ; DRUMMOND, Tom: Fusing points and lines for high performance tracking. In: *IEEE International Conference on Computer Vision* Bd. 2, URL http://edwardrosten.com/work/rosten_2005_tracking.pdf, October 2005, S. 1508–1511

[Rosten und Drummond 2006] ROSTEN, Edward ; DRUMMOND, Tom: Machine learning for high-speed corner detection. In: *European Conference on Computer Vision* Bd. 1, URL http://edwardrosten.com/work/rosten_2006_machine.pdf, May 2006, S. 430–443

[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

[Schweiger u. a. 2009] SCHWEIGER, Florian ; ZEISL, Berhnhard ; GEORGEL, Pierre ; SCHROTH, Georg ; STEINBACH, Eckehard ; NAVAB, Nassir: Maximum Detector Response Markers for SIFT and SURF. In: *Vision, Modeling and Visualization Workshop (VMV)*. Braunschweig, Nov 2009

[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

[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. – 10.1007/978-3-540-30301-5_38. – ISBN 978-3-540-30301-5

[Tomono 2009] TOMONO, M.: Detailed 3D mapping based on image edge-point ICP and recovery from registration failure. In: *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, Oct 2009, S. 1164–1169

[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, AK, USA, Mai 2010. – URL http://octomap.sf.net/. – Software available at http://octomap.sf.net/

A. Anhang

A.1. Projektionsvorrichtung

Die Anforderungen an die Projektionsvorrichtung ergaben sich im Wesentlichen aus der Zielsetzung der Arbeit. Es sollte vor allem günstig sein und weitgehend ohne Spezial-Hardware auskommen. Mit möglichst wenig Aufwand muss ein kompletter Raum abgedeckt werden können und die Aufstellung soll möglichst einfach vonstattengehen.

Daraus hat sich eine Konstruktion ergeben, bei der eine Spiegelkugel senkrecht von unten von einem Spot-Strahler beleuchtet wird und somit ein 360° Punktmuster in den Raum projiziert (Abbildung 27). Um eine möglichst geringe seitliche Abschattung bei gleichzeitig hoher Stabilität zu erreichen, wurde für die Aufhängung der Spiegelkugel das Prinzip eines *Hexapod* angewendet (Abbildung 28). Durch die sechs diagonale Verbindungen zwischen Basis und oberer Montageplatte findet eine Stabilisierung in allen sechs Freiheitsgraden statt. Zur einfachen Montage auf einem Stativ wurde im Schwerpunkt der Konstruktion ein stabiler Metallwinkel angebracht, an den eine Schnellwechselplatte montiert wurde (Abbildung 29).

Verwendet wurden die folgenden Teile:

- Eurolite Spiegelkugel mit 15 cm Durchmesser
- American DJ Pinspot mit einer 3 W LED und 12° Abstrahlwinkel
- Zwei Stück Multiplex 300 x 300 x 10 mm für die Ober- und Unterseite des Kopfteils
- Zwei Stück Spanplatte roh 1000 x 150 x 10 mm als Rückwand und Seitenteile (diagonal halbiert) der vertikalen Einheit
- Kleinteile wie Gewindestangen, Muttern, Lochband, Winkel und Schrauben

Die Gesamtkosten für die Konstruktion belaufen sich auf etwa 70 Euro (ohne Stativ). Sie ist leicht genug um von einer Person transportiert zu werden und die Kopfeinheit ließe sich in Zukunft auch demontierbar gestalten. Das Konstrukt ist in sich sehr stabil und verwindungssteif, sollte jedoch auch seitlich am Stativ abgestützt werden, da der Schwerpunkt insgesamt seitlich liegt und handelsübliche Stativköpfe daher mit dem Gewicht überfordert sind und zum Schwingen neigen.

Abbildung 27: Gesamtansicht des Projektors

Abbildung 28: Aufhängung der Spiegelkugel

Abbildung 29: Montage auf dem Stativ mittels Schnellwechseladapter

Versicherung über Selbstständigkeit

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe.

Hamburg, 29. Oktober 2015
Ort, Datum

Unterschrift