
Automatische Grundrisserstellung mittels Laserscandaten

Dorit BORRMANN, Florian LEUTERT, Ivan MAUROVIĆ, Marija SEDER, Andreas NÜCHTER

Zusammenfassung

In den letzten Jahren haben sich Laserscanner zum Stand der Technik bei der Erstellung von Gebäudemodellen entwickelt. Im Gegensatz zur vergleichsweise kurzen Aufnahmezeit bei der eigentlichen Erfassung der Umgebung mit dem Scanner nimmt die Nachbearbeitung der Daten einen deutlich höheren Zeitanteil ein. Eine manuelle Datenanalyse ist zeitaufwändig und fehleranfällig. Dies erhöht den Bedarf an automatischen Verfahren zur quantitativen Erfassung und Charakterisierung von Umgebungen.

In diesem Beitrag präsentieren wir eine automatische Grundrisserstellung basierend auf dem 3D Toolkit (3DTK - <http://www.threedtk.de>). Nach der autonomen Akquise mit einem mobilen Roboter werden die Daten mittels des iterativen Verfahrens der nächsten Punkte (engl. Iterative Closest Point (ICP)) registriert. Anschließend erfolgt eine automatische Vektorisierung eines 2D-Schnitts der Umgebung basierend auf Verfahren aus der Bildverarbeitung. Der so erzeugte Grundriss dient als Grundlage für eine semantische Karte.

1 Einleitung

Beim Laserscanning wird ein Abbild der Umgebung als 3D-Punktwolke erstellt. Eines der zahlreichen Anwendungsbereiche ist die As-built Dokumentation von Gebäuden. Die Punktwolke gibt hier die präzise Geometrie wieder. Für viele Anwendungen, wie die Visualisierung und die Modellierung von Umbauten ist aber eine intensive Nachbearbeitung nötig. Ein wichtiger Bestandteil dabei ist die Vektorisierung der Daten, bei der automatisch elementare, architektonische Elemente wie Wände, Fenster und Türen erkannt und Stördaten herausgefiltert werden. Eine übliche Darstellung ist der Grundriss, da dieser in einfacher Weise die räumlichen Verhältnisse innerhalb eines Gebäudes darstellt. In diesem Beitrag wird ein Ansatz zur vollautomatischen Grundrisserstellung mittels eines terrestrischen Laserscanners vorgestellt. Der Ansatz teilt sich in zwei Arbeitsschritte, die Aufnahme der Daten und die anschließende Verarbeitung ebendieser. Ziel ist, diese beiden Schritte vollautomatisch auszuführen.

2 Automatische Laserscanakquise mit einem mobilen Roboter

2.1 Hardware

Für die automatische Datenakquise wurde der Roboter Irma3D (Abk. engl.: Intelligent Robot for Mapping Applications in 3D) konzipiert (vgl. Abb. 1), der in der Lage ist, auto-

nom in einer unbekanntem Umgebung zu navigieren und eine Karte zu erstellen (NÜCHTER, ELSEBERG & BORRMANN 2013). Das Problem der gleichzeitigen Lokalisierung und Kartierung (engl. Simultaneous Localization and Mapping (SLAM)) ist ein in der Robotik wohl-bekanntes Forschungsgebiet. Hierbei werden zwei Konzepte miteinander kombiniert. Mit den aktuellen Sensorinformationen ist es möglich, sich in einer gegebenen Karte zu lokalisieren. Gleichzeitig ist es ein leichtes, bei bekannter Pose die aktuellen Sensorinformationen zu einer Karte zusammensetzen. Die Sensordaten des Roboters Irma3D stammen von einem Laserscanner. Beginnend von einer Startposition nimmt der Roboter einen Laserscan auf. Dann begibt er sich zum Rand des erfassten Gebietes, um dort mit weiteren Messungen die bestehende Karte zu erweitern. Dies wird solange wiederholt, bis die gesamte erreichbare Umgebung kartiert ist. Im Folgenden wird zuerst die verwendete Hardware genauer beschrieben, anschließend das Verfahren der Sensordatenakquise im Detail erklärt und an einem Beispieldatensatz verdeutlicht.



Abb. 1: Der Roboter Irma3D am Gauß'schen Punkt in Bremen (links) und die finale 3D-Punktwolke der Testumgebung (rechts)

Irma3D ist ein kleiner leichter Roboter, der auf einem Volksbot RT3 Unterbau aufgebaut ist und bei seinen Maßen von ca. $58\text{ cm} \times 52\text{ cm} \times 31,5\text{ cm}$ ein Gewicht von ca. 25 kg aufweist. Die Breite von 52 cm und das dreirädrige Design mit zwei 26 cm großen angetriebenen Luftgummireifen und einem kleineren passiven Kastorrad garantieren eine gute Manövrierbarkeit und ermöglichen die Navigation in engen Innenräumen. Der elektrische Zwei-Rad-Differentialantrieb erreicht seine Maximalgeschwindigkeit von 2,2 m/s durch zwei 150 W DC Maxon Motoren, die mit Drehzahlencodern ausgestattet sind um die Radumdrehungen zur Berechnung der zurückgelegten Strecke zu messen. Dies wird unterstützt durch die Daten eines XSens MTi, einer Inertialmesseinheit (engl. Inertial Measurement Unit (IMU)), die im hinteren Bereich des Roboters angebracht ist. Der Ort ist so gewählt, dass die Einflüsse durch die Magnetfelder der Motoren und der Laserscanner gering gehalten werden.

An der Vorderfront des Roboters ist ein SICK LMS100 Laserscanner angebracht. Dieser liefert Entfernungsmessungen mit einer Rate von 50 Hz in einem Öffnungswinkel von 270° vor dem Roboter. Diese Informationen werden zur Hindernisvermeidung und Navigation verwendet. Der Hauptsensor auf Irma3D ist ein 3D-Laserscanner vom Modell VZ-400 der

Firma Riegl Measurement GmbH. Dieser ist auf dem Volksbot-Gestell verschraubt und dient der Erfassung von 3D-Umgebungsdaten. Auf dem Laserscanner lassen sich weitere Sensoren anbringen. Für das Beispiel in diesem Beitrag befinden sich eine Logitech QuickCam Pro 9000 Webkamera und eine Optris PI160 Thermalkamera auf dem Laserscanner. Beide verfügen über einen USB-Anschluss, mit dem Stromversorgung und Datenübertragung gleichermaßen geregelt werden. Um die Geräte variabel austauschen zu können, ist ein USB-Hub verbaut.

Die Stromversorgung von Irma3D ist durch zwei Stromkreise realisiert, von denen jeder durch zwei 12 V 9,6 Ah Bleiakkus versorgt wird. Die Bleiakkus tragen mit ihrem Gewicht von jeweils ca. 2,5 kg erheblich zum Gesamtgewicht des Roboters bei. Durch leichtere Akkus ließe sich dieses deutlich verringern. Der eine Stromkreis ist ausschließlich für den 3D-Laserscanner, der andere Stromkreis versorgt neben dem Roboter auch die IMU und den 2D-Laserscanner. Die Steuerung des Roboters erfolgt über einen handelsüblichen Laptop. Eine verstellbare Halterung auf dem Gestell des Volksbot erlaubt es, beliebige Laptops in Standardgrößen zu verwenden. Im hier verwendeten Setup kam ein Samsung Q45 Aura Laptop mit einem Intel Core 2 Duo T7100 und 4 GB RAM zum Einsatz. An den Laptop wird der USB-Hub angeschlossen. Die IMU und die Volksbot Motor Controller benötigen jeweils einen seriellen RS232 Anschluss. Da diese auf modernen Laptops selten verbaut sind, werden diese über Adapter angeschlossen. Hierfür stehen zwei RS232-USB-Adapter oder alternativ, wie hier verwendet, ein PCMCIA auf RS232 Adapter zur Verfügung.

2.2 Positionsplanung

Das Positionsplanungsmodul erlaubt die vollautonome Exploration der Umgebung, um ein vollständiges dichtes 3D Modell ebendieser zu erhalten. Das Verfahren wird im Detail von BORRMANN ET AL. (2014) beschrieben. In der Robotik ist es üblich, das Positionsplanungsproblem auf eine horizontale 2D-Ebene auf Höhe des Roboters zu reduzieren. Oft wird dabei die Umgebung mit einem Belegtheitsgitter diskretisiert, bei dem in jedem Feld anhand der Sensormesswerte die Wahrscheinlichkeit vermerkt wird, dass dieses Feld belegt oder frei ist. Nie beobachtete Felder sind mit unbekannt vermerkt. Dies ist eine effiziente Lösung, die für viele Anwendungen gute Ergebnisse liefert. Für die Erstellung eines kompletten 3D-Modells ist dieses Verfahren jedoch nicht ausreichend. Dies wird bereits an einem einfachen Beispiel eines leeren Raumes deutlich. Mit einem horizontalen Öffnungswinkel von 360° kann der Roboter von jeder Position des Raumes aus alle Wände sehen und den Aufnahmeprozess somit als abgeschlossen betrachten. Dabei wird jedoch der vertikale Öffnungswinkel nicht berücksichtigt. Intuitive Überlegungen ergeben, dass es sinnvoller ist, die Aufnahmepositionen nahe der Ecken zu wählen, um durch die Entfernung den vertikalen erfassten Bereich zu maximieren. Aus Effizienzgründen ist es jedoch schwer möglich, die komplette Umgebung mit einem 3D-Belegtheitsgitter zu versehen, da man ein sehr großes Gitter verwenden müsste, um garantieren zu können, dass die ganze unbekannte Umgebung abgedeckt wird und dabei einen enormen Speicher- und Rechenaufwand hätte. Aus diesem Grund stellen BORRMANN ET AL. (2014) eine Kombination aus 2D- und 3D-Positionsplanung vor und zeigen, dass dadurch eine bessere Abdeckung der Umgebung erreicht wird.

Abb. 2 zeigt den Ablauf der Positionsplanung als Blockdiagramm. Der Ablauf der 2D- und 3D-Planung ist identisch. An der aktuellen Position wird ein Scan aufgenommen. Dieser wird in die bisherige Karte integriert um dann in dieser die nächstbeste Scanposition (engl.

Next Best View (NBV)) zu bestimmen. Der Algorithmus beginnt im 2D-Planungsmodus. Nach jeder Scanaufnahme wird der Raumdetektionsalgorithmus gestartet. Findet dieser einen Raum, so wird in den 3D-Planungsmodus übergegangen. Sobald der erkannte Raum komplett erfasst ist, erfolgt der Wechsel zurück in den 2D-Planungsmodus. Die einzelnen Schritte werden im Folgenden im Detail erklärt.

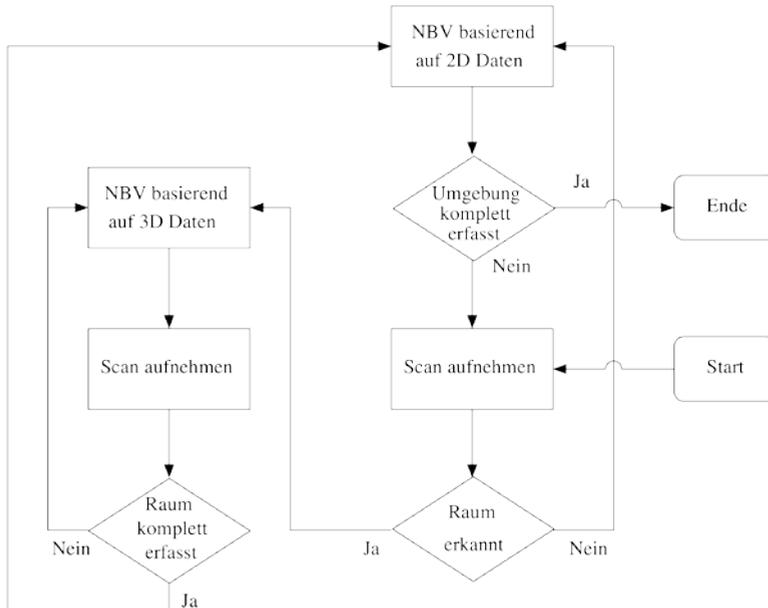


Abb. 2: Blockschema des Sensorplanungsalgorithmus.

2.2.1 2D-Positionsplanung

Der 2D-Positionsplanungsalgorithmus benötigt keine Vorabinformationen über die Umgebung. Die Exploration startet mit einer leeren Karte und gewinnt die Informationen nacheinander aus den neu aufgenommenen 3D-Laserscans. Hierfür wird parallel zum Boden ein Schnitt in die Karte projiziert, der die ganze Höhe des Roboters abdeckt, d.h. alles was zwischen 30 cm und 70 cm über dem Boden liegt. Durch den großen Abstand zum Boden haben Fehler, die durch eine leichte Schräglage des Roboters oder des Laserscanners entstehen, keinen Einfluss auf die resultierende Karte.

Das hierarchisch aufgebaute Umgebungsmodell besteht aus drei Abstraktionsebenen und wird inkrementell erzeugt. Auf der untersten Abstraktionsebene befindet sich das Belegtheitsgitter, das für die Pfadplanung und Hindernisvermeidung verwendet wird. Die nächste Abstraktionsebene besteht aus einer polygonalen Darstellung. Hierzu werden mittels PPHT (Abk. Progressive Probabilistic Hough Transform (MATAS, GALAMBOS & KITTLER 1998)) und der Methode der kleinsten Quadrate Liniensegmente aus der Projektion der Scanpunkte extrahiert. Diese werden zu einem Polygon P_i zusammengefügt, wobei i der Index der aktuellen Scanposition ist. Lücken im Polygon werden durch künstliche Kanten, sogenannte Sprungkanten, gefüllt. Um Sprungkanten in bereits zuvor explorierten Berei-

chen zu vermeiden, ist das neue Explorationspolygon EP_i die Vereinigung des neuen Polygons P_i und des Explorationspolygons EP_{i-1} aus den vorherigen Scans. Die Vereinigung zweier Polygone beinhaltet nur jene Kanten, die aus Sicht des Roboters die größte Entfernung haben.

Die potentiellen Scanpositionen bilden die höchste Abstraktionsebene. Vor jeder Sprungkante aus EP_i , deren Länge die Breite des Roboters übersteigt, wird anschließend eine dieser Positionen generiert. Die beste dieser Positionen (NBV, engl. Next Best View), und damit die nächste, wird heuristisch bestimmt. Hierbei gehen der Weg zur potentiellen Scanposition, aber auch die zu erwartenden Informationen, gegeben durch die Summe der Winkels zwischen potentieller Scanposition und allen Sprungkanten, gewichtet in die Heuristik ein. Abb. 3 veranschaulicht das Winkelkriterium anhand des Positionskandidaten p_1 .

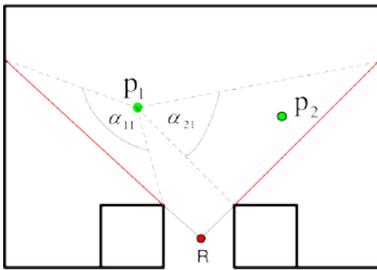


Abb. 3: Auswahlkriterium basierend auf Winkeln. Für jeden Positionskandidaten wird die Summe der Winkel der Dreiecke zwischen Position und den Sprungkanten bestimmt.

Sei N die Anzahl der Kanten in EP_i , d_j der Abstand zwischen den Positionen i und j , α_{jk} der Winkel im Dreieck, das aus j -tem Scankandidaten und k -ter Sprungkante definiert ist, und die Gewichtsparameter, w_1 die maximale Reichweite des 2D-Laserscanners und $w_2 = 1/N$, dann berechnet sich für Scankandidat j der heuristische Wert als

$$I_j = w_1 \frac{1}{d_j} + w_2 \sum_{i=1}^N \alpha_{jk}. \quad (1)$$

2.2.2 Raumerkennung

Ein elementarer Bestandteil des vorgestellten Planungsalgorithmus ist der Raumerkennungsalgorithmus, der den Wechsel von der 2D- zur 3D-Planung auslöst. Nach jedem aufgenommenen Scan in der 2D-Planungsphase wird nach einem umschlossenen Bereich gesucht. Im optimalen Fall handelt es sich dabei um einen Raum, aber die Korrektheit ist nicht kritisch für den Erfolg der Exploration. Dafür wird aus den 3D-Daten aller bisherigen Positionen ein horizontaler Schnitt in Deckennähe gewählt, da dieser Bereich erwartungsgemäß frei von Hindernissen sowie von Öffnungen, wie Fenstern und Türen, ist.

Sei A die Menge der Kanten des zuletzt aufgenommenen Scans. Beginnend von der ersten Linie in A beginnt der Algorithmus mit Hilfe anderer Segmente der kompletten Karte einen geschlossenen Linienzug zu finden. Hierbei wird jeweils zur aktuellen Kante die nächste aus der Karte gesucht. Wird dabei wieder der Anfang erreicht, ist der Linienzug geschlossen und der Raum detektiert. Als nächste Kante kommt jedes Segment in Frage, dessen Endpunkt einen kleineren Abstand zum Endpunkt der aktuellen Linie hat, als die erwartete Türbreite. Wird von keiner Kante aus A ausgehend ein geschlossener Linienzug gefunden, wurde kein Raum erkannt und der 2D-Planungsalgorithmus wird fortgesetzt.

2.2.3 3D-Positionsplanung

Bedingt durch die Vorgehensweise bei der Raumerkennung wurde im Erfolgsfall mindestens ein 3D-Scan innerhalb des Raumes aufgenommen. Der gefundene Linienzug liefert die Raumparameter, insbesondere die Bounding Box des Raumes. Das Modell für die Raumerkennung wird mit allen Scans, die innerhalb des Raumes aufgenommen wurden initialisiert. Hierfür wird ein Voxelgrid erstellt, das größer ist als der größte erwartete Raum in der Umgebung. Der Einfachheit halber wird die Form eines Quaders gewählt. Die Größe der Voxel richtet sich nach der Größe des Raumes und dem verfügbaren Speicher, im Beispiel beträgt die Größe ($0,2 \text{ m} \times 0,2 \text{ m} \times 0,2 \text{ m}$).

Jeder Voxel ist zunächst als *unseen* deklariert. Für alle Scans, die innerhalb des Raumes aufgenommen wurden wird ein Raytracingverfahren angewandt. Ein Strahl (engl. ray) zwischen der Scanposition und jedem Scanpunkt wird verfolgt. Jeder Voxel, der auf dem Weg zum Scanpunkt durchquert wird, erhält das Label *empty*. Der Voxel, der den Datenpunkt umschließt wird mit *occupied* markiert. Potentielle NBV-Kandidaten sind alle *empty* deklarierten Voxel auf der Höhe des 3D-Laserscanners, die für den Roboter erreichbar sind. Aus diesen Kandidaten ist die NBV-Position jene, von der die meisten *unseen* Voxel gesehen werden können. Dies wird ebenfalls mit einem Raytracing-Verfahren ausgehend von den NBV-Kandidaten bestimmt. Um die Anzahl der zu testenden Voxel zu reduzieren, werden nur jene *unseen* Voxel überprüft, die mindestens einen Nachbarn mit dem Label *empty* haben (BLAER & ALLEN 2007). Dies ist ähnlich dem Konzept der Sprungkanten aus der 2D-Positionsplanung, da auch hier die Grenze zwischen bekannten und unbekanntem Bereichen berücksichtigt wird. Ist eine NBV-Position gefunden, welche die Anzahl der *unseen* Voxel mit mindestens einem *empty* Nachbarn maximiert, wird diese angefahren um mit dem dort aufgenommenen Scan das Modell zu vervollständigen. Fällt die Anzahl der *unseen* Voxel, die von der NBV-Position gesehen werden können unter einen Schwellwert, gilt der Raum als vollständig erfasst und die 2D-Positionsplanung wird fortgesetzt. Auch während der 3D-Positionsplanung wird das Umgebungsmodell aus der 2D-Positionsplanung durch die neuen Daten aktualisiert.

2.3 Experimentbeschreibung

Als Testumgebung wurde ein Bereich des Research I Gebäudes der Jacobs University Bremen gewählt. Ein Teil eines Flures mit drei angrenzenden Räumen wurde eingegrenzt und sollte vom Roboter autonom erfasst werden. An jeder Scanposition wurde ein $360^\circ \times 100^\circ$ Laserscan mit einer horizontalen und vertikalen Auflösung von $0,04^\circ$ aufgenommen. Der Scannerkopf rotiert während der Aufnahme, um den horizontalen Öffnungswinkel zu erreichen. Dieses wird zur Abdeckung des horizontalen Blickfeldes mit jeweils 10 Kamerabilddern ausgenutzt. Nach dem Scanvorgang rotiert der Scanner zurück, um die Kamerabilder

aufzunehmen und gleichzeitig die USB-Kabel abzuwickeln. Die Sensoren sind geometrisch kalibriert, so dass die Scanpunkte mit den Informationen der beiden Kameras erweitert werden. Das hierzu verwendete Verfahren wird von BORRMANN ET AL. (2013) beschrieben.

Die Pose des Roboters wird durch Radencoder, IMU und Laserscanner geschätzt. Die Genauigkeit der Poseschätzung ist ausreichend, um die weitere Positionsplanung damit durchzuführen. Für die finale 3D-Karte reicht sie nicht aus. Deswegen wird im Anschluss an die Datenaufnahme die globale Variante des iterativen Algorithmus der nächsten Punkte (engl.: Iterative Closest Point (ICP)) verwendet, wie von NÜCHTER ET AL. (2008) beschrieben.

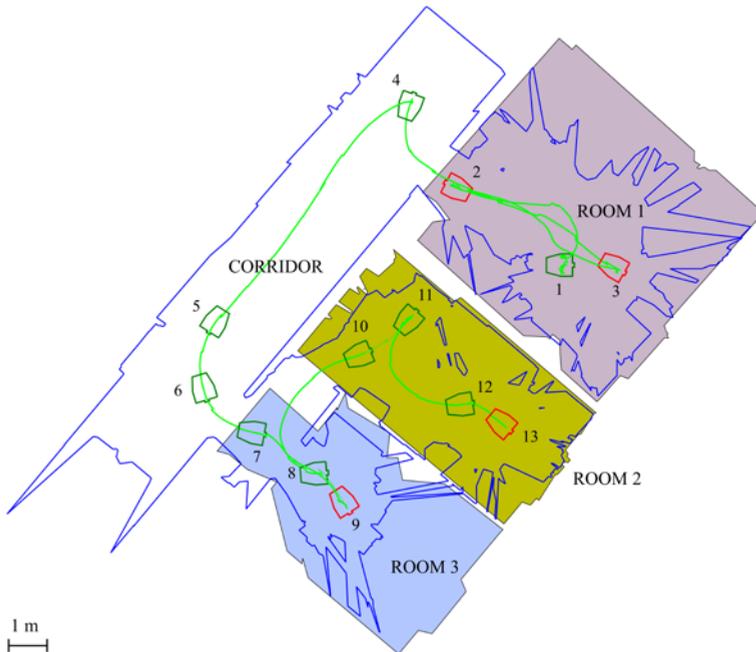


Abb. 4: Finale Karte. Eingezeichnet sind der Roboterpfad, die Scanposition und die erkannten Räume. Die rot markieren Scanpositionen wurden per 3D-Positionsplanung ausgewählt.

Abb. 4 veranschaulicht den Verlauf der Exploration, die Scanpositionen, den Roboterpfad und die erkannten Räume. Die rot markierten Scanpositionen sind die von 3D-Positionsplanung gewählten, d.h. *Room 1* wurde nach Scanposition p_1 erkannt, *Room 3* nach Scanposition p_8 und *Room 2* nach Scanposition p_{12} . Eine detailliertere Beschreibung der Exploration findet sich bei (BORRMANN ET AL., 2014). In der finalen Karte sieht man, dass der Roboter die komplette Umgebung erfolgreich abgefahren ist. Durch das blau eingezeichneten Explorationspolygon und die eingefärbten Räumen wird noch einmal deutlich, dass ein Schnitt in Bodennähe zwar zur Hindernisvermeidung gut ist, aber oftmals nicht die Geometrie der Szene widerspiegelt. Der Schnitt in Deckennähe, der zur Raumerkennung verwendet wurde, kommt der Realität deutlich näher. Aber auch hier treten aufgrund zahlreicher Verdeckungen Probleme auf. Wände sind nicht gerade und der kleine

Flur zwischen *Room 2* und *Room 3*, bei Scanposition p_7 , wird nicht als solcher erkannt. Im Folgenden wird gezeigt, wie die komplette 3D-Punktwolke nach der Exploration verwendet werden kann um den Grundriss zu extrahieren.

3 Grundrisserstellung

Ein Grundriss ist eine zweidimensionale Abbildung, die den horizontalen Schnitt eines Gebäudes wiedergibt. Für die Grundrisserstellung muss demzufolge die kontinuierliche 3D-Punktwolke in eine vektorisierte zweidimensionale Liniendarstellung übertragen werden. Hierzu wird in Zwischenschritten zunächst eine dreidimensionale Rasterdarstellung erzeugt, die dann zu einer zweidimensionalen Pixelgrafik reduziert wird, aus der wiederum die vektorisierte Darstellung berechnet wird. Im Folgenden wird vom Nivellement des Scanners während der Aufnahme ausgegangen.

Beim Aufnahmeverfahren mit einem 3D-Laserscanner nimmt bei zunehmender Entfernung zum Scanner die Punktdichte ab. Folglich sind die Punkte ungleich verteilt. Um eine Gleichverteilung zu erreichen, die Daten zu reduzieren und gleichzeitig ihre Verarbeitung zu vereinfachen, ist der erste Schritt der Grundrisserstellung die Diskretisierung der Daten zu einem Belegtheitsvoxelgrid. Hierzu wird die Bounding Box der Szene bestimmt und in einen Würfel mit frei wählbarer Seitenlänge w unterteilt. Im Beispiel ergeben sich für die Szene mit dem Ausmaß von $x = [-754,993 \text{ cm}; 912,012 \text{ cm}]$, $y = [0 \text{ cm}; 846,552 \text{ cm}]$, $z = [-1440,69 \text{ cm}; 509,344 \text{ cm}]$ und $w = 2 \text{ cm}$ über 350 Mio. Voxel. Für jeden so entstandenen Voxel wird vermerkt, ob er mindestens einen Punkt enthält. Decke und Boden zeichnen sich durch eine Häufung von Punkten in einem horizontalen Schnitt der Szene. In einem Histogramm, das die Anzahl der belegten Voxel eines jeden horizontalen Schnitts des Datensatzes zählt, entsprechen demnach die Maxima der Decke und dem Boden.

Abb. 5 zeigt das Histogramm des Beispieldatensatzes. Deutlich erkennbar sind die Maxima für den Boden mit dem Zählerstand 28.570 in Ebene 87 und die Decke mit dem Zählerstand 40.547 in Ebene 234. Zur weiteren Verarbeitung wird die Bounding Box und damit das Grid auf den Bereich zwischen Decke und Boden verkleinert

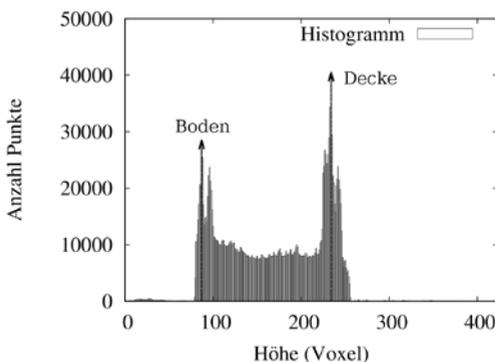


Abb. 5: Histogramm zum Erkennen von Decke und Boden. In jeder Ebene wird die Anzahl der belegten Voxel gezählt.

Aus der nun verbleibenden Szene wird durch gewichtete Projektion ein Graustufengrundriss erzeugt. Hierfür wird ein zweidimensionales Histogramm parallel zur Bodenebenen generiert, d.h. es wird die Anzahl der belegten Voxel pro Spalte gezählt. Skaliert um den Faktor $255.0/H$, wobei H der Höhe der Bounding Box in Anzahl an Voxel entspricht, lässt sich so die Szene als Graustufenbild darstellen. Das Ergebnis aus den Testdaten ist in Abb. 6 zu sehen. Durch die Konvertierung der 3D-Punktwolke in ein Graustufenbild lassen sich nun Bildverarbeitungsalgorithmen darauf anwenden. Je dunkler ein Pixel, desto größer ist die Wahrscheinlichkeit, dass es sich um eine durchgängige Wand handelt.

Zur weiteren Verarbeitung wird das Graustufenbild zunächst in ein Binär- (d.h. Schwarz-Weiß-)Bild umgewandelt. Für diese Aufteilung des Bildes in Vorder- und Hintergrund-Pixel gibt es verschiedene Verfahren, die von simpler Anwendung eines fixen Grenzwertes für die Klassifikation, bis zur Berechnung individueller Schwellwerte für jeden Pixel basierend auf Mittelwerten und Varianz der Nachbarschaft reichen. Gute Ergebnisse konnten hier erzielt werden, indem ein fixer Schwellwert aus dem Mittelwert μ und der Standardabweichung σ der Grauwerte als $s = \mu + 2\sigma$ berechnet wird (vgl. Abb. 6). Durch diese Binarisierung werden die wenig gewichteten Pixel (entsprechend Objekten der Umgebung mit einer niedrigen Höhe) aus den Scandaten entfernt. Im Bild verbleiben Hintergrundpixel (entsprechend freiem/gefiltertem Raum im Scan), sowie Vordergrundpixel, die Objekten mit einer gewissen Mindesthöhe entsprechen.

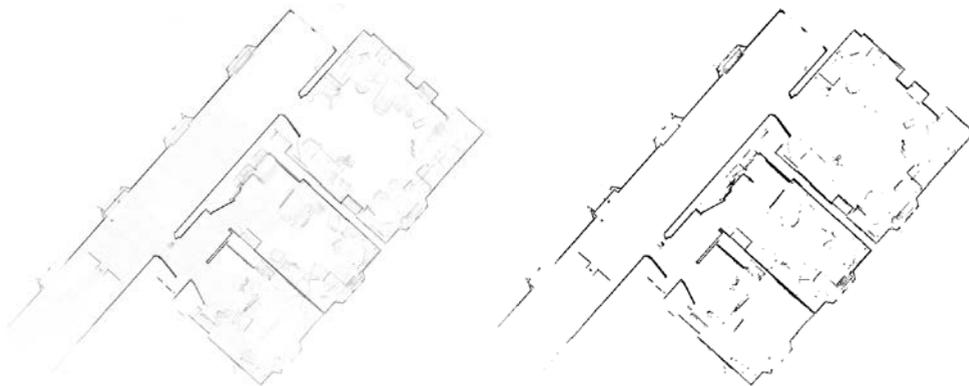


Abb. 6: Links: Histogramm als Graubild. Je dunkler der Pixel, desto mehr belegte Voxel befinden sich in dem Bereich. Rechts: Aus dem Graubild mittels Schwellwert errechnetes Binärbild.

Im nächsten Schritt sollen nun die Wände als lange, geradlinige Abschnitte im Scanbild bestimmt werden. Diese Wände dienen später zur Klassifikation der einzelnen Räume im Grundriss. Dafür werden im Binärbild zunächst die Vordergrundpixel ausgedünnt, bis alle Vordergrund-Segmente nur noch eine Breite von einem Pixel haben. Mithilfe eines Algorithmus ähnlich dem von LOWE (1987) werden anschließend zusammenhängende Linien-Segmente mit einer gewissen Mindestlänge im Bild bestimmt; nicht zu diesen Segmenten gehörige Vordergrundpixel werden aus dem Bild entfernt. Die verbleibenden Segmente sind in Abb. 7 dargestellt. Die Einschränkung auf eine Mindestlänge und eine gerade Form filtert den Scan also erneut: kurze oder runde Objekte (die nicht den gesuchten Wänden

entsprechen) werden aus dem Scan-Bild herausgenommen. Kreisförmige Räume können mit diesem Verfahren nicht bestimmt werden. Da in der gängigen Architektur Räume aber so gut wie immer aus geraden (nicht notwendigerweise rechtwinkligen) Wandabschnitten bestehen, stellt diese Einschränkung kein wesentliches Problem dar.

Im verbleibenden Scanbild werden anschließend mittels der Hough-Transformation (HOUGH 1962; FERNANDES & OLIVEIRA 2008) die wesentlichen dominanten Linien im Bild bestimmt. Diese entsprechen somit Wandabschnitten mit einer gewissen Länge - je länger der Abschnitt desto dominanter die jeweilige Linie. Auf diese Weise werden zu kurze Wandabschnitte oder verbleibende andere Objekte im Scanbild für die folgende Klassifikation unterdrückt. Häufig kommt es bei der Bestimmung von Linien mit der Methode (FERNANDES & OLIVEIRA 2008) zu mehreren lokalen Maxima im Hough-Raum, entsprechend mehreren, fast gleichen Kandidaten für die jeweilige Linie. Um solche Duplikate zu vermeiden, werden die bestimmten dominanten Linien in ähnliche Gruppen eingeteilt (mit vergleichbarem Winkel und Position in Bild), und aus den jeweiligen Gruppen wiederum nur die dominanteste (grüne Linien in Abb. 7) für die weitere Bearbeitung verwendet.



Abb. 7: Die Vektorisierung der Scanpunkte von links nach rechts: Liniensegmentierung, Bestimmung dominanter Geraden, erneute Liniensegmentierung. Zur Verbesserung der Sichtbarkeit wurden die Linien mit einer erhöhten Stärke gezeichnet.

Der finale Schritt zur Bestimmung des Grundrisses ist schließlich die Klassifikation einzelner Räume, basierend auf den bestimmten dominantesten Linien. Da eine Linie im Scanbild (entsprechend einem Wandabschnitt) durchaus Teil mehrerer Räume sein kann, werden dafür die bestimmten dominanten Linien an den Schnittpunkten nochmals unterteilt. Segmente, auf denen sich kein Scanpunkt befindet, werden entfernt. Ein Raum wird definiert als ein geschlossener Linienzug aus den verbleibenden Liniensegmenten. Um diese geschlossenen Linienzüge zu bestimmen, wird aus den einzelnen Segmenten ein Nachbarschaftsgraph aufgebaut: Beginnend an einem noch nicht bearbeiteten Segment wird dieses (als Wurzel) eingefügt, alle Nachbarn des Segments (als Knoten) in diesen Suchgraphen eingefügt und das Verfahren rekursiv auf diesen Nachbarn ausgeführt. Alle so erstellten Suchgraphen werden anschließend durchlaufen; ein geschlossener Linienzug wurde gefunden, wenn einer der Nachbarknoten dem Start-Knoten entspricht. Auf diese Weise werden geschlossene Linienzüge mehrfach gefunden; diese Duplikate müssen abschließend noch detektiert und entfernt werden. Die so bestimmten geschlossenen Linienzüge entsprechen dann einem Raum, und können als solcher automatisch aus den Scandaten im Grundriss bestimmt werden (vgl. Abb. 8).

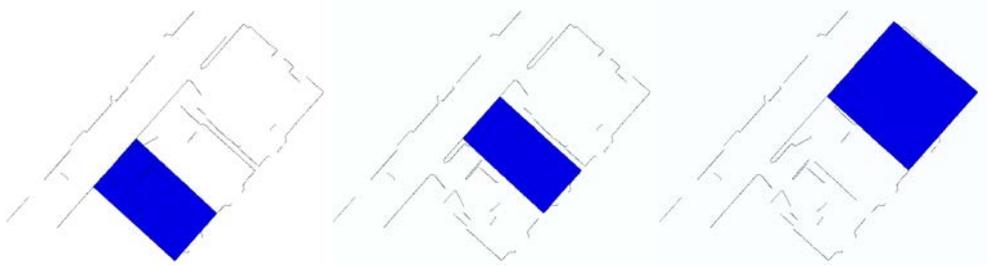


Abb. 8: Die graphbasierte Suche geschlossener Linienzüge ergibt als Endergebnis drei Räume.

4 Zusammenfassung

In diesem Beitrag wurde ein automatisches Verfahren zur Erstellung von Grundrissen mittels Laserscans vorgestellt. Die einzelnen Schritte, von der Aufnahme der Daten mit einem mobilen Roboter über die Registrierung bis hin zur eigentlichen Generierung des Grundrisses, werden dabei automatisch ausgeführt. Die 3D-Daten werden zunächst diskretisiert und auf eine 2D-Darstellung reduziert. Die so entstandene Pixelgrafik wird anschließend vektorisiert, um Linien zu erkennen und diese zu Räumen zu klassifizieren.

Im Beispieldatensatz erzeugt dieses Verfahren gute Ergebnisse. Zwar wird auch hier der kleine Flur zwischen den beiden Büros nicht separat erkannt und die Dicke der Wände wird ignoriert, dennoch liefern die gefundenen Räume eine Grundlage für weitere Berechnungen. Eine Möglichkeit ist, die Grundrissberechnung mit den Laserscans aus jedem der Räume separat zu wiederholen oder gezielt nach Linien parallel zu den Raumbegrenzungen zu suchen. Der Grundriss stellt die wesentlichen Elemente des Gebäudeteils in 2D dar. Überträgt man die enthaltenen Informationen nun wieder in die 3D-Darstellung, lassen sich die Ausmaße von Wänden, Decke und Boden extrahieren und so bereits ein einfaches Building Information Model (BIM) generieren. Der nächste Schritt ist dann die Bestimmung von Fenstern und Türen.

Danksagung

Diese Arbeit wurde gefördert aus dem DAAD/ZMOS Projekt “Creating thermal information models of existing buildings by active exploration” unter der Projekt-ID 57142988. Die Autoren bedanken sich bei Remus Dumitru, der Teile der Grundrisserstellung implementiert hat.

Literatur

- Blaer, P. S. & Allen, P. K. (2007): *Data acquisition and view planning for 3-D modeling tasks*. IEEE/RSJ International Conference on Intelligent Robots and, S. 417-422.
- Borrmann, D., Houshiar, H., Elseberg, J., & Nüchter, A. (2013): *Vom Kombinieren von 3D-Modellen mit Farb- und Temperaturinformationen*. Photogrammetrie Laserscanning Optische 3D-Messtechnik, Beiträge der Oldenburger 3D-Tage 2013, S. 246-253.
- Borrmann, D., Nüchter, A., Đakulovic, M., Maurovic, I., Petrovic, I., Osmankovic, D., & Velagic, J. (2014): *A mobile robot based system for fully automated thermal 3D mapping*. Advanced Engineering Informatics, 28(4), S. 425–440.
- Fernandes, L. A. F. & Oliveira, M. M. (2008): *Real-time line detection through an improved Hough transform voting scheme*. Pattern Recognition, 41, S. 299-314.
- Hough, P. V. C. (1962): *Method and Means for Recognizing Complex Patterns*. US Patent 3069654.
- Lowe, D. G. (1987): *Three-dimensional object recognition from single two-dimensional images*. Artificial Intelligence, 3(31), S. 355–395.
- Matas, J., Galambos, C., & Kittler, J. (1998): *Progressive Probabilistic Hough Transform*. Proceedings of the British Machine Vision Conference (1), S. 256–265.
- Nüchter, A., Elseberg, J., & Borrmann, D. (2013): *Irma3D – An Intelligent Robot for Mapping Applications*. Proceedings of the 3rd IFAC Symposium on Telematics Applications (3), Seoul, Korea.
- Nüchter A., Lingemann, K., Bormann, D., Elseberg, J., & Böhm, J. (2008): *Global Konsistente 3D-Kartierung mit Scanmatching*. Photogrammetrie Laserscanning Optische 3DMesstechnik, Beiträge der Oldenburger 3D-Tage 2008, S. 194-201.

Borrmann, Dorit
Julius-Maximilians-Universität Würzburg
borrmann@informatik.uni-wuerzburg.de

Leutert, Florian
Zentrum für Telematik, Würzburg
florian.leutert@telematik-zentrum.de

Maurović, Ivan
Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia
ivan.maurovic@fer.hr

Seder, Marija
Faculty of Electrical Engineering and Computing, University of Zagreb, Croatia
marija.seder@fer.hr

Nüchter, Andreas
Julius-Maximilians-Universität Würzburg
nuechter@informatik.uni-wuerzburg.de