

Eine voxelbasierte Metadatenstruktur zur Änderungsdetektion in Punktwolken großflächiger urbaner Gebiete

JOACHIM GEHRUNG^{1,2}, MARCUS HEBEL¹, MICHAEL ARENS¹ & UWE STILLA²

Zusammenfassung: Im Rahmen dieser Arbeit wird eine Methode vorgestellt, mit welcher sich eine volumetrische Approximation einer Punktwolke erstellen lässt. Weiterhin wird ein Verfahren zur Strahlverfolgung vorgestellt, welches einen auf Ebenen basierenden Filter einsetzt, um auf Grund der Diskretisierung des Raums entstandenen Artefakten an planaren Oberflächen entgegenzuwirken. Mittels beider Verfahren wird eine Metadatenstruktur aufgebaut, bei welcher es sich um eine effiziente Repräsentation der Umgebung handelt, in der Zusatzinformationen mit Ortsbezug abgelegt sind. Eine Evaluation beider Verfahren mit einem öffentlich verfügbaren Datensatz hat ergeben, dass sich aus geometrischer Sicht komplexe Strukturen abbilden lassen. Die Metadatenstruktur ist frei von besagten Artefakten. Die Ergebnisse der Untersuchung deuten darauf hin, dass eine gesonderte Behandlung von bewegten Objekten und Vegetation notwendig ist.

1 Einleitung

Die zuverlässige dreidimensionale Erfassung der Umgebung ist ein zunehmend wichtiger Bestandteil bei heutigen Mobile Mapping Systemen, beispielsweise im Hinblick auf Anwendungen wie die 3D-Rekonstruktion von Häuserfassaden für 3D-Stadtmodelle. Immer häufiger wird hierfür mobiles Laserscanning bzw. LiDAR eingesetzt (engl. *Light Detection and Ranging*). Im Gegensatz zu kamerabasierten photogrammetrischen Ansätzen und luftgebundenen 3D-Sensoren ist mobiles LiDAR dazu in der Lage, dichte und zugleich detailreiche Punktwolken urbaner Umgebungen zu erzeugen. Die hohe Qualität dieser Daten ist insbesondere für die automatische Änderungsdetektion attraktiv. Durch diese kann die Entwicklung einer urbanen Umgebung dokumentiert, automatisch ausgewertet und quantifiziert werden. Gerade im Hinblick auf statische Strukturen lassen sich nicht nur offensichtliche Änderungen wie abgerissene oder neu erbaute Häuser identifizieren, sondern auch eher subtile Details wie beschädigte oder fehlende Stadtmöbel.

1.1 Forschungsfragen

Änderungsdetektion in großräumigen urbanen Gebieten wirft im Wesentlichen zwei Forschungsfragen auf. Die erste betrifft die Aufbereitung der gesammelten Daten. Da der Fokus in dieser Arbeit auf baulichen und damit statischen Strukturen liegt, ist zu klären, wie sich bewegte Objekte aus den Punktwolken oder anderen Formen der Umgebungsrepräsentation entfernen lassen. Dies betrifft vor allem bewegte, aber auch temporär unbewegte Objekte. Beispiele für letzteres sind geparkte Fahrzeuge oder stillstehende Personen. Die Entfernung beider Objektklassen ist

¹ Fraunhofer-Institut für Optronik, Systemtechnik und Bildauswertung IOSB, Gutleuthausstraße 1, D-76275 Ettlingen, E-Mail: [joachim.gehrung, marcus.hebel, michael.arens]@iosb.fraunhofer.de

² Technische Universität München, Photogrammetrie und Fernerkundung, Arcisstr. 21, D-80333 München, E-Mail: stilla@tum.de

notwendig, da diese im Fall einer Änderungsdetektion als Änderungen erkannt werden, aber erwartbare Änderungen und damit ungewollte Detektionen darstellen.

Die zweite Forschungsfrage beschäftigt sich mit der Skalierung der Änderungsdetektion auf großflächige Gebiete. Aus methodischer Sicht ist zu klären, inwiefern Algorithmen und Datenstrukturen auch für große Areale skalieren. Datenstrukturen sind so zu konzipieren, dass sie möglichst speichereffizient sind und wenn möglich sogar eine fixe Speicherobergrenze aufweisen. Aus algorithmischer Sicht sind Methoden so zu entwerfen, dass sie möglichst laufzeiteffizient sind. Eine wichtige Rolle spielt hierbei auch die Möglichkeit zur Parallelisierung der Verfahren.

1.2 Lösungsansatz

Bei der Umgebungsrepräsentation durch Voxel handelt es sich um eine Datenstruktur, welche sowohl der Änderungsdetektion an sich als auch deren Anwendung auf große urbane Gebieten zuträglich ist. Eine solche volumetrische Repräsentation skaliert gut für große Umgebungen, da sie auf Grund der Diskretisierung des Raums nicht nur ähnliche Messdaten zusammenfasst, sondern auch eine fixe Speicherobergrenze aufweist. Für die Änderungsdetektion haben Volumen den Vorteil, dass hier neben reinen Oberflächeninformationen auch freie und bisher unbeobachtete Raumbereiche repräsentiert werden können. Diese Informationen spielen eine wichtige Rolle wenn es darum geht zu ermitteln, ob tatsächlich eine Änderung stattgefunden hat oder ob die potentielle Änderung daher rührt, dass der betroffene Raumbereich zum ersten Mal mit den Sensoren erfasst wurde. Im ersten Fall findet ein Zustandswechsel der betroffenen Voxel von frei nach belegt bzw. umgekehrt statt; im zweiten Fall erfolgt der Wechsel von ungesehen in einen beliebigen anderen Zustand. Für die volumetrische Repräsentation wird auf Methoden und Datenstrukturen aus unseren früheren Arbeiten zurückgegriffen, in denen eine ähnliche Datenstruktur bereits eingesetzt wurde (GEHRUNG et al. 2016).

1.3 Probleme volumenbasierter Repräsentationen

Der Nachteil eines auf Volumen basierten Ansatzes ist die Anfälligkeit für Artefakte, die durch die Diskretisierung des Raums bedingt sind (vergleiche Abb. 1). Durch diese kann es dazu kommen, dass Raumbereiche unterschiedlicher Natur wie z.B. Freiraum und belegter Raum im selben Voxel zusammengefasst werden. Ein Artefakt entsteht also, wenn ein Voxel von Sichtstrahlen durchlaufen wird, obwohl es auf Grund von anderen Observationen als belegt anzusehen ist. Hierdurch kommt es zu einem Aufweichen des Voxelzustands, da diesem durch die widersprüchlichen Beobachtungen nicht mehr eindeutig ein Label wie *frei* oder *belegt* zugeordnet werden kann. Diese sogenannten Traversionsartefakte treten an Grenzen zwischen Objekten und Freiraum auf und wurden in einer unserer vorangegangenen Arbeiten bereits thematisiert (GEHRUNG et al. 2017). Es ist zwischen zwei Arten dieser Artefakte zu unterscheiden. Die erste Kategorie tritt vereinzelt an Objektgrenzen auf. Eine stärker ausgeprägte Variante findet man vor allem an planaren Oberflächen. In flachem Winkel auftreffende Sichtstrahlen traversieren hier meist mehrere Voxel der Oberfläche, so dass es zu einer großflächigen Zustandsverfälschung kommt (siehe Abb. 1 (B)).

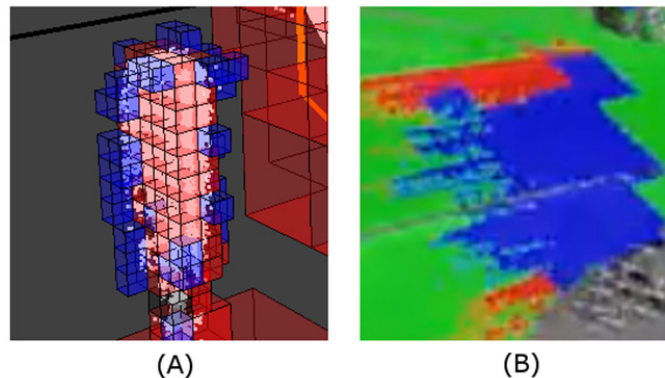


Abb. 1: Durch Diskretisierung bedingte Artefakte (A) an Objektgrenzen und (B) entlang planarer Flächen.

1.4 Beitrag dieser Arbeit

Diese Arbeit stellt einen weiteren Schritt in Richtung einer volumetrischen Repräsentation dar, die für die Änderungsdetektion in großen urbanen Gebieten geeignet ist. Es wird ein methodischer Ansatz vorgestellt, welcher eine volumetrische Approximation einer Punktwolke bzw. einer Menge an Sichtstrahlen ermöglicht, wobei sich die Auflösung der Volumina dynamisch an die lokale Geometrie anpasst. Weiterhin wird ein für diese Methode eingesetzter Algorithmus zur Strahlverfolgung (engl.: Raycasting) vorgestellt, welcher einen auf Ebenen basierten Filterschritt nutzt, der das Auftreten der oben beschriebenen Artefakte an planaren Oberflächen verhindert.

2 Stand der Forschung

Änderungsdetektion ist ein eigenständiger Forschungszweig, eine klare Abgrenzung zu anderen Bereichen ist jedoch nicht immer möglich. In eine ähnliche Richtung, jedoch mit einem weitaus kürzeren betrachteten Zeitfenster, geht beispielsweise *Detection and Tracking of Moving Objects (DATMO)*. Das Erkennen bewegter Objekte spielt vor allem im Forschungsumfeld von Robotik, autonomem Fahren und SLAM (*Simultaneous Localization and Mapping*) eine große Rolle, wenn auch mit unterschiedlichem Fokus. So entfernen z.B. LITOMISKY & BHANU (2013) mit einem clusterbasierten Ansatz bewegte Objekte aus Punktwolken, um damit die Performanz eines SLAM-Verfahrens zu verbessern. Viele Verfahren, die sich im weitesten Sinne mit Änderungsdetektion befassen, sind im Bereich des 2D-Maschinensehens angesiedelt. Da der Fokus dieser Arbeit jedoch auf der Verarbeitung von Punktwolken liegt, wird hier nur auf Verfahren mit eben diesem Schwerpunkt eingegangen.

2.1 Änderungsdetektion in Punktwolken

Auf gemessenen 3D-Punktwolken basierende Ansätze haben gegenüber der Verwendung von Bilddaten den Vorteil, dass sie schnell und effizient sind, da keine geometrische 3D-Rekonstruktion stattfinden muss und keine Verfälschung durch unterschiedliche Blickrichtungen eintritt. GIRARDEAU-MONTAUT et al. (2005) etwa organisieren Punktwolken räumlich in einem

Octree und stellen darauf aufbauend einige einfache Vergleichsverfahren für Teilpunktwolken vor. ZEIBAK & FILIN (2008) haben einen Ansatz zum Vergleich von Tiefenbildern vorgestellt, welche mittels eines stationären terrestrischen Laserscanners erzeugt wurden. Auf Grund der Einfachheit des Verfahrens ist nur der Vergleich von Aufnahmen möglich, die vom selben Standpunkt aus aufgenommen wurden. Beide Ansätze berücksichtigen jedoch keine Informationen über freien oder ungesehenen Raum.

2.2 Änderungsdetektion basierend auf Sichtstrahlen

Eine alternative Interpretation von Punktwolken sind Sichtstrahlen. Hierbei wird die Sensorposition als Ausgangspunkt des Strahls angenommen, der Messpunkt als Endpunkt. Durch die Verfolgung des Strahls lassen sich Informationen über freie und ungesehene Raumbereiche ableiten. UNDERWOOD et al. (2013) nutzen Strahlverfolgung in sphärischen Koordinaten, um einzelne Sensoraufnahmen im Sinne der Änderungsdetektion miteinander zu vergleichen. HEBEL et al. (2013) haben einen ebenfalls auf Sichtstrahlen basierten Ansatz zur Änderungsdetektion in LiDAR-Messungen fliegender Sensorträger vorgestellt. Einzelne Sichtstrahlen werden in einer voxelbasierten Index-Datenstruktur organisiert und mittels eines Evidenzmodells auf Basis der Dempster-Shafer Theorie miteinander verglichen. XIAO et al. (2015) nutzen einen ähnlichen Ansatz für das Auffinden von Änderungen in mobilen LiDAR-Daten. Ein Vorteil dieser Kategorie von Ansätzen ist es, dass die ursprünglichen Messinformationen nicht durch eine Diskretisierung reduziert werden. Dies hat jedoch zugleich den Nachteil, dass für eine Auswertung alle oder zumindest ein Teil der Sichtstrahlen herangezogen werden müssen. Für Änderungsdetektion in mobilen Laserdaten, die Millionen von Einzelmessungen pro Sekunde umfassen, ist dies im Hinblick auf Speicherbedarf und Laufzeit ein wenig praktikables Vorgehen.

2.3 Änderungsdetektion basierend auf Segmenten oder Objekten

Anstatt eine Änderungsdetektion auf den Messdaten an sich durchzuführen, kann auch eine vorgeschobene Vorverarbeitung der Punktwolken durchgeführt werden, beispielsweise durch Segmentierung oder Klassifikation. SCHACHTSCHNEIDER et al. (2017) nutzen eine Segmentierung basierend auf Region-Growing, um eine Punktwolke in Cluster zu unterteilen. Deren zeitliches Verhalten wird mittels eines Belegungsgitters ermittelt und ausgewertet. AIJAZI et al. (2013) setzen ebenfalls Segmentierung ein, klassifizieren dann allerdings die Cluster in Instanzen bekannter Objektklassen. Einige davon sind temporärer, andere permanenter Natur. Eine Ähnlichkeitskarte basierend auf einem Belegungsgitter wird eingesetzt, um Messdaten mehrerer Epochen zu fusionieren und eine dreidimensionale urbane Karte abzuleiten bzw. Änderungsdetektion durchzuführen. Diese Kategorie von Ansätzen setzt eine zuverlässig funktionierende Segmentierung bzw. Klassifikation voraus. Gerade letzteres stellt eine Einschränkung auf bekannte Objektklassen dar und somit auch eine Zwangsspezialisierung auf das jeweilige Anwendungsgebiet.

2.4 Änderungsdetektion in Belegungsgittern

Belegungs- bzw. Evidenzgitter approximieren einen Raum, indem dieser durch ein zwei- oder dreidimensionales Gitter in einzelne Zellen unterteilt wird. In den einzelnen Gitterzellen sind Informationen mit einem räumlichen Bezug abgelegt. PAGAC et al. (1996) setzen ein 2D-Belegungsgitter basierend auf der Dempster-Shafer Theorie zur Repräsentation des Umfeldes

eines autonomen Fahrzeugs ein. WOLF & SUKHATME (2004) stellen einen ähnlichen Ansatz vor, allerdings werden hier zwei Belegungsgitter eingesetzt, um statische und dynamische Umgebungselemente zu repräsentieren. Andere Ansätze nutzen einen populären dreidimensionalen Belegungs-Octree, welches unter der Bezeichnung OctoMap bekannt ist (HORNUNG et al. 2013). AZIM & AYCARD (2012) nutzen OctoMap, um bewegte Objekte in Echtzeit zu identifizieren. Hierzu wird eine Voxelliste kontinuierlich aktualisiert, indem eine Konfliktsuche zwischen der Umgebungsrepräsentation und den aktuellen LiDAR-Messungen durchgeführt wird. Die Voxel werden geclustert und gefiltert, um Objektkandidaten zu generieren. Das theoretische Rahmenwerk von OctoMaps wurde auch bei unseren vorangegangenen Arbeiten eingesetzt. Es wurde gezeigt, dass das Framework zur Identifikation bewegter Objekte eingesetzt werden kann, es hierbei jedoch zu den oben beschriebenen Artefakten kommt (GEHRUNG et al. 2016; GEHRUNG et al. 2017).

3 Volumenbasierte Approximation von Punktwolken

In diesem Abschnitt wird der Beitrag dieser Arbeit vorgestellt. Zunächst wird kurz auf das theoretische Rahmenwerk eingegangen, welches als Basis für die Umgebungsrepräsentation eingesetzt wird. Danach wird eine Methode zur Strahlverfolgung vorgestellt, welche einen auf Ebenen basierten Filterschritt nutzt, um die oben beschriebenen Artefakte an planaren Oberflächen zu verhindern. Darauf aufbauend wird ein Verfahren vorgeschlagen, mit dem sich eine volumetrische Approximation einer gegebenen Punktwolke aufbauen lässt.

3.1 Volumetrische Repräsentation

Die als Basis dieser Arbeit verwendete Metadatenstruktur basiert auf der in unseren vorangegangenen Arbeiten eingesetzten volumetrische Repräsentation (GEHRUNG et al. 2016). Räumliche Informationen werden in einem globalen kartesischen Koordinatensystem wie beispielsweise ECEF abgelegt. Mit Hilfe eines Gitters wird der Raum in würfelförmige Zellen zerlegt. Eine Zelle wird als *Kachel* bezeichnet und enthält neben den zugehörigen Sichtstrahlen auch eine volumetrische Repräsentation des eingeschlossenen Raums. Hierbei handelt es sich um einen Octree mit variabler Tiefe, der die effiziente Ablage von Volumen unterschiedlicher Auflösung ermöglicht. Jede Octree-Zelle beinhaltet Statistiken wie die Anzahl und Reihenfolge der Sichtstrahlen, die ein Voxel durchlaufen bzw. darin enden. Ebenfalls pro Zelle enthalten ist eine Tabelle mit Wahrscheinlichkeiten, welche die Zugehörigkeit eines Voxels zu einer Klasse wie *frei*, *belegt* oder *Restklasse* beschreibt. Aus Gründen von Datenorganisation und Visualisierung wurde in der praktischen Umsetzung die Kantenlänge einer Kachel auf 32 Meter festgelegt.

3.2 Strahlverfolgung mit Ebenenfilter

Die Absicht des vorgestellten Algorithmus zur Strahlverfolgung ist es, Diskretisierungsartefakte entlang planarer Strukturen zu vermeiden. Diese werden wie oben beschrieben durch Messstrahlen verursacht, die im flachen Winkel zur Oberfläche auftreffen. Die zu Grunde liegende Idee ist, parallel zu jeder planaren Struktur eine virtuelle Ebene zu definieren, an welcher der Sichtstrahl abgeschnitten wird. Unter Abschneiden ist hier zu verstehen, dass nur der zu traversierende Teil des Strahls gekürzt wird, jedoch nicht der Endpunkt entfernt wird. Um sicherzustellen, dass kein

Voxel traversiert wird, welches zu der planaren Struktur gehört, ist die Distanz zwischen beiden Ebenen die Diagonale eines Voxels. Da eine Ebene über eine unendliche Ausdehnung verfügt, ist sie nicht geeignet, um eine räumlich begrenzte Struktur zu repräsentieren. Aus diesem Grund wird anstelle einer Ebene eine orientierte Bounding Box (OBB) eingesetzt, welche als lokale Annäherung der Ebene fungiert. Besagte orientierte Bounding Box wird aus der Punktwolke berechnet. Durch Expansion der Box um die Voxeldiagonale kann der Schnitt mit der o.g. virtuellen Ebene als Schnittberechnung zwischen Sichtstrahl und der orientierten Bounding Box formuliert werden.

3.2.1 Extraktion planarer Oberflächen

Die Extraktion der planaren Oberflächen aus der Punktwolke erfolgt bei unserem Ansatz durch einen auf Normalen basierten Region-Growing-Algorithmus, der von RABBANI et al. (2006) vorgestellt wurde. Hierbei werden benachbarte Punkte zu Clustern zusammengefasst, wenn sie sich in Bezug auf ein Ähnlichkeitsmaß gleichen. Bei diesem handelt es sich um den Winkel zwischen den Oberflächennormalen an den Positionen beider Punkte. Ausgangspunkte für den Algorithmus werden in Regionen mit minimaler Krümmung ausgewählt, da so die Anzahl möglicher Cluster reduziert wird. Da ebenfalls eine Vielzahl kleiner Oberflächen gefunden wird, wurde ein zusätzlicher Filterschritt angefügt. Hierzu wird eine Liste der Cluster erstellt, die nach der Clustergröße – also der Anzahl der Messpunkte im Cluster – sortiert wird. Das untere Viertel der Liste wird verworfen, nur die verbliebenen Cluster werden für das Ebenenfilter verwendet.

3.2.2 Berechnung der orientierten Bounding Boxen

Eine orientierte Bounding Box approximiert eine Punktwolke lokal besser als eine achsenparallele Bounding Box (AABB), ist jedoch aufwendiger zu berechnen. OBBs werden hier erzeugt, indem zunächst mit Hilfe der Hauptkomponentenanalyse das lokale Koordinatensystem der Punktwolke berechnet wird. Die Eigenvektoren sowie der Schwerpunkt des Clusters werden dazu benutzt, die Transformation in das Koordinatensystem zu berechnen und alle Punkte zu transformieren. Der letzte verbliebene Schritt ist nun die Bestimmung der achsenparallelen Bounding Box. Da die Punktwolke lokal nicht immer vollständig planar ist, wird die Achse der Box mit dem kleinsten Eigenwert hier (vorläufig) auf null gesetzt. Die achsenparallele Bounding Box wird zusammen mit der Transformation gespeichert, um später den Schnitttest durchzuführen.

3.2.3 Strahlverfolgung mit Ebenenfilter

Die Transformation einer Punktwolke in die in dieser Arbeit verwendete volumetrische Repräsentation erfolgt durch Strahlverfolgung. Hierzu wird ein Sichtstrahl von der Sensorposition bis hin zum gemessenen Oberflächenpunkt traversiert. Der durchlaufene Bereich wird als Freiraum vermerkt, der Endpunkt des Strahls als belegter Raum (oder ebenfalls als Freiraum, wenn der Strahl an der Kachelgrenze abgeschnitten wurde). Die entsprechenden Voxel werden aktualisiert, indem für jeden der beiden Fälle ein Zähler inkrementiert wird. Die Strahlverfolgung mit Ebenenfilter ist ein zweistufiger Prozess:

1. Kürzen des Sichtstrahls durch Schnitt mit allen Bounding Boxen.
2. Traversal des Freiraums und behandeln der Endpunkte.

Um nun die Traversalion von Voxeln zu verhindern, die Teil von planaren Oberflächen sind, muss jeder Sichtstrahl zunächst mit allen Bounding Boxen geschnitten werden. Dies setzt voraus, dass der Anfangs- und Endpunkt des Strahls in das Koordinatensystem der orientierten Bounding Box transformiert werden. Hier findet dann ein Schnitttest mit einer achsenparallelen Bounding Box statt, beispielsweise der von WILLIAMS et al. (2003) vorgestellte. Bevor der Schnitttest durchgeführt wird, wird zunächst die (zuvor noch flache) Box um die Diagonale eines Voxels erweitert. Dies kann nicht bei der Extraktion der Box geschehen, da die jeweilige Voxelgröße und damit die benötigte Erweiterung der Bounding Box erst zum Zeitpunkt des Schnitttests bekannt sind. Liegt ein Schnitt vor, so liefert das Verfahren einen Offset t entlang des Strahls, welcher den Schnitt mit der Box angibt. Dieser wird dazu verwendet, den neuen Endpunkt entlang des untransformierten Strahls zu bestimmen.

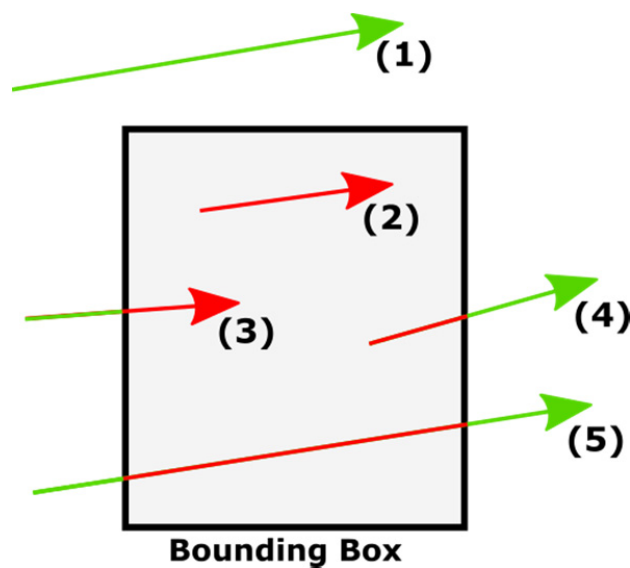


Abb. 2: Sonderfälle beim Schnitt zwischen Strahl und Box (1) Kein Schnitt. (2) Strahl komplett innerhalb. (3) Strahl beginnt außerhalb, endet innerhalb. (4) Strahl beginnt innerhalb, endet außerhalb. (5) Strahl startet und endet außerhalb, traversiert die Box.

Beim Schnitt zwischen Strahl und Bounding Box kann es zu mehreren Sonderfällen kommen (vergleiche Abb. 2). Die folgende Aufzählung erläutert die einzelnen Fälle und gibt an, wie mit diesen umzugehen ist.

1. Es liegt kein Schnitt zwischen Sichtstrahl und Box vor. Leerraum wird traversiert, der Endpunkt wird eingefügt.
2. Der Sichtstrahl liegt komplett innerhalb der Box. Keine Traversalion des Freiraums, der Endpunkt wird nur eingefügt, wenn er als belegt gilt.
3. Der Sichtstrahl startet außerhalb der Box und endet innerhalb. Nur der Teil des Strahls, der sich außerhalb der Box befindet, wird traversiert. Nur ein als belegt geltender Endpunkt wird eingefügt.
4. Der Strahl startet in der Box und endet außerhalb. Nur der Teil des Sichtstrahls außerhalb der Box wird traversiert. Der Endpunkt wird eingefügt; hierbei spielt es keine Rolle, ob er frei oder belegt ist.

5. Der Strahl startet und endet außerhalb der Box, durchläuft diese jedoch. Der erste Teil des Strahls wird traversiert, es wird jedoch kein Endpunkt eingefügt. Der Rest des Strahls wird der Liste noch zu bearbeitender Sichtstrahlen hinzugefügt.

Die Schnitte werden fusioniert, indem die Offsets $t_{start} = 0.0$ und $t_{end} = 1.0$ für Anfang und Ende des neuen Sichtstrahls angepasst werden. Dies erfolgt über die beiden Offsets t_{near} und t_{far} , welche den Schnitt des Strahls mit der nahen und fernen Seite der Bounding Box beschreiben. Beide Variablen werden vom Schnitttest geliefert und dienen unter anderem dazu, die oben genannte Fallunterscheidung durchzuführen. Folgende Aufzählung gibt wieder, wie der Strahl in den oben genannten Fällen angepasst wird:

1. Kein Schnitt, daher keine Anpassung notwendig.
2. Der Sichtstrahl ist komplett in der Bounding Box und wird nicht traversiert. Dies ist ein Abbruchkriterium für das Prüfen gegen weitere Ebenen.
3. Wenn $t_{near} < t_{end}$, dann $t_{end} = t_{near}$. Hierdurch wird sichergestellt, dass ein Sichtstrahl nur zum Start hin verkürzt wird.
4. Wenn $t_{far} > t_{start}$, dann $t_{start} = t_{far}$. Hierdurch wird sichergestellt, dass ein Sichtstrahl nur zum Endpunkt hin verkürzt wird.
5. Siehe Fall 3.

Nach der Kürzung des Sichtstrahls findet die eigentliche Strahlverfolgung statt. Der verbliebene Freiraum wird durchlaufen und die Volumen werden aktualisiert. Im Anschluss wird der freie oder belegte Endpunkt eingefügt; je nachdem, ob die Fallunterscheidung dies für den jeweiligen Typ des Endpunktes zulässt oder nicht.

3.3 Iterative Verfeinerung

Die im vorherigen Abschnitt beschriebene Methode zur Strahlverfolgung mit Ebenenfilter wird hier dazu eingesetzt, eine volumetrische Approximation einer gegebenen Punktwolke aufzubauen. Diese ist frei von Traversionsartefakten an planaren Oberflächen. Das Verfahren ist in eine Initialisierungsphase und eine iterative Verfeinerungsphase unterteilt.

Im Rahmen der Initialisierung wird eine Volumenrepräsentation der Umgebung mit grober, einheitlicher Auflösung aufgebaut. Für ein urbanes Umfeld hat sich eine Voxelgröße von einem Meter als praktikabel erwiesen, da diese Auflösung zum einen grob genug ist, um eine schnelle Laufzeit zu gewähren, zum anderen exakt genug, um die Anzahl der nötigen Verfeinerungsschritte zu minimieren. Die Minimierung der benötigten Verfeinerungsschritte ist auch der Grund, warum eine solche Initialisierung stattfindet. Eine Auflösung von einem Meter lässt sich erzielen, indem bei einer Kachel-Kantenlänge von 32 m die Tiefe 5 im Octree gewählt wird.

Die iterative Verfeinerungsphase findet für jedes Voxel einzeln statt. Hierzu wird das zu verfeinernde Voxel einer Queue zugewiesen. Diese wird solange abgearbeitet, bis sie leer ist; dann gilt das Voxel als maximal verfeinert. Zu Beginn einer Iteration wird das nächste Voxel aus der Queue entnommen. Sind weniger als eine Mindestanzahl an Beobachtungen (d.h. traversierende und darin endende Messstrahlen) darin enthalten, so wird dem Voxel das Label *Restklasse* zugewiesen und keine weitere Verfeinerung durchgeführt. Dieser Schwellwert ist notwendig, da der Voxelzustand umso robuster geschätzt werden kann, je mehr Beobachtungen vorliegen.

Liegt eine Mindestanzahl an Beobachtungen vor und die maximale Tiefe im Octree wurde noch nicht erreicht, dann wird für das Voxel eine naive Zustandsschätzung durchgeführt. Wird das

Voxel nur von Sichtstrahlen durchlaufen, dann wird ihm das Label *frei* zugewiesen. Enden hingegen nur Sichtstrahlen im Voxel, dann kann man davon ausgehen, dass es belegt ist. Da belegte Voxel jedoch weiter zu unterteilen sind, da es sonst nur zu einer sehr groben Annäherung der Punktwolke kommt, wird das Label *aufteilen-belegt* zugewiesen. Tritt der Fall auf, dass ein Voxel von Sichtstrahlen durchlaufen wird und zugleich andere Sichtstrahlen darin enden, dann gehört das Voxel entweder zum Rand einer Struktur oder eines Objekts. Es wird mit dem Label *aufteilen* versehen. Hat das Voxel nach Abschluss der naiven Zustandsschätzung entweder das Label *aufteilen* oder *aufteilen-belegt*, wird es in seine Kinds-knoten aufgespaltet. Für jeden dieser Knoten wird eine ebenen-gefilterte Strahlverfolgung durchgeführt. Die Laufzeit des Verfahrens lässt sich rapide senken, indem dafür nur die Sichtstrahlen herangezogen werden, welche bereits das übergeordnete Voxel durchlaufen haben. Anschließend werden die Kinds-knoten zwecks weiterer Verfeinerung der Queue hinzugefügt.

Wird ein Voxel der Liste entnommen und dessen Tiefe im Octree entspricht der maximalen Tiefe, dann wird ein finaler Labeling-Schritt vorgenommen. Voxel mit dem Label *aufteilen-belegt* wird das Label *belegt* zugewiesen. Bei Voxeln mit dem Label *aufteilen* kann es sich entweder um stationäre Strukturen handeln, um temporär stationäre Objekte wie geparkte Fahrzeuge, um von einem bewegten Objekt durchlaufenen Freiraum oder aber um Voxel an Objekträndern. Da die korrekte Zuweisung eines Labels keinesfalls trivial ist, wird diese Aufgabe als Gegenstand zukünftiger Arbeiten betrachtet.

4 Experimente

4.1 Verwendeter Datensatz

Die Evaluation der hier beschriebenen Verfahren wird auf Basis des *MLS1 – TUM City Campus* Datensatzes durchgeführt. Dieser wurde im Jahr 2016 mit dem Messfahrzeug MODISSA des Fraunhofer-Instituts für Optronik, Systemtechnik und Bildauswertung (IOSB) aufgezeichnet. Der Datensatz umfasst ca. 2.25 Milliarden Einzelmessungen zweier mobiler LiDAR-Sensoren, verteilt über eine Fläche von 0,17 km². Aufgenommen wurden das Stammgelände der TU München sowie der umliegende Straßenraum.

Die Aufnahmen wurden mit zwei Velodyne HDL-64E LiDAR-Sensoren durchgeführt, die in einem Winkel von 35° auf dem Dach im vorderen Bereich des Fahrzeugs montiert sind. Die Georeferenzierung der Daten basiert auf den Messungen eines Applanix POS LV Navigationssystems, welches zwei GPS-Antennen, sowie eine IMU und einen Raddrehgeber einsetzt. Die Navigationsdaten wurden nachprozessiert, um die Genauigkeit zu verbessern. Der Datensatz ist unter einer Creative Commons Lizenz³ frei verfügbar.

Für die Evaluation wurde die Kreuzung von Theresienstraße und Luisenstraße nordwestlich des TUM Stammgeländes ausgewählt. Die Kreuzung grenzt an mehrere mehrstöckige Gebäude, Vegetation in Form von Bäumen ist ebenfalls vorhanden. Im Straßenraum befinden sich Verkehrszeichen und Ampeln, Fußgänger und PKWs nehmen am Verkehr teil. Das Messfahrzeug kommt

³ <http://s.fhg.de/mls1>

von der Theresienstraße, wartet einige Sekunden an der Kreuzung und biegt dann in die Luisenstraße ab.

4.2 Qualitative Evaluation

Die qualitative Evaluation wird mittels manueller Überprüfung durch einen Beobachter anhand zuvor festgelegter Kriterien durchgeführt. Für dieses Vorgehen gibt es mehrere Gründe. Im Fall der volumetrischen Approximation ist die praktikabelste Möglichkeit nur der Vergleich mit einer bereits bestehenden volumetrischen Repräsentation. Zwar lässt sich prüfen, ob Messpunkte sich innerhalb eines als belegt gekennzeichneten Voxels befinden, für Freiraum jedoch ist dies nicht so ohne weiteres möglich. Beim Vergleich mit einer anderen volumetrischen Repräsentation tut sich zudem die Frage nach dem Grad der Diskretisierung auf, da es für dessen Wahl kein objektives Kriterium gibt. Im Fall der Strahlverfolgung mit Ebenenfilter ist eine Grundwahrheit notwendig, in der planare Flächen oder zumindest Gebäude und ebene Flächen gekennzeichnet sind. Fraglich ist hier jedoch, ab welcher Größe Flächen von Relevanz sind. In beiden Fällen ist es nicht so ohne weiteres möglich, eine objektive Grundwahrheit zu erstellen. Am praktikabelsten erscheint es also, beide Verfahren an einer möglichst komplexen Szene zu testen und dann eine manuelle Überprüfung anhand zuvor festgelegter Kriterien durchzuführen. Für die volumetrische Interpretation wurde folgendes Kriterium definiert:

- Wie gut ist die Approximation der Punktwolke durch die Volumen?

Die Evaluation der Strahlverfolgung mit Ebenenfilter wurde anhand der folgenden Kriterien durchgeführt:

- Welchen Einfluss hat die Strahlverfolgung mit Ebenenfilter auf die Repräsentation?
- Wurden alle relevanten planaren Oberflächen erkannt; d.h. fehlen Oberflächen oder wurden unerwünschte Oberflächen erkannt?
- Wurden planare Oberflächen im Hinblick auf die Segmentierung der Ebenen und Schätzung der Bounding Boxen richtig approximiert?

Die obigen Fragen werden beantwortet durch die Betrachtung einer farbkodierten Volumenrepräsentation in Kombination mit der akkumulierten Punktwolke, aus welcher erstere erstellt wurde. Neben den oben genannten Kriterien wurde auf sonstige Besonderheiten geachtet.

5 Diskussion der Ergebnisse

Qualität der Approximation – Das Ergebnis der volumetrischen Approximation für die im letzten Abschnitt beschriebene Beispielszene wird in Abb. 3 (C) gezeigt. Die Szene enthält viele Objekte, Vegetation und Strukturen und kann damit aus geometrischer Sicht also als komplex angesehen werden. Wie sich erkennen lässt sind auch feine geometrische Details gut erkennbar aufgelöst. Selbst Objekte wie beispielsweise Ampeln und Straßenschilder lassen sich in der volumetrischen Darstellung klar erkennen. Ebenfalls als solche zu erkennen sind Bäume und bewegte Verkehrsteilnehmer, wie etwa der Radfahrer in der Mitte der Kreuzung.

Einfluss der Strahlverfolgung – Um die Vorzüge der Strahlverfolgung mit Ebenenfilter zu demonstrieren wurde dieselbe Szene einmal mit und einmal ohne das Verfahren aufgebaut. Das Resultat kann in Abb. 3 (A, B) betrachtet werden. Im ersten Fall ist zu erkennen, dass ein Groß-

teil der belegten Voxel durch Messstrahlen traversiert wurde. Entsprechende Voxel sind gelb gekennzeichnet. Hierbei handelt es sich um die zu Beginn beschriebenen Traversionsartefakte an planaren Oberflächen. Die Anwendung der Strahlverfolgung mit Ebenenfilter führt dazu, dass alle Voxel einer korrekt identifizierten planaren Oberfläche als belegt erkannt werden. Eine Ausnahme bilden die grau gekennzeichneten Voxel der Restklasse, für die keine ausreichende Menge an Messpunkten vorliegt.

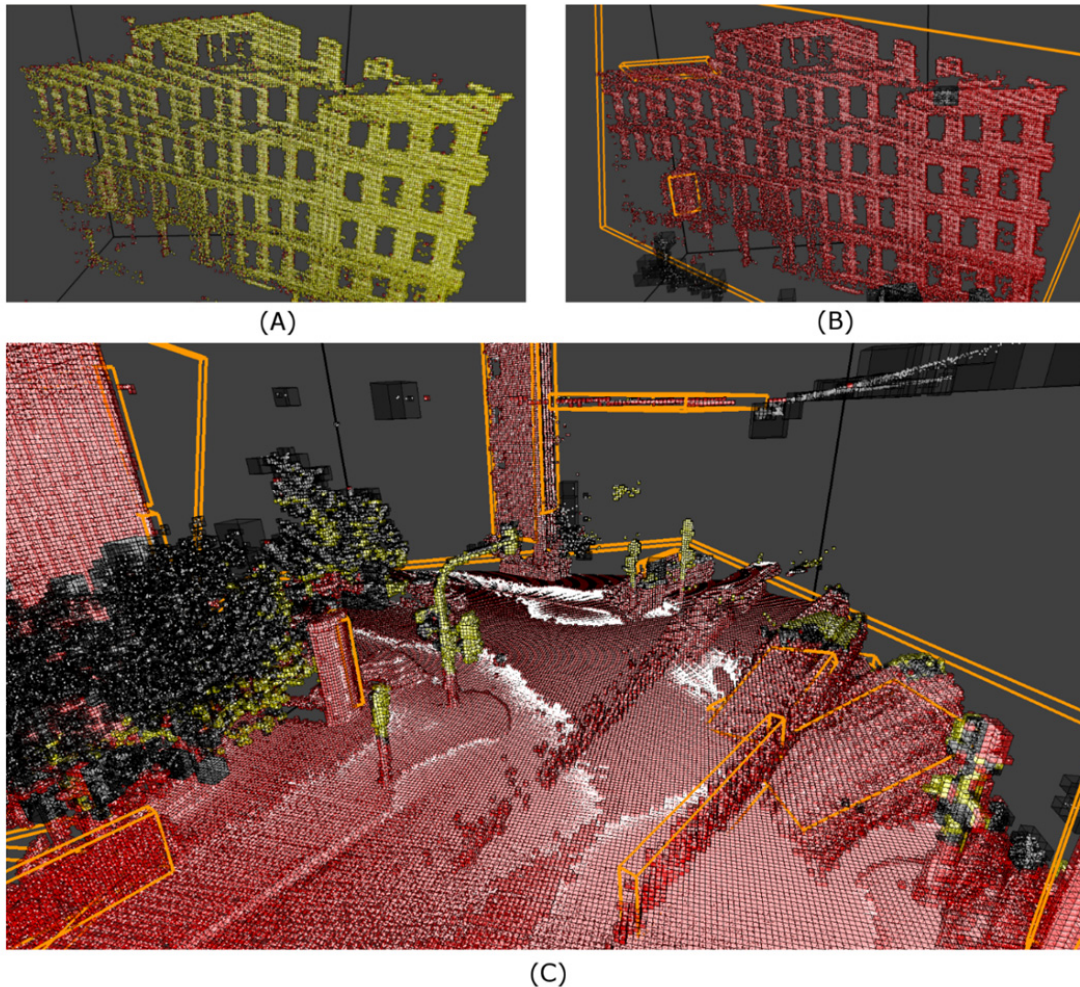


Abb. 3: Volumetrische Approximationen. (A) Ohne und (B) mit Strahlverfolgung mit Ebenenfilter. (C) Eine Beispielszene (rot=belegt, gelb=zu klassifizieren, grau=Restklasse)

Segmentierung und Approximation planarer Oberflächen – Die Methode zur Erkennung planarer Oberflächen ist dazu in der Lage, die für diese Arbeit relevanten dominanten Ebenen wie die Bodenebene und Gebäudefassaden zu identifizieren (vergleiche Abb. 4). Anomalien zeigen sich jedoch an sehr kleinen Flächen und an Ebenen, die nur aus wenigen Punkte bestehen. In einigen seltenen Fällen kommt es auch vor, dass in beliebigen Punktklustern fälschlicherweise Ebenenkandidaten identifiziert werden. Gelegentlich kommt es vor, dass Ebenen in bewegten Objekten erkannt werden. Da es sich bei der zu Grunde liegenden Datenbasis um akkumulierte

Einzelmessungen handelt, ziehen bewegte Objekte eine Art Schweif hinter sich her. Gerade bei PKWs kann die planare Oberfläche des Dachs dazu führen, dass diese auf Grund ihrer großen räumlichen Ausdehnung und des Schweifs als vielversprechender Kandidat erkannt wird.

In einigen wenigen Fällen sind Bounding Boxen nicht korrekt orientiert. Dies lässt sich nicht mit dem populären Fehler zu begründen, dass die Eigenvektoren auf Grund von Eigenarten des Eigenwert-Lösers kein rechtshändiges Koordinatensystem formen. Es wird angenommen, dass sich dieses Verhalten durch die Hauptkomponentenanalyse in Kombination mit der vorliegenden Punktwolke begründen lässt. Das Verhalten kann korrigiert werden, indem weitere Vorverarbeitungsschritte eingefügt werden oder eine Alternative zur Hauptkomponentenanalyse gefunden wird.

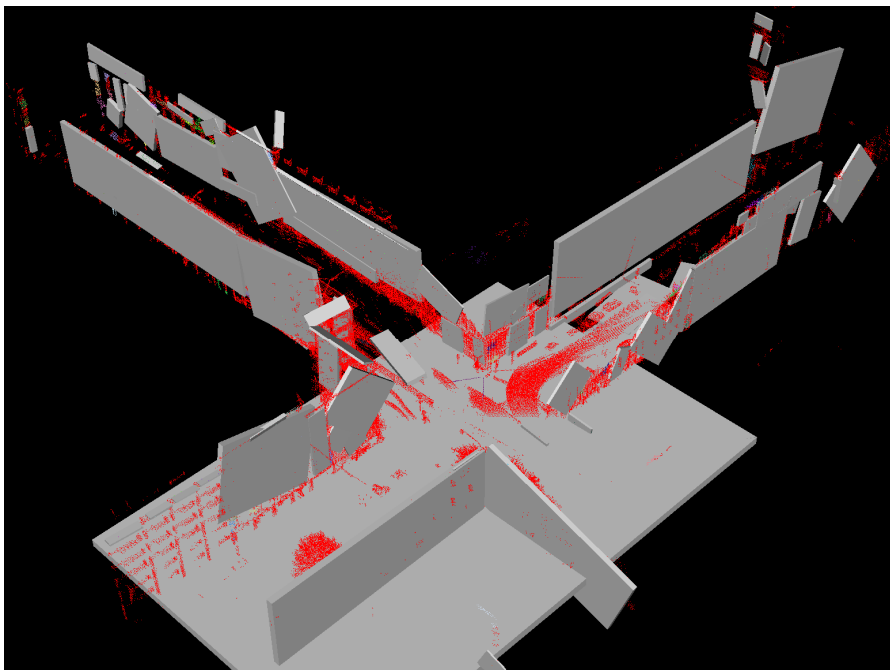


Abb. 4: Planare Oberflächen (grau), extrahiert aus der Punktwolke (rot). Ein Großteil aller planaren Oberflächen ist vorhanden und korrekt approximiert. Einige Bounding Boxen sind nicht korrekt ausgerichtet.

Sonstige Anomalien – Abb. 3 (C) zeigt, dass Traversionsartefakte an Objektgrenzen nach wie vor ein Problem darstellen. Ähnliches gilt auch für Bäume. Derselben Abbildung lässt sich entnehmen, dass diese entweder vom selben Artefakt betroffen sind oder auf Grund zu weniger Einzelmessungen der Restklasse zugewiesen wurden. Weiterhin ist anzumerken, dass bewegte Objekte auf Grund ihrer Nähe zur Bodenebene komplett oder teilweise als solide Struktur erkannt werden. Dies legt den Schluss nahe, dass für Vegetation und bewegte Objekte eine Sonderbehandlung notwendig ist.

6 Fazit & Ausblick

In dieser Arbeit wurde ein Verfahren zum Aufbau einer zur Änderungsdetektion geeigneten, auf Voxeln basierten Metadatenstruktur vorgestellt, bei dem eine volumetrische Approximation einer Punktwolke unter Nutzung einer Methode zur Strahlverfolgung mit Ebenenfilter durchgeführt wird. Letztere ermöglicht es, durch die Diskretisierung des Raums verursachte Artefakte an planaren Oberflächen zu verhindern. Eine Auswertung der Ergebnisse beider Verfahren hat ergeben, dass sich aus geometrischer Sicht komplexe Strukturen durch Volumen annähern lassen. Wichtige planare Oberflächen in der volumetrischen Repräsentation wie die Bodenebene und Häuserfassaden sind frei von Diskretisierungsartefakten. An Objektgrenzen zeigen sich wie erwartet weiterhin Artefakte. Dies und auch die Tatsache, dass gelegentlich planare Oberflächen in bewegten Objekten erkannt werden, legt eine Sonderbehandlung für diesen Fall nahe. Selbiges gilt für Vegetation. Detektiert man beide Objektklassen bereits vor Aufbau der Volumenrepräsentation in den Messdaten, so kann man diese gesondert einpflegen. Dies vereinfacht den Umgang mit den oben beschriebenen Artefakten deutlich.

7 Literaturverzeichnis

- AIJAZI, A.K., CHECCHIN, P. & TRASSOUDAIN, L., 2013: Detecting and Updating Changes in Lidar Point Clouds for Automatic 3D Urban Cartography. *ISPRS - Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, **II-5/W2**, 7-12.
- AZIM, A. & AYCARD, O., 2012: Detection, classification and tracking of moving objects in a 3D environment. *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, 802-807.
- GEHRUNG, J., HEBEL, M., ARENS, M. & STILLA, U., 2016: A Framework for Voxel-based Global Scale Modeling of Urban Environments. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, **42(2/W1)**, 45-51.
- GEHRUNG, J., HEBEL, M., ARENS, M. & STILLA, U., 2017: An approach to extract moving objects from MLS data using a volumetric background representation. *ISPRS - Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, **IV-1/W1**, 107-114.
- GIRARDEAU-MONTAUT, D., ROUX, M., MARC, R. & THIBAUT, G., 2005: Change detection on point cloud data acquired with a ground laser scanner. *ISPRS - International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, **36(3/W19)**, 30-35.
- HEBEL, M., ARENS, M. & STILLA, U., 2013: Change detection in urban areas by object-based analysis and on-the-fly comparison of multi-view ALS data. *ISPRS Journal of Photogrammetry and Remote Sensing*, **86**, 52-64.
- HORNUNG, A., WURM, K.M., BENNEWITZ, M., STACHNISS, C. & BURGARD, W., 2013: OctoMap: An Efficient Probabilistic 3D Mapping Framework Based on Octrees. *Autonomous Robots*, **34**, 189-206.
- LITOMISKY, K. & BHANU, B., 2013: Removing Moving Objects from Point Cloud Scenes. *Advances in Depth Image Analysis and Applications*, Springer Berlin Heidelberg, 50-58.
- PAGAC, D., NEBOT, E.M. & DURRANT-WHYTE, H., 1996: An evidential approach to probabilistic map-building. *Proceedings of IEEE International Conference on Robotics and Automation*, **1**, 745-750.

- RABBANI, T., VAN DEN HEUVEL, F.A. & VOSSELMAN, G., 2006: Segmentation of Point Clouds Using Smoothness Constraint. *ISPRS: Proceedings of the ISPRS Commission V Symposium*, **35**, 248-253.
- RABINER, L. R., 1989: A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, **77**, 257-286.
- SCHACHTSCHNEIDER, J., SCHLICHTING, A. & BRENNER, C., 2017: Assessing temporal behavior in LIDAR point clouds of urban environments. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, **42**(1/W1), 543-55.
- UNDERWOOD, J. P., GILLSJÖ, D., BAILEY, T. & VLASKINE, V., 2013: Explicit 3D change detection using ray-tracing in spherical coordinates. *IEEE International Conference on Robotics and Automation*, 4735-4741.
- WILLIAMS, A., BARRUS, S., KEITH, R. & SHIRLEY, M.P., 2003: An efficient and robust ray-box intersection algorithm. *Journal of Graphics Tools*, **10**, 54.
- WOLF, D. & SUKHATME, G.S., 2004: Online simultaneous localization and mapping in dynamic environments. *2004 IEEE International Conference on Robotics and Automation*, **2**, 1301-1307.
- XIAO, W., VALLET, B., BRÉDIF, M. & PAPANODITIS, N., 2015: Street environment change detection from mobile laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing, Elsevier*, **38**, 38-49.
- ZEIBAK, R. & FILIN, S., 2008: Change detection via terrestrial laser scanning. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, **36**, 430-435.