Universität Osnabrück; Universität Heidelberg

WEBBASIERTE SICHTBARKEITSANALYSE MIT LASERSCANNINGDATEN

Florian Hillen, Bernhard Höfle

Zusammenfassung: In diesem Beitrag werden zwei Ansätze zur Berechnung der direkten Sichtbarkeit auf unregelmäßigen Dreiecksnetzen (TINs) vorgestellt und in eine serviceorientierte Architektur integriert. Der erste von zwei entwickelten Ansätzen basiert auf der räumlichen Indexstruktur des R-Baums und einer schrittweisen Bearbeitung der direkten Sichtlinie. Diese Vorgehensweise liefert zwar korrekte Ergebnisse, ist jedoch mit einem hohen Berechnungsoverhead behaftet und somit nicht für größere Untersuchungsgebiete mit großen Sichtdistanzen geeignet. Der zweite Ansatz basiert hingegen auf einer topologischen Speicherung der TIN-Dreiecke. Aufgrund der daraus resultierenden effizienteren Prozessierung der direkten Sichtlinie sind die Berechnungen bis zu 10-mal schneller als der R-Baum-Ansatz. Die Integration der Berechnungen in eine serviceorientierte Architektur kann durch die Verwendung eines Web Processing Services (WPS) in Kombination mit einem zwischengeschalteten XML-RPC-Server zur performanten Handhabung des hohen Datenvolumens realisiert werden. In diesem Beitrag wird somit gezeigt, dass die on-the-fly-Berechnung der Sichtbarkeit auch auf hochgenauen Laserscanningdaten prinzipiell möglich ist, was unter anderem für Location-based Services sowie eine Vielzahl anderer räumlicher Anwendungen von besonderer Wichtigkeit ist.

Schlüsselwörter: 3D-Analyse, Sichtbarkeitsanalyse, Web Processing Service (WPS), Interoperabilität, Laserscanning, Location-based Services (LBS)

WEB-BASED VISIBILITY ANALYSIS WITH LIDAR DATA

Abstract: This work presents two approaches to calculate the direct visibility on triangulated irregular networks (TINs) and their integration into a service-oriented architecture. The first introduced approach is based on the R-tree, a spatial index structure, and the incremental processing of the line-of-sight. This procedure delivers correct results but is afflicted with a high calculation overhead and thus not appropriate for larger investigation areas with high view distances. The second approach on the other hand is based on a topological storage of the TIN triangles. Due to the more efficient line-of-sight processing, the calculations are up to 10 times faster than the R-tree approach. The integration of the algorithms into a service-oriented architecture is realized using a Web Processing Service (WPS) in combination with an intercalated XML-PRC server for the efficient management of the high data volumes. Hence, this work shows the possibility of the on-the-fly visibility calculation on highly accurate laser scanning data, which is of particular importance regarding Location-based Services as well as a multiplicity of other spatial application.

Keywords: 3D analysis, line-of-sight analysis, Web Processing Service (WPS), interoperability, laser scanning, location-based services (LBS)

Autoren

M. Sc. Florian Hillen
Universität Osnabrück
Institut für Geoinformatik und Fernerkundung
Barbarastraße 22b
D-49076 Osnabrück
E: florian.hillen@igf.uni-osnabrueck.de

Jun.-Prof. Dr. Bernhard Höfle
Universität Heidelberg
Geographisches Institut
Berliner Straße 48
D-69120 Heidelberg
E: bernhard.hoefle@geog.uni-heidelberg.de

1 EINLEITUNG

In den vergangenen Jahren hat sich das Laserscanning zu einer Standardmethode zur Gewinnung von Höhendaten der Geländeoberfläche entwickelt (Shan & Toth 2008). Dieses liegt vor allem an der direkten Höhenmessung durch Laserstrahlen-Laufzeitmessung sowie an der Möglichkeit, den Boden unter natürlichen Objekten (z. B. Bäume) erfassen zu können. Laserdaten bieten zudem ein hohes Potenzial für mobile Anwendungen wie Navigationssysteme oder insbesondere für Location-based Services (LBS), da hierfür 3D-Analysen eine immer wichtigere Rolle spielen und für die daher eine sehr genaue 3D-Datengrundlage benötigt wird (Wang & Wu 2009, Carswell et al. 2010, Trapp et al. 2011). Wichtig für jede mobile Anwendung ist jedoch eine geringe Wartezeit auf das Ergebnis einer Anfrage für den Benutzer. Diese Anforderung steht jedoch in Konflikt mit dem hohen Speicheraufkommen der hochvolumigen Laserdaten und dem daraus resultierenden rechenintensiven Analyseaufwand. Für mobile Anwendungen werden daher effiziente Algorithmen und Strukturen benötigt, die 3D-Analysen serverbasiert ausführen und deren Ergebnisse bereitstellen können. Wichtig für die Einbindung und Kopplung der 3D-Laserdatenanalyse mit weiteren Anwendungen ist die Bereitstellung eines interoperablen Webdiensts, was auf Basis eines OGC Web Processing Services (WPS) durchgeführt werden kann (OGC 2007).

Im Zuge dieses Beitrags soll somit die Forschungsfrage untersucht werden, mit welchen Methoden und Konzepten hochgenaue 3D-Daten in serviceorientierte Architekturen eingebunden und effektiv genutzt werden können. Anhand einer sehr wichtigen 3D-Analyse, der Sichtbarkeitsanalyse, soll das Potenzial der webbasierten 3D-Analyse mit hochvolumigen Laserscanningdaten für die Verwendung in zum Beispiel mobilen Anwendungen analysiert werden. Traditionelle Ansätze, die ihre Berechnungen auf rasterbasierten Digitalen Höhenmodellen durchführen, sind in der Regel nur für kurze Berechnungsdistanzen und geringe Datenauflösungen effizient einsetzbar. Für Rasterauflösungen von 1 m und besser liefern diese Ansätze jedoch keine Ergebnisse in einem angemessenen Zeitraum (vgl. Tabelle 1). Es werden daher zwei Algorithmen vorgestellt und verglichen, die auf einem unregelmäßigen Dreiecksnetz (Bill 2010, 467 ff.) (engl.: triangulated irregular network, kurz: TIN) die Sichtbarkeit ermitteln und zudem in einer objektrelationalen räumlichen Datenbank implementiert werden können. Die Dreiecke werden dabei 1) in einem R-Baum (Guttman 1984, Bill 2010, 426 f.), als räumliche Indexstruktur, und 2) in einer topologischen Datenstruktur organisiert. Die Analyse der Sichtbarkeit erfolgt bei beiden Ansätzen durch die Überprüfung der direkten Sichtlinie zwischen Betrachter und Ziel (engl.: line-of-sight, kurz: LOS). Diese Algorithmen werden entsprechend des Web Processing Service (WPS) Standards des OGC in eine serviceorientierte Architektur eingebunden, und können somit über das Internet gestartet und dessen Ergebnisse abgefragt werden.

Dieser Beitrag basiert auf ersten Untersuchungen von Hillen (2011) sowie Hillen & Höfle (2012) und ist dabei wie folgt aufgebaut: Kapitel 2 beschreibt den Stand der Forschung hinsichtlich Laserscanning sowie der Berechnung der Sichtbarkeit. In Kapitel 3 werden anschließend kurz das Untersuchungsgebiet und die hierfür verfügbaren Daten beschrieben. Das nachfolgende Kapitel 4 zur Methodik erläutert zunächst die wichtigsten Grundlagen zur Bestimmung der Sichtbarkeit und beschreibt in den folgenden Unterkapiteln die zwei entwickelten Sichtbarkeitsansätze sowie deren Integration in eine serviceorientierte Architektur. Die Ergebnisse zur Untersuchung der Effizienz der Ansätze werden in Kapitel 5 vorgestellt. Der Beitrag schließt mit einem Ausblick auf zukünftige Forschungsansätze.

2 STAND DER FORSCHUNG

Laserscanner sind aktive Fernerkundungssysteme, die einen Laserstrahl verwenden, um die Distanz zwischen Sensor und einem bestimmten Objekt (z. B. der Erdoberfläche), basierend auf der Lichtgeschwindigkeit, zu messen (Shan & Toth 2008, 4 ff.). Der Laserstrahl besitzt dabei die Eigenschaft, kleinste Lücken natürlicher Objekte, wie Lücken zwischen Baumblättern, zu durchdringen und somit auch Informationen über den Erdboden unterhalb dieser Objekte zu gewinnen. Durch diese Eigenschaft lassen sich hochgenaue Digitale Oberflächen- sowie Geländemodelle ableiten. Auch in der Forschung wurden die Durchdringungseigenschaften des Laserstrahls, z. B. von Næsset (1997) für die Bestimmung von Baumhöhen, eingesetzt. Jochem et al. (2011) nutzen Laserscanningdaten bereits für die Abschätzung der Biomasse in Wäldern. Im Zuge der Klimawandelforschung findet sich ein weiterer großer Einsatzbereich in der Glaziologie. Bereits frühere Ansätze verglichen Laserscanning und traditionelle photogrammetrische Techniken für das Monitoring und die Detektion von Veränderungen von Gletschern (Baltsavias et al. 2001, Geist et al. 2003, Arnold et al. 2006), während jüngere Studien die Möglichkeiten der Klassifikation von Gletscheroberflächen zur Quantifizierung der Veränderungen untersuchen (Höfle et al. 2007). Ein Überblick über die Verwendung von Laserscanning für geomorphologische Anwendungen und angrenzende Felder wurde von Höfle & Rutzinger (2011) sowie Höfle & Jochem (2012) erstellt.

Die Forschungen im Gebiet der Sichtbarkeitsanalysen unterscheiden sich grundlegend anhand der verwendeten Datenmodelle. Die traditionelle Repräsentation von Höhendaten war das Rastermodell, ein regelmäßiges Gitternetz in der x-y-Ebene mit einer bestimmten Auflösung (Kidner et al. 2000). Das Rastermodell stellt daher ein simples und einfach zu analysierendes Datenmodell (Andrade et al. 2011) dar, wird aber durch die abrupten Kantenübergange in geringen Auflösungen als weniger genau und nicht geeignet für die Modellierung des Geländes angesehen (Lee 1991). Höhere Auflösungen hingegen benötigen großen Speicherplatz, was jedoch heutzutage in der Regel nicht als limitierender Faktor angesehen wird (Andrade et al. 2011). Aufgrund der unkomplizierten Analysemöglichkeiten von Rasterdaten wurde eine Vielzahl an Studien über die Berechnung der direkten Sichtbarkeit (engl.: intervisibility) als auch des sichtbaren Umfelds um einen Punkt (engl.: viewshed) durchgeführt. Franklin et al. (1994) analysierten die Algorithmen R3, R2 und Xdraw für die Berechnung des Sichtbereichs sowie vier verschiedene Indexierungsalgorithmen hinsichtlich der Positionierung von Luftabwehrraketen. Cohen-Or & Shaked (1995) detektierten sichtbare und verdeckte Flächen,

		Rasterauflösung des Höhenmodells [m]			
		1	2	5	10
Maximale Berechnungsdistanzen [m]	1000	> 4 Min.	27,6 Sek.	2,8 Sek.	1,4 Sek.
	500	33,3 Sek.	10,4 Sek.	1,9 Sek.	1,0 Sek.
	300	15,2 Sek.	8,4 Sek.	1,7 Sek.	0,9 Sek.
	200	11,3 Sek.	7,6 Sek.	1,7 Sek.	0,9 Sek.
	100	9,3 Sek.	1,7 Sek.	1,7 Sek.	0,9 Sek.

Tabelle 1: Berechnungszeiten (in Sekunden) für Sichtbarkeitsanalysen in GRASS GIS hinsichtlich unterschiedlicher Rasterauflösungen und maximaler Berechnungsdistanzen

ausgehend von einem bestimmten Betrachterpunkt, direkt in der digitalen Höhenkarte durch einen Algorithmus, der diskrete Sichtbarkeitslinien (engl.: line-of-sight) prozessiert und die Geländeelemente entlang deren diskreten Querschnitts überprüft. Wang et al. (2000) präsentierten einen Algorithmus, der auf Referenzebenen zur Berechnung des Sichtbereichs basiert anstatt wie bisher auf Sichtlinien, und definierten einen Zielpunkt als sichtbar, sofern er oberhalb der Referenzebene des Betrachterpunkts und der umliegenden Punkte liegt. Izraelevitz (2003) stellte einen Ansatz zur Berechnung des sichtbaren Bereichs vor, der auf einer approximierten Sichtbarkeitslinie, basierend auf vorherigen Berechnungen der direkten Sichtbarkeit, beruht. Haverkort et al. (2009) beschreiben eine neue Anwendung der „Distribution Sweeping" Technik nach Goodrich et al. (1993), basierend auf dem Algorithmus nach Van Kreveld (1996), um den Sichtbereich in einem externen Speicher zu berechnen. Andrade et al. (2011) schlagen für diese Berechnung in einem externen Speicher einen weiteren Algorithmus, basierend auf der Arbeit von Franklin & Ray (1994) sowie Franklin (2002), vor.

Die unregelmäßige Dreiecksvermaschung von Höhenpunkten nach Peucker et al. (1978) stellt eine weitere Möglichkeit zur Modellierung des Geländes, ohne abrupte Datensprünge an den Kanten wie bei der Speicherung im Rasterformat, dar und ist die Grundlage für viele weitere Forschungsansätze. Goodchild & Lee (1989) demonstrierten, wie man die minimale Anzahl an Betrachterpunkten ermittelt, um eine Oberfläche komplett einzusehen, und wie viele Punkte es benötigt, um die sichtbare Fläche zu maximieren. Darüber hinaus identifizierten sie das TIN als die bestmögliche Datenstruktur für ihre Arbeit. Lee (1991) deckte daraufhin fünf Sichtbarkeitsprobleme auf und stellte dabei drei verschiedene heuristische Algorithmen, basierend auf einem TIN, zur Lösung dieser Probleme vor. Einen Überblick über Algorithmen zur Berechnung der Sichtbarkeit und des Horizonts sowie Sichtbarkeitsanfragen, basierend auf TINs und Regular Square Grids (RSGs), formulierten De Floriani & Magillo (1999) und De Floriani & Magillo (2003). Kidner et al. (2001) stellten einen Ansatz zur Analyse auf sog. „Multiscale Implicit Triangulated Irregular Networks" (Kidner et al. 2000) vor, der auf der Suche nach Schnittpunkten zwischen den LOS und dem Gelände sowie einem anschließenden Vergleich des Winkels mit dem Betrachterpunkt beruht. Rana & Morley (2002) optimierten die Sichtbarkeitsberechnung durch die Verwendung von topographisch markanten Punkten, die aus dem TIN abgeleitet werden können (vgl. Peucker & Douglas 1975), wie Gipfel, Täler und Pässe, und nutzen diese als Betrachterpunkte.

3 UNTERSUCHUNGSGEBIET UND DATENGRUNDLAGE

Das Untersuchungsgebiet dieses Beitrags befindet sich im Stadtgebiet von Osnabrück in Niedersachsen. Es handelt sich um ein 5,2 km x 3,6 km (in etwa 19 km²) großes Gebiet, das den gesamten Innenstadtbereich sowie einige zentrumsnahe Stadtteile abdeckt und somit überwiegend urbane Struktur beinhaltet. Für das Untersuchungsgebiet liegen Höhendaten in Form eines Digitalen Oberflächenmodells mit einer Auflösung von 1 m vor. Aus diesem wurden vier TINs mit unterschiedlichen Ausdünnungsgraden (2, 8, 10 und 15 m z-Toleranz) abgeleitet. Dabei wird ein initiales grobes TIN schrittweise verbessert, bis im gesamten TIN-Datensatz die maximale vertikale Abweichung zum Ausgangsraster nicht überschritten wird (vgl. Heckbert & Garland 1997).

4 METHODIK

4.1 BESTIMMUNG DER SICHTBARKEIT

Der Begriff „Sichtbarkeit" scheint eindeutig zu sein, bedarf im Zuge dieses Beitrags jedoch einer genaueren Definition. Wenn man sich z. B. einen Turm vorstellt, so ist es durchaus möglich, dass von einem Betrachungspunkt lediglich die Spitze dieses Turms, jedoch nicht der Turm in seiner Gesamtheit gesehen werden kann. Es muss also unterschieden werden, ob ein Objekt gänzlich oder nur zum Teil sichtbar sein soll, um von einem Algorithmus als sichtbar eingestuft zu werden. Für den Algorithmus bedeutet dies konkret, ob der Geländepunkt am potenziellen Ziel oder der Oberflächenpunkt (in diesem Beispiel also die Spitze des Turms) auf seine Sichtbarkeit überprüft werden soll (vgl. Kidner et al. 2001). In diesem Beitrag wird aufgrund der Datenverfügbarkeit daher ein Objekt als sichtbar erklärt, sobald ein Teil dieses Objekts von einem gegebenen Betrachterpunkt gesehen werden kann.

Darüber hinaus ist für die Bestimmung der Sichtbarkeit die Dimension des Ergebnisses von entscheidender Bedeutung. Diese kann in die Kategorien 1) Punktsichtbarkeit, 2) Liniensichtbarkeit und 3) Flächensichtbarkeit nach De Floriani & Magillo (1994) eingeordnet werden. Die Punktsichtbarkeit (i) oder der „direkte Sichtkontakt" von zwei oder mehreren Punkten basiert dabei auf der direkten Verbindung zwischen Betrachter- und Zielpunkt, der line-of-sight (LOS). Wenn die LOS von einem Teil der unterliegenden Oberfläche geschnitten wird, so ist das entsprechende Ziel nicht sichtbar. Das Ergebnis dieser Überprüfung ist die Menge der sichtbaren Punkte. Die Liniensichtbarkeit (2) basiert ebenfalls auf der LOS, liefert aber im Vergleich zur Punktsichtbarkeit (1) den tatsächlich sichtbaren Abschnitt der LOS als

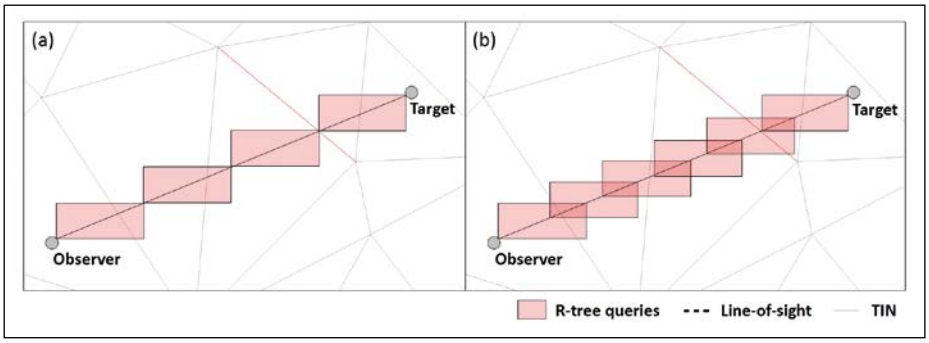


Abbildung 1: Grundprinzip des R-Baum-Algorithmus (a) mit nicht detektierter Schnittkante des TINs (rote Linie) und (b) mit überlappenden Abfragerechtecken (Hillen 2011)

Linienabschnitt zurück. In diesem Fall ist es somit irrelevant, ob das eigentliche Ziel sichtbar ist oder nicht. Die Flächensichtbarkeit (3) oder englisch „viewshed" ist die komplexeste Sichtbarkeitsberechnung und liefert den sichtbaren Anteil der Umgebung um den Betrachterpunkt, unabhängig von möglichen Zielpunkten, als Fläche zurück.

Die Analysen dieses Beitrags beschäftigen sich ausschließlich mit der Punktsichtbarkeit, basierend auf einem TIN (vgl. Kapitel 2). Kidner et al. (2001) haben hierfür einen Ansatz vorgestellt, der als Grundlage für die in diesem Kapitel vorgestellten Algorithmen dient. Die Grundidee basiert darauf, dass als Ausgangspunkt zunächst die beiden Dreiecke bestimmt werden müssen, die Betrachter- und Zielpunkt einschließen. Anschließend werden die Schnittpunkte der LOS mit den Dreieckskanten der TINs in der x-y-Ebene bestimmt und deren Höhenwerte durch lineare Interpolation ermittelt. Um die Sichtbarkeit zu bestimmen, wird anschließend der Winkel zwischen Betrachterpunkt und Schnittpunkt mit dem Winkel der LOS verglichen. Wenn der erstgenannte Winkel höher ist als der Winkel der LOS, so kann der Zielpunkt nicht mehr sichtbar sein, und die Überprüfung wird abgebrochen. Andernfalls werden die nachfolgenden Schnittpunkte entsprechend überprüft.

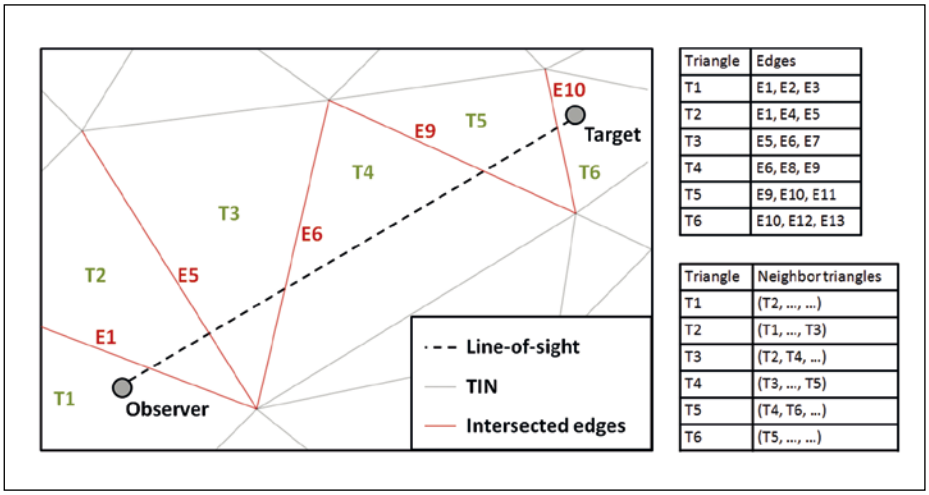
4.2 SICHTBARKEITSANALYSE AUF GRUNDLAGE DES R-BAUMS

Der R-Baum ist eine hierarchische Datenstruktur für räumliches Suchen, basierend auf dem höhenbalancierten B⁺-Baum, und wurde ursprünglich vorgestellt von Guttman (1984). Das Grundprinzip ist dabei die Organisation d-dimensionaler (hier d = 2) geometrischer Objekte durch die Speicherung derer minimal einschließenden Rechtecke (MER). Alle inneren Knoten des R-Baums enthalten somit die MERs ihrer Kindknoten. Wie auch der B⁺-Baum (Kemper & Eickler 2006), speichert der R-Baum die eigentlichen Datenobjekte in den Blättern (Guttmann 1984, Manolopoulos et al. 2005). Eine der häufigsten Anfragen auf einem R-Baum ist die Fenster- oder Überschneidungsabfrage, bei der alle MERs, die ein gegebenes Suchfenster schneiden, zurückgegeben werden (Arge et al. 2004). Ebenfalls möglich auf einem R-Baum ist die Abfrage des nächsten Rechtecks, was besonders dann sinnvoll ist, wenn keine Überschneidung mit einem Suchfenster erwartet wird.

Die generelle Idee des auf dem R-Baum basierenden Algorithmus besteht darin, dass die LOS in einem vorgegebenen Abstand schrittweise untersucht und mittels R-Baum-Abfragen die von der LOS geschnittenen Dreieckskanten des TINs ermittelt werden (siehe Abbildimg 1a). Das TIN muss dafür in einer R-Baum-Datenstruktur organisiert sein.

Ein Problem, das hierbei auftreten kann, ist das Nichterkennen einer Kante des Geländes, wenn diese exakt durch den Punkt verläuft, an dem zwei aufeinanderfolgende Abfragefenster ihren End- bzw. Startpunkt besitzen. Abbildung 1a verdeutlicht durch die rote TIN-Kante diesen eher seltenen Fall. Durch die Einführung eines Überlappungsfaktors für die Abfragefenster lässt sich dieses Problem vermeiden (Abbildung 1b). Für alle Berechnungen in diesem Beitrag wurde ein Überlappungsfaktor von 0,5 verwendet. Neue Abfragefenster beginnen somit einen halben Schritt hinter dem eigentlichen Startpunkt auf der LOS und haben somit einen Überlappungsgrad von 25 %.

Die eigentliche Bestimmung der Sichtbarkeit erfolgt nun über den in Kapitel 4.1 vorgestellten Algorithmus. Zunächst wird daher als Referenz der Winkel der LOS bestimmt. Anhand der Ausrichtung der LOS (in 2D) müssen nun die Abfragefenster erstellt und schrittweise abgefragt werden. Das Ergebnis dieser Abfrage beinhaltet alle möglichen TIN-Kanten, die im Abfragefenster eine Überschneidung mit der LOS aufweisen könnten. Anschließend werden alle Kanten auf Schnitt mit der LOS überprüft. Ist ein Schnittpunkt gefunden, so wird die Höhe dieses Punkts durch lineare Interpolation der Kantenpunkte ermittelt und der Winkel zwischen Betrachterpunkt und dem Schnittpunkt mit dem Winkel der LOS verglichen. Ist der erstgenannte Winkel größer, so wird die LOS-Überprüfung abgebrochen und der Zielpunkt als nicht sichtbar deklariert. Andernfalls wird die Überprüfung fortgesetzt, bis der Zielpunkt erreicht ist.



Triangle	Edges
T1	E1, E2, E3
T2	E1, E4, E5
T3	E5, E6, E7
T4	E6, E8, E9
T5	E9, E10, E11
T6	E10, E12, E13

Triangle	Neighbor triangles
T1	(T2, ..., ...)
T2	(T1, ..., T3)
T3	(T2, T4, ...)
T4	(T3, ..., T5)
T5	(T4, T6, ...)
T6	(T5, ..., ...)

Abbildung 2: Grundprinzip der Sichtbarkeitsanalyse auf Grundlage der topologischen Datenstruktur (Hillen & Höfle 2012)

4.3 SICHTBARKEITSANALYSE MIT TOPOLOGISCHER DATENSPEICHERUNG

Die Topologie hinsichtlich räumlicher Daten beschreibt die räumliche Beziehung zwischen benachbarten Objekten (Bill 2010, 28 f.). Eine topologische Datenstruktur repräsentiert dementsprechend diese Objekte inklusive ihrer gegenseitigen Beziehungen zueinander. Die drei Primitive Knoten, Kanten und Flächen (in diesem Fall Dreiecke) werden dabei verwendet, um zweidimensionale Daten zu beschreiben (Yeung & Hall 2007, Thalheim & Libkin 1998). Durch diese Aufteilung werden die Primitive ohne Redundanzen gespeichert. Für die Knoten werden daher für die Geometrie (Punkte) ihre x- und y-Werte gespeichert, wohingegen die Kanten durch ihre Start- und Endknoten repräsentiert werden. Die Dreiecke werden durch ihre drei Kanten gebildet und speichern zusätzlich ihre Nachbarn. Somit können alle Informationen über die Objekte und ihre Beziehungen untereinander extrahiert werden.

In diesem Beitrag wurde ein sehr ähnlicher Ansatz verwendet. Aufgrund der hohen Anzahl an Dreiecken für die verfügbaren TINs wurde jedoch eine einfachere Datenorganisation gewählt (Abbildung 2). Für jedes Dreieck werden zum einen die x-, y- und z-Koordinaten der Eckpunkte, wodurch implizit, durch Verbinden der Eckpunkte, die drei Kanten abgeleitet werden können, und zum anderen die jeweiligen Nachbarn gespeichert.

Die Grundidee des Algorithmus ist konträr zur Idee des R-Baum-Ansatzes aus Kapitel 4.2. Dort verursacht die Analyse der LOS in fest definierten Schrittweiten viele R-Baum-Abfragen sowie Rechenschritte, die womöglich keinerlei Ergebnis hervorbringen, da zwar viele TIN-Kanten im Abfragefenster liegen, jedoch keine einen Schnitt mit der LOS aufweisen. Die topologische Datenstruktur soll daher dazu dienen, dass der Algorithmus selbstständig durch das TIN „navigieren" kann und nur tatsächlich geschnittene TIN-Kanten überprüft. Dazu muss zunächst das Startdreieck, also das Dreieck, in dem der Betrachterpunkt liegt, gefunden werden. Hierfür wird die Abfrage des nächsten Rechtecks (siehe Kapitel 4.2) des R-Baums verwendet. Eine der drei Kanten dieses Dreiecks muss nun einen Schnitt mit der LOS aufweisen. Ist dieser Schnittpunkt gefunden, so kann durch die Überprüfung der Kanten des direkten Nachbardreiecks sofort der nächste Schnittpunkt ermittelt werden. Wird kein weiterer Schnittpunkt mit einer Dreieckskante gefunden, so ist das Zieldreieck erreicht. Anhand des Beispiels in Abbildung 2 wird diese Vorgehensweise verdeutlicht. Die Bestimmung der Sichtbarkeit erfolgt dabei analog zum Algorithmus aus Kapitel 4.2.

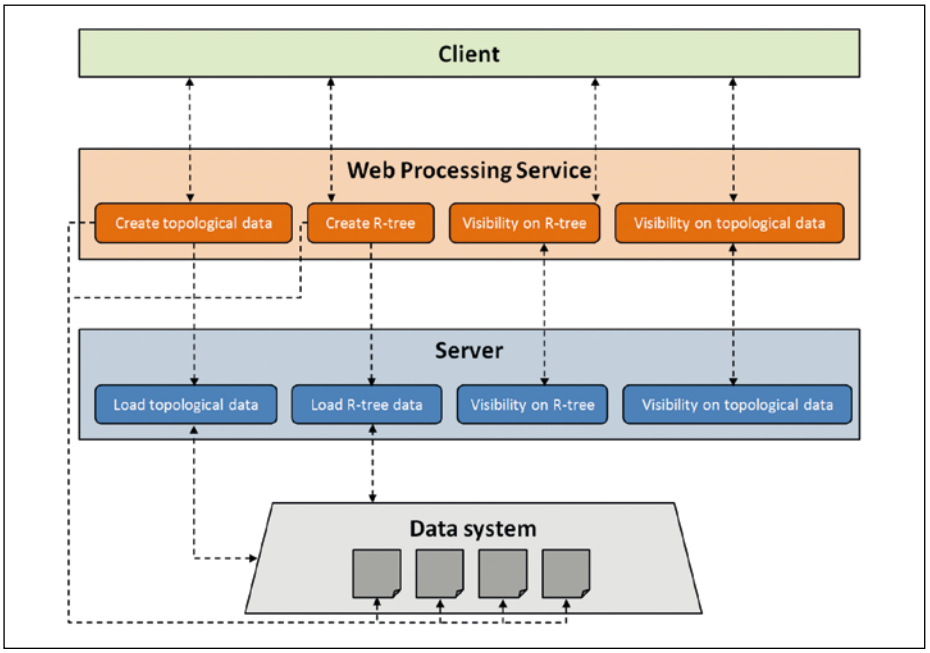


Abbildung 3: Schematische Darstellung des serviceorientierten Serversystems (Hillen & Höfle 2012)

4.4 EINBINDUNG IN SERVICEORIENTIERTE ARCHITEKTUR

Die Einbindung der vorgestellten Algorithmen in eine serviceorientierte Architektur soll unter Verwendung bestehender Standards des Open Geospatial Consortiums (OGC) realisiert werden. Die Web Processing Service (WPS) Spezifikation liefert die Möglichkeit, Geoprozesse über ein Netzwerk (in diesem Fall das Web) verfügbar zu machen. In diesem Beitrag wurde dazu die Python-Implementierung PyWPS (PyWPS 2011) verwendet.

Die naheliegende Möglichkeit ist es nun, die vorgestellten Ansätze über WPS-Prozesse direkt zu implementieren. Aufgrund der Datengrößen der TINs zwischen 35 MB und 1204 MB ist dies jedoch mit typischen Datenübertragungsraten nicht realisierbar. Daher wurde ein XML-RPC Server (XML-RPC 2010) als Unterbau implementiert, der die relevanten Daten im Arbeitsspeicher für den PyWPS vorhält und somit kurze Zugriffszeiten ermöglicht. Abbildung 3 zeigt einen Überblick über das Systemdesign und verdeutlicht die unterschiedlichen Zugriffsebenen. Der Client, in diesem Fall also ein Teil der serviceorientierten Architektur, oder jede beliebige Anwendung mit Zugriff auf das Web interagiert ausschließlich mit dem WPS, indem er die standardisierte Schnittstelle des WPS abfragt. Der WPS bietet dem Client dabei vier Operationen, zwei zur Erstellung der jeweiligen Datenstruktur und zwei zur Abfrage der Sichtbarkeit auf diesen Datenstrukturen anhand der vorgestellten Algorithmen. Der Client kann den zu prozessierenden Datensatz dabei mit einer individuellen Identifikationsnummer, die er nach Abschluss der Erstellung erhält, festlegen.

5 ERGEBNISSE

Zur Ermittlung der Effizienz der vorgestellten Ansätze wurde im Folgenden eine Evaluierung mit künstlich erstellten Testdaten für das Untersuchungsgebiet durchgeführt. Dazu wurden 52 zufällig verteilte Start-Zielpunkt-Paare erstellt, dessen Distanzen zueinander zwischen etwa 47 m und 5.900 m liegen. Die meisten Paare (ca. 65 %) sind dabei jedoch nicht weiter als 500 m voneinander entfernt, da dieser Entfernungsbereich mutmaßlich der relevanteste für Sichtbarkeiten in urbanen Gebieten ist.

Für die Testdaten wurden 50 Sichtbarkeitsberechnungen mit beiden Ansätzen auf den vier verfügbaren TIN-Auflösungen durchgeführt und die jeweiligen Berech-

	R-Baum			Topologische Datenstruktur		
TIN	Min.	Max.	Mittelwert	Min.	Max.	Mittelwert
TIN15	0,0079	1,7904	0,2195	0,0024	0,5579	0,0695
TIN10	0,0105	4,3966	0,5557	0,0021	1,1163	0,1522
TIN8	0,0149	7,4529	0,9157	0,0034	1,6491	0,2185
TIN2	0,2980	150,2021	15,7479	0,0069	13,2951	1,6332

Tabelle 2: Deskriptive Statistiken der mittleren Berechnungszeiten (in Sekunden) für die Ansätze auf dem R-Baum und der topologischen Datenstruktur

nungszeiten dokumentiert. Aus den Berechnungszeiten wurden nachfolgend die deskriptiven Statistiken abgeleitet. Tabelle 2 zeigt das Ergebnis der mittleren Berechnungszeiten für den R-Baum und den topologischen Ansatz. Es ist deutlich zu erkennen, dass der topologische Ansatz dem R-Baum-Ansatz deutlich überlegen ist. Für TIN2 ist die Sichtbarkeitsberechnung dabei im Mittel fast 10-mal schneller. Beide Ansätze sind zudem schneller als die traditionellen Berechnungen auf den Rasterdaten (vgl. Tabelle 1).

6 AUSBLICK

Basierend auf den Forschungen und Ergebnissen dieses Beitrags lassen sich einige zukünftige Arbeiten definieren. Durch die Implementierung mehrerer, zeitgleich ablaufender Prozesse (multi-threading) auf heutigen Mehrkernprozessoren zur Berechnung der Sichtbarkeit lassen sich beide Ansätze deutlich verbessern. Die schrittweise Verarbeitung der LOS beim R-Baum-Ansatz kann somit an mehreren Stellen zeitgleich starten und parallel durchgeführt werden, wodurch Verdeckungen durch die Geländeoberfläche schneller erkannt werden können. Gleiches gilt für den Ansatz auf der topologischen Datenstruktur. Hier kann die Prozessierung zusätzlich rückwärts, also ausgehend vom Dreieck, das den Zielpunkt umgibt, ablaufen und somit die Analysezeit deutlich verringern. Eine Auslagerung der Berechnungen auf den Grafikprozessor (GPU) der Grafikkarte, der speziell auf parallelisierte Prozessierung ausgelegt ist (Steinbach & Hemmerling 2012) durch Schnittstellen wie CUDA (Xia et al. 2010) oder WebGL (Auer 2012), würde die Prozessierungsgeschwindigkeit ebenfalls deutlich verbessern.

Darüber hinaus kann der topologische Ansatz als Grundlage für die Berechnung höherdimensionaler Sichtbarkeitsprobleme (vgl. Kapitel 4.1) genutzt werden. Zur Analyse der sichtbaren Fläche um einen Betrachterpunkt kann somit, ausgehend vom Startdreieck, die gesamte Umgebung bis zu einer bestimmten Entfernung durch die beschriebene Vorgehensweise überprüft werden.

Die Problematik der großen Datenmengen bedarf ebenfalls weiterer Untersuchungen, speziell hinsichtlich der Wiederverwertbarkeit von bereits prozessierten Datensätzen. Hier wäre die Einführung eines OGC Catalogue Service (CSW) denkbar, sodass Datensätze anhand bestimmter Metadaten identifiziert und somit erneut verwendet werden können. Zur Vermeidung von Datenredundanzen ist dies besonders dann sinnvoll, wenn mehrere Benutzer eigenständig ihre Datensätze in das System laden können.

Literatur

Andrade, M. V. A.; Magalhães, S. V. G.; Magalhães, M. A.; Franklin, W. R.; Cutler, B. M. (2011): Efficient viewshed computation on terrain in external memory. In: Geoinformatica, 15 (2), S. 381-397.

Arge, L.; de Berg, M.; Haverkort, H. J.; Yi, K. (2004): The Priority R-Tree: A Practically Efficient and Worst-Case Optimal R-Tree. In: Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data, S. 347-358.

Auer, M. (2012): Real-Time Web GIS Analysis Using WebGL. In: International Journal of 3-D Information Modeling (IJ3DIM), 1 (3), S. 49-61.

Baltsavias, E. P.; Favey, E.; Bauder, A.; Bosch, H.; Peteraki, M. (2001): Digital Surface Modelling by Airborne Laser Scanning and Digital Photogrammetry for Glacier Monitoring. In: Photogrammetric Record, 17 (98), S. 243-273.

Bill, R. (2010): Grundlagen der Geo-Informationssysteme. 5. Auflage. Wichmann Verlag, Berlin/Offenbach.

Carswell, J. D.; Gardiner, K.; Yin, J. (2010): Mobile Visibility Querying for LBS. In: Transactions in GIS, 14, S. 791-809.

Cohen-Or, D.; Shaked, A. (1995): Visibility and Dead-Zones in Digital Terrain Maps. In: Computer Graphics Forum, 14 (3), S. 171-180.

De Floriani, L.; Magillo, P. (1994): Visibility algorithms on triangulated digital terrain models. In: International Journal of Geographical Information Science, 8 (1), S. 13-41.

De Floriani, L.; Magillo, P. (1999): Intervisibility on Terrains. In: Longley, P. A.; Goodchild, M. F.; Maguire, D. J.; Rhind, D. W. (Eds.): Geographic Information Systems: Principles, Techniques, Management and Applications. John Wiley & Sons, Hoboken, NJ, S. 543-556.

De Floriani, L.; Magillo, P. (2003): Algorithms for Visibility Computation on Terrains: a Survey. In: Environment and Planning B – Planning and Design, 30 (5), S. 709-728.

Franklin, W. R. (2000): Applications of analytical cartography. In: Cartography and Geographic Information Systems, 27 (3), S. 225-237.

Franklin, W. R; Ray, C. K.; Mehta, S. (1994): Geometric Algorithms for Siting of Ait Defense Missile Batteries. Technical Report DAAL03-86-D-0001, Battelle, Columbus Division, Ohio.

Franklin, W. R.; Ray, C. K. (1994): Higher isn't necessarily better: Visibility algorithms and experiments. In: Waugh, T. C.; Healey, R. G. (Eds.): Advances in GIS Research: Sixth International

Symposium on Spatial Data Handling. Taylor & Francis, Edinburgh, S. 751-770.

Geist, T.; Lutz, E.; Stötter, J. (2003): Airborne laser scanning technology and its potential for applications in glaciology. In: Proceedings, ISPRS Workshop on 3-D reconstruction from airborne laserscanner and INSAR data, Dresden.

Goodchild, M. F.; Lee, J. (1989): Coverage Problems and Visibility Regions on Topographic Surfaces. In: Annals of Operations Research, 18, S. 175-186.

Goodrich, M. T.; Tsay, J.-J.; Vengroff, D. E.; Vitter, J. S. (1993): External-memory computational geometry. In: Proceedings of the IEEE Symposium on Foundations of Computer Science, 1993, S. 714-723.

Guttman, A. (1984): R-Trees: A Dynamic Index Structure for Spatial Searching. In: Proceedings of the 1984 ACM SIGMOD International Conference on Management of Data, S. 47-57.

Haverkort, H.; Toma, L.; Zhuang, Y. (2009): Computing Visibility on Terrains in External Memory. In: Proceedings of the 9th Workshop on Algorithm Engineering and Experiments/Workshop on Analytic Algorithms and Combinatorics (ALENEX/ANALCO 2007), S. 13-22.

Heckbert, P. S.; Garland, M. (1997): Survey of Polygonal Surface Simplification Algorithms. Technical Report, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA.

Hillen, F. (2011): Optimization of LiDAR data line of sight analysis in service-oriented architectures using the OGC Web Processing Service. Master Thesis (unveröffentlicht), Universität Osnabrück.

Hillen, F.; Höfle, B. (2012): Interoperable Web-based 3D Analysis of Laser Scanning Data for Location-based Mobile Applications. In: Löwner M.-O.; Hillen, F.; Wohlfahrt, R. (Hrsg.): Geoinformatik 2012 – Mobilität und Umwelt. Konferenzband. Shaker Verlag, Aachen, S. 343-350.

Höfle, B.; Geist, T.; Rutzinger, M.; Pfeifer, N. (2007): Glacier Surface Segmentation Using Airborne Laser Scanning Point Cloud and Intensity Data. In: International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVI (Part 3/W52), S. 195-200.

Höfle, B.; Jochem, A. (2012): 3D-Laserscanning-Punktwolken und GIS – Aktuelle Entwicklungen. In: gis.SCIENCE, 25 (2), S. 91-100.

Höfle, B.; Rutzinger, M. (2011): Topographic airborne LiDAR in geomorphology: A technological perspective. In: Zeitschrift für Geomorphologie, 55, Suppl. 2, S. 1-29.

Izraelevitz, D. (2003): A Fast Algorithm for Approximate Viewshed Computation. In: Photogrammetric Engineering and Remote Sensing, 69 (7), S. 767-774.

Jochem, A.; Hollaus, M.; Rutzinger, M.; Höfle, B. (2011): Estimation of Aboveground Biomass in Alpine Forests: A Semi-Empirical Approach Considering Canopy Transparency Derived from Airborne LiDAR Data. In: Sensors, 11 (1), S. 278-295.

Kemper, A.; Eickler, A. (2006): Datenbanksysteme: Eine Einführung. 6. Auflage. Oldenbourg Verlag, München/Wien, S. 215-216.

Kidner, D. B.; Sparkes, A. J.; Dorey, M. I.; Ware, J. M.; Jones, C. B. (2001): Visibility Analysis with the Multiscale Implicit TIN. In: Transactions in GIS, 5 (1), S. 19-37.

Kidner, D. B.; Ware, J. M.; Sparkes, A. J.; Jones, C. B. (2000): Multiscale Terrain and Topographic Modelling with the Implicit TIN. In: Transactions in GIS, 4 (4), S. 379-408.

Lee, J. (1991): Analyses of visibility sites on topographic surfaces. In: International Journal of Geographical Information Systems, 5 (4), S. 413-429.

Manolopoulos, Y.; Nanopoulos, A.; Papadopoulos, A. N.; Theodoridis, Y. (2005): R-Trees: Theory and Applications. Advanced Information and Knowledge Processing. Springer-Verlag, Berlin/Heidelberg/New York, S. 7-13.

Næsset, E. (1997): Determination of mean tree height of forest stands using airborne laser scanner data. In: ISPRS Journal of Photogrammetry and Remote Sensing, 52 (2), S. 49-56.

Open Geospatial Consortium (2007): OpenGIS Web Processing Service, Version 1.0.0. http://portal.opengeospatial.org/files/?artifact_id=24151, Zugriff 11/2012.

Peucker, T.; Douglas, D. (1975): Detection of Surface-Specific Points by Local Parallel Processing of Discrete Terrain Elevation Data. In: Computer Graphics and Image Processing, 4, S. 375-387.

Peucker, T. K.; Fowler, R. J.; Little, J. J.; Mark, D. M. (1978): The Triangulated Irregular Network. In: Proceedings of the DTM Symposium, ASPACSM, S. 24-31.

PyWPS (2011): Welcome to PyWPS – PyWPS. http://pywps.wald.intevation.org/, Zugriff 11/2012.

Rana, S.; Morley, J. (2002): Optimising visibility analyses using topographic features on the terrain. Centre for Advanced Spatial Analysis Working Paper Series, Paper 44.

Shan, J.; Toth, C. (2008): Topographic Laser Ranging and Scanning: Principles and Processing. CRC Press, Taylor & Francis Group, Boca Raton, FL.

Steinbach, M.; Hemmerling, R. (2012): Accelerating batch processing of spatial raster analysis using GPU. In: Computers & Geosciences, 45, S. 212-220.

Thalheim, B.; Libkin, L. (1998): Semantics in Databases. In: Lecture Notes in Computer Science, 1358. Springer-Verlag, Berlin/Heidelberg/New York, S. 124-130.

Trapp, M.; Schneider, L.; Lehmann, C.; Holz, N.; Döllner, J. (2011): Strategies for visualising 3D points-of-interest on mobile devices. In: Journal of Location Based Services, 5 (2), S. 79-99.

Van Kreveld, M. (1996): Variations on sweep algorithms: Efficient computation of extended viewsheds and classifications. In: Proceedings of the 7th Int. Symposium on Spatial Data Handling, 1996, S. 15-27.

Wang, J.; Robinson, G. J.; White, K. (2000): Generating Viewsheds without Using Sightlines. In: Photogrammetric Engineering and Remote Sensing, 66 (1), S. 87-90.

Wang, Z.; Wu, L.-X. (2009): Automated extraction of building geometric features from raw LiDAR data. In: Geoscience and Remote Sensing Symposium, IGARSS 2009, Vol. 2, S. 436-439.

Xia, Y.; Li, Y.; Shi, X. (2010): Parallel Viewshed Analysis on GPU Using CUDA. In: 2012 Fifth International Joint Conference on Computational Sciences and Optimization, S. 373-374.

XML-RPC (2010): XML-RPC Home Page. http://www.xmlrpc.com/, Zugriff 11/2012.

Yeung, A. K. W.; Hall, G. B. (2007): Spatial database systems: design, implementation and project management. In: GeoJournal Library, 87. Springer-Verlag, Berlin/Heidelberg/New York, S. 100-103.