

**Vektorielle Modellierung
von Forstbeständen
aus terrestrischen Laserscanner-Daten**

Schriftliche Hausarbeit
zur Erlangung des akademischen Grades

Bachelor of Science

an der Universität Trier
im Fachbereich VI
Angewandte Geoinformatik

vorgelegt von

Sebastian Lamprecht
Universitätsring 8
54296 Trier

Trier, im März 2013

1. Gutachter: Prof. Dr. Thomas Udelhoven
2. Gutachter: Dr. Sebastian Mader

Erklärung zur Bachelorarbeit

Hiermit erkläre ich, dass ich die Bachelorarbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt und die aus fremden Quellen direkt oder indirekt übernommenen Gedanken als solche kenntlich gemacht habe.

Die Arbeit habe ich bisher keinem anderen Prüfungsamt in gleicher oder vergleichbarer Form vorgelegt. Sie wurde bisher nicht veröffentlicht.

Trier, den 11. März 2013

Inhaltsverzeichnis

Abbildungsverzeichnis	i
Tabellenverzeichnis	iii
Abkürzungsverzeichnis	iii
1. Einleitung	1
1.1 Forschungsfragen	1
1.2 Modellierungsverfahren	1
1.2.1 Slice-Ansätze	1
1.2.2 TIN-Ansätze	2
1.2.3 Zylinder und Spline-Ansätze	3
1.2.4 Graph-Ansatz	3
1.3 Kritik an bestehenden Modellen	4
2. Motivation	5
2.1 Ziele dieser Arbeit	5
2.2 Erhoffte Vorteile eines Vektormodells gegenüber anderen Modellen	5
3. Datengrundlage	6
4. Vorbetrachtungen	7
4.1 Definitionen	7
4.1.1 Definition eines Baumes	7
4.1.2 Definition Astsegment	7
4.2 Abstrakte Betrachtung	8
4.3 Abstraktes Segmentmodell	8
4.4 Geometrisches Modell	9
4.5 Vektormodell	10
4.5.1 Modelleigenschaften	11
5. Modellierung	11
5.1 Modellierungsansatz	11
5.2 Layerselektion	12
5.2.1 Selektion des ersten Layers	12
5.2.2 Layerselektion	12
5.3 Clusteranalyse	13
5.4 Knotenbestimmung	14
5.4.1 Schritt 1: Anpassen einer Kugeloberfläche	15
5.4.2 Schritt 2: Rotation um den Kugelmittelpunkt	15
5.4.3 Schritt 3: Anpassen eines Kreises in 2D	16

5.4.4 Schritt 4: Rückrotation des Kreismittelpunktes	17
5.4.5 Vorteile des Vorgehens der Knotenbestimmung	17
5.5 Validierung	18
5.5.1 Plausibilitätsprüfung der Kreisanpassung	18
5.5.2 Plausibilitätsprüfung	19
5.5.3 Validierung eines Punktes an einem Segment	20
5.5.4 Validierung aller Punkte	21
5.6 Modellablauf	21
5.6.1 Aufbau der Modelllogik	21
5.6.2 Auffinden weiterer Vektoren	22
5.7 Entscheidungskette	23
5.7.1 Auswahl des nächsten zu analysierenden Knotens	23
5.7.2 Clusteranalyse	24
5.7.3 Knotenbestimmung	24
5.7.4 Validierung	24
5.7.5 Terminierung des Modells	25
5.8 Parametrisierung	25
5.8.1 Layerselektion	26
5.8.2 Clusteranalyse	26
5.8.3 Knotenbestimmung	27
6. Implementierung	27
6.1 Softwaredesign und Programmablauf	28
6.2 Tabellendesign	29
6.2.1 Datengrundlage	29
6.2.2 Modellierungsergebnis	29
6.2.3 Hilfstabellen	30
6.3 Implementierung der Modellkomponenten	30
6.3.1 Aufbau der Modelllogik	31
6.3.2 Auswahl eines Knotens	31
6.3.3 Layerselektion	31
6.3.4 Clusteranalyse	31
6.3.5 Knotenpunktbestimmung	32
6.3.6 Validierung	32
6.4 Konfigurationsdatei	32
6.5 Tests	33
6.5.1 Datengrundlage	33

6.5.2 Modellanpassung	34
6.5.3 Modelllaufzeit	34
7. Fazit	35
8. Ausblick	36
8.1 Erweiterungen des Modells	36
8.2 Integrität des Modells	37
8.3 Geobotanische Parameter	37
Literaturverzeichnis	I

Abbildungsverzeichnis

Abbildung 1: Querschnittserkennung für stark verzweigte Bäume. (BIENERT 2007)

Abbildung 2: Vorgehen der Segmentierung mit den Schritten Aufbau möglicher Graphen, Auswahl des Graphen mit kürzesten Kanten, Segmenterkennung mit Topologie, Aufbau eines 3D-Objektes über Voxel-Spaces. (GORTE, B.; PFEIFER, N. 2004)

Abbildung 3: Aufnahmegeometrie von Stammquerschnitten bei einer Einzelaufnahme. (KIRÁLY, G.; BROLLY, G. 2007)

Abbildung 4: Baum mit farblich markierten beliebigen Astsegmenten.

Abbildung 5: Segmentbetrachtung eines Baumes mit eingezeichneten Querschnitten und Knotenpunkten, sowie farblich hervorgehobene beliebige Segmente.

Abbildung 6: Beliebiges Abstraktes Modell eines Baumes mit farblich markierten Segmenten.

Abbildung 7: Geometrisches Modell mit eingezeichneten Kegelstümpfen und Kugeloberflächen um die Knotenpunkte.

Abbildung 8: Definition der Oberfläche eines Kegelstumpfes über eine Vektordarstellung.

Abbildung 9: Auswahl des ersten Layers mit horizontaler Ausrichtung über der Bodenoberfläche.

Abbildung 10: Querschnitt der Selektion eines Layers zu einem bekannten Knoten in Richtung des Parent-Vektors. Die Punkte des Layers liegen innerhalb der hier grün markierten Fläche.

Abbildung 11: Draufsicht auf verschiedene Buffer. Die Punkte innerhalb einer solchen Bufferhülle bilden einen Cluster.

Abbildung 12: Rotation des Clusters zur z-Achse über den orthogonalen Vektor.

Abbildung 13: Rotation des Kreismittelpunktes zurück zum Cluster zur Bestimmung des tatsächlichen Knotenpunktes.

Abbildung 14: Plausibilitätsprüfung der Kreisanpassung.

Abbildung 15: Bestimmung des Punktabstandes zu einer Kegelstumpfoberfläche.

Abbildung 16: Fehlerbetrachtung des Modells; Residuen $<5\%$ grün, 5 bis 10% gelb, -5 bis -10% hellblau, $>10\%$ rot, $<-10\%$ blau und nicht analysierte Punkte grau.

Abbildung 17: Selektion des ersten Layers, auffinden von Stammmittelpunkten und Projektion der Knoten auf die Bodenoberfläche.

Abbildung 18: Modellaufbau durch Layerselektion.

Abbildung 19: Parallele Äste können zum "Überspringen" der Vektoren führen, wenn die Äste Knoten mit größerem Radius zuerst abgearbeitet werden. Gewünschte Modellierung grün und "Überspringen" rot markiert.

Abbildung 20: Kreisanpassung (schwarzer Kreis) an größeren Cluster (grün) erzielt auch gute Anpassung an gegenüberliegenden Cluster (grau), während Anpassung an kleineren Cluster (roter Kreis) den größeren kaum approximiert.

Abbildung 21: Holzvolumenanalyse als Holzvolumen in Abhängigkeit vom gewünschten Astdurchmesser.

Abbildung 22: Softwaredesign des implementierten Vektormodells. Die Programmsteuerung und vereinzelte Berechnungen übernimmt R, während Postgis die räumlichen Analysen übernimmt. Das Modellierungsergebnis befindet sich in der Tabelle branches.

Abbildung 23: Tabellendesign der Modellimplementierung.

Tabellenverzeichnis

Tabelle 1: Parameter des implementierten Modells mit Wertebereich, Einheit und Beschreibung.

Abkürzungsverzeichnis

ca.	circa
cm	Centimeter
DTM	Digital Terrain Model (digitales Oberflächenmodell)
LIDAR	LIght Detection And Ranging
m	Meter
TIN	Triangulated Irregular Network (trianguliertes Dreiecksnetz)
z. B.	zum Beispiel

1. Einleitung

1.1 Forschungsfragen

Aufgrund modernerer LIDAR-Messgeräte und verbesserter Möglichkeiten zur Verarbeitung von Massendaten haben sich im Bereich der Forstwissenschaften Hoffnungen ergeben, aus Laserscanner-Daten Erkenntnisse zum Zustand und Aufbau der Vegetation zu erhalten. Während luftgestützte LIDAR-Messungen vorwiegend zur mesoskaligen Schätzung des Holzvolumens verwendet werden, können aus den geometrisch höher aufgelösten terrestrischen Messungen, detaillierte Informationen zu einzelnen Individuen abgeleitet werden.

Daher wird für eine begrenzte Anzahl an Individuen mit unterschiedlichen Verfahren versucht, forstliche und geobotanische Parameter abzuleiten, welche Rückschlüsse auf Eigenschaften der Makroskala, wie das Klima, erlauben. Beispiele hierzu sind neben dem absoluten Holzvolumen z. B. der beanspruchte Kronenraum, der Verästelungsgrad, der Blattflächenindex, das Wachstumsverhalten oder auch der Einfluss des Konkurrenzverhaltens zwischen mehreren Individuen auf die Struktur der Bäume.

Um aus den Daten einer terrestrischen LIDAR-Messung die gewünschten Parameter abzuleiten, ist eine aufwendige Analyse vor allem über numerische Modellierungsverfahren notwendig, die die ungeordneten Daten in mathematisch erfassbare geometrische Objekte überführen.

1.2 Modellierungsverfahren

Unterschiedliche Modellierungsverfahren aus der Literatur nutzen unterschiedliche Ansätze, die vorwiegend auf vorgefilterte Punktdaten zurückgreifen. Hier sollen zunächst beispielhaft verschiedene Verfahren zur Analyse von Bäumen aus terrestrischen LIDAR-Messungen beschrieben und deren Vor- und Nachteile kurz erläutert werden. Auf die Geometrie der LIDAR-Daten wird im Kapitel 3. (*Datengrundlage*) näher eingegangen.

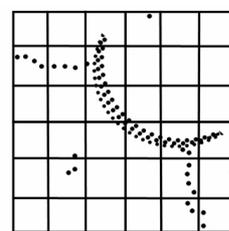
1.2.1 Slice-Ansätze:

Weit verbreitet sind wegen der Einfachheit des Ansatzes und der unproblematischen Analyse der Ergebnisse unterschiedliche Formen von Slice-Ansätzen, die im Allgemeinen aus dem Datensatz Teilmengen in Scheibenform parallel zur Bodenoberfläche auswählen. Innerhalb dieser Querschnitte werden über Clusterverfahren Querschnitte gesucht, um an sie jeweils eine Kreisform anzupassen.

Ein exemplarisches Beispiel für dieses Vorgehen ist die Arbeiten von BIENERT (2006), bei dem die Ebenen eine einheitliche Dicke aufweisen. Die Querschnitte werden über Clusteranalyse gefunden und anschließend wird deren Mittelpunkt ermittelt, wobei nur Cluster mit einer Mindestanzahl an Punkten betrachtet werden. Der Stammansatz wird hier auf das DTM (Digital Terrain Model) projiziert, um die Position des Baumes zu markieren. Es ergeben sich als Modellparameter nur die Scheibendicke und der Abstand der Scheiben untereinander, der zu meist gering gehalten wird. Als Ergebnis des Modells erhält man den Stamm als ein Zylindermodell, mit dem die Baumhöhe, die Baumposition, der Brusthöhendurchmesser und das Stammvolumen direkt ermittelt werden können.

Eine Verfeinerung des Modells für Nadelbäume findet sich in BIENERT (2007), bei dem der Einfluss von kleineren Ästen auf die Kreisanpassung des Stammes reduziert wird, indem der Stammquerschnitt genauer abgegrenzt wird. Dabei werden durch Rasterung zunächst die Punkte in Cluster zusammengefasst und daraufhin alle Cluster entfernt, deren Anteil an der Gesamtpunktzahl zu gering ist.

Das Slice-Modell zeichnet sich aufgrund seiner Einfachheit durch eine geringe Laufzeit aus, die durch Parallelisierung zusätzlich verbessert werden kann. Problematisch an einem solchen Ansatz ist, dass so nur der Stamm oder vertikale Äste modelliert werden können, da sonst keine Kreisform angepasst werden kann.



0	0	4	1	0	0
4	3	15	0	0	0
0	0	16	1	0	0
0	1	2	14	16	9
0	2	0	0	4	1
0	0	0	0	2	2

scan resolution: 0.1°
distance to scanner: 10 m
target number
on a raster element: 19
- 30%: 6
threshold: 13

0	0	4	1	0	0
4	3	15	0	0	0
0	0	16	1	0	0
0	1	2	14	16	9
0	2	0	0	4	1
0	0	0	0	2	2

Abbildung 1: Querschnittserkennung für stark verzweigte Bäume.

1.2.2 TIN-Ansätze:

Die Arbeit von ASCHOFF (2004) nutzt zur Filterung der Daten den Abstand und Winkel der Punkte untereinander. Die Bodenoberfläche wird dabei über ein trianguliertes DTM angepasst, sodass die überflüssigen Bodenpunkte entfernt werden können. Über die Auswahl verschiedener Schichten als Teilmenge parallel zur Bodenfläche (ähnlich wie bei Slice-Ansätzen) können über Bildanalyseverfahren die Stämme und über Approximation von Kreismittelpunkten die Mittelpunkte der Stämme aufgefunden werden. Die eigentliche Modellierung erfolgt daraufhin mit einem *power crust algorithm*, der über Triangulation die Oberfläche approximiert und damit einen Datensatz von ca. 800.000 Punkten auf ca. 50.000 repräsentative Punkte reduzieren kann. Das Ergebnis einer solchen Modellierung ist damit ein trianguliertes Drei-

ecksnetz (TIN) der Stämme, welches die Stammoberfläche gut approximiert. Zudem sind die Mittelpunkte der einzelnen Schichten bekannt. Die forstlichen Parameter können über die Mittelpunkte oder indirekt über die Oberfläche abgeleitet werden. Nachteile des Modells sind das weiterhin hohe Datenvolumen des TIN und die Nachteile eines Slice-Ansatzes, da die Positionen und Radien in Schichten über Kreisanpassung ermittelt werden.

1.2.3 Zylinder und Spline-Ansätze:

Eine übliche geometrische Anpassung an Bäume aus luftgestützten LIDAR-Messungen ist eine Zylinderanpassung an den Stamm. Solche Zylinder können bei den geometrisch wesentlich höher aufgelösten terrestrischen LIDAR an kleinere Segmente des Baumes angepasst werden, wie die Arbeit von PFEIFER UND WINTERHALDER (2004) zeigt. Hier wird durch ein Splinefitting, also die Anpassung von beliebigen Kurven, eine einfache Zylinderanpassung durch die Anpassung von überlappenden Zylindern optimiert werden. So kann die Oberfläche über die Splines mit einer Genauigkeit von 1cm bis 2cm modelliert werden. Das Datenvolumen des Modellierungsergebnisses ist wegen der mathematischen Definition der Kurven gering und die Parametrisierung ist überschaubar. Bei diesem konkreten Ansatz ist nur eine Anpassung an den Stamm möglich. Eine Anpassung an Äste wäre aber bei einer Zerlegung der Punktwolke denkbar.

1.2.4 Graph-Ansatz:

Einen rein mathematischen Ansatz mit sehr genauen Ergebnissen stellt die Modellierung von GORTE UND PFEIFER (2004) dar, bei dem zuerst alle Punkte gerastert werden. Danach wird über die Topologie ein Graph (*skeleton*) aufgebaut und einzelne Segmente abgeleitet. Auf diese Weise können auch die Astgabelungen erkannt werden. Um das Modell an das Baumvolumen anzunähern werden verschiedene 3D-Elemente zwischen den Punkten angepasst, sodass sich für das gesamte Modell ein *Voxel-Space*, also ein geometrisches Objekt im 3D-Raster, ergibt. Dies führt zu einer sehr detaillierten Modellierung, bei der die Volumenbestimmung einfach ist. Dennoch besteht hier das Problem, dass die geobotanischen Parameter wie Astwinkel und Durchmesser, nur indirekt abgeleitet werden können. Zudem ist der Aufbau des Graphen aufgrund der umfangreichen Topologieanalyse kostspielig, sodass die Analyse eines oder mehrerer Bäume zu höheren Laufzeiten führen dürfte.

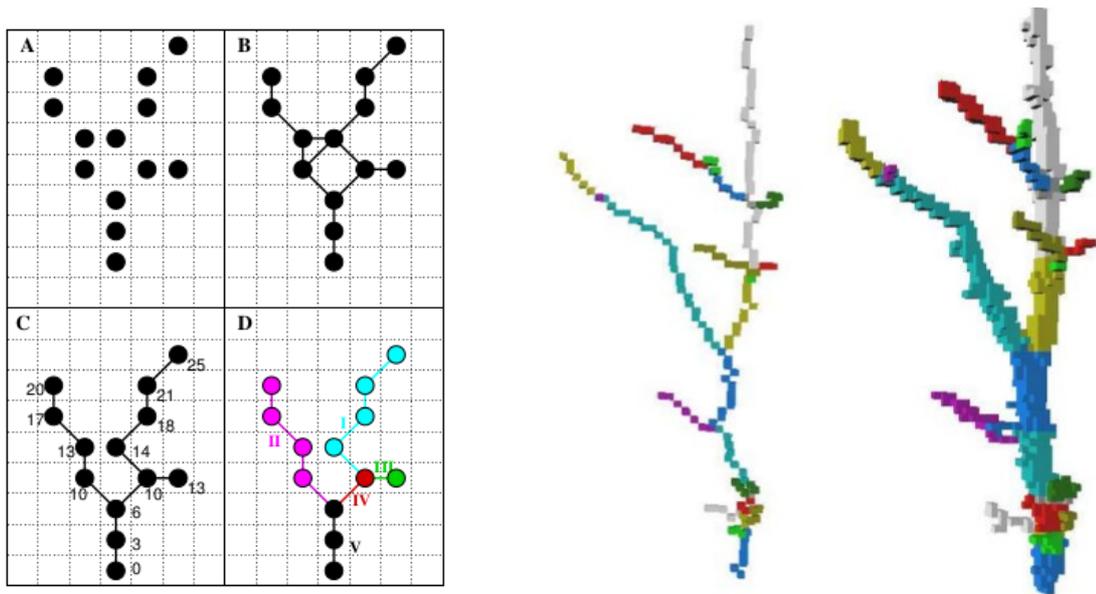


Abbildung 2: Vorgehen der Segmentierung mit den Schritten Aufbau möglicher Graphen, Auswahl des Graphen mit kürzesten Kanten, Segmenterkennung mit Topologie, Aufbau eines 3D-Objektes über Voxel-Spaces.

1.3 Kritik an bestehenden Modellen

Die bestehenden Modelle weisen alle unterschiedliche Vor- und Nachteile auf. Dazu zählen ein teilweise hohes Datenvolumen der Ergebnisse und eine weiterhin komplizierte Ableitung geobotanischer Parametern. Die Erfassung feinerer Äste wird aufgrund der Modellstruktur häufig nicht ermöglicht oder es entstehen Probleme beim Zusammensetzen einzelner Komponenten, sodass die Topologie der Äste nicht eindeutig geklärt werden kann. Bisher scheinen die Modelle entweder der Maxime einer sehr genauen Modellanpassung für den Stamm oder einer Erfassung der meisten Äste ohne Beachtung der Topologie zu folgen. Der Modellierungsaufwand unterscheidet sich dabei je nach Ansatz.

Es fällt auf, dass die geometrischen Eigenschaften der Bäume in den weit verbreiteten Slice-Ansätzen fast völlig ignoriert werden und damit eine Modellierung mit zu eng gefassten Annahmen erfolgt. Ein Ansatz, mit dem – wie bei einem Slice-Ansatz – ein geometrisches Modell erzeugt wird, welches auch den feineren Astbereich modelliert und eine einfache anschließende Analyse komplizierter geobotanischer Parameter ermöglicht, scheint zu fehlen.

2. Motivation

Um über die reine Erfassung des Stammholzes und einzelner größerer Äste hinauszukommen, müsste die Geometrie der Bäume stärker beachtet werden und ein Slice-Modell genau an diese Eigenschaften angepasst werden. Es wäre daher ein angepasstes Modell wünschenswert, bei dem feinere Äste erfasst und gleichzeitig in eine nachvollziehbare Vektordarstellung überführt werden.

2.1 Ziele dieser Arbeit

Aus den genannten Gründen ist das Hauptziel dieser Arbeit, ein vektorbasiertes Modell zu entwickeln, welches sich die spezifischen geometrischen Eigenschaften der Bäume zu Nutze macht. Dabei soll jeder Vektor die Orientierung eines Aststücks bzw. des Stammes erfassen und eine Annäherung des Holzvolumens und anderer forstlicher Parameter möglich sein. Zum Festlegen der Vektoren und Annäherung des Holzvolumens soll die Mittelpunktbestimmung aus einem Slice-Ansatz dienen. Als Datengrundlage ist die Punktwolke eines terrestrischen Laserscanners vorgesehen, die für die Anpassung der Oberfläche eine angemessene Auflösung bereitzustellen hat.

Aufgrund des sehr hohen Datenvolumens einer solchen Punktwolke ist der Modellierungsaufwand so gering wie möglich zu halten. Neben einer angemessenen Modelllaufzeit ist auch die praktikable Parametrisierung des Modells und eine möglichst gute Modellanpassung zu berücksichtigen. Daher ist neben der formalen Entwicklung des Modells auch eine mögliche Implementierung des Modells Teil dieser Arbeit.

2.2 Erhoffte Vorteile eines Vektormodells gegenüber anderen Modellen

Dieser Absatz greift der Arbeit etwas voraus, soll aber die Gründe der Motivation eines neuen Modellierungsansatzes weiter verdeutlichen.

Ein vektorieller Ansatz verspricht gegenüber einem Slice-Ansatz eine Erfassung von Ästen, die nicht vertikal orientiert sind. Da die Mittelpunktbestimmung aus diesem Ansatz übernommen wird, können die Vorteile dieses Modellierungsansatzes übernommen werden. Darunter fallen die einfache Berechnung des Holzvolumens, die hohe Informationsdichte des fertigen Modells und das damit verbundene geringe Datenvolumen. Zudem sind die topologischen Beziehungen der Vektoren und damit der Äste nachvollziehbar und das Modell konsistent.

Gegenüber einem Cluster- oder TIN-Ansatz verspricht das Vektormodell ein weitaus geringeres Datenvolumen und eine nachvollziehbare Geometrie. Zwar wird nicht jeder Punkt optimal durch das Modell erfasst, aber die räumlichen Beziehungen der Äste untereinander und der repräsentative Durchmesser können durch ein Vektormodell einfacher nachvollzogen werden. Auch eine Erfassung der Orientierung der Cluster, durch beispielsweise Hauptkomponenten, die zu einem Vektormodell führen könnten, leistet hier keine adäquate Abhilfe. Der Grund hierfür liegt darin, dass die Endpunkte der Vektoren nicht übereinstimmen würden, damit windschief im Raum lägen und so das Vektormodell nicht widerspruchsfrei wäre.

Da das Vektormodell vor allem darauf abzielt, auch feinere Äste zu erfassen, scheint eine Rasterung der Punktdaten – wegen des Verlustes der genauen geometrischen Information – ausgeschlossen. Dieser Effekt kann auch durch eine gesteigerte geometrischen Auflösung des Laserscanners nicht ausgeglichen werden.

3. Datengrundlage

Wie im bereits mehrfach erwähnt, soll der Modellierung als Datengrundlage die Messungen eines terrestrischen Laserscanners zugrunde liegen, da dieser eine hohe geometrische Auflösung bietet. Die Geometrie eines solchen Datensatzes entspricht einer dreidimensionalen Punktwolke $P \subseteq \mathbb{R}^3$, wobei jeder Aufnahmepunkt zusätzlich über eine Reflexionseigenschaft der Aufnahmewellenlänge verfügt, die aber im Folgenden aufgrund des strittigen Informationsgehaltes ignoriert wird.

Da das Datenvolumen einer solchen Punktwolke durch Messung von Bodenpixeln meist übermäßig groß ist und eine Vielzahl von unplausiblen Messungen auftreten, wie offensichtlich falsche Geometrie (Mehrfachreflexion, Streuung, Wind), zu geringe Reflexion oder distanzabhängige geometrische Messungenauigkeiten, sollte eine geeignete Datenvorfilterung vorgenommen werden. Die geometrische Auflösung der Punktwolke hängt vom Laserscanner, den Geräteeinstellungen, der Distanz zum Sensor und den Messfehlern ab.

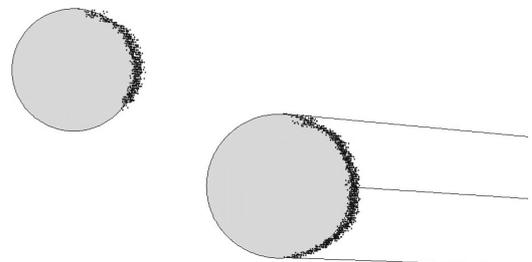


Abbildung 3: Aufnahmegeometrie von Stammquerschnitten bei einer Einzelaufnahme.

Wie auch der Slice-Ansatz wird sich das in dieser Arbeit dargestellte Modell die Sichelform der gemessenen Stämme und Äste zu Nutze machen. Diese ergeben sich aus der meist runden

Form der Stammquerschnitte und der festgelegten Aufnahme­richtung. Durch miteinander ver­ schnittenen Punktwolken aus mehreren Messrichtungen können die Abschattungseffekte weit­ gehend reduziert und damit die Oberfläche weiter angenähert werden. Nachteilig sind hier aber die Verschneidungsfehler, das höhere Datenvolumen und der Mehraufwand zu nennen. Näheres zum Aufbau einer Messreihe, Geländemodell, Datenfilter und Punktgeometrie findet sich beispielsweise in der Arbeit von SIMONSE (2003).

4. Vorbetrachtungen

Aufgrund der im Kapitel 2. (*Motivation*) beschriebenen Problematik der Bedeutung der Baumgeometrie für die Modellierung wird in dieser Arbeit davon ausgegangen, dass ein opti­ males Modellierungsergebnis erreicht werden kann, wenn die Geometrie des Baumes so gut wie möglich berücksichtigt wird. Daher wird zunächst die Form eines Baumes näher spezifi­ ziert, um daraufhin über eine abstraktere Betrachtung zu einem geometrischen Modell und da­ mit zu einem Vektormodell zu kommen.

4.1 Definitionen

4.1.1 Definition eines Baumes:

Ein Baum im Sinne dieser Arbeit ist zunächst ein geometrisch zusammenhängendes Objekt im dreidimensionalen Raum, welches unter anderem auch über Reflexionseigenschaften ver­ fügt. Grundlegende Eigenschaft eines Baumes ist, dass er als ein einziges Astsegment aus dem Boden entspringt, welches eine annähernd verti­ kale Wuchsrichtung aufweist.

4.1.2 Definition Astsegment:

Unter einem Astsegment versteht man dabei ein beliebiges in Wuchsrichtung des Baumes lie­ gendes Teilstück mit einem senkrecht zu dieser Richtung liegenden Querschnitt. Das erste aus

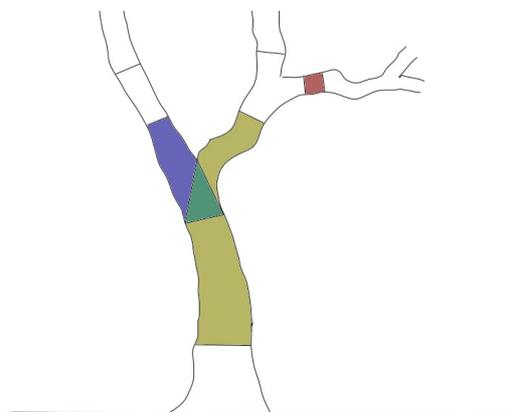


Abbildung 4: Baum mit farblich markier­ ten beliebigen Astsegmenten.

dem Boden entspringende Teilstück ist hierbei das, was man im Alltagssprachgebrauch als Stamm bezeichnet. Von diesem ersten Teilstück wie auch den folgenden Teilstücken können beliebig viele Astsegmente abzweigen, die alltagssprachlich als Äste oder Zweige bezeichnet werden.

4.2 Abstrakte Betrachtung

Da diese sehr allgemeine geometrische Definition eines Baumes kaum verbesserte Ansätze zur Modellierung ermöglichen kann, müssen noch weitere Annahmen getroffen werden.

Ähnlich wie bei Slice-Ansätzen wird davon ausgegangen, dass annähernd runde Stammquerschnitte existieren. Übertragen auf die Definition bedeutet das, dass Astsegmente existieren, deren Querschnitte zur Wuchsrichtung an beiden Segmentenden annähernd rund sind. Ein

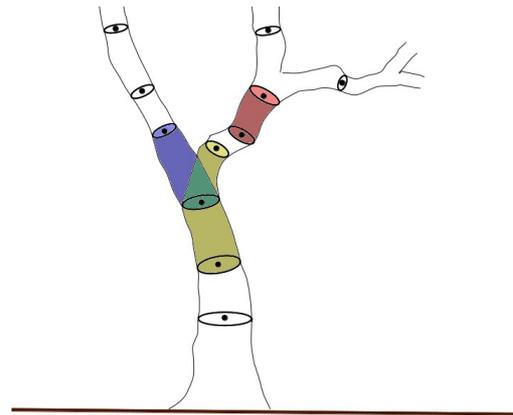


Abbildung 5: Segmentbetrachtung eines Baumes mit eingezeichneten Querschnitten und Knotenpunkten, sowie farblich hervorgehobene beliebige Segmente.

solches Astsegmente soll nun nur noch als *Segment* S_i aus der Menge aller Segmente S bezeichnet werden. Die Kreismittelpunkte der genannten Querschnitte werden im Folgenden als zum Segment gehörige *Knotenpunkte* bezeichnet. Als *Knoten* $k_i \in K$ wird der Knotenpunkt $kp_{k_i} \subseteq \mathbb{R}^3$ zusammen mit der Eigenschaft des Radius $r_{k_i} \in \mathbb{R}$ der Kreisform bezeichnet. Dabei entspricht K der Menge aller Knoten.

Damit können offensichtlich auch Segmente gefunden werden, bei denen mindestens ein Knotenpunkt mit dem eines anderen Segmentes übereinstimmt. Das gegen die Wuchsrichtung liegende Segment wird dabei als *Parent-Segment* und das in Wuchsrichtung liegende Segment als *Child-Segment* bezeichnet. Entsprechend werden auch die Knoten *Parent-Knoten* und *Child-Knoten* genannt.

4.3 Abstraktes Segmentmodell

Aus den zuletzt genannten Segmenten kann ein abstraktes Modell des Baumes generiert werden, bei dem weitere Einschränkungen getroffen werden. Im Folgenden besteht ein solches Modell aus einer Reihe von Segmenten, sodass zu jedem Segment ein Pfad von Parent- oder Child-Segmenten gefunden werden kann, mit dem ein beliebiges Segment des Modells er-

reicht werden kann. Zudem soll zu jedem Segment stets nur ein eindeutiges Parent-Segment existieren. Aus der Definition ergibt sich, dass ein Segment des Modells als verästelt gilt, falls es über mehrere Child-Segmente verfügt. Diese Betrachtung legt nahe, dass ein Segment S_i stets zu seinem Parent-Segment S_j definiert ist und damit die Schreibweise S_{i,S_j} angebracht ist. Da ein Baum definitionsgemäß nur über einen eindeutigen Stammansatz verfügt, existiert auch ein bodennahes Segment S_0 ohne ein eigenes Parent-Segment.

Eine solche abstrakte Modellbetrachtung setzt voraus, dass aus der Vielzahl an möglichen Segmenten eine begrenzte Anzahl ausgewählt wird. Da unterschiedliche Auswahlmöglichkeiten existieren, gibt es mehrere mögliche Modelle, von denen nur eines weiter betrachtet werden muss. Da runde Querschnitte benötigt werden, ist es offensichtlich, dass nicht der gesamte Baum durch ein solches Modell beschrieben werden kann. Vor allem die Endstücke von Ästen und das Laubwerk können durch das Modell nicht erfasst werden.

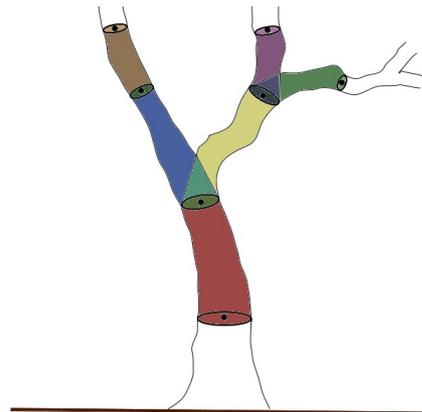


Abbildung 6: Beliebiges Abstraktes Modell eines Baumes mit farblich markierten Segmenten.

4.4 Geometrisches Modell

Als Grundlage für ein mathematisch-geometrisches Modell dient ein beliebiges abstraktes Segmentmodell, bei dem die natürlichen Oberflächen der Segmente durch geometrische Objekte angenähert werden. Die Annäherung eines Segmentes erfolgt über einen Kegelstumpf mit der Grundfläche des zum Parent-Knoten gehörigen Querschnittes und der Deckfläche des zum Child-Knoten gehörigen Querschnittes. Die Knotennahen Bereiche werden durch eine Kugel mit Radius des Knotenquerschnittes angenähert. Dabei sollen die Kugeln stets disjunkt sein, um zu kleine Segmente zu vermeiden.

Ein solches geometrisches Modell ist nur in der Lage, die Oberfläche der Segmente anzunähern, da die Segmente im Bereich zwischen den Querschnitten der Knoten eine gewisse Variation aufweisen. Es liegt daher nahe, dass bei Seg-

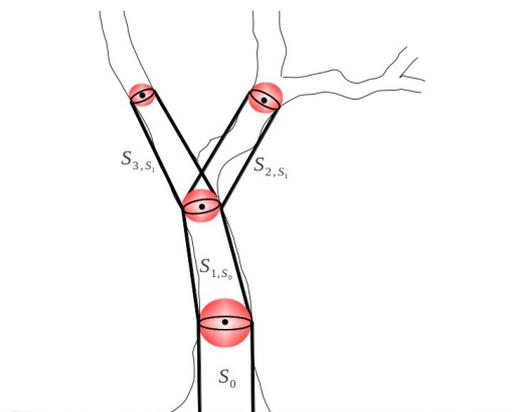


Abbildung 7: Geometrisches Modell mit eingezeichneten Kegelstümpfen und Kugeloberflächen um die Knotenpunkte.

menten mit geringer Variation eine Annäherung mit geringeren Residuen erreicht werden kann. Um die Variation innerhalb eines Segmentes so gering wie möglich zu halten, sollte daher als Grundlage ein abstraktes Segmentmodell mit möglichst kurzen Segmenten dienen. Beispielsweise wäre es in der gezeigten Abbildung förderlich, das Segment S_{2,S_1} in zwei oder mehrere Segmente zu unterteilen. Die Segmente S_{1,S_0} und S_{3,S_1} weisen aufgrund ihrer geringen Variation eine deutlich bessere Anpassung auf. Daher sollte sich die Segmentlänge möglichst an der zu erwartenden Variation des Astes orientieren, also als Grundlage ein abstraktes Segmentmodell dienen, bei dem für dünnere Äste kleinere Segmentlängen gewählt werden, während bei dickeren Ästen bzw. dem Stamm aufgrund der geringeren Variation auch größere Segmentlängen akzeptabel sind.

4.5 Vektormodell

Eine Betrachtungsweise des geometrischen Modells ist eine Vektordarstellung der Segmente. Hierbei wird zwischen den Knotenpunkten eines Segmentes S_{i,S_j} ein Vektor \vec{v}_i aus dem Vektorraum $V \subseteq \mathbb{R}^3$ aufgespannt, sodass $\vec{v}_i := kp_i - kp_j$. Der Stützvektor dieses Vektors \vec{v}_i entspricht hierbei dem Vektor zwischen Koordinatennullpunkt und Parent-Knotenpunkt kp_j . Der Vektor \vec{v}_i gibt neben der Orientierung auch die Höhe des Kegelstumpfes an, da sie der Vektorlänge $\|\vec{v}_i\|$ mit der euklidischen Norm entspricht. Der Radius der Grundfläche des Kegels ergibt sich aus dem Radius r_{k_j} des Parent-Knotens k_j . Entsprechendes gilt für die Deckfläche des Kegels, die sich aus dem Radius r_{k_i} des Child-Knotens k_i ergibt. Da auch die Kugeloberfläche mit diesen Radien beschrieben werden kann, kann ein solches Vektormodell das geometrische Modell vollständig abbilden.

Im Folgenden wird daher zur Vereinfachung die Länge eines Kegelstumpfes nur noch als *Vektorlänge* bezeichnet, sodass diese Länge auch als repräsentativ für die Länge des Segmentes angesehen werden kann.

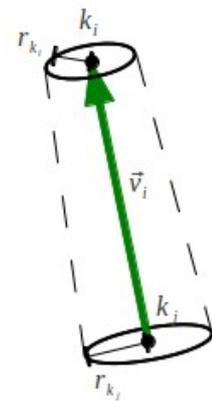


Abbildung 8: Definition der Oberfläche eines Kegelstumpfes über eine Vektordarstellung.

4.5.1 Modelleigenschaften:

Im Folgenden wird nur noch das zuletzt erläuterte Vektormodell betrachtet, dessen Eigenschaften hier näher betrachtet werden sollen.

Die Oberfläche eines Baumes wird durch eine verzweigte Kette von Kegelstümpfen angenähert. An den Verbindungsstücken der Kegel erfolgt die Annäherung über Kugeloberflächen. Die Residuen des Modells ergeben sich aus dem Abstand der tatsächlichen Baumoberfläche zu der Modelloberfläche, also zu dem Abstand des am nächsten liegenden Kegels oder einer Kugel.

Die Orientierung der Segmente ergibt sich aus den Vektoren des Modells, wobei das erste Segment eines Baumes in Richtung der z-Achse zeigt. Über die Orientierung der Vektoren kann die lokale Wuchsrichtung des Baumes angenähert werden. Der Raumwinkel zwischen zwei Vektoren mit Parent-Child-Beziehung liegt stets unter 180° . Eine Verzweigung ergibt sich, falls auf einen Vektor mehrere Child-Vektoren folgen.

Grundvoraussetzung für das Modell ist das Auffinden annähernd runder Querschnitte, wobei hier der Begriff annähernd vom Modellierer geklärt werden muss. Über eine vom Stamm ausgehende Suche könnte das Modell sukzessive aufgebaut werden, wobei darauf zu achten ist, die Vektorlängen entsprechend der zu erwartenden Variation zu wählen, damit die Residuen angemessen gering ausfallen.

5. Modellierung

5.1 Modellierungsansatz

Die Vorbetrachtungen haben zu einem mathematisch gut erfassbaren Vektormodell geführt, welches theoretisch die Oberfläche eines Baumes mit hinreichender Genauigkeit beschreiben kann. Um aus der Theorie ein praktikables Modell herzuleiten, muss eine Möglichkeit gefunden werden, die Segmente bzw. die Knoten aus einer Punktwolke zu generieren.

Es wurde hierzu ein iterativer Ansatz gewählt, der sich in die vier Hauptschritte *Layerselktion* (Auswahl einer Teilmenge aus der Punktwolke), *Clusteranalyse* (Auffinden möglicher Querschnitte), *Knotenbestimmung* (Auffinden des Mittelpunktes eines Querschnittes über Anpassen eines Kreises) und *Validierung* (Plausibilitätsprüfung und Fehleranalyse) unterteilt. Diese Komponenten der Modellierung sollen zunächst genauer erläutert werden, um daraufhin das Vorgehen der Modellierung genauer zu beschreiben.

5.2 Layerselektion

Ein *Layer* im Sinne dieser Arbeit soll eine überschaubare Teilmenge der Punktwolke $L \subseteq P$ darstellen, die zur weiteren Analyse verwendet werden kann. Durch die Auswahl eines solchen Layers soll im Besonderen der Analyseaufwand reduziert werden. Die Selektion des ersten Layers muss von der Auswahl aller weiteren Layer unterschieden werden, da zu Beginn der Modellierung noch keine Knoten als Ausgangspunkte und Vektoren zur Richtungsabschätzung bekannt sind.

5.2.1 Selektion des ersten Layers:

Falls noch kein Knotenpunkt gefunden wurde, wird in einem ersten Schritt gemäß eines Slice-Ansatzes eine horizontale Teilmenge aus der Punktwolke P mit begrenzter vertikaler Ausdehnung h in einer angemessenen Höhe s über Grund (z. B. Brusthöhe) gewählt, um zunächst die Bäume zu identifizieren.

Damit ist der erste Layer wie folgt definiert:

$$L_{0,s,h} := \{ p \in P : s - \frac{h}{2} \leq p_z \leq s + \frac{h}{2} \}$$

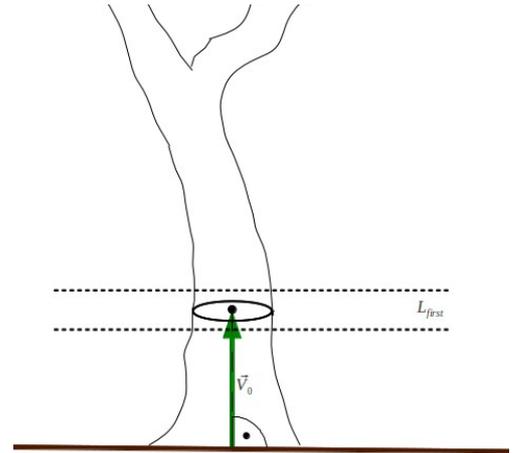


Abbildung 9: Auswahl des ersten Layers mit horizontaler Ausrichtung über der Bodenoberfläche.

5.2.2 Layerselektion:

Alle weiteren Layer richten sich stets nach einem bereits bekannten Knoten k_i und dem entsprechenden Vektor \vec{v}_i . Ein solcher Layer dient dazu, alle möglichen Querschnitte um einen

Knotenpunkt kp_{k_i} zu erfassen. Daher werden alle Punkte aus der Punktwolke P , die in einem bestimmten Abstandsbereich zum Knotenpunkt liegen, als ein sphärischer Teilausschnitt ausgewählt. Zur vereinfachten Auswahl wird ein Hilfspunkt $p_0 \in \mathbb{R}^3$ definiert, sodass die Zugehörigkeit eines Punktes zum Layer sich allein durch die Abstände zu den Punkten kp_{k_i} und p_0 ermitteln lässt.

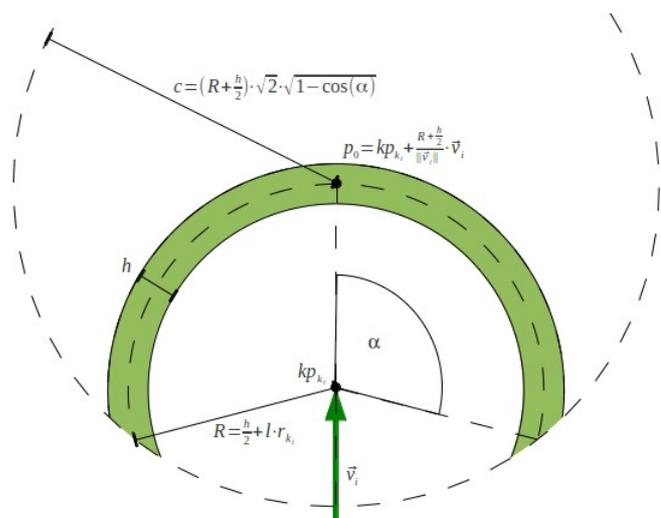


Abbildung 10: Querschnitt der Selektion eines Layers zu einem bekannten Knoten in Richtung des Parent-Vektors. Die Punkte des Layers liegen innerhalb der hier grün markierten Fläche.

Die entscheidenden Parameter sind der Faktor der Vektorlänge $l \in \mathbb{R}^+$, der Raumwinkel α und die Ausdehnung des Layers $h \in \mathbb{R}$. Mithilfe des Radius r_{k_i} des Knotens definiert sich der Radius $R_l := \frac{h}{2} + l \cdot r_{k_i}$. Des Weiteren folgt auf diese Weise der Radius des Hilfskreises $c := (R_l + \frac{h}{2}) \cdot \sqrt{2} \cdot \sqrt{1 - \cos(\alpha)}$ zum Hilfspunkt p_0 , dessen Koordinate sich über den Vektor \vec{v}_i errechnen lässt. Damit wird der Punkt p_0 über die Vektorlänge $\|\vec{v}_i\|$ und den Knotenpunkt kp_{k_i} definiert, sodass $p_0 := kp_{k_i} + \frac{R_l + \frac{h}{2}}{\|\vec{v}_i\|} \cdot \vec{v}_i$ gilt. Ein Layer zu einem Segment S_i lässt sich demnach wie folgt über $L_{k_i, \vec{v}_i, \alpha, l, h} := \{ p \in P : R_l \leq \|p - kp_{k_i}\| \leq R_l + h \wedge \|p - p_0\| \leq c \}$ definieren.

Über diese Layerdefinition ist es möglich, die Auswahl der Punkte um einen Knotenpunkt auf einen bestimmten Raumwinkel zu reduzieren und damit zu gewährleisten, dass die Punkte und damit auch die nächsten Segmente in Wuchsrichtung liegen. Des Weiteren passt sich die Vektorlänge der zu erwartenden Variation an, indem der Radius des Knotens r_{k_i} mit dem Faktor l in die Skalierung des Layers einbezogen wird.

Um die Auswertung des Modells, vor allem bei mehreren unterschiedlichen Vektorlängen zu einem Knoten, effizienter zu gestalten, kann zur Layerselektion eine Vorauswahl an Punkten verwendet werden. Dabei wird zunächst eine Teilmenge $B_{k_i, l_{max}}$ aus der Punktwolke P selektiert, sodass die Entfernung aller Punkte $p \in B_{k_i, l_{max}}$ zum Knotenpunkt kp_{k_i} nicht größer ist als der Radius $R_{l_{max}} := \frac{h}{2} + l_{max} \cdot r_{k_i}$ des maximalen Faktors l_{max} . Diese Teilmenge wird im Folgenden *Branchpoints* genannt und ergibt sich aus $B_{k_i, l_{max}} := \{ p \in P : \|p - kp_{k_i}\| \leq R_{l_{max}} \}$. Die Auswahl eines Layers kann nun mit deutlich weniger zu analysierenden Punkten über $L_{k_i, \vec{v}_i, \alpha, l, h} = \{ p \in B_{k_i, l_{max}} : R_l \leq \|p - kp_{k_i}\| \leq R_l + h \wedge \|p - p_0\| \leq c \}$ erfolgen.

5.3 Clusteranalyse

Um in einem Layer mögliche runde Querschnitte zu finden, werden zunächst Einzelquerschnitte gesucht, deren Punkte topologisch zusammenhängen. Dabei werden die Cluster zu einem aus dem Layer zu wählenden Punkt $p_0 \in L_j$ aufgebaut. Ein Punkt $p \in L_j$ gehört so zum Cluster C mit dem Ausgangspunkt p_0 und dem Abstand $\delta \in \mathbb{R}^+$, falls sein euklidischer Abstand $\|\cdot\|$ zum Punkt p_0 oder einem anderen Punkt des Clusters nicht größer ist als δ , also gilt $C_{i, L_j, p_0, \delta} := \{ p \in L_j : (\|p - p_0\| \leq \delta) \vee (\exists p_c \in C_{i, L_j, p_0, \delta} : \|p - p_c\| \leq \delta) \}$.

Eine vereinfachte Betrachtung dieser formalen Definition ist die eines dreidimensionalen Buffers um die Punkte des Layers. Die einzelnen Buffer werden zu einer dreidimensionalen Hülle

zusammengefasst, was einer Union-Operation entspricht. Alle Punkte, die in einer der disjunkten Teilhüllen liegen, gehören zu einem Cluster.

Der entscheidende Modellparameter bei der Clusteranalyse ist der Abstand δ , da dieser darüber entscheidet, welche Punkte als topologisch zusammenhängend betrachtet werden. Größere δ führen tendenziell zu wenigen, aber größeren Clustern, während bei sehr kleinen δ unter Umständen Querschnitte in viele Einzelcluster zerfallen. Ein Querschnitt sollte sich im Allgemeinen aus vielen Punkten zusammensetzen, sodass ein größerer Cluster zu erwarten ist.

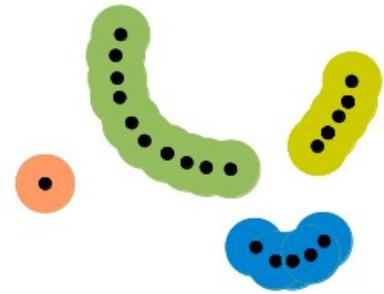


Abbildung 11: Draufsicht auf verschiedene Buffer. Die Punkte innerhalb einer solchen Bufferhülle bilden einen Cluster.

Deshalb können Cluster mit wenigen Punkten vernachlässigt werden. Damit rechnerisch eine spätere Anpassung eines Kreises möglich ist, muss ein Cluster über mindestens drei Punkte verfügen. Um aber auch eine statistisch sinnvolle Auswertung zu ermöglichen, sollten dennoch nur Cluster betrachtet werden, die wesentlich mehr Punkte beinhalten. Ein Cluster mit mindestens $n \in \mathbb{N}_{\geq 3}$ unterschiedlichen Punkten kann also im Folgenden als möglicher Querschnitt betrachtet werden.

5.4 Knotenbestimmung

Die Knotenbestimmung nutzt als Analysegrundlage einen Cluster C_{i,L_j} , der als möglicher Querschnitt identifiziert wurde. Hier soll geprüft werden, ob der Querschnitt annähernd rund ist und ob dieser damit ein Querschnitt eines Child-Segmentes zum vorherigen Segment S_j sein kann. Dabei soll verhindert werden, dass ein Querschnitt eines benachbarten Astes betrachtet wird. Daher muss neben einer Kreisanpassung auch sichergestellt werden, dass die Kreisfläche senkrecht zur Vektorrichtung liegt.

Aus diesen Gründen wird die Knotenbestimmung in die vier Teilschritte *Anpassen einer Kugeloberfläche*, *Rotation um den Kugelmittelpunkt*, *Anpassen eines Kreises in 2D* und *Rückrotation des Kreismittelpunktes* zerlegt, die nacheinander ausgeführt werden. Für den ersten Layer beschränkt sich die Knotenbestimmung auf *Anpassen eines Kreises in 2D*, wobei die z-Koordinate über den Mittelwert der z-Werte der Clusterpunkte hergeleitet wird.

5.4.1 Schritt 1: Anpassen einer Kugeloberfläche:

Da davon ausgegangen wird, dass die Punkte im Cluster C_{i,L_j} annähernd kreisförmig angeordnet sind, kann auch näherungsweise eine Kugeloberfläche angepasst werden, die einen vorläufigen Kreismittelpunkt liefert.

Mit der Idee von BRÜNNER (2009) zur Anpassung eines Kreises an drei Punkte wird hier über die Koordinatengleichung einer Kugel $r = \sqrt{(x-x_m)^2 + (y-y_m)^2 + (z-z_m)^2}$ ein lineares Gleichungssystem hergeleitet, welches zur Kugelanpassung genutzt wird. Dabei steht $r \in \mathbb{R}^+$ für den Kugelradius, $(x_m, y_m, z_m)^T$ für die Raumkoordinate des Kugelmittelpunktes und $(x, y, z)^T$ für einen beliebigen Punkt auf der Kugeloberfläche bzw. im Cluster C_{i,L_j} . Das Gleichungssystem kann damit wie folgt hergeleitet werden:

$$\begin{aligned} r &= \sqrt{(x-x_m)^2 + (y-y_m)^2 + (z-z_m)^2} \\ r^2 &= (x-x_m)^2 + (y-y_m)^2 + (z-z_m)^2 \\ -2x_m \cdot x - 2y_m \cdot y - 2z_m \cdot z + x_m^2 + y_m^2 + z_m^2 - r^2 &= -(x^2 + y^2 + z^2) \\ A \cdot x + B \cdot y + C \cdot z + D &= -(x^2 + y^2 + z^2), \text{ mit} \\ A &:= -2 \cdot x_m, \quad B := -2 \cdot y_m, \quad C := -2 \cdot z_m \quad \text{und} \quad D := x_m^2 + y_m^2 + z_m^2 - r^2 \end{aligned}$$

Die Lösung des Gleichungssystems über die Methode der kleinsten Quadrate führt zu der gesuchten Mittelpunktkoordinate mit $x_m = -0.5 \cdot A$, $y_m = -0.5 \cdot B$, $z_m = -0.5 \cdot C$ und dem Kugelradius $r = \sqrt{x_m^2 + y_m^2 + z_m^2 - D}$.

Über diese Anpassung ist eine Minimierung der Residuen, also dem Abstand der Punkte zu der Kugeloberfläche, gegeben. Mit dem Kugelmittelpunkt kann im Folgenden weitergearbeitet werden, da dieser, falls es sich bei dem Cluster um einen der gesuchten Querschnitte handelt, annähernd dem Kreismittelpunkt entspricht.

5.4.2 Schritt 2: Rotation um den Kugelmittelpunkt:

Auf Grundlage des im vorherigen Schritt hergeleiteten Kugelmittelpunktes kann nun der Cluster C_{i,L_j} in Richtung der z-Achse rotiert werden. Auf diese Weise ist es möglich, die z-Koordinate der Clusterpunkte zu ignorieren und damit eine Anpassung eines Kreises in der Ebene zu ermöglichen. Da der Kugelmittelpunkt bei einer annähernd runden Anordnung der Clusterpunkte in etwa dem Kreismittelpunkt entsprechen sollte, ist anzunehmen, dass die Punkte der Astoberfläche nach der Rotation nahezu senkrecht übereinander liegen, sodass die Verzerrung nur gering ist.

Um eine Rotation vorzubereiten, muss zunächst der Rotationswinkel α bekannt sein. Dieser wird über die Formel $\angle(\vec{v}, \vec{w}) = \arccos\left(\frac{\vec{v}^T \cdot \vec{w}}{\|\vec{v}\| \cdot \|\vec{w}\|}\right) = \arccos\left(\frac{v_x \cdot w_x + v_y \cdot w_y + v_z \cdot w_z}{\sqrt{v_x^2 + v_y^2 + v_z^2} \cdot \sqrt{w_x^2 + w_y^2 + w_z^2}}\right)$ zur Berechnung von

Raumwinkeln hergeleitet. Damit entspricht der Rotationswinkel α dem Winkel zwischen dem Vektor $\vec{v} := b_m - kp_{k_j}$

und der z-Achse $\vec{z}_0 = (0, 0, 1)^T$, sodass gilt:

$$\alpha = \angle(\vec{z}_0, \vec{v}) = \arccos\left(\frac{0 \cdot v_x + 0 \cdot v_y + 1 \cdot v_z}{\sqrt{0^2 + 0^2 + 1^2} \cdot \sqrt{v_x^2 + v_y^2 + v_z^2}}\right) = \arccos\left(\frac{v_z}{\sqrt{v_x^2 + v_y^2 + v_z^2}}\right)$$

Die Rotation wird über den Drehvektor \vec{o} durchgeführt, der orthogonal zu den Vektoren \vec{z}_0 und \vec{v} stehen muss. Die Orthogonalität zwischen zwei Vektoren ist gegeben, falls ihr Skalarprodukt gleich Null ist (vgl. PESTER 2013), also gilt $\vec{o} \cdot \vec{v} = 0$ und $\vec{o} \cdot \vec{z}_0 = 0$. Damit lässt sich der Drehvektor \vec{o} wie folgt herleiten:

Es gilt $\vec{o} \cdot \vec{z}_0 = o_x \cdot z_{0x} + o_y \cdot z_{0y} + o_z \cdot z_{0z} = o_x \cdot 0 + o_y \cdot 0 + o_z \cdot 1 = o_z$, also $o_z = 0$. Des Weiteren gilt $\vec{o} \cdot \vec{v} = o_x \cdot v_x + o_y \cdot v_y + o_z \cdot v_z = o_x \cdot v_x + o_y \cdot v_y + 0 \cdot v_z = o_x \cdot v_x + o_y \cdot v_y$. Folglich gilt die Gleichung $o_x \cdot v_x = -o_y \cdot v_y$, die mit $o_x = v_y$ und $o_y = -v_x$ gelöst werden kann, sodass der Drehvektor mit $\vec{o} = (v_y, -v_x, 0)^T$ gegeben ist.

Die eigentliche Rotation der Punkte des Clusters C_i erfolgt über eine Rotationsmatrix M (vgl. TORNAU 2013) durch Matrixmultiplikation. Die Rotationsmatrix wird über den Drehwinkel α und Drehvektor \vec{o} aufgebaut, sodass gilt:

$$M_{\alpha, \vec{o}} = \begin{pmatrix} \cos(\alpha) + \vec{o}_x^2 \cdot (1 - \cos(\alpha)) & \vec{o}_x \cdot \vec{o}_y \cdot (1 - \cos(\alpha)) - \vec{o}_z \cdot \sin(\alpha) & \vec{o}_x \cdot \vec{o}_z \cdot (1 - \cos(\alpha)) + \vec{o}_y \cdot \sin(\alpha) \\ \vec{o}_y \cdot \vec{o}_x \cdot (1 - \cos(\alpha)) + \vec{o}_z \cdot \sin(\alpha) & \cos(\alpha) + \vec{o}_y^2 \cdot (1 - \cos(\alpha)) & \vec{o}_y \cdot \vec{o}_z \cdot (1 - \cos(\alpha)) - \vec{o}_x \cdot \sin(\alpha) \\ \vec{o}_z \cdot \vec{o}_x \cdot (1 - \cos(\alpha)) - \vec{o}_y \cdot \sin(\alpha) & \vec{o}_z \cdot \vec{o}_y \cdot (1 - \cos(\alpha)) + \vec{o}_x \cdot \sin(\alpha) & \cos(\alpha) + \vec{o}_z^2 \cdot (1 - \cos(\alpha)) \end{pmatrix}$$

Die Punkte des rotierten Clusters C' ergeben sich über die Gleichung $C' = M_{\alpha, \vec{o}} \cdot C$ mit:

$$C = \begin{pmatrix} p_{1x} & p_{1y} & p_{1z} \\ p_{2x} & p_{2y} & p_{2z} \\ \vdots & \vdots & \vdots \\ p_{nx} & p_{ny} & p_{nz} \end{pmatrix}, \text{ wobei } p_{1 \dots n} = (p_{1 \dots n_x}, p_{1 \dots n_y}, p_{1 \dots n_z})^T \in C_i$$

5.4.3 Schritt 3: Anpassen eines Kreises in 2D:

Die weitere zweidimensionale Analyse kann nun mit dem rotierten Cluster C' erfolgen, indem – wie bei BRÜNNER, A. (2009) im ersten Schritt der Knotenbestimmung – ein Kreis an dem rotierten Cluster C' über die Koordinatengleichung $r = \sqrt{(x - x_m)^2 + (y - y_m)^2}$, den Kreisradius $r \in \mathbb{R}^+$ und den Kreismittelpunkt $(x_m, y_m)^T$ angepasst wird. Für einen beliebigen Punkt $(x, y)^T \in C'$ gilt also:

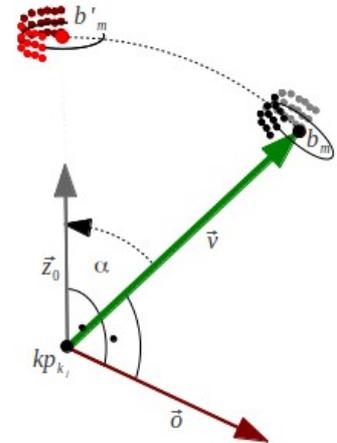


Abbildung 12: Rotation des Clusters zur z-Achse über den orthogonalen Vektor.

$$\begin{aligned}
r &= \sqrt{(x-x_m)^2 + (y-y_m)^2} \\
r^2 &= (x-x_m)^2 + (y-y_m)^2 \\
-2x_m \cdot x - 2y_m \cdot y + x_m^2 + y_m^2 - r^2 &= -(x^2 + y^2) \\
A \cdot x + B \cdot y + D &= -(x^2 + y^2) \text{ mit} \\
A &:= -2 \cdot x_m, \quad B := -2 \cdot y_m \text{ und } D := x_m^2 + y_m^2 - r^2
\end{aligned}$$

Durch Lösen des Gleichungssystems für alle Punkte des Clusters ergeben sich die Koordinatenwerte $x_m = -0.5 \cdot A$, $y_m = -0.5 \cdot B$ und der Kreisradius $r = \sqrt{x_m^2 + y_m^2 - D}$.

Der Kreisradius r entspricht nun dem Radius r_{k_i} des gesuchten Knotens k_i . Die ermittelte Mittelpunktkoordinate befindet sich zunächst nur in einer Ebene. Daher wird die Raumkoordinate kp'_{k_i} des vorläufigen Knotens k'_i aus der Mittelpunktkoordinate und dem gemittelten z-Wert des Clusters C' als $kp'_{k_i} := (x_m, y_m, \bar{z})^T$ definiert.

5.4.4 Schritt 4: Rückrotation des Kreismittelpunktes:

Da die Raumkoordinate des Kreismittelpunktes kp'_{k_i} auf den rotierten Punkten des Clusters C' beruhen, muss der Punkt auf die tatsächliche Koordinate zurück in den Cluster C rotiert werden. Hierzu wird die Rotationsmatrix $M_{-\alpha, \vec{o}}$ aus dem Drehvektor \vec{o} und dem invertierten Drehwinkel $-\alpha$ aus dem zweiten Schritt hergeleitet. Die Rückrotation des Punktes erfolgt demnach nach der Formel $kp_{k_i} = M_{\alpha, \vec{o}} \cdot kp'_{k_i}$.

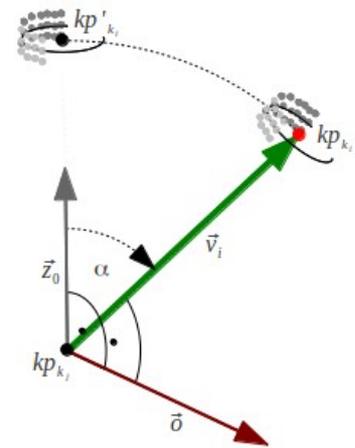


Abbildung 13: Rotation des Kreismittelpunktes zurück zum Cluster zur Bestimmung des tatsächlichen Knotenpunktes.

Folglich definiert sich der Knoten k_i aus dem hier bestimmten Knotenpunkt kp_{k_i} und dem Radius r_{k_i} des im Schritt 3 bestimmten Kreises. Der Vektor vom Parent-Knotenpunkt kp_{k_j} zum Child-Knotenpunkt kp_{k_i} entspricht dem gesuchten Vektor \vec{v}_i .

5.4.5 Vorteile des Vorgehens der Knotenbestimmung:

Eine vereinfachte Anpassung einer Kugeloberfläche aus Schritt 1 ohne die Schritte 2 bis 4 und die Übernahme des Kugelmittelpunktes mit dem Kugelradius als Knoten kp_{k_i} erscheint zwar auch denkbar, führt aber zu mehreren Nachteilen. Hier ist zunächst die weniger optimale Anpassung des Radius zu nennen, da ein Einfluss der Layerdicke auf die Mittelpunktbestimmung

gegeben ist. Folglich ist der Radius auch bei einer perfekten Kreisform nicht korrekt angepasst, sodass die Residuen räumlich abhängig sind. Des Weiteren besteht der Nachteil, dass bei parallel verlaufenden Ästen die Gefahr besteht, dass eine Kugelanpassung an den parallel verlaufenden Ast wahrscheinlich ist. Damit wird irrtümlicherweise eine Verzweigung zu diesem Ast angenommen. Auf diesen Effekt und einen weiteren Ansatz zur Lösung dieses Problems wird im Kapitel 5.6 eingegangen.

5.5 Validierung

Nach dem Auffinden eines neuen Knotens muss eine ausführliche Plausibilitätsüberprüfung durchgeführt werden, um sicherzustellen, dass tatsächlich ein Segment vorliegt. Hierzu wird sowohl die Kreisanpassung überprüft, als auch getestet, ob der neu gefundene Knoten kein Widerspruch zum bestehenden Modell darstellt, also die Knoten disjunkt sind bzw. der neue Knoten nicht innerhalb eines Segmentes liegt.

5.5.1 Plausibilitätsprüfung der Kreisanpassung:

Die Plausibilität der Kreisanpassung kann über verschiedene Verfahren geprüft werden. Hier sind mehrere Vorschläge aufgeführt, die jedoch auch durch andere Verfahren ersetzt werden können.

Der mittlere quadratische Fehler mse stellt die Güte der Kreisanpassung dar, da er sensitiv auf Abweichungen von der Kreisform reagieren kann. Dazu wird das Verhältnis der erklärten Varianz $var_E = R$ zur Gesamtvarianz

$var_G = \sqrt{(x - x_m)^2 + (y - y_m)^2}$, mit dem Kreisradius

$R \in \mathbb{R}^+$, den Kreismittelpunktkoordinaten (x_m, y_m) und den Punktkoordinaten (x, y) betrachtet. Der mittlere quadratische Fehler definiert sich damit nach $mse = 1 - \sum \frac{(var_{ges} - R)^2}{var_{ges}^2}$ und liegt damit im Wertebereich $[0, 1]$. Ein Ausschlusskriterium für einen Knoten kann also die Voraussetzung eines Mindestwertes für den mse sein, wie $mse \geq 0.95$ für einen maximalen mittleren Fehler von 5%.

Zusätzlich macht eine Betrachtung des maximalen relativen Fehlers res_{max} Sinn, um die Kreisform des Clusters zu garantieren. Hierzu wird das Verhältnis der maximalen Gesamtvarianz zur erklärten Varianz, also dem Radius, betrachtet. Der maximale relative Fehler

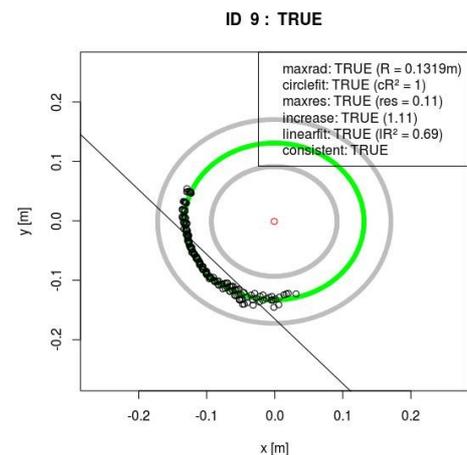


Abbildung 14: Plausibilitätsprüfung der Kreisanpassung.

$res_{max} = \frac{\max(\text{var}_a)}{R} - 1$ hat einen Wertebereich $[0, 1]$ und kann als Ausschlusskriterium dienen. Dabei wird z. B. ein maximaler $res_{max} = 0.3$ angenommen, um zu garantieren, dass kein Residuum größer als 30% des Radius ist.

Falls die Punkte zur Kreisanpassung annähernd geradlinig verlaufen wird für gewöhnlich ein Kreis mit sehr großem Radius angepasst. Dadurch fallen der mse und res_{max} sehr günstig aus, da der Anteil der erklärten Varianz an der Gesamtvarianz durch den großen Radius sehr hoch ist. Da das Modell kreisförmige Querschnitte voraussetzt, müssen andere Ausschlussverfahren gewählt werden, die die Linearität der Punkte prüfen. Eine Möglichkeit ist die Anpassung einer Geraden an die Punkte durch lineare Regression und anschließende Prüfung des Bestimmtheitsmaßes R^2 . So könnte ein Knotenpunkt verworfen werden, der eine signifikante Linearität aufweist, falls z. B. $R^2 \geq 0.95$.

5.5.2 Plausibilitätsprüfung:

Neben einer reinen Betrachtung der Güte der Kreisanpassung sollte die Plausibilität der Ergebnisse geprüft werden. Darunter fällt neben der Prüfung, ob das Modell durch Aufnahme des neuen Punktes widersprüchlich wird, auch eine Prüfung der Werte aufgrund von Beobachtungen und Erfahrungen.

Eine erfahrungsbasierte Prüfung ist z. B. die Entscheidung, ob ein Radius im Untersuchungsgebiet überhaupt auftreten kann. Falls ein zu großer Radius angenommen wird, wird nicht nur die Modellierung ungenau, sondern bei der Suche des nächsten Knotenpunktes wird ein viel zu großer Layer ausgewählt, da diese abhängig vom Radius ist. Daher sollte ein Maximaler Radius r_{max} definiert werden, der nicht überschritten werden darf.

Allgemein kann davon ausgegangen werden, dass Äste sich in Wuchsrichtung leicht verzweigen. Daher sollte geprüft werden ob das Verhältnis $w := \frac{r_i}{r_j}$, das Verhältnis zwischen Parent- und Child-Radius, einen maximalen Wert nicht überschreitet. Auf diese Weise kann garantiert werden, dass der Astdurchmesser nicht ungewöhnlich stark zunimmt, indem w nahe 1 gewählt wird.

Eine Widerspruchsprüfung des Modells erfolgt durch Testen, ob ein neuer Knoten im Widerspruch zum bereits bestehenden Modell liegt. Hierzu muss sichergestellt werden, dass sich die Kugel des neuen Knotens nicht mit der Kugel eines anderen Knotens oder mit dem Kegelstumpf eines vorhandenen Segmentes überschneidet bzw. innerhalb liegt.

Der neue Knoten mit Knotenpunkt p und Radius r überschneidet sich damit mit einer Kugel eines bestehenden Knotens k_i , falls $\|kp_{k_i} - p\| < r_{k_i} + r$. Zudem schneidet er sich mit einem

Segment, falls ein Kegelstumpf des bestehenden Modells existiert mit $h < R_{mod} + r$. Dabei bezeichnet h den Abstand des Punktes p zum Vektor und R_{mod} den Radius des Kegelstumpfes auf Höhe des Punktes. Für die Prüfung, ob sich der Punkt in einem Kegelstumpf befindet, wird die im Folgenden beschriebene Validierung verwendet, die klären soll, in welchem Abstand sich ein Punkt zur Modelloberfläche befindet.

5.5.3 Validierung eines Punktes an einem Segment:

Um zu bestimmen, ob ein beliebiger Punkt $p \in \mathbb{R}^3$ durch das Vektormodell beschrieben werden kann, muss sein Abstand zur Modelloberfläche bestimmt werden. Da sich die Oberfläche vornehmlich aus den Kegelstümpfen ergibt, wird neben der Fehleranalyse der Cluster über den mse die Fehleranalyse des Punktes p zu der Oberfläche des Kegelstumpfes benötigt. Zu diesem Zweck wird der Vektor \vec{v}_i eines Kegelstumpfes als Referenz definiert, sodass der Abstand h des Punktes zu diesem Vektor zu bestimmen ist. Da der Vektor \vec{v}_i durch die Knoten k_j und k_i definiert wird, spannen die Knotenpunkte $k_{p_{k_j}}$, $k_{p_{k_i}}$ und der Punkt p ein Dreieck auf, falls keiner der

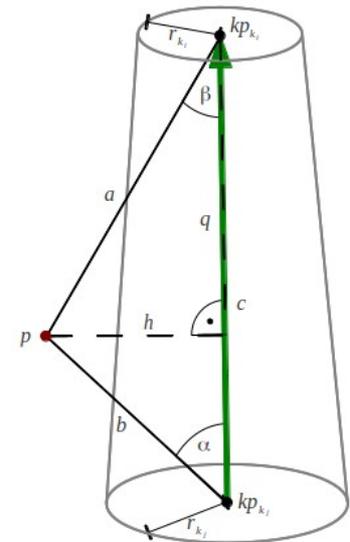


Abbildung 15: Bestimmung des Punktabstandes zu einer Kegelstumpfoberfläche.

Punkte mit einem der anderen identisch ist. Damit ergeben sich die Seitenlängen des Dreiecks $a = \|kp_{k_i} - p\|$, $b = \|p - kp_{k_j}\|$ und $c = \|\vec{v}_i\|$. Die Höhe des Dreiecks zwischen dem Punkt p und dem Vektor \vec{v}_i entspricht dem Abstand h des Punktes zum Vektor. Die Seiten a , h und q definieren auf diese Weise ein rechtwinkliges Dreieck. Damit die Oberfläche des Kegelstumpfes dazu geeignet ist, den Punkt zu repräsentieren, muss zusätzlich gelten, dass der Punkt neben dem Vektor liegt, also $\beta \leq 90^\circ$ und $q \leq c$. Der Abstand h zum Punkt p wird nun über Winkelsätze hergeleitet, sodass $h = a \cdot \sin(\beta)$ mit $\beta = \arccos\left(\frac{a^2 + c^2 - b^2}{2 \cdot a \cdot c}\right)$ gilt. Um die für diesen Punkt repräsentative Oberfläche des Kegelstumpfes zu bestimmen, wird der Radius R_{mod} des Kegelstumpfes auf Höhe des Punktes benötigt, welcher durch das gewichtete arithmetische Mittel der Knotenradien r_{k_i} und r_{k_j} mit dem Gewichtungsfaktor $\frac{a}{c}$ über die Formel $R_{mod} = r_{k_i} + \frac{a}{c} \cdot (r_{k_j} - r_{k_i})$ berechnet werden kann.

Der relative Fehler des Modells ergibt sich über $res = \frac{h}{R_{mod}} - 1$, der neben der relativen Abweichung vom Modell auch angibt, ob der Punkt außerhalb ($res > 0$) oder innerhalb ($res < 0$) der Modellhülle liegt.

5.5.4 Validierung aller Punkte:

Um alle Punkte mithilfe des zuvor genannten Vorgehens zu validieren, werden nach jeder Knotenbestimmung alle um den Knotenpunkt kp_{k_j} liegenden Punkte, also die der in Kapitel 5.2 (*Layerselektion*) definierten Teilmenge $B_{k_j, l_{max}}$, für den neuen Vektor \vec{v}_i zwischen den Knoten k_j und k_i untersucht. Dadurch werden aus Gründen der Performance nur die Punkte analysiert, die sich in der Nähe des Modells befinden. Da ein Punkt durch mehrere Vektoren beschrieben werden könnte (z. B. an einer Astgabelung), sollte am Ende der Modellierung eine *Postvalidierung* vorgenommen werden, bei dem jedem Punkt das Segment mit dem geringsten relativen Fehler zugeordnet wird.

Diese Art der Validierung eignet sich ausschließlich zur Betrachtung der Fehlerverteilung des Modells, denn es kann daraus nicht abgeleitet werden, ob ein Messpunkt zur Oberfläche des analysierten Baumes gehört, da keine absolute Referenz existiert und keine absolute Grenze zwischen erklärtem Punkt und nicht erklärtem Punkt gezogen werden kann.

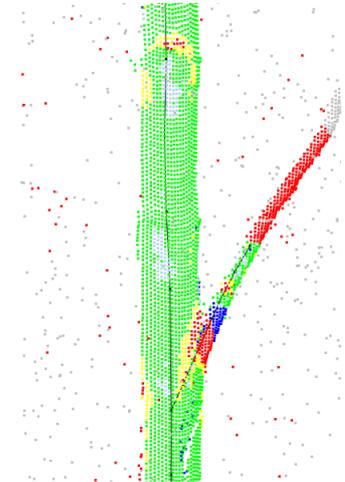


Abbildung 16: Fehlerbetrachtung des Modells; Residuen $< 5\%$ grün, 5 bis 10% gelb, -5 bis -10% hellblau, $> 10\%$ rot, $< -10\%$ blau und nicht analysierte Punkte grau.

5.6 Modellablauf

Die in den vorherigen Abschnitten erläuterten Teilschritte stellen Einzelmodule der Modellierung dar und können daher nacheinander aufgerufen werden. Dabei spielt vor allem der Aufbau der Modelllogik, die Abfolge der Schritte und die Auswahl der Ausgangsknoten eine wichtige Rolle. Im Folgenden wird daher der Modellablauf näher beschrieben und die Entscheidungskette erläutert.

5.6.1 Aufbau der Modelllogik:

Da zu Beginn der Modellierung noch keine Knoten bekannt sind, muss die Auswahl der Punkte über die Definition der Layerselektion des ersten Layers erfolgen. Dies liefert eine Teilmenge an Punkten, was zum Auffinden der Baumstämme genutzt werden kann. Auf diesen ersten Layer $L_{0,h,d}$ kann nun der Schritt der Clusteranalyse angewendet werden, sodass man eine Anzahl von Clustern $C_{i,L_0,p_0,\delta}$ erhält, die jeweils für einen Stammquerschnitt stehen sollten. Danach wird für jeden der Cluster die Knotenpunktbestimmung mit anschließender Validie-

rung durchgeführt, sodass man für jeden Stammquerschnitt einen Knotenpunkt mit annähernd der Höhe des angegebenen Parameters s erhält. Um eine vertikale Wuchsrichtung zu erhalten und den Stammansatz abzubilden wird der jeweilige Knotenpunkt auf die Höhe 0 – also auf die angenommene Bodenoberfläche – projiziert. Die Punkte $B_0 = \{ p \in P : 0 \leq p_z \leq h \}$ zwischen dem ersten Layer und der Bodenfläche werden daraufhin validiert.

Das Ergebnis dieser ersten Schritte sind die Knoten und Vektoren der Stammansätze. Auf diese Weise wird die Modelllogik für alle weiteren Schritte aufgebaut und die Identifizierung eines Baumes kann über den ersten Knoten erfolgen, also einen Knoten, der keine Parent-Beziehung aufweist und damit den Stammansatz darstellt.

5.6.2 Auffinden weiterer Vektoren:

Für die weitere Modellierung kann nun auf die bereits bestehenden Knoten zurückgegriffen werden, sodass das Modell iterativ erweitert werden kann. In jeder Iteration werden daher die Schritte *Layerselektion*, *Clusteranalyse*, *Knotenpunktbestimmung* und *Validierung* nacheinander ausgeführt.

Zu einem bestehenden Knoten k_i , bei dem noch nicht alle Vektorlängen abgearbeitet wurden, werden zunächst alle in der Nähe liegenden Punkte, also $B_{k_i, l_{max}}$ bestimmt, falls diese nicht bereits bekannt sind. Danach wird von den Punkten

aus $B_{k_i, l_{max}}$ ein Layer $L_{k_i, \vec{v}_i, \alpha, l, h}$ mit der gewünschten Vektorlänge l , der Ausdehnung h und Raumwinkel α , ausgehend vom Knotenpunkt kp_{k_i} in Richtung \vec{v}_i ausgewählt. Die Punkte des Layers können nun geclustert werden, um mögliche Querschnitte aufzufinden. Die als Querschnitte identifizierten Cluster werden nacheinander zur Knotenpunktbestimmung genutzt, wobei nur die Knoten in das Modell aufgenommen werden, die entsprechend den Ergebnissen der Validierung sinnvoll erscheinen und dem bestehenden Modell nicht widersprechen. Nachdem alle Knoten der neuen Cluster bestimmt wurden, kann erneut ein Knoten gewählt werden.

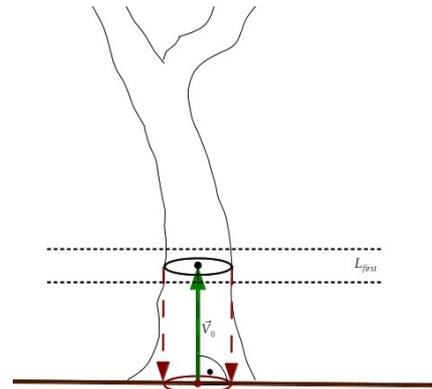


Abbildung 17: Selektion des ersten Layers, auffinden von Stammmittelpunkten und Projektion der Knoten auf die Bodenoberfläche.

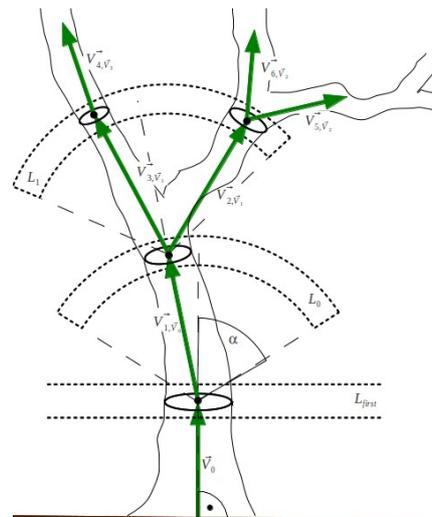


Abbildung 18: Modellaufbau durch Layerselektion.

5.7 Entscheidungskette

Der schleifenhafte Modellablauf setzt geeignete Auswahl- und Abbruchkriterien voraus, die im Folgenden für jeden Teilschritt separat erläutert werden.

5.7.1 Auswahl des nächsten zu analysierenden Knotens:

Von besonderer Bedeutung ist – weil eine Parallelisierung aus Gründen der Modellkonsistenz nicht vorgesehen ist – die Auswahl des nächsten zu analysierenden Knotens, da dieser die Auswahl des nächsten Layers bestimmt.

Allgemein ist bei der Auswahl des Knotens darauf zu achten, dass die Vektorlänge und damit die Entfernung zu den Punkten des Layers möglichst klein ist, da sonst – wie in der Grafik gezeigt – ein „Überspringen“ der Vektoren droht. Dieser Effekt entsteht vor allem bei nahe beieinander liegenden oder auch parallelen Ästen. Wenn einer der Äste dicker als der andere ist, also die Radien der Knoten größer sind, erfolgt die Auswahl der Punkte des Layers auch in den Bereich der kleineren Äste hinein. Dies führt, falls der größere Ast zuerst ausgewertet wird, dazu, dass nach jedem Knoten ein Vektor zum kleineren Ast abzweigt. Die Bestimmung der Knotenpunkte mit der in Kapitel 5.4 beschriebenen Drehung verringert zwar die Wahrscheinlichkeit, dass solche nicht in Vektorrichtung liegende Äste erfasst werden, schließt den Effekt aber nicht aus.

Daher ist eine effektivere Variante zur Lösung dieses Problems immer, den Knoten auszuwählen, der eine möglichst kleine Vektorlänge garantiert. Dadurch ist die Entfernung der ausgewählten Punkte zum Knoten so gering, dass wahrscheinlich keine Punkte eines anderen entfernt liegenden Astes gewählt werden. Falls dennoch Punkte eines anderen größeren Astes erfasst werden, kann davon ausgegangen werden, dass der Querschnitt nicht vollständig ist und damit eine Kreisanpassung fehlschlägt. Aus diesen Gründen ist ein Überspringen seitens des kleineren Astes unwahrscheinlich. Ein späteres Überspringen seitens des größeren Astes ist daraufhin auch nicht möglich, da dabei gefundene Knoten innerhalb des bestehenden Modells liegen und damit verworfen würden.

Dieser Ansatz impliziert, dass das Vorgehen der Punktauswahl nicht geordnet vom Stamm beginnend bis zur Krone

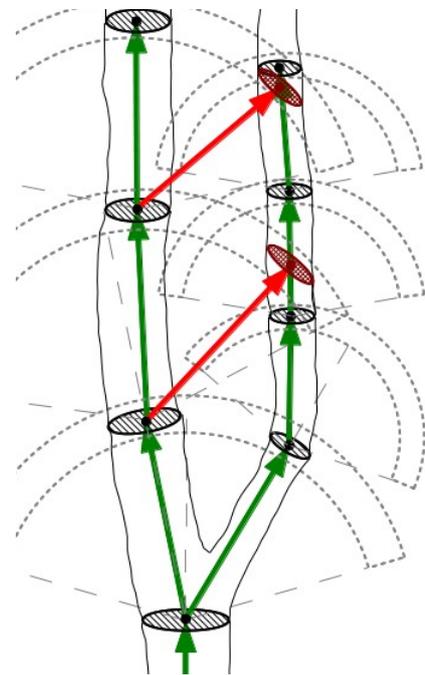


Abbildung 19: Parallele Äste können zum "Überspringen" der Vektoren führen, wenn die Äste Knoten mit größerem Radius zuerst abgearbeitet werden. Gewünschte Modellierung grün und "Überspringen" rot markiert.

erfolgt, sondern dass sich die Analyse stets an den kleinsten analysierbaren Ästen orientiert. Dies führt auch dazu, dass die Analyse bei mehreren Bäumen einmal an einem und ein anderes Mal an einem anderen Baum erfolgen kann. Über die Parent-Child-Beziehungen kann jedoch ein Baum eindeutig identifiziert und rekonstruiert werden.

5.7.2 Clusteranalyse:

Da alle Cluster eines Layers disjunkt sind, hat die Reihenfolge der Auswertung keinen Einfluss auf das Ergebnis und die Clusteranalyse kann parallel verlaufen.

5.7.3 Knotenbestimmung:

Die Reihenfolge der Cluster zur Knotenbestimmung hat jedoch einen größeren Einfluss, da diese über den als nächstes einzufügenden Knoten entscheidet. Allgemein ist davon auszugehen, dass Cluster mit vielen Punkten eine höhere Genauigkeit bei der Knotenbestimmung erreichen, da die Kreisanpassung weniger Unsicherheiten aufweist. Daher sollten in einem Layer zunächst die Cluster für die Knotenbestimmung genutzt werden, die über viele Punkte verfügen. Das hat zudem den Vorteil, dass bei Querschnitten mit Schatteneffekten, bei denen der Querschnitt in zwei oder mehrere Cluster zerfällt, die Anpassung an den größeren Cluster erfolgt, der höchstwahrscheinlich den Radius gut genug approximiert, um auch die anderen Teilquerschnitte zu beschreiben. Die Knotenbestimmung eines kleineren Clusters führt danach entweder dazu, dass der Knoten innerhalb des Knotens des größeren Clusters liegt, sodass er verworfen, oder zu einem neuen Zweig wird.

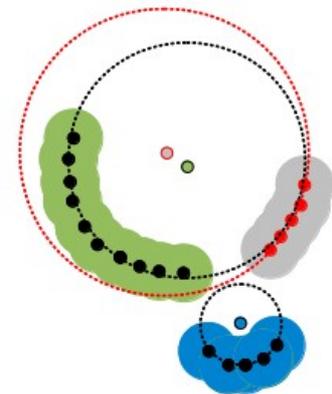


Abbildung 20: Kreisanpassung (schwarzer Kreis) an größeren Cluster (grün) erzielt auch gute Anpassung an gegenüberliegenden Cluster (grau), während Anpassung an kleineren Cluster (roter Kreis) den größeren kaum approximiert.

5.7.4 Validierung:

Die Validierung während der Modellierung erfüllt mehrere Aufgaben, da neben der Information über die Güte einer neuen Querschnittsanpassung auch Punkte, die durch das Modell gut beschrieben werden, aus der weiteren Analyse ausgeklammert werden können. Da ein neuer Layer aus den Punkten $B_{k_i, l_{max}}$ ausgewählt wird, ist bei der Layerselektion nur ein Abgleich mit den validierten Punkten aus dieser Menge erforderlich, was den Analyseaufwand begrenzt. Hilfreich wird dieses Vorgehen vor allem, wenn bereits ein Querschnitt mit einer ho-

hen Vektorlänge in den Branchpoints $B_{k_i, l_{max}}$ zum Knoten k_i gefunden wurde, sodass ein Großteil der Punkte bereits durch den Vektor approximiert wurde. Folglich wird die Menge der Punkte bei der Layerselektion wahrscheinlich so gering sein, dass eine Knotenbestimmung nicht mehr erforderlich ist.

Die bereits validierten Punkte aus allen Branchpoints können nach Ende der eigentlichen Modellierung auch zu der Postvalidierung verwendet werden, indem für jeden Punkt aus allen Branchpoints der Wert mit dem geringsten relativen Fehler herausgesucht wird. Das Ergebnis ist die Fehlerverteilung für alle analysierten Punkte.

5.7.5 Terminierung des Modells:

Die Modellierung eines Datensatzes endet erst, wenn zu jedem bestehenden Knoten kein weiterer Child-Knoten mehr gefunden werden kann. Hierzu werden alle gewünschten Vektorlängen zur Layerselektion abgearbeitet.

Es werden keine neuen Knoten mehr ins Modell integriert, falls entweder keine ausreichend großen Cluster mehr gefunden werden, da z. B. die Querschnitte der Äste zu klein sind, oder die Anpassung eines Knotens keine akzeptablen Ergebnisse mehr liefert. Darunter fällt eine unzureichende Kreisanpassung oder ein Widerspruch zum bestehenden Modell.

Da die aufgespannten Kugeln für alle Knoten disjunkt sind, wird für jeden Knoten ein Teil des Raumes der Punktwolke P für das Setzen eines weiteren Knotens ausgeschlossen. Dies ist auch der Fall, wenn der neue Knotenpunkt außerhalb des Wertebereiches der Punktwolke liegt, da die aufgespannte Kugel in den Raum hineinreichen muss. Weil der Raum dadurch irgendwann gefüllt sein wird und die Anzahl der verfügbaren Vektorlängen zur Layerselektion ebenfalls begrenzt ist, wird in jedem Fall die Modellierung nach endlicher Zeit terminieren.

5.8 Parametrisierung

Die Auswahl aller Modellparameter muss für ein optimales Modellierungsergebnis an die Problemstellung angepasst werden. Dabei sollten die Parameter aufeinander abgestimmt sein, da ein bestimmtes Modellverhalten über verschiedene Parameterkombinationen erreicht werden kann. Die Modellparameter sind den einzelnen Schritten des Modells zugeordnet, sodass im Folgenden auf die Bedeutung der Parameter auf den jeweiligen Schritt eingegangen wird und der Einfluss auf bestimmte Modellierungseffekte geklärt wird.

5.8.1 Layerselektion:

Über die *Höhe des ersten Layers* s kann gesteuert werden, wie lang das erste Segment, also der Stammansatz ist. Daher wird bei einem kleinen Wert dieses Parameters die Modellierung des Stammansatzes genauer, jedoch kann dies auch dazu führen, dass Stämme, die bodennah beschattet sind, nicht erfasst werden. Bei einer Punktwolke mit mehreren Bäumen sollte deshalb darauf geachtet werden, dass alle Bäume auf dieser Höhe einen Querschnitt abbilden.

Die *Ausdehnung eines Layers* h ist über die gesamte Modellierung konstant und garantiert damit, dass ein Querschnitt in Wuchsrichtung genügend viele Punkte übereinander erfasst. Der Parameter muss daher genügend groß gewählt werden, sollte aber nicht so groß sein, dass die Höhenausdehnung des Querschnittes größer ist als die Vektorlängen, da sonst im nächsten Schritt im Allgemeinen die Größe der Cluster zu optimistisch eingeschätzt wird.

Die *Vektorlänge* l beeinflusst die Auswahl eines Layers hinsichtlich der Dimensionierung sehr stark und ist – da er aus einer Liste von Werten ausgewählt werden kann – einer der entscheidenden Parameter, um neue Querschnitte zu finden. Da dieser Parameter einen Faktor darstellt, der mit dem entsprechenden Knotenradius verrechnet wird, ist nicht vorherzusehen, welche tatsächlichen Längen die Vektoren annehmen. Zudem müssen die Vektorlängen nicht unbedingt geordnet vorliegen, sodass die Suche nach Querschnitten an die Eigenschaften der Baumart angepasst werden können und sollten.

Der *Winkel* α kann ebenfalls aus einer Liste von Werten ausgewählt werden. Da er auch die räumliche Ausdehnung eines Layers bestimmt, ist darauf zu achten, dass der Winkel groß genug ist, um eine angemessen große Sphäre zu erhalten. Zu große Winkel begünstigen jedoch das Überspringen eines Vektors zu parallelen Ästen. Daher sollten die Winkel im Allgemeinen mit Bedacht gewählt werden. Eine zu starke Einschränkung kann nämlich auch dazu führen, dass ein von einem Knoten abzweigender Ast nicht erkannt wird. Zudem können Werte wesentlich größer als 90° zusätzlich dazu führen, dass ein Vektor entgegen der Wuchsrichtung angepasst wird.

Da jeder Vektorlänge l ein Winkel α zugeordnet wird, beeinflussen beide über die Anzahl der möglichen Paare den Analyseaufwand des Modells essentiell, weil für jedes Paar eine Layerselektion mit anschließender Analyse durchgeführt werden muss.

5.8.2 Clusteranalyse:

Der *Bufferradius* δ entscheidet über die Clustergröße innerhalb eines Layers. Es ist darauf zu achten, dass der Bufferradius oberhalb der geometrischen Auflösung des Laserscanners liegt, da sonst die Punkte nicht als topologisch zusammenhängend erfasst werden können und damit

die Querschnitte zerfallen. Ein zu großer Bufferradius führt hingegen zum Einbeziehen vieler Punkte, die nicht zu einem Querschnitt gehören.

Die *Mindestanzahl der Punkte eines Clusters* n stellt eine Vorauswahl an akzeptablen Querschnitten dar, sodass dieser Parameter ein hartes Abbruchkriterium für die Modellierung darstellt. Er ist aber für die gewünschte Güte des Modells erforderlich, da bei zu wenigen Punkten eine Kreisanpassung statistisch oder rechnerisch nicht sinnvoll ist.

5.8.3 Knotenbestimmung:

Wie bereits in Kapitel 5.4 (*Knotenbestimmung*) beschrieben, werden zur Knotenbestimmung Parameter verwendet, die die Kreisanpassung validieren. Dabei wird der Knotenpunkt ausgeschlossen, falls die Grenzwerte über- bzw. unterschritten werden, sodass es sich um harte Abbruchkriterien handelt. Weichere Grenzen können zwar von Vorteil sein, um auch kleinere Äste modellieren zu können, jedoch führt dies im Allgemeinen zu einer höheren Ungenauigkeit und zur Gefahr von überspringenden Vektoren.

6. Implementierung

Zur Implementierung des Modells war es von Vorteil, spezialisierte Softwarelösungen zu verwenden, um einerseits die Geodaten effektiv im dreidimensionalen Raum analysieren zu können und andererseits eine einfache Programmsteuerung zu erhalten. Da die Geodatenbank *Postgis*, die auf *Postgres* basiert, Möglichkeiten zur Speicherung und Analyse von dreidimensionalen Geodaten liefert, wird diese zur räumlichen Analyse der Punktwolke verwendet. Die Verarbeitung der Daten erfolgt bis auf wenige Ausnahmen ausschließlich über SQL-Abfragen, die verschiedene Funktionen zur räumlichen Analyse oder Abfrage aufrufen, sowie die Daten in weiteren Tabellen speichern. Die Programmsteuerung, wie das Einlesen einer Konfiguration, die Steuerung der verschiedenen Modellierungsschritte, Lösen von Gleichungssystemen, Regressionsrechnungen oder die Ausgabe von Grafiken erfolgt skriptbasiert in der Programmiersprache bzw. Statistiksoftware *R*. Da das Modell ausschließlich über R-Skripte aufgerufen wird, ist die Installation des Modells durch einfaches Kopieren der Dateien, die sich im digitalen Anhang befinden, möglich. Im Folgenden werden das Softwaredesign und der Programmablauf genauer beleuchtet.

6.1 Softwaredesign und Programmablauf

Da Postgis auf der objektrelationalen Datenbank *Postgres* basiert, muss auf dem Server erst diese mit der Postgis-Erweiterung – möglichst ab *Postgres*-Version 9.1 und Postgis ab Version 2.0 – installiert sein. Zudem muss der Zugriff auf eine der Postgis-Datenbanken mit Schreibrechten für Tabellen gewährleistet sein. Der eigentliche Programmaufruf wird über das R-Skript *MainModel.r* getätigt, welches alle anderen Skripte und die Konfiguration importiert. Die Konfigurationsdatei *ModelConfig.conf* befindet sich auf der gleichen Verzeichnisebene wie das Hauptskript. Die graphische Modellausgabe wird, falls gewünscht, in das Verzeichnis *output* gespeichert. Im Verzeichnis R liegen die Skripte *run.r* (steuert die Modellierung), *modelling.r* (beinhaltet die Funktionen der Modellierungsschritte), *helpers.r* (liefert Hilfsfunktionen wie das Laden der Konfiguration), *plotting.r* (zur graphischen 3D-Ausgabe) und *analysis.r* (stellt Funktionen zur Ergebnisauswertung bereit).

Wie bereits beschrieben werden im Skript *MainModel.r* die notwendigen Skripte geladen, sodass die Funktionen aufgerufen werden können. Die eigentliche Modellierungsroutine übernimmt die Datei *run.r*. Diese stellt neben einer Funktion zum automatischen Laden der Punkte auch die Funktion *run()* zur Verfügung, welche alle Schritte der Modellierung koordiniert und die Parameter aus der Konfiguration beim Funktionsaufruf lädt. Dabei wird in einem ersten Schritt die Tabellenstruktur erstellt, danach die Bäume über die Selektion und Clusteranalyse des ersten Layers identifiziert und damit die Modelllogik aufgebaut. Das Modell wird in einem weiteren Schritt iterativ erweitert.

Die Modellierung erfolgt dabei über Aufruf der Funktionen aus der Datei *modelling.r*, welche alle Funktionen zur räumlichen Analyse enthält. Die Auswahl der nächsten Knoten, die Auswahl von Punkten sowie das Speichern der Ergebnisse erfolgt stets über SQL-Abfragen. Damit übernimmt die Datenbank den Hauptteil des Analyseaufwands. Das eigentliche Modellierungsergebnis wird in die Tabelle *branches* geschrieben. Es werden aber noch weitere Tabellen abgelegt, die im nächsten Abschnitt näher beschrieben werden.

Nach der bisherigen Struktur müssen für ein automatisiertes Laden der LIDAR-Messungen in die Tabelle *poi* die Daten in Form einer Textdatei in einem beliebigen Verzeichnis auf dem Postgis-Server liegen. Die Datei muss dabei über die Spalten *x*, *y*, *z* und *ref* verfügen, wenn das Laden der Daten in die Tabelle *poi* automatisiert erfolgen soll. Es ist zu beachten, dass bei einem Programmaufruf alle vorhandenen Tabellen überschrieben werden, außerdem das Verzeichnis *output* geleert wird, sodass die Ergebnisse einer vorigen Analyse bei Bedarf gesichert werden sollten. Im Anhang befindet sich eine Grafik zum schematischen Programmdesign mit allen Skripten, Dateien und Tabellen, die angelegt werden.

6.2 Tabellendesign

Die Modelllogik wurde für die Implementierung vollständig in die Datenbank übertragen, um die Datenverfügbarkeit auch nach Programmende zu garantieren. Dabei werden verschiedene Tabellen angelegt, deren Aufbau und Bedeutung hier genauer beschrieben werden sollen. Im Anhang befindet sich ein Diagramm zu den Abhängigkeiten der Tabellen.

6.2.1 Datengrundlage:

Die Punktwolke der LIDAR-Messungen wird zu Beginn der Modellierung in die Tabelle *poi* geladen. Die Tabelle verfügt über die Spalten *pid* (eindeutige ID des Punktes), *geom* (Geometrieobjekt der Raumkoordinate) und *ref* (Reflexionswert des Punktes), sodass jeder gemessene Punkt eine Zeile bildet. Die weiteren Tabellen, die einen Messpunkt übernehmen, verwenden zwar zur Identifikation die *pid*, jedoch werden trotz Redundanz und der damit verbundenen Gefahr der Inkonsistenz alle Informationen übernommen, da dadurch eine bessere Performance erzielt werden kann. Dies liegt unter anderem daran, dass die Tabelle *poi* naturgemäß die Tabelle mit den meisten Einträgen darstellt und damit eine Abfrage auf dieser – vor allem, wenn es sich um eine räumliche Abfrage handelt – sehr kostspielig ist.

6.2.2 Modellierungsergebnis:

Das eigentliche Ergebnis der Modellierung wird in der Tabelle *branches* abgelegt. Jede Zeile dieser Tabelle steht für einen Knoten des Modells. Daher erhält jede Zeile auch eine eindeutige *branchid* als Referenz, eine Koordinate, die als Geometrieobjekt *geom* abgebildet ist, und den Knotenradius *r*. Von besonderer Bedeutung ist der Eintrag *parentid*, welcher der *branchid* eines anderen Knotens entspricht und damit die Parent-Beziehung ausdrückt. Über Abfragen kann also zu jedem Knoten sein Parent- sowie seine Child-Knoten identifiziert werden. Der Vektor ergibt sich aus der eigenen Geometrie sowie der der Parent-Zeile.

Da die Reihenfolge der Clusteranalyse nicht bekannt ist und die Information über die Cluster beibehalten werden soll, wird für jeden ausreichend großen Cluster ein eigener Eintrag in *branches* erzeugt, der zu Beginn nur aus seiner *branchid* und *parentid* besteht. Erst bei erfolgreicher Knotenbestimmung werden die weiteren Werte aufgefüllt, sodass eine Reihe von Knoten ohne Radius und Koordinate am Ende der Modellierung übrig bleibt.

Da für einige Analysen Zusatzinformationen benötigt werden, werden diese während der Modellierung bereits erzeugt, sodass die Tabelle *branches* neben den unbedingt notwendigen Informationen auch über die Spalten *vector* (berechneter Vektor zwischen Parent- und Child-Knoten), *angle* (Winkel zwischen Vektor und z-Achse) und *r_mse* (bei der Knotenbestim-

mung errechneter Fehler) verfügt. Die Vektorlänge wird aus der Liste der Vektorlängen über den Spalteneintrag *vectorindex* ausgewählt, indem das Element an der Stelle des *vectorindex* aus der Liste verwendet wird. Nach dem Abarbeiten einer Vektorlänge wird der Eintrag in *vectorindex* um eins erhöht.

6.2.3 Hilfstabellen:

Die folgenden genannten Tabellen kann man als Hilfstabellen bezeichnen, da sie während der Modellierung zum Speichern der bedeutenden Zwischenergebnisse verwendet werden und damit die Performance erhöhen. Sie gestalten sich aber als besonders hilfreich bei der Betrachtung des Modellverhaltens und beim Debugging.

Die Tabelle *branchpoints* wird zum Speichern der Branchpoints verwendet und dient zur beschleunigten Layerauswahl. Sie referenziert einen Punkt aus der Punktwolke über die Spalte *pid* mit einem Knoten über die Spalte *branchid*. Zusätzlich werden die Informationen des Punktes *geom* und *ref* sowie Zwischenberechnungen, wie die Distanz zum Knoten *dist*, abgespeichert. Des Weiteren dient diese Tabelle zur Ablage der Zwischenergebnisse zur Fehleranalyse, indem der Abstand des Punktes zum Vektor, der erwartete Abstand und der sich daraus ergebende relative Fehler in die Spalten *varianz*, *mod* und *error* eingetragen werden.

Da die Layer über verschiedene Vektorlängen gebildet und ausschließlich für die Clusteranalyse benötigt werden, wird nur jeweils eine temporäre Tabelle erzeugt, die am Ende der Berechnungen verworfen wird.

Weil die Cluster während der Auswertung und zu einer nachträglichen Analyse benötigt werden, werden die Punkte eines Clusters mit den Spalten *pid*, *geom* und *ref* in der Tabelle *clusterpoints* abgelegt. Da die Cluster stets einem Knoten zugeordnet sind, erfolgt eine Referenzierung über die *branchid*.

Die abschließende Validierung wird in die Tabelle *consideredpoints* eingetragen, die jedem analysierten Punkt über die *branchid* dem Vektor zuordnet, der ihn mit dem geringsten Fehler modelliert hat. Diese Postvalidierung wird über die Tabelle *branchpoints* realisiert, sodass hier auch die Spalten *pid*, *geom*, *ref*, *varianz*, *mod* und *error* übernommen werden.

6.3 Implementierung der Modellkomponenten

Im Folgenden wird auf die Eigenheiten der implementierten Modellkomponenten eingegangen. Die entsprechenden Funktionen hierzu finden sich im Skript *modelling.r* und werden bei der Modellierung über die Funktion *run()* in der Datei *run.r* aufgerufen.

6.3.1 Aufbau der Modelllogik:

Um die Modelllogik aufzubauen, müssen zunächst die im vorigen Abschnitt definierten Tabellen über die Funktion *createTables()* erstellt werden. Danach erfolgt die Auswahl des ersten Layers über die Funktion *firstlayer()*. Dieser Layer wird daraufhin mit der Funktion *clusterFirstlayer()* auf geeignete Querschnitte untersucht, wobei abweichend von der Definition hier die Mindestgröße der Cluster über ihre Fläche bestimmt wird, sodass eine differenziertere Identifikation der Stammansätze möglich ist. Die Knoten werden daraufhin gesondert über die Funktion *insertFirstBranchpoint()* bestimmt, indem eine Kreisanpassung über Lösen des entsprechenden Gleichungssystems (siehe Abschnitt 5.4.3 *Schritt 3: Anpassen eines Kreises in 2D*) in R durchgeführt wird. Die Projektion der Knoten auf die Höhe 0 wird über die Funktion *insertFirstbranches()* erreicht. Alle weiteren Modellierungskomponenten werden in einer Schleife aufgerufen, bis keine weiteren Knoten mehr gefunden werden können.

6.3.2 Auswahl eines Knotens:

Die Auswahl eines neuen Knotens erfolgt über die Funktion *getNextBranch()* und richtet sich gemäß Kapitel 5.7 (*Entscheidungskette*) nach den entsprechenden Auswahlkriterien, sodass derjenige Knoten gewählt wird, der als nächstes eine möglichst kleine Vektorlänge aufweist. Hier wird die Spalte *vektorindex* zur Auswahl eines Elements aus der Liste der Vektorlängen genutzt, um die für die Layerselektion entscheidende Vektorlänge zu errechnen.

6.3.3 Layerselektion:

Der Layerselektion ist die Auswahl der Branchpoints über die Funktion *getBranchPoints()* vorgeschaltet, falls noch kein Eintrag zu dem entsprechenden Knoten in der Tabelle *branchpoints* existiert. Hierbei werden die Punkte über die Postgis-Funktion *ST_3DDWithin()*, aus der Tabelle *poi* ausgewählt sowie die Werte der Spalten berechnet. Die Auswahl eines Layers erfolgt danach aus der Tabelle *branchpoints* mit der Funktion *layer()*, indem der Abstand *dist* mit der gewünschten Vektorlänge verglichen wird und die im Kapitel 5.2 (*Layerselektion*) genannten Berechnungen durchgeführt werden.

6.3.4 Clusteranalyse:

Die Clusteranalyse wird aus Performancegründen in der Funktion *clustering()* abweichend von der eigentlichen Clusterdefinition durchgeführt. Dabei wird die Postgis-Funktion *ST_3dMakeBox()* genutzt, um für jeden Punkt einen umgebenden Würfel mit den Seitenlängen des doppelten Bufferradius zu erzeugen. Diese Würfel werden daraufhin über die Kombi-

nation der Funktionen *ST_dump(ST_Union())* zu einer dreidimensionalen Hülle der Cluster zusammengefasst. Die Punkte können danach über *ST_Intersects()* den Clustern zugeordnet und damit in die Tabelle *clusterpoints* eingetragen werden.

6.3.5 Knotenpunktbestimmung:

Die Knotenbestimmung erfolgt schrittweise über die Punkte eines Clusters aus der Tabelle *clusterpoints* wie im Kapitel 5.4 (*Knotenbestimmung*) beschrieben. Dabei werden sowohl die Kugelanpassung als auch die Kreisanpassung in R durch Lösen der Gleichungssysteme ermöglicht. Für die Rotationen wird die Postgis-Funktion *ST_Affine()* verwendet, die als Argument die Rotationsmatrix erhält. Der neue Knotenpunkt wird nach einer erfolgreichen Validierung in die Tabelle *branches* eingetragen. Zudem besteht die Möglichkeit einer graphischen Ausgabe der Kreisanpassung zur Analyse der Parameterwahl und zur Plausibilitätsprüfung.

6.3.6 Validierung:

Die Validierung eines neuen Knotens erfolgt, wie im Kapitel 5.5 (*Validierung*) beschrieben, sowohl durch eine Plausibilitätsprüfung des neuen Knotens als auch durch Abschätzung der Modellfehler. Hierzu werden die entsprechenden Berechnungen zur Klärung des Fehlers zu einem Segment über SQL-Abfragen der Funktion *validate()* durchgeführt. Die Ergebnisse werden jeweils in die Tabelle *branchpoints* eingetragen. Nach der Modellierung folgt zuletzt der Funktionsaufruf *postvalidate()*, der für alle Punkte der Tabelle *branchpoints* – durch Auswahl des Eintrags mit dem geringsten Modellfehler – die Tabelle *consideredpoints* aufbaut.

6.4 Konfigurationsdatei

Aus Gründen der Übersichtlichkeit werden hier die Einträge der Konfigurationsdatei tabellarisch zusammengefasst. Dabei steht ein *Keyword* für den Namen des Parameters, während die Werte in den nächsten Zeilen folgen und Kommentare durch eine führende Raute gesetzt werden. Eine mögliche Konfiguration findet sich im Anhang dieser Arbeit.

Modellparameter	Keyword	Wertebereich	Einheit	Beschreibung
<i>Postgis-Datenbankverbindung</i>				
-	HOST	-	String	Datenbankserver
-	PORT	-	Integer	Port der Datenbankverbindung
-	USERNAME	-	String	Nutzername
-	PASSWORD	-	String	Passwort

-	DBNAME	-	String	Name der Datenbank
<i>Laden der Daten</i>				
-	LOADDATA	true/false	Boolean	Laden der Daten durchführen?
-	FILE	-	String	LIDAR-Datendatei auf dem Server
-	FILTER	true/false	Boolean	Datenfilterung durchführen?
-	MINREF	\mathbb{R}^+	[]	Minimale Reflexion der Punkte
-	BBOX	\mathbb{R}^6	[m]	Datenbegrenzungen $[x_{min}, y_{min}, z_{min}, x_{max}, y_{max}, z_{max}]$
<i>Modellverhalten</i>				
-	PLOT	true/false	Boolean	Knotenbestimmung plotten?
-	EXTEND	true/false	Boolean	Bestehendes Modell erweitern?
<i>Parameter zur Layerselektion</i>				
s	FIRSTLAYERHEIGHT	\mathbb{R}^+	[m]	Höhe des ersten Layers
h	LAYERTHICKNESS	\mathbb{R}^+	[m]	Dicke der Layer
l	VECTORLENGTH	\mathbb{R}^+	[]	Liste von Faktoren zur Berechnung der Vektorlänge
α	BRANCHANGEL]0,180[[°]	Liste von Winkeln zur Berechnung der Layer
<i>Parameter zur Clusteranalyse</i>				
δ	DELTA	\mathbb{R}^+	[m]	Bufferradius
-	MINAREA	\mathbb{R}^+	[m ²]	Mindestfläche der Cluster des ersten Layers
n	MINPOINTS	$\mathbb{N}_{\geq 3}$	Integer	Mindestanzahl an Punkten eines Querschnitts
<i>Parameter zur Knotenbestimmung</i>				
mse	ERROR	[0,1]	[]	Mittlerer quadratischer Fehler der Kreisanpassung
res_{max}	MAXRES	[0,1]	[]	Maximaler Fehler der Kreisanpassung
R^2	MAXLINEAR	[0,1]	[]	Maximale lineare Anpassung
r_{max}	MAXRAD	\mathbb{R}^+	[m]	Maximal erwarteter Radius eines Querschnittes
w	MAXINCREASE	\mathbb{R}^+	[]	Maximale Steigerung im Astdurchmesser

6.5 Tests

Um die Güte des Modells und die Korrektheit der Implementierung zu prüfen, wurden zahlreiche Modellläufe durchgeführt. Hier sollen einige Aspekte der Modellierung genannt werden.

6.5.1 Datengrundlage:

Als Datengrundlage diente eine von der Universität Trier durchgeführte terrestrische LIDAR-Messung, die auf einer Versuchsfläche in Merzalben (Saarland, Deutschland) nahe der Wetterstation durchgeführt wurde. Die verwendete ungefilterte Teilmenge, aus dem Konkurrenzbereich zwischen Buche und Douglasie verfügte über mehr als zwanzig Bäume mit ca. 8.3 Mil-

tionen Punkten. Da in diesem Datensatz nur eine Aufnahme­richtung vorherrschte, lagen die Querschnitte in Sichelform vor.

Wegen der nur unvollständigen Querschnitte, ungefilterter Punkte und zahlreicher Lücken durch Schatteneffekte schien dieser Datensatz ideal zur Optimierung des Modells, da dadurch einige Verbesserungen des Algorithmus notwendig waren. Zusätzlich stellt die Buche, wegen der zahlreichen Verästelungen eine gute Möglichkeit dar, die Funktionalität dieses Modellan­satzes zu testen. Des Weiteren konnten auch überraschend gute Ergebnisse für Duglasien erzielt werden, obwohl das Modell für deren geometrische Eigenschaften als Nadelbaum nicht optimiert wurde.

Die Erfahrungen aus der Modellierung haben gezeigt, dass als Datengrundlage bereits eine Aufnahme­richtung ausreicht, da hier Effekte durch Windbewegungen geringer ausfallen und sich das Modell als stabil gegenüber Lücken in den Daten – z. B. durch Schatteneffekte – erwiesen hat. Um eine Verbesserung der Modellierung von kleineren Ästen zu erzielen, sollte die geometrische Auflösung zur Krone hin erhöht werden. Für die Modellierung des Stam­mansatzes reicht, aufgrund der gut bestimmten Kreisform der Stammquerschnitte, hingegen eine geringe Auflösung.

6.5.2 Modellanpassung:

Die Modellanpassung erzielte optisch weitgehend gute Ergebnisse, da die Residuen über wei­te Bereiche der Segmente gering ausfielen. Zudem war es bei guter Parametrisierung in eini­gen Fällen möglich, Äste bis zu einem Durchmesser von ca. 2,5 Zentimeter zu modellieren. Hier ist die Modellgüte zwar gering, jedoch kann voraussichtlich das nutzbare Holzvolumen bis zu einem Astdurchmesser von ca. fünf Zentimetern erfasst werden. Zusätzlich ist die An­passung an flache Äste im Vergleich zu Slice-Ansätzen positiv zu nennen.

Das Modellverhalten im Bereich der Astgabelungen stellt bei dem vorliegenden Ansatz noch ein Problem dar, da häufig die Astgabelung um ein Segment verpasst wird und damit die Ab­zweigung erst verspätet erfolgt. Hier könnte eine nachträgliche Modellkorrektur, z. B. durch neue Kombination der Vektorverbindungen zwischen den Knoten, notwendig sein.

6.5.3 Modelllaufzeit:

Die Laufzeit des Modells ist vor allem vom Datenvolumen der Punktwolke, den technischen Voraussetzungen und der Konfiguration (hier besonders durch die Anzahl der Vektorlängen und der Anzahl der betrachteten Bäume) abhängig. Die Laufzeit erhöht sich natürlich, falls

graphische Ausgaben gewünscht sind, da hierfür eine komplette Knotenbestimmung benötigt wird, selbst wenn bereits ein Abbruchkriterium gegeben ist.

Bei den vorgenommenen Tests wurde ein Lenovo U310 mit einem Intel Core i7-3517U Prozessor und 4GB Arbeitsspeicher unter einem Ubuntu 10.04 verwendet. Für die Datenbank wurde eine virtuelle Maschine mit einem Ubuntu 10.04 Server aufgesetzt, der einen Prozessor und 1540MB Arbeitsspeicher zugewiesen waren. Dabei wurde für die meisten Testläufe mit einem einzelnen Baum (ca. 200.000 Punkte) und einer einfachen Konfiguration mit wenigen Vektorlängen eine Modelllaufzeit von unter fünf Minuten erreicht. Hinzu kommt die Zeit zum Laden der Punkte in die Datenbank. Falls weit mehr als fünf Vektorlängen konfiguriert werden, kann die Laufzeit entsprechend bis ca. 10 Minuten ansteigen. Zudem steigt die Laufzeit auch mit der Anzahl der gefundenen Knoten an. Falls mehrere Bäume gleichzeitig analysiert werden (drei Bäume mit ca. 600.000 Punkten), kann die Laufzeit auch auf über 40 Minuten ansteigen.

Allgemein macht es daher Sinn, einen größeren Datensatz mit vielen Bäumen in je einen Datensatz pro Baum zu zerlegen, da so der Analyseaufwand vor allem bei der Auswahl der Branchpoints sinkt.

7. Fazit

Das implementierte Modell zeichnet sich durch die verhältnismäßig geringe Laufzeit und eine einfache und nachvollziehbare Konfiguration aus. Die Sensitivität der Parameter muss jedoch noch systematisch geprüft werden. Weiterhin ist die Anpassung an flache Äste als vorteilhaft zu sehen, da hier ein hohes Potential zur Analyse des Kronenraums zu erwarten ist.

Die Effizienz des Modells könnte, vor allem für kleine Äste, durch verbesserte Verfahren zur Querschnittsanpassung gesteigert werden, sodass die Vorteile dieser Art der Modellierung weiter ausgeschöpft werden können. Die Projektion des Stammansatzes auf die Höhe 0 stellt zudem keine Ideallösung dar, da hier die Modellfehler unnötig hoch und statistisch abhängig auftreten. Der Aufwand der Modellierung mit dieser Implementierung ist wegen der notwendigen Datenbankanbindung je nach Infrastruktur relativ hoch. Zudem kommen zur eigentlichen Modellierung noch die Ladezeiten beim Import der Daten in die Datenbank hinzu.

Die Güte des Modells konnte bisher nur durch Validierung an den eigenen Daten, also den Vektoren, festgestellt werden, sodass die Fehleranalyse vorwiegend optisch erfolgte.

Insgesamt stellt dieser Modellierungsansatz eine Erweiterung eines Slice-Ansatzes dar und sollte daher mit diesem Verhalten der Modellierung verglichen werden. Die Ergebnisse der bisherigen testweisen Modellierung sind vielversprechend, da weitaus mehr Holz als der reine Stammbereich erfasst werden kann. Durch zielgerichtete Optimierung der Komponenten, wie eine Anpassung der Knotenbestimmung an bewährte Verfahren aus Slice-Ansätzen, könnte die Modellgüte weiter erhöht werden. Der Nutzen des Modells wird sich erst durch seine Anwendungen zeigen, jedoch könnte die Erfassung der feineren Äste ein bedeutendes Argument für die Nutzung dieses Ansatzes sein.

8. Ausblick

Es wurde dargestellt, dass das in dieser Arbeit entwickelte Modell grundsätzlich funktional ist. Nun bleibt zu zeigen, ob es praktikabel einsetzbar ist. Hier werden daher einige Beispiele für einen möglichen Einsatz genannt und dabei beschrieben, wie die Güte des Modells ermittelt werden könnte.

8.1 Erweiterungen des Modells

Neben einer Verfeinerung des Modells, z. B. durch eine verbesserte Erfassung der Querschnittsradien, wäre es auch denkbar, andere Daten – wie die Reflexion – in die Modellierung miteinzubeziehen. Hierzu könnten beispielsweise zusätzlich zu den einfachen LIDAR-Messungen auch Aufnahmen mit einem passiven optischen System, wie einer Kamera, in Aufnahmerichtung erfolgen. Da das Modell zunächst mit Rohdaten arbeitet ist ein geeigneter Datenfilter anzuwenden. Dabei könnten je nach Neigung des Geländes, wie bei ASCHOFF (2004), über die Anpassung eines DTM erst die Bodenpunkte entfernt werden. Zusätzlich wäre je nach Baumart eine weitere Einschränkungsmöglichkeit der Layer denkbar, sodass neben einem Suchfenster mit maximalen Winkel auch ein minimaler Winkel angegeben werden kann und so z. B. vorgegeben wird, dass die Winkel eines Baumes in einem bestimmten Bereich liegen müssen.

8.2 Integrität des Modells

Da im Rahmen dieser Arbeit die Validierung des Modells nur an den eigenen Daten vorgenommen wurde und damit keine Unterscheidung zwischen einem modellierten Punkt und einem Störpunkt getroffen werden kann sollten noch Validierungsverfahren verwendet werden. Darunter könnte eine Validierung an harten Daten fallen, wie dem real ermitteltem Holzvolumen oder den Ergebnissen anderer Modelle. Als Alternative wäre auch eine Kreuzvalidierung denkbar, indem bei gleicher Parametrisierung die Modellierung für einen jeweils zufällig gefilterten Datensatz durchgeführt wird. Falls das Modell stabil gegenüber zufälligen Datenkonstellationen ist, sollte das Ergebnis der Modellierung in allen Fällen annähernd gleich sein. Die Sensitivität der Modellparameter müsste zusätzlich noch systematisch geprüft werden. Der Einfluss der Parameterkombinationen, vor allem im Hinblick auf die Vektorlängen und Winkel für die Layerselektion, sind dabei im Besonderen zu klären.

8.3 Geobotanische Parameter

Das in dieser Arbeit entwickelte Modell könnte wie erhofft als Grundlage für die Bestimmung geobotanischer Parameter dienen, da wichtige geometrische Größen direkt oder indirekt aus dem Modell abgeleitet werden können.

Neben einfachen Parametern, wie dem Brusthöhendurchmesser, der sich über das geometrische Modell unkompliziert ermitteln lässt und zur Abschätzung des Blattflächenindex verwendet wird, ist es denkbar, den Blattflächenindex über das gesamte Modell abzuschätzen. Hierzu könnte versucht werden, das Modell so gut wie möglich an einen belaubten Baum anzupassen, sodass über das Modell der Holzanteil entfernt werden könnte. Über die übrigen Punkte der Krone könnte man die Blattfläche schätzen.

Eine weitere Möglichkeit ist es, die Astwinkel des modellierten Baumes zu analysieren, indem der Raumwinkel, wie im Abschnitt 5.4.2 (*Schritt 2: Rotation um den Kugelmittelpunkt*) beschrieben, für jeden Vektor zur z-Achse berechnet wird. Durch den Vergleich von Bäumen unterschiedlicher Art, Standort oder Konkurrenzverhalten könnten Erkenntnisse über diese Faktoren gesammelt werden. Durch Messung und Modellierung einer größeren Anzahl von Bäumen könnten daraus statistisch valide Ergebnisse erzielt werden. Gleiches gilt auch für die Analyse des Verästelungsgrades, indem alle Knoten des Modells betrachtet werden, die über mehrere Child-Knoten verfügen.

Eine angenäherte Holzvolumenbestimmung des Baumes lässt sich über die Summe aller Teilelemente des Modells erzielen. Hierbei berechnet sich das Kugelvolumen eines Knotens k_i

mit dem Parent-Knoten k_j über die Formel $VB_{k_i} = \frac{4}{3} \cdot \pi \cdot r_{k_i}^3$ und das Kegelstumpfvolumen mit dem Parent-Knoten k_j über $VK_{k_i} = \frac{|V_i| \cdot \pi}{3} \cdot (r_{k_j}^2 + r_{k_j} \cdot r_{k_i} + r_{k_i}^2)$. Um die Überlappungsbereiche zwischen den Kegelstümpfen zu verringern, wird das jeweilige Kugelvolumen subtrahiert, sodass sich das angenäherte Gesamtvolumen des Modells über $V = \sum_{i=1}^n VK_{k_i} - \frac{VB_{k_j} + VB_{k_i}}{2} + \sum_{i=1}^n VB_{k_i}$ errechnen lässt.

So lässt sich beispielsweise vereinfacht das nutzbare Holzvolumen abhängig vom gewünschten Durchmesser annähern, indem nur die Knoten betrachtet werden, die den Mindestdurchmesser aufweisen.

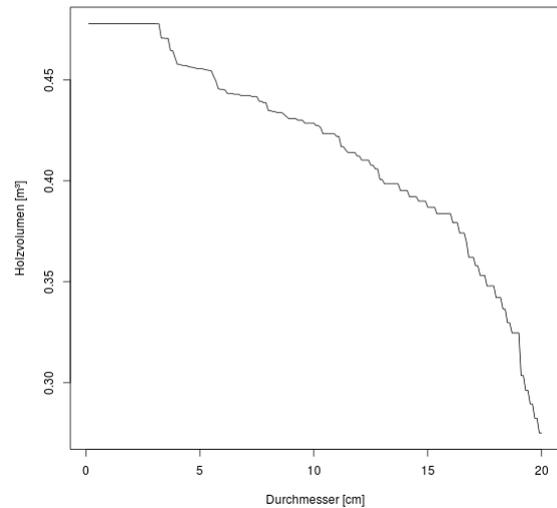


Abbildung 21: Holzvolumenanalyse als Holzvolumen in Abhängigkeit vom gewünschten Astdurchmesser.

Literaturverzeichnis

ASCHOFF, T.; THIES, M.; SPIECKER H. (2004): Describing forest stands using terrestrial laser-scanning. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 25: S. 237- 241.

BIENERT, A.; SCHELLER, S.; KEANE, E.; MULLOOLY, G.; MOHAN, F. (2006): Application of terrestrial laser scanners for the determination of forest inventory parameters. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 26: Part 5.

BIENERT, A.; MAAS, H.-G.; SCHELLER, S. (2006): Analysis of the information content of terrestrial laserscanner point clouds for the automatic determination of forest inventory parameters. *ISPRS WG VIII/11 & EARSeL joint Conference '3D Remote Sensing in Forestry', Vienna, 14-15 February*.

BIENERT, A.; SCHELLER, S.; Keane, E.; Mohan, F.; Nugent, C. (2007): Tree detection and diameter estimations by analysis of forest terrestrial laserscanner point clouds. *Proceedings ISPRS Workshop Laser Scanning and SilviLaser 2007, Helsinki, International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36: Part3/W52.

BRÜNNER, A. (2009): Kreis durch drei Punkte, URL:
<http://www.arndt-bruenner.de/mathe/scripts/kreis3p.htm>. Stand 01.03.2013, 15.58 Uhr.

GORTE, B.; PFEIFER, N. (2004): Structuring laser-scanned trees using 3D mathematical morphology. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 35: Part B5, S. 929-933.

KIRÁLY, G.; BROLLY, G. (2007): Tree height estimation methods for terrestrial laser scanning in a forest reserve. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 26: Part 3 / W52.

PESTER, A. (2013): Das Skalarprodukt zweier Vektoren, URL: 1. März 2013.
http://www.mathe-online.at/materialien/Andreas.Pester/files/Vectors/skalarprodukt_zweier_vektoren.htm. Stand 01.03.2013, 15.59 Uhr.

PFEIFER, N.; WINTERHALDER, D. (2004): Modelling of tree cross sections from terrestrial laser scanning data with free-form curves. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36: Part 8 / W2, S. 76–81.

SIMONSE, M.; ASCHOFF, T.; SPIECKER, H.; THIES, M. (2003): Automatic determination of forest inventory parameters using terrestrial laserscanning. *Proceedings of the ScandLaser Scientific Workshop on Airborne Laser Scanning of Forests, Umeå*, S: 251- 257.

TORNAU, C. (2013): Rotationsmatrix, URL:
<http://www.grundstudium.info/animation/node14.php>. Stand 01.03.2013, 16.29 Uhr.

Anhang

6.1 Softwaredesign und Programmablauf

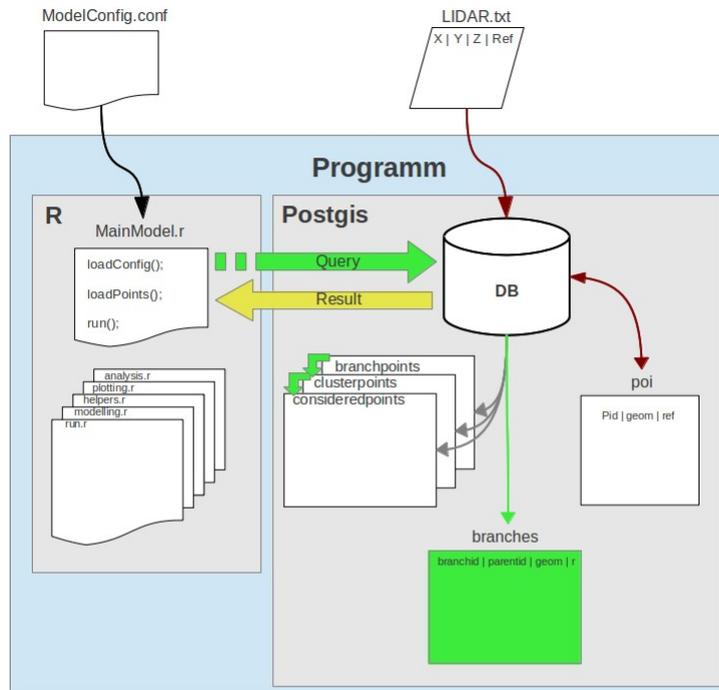


Abbildung 22: Softwaredesign des implementierten Vektormodells. Die Programmsteuerung und vereinzelt Berechnungen übernimmt R, während Postgis die räumlichen Analysen übernimmt. Das Modellierungsergebnis befindet sich in der Tabelle branches.

6.2 Tabellendesign

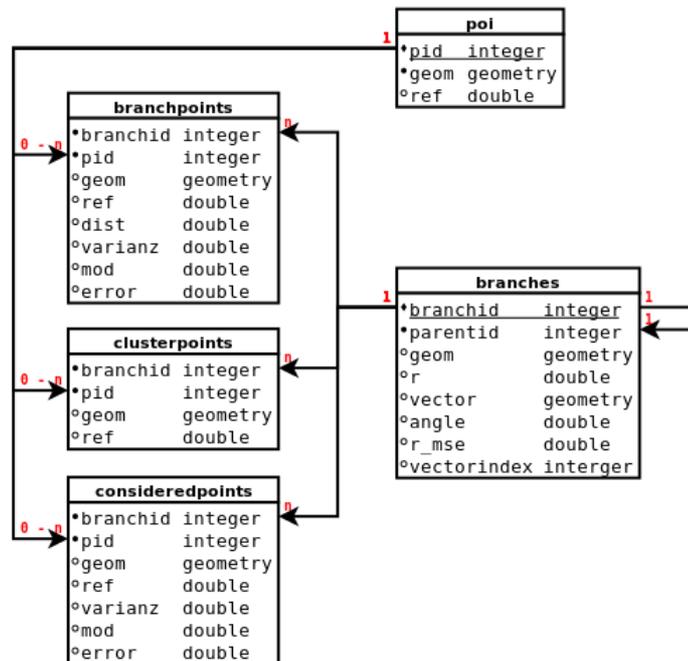


Abbildung 23: Tabellendesign der Modellimplementierung.

6.4 Konfigurationsdatei

Dies ist ein Beispiel für eine mögliche Konfiguration des implementierten Modells:

```
# Konfigurationsdatei:
# Vektorielle Modellierung von Forstbeständen aus terrestrischen Laserscanner-Daten

#####
# Postgis-Datenbankverbindung #
#####

HOST      # (Keyword, do not edit!)
192.168.56.101    # Postgis-Serveradresse

PORT      # (Keyword, do not edit!)
5432      # Postgis-Port

USERNAME  # (Keyword, do not edit!)
postgres  # Nutzernamen (Schreibrechte auf der Datenbank werden benötigt)

PASSWORD  # (Keyword, do not edit!)
Spaceshuttle    # Passwort

DBNAME    # (Keyword, do not edit!)
lidar        # Name der DB

#####
# Laden der Daten #
#####

LOADDATA  # (Keyword, do not edit!)
false     # (true/false) Laden der Daten auf dem Server? Falls true, wird der Pfad zur Datei auf dem Server FILE als Datengrundlage
genutzt.

FILE      # (Keyword, do not edit!)
/home/postgis/lidar/data.xyz    # Ort der xyz-Datei auf dem Server (Format x y z ref)

# Rudimentärer Datenfilter

FILTER    # (Keyword, do not edit!)
true      # (true/false) Legt fest ob die Daten beim laden gefiltert werden.

MINREF    # (Keyword, do not edit!)
700       # [?] Minimale Reflexion, die ein Punkt haben sollte

BBOX      # (Keyword, do not edit!)
8         # [m] Datenbegrenzungen [x_min,y_min,z_min,x_max,y_max,z_max]
5
0.5
14
11
17.5

#####
# Modellverhalten #
#####

PLOT      # (Keyword, do not edit!)
false     # (true/false) Falls true gibt das Modell, Grafiken zum Kreisfitting aus

EXTEND    # (Keyword, do not edit!)
false     # (true/false) Falls true wird das Modell um die zusätzlich eingetragenen Parametern VECTORLENGTH und
MAXBRANCHANGEL erweitert, sodass es nicht neu berechnet werden muss

#####
# Parameter zur Layerselektion #
#####

FIRSTLAYERHEIGHT    # (Keyword, do not edit!)
1.30                # [m] Höhe des ersten Layers über Grund. Bestimmend für die Baumposition.
```

LAYERTHICKNESS #(Keyword, do not edit!)

0.05 # [m] Dicke der Layer.

VECTORLENGTH #(Keyword, do not edit!)

2.5 # [] Faktor zum bestimmen der Länge der Segmente. (Segmentlänge = R * VECTORLENGTH).

3
3.5
4
1
5

BRANCHANGEL #(Keyword, do not edit!)

120 # [°] Maximal zu erwartender Winkel einer Astgabel. Winkel von >0 bis <180°. (Schränkt die Sphärischen Subsets ein).

110
90
70
110
50

Achtung: Die länge der Liste zu VECTORLENGTH muss mit der Länge der Liste BRANCHANGEL übereinstimmen

Parameter zur Clusteranalyse #
#####

DELTA #(Keyword, do not edit!)

0.02 # [m] Maximaler Punktabstand (wird zur identifizierung der Cluster benötigt und entspricht einem Bufferradius).

MINAREA #(Keyword, do not edit!)

0.005 # [m²] Minimale Clusterfläche für ersten Layer

MINPOINTS #(Keyword, do not edit!)

10 # [] Minimale Anzahl an Punkten in Cluster (gilt nicht für erstern Layer)

Parameter zur Knotenbestimmung #
#####

ERROR #(Keyword, do not edit!)

0.05 # (0-1) Signifikanzniveau (maximaler relativer Fehler des Kreisfittings)

MAXRES #(Keyword, do not edit!)

0.3 # (0-1) Maximales Residuum des angepassten Kreises

MAXLINEAR #(Keyword, do not edit!)

0.95 # (0-1) Maximales Bestimmtheitsmaß einer linearen Regression

MAXRAD #(Keyword, do not edit!)

0.4 # [m] Maximaler Astradius

MAXINCREASE #(Keyword, do not edit!)

2 # [] Maximale Steigerung im Astdurchmesser