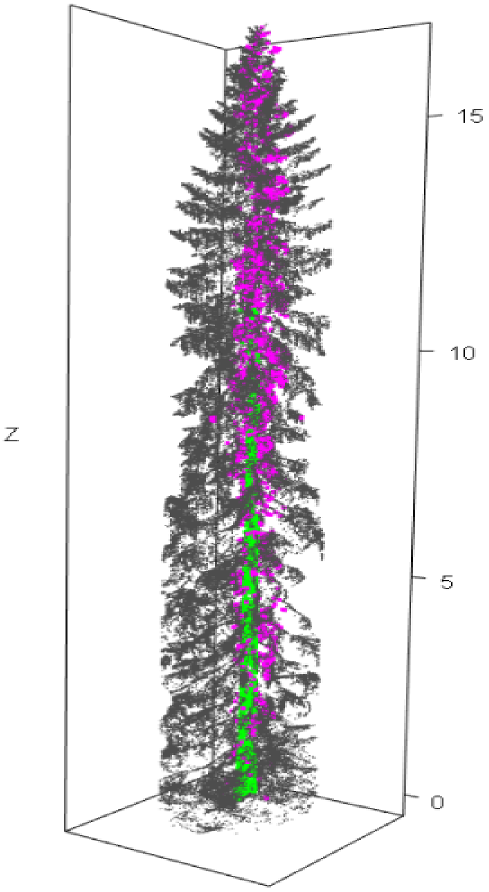
Performance of tree stem isolation algorithms for terrestrial laser scanning point clouds



Tiago de Conto

Supervisors: Kenneth Olofsson Department of Forest Resource Management
Eric Agestam Southern Swedish Forest Research Centre

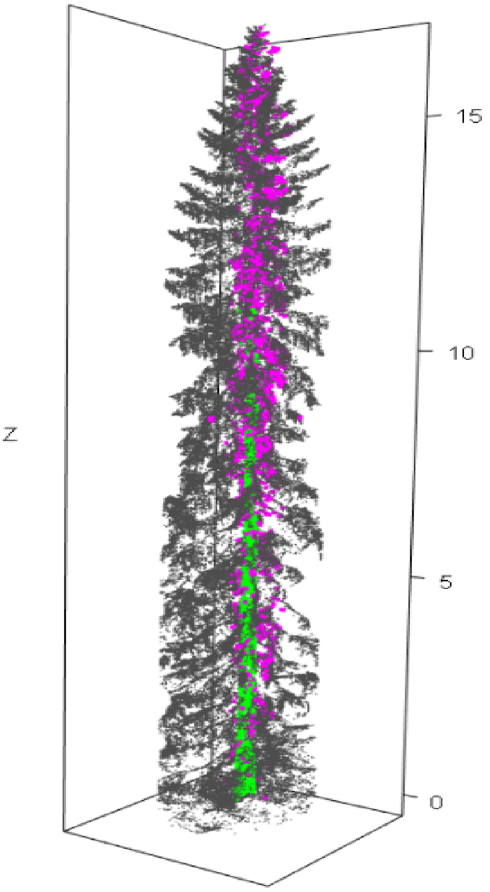
Swedish University of Agricultural Sciences

Master Thesis no. 262

Southern Swedish Forest Research Centre

Alnarp 2016

Performance of tree stem isolation algorithms for terrestrial laser scanning point clouds



Tiago de Conto
Supervisors: Kenneth Olofsson Department of Forest Resource Management
Eric Agestam Southern Swedish Forest Research Centre
Examiner: Per Magnus Ekö Southern Swedish Forest Research Centre

Swedish University of Agricultural Sciences
Master Thesis no. 262
Southern Swedish Forest Research Centre
Alnarp 2016

Master thesis in Forest Management
Erasmus Mundus MSc in Sustainable Forest and Nature Management SUFONAMA
Advanced Level (A2E), SLU course code EX0630, 30 ECTS

ABSTRACT

LiDAR sensors present increasing popularity in the Forestry sector, due to its capability of acquiring high resolution three dimensional data of the forest, useful to a variety of applications. Terrestrial Laser Scanning (TLS) consists of a way to collect large amounts of forest data at the plot level, making further 3D modelling and tree reconstruction possible, thus allowing foresters to extract dendrometric variables with high accuracy from those point clouds. In order to enjoy the full potential of TLS technology for forest inventory, tool sets to extract useful information from TLS point clouds of a variety of tree species are required, starting by stem isolation, which is the core of silviculture and the most targeted outcome of forest management everywhere. The present study aimed to assess the performance of three different methods of stem isolation from TLS point clouds of single trees, both boreal and tropical species. At the same time making the algorithms available as an open source R package. The methods were adapted from three main authors. They rely on finding one main trunk in the point cloud, followed by a circle or cylinder fitting procedure on trunk sections to precisely extract only the stem points. The circle-fit based method had better performance in most cases. Accuracy was higher for all algorithms when tested on boreal trees point clouds, with stem diameter RMSEs ranging from 1.53 to 3.15 cm. For the tropical species the RMSEs ranged from 3.50 to 7.54 cm. Best diameter estimations were obtained for *Pinus sylvestris*, followed by *Picea abies*, *Eucalyptus sp.* and *Pinus taeda*, respectively. All point clouds had reduced density, keeping less than 300 thousand points per tree, and processing time varied from a few seconds up to 20 min/tree, depending on the method applied and point cloud size.

Keywords: LiDAR, cylinder/circle fit, taper, tropical and boreal tree species, robust estimation

Table of contents

1 Introduction

1.1 LiDAR

LiDAR (Light Detection and Ranging) is a type of technology used to measure distances in three-dimensional (3D) space based on laser pulses (Leeuwen & Nieuwenhuis 2010). Every laser pulse return is stored in a 3D coordinate database: a *point cloud*. LiDAR sensors have been increasingly applied in many areas of expertise, due to its high accuracy and speed for acquiring large amounts of data (Schwarz 2010), thus allowing virtual reconstruction of scanned objects from LiDAR point clouds.

The most common implementation of LiDAR sensors is in airborne laser scanning (ALS), where large areas are mapped in 3D by attaching those sensors into airplanes. This technology allows extraction of information at high spatial resolutions, from which one can extract digital terrain models, canopy height models, land cover classification etc. (Dassot et al. 2012). For fine scale precision in measurements of individual objects ALS is not suitable though, mainly due to a large laser footprint – i.e. the further from the laser beam source the targeted object is, the larger is the area covered by a laser pulse when it reaches the target (its *footprint*), reducing the level of detail captured by the sensor. Other reasons for the lack of detail in ALS surveys are the occlusion effect from the upper canopy, lower pulse density and scanning system oscillation. For targeting high accuracy at fine scales, approaches other than ALS are adopted, such as Ground Based Laser Scanning, or Terrestrial Laser Scanning.

1.2 Terrestrial Laser Scanning

TLS works in a similar way to ALS in the sense that it measures distances by means of a LiDAR sensor, but it is positioned on the ground instead. An advantage with TLS systems is that the sensor is located closer to the targeted objects when scanning, reducing laser footprint size, thus catching more detail, with the trade-off that the total area covered by TLS is much smaller than in ALS surveys (Hackenberg et al. 2014). While ALS is suitable for scanning large areas and building terrain models, TLS excels in highly accurate 3D reconstruction of individual objects.

ALS and TLS are rather complementary tools, since their fields of application may differ widely. While ALS has become very popular for surface mapping, in surveying natural resources and geo-sciences in general (Hudak et al. 2009), TLS is more suited for engineering applications, such as 3D modeling and reconstruction of objects (Lichti et al. 2002). In Forestry, for example, both tools are used, either individually or in complement to each other.

1.3 LiDAR and TLS in Forestry

Just as in many other fields, LiDAR sensors have increased in popularity amongst foresters, being applied in planning, management, inventory, biometrics etc. In the future, LiDAR may be applied to automate tasks such as harvesting and weed control. Large ALS surveys over forested areas are common practice in forest enterprises, providing detailed description of forest land at the stand level, gathering input data for models used to estimate several variables of interest for forest management, such as standing volume, basal area, carbon stock, leaf area index etc. (Hudak et al. 2009).

TLS, on the other hand, is capable of gathering data on the individual tree level in detail, which can’t be reached by ALS alone. Furthermore, since ALS operates from above the forest, it is partially blocked by the canopy and data collection aptitude for vertically oriented objects, such as tree stems, becomes severely limited due to the objects being parallel to the laser pulse directions. By operating from below the canopy and scanning radially, TLS collects data that may allow full digital reconstruction of individual trees, making possible direct extraction of variables of interest, such as diameters, tree taper and volume. LiDAR data produced in such ways provide means for the direct measurement of trees instead of just serving as input for predictive modeling. The high resolution of LiDAR sensors in short distances is another factor that contributes for preserving scanned objects' geometrical features in high detail and accuracy. For these reasons, there is a great interest in tools for extraction of tree features based on TLS point clouds. Figure 1 shows the outcome of ALS and TLS surveys side by side.

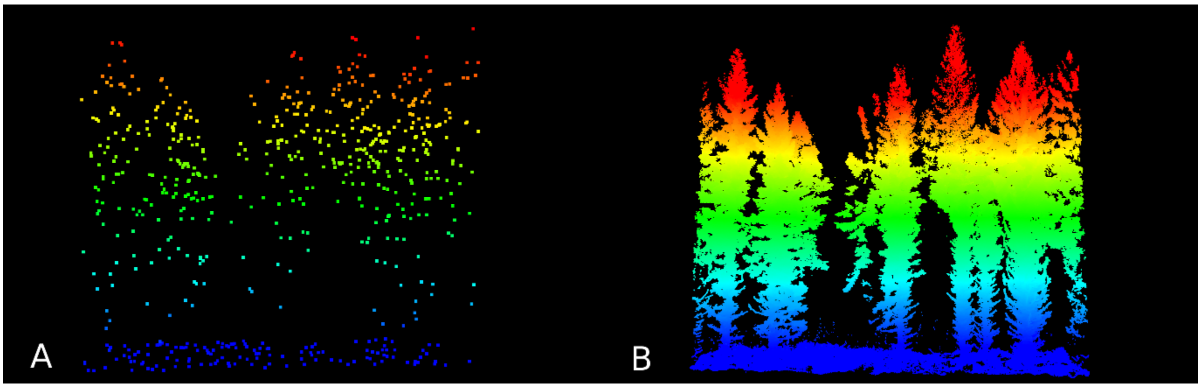


Figure 1. (A) ALS and (B) TLS point clouds from the same scanned area.

Given the level of detail acquired by TLS surveys, one can see that TLS based inventory methods can achieve much finer scales that are not achievable by other state of the art, remote sensing based, forest inventory methods, such as ALS and KNN – K nearest neighbor (Meng et al. 2007). Once time efficient tools are validated and available, TLS will likely become a widely popular tool applied in forest inventory and biometrics. Outside the quantitative realm in forestry, TLS has potential to be used for other purposes. LiDAR sensors can collect information other than coordinates alone, such as laser pulse return intensity and reflected wavelengths. It also can be combined with cameras taking pictures simultaneously to the laser scanning, thus, combining high resolution spatial data with spectral data. This range of capabilities of TLS open doors for methods aiming to automate other procedures, e.g. tree species identification.

1.4 Tree structure and geometry

Several researchers have developed routines for processing TLS point clouds containing tree data. Applications range from isolation of single trees in scanned plots (Liang et al. 2012; Olofsson et al. 2014) to segmentation of tree components (e.g. branches, stem and canopy) (Raumonen et al. 2013; Hackenberg et al. 2014). In order to automate the process of isolating tree components in point clouds, general assumptions need to be made about a tree's geometrical structure. A trained algorithm for stem isolation, for example, requires a level of generalization that allows it to capture the essential information of point clouds from a variety of tree shapes.

Some commonly adopted assumptions that roughly apply to most trees is the vertical orientation of the bole (approximately orthogonal angle with the ground), approximately circular stem cross sections

(Olofsson et al. 2014) and approximately cylindrical stem segments. Those assumptions are the core of many already implemented algorithms, but a key issue lies underneath those assumptions: fitting circles and cylinders works well on *clean* point clouds (with the stem already isolated), but can't be applied directly upon unfiltered ones. Good methods for noise (or outliers) filtering are paramount for making a robust stem isolation algorithm – robust means that a method performs well in the presence of outliers.

Furthermore, tree species present a variety of morphological features, and therefore may present a more or less complex architecture. For example, tropical species usually present a more branched and complex structure when compared to conifer species, which are generally straight. Some properties that should be borne in mind when fitting an algorithm for stem isolation are: branchiness, canopy density, leaning stem segments, more or less circular cross sections and understory density.

1.5 Rationale and objectives

Tool sets focusing on automatic processing of TLS data and LiDAR point clouds in general will assist foresters in the future, reducing need of laborious field work and destructive sampling, as well as providing quick and accurate estimations of tree or forest variables based on 3D data sets, reducing costs for the forest enterprise and allowing foresters to focus more on planning and management activities.

Motivated by the issues mentioned above, the present study aims to evaluate the performance of three main different approaches for stem isolation on TLS point clouds containing single trees, by testing them on point clouds taken from a Swedish Spruce/Pine boreal forest, and tropical plantations of Eucalypt and Pine in Brazil. A side objective is to develop and make available an open source package in R containing functions for dealing with single tree TLS point clouds, focusing on stem isolation.

2 Methods

In the current study, all algorithms were implemented in the R language (R Core Team 2016). The packages rLiDAR (Silva et al. 2015) and rgl (Adler et al. 2016) were used for reading LAS files and 3D visualization, respectively. Three main approaches were implemented based on the work published by Olofsson et al. (2014), Liang et al. (2012) and Raumonen et al. (2013). All algorithms consist of two main phases, respectively: *pre-filtering*, to deal with noise filtering and rough trunk isolation; and *fitting*, to accurately fit circles/cylinders to the trunk, removing all remaining outliers outside the fitted bole. This division allows the user to apply any desired combination of pre-filtering and fitting methods – 9 pairwise combinations.

2.1 Data sources, acquisition and its properties

All TLS data was acquired by means of multiple scan setups, which require co-registration of point clouds of the same area from different positions. In order to do so, targets with known exact positions are fixed inside the plot, and their coordinates are used to convert point clouds from all scanning positions to the same coordinate system (Hilker et al. 2012).

2.1.1 Boreal species

2.1.1.1 Tree data

The trees were selected from test plots in Northern Sweden. Six plots dominated by Scots pine (*Pinus sylvestris* L.) or Norway spruce (*Picea abies* L. Karst.) were measured in the period October-December, 2013. Average age and tree density varied among plots. Three trees were measured manually in every plot. The trees were calipered along the stem with a height distance of approximately 1 meter, starting from the ground level up to 15 m above ground.

2.1.1.2 Scanning data

Scanning was carried out in August, 2013. A Leica ScanStation C10 with high scanning mode was used with a distance of 0.5 cm between measurements in the horizontal and vertical direction, at a distance of 10 m. The scanner had a green laser, 532 nm, beam divergence of 0.1 mrad, scanning pulse rate up to 50 kHz, and horizontal and vertical field of view of 360 and 270 degrees, respectively. A multiple scan setup was adopted with three scan positions in each plot, approximately 25 m apart from each other – forming a triangle surrounding the plot, scanning it from outside.

2.1.2 Tropical species

2.1.2.1 Tree data

Eucalyptus sp. plots were inventoried on June 2016 in an experimental station from the University of São Paulo, in the city of Itatinga, São Paulo, Brazil. The trees were 7 years old, planted in a density of 1667 trees/ha. Some trees were felled and had their stems calipered every 30 cm up to 1.3 m, and every 1 m from 1.3 m up to the total height.

Pinus taeda L. trees were also scanned, in an uneven-aged, seed generated stand belonging to the Santa Catarina State University, in the city of Lages, Santa Catarina, Brazil. No field inventory was carried at the stand and manual measurements on the cloud were performed for reference data. Diameters were extracted at 0.5 m and 1 m above ground and then every two meters up to heights with visible stem portions.

2.1.2.2 Scanning data

A multiple scan setup with five scan positions per plot was adopted for the *Eucalyptus sp*. Scanning was performed on October 2015 and the equipment used was a RIEGL VZ-400 V-Line® 3D Terrestrial Laser Scanning. The phase shift technique was used to measure distances in the near infra-red spectrum (1550 nm), with beam divergence of 0.35 mrad, pulse rate of 122 kHz, field of view of 360 and 100 degrees, horizontally and vertically, respectively.

A multiple scan setup with five scan positions per plot was also applied on the *Pinus taeda* stand. The equipment used was a Faro Focus 3D, with a pulse rate of 244 kHz and field of view of 360 and 305 degrees in the horizontal and vertical directions, respectively.

On both multiple scan setups, one sensor scanned the plot from its center, while the other four scanned it from outside – in a "X" shape.

2.2 Method M1: Hough transformation and RANSAC circle fit

This method was adapted from Olofsson et. al. (2014). It focuses on identifying circular patterns in two dimensional representations of tree segments based on the Hough transformation (Illingworth & Kittler 1987), following by the identification of best circle parameters iteratively by least squares fit in a RANdom SAmple Consensus (RANSAC) approach (Choi et al. 2009).

The tree point cloud is first subdivided vertically, generating t vertical segments of l length in the $z = [0,0,1]$ direction. Each stem sub-section is assumed to have an approximate vertical orientation (parallel to z). The sub-sections are further processed individually, working as input for the subsequent steps.

At first, a *baseline* section is defined at a specified height interval (e.g. 1 to 2 m above ground or from 5% to 10% of the tree's total height). Its approximate circle parameters are then estimated (x and y center coordinates and radius) by applying the Hough transformation (explained below). Up to the height of the baseline section, all points outside a range of the radius plus an inflation factor (e.g. 5 cm) from its estimated center are disregarded. Above the baseline section's height, the search for circles in each segment is constrained to the range found for the previous section in the x and y dimensions. The outcome of this filtering step is a point cloud that contains the stem and some noise, which is further refined by an adapted RANSAC algorithm.

2.2.1 The Hough transformation

The Hough transformation is a robust method for shape detection in two dimensions (2D). It is applied on a raster layer and finds specific patterns according to a targeted primitive 2D shape (in this case, circles). The stepwise Hough transformation based filtering applied on this study is outlined below.

A raster layer is generated from the input point cloud section. Each pixel displays the local relative point density (Eq. 1) inside the raster's x and y limits.

Eq. 1

$$pd = \frac{n_i}{n_{max}}$$

Where:

pd = point density

n_i = number of points counted inside cell *i*

n_{max} = number of points in the pixel with the highest count

Over the generated raster, all pixels with values above a specified threshold (e.g. 30% density) are used as centers for generating circles in a range of radii in between a lower and upper limit, then an iterative process is conducted. Each iteration generates circles of radius r_i centered on all applicable pixels. Another raster layer is then created, and each pixel counts the number of circles that intersect with it – further referred in the text as *votes*. The pixel with the highest votes count (or *score*) on iteration *i* is assigned as center for radius r_i. Out of *m* iterated radii, the best candidate as true radius is the one whose center has the most votes (Figure 2). Its respective center is therefore also considered as the best candidate for the true center.

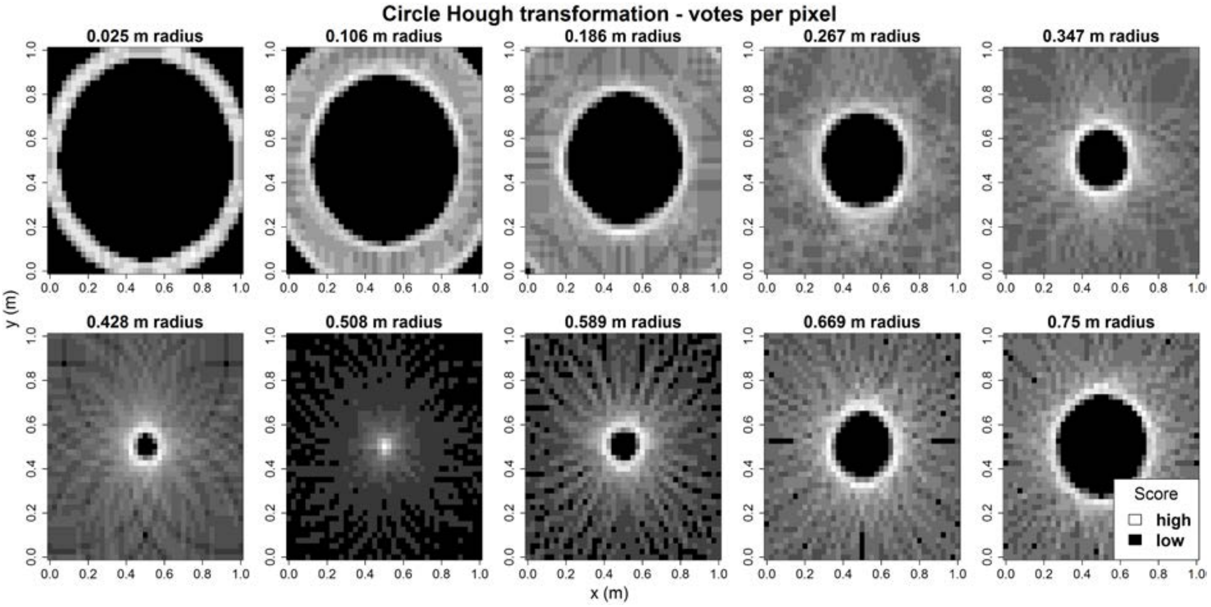


Figure 2. 10 iterations of the Hough transformation searching for circles in raster layers of 2.5 cm pixel size, depicting a vertical cylinder of 0.5 m radius. Each pixel counts the number of votes received for the tested radius stated. The best candidate for true circle is iteration number 7, for a radius of 0.508 m. All other tested radii present many possible conflicting centers. The radius of 0.508 m is the closest one to have all outer circles converging at the same pixel, having the highest score, thus outputting the best circle estimates. All other radii present many conflicting centers with lower scores (a *ring* instead of a *point* at the center).

The pixel size gives the maximum level of detail extracted from the image, therefore it should be small enough to cover the maximum level of detail of the represented object, but large enough to allow reasonable computation time. It may happen that very small pixels are not able to catch information enough to properly represent the data, e.g. most pixels may present similar densities. Figure 3 shows the Hough transformation applied over varying pixel sizes for a section of a sample tree.

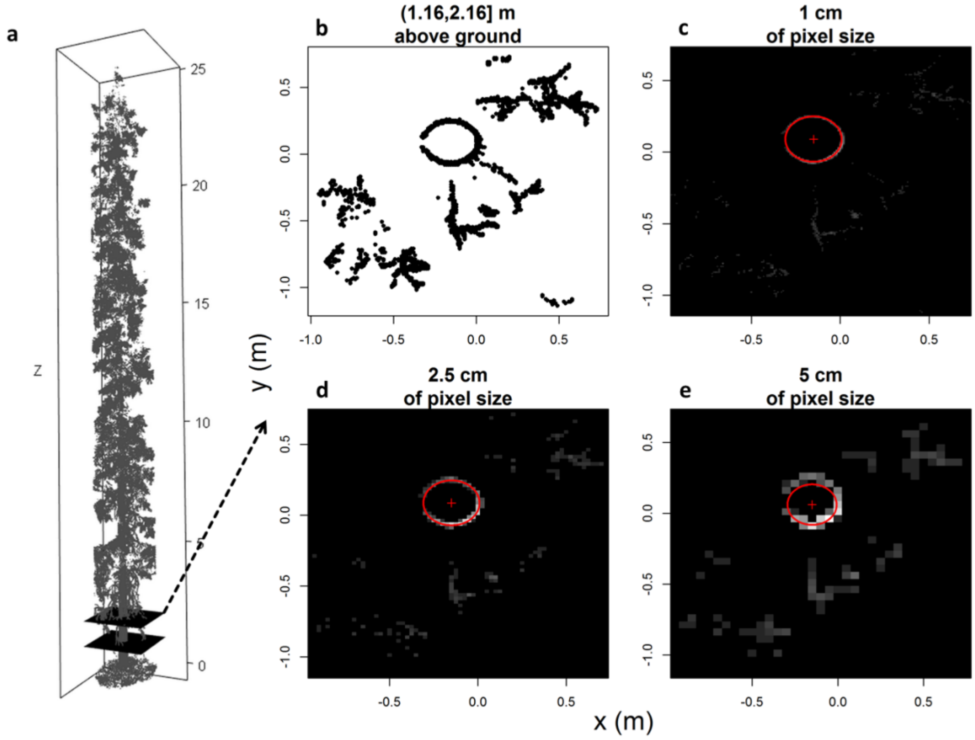


Figure 3. Hough transformation applied over raster layers of different spatial resolutions for the same tree segment –sample tree point cloud (a) with cropped segment detached (b), converted to raster layers of (c) 1 cm, (d) 2.5 cm and (e) 5 cm of spatial resolution, respectively.

2.2.2 RANSAC circle fit

RANSAC stands for Random Sample Consensus and considers the probability that all observations (points) in a random sample belong to a model (Eq. 2) – i.e. the probability that a sample contains no outliers.

Eq. 2

$$N = \frac{log(1-P)}{log(1-p^n)}$$

Where:

N = number of iterations required to find at least one sample without outliers with a probability *P*

P = confidence level

n = number of observations in the sample

p = probability that one point belongs to the model, which is estimated by: $\frac{M}{D}$, with *M* being the number of inliers in the dataset and *D* the total number of points in the dataset.

In this study, the RANSAC algorithm was applied to all output segments from the Hough transformation filter, and a least squares circle adjustment was performed by minimizing the root sum squared error (Eq. 3) in every iteration. The parameter estimations were then refined through general purpose optimization, taking the least squares output as initial parameter estimates. The parameters (*x* and *y* center coordinates and radius) that yielded the lowest RSSE amongst all iterations were chosen as best fit. An example of RANSAC circle fit on a stem segment is given on Figure 4.

Eq. 3

$$RSSE = \sqrt{\sum_{i}^{n}(d_i - r)^2}$$

Where:

RSSE = root sum squared error

d_i = Euclidean distance from a sample point to the estimated circle center in *x* and *y*

r = estimated radius

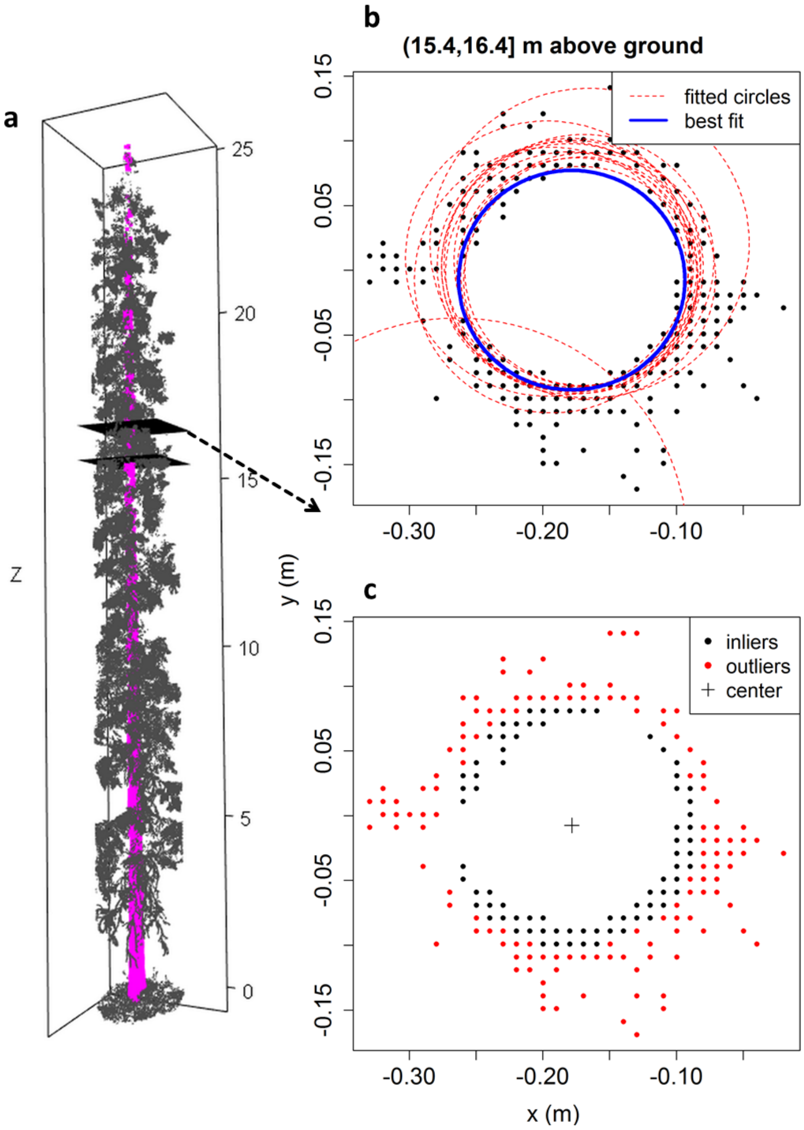


Figure 4. RANSAC circle fit on sample tree segment – (a) sample tree with highlighted remaining points from the pre-filtering phase and cropped segment detached, (b) fitted circles over a trunk segment calculated from different samples of 15 points, (c) inliers and outliers according to the best fitted circle.

2.3 Method M2: Spectral decomposition of 3D surfaces and robust cylinder fit

This method was adapted from Liang et al. (2012). It surveys the point cloud without reducing its dimensionality. Its pre-filtering phase searches for approximately flat and vertically oriented surface patches, assuming that they belong to the stem. Upon the areas that pass such criteria, a robust cylinder model is fitted in an iterated reweighted least squares fashion.

First, the algorithm evaluates the neighborhood of every point belonging to the point cloud. A point is characterized by its k closest neighbors in the 3D space (in terms of Euclidian distance), forming a surface. Spectral decomposition (explained below) is carried out over all neighborhoods and two main features are extracted based on it: surface flatness and its normal vector direction.

2.3.1 Spectral decomposition pre-filtering

The spectral decomposition extracts the eigenvalues and eigenvectors of a dataset, which are as many as the number of variables available (e.g. in the case of *xyz* coordinates there are 3 variables or dimensions). Those values are extracted in such a way that one eigenvector refers to the direction of most variability in the dataset, with the condition that it has to be perpendicular to the previously extracted eigenvector. An eigenvalue is the variance of the data in the direction of its respective eigenvector, therefore the first value represents the direction of highest variability.

In a flat three-dimensional surface represented by a point cloud, the last eigenvalue is expected to be much smaller than the first two, with its eigenvector representing the normal direction to the surface. Under those assumptions, the flatness of a 3D point cloud is defined by Eq. 4. An example of surface neighborhood is shown in Figure 5.

Eq. 4

$$FL = 1 - \frac{\lambda_3}{\lambda_1 + \lambda_2 + \lambda_3}$$

Where:

FL = point cloud flatness

λ = eigenvalue, with $\lambda_1 \geq \lambda_2 \geq \lambda_3$

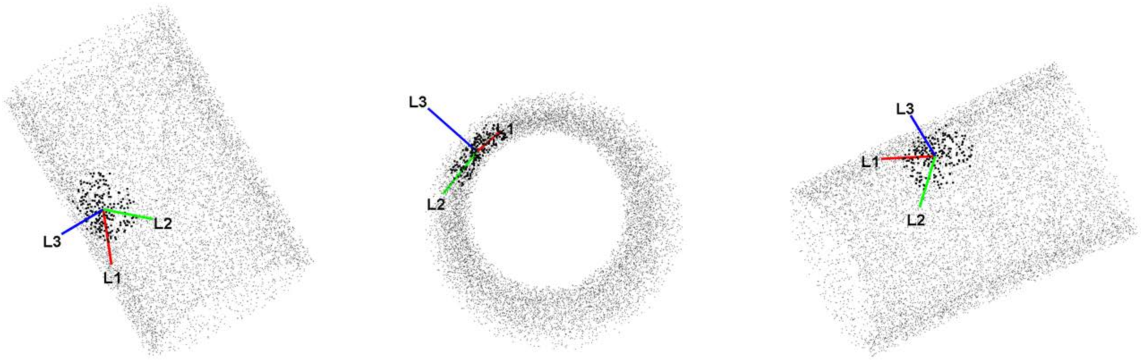


Figure 5. Local neighborhood of a point on the surface of a model cylinder point cloud. *L1*, *L2* and *L3* represent the neighborhood's eigenvectors that relate to λ_1, λ_2 and λ_3, respectively. Note that *L3* is perpendicular to the cylinder's axis, also representing the direction of least variability.

In a point cloud from a scanned tree, sample surfaces belonging to the stem are expected to be approximately flat and having normal vectors approximately perpendicular to *z = [0,0,1]*. In the current method, all points whose neighborhoods comply with those properties are kept for further assessment.

The remaining points are grouped according to their distances in the *xyz* space. Two points belong to the same group if the Euclidean distance between them is smaller than an arbitrary distance *d*. Further on, vectors passing by the center of gravity of each group are projected onto the *xy* plane, and groups whose centers have close projections on *xy* (less than a distance d_{xy} from each other) are merged. This ensures that groups belonging to the same stem are connected even if large gaps occur in the data set. Groups with a large amount of points by the end of this step are considered as part of the trunk – i.e. all groups whose number of points is larger than a proportion *p* of the largest group.

2.3.2 Robust cylinder fit

The output cloud from the previous step is subdivided in vertical sections, like in the first method, and upon each section a cylinder is fitted. A cylinder in three dimensions is defined by Eq. 5. Five parameters (ρ, θ, φ, α, r) are necessary to fit a cylinder, which are shown in equations 5, 6 and 7. The stepwise cylinder parameterization procedure is described by Lukács et al. (1998).

Eq. 5

$$|(p-q)\times a|-r=0$$

Where:

p = a point on the cylinder's surface

q = a point on the cylinder's axis (Eq. 6)

a = unitary vector of cylinder's direction (Eq. 7)

r = cylinder's radius

|…| is the Euclidean norm of the vector and × is the vector cross product

Eq. 6

$$q=(\rho+r)\cdot n$$

Where:

ρ = distance from the origin [0,0,0] to the closest point on the cylinder surface

n = (cos φ sin θ, sin φ sin θ, cos θ) – unitary vector orthogonal to the cylinder's axis, starting from the origin

φ = angle between the projection of *n* on the *xy* plane and *x = [1,0,0]*

θ = angle between *n* and *z = [0,0,1]*

Eq. 7

$$a=n^{\theta}cos\alpha+n^{\varphi}sin\,\alpha/sin\,\theta$$

Where:

α = angle between *a* and n^{θ}

n^{θ} and n^{φ} = partial derivatives of *n*

Figure 6 shows the behavior of vectors *n* and *a* when fitting a cylinder in three dimensions.

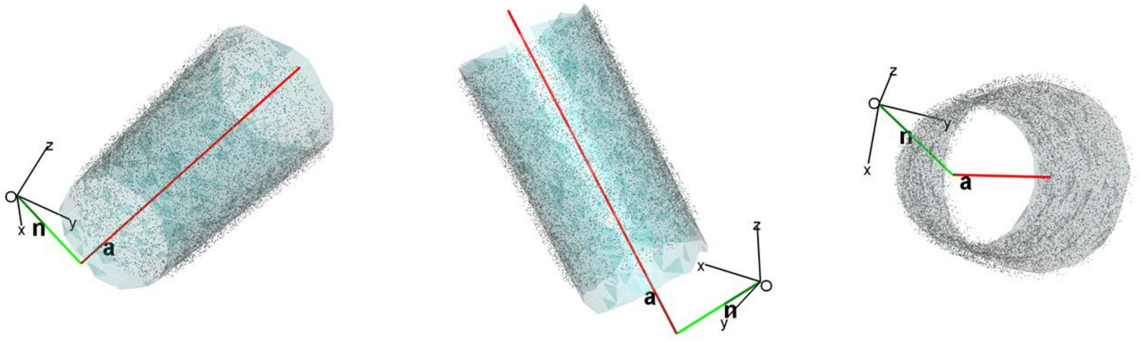


Figure 6. Point cloud cylinder fit using Eq. 5 – *n* is the dark green vector, it starts at the origin and has length 1, being orthogonal to the cylinder's axis *a*, shown in red. The starting point for *a* is *q* (Eq. 6), which is a point at a distance ($\rho + r$) from the origin to the cylinder's axis in *n*'s direction, represented by the light green line.

The output point cloud from 2.2.1 is likely to contain outliers (i.e. points not belonging to the stem). In order to reduce the influence of outliers on the cylinder fitting procedure, a point weighting system is adopted. A robust estimation is performed in an iterated reweighted total least-squares fashion (IRTLS). The best cylinder fit parameters (ρ, θ, φ, α, r) are chosen in a way they minimize the weighted sum of squared residuals to the cylinder's surface (Eq. 8).

Eq. 8

$$\sum_{i=1,j=1}^{n,m} w_{i(j-1)} \cdot RES_{ij}^{2}$$

Where:

$RES = |(p - q) \times a| - r$

m = number of iterations

n = number of points in the rough cylinder point cloud

w = weight

Weights were calculated using M-estimators, more specifically, the Tukey's biweight function (Eq. 9).

Eq. 9

$$w = \begin{cases} \left[1 - \left(\frac{y}{c}\right)^{2}\right]^{2} & , if\ y \leq c \\ 0 & , if\ y > c \end{cases}$$

Where:

w = weight

$$y = \frac{RES}{1.4826 \cdot MAD(RES)}$$

MAD = median absolute deviation

1.4826 = scale factor that ensures MAD is asymptotically unbiased if $RES \sim N(0,\sigma^2)$

c = efficiency constant at the normal distribution (5 corresponds to 95% efficiency)

A general-purpose optimization routine was adopted for minimizing Eq. 8, which doesn't require initial estimation of the right hand side of the model (dependent variable), but good initial estimates for the five cylinder parameters are necessary. For the first iteration of the algorithm the parameters are estimated based on the point cloud's center of gravity, and the value of one is assigned to all weights. In the following iterations weights and parameters are recalculated, with the parameters calculated in one iteration used as initial estimates for the next one. This procedure is repeated until convergence or a specified maximum number of iterations is achieved. Figure 7 shows an example of cylinder fitted to a stem section by IRTLS.

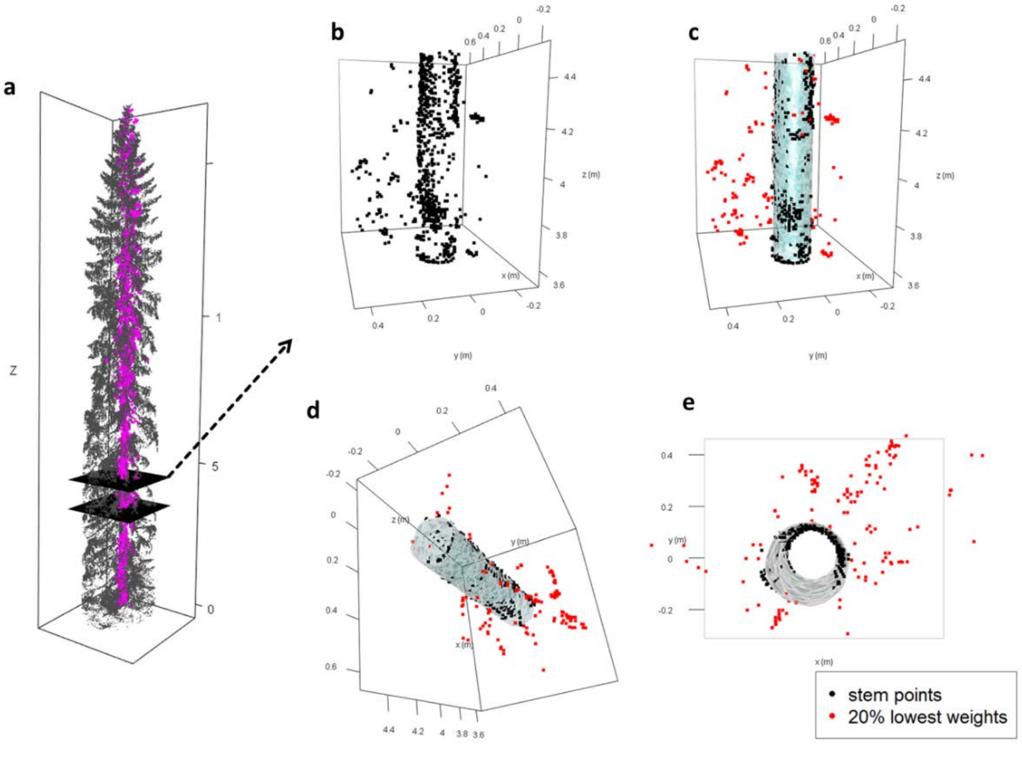


Figure 7. IRTLS cylinder fit for a tree segment – (a) tree cloud with highlighted points from the pre-filtering phase and cropped section detached, (b) pre-filtered stem segment point cloud, (c, d, e) fitted cylinder with highlighted potential outliers – lowest weights according to Tukey's biweight function (Eq. 9).

Once the IRTLS fit is done for a section, all points at an orthogonal distance of d_s or more from the cylinder surface are disregarded. The remaining points for all sections are then merged in a single dataset containing only the tree stem.

2.4 Method M3: Voxel space neighborhoods and RANSAC cylinder fit

This method was developed by Raumonen et al. (2013) and was slightly changed to provide a simpler assessment of the results expected in this study. In their original algorithm the authors segmented trees in more detail, also isolating branches. Here we focus solely on the stem extraction. Furthermore, the authors developed a heuristic method to isolate the desired segments and then directly fitted a cylinder model to the data segments using the least squares method. In the present study a modified RANSAC method for cylinder fitting was applied instead.

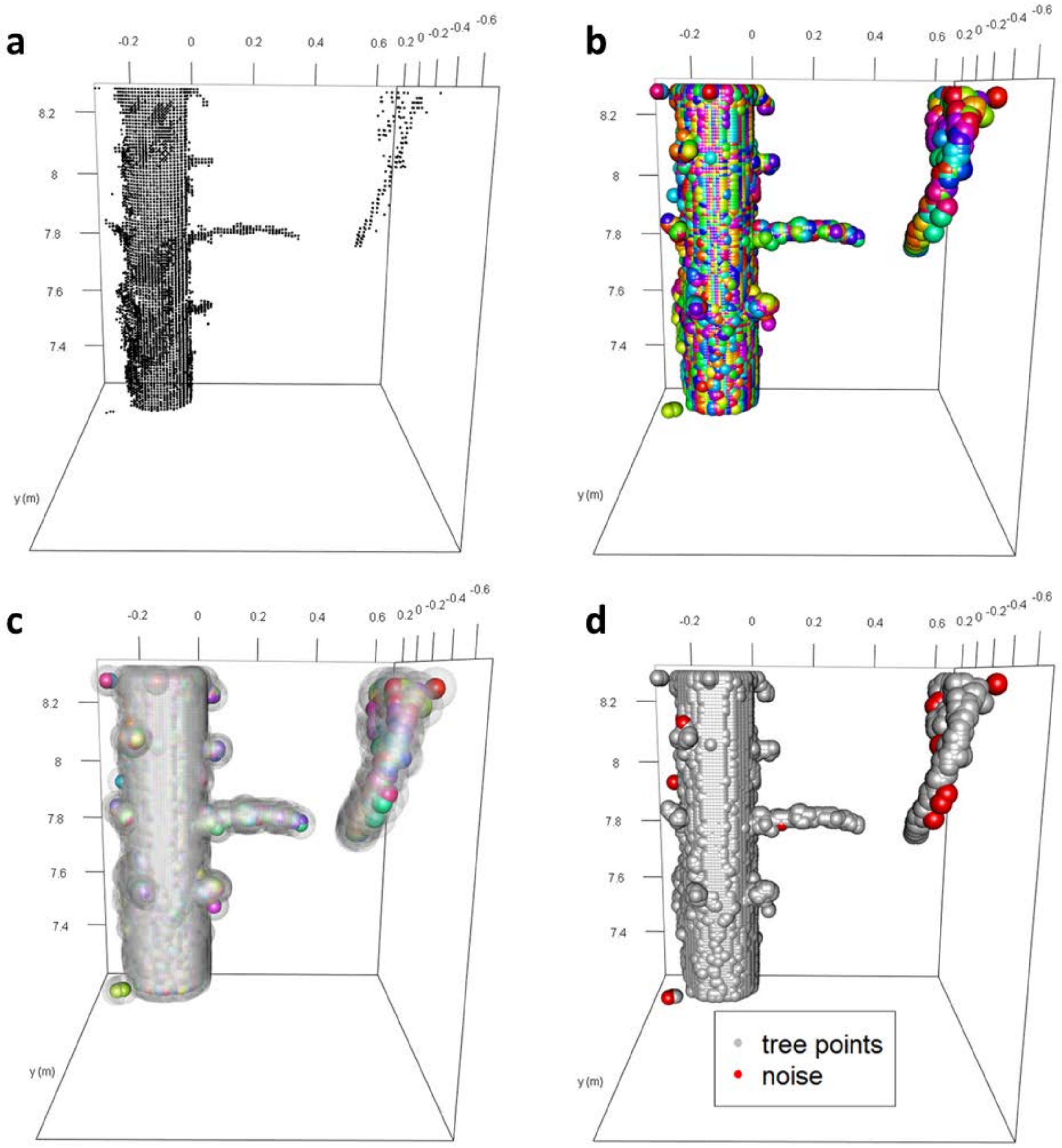


Figure 8. Stepwise rough noise filtering – (a) sample data taken from a tree point cloud, (b) spheres with 2.5 cm radius generated around every point, (c) spheres with 5 cm radius generated around all spheres in *b*, (d) highlighted noise. All points inside spheres in *b* with less than 2 points or inside spheres in *c* containing less than 3 smaller spheres were assigned as noise.

This method has many features in common with M2, described above. At first, on the pre-filtering phase, a rough noise filter is applied to remove isolated points, then patches of the point cloud are merged based on their distribution in an assembled voxel space. Spectral decomposition is carried out on the point neighborhoods and further filtering is done based on flatness, normal vector direction and neighborhood size.

2.4.1 Noise removal and voxel neighborhood aggregation

The first pre-filtering step uses spheres with radius r_1 centered at each point p in the point cloud. All

points inside spheres with less than m points are considered noise and disregarded. A second, and optional, step is to apply the noise filtering a second time, generating spheres with radius r_2 ($r_2 > r_1$) and disregarding all points inside spheres that contain less than n smaller spheres. This procedure is illustrated on Figure 8.

Following the noise removal step, the point cloud is sampled in a regular 3D grid, which serves as base for generating a voxel space. The minimum distance d in between sample points is the voxel side length. The point cloud is then divided into local point neighborhoods. A neighborhood is defined by u^3 neighboring voxels, with u being the neighborhood order, or the number of voxels to merge in one dimension to construct a neighborhood. All points inside voxels belonging to the same neighborhood form a *cover set*. For $u > 1$, neighborhoods, and therefore cover sets, will overlap, since every voxel is used as center of one neighborhood, as illustrated in Figure 9. Figure 10 shows the stepwise procedure on a point cloud sample, after filtered on the previous step.

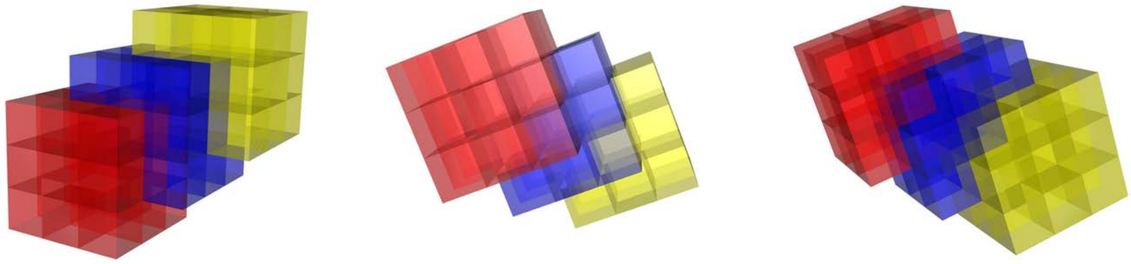


Figure 9. Example of three overlapping voxel neighborhoods for $u = 3$ viewed from 3 different angles. Voxels with the same color belong to the same neighborhood. Notice that all neighborhoods have voxels in common – the central voxel of one neighborhood becomes an outer voxel of the next neighborhood. Overlapping avoids point misclassification on the next step, that classifies stem points based on spectral decomposition of cover sets.

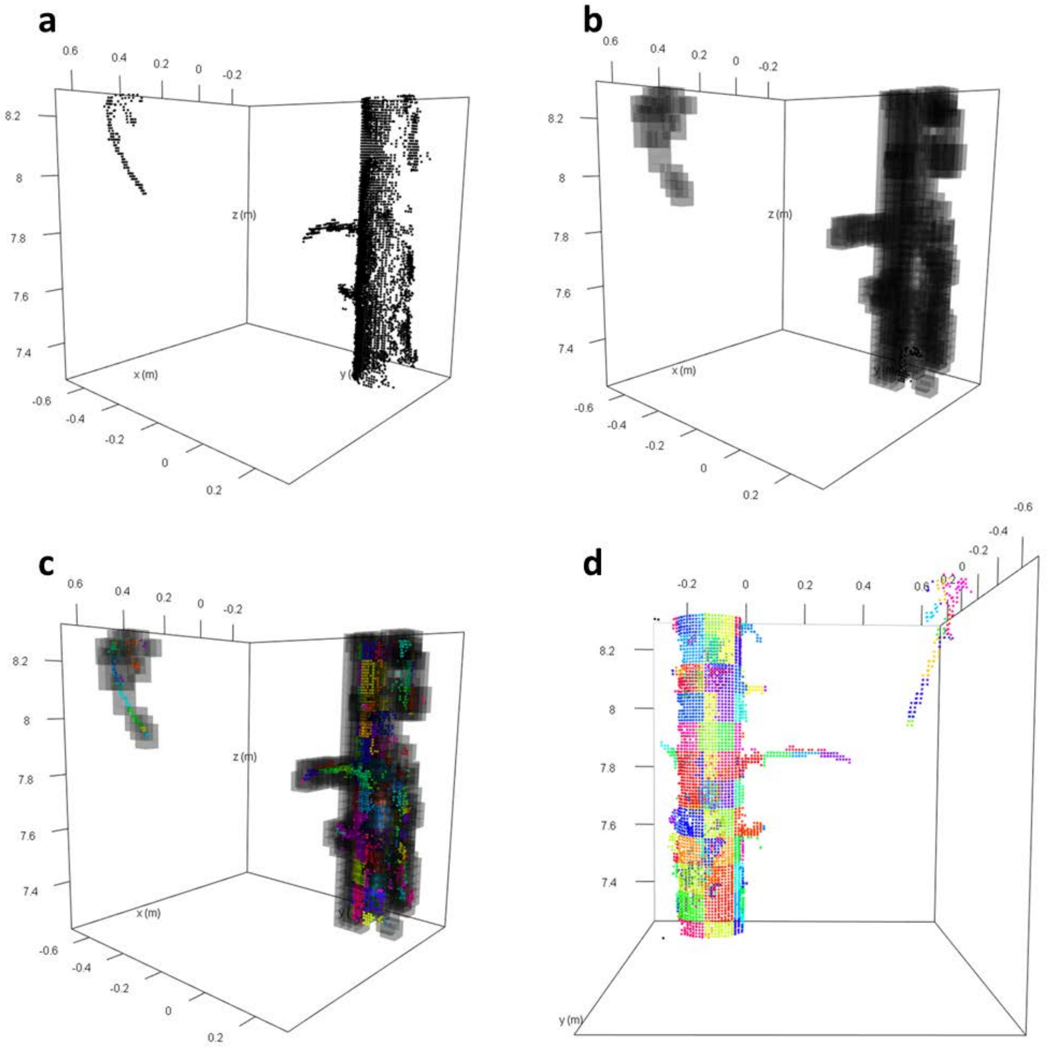


Figure 10. Voxel neighborhoods and covers sets construction – (a) sample cloud after rough noise filtering (Figure 5), (b) construction of voxel space with 3 cm of voxel side length, (c) cover sets originated from neighborhoods of order 3, (d) clear view of cover sets.

Spectral decomposition is performed on all cover sets, their flatness (Eq. 4) and normal vector are extracted, analogously to M2. All points belonging only to cover sets with flatness and angle between its normal vector and *z* below arbitrary thresholds are disregarded. The voxel space is then reorganized into larger neighborhoods and new cover sets are obtained. The largest cover set (with highest number of points) is used to estimate a global axis for the stem. In this step the stem is assumed to be parallel to *z*, therefore only *x* and *y* coordinates need to be estimated for positioning the axis. Cover sets further than d_a meters from the estimated axis (Euclidian distance) and containing too little points (i.e. lower than a specified percentage of the largest cover set) are also disregarded. The remaining point cloud is used as input for the next phase.

In the original method proposed by the authors (Raumonen et al. 2013), cover sets are defined according to spherical neighborhoods in random coordinates instead of voxels. Here voxels are used in order to reduce processing time.

2.4.2 RANSAC cylinder fit

This method intertwines the fitting phases on M1 and M2. Here the RANSAC approach is used to fit cylinder parameters (Eq. 5, 6 and 7) in a total least squares fashion. In order to do so, the output trunk point cloud from previous stem is segmented in intervals of length *l* in *z's* direction. A general-purpose optimization method is then iteratively applied on at least *N* random point samples (Eq. 2) of every vertical segment, aiming to minimize the sum of squared residuals in Eq. 10. Figure 11 shows an example of the RANSAC cylinder fit on a sample stem segment.

Eq. 10

$$\sum_{i=1}^{n} RES_i^2$$

Where:

$RES = |(p - q) \times a| - r$

n = size of random sample

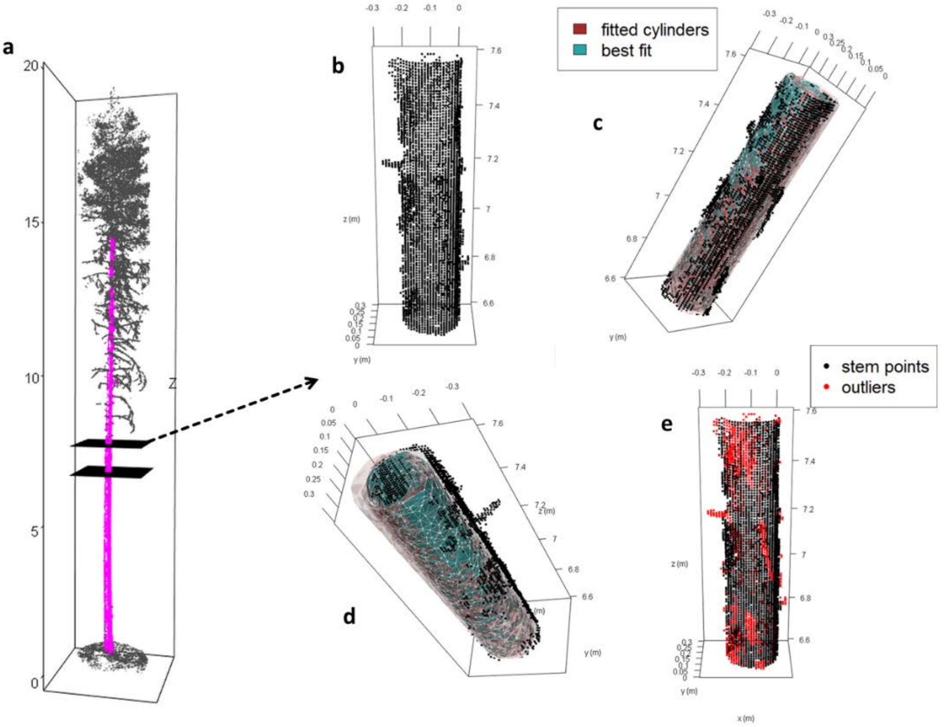


Figure 11. RANSAC cylinder fit on a TLS tree cloud segment – (a) tree cloud with highlighted points from the pre-filtering phase and cropped section detached, (b) pre-filtered segment point cloud, (c, d) fitted cylinders over samples of 20 points with the adapted RANSAC cylinder fit algorithm, (e) outliers – points further than 1 cm from the cylinder surface.

A baseline segment is defined, in analogy to what is described for M1, upon which minimization of Eq. 10 is performed, taking as initial parameters estimates based on its center of gravity. The parameters that yield the best fit for a segment are used as initial estimates for the next one, fitting cylinders of length l from bottom to top of the tree.

2.5 Functions parameterization

2.5.1 M1: Hough transformation and RANSAC circle fit

For the pre-filtering phase, the trees were segmented into vertical sections of $l = 0.5$ m. Each segment was converted to a raster layer with 2.5 cm of pixel size, over which the Hough transformation was applied, generating circles centered in all raster cells with point density $pd > 0.3$.

In the fitting phase, the chosen parameters for the RANSAC number of iterations were $n = 15$, $p = 0.8$ (estimated, same value was used by Olofsson et al. 2014), $P = 0.99$, and thus $N = 129$. $N \times 5 = 645$ iterations were carried out in every circle fitting procedure. Tree segments with less than 15 points were ignored.

2.5.2 M2: Spectral decomposition of 3D surfaces and robust cylinder fit

On the pre-filtering phase all points belonging to surfaces ($k = 30$ closest neighbors) with $FL < 0.9$ or whose normal vector presented angle with z outside the 80° - 100° interval were ignored. Points with less than 10 cm ($d < 0.1$ m) apart in the xyz space were grouped, and groups less than 20 cm apart ($d_{xy} < 0.2$ m) in the xy plane were merged. All groups with $p > 0.25$ the amount of points of the largest group were retained in the output of the pre-filtering phase. For some trees, though, the relative amount of points had to be reduced, otherwise significant portions of the stem would be lost, thus the largest group proportion criterion was reduced to $p > 0.1$.

On the fitting phase, cylinders were fitted from bottom to top in segments of 0.5 m in z's direction. The IRTLS algorithm ran until the difference in the weighted sum of squares (Eq. 8) between consecutive iterations was lower than 0.001 m or until 10 iterations were reached. If a cylinder had estimated radius of more than 0.5 m, its fitted parameters were ignored and the ones from the segment below were adopted. All points with Euclidean distance $d_s > 0.01$ m from the cylinder surface were considered outliers.

2.5.3 M3: Voxel space neighborhoods and RANSAC cylinder fit

On the pre-filtering phase, the following values were adopted for the rough noise removal: $r_1 = 0.05$ m, $m = 2$ points, $r_2 = 0.1$ m, $n = 3$. Following, a voxel space with $d = 0.03$ m of spatial resolution was generated for the point clouds and neighborhoods of order $u = 3$ were merged. Spectral decomposition was performed on all cover sets originated on the previous step. All points belonging exclusively to cover sets with $FL < 0.9$ or whose normal vectors' angle with z lied outside the 80° - 100° interval were disregarded. Over the remaining points, a new voxel space with spatial resolution $d = 0.05$ m was built and neighborhoods of order $u = 4$ were merged. Cover sets smaller than an arbitrary threshold p_n, or with geometric center further than $d_a = 0.5$ m from an estimated global axis for the tree were further disregarded. An automated method for estimation of p_n based on cover set size distribution was applied – once defined for a tree, all cover sets are sorted according to their amount of points, the largest and smallest covers sets are then connected by a straight line, and the cover set which is further away from the line gives the size threshold for keeping or eliminating cover sets from the point cloud, i.e. threshold equals point of maximum curvature (p_n) in Figure 12. For leaning trees the distance from the global axis had to be increased, thus d_a = 1.5 m was used.

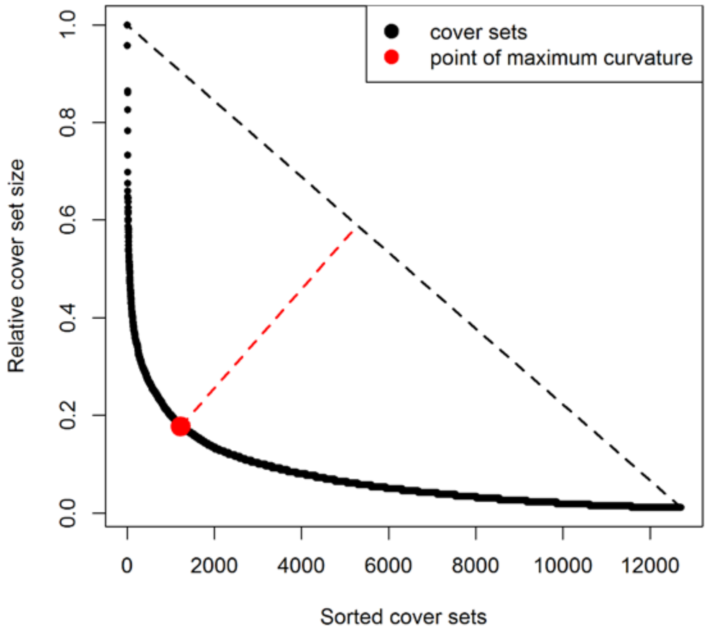


Figure 12. Rough estimation of maximum curvature (p_n) for cover sets sorted by size. Relative cover set size is the ratio between the number of points in a cover set and the number of points in the largest cover set.

The fitting phase was carried out on tree segments of l = 0.5 m. The following parameters were adopted for the RANSAC: n = 20, p = 0.9, P = 0.99, thus N = 36 and N×2 = 72 iterations were carried out for every stem segment.

2.6 Performance evaluation

The performance of the methods was assessed by analysis of residuals, comparing TLS estimated diameters with field measurements of diameters along the stems (taper). Root mean square error and bias were calculated for each method, according to Eq. 11 and 12, as well as Pearson's correlation between estimated and measured diameters. Results were first split in between boreal and tropical datasets, and then amongst tree species.

Eq. 11

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\widehat{d_\imath} - d_i)^2}$$

Eq. 12

$$bias = \frac{1}{n}\sum_{i=1}^{n}(\widehat{d_\imath} - d_i)$$

Where:

RMSE = root mean square error

$\hat{d}$ = estimated diameter at the i^{th} observation

d = observed diameter at the i^{th} observation

n = number of observations (fitted cylinders/circles)

Benchmarking tests were also performed with reduced point cloud densities to assess quality of estimation and processing time in R for all three methods.

3 Results and discussion

All described methods on the section above were applied on the same TLS point clouds containing single trees. In order to reduce computation time, all TLS clouds had their original densities reduced to a maximum of 10^4 points per meter in z's direction (above ground) – resulting in 10^5 to 3×10^5 points per tree. Ten trees were randomly selected from the boreal datasets for the benchmarking tests, having their point cloud densities further reduced to 50% and 20% of the previously stated one – to a maximum of 5×10^3 and 2×10^3 points per meter above ground, respectively. TLS point clouds are generally very dense and all tree geometrical features can be represented by much smaller samples without significant losses in accuracy (Raumonen et al. 2013).

All sample trees were measured on field and several diameters along the stem were obtained. For a given diameter measured at a specific height, a cylinder of 0.5 m of length was cropped from the stem point cloud, centered at the same height, and a circle or cylinder was fitted using only the points on the cropped segment.

Unfeasibly large diameters obtained from this procedure were re-estimated by local linear functions. The linear functions were built based solely on the laser data of individual trees. Radii were estimated from bottom to top, every 0.5 m along z. A reference taper function was used to detect large radii along the stems (Eq. 13) – a reference cone with radius of 30 cm at the ground level and same height as the tree's total height. All radii larger than the value at correspondent height on the reference cone were disregarded, and a linear model estimating radius from height above ground was fitted by ordinary least squares using the remaining estimates. These models (one per tree) were further used to re-estimate previously disregarded diameters when comparing TLS and field measurements.

Eq. 13

$$r_h = r_{base}\sqrt{(\frac{H-h}{H})^b}$$

Where:

r_h = maximum radius tolerated at a given height h

r_{base} = maximum radius tolerated at ground level

H = total tree height

b = form factor: 0 for cylinder, 1 for paraboloid, 2 for cone and 3 for neiloid

3.1 Quality of estimation

3.1.1 Boreal species

The estimated diameters from TLS are plotted against field measurements on Figure 13. All methods had overall good estimates of diameters with RMSE values lower than 3 cm in all methods. The Hough transformation followed by RANSAC circle fit had the best performance in terms of RMSE, bias and linear correlation to field measurements (Figure 13 - M1). Such an outcome was possible since Pine and Spruce trees are generally straight, thus stem cross sections parallel to the ground are well approximated by circular shapes on the *xy* plane. The method's assumption of vertical straight stems in both pre-filtering and fitting phases corresponds well to the natural geometry of those species. Furthermore, on the fitting phase, good initial estimates for the cylinder and circle parameters are required for enhanced performance. For cylinders those values were roughly estimated using the gravity center of trunk segments, while for two-dimensional circles, good initial parameter estimates are easily obtained through least squares fit.

All methods performed slightly better on Pine clouds, rather than Spruce. As seen in Figure 13, most of the values far from the red line (representing a perfect fit) belong to spruce trees, especially for the second method. Spruce trees present denser canopy, making it harder for a laser pulse to reach the stem, thus causing occluded patches to happen more often in Spruce point clouds (Figure 14). The pre-filtering phase of the second method relies on surface reconstruction and requires a high frequency of returns belonging to the stem in order to be effective, making estimates of occluded segments less accurate.

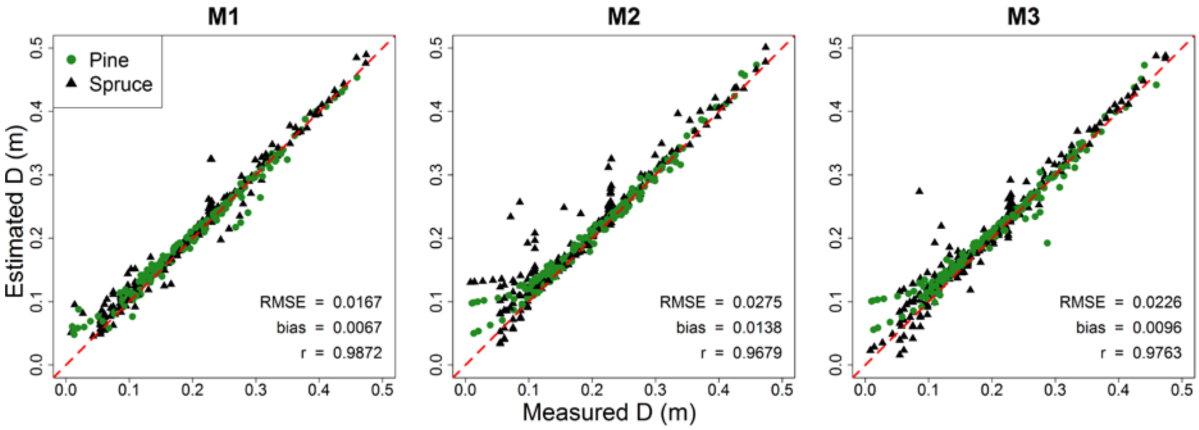


Figure 13. Estimated vs field measured diameters for boreal species (D = diameter; r = Pearson's linear correlation).

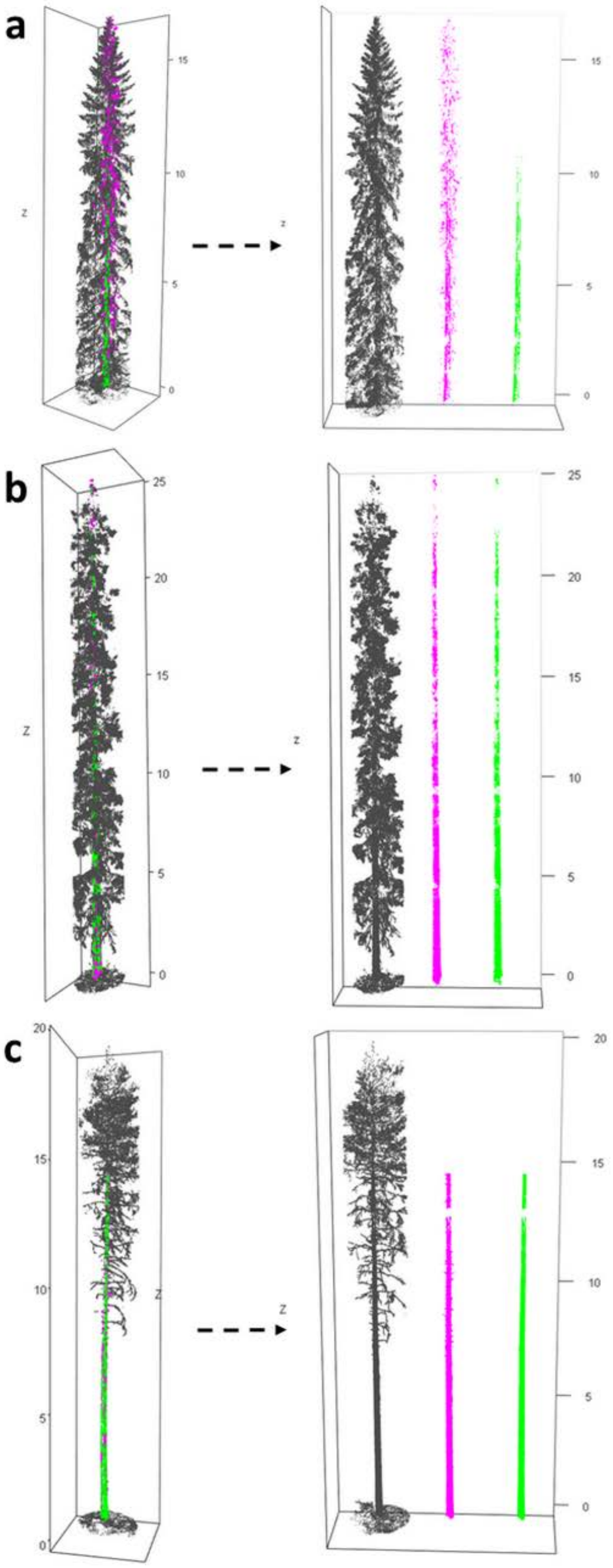


Figure 14. TLS point clouds from (a) a young Spruce, (b) an older Spruce and (c) a Pine tree. Pre-filtering and fitting phase outputs are shown in pink and green, respectively. Methods M2, M1 and M3 were applied, respectively.

The algorithms presented poorer performance on segments of lower diameters, with estimations tending to be more upwardly biased for stem segments smaller than 10 cm in diameter (Figure 15). As the top height of a tree is approached, its stem diameter decreases, and its occlusion by the canopy as well as distance from the LiDAR sensor increase, therefore the poorer performance is not entirely related to diameter size, but is more likely to happen due to the latter two stated reasons. Figure 16 shows residuals distributed in accordance to the portion of the tree from which the diameters were obtained.

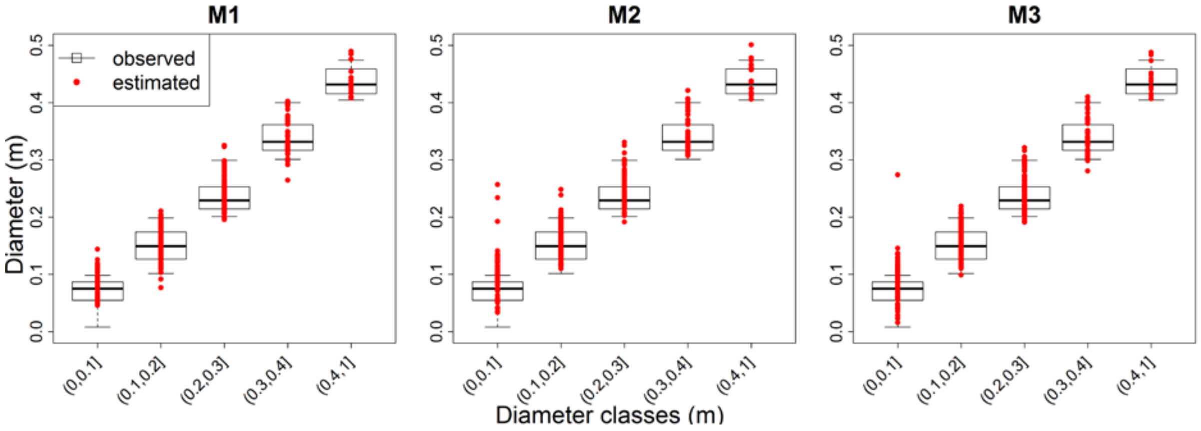


Figure 15. Estimated vs measured diameters distributed in different classes for boreal species.

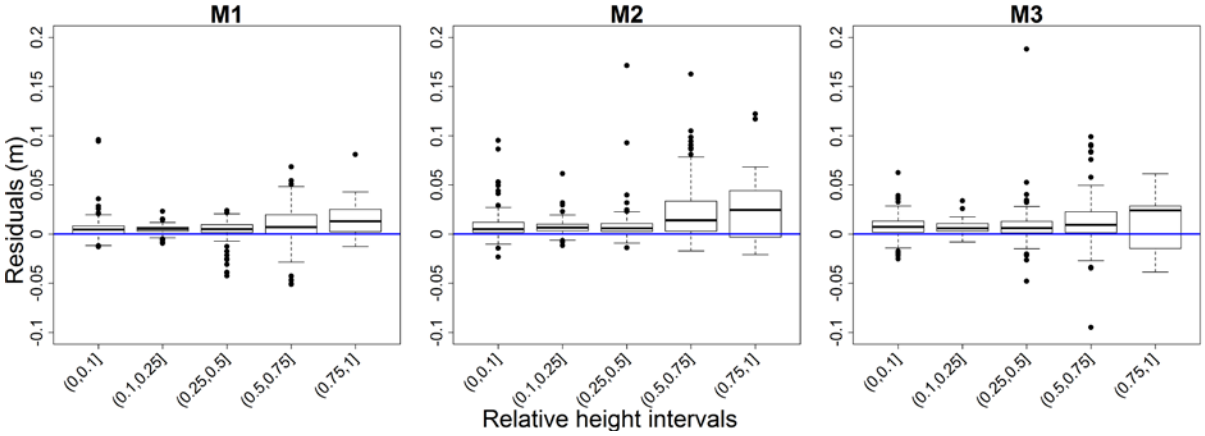


Figure 16. Residuals distribution according to relative tree height, which is the ratio between the height on which a diameter was measured and the tree's top height.

3.1.2 Tropical species

Scanned plots of Eucalypt trees were heavily affected by wind during the TLS survey, which associated to the young, slender trees, yielded some point clouds of poor quality. In order to overcome this issue, the trees were manually "cleaned" prior to any tests, mainly to remove "ghost stems" that largely inflate diameter estimations by the algorithms, due to difficulties in distinguishing overlapping cylinders. Further research should be carried out to automate the cleaning step for defective point clouds.

The fitting quality assessment for the tropical trees point clouds is seen on Figure 17. For all methods the accuracy was lower when compared to the fitted boreal species, but RMSEs remained smaller than 5 cm for all methods. M2 was slightly more precise than the other methods, but presented the highest bias (overestimation). M1 was the least precise method, but on the other hand it presented the least bias (around half of the other methods). The tropical trees often presented stems not as straight as the boreal ones, leaning specially at higher portions (Figure 18). This reflects on a better performance of the cylinder-based fitting methods in comparison to the horizontal cross section circle-fit (M1).

The observed overestimation, for the Eucalypt, is likely to be a residual effect of the wind, enlarging some stem cross-sections in the point cloud due to sideways movement during laser scanning. As for the Pine trees, notice that the measured diameters were manually extracted from the point clouds and not from field inventory, thus unknown errors may have been introduced.

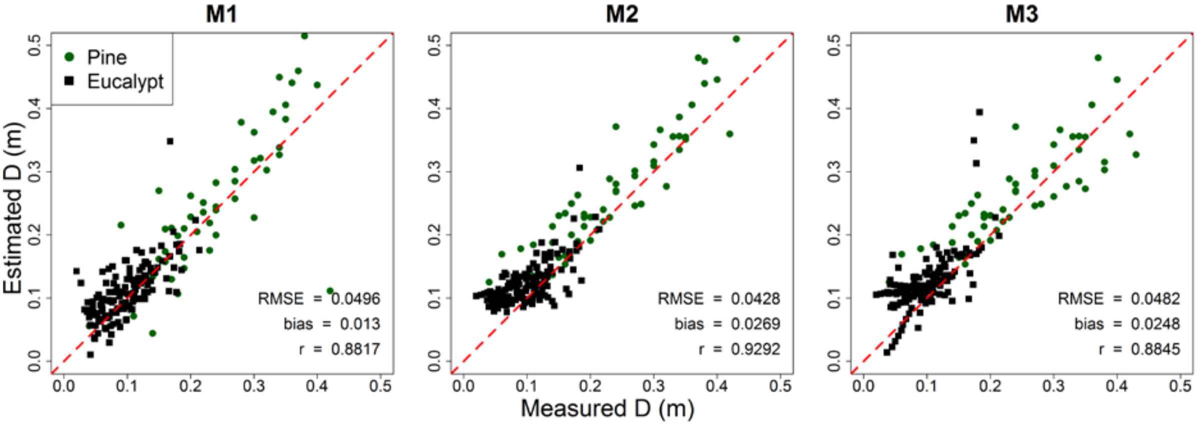


Figure 17. Estimated vs field measured diameters for tropical species (D = diameter; r = Pearson's linear correlation).

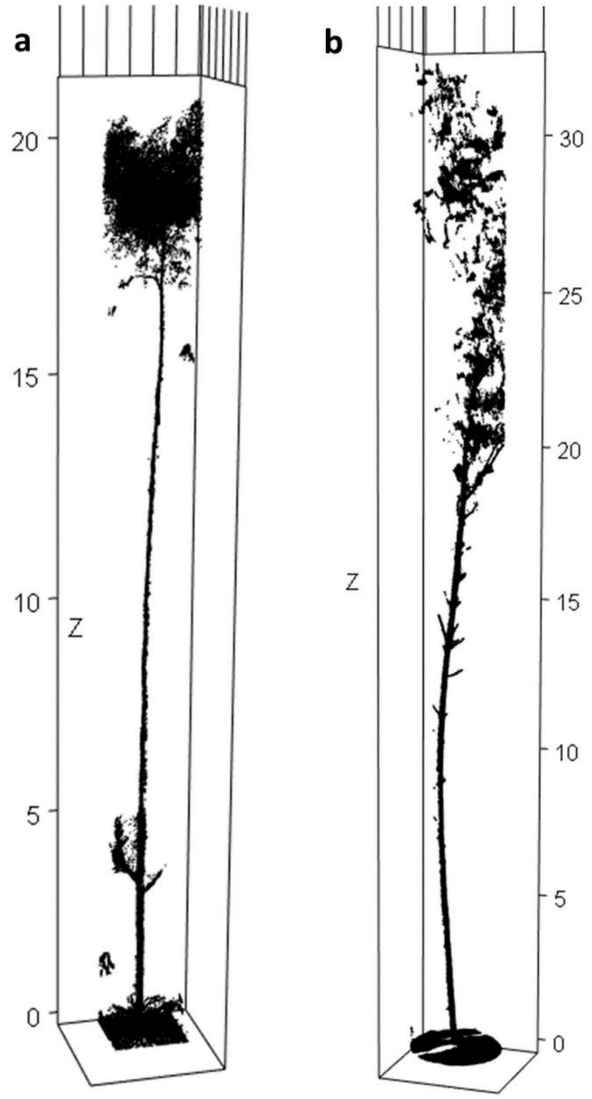


Figure 18. *Eucalyptus sp.* (a) and *Pinus taeda* (b) TLS point clouds.

The quality of estimation was similar throughout all diameter classes (Figure 19) and relative height intervals (Figure 20). Notice that for M2, however, overestimation at the lowest diameters outstands. Both species have sparse canopies and no lower foliage, which relates to their silvicultural treatments, since those species are planted in high densities, which promotes fast growth in height and persistence of canopy only (or mainly) at the highest portions of the tree. Stem occlusion, therefore, is not a big issue when scanning those species, allowing the algorithms to perform alike regardless of height.

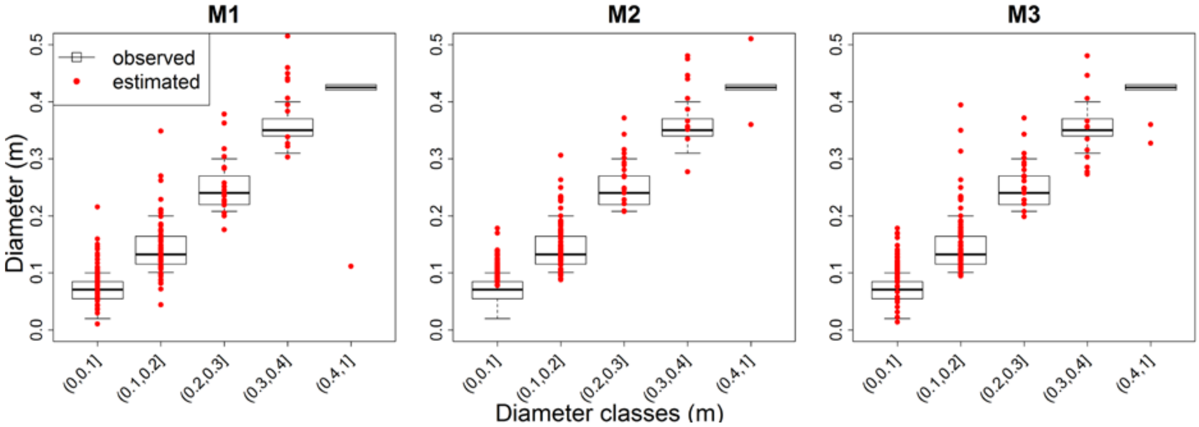


Figure 19. Estimated vs measured diameters distributed in different classes for tropical species.

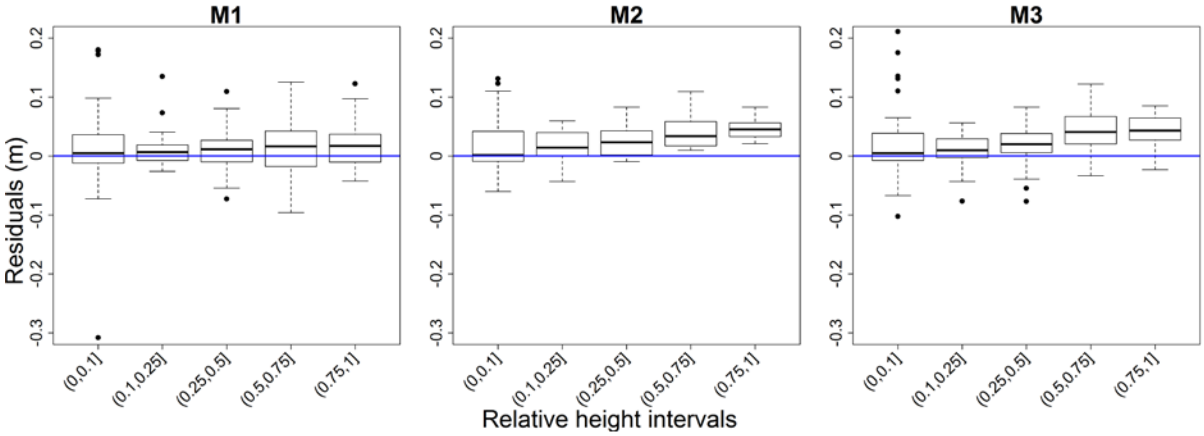


Figure 20. Residuals distribution according to relative tree height, which is the ratio between the height on which a diameter was measured and the tree's top height.

3.1.3 Summarized results

The results are summarized per species on Table 1. Best performance was achieved for *Pinus sylvestris*, followed by *Picea abies*, *Pinus taeda* and *Eucalyptus sp.*, respectively. The Hough transformation followed by RANSAC circle fit (M1) had best performance overall, presenting the least bias for all species and often the least RMSE, except when fitting *Pinus taeda* trees.

Table 1. Summarized assessment measures by method and species. Relative values are related to the average measured diameter for the species.

Method	RMSE (cm)	Rel RMSE	Bias (cm)	Rel bias	r
			Eucalyptus sp.		
M1	3.50	35.3%	1.09	11.0%	0.69
M2	3.82	38.4%	2.39	24.0%	0.69
M3	4.55	45.8%	2.61	26.2%	0.65
			Picea abies		
M1	1.75	9.3%	0.84	4.5%	0.99
M2	3.15	16.7%	1.63	8.6%	0.96
M3	2.28	12.1%	0.99	5.2%	0.98
			Pinus sylvestris		
M1	1.53	7.5%	0.38	1.9%	0.99
M2	1.92	9.4%	0.98	4.8%	0.99
M3	2.22	10.9%	0.92	4.5%	0.98
			Pinus taeda		
M1	7.54	31.3%	1.83	7.6%	0.82
M2	5.31	22.0%	3.49	14.5%	0.91
M3	5.48	22.7%	2.15	8.9%	0.85

3.2 Processing time benchmarking in R

Ten point clouds were randomly selected among the boreal trees, which underwent two extra point cloud density reductions. Most of the function parameters were kept the same, but for much lower densities, when applying M2, *k* was reduced from 30 to 10 points, and when applying M3, the sphere radii for the rough noise filtering were increased to r_1 = 0.1 m and r_2 = 0.2 m. All tests were carried out using a processor Intel® Core™ i7-6500U, base frequency of 2.5 GHz and 16 GB of RAM.

Figure 21 shows the time necessary to process one tree depending on its point cloud size. M2 presented significant improvement in time processing for reduced point cloud size, in special at the pre-filtering phase, being 3 times faster for a density reduction of 50% and 8 times faster for 80% of density reduction. This method evaluates the neighborhood of every point belonging to the point cloud, performing spectral decomposition as many times as the total number of points. By reducing the cloud size, significant time reduction is achieved due to a smaller number of neighborhoods on which spectral decomposition is performed. At the pre-filtering phase, considering the tested point cloud densities as 100%, 50% and 20%, respectively, average processing time was: 68s, 53s and 48s for M1; 703s, 231s and 87s for M2; and 241s, 138s and 108s for M3. Time efficiency for M1 depends more on the pixel size adopted for the Hough transformation, rather than the point cloud size, and since the pixel size was kept as 2.5 cm in all cases, no noteworthy processing time improvement was shown. M3 shares many similarities with M2, but the spectral decomposition is performed on pre-selected neighborhoods from early steps of the algorithm, instead of all possible point neighborhoods, making it more time efficient.

On the fitting phase, there was no time difference for different point cloud densities for the RANSAC based methods (M1 and M3), since the speed is dependent on the number of iterations performed and not on the point cloud size. All RANSAC parameters were kept the same, thus no change in processing speed was detected. For the cylinder fit based on IRTLS (M2), processing time decreased, in average, by factors of 1.4 and 2.2, for 50% and 20% of point cloud size, respectively.

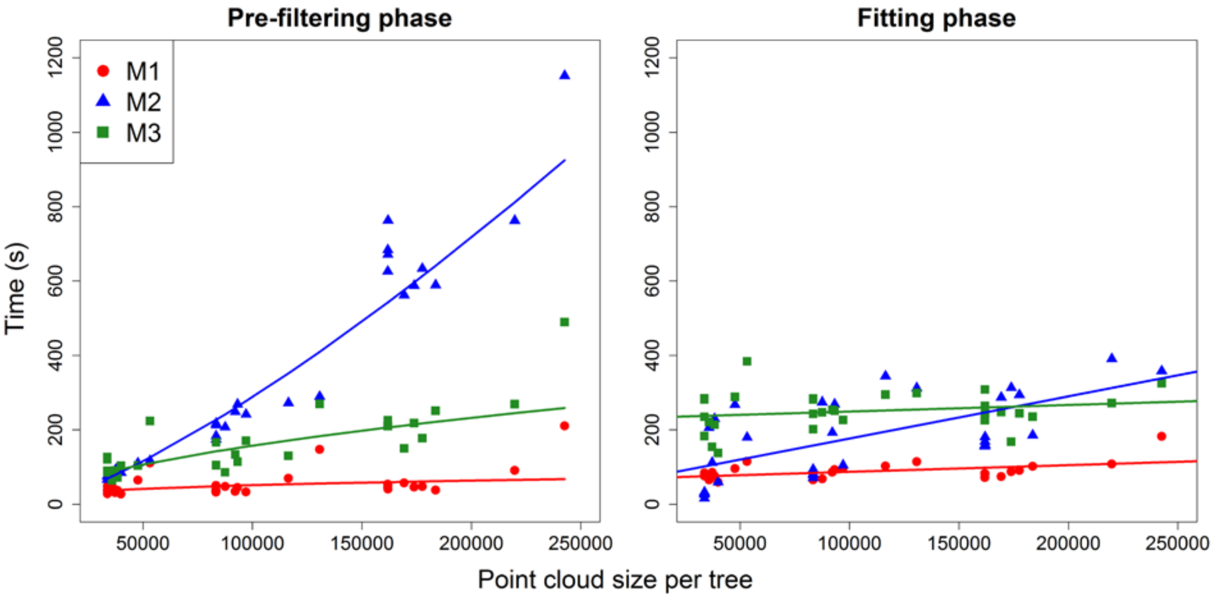


Figure 21. Processing time of the three different methods applied on point clouds of different size.

Estimation quality was checked for the reduced point clouds for all methods (Figure 22). Reducing the point cloud density from 10^4 to 5×10³ points per meter above ground had little effect on the estimation quality. Reducing it further, to 2×10³ points per meter above ground, had no effect on quality for M1, but increased the RMSE by 59% on M3. For M2, the latter density reduction unveiled poor overall performance, yielding a 9 times higher RMSE (from 2.75 cm to 25 cm). The values on Figure 17 considers 10 trees combined, but all the observed outliers belong to estimates made for Spruce. For Pine trees, all methods, regardless of point cloud density, performed well, with all RMSE values below 2.5 cm and bias below 1 cm – e.g. M2, 2×10³ pts/m above ground, Spruce: RMSE = 0.2983 m, bias = 0.0942 m and r = 0.0688; Pine: RMSE = 0.0149 m, bias = 0.0065 m and r = 0.9845. This outcome is, again, probably due to more frequent stem occlusion in LiDAR point clouds taken from Spruce trees, in special young ones that have also persistent lower canopy, decreasing estimation quality alongside spatial resolution. M2 presented better fitting statistics for a density of 5×10³ pts/m than 10^4 pts/m (Figure 13), which is consequence of the reduced tree sample, that ended up removing some of the hardest trees to model.

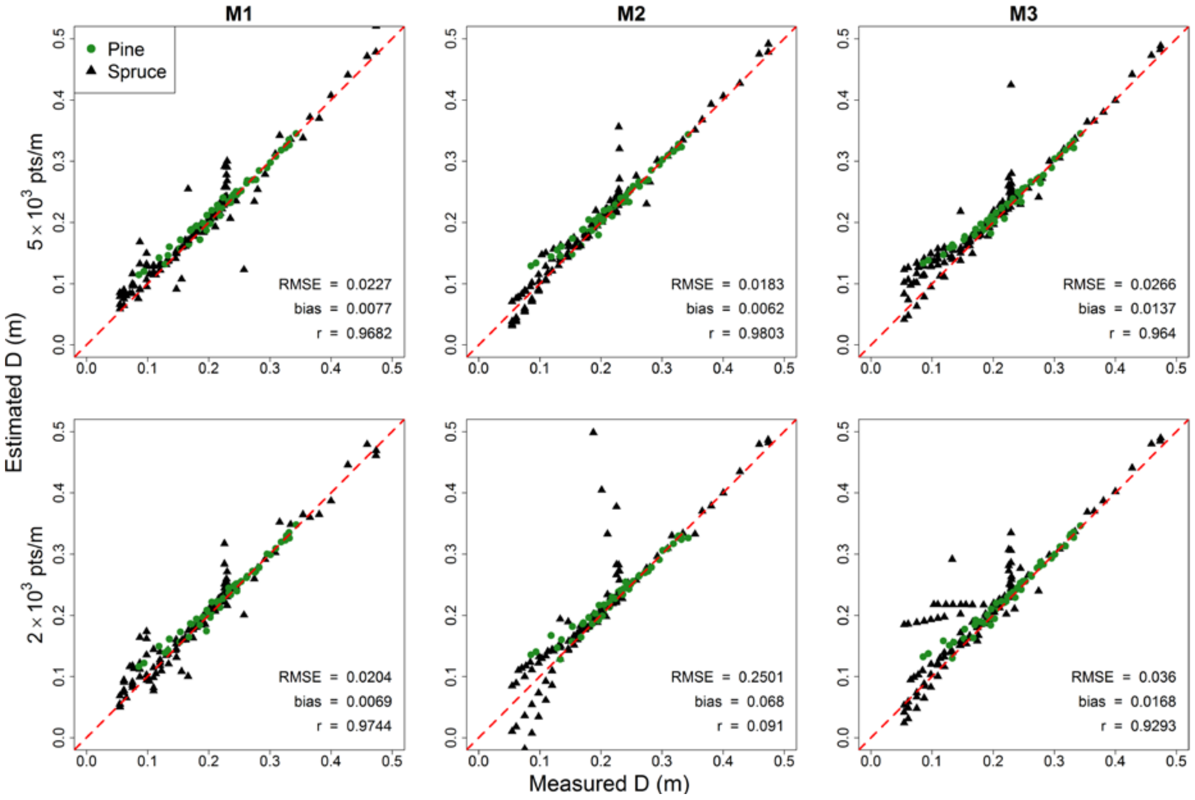


Figure 22. Estimated vs field measured diameters for different point cloud densities (D = diameter; r = Pearson's linear correlation).

No tests were performed on further reduced cloud densities, but for point clouds with a clear view of the lower parts of the bole (i.e. first two meters) it may still be possible to perform accurate stem extraction in even smaller point clouds, since all estimations are made from bottom to top, taking estimates made for one tree section as input for the next one, in a heuristic manner.

3.3 Further discussion and remarks

Overall, the fitting quality observed can be considered satisfactory in most cases. Liang et al. (2014) compiled results from circle and cylinder fitting methods for stem isolation in TLS point clouds published between 2008 and 2012, finding that circle based techniques, with RMSEs ranging from 1.48 to 5.69 cm, were generally more precise than cylinder based ones, with RMSEs ranging from 3.7 to 9.17 cm. Olofsson et al. (2014) found diameter RMSE values in the range of 2.4 cm to 7.5 cm. They tested their adapted circle fit RANSAC algorithm in boreal species point clouds, obtaining best results for Pine (RMSE = 2.38 cm, bias = -0.6 cm), followed by Spruce (RMSE = 6.18 cm, bias = 1.54 cm) and deciduous species (RMSE = 7.48 cm, bias = 2.52 cm). Liang et al. (2014) also compared the TLS automated estimations with field taper measurements, finding a bias of 0.15 cm and RMSE values of 1.17 cm and 1.08 cm for Pine and Spruce, respectively. Furthermore, they also reported that automated estimations were as precise as manual measurements made on the point cloud, and around 3 times less biased. Hackenberg et al. (2015) tested the method described by Raumonen et al. (2013) to estimate above ground biomass for different species, obtaining an average deviation from ground truth measurements ranging from 2% to 21% for whole trees.

In the current study, all methods were applied on point clouds with reduced spatial resolution, with no more than 3×10^5 points per cloud and good accuracy was still achieved. On the referenced studies, the authors processed clouds with much higher resolutions, up to several million points per cloud. Kelbe et al. (2015) estimated DBH from single scan low resolution TLS point clouds, achieving an RMSE of 6.4 cm. Further improvement in TLS-based estimation of tree attributes could be achieved by using higher resolutions (Liang et al. 2014), but this study suggests that reduced point clouds can already provide precise estimates, and rather than increasing resolution globally, keeping higher resolutions only at higher portions of the trees may improve the algorithms overall performance.

Further errors may be introduced by sources not taken into account during the scanning. Other than wind, as already stated, the distance from the trees to the sensor may affect the quality of derived models and estimations based on the point cloud. Olofsson et al. (2014) observed that the precision of estimation decreases the further the trees are from the sensor, which is likely due to shading from neighboring trees closer to the sensors. On the present study, the effect of distance from the trees to the LiDAR sensors was not investigated – all studied single trees were located between 0 and 10 meters away from the laser scanner.

Fitting circles or cylinders on narrow features is troublesome, since the algorithms tend to falsely identify those patterns in small diameters. Here we observed that estimations made for stem segments below 7 cm of diameter are generally not accurate, with modelling being required for those stem portions.

An advantage of method M1, as seen on Figure 20, is that it’s processing time is not dependent on the point cloud size, and it can be applied over point clouds denser than 10^5 pts/m above ground. This method has proven to be precise in most cases. When fitting crooked or leaning stems, in order to take advantage of M1’s time efficiency, circles can be fitted in shorter height intervals, on denser point clouds. Some precision is likely to be lost depending on how much a cylinder is tilted, but it’s a potential time saving alternative for methods M2 and M3.

The pre-filtering phase on M2, based on spectral decomposition of flat surfaces, was first proposed as a stem mapping technique by the original authors (Liang et al. 2012). Thus, an advantage of method M2 is the possibility of using its pre-filtering phase for stem mapping and single tree isolation on plot derived point clouds. The pre-filtering phase in M1, for instance, was tuned to return only the main circle found in the point cloud, thus only being applicable for single tree point clouds. Stem mapping can also be performed by M3, by removing the distance from the main axis criterion from the pre-filtering phase.

Further research will contribute to enhance the quality of TLS-based tree measurements, for instance, allowing implementation of functions to automatically clean point clouds heavily affected by wind; automated best choice of function parameters according to distance from the tree to the laser scanner and point cloud density; simultaneous fitting of forking stems etc. The combination of different pre-filtering and fitting phases should also be evaluated, which will potentially enhance estimation quality for different scenarios.

4. Conclusions

All tested methods have proven to be accurate and precise on good quality single tree point clouds. They perform generally better at lower portions of the stem. The Hough transformation followed by the RANSAC circle fit method presented best overall performance for most species.

All functions developed during this study are available as an open source package on GitHub: https://github.com/tiagodc/TreeLS. Currently the package provides basic functions for manipulation of TLS point clouds, automatic stem isolation from individualized trees and taper extraction.

Acknowledgments

We would like to thank the Sven and Hildur Wingquist foundation for forest research for financing the Swedish data acquisition, managed by the forest remote sensing section at SLU Umeå. The LiDAR study group in the Forestry faculty of the University of São Paulo, ESALQ/USP, and specially Gustavo Almeida, for providing the *Eucalyptus sp.* data. The geomatics laboratory, part of the Forestry department of the Santa Catarina State University, CAV/UDESC, and specially Rorai Pereira Martins Neto, for providing the *Pinus taeda* point clouds. The Erasmus Mundus programme in Sustainable Forest and Nature Management, SUFONAMA, for the scholarship. The Southern Swedish Forest Research Centre, SLU Alnarp, for all the support throughout the year. Dr. Eric Görgens and Dr. Luiz Carlos Estraviz Rodriguez, professors at the UFVJM and USP, respectively, for all their support and valuable inputs throughout this work.

5. References

Adler, D., Murdoch, D. & others, 2016. rgl: 3D Visualization Using OpenGL. R package version 0.95.1441. https://CRAN.R-project.org/package=rgl

Choi, S., Kim, T., Yu, W., 2009. Performance evaluation of RANSAC family. *Br. Mach. Vis. Assoc.* 81, doi:10.5244/C.23.81.

Dassot, M. et al., 2012. Terrestrial laser scanning for measuring the solid wood volume, including branches, of adult standing trees in the forest environment. *Computers and Electronics in Agriculture*, 89, pp.86–93.

Hackenberg, J. et al., 2014. Highly accurate tree models derived from terrestrial laser scan data: A method description. *Forests*, 5(5), pp.1069–1105.

Hackenberg, J. et al., 2015. SimpleTree —An Efficient Open Source Tool to Build Tree Models from TLS Clouds. *Forests*, 6(12), pp.4245–4294.

Hilker, T. et al., 2012. A simple technique for co-registration of terrestrial LiDAR observations for forestry applications. *Remote Sensing Letters*, 3(3), pp.239–247.

Hudak, A.T., Evans, J.S. & Stuart Smith, A.M., 2009. LiDAR Utility for Natural Resource Managers. *Remote Sensing*, 1(4), pp.934–951.

Illingworth, J. & Kittler, J., 1987. The Adaptive Hough Transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(5), pp.690–698.

Kelbe, D. et al., 2015. Single-Scan Stem Reconstruction Using Low-Resolution Terrestrial Laser Scanner Data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(7), pp.3414 – 3427.

Leeuwen, M. van & Nieuwenhuis, M., 2010. Retrieval of forest structural parameters using LiDAR remote sensing. *European Journal of Forest Research*, 129(4), pp.749–770.

Liang, X. et al., 2014. Automated stem curve measurement using terrestrial laser scanning. *IEEE Transactions on Geoscience and Remote Sensing*, 52(3), pp.1739–1748.

Liang, X. et al., 2012. Automatic stem mapping using single-scan terrestrial laser scanning. *IEEE Transactions on Geoscience and Remote Sensing*, 50(2), pp.661–670.

Lichti, D.D., Gordon, S.J. & Stewart, M.P., 2002. Ground-based laser scanners: operation, systems and applications. *Geomatica*, 56(1), pp.21–33.

Lukács, G., Martin, R.R. & D., M. a, 1998. Faithful least-squares fitting of spheres, cylinders, cones and tori forreliable segmentation. *Proc. of European Conference on Computer Vision*, 1, pp.671–686.

Meng, Q. et al., 2007. K Nearest Neighbor Method for Forest Inventory Using Remote Sensing Data. *GIScience & Remote Sensing*, 44(2), pp.149–165.

Olofsson, K., Holmgren, J. & Olsson, H., 2014. Tree stem and height measurements using terrestrial laser scanning and the RANSAC algorithm. *Remote Sensing*, 6(5), pp.4323–4344.

R Core Team, 2016. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Raumonen, P. et al., 2013. Fast Automatic Precision Tree Models from Terrestrial Laser Scanner Data.

Remote Sensing, 5(2), pp.491–520.

Schwarz, B., 2010. LiDAR: Mapping the world in 3D. *Nature Photonics*, 4(7), pp.429–430.

Silva, C. A., Crookston, N. L., Hudak, A. T. & Vierling, L. A., 2015. rLiDAR: LiDAR Data Processing and Visualization. R package version 0.1. https://CRAN.R-project.org/package=rLiDAR