

Automatic Tree Detection and Diameter Estimation in Terrestrial Laser Scanner Point Clouds

Anita Schilling, Anja Schmidt and Hans-Gerd Maas
Institute of Photogrammetry and Remote Sensing,
Dresden University of Technology, Germany
anita.schilling@tu-dresden.de

Abstract. *We present a method to detect trees in 3D point clouds of forest area acquired by a terrestrial laser scanner. Additionally, a method to determine the diameter at breast height of the detected trees is shown. Our method is able to process large data sets bigger than 20 GB in a reasonable amount of time. Results from scans on our test site with different seasonal vegetation are shown. Tree diameters can be reliably determined from the same trees in different scans.*

1. Introduction

Terrestrial laser scanners gained widespread popularity in the last years because point cloud representations of 3D objects can be acquired rapidly and easily. Applications cover a wide range from documentation of cultural heritage sites or accidents to environmental change detection or industrial engineering. We are interested in the development of methods to extract forestry related parameters from scans of forest area. These so-called inventory parameters for a particular forest site, e.g. tree height, diameter at breast height, crown diameter and basis, are important for forest monitoring and management. Usually, a sample set of trees is measured manually by time intensive methods to determine values for a forest. In some cases destructive methods cannot be avoided to obtain reliable results.

Laser scanning is especially attractive for this kind of tasks since it allows fast capturing of scenes in a non-destructive way. The scene analysis can then be performed off-site and already acquired point clouds can always be processed again if other parameters are needed.

Our aim here is the automatic generation of a map of the trees within the laser scanned scene. The

actual number of trees within the area is unknown. Each tree has to be characterized by its diameter at breast height defined at 1.3m w.r.t. the lowest tree trunk point. Furthermore, the tree position is considered to be the center point of the circle from which the diameter is obtained.

A lot of work has also been done on detecting and segmenting trees in airborne laser scans as reported in [11]. Our focus lies on terrestrial laser scanning within the scope of our ongoing project to recover the 3D forest structure, from which we present preliminary results. Similar work on tree detection and diameter estimation was described in [1], but the studied test site was less than half the size of ours. Our study site consists of a birch stock covering an area (160m × 80m) of about 1.3ha. The site was captured from 12 separate scanning positions in winter and spring 2010. The scenes show substantial seasonal changes in vegetation. Because of the size of the test area, our focus is on developing a robust method that can calculate the tree diameter reliably with different understorey vegetation present.

The paper is organized as follows: Section 2 gives an overview of laser scanner techniques and the data specifications. In section 3, the investigated methods are explained in detail. Following, experimental results using scans from our test site are presented in section 4. Finally, section 5 summarizes our findings.

2. Data Acquisition

A terrestrial laser scanner determines the distance to an object by emitting a laser pulse and measuring the time of flight until the reflection of an object is observed at the device or by a phase comparison of the reflection to the initial value. Usually, a laser in the near-infrared is utilized. The strength of the reflected laser pulse affects the measurement accuracy

and is dependent on the incident angle and object material properties.

Most scanners work in their own polar coordinate systems with the scanning mechanism as origin. The vertical and horizontal directions are divided by an angular sampling interval of α_r degrees obtaining a spherical grid around the scanner head. A laser beam is sent through each of the spherical grid points (θ, ϕ) . The distance d to the first object hit by the laser beam is measured. Thus, a particular object can only be measured if the line of sight between scanning mechanism and object is unobstructed. For this reason lower trunk parts are occasionally insufficiently represented due to understorey vegetation which is closer to the scanner than the targeted trees. Additionally, the laser exhibits a beam divergence resulting in an increasing beam diameter with distance. Therefore, several objects might be hit by one laser beam resulting in multiple reflection at the scanner. Some scanner models utilizing phase-comparison average the range values from several observed reflections ([1]), which decreases the accuracy of the scene representation.

The point cloud acquired in polar coordinates (θ, ϕ, d) is then converted to Cartesian coordinates (x, y, z) . The resulting point cloud is a sampled representation of the object surfaces around the scanner. To represent an object from all sides, several scans have to be acquired providing full object coverage. The separate scans need to be registered to the same coordinate system using natural or artificial markers. Although the basic principle of laser scanners is straightforward and provides 3D coordinates of the objects around the device, accuracy depends on the characteristics of the utilized device as well as the object properties. An in-depth description of terrestrial laser scanning can be found in [11].

We used the terrestrial laser scanner Imager 5006i from Zoller+Fröhlich to capture the test site. The scanner uses a phase comparison technique which can resolve distances up to maximal 79m ([12]). Object points which are hit further away are treated as if they would lie within the maximum distance, i.e. $d = d - 79m$, resulting in *ghost points*. As reported in [1], these points have usually a very low reflection strength and can be removed by applying a suitable threshold. Therefore, we set the threshold value for the reflection strength to 0.005. The reflection strength of the measurements is in the interval $[0 \dots 1]$. Only points with a reflection strength

	Session 1	Session 2
time of scan	March	May
binary file size	24 GB	27.6 GB
total no. points	1,738,900,000	2,005,000,000
no. of points used	1,269,056,557	1,471,058,980

Table 1: Laser scanner data specifications. One point consists of 3D coordinates and a value indicating the reflection strength of the particular measurement as floats. The number of points used denotes points with a reflection strength greater than 0.005.

greater than the threshold are used. This reduces the point cloud sizes by about 15% to 29%. Since natural materials, e.g. bark or leaves with low incident angle, also yield low reflection strengths, it cannot be ruled out that a fraction of those are removed as well.

The angular resolution used was 0.0018° resulting in 20,000 range measurements per 360° . The field of view in the vertical direction is limited to 310° due to the scanner tripod. The test site is captured by 12 scans from fixed positions as indicated by figure 1. The 12 separate scans for each scanning session were co-registered by fixed spherical markers mounted on same trees. Registration was performed manually with the Zoller+Fröhlich scanner software. Each separate point cloud was limited to a radius of 37m around the scanner and exported as 3D Cartesian coordinates. The data specifications are summarized in table 1. The first session was scanned in March 2010 when there was no foliage on the trees and the understorey vegetation had been freshly pruned. In May 2010, the second session was acquired when the vegetation had grown significantly and trees were covered by foliage again.

3. Methods

The generation of a Digital Terrain Model (DTM) is necessary to determine the lowest trunk point of each tree. The DTM represents the ground as 2D matrix containing height values as elements. For the DTM generation a method presented in [3] is applied. The actual detection of trees within the point clouds is based on the assumption that the highest density of scan points is on the tree trunks. This was also exploited in [5], [4], [8] and [1]. A problem of the tree detection is the possible mutual occlusion of trees and other vegetation in the scans at different heights. Therefore, the detection method needs to consider several different heights. The targeted birch

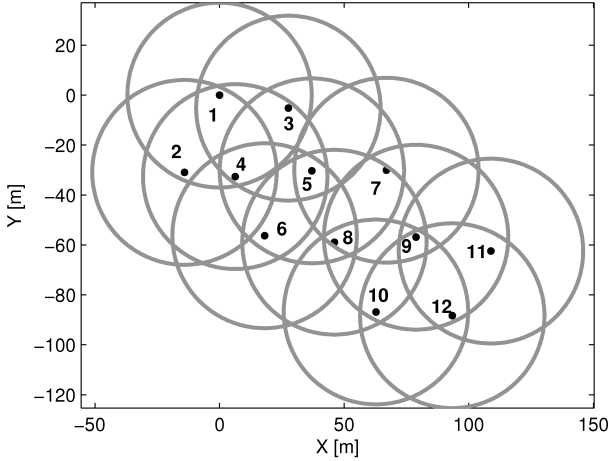


Figure 1: Distribution of the 12 scanner positions per session with radius of $37m$.

trees in the area are $38m$ in height. Smaller trees and some coniferous trees are also present within the area as well as shrubs of different extent. As the number of trees within the test site is unknown, the detection method needs to be robust enough so that no birch trees are missed.

Following tree detection, the points contributing to single trees are analysed separately to find points in breast height. Based on the previously determined DTM plane, the breast height of a particular tree is computed. Then, points at breast height are used to fit a circle to obtain the trunk diameter. Since a tree usually does not grow up perfectly straight, it can hardly be completely located using one 3D point. In spite of this, we use the center point of the fitted circle to indicate the position of a tree. The tree position is used to create an overview map of the test site. For further processing an ample radius around the reported position has to be considered.

The main issue is to determine the boundary of the tree trunk in breast height. This is complicated by the fact that in lower heights many scan points represent other vegetation partially obstructing the trunks. In [1], this task was performed with only few trees on a very small test site, taking about $10h$ processing time. We present a method to achieve reliable results in a reasonable amount of time for a comparatively large data set. The method is summarized in algorithm 1. The position and diameter at breast height values for the trees are eventually summarized as a map of the trees on the test site.

1. determine DTM plane g_{DTM} for 3D point set E from all positions of a scan session
2. detect trees and calculate a set of tree position estimates T (see algorithm 2)
3. for each tree position $t \in T$
 - (a) load 3D points within bounding volume from E ,
 $P_t = \{p \in E : t_x - b_x \leq p_x \leq t_x + b_x \wedge t_y - b_y \leq p_y \leq t_y + b_y\}$
 - (b) if $|P_t|$ is sufficient determine DTM plane t_{DTM} from P_t , otherwise use g_{DTM}
 - (c) calculate circle estimate c in height h_s (see algorithm 3)
 - (d) compute lowest trunk point height k_z^1 by projecting the circle center onto the DTM plane
 - (e) calculate circle update c (see algorithm 3)
 - (f) compute new lowest trunk point height k_z^2
 - (g) if $|k_z^1 - k_z^2| > \epsilon$ then repeat starting at step (c) with $h_s = h_s + h_o$
 - (h) log resulting diameter at breast height $t_{dbh} = 2 \cdot c_{radius}$ and tree position $t_p = (c_x, c_y, c_z)$ for the tree

Algorithm 1: Scheme of subtasks for tree detection and diameter calculation.

3.1. Digital Terrain Model generation

The method to generate the DTM that is summarized here, was originally presented in [3]. The xy -plane is partitioned into a 2D grid with cell size s_c . When projected onto this plane, several 3D points lie in the same cell. For each single cell, the z -axis is divided in several bins each covering a height interval of s_l . Points which are located within the current cell are counted in the bins corresponding to their z coordinate. Thus a height histogram is built for each cell from the point numbers. The histogram bin with the highest number is assumed to be the ground and the bin height is assigned to the current cell.

If a tree trunk was occluded by vegetation closer to the scanner, then there are hardly any points at the real ground height. In this case lower histogram bins are empty because the trunk points are only contributing to higher bins of the particular cell. The

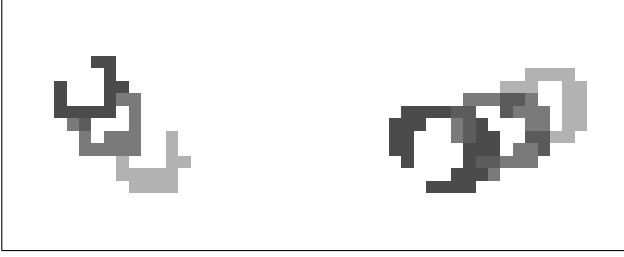


Figure 2: Arc- or circle-like shapes caused by tree trunks in different height layers indicated by gray value.

maximum bin is determined far to high resulting in a false height value. Therefore, the grid cell heights need to be filtered. If a cell height is too high in comparison to its neighbouring cells and a determined threshold then the cell value is removed.

Afterwards, cells with missing height values are interpolated using neighbouring grid cell heights. The 2D index of each cell is converted to (x, y) coordinates in the point cloud coordinate system with the cell height as z coordinate. Finally, an adjusted plane is fitted to this 3D point set.

A DTM is generated for each point cloud of one scan session separately. To obtain a general DTM for the entire scan session, the separate DTMs are merged. The DTMs of separate point clouds are overlapping in several parts of the test area. In these cases uninterpolated height values were preferred and averaged if multiple values were available.

3.2. Tree Detection

Our tree detection method is presented in algorithm 2. As already mentioned, it is based on the assumption that in the forest area the highest density of scan points are located at the tree trunks. To benefit from the nearly full trunk coverage in the overlapping parts of the point clouds, the entire scan session needs to be processed at once. Therefore a height slice of the scan session is considered. Points within that slice are projected to a 2D grid that partitions the xy -plane. For each cell, the number of points within the cell is counted. Grid cells with a point count less than a defined threshold $minNbPoints$ are cleared. If a suitable threshold is applied, the non-zero cells are likely to correspond to positions at the tree trunks. The trunk boundaries appear as components with an arc- or circle-like shape as shown in figure 2, though the cross section of a trunk is rarely a perfect circle. A more detailed analysis of the trunk points is neces-

1. at different heights h_i , slices of thickness t , project all points within onto a plane l_i parallel to xy -plane
2. partition l_i by a 2D grid g_i , count no. of points in each grid cell
3. grid g_i represented by an $m \times n$ matrix \mathbf{I}^i where
$$\mathbf{I}^i(m, n) = \begin{cases} 1 & g_i(m, n) > minNbPoints \\ 0 & otherwise \end{cases}$$
4. concatenate matrices \mathbf{I}^i with OR operation, thus
$$\mathbf{K}(m, n) = \begin{cases} 1 & \mathbf{I}^i(m, n) = 1 \\ 0 & otherwise \end{cases}$$
5. dilate \mathbf{K} with square structure element of size $s \times s$
6. find and uniquely label components in \mathbf{K} by connected component labelling
7. find components in \mathbf{I}^i by connected component analysis, join components c by component number from \mathbf{K} thus
$$M[\mathbf{K}(c_m, c_n)] = M[\mathbf{K}(c_m, c_n)] \cup (c_m, c_n)$$
8. for each index list in M calculate 2D centroid from indices, convert to point cloud coordinate system, resulting 2D coordinates are tree position estimate

Algorithm 2: Detection of trees in point clouds.

sary for each tree in any case, therefore determining approximate coordinates of the tree location is sufficient for the tree detection step.

It is possible that a tree does not appear on the 2D grid of a particular height because of occlusions. Hence, several different heights have to be analysed. The components in each of the 2D grids are detected by a connected component labelling algorithm ([9]). Because of the skewed tree growth, components corresponding to the same tree in grids of different heights do not necessarily cover the same grid cells. But components of the same tree are inevitably close together and are joined to clusters. Seldomly, components resulting from branches with high scan coverage produce separate clusters, which are at the moment treated as valid detections as well. The 2D centroid of each cluster is computed and constitutes the tree position.

Finally, for each estimated tree position, all points

located within a bounding volume are exported to a separate file. The bounding volume is a box of square base with the position estimate at its center. The generation of these smaller point clouds for each presumed tree is the most time intensive part using standard hardware, because of the high number of read and write operations. If sufficient memory, i.e. at least 30 GB, could be provided such that all point clouds of a scan session can be hold within memory, the creation of temporary point clouds for the tree position estimates would be unnecessary.

3.3. Tree Location and Diameter Determination

For tree location and diameter determination, each point cloud section belonging to an estimated tree position is processed separately. First, a DTM is calculated for the point cloud section. If this fails because of an insufficient number of points, an adjusted plane is used instead that is fitted to the 3D points of the respective section of the session DTM.

A first computation of the trunk circle center is necessary to determine the lowest trunk point height accurately. The largest aggregation of 3D points within a circular slice at height h_s around the estimated position is assumed to be the trunk. This subset of points can be found by taking the maximum and neighbouring bins greater than a predefined threshold from histograms of point numbers along the x and y axis as indicated in figure 3. A circle is calculated with Kasa method ([6]) using the 3D point subset. The circle equation is rearranged to

$$-2c_x x - 2c_y y + c_x^2 + c_y^2 - c_r = -(x^2 + y^2) \quad (1)$$

and transformed with the given point set to matrix representation

$$\mathbf{A}_{n \times 3} \cdot \mathbf{k}_{3 \times 1} = \mathbf{l}_{n \times 1} \quad (2)$$

with n denoting the number of the considered points. The solution vector \mathbf{k}

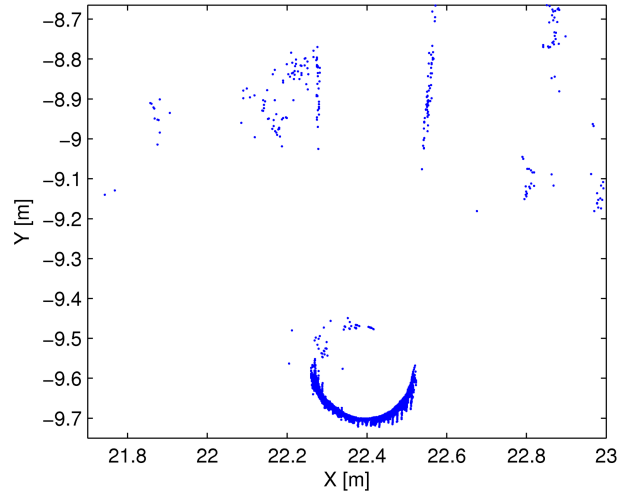
$$\mathbf{k} = \left[-2c_x \quad -2c_y \quad c_x^2 + c_y^2 - c_r \right]^T \quad (3)$$

is obtained by least-squares minimization

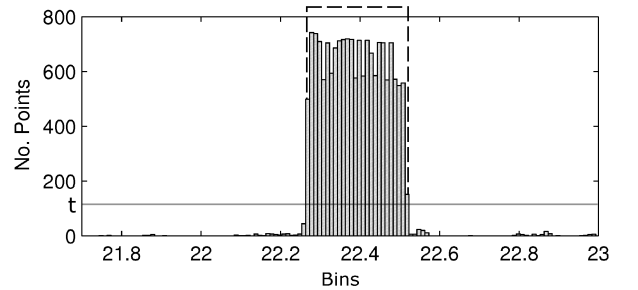
$$\mathbf{k} = (\mathbf{A}^T \mathbf{A})^{-1} \cdot \mathbf{A}^T \cdot \mathbf{l} \quad (4)$$

of the algebraic distances. Following, the elements of \mathbf{k} have to be solved for the circle parameters.

The 2D center point (c_x, c_y) is projected onto the DTM plane to calculate the trunk point height k_z^1 . In



(a) Plot of points on xy -plane



(b) Histogram with bin size of 0.01m along x axis.

Figure 3: Determination of tree circle estimate using a histogram of point amounts along the x and y axis with predefined threshold t .

a defined height of 1.3m w.r.t. the lowest trunk point, a new set of points within a circular slice around the calculated circle center is considered as summarized in algorithm 3. To find a cluster of points in the set resembling a circle the Circular Hough Transform as reported in [10] is utilized. The Circular Hough Transform is based on the fact, that the distance of every point on the perimeter of a circle c_m with known radius r is r . When a circle c_p of the same radius r is drawn around each perimeter point, all circles c_p will necessarily meet at the center point of circle c_m as shown in figure 4.

In [1] the Circular Hough Transform was applied to the non-empty cells of a 2D grid. Previously, the point set was projected onto this grid partitioning the xy -plane and thresholded like explained in section 3.2. Present on the 2D grid are arc- or circle-like shapes from trunks, but also components caused by branches. The accumulation of circles around the component cells on the grid results in only weak sup-

port for a particular circle. Instead of the few number of non-empty grid cells, we use each 3D point of the considered set for the circle accumulation. We initialize an empty 2D grid partitioning the xy -plane. The 3D points of the considered slice are projected onto the grid and a circle of size r is drawn around each of the points. For each cell the number of circles passing through it are counted.

In this way many more points are voting for the same circle center resulting in a distinct peak in the grid. Because only an estimate c_r of the precise radius is known, the Circular Hough Transform is applied several times with an increasing radius r . The maximum peak on the grid over all iterations denotes the new circle center (c_x, c_y) . The circle radius c_r is updated with the radius r of the corresponding iteration. Finally, a new set of 3D points S is considered containing only points at the previously determined height $h_b \pm \frac{t}{2}$ within a radius defined by c_r with an additional offset d_3 . Again, the algebraic circle fit of equation 1 to 4 is used to calculate values

$$\mathbf{c} = (c_x \ c_y \ c_r)^T \quad (5)$$

for the circle parameters. The circle is then fitted ([7]) by a least-squares minimization as in equation 4 with

$$\mathbf{A} = \begin{bmatrix} -\frac{x-c_x}{c_r} & -\frac{y-c_y}{c_r} & -\mathbf{1} \end{bmatrix}_{n \times 3} \quad (6)$$

and

$$\mathbf{1} = \left[c_r - \sqrt{(x - c_x)^2 + (y - c_y)^2} \right]_{n \times 1} \quad (7)$$

to minimized the geometric distances. The resulting improvements in vector \mathbf{k} are added to the circle parameters. The center point of the adjusted circle is the location of the tree t_c . The diameter $t_{DBH} = 2 \cdot c_r$ is obtained from the circle radius.

The lowest trunk point is determined again by projecting the circle center point onto the DTM. If the resulting trunk point differs from the previously defined height in comparison to a suitable threshold, then a new iteration of the method is performed unless the maximum number of iterations is reached. In this case the first circle estimate is determined anew, starting at a height of $h_s = h_b + h_o$.

4. Experiments

We applied our method to both scan sessions of the test site. The results are summarized in table

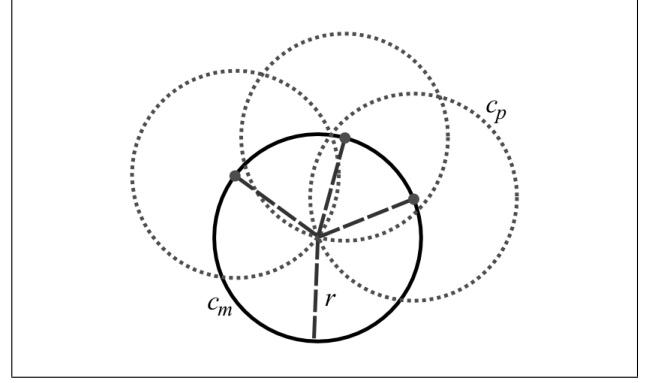


Figure 4: Circular Hough Transform

1. create 3D point set

$$S = \left\{ p \in P_t : h_b - \frac{t}{2} \leq p_z \leq h_b + \frac{t}{2} \wedge d(p, t_c) < c_r + d_2 \right\}$$
2. adjust $r_{min}, r_{max}, r_{step}$ according to current circle radius estimate
3. for $r = r_{min}, r < r_{max}, r = r + r_{step}$
 - (a) project all 3D points of S onto a plane l parallel to xy -plane
 - (b) draw a circle with radius r around each point $s \in S$
 - (c) partition plane l by a 2D grid g with cell size s_c , in each cell count no. of circles passing through
4. determine radius r of iteration with maximum cell value in grid g
5. convert grid indices to point cloud coordinates (c_x, c_y) for circle c and update circle radius c_r with r
6. recreate 3D point set

$$S = \left\{ p \in P_t : h_b - \frac{t}{2} \leq p_z \leq h_b + \frac{t}{2} \wedge d(p, t_c) < c_r + d_3 \right\}$$
7. update circle c with a new circle estimate using S
8. calculate adjusted circle fit with c and S

Algorithm 3: Determination of tree points in breast height and calculation of radius by circle fitting.

2 and processing times are reported in table 3. We are not able to assess the results of the tree detection method regarding its completeness, because the

number of birch trees actually present on the test site is not documented. For this reason, the number of false negatives, i.e. trees which were not detected, is also unknown.

We evaluated the results of the detection method manually. 99% of all detected items in session one and 97% in session two are actual trees present on the test site. 91% and 92% of all detections in the respective session are the targeted birch trees, while 8% and 5% are other small or coniferous trees. 1% and respectively 3% of all cases are false positives, which means that structures have been detected which are not trees. These detections were caused by branches with high scan coverage. In the second session more items were detected falsely which is probably due to the foliage present on branches.

We do not have ground truth values for the DBHs of the trees. The evaluation of the DBH only on basis of the computed values is not reliable. Tree diameters are quite variable, which makes the definition of a particular interval difficult. Furthermore, a circle fitted wrongly to a set of points belonging to a shrub nearby the sought-after trunk can also yield a diameter value, which is typical for birch trees.

For this reason, it was verified visually whether the points used for the calculation of the DBH are actually located at the respective tree trunk. For 95% of all detected birch trees in the first and 92% in the second session sufficient correctly located points were selected and therefore an accurate DBH value could be calculated. The averaged standard deviation of the point sets to the fitted circles is $7mm$ and $8mm$. For 5% and respectively 8% of the birch trees the trunk was not sufficiently covered by scan points or the points used for DBH calculation were not localized on the trunk yielding an invalid DBH value.

We are interested in the seasonal change of the vegetation. Before a comparison of the tree appearance in both sessions is possible, the scan sessions have to be registered to each other. The scan sessions exhibit a rotation to each other, but the correspondences and coordinates of the sphere targets are known. The sphere targets were previously used to register separate point clouds of one session to each other. We applied the Iterative Closest Point algorithm ([2]) to obtain a transformation matrix M using sphere target correspondences. With M the tree positions of the second scan session could be transformed to the coordinate system of the first session. Then we established correspondences between tree

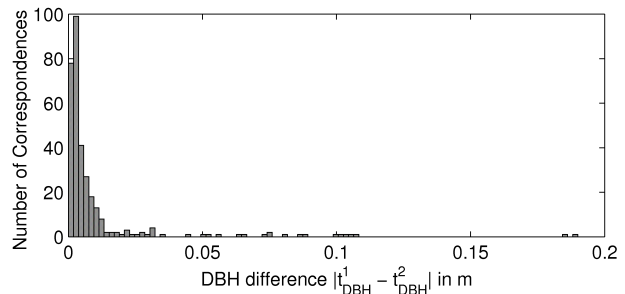


Figure 5: Histogram of DBH differences between the corresponding birch trees.

	session 1	session 2
total detections	363	368
detected birch trees	331	325
false detections	3	11
other detected trees	29	32
valid point set for DBH	316	299
correspondences	323	

Table 2: Results of first and second scan session. All detections were manually checked. The number of false detections is caused by branches which were interpreted as separate vegetation structures.

total processing times	session 1	session 2
DTM generation	$9min$	$9min\ 28s$
tree detection	$4min\ 57s$	$5min\ 46s$
tree separation	$147min$	$157min$
DBH calculation	$20min$	$17min$

Table 3: Processing times for the first and second scan session.

positions from both sessions manually.

A total of 323 distinct birch tree correspondences were found. For this set the DBH values were compared. The absolute differences of the DBH values of each pair $\epsilon = |t_{DBH}^1 - t_{DBH}^2|$ were calculated and are shown in figure 5 as histogram. 90% of the correspondences exhibit a DBH deviation of less than $2cm$ and even 63% of less than $5mm$. Regarding a maximum DBH difference of $1cm$, the DBH value could reliably determined in both scan sessions for an amount of 268 birch trees present on the test site.

5. Conclusion

We have shown that the generation of a map of trees on a comparably large test site is feasible in a reasonable amount of computation time. Although

we cannot entirely evaluate the detections on the test site concerning their completeness, the results look promising. The greatest amount of detections are the targeted birch trees and their diameters at breast height could be determined precisely from the available terrestrial laser scanner point clouds.

There are still a lot of possibilities for improvements. The tree detection method needs to be evaluated whether actually all trees are detected. Further processing would profit from a more detailed analysis by which kind of vegetation structure the detection was caused. False detections from branches or smaller, unwanted trees on the test site could be avoided. Additionally, it is necessary to improve on the diameter calculation. The diameter at breast height has not been accurately calculated for all birch trees though the scan coverage was sufficient.

The DTM generation from scans with dense understorey vegetation is more error-prone, because the actual ground is not sampled enough. We will try to use the DTM from winter scans as basis for all sessions to obtain more reliable height values. The mutual registration of the scan sessions will be necessary for that. The calculation of an appropriate transformation matrix might be improved by utilizing the established tree position correspondences as well.

We have two more scan sessions captured in July and October 2010 exhibiting considerably more seasonal change in comparison to the first session. Therefore, the changing of parameter values is probably not appropriate and a way to adaptively adjust parameter values of the processing step would be beneficial. Furthermore, the representation of the tree location as a single 3D point is in fact not sufficient. Instead, we will aim to capture the topology of a tree directly.

References

- [1] T. Aschoff and H. Spiecker. Algorithms for the automatic detection of trees in laser scanner data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(8/W2), 2004. 1, 2, 3, 5
- [2] P. Besl and N. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992. 7
- [3] A. Bienert, S. Scheller, E. Keane, G. Mullooly, and F. Mohan. Application of terrestrial laser scanners for the determination of forest inventory parameters. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36, 2006. 2, 3
- [4] J. G. Henning and P. J. Radtke. Detailed stem measurements of standing trees from ground-based scanning lidar. *Forest Science*, 52:67–80, 2006. 2
- [5] C. Hopkinson, L. Chasmer, C. Young-Pow, and P. Treitz. Assessing forest metrics with a ground-based scanning lidar. *Canadian Journal for Forest Research*, 34:573–583, 2004. 2
- [6] I. Kasa. A curve fitting procedure and its error analysis. *IEEE Transactions on Instrumentation and Measurement*, 25:8–14, 1976. 5
- [7] T. Luhmann. *Nahbereichsphotogrammetrie: Grundlagen, Methoden und Anwendungen*. Wichman Verlag, Heidelberg, 2000. 6
- [8] H. Maas, A. Bienert, S. Scheller, and E. Keane. Automatic forest inventory parameter determination from terrestrial laserscanner data. *International Journal of Remote Sensing*, 29(5):1579–1593, 2008. 2
- [9] L. G. Shapiro and G. C. Stockmann. *Computer Vision*. Prentice Hall, 2001. 4
- [10] M. Sonka, V. Hlavac, and R. Boyle. *Image processing, analysis, and machine vision*. PWS Publishing, second edition, 1998. 5
- [11] G. Vosselman and H. Maas, editors. *Airborne and Terrestrial Laser Scanning*. Whittles Publishing, 2010. 1, 2
- [12] Zoller+Fröhlich GmbH. Imager 5006i - An improved system based upon the highly regarded Z+F IMAGER 5006 http://www.zf-laser.com/BROSCHUERE%20Z+FIMAGER_5006I_E_01.07.09.kompr.pdf, July 2009. 2