Computers and Electronics in Agriculture 143 (2017) 165–176

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

Performance of stem denoising and stem modelling algorithms on single tree point clouds from terrestrial laser scanning

Tiago de Conto[a,*], Kenneth Olofsson[b], Eric Bastos Görgens[c], Luiz Carlos Estraviz Rodriguez[d], Gustavo Almeida[d]

[a] Swedish University of Agricultural Sciences (SLU), Sundsvägen 6, 230 53 Alnarp, Sweden
[b] Swedish University of Agricultural Sciences (SLU), Skogsmarksgränd, 907 36 Umeå, Sweden
[c] Universidade Federal dos Vales do Jequitinhonha e Mucuri (UFVJM), Campus JK, Alto da Jacuba, 39100-000 Diamantina, MG, Brazil
[d] Escola Superior de Agricultura Luiz de Queiroz – Universidade de São Paulo (ESALQ-USP), Avenida Pádua Dias, 11, 13418-900 Piracicaba, SP, Brazil

ARTICLE INFO

Keywords:
LiDAR
Cylinder/circle fit
Tropical and boreal tree species
Robust estimation
RANSAC

ABSTRACT

The present study assessed the performance of three different methods of stem denoising and three different methods of stem modelling on terrestrial laser scanner (TLS) point clouds containing single trees – thus validating all tested methods, which were made available as an open source software package in the R language. The methods were adapted from common TLS stem detection techniques and rely on finding one main trunk in a point cloud by denoising the data to precisely extract only stem points, followed by a circle or cylinder fitting procedure on stem segments. The combination of the Hough transformation stem denoising method and the iteratively reweighted total least squares modelling method had best overall performance – achieving 2.15 cm of RMSE and 1.09 cm of bias when estimating diameters along the stems, detecting 80% of all stem segments measured on field surveys. All algorithms performed better on point clouds of boreal species, in comparison to tropical Eucalypt. The point clouds underwent reduction of point density, which increased processing speed on the stem denoising algorithms, with little effect on diameter estimation quality.

1. Introduction

Terrestrial Laser Scanning (TLS) sensors are increasing in popularity in the Forestry sector, due to the high resolution and precision of the three-dimensional information provided at the sample plot level, making further 3D modelling and tree reconstruction possible and allowing foresters to extract dendrometric variables with high accuracy from point clouds (Liang et al., 2016). In order to apply the TLS technology in forest inventory at its full potential, robust tool sets to extract useful information from TLS point clouds, and that perform well on a variety of circumstances are required – especially stem denoising and stem modelling, since stem volume is the most targeted variable in commercial forest management. TLS is a technique that could be used to develop locally adjusted stem taper functions (Trincado and Burkhart, 2006; Sun et al., 2016).

Routines for processing TLS point clouds containing tree data have been tested, with applications ranging 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 reliable algorithm for stem isolation, thus, requires a level of generalization that allows it to capture the essential information of a variety of tree shapes in a point cloud.

Some commonly adopted assumptions are the vertical orientation of the bole (approximately orthogonal angle with the ground), approximately circular stem cross sections and approximately cylindrical stem segments. Those assumptions are the core of many already implemented algorithms, but a key issue must be tackled beforehand: fitting circles and cylinders works well on point clouds with the stem clearly visible, but can't be applied directly upon noisy point clouds. Good methods for noise filtering are paramount for a robust stem isolation algorithm.

There is a handful of methods to denoise stem point cloud data. One early technique is the Hough transformation, used by Simonse et al. (2003), Rabbani and van den Heuvel (2005) and Olofsson et al. (2014). The Hough transformation is a technique in which every data point "votes" for a cylinder center. Another method is to use saliency features of a point dataset to classify different parts of a TLS dataset of a forest

* Corresponding author.
E-mail address: tiagodc@usp.br (T. de Conto).

http://dx.doi.org/10.1016/j.compag.2017.10.019
Received 21 January 2017; Received in revised form 17 October 2017; Accepted 22 October 2017

plot into, for instance, stems and canopy. Several authors use this technique, such as Lalonde et al. (2006), Liang et al. (2012) and Xia et al. (2015). Another way to denoise stem data is to store the TLS data in a 3D voxel space and use morphological operations, as in the studies by Gorte and Pfeifer (2004), Gorte and Winterhalder (2004), Vonderach et al. (2012) and Raumonen et al. (2013).

The most common way to model the volume or biomass of a tree's stem is to fit circles or cylinders to TLS data points by non-linear, or linearized, least square adjustment (Simonse et al., 2003; Aschoff and Spiecker 2004; Hopkinson et al., 2004; Pfeifer and Winterhalder, 2004; Thies et al., 2004; Forsman and Halme 2005; Watt and Donoghue 2005, Henning and Radtke, 2006, Lalonde et al., 2006; Brolly and Király, 2009; Lindberg et al., 2012, Pueschel et al., 2013, Raumonen et al., 2013). Some studies use robust shape fitting of cylinders, like Liang et al. (2012, 2014b), who used iteratively reweighted least squares and *m* estimators. Another option is to use the RANSAC method (Chum 2005; Choi et al., 2009; Schnabel et al., 2007; Olofsson et al., 2014) which is a robust iterative method that can be used to find circles or cylinders directly in point cloud data.

Many studies only evaluate one algorithm, often focusing on tree detection and estimating diameter at breast height (dbh). A few studies evaluated the performance of TLS on measurements of stem taper contrasted to field data, Henning and Radtke (2006), Maas et al. (2008), Liang et al. (2014b), and Mengesha et al. (2015). One feature of interest not yet investigated is to test various combinations of stem denoising and stem modelling algorithms to evaluate which combinations work best.

The emphasis of this study is not to develop a tree detection algorithm that operates on TLS data, as other studies have already shown ways to do that. In this study we compare alternatives to extract stems from noisy data. Three different algorithms for stem denoising and three algorithms for stem modelling were chosen and all combinations of those were tested and evaluated. Also, the algorithms were tested on single tree point clouds with different densities, aiming to investigate the performance under lower density surveys. The code during the study is available as an open source package in R language, containing functions for dealing with TLS forest data.

2. Methods

2.1. Data acquisition

The TLS data was acquired through multiple scan setups (Hilker et al., 2012). Mixed plots of Norway spruce (mean dbh = 29.57 cm, mean Ht = 22.4 m) and Scots pine (mean dbh = 25.04 cm, mean Ht = 20.7 m) of varying ages were scanned in Northern Sweden. Seven years old eucalypt trees were taken from a scanned plot in South-Eastern Brazil (mean dbh = 12.80 cm, mean Ht = 21.28 m) – dbh: diameter at breast height, Ht: total height. Table 1 summarizes the laser scanner settings of both surveys. All point clouds contained single trees only – 1.5 m radius from the tree's approximate center. Tree detection at the plot level is not addressed in our study – such techniques were described in detail by Liang et al. (2012) and Olofsson et al. (2014), for instance.

Table 1
Laser scanner settings of both surveys.

	Sweden	Brazil
Tree species	*Pinus sylvestris* L. *Picea abies* L. Karst.	*Eucalyptus* sp.
Scanning period	August 2013	October 2015
Laser scanner	Leica ScanStation C10	RIEGL VZ-400 V-Line
Wave length (nm)	532	1550
Beam divergence (mrad)	0.1	0.35
Pulse rate (kHz)	50	122
Field of view (degrees)*	360h 270v	360h 100v

* h and v refer to the horizontal and vertical planes, respectively.

Twenty-five scanned trees – 10 spruces, 9 pines and 6 eucalypts were manually calipered – at 1.3 m above ground (dbh) and every meter from the ground level up to 15 m above ground on the Norway spruce and Scots pine trees, and up to top height on the eucalypt trees. Those trees were processed using our implementations of the following methods. The callipered diameters were used as ground truth for validation, thus only trees with both laser data and field measurements were analyzed. On the present study, the effect of distance from the trees to the LiDAR sensor was not investigated – all studied trees were between 0 and 10 meters from the laser scanners on multi-scan setups.

2.2. Denoising techniques

2.2.1. Hough transformation (D1)

The Hough transformation is a technique that searches for primitive shapes (e.g. lines, circles, etc.) on raster datasets (Illingworth and Kittler, 1987). On our study, we adapted the algorithm to find circular shapes on two dimensional horizontal layers of the point clouds.

Taking a tree point cloud, it is first sliced from bottom to top – every 0.5 m. Every slice contains a stem segment assumed orthogonal to the ground – i.e. parallel to $z = [0, 0, 1]$. A baseline segment is defined at an arbitrary height interval (1–2 m above ground) and its approximate circle parameters are estimated – center coordinates (x, y) and radius, by applying the Hough transformation (Fig. 1). Points below the upper height limit of the baseline segment, which were far from its center (baseline's radius plus 5 cm), were disregarded. Above the baseline segment, the search for circle centers in each segment is constrained to the x and y range found for the previous segment. For those segments, points further than its radius plus 1 cm from its estimated center are disregarded. Every pixel displays its local point density (ratio between its number of points and the pixel containing the most points). All pixels with more than 30% density are tested as center coordinates, and several radii between arbitrary limits (from 2.5 to 30 cm) are tested on each valid pixel. The radius between iterations increases by the pixel size – here we adopted 2.5 cm. Another, initially empty, raster layer with the same extent is used to count the number of circle intersections per pixel (votes). The pixel with most votes from circles with same radius is the estimated center – for that radius. The image displays 3 iterations (3 radii) of the Hough transformation. The same pixels are used as centers in all 3 iterations, the first iteration tests a small radius (left), the second iteration tests the true radius (center) and the third iteration tests a large radius (right). Note the central image contains the pixel with most votes, and thus best estimates the circle parameters.

2.2.2. Eigen decomposition of flat surfaces (D2)

This method starts investigating the cloud pointwise. Every point has flatness (Eq. (1)) and verticality (angle between its normal vector and z = $[0, 0, 1]$) values assigned to it by performing eigen decomposition on its immediate point neighborhood of 30 points (Fig. 2). All (approximately) non-flat (FL < 0.8), non-vertical (angle with z $> 10°$) points are disregarded, then points which are close in the 3D space are clustered (up to 10 cm distant), and overlapping projections of those clusters on the xy plane are merged. This ensures that points belonging to the trunk are grouped together, even if gaps occur in the point cloud. Small clusters are assumed to be noise and also disregarded.

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

where

FL = flatness;
λ_i = eigenvalues, where $\lambda_1 \geq \lambda_2 \geq \lambda_3$.