Received: 3 May 2018 | Accepted: 29 September 2018

DOI: 10.1111/2041-210X.13121

APPLICATION

Methods in Ecology and Evolution | BRITISH ECOLOGICAL SOCIETY

Extracting individual trees from lidar point clouds using *treeseg*

Andrew Burt[1] | Mathias Disney[1, 2] | Kim Calders[3]

[1]Department of Geography, University College London, London, UK

[2]NERC National Centre for Earth Observation (NCEO), Leicester, UK

[3]CAVElab - Computational & Applied Vegetation Ecology, Ghent University, Ghent, Belgium

Correspondence
Andrew Burt
Email: a.burt@ucl.ac.uk

Funding information
Natural Environment Research Council, Grant/Award Number: NE/J016926/1 and NE/N00373X/1

Handling Editor: Sarah Goslee

Abstract

1. Recent studies have demonstrated the potential of lidar-derived methods in plant ecology and forestry. One limitation to these methods is accessing the information content of point clouds, from which tree-scale metrics can be retrieved. This is currently undertaken through laborious and time-consuming manual segmentation of tree-level point clouds from larger-area point clouds, an effort that is impracticable across thousands of stems.
2. Here, we present *treeseg*, an open-source software to automate this task. This method utilises generic point cloud processing techniques including Euclidean clustering, principal component analysis, region-based segmentation, shape fitting and connectivity testing. This data-driven approach uses few *a priori* assumptions of tree architecture, and transferability across lidar instruments is constrained only by data quality requirements.
3. We demonstrate the *treeseg* algorithm here on data acquired from both a structurally simple open forest and a complex tropical forest. Across these data, we successfully automatically extract 96% and 70% of trees, respectively, with the remainder requiring some straightforward manual segmentation.
4. *treeseg* allows ready and quick access to tree-scale information contained in lidar point clouds. *treeseg* should help contribute to more wide-scale uptake of lidar-derived methods to applications ranging from the estimation of carbon stocks through to descriptions of plant form and function.

KEYWORDS
extraction, forests, laser scanning, lidar, point cloud, segmentation, trees

1 | INTRODUCTION

Advances in high-precision laser scanning have led to its application in a wide and growing range of fields across the environmental sciences. Current off-the-shelf terrestrial and UAV-mounted lidar instruments are now routinely used to capture high-density, millimeter accurate 3D point clouds in forest scenes. Applications range from retrieval of traditional parameters of forest structure such as stem diameter and tree height (Maas, Bienert, Scheller & Keane, 2008), through to quantitative descriptions of plant material distribution and leaf area index (Jupp et al., 2009).

Recently, new shape-fitting methods have been proposed to utilise such point clouds to construct models that describe the 3D woody structure of individual trees (Raumonen et al., 2013). So far, these so-called quantitative structure models (QSMs) have primarily been used for the estimation of above-ground biomass and carbon stocks via volume estimation (Calders et al., 2015; Gonzalez de Tanago Menaca et al., 2018). Other, new applications of QSMs include species identification (Åkerblom, Raumonen, Mäkipää & Kaasalainen, 2017), calibration and validation of remote sensing instrumentation (Armston et al., 2016), development of allometric

models (Burt, 2017), and empirical testing of plant form and function hypotheses (Lau Sarmiento et al., 2015).

Limiting the adoption of these new lidar-derived methods is the difficulty associated with accessing the information content of point clouds. To retrieve tree-scale metrics from these data, individual tree-level point clouds must be extracted from the larger-area point cloud. This generally involves laborious and time-consuming manual segmentation, a task that is impracticable for many trees or plots. The more complex the scene, the more difficult and time-consuming this becomes.

Several methods have looked to automate this process using a variety of techniques; the approaches of Raumonen et al. (2015) and Trochta, Krůček, Vrška and Král (2017) organised the larger-area point clouds into clusters, from which tree-level point clouds were grown through fixed inter-cluster assumptions of distance and orientation to infer connectivity. A variant of this approach by Zhong et al. (2017) employed graph theory for this organisation, while the methods of Tao et al. (2015) introduced ecological theory via metabolic scaling theory (West, Brown & Enquist, 1997), to determine belonging on a point-by-point basis. Limiting these methods were their exclusive application to TLS data from structurally simple and sparse forest types, with large distances and minimal interaction between crowns, such that transferability of these methods and their underlying assumptions of tree architecture and/or data quality to more complex forest types, is likely difficult.

Here, we present *treeseg*, an open-source software for the near-automatic extraction of tree-level point clouds from larger-area point clouds. The method has been designed around the principles of being both independent of forest type and instrument, and is demonstrated here through application to lidar data acquired from both simple open forest and complex tropical forest. The method is driven by the data themselves, free from fixed assumptions of tree architecture and data quality, and unique in its attempt to organise the larger-area point cloud into each separate underlying surface comprising the scene.

2 | OVERVIEW OF *TREESEG*

The *treeseg* method has been developed to near-automatically extract tree-level point clouds from larger-area point clouds. The source code, released under the MIT license, is hosted at https://github.com/apburt/treeseg/. The method is implemented in C++, and makes extensive use of the Point Cloud Library (Rusu & Cousins, 2011). Figure 1 illustrates *treeseg*, where it has been applied to a small section of TLS data acquired in tropical forest. Broadly, the method consists of: (i) identification of individual stems, (ii) segmentation of each stem up to first branching, and (iii) isolation of each crown from the canopy. Here, first, we outline the generic point cloud processing techniques underpinning *treeseg*; and the subsequent sections then describe these three main steps of the method.

These steps are illustrated using TLS data collected from 1 ha of tropical forest (moist, Terra Firma, lowland, mixed species, old-growth) in Nouragues Nature Reserve, French Guiana (designation: NOU-11, loc: 4.08°N 52.68°W); and from 0.25 ha of *Eucalyptus* spp. open forest in Karawatha Forest Park, Australia (designation: KARA-001, loc: 27.32°S 153.07°E) (Armston, 2013). These two plots, respectively, comprise 425 and 40 stems with DBH < 0.1 m, basal areas of 33.13 m^2/ha and 10.43 m^2/ha, and approximate canopy heights of 45 and 25 m. Lidar data were collected from both plots using a RIEGL VZ-400 from 121 and 5 scan positions, providing larger-area point clouds containing *c*. 4 billion and *c*. 50 million points respectively.

FIGURE 1 Using *treeseg* to extract a tree-level point cloud (red) from a larger-area point cloud (black) (35 × 25 × 47 m L × W × H) acquired from terrestrial laser scanning in tropical forest in Nouragues Nature Reserve, French Guiana

3 | POINT CLOUD PROCESSING

This section describes the generic point cloud processing techniques used in the *treeseg* method.

3.1 | Nearest neighbour distance

The Euclidean distance, d, between two points, p_1 and p_2, denoted in the usual Cartesian coordinate system, is defined as:

$$d(p_1,p_2)=\sqrt{(x_{p_1}-x_{p_2})^2+(y_{p_1}-y_{p_2})^2+(z_{p_1}-z_{p_2})^2} \quad (1)$$

The distance between the point p_1 and its nearest neighbour, $d_{NN}(p_1)$, in the point cloud, P, $p_i \in P$, is defined as:

$$d_{NN}(p_1)=\min_{\{p_i \in P : p_i \neq p_1\}} d(p_1,p_i) \quad (2)$$

Across P, the mean nearest neighbour distance, $\overline{d_{NN}}$, is defined as:

$$\overline{d_{NN}} = \frac{1}{N(P)} \sum_{i=1}^{N(P)} d_{NN}(p_i) \quad (3)$$

Further, this can be vertically resolved, $\overline{d_{NN}}(z)$, using bins of width, dz, as:

$$\overline{d_{NN}}(z) = \frac{1}{N(P)} \sum_{i=1}^{N(P)} d_{NN}(p_i) \Big| \{p_i : z < z_{p_i} < z + dz\} \quad (4)$$

3.2 | Downsampling

Point cloud downsampling can manage the distribution of $\overline{d_{NN}}(z)$ across P, or reduce computational complexity. This is typically undertaken by partitioning the bounding volume of P into voxels with an edge-length of l_V. The points inside each voxel, $p_i \in V$, are aggregated into a single point, p_c.

3.3 | Euclidean clustering

Clustering is a tool used to order an unorganised point cloud (e.g., P) into an organised set of point clouds, $\{C\}$, $\{C\} \subset P$, based on Euclidean distances. The cluster C is formed when enough constituent points ($\geq N_{min}$) have an intra-d_{NN} less than or equal to d_{max}. $p_i \in C$ are then segmented from P and the set is added to $\{C\}$, and iterated until remaining $p_i \in P$ are exhausted.

3.4 | Principal component analysis

Principal component analysis is used to describe the arrangement of points in a point cloud (e.g., $p_i \in P$) through eigen decomposition:

$$(\Sigma - \lambda I)v = 0 \quad (5)$$

where, λ and v are the eigenvectors and eigenvalues respectively; Σ, the covariance matrix, is defined as:

$$\Sigma = \begin{bmatrix} \mathrm{var}(x,x) & \mathrm{cov}(x,y) & \mathrm{cov}(x,z) \\ \mathrm{cov}(y,x) & \mathrm{var}(y,y) & \mathrm{cov}(y,z) \\ \mathrm{cov}(z,x) & \mathrm{cov}(z,y) & \mathrm{var}(z,z) \end{bmatrix} \quad (6)$$

where, for example, cov(x,y), is defined as:

$$\mathrm{cov}(x,y) = \sum_{i=1}^{N(P)} \frac{(x_{p_i} - x_{p_c})(y_{p_i} - y_{p_c})}{N(P) - 1} \quad (7)$$

Obtained from the resultant Σ, the cubic polynomial can be solved for λ through factorisation, from which v can be solved from each respective resultant linear equation. The principal axes of P are then described by λ (eigenvectors), alongside their associated magnitudes v (eigenvalues).

3.5 | Surface normals

One extension to PCA is surface normal estimation. Across an organised point cloud, C, representing a common underlying surface such as a leaf, the normal to this surface is the vector perpendicular to a plane fitted through C. By definition, λ_1 across C, derived via PCA, is the vector of this plane. As each eigenvector is orthogonal to the next, λ_3 defines the normal of this plane (i.e., an estimate of the surface normal). Across unorganised point clouds, each point is attributed a surface normal estimate by fitting such a plane to either the nearest N neighbours, or those neighbours whose distance is less than d_{max}.

3.6 | Region-based segmentation

A further extension to PCA and surface normal estimation is region growing segmentation. This partitions P into a set of regional point clouds, $\{R\}$, $\{R\} \subset P$, based on neighbourhood point commonality, such that it is inferred $p_i \in R$ share some common underlying surface. Here, the metric used to determine commonality between points is the angle between surface normals:

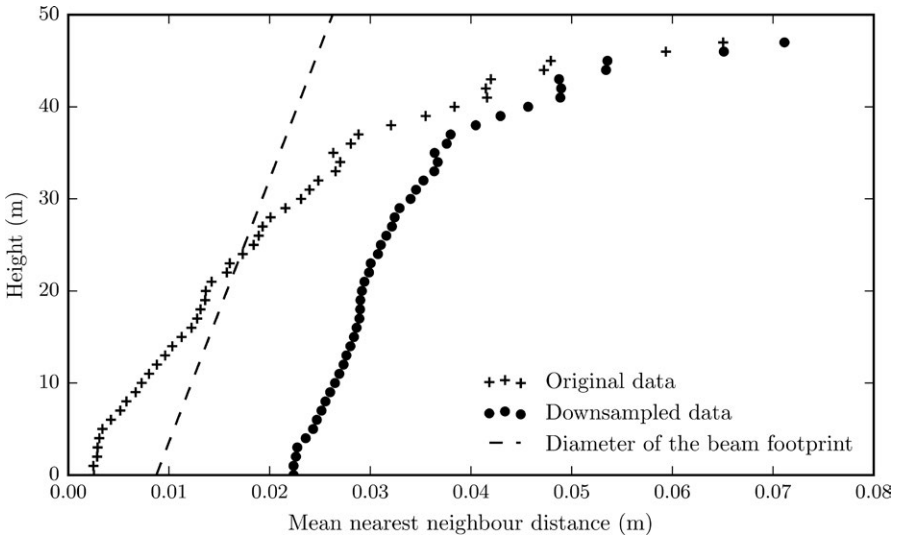

FIGURE 2 Vertically resolved nearest neighbour distance across the NOU-11 point cloud both pre- and post-downsampling. The dashed line represents the assumed pulse footprint at the respective height (i.e., assuming no lateral travel)

$$\theta(A,B) = \arccos\left(\frac{A \cdot B}{|A||B|}\right) \quad (8)$$

where A and B represent λ_3 of two arbitrary points. Each region cloud, R, is grown from a random seed point in P by incorporating neighbours provided the angle between surface normals is below some threshold, θ_{max}, until exhaustion of the neighbourhood. R is then segmented from P and added to $\{R\}$, and the process is iterated until all seeds are considered.

3.7 | Shape fitting

The final method used in *treeseg* is shape fitting via random consensus model fitting (RANSAC). The only geometric primitive considered here is a cylinder, defined by: a centreline with the vector, v, a point on the centreline, p_0, and a radius, r. If R is an organised point cloud whose underlying surface belongs to a cylindrical section of stem, v can be equated to λ_1. That is, a plane described as:

$$\lambda_{(1,x)}x + \lambda_{(1,y)}y + \lambda_{(1,z)}z + -\lambda_{(1,x)}x_{p_c} - \lambda_{(1,y)}y_{p_c} - \lambda_{(1,z)}z_{p_c} = 0 \quad (9)$$

where p_0 can be described as the centroid point, p_c, of R. Subsequently, r is defined as the mean distance between $p \in R$ and the plane. This shortest distance between some arbitrarily selected point, p_1, and the plane, d, is defined as:

$$d = \frac{|\lambda_{(1,x)}x_{p_1} + \lambda_{(1,y)}y_{p_1} + \lambda_{(1,z)}z_{p_1} - \lambda_{(1,x)}x_{p_c} - \lambda_{(1,y)}y_{p_c} - \lambda_{(1,z)}z_{p_c}|}{\sqrt{\lambda_{(1,x)}^2 + \lambda_{(1,y)}^2 + \lambda_{(1,z)}^2}} \quad (10)$$

For the cylinder to be finite, then the length, l, must be characterised. To determine l from R using Euclidean distance, it is necessary to transform λ_1 such that the centreline z-axis will be rotated into (0,0,1):

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (11)$$

where θ, the angle between λ_1 and (0,0,1) is calculated through Equation 8. This reduces the estimate of l down to:

$$l = p_{z_{max}} - p_{z_{min}} \quad (12)$$

This cylinder fitting can then be applied in a RANSAC framework, a brute force approach permitting the removal of outliers from noisy data.

4 | THE LARGER-AREA POINT CLOUD

treeseg ingests a point cloud (with each point attributed an x, y, z-coordinate) such as the one illustrated in Figure 1. The two expectations of these data are: first, the point cloud provides a contiguous sample of the scene from which structural elements of individual trees, down to the highest-order branching, are clearly discernible. That is, the sampling protocol employed to acquire the larger-area point should minimise occlusion effects, with no significant gaps

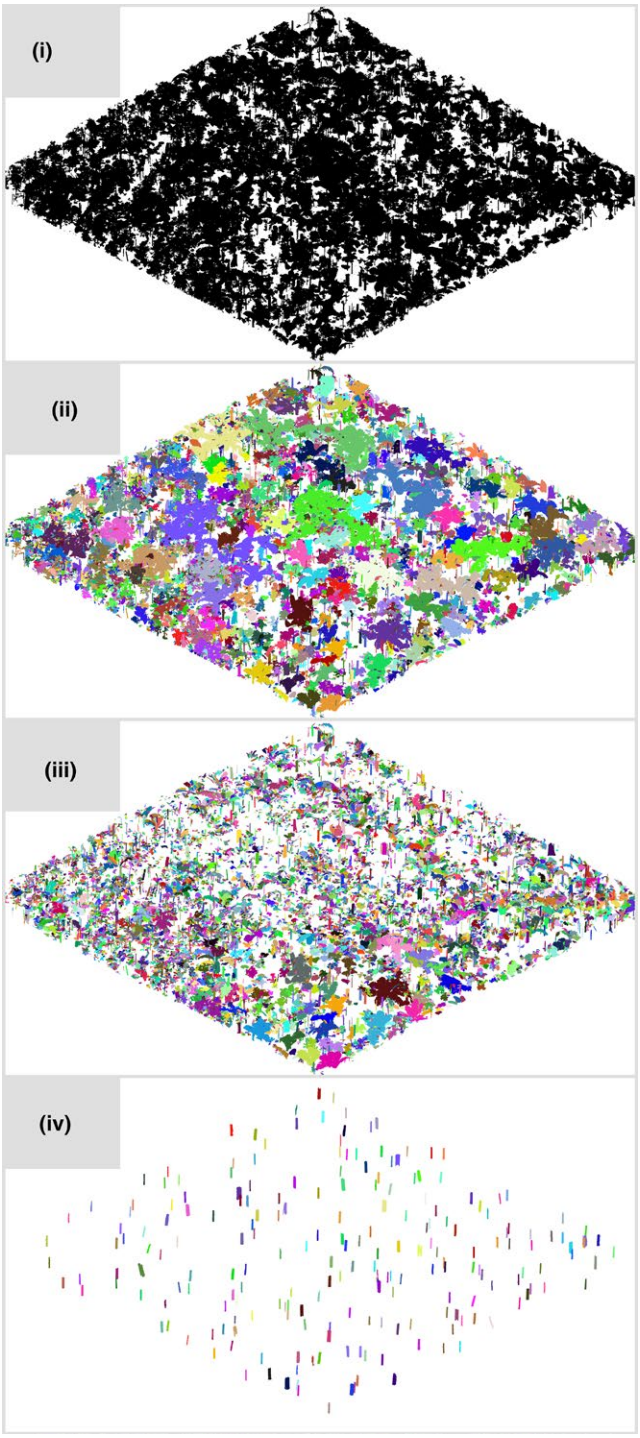

FIGURE 3 Identifying stems in the NOU-11 larger-area point cloud: (i) a slice in the z-axis is segmented from the plot-level point cloud, as driven by the underlying DTM, (ii) the slice is organised via Euclidean clustering, (iii) each of these clusters are further organised into their underlying surfaces via region-based segmentation, and (iv) stems are identified from each region through RANSAC cylinder fitting

present in the data. Second, for data requiring co-registration (e.g., multiple scans from various locations using a tripod-mounted TLS), error in this registration should be similar to the ranging accuracy of the instrument.

Due to beam divergence, increasing angular distance between sequentially fired pulses at a given range, and general occlusion, there is likely to be strong variation in $\overline{d_{NN}}(z)$ across the larger-area point cloud. Because many of the aforementioned point cloud processing techniques are ill-suited to such variations, downsampling via voxel grid aggregate downsampling can somewhat homogenise the larger-area point cloud. Figure 2 illustrates $\overline{d_{NN}}(z)$ across the NOU-11 larger-area point cloud, both pre- and post-downsampling, where $l_v = 0.04$ m. Noticeably, between a height of 0–30 m, $\frac{d(\overline{d_{NN}}(z))}{dz}$ is steeper in the downsampled data, while the range of $\overline{d_{NN}}(z)$ values across the height has reduced by an order of magnitude.

5 | TREE IDENTIFICATION

The initial stage of *treeseg* is to identify individual trees from the larger-area point cloud, *P*. Illustrated in Figure 3, this stage comprises:

1. The generation of a digital terrain model (DTM) across *P*, from which a point cloud slice, *S*, in the z-axis, is generated from *P* (Figure 3i).
2. Euclidean clustering is applied across *S* to order the slice into the set of point clouds, $\{C\}$, $\{C\} \subset S$ (Figure 3ii).
3. For $C \in \{C\}$, region-based segmentation is used to order $\{C\}$ into the set of point clouds $\{R\}$, $\{R\} \subset \{C\}$, based on common underlying surfaces (Figure 3iii).
4. Finally, for $R \in \{R\}$, RANSsAC cylinder fitting is applied to segment cylinders. These cylinders are assumed to be stems based on: (i) the residual error of the fitting and (ii) the angle between the vector of the cylinder centreline and the vector perpendicular to underlying DTM tile plane (Figure 3iv).

6 | STEM SEGMENTATION

The second stage of *treeseg* is, for each identified tree, to extract the stem up to the position of first branching. The approach, consisting of the four following steps, is illustrated in Figure 4:

1. The cylinder belonging to each identified tree acts as a pass-through filter across the larger-area point cloud to extract the point cloud, *V*, $V \subset P$, containing: the stem of the tree in question, neighbouring vegetation, and returns from the ground (Figure 4i).
2. A plane is fitted through the lower section of *V* via the RANSAC framework, with the inliers, considered to represent the ground, segmented from *V* (Figure 4ii).
3. Euclidean clustering and region-based segmentation is applied across *V* to order the cloud into its underlying surfaces, from which neighbouring vegetation is removed (Figure 4,iii,iv,v).

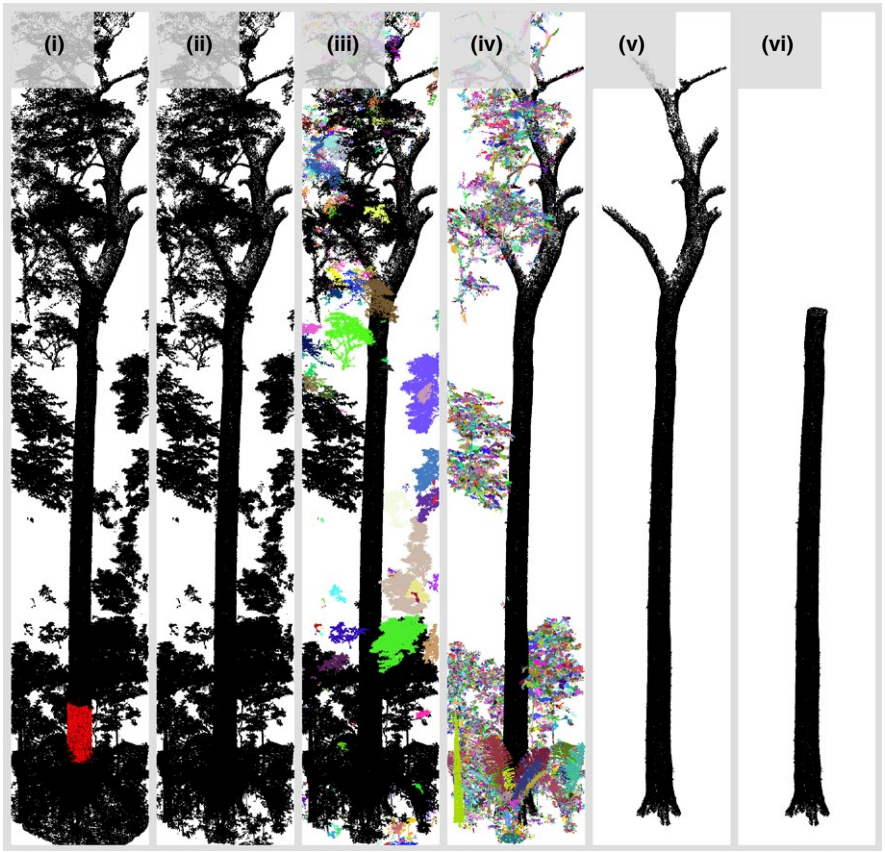

FIGURE 4 Segmenting the stem: (i) the cylinder fit for each identified tree (red) acts as a pass-through filter (black), (ii) points from the ground are removed through RANSAC plane fitting, (iii), (iv), (v) Euclidean clustering and region-based segmentation are used to remove neighbouring vegetation, and (vi) RANSAC cylinder fitting is used to determine the location of first branching

4. From the base of the stem, variation in the goodness of RANSAC cylinder fits through the stem are assessed to determine the position of first branching (Figure 4vi).

7 | CROWN SEGMENTATION

The final stage of *treeseg* is to extract the crown associated with each segmented stem, as illustrated in Figure 5.

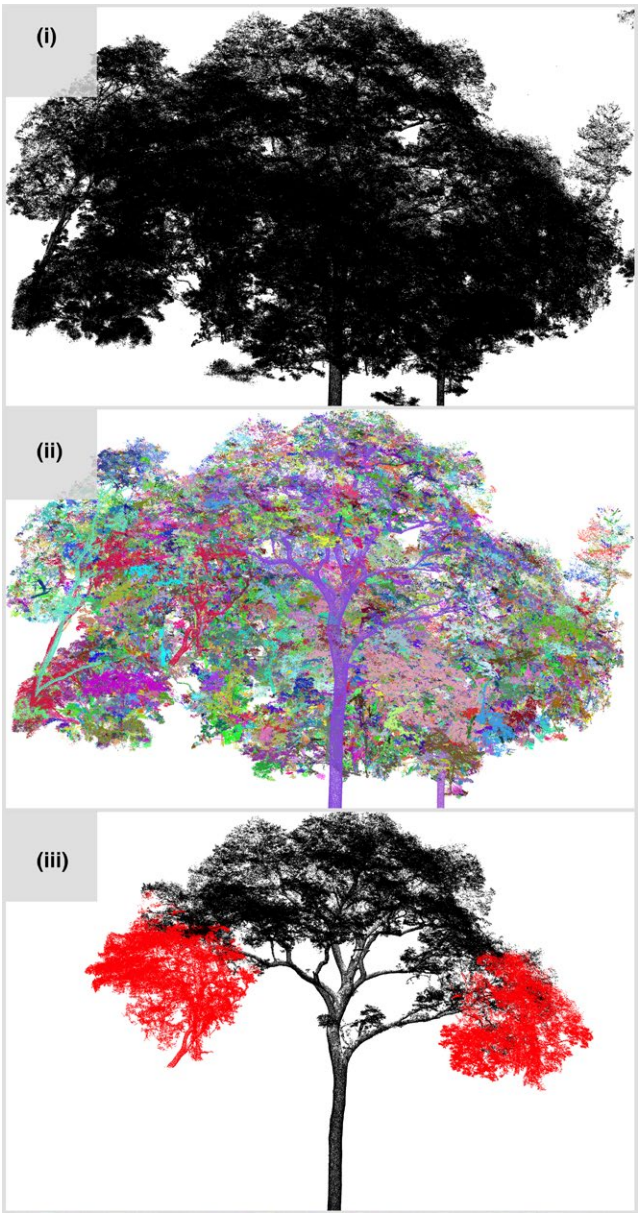

FIGURE 5 Segmenting the crown from the canopy: (i) a section of point cloud containing the crown is extracted from the larger-area point cloud using allometric relationships, (ii) this volume is organised into underlying surfaces via region-based segmentation, and (iii) the crown is grown via connectivity testing driven by $\overline{d_{NN}}(z)$ (black); occasionally manual segmentation may be required to remove neighbouring vegetation (red)

1. The volume of canopy containing the crown of each stem is determined using allometric relationships relating stem diameter to tree height and crown extent. These dimensions, further extended by a user-modifiable fixed distance to ensure complete capture of the crown, are used to segment the point cloud, V, $V \subset P$ (Figure 5i).
2. Region-based segmentation is used to order V into the set of point clouds $\{R\}$, $\{R\} \subset V$, based on common underlying surfaces (Figure 5ii).
3. Commencing from the isolated stem, the crown is grown through appendage of R_i via connectivity testing, provided that: (i) the distance between each R is approaching $\overline{d_{NN}}(z)$ and (ii) that R is smaller than its parent (Figure 5iii); where the length of R is determined through transformation about the principal eigenvector.
4. Finally, due to the conservative nature of 3) tending towards commission error, if necessary, manual segmentation is used to remove any erroneously segmented neighbouring vegetation (Figure 5iii). This can be undertaken using software such as CloudCompare (2018).

8 | CONCLUSIONS

In this applications paper, we have presented a method for the near-automatic extraction of tree-level point clouds from larger-area point clouds. We have applied the method to the KARA-001 and NOU-11 downsampled point clouds of 17 million points (197 MB) and 338 million points (3:8 GB), to segment 28 and 155 trees with DBH greater than 0:2 m (this is an arbitrary user-defined threshold, and not a limitation of the method). Automatic segmentation took 2 days and 1 week respectively using a 24-core 2.40 GHz Intel Xeon E5-2620v3 (15 MB L3 cache) node with 72 GB DDR4 RAM and mechanical hard drive. In its current form, each of the three steps of

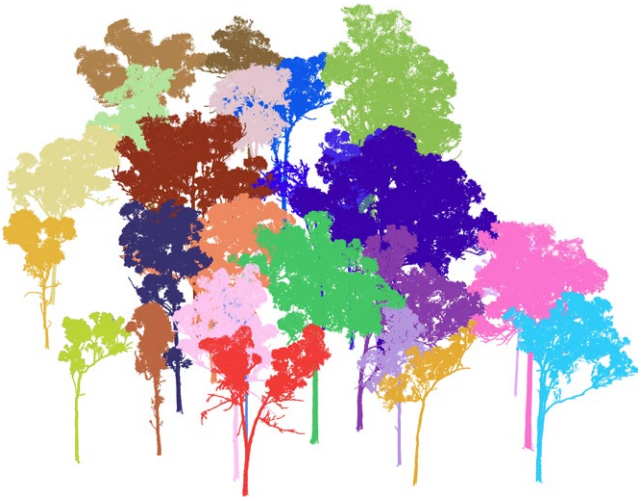

FIGURE 6 The 28 tree-level point clouds extracted from the KARA-001 (*Eucalyptus* spp. open forest) larger-area point cloud via *treeseg*

FIGURE 7 The 155 tree-level point clouds extracted from the NOU-11 (moist, Terra Firma, lowland, mixed-species, old-growth tropical forest) larger-area point cloud via *treeseg*

treeseg are single-threaded and parallelisation would reduce these timings.

Of the trees inside tropical forest plot NOU-11, approximately 30% required further manual segmentation, taking two days using the segment feature of *CloudCompare* (2018). One tree inside open forest plot KARA-001 also required manual segmentation, representing a 96% success rate. With our data-driven approach free from fixed assumptions of tree architecture and data quality, our conservative connectivity testing ensured this manual segmentation was straightforward and intuitive as it required only removal of small sections of neighbouring vegetation.

Figures 6 and 7 present both sets of extracted tree-level point clouds. *treeseg* provides a significant advance in accessing tree-level point clouds in a timely and consistent way from larger-area point clouds, and can be used with any high density lidar point cloud providing a contiguous (gapless) sample of the scene, whether from UAV or TLS.

ACKNOWLEDGEMENTS

A.B. was supported by NERC CASE PhD NE/J016926/1 with EADS Astrium; additional funding was via the NERC National Centre for Earth Observation (NCEO), and NERC funding awards NE/N00373X/1, NE/P011780/1 and NE/K002554/1; M.D. is also supported by the EU Horizon2020 project (BACI project funded by the EU's Horizon 2020 Research and Innovation Programme under grant agreement 640176). K.C. is funded by BELSPO (Belgian Science Policy Office) in the frame of the STEREO III programme – project 3D-FOREST (SR/02/355).

TLS data collection in Nouragues Nature Reserve, French Guiana (NOU-11) were supported via funding from the CNRS Nouragues Travel Grants Program, and we gratefully acknowledge Jéróme Chave and Blaise Tymen for help with this.

AUTHORS' CONTRIBUTIONS

All authors conceived and designed the methods. A.B. wrote the software and manuscript, with contributions from M.D. and K.C.

DATA ACCESSIBILITY

treeseg is available at https://github.com/apburt/treeseg. The version described in this paper, v0.1.0, is archived at https://doi.org/10.5281/zenodo.1471621. The TLS data from Nouragues Nature Reserve (NOU-11) are available from the NERC CEDA archive via http://dx.doi.org/10.5285/33592524dd9a4b2f897edbddb69b3381. The TLS data from Karawatha Forest Park (KARA-001) are available from TERN AusCover via the persistent URL http://www.auscover.org.au/purl/tls-riegl-qa-all-sites.

ORCID

Andrew Burt https://orcid.org/0000-0002-4209-8101
Mathias Disney http://orcid.org/0000-0002-2407-4026
Kim Calders https://orcid.org/0000-0002-4562-2538

REFERENCES

Åkerblom, M., Raumonen, P., Mäkipää, R., & Kaasalainen, M. (2017). Automatic tree species recognition with quantitative structure models. *Remote Sensing of Environment*, *191*, 1–12. https://doi.org/10.1016/j.rse.2016.12.002

Armston, J. (2013). *Terrestrial laser scans - Riegl VZ400, raw instrument files and ancillary data, Australian field sites*. Version 1.0. Remote Sensing Centre, Queensland Department of Environment and Science. Retrieved from http://www.auscover.org.au/purl/tls-riegl-qa-all-sites, made available by the AusCover facility (http://www.auscover.org.au) of the Terrestrial Ecosystem Research Network (TERN, http://www.tern.org.au).

Armston, J., Tang, H., Hancock, S., Hofton, M. A., Dubayah, R., Duncanson, L., … Disney, M. (2016). Integration of ALS and TLS for calibration and validation of LAI profiles from large footprint lidar. In: American Geophysical Union, Fall General Assembly 2016, San Francisco, USA, December 12-16. Retrieved from http://adsabs.harvard.edu/abs/2016AGUFM.B54B.04A

Burt, A. (2017). New 3D measurements of forest structure. PhD thesis. University College London. Retrieved from: http://discovery.ucl.ac.uk/id/eprint/1575534

Calders, K., Newnham, G., Burt, A., Murphy, S., Raumonen, P., Herold, M., … Kaasalainen, M. (2015). Nondestructive

estimates of above-ground biomass using terrestrial laser scanning. *Methods in Ecology and Evolution*, *6*(2), 198–208. https://doi.org/10.1111/2041-210X.12301

CloudCompare. (2018). 3D point cloud and mesh processing software, Version 2.9.1. http://cloudcompare.org

Gonzalez de Tanago Menaca, J., Lau, A., Bartholomeusm, H., Herold, M., Avitabile, V.Raumonen, P., … Calders, K. (2018). Estimation of above-ground biomass of large tropical trees with terrestrial LiDAR. *Methods in Ecology and Evolution*, *9*(2), 223–234. https://doi.org/10.1111/2041-210X.12904

Jupp, D. L. B., Culvenor, D. S., Lovell, J. L., Newnham, G. J., Strahler, A. H.. (2009). Estimating forest LAI profiles and structural parameters using a ground-based laser called Echidna. *Tree Physiology*, *29*, 171–181. https://doi.org/10.1093/treephys/tblpn022

Lau Sarmiento, A. I., Bartholomeus, H., Herold, M., Martius, C., Malhi, Y., Bentley, L.P., … Raumonen, P. (2015). Application of terrestrial LiDAR and modelling of tree branching structure for plant-scaling models in tropical forest trees. In: Retrieved from https://silvilaser2015.teledetection.fr/files/Proceedings_Silvilaser_22_09_2015_2.pdf

Maas, H.-G., Bienert, A., Scheller, S., & Keane, E. (2008). Automatic forest inventory parameter determination from terrestrial laser scanner data. *International Journal of Remote Sensing*, *29*(5), 1579–1593. https://doi.org/10.1080/01431160701736406

Raumonen, P., Casella, E., Calders, K., Murphy, S., Åkerblom, M., Kaasalainen, M. (2015). Massive-scale tree modelling from TLS data. In: International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, II-3/W4, Munich, Germany, March 25-27. https://doi.org/10.5194/isprsannals-II-3-W4-189-2015.

Raumonen, P., Kaasalainen, M., Åkerblom, M., Kaasalainen, S., Kaartinen, H., Vastaranta, M., … Lewis, P. (2013). Fast automatic precision tree models from terrestrial laser scanner data. *Remote Sensing*, *5*(2), 491–520. https://doi.org/10.3390/rs5020491

Rusu, R. B., & Cousins, S. (2011). 3D is here: Point Cloud Library (PCL). In: IEEE International Conference on Robotics and Automation, Shanghai, China, May 9–13. https://doi.org/10.1109/ICRA.2011.5980567.

Tao, S., Wu, F., Guo, Q., Wang, Y., Li, W.Xue, B., … Fang, J. (2015). Segmenting tree crowns from terrestrial and mobile LiDAR data by exploring ecological theories. *ISPRS Journal of Photogrammetry and Remote Sensing*, *110*, 66–76. https://doi.org/10.1016/j.iisprsjprs.2015.10.007

Trochta, J., Krůček, M., Vrška, T., & Král, K. (2017). 3D Forest: An application for descriptions of three-dimensional forest structures using terrestrial LiDAR. *PLOS ONE*, *12*(5), e0176871. https://doi.org/10.1371/journal.pone.0176871

West, G. B., Brown, J. H., & Enquist, B. J. (1997). A general model for the origin of allometric scaling laws in biology. *Science*, *276*(5309), 122–126. https://doi.org/10.1126/science.276.5309.122

Zhong, L., Cheng, L., Xu, H., Wu, Y., Chen, Y., Li, M. (2017). Segmentation of individual trees from TLS and MLS data. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *10*(2), 774–787. https://doi.org/10.1109/JSTARS.2016.2565519

How to cite this article: Burt A, Disney M, Calders K. Extracting individual trees from lidar point clouds using *treeseg*. *Methods Ecol Evol*. 2018;00:1–8. https://doi.org/10.1111/2041-210X.13121