Environmental Sciences Proceedings
MDPI

Proceedings

FORTLS: an R Package for Processing TLS data and Estimating Stand Variables in Forest Inventories

Juan Alberto Molina-Valero[1], Maria José Ginzo Villamayor[2], Manuel Antonio Novo Pérez[3], Juan Gabriel Álvarez-González[1], Fernando Montes[4], Adela Martínez-Calvo[1] and César Pérez Cruzado[5,]*

[1] Unidad de Gestión Ambiental y Forestal Sostenible (UXAFORES), Departamento de Ingeniería Agroforestal, Escuela Politécnica Superior de Ingeniería, Universidade de Santiago de Compostela, Benigno Ledo s/n, Campus Terra, 27002, Lugo, Spain; juanalberto.molina.valero@usc.es; juangabriel.alvarez@usc.es; adela.martinez.calvo@usc.es
[2] Departamento de Estadística, Análisis Matemático y Optimización, Universidade de Santiago de Compostela, Facultad de Matemáticas, Rúa Lope Gómez de Marzoa s/n. Campus Vida. Santiago de Compostela, Spain. C.P. 15782; mariajose.ginzo@usc.es
[3] Instituto Tecnológico de Matemática Industrial (ITMATI), Edificio Instituto Investigaciones Tecnológicas, planta-1. Rúa de Constantino Candeira s/n. Campus Vida. Santiago de Compostela, Spain. C.P.15782; manuelantonio.novo.perez@usc.es
[4] INIA-Centro de Investigación Forestal, Ctra. De la Coruña km 7,5, 28040 Madrid; fmontes@inia.es
[5] Unidad de Gestión Ambiental y Forestal Sostenible (UXAFORES), Departamento de Producción Vegetal y Proyectos de Ingeniería, Escuela Politécnica Superior de Ingeniería, Universidade de Santiago de Compostela, Benigno Ledo s/n, Campus Terra, 27002, Lugo, Spain; cesar.cruzado@usc.es
* Correspondence:Correspondence: cesar.cruzado@usc.es

Published: 25 October 2020

Abstract: Terrestrial Laser Scanning (TLS) enables rapid, automatic and detailed 3D representation of surfaces with an easily handled scanner device. TLS therefore shows great potential for use in Forest Inventories (FIs). However, the lack of well established algorithms for TLS data processing hampers operational use of the scanner for FI purposes. Here we present FORTLS, an R package specifically developed to automate TLS point cloud data processing for forestry purposes. The FORTLS package enables (i) detection of trees and estimation of their diameter at breast height (*dbh*), (ii) estimation of some stand variables (e.g. density, basal area, mean and dominant height), (iii) computation of metrics related to important tree attributes estimated in FIs at stand level and (iv) optimization of plot design for combining TLS data and field measured data. FORTLS can be used with single-scan TLS data, thus improving data acquisition and shortening the processing time, as well as increasing sample size in a cost-efficient manner. The package also includes several features for correcting occlusion problems in order to produce improved estimates of stand variables. These features of the FORTLS package will enable the operational use of TLS in FIs, in combination with inference techniques derived from model-based and model-assisted approaches.

Keywords: Forest inventory; LiDAR; remote sensing; R-package; software; stand-level; TLS.

1. Introduction

Information about forest resources is essential for sustainable forest management and development of forest policies. In this regard, forest inventories (FIs) are used as the main approach to estimating and monitoring the state and evolution of the main variables of interest. FIs have improved since they were first introduced, as a result of the continuous appearance of new technologies, such as Terrestrial Laser Scanning (TLS), considered of great potential value for enhancing FIs [1-2]. However, TLS has not been yet adopted in FIs for several reasons [3], although many studies agree that affordability is the main key challenge to overcome, emphasizing that automation of the point cloud processing with attainable and easy-to-use software able to extract information related to important forest attributes is essential [1-5].

As TLS data sets comprise millions of points, sophisticated methods for automatic processing are necessary. In this respect, many algorithms with a high level of automation and that are able to extract tree attributes (diameter at breast height, *dbh*, height, volume, etc.) have been developed in the last few decades [6]. Some of these algorithms have also been included in software applications, e.g. SimpleTree [7], 3D Forest [8] and AutoStem™ [9]. However, there are some drawbacks to using these applications in FIs: (i) single-tree instead of stand-level approaches (SimpleTree); (ii) semiautomatic processing (3D Forest); and (iii) commerciality (not suitable for all users) (AutoStem™).

Here we present `FORTLS`, an `R` package developed with the objective of automating TLS point cloud data processing and estimating variables for forestry purposes. `FORTLS` can be used with single-scan TLS data and enables (i) detection of trees and estimation of *dbh*, (ii) estimation of some stand variables, such as density (*N*, trees ha^{-1}), basal area (*G*, m^2 ha^{-1}) and mean and dominant height, defined as the mean height of the 100 largest trees ha^{-1} (h_m and H_0 respectively), (iii) computation of metrics related to important tree attributes estimated in FIs at stand level, and (iv) optimization of plot design for combining TLS data and field measured data. These features of the FORTLS package will enable the operational use of TLS in FIs, in combination with model-based or model-assisted inference approaches.

2. Materials and Methods

The steps involved in the TLS data processing algorithms are described in the following sections.

2.1. Detection of trees and estimation of dbh

This first algorithm detects trees and estimates their *dbh*, which is the basis for further computations. This is done by the `normalize` function (Figure 1), which obtains coordinates relative to the plot centre and the digital terrain model. This function also applies the point cropping process as a criterion for reducing point density homogeneously in space and proportional to object size [10]. The output generated is then used as input for the `tree.detection` function, which detects as many trees as possible from point clouds in the TLS scans. In addition, for every tree detected, the function calculates the coordinates of the section centre, estimates *dbh*, and classifies it as fully visible or partially occluded. Finally, this function obtains the number of points corresponding to 1.3 m height sections of trees (i.e. the *dbh*) for original and reduced point clouds (by applying point cropping process), as well as their estimations.

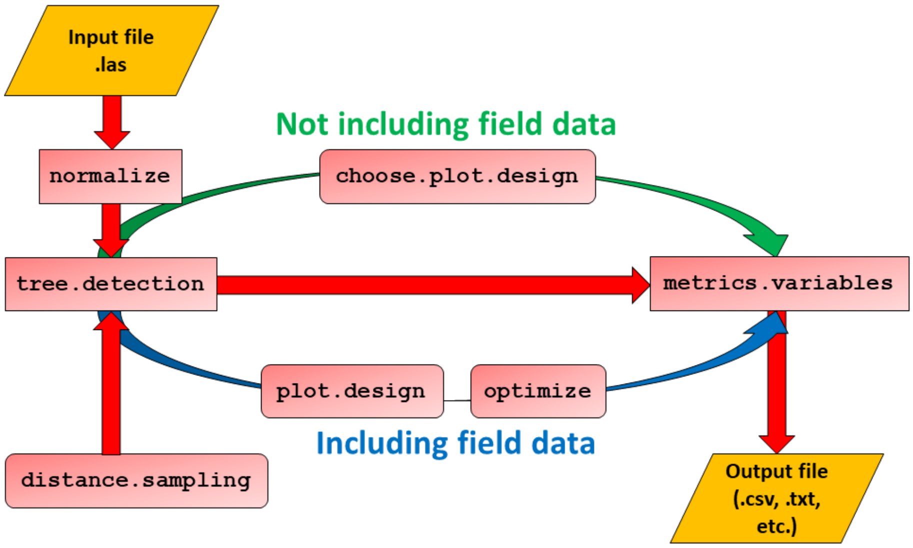

Figure 1. Schematic workflow of `FORTLS`. The pathway shown in red represents the shortest possible procedure for estimating variables and metrics. The pathway shown in green includes `choose.plot.design` as a previous step for assessing the stability of estimations, based only on TLS data. The pathway shown in blue includes `plot.design` and `optimize` functions with the objective of determining the best plot design according to field measured data. `distance.sampling` is an optional function which can be used in both approaches.

2.2. Computation of variables and metrics related to attributes estimated in FIs at stand level

Once trees have been detected, the next application of `FORTLS` is to compute variables and metrics at plot level. For this purpose, the `metrics.variables` function produces a set of TLS-based variables and metrics related to forest attributes. These can be obtained for different plot designs (circular fixed area, k-tree and angle-count) if specified in the arguments. This function also includes features for correcting occlusion problems generated in TLS point clouds. These features are based on correcting the shadowing effect [11] and gap probability attenuation with distance to TLS [12]. Apart from these features, others based on distance sampling methods can be applied with the `distance.sampling` function by implementing point transects methods with the trees detected [13]. This calculates the detection probability for every tree by fitting probability detection functions to the histogram of trees distribution according to their distance from plot centre. As in previous studies by the same authors [13], half normal and hazard rate probability functions without and with *dbh* as covariate were used.

Before using the `metrics.variables` function, previous steps are recommended in order to select the most appropriate values for the radius, k-tree and BAF (Basal Area Factor) in the function arguments. This can be done with or without field data.

2.2.1. Field measurements not available

In this case, the `choose.plot.design` function can be used to plot empirical charts of *N* and *G* estimates as a function of the plot size (estimation-size charts) for different plot designs (circular fixed area, k-tree and angle-count), through continuous size increments (radius, k and BAF respectively). These size-estimation charts represent the consistency in predicting the stand variables across different values of radius, k and BAF. Size-estimation charts can be drawn for individual sample plots (including all plots together in the same charts) or for mean values (global mean computed for all the sample plots, or for group means if different strata are considered). Finally, different plot designs can be compared if specified in the arguments, producing one size-estimation chart per variable (*N* and *G*).

2.2.2. Field measurements available

When field measurements are available for the same positions of TLS single-scans, the `plot.design` and `optimize` functions can be used to assess the performance of TLS-based metrics and variables relative to field measurements. The `plot.design` function examines correlations (Pearson and Spearman) and the relative deviance between TLS-based estimates and field measurements, through plots with continuous size increment. This is done for different plot designs and by default for the most common metrics and variables, although other metrics/variables may be considered in the arguments. In a second step, those metrics and variables most closely correlated with the variables of interest are evaluated by the `optimize` function. This function generates heatmaps (one per plot design) in which correlations between TLS metrics-variables and estimations of variables based on field data can be evaluated for all variables and plot sizes.

3. Results

The outputs of the above-mentioned functions are reported below.

3.1. Detection of trees and estimation of their dbh

The result of these applications is a list of the trees detected with the aforementioned `tree.detection` function, which is a data frame object containing attributes for every tree detected (Table 1).

Table 1. Data frame with detected trees and their estimated attributes.

id	file	tree	x	y	phi	phi.left	phi.right	*dbh*	horizontal.distance
numeric or character	character (id.txt)	numeric (n)	numeric (m)		numeric (rad)			numeric (m)	

num.points	num.points.est	num.points.hom	num.points.hom.est	partial.occlusion
numeric (n)				numeric (0-1)

id: identification assigned to a sample plot and which coincides with the file name. file: file name, consisting of the id and the respective extension (.txt, .csv, etc.). tree: number assigned to every detected tree (1, 2, …, n). x: x coordinate of tree centre relative to plot centre. y: y coordinate of tree centre relative to plot centre. phi: azimuth of tree centre from plot centre. phi.left: azimuth corresponding to left border of tree section detected. phi.right: azimuth corresponding to right border of detected tree section. *dbh*: estimated diameter at breast height. horizontal.distance: horizontal distance from sample plot centre to tree centre. num.points: number of points corresponding to normal tree section (1.3 ± 0.05 m). num.points.est: estimated number of points corresponding to normal tree section (1.3 ± 0.05 m). num.points.hom: number of points corresponding to normal tree section (1.3 ± 0.05 m) after point cropping process. num.points.est: estimated number of points corresponding to normal tree section (1.3 ± 0.05 m) after point cropping process. partial.occlusion: tree fully visible (0) or partial occluded (1). .

3.2. Computation of variables and metrics related to attributes estimated in FIs at stand level

The output of the function `metrics.variables` is a list with three data frames, one per plot design (circular fixed area, k-tree and angle-count plot), containing the following metrics and variables computed from TLS data:

Table 2. Structure of list containing metrics and variables.

	N	***G***	***V***	***dbh_m***	***dbh_0***	**Number of points belonging to normal sections**	**Percentiles**
fix.plot k.tree angle.count	N N.hn[1] N.hr[1] N.hn.cov[1] N.hr.cov[1] N.sh[1] N.corr[2]	G G.hn [1] G.hr [1] G.hn.cov[1] G.hr.cov[1] G.sh[1] G.corr[2]	V V.hn[1] V.hr[1] V.hn.cov[1] V.hr.cov[1] V.sh[1] V.corr[2]	dbh.arit dbh.sqrt dbh.geom dbh.harm	dbh.dom.arit dbh.dom.sqrt dbh.dom.geom dbh.dom.harm	num.points num.points.est num.points.hom num.points.hom.est	P_1, P_5, P_{10}, P_{20}, P_{25}, P_{30}, P_{40}, P_{50}, P_{60}, P_{70}, P_{75}, P_{80}, P_{90}, P_{95}, P_{99}

N: all N variables are direct estimates of *N*, estimated using trees detected in TLS data. They are computed without considering occlusion corrections (N) and by implementing distance sampling methodologies (N.hn, N.hr, N.hn.cov, N.hr.cov), shadowing effect (N.sh) and gap probability attenuation with distance from TLS (N.corr). G: all G variables are direct estimates of *G*, estimated using detected trees from TLS data. They are computed without considering occlusion corrections (G) and by implementing distance sampling methodologies (G.hn, G.hr, G.hn.cov, G.hr.cov), shadowing effect (G.sh) and gap probability attenuation with distance from TLS (G.corr). V: all V variables are direct estimates of *V*, estimated using trees detected in TLS data. They are computed without considering occlusion corrections (V) and by implementing distance sampling methodologies (V.hn, V.hr, V.hn.cov, V.hr.cov), shadowing effect (V.sh) and gap probability attenuation with distance from TLS (V.corr). dbh_m: estimated *dbh* mean for detected trees using arithmetic (dbh.arit), square (dbh.sqrt), geometric (dbh.geom) and harmonic means (dbh.harm). *dbh_0*: estimated dominant *dbh* mean (considering the 100 largest trees ha^{-1}) for trees detected using arithmetic (dbh.dom.arit), square (dbh.dom.sqrt), geometric (dbh.dom.geom) and harmonic means (dbh.dom.harm). Number of points belonging to normal sections: sum of points belonging to normal sections of all trees detected from the original point cloud (num.points) and reduced point cloud, reduced using point cropping process (num.points.hom), and number of points estimated from the original point cloud (num.points.est) and reduced point cloud, reduced using the point cropping process (num.points.hom.est). Percentiles: percentile of z coordinate (m) relative to ground level. .

[1] Variables are just computed for fix area and k-tree plots.

[2] Variables are just computed for angle count plots.

3.2.1. Plot design when field measurements are not available

Figure 2 is an example of `choose.plot.design` output when no arguments are defined. In these graphical representations, it can be observed that estimations of *N* and *G* become approximately stable from a radius of 8 m (circular fixed area plot) and 10 trees (k-tree plot) and at between 1 and 2 for BAF (angle count plot).

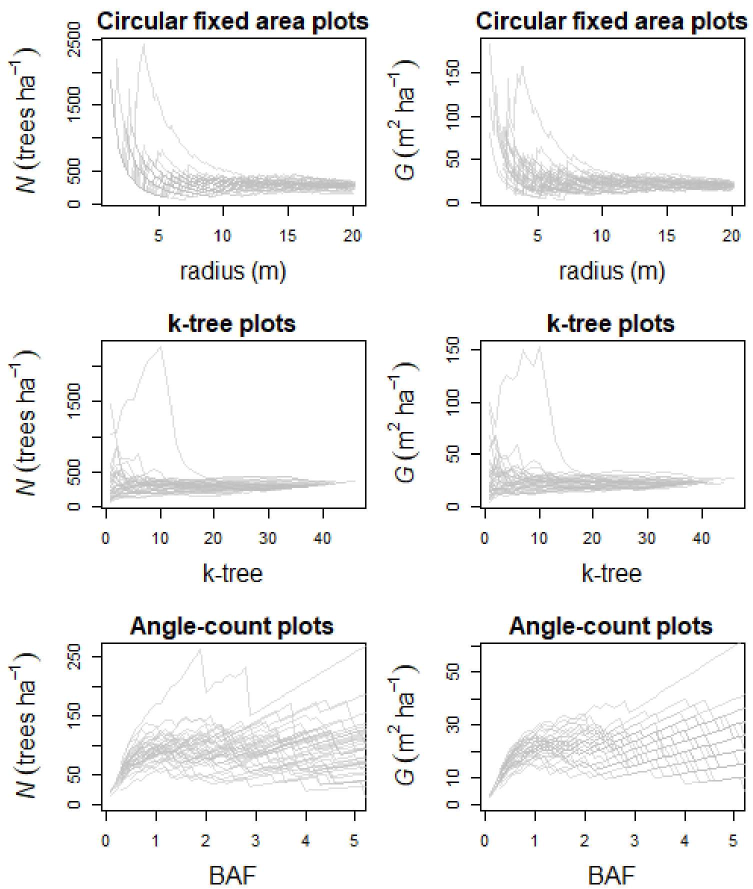

Figure 2. Line charts with estimated values of *N* and *G* for different plot designs (circular fixed area, k-tree and angle-count), through continuous size increments (radius, k and BAF respectively). Each grey line represents a sample plot.

3.2.2. Plot design when field measurements are available

The outputs of the `plot.design` function are line charts showing correlation patterns and relative deviance for TLS derived metrics-variables and estimations of variables based on field data, for different designs and sizes of plots. One interactive chart (html file) per plot design (circular fixed area, k-tree and angle-count plot) and variables of interest (N, G, V, h_m, H_0, dbh_m, dbh_0) (Figure 3), as well as their associated database as (csv file), are saved in the work directory.

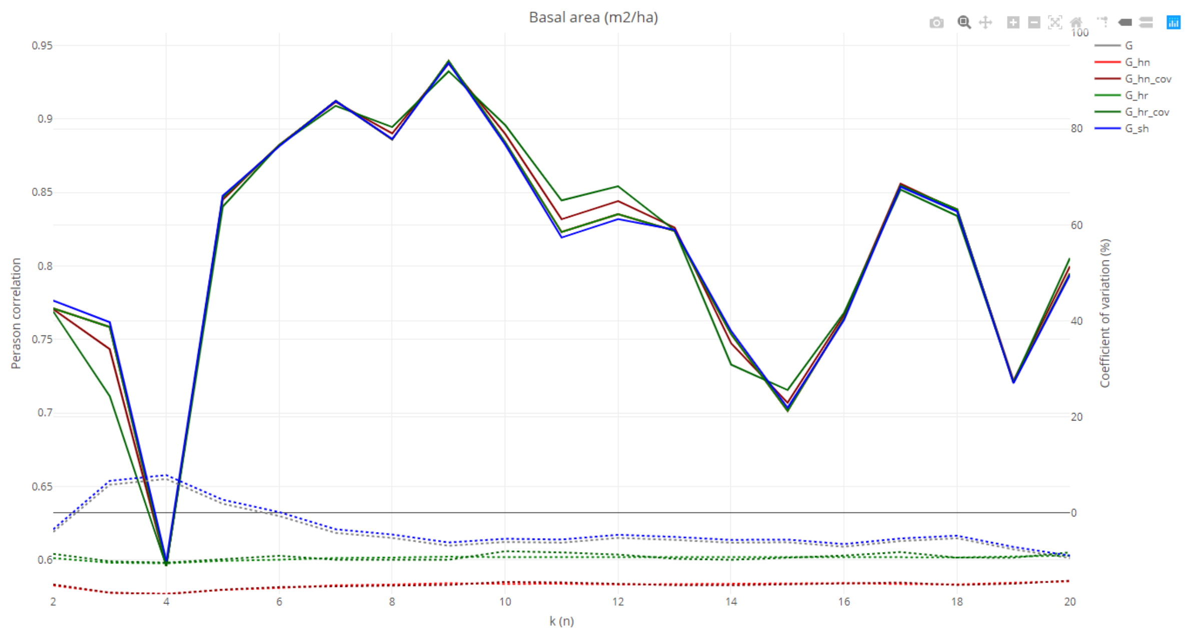

Figure 3. Line chart showing Pearson correlation (continuous line) and relative deviance (dotted line) for basal area estimation based on field data and the TLS derived metrics and variables: G (direct estimates of *G* estimated using trees detected in TLS data), G.hn, G.hr, G.hn.cov, G.hr.cov (considering occlusion corrections based on distance sampling methodologies) and G_sh (considering occlusion corrections based on shadowing effect) through continuous size increment (k) for the k-tree plot design.

Once all TLS metrics and variables have been assessed according to how they are correlated with the variables of interest, the next step is to evaluate them with the `optimize` function. This function generates interactive heatmaps (one per plot design) in which the behaviour of those metrics showing the best correlations across continuous plot size increment can be observed (Figure 4). The color palette gives warm and cold colours to highly positive and negative correlations respectively.

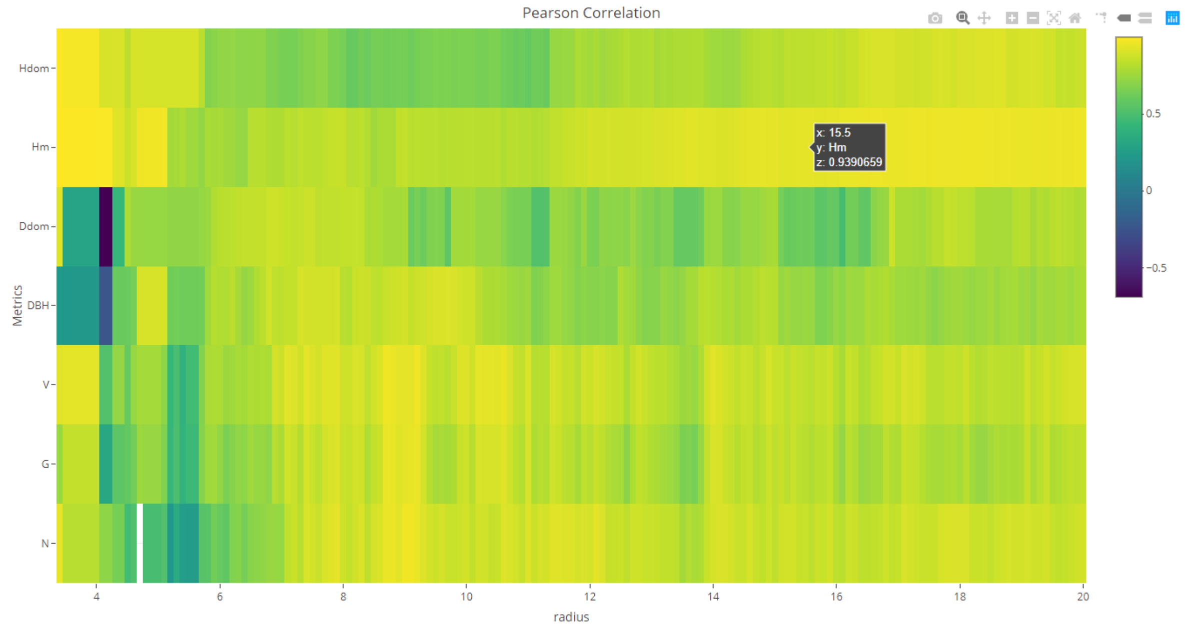

Figure 4. Heatmap showing correlations between variables of interest and TLS variables-metrics.

4. Discussion

FORTLS enables automated processing of TLS point clouds and production of variables and metrics related to relevant information about forest attributes. As some of the functions assess the performance of variable estimations for different plot designs, the application finds the best possible sampling design for any case. This attribute makes FORTLS a flexible application for FIs purposes and valid for several types of forests.

Although FORTLS can be used without including conventional field data, its use is optimal when TLS data and field measured data are combined and assessed with the plot.design and optimize functions. This enables optimization of plot design by assessing correlations between variables of interest (*dbh*, *H*, *G*, etc.) and metrics and variables computed for TLS data, which enables selection of the most appropriate plot designs for each situation. In the best case, those metrics and variables for which low deviations from field measurements are obtained can be used to estimate variables, as in other conventional methods. However, occlusions caused by trees, especially in single-scan data, represent the main problem in this approach [3]. This drawback may be solved with some of the occlusion correction features implemented in this package, as assessed in previous studies [11-13].

The utility of the R package FORTLS for operational use of TLS in FIs has been demonstrated, confirming previous conclusions considered a guideline for further research on TLS in forestry [5]. As FORTLS works with single-scan data, co-registration of point clouds in specific software and placement of targets at field measurements are not required. This improves data acquisition and shortens the processing time, as well as increasing sample size in a cost-efficient manner, which is one of the most desirable features of TLS in FIs [3]. Further research with study cases and considering different metrics that are potentially highly correlated with forest attributes is necessary in order to consolidate this R package.

Funding: This research was funded by Spanish Ministry of Science, Innovation and Universities, AGL2016-76769-C2-2-R. JAMV was funded by the Spanish Ministry of Education through the FPU program (FPU16/03057).

Acknowledgments: The authors thank Mario López Fernández for help with fieldwork.

References

1. Dassot, M.; Constant, T.; Fournier, M. The use of terrestrial LiDAR technology in forest science: application fields, benefits and challenges. *Annals of forest science* **2011**, *68*(5), 959-974.

2. White, J. C.; Coops, N. C.; Wulder, M. A.; Vastaranta, M.; Hilker, T.; Tompalski, P. Remote sensing technologies for enhancing forest inventories: A review. *Canadian Journal of Remote Sensing*, **2016**, *42*(5), 619-641. https://doi.org/10.1080/07038992.2016.1207484

3. Liang, X.; Kankare, V.; Hyyppä, J.; Wang, Y.; Kukko, A.; Haggrén, H.; ... Holopainen, M. Terrestrial laser scanning in forest inventories. *ISPRS Journal of Photogrammetry and Remote Sensing* **2016**, *115*, 63-77. https://doi.org/10.1016/j.isprsjprs.2016.01.006

4. Newnham, G. J.; Armston, J. D.; Calders, K.; Disney, M. I.; Lovell, J. L.; Schaaf, C. B.; ... Danson, F. M. Terrestrial laser scanning for plot-scale forest measurement. *Current Forestry Reports* **2015**, *1*(4), 239-251.

5. Liang, X.; Kukko, A.; Hyyppä, J.; Lehtomäki, M.; Pyörälä, J.; Yu, X.; ... Wang, Y. In-situ measurements from mobile platforms: An emerging approach to address the old challenges associated with forest inventories. *ISPRS Journal of Photogrammetry and Remote Sensing*, **2018**, *143*, 97-107. https://doi.org/10.1016/j.isprsjprs.2018.04.019

6. Liang, X.; Hyyppä, J.; Kaartinen, H.; Lehtomäki, M.; Pyörälä, J.; Pfeifer, N.; ... Huang, H. International benchmarking of terrestrial laser scanning approaches for forest inventories. *ISPRS journal of photogrammetry and remote sensing* **2018**, *144*, 137-179. https://doi.org/10.1016/j.isprsjprs.2018.06.021

7. Hackenberg, J., Spiecker, H., Calders, K., Disney, M., & Raumonen, P. (2015). SimpleTree—an efficient open source tool to build tree models from TLS clouds. *Forests*, *6*(11), 4245-4294. https://doi.org/10.3390/f6114245

8. 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). https://doi/10.1371/journal.pone.0176871
9. Bienert, A., Scheller, S., Keane, E., Mohan, F., & Nugent, C., 2007. Tree detection and diameter estimations by analysis of forest terrestrial lasers canner point clouds. In *ISPRS workshop on laser scanning* (Vol. 36, pp. 50-55).
10. Molina Valero, J. A., Ginzo Villamayor, M. J., Novo Pérez, M. A., Álvarez-González, J. G., Pérez-Cruzado, C. 2019. Estimación del área basimétrica en masas maduras de *Pinus sylvestris* en base a una única medición del escáner láser terrestre (TLS). Cuadernos de la Sociedad Española de Ciencias Forestales 45(3): 97-116. DOI: 10.31167/csecfv0i45.19887
11. Seidel, D., & Ammer, C. (2014). Efficient measurements of basal area in short rotation forests based on terrestrial laser scanning under special consideration of shadowing. *iForest-Biogeosciences and Forestry*, *7*(4), 227. https://doi.org/10.3832/ifor1084-007
12. Strahler, A. H., Jupp, D. L., Woodcock, C. E., Schaaf, C. B., Yao, T., Zhao, F., ... & Ni-Miester, W. (2008). Retrieval of forest structural parameters using a ground-based lidar instrument (Echidna®). *Canadian Journal of Remote Sensing*, *34*(sup2), S426-S440. https://doi.org/10.5589/m08-046
13. Astrup, R., Ducey, M. J., Granhus, A., Ritter, T., & von Lüpke, N. (2014). Approaches for estimating stand-level volume using terrestrial laser scanning in a single-scan mode. *Canadian Journal of Forest Research*, *44*(6), 666-676. https://doi.org/10.1139/cjfr-2013-0535