A SHORTEST PATH BASED TREE ISOLATION METHOD FOR UAV LIDAR DATA

Pasi Raumonen[1], Benjamin Brede[2], Alvaro Lau[2], Harm Bartholomeus[2]

[1]Tampere University, Mathematics, Korkeakoulunkatu 1, 33720 Tampere, Finland
[2]Wageningen University & Research, Laboratory of Geo-Information Science and Remote Sensing, Droevendaalsesteeg 3, Wageningen 6708 PB, the Netherlands

ABSTRACT

A description and test results for a new method for automatically isolating individual trees from UAV LiDAR point clouds is presented. The isolation method is based on shortest path computations with height from the ground working as a restriction. The method is tested on a 4 ha tropical forest plot, which is also scanned with TLS to provide comparison data for the isolation results. The comparison results show that on average the intersection volume of the common voxel volume between TLS and UAV trees covers 56 % of the TLS tree and 44 % of the UAV trees.

***Index Terms*—** UAV laser scanning, LiDAR, tree isolation, tree extraction, point cloud

1. INTRODUCTION

A UAV carrying a LiDAR instrument can measure large areas of forests quickly by producing detailed 3D point clouds of the areas. To measure and analyze the individual trees in the data some way requires that those individual trees must be isolated from the point cloud covering many hectares of forest and containing thousands of trees. Terrestrial laser scanner (TLS) produced point clouds have the similar problem of how to automatically and accurately isolate the individual trees. There are many published methods for this challenge such as [1, 2, 3], but there is still a lot of room for improvements, especially in the more challenging case of UAV data that has lower point density and more occlusion.

Here we present a new tree isolation method based on shortest paths calculation. The idea is that despite overlapping canopy and touching branches, most part of any given tree in a forest is defined by the property that the shortest continues path along the vegetation from a point on the tree to the ground stays on the same tree. The challenge is occlusion that disconnects many of these paths and our method is designed to overcome this difficulty with a simple restriction on path length based on the height of the starting point. We test the method with UAV data from a tropical forest plot and compare the isolation results with manually isolated trees from TLS data from the same plot.

2. DATA

UAV laser scanner data was collected with a Riegl RiCopter with VUX-1UAV [4] over moist tropical forest during a one month field campaign at the Paracou Research Station (French Guiana, France). In total, 8 flights have been performed over a 4 ha part of Paracou plot 6. Of these 6 covered the complete 4 ha, while 2 only covered the NE 1 ha of the plot, resulting in higher pulse density in this area of the plot.

Simultaneously, TLS data was collected with a Riegl VZ-400 for the same area covered by the UAV. The area was scanned at 441 scan positions on a 10 m-spaced grid according to recommendations for dense tropical forest [5].

UAV and TLS were processed to point clouds according to current practices [4, 5] and manually co-registered with common ground features like large tree branches.

3. TREE ISOLATION METHOD

There are multiple steps in the method. First, the height of the points from the ground level is estimated. Second, the point cloud is covered with small subsets that correspond mostly to connected surface patches. These patches are the smallest "blocks" from which the trees are built by determining their shortest paths to the base layer above ground. In the third step, the shortest paths are determined using a modified version of the Dijkstra's algorithm. The path determination simultaneously updates the neighbour relation of the patches so that gaps in the data are bridged to form connected and more correctly defined trees. Fourth, the trees are defined based on shortest paths as certain collection of these patches.

3.1. Height of the points and cover

The height of the points from the ground was determined by modelling the ground level with triangulation: From each non-empty 5x5 m^2 square the mean of the lowest 30 cm of points defines a vertex and then the Delaunay triangulation is applied to the vertices.

The point cloud was next randomly covered with relatively small patches (subsets with diameter between 15 and 30 cm) and their neighbours were defined [1]. The

neighbour distances between the centre points of the patches define a network for the shortest path computations.

3.2. The shortest path computations

The shortest paths were computed from every patch via the chain of neighbours to the base layer. The base layer is defined as the bottom 5 m to make sure that every tree has at least some points in the base. The computations of the path lengths are based on Dijkstra's algorithm [6]. However, there are gaps in the data resulting in separate components that cannot be reached from the base via a chain of neighbours. Secondly, in some cases gaps and touching branches result in wrong paths: for example, a branch could be not connected to the stem from its base but via another touching branch (Figure 1). The resulting paths from the branch are wrong and over-estimated. Similarly, touching branches from neighbouring trees can cause wrong paths. Therefore, Dijkstra's algorithm was modified to alleviate these problems.

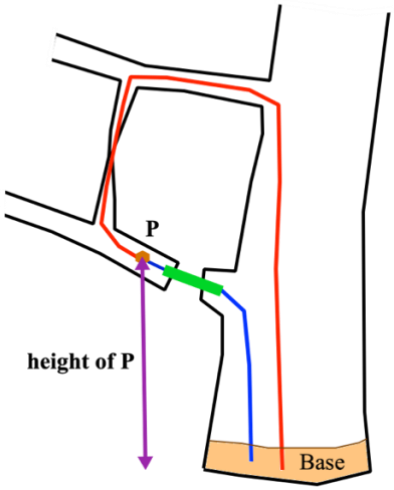

Figure 1. Schematic presentation of the shortest path method. A gap in the point cloud data can disconnect a branch at its base from the stem but via touching branches it is still connected to tree base (red path from point P). By creating a new connection (green line) that bridges the gap, shorter and more truthful path (blue) from point P is created. The red path has large path length/height ratio compared to blue path. Preventing paths with large length/height-ratio allows the creation of the new path.

At first the goal is to define the stems and therefore the paths are prevented expanding too much into branches by restricting the ratio of the path length to the height of the patch (L/H-ratio). Therefore, the maximum allowed L/H-ratio was set to 1.025. Initial path computations go as far possible with this restriction. But then potential gaps in the data need to be bridged. For the stems quite large gaps (gaps size = GS = 80 cm) are allowed to be bridged with a new connection by defining new neighbours (see Figure 1). To create new connections, all the patches close to the patches with paths defined so far are selected and the gap distances and path lengths are computed. Then new paths from the smallest gap (if below 80 cm and L/H < 1.025) are expanded and this is done as long as there are suitable patches.

Next, the computation of shortest paths is continued iteratively, such that the maximum allowed L/H and GS are increased for each iteration. The idea is to reach further and further into branches with each iteration and at the same time to create probable new connections so that the resulting paths are more realistic. At the start of the iterations the parameters are L/H = 1.05 and GS = 40 cm. Then they are increased by 0.05 and 20 cm, respectively, after each iteration. After each iteration the shortest paths are computed again from the base using the updated network of patches. The iterations stop when all the patches are connected to the base, or the maximum allowed L/H-ratio is 1.3.

3.3. Tree isolation

The isolation is based on the shortest paths and the number of such paths ending in the base layer. The idea for the isolation is that the patches in the same tree, at least mostly, have the property that their shortest path to the base layer ends at the same cluster of patches that cover the perimeter of the tree's base.

The initial tree candidates are defined iteratively. First, the patch B_{max} with the highest number of the shortest paths ending at it is selected. Then B_{max} is expanded with two layers of its neighbours to form the subset B that now hopefully covers the perimeter of a stem base. The tree candidate is then defined as all the patches whose shortest path ends at B. Next, the patch with the highest number of shortest paths not in the candidates defined so far is selected and the process is repeated. Candidates are defined this way until the highest number of paths is below 20.

For larger trees or trees whose stem was occluded during data acquisition, the whole perimeter of the tree base may not be covered with the subset B. In that case the above process may produce two or more candidates that are actually from the same tree. Thus, when all the candidates are defined by the above process, the candidates are tested if some of them form the same tree. A subset S of patches is selected from a candidate such that their path length is between 1 m and 3 m from the base and L/H-ratio < 1.05. Then S is expanded by eight layers of neighbours and if the expansion contains patches from other candidates, then these are checked if they belong to the same tree: A similar subset S is selected from each candidate and cylinders are fitted to them. Those candidates whose cylinders overlap enough are merged.

Finally, the candidates need to be completed towards the ground level, because the base layer was defined as the bottom 5 m and most of the candidates start about at 5 m above the ground. This is done similarly as the above parts of the trees, i.e., using the shortest paths. The lowest patches B of each tree are selected as bases where the path lengths are

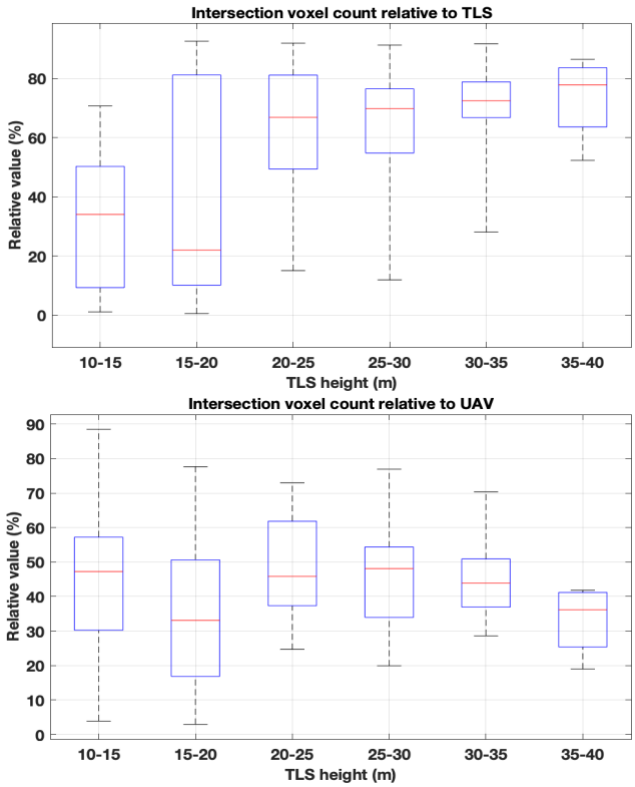

Figure 2. Comparison of the number of common voxels to the total number of voxels in the TLS (top) and UAV (bottom) trees. The red line inside the box is the median, the bottom and top of the box are 25 and 75 percentiles, and the bottom and top whiskers are the minimum and maximum, respectively.

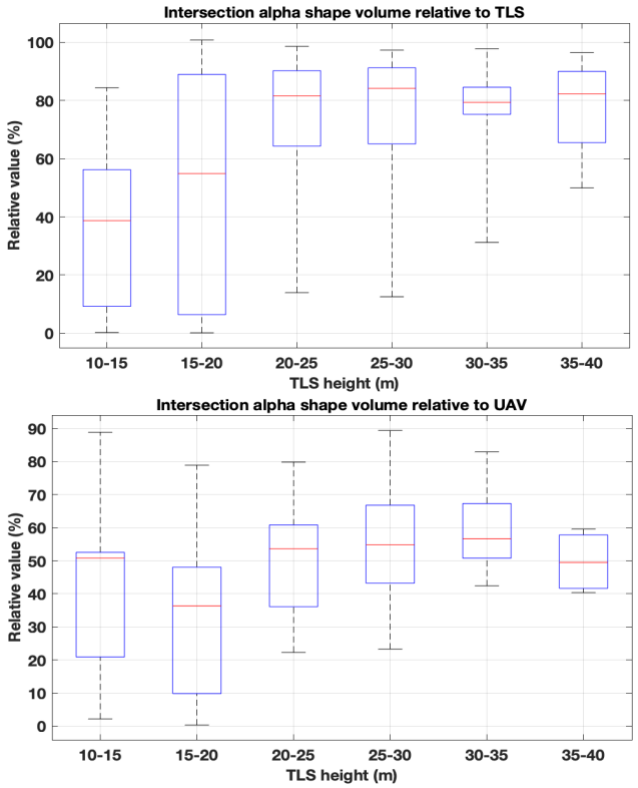

Figure 3. Comparison of the alpha shape volumes of the intersection to the alpha shape volumes of the TLS (top) and UAV (bottom) trees.

computed, and the height of the patches is reversed so that ground level has the largest height, and the zero height is for the actually highest patch in the base. Then each tree is completed with the patches whose shortest path end at the base *B* of the tree and their L/H-ratio is less than 1.3 and the path length is smaller than the height of *B* plus 75 cm.

4. RESULTS

The tree isolation method was applied to 210 x 225 m^2 part of the UAV point cloud and it resulted in 6607, 2877, 1200 and 422 trees with height above 5, 10, 20 and 30 m, respectively.

To evaluate the isolation quantitatively, some of the UAV trees were matched and compared to the same tree manually isolated from the TLS data. The trees were matched as follows: The centers (averages) of the lower section between 0-1 m and the upper section between 4-5 m of the UAV and TLS trees were computed as well as the tree heights. Then the matching distance was defined as $d_l + d_u + 1/3\ h$, where d_l and d_u are the horizontal distances between the lower and upper sections of the TLS and UAV trees, respectively, and h is the absolute height difference of the TLS and UAV trees.

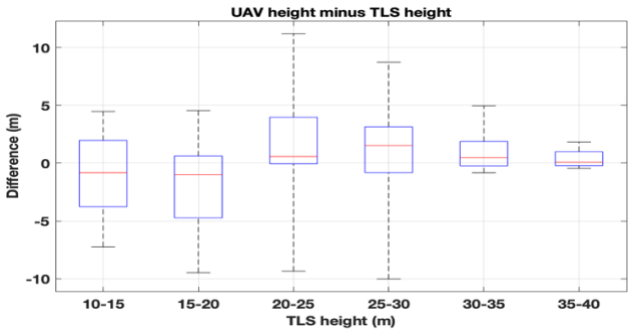

Figure 4. Comparison of the alpha shape volumes of the intersection to the alpha shape volumes of the TLS (top) and UAV (bottom) trees.

The UAV tree with the smallest matching distance, if less than 5, was selected for a given TLS tree. This resulted in 109 matched trees from the available 120 TLS trees.

The matched trees were compared in terms of voxel and alpha shape volumes of their intersections and the height difference. The point clouds were voxelized into 30 cm voxels and the number of voxels per tree were computed. The number of common voxels were compared to the total number of voxels in TLS and UAV trees. Figure 2 shows these relative voxel counts for different height classes. The alpha shapes [7] of the matched point clouds were computed with $\alpha = 2.5$. The alpha shape of the intersection of the trees,

i.e., the TLS points inside the alpha shape of the UAV tree and the UAV points inside the alpha shape of the TLS tree, was also determined. Fig. 3 shows the relative alpha shape volumes for different height classes.

Figure 4 show the height difference between UAV and TLS trees for different height classes. Figure 5 shows the best and worst matched trees in terms of their intersection alpha shape volume compared to the volume of the TLS tree.

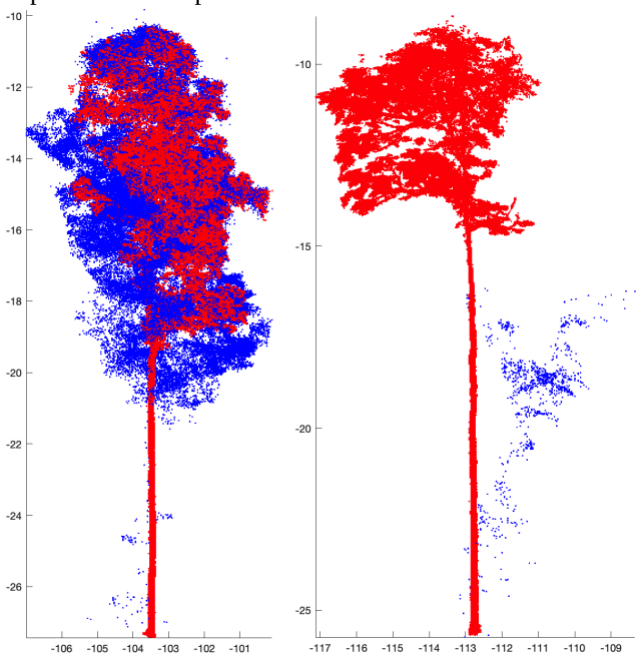

Figure 5. The best (left) and worst (right) matched trees. The blue points are from UAV and the red points from TLS. The intersection alpha shape volume covers 100% and 0.1% of the TLS tree in the best and the worst case, respectively.

5. DISCUSSION

The TLS and UAV tree comparisons show that there is large variability in the isolation results, at least in terms of how much common volume (in terms of voxels or alpha shapes) the trees occupy and in terms of tree height. In general, the intersection covers more of the TLS tree than the UAV tree, which indicates that the UAV trees contain parts not in the TLS trees. It is quite expected that the intersection cannot cover the TLS tree fully as there are lot of structure that cannot be seen in UAV. But why the intersection covers generally so little of the UAV tree may be due to L/H and GS parameter values used in the isolation process and it requires further study. However, when the trees get taller and thus usually also bigger, the isolation results get better: For example, the trees below 20 m in height (TLS) have median intersection volumes (voxel or alpha shape) between 22 % and 56 %, whereas for the trees above 20 m in height the same measures are between 37 % and 84 %. This is quite expected as shorter trees are often covered by the canopy of the higher neighboring trees and thus their visibility from UAV is less than for the taller trees. For the tree height the median difference is close to zero in all height classes, but the variability is large for all but the tallest trees. This is again quite expected because the shorter trees have less visibility.

The results of the tree isolation are promising, at least when considering that the larger errors are mostly in the small trees that are less visible in the UAV data. More detailed analysis of the comparisons is still needed to get better sense how well the isolation is and where in the process the greatest needs for improvements are. For example, the woody volumes of the matched trees should be estimated with QSMs [1] to get better volume estimates than voxel counts and alpha shapes which can be quite sensitive to small errors in isolation or small changes in the used parameter values. There probably are ways that could improve the method, for example, adding more information than just the height of points for the shortest path computations could help to define the individual trees better.

REFERENCES

[1] P. Raumonen, E. Casella, K. Calders, S. Murphy, M. Åkerblom, M. Kaasalainen, "Massive-scale Tree Modelling from TLS Data," *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Volume II-3/W4, 189-196.

[2] A. Burt, M. Disney, K. Calders, "Extracting individual trees from lidar point clouds using *treeseg*," *Methods in Ecology and Evolution*, 2019, vol. 10, Issue 3, 438-445.

[3] Di Wang, "Unsupervised semantic and instance segmentation of forest point clouds," ISPRS Journal of Photogrammetry and Remote Sensing, 2020, vol. 165, 86-97.

[4] B. Brede, A. Lau, H. M. Bartholomeus, and L. Kooistra, "Comparing RIEGL RiCOPTER UAV LiDAR Derived Canopy Height and DBH with Terrestrial LiDAR," *Sensors*, vol. 17, no. 10, p. 2371, 2017.

[5] P. Wilkes, A. Lau, M. Disney, K. Calders, A. Burt, J. Gonzalez de Tanago, H. Bartholomeus, B. Brede, M. Herold, "Data Acquisition Considerations for Terrestrial Laser Scanning of Forest Plots," *Remote Sens. Environ.*, vol. 196, pp. 140–153, 2017.

[6] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*. 1959, 1: 269–271.

[7] H. Edelsbrunner, D. G. Kirkpatrick, R. Seidel, "On the shape of a set of points in the plane," *IEEE Transactions on Information Theory*, 1983, 29 (4), 551–559.