

An efficient approach to 3D single tree-crown delineation in LiDAR data

Domen Mongus*, Borut Žalik

University of Maribor, Faculty of Electrical Engineering and Computer Science, Smetanova 17, SI-2000 Maribor, Slovenia



ARTICLE INFO

Article history:

Received 1 December 2014

Received in revised form 12 August 2015

Accepted 15 August 2015

Available online 7 September 2015

Keywords:

LiDAR

Single tree-crown delineation

Mathematical morphology

Watershed

LoFS

Verticality

ABSTRACT

This paper proposes a new method for 3D delineation of single tree-crowns in LiDAR data by exploiting the complementaries of treetop and tree trunk detections. A unified mathematical framework is provided based on the graph theory, allowing for all the segmentations to be achieved using marker-controlled watersheds. Treetops are defined by detecting concave neighbourhoods within the canopy height model using locally fitted surfaces. These serve as markers for watershed segmentation of the canopy layer where possible oversegmentation is reduced by merging the regions based on their heights, areas, and shapes. Additional tree crowns are delineated from mid- and under-storey layers based on tree trunk detection. A new approach for estimating the verticalities of the points' distributions is proposed for this purpose. The watershed segmentation is then applied on a density function within the voxel space, while boundaries of delineated trees from the canopy layer are used to prevent the overspreading of regions. The experiments show an approximately 6% increase in the efficiency of the proposed treetop definition based on locally fitted surfaces in comparison with the traditionally used local maxima of the smoothed canopy height model. In addition, 4% increase in the efficiency is achieved by the proposed tree trunk detection. Although the tree trunk detection alone is dependent on the data density, supplementing it with the treetop detection the proposed approach is efficient even when dealing with low density point-clouds.

© 2015 International Society for Photogrammetry and Remote Sensing, Inc. (ISPRS). Published by Elsevier B.V. All rights reserved.

1. Introduction

By allowing for fast and accurate monitoring of vast geographical areas with resolutions reaching beyond square decimetres, airborne Light Detection and Ranging (LiDAR) technology has triggered a new wave of research into Earth observations. The unprecedented information contained within the acquired datasets, however, imposes great challenges for data processing and drives the demand for more efficient object recognition algorithms capable of dealing with those topologically unstructured and often noisy data. When considering the obviously huge environmental impacts of forests, capturing and processing forest data is an especially important yet difficult task due to their large extends and the complex geometries of the trees.

Much of the research into the structural properties of forests (e.g. estimation of leaf area index ([Solberg et al., 2009](#)), fuel metrics ([Erdödy and Moskal, 2010](#)), and other common forest inventory variables ([Andersen et al., 2005](#))) has utilised area-based methods

([Breidenbach et al., 2010](#)). However, this approach is limited in resolution and unable to consider the spatial distribution of objects ([Richardson and Moskal, 2011](#)). By identifying and delineating individual trees, the so-called individual tree-crown approach provides an improved methodological framework for forest characterisation. In the more common cases, delineation of individual trees is utilised by segmentation of a canopy height model (CHM) ([Hyyppä et al., 2001; Persson et al., 2002; Chen et al., 2006; Li et al., 2012](#)). Generated by subtracting the digital terrain model (DTM) from the digital surface model (DSM), CHM provides estimations of tree heights at discrete space intervals (usually grid). Local maxima are then used to define the positions of treetops, while the segmentation is commonly achieved by applying region growing ([Hyyppä et al., 2001; Persson et al., 2002](#)) or watershed algorithms ([Yu et al., 2011; Alonso et al., 2014](#)). Although this is an elegant approach, there are several issues related to it. Namely, due to the complex geometry and noise contained within the LiDAR data, defining treetops by local maxima alone inevitably results in oversegmentation. Several solutions have already been proposed for overcoming this issue, including smoothing CHM as described by [Persson et al. \(2002\)](#), controlling the distances between treetops ([Chen et al., 2006; Li et al., 2012; Alonso et al., 2014](#)).

* Corresponding author. Tel.: +386 2 220 7408; fax: +386 2 220 7272.

E-mail address: domen.mongus@um.si (D. Mongus).

URL: <http://gemma.feri.um.si> (D. Mongus).

2014), and data scaling based on the minimum curvature estimation as considered by Yu et al. (2011). Nevertheless, by assuming that individual trees form distinct peaks in the CHM, these methods tend to perform well in structurally homogeneous plantations, single-species dominated stands, and forests with widely spaced trees. However, significantly lower accuracies have been reported in more complex cases, where trees are often clumped together and may have well developed under and midstorey canopy layers consisting of smaller, shade tolerant trees (Richardson and Moskal, 2011; Lu et al., 2014). In order to improve accuracy in these cases, further research is directed towards 3D single tree-crown delineation (Reitberger et al., 2009; Lu et al., 2014).

According to our study, 3D approaches for single tree-crown delineation can be classified as either statistical or clustering based. Statistical methods rely on so-called vertical density profiles that are essentially probability density functions of point heights estimated within the local neighbourhoods (Coops et al., 2007; Wang et al., 2008; Dean et al., 2009; Duncanson et al., 2014). An early approach towards the single tree crown delineation was proposed by Wang et al. (2008). Their method divides the input point-clouds into a grid and vertical density profiles are estimated for each grid-cell. A series of horizontal 2D projections are then made at the different height levels and the tree crowns are delineated by extracting their contours at each projection. On the other hand, some authors have delineated vegetation strata by fitting continuous probability distributions to the vertical density profiles (Coops et al., 2007; Dean et al., 2009). Recently Duncanson et al. (2014) applied treetop detection on smoothed CHMs, where its delineation is achieved based on a marker-controlled watershed. Vertical density profiles are then estimated for each watershed region, and the troughs detected within them, are used to perform 3D segmentation by point-cloud refinement. In any case, methods based on vertical density profiles have some common disadvantages. Namely, as horizontal branches may form distinct peaks within the vertical density profiles, they are often recognised as individual tree crowns, while overlapping tree crowns of similar heights cause their corresponding peaks to merge and thus they cannot be separated.

On the other hand, clustering-based methods rely on the assumption that the centres of tree crowns are characterised by a high density of points (Reitberger et al., 2009; Ferraz et al., 2012). Reitberger et al. (2009) proposed the first method towards this direction. Their approach uses distance-based hierarchical clustering for defining the number of tree trunks within a single watershed region, while the RANSAC algorithm is used for estimating their positions by detecting points lying along vertical lines. Finally, three-dimensional tree-crown delineation is achieved by normalised cuts as proposed by Shi and Malik (2000). Consequentially, this method is computationally demanding and, at the same time, requires very high point densities as obtained by full-waveform LiDAR systems (Reitberger et al., 2009). Ferraz et al. (2012) presented a method for the characterisation of multi-layered forests from LiDAR data using iterative clustering based on a mean shift algorithm. Unlike other methods, this approach does not rely on CHM and directly applies to the 3D point clouds. Although this has proven to be an advantage in the cases of Mediterranean forests that are characterised by an open dominant canopy, this method might be inefficient when dealing with overlapping tree crowns, where clusters of dense points often appear within the conjunction areas. In addition, clusters of dense points are also characteristic for open branches with dense leaves or needless. However, both of these issues can be mitigated by additionally considering the positions of tree trunks and clustering the points accordingly. Recently, Lu et al. (2014) proposed a new approach in this direction by detecting tree trunk points based on their presumably higher intensity values. As this might not

always be the case, especially when dense leaves shade the tree trunks, the proposed intensity based thresholding is recommended for use only under leaf-off conditions. This method additionally relies on a high penetration rate that is achieved in this way.

This paper considers a clustering-based approach for 3D single tree crown delineation that is capable of exploiting the complementarities of treetop and tree trunk detections, while several improvements on them are additionally proposed. Firstly, we introduce a new definition of treetops that is based on local neighbourhoods of concave shapes, detected within CHM. In comparison to the traditionally used detection of local maxima on smoothed CHM, the proposed approach is better adapted for dealing with tree crowns of different sizes and anisotropic shapes that consequently ensure its improved delineation, while possible oversegmentation is mitigated by merging the obtained regions based on their heights, areas, and shapes. A new approach for tree trunk detection is then proposed based on estimating the verticalities of points' distributions. The high contrast between tree trunks and branches that is achieved in this way allows for efficient definitions of markers for density-based 3D clustering by which delineation of additional tree crowns from the mid- and understorey layers is achieved, while boundaries of delineated tree crowns from the canopy layer are used to prevent overspreading of regions. We have derived our conclusions from the graph-theoretical perspective in order to establish a common mathematical framework for all the concepts being used. Thus, 3D single tree delineation is controlled by manipulating over the edge weight functions, while all the segmentations are achieved using marker-controlled watershed-cuts. The necessary graph theoretical definitions are explained in Section 2. Section 3 proposes a new framework for delineating single-trees, where the definitions of markers and edge weight functions are explained in detail. Sections 4 and 5 respectively provide the results and the discussion, while Section 5 concludes the paper.

2. Theoretical foundations

This section provides the basic definitions used in this paper. In general, the sets and discrete functions acting on them are denoted with capital letters, while the elements of a set and constants are denoted with lower-case letters. Capital Greek letters are used for denoting continuous functions and lower-case Greek letters are reserved for known constants such as π . We define an undirected graph G by the set of vertices $V(G)$ and an unordered set of edges $E(G)$. An edge $e_{ij} \in E(G)$ is defined by a pair of vertices $e_{ij} = (v_i, v_j)$, where $v_i, v_j \in V(G)$. A graph $G_A \subseteq G$ is a subgraph of G if $V(G_A) \subseteq V(G)$ and $E(G_A) \subseteq E(G)$. An arbitrary vertex attribute-function, denoted as A , is a mapping $A : V(G) \rightarrow \mathbb{R}$ (e.g., spatial coordinates of vertex v_i are given as $X[v_i], Y[v_i]$, and $Z[v_i]$), while the edge weight-function is the mapping $W : E(G) \rightarrow \mathbb{R}$. An edge-weighted graph is thus given by a pair (G, W) . When considering operations on weight-functions, we denote the infimum (i.e. minimum) by \wedge and \vee is used to denote the supremum (i.e. maximum).

Let an ordered sequence of vertices $P_{i_0, i_L} = \{v_{i_0}, v_{i_1}, \dots, v_{i_L}\}$, we say that P_{i_0, i_L} is a path from v_{i_0} to v_{i_L} in a graph G if $\forall l \in [1, L], e_{i_{l-1}, i_l} \in E(G)$. The length of the path is given by L , or equivalently $|P_{i,j}| - 1$, where notation $\widehat{P}_{i,j}$ is used to emphasise that $P_{i,j}$ is the shortest path between v_i and v_j . If there exists $P_{i,j}$, we say that vertices v_i and v_j are linked and a graph G is called connected if $\forall v_i, v_j \in V(G)$ there exists $P_{i,j}$ in G . A connected component $C_i \subseteq G$ is defined as a maximal connected subgraph of G . Each C_i is uniquely addressed by any member vertex $v_i \in C_i$. When considering two non-empty subgraphs $G_A, G_B \subseteq G$, G_B is called an extension of G_A in G if $G_A \subseteq G_B$ and any connected component of G_B contains

exactly one connected component of G_A . Consequently, a set of edges $E(G_C) \subseteq E(G)$ is a graph-cut for G_A if $\overline{E(G_C)}$ is an extension of G_A and if $E(G_C)$ is minimal for this property (Diestel, 2005; Cousty et al., 2009). Note that $\overline{E(G_C)}$ is a complement graph of $E(G_C)$, while we denote a graph-cut for G_A as $E^C(G_A)$.

Another particularly important subgraph is a graph of local minima $M(G, W) \subseteq (G, W)$ of the edge-weighted graph (G, W) . $M(G, W)$ is defined as a set of connected subgraphs of equally weighted edges, the adjacent edges of which have strictly greater values.

2.1. Marker controlled-watershed

There exist several approaches that allow for computation of watershed (Vincent and Soille, 1991; Bertrand, 2005; Couprise et al., 2005). Indeed, the proposed method for single tree crown delineation may be implemented using any of these. However, we rely here on a graph-theoretical definition of watershed, as it is independent of the data dimensionality, connectivity, and attribute-function used for its estimation. It can be given by the notion of minimum spanning forest $MSF(G, W)$, constructed on (G, W) , relative to the set of local minima $M(G, W)$. Let $G_A \subseteq G$, we say that G_A is a forest relative to $M(G, W)$ if G_A is an extension of $M(G, W)$ and for any extension G_B of $M(G, W)$, such that $G_B \subseteq G_A$, $V(G_A) = V(G_B)$ implies $G_A = G_B$. In other words, if G_A is a forest, then we cannot remove any edge from G_A without affecting its vertex-set. Additionally, when $V(G_A) = V(G)$, we say G_A is a spanning forest relative to $M(G, W)$ for G . $MSF(G, W)$ relative to $M(G, W)$ for (G, W) is, thus, a spanning forest relative to $M(G, W)$, the sum of the edge-weights of which is less than or equal to the sum of the edge-weights of any other spanning forest relative to $M(G, W)$ for (G, W) . As shown by Cousty et al. (2009), the watershed is a graph-cut $E^C(MSF(G, W))$ for $MSF(G, W)$ or, equivalently, watershed regions are connected components $C_i \in MSF(G, W)$.

In addition to its efficient implementation (see the pseudocode proposed by Cousty et al. (2009)), the given watershed definition allows for a flexible graph segmentation by manipulating over the edge weight-functions. When considering the classical watershed problem where those “valleys” separated by “ridges” are being segmented, edge-weights are defined by the lowest attribute value of the corresponding vertices, i.e. $W[e_{ij}] = \wedge\{A[v_i], A[v_j]\}$. Inversely, the segmentation of peaks is achieved by $W[e_{ij}] = \wedge\{-A[v_i], -A[v_j]\}$. Furthermore, a marker-controlled watershed can be achieved by filtering the underlying attribute-function A . In our case, morphological filters are used for this purpose (Serra and Vincent, 1992). As these are well-known in LiDAR data processing, only the fundamental notations for graph based mathematical morphology are given next, while additional details can be found in (Najman and Talbot, 2013). We define the neighbourhood of a vertex v_i (i.e. structuring element) by a subgraph $S_i^s \subseteq G$ of size s , the vertex-set and edge-set of which are:

$$\begin{aligned} V(S_i^s) &= \{v_j \in V(G) : \exists \hat{P}_{ij}, |\hat{P}_{ij}| - 1 \leq s\} \cup \{v_i\}, \\ E(S_i^s) &= \{e_{j,k} \in E(G) : v_j, v_k \in V(S_i^s)\}. \end{aligned} \quad (1)$$

In other words, the vertex-set of S_i^s consists of vertices whose shortest paths from v_i are no longer than s and its edge-set contains the corresponding edges from $E(G)$. Note that in the case of traditional 8-connectivity on a regular 2D grid, this definition is equivalent to an $s \times s$ square structuring element or cross structuring element in the case of 4-connectivity. A morphological dilation of an attribute function A with S^s is denoted as $\delta^s(A)$, $\epsilon^s(A)$ is morphological erosion and $\epsilon_A^R(A^M)$ is reconstruction by erosion, where A^M is a marker function. As proposed by Vachier and Meyer (2005), a straightforward approach to the marker-controlled watershed can

be achieved by defining $A^M[v_i] = A[v_i]$, if v_i is marked, and $A^M[v_i] = \infty$ otherwise. The edge-weight function for the marker-controlled watershed is then given as

$$W[e_{ij}] = \wedge\{\epsilon_A^R(A^M)[v_i], \epsilon_A^R(A^M)[v_j]\}, \quad (2)$$

or $W[e_{ij}] = \wedge\{-\epsilon_A^R(A^M)[v_i], -\epsilon_A^R(A^M)[v_j]\}$ when peaks are being segmented rather than valleys. Based on these notions, a new approach for single-tree delineation in LiDAR data is proposed next.

3. Single tree-crown delineation

This section provides a new framework for single-tree delineation within LiDAR data (see Fig. 1). In the initial step of the method the topology construction over the input LiDAR point-cloud is done consistent with the graph-theoretical approach. Although any type of undirected graph can be used, sufficient memory and computational efficiency is achieved by constructing a regular 2D grid for the representation of CHM during canopy layer analysis, while tree trunk detection is performed within the voxel-space. During the final step, the voxel-space is segmented in order to achieve 3D delineations of individual trees.

3.1. Initialisation

During initialisation, connectivity over the input set of LiDAR points is established by constructing two graphs G_2 and G_3 in the forms of regular 2D and 3D grids, where standard 8- and 26-connectivity are used, respectively. For simplicity they are both of equal resolution r , thus G_2 is equal to the projection of G_3 to 2D, i.e. $G_2 = \text{proj}_2(G_3)$. A digital surface model is then defined as a mapping $DSM : V(G_2) \rightarrow \mathbb{R}$, where $DSM[v_i]$ is equal to the highest point within the neighbourhood of v_i (i.e. the considered grid-cell). When there are no LiDAR points within the neighbourhood of v_i , $DSM[v_i]$ is interpolated, as explained by Mongus et al. (2013). DTM is in our case constructed as described by Mongus et al. (2014) and subtracted from DSM . In addition, morphological closing φ_{w_1} and opening γ_{w_1} with structuring element w_1 of size

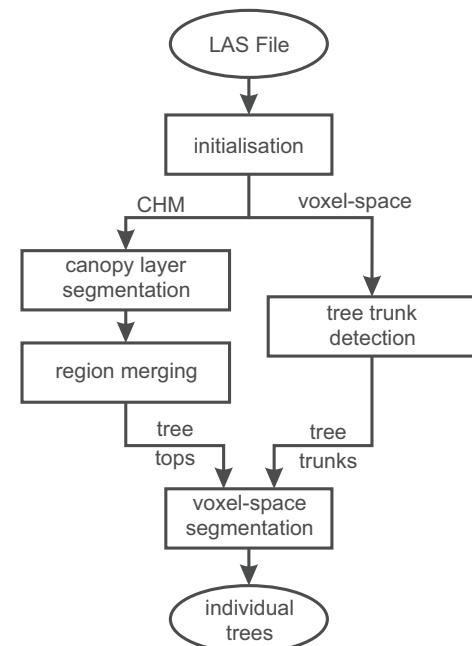


Fig. 1. The proposed framework for 3D delineation of single-trees.

1m are applied in order to remove any possible outliers. Thus, normalised canopy height model CHM is defined as

$$CHM[v_i] = \gamma_{w_1} (\varphi_{w_1}(DSM - DTM))[v_i]. \quad (3)$$

Finally, the vertex attribute-function $N : V(G_3) \rightarrow \mathbb{Z}$ is defined by the number of LiDAR points lying within the neighbourhood of $v_i \in G_3$ (i.e. within the voxel-cell). When estimating N , LiDAR points lying below DTM or higher than CHM are treated as outliers and are, therefore, ignored. In this way, $N[v_i] = 0, \forall v_i \in G_2$ such that $Z[v_i] - DTM[v_i] > CHM[v_i]$ or $Z[v_i] - DTM[v_i] < 0$.

Note that an implementation of the proposed method does not require actual construction of G_2 and G_3 nor to maintain LiDAR points within main memory during its execution. Namely, G_2 and G_3 only define the bounding box, the size, and the resolution of matrices used for the implementation of discrete functions, where DTM , DSM , and CHM are implemented as 2D matrices of floating-point values and N is a 3D matrix of integers. A vertex v_i refers to the coordinates of a grid cell, while the real-world coordinates of a vertex (i.e. $X[v_i], Y[v_i]$, and $Z[v_i]$) can be estimated on the fly for improved memory efficiency.

3.2. Segmentation of canopy layer

As already shown by the related methods considered in the introduction, CHM contains sufficient information for delineating single-trees within the canopy layers of forests. In this section, we first propose an improved treetop detection based on locally-fitted surfaces (LoFS). This is a computationally efficient alternative for well-known Hough transforms (Fernandes and Oliveira, 2008) that allows for the extraction of vertices following an arbitrary predefined distribution by applying so-called fitting and linking steps. During the fitting step, a set of candidates describing surface patches is obtained by estimating best-fitted surfaces within the local neighbourhood of each vertex. These are then filtered during the linking step according to some predefined conditions, while vertices that conform to the preserved patches are finally linked to them. This approach was first proposed by Mongus et al. (2014), where LoFS were applied for the extraction of planar vertices for building detection in LiDAR data, while we extend it here by using least-square fitting of the second degree polynomials in order to achieve an improved definition of treetops. As the least-square fitting is well-known, it is not discussed at this point. An in-depth study of the least-squares was provided by Bevington and Robinson (2002) and Schneider and Eberly (2002), while an efficient implementation of the best-fitted polynomials is available at <http://www.geometrictools.com/>. Let $\Pi^s(G_2, CHM)$ be a set of polynomials, best-fitted to CHM within the neighbourhood of a given vertex v_i (note that the neighbourhood of v_i is defined by a subgraph $S_i^s \subseteq G_2$). The best-fitted polynomial $\Pi^s(G_2, CHM)[v_i]$ is, for any given vertex $v_j \in V(S_i^s)$, defined in an explicit form as

$$\begin{aligned} \Pi^s(G_2, CHM)[v_i](v_j) = & c_5 X[v_j]^2 + c_4 X[v_j]Y[v_j] + c_3 Y[v_j]^2 \\ & + c_2 X[v_j] + c_1 Y[v_j] + c_0. \end{aligned} \quad (4)$$

When considering coefficients c_k (where $k \in [0, 5]$) defining Π^s , the neighbourhood of v_i at scale s can be segmented into convex, concave, or neither of those, i.e. a saddle (Gradshteyn and Ryzhik, 2000). Namely, it can be shown that $\Pi^s(G_2, CHM)[v_i]$ is

- 1. **saddle** if $c_4^2 - 4c_5c_3 \geq 0$, otherwise
- 2. **convex** if $c_5 \geq 0$, and
- 3. **concave** if $c_5 < 0$.

Intuitively, vertices with concave neighbourhoods represent treetops, while saddle and convex neighbourhoods appear near

the boundaries of tree-crowns. As the used surface approximation is, by itself, capable of absorbing a fair amount of noise, treetop detection is significantly more reliable in this way (see Fig. 2). It is also better adapted for dealing with anisotropic tree-crowns than e.g. Gaussian filtering at variable scales (Lowe, 2004). More importantly, the proposed approach allows for controlling the curvatures of the treetops by applying a threshold \tilde{t}_M instead of using a constant 0. Thus, the marker function CHM_M used for the marker-controlled watershed is given as

$$CHM_M[v_i] = \begin{cases} CHM[v_i], & \text{if } c_4^2 - 4c_5c_3 < 0 \text{ and } c_5 < \tilde{t}_M, \\ \infty, & \text{otherwise.} \end{cases} \quad (5)$$

The edge-weight function, allowing for watershed segmentation based on LoFS, is defined as

$$W_\Pi[e_{ij}] = \bigwedge \{-\epsilon_{CHM}^R(CHM_M)[v_i], -\epsilon_{CHM}^R(CHM_M)[v_j]\}. \quad (6)$$

Note that Eq. (6) is equivalent to the general definition of marker-controlled watershed, given by Eq. (2), where CHM is used instead of an arbitrary attribute function A and is multiplied by -1 as peaks of CHM are segmented rather than valleys. Watershed is thus obtained by a cut $E^C(MSF(G_2, W_\Pi))$ and watershed regions defining individual trees are connected components $C_i \in MSF(G_2, W_\Pi)$. In order to simplify the notations, E^C is in continuation used to denote the set of edges defining watershed-cut $E^C(MSF(G_2, W_\Pi))$ and MSF denotes the set of watershed-regions $MSF(G_2, W_\Pi)$.

Although the results, as demonstrated in Fig. 2, show improved treetop detection in comparison with local maxima search on smoothed CHM , the LoFS-based approach may still lead to oversegmentation when individual trees form two or more distinct peaks in CHM . Therefore, the merging of watershed-regions $C_i \in MSF$ is proposed next.

3.3. Merging of watershed regions

This section proposes an approach that allows for the integration of arbitrary geometric attributes within the definitions of individual trees. This is achieved by filtering edges of the watershed-cut E^C according to the heights, areas, and shape compactnesses of the connected components they are linking (note that $e_{ij} \in E^C$ implies $C_i \neq C_j$). For this purpose, we consider the boundary of C_i as a graph $B(C_i)$, the vertex- and edge-sets of which are, respectively, given as

$$\begin{aligned} V(B(C_i)) &= \{v_j \in V(C_i) : \exists e_{j,k} \in E^C, v_k \notin V(C_i)\}, \\ E(B(C_i)) &= \{e_{j,k} \in E^C : v_j \in V(C_i), v_k \notin V(C_i)\}. \end{aligned} \quad (7)$$

Let connected components $C_i, C_j \in MSF$, we then define:

- **the height** $W_H(e_{ij})$ of the edge e_{ij} by the elevation differences between the boundary vertices v_i and v_j and the peaks of the corresponding connected components C_i and C_j , respectively. The smaller elevation difference between those is selected, as shown in Fig. 3a, where the value of w_H^j is used. Formally,

$$\begin{aligned} W_H(e_{ij}) &= \bigwedge \{w_H^i, w_H^j\}, \\ w_H^i &= \bigvee_{v_n \in V(C_i)} CHM[v_n] - CHM[v_i], \\ w_H^j &= \bigvee_{v_m \in V(C_j)} CHM[v_m] - CHM[v_j]. \end{aligned} \quad (8)$$

- **the area** $W_A(e_{ij})$ linked by e_{ij} as

$$W_A(e_{ij}) = |V(C_i)| + |V(C_j)|. \quad (9)$$

Note that when area is given in m^2 , $W_A(e_{ij})$ should be multiplied by the square of grid resolution r^2 .

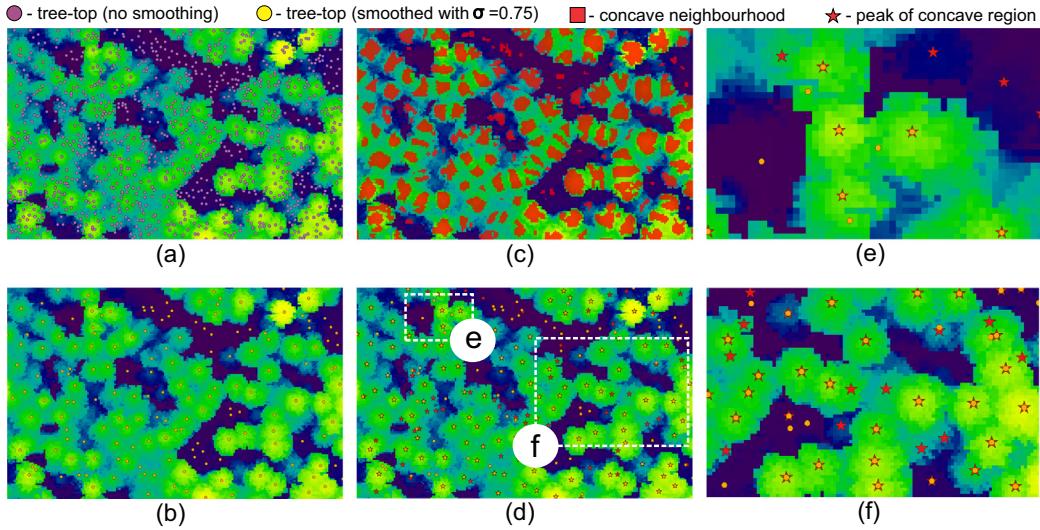


Fig. 2. Treetop detection based on the local maxima and LoFS, where (a,b) show a decrease of false positives achieved by progressive Gaussian filtering (without smoothing and smoothing with $\sigma = 0.75$, respectively) and (c) shows detection of concave regions with LoFS (using $t_M = 0.01$). The comparison of (d) the peaks of concave regions and local maxima shows that later results in (e) oversegmentation due to noise, while at the same time (f) CHM is oversmoothed and several treetops are lost.

- **the shape compactness** $W_{SC}(e_{ij})$ of the union of C_i and C_j as (Montero and Bribiesca, 2009)

$$W_{SC}(e_{ij}) = \frac{|V(B(C_i \cup C_j))|^2}{4\pi|V(C_i \cup C_j)|}, \quad (10)$$

where the set of boundary vertices B is defined by Eq. (7). Note that by the same principle, the shape compactness of C_i alone is defined as $SC(C_i) = \frac{|V(B(C_i))|^2}{4\pi|V(C_i)|}$.

Based on the assumption that peaks belonging to the same tree are separated by the edges of low heights W_H , we now propose merging connected components $C_i \in MSF$ by adding edges $e_{ij} \in E^C$ from the watershed-cut E^C according to the given merging criterion K . Let $\hat{e}(B(C_i), K)$ be the lowest edge from the boundary of C_i , satisfying a given merging criterion K . Formally, $\hat{e}(B(C_i), K)$ is defined as

$$\hat{e}(B(C_i), K) = e_{ij} : \forall e_{n,m}, W_H[e_{ij}] \leqslant W_H[e_{n,m}], \quad (11)$$

where $e_{ij}, e_{n,m} \in E(B(C_i))$ such that $K(e_{ij})$ and $K(e_{n,m})$ are true. When several edges satisfy Eq. (11), one is selected at random. In our case, the following merging criteria are used:

- **merging criterion by height** $K_H^{\hat{t}_H}$ is defined by the threshold \hat{t}_H as

$$K_H^{\hat{t}_H}(e_{ij}) = \begin{cases} \text{true}, & \text{if } W_H(e_{ij}) \leqslant \hat{t}_H, \\ \text{false}, & \text{otherwise.} \end{cases} \quad (12)$$

In simple terms, C_i and C_j are merged if the height of the lowest edge $e_{ij} \in E^C$ between them is less than or equal to \hat{t}_H .

- **merging criterion by area** K_A is given by the smallest \hat{t}_A and the largest \check{t}_A area-thresholds. Thus,

$$K_A^{\hat{t}_A, \check{t}_A}(e_{ij}) = \begin{cases} \text{true}, & \text{if } \wedge\{|C_i|, |C_j|\} \leqslant \hat{t}_A \text{ and } W_A(e_{ij}) \leqslant \check{t}_A, \\ \text{false}, & \text{otherwise.} \end{cases} \quad (13)$$

Accordingly, C_i and C_j are merged if the area of C_i or C_j is less than or equal to \hat{t}_A , while the sum of their areas is no greater than the value of \check{t}_A , as shown in Fig. 3c.

- **merging criterion by shape-compactness** K_{SC} is designed to minimise the shape-compactness and keep it below the given threshold \check{t}_{SC} . In this way,

$$K_{SC}^{\check{t}_{SC}}(e_{ij}) = \begin{cases} \text{true}, & \text{if } \check{t}_{SC} \leqslant \vee\{SC(C_i), SC(C_j)\} \geqslant W_{SC}(e_{ij}), \\ \text{false}, & \text{otherwise.} \end{cases} \quad (14)$$

In other words, C_i and C_j are merged if the shape-compactness of at least one of them is greater than \check{t}_{SC} , while lower shape-compactness is achieved by merging them, as achieved by removing the red part of the boundary in Fig. 3b.

Prior to merging, a set of boundary edges $\cup_i E(B(C_i))$ is ordered according to their heights W_H (from smallest to largest) and merging is sequentially applied until all the edges satisfying $K_{CHM}^{\hat{t}_H, \check{t}_A, \check{t}_{SC}} = K_H^{\hat{t}_H} \wedge K_A^{\hat{t}_A, \check{t}_A} \wedge K_{SC}^{\check{t}_{SC}}$ are added to MSF . Thus, a single tree-crown within the canopy layer is defined as a connected component $C_i \in T_{CHM}^{\hat{t}_H, \check{t}_A, \check{t}_{SC}}(G_2)$ of the graph $T_{CHM}^{\hat{t}_H, \check{t}_A, \check{t}_{SC}}(G_2)$, given by

$$T_{CHM}^{\hat{t}_H, \check{t}_A, \check{t}_{SC}}(G_2) = MSF \bigcup_{C_i \in MSF} \hat{e}(B(C_i), K_{CHM}^{\hat{t}_H, \check{t}_A, \check{t}_{SC}}). \quad (15)$$

3.4. Detection of tree trunks

As tree trunks traditionally grow vertically, the likelihood of a vertex being a tree trunk is correlated with the likelihood that a vertex lies along a vertical line. In this section we introduce a convolution kernel designed to estimate the likelihood that a given neighbourhood is vertical (see Fig. 4).

Let a 26-connected voxel-space in the form of a graph G_3 and a vertex-attribute function $N : V(G_3) \rightarrow \mathbb{Z}$ defined by the number of LiDAR points contained within the corresponding voxel. Given $v_i \in G_3$, such that $N[v_i] \geqslant 1$, we may assume that if v_i lies on a vertical line, there exists a set of vertices $\{v_j\} \subseteq V(S_i^s)$ within the neighbourhood S_i^s , such that $X[v_i] = Y[v_j], Y[v_i] = Y[v_i]$, and $Z[v_i] \neq Z[v_j]$, where $N[v_j] \geqslant 1$. In other words, if a given vertex is along a vertical line, several vertices from its neighbourhood are above or below it (see Fig. 5). However, the directions of the lines may deviate from being perfect vertical as tree trunks can be skewed or inclined, meaning that vertices may also differ by

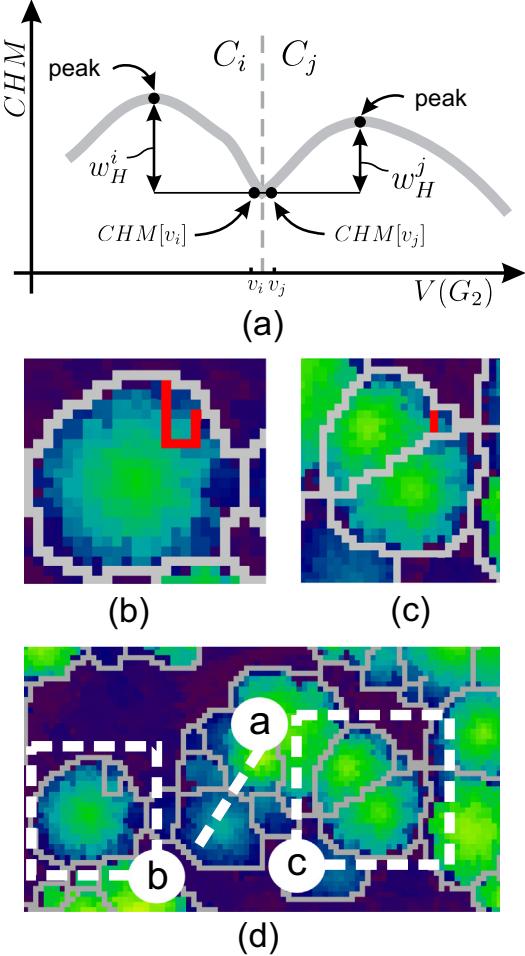


Fig. 3. Region merging criterion, where (a) shows a graphic definition of $W_H(e_{ij})$ given by the smallest elevation difference $\wedge\{w_H^i, w_H^j\}$ between the boundary vertices v_i and v_j and the peaks (i.e. $\vee_{v_n \in V(C_i)} CHM[v_n]$ and $\vee_{v_m \in V(C_j)} CHM[v_m]$) of the corresponding connected components C_i and C_j , respectively (b) demonstrates region merging by shape compactness W_{SC} (note that lower value of shape compactness is achieved by removing the red part of the boundary), (c) demonstrates region merging by area W_A , and (d) displays the initial watershed, boundaries of which (i.e. E^*) are marked grey.

X- and Y-values, not only by Z. Accordingly, we use Gaussian function for describing the probability of v_i being along a vertical line in regards to its XY-difference from v_j . On the other hand, when v_i is on a branch, horizontal distribution of points is expected. Thus, when there exists a set of points $\{v_k\} \subseteq V(S_i^s)$, such that $X[v_i] \neq X[v_k], Y[v_i] \neq Y[v_k]$, and $Z[v_i] = Z[v_k]$, where $N[v_k] \geq 1$, the

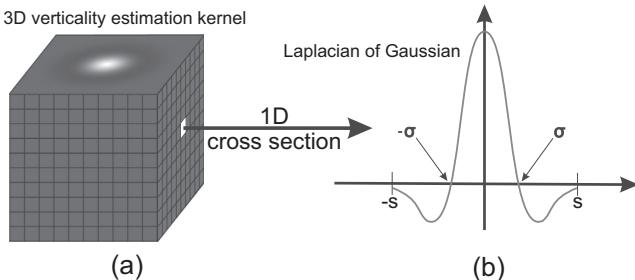


Fig. 4. Estimation of verticality with (a) the 3D verticality detection kernel based on (b) Laplacian of Gaussian function modelled by the standard deviation σ and the scale s .

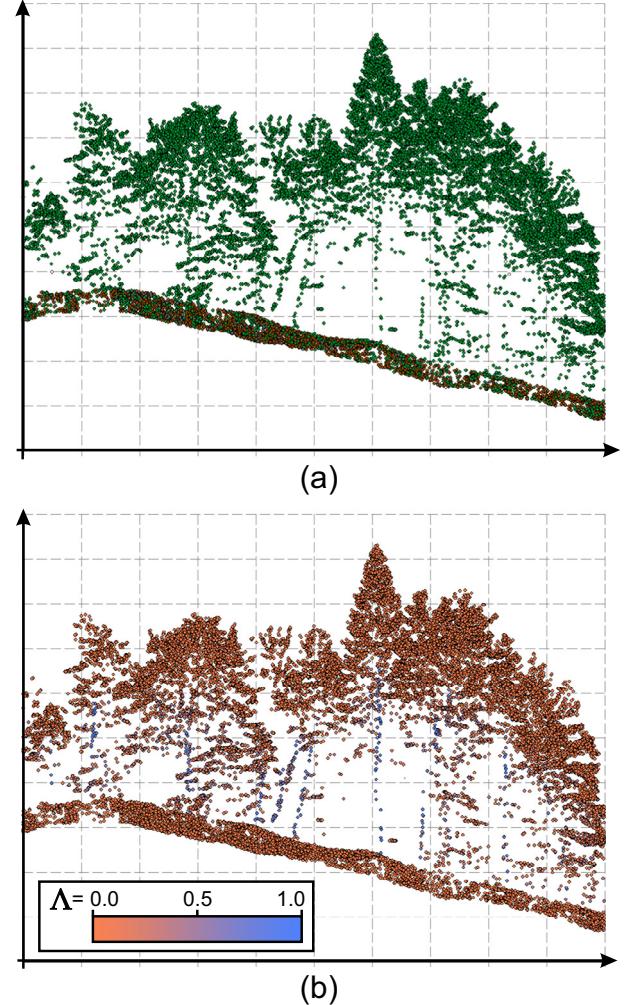


Fig. 5. Estimation of points' verticalities on (a) the input LiDAR dataset, where (b) the likelihood of each vertex v_i (and corresponding LiDAR point) being on a vertical line is estimated (points with low verticality are represented by orange and points with high verticality are blue).

probability of v_i being along a vertical line is decreased. Accordingly, these points should be weighted negatively. This leads to the conclusion that the Laplacian of Gaussian function, as shown in Fig. 4, is appropriate for modelling the probability of v_i being a tree trunk point in regards to its neighbourhood S_i^s , while its efficient estimation can be achieved by convolution. Let $\Delta X_{ij} = X[v_i] - X[v_j]$ and $\Delta Y_{ij} = Y[v_i] - Y[v_j]$, we define the convolution at v_i with the Laplacian of Gaussian kernel as a mapping $\Lambda : V(G_3) \rightarrow \mathbb{R}$, given by

$$\Lambda_\sigma[v_i] = \sum_{v_j \in V(S_i^s), N[v_j] \geq 1} \frac{1}{\pi \sigma^4} \left[1 - \frac{\Delta X_{ij}^2 + \Delta Y_{ij}^2}{2\sigma^2} \right] e^{-\frac{\Delta X_{ij}^2 + \Delta Y_{ij}^2}{2\sigma^2}}, \quad (16)$$

where σ is, in our case, the standard deviation used for modelling Λ and s defines the scale of the neighbourhood being observed. In other words, Λ is the sum of the values of the Laplacian of Gaussian at $(\Delta X_{ij}, \Delta Y_{ij})$, where $N[v_i], N[v_j] \geq 1$. As the range of Λ depends on the size of the neighbourhood s and the standard deviation σ used for its definition, a further normalisation step complementing Eq. (16) is used to map the values of $\Lambda[v_i]$ within the range $[0, 1]$.

As shown in Fig. 5, Λ achieves high contrast between tree trunks and branches, where thresholding can be applied

straightforwardly. However, the optimal threshold value t_A depends on the data, in particular, the canopy penetration-rate achieved during its acquisition and the resulting number of actual tree trunk-points. Generally, $t_A = 0.5$ should be sufficient as it thresholds those vertices that are, according to A , more likely to be tree trunks than branches. However, a slightly lower value of \hat{t}_A is used in our case for efficiently detecting tree trunks even at low-penetration rates, while additional filtering is used for the compensation. As the major cause for false-positives are branches at the canopy layer where they are often vertically skewed, they can be successfully avoided by inspecting their context. Namely, as tree trunks are higher, they are defined by a larger number of vertical vertices, while branches usually appear in bundles and are, therefore, often surrounded by horizontal ones. Given the neighbourhood S_i^s of a vertex v_i , efficient context-based filtering can be achieved by extracting two sets of vertices $\tilde{S}_i^{s,t_A}, \hat{S}_i^{s,t_A} \subseteq V(S_i^s)$, such that $\forall v_j \in \tilde{S}_i^{s,t_A}, N[v_j] \geq 1$ and $A[v_j] \geq t_A$, while $\forall v_k \in \hat{S}_i^{s,t_A}, N[v_k] \geq 1$ and $A[v_k] < t_A$. Thus, \tilde{S}_i^{s,t_A} is a set of vertices that are more likely to be tree trunks and \hat{S}_i^{s,t_A} is a set of candidates for branches. If there are more tree trunk- than branch-candidates, a given neighbourhood is more likely to contain a tree trunk and vice versa. Formally, a graph of thresholded tree trunks T_A^{s,t_A} is defined by

$$\begin{aligned} V(T_A^{s,t_A}(G_3)) &= \{v_i \in V(G_3) : N[v_i] \geq 1, A[v_i] \geq t_A, \text{ and } |\tilde{S}_i^{s,t_A}| \geq |\hat{S}_i^{s,t_A}|\}, \\ E(T_A^{s,t_A}(G_3)) &= \{e_{ij} \in E(G_3) : v_i, v_j \in V(T_A^{s,t_A}(G_3))\}. \end{aligned} \quad (17)$$

3.5. Voxel-space segmentation

In the last step of the proposed method, 3D delineation of single tree-crowns is achieved. We rely on LiDAR point density for this purpose, as tree-crowns are characterised by dense branches and leaves at their top-outer sides, where (due to the geometry of LiDAR data acquisition) the highest number of points is expected. Here, we propose density-based point clustering using watershed segmentation of a grid G_3 (i.e. voxel-space). By denoting the number of LiDAR points corresponding to a given $v_i \in G_3$ as $N[v_i]$ (see Section 3.1 for details), we define point density as a mapping $D : G_3 \rightarrow \mathbb{R}$, given by

$$D[v_i] = \frac{|\{v_j \in V(S_i^s) : N[v_j] \geq 1\}|}{|V(S_i^s)|}, \quad (18)$$

where S_i^s defines the observed neighbourhood. In simple terms, D estimates that portion of the neighbourhood S_i^s of v_i , occupied by LiDAR points, while local maxima of D correspond to the centres of points' clusters. Still, applying watershed directly on D obviously results in oversegmentation because not all of the clusters are actual tree crowns. For example, some isolated tree trunks may be segmented as separate watershed regions because they are surrounded by vertices of zero density. On the other hand, dense clusters that often appear in the conjunction areas of neighbouring tree crowns may cause undersegmentation (see Fig. 6a). In order to deal with these issues, merging of clusters is achieved first by using tree trunks (detected by $T_A^{s,t_A}(G_3)$) for defining the markers of the watershed segmentation, while canopy layer segmentation (achieved by $T_{CHM}^{t_H, \hat{t}_A, \hat{t}_B, \hat{t}_{SC}}(G_2)$) is used to prevent overspreading of regions and dividing conjunction areas in the canopy layer. As illustrated in Fig. 6, this is achieved over the following three steps:

- New markers are added first in those cases where tree trunks were unsuccessfully detected but the corresponding tree-crowns within the canopy-layer are well-defined by

$T_{CHM}^{t_H, \hat{t}_A, \hat{t}_B, \hat{t}_{SC}}(G_2)$. This is especially apparent when processing low-density data or when achieving low canopy penetration-rate (see Fig. 6a).

- Markers are then extended in order to connect those tree trunks detected in fragments as shown in Fig. 6b. As each connected component of markers defines a distinct watershed region, oversegmentation is prevented in this way.
- Finally, 3D delineation of single tree crowns is achieved by marker-controlled watershed segmentation, using boundaries of regions from canopy layer segmentation $T_{CHM}^{t_H, \hat{t}_A, \hat{t}_B, \hat{t}_{SC}}(G_2)$ to limit the extends of each region. As shown in Fig. 6c, conjunction areas of neighbouring trees in canopy layer are successfully separated in this way.

Given a delineated single tree-crown defined by connected-component $C_i^2 \in T_{CHM}^{t_H, \hat{t}_A, \hat{t}_B, \hat{t}_{SC}}(G_2)$, the number of tree trunks lying within C_i^2 is estimated first. This is defined by the number of connected-components of the intersection between C_i^2 and the projection $proj_2(T^{s,t_A}(G_3))$ of $T^{s,t_A}(G_3)$ to 2D. Thus, a tree with an undetected trunk is straightforwardly recognised by a connected-component C_i^2 , such that $C_i^2 \cap proj_2(T^{s,t_A}(G_3)) = \emptyset$. In this case, the undersegmentation can successfully be avoided by adding a vertex to the graph of thresholded tree trunks $T_A^{s,t_A}(G_3)$. As a tree trunk is more likely to be below the centre of the tree-crown, a vertex of maximal density $\check{v}_i \in V(G_3)$, such that $\forall v_j, proj_2(v_j) \in C_{proj_2(\check{v}_i)}^2$ implies that $D[v_j] \leq D[\check{v}_i]$ is used for this purpose. Formally,

$$V(T_A^{s,t_A}(G_3)) = V(T_A^{s,t_A}(G_3)) \cup \{\check{v} \in G_3 : C_{proj_2(\check{v}_i)}^2 \cap proj_2(T^{s,t_A}(G_3)) = \emptyset\}, \quad (19)$$

where connected-component $C_{proj_2(\check{v}_i)} \in T_{CHM}^{t_H, \hat{t}_A, \hat{t}_B, \hat{t}_{SC}}(G_2)$ is identified by the projection of \check{v}_i . Thus, $\delta(T_A^{s,t_A}(G_3))$ now ensures that each single tree, delineated by $T_{CHM}^{t_H, \hat{t}_A, \hat{t}_B, \hat{t}_{SC}}(G_2)$, contains at least one marker (seen Fig. 6a). This prevents undersegmentation while markers are extended (as seen in Fig. 6b) in order to avoid oversegmentation. Let sets of vertices above and below a given $v_i \in G_3$, denoted as $V^\dagger(G_3, v_i)$ and $V^\perp(G_3, v_i)$ respectively, be defined as (see Fig. 7)

$$V^\dagger(G_3, v_i) = \{v_j \in V(G_3) : X[v_j] = X[v_i], Y[v_j] = Y[v_i], Z[v_j] \geq Z[v_i]\}, \quad (20)$$

$$V^\perp(G_3, v_i) = \{v_j \in V(G_3) : X[v_j] = X[v_i], Y[v_j] = Y[v_i], Z[v_j] < Z[v_i]\}. \quad (21)$$

The vertex of maximal density above given v_i is defined as $\check{v}_i^\dagger \in V^\dagger(G_3, v_i) : \forall v_j \in V^\dagger(G_3, v_i) \rightarrow D[\check{v}_i^\dagger] \geq D[v_j]$. When two or more vertices of maximal density exist, the one with the largest Z -value is selected. The set of dilated tree trunks $\delta(T_A^{s,t_A}(G_3), D)$ is then obtained by

$$V(\delta(T_A^{s,t_A}(G_3), D)) = \bigcup_{v_i \in T_A^{s,t_A}(G_3)} V^\perp(G_3, \check{v}_i^\dagger), \quad (22)$$

$$E(\delta(T_A^{s,t_A}(G_3), D)) = \{e_{ij} : v_i, v_j \in V(\delta(T_A^{s,t_A}(G_3), D))\}.$$

Note that δ , as defined by Eq. (22), is not morphological dilation. Although it could be defined as such (by constructing a marker-graph below vertices of maximal density and applying binary reconstruction by dilation on $T_A^{s,t_A}(G_3)$ with a vertical structuring element), a simpler definition is used here for clarity. Moreover, by defining tree trunks straightforwardly as a union of sets of vertices below the vertex of maximal density \check{v}_i^\dagger above a given $v_i \in T_A^{s,t_A}(G_3)$ and the corresponding edges, we avoid construction of a marker-graph. Oversegmentation is sufficiently prevented in this way, as all the fragments of the tree trunk are merged. At the same time, tree trunks that are isolated by zero-vertices are extended to the clusters of best candidates for the corresponding

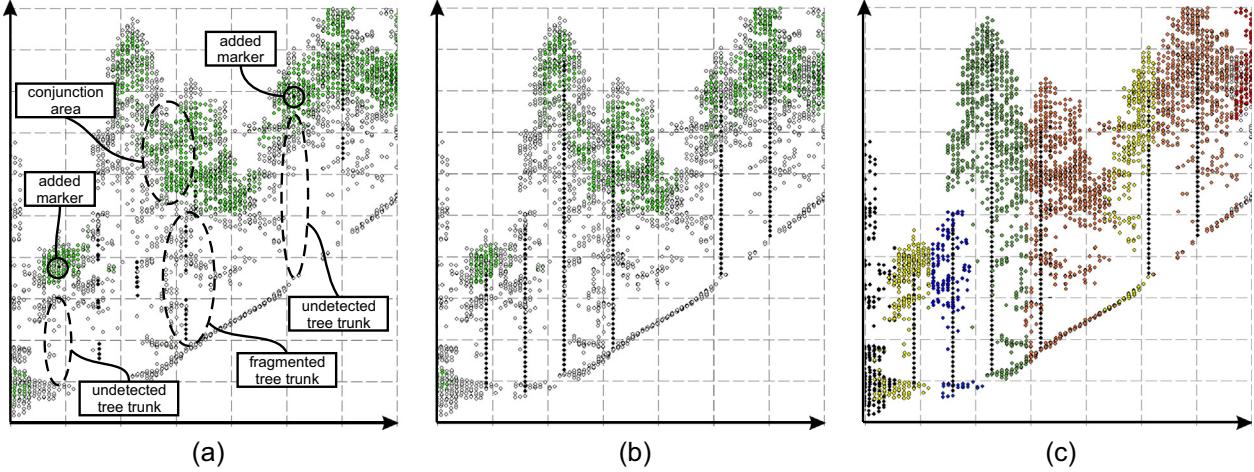


Fig. 6. Single tree-crown delineation in voxel-space based on point-density (low-density points are marked white and high density points are green), where (a) detected tree trunk-points (marked black) are extended to form (b) a set of markers for (c) watershed segmentation.

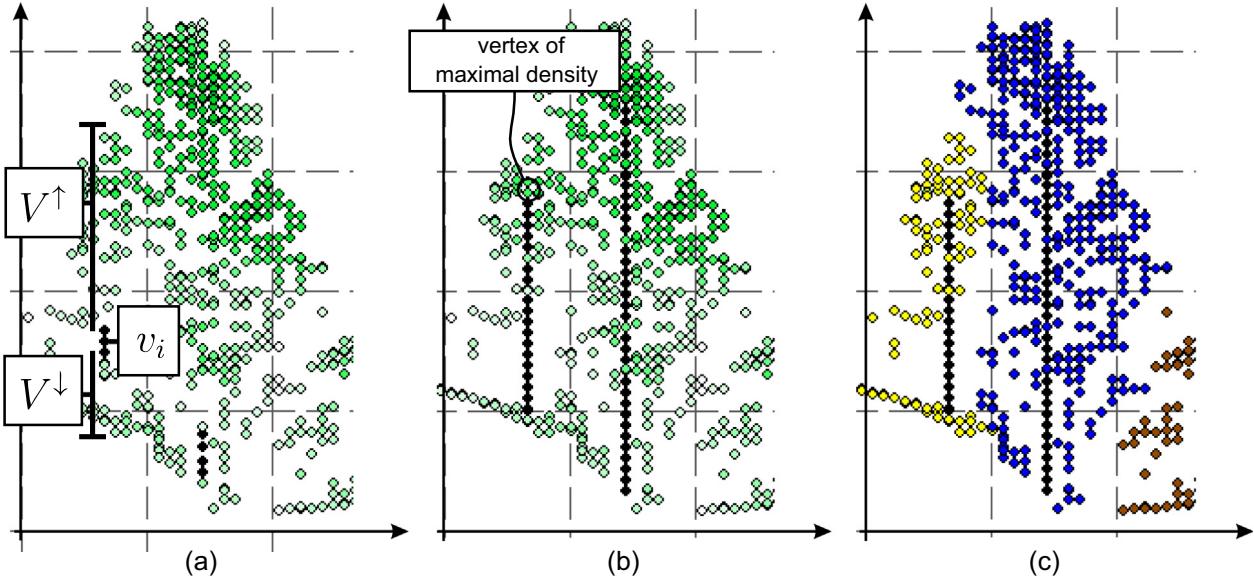


Fig. 7. Extended markers (marked black) based on (a) the definition of $V^{\uparrow}(G_3, v_i)$ and $V^{\downarrow}(G_3, v_i)$, where (b) vertex of maximal density above a given v_i is used for improved (c) 3D delineation single tree crowns.

tree-crowns. Finally, single-tree delineation in G_3 can be defined by the watershed based on the density D , as seen in Fig. 6c. Let an edge-weighted function $W_D : G_3 \rightarrow \mathbb{R}$, given as

$$W_D[e_{ij}] = \begin{cases} 1 & \text{if } proj_2(v_i) \in B(C_{proj_2(v_i)}) \vee proj_2(v_j) \in B(C_{proj_2(v_j)}), \\ \wedge & \{-D[v_i], -D[v_j]\} \text{ otherwise.} \end{cases} \quad (23)$$

Note that the condition $W_D[e_{ij}] = 1$ if $proj_2(v_i) \in B(C_{proj_2(v_i)})$ or $proj_2(v_j) \in B(C_{proj_2(v_j)})$ prevents overspreading of watershed regions beyond the boundaries of delineated tree-crowns within the canopy layer. A single tree delineation is finally achieved by estimating a minimum spanning tree $MSC(G_3, W_D)$, as shown in Fig. 6c.

Fig. 6c demonstrates successful 3D delineation of single tree crowns, achieved by the proposed method. Orange coloured trees in Fig. 6c were detected during the segmentation of the canopy layer based on LoFS and their corresponding tree trunks were also successfully found. Thus, tree trunk detection did not affect their delineation. Similarly, the delineation of the yellow coloured trees

was successfully achieved during the segmentation of the canopy layer, however, their corresponding tree trunks were undetected. In order to prevent their merging with neighbouring trees, additional markers were introduced at vertices of maximal density below their corresponding treetops (see Fig. 6a), as defined by Eq. (19). On the other hand, the blue coloured tree was not delineated during the segmentation of the canopy layer as it is covered by the green coloured tree. Still, by detecting its trunk, accurate delineation of blue coloured tree was achieved during the segmentation of voxel-space.

4. Results

4.1. Ground truth data

The proposed approach was evaluated on six different LiDAR datasets. Although they were all acquired on the outskirts of the Slovenian Alps, they differed according to the forest type, terrain

configuration, and data density, as seen in [Appendix A](#). In all the cases, except for **DS3**, the forest was managed. Details on test datasets are shown in [Table 1](#).

Within these datasets, 11 test areas were selected where ground truth measurements were acquired. The centres of each test area were defined first and the positions of the trees within a radius of 12 m were measured.

In total, the positions of 232 trees (approximately half of them coniferous and half of them deciduous) were sampled using *Garmin GPSMAP 60CSx*. The number of measured trees per data is shown in [Table 1](#). The column “total” refers to the total number of measured trees, while the column “canopy” gives the number of trees present within the canopy layer of each dataset. In the following validation, the total number of trees is used for evaluating the efficiency of the proposed method, while the number of trees present in the canopy layer is reported only as a complementary information about the proportions of canopy and understory trees. The accuracy of ground truth data was determined by measuring the referenced objects (such as buildings, chapels, and crossroads close to the test areas). The estimated average error was approximately 2 m in all the test cases with the standard deviation equal to 0.5 m. As the errors were normally distributed, this meant that more than 99% of the ground truth tree trunks were, according to 68–95–99.7 rule ([Mann, 2010](#)), within the 3.5 m range from their estimated positions. Note that only planimetric coordinates of trees were measured, without measuring their heights. In view of this, we were able to estimate only the number of trees within the canopy layer by visual inspection of LiDAR data and supplementary orthophoto images. Moreover, planimetric distance measurements were used during the following validation procedure, although 3D measurements would have been more appropriate.

4.2. Validation procedure

In the first step of the validation procedure, detected trees within the test areas were extracted first by thresholding them based on their distances from the closest centres of the test areas. These were then matched by linking them with the closest ground truth positions of tree trunks. Thus, each estimate was matched with exactly one ground truth tree trunk, however, ground-truth tree trunks may be matched with zero or more estimates. Accordingly, the following measurements were obtained:

- *tp* are true positives, defined by the number of ground truth trees that were detected. Note that a ground truth tree is considered to be detected if there is at least one estimated position within a 3.5 m range from it.
- *fp* are false positives. These occur when a given ground truth tree is matched with more than one estimate. Amongst them, one is considered to be *tp* (providing it satisfies the previous condition), while the rest are assigned to *fp*.
- *fn* are false negatives, defined by the number of ground truth trees that were not matched.

Table 1
Description of test datasets.

Dataset	Number of test areas	Data density [p/m ²]	Terrain type	Vegetation type	Number of trees	
					Total	Canopy
DS1	2	57.3	Hill	Mixed	50	30
DS2	1	25.6	Valley	Coniferous	28	17
DS3	4	51.2	Flat	Deciduous	109	74
DS4	2	63.3	Valley	Coniferous	56	31
DS5	1	47.1	Hill	Mixed	15	8
DS6	1	96.6	Flat	Mixed	20	11

Accordingly, the following statistical measurements of the performance were used ([Mann, 2010](#)):

- **Recall** or true positive rate *tpr* describing the rate of the detected trees by:

$$tpr = tp / (tp + fn). \quad (24)$$

- **Precision** or positive predictive value *ppv* describing the rates of correct detections. Its definition being given as

$$ppv = tp / (tp + fp). \quad (25)$$

- **f1-score** is the harmonic mean of precision and recall, given by

$$f1 - score = 2tp / (2tp + fp + fn). \quad (26)$$

4.3. Validation

4.3.1. Sensitivity analysis

The proposed method relies on nine input parameters. In order to estimate their optimal values, a sensitivity analysis was performed first. A two stage approach has been used for this purpose, where the input parameters were intuitively defined first, while so-called “one-at-a-time” approach ([Czitrom, 1999](#)) has been adopted for their systematic tuning. In this way, we defined baseline values of grid resolution as $r = 0.5$ m, observed neighbourhood size as $s = 1.5$ m, curvature threshold as $\hat{t}_M = 0.01$, edge height threshold as $\hat{t}_H = 0.1$ m, minimal area threshold as $\hat{t}_A = 5$ m², maximal area threshold as $\hat{t}_{A'} = 25$ m², shape compactness threshold as $\hat{t}_{SC} = \pi$, standard deviation of LoG function as $\sigma = 0.5$, and verticality threshold as $\hat{t}_V = 0.4$. Further, their effects on the performances of the proposed method are studied by changing one parameter at a time, while keeping all others at their baseline values. The obtained results are shown in [Fig. 8](#).

Resolution of grids, given by r , obviously had the largest impact on the efficiency of the proposed method. Namely, with increasing resolution ($r < 0.5$ m in our case), more points lying below the top surface (i.e. mid- and understory or even ground points) were added to the definition of DSM. The final CHM therefore became rough, leading to oversegmentation. On the other hand, tree crowns have often been absorbed by their dominant neighbours when low resolutions were used ($r > 0.5$ m in our case). Thus, the optimal resolution is bounded by the point-density on one side and the geometrical features of tree crowns on the other, where the results show $r = 0.5$ m is optimal in our case (see [Fig. 8a](#)). Similarly, geometric features of trees are important for achieving optimal definitions of several other parameters. As the tree tops are defined by those neighbourhoods where CHM is of the convex shape (see Eq. (4)), the optimal size of the observed neighbourhood s was inevitably related to the sizes of the tree crowns within the datasets. Consequently, some distinct peaks caused by strong branches were detected as separate tree crowns in canopy layer segmentation, when using $s < 1.5$ m. As s directly correlates with the number of vertically distributed points needed for tree trunk detection, the proposed method also had difficulties in distinguishing between them and the vertical branches. In addition, the accuracy of the proposed method became gradually lower when using $s > 1.5$ m, as increasing number of vertically distributed points were required for detecting a tree trunk. Thus, $s = 1.5$ m is an optimum in our case, as confirmed by [Fig. 8b](#). Complementary to s , the curvature threshold \hat{t}_M used in the definition of treetops allowed for absorbing small variation in CHM. This means that with the increasing value of \hat{t}_M the set of markers used for canopy layer segmentation was gradually reduced, simultaneously affecting recall as well as the precision of the proposed method. Namely, when using $\hat{t}_M < 0.01$, undersegmentation was achieved as closely

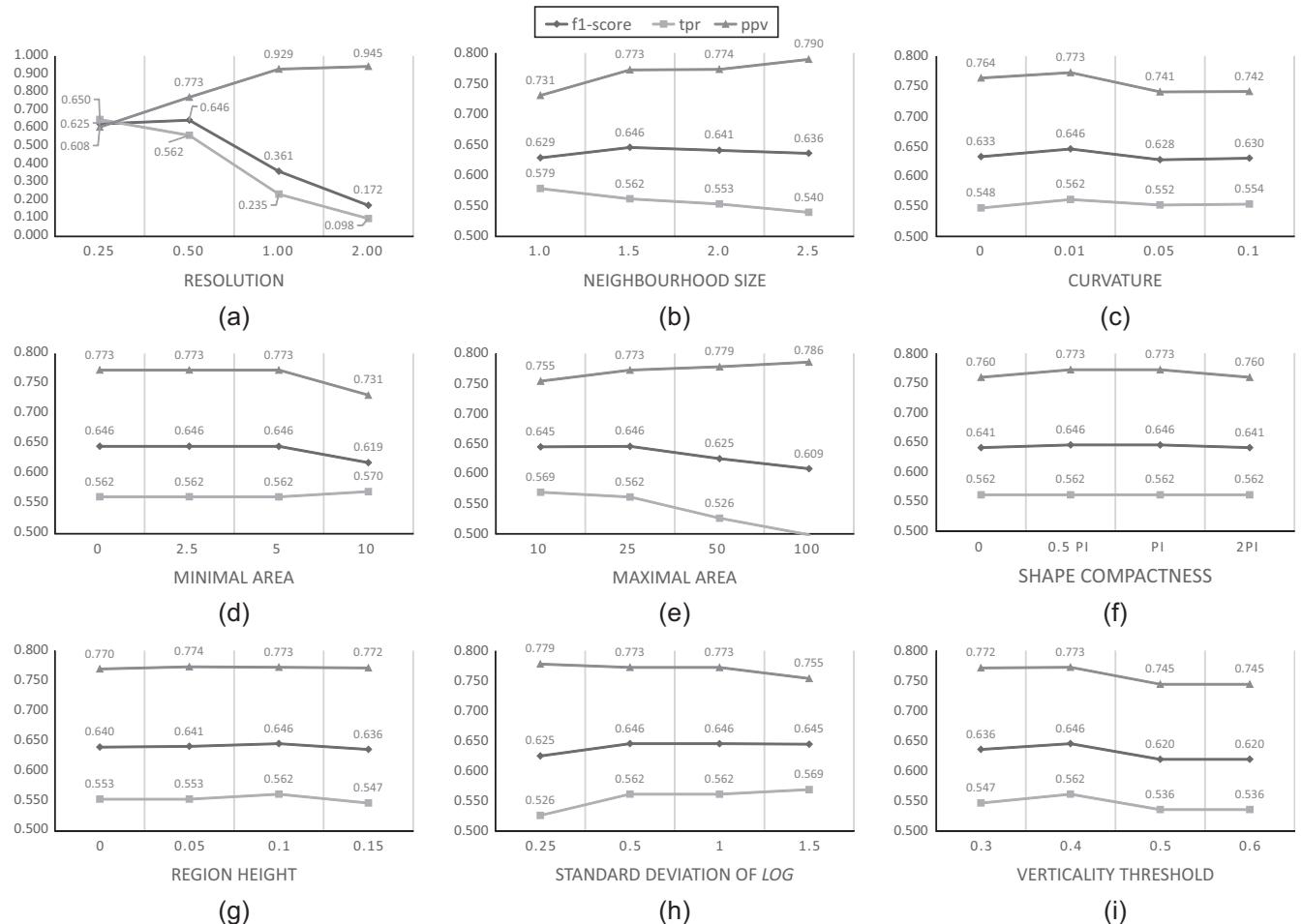


Fig. 8. The effects caused by changing (a) resolution r , (b) observed neighbourhood size s , (c) marker curvature threshold \check{t}_M , (d) minimal area threshold $\check{t}_A = 5 \text{ m}^2$, (e) maximal area threshold $\check{t}_A = 25 \text{ m}^2$, (f) shape compactness threshold \check{t}_{SC} , (g) region height threshold \check{t}_H , (h) standard deviation for LoG denoted by σ , and (i) verticality threshold for stem detection \check{t}_A .

connected treetops were detected as a single region, while small variations in *CHM* caused by strong branches were the reason for the oversegmentation. On the other hand, the oversegmentation was more critical when $\check{t}_M > 0.01$ was used, as the treetop markers broke into smaller regions. The optimal trade-off was in our case defined by $\check{t}_M = 0.01$ (see Fig. 8c).

Other parameters related to the geometry of the trees are used in the canopy layer segmentation. While minimal and maximal area thresholds, respectively given by \check{t}_A and \check{t}_{A1} , defined the extent to which region merging was achieved, their effects on the results are opposite. Obviously, increasing \check{t}_A or decreasing \check{t}_{A1} results in less merging occurring and vice versa. In our case, a suitable trade-off has been defined by $\check{t}_A = 5.0 \text{ m}^2$ and $\check{t}_{A1} = 25.0 \text{ m}^2$, where the largest f -scores were measured (see Figs. 8d and e). Similarly, the extent to which region merging occurs, can be limited by shape compactness threshold \check{t}_{SC} . However, the effects of \check{t}_{SC} are significantly less obvious (see Fig. 8f), as it only defines the level to which minimisation of the shapes' compactness is achieved (see Eq. (14)). In contrast to the parameters discussed earlier, the optimal value of the height threshold \check{t}_H used for region merging is related to the distribution of trees rather than to their geometric properties. As shown in Fig. 8g, the optimal value was in our case given by $\check{t}_H = 0.1 \text{ m}$ due to clusters of dense trees, while higher threshold value could be used when dealing with trees that are further apart in order to increase the number of merging and, consequently, increase the method's precision.

Finally, tree trunk detection is controlled by the standard deviation of LoG defined by σ and with the verticality threshold \check{t}_A . While the optimal definition of σ is related to the skewness of tree trunks and their inclines, the definition of r was considered when estimating its value. In our case, optimal results were obtained using $\sigma = 0.5$ or $\sigma = 1.0$ (see Fig. 8h), indicating that most of the measured trees grow perpendicularly to the ground. On the other hand, due to the low canopy penetration rate achieved in most of the cases, the verticality threshold $\check{t}_A = 0.4$ was optimal (see Fig. 8i). As tree trunks are represented with the low number of the vertically distributed points, the recall was significantly decreased when using $\check{t}_A > 0.4$, while applying $\check{t}_A < 0.4$ resulted in a decrease of precision.

With the exception of resolution r , the achieved f -scores were within the range [0.62, 0.65] in all the discussed cases, indicating the robustness of the proposed method against the non-optimal definition of input parameters. However, by comparing the lowest and the highest f -scores achieved during this analysis, tuning of the input parameters may contribute up to 5% to the efficiency of the method. The optimal results are visually presented in Appendix A.

4.3.2. Comparison of segmentation accuracy

This section provides the validation of the proposed method. The proposed approach for delineation of canopy layer based on LoFS was first compared with traditional approach relying on the

local maxima-based definition of the watershed markers, while contributions of the understory segmentations based on tree trunk detection were analysed later. Except when specifically stated otherwise, the same set of input parameters as defined by the sensitivity analysis (see Section 4.3.1) was used. As the compared method relies on smoothing in order to cope with oversegmentation, different smoothing levels (i.e. $\sigma \in \{0.5, 0.75, 1.0, 1.25, 1.5\}$) were applied. The obtained results are shown in Table 2, where the optimal results in regards to $f1 - score$ achieved in each particular case are marked as bold.

Table 2 shows that the optimal smoothing level in terms of $f1 - score$ depends on the dataset being processed. On average, the higher $f1 - score$ values were achieved when $\sigma=0.75$ and the values of tpr and ppv were balanced (see Fig. 9a) as σ positively correlated with ppv , while its correlation with tpr was negative. Although only the higher $f1 - scores$ achieved by the compared method were used in further comparisons, Table 3 shows significant improvement on the results with the proposed LoFS-based definition of markers.

Similarly to the traditional approach, Table 4 shows that the results of the proposed method had no correlation with the terrain or the vegetation types. However, both methods performed slightly better in the cases of DS1, DS2, and DS3. Closer inspection of these datasets revealed that there were less trees present within the understory layers and, thus, the greater percentages of trees could be detected by analysing CHM alone (see Table 1).

By using the LoFS-based definition of treetops, higher $f1 - score$ values were obtained in all the cases due to the significant increase in ppv . The proposed approach was, therefore, less prone to oversegmentation, while both methods achieved comparable tpr in most of cases, except for DS1 and DS6. Although a considerably higher tpr value was achieved by the traditional approach in the case of DS6, the significantly lower ppv value clearly indicated oversegmentation occurring in this particular case. A similar but less severe example was also DS1. The reasons for lower average tpr achieved by the proposed method could, therefore, be attributed to the limitations of the validation procedure. Namely, due to the lack of tree heights, 2D distances had to be used for matching and overdetected trees from the canopy layer may have been matched coincidentally with the understory trees because of this. However, as the proposed method is less prone to oversegmentation, this did not significantly affect the results of region merging and tree trunk detection steps, as shown in Table 4.

Table 4 demonstrates the effectiveness of the proposed region merging and tree trunk detection for improving the results of the proposed method. As shown in Fig. 9b, region-merging could not

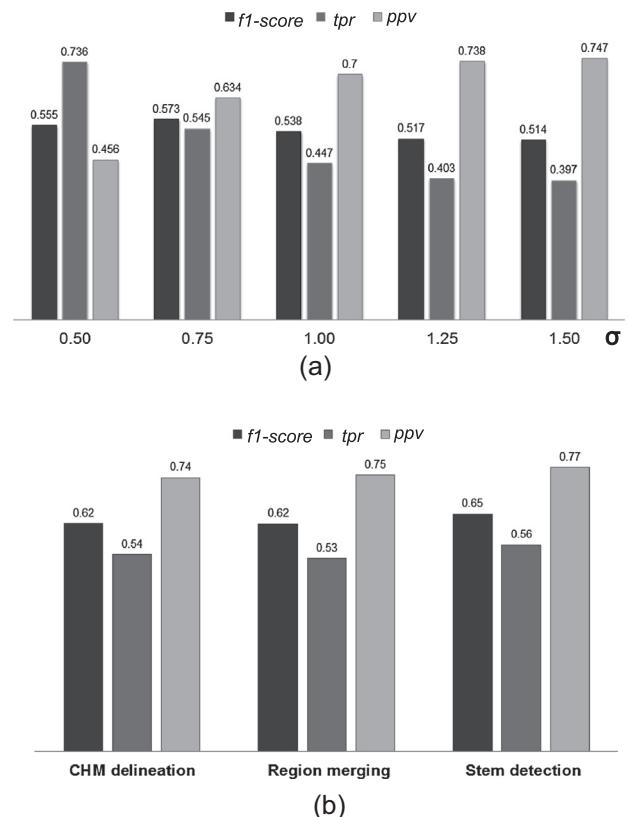


Fig. 9. The accuracy of single tree-crown extraction of (a) the compared method at different smoothing levels and (b) the progressive increase in the efficiency achieved by the proposed method.

Table 3
Single tree-crown detection in canopy layer-based on LoFS.

	The proposed method			The traditional approach		
	f1-score	tpr	ppv	f1-score	tpr	ppv
DS1	0.659	0.600	0.732	0.662	0.660	0.650
DS2	0.652	0.536	0.833	0.571	0.500	0.667
DS3	0.647	0.514	0.875	0.615	0.514	0.767
DS4	0.587	0.482	0.750	0.577	0.500	0.683
DS5	0.593	0.533	0.667	0.560	0.467	0.700
DS6	0.579	0.550	0.611	0.531	0.850	0.386
AVG	0.620	0.536	0.745	0.586	0.582	0.642

Table 2

Single tree-crown delineation based on the local maxima of smoothed CHM at levels $\sigma \in \{0.5, 0.75, 1.0, 1.25, 1.5\}$.

σ	DS1			DS2			DS3		
	f1-score	tpr	ppv	f1-score	tpr	ppv	f1-score	tpr	ppv
0.50	0.503	0.860	0.355	0.580	0.714	0.488	0.602	0.679	0.540
0.75	0.627	0.740	0.544	0.571	0.500	0.667	0.615	0.514	0.767
1.00	0.662	0.660	0.650	0.558	0.429	0.800	0.593	0.468	0.810
1.25	0.619	0.520	0.765	0.571	0.429	0.857	0.594	0.450	0.875
1.50	0.612	0.520	0.743	0.537	0.393	0.846	0.588	0.431	0.922
MAX	0.662	0.660	0.650	0.571	0.500	0.667	0.615	0.514	0.767
	DS4			DS5			DS6		
0.50	0.558	0.643	0.493	0.556	0.667	0.476	0.531	0.850	0.386
0.75	0.577	0.500	0.683	0.560	0.467	0.700	0.489	0.550	0.440
1.00	0.482	0.357	0.741	0.560	0.467	0.700	0.375	0.300	0.500
1.25	0.436	0.304	0.773	0.560	0.467	0.700	0.323	0.250	0.455
1.50	0.462	0.321	0.818	0.560	0.467	0.700	0.323	0.250	0.455
MAX	0.577	0.500	0.683	0.560	0.467	0.700	0.531	0.850	0.386

Table 4

Improved detection of single trees by region-merging and tree trunk detection.

	Region merging			Tree trunk detection (without region merging)			Tree trunk detection (with region merging)		
	f1-score	tpr	ppv	f1-score	tpr	ppv	f1-score	tpr	ppv
DS1	0.659	0.600	0.732	0.640	0.620	0.675	0.659	0.600	0.732
DS2	0.652	0.536	0.833	0.667	0.536	0.882	0.667	0.536	0.882
DS3	0.647	0.514	0.875	0.617	0.523	0.784	0.644	0.514	0.862
DS4	0.587	0.482	0.750	0.626	0.554	0.745	0.645	0.536	0.811
DS5	0.593	0.533	0.667	0.593	0.533	0.667	0.593	0.533	0.667
DS6	0.579	0.550	0.611	0.642	0.650	0.654	0.667	0.650	0.684
AVG	0.620	0.536	0.745	0.631	0.569	0.735	0.646	0.562	0.773

detect any additional trees and, thus, it is used only for improving the precision of the tree extraction. Consequently, the *tpr* value could only be decreased during this step while the *ppv* value improved. More precisely, due to the possibility of mismatching over-detected trees from the canopy layer with the understorey trees, the *tpr* value is likely to be decreased during this step. Consequently, maintaining *f1 – scores* at the same level while increasing the *ppv* value, as seen by comparing the results from [Tables 3 and 4](#), was more or less sufficient. On the other hand, tree trunk detection could be seen as a complementary operation where additional tree crowns were delineated when more than one tree trunk was detected within the corresponding region from *CHM* delineation. Consequently, the *tpr* value could only increase, while *ppv* decreased if overdetection occurred (see [Fig. 9b](#)). Thus, region-merging could be understood as a counterbalance to tree trunk detection that essentially leads to improved overall *f1 – scores* (see average *f1 – scores* achieved after tree trunk detection with and without region merging in [Table 4](#)).

The proposed approach for the tree trunk detection significantly increased *f1 – scores* in all cases except for **DS2**. On the other hand, the highest increase in *f1 – scores* was achieved in the case of **DS6** where the point density was higher than the rest of the datasets. Additional tests were therefore performed in order to evaluate the performance of the proposed method in regards to the point-density.

4.3.3. Influences of point-density

For evaluating the effects of different point-densities on the performance of the proposed method, test datasets from [Table 1](#) were decimated by factors of 2, 4, 8, and 16, i.e. from each test dataset only each second, fourth, eighth, and sixteenth points being selected and all the rest removed. Note that when using decimation factors 8 and 16, optimal results were achieved at increased resolution $r = 1.0m$.

As expected, [Fig. 10](#) shows gradual decreases for all efficiency indicators when increasing the decimation factor. However, only a slight reduction in the efficiency of LoFS-based *CHM* analysis was noticed when the average point density was decreased from 56.85 to 3.56 points per square metre (see [Fig. 10a](#)). This confirmed that the proposed LoFS based definition of markers is well-capable of absorbing the noise within the data and is adapted for dealing with the anisotropic shapes of tree-crowns of different sizes. A larger decrease in *ppv* in comparison to *tpr* indicated that oversegmentation was slightly increased in this way due to the loss of geometric details that cause the concave regions within *CHM* to break-up. Typically, false convex regions appear when no treetop point is contained within a given grid-cell and the value of *CHM* is, consequently, lower than the neighbourhood. On the other hand, the lowest data density used was still enough for adequate representation of trees, as most of the tree crowns were loosely clamped together and the convex boundary between them remained preserved. On the other hand, a greater decrease in the

efficiency was noticed when considering the tree trunk detection (see [Fig. 10](#)). Namely, tree trunks are usually poorly presented in airborne LiDAR as they are shaded by neighbouring branches. Reduction of data density had, therefore, a much more severe effect on their detection as critical geometric details were lost in this way. More precisely, while the average *tpr* value was at a similar level when decimation by the factor two was performed (i.e. from 56.85 to 28.43 points per square metre), a significant decrease was observed when the decimation factor was increased to four (i.e. 14.21 points per square metre) or even eight (i.e. 7.1 points per square metre), where the results became identical to the *CHM* analysis alone (see *tpr* curve in [Fig. 10](#)). It was, therefore, obvious that the efficiency of tree trunk detection was highly correlated with the point density as sufficient numbers of tree trunk points had to be present in order to recognise them.

5. Discussion

The proposed approach combines treetop and tree trunk detection for improved 3D single tree delineation in LiDAR data. In total, it relies on six user defined threshold values, where verticality and curvature thresholds are applied for tree trunk and treetop detection respectively, while minimal and maximal areas, minimal region height, and maximal shape compactness are used during the region merging process. In addition, user definition of the observed neighbourhood size is required. In our study, the same set of threshold values and the same neighbourhood size were used in all cases, indicating that the method does not require any significant recalibration when dealing with forests of different types. This was also confirmed by the sensitivity analysis, provided in [Section 4.3.1](#).

When comparing the performances of the methods throughout the results, we mostly refer to the *f1 – score* that is a harmonic mean between *tpr* and *ppv*. As for some applications one is more important than the other, we should not compromise between them and, if possible, try to increase them both. As shown in [Fig. 9](#), the proposed method achieves significantly higher *ppv* in comparison with the traditional approach. As the greater majority of the trees from the canopy layer were successfully detected (as concluded by comparing *tpr* values from [Table 3](#) with the number of trees within the canopy layer from [Table 1](#)), the reason for lower increase in *tpr* can be related to the tree trunk detection. Namely, as the proposed approach for tree trunk detection requires their geometries to be sufficient, it is to some extent limited during their extraction. In order to overcome this issue, full-waveform LiDAR systems can be used for gaining improved geometric descriptions by acquiring higher point densities, as demonstrated by [Reitberger et al. \(2009\)](#). Moreover, note that the proposed tree trunk detection serves only for defining a set of markers for density-based clustering (i.e. watershed on density function) and, thus, improving the accuracies of single tree crown delineations within the midstorey and understorey layers. When dealing with

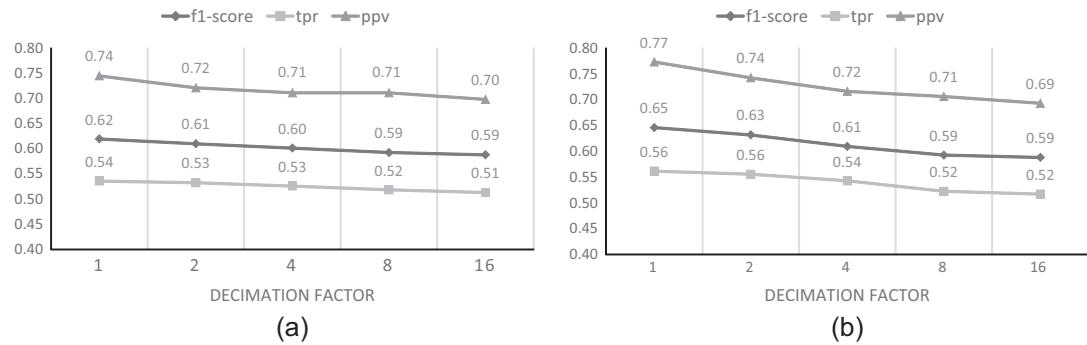


Fig. 10. Influence of point-density reduction on the average $f1 - score$, tpr , and ppv of (a) the proposed *LoFS* based definition of markers for CHM analysis alone and (b) CHM analysis together with the proposed tree trunk detection.

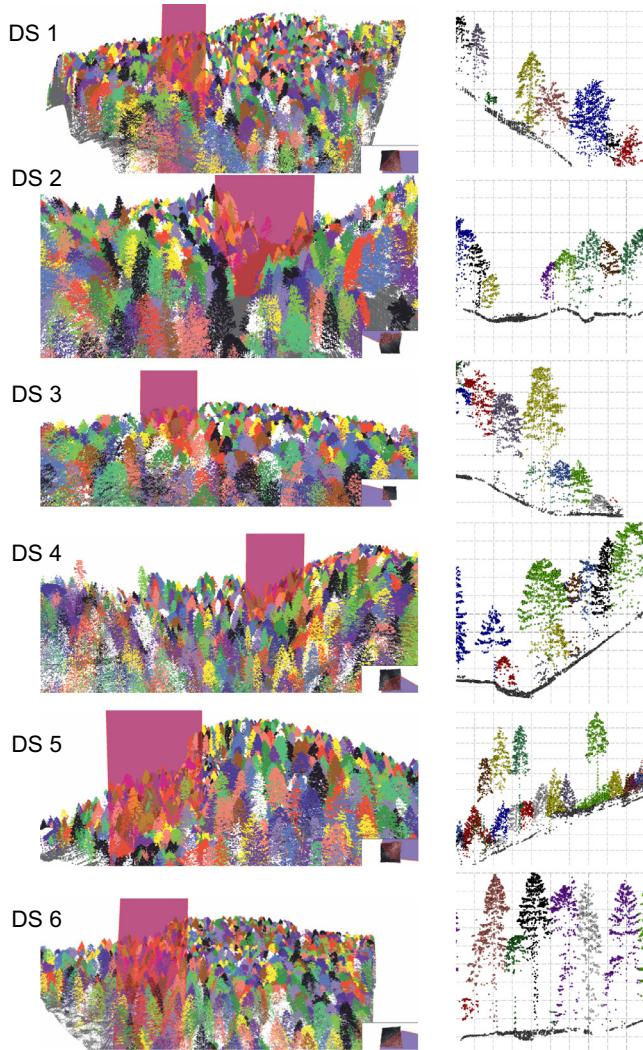


Fig. A.11. Test dataset **DS1**, **DS2**, **DS3**, **DS4**, **DS5**, and **DS6** with cross section planes (left), and visualisations of cross sections (right).

those particular cases, where tree trunks' geometries are obscured, a number of methods can be employed instead, where the RANSAC based approach as proposed by Reitberger et al. (2009), intensity based thresholding as proposed by Lu et al. (2014), or mean shift algorithm for defining the centres of tree crowns as proposed by Ferraz et al. (2012) are just some examples. These approaches should, therefore, be viewed as complementary techniques that

may improve the performance of the proposed method in the discussed cases.

Nevertheless, the point density required for tree trunk detection is still relatively high. As this may raise the issue of costs related to LiDAR data acquisition, storage, and processing, it is firstly important to notice that the computational efficiency of the proposed approach is relatively high. Namely, by subsampling the data into a regular grid and avoiding any iterative approaches, the proposed method achieves linear time complexity in regards to the number of vertices used for its graph-based representation. On the other hand, while full-waveform LiDAR systems may also provide an efficient approach for obtaining sufficiently high point-density without significant increase in the costs of data acquisition, large data storage is required for this purpose and it cannot be avoided for successive tree-trunk detection. However, as shown in Fig. 10, the proposed *LoFS*-based definition of markers for delineating single tree crowns within the canopy layer is capable of achieving relatively high extraction rates even when dealing with low density point-clouds.

6. Conclusion

This paper proposes several improvements on CHM analysis for single tree crown delineation and a new approach for tree trunk detection, while a unified framework is achieved for exploiting their complementaries. The main improvement on CHM-based single tree-crown delineation is achieved by *LoFS*-based definitions of markers that are better adapted to dealing with the various sizes and anisotropic shapes of tree-crowns. Furthermore, watershed definition based on minimum spanning forest allows for straightforward implementation of region-merging, leading to improved precision of tree crown delineation within canopy layers. Although this does not affect the overall performances of the proposed approach in itself, it still significantly contributes to the improved final results achieved by the tree trunk detection. For this purpose, a new estimation of verticality regarding points' distributions is proposed based on convolution with the Laplacian of Gaussian kernel, the shape of which accurately models the probability of a point being a tree trunk in regards to the positions of points from its neighbourhood. As confirmed by the results, this leads to an approximately 10% increase in the average $f1 - score$ (from 0.59 to 0.64 in absolute values) in regards to the compared method, where improved segmentation of the canopy layer contributes approximately 6% and the rest is achieved by delineating the mid- and understory layers based on tree trunk detection. While both method achieve comparable average recall, the precision of the proposed method is significantly higher. The main advantage of the proposed method is, therefore, its robustness against oversegmentation while it is still capable of achieving sufficient

detection rate due to the delineation of the understorey layer based on the tree trunk detection. Although the tree trunk detection and the delineation of the mid- and understorey layers depend on the point density of LiDAR data, we have shown that sufficient accuracy of single tree delineation achieved within in the canopy layer even when dealing with low density datasets. Sensitivity analysis showed that the proposed method is also robust against non-optimal definitions of input parameters.

Acknowledgements

This work was supported by the Slovenian Research Agency under Grants J2-6764 and P2-0041. Special thanks goes to the Slovenia Forest Service for providing us with LiDAR data and ground truth measurements.

Appendix A

See Fig. A.11.

Appendix B

The following table summarises the notations used in this paper, ordered by their appearance in the paper.

Symbol	Meaning
G	- A graph
$E(G)$	- Edge-set of a graph G
$V(G)$	- Vertex-set of a graph G
v_i	- A vertex
$e_{i,j}$	- Edge between vertices v_i and v_j
A	- An arbitrary vertex-attribute function
$X[v_i]$	- x -coordinate of a vertex v_i
$Y[v_i]$	- y -coordinate of a vertex v_i
$Z[v_i]$	- z -coordinate of a vertex v_i
W	- An edge-weight function
\hat{P}_{i_0, i_l}	- The shortest path from v_{i_0} to v_{i_l}
$E^C(G_A)$	- A graph cut for $G_A \subseteq G$
(G, W)	- Edge-weighted graph
$M(G, W)$	- A graph union of local minima
$MSF(G, W)$	- Minimum spanning forest for (G, W) relative to $M(G, W)$, i.e. watershed
C_i	- A connected-component of a graph, containing v_i
$B(C_i)$	- A boundary-graph of a connected-component C_i
S_i^s	- A structuring element of size s , translated to the vertex v_i
δ^s	- A morphological dilation with structuring element of size s
δ_A^s	- A geodesic dilation of size s under the mask A
δ_A^R	- A reconstruction by erosion, under the mask A
ϵ^s	- A morphological erosion with structuring element of size s
ϵ_A^s	- A geodesic erosion of size s under the mask A
ϵ_A^R	- A reconstruction by dilation, under the mask A
$proj_2$	- A projection to the 2D space
CHM	- Canopy height model
DTM	- Digital terrain model
DSM	- Digital surface model
$LoFS$	- Locally fitted surfaces

(continued)

Symbol	Meaning
Π	- A set of locally fitted polynomial
T	- A thresholded graph
t	- A threshold, where \hat{t} indicates upper- and \check{t} lower-bound
K	- A criterion function
A	- Verticality estimation function
D	- Density function
N	- A function defining the number of LiDAR points within a the corresponding voxel

References

- Alonso, M., Bookhagen, B., Roberts, D.A., 2014. Urban tree species mapping using hyperspectral and LiDAR data fusion. *Remote Sens. Environ.* 148, 70–83.
- Andersen, H.E., McGaughey, R.J., Reutebuch, S.E., 2005. Forest measurement and monitoring using highresolution airborne LiDAR. Technical Report, USDA Forest Service, Service, Pacific Northwest Research Station – General Technical Report PNW 642, pp. 109–120.
- Bertrand, G., 2005. On topological watersheds. *J. Math. Imag. Vis.* 22 (2–3), 217–230.
- Bevington, P., Robinson, D.K., 2002. Data Reduction and Error Analysis for the Physical Sciences, third ed. McGraw-Hill, New York.
- Breidenbach, J., Næsset, E., Lien, V., Gobakken, T., Solberg, S., 2010. Prediction of species specific forest inventory attributes using a nonparametric semi-individual tree crown approach based on fused airborne laser scanning and multispectral data. *Remote Sens. Environ.* 114 (4), 911–924.
- Chen, Q., Baldocchi, D., Gong, P., Kelly, M., 2006. Isolating individual trees in a savanna woodland using small footprint LiDAR data. *Photogram. Eng. Remote Sens.* 72 (8), 923–932.
- Coops, N.C., Hilker, T., Wulder, M.A., St-Onge, B., Newnham, G., Siggins, A., Trofymow, J.T., 2007. Estimating canopy structure of douglas-fir forest stands from discrete-return LiDAR. *Trees* 21 (3), 295–310.
- Couprise, M., Najman, L., Bertrand, G., 2005. Quasi-linear algorithms for the topological watershed. *J. Math. Imag. Vis.* 22 (2–3), 231–249.
- Cousty, J., Bertrand, G., Najman, L., Couprise, M., 2009. Watershed cuts: minimum spanning forests and the drop of water principle. *IEEE Trans. Pattern Anal. Machine Intell.* 31 (8), 1362–1374.
- Czitrom, V., 1999. One-factor-at-a-time versus designed experiments. *Am. Stat.* 53 (2), 126–131.
- Dean, T.J., Cao, Q.V., Roberts, S.D., Evans, D.L., 2009. Measuring heights to crown base and crown median with LiDAR in a mature, even-aged loblolly pine stand. *Forest Ecol. Manage.* 257 (1), 126–133.
- Diestel, R., 2005. Graph Theory, third ed. Springer-Verlag Heidelberg, New York.
- Duncanson, L., Cook, B., Hurtt, G., Dubayah, R., 2014. An efficient, multi-layered crown delineation algorithm for mapping individual tree structure across multiple ecosystems. *Remote Sens. Environ.* 154, 378–386.
- Erdody, T.L., Moskal, L.M., 2010. Fusion of LiDAR and imagery for estimating forest canopy fuels. *Remote Sens. Environ.* 114 (4), 725–737.
- Fernandes, L.A., Oliveira, M.M., 2008. Real-time line detection through an improved Hough transform voting scheme. *Pattern Recogn.* 41 (1), 99–314.
- Ferraz, A., Bretar, F., Jacquemoud, S., Gonçalves, G., Pereira, L., Tomé, M., Soares, P., 2012. 3-D mapping of a multi-layered mediterranean forest using ALS data. *Remote Sens. Environ.* 121, 210–223.
- Gradshteyn, I.S., Ryzhik, I.M., 2000. Tables of Integrals, Series, and Products, sixth ed. Academic Press, San Diego, CA.
- Hyyppä, J., Kelle, O., Leikonen, M., Inkkinen, M., 2001. A segmentation-based method to retrieve stem volume estimates from 3-D tree height models produced by laser scanners. *IEEE Trans. Geosci. Remote Sens.* 39 (5), 969–975.
- Li, W., Guo, Q., Jakubowski, M.K., Kelly, M., 2012. A new method for segmenting individual trees from the LiDAR point cloud. *Photogram. Eng. Remote Sens.* 78 (1), 75–84.
- Lowe, D.G., 2004. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* 60 (2), 91–110.
- Lu, X., Guo, Q., Li, W., Flanagan, J., 2014. A bottom-up approach to segment individual deciduous trees using leaf-off LiDAR point cloud data. *ISPRS J. Photogram. Remote Sens.* 94, 1–12.
- Mann, P.S., 2010. Introductory Statistics, seventh ed. John Wiley & Sons Inc.
- Mongus, D., Lukač, N., Obrul, D., Žalik, B., 2013. Detection of planar points for building extraction from LiDAR data based on differential morphological and attribute profiles. *ISPRS Ann. Photogram. Remote Sens. Spatial Inform. Sci.* 2 (Part 3/W1), 21–26.
- Mongus, D., Lukač, N., Žalik, B., 2014. Ground and building extraction from lidar data based on differential morphological profiles and locally fitted surfaces. *ISPRS J. Photogram. Remote Sens.* 93, 145–156.

- Montero, R.S., Bibiesca, E., 2009. State of the art of compactness and circularity measures. *Int. Math. Forum* 4 (27), 25–28.
- Najman, L., Talbot, H., 2013. Mathematical Morphology. John Wiley & Sons.
- Persson, Å., Holmgren, J., Söderman, U., 2002. Detecting and measuring individual trees using an airborne laser scanner. *Photogramm. Eng. Remote Sens.* 68 (9), 925–932.
- Reitberger, J., Schnörr, C., Krzystek, P., Stilla, U., 2009. 3D segmentation of single trees exploiting full waveform LiDAR data. *ISPRS J. Photogramm. Remote Sens.* 64 (6), 561–574.
- Richardson, J.J., Moskal, L.M., 2011. Strengths and limitations of assessing forest density and spatial configuration with aerial LiDAR. *Remote Sens. Environ.* 115 (10), 2640–2651.
- Schneider, P., Eberly, D.H., 2002. Geometric Tools for Computer Graphics. Morgan Kaufman.
- Serra, J., Vincent, L., 1992. An overview of morphological filtering. *Circ. Syst. Signal Process* 11 (1), 48–107.
- Shi, J., Malik, J., 2000. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Machine Intell.* 22 (8), 888–905.
- Solberg, S., Brunner, A., Hanssen, K.H., Lange, H., Næsset, E., Rautiainen, M., Stenberg, P., 2009. Mapping LAI in a Norway spruce forest using airborne laser scanning. *Remote Sens. Environ.* 113 (11), 2317–2327.
- Vachier, C., Meyer, F., 2005. The viscous watershed transform. *J. Math. Imag. Vis.* 22 (2–3), 251–267.
- Vincent, L., Soille, P., 1991. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Trans. Pattern Anal. Machine Intell.* 13 (6), 583–598.
- Wang, Y., Weinacker, H., Koch, B., 2008. A LiDAR point cloud based procedure for vertical canopy structure analysis and 3D single tree modelling in forest. *Sensors* 8 (6), 3938–3951.
- Yu, X., Hyppä, J., Vastaranta, M., Holopainen, M., Viitala, R., 2011. Predicting individual tree attributes from airborne laser point clouds based on the random forests technique. *ISPRS J. Photogramm. Remote Sens.* 66 (1), 28–37.