Individual tree crown modelling and change detection from airborne lidar data

Wen Xiao, Sudan Xu, Sander Oude Elberink, and George Vosselman

***Abstract*—Lidar (light detection and ranging) provides a promising way of detecting changes of trees in 3D because laser beams can penetrate through the foliage and therefore provide full coverage of trees. The aim is to detect changes in trees in urban areas using multi-temporal airborne lidar point clouds. Three datasets covering a part of Rotterdam, the Netherlands, have been classified into several classes including trees. A connected components algorithm is applied first to cluster the tree points. However, closely located and intersected trees are clustered together as multi-tree components. A tree shaped model-based continuously adaptive mean shift (CamShift) algorithm is implemented to further segment these components into individual trees. Then, the tree parameters are derived in two independent methods: a point-based method using the convex hull; and a model-based method which fits a tree shaped model to the lidar points. At last changes are detected by comparing the parameters of corresponding tree models which are matched by a tree-to-tree matching algorithm using overlapping bounding boxes and point-to-point distances. The results are visualized and statistically analysed. The CamShift using a tree model kernel yields high segmentation accuracies. The model-based change detection is consistent with the point-based method according to the small differences between the parameters of single trees. The highlight is that it is more robust to data noise and to the segmentation of multi-tree components compared to the point-based method. The detected changes show the potential of the method to monitor the growth of urban trees.**

***Index Terms*—change detection, urban tree, 3D modelling, airborne lidar, point cloud**

I. Introduction

CHANGE detection has become a major application of remote sensing techniques which provide viable data of repetitive coverages at short intervals and of consistent quality. Especially, changes in vegetation covered areas are of great interest because they are crucial for ecosystem monitoring where digital change detection methods are widely used [1]. Vegetation in urban areas is a vital part of our living conditions. The location, density, coverage, and connectivity of trees are important factors for urban planning. Since trees are growing over time, the changes of trees should be constantly estimated and monitored.

As a relatively new remote sensing technology, airborne lidar (also referred to as airborne laser scanning) provides a promising way of change detection of trees in 3D because the laser beam can penetrate through the foliage reaching the lower crown and even the trunk. Hence it has a full coverage of trees with accurate 3D coordinates. Using high density point clouds, the changes in both coverage and height can be detected [2]. Vegetation changes in forestry at plot level, such as biomass or average height, have been studied [3]. Lidar data processing, e.g. feature extraction [4], classification [5], has been discussed extensively and vegetation have normally been taken into consideration.

In lidar data, vegetation is represented by irregularly distributed points. High vegetation has been detected in urban areas using airborne lidar data [6]. Parameters of individual trees are generated directly from the point clouds. Yu et al. [7] developed an approach for extracting individual tree attributes, i.e. height, diameter at breast height (DBH) and stem volume. They also detected harvested trees and forest growth using airborne lidar data [8]. The estimation of height growth was accomplished by individual tree delineation and a tree to tree matching algorithm. Individual tree height growth was also detected [3]. Three change detection methods, i.e. differentiation between digital surface models (DSMs) and canopy height models (CHMs), canopy profile comparison and analysis of height histograms, were presented.

In addition, 3D tree modelling in lidar data has becomes a popular topic. Models used in traditional remote sensing techniques and computer science are commonly utilized on lidar data. A fixed shape model or individual tree-wise models are applied in both mobile laser scanning (MLS) and airborne laser scanning (ALS) data. Rutzinger et al. [9] utilized four different crown shapes with different diameters at three height levels using 2D enclosing circles. Then trees were modelled by an open source framework OpenAlea. Wang et al. [10] analysed the vertical canopy structure of forest and also modelled trees in 3D. A voxel based method for individual trees delineation was implemented at different height levels. Then tree crowns were modelled and several crown parameters, e.g. tree height, crown height, crown diameter and volume are derived. Vosselman [11] detected trees in airborne lidar data by computing the local maxima with a detection rate of 97%. The tree crown was modelled using a fixed shape whose diameter was adaptive to the height of the local maximum. Instead of regression models, a "wrapped surface reconstruction" method was also proposed [12]. Tree parameters, e.g. tree height, crown diameter, crown base and volume were derived by the wrapped surfaces. And the results were validated by comparison with total station surveyed field measurements.

In this paper, we aim to develop a highly automated method for accurate individual tree geometry extraction and then change detection in urban areas using airborne lidar data. Tree changes can be affected by many factors other than real

W. Xiao is with the Newcastle University, School of Civil Engineering and Geosciences, Newcastle upon Tyne, NE1 7RU, United Kingdom. e-mail: xwshawn@qq.com.

S. Xu, S. Oude Elberink and G. Vosselman are with ITC, University of Twente, the Netherlands.

growth, e.g. seasonal effect, scanning angle/perspective, point density, etc. Some of them are very difficult to be calculated directly. However, we believe with accurate tree models, the other factors can be modelled provided that there are enough data series. Here, we focus on the accurate geometric changes. A system composed of a series of algorithms to detect the changes of trees is proposed. Based on our preliminary work [13], in which multi-tree components are treated as single objects and not modelled, a mean shift based segmentation is investigated to further segment these components so that individual tree changes, including height changes, can be detected. Another contribution is that the model-based parameter derivation is proved to be better than a point-based method for recovering the overlapped part that has been cut off during segmentation. The whole system is introduced in Section II. A flowchart is firstly depicted, then each step is described in detail. Datasets are presented in the beginning of Section III. Then experimental results are visualized and statistically analysed. Discussions based on the analysis are followed. Conclusions are drawn in the last section.

II. Method

Point clouds are processed by a sequence of algorithms. They are assumed to be classified already so that tree points are taken as the input to the system. First of all, they are clustered by the connected components algorithm, and then further segmented by the continuously adaptive mean shift (CamShift) [14]. Crowns are extracted by removing the trunk points and the points inside the crowns using the 3D alpha shapes, which will also reduce the data size. Then two independent methods, point-based (Convex hull) and model-based (Pollock model), are implemented to derive tree crown parameters which are then used for comparison. Corresponding trees are matched by the overlapping of bounding boxes and point-to-point distances. In the end, visual inspection, comparison of different methods and generic knowledge are proposed to assess the modelling and change detection results. The flowchart of the method is depicted as Figure 1.

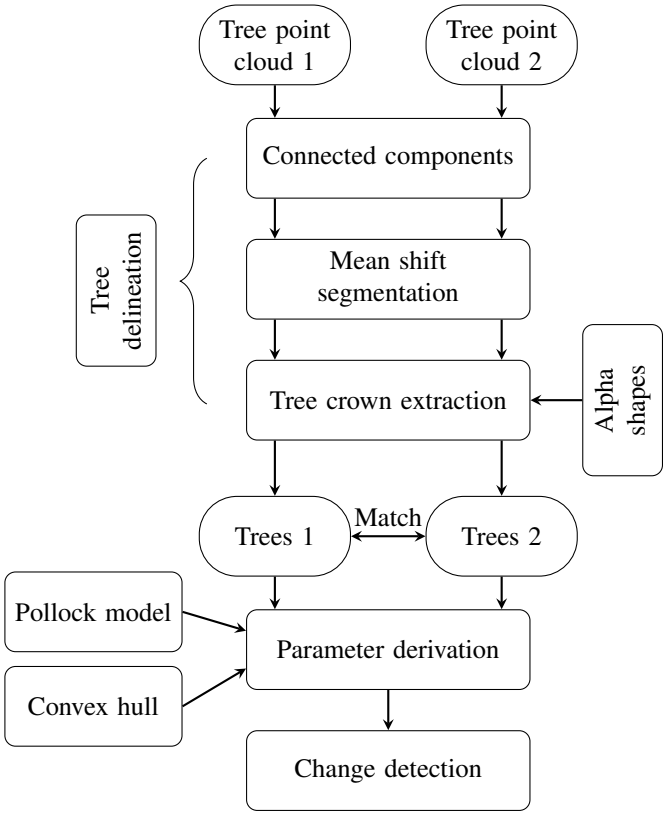


Fig. 1: Flowchart of the full system.

A. Tree Delineation

Typically, after classification points are labelled as certain classes but not yet as objects, meaning object ID has not been identified. So points belonging to the same tree need to be clustered. Moreover, commission and omission errors are inevitable during classification. One can expect many non-tree points classified as trees, especially in the general classification case, i.e. not a tree exclusive classification. Thus the detected trees need to be segmented and refined.

1) Connected Components: The *connected components* algorithm is firstly implemented to cluster the points of a tree. Points within a certain distance, e.g. 2 m, in 3D are connected as one component. The attributes of components are used to distinguish tree components from others. In general, most of the misclassified components are small fragments. Therefore the components features can be used to differentiate trees from fragments. Some geometric features are extracted, e.g. component size (number of points), height span (the distance from the lowest point to the highest point). minimum height of a component. Besides, normal distribution of the components is used to remove fragments of regular shapes. In addition to the geometric attributes, spectral information, e.g. reflectance, intensity and true color are also used since, for example, the reflectance of trees is different from other objects such as building roofs.

2) Mean-shift segmentation: The connected components algorithm is rather simple and it will cluster attached or overlapped trees as one single component, meaning a component can contain multiple trees. One change detection strategy is to treat these multi-tree components as compound objects, then changes are detected directly from these compound objects as in our preliminary work [13]. However, the disadvantage of this strategy is that the derived parameters are not accurate enough since the parameters of a compound object do not necessarily correspond to these from each individual tree. Some space between the connected trees is inevitably included into the compound tree parameter derivation using a point-based method. Thus, in this paper, the compound tree components are further segmented into individual trees.

Individual tree segmentation methods are summarized from 2001 to 2012 by [15], in which *local maxima*, *region growing*, *watershed*, and *normalized cut* are mostly used. We have tested the local maxima method in our preliminary work. It is able to differentiate single-tree components from multi-tree components, however, it is not really able to detect the exact number of trees in a multi-tree component. A further step is needed to assign each point to its corresponding local

maximum. Region growing is adopted by [15], but first it needs precise tree detection as seeds. Trees are detected by thresholding the intensity of points to extract tree trunks, which have higher intensity. Then other points are assigned to each tree by region growing. This method depends heavily on the detection of tree trunks which may fail when there are not many trunk points, especially in leaf-on seasons. Papers using watershed have converted lidar points to crown height images, e.g. CHM. Reported accuracy are not high, from 64% to 73% [15]. Normalized cut needs the number of clusters as input, so that it is limited by the number detection accuracy. Mean shift has not been included in the summary, however it has been proven to work well for multi-layered forest trees [16], [17]. Inspired by [16], mean shift for urban tree segmentation is explored. Specifically, mean shift with a tree-shaped 3D kernel and continuously adaptive size (bandwidth) is investigated.

a) Mean shift: Mean shift is a nonparametric mode-seeking algorithm proposed by [18]. It has been widely used for clustering such as image segmentation in feature space [19], [20]. In our case, the algorithm is directly applied to lidar point $\mathrm{x} \in \mathbb{R}^d$ (d means dimension). Each point of x_i (i = 1,...,n, where n is the number of points) contributes to the probability density function regarding the distance to the point x with kernel $K(\mathrm{x})$ and the radius h of the region of interest. The multivariate *kernel density estimator* is

$$f_{h,K}(\mathrm{x}) = \frac{1}{nh^d}\sum_{i=1}^{n} K(\frac{\mathrm{x}-\mathrm{x}_i}{h}) \tag{1}$$

in which h is normally called the bandwidth, K is defined as a radially symmetric kernel satisfying $K(\mathrm{x}) = c_{k,d}k(\|\mathrm{x}\|^2)$, where $c_{k,d}$ is a normalization constant assuring $K(\mathrm{x})$ add up to 1, and k is the kernel profile, determining the kernel shape. The modes of the density function are located at the zeros of the gradient function $\nabla f_{h,K}(\mathrm{x}) = 0$. Assuming the derivative of the kernel profile k exists, given $g(x) = -k^{'}(x)$ and corresponding kernel $G(\mathrm{x}) = c_{g,d}g(\|\mathrm{x}\|^2)$, the gradient of the density estimator is

$$\begin{aligned}
&\nabla f_{h,K}(\mathrm{x}) \\
&= \frac{2c_{k,d}}{nh^{d+2}}\sum_{i=1}^{n}(\mathrm{x}_i-\mathrm{x})g(\|\frac{\mathrm{x}-\mathrm{x}_i}{h}\|^2) \\
&= \frac{2c_{k,d}}{nh^{d+2}}\sum_{i=1}^{n} g(\|\frac{\mathrm{x}-\mathrm{x}_i}{h}\|^2)\left[\frac{\sum_{i=1}^{n}\mathrm{x}_i g(\|\frac{\mathrm{x}-\mathrm{x}_i}{h}\|^2)}{\sum_{i=1}^{n} g(\|\frac{\mathrm{x}-\mathrm{x}_i}{h}\|^2)} - \mathrm{x}\right]
\end{aligned} \tag{2}$$

The first term is proportional to the kernel G density estimator $f_{h,G}(\mathrm{x})$ and the second term is the *mean shift* vector

$$\mathbf{m}_{h,G}(\mathrm{x}) = \frac{\sum_{i=1}^{n}\mathrm{x}_i g(\|\frac{\mathrm{x}-\mathrm{x}_i}{h}\|^2)}{\sum_{i=1}^{n} g(\|\frac{\mathrm{x}-\mathrm{x}_i}{h}\|^2)} - \mathrm{x} \tag{3}$$

which is the difference between the kernel G weighted mean and the kernel center x. More detailed explanations can be found in [20]. In higher dimension space, the kernel can be weighted in different directions, e.g. in 3D, vertical weight and horizontal weight can be independent [16].

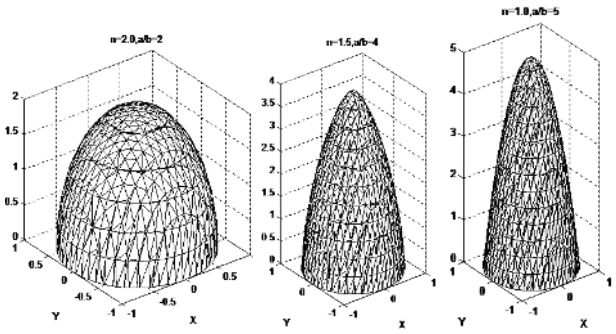


Fig. 2: Parametric crown model proposed by Pollock [21].

b) Continuously adaptive mean shift: Classical 2D mean shift searches for the mode in a circular kernel, i.e. the location with the highest point density in our case. It works well for specific cases, with carefully tuning of the bandwidth. However, this bandwidth is most probably not suitable for other multi-tree components. It means that the bandwidth should be adaptable to each component, or even to each tree.

In 3D, points are mostly evenly distributed, still higher density may be found on top of crowns. Thus points are weighted by height in favour of the mean moving upwards to the crown local maximum. As aforementioned, the bandwidth should be adaptable to each component. Assuming that higher trees have wider crowns, the bandwidth can be adaptive to the tree height. Note that there may be many trees of different heights in one component. The bandwidth should be adaptable to the height of each tree so that it can be used for not only one particular component but all the single-tree and multi-tree components in the data. Hence, the idea of continuously adaptive mean shift (CamShift) is adopted, in which case, the bandwidth h is chosen to be adaptable to the absolute height z of the kernel center x for each iteration during shifting ($h_t = z_t/c$, c is a general constant explaining the ratio between the crown size and tree height, t refers to the iteration number). For example, if a mean starts from the bottom of a component, the bandwidth will be small. Due to the higher weight in the upper part of the kernel, it will move upwards to the top, then the bandwidth will become bigger.

c) Tree crown model: Typically, a mode is searched in a single value controlled sphere in 3D. Alternatively, to better fit to the tree crown, the parametric tree crown model proposed by Pollock [21] is adopted as the mean shift kernel.

$$P(\mathrm{x}) = \frac{z^n}{a^n} + \frac{\sqrt{(x^2+y^2)^n}}{b^n} = 1 \tag{4}$$

where $\mathrm{x} = (x, y, z)$ w.r.t the crown center, a is the radius of the intersection of the model surface with the z axis, b is the radius of the crown circle in xy plane, and n is a positive real number that determines the shape of the crown surface. When $n = 2$ the model is an ellipsoid, and as n decreases to 1 the model becomes a cone (Figure 2).

Vertically, the kernel is weighted by the relative height $\frac{z-z_{min}}{z_{max}-z_{min}}$, and horizontally by a Gaussian function

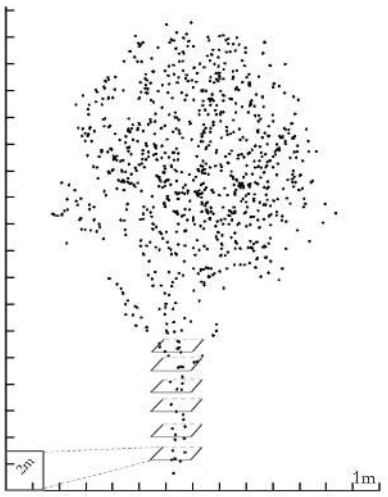


Fig. 3: Tree crown extraction.

$e^{-\lambda\|\mathrm{x}-\mathrm{x}_i\|^2}$, then the kernel profile g is defined as

$$g(\mathrm{x}) = \begin{cases} \frac{z-z_{min}}{z_{max}-z_{min}} e^{-\lambda\|\mathrm{x}-\mathrm{x}_i\|^2} & \text{if } P(\mathrm{x}) \leq 1 \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Now, the Pollock model-based CamShift is used to segment tree components into individual trees. Next tree crowns are extracted for modelling.

3) Tree crown extraction: To model the tree crown, the start point of the lower crown has to be determined. Due to the penetrability of trees, some have many points on the trunks but others do not. This inconsistency will affect the process of tree modelling, which needs a standard definition of the crown. So it is necessary to remove the points on the trunks before crown modelling. A simple crown truncating method is proposed. For each tree component, it is partitioned vertically into slices with a certain height starting from the bottom. Together with the previous slice below, a 2D bounding box of each slice is computed. If the hypotenuse of the bounding box is smaller than a predefined threshold (e.g. 2 m), the points within the slice are considered as trunk points. The bounding box will keep moving upwards until the hypotenuse is greater than the threshold. In the end, a reference height will be obtained. Points above the height are assigned to the crown (Figure 3).

B. Tree Parameter Derivation

The Pollock model is a crown surface model, whereas the lidar data have many points inside the crowns. So the interior crown points are firstly filtered out.

1) Alpha Shapes: The alpha shape algorithm is famous for shape reconstruction from a dense unorganized set of points. Indeed, an alpha shape is a linear approximation of the original shape [22]. The definition of alpha shapes is based on an underlying triangulation. As for 2D alpha shapes, circles with a certain radius (α) approach the data points until they touch points on the edges of the triangles. As shown in Figure 4, the edges touched with circles describe an approximate shape of the original points. For 3D alpha shapes [23], a triangulation is calculated first, and then spheres, instead of circles in 2D, will pass through the triangles. So the triangles that are touching spheres will represent the original shape of the data points. The points belonging to the vertices of the triangles are sufficient to describe the shape of a tree, so the points inside the alpha shape are eliminated and the data size (number of points) is reduced significantly.

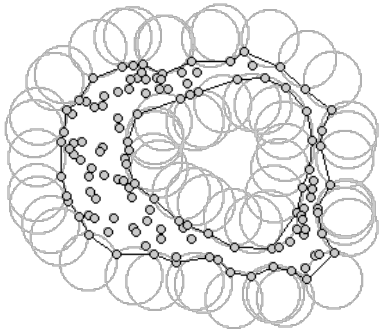


Fig. 4: 2D alpha shape algorithm.

The optimized α value is defined as the smallest value such that the complex, i.e. reconstructed shape model, has one solid component. Hence each tree has its own optimal α value. However, for certain tree component, a bigger α value is necessary because the optimized α valued sphere can go inside the crown, i.e. the alpha shape is over detailed. In this case, the volume of the alpha shapes is actually smaller than the real crown volume. Furthermore, if the α values of the same tree from two epoch datasets are quite different from each other, the change detection result will also be affected. To avoid these problems, a consistent and big enough α value (e.g. 10 m [9]) is set for all tree components. If the α value is positive infinity ($\alpha \to \infty$), an alpha shape is actually a convex hull. The areas and volumes of tree components are derived by convex hulls for comparison.

2) 3D tree modeling: The convex hull is a point-based parameter derivation method which has both advantages and disadvantages. A point-based method is straight forward, simple to process but sensitive to noise and outliers. On the contrary, model-based method is more robust. These two methods can be compared so as to assess the parameter derivation results. In this paper, tree crowns are modelled by the Pollock crown model, and the results are compared with those from point-based methods. One advantage of the Pollock model [21] is that the crown shape can be adjusted by the parameter n (Equation 4) from an ellipsoid to a cone.

a) Adjusted Pollock model: In reality, a tree crown base, i.e. 2D crown outer boundary, is hardly of any regular shape. The original model treats it as a perfect circle, whereas an ellipse approximates more to a tree crown and it also shows the orientation. Thus the model crown base is adjusted to an ellipse which becomes anisotropic. The rotation in xy plane is needed together with the center shifting as follows:

$$
\begin{aligned}
x &= (X - X_0)cos\beta - (Y - Y_0)sin\beta \\
y &= (X - X_0)sin\beta + (Y - Y_0)cos\beta \\
z &= Z - Z_0 \\
\frac{z^n}{c^n} &+ \sqrt{(\frac{x^2}{a^2} + \frac{y^2}{b^2})^n} = 1
\end{aligned} \tag{6}
$$

in which, a and b are the two semi-axes of the crown base ellipse, c is the semi-axis in z direction, n is still the real number that determines the crown shape, (X_0, Y_0, Z_0) are the coordinates of the crown center in the global coordinate system, and β is the rotation angle.

b) Crown fitting: The adjusted Pollock model-to-crown points fitting is implemented using the nonlinear least square fitting in three steps: (i) 2D crown base fitting, (ii) upper crown fitting, and (iii) lower crown fitting (Figure 5).

To precisely model a tree crown, it is divided into upper crown and lower crown because they are of different shapes. Then it is crucial to find the position of the crown center, especially the height, which will determine the crown division. We propose to first extract the crown base boundary in 2D by the convex hull. Points on the convex hull are the extreme outer points hence are assumed to be the crown base points, by which the crown center height Z_0 can be determined, and then the crown base parameters (a, b) are estimated by fitting an ellipse. Next, the upper and lower crowns are fitted separately, where crown height c and shape n are determined. Because the upper crown normally has more points and is bigger than the lower crown, the crown base is fixed using the upper crown so that the whole model has a continuous smooth surface. The whole procedure is detailed as follows:

1) the 2D centre (X_0, Y_0) and rotation angle β are firstly initialized using principal component analysis, and the points are then translated to the local coordinate system;
2) a convex hull is generated, and the crown center height Z_0 is computed as the average height of the points on the convex hull;
3) the crown base ellipse is fitted with the points on the convex hull in 2D, so the crown base shape (a, b) are initialized;
4) points above Z_0 (upper crown) are modelled by nonlinear least square fitting, where crown shape $(a, b, c_{upper}, n_{upper})$ and position (X_0, Y_0) are fitted;
5) points below Z_0 (lower crown) are modelled in the same way, but crown base (X_0, Y_0, a, b) are treated as constant, and only c_{lower} and n_{lower} are fitted.

After the 3D model fitting, the height of the tree is the crown center height plus the upper crown height $Z_0 + c_{upper}$. The crown area is the area of the fitted ellipse πab, and the volume is the sum of the upper and lower crown volumes.

C. Model-based Change Detection

After deriving the parameters of the trees in each dataset, the corresponding trees are identified based on the locations since all the data are georeferenced in the same world coordinate system.

The tree-to-tree matching is accomplished by calculating the overlaps of bounding boxes and point-to-point distances.

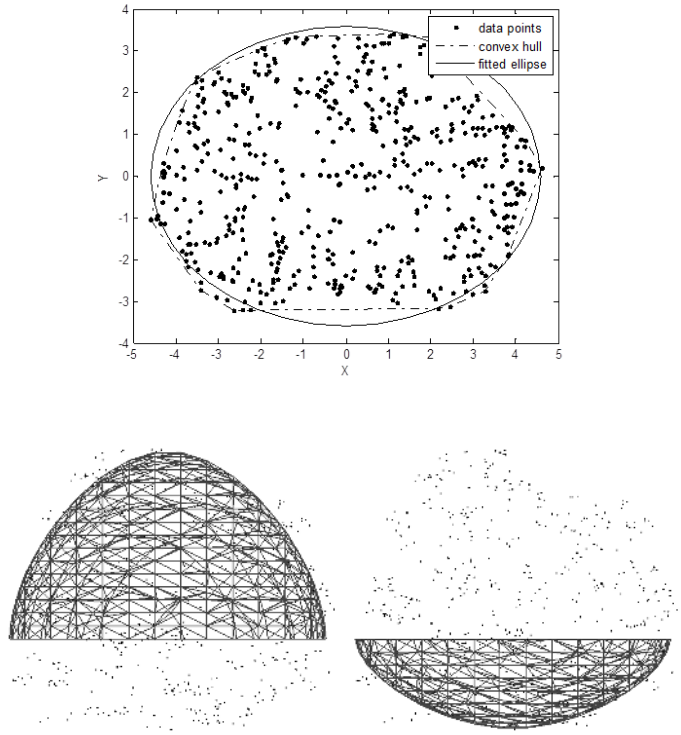


Fig. 5: Adjusted Pollock model fitting: 2D crown base fitting (top), upper crown fitting (bottom left), lower crown fitting (bottom right).

First of all, for each component in dataset 1, a bounding box (BBox) is derived. Then the overlapping BBox in dataset 2 is searched. One of the overlapped BBoxes must contain the other's center. To further check whether these two BBoxes are the corresponding components, the distances from points in dataset 1 to points in dataset 2 are calculated. If the number of distances that are smaller than 1m is greater than a certain percentage (50% based on experimental tests) of the smaller size of the two components, they are considered as corresponding trees. It happens that several components in one dataset correspond to one component in another due to under or over segmentation. So the number of nearby points is compared with the smaller one between the two components. The two corresponding components are given the same label in both dataset. Trees appearing and disappearing, i.e. trees without correspondence, are also detected.

Three parameters are extracted to analyse the changes: *height growth*, *area growth* and *volume growth*. The growth rate is the difference between the compared two data over the value of the earlier one, i.e. percentage of growth over time. As explained above, the corresponding relation between the components in two datasets may be many-to-one. In this case, the parameters of the multiple components are added up and then compared with the corresponding component.

III. Experiments and Analysis

A. Study Site and Data Preparation

Three datasets were acquired on behalf of the municipality of Rotterdam, the Netherlands. Two of them were in 2008 (March and November), named 0803 and 0811, and the third one in April 2010 (named 1004). They are all under the Dutch RD coordinate system. Point density of the data in March of 2008 is around 10 to 15 pts/m^2 (points per square meter), while the other two are about 30 to 50 pts/m^2. A part of the small island (Noordereiland) along the river in Rotterdam (Figure 6) is selected as the study area since there are plenty of trees which vary in both size and shape. The dominant tree genera on the test site are Acer, commonly known as Maple, and Castanea.

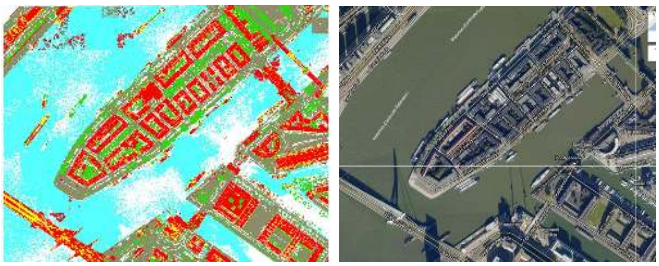


Fig. 6: Study area in Rotterdam, (left) Lidar points (vegetation: green), (right) Google Map satellite image.

The datasets have been classified into several predefined classes in which vegetation is one of them (green in Figure 6) using the method proposed by [5]. The overall accuracy of the classification is as high as 97.0%, while the accuracy of vegetation is 90.1% [5]. Both commission and omission errors are observed in the classification results. By visual inspection, several kinds of commission errors can be recognized in the dataset. Small segments such as walls, roofs of complex shape on buildings as well as cars, poles and even some ground points are classified as vegetation. On the other hand, some vegetation points are classified as other classes such as buildings and ground. Typically, the omission errors are either a few misclassified points which are a minority with regard to the amount of points of a tree hence will hardly affect the parameter derivation result, or small trees of few points which will result in the disappearance or appearance of trees. These non-geometric changes have to be verified by the end-user.

The classified vegetation, which includes both bushes and trees, and other false positives, is refined after clustering using *connected components* (Section II-A1) by filtering components features. The recall (R), precision (P) and F-score (F) are used to evaluate the results. Numbers of components before and after refinement are also presented in Table I. There are still non-tree components in the refined results. Since this is a part of the classification problem whereas we focus on tree change detection, to evaluate the feasibility of our method, trees are manually selected for experiments. In practice, the precise locations and even the trees of interest can be specified by the end users using a existing database.

TABLE I: Number of components before and after refinement, and recall (R), precision (P) and F-score (F).

Data	before	after	R%	P%	F%
0803	1451	306	93	86	89.4
0811	1169	229	97	97	97
1004	2118	275	98	93	95.4

B. Single tree delineation

The multi-tree components are further segmented by mean shift. Figure 7 shows the 2D segmentation result. The 8-tree component is correctly segmented by the 2D kernel with a proper bandwidth $h = 2.6$. However, this bandwidth value is specific to this particular component, meaning it is not suitable for other components. With more tree components (18 trees in total), the algorithm over-segments the other tree components, resulting in 24 trees. This is the reason that we propose to segment trees in 3D and adapt the bandwidth to each tree.

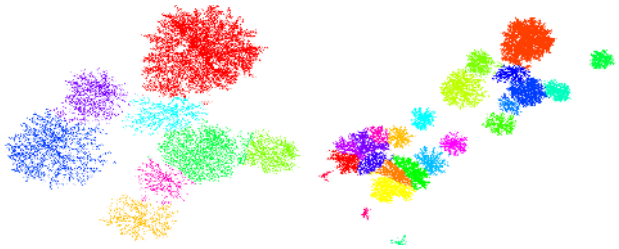
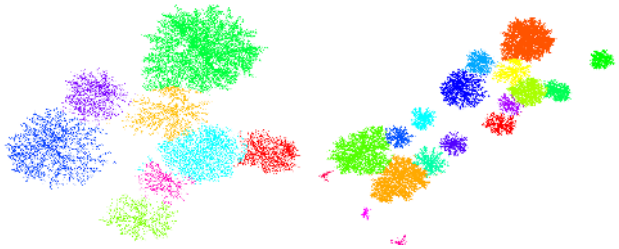


Fig. 7: 2D mean shift segmentation. Left: a 8-tree component correctly segmented with bandwidth h = 2.6; right: 18 trees including the 8-tree component using the same bandwidth, over segmented to 24 trees.

The bandwidth is continuously adaptive to the kernel height ($h_t = z_t/c$), and is only determined by the constant c. The Pollock model kernel shape is governed by the horizontal circle radius a, vertical z axis parameter b and shape parameter n. To have a kernel shape that fits a general shape of a tree, n is set as the middle value 1.5, b is set as two times as a. In this case, there is only one tunable parameter a, which is the bandwidth. As illustrated in Figure 8, both data are correctly segmented using the same bandwidth.



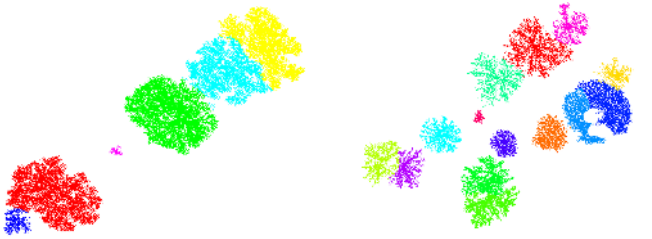
(a) (b)

Fig. 8: 3D mean shift segmentation. Left: a 8-tree component correctly segmented with bandwidth $h_t = z_t/3.2$; right: 18 trees including the 8-tree component correctly segmented with the same bandwidth.

Now, trees of different heights have different kernel bandwidth, still, the exact bandwidth size is determined by c. Ideally, the same c value is expected to be suitable for all trees. To figure out its sensitivity, more trees are experimented with different constant values. As shown in Table II, datasets of different numbers of trees are correctly segmented by various c values. In some cases, c can vary from 3.0 to 3.4, but in others, the value range is more limited. A mutual value (3.2) suitable to all datasets is found.

TABLE II: Continuously adaptive mean shift tree segmentation.

Constant c	6 trees	8 trees	12 trees	18 trees	21 trees	24 trees
3.0	6	8	14	17	21	24
3.1	6	8	14	17	21	24
3.2	6	8	14	18	21	24
3.3	6	8	14	18	21	25
3.4	6	8	14	18	21	25



(a) 6 trees (b) 12 trees

(c) 21 trees (d) 24 trees

Fig. 9: More 3D mean shift segmentation examples with different number of trees using the same bandwidth $h_t = z_t/3.2$.

More segmentation results are depicted in Figure 9. Note that trees are correctly segmented regardless of their diverse sizes and shapes. Two missed segmentations are observed in Figure 9b, in which two trees are over segmented (also shown in Table II). One tree is incomplete due to misclassification. The other one has an extremely flat crown as shown in Figure 10. The segmentation results of all the three datasets using both connected components (CC) and CamShift are illustrated in Table III, which suggests that the CamShift significantly improves the segmentation with very high overall accuracies.

After segmentation, each tree crown is extracted by removing the trunk points, and the 3D alpha shape algorithm is implemented to filter out the points inside the crown. Figure 11 shows the extracted crowns.

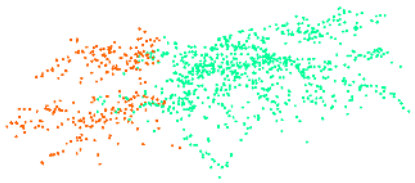


Fig. 10: Incorrect tree segmentation example: over segmented.

TABLE III: Segmentation results using connected components (CC)(number of components) and CamShift, including number of components, under-segmentation, over-segmentation, percentage of multi-tree components and overall accuracy.

Data	CC	CamShift	Under	Over	Multi-tree%	Accuracy%
0803	141	194	0	2	38.1	99.0
0811	153	230	6	2	46.5	96.5
1004	168	234	4	2	40.6	97.4

Fig. 11: Tree crown extraction and interior points filtering.

C. Tree Models and Derived Parameters

After tree modelling, height, area, and volume parameters are extracted from models for later change detection. As described in Section II-B2, the height is the crown base height plus the upper crown height, the area is the ellipse area of the crown base, and the volume is the sum of the lower and upper crown volumes. Figure 12 shows the modelling results of different trees. To evaluate the parameters, they are compared with those derived from a point-based method. Area and volume parameters are extracted by the convex hull.

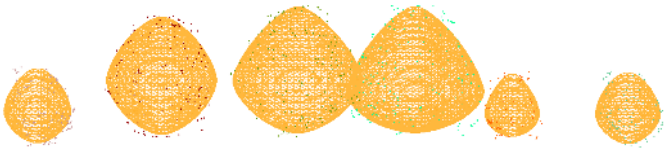
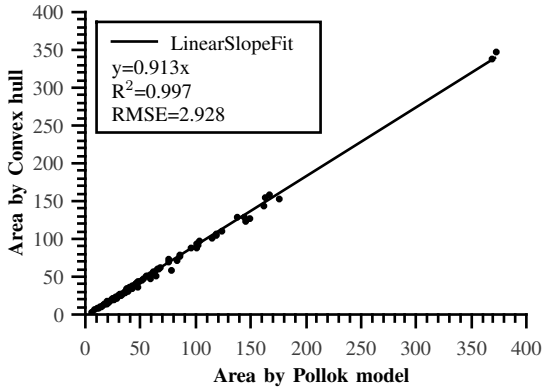
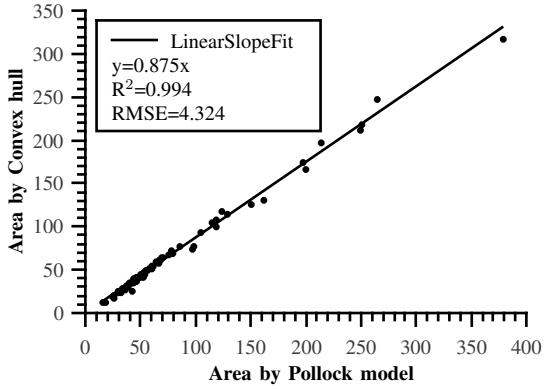


Fig. 12: 3D tree models.

Multi-tree components are further segmented into individual trees, i.e. they are partitioned through the middle of their intersections. So these segmented trees are incomplete (still referred to as multi-tree components to distinguish from the single-tree components). In this case, point-based methods, e.g. convex hull, will not be able to recover the part that has been cut off. However, model-based methods are assumed to be, to some extend, robust to this situation. For heavily intersected trees, point-based parameters are supposed to be smaller than the model-based ones.



(a) Single-tree components



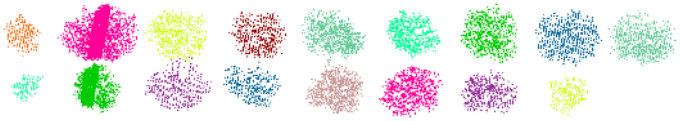
(b) Multi-tree components

Fig. 13: Area comparison between Pollock model and convex hull for single-tree (a) and multi-tree components (b).

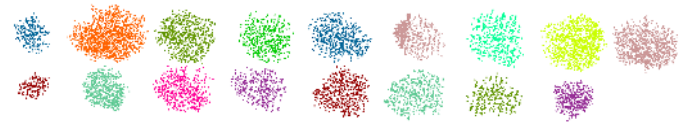
To verify this assumption, the area and volume parameters derived from the two methods are compared. Parameters of single-tree and multi-tree components are compared separately. Figure 13 illustrates the linear correlation $y = Ax$ between the area parameters derived from the two methods. In both single-tree and multi-tree cases, the parameters are well correlated, whilst Pollock model values are slightly greater ($A < 1$). It is clear that in the multi-tree case, $A^{multi}_{area} = 0.875$ is notably smaller than $A^{single}_{area} = 0.913$, meaning in general the point-based areas are smaller than model-based ones to a greater degree compared to the single-tree case. The same comparison is conducted for volume parameters. This time the convex hull parameters are greater than Pollock models ($A > 1$). But still, the slope of the single-tree case $A^{single}_{volume} = 1.133$ is greater than $A^{multi}_{volume} = 1.124$ of the multi-tree case.

Besides, the difference between the two results w.r.t the Pollock model method, i.e. (Pollock model - Convex hull)/ Pollock model, is computed. The standard deviation of the difference of area for single-tree case $\sigma^{single}_{area} = 4.74\%$, whereas for multi-tree case $\sigma^{multi}_{area} = 5.68\%$. And the standard deviation of the difference of volume for single-tree case $\sigma^{single}_{volume} = 7.82\%$, whereas $\sigma^{multi}_{volume} = 8.79\%$. This means that the parameters from the two methods are less consistent in the multi-tree case, which can be caused by the fact that the degree of intersection varies, i.e. trees can be slightly or heavily intersected in multi-tree components.

To further verify the modelling results, different strips of the 0803 data are used. The time difference is small enough to be ignored, meaning trees are not supposed to be changed. However, the scanning perspectives are different and the point density may vary due to different strip flying speeds. Figure 14 depicts the example of trees from different strips. It is observable that same trees have different point distributions. Table IV shows the differences of height, area and volume in percentage w.r.t one of the compared strips. The differences are small. However if the detected tree changes are also small, these differences have to be taken into consideration, meaning the changes may just be the inconsistency of tree models affected by scanning perspective and point density.



(a) Strip 1



(b) Strip 2

Fig. 14: Example of trees from different strips.

TABLE IV: Model parameter comparison of the same trees from different strips. Differences in % w.r.t strip1.

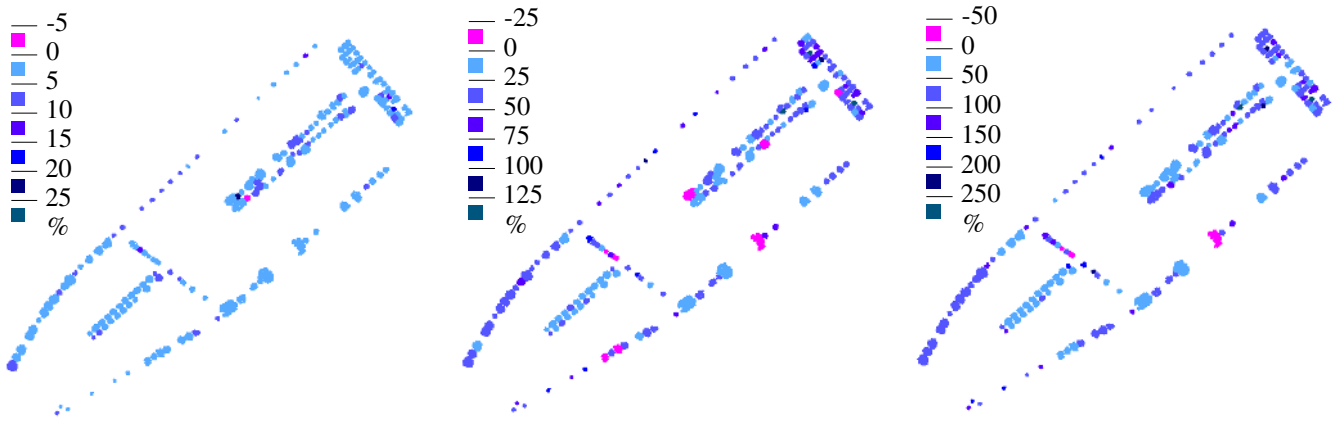
Compared Data	Height %		Area %		Volume %	
	mean	σ	mean	σ	mean	σ
strip1 vs. strip2	0.06	2.19	-1.22	7.75	-2.55	8.53

D. Detected Changes and Analysis

The extracted parameters are then assigned back to each component, and corresponding components are matched. In the case of many-to-one matching, area and volume parameters are added up among the merged components, and the maximum height is taken as the overall height.

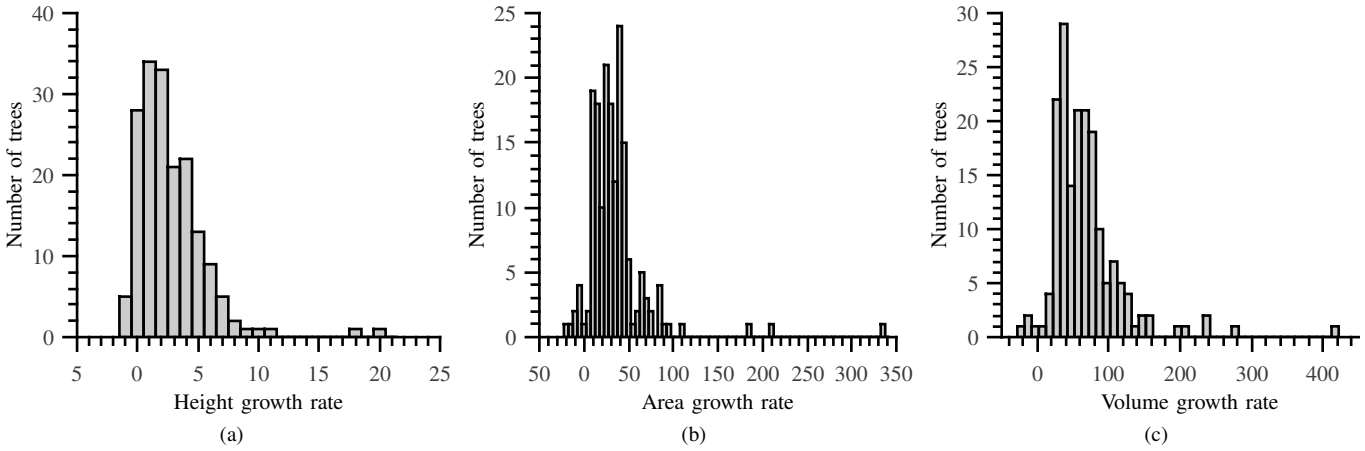
Figure 15 depicts the change detection results of data 0803 compared with data 0811. Height, area, volume growth rates w.r.t to the former data are presented. In general, most of the trees increase in height, area and volume. Similar sized trees have similar growth rates and smaller trees have generally greater growth rates. The histogram of each growth rate is illustrated in Figure 16. With few exceptions, most trees have positive growth rates. The majority of height growth rate lies between 0 to 5%, and the area growth rate ranges from 0 to 100%, while the volume growth rate varies from 0 to almost 200%, excluding some outliers that have much higher rates.

Earlier epoch datasets are compared with later epochs, i.e. data 0803 vs. 0811, 0803 vs. 1004, 0811 vs. 1004. Table V



(a) Height growth rate (b) Area growth rate (c) Volume growth rate

Fig. 15: Change detection results of data 0803 compared to data 0811.



(a) (b) (c)

Fig. 16: Growth rate histogram of data 0803 compared to data 0811.

TABLE V: Change detection results, including the mean and standard deviation of height, area, and volume growth rate, among the three datasets.

Compared Data	Height %		Area %		Volume %	
	mean	σ	mean	σ	mean	σ
0803 vs. 0811	3.23	2.83	37.84	35.92	72.25	50.82
0803 vs. 1004	2.60	3.20	24.11	24.21	43.34	42.98
0811 vs. 1004	-0.50	3.68	-6.85	26.61	-12.97	33.11

shows the mean and standard deviation of the height, area, and volume growth rates among the three datasets.

According to the change detection results, height growth rate is much smaller than area and volume growth rates. And volume growth rate is the highest among the three, which is reasonable because volume growth accounts for changes in three dimensions. Note that the growth rates of data 0803 compared to data 1004 are even smaller than those compared with data 0811, which is five months earlier. This is also reflected by the results of the third comparison, 0811 vs. 1004, in which the growth rates are negative, meaning the trees are getting smaller. One plausible explanation is that those trees are pruned during these five months since it is normal that urban trees are pruned by the local municipality on a regular basis. However a more likely reason is that the point densities on trees of those two data differ significantly, since the former is still in leaf-on season whereas the later is in leaf-off season. Different seasons can have a big impact on the point density therefore the derived parameters [24], [25]. So no concrete conclusion can be drawn from this part. Still data 0803 and 1004 are both in leaf-off season, so even without ground truth comparison, the change results reveal the growing trend. More data are needed for further investigation. Note that the detected changes are significantly higher than the differences of models of the same trees from different strips.

IV. Discussions

The appearance and disappearance of trees can also be detected by our method. However, it does not necessarily mean that trees are planted or cut off. Empirical evidence

suggest that they are mostly caused by omission errors during the classification step. Thus these two categories of changes are not studied in this paper. In the classification refinement step, by observing the number of components before and after refinement, it is clear that there is a remarkable improvement. The only factor is the geometric and spectral features at object/component level. These features can be integrated into a tree-oriented classification, in which case this refinement step can be trivial.

In our preliminary work [13], multi-tree components were treated as compound trees and only point-based method was used for parameter derivation. The Pollock crown model was not used since a compound tree does not hold an original crown shape. Tree height changes were not detected. Since point-based methods will inevitably include a part of the empty space between connected trees. The detected changes are inaccurate. So in this paper, the multi-tree components are further segmented by the CamShift using Pollock model as the kernel. The advantage is that there is only one tunable parameter, i.e. the kernel bandwidth, which is rather independent from different tree shapes and sizes. The same bandwidth has been used for all the three datasets, resulting in really high accuracies. Compared with other segmentation methods mentioned in [15], this crown shaped kernel will guarantee that the segmented components have at least the same, if not greater, size as the kernel, which inherently avoids trees being segmented into over small components. Still, the kernel is only adaptive to the height not to the shape. Thus, if the tree crown shapes vary significantly in the dataset, a single bandwidth may not be suitable. The CamShift using tree shaped kernel works well for segmentation at tree level. But it can be observed that some points on the edges between two trees are sometimes miss-segmented. This is because the kernel mean is randomly initialized, and all the points that been visited by the kernel will be clustered into the same segment. So the points between trees maybe visited by the neighbour tree. This can be tackled by running the kernel on each point instead of random initialization. However, it will take much longer time as tested, especially for big components with many trees.

The adjusted Pollock model enhances the 3D tree modelling by adding one more degree of freedom to the crown base, which is more realistic. Apart from the compared parameters, other information, e.g. accurate crown 3D position, lower crown start point, can also be extracted from the model. In general, the model-based method delivers more robust information than point-based methods since it is barely affected by outliers. More importantly, the latter can not recover the overlapped part that has been cut off from trees in multi-tree components since it is strictly restricted by the points, whereas the proposed model is expected to have a better recovery. This is proved by treating the correlation of single-tree component parameters between these two methods as standard, and showing that the point-based parameters are getting smaller w.r.t model-based ones from multi-tree components. However, the absolute accuracy of the model reconstruction of the missing part is not evaluated.

Considering the errors of pre-processing and modelling, every tree correspondence will be slightly different even if no change has happened. The inconsistency of tree parameters from different strips is significantly lower than the detected average changes in terms of both mean and standard deviation. Still very small changes should be taken care of since they might not be real changes. Based on visual inspection, trees of similar sizes have similar behaviours of changes, especially for those located at the same areas. Moreover, the change rates of small trees are generally greater than bigger trees. In reality, smaller trees grow faster. Nevertheless, ground truth is needed to verify the absolute values of the changes.

V. Conclusion

A model-based change detection system is presented to potentially monitor the growth of urban trees. The proposed Pollock model-based CamShift successfully segmented connected components into individual trees with high overall accuracy. The 3D alpha shapes algorithm significantly reduces the data sizes and more importantly extracts the points on the outer boundary of the crowns, so that the remaining points can be directly used for 3D tree modelling and other point-based parameter derivation methods. The Pollock model introduces the crown shape parameter n so that every single tree has its own crown shape. The separation of upper crown fitting and lower crown fitting is innovative. The derived parameters have high linear correlation with convex hull which is based directly on the points. The standard deviations of the parameter differences for single-tree components between the Pollock model and convex hull are very small, suggesting the consistency between the Pollock model-based and point-based methods. Also the differences of models of the same trees from different strips are small. The advantage of model-based parameter derivation is its robustness against noise and tree incompleteness due to segmentation. Moreover the tree models can be used for visualization purposes. The proposed system provides a guideline for change detection of trees in multi-temporal airborne lidar point clouds. The growths of trees are successfully detected. Furthermore the system is highly automatic.

Future work will focus on the influence of other factors of tree changes, e.g. seasonal effect, wind effect, and multiple sources. The fusion with other types of data can be helpful to improve the performance of the system. Imageries obtained simultaneously with the point clouds are useful to identify the commission and omission errors at tree level. Mobile mapping system (MMS) is capable of capturing the details of both trunk and lower crown of trees. So the fusion of mobile and airborne lidar data will improve the modelling of trees hence facilitate the change detection.

Acknowledgment

The authors would like to thank Fugro Aerial Mapping for acquiring and the Municipality of Rotterdam, the Netherlands for providing the lidar data.

REFERENCES

[1] P. Coppin, I. Jonckheere, K. Nackaerts, B. Muys, and E. Lambin, "Digital change detection methods in ecosystem monitoring: a review," *International Journal of Remote Sensing*, vol. 25, no. 9, pp. 1565–1596, 2004.

[2] C. J. Houldcroft, C. L. Campbell, I. J. Davenport, R. J. Gurney, and N. Holden, "Measurement of canopy geometry characteristics using lidar laser altimetry: a feasibility study," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 43, no. 10, pp. 2270–2282, 2005.

[3] X. Yu, J. Hyyppä, A. Kukko, M. Maltamo, and H. Kaartinen, "Change detection techniques for canopy height growth measurements using airborne laser scanner data," *Photogrammetric engineering and remote sensing*, vol. 72, no. 12, p. 1339, 2006.

[4] M. Weinmann, B. Jutzi, S. Hinz, and C. Mallet, "Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 105, no. 0, pp. 286 – 304, 2015.

[5] S. Xu, G. Vosselman, and S. Oude Elberink, "Multiple-entity based classification of airborne laser scanning data in urban areas," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 88, pp. 1–15, 2014.

[6] M. Rutzinger, B. Höfle, and N. Pfeifer, "Detection of high urban vegetation with airborne laser scanning data," in *ForestSAT07*, 2007.

[7] X. Yu, J. Hyyppä, M. Vastaranta, M. Holopainen, and R. Viitala, "Predicting individual tree attributes from airborne laser point clouds based on the random forests technique," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 66, no. 1, pp. 28–37, 2011.

[8] X. Yu, J. Hyyppä, H. Kaartinen, and M. Maltamo, "Automatic detection of harvested trees and determination of forest growth using airborne laser scanning," *Remote Sensing of Environment*, vol. 90, no. 4, pp. 451–462, 2004.

[9] M. Rutzinger, A. K. Pratihast, S. J. Oude Elberink, and G. Vosselman, "Tree modelling from mobile laser scanning data-sets," *The Photogrammetric Record*, vol. 26, no. 135, pp. 361–372, 2011.

[10] Y. Wang, H. Weinacker, and B. Koch, "A lidar point cloud based procedure for vertical canopy structure analysis and 3d single tree modelling in forest," *Sensors*, vol. 8, no. 6, pp. 3938–3951, 2008.

[11] G. Vosselman, "3d reconstruction of roads and trees for city modelling," in *International Archives of Photogrammetry and Remote Sensing*, vol. 34, 2003, pp. 231–236.

[12] A. Kato, L. Moskal, P. Schiess, M. Swanson, D. Calhoun, and W. Stuetzle, "Capturing tree crown formation through implicit surface reconstruction using airborne lidar data," *Remote Sensing of Environment*, vol. 113, no. 6, pp. 1148–1162, 2009.

[13] W. Xiao, S. Xu, S. Oude Elberink, and G. Vosselman, "Change detection of trees in urban areas using multi-temporal airborne lidar point clouds," in *SPIE Remote Sensing*, vol. 853207, 2012, pp. 1–10.

[14] G. R. Bradski, "Computer vision face tracking for use in a perceptual user interface," in *IEEE Workshop on Applications of Computer Vision*, 1998, pp. 214–219.

[15] X. Lu, Q. Guo, W. Li, and J. Flanagan, "A bottom-up approach to segment individual deciduous trees using leaf-off lidar point cloud data," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 94, pp. 1–12, 2014.

[16] A. Ferraz, F. Bretar, S. Jacquemoud, G. Gonçalves, L. Pereira, M. Tomé, and P. Soares, "3-d mapping of a multi-layered mediterranean forest using als data," *Remote Sensing of Environment*, vol. 121, pp. 210–223, 2012.

[17] Y. Qin, A. Ferraz, C. Mallet, and C. Iovan, "Individual tree segmentation over large areas using airborne lidar point cloud and very high resolution optical imagery," in *IEEE International Geoscience and Remote Sensing Symposium*, 2014, pp. 800–803.

[18] K. Fukunaga and L. D. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *Information Theory, IEEE Transactions on*, vol. 21, no. 1, pp. 32–40, 1975.

[19] Y. Cheng, "Mean shift, mode seeking, and clustering," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 17, no. 8, pp. 790–799, 1995.

[20] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 5, pp. 603–619, 2002.

[21] R. J. Pollock, "Model-based approach to automatically locating tree crowns in high spatial resolution images," in *Image and Signal Processing for Remote Sensing*, vol. 2315, 1994, pp. 526–537.

[22] H. Edelsbrunner, D. G. Kirkpatrick, and R. Seidel, "On the shape of a set of points in the plane," *Information Theory, IEEE Transactions on*, vol. 29, no. 4, pp. 551–559, 1983.

[23] H. Edelsbrunner and E. P. Mücke, "Three-dimensional alpha shapes," *ACM Transactions on Graphics (TOG)*, vol. 13, no. 1, pp. 43–72, 1994.

[24] E. Næsset, "Assessing sensor effects and effects of leaf-off and leaf-on canopy conditions on biophysical stand properties derived from small-footprint airborne laser data," *Remote Sensing of Environment*, vol. 98, no. 2, pp. 356–370, 2005.

[25] T. Brandtberg, "Classifying individual tree species under leaf-off and leaf-on conditions using airborne lidar," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 61, no. 5, pp. 325–340, 2007.

Wen Xiao received the B.S. in Geodesy and Geomatics from Wuhan University, China, in 2010, the M.S. in Geoinformatics with Cum Laude from ITC, University of Twente, the Netherlands, in 2012, and the Ph.D in Geoinformation Science and Technology from IGN, Université Paris-EST, France, in 2015. He has worked at IGN as a research engineer and visited ETH Zurich as an academic guest. Since 2016, he has been a Lecturer in photogrammetric computer vision and laser scanning in the Newcastle University, United Kingdom. His research interests include mobile mapping, laser scanning and photogrammetry, with a focus on change detection, 3D modeling and classification.

Sudan Xu received the B.S. and M.S. degrees in Geodesy and Geomatics Engineering from the Wuhan University, China, in 2004 and 2006 respectively, and the Ph.D in Geoinformatics from ITC, University of Twente in 2015. She worked at the Institute of Measurement Bureau, Fujian, China in 2006, and in the Department of Computer Science in Kunsan University, South Korea in image processing in 2007. In 2008, she worked at the Zhongke Remote Sensing Inc., Tianjin, and then at the Guangzhou Urban Planning and Design Survey Research Institute in 2009. Her research interests include image and point cloud processing, especially the classification and change detection of laser scanning data.

Sander Oude Elberink graduated as Geodetic Engineer from Delft University of Technology in 2000, and received the PhD in 2010. After his master research on the classification of airborne laser scanner data, he joined the Section of Photogrammetry and Remote Sensing as researcher. From 2001 till 2005 he worked as research consultant and project manager at the Survey department of Rijkswaterstaat in Delft. He received a young author's award for best papers at the ISPRS congress in Beijing, China in 2008. From September 2009 he holds a position of assistant professor at the department of Earth Observation Science at ITC, University of Twente. Since 2012, he is the Chair of WG III/2 of the ISPRS. His main research interests are (semi-)automated acquisition of geo-information, 3D modelling/reconstruction and information extraction.

George Vosselman received the M.S. in Geodetic Engineering in 1986 from the Delft University of Technology, the Netherlands, and the Ph.D from the Rheinische Friedrich Wilhelms University of Bonn, Germany, in 1991. He worked as a researcher at the Institute of Photogrammetry of the Stuttgart University, Germany, until 1992. After a year as visiting scientist at the University of Washington, U.S.A., he was appointed professor of Photogrammetry and Remote Sensing at the Delft University of Technology in 1993. In 2004 he joined ITC, University of Twente as professor of Geo-Information Extraction and Remote Sensing. From 2005 until 2012 he was Editor-in-Chief of the ISPRS Journal of Photogrammetry and Remote Sensing. As of 2012 he is head of the Department of Earth Observation Science. The research of his chair group at ITC focuses on the utilisation of advancements in sensor technology for the efficient production of large scale geo-information. He has published over 150 journal and conference papers on photogrammetry and laser scanning.