# Graduation Plan

| **Personal information** | |
|---|---|
| Name | **Shenglan Du** |
| Student number | **4718291** |
| Address | |
| Postal code | |
| Place of residence | |
| Telephone number | |
| E-mail address | |
| **Graduation committee** | |
| Main mentor | **Dr. Liangliang Nan**<br>Assistant professor, Faculty of Architecture and the Built Environment, Delft University of Technology |
| Second mentor | **Dr. Roderik Lindenbergh**<br>Assistant professor, Faculty of Civil Engineering and Geosciences, Delft University of Technology |
| **Title** | |
| Title of the graduation research | Accurate, Detailed, and Automatic Tree Modeling from Point Clouds |
| **Research** | |
| See full proposal attached | |

# Accurate, Detailed, and Automatic Tree Modeling from Point Clouds

Shenglan Du

January 7, 2019

## Contents

# 1 Introduction

Trees are an important component throughout the world. They form and function in the natural ecosystem, the human-made environments, for instance parks and gardens, and intermediate environments such as lands recolonized by vegetation (Deussen et al. 1998). Models of trees, therefore, have a wide range of applications, including urban landscape design, ecological simulation, forestry management, and entertainment visualization.

While applications such as landscape design and visualization only require modelling virtual trees, lots of other applications relevant with ecological modelling and forestry management require accurate measuring of tree parameters (tree height, tree stem diameter, etc). The traditional way of measuring trees is to manually conduct field work, which is usually tedious and time consuming. With the development of digital image processing technologies, researchers had tried to reconstruct digital tree models from photographs (Reche-Martinez et al. 2004). Due to structural and geometrical complexity of the trees, the modelling still remains a challenging problem.

In recent years, the laser scanning technology has been widely used in urban and environment-related studies. Measurements from laser scanning enable people to acquire millimeter-level of detail from the surrounding area, and that allows rapid, automatic and periodical estimates of many important forest inventory attributes (Liang et al. 2016). With laser scanning it is easy to obtain 3D point cloud data of trees with high quality and sufficient density, thus making it possible to obtain accurate tree models.

The aim of this thesis project is to develop a robust approach to *automatically reconstruct accurate and detailed 3D tree models from point clouds.*

The inputs to the proposed approach are the point clouds of real-world trees collected from various scanning devices. The outputs are 3D tree models with accurate and detailed branch-structure. In order to evaluate the proposed method, the accuracy of the reconstructed models (defined as the distance between the input points and output model) will be computed and assessed.

This proposal is organized as follows. In the next chpater, the related work is introduced and elaborated on. Chapter 3 gives a detailed statement of the research problem and chapter 4 presents the methodology and the working pipeline. This proposal will end with a schedule plan together with a brief description about the tools and data to be used.

# 2 Related Work

## 2.1 Modelling "icon" trees

Currently the most common approach is to model the trees as icons, which is easy to generate and maintain. Gobeawan et al. (2018) constructed 3D tree models in the format of CityGML for the virtual Singapore city (Figure 1). The trees were modeled in various level of details. Among them, the LOD1 is a simplified 3D proxy with a height representation, the LOD2 is a crown and trunk representation, and the LOD3 is a more realistic representation at the species level of details including leaves and branches. This approach enables modelling 3D trees dynamically on a large scale. However, the model reconstructed is not accurate or detailed enough.

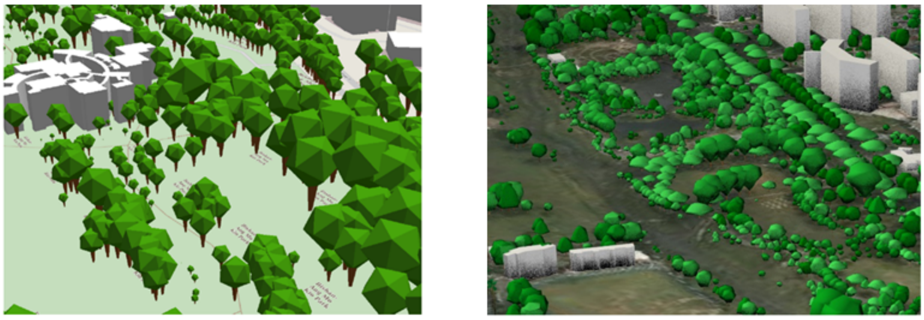

Figure 1: Tree models of the Virtual Singapore (Gobeawan et al. 2018)

### 2.2 Modelling trees based on cylinder-fitting

As the laser scanning technology has been widely applied in forestry and urban studies, it is feasible to acquire highly dense point clouds, which lays the foundation for accurate tree modelling. Tree attributes, such as tree height and tree stem, can be automatically measured from the laser scanning data (Olofsson et al. 2014). Furthermore, to accurately model the geometry of the tree branches, cylinder fitting is a primary strategy in literature (Wang et al. 2016).

Hackenberg et al. (2014) described a method for fitting cylinders into a point cloud. In this work, cylinders are stored as a hierarchical tree-like data structure, which encapsulates parent-child neighbour relations and also takes into consideration the tree's direction of growth. The cylinder structure enables the efficient extraction of tree components, such as the stem or a single branch. Nevertheless, this approach is not fully automated, as tree extraction and pre-processing is performed manually.

Raumonen et al. (2013) proposed another method for constructing precise tree models efficiently from point clouds. This approach is based on a step-by-step collection of small connected surface patches, which are small local subsets of the point cloud. The neighbor-relations and geometrical properties of these patches are extracted and used to segment the point cloud into branches, after which the branches are modeled as collections of cylinders. One drawback of this method is that it requires high quality input point clouds. An individual tree often needs to be scanned multi-times from various view points to ensure a good output model.

By applying a cylinder-based approach, Calders et al. (2018) reconstructed forest stands from scanning points for radiative transfer modelling and simulation. They developed a semi-automatic workflow which enables the segmentation of tree point clouds and the cylinder fitting along the tree skeleton. Furthermore, the TLS data is matched with traditional census data to determine the species of each individual tree and allocate species-specific radiometric properties. This method is generic, highly transferable and adjustable to data, and also capable of producing realistic virtual forests.

While most of the methods focus on the tree modelling in a flat area, the work conducted by Wang et al. (2016) paid attention to the tree stem reconstruction in landslide-affected areas, where the terrain is usually steep and the tree stems are often growing in an irregular direction. In their work, the tree stems are modelled by fitting a series of cylinders using a 2D-3D random sample consensus-based approach. The resulted stem parameters can be employed in biomass estimation, tree growth quantification, and other forest-related studies.

### 2.3 Skeleton-based approaches

Compared with methods that fit tree branches based on cylinder fitting, a more natural approach is to construct the tree skeleton first and apply the geometry inflation. According to Cornea et al. (2007), curve-skeletons are thinned 1D representations of 3D objects, which is useful for many visualization tasks including virtual navigation, reduced-model formulation, visualization improvement, animation, etc.

Existing literature on tree skeleton extraction is vast. Verroust and Lazarus (1999) proposed an algorithm which constructs a neighbourhood graph among input 3D points and then extracts consecutive skeletal curves from computing the length of edges. Instead of extracting skeleton curves directly from point clouds, Bucksch et al. (2009) applied another method, organizing points into the octree structure and generating the skeletal curves from the octree cells. This approach has good performance in terms of computation efficiency. However, the results might turn out to be unreliable when the branches don't have the uniform spatial distributions.

Livny et al. (2010) introduced a new approach which computes the minimum spanning graph from point clouds to obtain an initial skeletal structure of the tree. Furthermore, several global optimizations are applied to remove noises and better model the tree structure. This method gives us a lot of inspiration, from which our research work will be mainly based on. One significant strength of this method is its capability of generating the tree geometry from the pure point cloud data, instead of requiring additional information (e.g., photographs) and user interaction. Nevertheless, this approach has multiple drawbacks. One limitation is that it is predominantly data-driven. It is not designed to reconstruct skeletal structures over large regions of missing data. Moreover, it doesn't guarantee a geometrically-correct tree model, i.e., the output model doesn't fit well to the input point cloud data.

Having attempted to reconstruct the skeleton structure of tree branches, some follow up works focus on the modelling of tree foliage and tree leaves. Livny et al. (2011) encoded a lobe-based tree representation from sample trees and later synthesized it into a fully detailed tree model that visually resembles the sample tree data. This approach requires the prior knowledge of trees classification information. Xie et al. (2016) utilized examples of real trees to enhance tree creation with realistic structures and fine-level details. 3D real-world trees are captured in great detail and processed as exemplar parts, and these finer-level geometric details are transferred to the reconstructed tree models, which enables that the generated tree models inherit geometric details from the example models.

## 3 Research Objectives

### 3.1 Objectives

The goal of this research is to develop a method that can automatically reconstruct 3D tree models from the point cloud data. The output tree models should have accurate and detailed branch structure. From the reconstructed 3D models the tree parameters such as tree height and tree-stem diameter can be measured. To achieve this goal, several relevant sub-questions are listed as follows:

- How to extract the tree structure from the point cloud? Which data structure is most suitable for storing and representing the tree skeleton structure?
- Which primitive should be used to model the geometry of tree branches? Which algorithm is most suitable for tree branch reconstruction?
- How to realistically render the tree models?
- How to assess the quality of the output tree models? What is the accuracy of the reconstructed models? Where are the errors from and how can we improve the accuracy?

### 3.2 Scope of research

This thesis project will focus on 3D tree modelling from point clouds, aiming to achieve accurate and detailed branch reconstruction of the trees. The reconstruction of tree leaves is therefore out of the scope of this research. Furthermore, this research focuses on modelling and reconstruction of the individual trees and we assume that each tree is segmented out from the point cloud (using existing tools such as Mapple).

### 3.3 Research challenges

3D tree modelling from pure point clouds is a challenging topic. Compared to other man-made objects such as buildings, furniture, and mechanical parts, trees typically have more complicated appearances and structures. Due to the variety of tree species, branches as well as leaves, it is difficult to achieve accurate and detailed 3D modelling of trees. On the other side, data quality is also an important issue that affects the result of modelling. Since occlusion cannot be avoided during the tree scanning process, it is common to obtain tree point clouds with missing data and incomplete branches, thus making the problem more difficult to tackle. To sum up, 3D tree modelling is a valuable research problem. However, multiple challenges need to be addressed to solve this problem.

## 4 Methodology

To achieve our research objectives, the proposed method follows several previous work, in particular Livny et al. (2010). Since the reconstruction results of Livny et al. (2010) don't fit well to the input point clouds, we will further explore cylinder fitting to improve the reconstruction accuracy. Our proposed tree reconstruction approach has four major steps. Firstly, Minimum Spanning Tree (MST) will be extracted from the input point cloud. Secondly, the tree branch skeleton will be reconstructed from the MST graph. Thirdly, branch geometry is obtained by fitting cylinders to each branch. Finally, random tree leaves are synthesized to add realism to tree model. After the tree modelling process, we assess the quality of the reconstructed models by computing their distance to the input point clouds. The overview of the methodology is depicted in Figure 2.

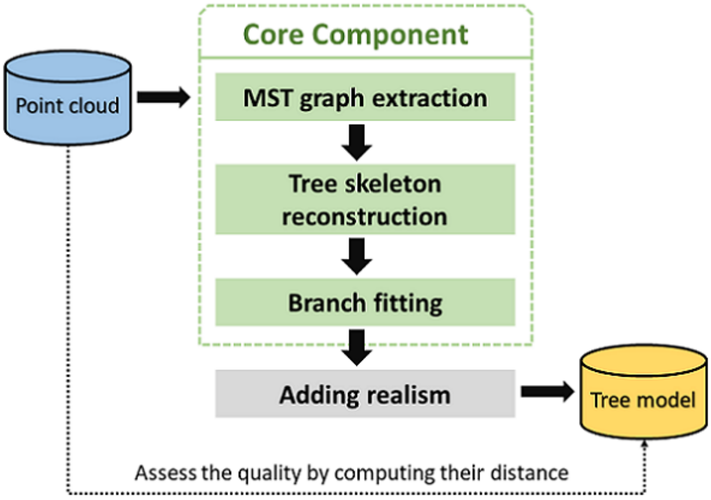


Figure 2: An overview of the proposed approach

### 4.1 MST graph extraction

The Minimum Spanning Tree (MST) is a spanning tree of an undirected and weighted graph where the sum of the edges weights is minimized. It can roughly estimate the intrinsic structure of a dataset (Zhong et al. 2015). An MST graph is very suitable for representing the tree skeletal structure, as it constructs the least-expensive path for the tree points. To extract the MST graph from point clouds, two steps need to be taken:

Firstly, we apply Delaunay triangulation to construct an initial graph from the input points. Delaunay triangulation defines a piecewise linear interpolation function and produces a well-shaped spatial tessellation (Gudmundsson et al. 2002). It lays the foundation for MST computation as most efficient approaches find a minimum spanning tree among edges in the Delaunay triangulation of the points (Zhou et al. 2001). Additionally, Delaunay triangulation helps to complete the missing region or incomplete branches, which will make the MST extraction more robust to the input point clouds with poor data quality;

After we obtain the initial graph, we weight all the edges using their lengths defined in the Euclidean space. Then the Dijkstra shortest path algorithm will be utilized to compute the minimum-weight spanning tree from the initial graph. Figure 3 shows the MST extracted from an example tree point cloud.

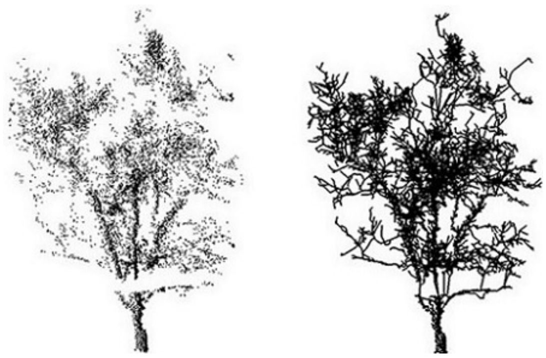

Figure 3: MST extraction (Left: input point cloud; Right: MST graph)

### 4.2 Tree skeleton reconstruction

Having extracted the MST graph from the input point clouds, the graph has a large amount of redundant vertices and edges. Most of the redundant vertices and edges don't contribute to the tree skeleton shape and thus are of little importance. Those unimportant components should be removed to simplify the tree graph. The simplification is conduct by checking the proximity between adjacent vertices. Two scenarios need to be considered:

- If the vertex to be checked has only one child vertex, the proximity to its child will be checked. Figure 4 (left) shows how the proximity is measured. If the radial distance from the child vertex to the current edge is within a specified distance threshold (i.e., the child vertex is close enough to the current vertex), we merged the two vertices;
- If the vertex has multiple child vertices, we still use the similar approach as above, but we merge only the one with the smallest distance. We do this iteratively until no more vertices can be merged.

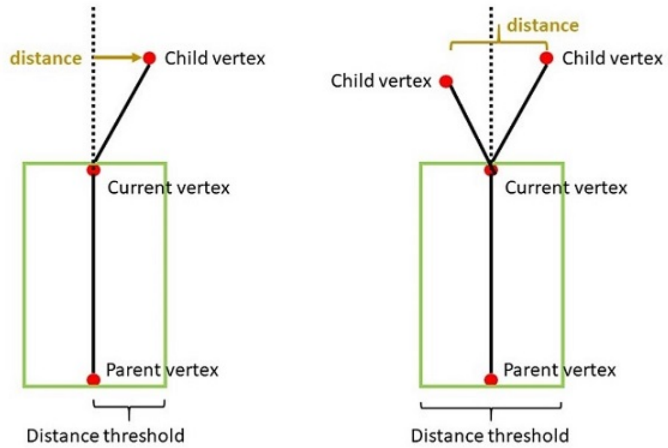


Figure 4: Vertices merge strategy (Left: single child; Right: multiple children)

The distance threshold of an edge in the tree graph controls the simplification process. It can be obtained from a rough estimation of the stem edge radius. Specifically, each edge is assigned with a corresponding radius proportional to its subtree length. After the tree graph simplification, we obtain a concise tree skeletal structure. Figure 5 shows such an example.

Figure 5: Tree skeleton reconstruction (Left: MST graph; Right: tree skeleton)

### 4.3 Branch fitting

In this step, we fit accurate branch geometry to all the tree branches. Similar to Hackenberg et al. (2014), we fit cylinders to approximate the geometry of the tree branches. We will apply a global optimization-based approach to obtain the accurate branch geometry. The objective function will be defined as the sum of the squared distance from the points to the branch cylinder. This is a typical non-linear least squares problem and we can use the Levenberg-Marquardt algorithm to solve it (Moré 1978). If the branch doesn't have enough points to support the fitting process (i.e., incomplete branches or missing data), we will assign the branch with a proper radius derived from the radius of its adjacent branches. Figure 6 shows such an example.

Figure 6: Geometric fitting for branches (Left: tree skeleton; Right: tree model with branches)

### 4.4 Adding realism

To further add realism, we add leaves to the reconstructed tree models. Since it's almost impossible for us to capture the geometry and texture characteristics of leaves using TLS technology, we are unable to reconstruct accurate leaves purely from tree point cloud. Therefore, leaves reconstruction is out of the research scope. However, we would like to render leaves to convey the realism of the model. We generate random oriented leaves at the end of each branch. The leaves will be rendered as textured quads. Figure 7 shows the final reconstruction.

Figure 7: Adding synthesized leaves to the tree model (Left: tree model with branches; Right: final model with leaves)

# 5 Time Schedule

To achieve the research objectives, the following schedule is set up which includes multiple activities, as shown in Table 1.

| Start | End | Activities |
|---|---|---|
| 15 Oct | 10 Nov | Exploring possible thesis topics |
| | | **P1 - Progress review Graduation Plan** |
| 15 Nov | 31 Dec | Literature study |
| 15 Nov | 15 Dec | Minimum Spanning Tree extraction |
| 16 Dec | 31 Dec | Explore MST graph simplification and prepare for P2 |
| | | **P2 - Formal assessment Graduation Plan** |
| 15 Jan | 31 Jan | Implement MST graph simplification |
| 1 Feb | 18 Feb | Implement tree branch fitting |
| | | **P3 - Colloquium midterm** |
| 1 Mar | 30 Mar | Synthesize leaves; Integrate all steps into a software prototype |
| 1 Apr | 12 May | Implement tree rendering; Write manuscript for publication; Write thesis |
| | | **P4 - Formal process assessment** |
| 15 May | 22 Jun | Finalize thesis and prepare for the presentation |
| | | **P5 - Public presentation and final assessment** |

Table 1: Schedule plan

# 6 Tools and Data

## 6.1 Tools

The programming language to be used is C++. The development environment is Microsoft Visual Studio 2017. To enhance the modelling visualization and user interactivity, the Qt library will be used for User Interface design. Mapple (a tool for visualizing and editing 3D point clouds) will be used for the visualization and segmentation individual trees from the point clouds).

## 6.2 Data

To develop and test the proposed tree reconstruction method, several point cloud datasets have been collected. These test datasets contain point clouds from publicly available point cloud repositories, the Floriade Project of Almere, and the AHN dataset. These point clouds cover a wide variety of the date sources, i.e., static laser scans, mobile laser scans, and airborne laser scans.

## References


Bucksch, A., Lindenbergh, R. C., and Menenti, M. (2009). Skeltre-fast skeletonisation for imperfect point cloud data of botanic trees. Eurographics.

Calders, K., Origo, N., Burt, A., Disney, M., Nightingale, J., Raumonen, P., Åkerblom, M., Malhi, Y., and Lewis, P. (2018). Realistic forest stand reconstruction from terrestrial lidar for radiative transfer modelling. *Remote Sensing*, 10(6):933.

Cornea, N. D., Silver, D., and Min, P. (2007). Curve-skeleton properties, applications, and algorithms. *IEEE Transactions on Visualization & Computer Graphics*, (3):530–548.

Deussen, O., Hanrahan, P., Lintermann, B., Měch, R., Pharr, M., and Prusinkiewicz, P. (1998). Realistic modeling and rendering of plant ecosystems. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 275–286. ACM.

Gobeawan, L., Lin, E., Tandon, A., Yee, A., Khoo, V., Teo, S., Yi, S., Lim, C., Wong, S., Wise, D., et al. (2018). Modeling trees for virtual singapore: From data acquisition to citygml models. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 42.

Gudmundsson, J., Hammar, M., and van Kreveld, M. (2002). Higher order delaunay triangulations. *Comput. Geom.*, 23(1):85–98.

Hackenberg, J., Morhart, C., Sheppard, J., Spiecker, H., and Disney, M. (2014). Highly accurate tree models derived from terrestrial laser scan data: A method description. *Forests*, 5(5):1069–1105.

Liang, X., Kankare, V., Hyyppä, J., Wang, Y., Kukko, A., Haggrén, H., Yu, X., Kaartinen, H., Jaakkola, A., Guan, F., et al. (2016). Terrestrial laser scanning in forest inventories. *ISPRS Journal of Photogrammetry and Remote Sensing*, 115:63–77.

Livny, Y., Pirk, S., Cheng, Z., Yan, F., Deussen, O., Cohen-Or, D., and Chen, B. (2011). *Texture-lobes for tree modelling*, volume 30. ACM.

Livny, Y., Yan, F., Olson, M., Chen, B., Zhang, H., and El-Sana, J. (2010). Automatic reconstruction of tree skeletal structures from point clouds. In *ACM Transactions on Graphics (TOG)*, volume 29, page 151. ACM.

Moré, J. J. (1978). The levenberg-marquardt algorithm: implementation and theory. In *Numerical analysis*, pages 105–116. Springer.

Olofsson, K., Holmgren, J., and Olsson, H. (2014). Tree stem and height measurements using terrestrial laser scanning and the ransac algorithm. *Remote sensing*, 6(5):4323–4344.

Raumonen, P., Kaasalainen, M., Åkerblom, M., Kaasalainen, S., Kaartinen, H., Vastaranta, M., Holopainen, M., Disney, M., and Lewis, P. (2013). Fast automatic precision tree models from terrestrial laser scanner data. *Remote Sensing*, 5(2):491–520.

Reche-Martinez, A., Martin, I., and Drettakis, G. (2004). Volumetric reconstruction and interactive rendering of trees from photographs. In *ACM transactions on graphics (ToG)*, volume 23, pages 720–727. ACM.

Verroust, A. and Lazarus, F. (1999). Extracting skeletal curves from 3d scattered data. In *Proceedings Shape Modeling International'99. International Conference on Shape Modeling and Applications*, pages 194–201. IEEE.

Wang, D., Hollaus, M., Puttonen, E., and Pfeifer, N. (2016). Automatic and self-adaptive stem reconstruction in landslide-affected forests. *Remote Sensing*, 8(12):974.

Xie, K., Yan, F., Sharf, A., Deussen, O., Huang, H., and Chen, B. (2016). Tree modeling with real tree-parts examples. *IEEE transactions on visualization and computer graphics*, 22(12):2608–2618.

Zhong, C., Malinen, M., Miao, D., and Fränti, P. (2015). A fast minimum spanning tree algorithm based on k-means. *Information Sciences*, 295:1–17.

Zhou, H., Shenoy, N., and Nicholls, W. (2001). Efficient minimum spanning tree construction without delaunay triangulation. In *Proceedings of the 2001 Asia and South Pacific Design Automation Conference*, pages 192–197. ACM.