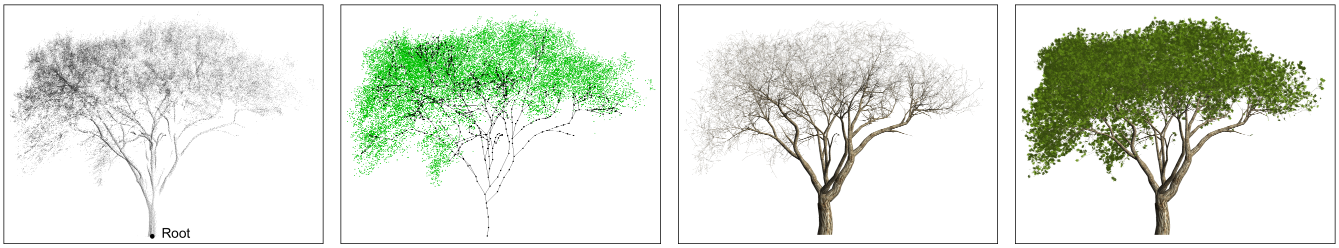
Knowledge-Based Modeling of Laser-Scanned Trees

Hui Xu*
University of Minnesota

Nathan Gossett*
University of Minnesota

Baoquan Chen*
University of Minnesota



(a) (b) (c) (d)

Figure 1: Modeling a real tree: (a) A point cloud produced by a single scan of a real tree. (b) A skeleton is produced from the graph, and leaf locations (green dots) are identified. (c) The skeleton is extended by synthesizing new small branches growing into the tree crown and a mesh is reconstructed based on the skeleton. (d) Leaves are added and the full tree model is completed.

1 Introduction

The increasing availability and power of range scanners has enabled us to scan larger and more complex objects. Trees, however, pose special problems for such scanning. Unlike buildings, trees have relatively more complex geometry, and the scanned tree data is inherently incomplete due to limited scanning resolution and occlusion from leaves and branches. Standard mesh generation techniques often fail because of this inadequate data.

Our purpose is to use knowledge about the structure of trees and allometric theory to produce full polygonal meshes from the point clouds obtained from the scans. We aim to produce a mesh that plausibly recreates the tree that was originally scanned, but with extrapolated details that were missing in the point cloud. These surrogate models can then be used in place of the original point clouds for rendering.

2 Overview

Given an under-sampled point cloud of a laser-scanned tree as shown in Figure 1(a), a skeleton of the trunk and main branches of the tree is first produced (Figure 1(b)). Steps are then taken to synthesize additional branch skeletons to produce plausible support for the tree crown. Appropriate dimensions for each branch section are estimated using allometric theory. Using this information, a mesh is produced around the full skeleton (Figure 1(c)). Finally, leaves are connected to nearby branches (Figure 1(d)).

To produce a skeleton of the trunk and main branches, we connect every point to all other points within its local neighborhood, forming a weighted graph. The weight of each edge in the graph is defined by its length. Due to the inherent incompleteness of the input data, the graph is usually disconnected, but has many connected subgraphs. We start producing the skeleton from the connected subgraph containing the root point. To do so, shortest paths from the root to all other points in the subgraph are calculated. The lengths of the shortest paths are quantized and the points are clustered into bins based on these lengths. The skeleton is formed by connecting the centroids of adjacent bins. In the next step, we extend this skeleton to include all other connected subgraphs that also represent main branches. We use a bi-directional probing process to find such subgraphs. Their skeletons are then produced using the same algorithm described above and connected to the main skeleton. The remainder points of the tree are classified as *leaf locations*.

The small twigs in the crown are usually inadequately sampled, so it is impractical to construct such fine branches from the point cloud. Therefore, to produce plausible support for the tree crown, we synthesize small branches based on the existing skeleton iteratively. If a leaf location is far away from the existing skeleton, it is called *unsupported*. The branch synthesis process is driven by the elimination of unsupported leaf locations. In each iteration, skeletons from some parts of the existing skeleton are selected, copied, scaled, transformed, and attached to the main skeleton to eliminate as many unsupported leaf locations as possible. The iteration process stops when there are no more unsupported leaf locations or a maximal number of iterations is reached.

Once a full skeleton has been produced, we determine the thickness of each branch section and construct a mesh along the skeleton. Due to the incomplete and under-sampled data, only the base of the trunk and possibly the larger sections of the lower branches are scanned in sufficient detail to determine thickness. Therefore, we only estimate the thickness for the trunk and use allometric theory to produce plausible dimensions for the rest of the branches.

Finally, leaves, modeled using alpha textured quads, are oriented and connected to the nearest branches to complete the tree model. Our plausible tree models can then be inserted to other virtual scenes as shown in Figure 2, and can also be used in place of their original point clouds in a scanned environment.

Figure 2: Plausible tree models are inserted in the Ephesos excavation site scanned in Turkey (provided by Michael Wimmer).

*e-mail: {hxu,gossett,baoquan}@cs.umn.edu