Volume xx (200y), Number z, pp. 1–13

Variational Tree Synthesis

Rui Wang, Yinhui Yang, Hongxin Zhang†, and Hujun Bao

State Key Lab of CAD&CG, Zhejiang University, China

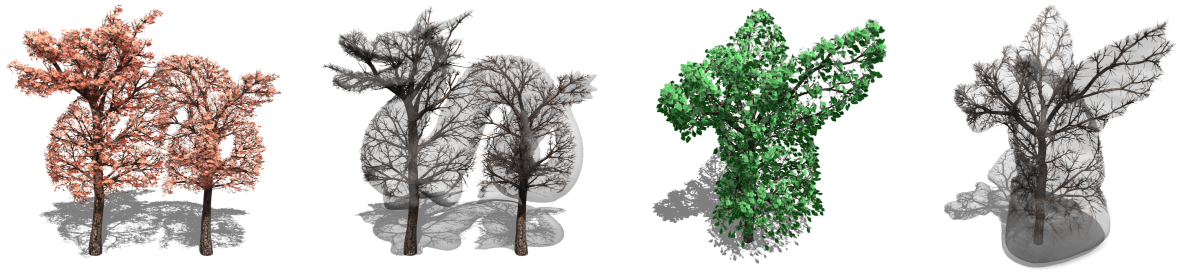


Figure 1: *Shape-guidance tree modeling using our approach. Left: The Dragon model as a guidance shape. Right: The Gargoyle model as a guidance shape. Note that small details of the shapes, such as the tongue and tail of the Dragon and the ears and wings of the Gargoyle, are faithfully generated by our method.*

Abstract

Modeling trees according to desired shapes is important for many applications. Despite numerous methods having been proposed in tree modeling, it is still a non-trivial task and challenging. In this paper, we present a new variational computing approach for generating realistic trees in specific shapes. Instead of directly modeling trees from symbolic rules, we formulate the tree modeling as an optimization process, in which a variational cost function is iteratively minimized. This cost function measures the difference between the guidance shape and the target tree crown. In addition, to faithfully capture the branch structure of trees, several botanical factors, including the minimum total branches volume and spatial branches patterns, are considered in the optimization to guide the tree modeling process. We demonstrate that our approach is applicable to generate trees with different shapes, from interactive design and complex polygonal meshes.

Keywords: tree modeling,variational model,mesh generation

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—I.3.6 [Computer Graphics]: Methodology and Techniques—Interaction Techniques I.6.8 [Simulation and Modeling]: Types of Simulation—Visual

1. Introduction

Realistic tree models are essential for many computer graphics applications. Numerous methods and pieces of software have been proposed to synthesize realistic trees. However, with the rapid development of applications, special requirements arise on the shapes of synthesized trees [SLS*10].

L-Systems [Lin68, PL90] are typical and powerful solutions for tree modeling. Although the original rule-based generative models of L-Systems are capable of generating various types of trees, it is limited at modeling trees under specific global constraints such as crown shapes. Extra efforts [PJM94, RLP07, PHL*09] have been made to en-

† {rwang,yangyinhui,zhx,bao}@cad.zju.edu.cn, corresponding author: Hongxin Zhang

hance the original L-Systems for better shape control. These methods, in a manner, adopted self-organizing and locally adaptive strategies to generate trees may require a priori knowledge of how different species of trees response to the environment. Sketch-based modeling [OOI06, CNX*08, LRBP12] , however, from the view angle of user interface, a trade-off has to be made between the amount of user-involved sketching and the fidelity of resultant models. Thus, it is still meaningful and valuable to explore new tree modeling methods with new perspective.

In the physical world, there exists diverse branching systems and many of them share a tree-like structure. Irrigation and draining systems, plants and trees together with their root systems, lungs and cardiovascular systems have a common morphology which seems to derive from topological constraints together with energy saving requirements [MTM03]. Considering such requirements, there exists minimum energy of branch structures in biology, such as the minimum length, area or volume of branches and etc [Mur27, Leo71, Zam76]. Furthermore, the work of [WZY01] indicated that the model of volume minimization is better than other cost minimization models in fitting data of branching structures of plants.

In this paper, inspired by previous works on branch structure generation [RFL*05, HGP05], we propose a new variational tree modeling approach targeting on generating realistic trees in specific shapes. Instead of directly modeling trees from symbolic rules, we formulate the shape-guidance tree modeling problem in a multi-level variational optimization framework. At each level, a variational cost function is defined on the difference between the guidance shape and the target tree crown which is decomposed into a set of subtrees. To faithfully capture the branch structure of trees, each subtree is parameterized in several botanical parameters and constructed under botanical factors, including the minimum total branches volume and spatial branches patterns. These constraints are all considered in the optimization to guide the synthesis of branch structures. To optimize such a variational problem efficiently, we develop a three-step iterative algorithm. At the first step, initial tree branches are generated. At the second step, we fix parameters of subtrees and cluster sample points on shapes to find the best shape approximation of subtrees. In the final step, subtrees are optimized to best fit these clustered sample points in the previous step. After several iterations, the final optimized tree is obtained. In Figure 3, the overall process and the resulting tree structures are illustrated.

To demonstrate our variational tree model and the optimization framework, we develop a prototype system for modeling trees. Guidance shapes are given by sketches of users or from complex polygonal meshes. An overview of our modeling system is shown in Figure 2. The system takes the user specified guidance shape and botanical rules as inputs to generate an optimized tree skeleton. Then, branch

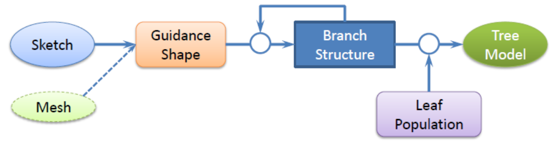


Figure 2: *Overview of our modeling system.*

surfaces are constructed and leaves are populated to synthesize a final tree. Trees generated in Dragon and Gargoyle shapes are illustrated in Figure 1.

2. Related Work

Tree modeling is an active research topic and receives considerable attention in the past decades [Hon71, RB85]. L-Systems, as a traditional tree modeling solution, has been widely studied. We recommend readers to [PL90] for a systemic introduction and [DL05] for work in this category of approaches. The fundamental mechanism of L-Systems is an iterative system based on a small set of symbolic rules. It works well for modeling of trees, especially at synthesizing local botanical structure details. This makes it an elegant solution for generating complex tree structures in many applications. However, one main disadvantage of conventional L-Systems is the lack of ability to control the growth of trees under certain shape constraints.

Such a limitation motivated a lot of tree modeling techniques. Prusinkiewicz *et al.* [PJM94, MP96] extended the L-Systems to the interaction with environments and modeled a topiary tree by pruning branches that grow outside a predefined shape. Our variational approach directly optimizes branches into particular forms rather than pruning. Runions *et al.* [RLP07] proposed a space colonization method but it did not consider the botanical constraints of branches patterns, e.g. phyllotaxy. Then, a generative model with more botanical rules was presented in [PHL*09] to solve the space colonization problem. In such a generative model, shape constraints are optimized by local branches growth and competition. Compared with bottom-up methods, our variational approach employs a global optimization that minimize the variational error top-down so as to obtain a global optimal tree structures under the shape constraints.

Talton *et al.* [TLLD11] developed the first inverse L-Systems method based on Markov Chain Monte Carlo (MCMC) optimization that iteratively fits a given desired shape. The objective function of [TLLD11] in terms of post-prior is mainly measured by the shape difference. Their method does not explicitly optimize the energy of branch structures, but our method does. Moreover, using the MCMC, their optimization requires hundreds and thousands of iterations to converge which is a great limitation for its usage in many applications.

In [PSK*12], a static input tree model is analyzed and

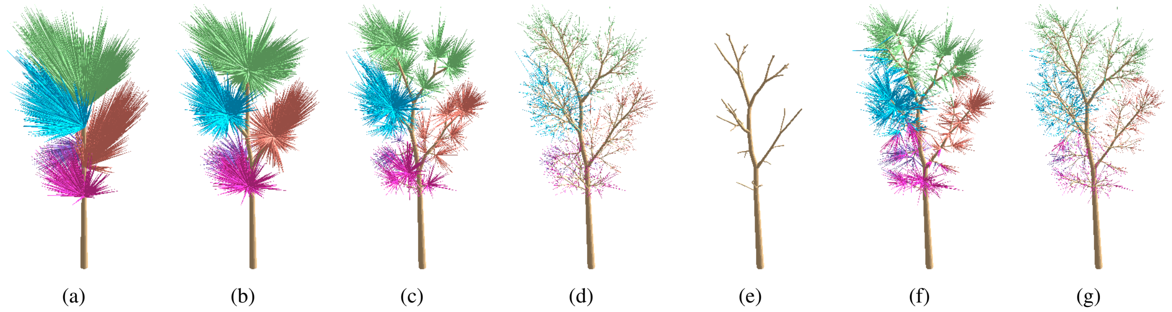


Figure 3: *An overview of our variational tree synthesis algorithm. (a) The shape nodes are clustered and directly connected to the initial structure to form a k-partition. The initial structure is shaded in a color of burly wood and subtree clusters are indicated by different colors. (b) Updated subtrees after optimization. (c) The intermediate branching structure after three levels. (d) The branching structure after reaching a local minima. (e) Branches under a certain order are pruned. (f) The shape nodes are re-connected to their closest branch node in the remaining structure from (e). (g) The final tree obtained after 32 steps of iterations through (a) to (f).*

the structure of the tree can be modified according to the change of the environment. Also, [PNDN12] proposed an automatic method for analyzing a static tree model to generate developmental stages and the tree can be edited by applying growth development to parts of the tree. While these methods aimed to create models from a small number of existing static tree models, our variational approach is dedicated to generate such static tree models from scratch.

[FG00] proposed a method to compare plant individuals based on a distance metric, which is defined as the minimum cost of the sequence of elementary edit operations needed to transform one tree graph into another. The variational cost defined in our work is a combination of the shape constraint and several botanical factors. The tree structure is generated by minimizing such a cost.

Stroke-based sketch methods, e.g. [BPF*03, OOI06, IOI06, CNX*08, TFX*08, WBCG09], provided a designing-friendly way to generate tree models guided by strokes. These well-designed user interfaces greatly enhanced the L-Systems approach. But the geometry shape of the tree models still cannot be fully constrained or optimized. Moreover, compared with our method, branch structures in [WBCG09] are specified by sketches in a step-by-step manner. While their method infers the branches in 2D and then extends them to 3D, our method is fully performed in 3D. The latest work along such a direction is an interactive procedural modeling approach to model trees by sketches on a tablet [LRBP12]. A friendly interface and the fast generation of trees make this method be a powerful tree modeling tool. However, since the tree model is extended from [PHL*09], such a method bears the similar limitations of that in [PHL*09]. Compared with these sketch-based plant modeling approaches, our method has a similar motivation that generates trees by an inverse modeling process. But, instead of generating trees from rules, our method provides an alternative and new tree modeling approach.

Besides users' interactive input, many researchers proposed tree reconstruction methods to model trees from acquired data, such as from images [SRDT01, NFD07], video data [TZW*07, LDS*11], or from scanned point sets [XGC07, BLM09] and the recent work is [LYO*10, LPC*11]. These methods tend to accomplish a reconstruction mission rather than a modeling issue. Only a few of them [LYO*10, LPC*11] are aware of alternative ways to globally optimize and approximate shapes by branch patterns. However, the branching structure in [LYO*10, LPC*11] is not generated from an unified cost function. Instead, in each optimization iteration, they first connect scanned points using distance function to obtain the branching structures, and then use other specific botanical criteria to refine them. In contrast, our method employs an unified cost function to optimize the branching structures in a variational way. This makes our method be able to synthesize trees only using shape guidance.

The research on variational approximation, which inspires our approach, has existed in different areas. Cohen-Steiner *et al.* [CSAD04] proposed a method to approximate shape by planes. Sphere set [WZS*06] and ellipsoids [LCWK07] are then used to approximate shapes. Compared with these parametric primitives, the morphological and botanical structure of a tree to be synthesized in this paper is more complicated and challenging, which requires new optimization model and framework.

3. Variational Tree Synthesis Model

In this section, we present a computational model for the modeling of botanical branching structures. To precisely explain the idea, we introduce several related terminologies first.

3.1. Preliminaries

In a tree structure, we group an insertion zone of a leaf on the stem as a *node*. The stem portion between two successive nodes is called *internode*. The basic structural unit of a plant body is known as *metamer* formed by a node, associated with its leaf (or leaves) and axillary bud(s) plus the subtending internode [BC07].

For computing purpose, we define an abstract branching model $\mathcal{B}$ of tree as a quadruples:

$$\mathcal{B} := < \mathcal{V}, \mathcal{P}, \mathcal{D}, \mathcal{T} >, \quad (1)$$

where $\mathcal{V} := \{v_i\}_{i=0}^{\mathbf{N}}$ defines $\mathbf{N}+1$ node indices in the tree structure with v_0 representing the root node, $\mathcal{P} := \{\mathbf{p}_i \in R^3\}_{i=0}^{\mathbf{N}}$ and $\mathcal{D} := \{d_i > 0\}_{i=0}^{\mathbf{N}}$ represent the corresponding spatial positions and the diameters of the node set $\mathcal{V}$, respectively. The topological relationship of $\mathcal{V}$ in $\mathcal{B}$ is then encoded in the tree data structure

$$\mathcal{T} := \{B_{ij} := v_i \to v_j | v_i, v_j \in \mathcal{V}, v_j \text{ is a child of } v_i\},$$

which forms a special case of a directed acyclic graph (DAG). Note that $\mathcal{B}$ is an approximation of a real tree by representing a given tree branch segment as an ideal cylinder.

3.2. Botanical Rules

Different from general graph or previous variational shape approximation problems [WZS*06], our optimization targets are realistic tree structures. Therefore, we incorporate several botanical rules in the optimization and generate branch structure patterns of trees instead of dealing with symbolic iterations in L-Systems.

First of all is the *pipe model*, which is proposed in botanical literature [SYHK64] according to accumulated observation data over years. In this model, a tree is idealized as a complex flow transmission system consisting large amount of pipes. Intuitively speaking, each unit pipe supports a constant number of leaves or a set of "photosynthetic organs".

Each pipe connects a leaf to the tree's trunk, i.e. the pipe starts at a corresponding leaf and goes down the entire length of the tree to the base of its trunk. Thus, a trunk can be seen as a bundle of pipes connecting leaves. According to the statistical data, it is known that the diameter of a parent branch has a relationship with its child branches [Mac83]. This gives one botanical constraint on the branching structure:

$$d_i^{\alpha} = \sum_{B_{ij} \in \mathcal{T}} d_j^{\alpha}, \quad (2)$$

where α is a branch transmission coefficient which differs in tree species. In our approach, α is taken between $(2.0, 3.0)$ according to different type of trees.

Following the analysis of plant architecture in [BC07], the major morphological features of a tree are parameterized in several parameters listed in Table 1. The usage of such parameters to generate branches is illustrated in Figure 4. In this representation, parameter N denotes the metamer numbers of a stem and works as a branch density control. The form of woody plants is determined by the differential elongation of buds and branches, and the expression of a particular growth habit is usually associated with the phenomenon of apical dominance. The apical dominance connotes bud inhibition by an active apex on currently elongating shoots [BMK67]. Given a shape constraint, parameter $P \in (0.0, 1.0]$ is a depiction of the fate of a stem's apical bud and is defined as the ratio of the actual length of the stem to its maximal length.

Parameter	Description
N	number of metamers on current axis
P	the range of a stem formed by apical growth process
n	number of sub-branches
ϕ	rotation angle of leaves associated with two successive nodes
θ	angle between a lateral branch and its bearing axis

Table 1: *Botanical parameters of branches structure.*

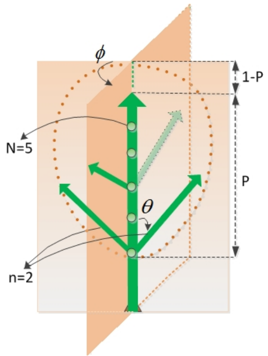


Figure 4: *Branch structure generated by botanical parameters.*

Phyllotaxy is an important factor to model the arrangement of leaves on a stem [DC96]. Phyllotaxic patterns are generated whenever a vascular plant repeatedly produces similar botanical elements at its tip, i.e., leaves, branches and florets etc. As the position of the leaf upon the stem marks the position on the axillary bud, it follows that the order of the leaf-arrangement will be the order of the branches also. We therefore employ n to denote the number of leaves associated with a node. Similar to [VEJA89], the angle coefficient θ presents the branching angle or insertion angle which is the angle between a lateral branch and its bearing axis. The coefficient ϕ is used for specifying the rotation angle of branches associated with two successive nodes.

Hence we represent the structure of tree branches in a 5-

dimensional vector as

$$\mathbf{R} = (N, P, n, \theta, \phi),$$

in the botanical parameter space. In our computing model, a tree is represented as a hierarchical structure, taking the notion of 'levels of recursion' in [WP95]. The trunk is corresponding to the first level and the next level consists of all the child branches that are directly connected to the current level. Branches in different level of a tree hierarchy can be either classified with a similar set of parameters $\mathbf{R}_i$ or specified with different values $\mathbf{R}_j$, where $i < j$ for endogenous effects. To simplify the representation, we combine all parameter vectors of different levels to be a botanical arguments set $\mathcal{R} = \{\mathbf{R}_0, \mathbf{R}_1, ...\mathbf{R}_L\}$. In general, 2 or 3, mostly 6, levels are good enough for tree modeling in this paper. The actual branches positions of subtrees will be computed from the variational model introduced in the following section.

3.3. Variational Model

To formulate our variational computing approach, we use following notions to define a *guidance shape* which is usually the description of the desired tree crown, see Figure 5 (a). That is we collect a number of small spheres (called *shape nodes*) to form the constraint set $\mathcal{C}$, i.e., the guidance shape:

$$\mathcal{C} = \{\mathcal{U}, \mathcal{Q}, \mathcal{D}\}.$$

where $\mathcal{U} := \{u_i\}_{i=0}^{M}$ defines $M+1$ nodes to represent the guidance shape in discrete form, $\mathcal{Q} := \{\mathbf{q}_i \in R^3\}_{i=0}^{M}$ and $\mathcal{D} := \{d_i > 0\}_{i=0}^{M}$ represent spatial positions and the diameters of corresponding nodes.

Our primary modeling goal is to approximate the input guidance shape $\mathcal{C}$. Thus, we propose the following optimization problem:

$$\mathcal{B}_{opt} = \arg\min_{\mathcal{B}} F(\mathcal{B}; \mathbf{p}_0, \mathcal{C}, \mathcal{R}) \tag{3}$$

where $F(\cdot)$ is an objective function to present modeling cost. Function $F(\cdot)$ consists of the intrinsic structure energy F_I, the exterior shape-guidance energy F_E due to the guidance shape $\mathcal{C}$, and the parameter control penalty F_P from the botanical parameter set $\mathcal{R}$. Note $\mathbf{p}_0$ is the position of the root node. By assembling these three terms together, we have

$$F(\mathcal{B}; \mathbf{p}_0, \mathcal{C}, \mathcal{R}) = F_I + \lambda F_E + \mu F_P \tag{4}$$

where λ and μ are two normalization terms for F_E and F_P respectively to unify the dimensionality.

The intrinsic structure energy, F_I, is defined based on a widely existing observation that the geometrical structure of branching of arteries, rivers and trees is an optimized structure as stated previously. According to the accumulated data of years in botanical research [Mur27, WZY01], the most wildly applied model to depict the optimized structure of tree is to minimize the total volume of branches. Thus, mathematically, we define the intrinsic energy of branching structure as

$$F_I(\mathcal{B}) = \int_{\mathcal{B}} dV \simeq \sum_{B_{ij} \in \mathcal{T}} \mathrm{Vol}(B_{ij}). \tag{5}$$

The volume function for a given internode B_{ij} is calculated by

$$\mathrm{Vol}(B_{ij}) = \|\mathbf{p}_i - \mathbf{p}_j\| d_j^{\beta}. \tag{6}$$

where d_j is computed by Equation (2) in the pipe model, shape coefficient β is employed for different cost models. Therefore the cost function is adopted to minimize total surface area when $\beta = 1.0$, and to minimize total volume when $\beta = 2.0$. And some authors proposed to use $\beta = 2n - 2$ for n-split branches [RW82]. In general, $\beta = 2.0$ is sufficient to give satisfied results, but the value of α in Equation 2 should be greater than β to make sure the cost of F_I will be decreased according to the three-step optimization described in Section 4. Note that we omit the constant $\pi/4$ as it does not affect the final optimization results.

To determine the shape-guidance energy, F_E, we define a distance-aware measurement to compute the similarity between the target structure $\mathcal{B}$ and the shape constraint set $\mathcal{C}$

$$F_E(\mathcal{B}; \mathbf{p}_0, \mathcal{C}, \mathcal{R}) := \sum_{i=1}^{M} d_i \min_j \|\mathbf{q}_i - \mathbf{p}_j\| / \sum_{i=1}^{M} d_i. \tag{7}$$

For the energy from the botanical parameter control penalty, F_P, it is also formulated as a similarity measurement

$$F_P(\mathcal{B}; \mathbf{p}_0, \mathcal{C}, \mathcal{R}) :=$$
$$\sum_{B_{ij} \in \mathcal{T}} [\| \bar{\theta}_{ij} - \theta_{ij} \| + \| \bar{\phi}_{ij} - \phi_{ij} \| + \| \bar{l}_{ij} - l_{ij} \|], \tag{8}$$

where l_{ij} is the length of B_{ij} and symbols with *bar* are denoted for the evaluated parameters in $\mathcal{B}$ by minimizing Equation (10).The integer parameter N and n are used to generate an initial topology, on which our constraints are imposed. We omit the comparison of these two parameters in F_P. The function F_p is applied as a soft constraint term to enforce the synthesized result $\mathcal{B}$ according to the structure pattern which is defined by $\mathcal{R}$.

4. Optimization Algorithm

Based on our variational model, the optimization goal is challenging since we need to optimize topology structure $(\mathcal{V}, \mathcal{T})$ and geometry parameters $(\mathcal{P}, \mathcal{D})$ simultaneously. With a fixed set of parameters $\mathcal{R}$, the above problem is equivalent to a weighted version of the famous Steiner tree problem [HR92] which is NP-complete. It turns out that directly solving the optimization problem is time consuming. Therefore, in this paper, we present a greedy optimization algorithm to iteratively minimize the structure cost function F.

4.1. Algorithm Overview

Our algorithm starts from an initial branch structure $\mathcal{B}_0$, in which all branches are directly generated by the botanical arguments, $\mathcal{R}_0$, see Figure 4 for an illustration. Then, a tree hierarchy is progressively optimized from the root level to the leaf level. In each level, it is still hard to minimize the three cost functions at the same time. In Figure 5, a procedure of two level optimization is illustrated.

Inspired by previous variational approaches [CSAD04], we take a three-step optimization. The first step is *structure initialization*. A simple structure $\mathcal{B}_i$ is initialized in which all branches are directly generated by the botanical arguments, $\mathcal{R}_i$. The second step is *shape partition*. We fix the positions of branches and minimize the cost function to find the best guidance shape for each subtree (Figure 3(a)). Then the final step is *structure update*. We fix the shape partition and only update positions of branches (Figure 3(b)). These later two steps are iteratively carried out until there is no noticeable change of the structure or a fixed iteration number is reached. After the optimization taken on all levels (Figure 3(c)-(d)), to avoid the local minima trap for our greedy strategy, we also integrate an additional global topological refinement step (Figure 3(e)-(f)).

4.2. Iterative Optimization at One Level

For each level of the tree, we need to find a number of suitable intermediate *branch nodes*. For each branch node v_j ($j = 1,\ldots,\mathbf{N}$), it connects a cluster of shape nodes which constitutes a sub-tree $\#j$ rooted at node v_j (Figure 5 (c)). The optimization needs to solve three issues: (1) how many intermediate branch nodes we need, (2) how to approximate the guidance shape and, (3) where to locate the optimal branch nodes. There are three corresponding steps in our algorithm which can solve these issues one by one.

Structure initialization. Recall that we use five parameters to define the branching structure in one specific level (see Figure 4). These botanical parameters are used to create the initial topology and nodes' positions of branches in the current level of the tree. First, we generate a main stem with $N+1$ nodes. A maximum length of the stem can be derived from the guidance shape and the actual length of the stem is computed by scaling it according to P. Then the positions of the nodes are set to be evenly distributed on the main stem. Second, we attach n lateral branches for each node on the main stem and the initial direction of each lateral branch can be computed according to θ and ϕ. In Figure 5 (b), we illustrate one case for an example. Note $\mathcal{B}_{i+1}$ contains all the nodes in $\mathcal{B}_i$.

Shape partition. To solve the second issue, we recall the variational cost function that we need to solve. According to Equation (4), it can be observed that given the branch structure $\mathcal{B}$, the cost function F_I and F_P are determined, the cost function to be minimized is only the shape-guidance energy, F_E.

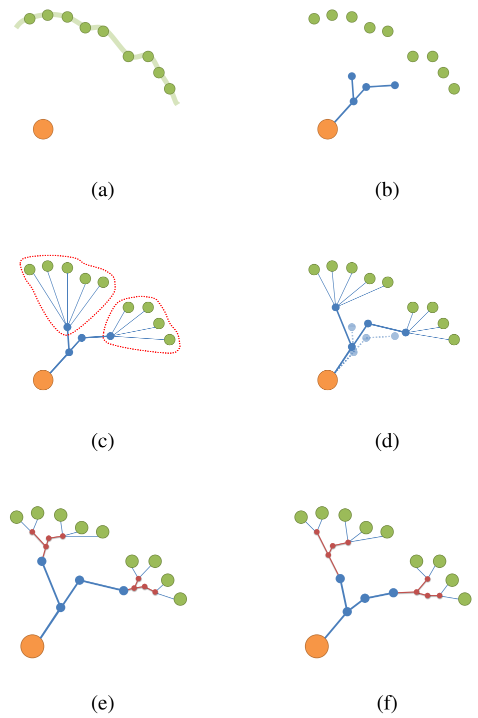


Figure 5: *Branching structure optimization. (a) Input guidance shape, the nodes in green are the shape nodes and the root node* $\mathbf{p}_0$ *is in orange. (b) After the structure initialization, initial topology and nodes of branches are generated according to* $\mathcal{R}_0$*. (c) After the shape partition, shape sample nodes are assigned to different branch nodes and each cluster indicated by the red dashed line constitutes a subtree. (d) After the structure update, the positions of nodes are optimized. (e) When level one optimization is completed, the initial branch structures of level two are generated (nodes in dark red). (f) The final tree structure after the level two optimization. Note that the level two optimization does not only optimize nodes generated in level two, but all notes' positions are optimized.*

$$F(\mathcal{B};\mathbf{p}_0,\mathcal{C},\mathcal{R}) = F_E \tag{9}$$

As illustrated in Figure 5 (c), we form an initial partition of the shape by clustering the shape nodes according to the energy defined in F_E. Note that each cluster constitutes a subtree which is used as an initial setup for a further processing discussed below.

Structure update. According to Equation (4), once the clusters of shape nodes are determined, we can further update the branch structure of subtrees. To avoid minimizing the F_E term in this step, we directly connect shape nodes to the branches. In such a case, the optimization of internodes will not break the shape-guidance, thus, the overall cost function is

$$F(\mathcal{B};\mathbf{p}_0,\mathcal{C},\mathcal{R}) = F_I + \mu F_P \tag{10}$$

It is hard to find an analytical solution that minimizes Equation (10). We use Powell's discrete multidimensional minimization [PTVF92] to find the minimal nodes' position of branches. It is a gradient descent method with numerically gradient evaluations. In Figure 5(d), we use different color to illustrate the roles that the optimization process played on the branch structure. Note that all nodes' positions are optimized in this step. Please also refer to Figure 5(e)-(f) for a further illustration.

To minimize the global energy in Equation (4), **Shape partition** and **Structure update** are carried out iteratively. To this end, we employ a weighted variant of the Lloyd's algorithm [Llo82] which fits our condition naturally. The original Lloyd's algorithm, also known as K-means, involves dividing a set of high dimensional points into non-overlapping clusters where points belonging to a region are closer by distance-based measurement of proximity to one another than to the points in other clusters. Similarly, in our approach, every cluster can be characterized by the branch node position $\mathbf{p}_j$, and the set of all k regions is called a k-partition. This algorithm provides deterministic iterations to generate such a partitioning. The major procedure is intuitively simple: after given initial K subtrees $\{\mathbf{p}_j\}_{j=1}^{K}$, these two steps, **Shape partition** and **Structure update**, are iteratively taken. First, all the shape nodes are partitioned into K clusters by assigning each node according to F_E in **Shape partition**. Then the resultant branch node position $\mathbf{p}_j$ is updated in **Structure update**.

4.3. Topology Refinement

The proposed optimization algorithm is greedy, thus it will be stuck into local minimal. To jump out from the local minimal trap, an additional step for topology refinement is performed after a local convergent is reached. We define the *order* of a given node to be the maximal steps connecting shape nodes in $\mathcal{B}$. Using this operator, branches in smaller order are pruned away from the structure $\mathcal{B}$ and the corresponding shape sample nodes are reconnected to their nearest branch nodes in the remaining skeletal structure(Fig 3.(e)-(f)). In our experiments, we find the topology refinement is useful to escape the local minima.

To summarize aforementioned discussion, our optimization algorithm consists of two loops. The interior loop is employed to find a local minima and the exterior loop provides the mechanism for global optimization. The algorithm is terminated when no more decreasing of the structure cost. In general, it is terminated in less than 60 steps of iterations in all our presented examples.

5. System and Modeling Interface

To demonstrate our variational tree modeling approach, we develop a system to input guidance shapes, synthesizing trees and visualize them. The components of our system are shown in Figure 2. Firstly, the user needs to sketch a few strokes of a silhouette to create a guidance shape, or alternatively import a triangular mesh. Secondly, the botanical arguments $\mathcal{R}$ are specified according to the desired tree species. Finally, the branches are optimized from our variational branching model. After the optimizing process is completed, users can freely visualize the model and output desired models after the additional step of leaf population. The total process is commonly accomplished in less than several minutes even to generate a complex realistic tree model.

Sketching guidance shape. We apply a Teddy-like user interface [IMT99] for interactively modeling tree crown. In our prototype system, users only need to sketch silhouettes, and the shape of crown is automatically generated. As shown in Figure 6, we first compute out several cross profiles defined by Bezier curve in polar coordinates from closed sketching curve. And then a guidance crown shape is created by sweeping all these cross profiles. Users can further drag silhouettes of editing control volume to tune cross profiles, so as to create new asymmetric crown shapes. Comparing with conventional parametric based crown representations [BPF*03], our approach is more intuitive and convenient to create complex crown shape.

Importing mesh. It is very useful to synthesize specific shape of trees via importing triangular or polygonal meshes. For example, people can apply this feature in film production to create special effects, such as 'monster tree'. Examples are demonstrated in Figure 1 for this feature. Users can simply import a target mesh and constrained points are sampled from this guidance shape. Therefore a couple of trees with special complex shape constraints can be automatically generated with different parameter combinations.

Generation of shape sample nodes. Once having the shape represented in triangles, we distribute uniform sample nodes on these triangles. The averaged area, A, of each sample is computed and then converted to the diameter, d, by assuming the sample region is covered by a sphere as $d = 2\sqrt{A/2\pi}$. We can also distribute uniform sample nodes in the shape volume with the same d and all of these samples form the constraint set $\mathcal{C}$ in our variational model.

Leaf population. After generating the tree model, we attach leaves or flowers to resultant tree structure according to specified tree species. In our system, we assume the leaf geometry and small branch structure are regular for a given tree type. We therefore replicate leaves and place them on

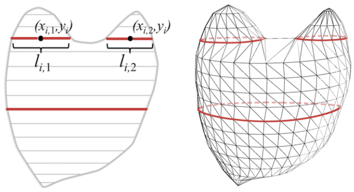


Figure 6: *Sketching guidance shape.*

branches based on botanical laws governing leaf arrangement. For example, the leaves on the same branch can be directly opposite or alternate with a deviation angle. The leaf population and geometry tessellations are controlled by users in parameters. To create realistic stems, standard cylinders in computed out branch structure $\mathcal{B}$ are replaced by sweeping cylinders which are generated from cubic Bezier curve with given diameters [Blo85] and randomly perturbing the second and the third control points [LD99].

6. Results

The prototype system based on our variational tree synthesis approach is written in C++ with Qt. All results were acquired in a standard PC with a 3.4 GHz i7 2600 CPU and rendered using Mental Ray by exporting data from our system to Maya.

Modeling abilities. To show the modeling abilities of our approach, we demonstrate variety modeling examples. In Figure 7, we demonstrate modeling different types of tree structures with the same guidance shape by choosing the botanical parameter N, P, n and θ, respectively. In Figure 1, we illustrate that the main goal of our approach is modeling optimal trees within guidance shapes. Small details of the shapes, such as the tongue and tail of the Dragon and the ears and wings of the Gargoyle, are captured by our method. A more complicated example is shown in Figure 13. These trees faithfully reflect these characters but still retain nature and with botanical rules. More results on modeling different kinds of trees are shown in Figure 14, where 9 trees, maple, Himalayan pine, white ash, camphor, Scarlet oak, Siberian elm, European spruce, birch and Acer negundo are modeled.

Apical and lateral control. In our approach, apical and lateral controls are well supported by our model. As illustrated in Figure 12, a wide diversity of forms can be obtained by different control parameters which apical control is removed in the main stem or the lateral branches. For example, in the left of Figure 12, we set $P = 1.0$ to let the apical control be limited to the main stem. In the center image of Figure 12, to remove the apical control in the main stem and lateral branches earlier, we can optimize tree structure with $P = 0.3$, $N = 3$ in level 0 and $P = 0.5$, $N = 3$ in level 1. In Figure 12 right, we set $P = 0.2$ in level 0 and $P = 1.0$ in level 1, then it can be observed that the apical control is removed earlier in the main stem, but is persistent in lateral branches.

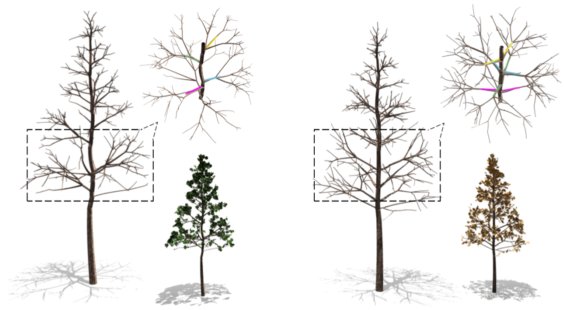


Figure 8: *Alternate (left) vs. opposite (right) laterals.*

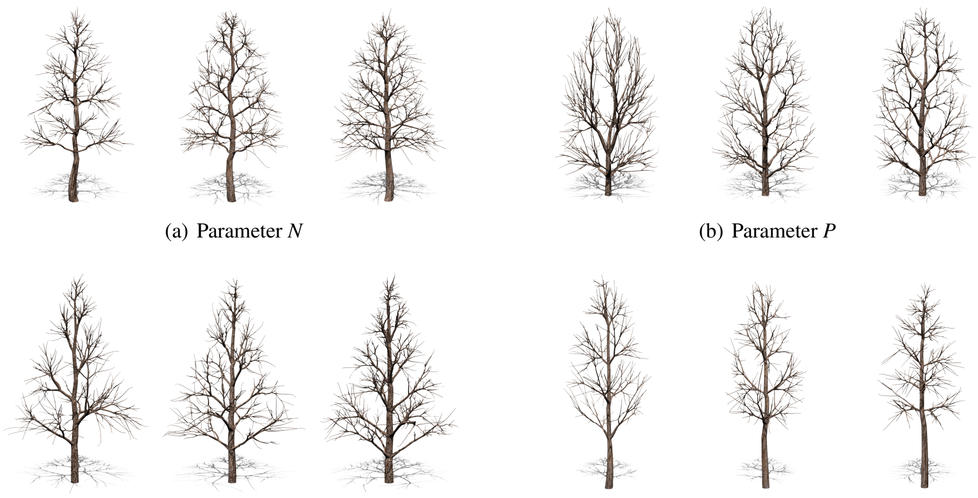
Figure	Constraints	Branches	Generation time
Fig. 12 left	3500	5741	33.5 sec
Fig. 12 center	4144	6845	37.4 sec
Fig. 12 right	1792	2565	12.6 sec

Table 2: *Sample tree generation times.*

Alternate vs. opposite lateral. Different species of trees may have alternate or opposite laterals [Bel91]. In our model, alternate or opposite laterals are determined by parameter $n = 1$ or 2 respectively (see Figure 8). Moreover, adjacent branches of a tree with opposite laterals are often forming in cross. This phenomenon is called *decussate* which is also illustrated in the right example of Figure 8. Please verify from the top view. In order to have a clear visualization, we remove some leaf level branches.

Comparison. Figure 9 is an example to demonstrate the difference between our global optimization approach and existing local adaptive methods [PJM94] and [PHL*09]. For fair comparison, all three trees are synthesized according to a same shape bending to right. And all of them share a common branching configuration, and the later two apply a simple branching rule "/[+F][-F]F" in a L-systems representation. By comparing these results, it is easy to see that the overall trend of the trunk generated by our approach is bending to right as the given guidance shape, which reflects the effects of global constraints. While the tree generated by [PJM94] has dense top part in a bush-like style due to iterative pruning and recall mechanism. Although the self-organizing [PHL*09] is good at local control and is able to create rich effects of exogenous and endogenous regulation of the tree interacting with the local environment, e.g. space and light, as depicted in the right image of Figure 9. However, it is still lack of straightforward control to globally affect the growing trend of the main stem. In such application scenarios, our method efficiently generate desired results with intuitive global controls, which is a main difference with those local adaptive methods [PJM94, PHL*09, TLLD11].

Local minima and topology refinement. In our variational model, a topology refinement step is used to jump out local minima. To show its effects, in Figure 10, we plot the



(a) Parameter N (b) Parameter P

(c) Parameter n (d) Parameter θ

Figure 7: *(a) Lateral density control with parameter N. From left to right, the values of N are 5, 7, 9. (b) Apical growth control with parameter P. From left to right, the values of P are 0.2, 0.6, 1.0. (c) Sub-branch control with parameter n. From left to right, the values of n are 1, 2, 3. (d) Angle control with coefficient* θ*. From left to right, the values of* θ *are* 25°, 45°, 65°.

Figure 9: *Tree branching structures generated by different methods under a same guidance shape. From left to right, the tree is generated by our variational approach, [PJM94] and [PHL*09].*

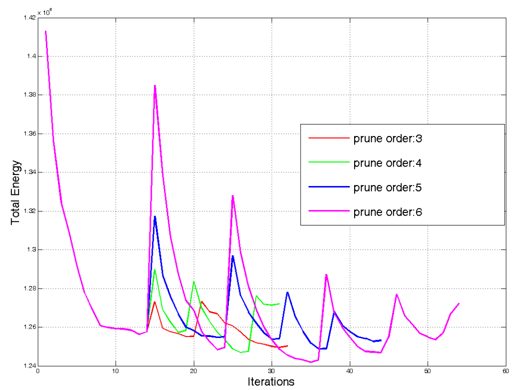


Figure 10: *Energy during optimization. Each curve corresponds to one prune order for the same tree.*

total energy (Equation(4)) with iterations at different prune orders. The overall decreases of curves in Figure 10 reflect the optimization described in Section 4. From these curves, it can be seen that there is a repeatedly monotonic decrease of the energy before reaching a local minima. Though there is an obvious increase of the energy just after we apply the topology update, after several iterations, the overall energy still decreases. However, the topology update can not always guarantee the global decrease of total energy. There might be fluctuation of the local minima after several topology update steps. In this case, we always choose the tree model with the lowest energy as the final result. In Figure 11, we show some resultant trees at different iteration steps corresponding to the curve with prune order 6. It can be observed that the topology refinement has the effect of adjusting the structure of the tree to avoid unrealistic long internodes connected to the shape nodes.

Performance. In Table 2, we show the synthesizing times of three trees which are illustrated in Figure 12. It takes dozens of seconds to generate these complex trees. According to this typical data, the performance of our approach is related to the number of branches as well as the complexity of the guidance shape. Our method can achieve high modeling quality with interactive response time.

7. Conclusions and Discussion

In this paper, we introduce a novel approach for efficiently modeling trees with guidance shapes. Our method employs a variational optimization framework and measures differ-

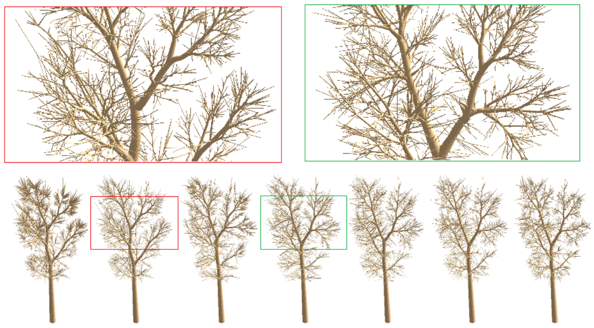


Figure 11: *Resultant trees at different iteration steps. Bottom: from left to right, tree at iteration 6 and different local minima. Top: the highlighted parts corresponding to two trees selected at the first local minima(red) and the global minima(green).*

ent factors, including the difference between the guidance shape and the target tree crown, the minimum total branches volume and spatial branches patterns. A variety of natural-looking tree models are generated from freehand sketching or from given meshes.

Limitations. Our method also bears some limitations. First, to depict branch patterns, we only consider a small set of hierarchical botanical rules with five parameters for each level. Though such a set of botanical rules is capable of generating many branching structures of trees, it is still a small subset of a large number of parameters to depict geometric shapes and branches of most realistic trees [WP95, LD99]. This limits our method to model some other plants like flowers and vines. Additionly, by lacking some richer controlling parameters, such as the vertical attraction parameters used in [WP95], our method is unable to model some specific trees like willows. Integration of more botanical rules and parameters to represent trees with more complex shapes or other plants with different morphology will be a future direction of our method. Second, we does not consider the environmental factors, such as the gravity or sunshine in our method. It will be an interesting topic to add these factors into our tree modeling framework. Third, all botanical rules are counted in the optimization by soft constraints. This may produce some unnatural results, for example, the bottom of region of the Dragon in Figure 1. But under the constraint of the shape, our method produce an optimal result even some branches have large botanical errors. Finally, in our prototype system implementation, we uniformly generate shape sample nodes on triangles, which may not be the final distribution of branches wanted by users. More tools for user to input shape or shape sample nodes will be one of future work.

Future work. Besides these limitations, in future, we will explore better algorithms and tools to distribute sample nodes according to some botanical patterns with and within the shape. Additionally, we will also try to integrate our proposed approach with other image based techniques for tree geometry reconstruction. Finally, since our approach is a parametric driven one, it might be an interesting direction to integrate learning based user interface as that did in [TGY*09].

Acknowledgements

We would like to thank the reviewers for their thoughtful comments. We thank Kaimo Lin for help with developing the prototype system, and Bolei Qian for help with art assets. This work was partially supported by NSFC (No. 61272301, No. 61232011), the 863 program of China (No.2012AA011902) and the Fundamental Research Funds for the Central Universities (No.2014FZA5017).

References

[BC07] BARTHÉLÉMY D., CARAGLIO Y.: Plant architecture: a dynamic, multilevel and comprehensive approach to plant form, structure and ontogeny. *Annals of Botany 99*, 3 (2007), 375–407. 4

[Bel91] BELL A.: *Plant form: an illustrated guide to flowering plant morphology*. Oxford University Press,Inc., 1991. 8

[BLM09] BUCKSCH A., LINDENBERGH R., MENENTI M.: Skeltre – fast skeletonisation for imperfect point cloud data of botanic trees. In *EG Workshop on 3D Object Retrieval* (2009), pp. 13–27. 3

[Blo85] BLOOMENTHAL J.: Modeling the mighty maple. In *Proceedings of the 12th annual conference on Computer graphics and interactive techniques* (1985), SIGGRAPH '85, pp. 305–311. 8

[BMK67] BROWN C. L., MCALPINE R. G., KORMANIK P. P.: Apical dominance and form in woody plants : A reappraisal. *American Journal of Botany 54*, 2 (1967), 153–162. 4

[BPF*03] BOUDON F., PRUSINKIEWICZ P., FEDERL P., GODIN C., KARWOWSKI R.: Interactive design of bonsai tree models. *Computer Graphics Forum 22*, 3 (2003), 591–600. 3, 7

[CNX*08] CHEN X., NEUBERT B., XU Y.-Q., DEUSSEN O., KANG S. B.: Sketch-based tree modeling using markov random field. *ACM Transactions on Graphics 27*, 5 (2008), 109. 2, 3

[CSAD04] COHEN-STEINER D., ALLIEZ P., DESBRUN M.: Variational shape approximation. *ACM Trans. Graph. 23*, 3 (Aug. 2004), 905–914. 3, 6

[DC96] DOUADY S., COUDER Y.: Phyllotaxis as a dynamical self organizing process part i: The spiral modes resulting from time periodic iterations. *Journal of Theoretical Biology 178*, 3 (1996), 255–27. 4

[DL05] DEUSSEN O., LINTERMANN B.: *Digital Design of Nature: Computer Generated Plants and Organics*. Springer-Verlag New York, Inc., 2005. 2

[FG00] FERRARO P., GODIN C.: A distance measure between plant architectures. *Annals of Forest Science 57*, 5/6 (2000), 445–461. 3

[HGP05] HAHN H., GEORG M., PEITGEN H.-O.: Fractal aspects of three-dimensional vascular constructive optimization. *Fractals in Biology and Medicine 4* (2005), 55–66. 2

[Hon71] HONDA H.: Description of the form of trees by the parameters of the tree-like body: Effects of the branching angle and

Figure 12: *Sample tree structures created by using multi-level combinations of botanical parameters.*

Figure 13: *Sample trees to represent GRAPHICS (zoom in for details).*

the branch length on the shape of the tree-like body. *Journal of Theoretical Biology 31*, 3 (1971), 331–338. 2

[HR92] HWANG F. K., RICHARDS D. S.: Steiner tree problems. *Networks 22*, 1 (1992), 55–89. 5

[IMT99] IGARASHI T., MATSUOKA S., TANAKA H.: Teddy: A sketching interface for 3d freeform design. In *SIGGRAPH* (1999), pp. 409–416. 7

[IOI06] IJIRI T., OWADA S., IGARASHI T.: The sketch lsystem: Global control of tree modeling using free-form strokes. *Smart Graphics* (2006), 138–146. 3

[LCWK07] LU L., CHOI Y.-K., WANG W., KIM M.-S.: Variational 3d shape segmentation for bounding volume computation. *Comput. Graph. Forum 26*, 3 (2007), 329–338. 3

[LD99] LINTERMANN B., DEUSSEN O.: Interactive modeling of plants. *IEEE Computer Graphics and Applications 19*, 19 (1999), 56–65. 8, 10

[LDS*11] LI C., DEUSSEN O., SONG Y.-Z., WILLIS P., HALL P.: Modeling and generating moving trees from video. *ACM Trans. Graph. 30*, 6 (Dec. 2011), 127:1–127:12. 3

[Leo71] LEOPOLD L. B.: Trees and streams: The efficiency of branching patterns. *Journal of Theoretical Biology 31*, 2 (1971), 339–354. 2

[Lin68] LINDENMAYER A.: Mathematical models for cellular interactions in development ii. simple and branching filaments with two-sided inputs. *Journal of Theoretical Biology 18*, 3 (1968), 300–315. 1

[Llo82] LLOYD S.: Least square quantization in pcm. *IEEE Trans. Inform. Theory 28* (1982), 129–137. 7

[LPC*11] LIVNY Y., PIRK S., CHENG Z., YAN F., DEUSSEN O., COHEN-OR D., CHEN B.: Texture-lobes for tree modelling. *ACM Transactions on Graphics 30* (2011), 53:1–53:10. 3

[LRBP12] LONGAY S., RUNIONS A., BOUDON F., PRUSINKIEWICZ P.: Treesketch: interactive procedural modeling of trees on a tablet. In *Proceedings of the International Symposium on Sketch-Based Interfaces and Modeling* (Aire-la-Ville, Switzerland, Switzerland, 2012), SBIM '12, Eurographics Association, pp. 107–120. 2, 3

[LYO*10] LIVNY Y., YAN F., OLSON M., CHEN B., ZHANG H., EL-SANA J.: Automatic reconstruction of tree skeletal structures from point clouds. *ACM Trans. Graph. 29*, 6 (Dec. 2010), 151:1–151:8. 3

[Mac83] MACDONALD N.: *Trees and networks in biological models*. J. Wiley, Chichester [West Sussex] New York, 1983. 4

[MP96] MĚCH R., PRUSINKIEWICZ P.: Visual models of plants interacting with their environment. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques* (1996), SIGGRAPH '96, pp. 397–410. 2

[MTM03] MADDALENA F., TAGLIALATELA G., MOREL J.-M.:

Figure 14: *More tree modeling results of our method. Top row, from left to right, are maple, Himalayan pine and white ash. Middle row, from left to right, are camphor, Scarlet oak and Siberian elm. Bottom row, from left to right, are European spruce, birch and Acer negundo.*

A variational model of irrigation patterns. *Interfaces and Free Boundaries* (2003), 391–416. 2

[Mur27] MURRAY C. D.: A relationship between circumference and weight in trees and its bearing on branching angles. *Journal of General Physiology 10* (1927), 725–729. 2, 5

[NFD07] NEUBERT B., FRANKEN T., DEUSSEN O.: Approximate image-based tree-modeling using particle flows. *ACM Trans. Graph. 26*, 3 (2007), 88. 3

[OOI06] OKABE M., OWADA S., IGARASHI T.: Interactive design of botanical trees using freehand sketches and example-based editing. In *ACM SIGGRAPH 2006 Courses* (New York, NY, USA, 2006), SIGGRAPH '06, ACM. 2, 3

[PHL*09] PALUBICKI W., HOREL K., LONGAY S., RUNIONS A., LANE B., MECH R., PRUSINKIEWICZ P.: Self-organizing tree models for image synthesis. *ACM Trans. Graph. 28*, 3 (2009). 1, 2, 3, 8, 9

[PJM94] PRUSINKIEWICZ P., JAMES M., MECH R.: Synthetic topiary. In *ACM SIGGRAPH* (1994), pp. 351–358. 1, 2, 8, 9

[PL90] PRUSINKIEWICZ P., LINDENMAYER A.: *The algorithmic beauty of plants*. Springer-Verlag New York, Inc., New York, NY, USA, 1990. 1, 2

[PNDN12] PIRK S., NIESE T., DEUSSEN O., NEUBERT B.: Capturing and animating the morphogenesis of polygonal tree models. *ACM Trans. Graph. 31*, 6 (Nov. 2012), 169:1–169:10. 3

[PSK*12] PIRK S., STAVA O., KRATT J., SAID M. A. M., NEUBERT B., MĚCH R., BENES B., DEUSSEN O.: Plastic trees: interactive self-adapting botanical tree models. *ACM Trans. Graph. 31*, 4 (July 2012), 50:1–50:10. 2

[PTVF92] PRESS W., TEUKOLSKY S., VETTERLING W., FLANNERY B.: *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, New York, NY, USA, 1992. 7

[RB85] REEVES W. T., BLAU R.: Approximate and probabilistic algorithms for shading and rendering structured particle systems. In *ACM SIGGRAPH 1985* (1985), vol. 19, p. 313ÍC322. 2

[RFL*05] RUNIONS A., FUHRER M., LANE B., FEDERL P., ROLLAND-LAGAN A., PRUSINKIEWICZ P.: Modeling and visualization of leaf venation patterns. *ACM Transactions on Graphics 24*, 3 (2005), 702–711. 2

[RLP07] RUNIONS A., LANE B., PRUSINKIEWICZ P.: Modeling trees with a space colonization algorithm. In *Proceedings of the Eurographics Workshop on Natural Phenomena, NPH 2007, Prague, Czech Republic* (2007), pp. 63–70. 1, 2

[RW82] ROY A. G., WOLDENBERG M. J.: A generalization of the optimal models of arterial branching. *Bulletin of Mathematical Biology 44* (1982), 349–360. 5

[SLS*10] SHEK A., LACEWELL D., SELLE A., TEECE D., THOMPSON T.: Art-directing disney's tangled procedural trees. In *ACM SIGGRAPH 2010 Talks* (2010), SIGGRAPH '10, pp. 53:1–53:1. 1

[SRDT01] SHLYAKHTER I., ROZENOER M., DORSEY J., TELLER S.: Reconstructing 3d tree models from instrumented photographs. *Computer Graphics and Applications, IEEE 21*, 3 (2001), 53–61. 3

[SYHK64] SHINOZAKI K., YODA K., HOZAMI K., KIRA T.: A quantitative analysis of plant form – the pipe model theory – i basic analysis. *Japanese Journal of Ecology 14*, 3 (1964), 97–105. 4

[TFX*08] TAN P., FANG T., XIAO J., ZHAO P., QUAN L.: Single image tree modeling. *ACM Transactions on Graphics 27*, 5 (2008), 108. 3

[TGY*09] TALTON J. O., GIBSON D., YANG L., HANRAHAN P., KOLTUN V.: Exploratory modeling with collaborative design spaces. *ACM Trans. Graph. 28*, 5 (2009). 10

[TLLD11] TALTON J. O., LOU Y., LESSER S., DUKE J.: Metropolis procedural modeling. *ACM Transactions on Graphics 30*, 2 (2011), 1–14. 2, 8

[TZW*07] TAN P., ZENG G., WANG J., KANG S. B., QUAN L.: Image-based tree modeling. *ACM Transactions on Graphics 26*, 3 (2007), 87. 3

[VEJA89] VIENNOT X., EYROLLES G., JANEY N., ARQUÉS D.: Combinatorial analysis of ramified patterns and computer imagery of trees. *SIGGRAPH Comput. Graph. 23*, 3 (July 1989), 31–40. 4

[WBCG09] WITHER J., BOUDON F., CANI M.-P., GODIN C.: Structure from silhouettes: a new paradigm for fast sketch-based design of trees. *Comput. Graph. Forum 28*, 2 (2009), 541–550. 3

[WP95] WEBER J., PENN J.: Creation and rendering of realistic trees. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques* (1995), SIGGRAPH '95, pp. 119–128. 5, 10

[WZS*06] WANG R., ZHOU K., SNYDER J., LIU X., BAO H., PENG Q., GUO B.: Variational sphere set approximation for solid objects. *Vis. Comput. 22*, 9 (Sept. 2006), 612–621. 3, 4

[WZY01] WANG Z., ZHAO M., YU Q.-X.: Modeling of branching structures of plants. *Journal of Theoretical Biology 209*, 4 (2001), 383–394. 2, 5

[XGC07] XU H., GOSSETT N., CHEN B.: Knowledge and heuristic-based modeling of laser-scanned trees. *ACM Transactions on Graphics 26*, 4 (2007), 19:1–13. 3

[Zam76] ZAMIR M.: Optimality principles in arterial branching. *J Theor Biol 62*, 1 (1976), 227–51. 2