The Shape Space of 3D Botanical Tree Models

GUAN WANG, Tongji University (China)
HAMID LAGA, Murdoch University (Australia) and The Phenomics and Bioinformatics Research Centre, University of South Australia (Australia)
NING XIE, University of Electronic Science and Technology of China
JINYUAN JIA, Tongji University (China)
HEDI TABIA, ETIS UMR 8051, Paris Seine University, University of Cergy-Pontoise, ENSEA, CNRS (France)

We propose an algorithm for generating novel 3D tree model variations from existing ones via geometric and structural blending. Our approach is to treat botanical trees as elements of a tree-shape space equipped with a proper metric that quantifies geometric and structural deformations. Geodesics, or shortest paths under the metric, between two points in the tree-shape space correspond to optimal deformations that align one tree onto another, including the possibility of expanding, adding, or removing branches and parts. Central to our approach is a mechanism for computing correspondences between trees that have different structures and a different number of branches. The ability to compute geodesics and their lengths enables us to compute continuous blending between botanical trees, which, in turn, facilitates statistical analysis, such as the computation of averages of tree structures. We show a variety of 3D tree models generated with our approach from 3D trees exhibiting complex geometric and structural differences. We also demonstrate the application of the framework in reflection symmetry analysis and symmetrization of botanical trees.

Categories and Subject Descriptors: I.3.5 [**Computer Graphics**]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations

General Terms: Tree Modeling, Plant Modeling, Geodesics, Plant Animation, Tree Statistics, Symmetry Analysis and Symmetrization

Additional Key Words and Phrases: 3D tree synthesis, geodesics, 3D tree blending, structural variation

ACM Reference format:
Guan Wang, Hamid Laga, Ning Xie, Jinyuan Jia, and Hedi Tabia. 2017. The Shape Space of 3D Botanical Tree Models. *ACM Trans. Graph.* 37, 1, Article 7 (January 2018), 18 pages.
https://doi.org/10.1145/3144456

This work was supported in part by the National Natural Science Foundation of China under Project 61602088.
Authors' addresses: G. Wang and J. Jia, School of Software Engineering, Tongji University, 4800 Cao'An Highway, Shanghai, China, 201804; emails: {guan.wang, jyjia}@tongji.edu.cn; H. Laga, School of Engineering and IT of Murdoch University, (90 South Street, Murdoch WA6150, Australia) and with the Phenomics and Bioinformatics Research Centre, University of South Australia; email: H.Laga@murdoch.edu.au; N. Xie, Center for Future Media, and the School of Computer Science and Engineering, University of Electronic Science and Technology of China, 2006 Xiyuan Ave., West Hi-Tech Zone, Chengdu, China, 611731; email: seanxiening@gmail; H. Tabia, ETIS UMR 8051, Paris Seine University, University of Cergy-Pontoise, ENSEA, CNRS, 6, avenue du Ponceau, 95014, Cergy-Pontoise, France; email: hedi.tabia@ensea.fr.

https://doi.org/10.1145/3144456

1 INTRODUCTION

Trees, bushes, crops, or plants in general are ubiquitous in urban and landscape environments. Their 3D models are used in many application areas, including agriculture, plant biology, architecture, urban modeling, games, and movies. In computer graphics, realistic 3D models of plants can add a great deal of realism to a virtual environment. Trees and plants belong to almost every animated movie or computer game (Bradley et al. 2013). As a consequence, various methods and commercial tools have been proposed to ease the modeling process while creating realistic models. Some of these methods capture models of plants directly from the real world (Quan et al. 2006; Neubert et al. 2007; Bradley et al. 2013). Others focused on interactive modeling tools, such as freehand sketching or example-based editing (Okabe et al. 2007; Chen et al. 2008), or ball B-spline curves (Wu et al. 2009). Nevertheless, the process of creating new 3D plant models requires intensive and expensive work since plants, which have complex geometry and appearance, are difficult to model even manually. This is particularly true when populating large virtual environments in which modeling each plant individually can be tedious even with the best image-based or interactive tools.

In this article, we focus on botanical trees and propose a data-driven mechanism for creating new 3D tree variations from existing ones via joint geometric and structural blending. Our idea is that, instead of individually modeling every 3D tree in a scene, users can benefit from tools that automatically synthesize a variety of new, distinct 3D trees from only a few existing models. The few initial models can be created using one of the existing 3D reconstruction and modeling tools or imported from the various repositories that are currently available on the Internet.

When dealing with anatomical models—for example, human body shapes or man-made ones—methods for automatically synthesizing new shape variations exist (Blanz and Vetter 1999; Allen et al. 2003; Kurtek et al. 2013; Kalogerakis et al. 2012a; Alhashim et al. 2014; Laga et al. 2017). For instance, one can take a pair of 3D human shapes and automatically generate an optimal in-between blending, also referred to as a *geodesic* when a proper metric is defined. The ability to compute geodesics in a well-defined shape space equipped with a proper metric is fundamental to statistical 3D shape analysis and shape-space exploration. If these ingredients are available, one can fit to a population of shapes a probability distribution, for example, multivariate Gaussian, and sample from it—either randomly or in a controlled fashion—new instances of shapes. These concepts are well studied in the case of 3D shapes that deform only in their geometry (Blanz and Vetter 1999; Allen

Fig. 1. Smooth geometry and structure blending between two botanical trees. Botanical trees are represented as points in a tree-shape space in which paths correspond to geometric and topological variations, including the possibility of expanding, adding, or removing branches and parts.

et al. 2003; Laga et al. 2017; Kurtek et al. 2013). Our key observation, however, is that, unlike such anatomical shapes, most of the rich variations in the shape of botanical trees resides more in their structure than in their geometric details.

In this article, we consider 3D models of botanical trees that possess rich structural variations. We focus on joint geometric and structural blending and on mean tree-shape computation. The technical challenges to this process are two-fold. First, it requires computing accurate correspondences between tree models that differ in the number and structure of their branches. Second, in addition to geometric blending, it also requires a plausible continuous blending of structures. Our approach is to represent 3D tree models using graphs that encode both their geometry and structure. The graph edges represent the medial abstractions of the tree's segments augmented with geometric attributes of the segments, while the nodes represent bifurcation points. Since we are dealing with botanical trees, the graph has a tree topology (i.e., with no cycles), which we refer to as a *tree graph*.

With this representation, structural variations between botanical trees can be modeled as variations in the structure of the tree graph, for example, by adding or removing nodes, altering edge connectivity, merging nodes, or splitting edges, while geometric variations are captured by changes in the edge attributes. Thus, the problem of generating smooth optimal blendings between botanical trees becomes a problem of computing geodesics between the tree graphs that represent them. To do so, we treat each tree graph as an element of a tree-shape space equipped with a proper metric, namely, the Quotient Euclidean Distance (QED), originally proposed in Feragen et al. (2013a). This formulation facilitates statistical analysis tasks such as the computation of geodesics and averages. Central to our approach is a mechanism for computing one-to-many correspondences between tree models that often have a different number of branching parts.

Contributions. This article focuses on automatic synthesis of 3D tree variations from existing ones via geodesic interpolation in tree-shape space. The key contributions can be summarized as follows:

- A novel approach for 3D botanical tree shape synthesis via geodesic interpolations in the shape space of trees. It builds on the concept of tree-graph statistics introduced in Billera et al. (2001) and Feragen et al. (2013a). We define a tree-shape space and a metric on this space, which allows not only the computation of geodesics but also the computation of summary statistics such as the average tree shape. To the best of our knowledge, this is the first approach that produces joint continuous geometric and structural variations from a few 3D models of trees.
- Central to our framework is an efficient process for computing branchwise correspondences between botanical trees that differ not only in geometry but also in their structure and in the number of their branches and bifurcation points. This enabled us not only to produce smooth and natural geodesics but also improved significantly the computation time. Compared to inverse procedural modeling techniques (Stava et al. 2014), our approach is one order of magnitude faster.
- We demonstrate in the experimental results that our formulation generates continuous and smooth 3D tree interpolations that cannot be obtained with standard statistical 3D shape analysis techniques and that are purely geometry based, and with inverse procedural modeling techniques.
- In addition to computing summary statistics such as mean tree shapes, we also demonstrate the utility of the proposed framework in reflection symmetry analysis and symmetrization of 3D botanical trees. To the best of our knowledge, this cannot be achieved with existing techniques such as inverse procedural modeling.

The ability to compute geodesics and statistical averages is a significant leap forward toward a full framework for joint statistical analysis of the geometry and structure of 3D botanical tree models.

1.1 Related Work

Producing high-quality 3D tree models is a challenging task. Previous approaches have focused either on developing capture techniques that reconstruct trees from the real world or on providing tools to alleviate the difficult modeling process. The framework proposed in this article falls within the second category of methods, which we further divide into interactive techniques and generative models. We focus our literature review on the methods that fall in this category and refer the reader to Deussen and Lintermann (2004) and Zhang and Pang (2008) for a comprehensive survey of many other 3D tree modeling algorithms.

Rule-based models, such as L-systems (Lindenmayer 1968; Prusinkiewicz 1986; Prusinkiewicz and Lindernmayer 1990) and procedural modeling (Deussen and Lintermann 2004), along with particle systems (Reeves and Blau 1985) and space colonization (Greene 1989) frameworks are traditionally used to generate new trees by simulating their branching structures. They have been since extended in various ways to enable the representation of growth processes (Prusinkiewiczy et al. 1996; Měch and Prusinkiewicz 1996) and to model the reaction of plants to their environment (e.g., competition for resources), enabling them to self-organize (Měch and Prusinkiewicz 1996; Palubicki et al. 2009; Hua and Kang 2011). These techniques are capable of creating many distinct individual trees. However, they usually require

expert-level knowledge about modeling and thus are best suited to technical users.

Honda (1971) proposed modeling trees as recursive branching structures characterized by a small number of geometric attributes, such as branching angles and length ratios of consecutive branch segments. This basic principle underlies most recursive generative tree models proposed to date. For instance, Aono and Kunii (1984), Bloomenthal (1985), and Oppenheimer (1986) adapted Honda's model to computer graphics. Other works, such as Reeves and Blau (1985), Weber and Penn (1995), Lintermann and Deussen (1999), and Prusinkiewicz et al. (2001) improved the visual realism of recursive tree models by introducing random and organized variation of parameter values as a function of position of the affected branches within the tree structure.

In contrast, Ulam (1962) considered trees as self-organizing structures in which branching patterns emerge from the competition of individual units for space. This idea, originally introduced in terms of 2D cellular automata, has been adopted by many graphics papers. For example, Open L-Systems (Měch and Prusinkiewicz 1996) allow L-Systems to react to their environment, thus to actively self-organize. Greene (1989) and Beneš and Millán (2002) simulated climbing plants that grow on support structures and which are influenced by light density in subvolumes of the 3D scene. Pirk et al. (2012b) proposed a system that allows trees to dynamically react to their environment by computing the temporal light conditions and the inverse tropism of a tree model. They consider the environmental conditions for a given tree. Pirk et al. (2012a) further enable continuous animations of the developmental stages.

The main emphasis of these papers and many others, such as Bornhofen and Lattaud (2009), was two-fold: the ability to create individual 3D trees that look realistic and the ability of these models to adapt to their surrounding space so that they can be seamlessly integrated into complex virtual environments. The complexity and visual appearance of the models that can be created have been significantly enhanced over the years. However, creating such models is still a tedious task. To overcome this problem and reduce the modeling effort, a number of sketch-based (Power et al. 1999; Prusinkiewicz et al. 2001; Chen et al. 2008; Okabe et al. 2007), image-based (Shlyakhter et al. 2001; Quan et al. 2006; Neubert et al. 2007; Tan et al. 2008; Reche-Martinez et al. 2004; Livny et al. 2011; Bradley et al. 2013), video-based (Li et al. 2011), and range-scan-based (Livny et al. 2010) reconstructions methods have been proposed.

Statistical 3D shape analysis and exploration. Our approach is data driven. The ultimate goal is to be able to automatically synthesize new instances of 3D trees from a few examples created using the wide range of techniques that are currently available. The idea is to model, in an analysis stage, the statistics of the available dataset and then enrich it by generating new variations. From a different perspective, the problem of synthesizing new instances of 3D trees from examples can be seen as a statistical shape analysis problem. The attempts to model the statistics of botanical growth dates back to de Reffye et al. (1988), who developed a model that integrates botanical knowledge about the architecture of the trees (e.g., how they grow; how they occupy space; where and how leaves, flowers, or fruits are located) and produces a variety of new 3D tree models. Most relevant to our work are recent approaches for modeling 3D shape variability. Methods such as 3D morphable models, originally proposed for the analysis of 3D faces (Blanz and Vetter 1999) and later extended to the analysis of human body shapes (Allen et al. 2003), enable shape creation from existing ones via shape-space exploration. There has been also a growing interest in analyzing the variability in 3D shape collections using tools from differential geometry and the concepts of Riemannian shape spaces (Kurtek et al. 2013; Kilian et al. 2007). While these tools are based on strong mathematical foundations, they cannot capture structural variations and thus are not suitable for the analysis of tree shapes.

Other approaches enable shape creation via part replacement or recombination (Jain et al. 2012; Kalogerakis et al. 2012b; Zheng et al. 2013) or via structural blending (Alhashim et al. 2014). 3D botanical trees, however, are composed of branches that are geometrically and topologically very similar. Thus, the range of variations that can be generated by swapping branches between trees is very limited. This article builds on recent theoretical developments in the field of graph and tree statistics, pioneered by Billera et al. 2001, who proposed the concept of continuous tree-space and its associated geodesic distance metric as a natural way to embed and compare phylogenetic trees. This concept has been used by various authors for the statistical analysis of tree-structured data, for example, Owen and Provan (2011) and Alfaro et al. (2014). However, these works focused only on structure, ignoring the geometry and shape of the edges. Feragen et al. (2013a, 2013b) extended this concept and proposed a tree-shape space and a set of metrics for computing geodesics and statistics of airway trees. However, due to their computational complexity, these methods have been applied only to relatively small trees that exhibit only minor structural differences. This article builds on these theoretical developments and proposes a framework for computing smooth blendings between pairs of complex botanical trees and for computing the mean tree from a collection of trees that can significantly differ in their branching structure as well as in the number of their branches. The recent article of Wang et al. (2016) also proposed a tree synthesis method via hierarchical topology-preserving blending between given trees. Due to the stochastic nature of the method, the generated blendings are not smooth. Furthermore, the method does not define a proper tree-shape space and a metric. Thus, unlike our approach, tasks such as tree statistics (mean tree) and tree symmetrization cannot be performed with the approach of Wang et al. (2016).

In contrast to our approach, rule-based and procedural-based methods allow users to control the modeling process using a set of biologically motivated parameters and rules. The space of such parameters can be seen as a tree-shape space since varying these parameters will result in different botanical tree models and species. Finding the set of parameters that would generate a given tree model is a complex task, however. This problem, known as *inverse procedural modeling*, has been investigated in the context of procedural modeling of urban facades (Livny et al. 2010; Vanegas et al. 2012) as well as botanical trees (Livny et al. 2010; Cournede et al. 2011; Stava et al. 2014). In particular, Stava et al. (2014) proposed a compact parametric procedural model that describes a wide

variety of trees and a Monte Carlo Markov Chain (MCMC)–based optimization method for estimating these parameters for a given tree. The estimated parameters can then be used to generate a wide variety of tree models. In this approach, however, the mapping from the parameter space to the space of 3D tree models is not one-to-one, that is, one set of parameters (and thus a point in the parameter space) corresponds to many trees. As a consequence, the trees that can be generated from a set of parameters are random due to the stochastic nature of the sampling procedure. Moreover, smooth interpolations in the parameter space do not correspond to smooth deformations between trees.

1.2 Overview

The core of the framework proposed in this article is an algorithm that takes two mesh models of botanical trees, S and T, hereinafter referred to as the source and target, and produces an optimal sequence of deformations, which, when applied to the source, generates a sequence of blended 3D tree models. We refer to this sequence as a geodesic since it corresponds to the shortest path with respect to a well-defined metric, which connects the source and target in the space of tree-like shapes. The midpoint of the geodesic corresponds to the mean tree of the source and target. The algorithm also computes the mean tree of a collection of tree models that can be used to build generative models of botanical trees. Figure 2 summarizes the process. Below, we briefly overview each step.

First, in a preprocessing step (Section 2), we abstract each input 3D tree model into a curve representation via skeleton extraction. The representation is then used to segment trees into branches and to detect bifurcation points. To compute correspondence between the source and target tree, we propose a three-step algorithm (Section 3) consisting of an approximate elastic matching of two corresponding curves, a refinement step that aligns accurately the bifurcation points, and a recursive iteration over the subtrees. The approach is fully automatic and is able to handle trees of different structures.

Next, we jointly encode the geometry and structure of a botanical tree as a tree graph $G = (V, E, A)$. The list of nodes V and edges E encode the structure of the tree. The set A encodes the geometric attributes of each edge $e \in E$, which corresponds to a branch or a segment that connects two bifurcation points in the tree. In order to measure the distance (or dissimilarity) and compute geodesics between trees of arbitrary structure, we parameterize the tree graph G with a maximal binary tree whose edges that have zero attributes represent edge collapse and thus encode topological changes (Section 4.1).

Given this parameterization, botanical trees become elements of a tree-shape space that is composed of a concatenation of multiple subspaces. Trees within the same subspace have the same structure. Thus, paths within the same subspace correspond to variation in the geometry of trees that have the same structure. Transitions from one subspace to another correspond to variations in structure (Section 4). Endowing the space with a proper metric (Section 4.2) enables the computation of geodesics and summary statistics such as the average tree of a collection of trees.

Our results (Section 5) demonstrate that the proposed framework is able to generate smooth joint blending of geometry and structure and that it is able to compute average tree shapes from a collection of botanical trees that exhibit large geometric and structural variations. We also show that the proposed approach can be used for symmetry analysis and symmetrization of botanical trees.

2 PREPROCESSING

The input shapes are 3D polygonal models of botanical trees. For each shape, we apply the skeletonization method of Zhang et al. (2015). Since botanical trees are composed of nearly tubular branches, the approach extracts curve skeletons, which we automatically segment to extract the curves that correspond to branches in the trees. We organize the branches—and thus their corresponding curve skeletons—into levels, with the tree trunk assigned to level zero. This will be used as a constraint to facilitate the process of finding correspondences between pairs of trees. If a tree has more than one trunk, then the longest is selected as the main trunk and the others are treated as lateral branches.

Next, we sample $n = 50$ equidistant points on each curve skeleton and fit a cubic B-spline to obtain a 1D parameterized curve f embedded in $\mathbb{R}^3$, that is, $f : [0, 1] \rightarrow \mathbb{R}^3$, that best fits the sampled points. We also augment the curve skeletons with information that describes the geometry of the branches that they represent. There are several descriptors that can be used for this purpose (Biasotti et al. 2015). In our implementation, we simply use the local thickness r_i of the branches, computed at the sampled points, since botanical trees tend to have branches that are locally cylindrical.

With this setup, a botanical tree S can be represented recursively by writing $S = \{f_S, S_1, \ldots, S_{m_f}, s_1, \ldots s_{m_f}\}$ (Figure 3(a)), where:

- $f_S : [0, 1] \rightarrow \mathbb{R}^3$ is the curve skeleton corresponding to the main trunk.
- $S_i, i = 1 \ldots m_f$, are the subtrees that are connected to the main trunk, and
- $s_i \in [0, 1], i = 1 \ldots m_f$, are the locations of the bifurcation points, that is, points where subtrees are connected to their parent branch.

Each subtree that is connected to the main trunk at a bifurcation point is recursively defined in the same manner.

3 CORRESPONDENCE COMPUTATION

Computing geodesics requires one-to-one correspondence between the source and target trees. Solving, however, the general correspondence problem is very hard, especially for botanical trees that can be structurally different since they often have a different number of branches and bifurcation points. In this article, we assume that the source and target trees, hereinafter denoted by S and T, respectively, are upright oriented, their main trunks have been already identified, and that their branches are organized into levels, with level zero corresponding to their main trunks. If a tree has more than one trunk, then the longest one is selected as the main one and the others are treated as lateral branches. We also assume that levelwise correspondences are given, that is, branches located at level l on the source tree can have their counterparts only at level l on the target tree. This last assumption is biologically plausible, although it might be violated in some cases, since bifurcation ratios

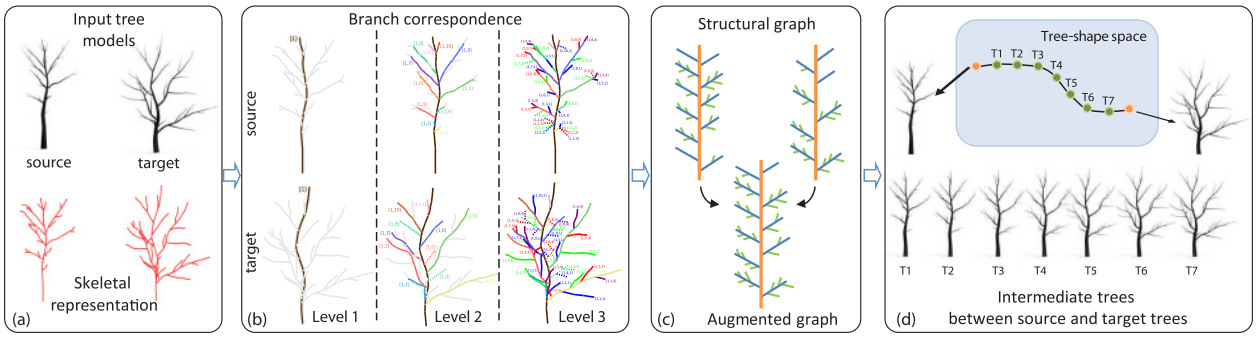

Fig. 2. Overview of the proposed framework. Given a source and target botanical trees, we first (a) compute their skeletal representations and then (b) find branch correspondences. Here, branchwise correspondences are color coded. Each botanical tree is then represented with a tree graph that encodes jointly its geometry and structure. (c) The trees are augmented, binarized, and parameterized with a maximal binary tree. For simplicity, we only show the nodes and edges that are in the first three layers of the trees. (d) Finally, trees are represented as points in a tree-shape space equipped with a proper metric. Geodesics in the tree-shape space correspond to smooth geometric and structural blending between the source and target trees.

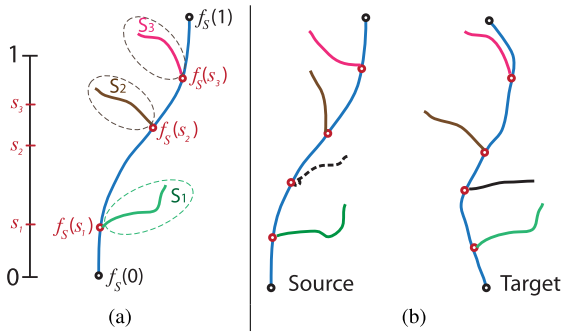

Fig. 3. (a) Recursive representation of a simple botanical tree. (b) Example of branchwise correspondences. Artificially added branches (of zero attributes) are illustrated with dashed lines.

are often highly correlated with growth stages of plants (Borchert and Slade 1981).

With these assumptions, the search for correspondences is restricted within the same level on the trees. Thus, one can proceed hierarchically to establish full branch-to-branch correspondence between the source S and target T. In what follows, we write $S = \{f_S, S_1, \ldots, S_m, s_1, \ldots s_m\}$ and $T = \{f_T, T_1, \ldots, T_k, t_1, \ldots, t_k\}$; see Section 2 for the notation. To find correspondences between S and T, we first perform an approximate elastic registration of the curves f_S and f_T and then refine the registration by finding correspondences between the bifurcation points. The purpose of the first step is to reduce the search space for bifurcation point correspondences into a small neighborhood around each bifurcation point. Finally, we repeat this process for every pair of corresponding subtrees in S and T. Below, we detail each step of this process.

Elastic curve matching. The goal of this step is to find an optimal rotation $O \in SO(3)$ and re-parameterization $\gamma : [0,1] \to [0,1]$ such that the distance between f_S and $O(f_T \circ \gamma)$ is minimized. To solve this problem, we represent skeletal curves $f : [0,1] \to \mathbb{R}^3$ using the Square Root Velocity Function (SRVF) (Srivastava et al. 2011; Laga et al. 2014) denoted by $q = SRVF(f)$ and given by $q(s) = \frac{\dot{f}(s)}{\sqrt{\|\dot{f}(s)\|}}$, $s \in [0,1]$, and $\dot{f} = \frac{\partial f}{\partial s}$. The use of derivatives with respect to the parameter $s \in [0,1]$ ensures that the representation is invariant to translation. Prior to the analysis, we rescale each curve to have unit length. Under this representation, the elastic registration of two skeletal curves, f_S and f_T, becomes a problem of solving the following optimization:

$$(O^*, \gamma^*) = \arg \min_{O \in SO(3), \gamma \in \Gamma} d_g(q_1, O(q_2, \gamma)), \tag{1}$$

where $q_1 = SRVF(f_S)$, $q_2 = SRVF(f_T)$, $(q, \gamma) = \sqrt{\dot{\gamma}(t)}(q \circ \gamma)$, and $d_g(q_1, q_2)$ is defined as the Euclidean distance between q_1 and q_2, which is equivalent to measuring the amount of bending and stretching needed to align f_S onto f_T. See Srivastava et al. (2011) for a full proof and discussion about the SRVF representation and its properties. Compared to a straightforward approach for matching curves, the proposed approach is particularly suitable for finding correspondences between skeletal curves that bend and stretch.

Bifurcation point matching and tree augmentation. While the elastic matching procedure described above provides relatively acceptable registration, bifurcation points may still remain inaccurately registered onto each other. Moreover, the correspondence between bifurcation points is often not one to one since their number on the source and target trees may differ. The goal of this step is to refine the elastic registration above in order to find a subset of bifurcation points in f_S that have one and only one counterpart in f_T. Let $s_i \in [0,1]$ be the location of a bifurcation point on f_S and $\gamma^*(s_i) \in [0,1]$ its corresponding location on f_T. Note that $\gamma^*(s_i)$ does not necessarily fall exactly on one of the bifurcation points in f_T. The goal is then to find, if it exists, a bifurcation point $t_{i'}$ on f_T that is as close as possible to $\gamma^*(s_i)$ while fulfilling the following two constraints:

- Uniqueness constraint: Each bifurcation point on f_S will have at most one counterpart in f_T and vice versa.
- Ordering constraint: If $s_i > s_j$, then $t_{i'} > t_{j'}$ where $t_{k'}$ is the counterpart of s_k on T.

We formulate this problem as a constrained energy minimization, which we solve using dynamic programming. The energy being minimized is defined as

$$E = \sum_{i=1}^{n} d(\gamma^*(s_i), t_{i'})^2, \tag{2}$$

where

- $d(\gamma^*(s_i), t_{i'})$ is the distance along the curve f_T between $\gamma^*(s_i)$ and $t_{i'}$ if the counterpart of s_i exists.
- $d(\gamma^*(s_i), t_{i'}) = \epsilon$ in case s_i does not have a counterpart in f_T. Here, ϵ can be seen as the cost of not assigning s_i to any bifurcation point on T. In our implementation, we set $\epsilon = 0.1$.

The output of this procedure is a subset of bifurcation points in f_S that have exactly one counterpart on f_T. Next, we augment the trees S and T by adding artificial bifurcation points, which correspond to branches of zero attributes along the skeletal curves f_S and f_T such that the correspondence becomes one to one; see Figure 3(b). To do so, for each bifurcation point s_i in f_S that does not have a counterpart in f_T, we create an artificial bifurcation point $t_{i'} = \gamma^*(s_i)$ on f_T and attach to it a virtual subtree that has the same structure as S_i but of zero length branches. Similarly, for each bifurcation point t_i in f_T that does not have a counterpart in f_S, we create an artificial bifurcation point $s_{i'}$ on f_S such that $t_i = \gamma^*(s_{i'})$ and attach to it a virtual subtree that has the same structure as T_i but of zero length branches.

Having established one-to-one correspondence between the bifurcation points (including the artificially added ones), we have also established one-to-one correspondence between the subtrees that are attached to them. Finally, in order to find full correspondence between entire trees, we apply the procedure described above recursively, starting with the main trunks and iterating over the subtrees attached to them. Figure 2(b) shows the result of this procedure on an example of two trees in which correspondences are color coded.

With a slight abuse of notation, we also refer to S and T as the augmented trees, which contain virtual branches of length zero.

Multiple correspondences. The pairwise correspondence computation approach described above can be extended to finding correspondence between multiple trees, which is required when computing statistics such as the average tree of a collection of botanical trees $\{S^i, i = 1, \ldots, n\}$, where $S^i = \{f_S^i, S_1^i, \ldots, S_{m_i}^i, s_1^i, \ldots, s_{m_i}^i\}$. Similar to pairwise correspondences, we proceed hierarchically to establish branchwise correspondences between multiple trees. Hereinafter, to simplify notation, we use f_i to denote f_S^i. First, we compute the Karcher mean μ of the curves $f_i, i = 1, \ldots, n$. It is given as

$$(q_\mu, O_i^*, \gamma_i^*) = \arg\min_q \sum_{i=1}^{n} d_g(q, O_i(q_i, \gamma_i))^2. \tag{3}$$

Here, q_i and q_μ are, respectively, the SRVF representation of the curve f_i and of the mean curve μ. We solve this optimization problem using a gradient descent approach as described in Srivastava et al. (2011) and Laga et al. (2014). Importantly, the process of computing the Karcher mean also finds the approximate elastic registration of the input curves to the computed mean. The transformation $(O_{ij}^*, \gamma_{ij}^*)$ that aligns a curve f_i to another curve f_j can then be inferred in a straightforward manner from (O_i^*, γ_i^*) and (O_j^*, γ_j^*).

Next, to match the bifurcation points, we first select the curve that has the maximum number of bifurcation points and use it as a reference. Let us denote it by f. Then,

- For each curve f_i, other than f, match its bifurcation points with the bifurcation points of f and add, if necessary, artificial bifurcation points to *only* f as described above for pairwise matching.
- Once all curves have been treated and an augmented reference curve f is obtained, then match again the bifurcation points of f to all the other curves f_i, this time adding artificial bifurcation points to f_i.

Similar to pairwise registration, this process is applied recursively, starting with the main trunks and recursively iterating over the subtrees attached to them.

Joint geometry and structure representation. Finally, both the source and target botanical trees will be represented each with a tree graph $G = (V, E, A)$ that encodes both their geometry and structure. The nodes V correspond to bifurcation and endpoints of the tree's skeleton curves. An edge $e \in E$ corresponds to a segment of a branch on the augmented tree. Thus, the pair (V, E) encodes the structure of the tree T. Each edge $e \in E$ is also annotated with additional geometric information that describes the geometry of its corresponding branch segment. In our implementation, we represent the shape of a segment using a set of pairs $\{(p_i, r_i), i = 1 \ldots, 5\}$, where p_i are the skeletal points interpolated with a cubic B-spline and r_i is the branch thickness at p_i. Thus, each edge e is a point in the Euclidean space $X_e = \mathbb{R}^{20}$.

4 GEODESICS AND MEAN TREE COMPUTATION

Once correspondences have been established between the source and target trees, the next step is to compute smooth blending paths between the two trees. In this article, we focus on optimal blending, termed *geodesics*, which is obtained by optimally (and smoothly) deforming the geometry and structure of one tree onto the other. Our approach builds on the tree statistics approach pioneered by Billera et al. (2001) and recently extended to the analysis of airway trees in human lungs (Feragen et al. 2013a). Our contribution is the adaptation of this theoretical framework to botanical tree modeling, which is an important problem in computer graphics.

The first step is to unify the representations of the trees T and S, which are often of different sizes and structures, in order to compare them (Section 4.1). We refer to this step as joint geometry and structure parameterization. Next, we treat the parameterized trees as points in a tree-shape space equipped with a proper metric. Paths in the tree-shape space correspond to geometric and structural deformations, while shortest paths, or geodesics under the metric, correspond to optimal deformations that align one tree onto the other (Section 4.2).

4.1 Unified Binary Tree Representation

We first convert T and S into binary trees by introducing constant edges of length zero and of zero geometric attributes. We

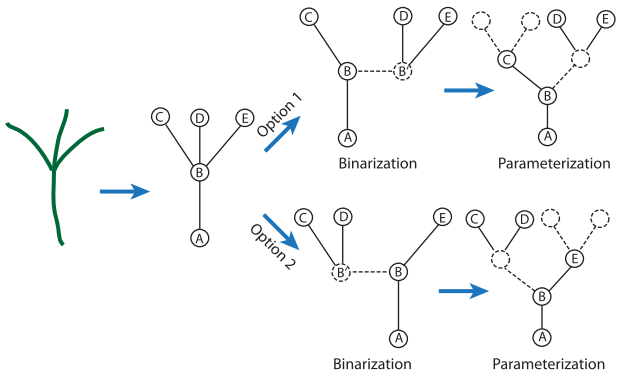

Fig. 4. Converting a 3-order tree into a binary tree by introducing virtual nodes and edges of length zero (dashed line), referred to as structure parameterization. Note that the same tree can have multiple possible parameterizations. This, however, does not affect the subsequent steps since the process of computing geodesics is invariant to the way the structures of the input trees are parameterized using binary trees.

then unify the representations by encoding every tree shape using the same binary tree $\mathcal{T} = (V, E)$, called *the maximal binary tree*. $\mathcal{T}$ can be seen as a means of parameterizing geometric trees. The tree-shape space is then defined as $X = \prod_{e \in E} (\mathbb{R}^d)^n$, where $n = 5$ is the number of points (hereinafter referred to as landmarks) sampled on the skeletal curve of each edge and $d = 4$ is the dimension of the geometric attributes used to characterize each landmark. In our implementation, we encode each landmark with its 3D coordinates augmented with the thickness of the branch at the landmark.

Figure 4 illustrates the parameterization process. We first convert all trees into binary trees by splitting nodes that have more than two children and connecting them with a virtual edge of zero attributes. In the example of Figure 4 (option 1), the node B is split into two nodes, B and B', connected with a virtual edge (the dashed line) of length zero. If after split the new nodes B or B' still have more than two children, then the same process is recursively repeated until the graph becomes binary.

Next, we collect all binary trees, $C = \{T_1, \ldots, T_M\}$, that represent the tree shapes in our collection, select the binary tree with maximum depth d_{max}, then define the maximal binary $\mathcal{T}$ as the full binary tree (i.e., every node has exactly one parent and two children, except the root node that has no parent and the leaf nodes, which do not have children). Finally, we parameterize every tree $T \in C$ using the maximal tree $\mathcal{T}$ by setting the geometric attributes of the edges of $\mathcal{T}$ that do not exist in T to zero. Thus, T can be reconstructed from $\mathcal{T}$ by collapsing edges of zero attributes.

Note that when computing the average (mean) tree of a collection C of M trees, the number of nodes in $\mathcal{T}$ is $2d_{max} - 1$, where d_{max} is the depth of the largest tree in C. When computing the geodesic between a pair of trees, however, one needs to consider only the largest tree among the two.

Finally, as shown in Figure 4, the parameterization is, in general, not unique. In fact, a same tree shape can have several representations in the tree-shape space X, that is, there will be several points in X that correspond to the same tree. To avoid the search over all possible parameterizations when computing the geodesic between a pair of trees, we use the correspondences computed in Section 3 to guide the parameterization process. When dealing with two tree shapes S and T that are in correspondence, we first compute one parameterization of S. This parameterization will serve as a reference. Then, all edges of the target tree T are embedded in the same location as their counterparts in S. When dealing with more than two trees (for computing the mean tree), we select the largest one as a reference to guide the parameterization of the others. Note, however, that the subsequent steps are not affected by the way the reference tree S is parameterized since the process of computing geodesics (Section 4.2) is parameterization invariant.

4.2 Geodesics and Means on the Tree-Shape Space

With the unified parameterization defined in Section 4.1, every tree T can be seen as a point $x = (x_e)_{e \in E}$ in a tree-shape space $X = \prod_{e \in E} X_e$, where $X_e = (\mathbb{R}^d)^n$. The vector x_e describes the shape of the edge e, $n = 5$ is the number of points per edge, and $d = 4$ is the dimension of the geometric attributes used to characterize each point on the edge e. Billera et al. (2001), Feragen et al. (2013a), and Owen et al. (2011) provide a good analysis of the mathematical properties of this space. Below, we describe how we compute geodesics in this tree-shape space.

The first important property is that in X each tree topology is associated with a unique k−dimensional subspace, called orthant. Trees belonging to the same orthant of dimension k have the same topology, with k nonzero edges, but varying edge shapes. If $(e_1, \ldots, e_k)$ are the nonzero edges, then the orthant is defined as the subspace $X_k = \prod_{i=1}^{k} X_{e_i}$. Since X_k is Euclidean, then one can use the Euclidean distance to compare two points x and y in X_k that represent two trees of same topology. If the edges of the trees represented by x and y are $(e_1^x, \ldots, e_k^x)$ and $(e_1^y, \ldots, e_k^y)$, respectively, then

$$d^2(x, y) = \sum_{i=1}^{k} \|x_{e_i} - y_{e_{\phi(i)}}\|^2, \tag{4}$$

where x_{e_i} and $y_{e_i} \in X_e$ are the vector representations of the edges e_i^x and e_i^y, respectively. $y_{e_\phi(i)}$ is the edge on y that corresponds to the ith edge on x. Using this metric, the geodesic between x and y is the straight line that connects them. Since both trees are within the same orthant, the geodesic corresponds to deformations in the shape of the edges while keeping the structure constant.

The second important property is that edge collapses correspond to projections to lower-dimensional spaces. Let X_k be an orthant of dimension k spanned by the edges $(e_1, \ldots, e_k)$. Collapsing an edge—say, e_k for example—corresponds to setting all of its geometric attributes to zero. This is equivalent to projecting a point $x \in X_k$ onto X_{k-1}, which is spanned by the edges $(e_1, \ldots, e_{k-1})$. Thus, topological changes can be modeled as paths across subspaces; moving from one subspace to another of lower dimension corresponds to edge collapse, while moving to another of a higher dimension corresponds to a node split (or insertion of a new edge). Thus, the tree-shape space is composed of multiple orthants, one for each possible tree topology, glued together to form a connected space. The geodesic between two trees x and y of different structures, and thus belonging to two different orthants X_k and $Y_{k'}$, passes through one or more intermediate orthants (see Figure 5).

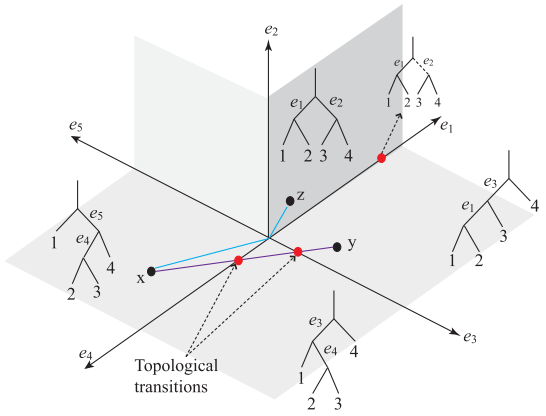

Fig. 5. Illustration of the tree-shape space, which is composed of concatenated subspaces (orthants). Each orthant corresponds to one tree topology. Dashed lines in the tree graphs indicate collapsed edges.

It is composed of a finite number of straight line segments, each one belonging to one orthant. Its length is the sum of the lengths of these segments measured using the Euclidean distance of Equation (4). It is the solution of the following optimization problem:

$$\arg\min \left\{ d_g^2(x,y) = \sum_{t=1}^{N-1} d^2(x_t, x_{t+1}) \right\}, \tag{5}$$

where $x_1 = x$, $x_N = y$, and x_t and x_{t+1} sit on the boundaries of the same orthant. $d^2(\cdot,\cdot)$ is given by Equation (4). The geodesic is the sequence $\{x = x_1, x_2, \ldots, x_N = y\}$ that minimizes Equation (5). Under this metric, which is called the Quotient Euclidean Distance (QED), geodesics are locally unique (Feragen et al. 2013a). Replacing the $\mathbb{L}_2$ norm in Equation (4) by the $\mathbb{L}_1$ norm, the metric becomes the standard Tree Edit Distance (Feragen et al. 2013a). Figure 5 illustrates the structure of the tree-shape space.

Since the optimization of Equation (5) over all possible reparameterizations (and thus correspondences) and paths in the tree-shape space is NP-hard, Feragen et al. (2013a) considered only relatively small trees and assumed that the number of structural transitions along a geodesic is at most one. Such assumptions, however, are very restrictive when dealing with botanical trees, which are often composed of a large number of nodes and branches and exhibit large structural variations. In this article, we observe that under fixed parameterization of the source and target trees (i.e., if correspondences are precomputed), the geodesic can be efficiently computed in a polynomial time. Thus, we propose a three-step algorithm, which can be summarized as follows;

—**Step 1 - Correspondence and parameterization.** This is done using the procedure described in Sections 3 and 4.1.

—**Step 2 - Initial path computation.** The goal of this step is to find the sequence of subspaces (or orthants) through which the geodesic transits. First, observe that the transition from one orthant to another is equivalent to a sequence of two operations: projection to a lower-dimensional subspace (equivalent to the collapse of an edge), followed by a projection to a different subspace of higher dimension (equivalent to the insertion of a new edge). Thus, we start with a path that has a single point, which is the source tree x_1. This path, which has zero-length with respect to the metric, is inserted into a queue of paths ordered with respect to their lengths. We proceed iteratively. At each iteration, we take from the queue the shortest path and expand it either by projection onto a lower-dimensional subspace by edge collapse or to a higher one with an edge expansion. The new path is inserted to the queue. The procedure stops once we reach the orthant that contains the target tree $x_N = y$. Algorithm 1 summarizes this procedure. In this algorithm, if x_e are edges of a tree x, $\phi(x_e)$ denotes their corresponding edges on $x_N = y$.

When projecting to a subspace of higher dimension (Step 5 of Algorithm 1), the initial length of the newly inserted edge $e \in x_e$ is set to 10% of the length of the shortest nonzero edge in the initial binary tree. It is set to be a straight line in the direction of one of the edges connected to one of its nodes.

—**Step 3 - Path straightening.** The initial path provides the sequence of subspaces (or orthants) through which the

ALGORITHM 1: Initial path computation.

Input: Initial tree $x_1 = x$ and target tree $x_N = y$.
Output: Initial geodesic path $\{x_1, x_2, \ldots, x_N\}$ such that $x = x_1$, $y = x_N$, and x_i, $i = 2, \ldots, N-1$, is on the boundary of an orthant.
Let Q be a queue that will contain the list of paths to expand. They are sorted according to their length. Each path in Q starts at x_1.

1: $p = \{x_1\}$. $Q \leftarrow Q \cup p$;
Repeat
2: $p \leftarrow$ first path in Q. Remove p from the queue;
3: Let x be the last point (i.e., extremity) of p.
4: **If** the last operation on p was edge expansion **then**
— Let x_e be the edges of x that are collapse candidates, i.e., the length of the edges $\phi(x_e)$ is zero.
— **For** every edge $e \in x_e$
— Project x to a subspace of lower dimension by setting the length of e to zero. Let x_n be the new tree.
— Remove x from the path p.
— $p_n = \{p, x_n\}$.
— Insert the new path back to the ordered queue: $Q = Q \cup \{p_n\}$.
— Mark the last operation on the newly inserted path as edge collapse.
— **End**
5: **else** (here, the last operation on p was edge collapse)
— Let y_e be the edges of $x_N = y$ that are collapse candidates, i.e., $x_e = \phi^{-1}(y_e)$ are candidates for expansion.
— **If** y_e is empty then terminate and return the path p.
— **For** every edge $e \in x_e$
— Project x to a subspace of higher dimension by setting the length of e to a positive constant. Let x_n be the new tree.
— $p_n = \{p, x_n\}$.
— Insert the new path back to the ordered queue: $Q = Q \cup \{p_n\}$.
— Mark the last operation on the newly inserted path as edge expansion.
— **End**
6: Return p and the sequence of subspaces through which it passes.

Fig. 6. Examples of geodesics (optimal blending paths) automatically generated with our approach. In each row, the first and last trees are the source and target, respectively, which although belong to the same species, they exhibit rich geometric and structural variations.

geodesic transits. The last step is to straighten this path, that is, minimize its length with respect to the metric while keeping the path within the same sequence of subspaces. This is efficiently solved using the path-straightening algorithm of Owen et al. (2011).

The experimental results section discusses the practical benefits of this approach compared to Feragen et al. (2013a).

Mean tree-shape computation. The point on the geodesic between x and y that is exactly at equidistance from x and y is the average of the two trees. When given a set of N botanical trees represented as points $\{x_i, i = 1, \ldots, N\}$ in X, their average tree is defined as the point that is at the minimum overall distance to all the points in the set. It is given by:

$$\mu = \arg\min_{x \in X} \sum_{i=1}^{N} d_g^2(x, x_i). \tag{6}$$

This is known as the Frechet mean and, since correspondences have been computed using the approach described in Section 3, it can be efficiently computed using Sturm's algorithm (Sturm 2003).

Post processing. Finally, the computed in-between 3D tree models may have small branches that can be visually implausible. To improve the visual quality of the synthesized trees, we design a filter that removes small branches that are not natural. To do so, we first assume that the length L of a branch is a function of its location $s \in [0, 1]$ on its parent branch: $L = \beta(s)$. This is biologically motivated since often branches located near the end tips of their parent branch tend to be short. Next, for each level l of the trees' hierarchy ($l = 0$ corresponds to the trunk), we learn from the data a function β_s^l for the source tree and another function β_t^l for the target tree. In our implementation, the β functions are approximated using second-order polynomials. See Weber and Penn (1995) for a discussion on the biological motivation of this choice.

For a newly synthesized tree and for each branch of the lth level, $l > 0$, located at s on its parent branch, if its length is less than $\max(0, \min(\beta_1^l(s), \beta_2^l(s)) - \alpha|\beta_1^l(s) - \beta_2^l(s)|)$ then the branch is considered to be too small and is deleted. In all our experiments, we set $\alpha = 0.5$.

5 RESULTS

In this section, we show results of geodesics and mean tree shapes computed between botanical trees that exhibit different degrees of geometric and structural variations. We consider trees belonging to the same species and those belonging to different species. We also report the timing and compare our results to the state-of-the-art. Additional results as well as a video demonstrating the blending paths and mean trees generated with the proposed approach are included in the supplementary material.

Input models. We collected, from online resources, 120 polygonal tree models with rich structural variations. The trees belong to various species. In total, we have 30 species with 3 to 5 tree models per species. All the tree models came with a correct upright orientation. They were then automatically skeletonized and converted into a tree-graph representation as described in the previous sections.

Geodesics computation. We consider pairs of 3D tree shapes, the source and target, and generate new 3D tree models by computing the geodesic (or optimal blending) that connects the source and target trees. Figure 6 shows two geodesics computed with the proposed approach. In this example, although the source and target trees belong to the same species, they exhibit various forms of geometric and structural variations including branch elongation, bifurcation (split), and merge. Figures 1 and 7, on the other hand, show geodesics between a source and a target tree that belong to different species, while Figure 8 shows a geodesic between a small tree and a big one. Note that despite the substantial geometric and structural variations, our approach is able to find correct correspondences between the trees and to generate smooth in-between tree shapes. The newly generated 3D trees can be further

Fig. 7. Examples of geodesics automatically generated with our approach. In each row, the first and last trees are the source and target, respectively, which belong to two different species. Observe the smooth structural changes in the generated in-between trees.

Fig. 8. Example of a geodesic between a small tree (most left) and a big tree (most right), which have a significantly different number of branches. Observe how new branch structures grow along the geodesic paths.

interpolated with other trees, which will allow the exploration of the tree-shape space.

Figure 9 shows the effect of rotation on the generated blendings. In this example, two geodesics are computed for each pair of trees; the first one is between S and T and the second one between S and T', which is exactly the same tree as T but rotated 180° around the vertical axis. Observe that different orientations of the target tree result in different smooth geodesics. While this can be seen as a limitation in statistical shape analysis since one looks for an analysis framework that is invariant to all shape-preserving transformation, it is beneficial for 3D modeling since a rich set of variations can be generated by just rotating one of the endpoint trees.

Additional examples of geodesics generated with the proposed framework are included in the attached supplementary material. The supplementary material also illustrates and discusses the quality of the branchwise correspondences computed using the proposed framework.

Finally, the length of the geodesic between two botanical trees is a measure of dissimilarity between these trees. To illustrate the utility of this measure for botanical tree classification, we took 30 trees and computed their pairwise geodesic distance matrix. Figure 10 shows the dendrogram plot of this pairwise distance matrix. The plot is generated using MATLAB's linkage function and using the shortest distance as a measure of dissimilarity between clusters. As one can see, the metric clusters trees that have similar geometry and structure.

Comparison to the state-of-the-art. We compare the geodesics obtained using our approach with geodesics computed using the approach of Feragen et al. (2013a) and the inverse procedural

Fig. 9. Effect of rotation on the computed geodesics. For each source tree S and target tree T belonging to different species, we compute two geodesics: one between S and T and another between S and T' obtained by rotating T 180° around the vertical axis. Observe that, in both cases, the computed geodesics are smooth and natural. More examples are included in the supplementary material.

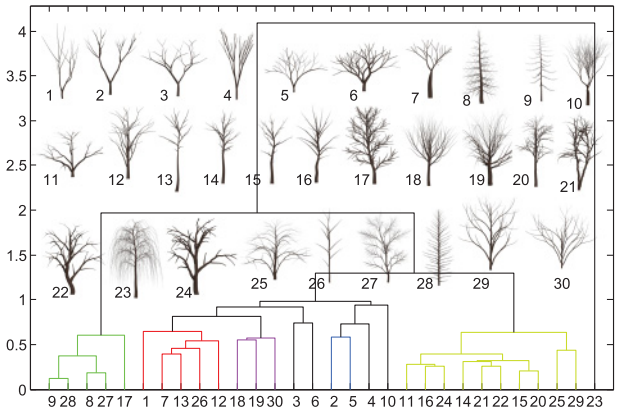

Fig. 10. Dendrogram plot of the pairwise distance matrix between 30 tree models computed using the metric proposed in this article. The plot is generated using MATLAB's linkage function, which uses the shortest distance as a measure of dissimilarity between clusters.

modeling of Stava et al. (2014). We take two sets of parameters of a procedural model learned using the approach of Stava et al. (2014), use one set to generate a source tree and another to generate a target tree. We then compute a geodesic between the source and target trees using the approach of Feragen et al. (2013a), our approach, and the approach of Stava et al. (2014). For the last, we took the parameters of the source and target trees, interpolated them, and then generated their corresponding trees (code kindly provided by the authors). The results of the computed geodesics with the three approaches are shown in Figure 11.

As one can see, since the approach of Feragen et al. (2013a) fails to find correct correspondences between the source and target trees, it does not produce plausible geodesics. In particular, the higher the complexity of the source and target trees, the lower the quality of the computed geodesics. On the other hand, the approach of Stava et al. (2014) does produce biologically plausible trees. However, due to its stochastic nature, the produced in-between trees are not smooth. In fact, the intermediate trees are random trees that look either similar to the source or to the target. More important, the mapping from the space of the procedural model parameters to the space of botanical trees is not one to one; one point in the parameter space corresponds to multiple trees. As a consequence, the approach of Stava et al. (2014) can be used to generate random trees but not for computing geodesics between a pair of trees. Compared to these state-of-the-art works, our approach produces smooth geodesics even between complex trees.

User study. We conducted a user study on 62 participants aimed at comparing our method with the approach of Feragen et al. (2013a) and the approach of Stava et al. (2014). Each participant is shown three geodesics between a pair of trees generated with the three methods (as in Figure 11). The participants then assign a score between 0 and 10 for each example. The higher the score, the better the result from the user perspective. We have collected the user scores and summarized them in Table 1. The table shows for each method how many participants among the 62 assigned a specific score to that method. It also provides the average and median scores. The last column shows for each example and for each method how many participants selected that method as the best in terms of the quality of the deformation path. As one can see, all the statistics show that the results produced with our method outperformed those of Feragen et al. (2013a) and Stava et al. (2014). The supplementary material includes additional discussion about the results of the user study.

Mean tree-shape computation. The average of a pair of trees is a by-product of the geodesic computation process. In fact, the

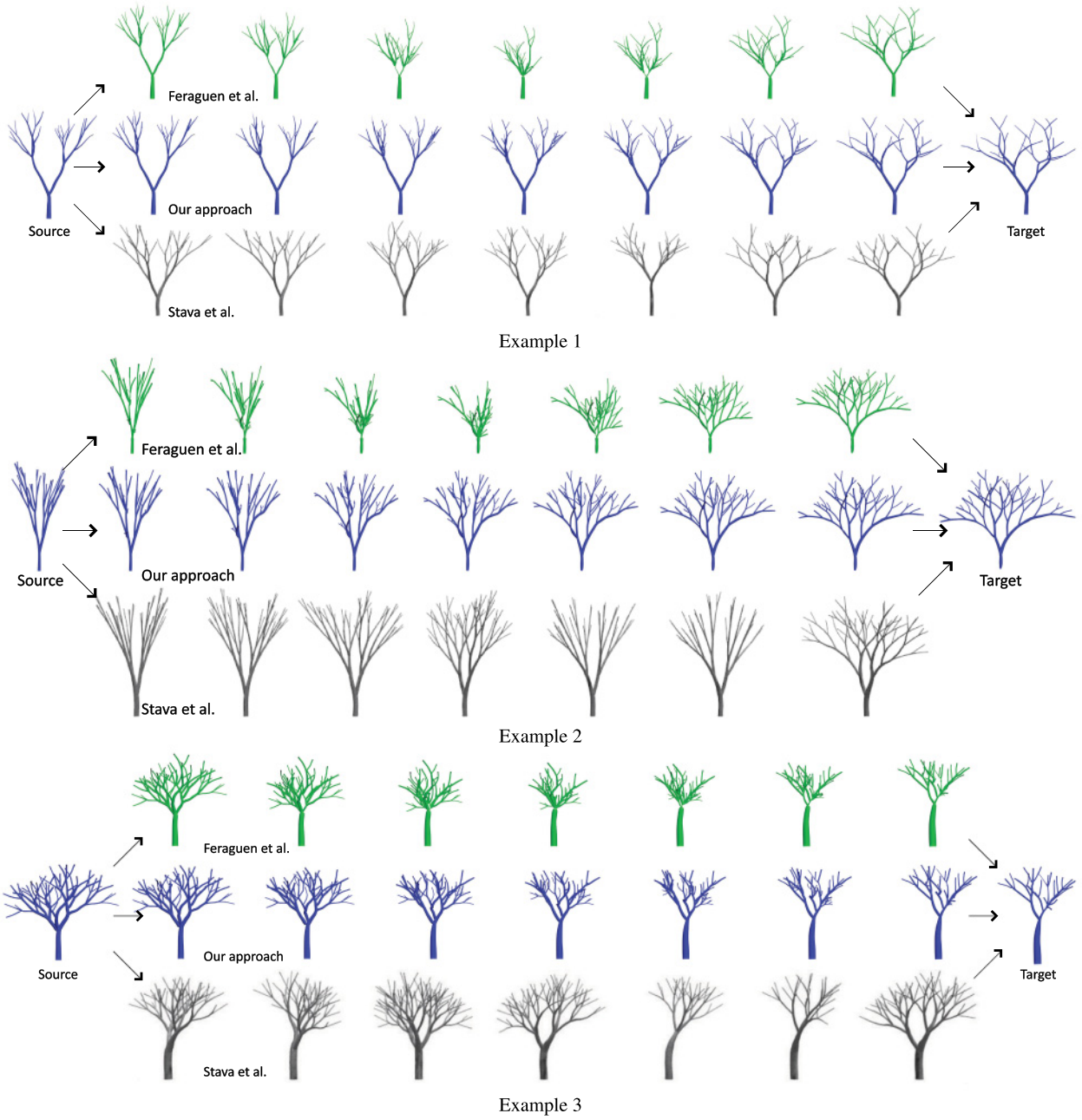

Fig. 11. Comparison between our approach, the approach of Feragen et al. (2013a), and inverse procedural modeling of Stava et al. (2014).

tree that is at equidistance from the source and target trees is exactly the mean tree of these two. Thus, the tree that is exactly at the middle of each geodesic in Figures 6, 7, 8, and 9 corresponds to the average of the leftmost and the rightmost trees.

Figures 12 and 13 show examples of mean trees computed from several other trees belonging to the same species and to different species. The input 3D trees are shown on the left while the generated mean trees, with and without synthesized leaves, are shown on the right. Note that the correspondence for all examples shown in these figures requires no user assistance other than setting their upright orientation. All subsequent steps are automatic, including the main trunk identification and tree skeletonization.

Finally, we compare the proposed approach for computing the average tree with baseline techniques, such as morphable models (Blanz and Vetter 1999; Allen et al. 2003), which are geometry based in contrast to the approach proposed in this article, which combines both geometry and structure. Figure 13 shows the average trees computed using the proposed framework. Figure 14

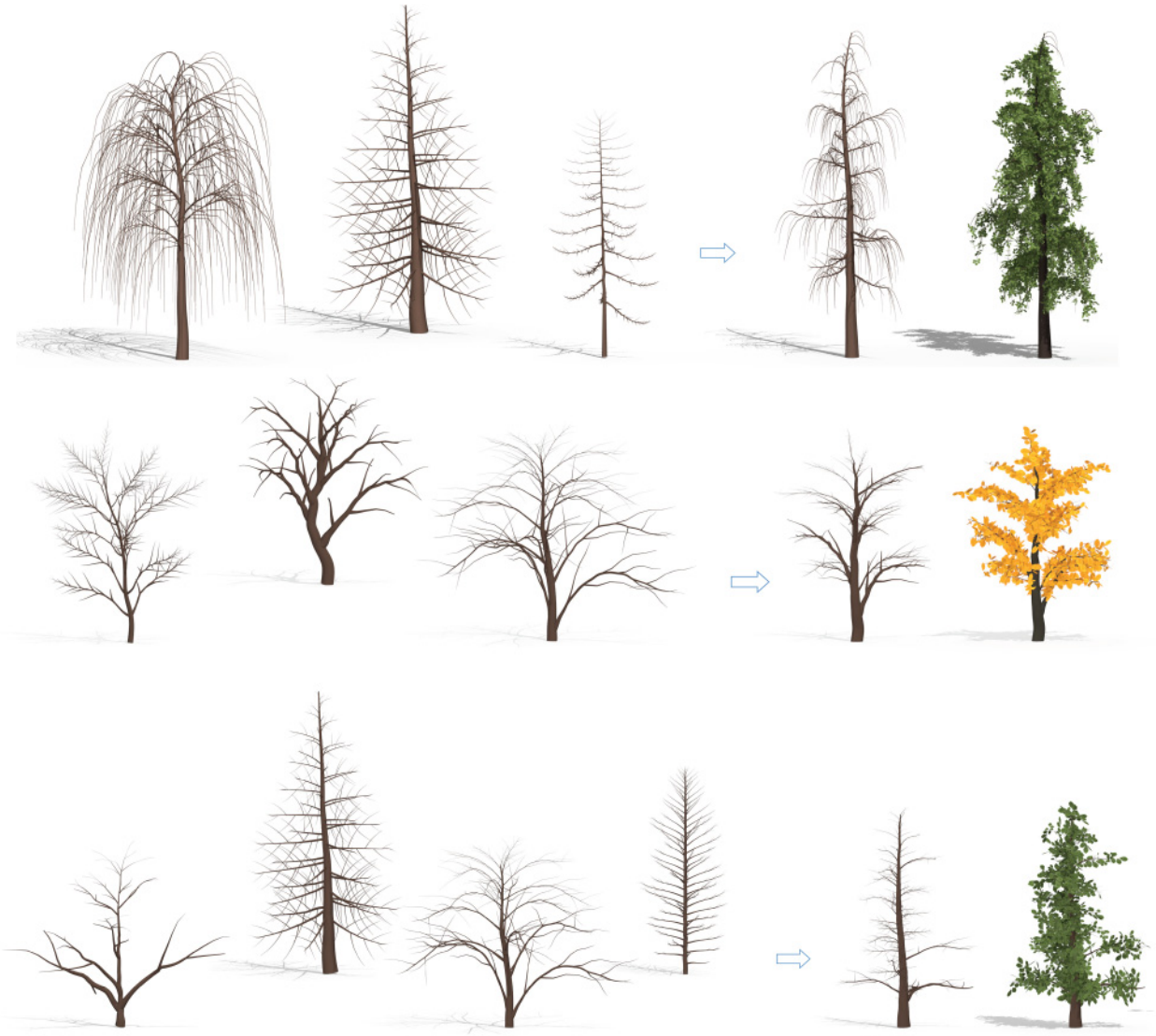

Fig. 12. Examples of average trees computed with trees taken from various species. The input 3D tree models are shown on the left. The computed average 3D tree, without and with synthesized leaves, is shown on the right.

shows the mean tree of the same input 3D trees of Figure 13 but obtained using the standard linear averaging employed in morphable models. To obtain these results, we first put the skeletons of the input trees in correspondence using the approach described in Section 3. Observe that previous techniques produce mean tree shapes with several artifacts and degeneracies in contrast to our approach, which produces mean trees with a richer structure, reflecting the properties of the data being averaged.

Additional results of mean trees computed with the proposed framework are included in the attached supplementary material.

Timing. On average, the processing time needed to compute the geodesics shown in this article is 3 minutes on an Intel(R) Core(TM) i7, 5500U CPU with 2.40GHz and 8GB of RAM. All codes have been implemented in MATLAB except the geodesic computation part, which is implemented in Java. Table 2 provides a breakdown of this processing time.

For comparison, in the approach of Stava et al. (2014), which does not produce smooth geodesics, the computation time for estimating the parameters of one tree model ranges from 12 to 85 minutes for trees of 298 to 587 nodes and 270 minutes for a tree of 2307 nodes. Once the parameters are learned, generating one tree model is instantaneous (less than one second). Thus, the computation of one interpolation would require 24 to 540 minutes. Ours only takes 18 seconds to 11 minutes per geodesic for trees of similar complexity (28 to 790 nodes). This is a significant advantage over inverse procedural modeling. In addition, the geodesics computed with our approach are smooth and natural.

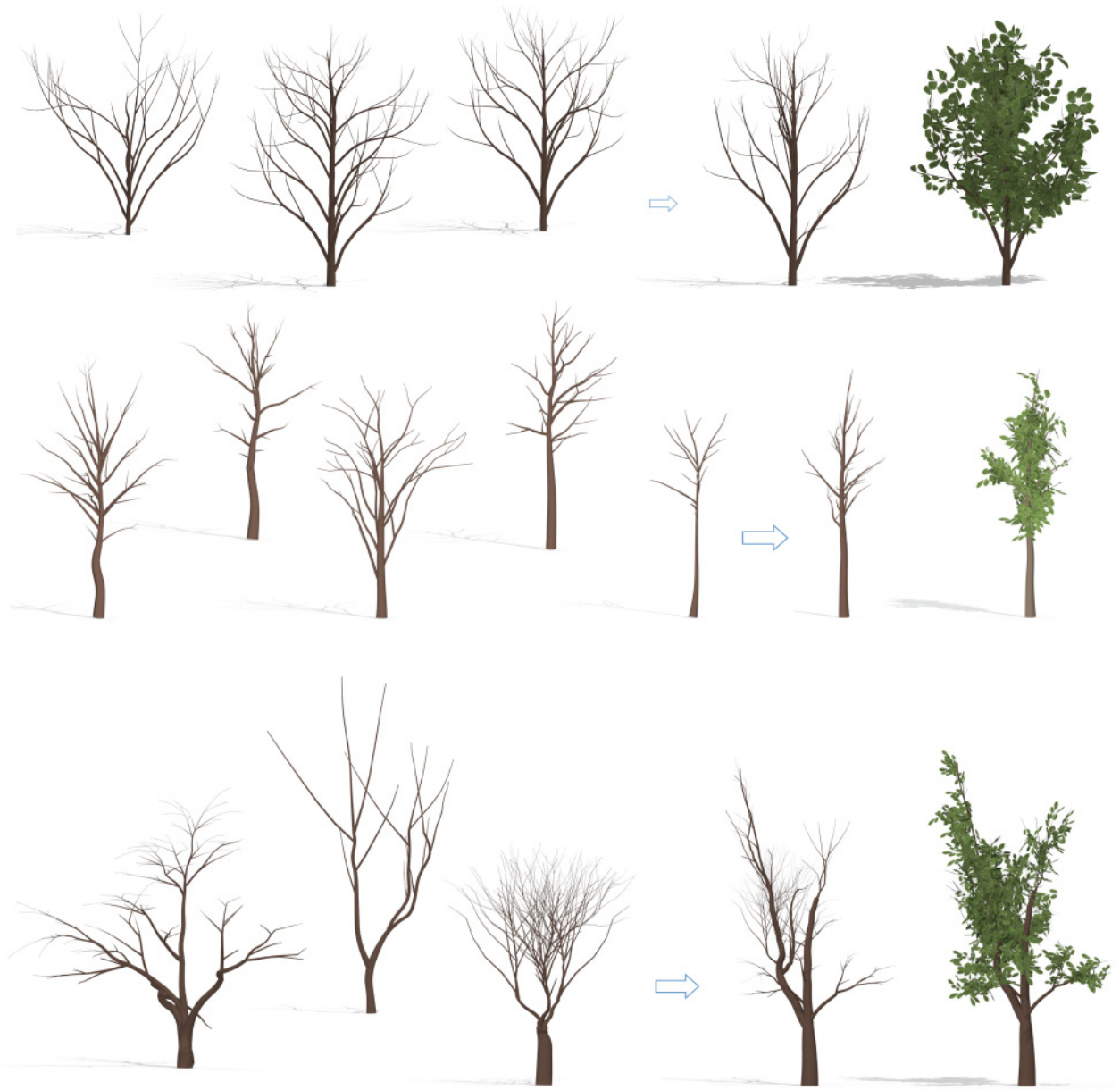

Fig. 13. Examples of average trees computed with trees taken from various species. The input 3D tree models are shown on the left. The computed average 3D tree, without and with synthesized leaves, is shown on the right.

Using the same setup, computing the mean tree of five trees takes 10 minutes for Figure 12 (top), 142 seconds for Figure 12 (middle), 5 minutes for Figure 12 (bottom), 7 seconds for Figure 13 (top), 5 seconds for Figure 13 (middle), and 77 seconds for Figure 13 (bottom). Note that the processing time is not affected by the number of polygons but rather by the complexity of their structure, that is, the depth of the trees and the number of their branches.

Finally, note that much of the existing software for interactive 3D tree modeling requires from the users manual specification of every detail, such as the positions, lengths, and bending degrees of the branches. They are time consuming and labor intensive even for skilled users. The framework proposed in this article is fully automatic and does not require any expertise from the users.

6 APPLICATION TO REFLECTION SYMMETRY ANALYSIS AND SYMMETRIZATION OF TREES

Symmetry is an important feature of an object and can be useful in many different applications. We refer the reader to the survey by Mitra et al. (2013), which discusses the importance of symmetry analysis and symmetrization of natural and manmade objects from the biological as well as computer graphics perspectives. It also surveys the different techniques used for symmetry analysis,

Table 1. User Study Results

		Avg score	Median score	Best path votes
Exple 1	Feragen et al. (2013a)	5.53	5	13
	Our method	**7.64**	**8**	**36**
	Stava et al. (2014)	6.33	6.5	10
Exple 2	Feragen et al. (2013a)	5.5	6	8
	Our method	**7.96**	**8.5**	**39**
	Stava et al. (2014)	6.41	7	11
Exple 3	Feragen et al. (2013a)	6.22	6.5	8
	Our method	**7.77**	**8.5**	**33**
	Stava et al. (2014)	5.98	7	16

Note: For each example and for each method, we take the average and median scores. The last column shows the number of users that selected the result of each method as the best.

Table 2. Complexity, In Terms of Number of Nodes, of the Trees Used in this Article and Geodesic Computation Time

	Number of nodes		Computation time (secs)		
Figure	Source	Target	Correspondence	Geodesic	Total
1	118	484	3.67	171.40	175.07
6 (top)	362	226	4.49	195.96	200.45
6 (bottom)	302	372	3.15	105.94	109.09
7 (top)	118	338	2.63	78.69	81.32
7 (bottom)	118	430	3.70	145.94	149.64
8	28	790	6.08	658.71	664.79
9	118	28	0.92	17.02	17.94

Note: The columns Source and Target indicate the number of nodes in the source and target trees.

which are mainly based on feature detection and matching. Because the proposed framework provides a proper metric and a mechanism for computing geodesics, it can be used in a straightforward manner for analyzing the reflection symmetry and for symmetrizing botanical trees.

To analyze the level of asymmetry of a given botanical tree T using our framework, we first obtain its reflection with respect to an arbitrary plane Δ that is orthogonal to the ground plane ($x-z$ plane in our case). Let $v \in \mathbb{R}^3$ be a normal vector to this plane. Then, the reflection of the tree T with respect to Δ is given by

$$\tilde{T} = \left(I - 2\frac{vv^T}{v^T v}\right) T. \tag{7}$$

Here, I is the 3×3 identity matrix and v^T refers to the transpose of the vector v. Next, we use the approach described in this article to compute a geodesic path, F^*, between T and $\tilde{T}$. F^* provides valuable information about the symmetry of T. First, its length gives a formal measure of asymmetry of T. Second, the halfway point along this geodesic—that is, $F^*(0.5)$—is symmetric. Moreover, among all symmetric trees, $F^*(0.5)$ is the nearest to T. Thus, the process of computing a geodesic between a botanical tree and its reflection is equivalent to symmetrizing the tree.

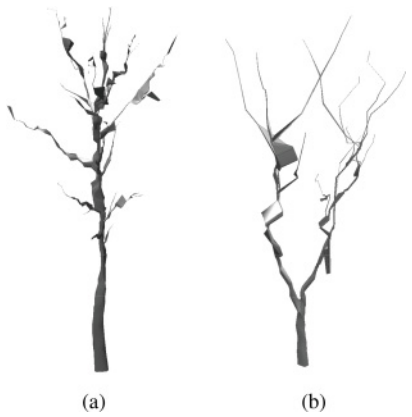

Fig. 14. Standard linear average of the trees in the first (a) and second (b) rows of Figure 13. The averages are obtained using standard principle component analysis, as in morphable models (Blanz and Vetter 1999; Allen et al. 2003), which are geometry based.

In Figure 15, we present several examples of symmetrizing highly complex botanical trees. Note that the highlighted midpoints of the presented geodesics are symmetric; thus, these paths provide natural symmetrizations of the given trees.

7 CONCLUSION

We presented in this article a new framework for generating smooth blending between 3D botanical tree models via geodesic analysis in the shape space of trees. To the best of our knowledge, this is the first data-driven approach that is specifically designed to capture both the geometric, structural, and topological variations that are present in botanical trees. We demonstrated several examples of geodesics computed between trees that significantly differ in the number of branches and in their structure. The framework also provides a proper metric, which can be used for comparing trees; but, more important, it allows us to compute average statistics, such as the mean tree shape. We have also demonstrated that the framework can be used for reflection symmetry analysis and symmetrization of 3D botanical trees. The approach is fully automatic, except when defining the upright orientation of the trees, which further demonstrates the practical utility of the framework.

The supplementary materials included with this article present additional results and animations.

Limitations and future work. There are several aspects of the proposed framework that require improvement. First, correspondences are found levelwise under the assumption that each tree's trunk has been identified. We will explore in the future the possibility of relaxing this condition using, for example, fuzzy correspondences, as in Wang et al. (2016). Second, the proposed framework considers only the tree branches and ignores the tree leaves. Leaves can be easily synthesized on the generated trees, as commonly done in most of the previous works, for example, Bradley et al. (2013). However, we need to ensure that the foliage structure changes smoothly along the blending paths. This cannot be done unless both foliage and branching structures are jointly represented in a common shape space.

Fig. 15. Examples of reflection symmetry analysis and symmetrization of botanical tree models. The highlighted midpoint of the geodesic between the leftmost tree and its reflection (the rightmost tree) is a symmetrized version of the leftmost tree.

Third, the quality of the computed geodesics indicates that the proposed approach for finding correspondences is efficient. In this article, did not perform a quantitative evaluation of the quality of the correspondences due to lack of a proper benchmark. This is left for future work. Fourth, although the computation time (2 minutes per geodesic) is significantly less than the time required in manual as well as in inverse procedural modeling of 3D trees, we still believe that it can be significantly improved by optimizing the implementation, for example, by using a hierarchical procedure since tree branches are organized into levels.

On the technical side, generated geodesics are not invariant to the rotation of the trees around the vertical axis. That is, rotating the source or target trees around the vertical axis will result in a new geodesic. While from the shape synthesis perspective this can be seen as an advantage since it enables the user to generate new variations by rotating just one of the endpoint trees, it is not desirable from the statistical shape analysis perspective since one looks for an analysis framework that is invariant to all shape-preserving transformations. One way to address this is to find the minimal geodesic over all possible orientations around the vertical axis.

Another limitation of the framework is related to the use of the QED for the computation of geodesics. QED is essentially a piecewise Euclidean distance. Geodesics under this metric are piecewise linear segments; each segment lies in one orthant. As such, in the presence of branches that significantly bend, the branches will shrink along the geodesic. One way to address this is to replace the QED with an elastic metric similar to those proposed in Srivastava et al. (2011) and Laga et al. (2014, 2017). In these works, curves are represented with their SRVFs, which are essentially normal vectors scaled by the square root of the local length of the curves. Geodesics computed with the Euclidean metric in the space of SRVFs do not suffer from shrinkage artifacts. Using these representations, however, is computationally expensive. In addition, the mapping from the SRVF space back to the original space is unique up to translation. This makes them difficult to use for computing geodesics in tree-structured data since mapping back to the original spaces loses the location of the branches.

Another limitation is that the proposed approach relies on a common parameterization, using a maximal binary tree of the structure of all ttrees being analyzed. While it is not an issue when computing pairwise geodesics, it might be expensive in terms of memory requirement when dealing with a large collection of complex 3D tree models.

While the proposed approach synthesizes visually plausible blendings between 3D tree models, we do not claim that the synthesized blendings are biologically correct. However, having a well-defined tree-shape space with a proper metric and a mechanism for computing geodesics are important steps toward modeling biological growth and evolution. With this framework, one can treat growth, evolution, or even environmental effects (such as wind, light sources, and obstacles) on the shape of trees as trajectories in the tree-shape space and use the rich literature of curve analysis to model them mathematically. One can also use the concept of parallel transport to transfer the growth of a certain tree to

another. This article provides the first building blocks, which are correspondence, tree-shape space, a metric, and a mechanism for computing geodesics and statistical summaries.

Finally, having a well-defined tree-shape space with a proper metric and a mechanism for computing geodesics are also important steps toward full statistical modeling of shape variation in botanical trees. With these computational tools, populations of trees can be characterized with probability distributions and new instances can be created by sampling from these distributions. This will generate richer structural variations compared to what can be created by just blending pairs of trees. This article provides the first, but important, steps toward this ultimate goal.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers and the Associate Editor for their valuable feedback. The authors are grateful to Aasa Feragen and colleagues for various discussions and for making publicly available the codes of their work on tree-shape spaces. The authors would like also to thank Ibraheem Alhashim and colleagues for the discussion about their work on Topology-Varying 3D Shape Creation via Structural Blending, and Bedrich Benes, Sören Pirk, and Ondrej Stava for the discussion about inverse procedural modeling and for sharing their codes. The trees used in this article are from https://archive3d.net/ and https://xfrog.com/. This research is partially supported by the Fundamental Research Funds for Chinese Central Universities (Project 2100219066 and 0200219153).

REFERENCES

Carlos A. Alfaro, Burcu Aydın, Carlos E. Valencia, Elizabeth Bullitt, and Alim Ladha. 2014. Dimension reduction in principal component analysis for trees. *Computational Statistics & Data Analysis* 74, 157–179.

Ibraheem Alhashim, Honghua Li, Kai Xu, Junjie Cao, Rui Ma, and Hao Zhang. 2014. Topology-varying 3d shape creation via structural blending. *ACM Transactions on Graphics* 33, 4, Article 158.

Brett Allen, Brian Curless, and Zoran Popović. 2003. The space of human body shapes: Reconstruction and parameterization from range scans. *ACM Transactions on Graphics* 22, 3, 587–594.

Masaki Aono and Tosiyasu Kunii. 1984. Botanical tree image generation. *IEEE Computer Graphics and Appllications* 4, 5, 10–34. DOI: https://doi.org/10.1109/MCG.1984.276141

Bedrich Beneš and Erik Uriel Millán. 2002. Virtual climbing plants competing for space. In *Proceedings of the Computer Animation (CA'02)*. IEEE Computer Society, Washington, DC, USA, 33–. http://dl.acm.org/citation.cfm?id=791218.791582

S. Biasotti, A. Cerri, A. Bronstein, and M. Bronstein. 2015. Recent trends, applications, and perspectives in 3D shape similarity assessment. In *Computer Graphics Forum*.

Louis J. Billera, Susan P. Holmes, and Karen Vogtmann. 2001. Geometry of the space of phylogenetic trees. *Advances in Applied Mathematics* 27, 4, 733–767.

Volker Blanz and Thomas Vetter. 1999. A morphable model for the synthesis of 3D faces. In *Siggraph*. ACM Press/Addison-Wesley Publishing Co., New York, NY, 187–194.

Jules Bloomenthal. 1985. Modeling the mighty maple. *SIGGRAPH* 19, 3 (July 1985), 305–311. DOI: https://doi.org/10.1145/325165.325249

Rolf Borchert and Norman A. Slade. 1981. Bifurcation ratios and the adaptive geometry of trees. *Botanical Gazette* 142, 3, 394–401.

Stefan Bornhofen and Claude Lattaud. 2009. Competition and evolution in virtual plant communities: A new modeling approach. *Natural Computing* 8, 2, 349–385.

Derek Bradley, Derek Nowrouzezahrai, and Paul Beardsley. 2013. Image-based reconstruction and synthesis of dense foliage. *ACM Transactions on Graphics* 32, 4, 10.

Xuejin Chen, Boris Neubert, Ying-Qing Xu, Oliver Deussen, and Sing Bing Kang. 2008. Sketch-based tree modeling using Markov random field. *ACM Transactions on Graphics* 27, 5, Article 109, 9 pages. DOI: https://doi.org/10.1145/1409060.1409062

P.-H. Cournede, Veronique Letort, Amélie Mathieu, Meng Zhen Kang, Sébastien Lemaire, Samis Trevezas, François Houllier, and Philippe De Reffye. 2011. Some parameter estimation issues in functional-structural plant modelling. *Mathematical Modelling of Natural Phenomena* 6, 2, 133–159.

Phillippe de Reffye, Claude Edelin, Jean Françon, Marc Jaeger, and Claude Puech. 1988. Plant models faithful to botanical structure and development. *SIGGRAPH* 22, 4, 151–158. DOI: https://doi.org/10.1145/378456.378505

Oliver Deussen and Bernd Lintermann. 2004. *Digital Design of Nature: Computer Generated Plants and Organics*. Springer, New York, NY.

Aasa Feragen, Pechin Lo, Marleen de Bruijne, Mads Nielsen, and François Lauze. 2013a. Toward a theory of statistical tree-shape analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 8, 2008–2021.

Aasa Feragen, Megan Owen, Jens Petersen, Mathilde M. W. Wille, Laura H. Thomsen, Asger Dirksen, and Marleen de Bruijne. 2013b. Tree-space statistics and approximations for large-scale analysis of anatomical trees. In *Information Processing in Medical Imaging (IPMI'13)*. 74–85. DOI: https://doi.org/10.1007/978-3-642-38868-2_7

N. Greene. 1989. Voxel space automata: Modeling with stochastic growth processes in voxel space. *SIGGRAPH* 23, 3, 175–184. DOI: https://doi.org/10.1145/74334.74351

Hisao Honda. 1971. Description of the form of trees by the parameters of the tree-like body: Effects of the branching angle and the branch length on the shape of the tree-like body. *Journal of Theoretical Biology* 31, 2, 331–338. DOI: https://doi.org/10.1016/0022-5193(71)90191-3

Jing Hua and Mengzhen Kang. 2011. Functional tree models reacting to the environment. In *ACM SIGGRAPH 2011 Posters*. ACM, New York, NY, USA, Article 60, 1 pages. DOI: https://doi.org/10.1145/2037715.2037783

Arjun Jain, Thorsten Thormählen, Tobias Ritschel, and Hans-Peter Seidel. 2012. Exploring shape variations by 3D-model decomposition and part-based recombination. In *Computer Graphics Forum*, Vol. 31. Wiley Online Library, Hoboken, NJ, 631–640.

Evangelos Kalogerakis, Siddhartha Chaudhuri, Daphne Koller, and Vladlen Koltun. 2012a. A probabilistic model for component-based shape synthesis. *ACM Transactions on Graphics* 31, 4, Article 55, 11 pages.

Evangelos Kalogerakis, Siddhartha Chaudhuri, Daphne Koller, and Vladlen Koltun. 2012b. A probabilistic model for component-based shape synthesis. *ACM Transactions on Graphics* 31, 4, 55.

Martin Kilian, Niloy J. Mitra, and Helmut Pottmann. 2007. Geometric modeling in shape space. *ACM Transactions on Graphics* 26, 3, 64.

Sebastian Kurtek, Anuj Srivastava, Eric Klassen, and Hamid Laga. 2013. Landmark-guided elastic shape analysis of spherically-parameterized surfaces. *Computer Graphics Forum* 32, 2pt4, 429–438.

Hamid Laga, Sebastian Kurtek, Anuj Srivastava, and Stanley J. Miklavcic. 2014. Landmark-free statistical analysis of the shape of plant leaves. *Journal of Theoretical Biology* 363, 41–52.

Hamid Laga, Qian Xie, Ian H. Jermyn, and Anuj Srivastava. 2017. Numerical inversion of SRNF maps for elastic shape analysis of genus-zero surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39, 12 (2017), 2451–2464.

Chuan Li, Oliver Deussen, Yi-Zhe Song, Phil Willis, and Peter Hall. 2011. Modeling and generating moving trees from video. *ACM Transactions on Graph.ics* 30, 6, Article 127, 12 pages. DOI: https://doi.org/10.1145/2070781.2024161

Aristid Lindenmayer. 1968. Mathematical models for cellular interactions in development I. Filaments with one-sided inputs. *Journal of Theoretical Biology* 18, 3, 280–299. DOI: https://doi.org/10.1016/0022-5193(68)90079-9

Bernd Lintermann and Oliver Deussen. 1999. Interactive modeling of plants. *IEEE Computer Graphics and Application* 19, 1, 56–65. DOI: https://doi.org/10.1109/38.736469

Yotam Livny, Soeren Pirk, Zhanglin Cheng, Feilong Yan, Oliver Deussen, Daniel Cohen-Or, and Baoquan Chen. 2011. Texture-lobes for tree modelling. *ACM Transactions on Graphics* 30, 4, Article 53, 10 pages. DOI: https://doi.org/10.1145/2010324.1964948

Yotam Livny, Feilong Yan, Matt Olson, Baoquan Chen, Hao Zhang, and Jihad El-Sana. 2010. Automatic reconstruction of tree skeletal structures from point clouds. *ACM Transactions on Graphics* 29, 6, Article 151, 8 pages. DOI: https://doi.org/10.1145/1882261.1866177

Niloy J. Mitra, Mark Pauly, Michael Wand, and Duygu Ceylan. 2013. Symmetry in 3D geometry: Extraction and applications. In *Computer Graphics Forum*, Vol. 32. Wiley Online Library, Hoboken, NJ, 1–23.

Radomír Měch and Przemyslaw Prusinkiewicz. 1996. Visual models of plants interacting with their environment. In *Proceedings of SIGGRAPH*. ACM, New York, NY, USA, 397–410. DOI: https://doi.org/10.1145/237170.237279

Boris Neubert, Thomas Franken, and Oliver Deussen. 2007. Approximate image-based tree-modeling using particle flows. *ACM Transactions on Graphics* 26, 3, Article 88.DOI: https://doi.org/10.1145/1276377.1276487

Makoto Okabe, Shigeru Owada, and Takeo Igarashi. 2007. Interactive design of botanical trees using freehand sketches and example-based editing. In *ACM SIGGRAPH 2007 courses*. ACM, Article 26. DOI: https://doi.org/10.1145/1281500.1281537

Peter E. Oppenheimer. 1986. Real time design and animation of fractal plants and trees. *SIGGRAPH* 20, 4, 55–64. DOI: https://doi.org/10.1145/15886.15892

Megan Owen and J. Scott Provan. 2011. A fast algorithm for computing geodesic distances in tree space. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 8, 1, 2–13.

Wojciech Palubicki, Kipp Horel, Steven Longay, Adam Runions, Brendan Lane, Radomír Měch, and Przemyslaw Prusinkiewicz. 2009. Self-organizing tree models for image synthesis. *ACM Transactions on Graphics* 28, 3, Article 58, 10 pages. DOI: https://doi.org/10.1145/1531326.1531364

Sören Pirk, Till Niese, Oliver Deussen, and Boris Neubert. 2012a. Capturing and animating the morphogenesis of polygonal tree models. *ACM Trans. Graph.* 31, 6, Article 169 (Nov. 2012), 10 pages. DOI: https://doi.org/10.1145/2366145.2366188

Sören Pirk, Ondrej Stava, Julian Kratt, Michel Abdul Massih Said, Boris Neubert, Radomír Měch, Bedrich Benes, and Oliver Deussen. 2012b. Plastic trees: Interactive self-adapting botanical tree models. *ACM Transactions on Graphics* 31, 4, Article 50, 10 pages. DOI: https://doi.org/10.1145/2185520.2185546

Joanna L. Power, A. J. Bernheim Brush, Przemyslaw Prusinkiewicz, and David H. Salesin. 1999. Interactive arrangement of botanical L-system models. In *Proceedings of the 1999 Symposium on Interactive 3D graphics*. ACM, 175–182. DOI: https://doi.org/10.1145/300523.300548

P. Prusinkiewicz. 1986. Graphical applications of L-systems. In *Proceedings on Graphics Interface'86/Vision Interface'86*. Canadian Information Processing Society, Toronto, ON, Canada, 247–253. http://dl.acm.org/citation.cfm?id=16564.16608

P. Prusinkiewicz and A. Lindernmayer. 1990. *The Algorithmic Beauty of Plants*. Springer, New York, NY.

Przemyslaw Prusinkiewicz, Lars Mündermann, Radoslaw Karwowski, and Brendan Lane. 2001. The use of positional information in the modeling of plants. In *Proceedings of SIGGRAPH*. ACM, New York, NY, USA, 289–300. DOI: https://doi.org/10.1145/383259.383291

Przemyslaw Prusinkiewiczy, Mark Hammely, Jim Hananz, and Radom Mech. 1996. L-System: From the theory to visual models of plants. *CSIRO Symposium on Computational Challanges in Life Sciences*.

Long Quan, Ping Tan, Gang Zeng, Lu Yuan, Jingdong Wang, and Sing Bing Kang. 2006. Image-based plant modeling. *ACM Transactions on Graphics* 25, 3, 599–604. DOI: https://doi.org/10.1145/1141911.1141929

Alex Reche-Martinez, Ignacio Martin, and George Drettakis. 2004. Volumetric reconstruction and interactive rendering of trees from photographs. *ACM Transactions on Graphics* 23, 3, 720–727. DOI: https://doi.org/10.1145/1015706.1015785

William T. Reeves and Ricki Blau. 1985. Approximate and probabilistic algorithms for shading and rendering structured particle systems. *SIGGRAPH* 19, 3, 313–322. DOI: https://doi.org/10.1145/325165.325250

Ilya Shlyakhter, Max Rozenoer, Julie Dorsey, and Seth Teller. 2001. Reconstructing 3D tree models from instrumented photographs. *IEEE Computer Graphics and Application* 21, 3, 53–61. DOI: https://doi.org/10.1109/38.920627

Anuj Srivastava, Eric Klassen, S. Joshi, and Ian Jermyn. 2011. Shape analysis of elastic curves in Euclidean spaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 99, 1–1.

Ondrej Stava, Sören Pirk, Julian Kratt, Baoquan Chen, R. Měch, Oliver Deussen, and Bedrich Benes. 2014. Inverse procedural modelling of trees. In *Computer Graphics Forum*, Vol. 33. Wiley Online Library, Hoboken, NJ, 118–131.

Karl-Theodor Sturm. 2003. Probability measures on metric spaces of nonpositive. *Heat Kernels and Analysis on Manifolds, Graphs, and Metric Spaces: Lecture Notes from a Quarter Program on Heat Kernels, Random Walks, and Analysis on Manifolds and Graphs: April 16-July 13, 2002, Emile Borel Centre of the Henri Poincaré Institute, Paris, France* 338, 357.

Ping Tan, Tian Fang, Jianxiong Xiao, Peng Zhao, and Long Quan. 2008. Single image tree modeling. *ACM Transactions on Graphics* 27, 5, Article 108, 7 pages. DOI: https://doi.org/10.1145/1409060.1409061

Stanislaw Ulam. 1962. On some mathematical problems connected with patterns of growth of figures. In *Proceedings of Symposia in Applied Mathematics*, Vol. 14. Am. Math. Soc. Vol. 14, Providence, 215–224.

Carlos A. Vanegas, Ignacio Garcia-Dorado, Daniel G. Aliaga, Bedrich Benes, and Paul Waddell. 2012. Inverse design of urban procedural models. *ACM Transactions on Graphics* 31, 6, 168.

Yutong Wang, Xiaowei Xue, Xiaogang Jin, and Zhigang Deng. 2016. Creative virtual tree modeling through hierarchical topology-preserving blending. *IEEE Transactions on Visualization and Computer Graphics* 23, 12 (2016), 2521–2534.

Jason Weber and Joseph Penn. 1995. Creation and rendering of realistic trees. In *SIGGRAPH*. ACM, New York, NY, USA, 119–128. DOI: https://doi.org/10.1145/218380.218427

Zhongke Wu, Mingquan Zhou, and Xingce Wang. 2009. Interactive modeling of 3D tree with ball b-spline curves. *International Journal of Virtual Reality* 8, 1.

Dejia Zhang, Ning Xie, Shuang Liang, and Jinyuan Jia. 2015. 3D tree skeletonization from multiple images based on PyrLK optical flow. *Pattern Recognition Letters* 76, 49–58.

Qi-Long Zhang and Ming-Yong Pang. 2008. A survey of modeling and rendering trees. In *Proceedings of the 3rd International Conference on Technologies for E-Learning and Digital Entertainment*. Springer-Verlag, 757–764. DOI: https://doi.org/10.1007/978-3-540-69736-7_80

Youyi Zheng, Daniel Cohen-Or, and Niloy J. Mitra. 2013. Smart variations: Functional substructures for part compatibility. In *Computer Graphics Forum*, Vol. 32. 195–204.

Received October 2016; revised September 2017; accepted September 2017