A Procedural Model for Diverse Tree Species

Rama Karl Hoetzlein

ramakarl.com

Quanta Sciences, Ithaca, NY

Assistant Professor, Digital Media Design
School of Entrepreneurship
Florida Gulf Coast University

Background

2005 © Public domain

Lindenmayer systems, 1968

Przemyslaw Prusinkiewicz, et al., 1997

L-systems: from the Theory to Visual Models of Plants

Przemyslaw Prusinkiewicz[1], Jim Hanan[2], Mark Hammel[1] and Radomir Mech[1]

[1] Department of Computer Science, University of Calgary, Calgary, Alberta, Canada T2N 1N4
[2] CSIRO — Cooperative Research Centre for Tropical Pest Management, Brisbane, Queensland, Australia

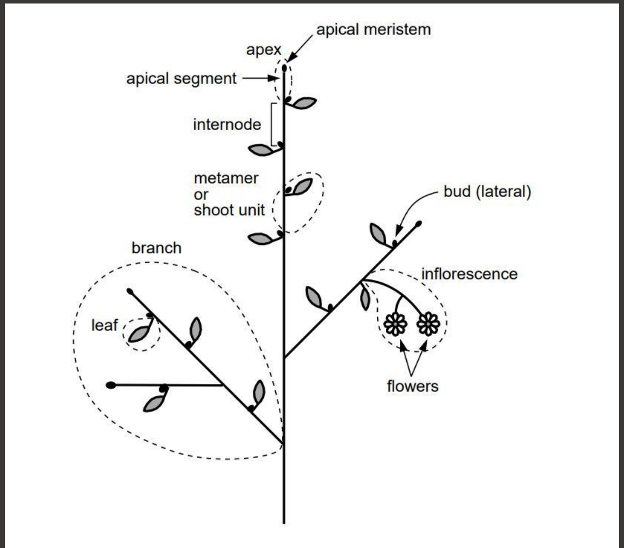


Figure 1: Selected modules and groups of modules (encircled with dashed lines) used to describe plants

L-systems	Growth as string rewrite.
Parametric L-systems	Probablistic /w parameters.

Motivation

Motivation

Gumbo Limbo tree

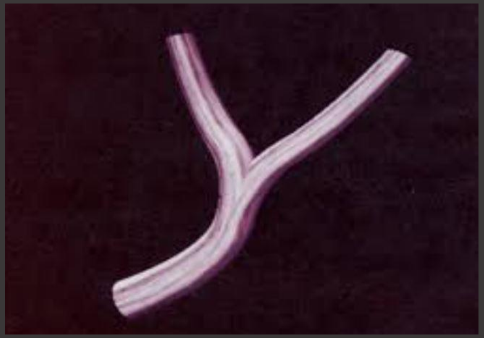
Dwarf Beech (Fagus sylvatica tortuosa)

Motivation

Twisting is not even understood by biologists.

It is related to mechanical stresses during cell growth, but some species (genetics) exhibit more twisting. Harder woods don't always twist less.

Background

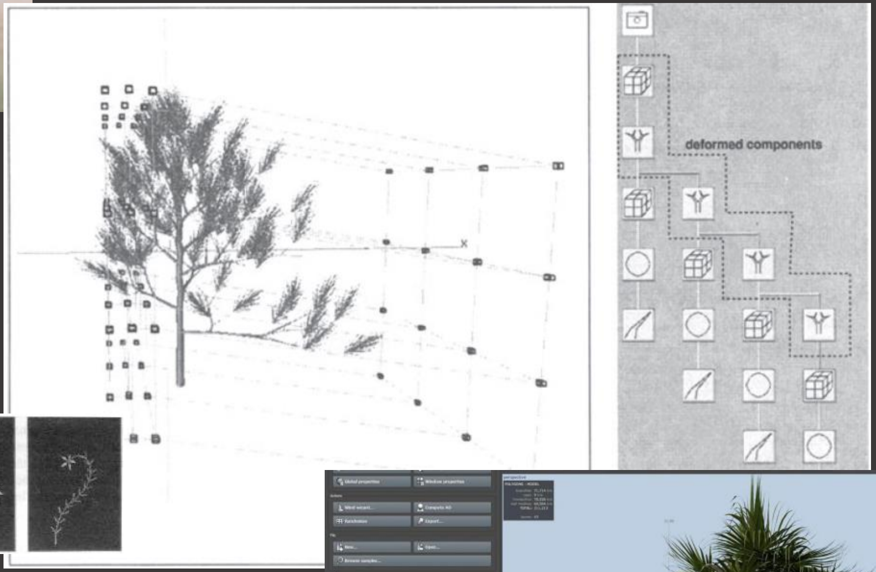


Holton:
Introduced
Bezier curves
for branches.

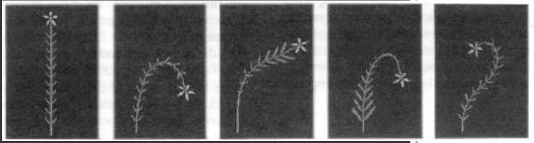
User-based Modeling

- 1994, Holton.
  Strands, Gravity and Botanical Tree Imagery

- 1996, Lintermann and Deussen.
  Interactive Modelling and Animation of Branching Botanical Structures



- 1999, Joanna Power et al.
  Interactive Arrangement of Botanical L-System Models



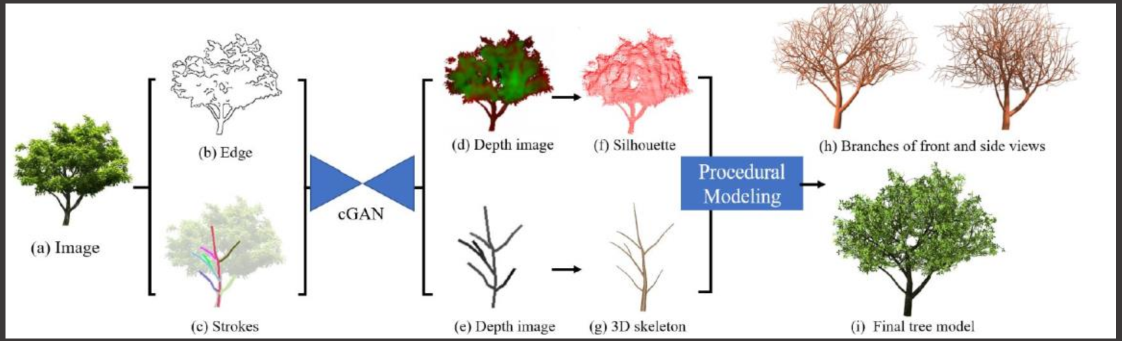
- 2008, Xeujin Chen et al.
  Sketch-Based Tree Modeling Using Markov Random Fields

Speedtree

Background

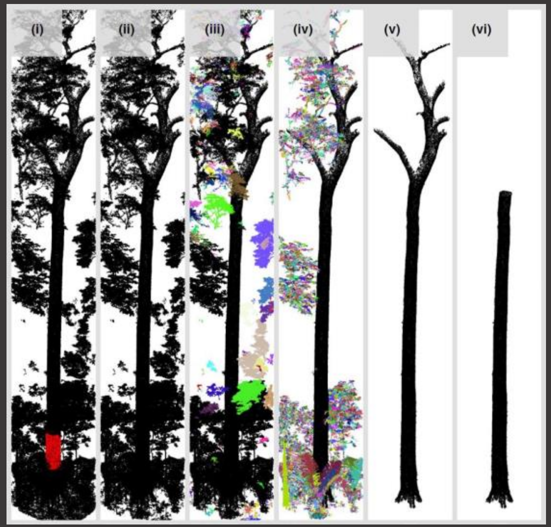
Data-driven Modeling

- 2015, Miller et al.
  3D Modeling of Individual Trees using a handheld camera

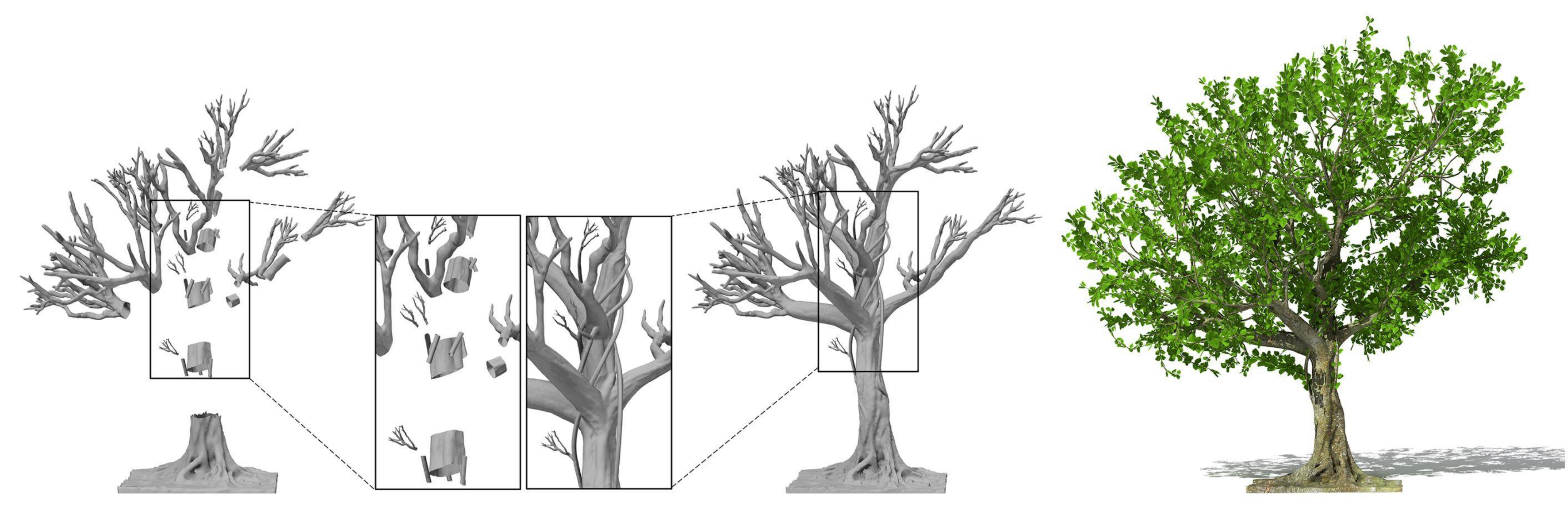


- 2021, Liu et al.
  Single Image Tree Reconstruction via Adversarial Network

- 2018, Burt, Disney, Calders
  Extracting individual trees from lidar point clouds using *treeseg*



Data-driven Modeling



2016, Xie et al.

Tree Modeling with Real Tree-Part Examples

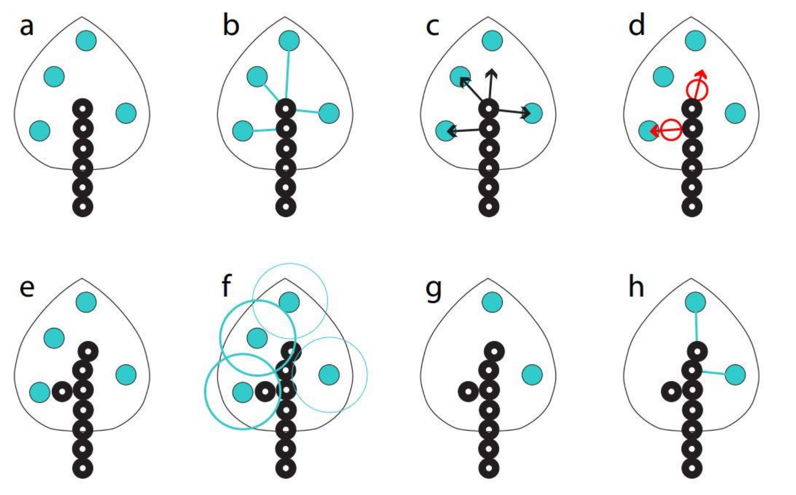
Data-driven Modeling

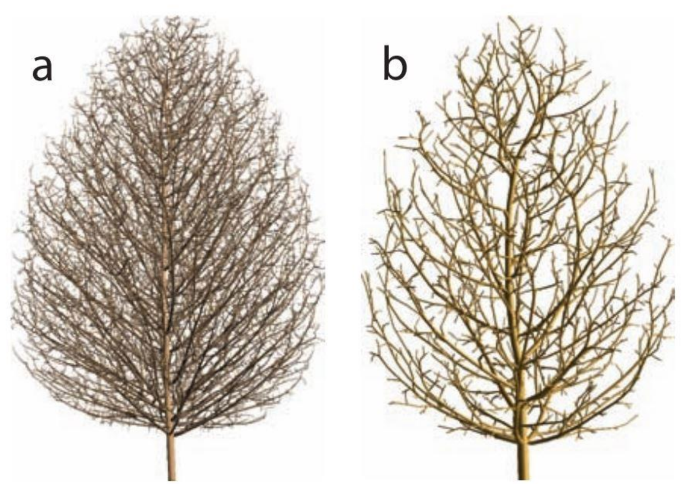
Data-driven Tree modeling commercialized by Quixel Megascans with massive models rendered in Unreal Engine 5 using Nanite virtualized geometry technology.

Requires professional capture team, extensive post-processing, and massive data models.

Background

Fully Procedural Models





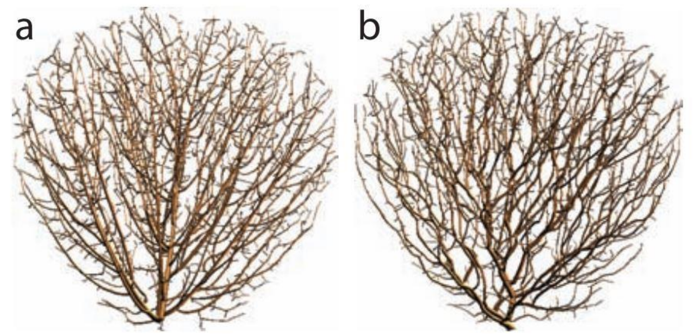


Figure 4: *Impact of the radius of influence d_i on the form of shrubs. a) $d_i = \infty$; b) $d_i = 17D$.*

20076, Runions, Lane, Prusinkiewicz

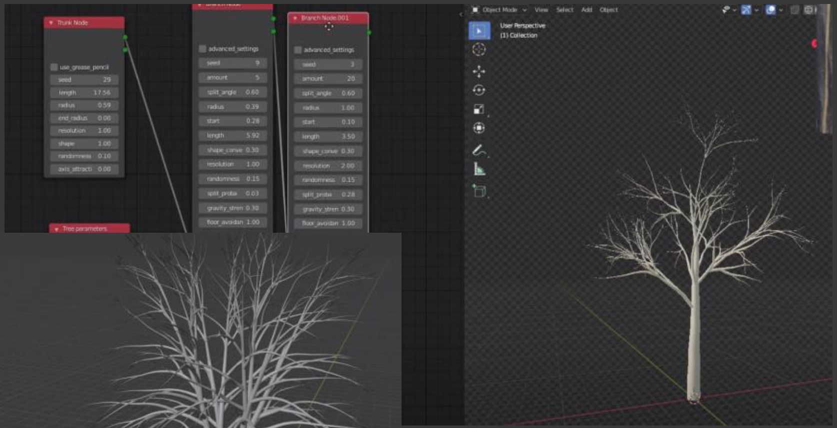
Modeling Trees with a Space Colonization Algorithm

The cornerstone of the proposed method is the space colonization algorithm (Figure 1b and c), which treats competition for space as the key factor determining the branching structure of trees

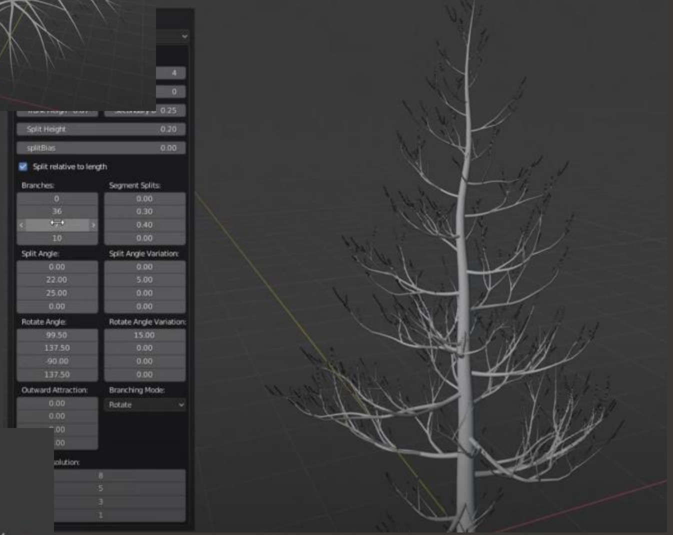
Motivation

Youtube Blender Tutorials

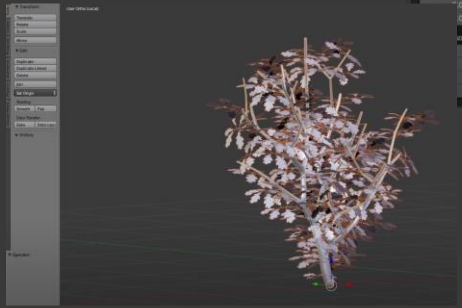
The Best Way to Make Tree! (Blender Tutorial)



How to Generate a Tree with Looping Wind



Create realistic trees in Blender 2.82



How to Create Realistic Trees

Trees are not a solved problem in computer graphics.

These trees are realistically rendered, but very limited procedurally and structurally.

Goals

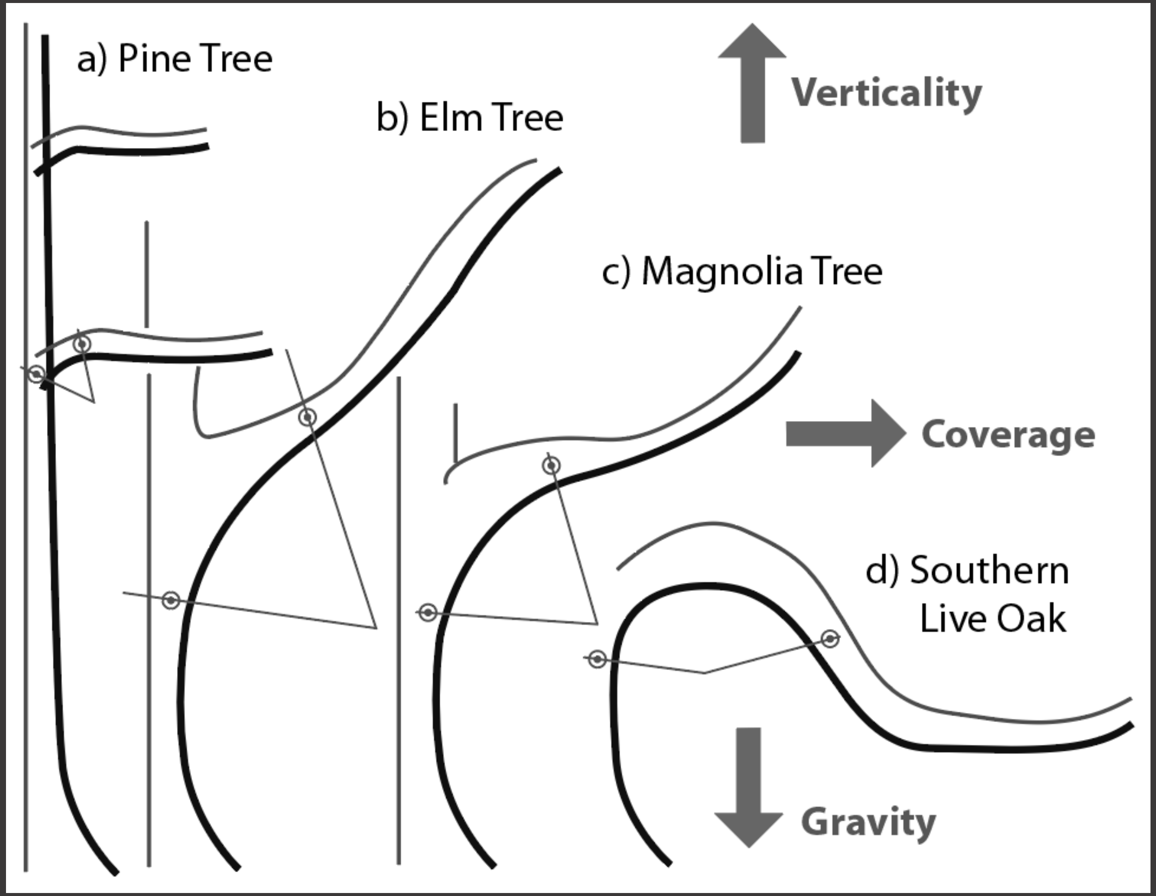
- Capture a wider variety of real branch shapes
- No manual modeling of branch curves
- Fully procedural model
- Species & individual variation
- Compact model – not data-driven
- Efficient – generate variations quickly

Algorithm Design

Key idea:
Tree branches are not arbitrary 3D curves.

They are influenced by common competing forces such as *gravity*, *coverage* and *verticality* (for sunlight)

Constraining the branch shape space leads to several useful qualities.



Algorithm Design

Key idea:

Tree branches are not arbitrary 3D curves.

Algorithm:

1. A Harmonic Procedural Model for Branch Shape
2. Particle-based Semantic Model for Tree Growth
3. Particle-to-Mesh Model for Surfacing

Algorithm Design

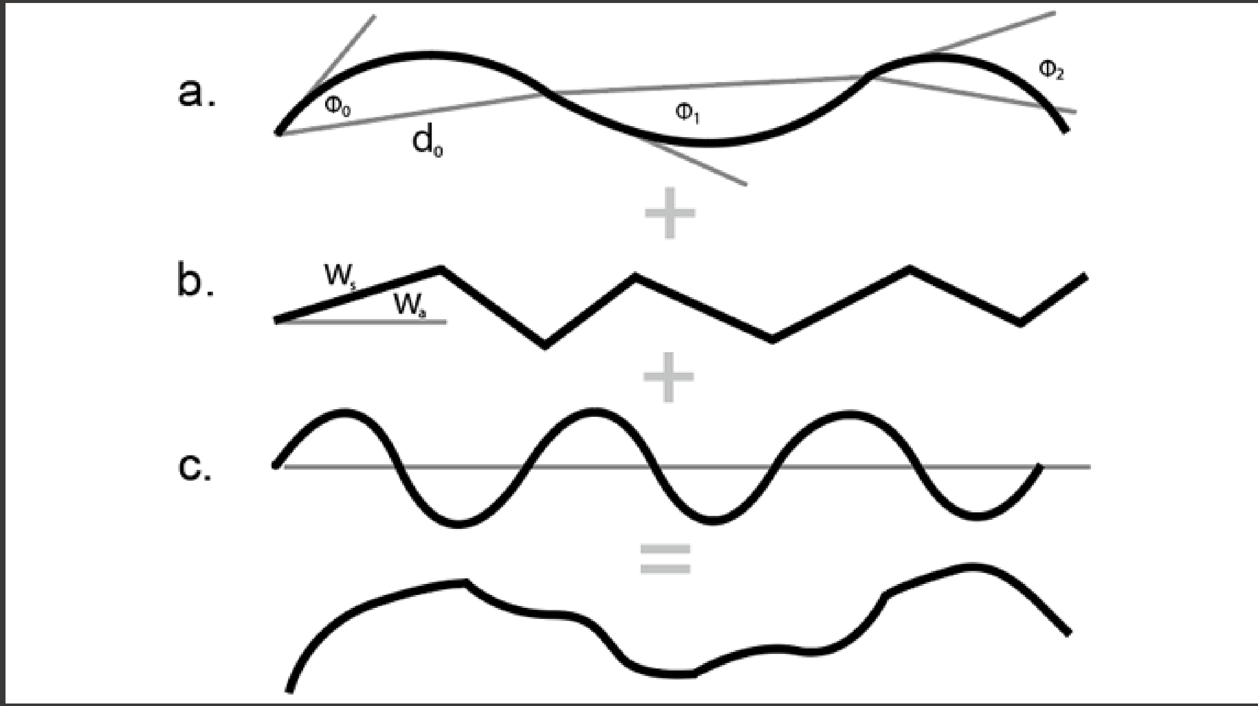
1. A Harmonic Procedural Model for Branch Shape

Branches are not arbitrary curves:

a) Arc sections compactly capture primary directional force changes

b) Linear terms capture angular branch deflections

c) Harmonic terms capture periodic and oscillating motion



$$
\begin{aligned}
B(t+1) &= B(t)\, R_i(t)\, W_i(t)\, H_i(t) && (1)\\
\text{where} & \\
R_i &= \phi_k, \text{for } k, s_k < t < s_{k+1} && (2)\\
W_i &= \begin{cases} w_a, & t \mod w_s \\ 0, & \text{otherwise} \end{cases} && (3)\\
H_i &= sin(t * f_i) * a_i && (4)
\end{aligned}
$$

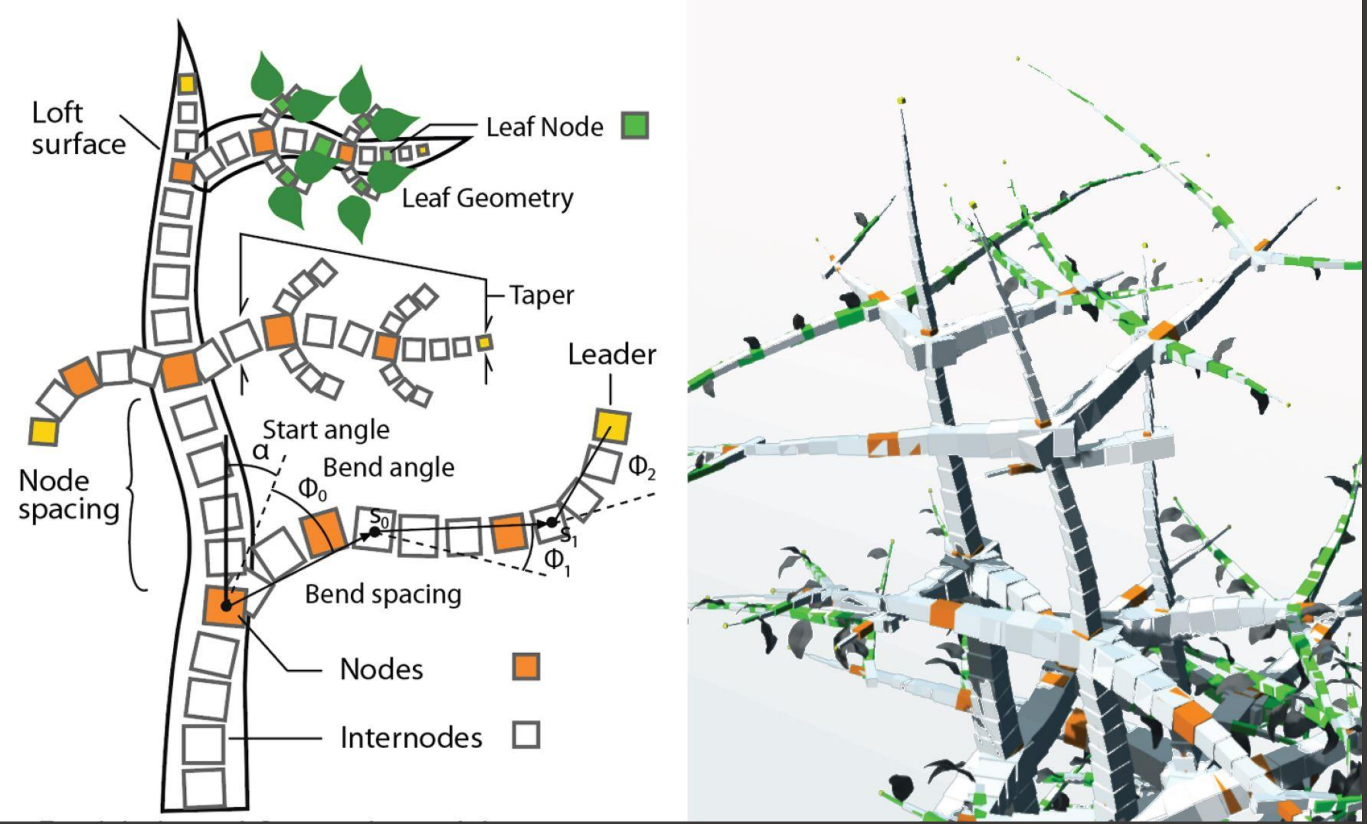
Algorithm Design

2. Particle-based Semantic Model for Tree Growth

For branch trajectories, particle systems offer the most flexibility.

Leader particles change direction based on the harmonic model terms.

Node semantics determines implicit tree hierarchy. New branch particles being at Nodes. Trailing particles are internodes.



Algorithm Design

3. Particle-to-Mesh Model for Surfacing

Particles must be converted to branch surfaces and leaves.
Identify particle chains that define each sub-branch.

Cross-sectional Lofting - Produces smooth, tapered, mesh geometry. Suitable for bump or displacement maps.

Geometry Instancing - Particles specify the orientation of tree parts. Replaced with GPU geometry instances.

Results

COMPLEX BRANCHES

Growth-based Harmonic model can capture complex branch shapes.

Results

INDIVIDUAL TREES

These are not instances.

Each is a uniquely generated individual tree. Individuals are expressed as random variations in branch directionality.

Results

SPECIES VARIATION

The constrained model space (ie. branches are not arbitrary curves) helps to limit the latent parameter space to more plausible trees. These are 36 trees from randomly generated parameters.

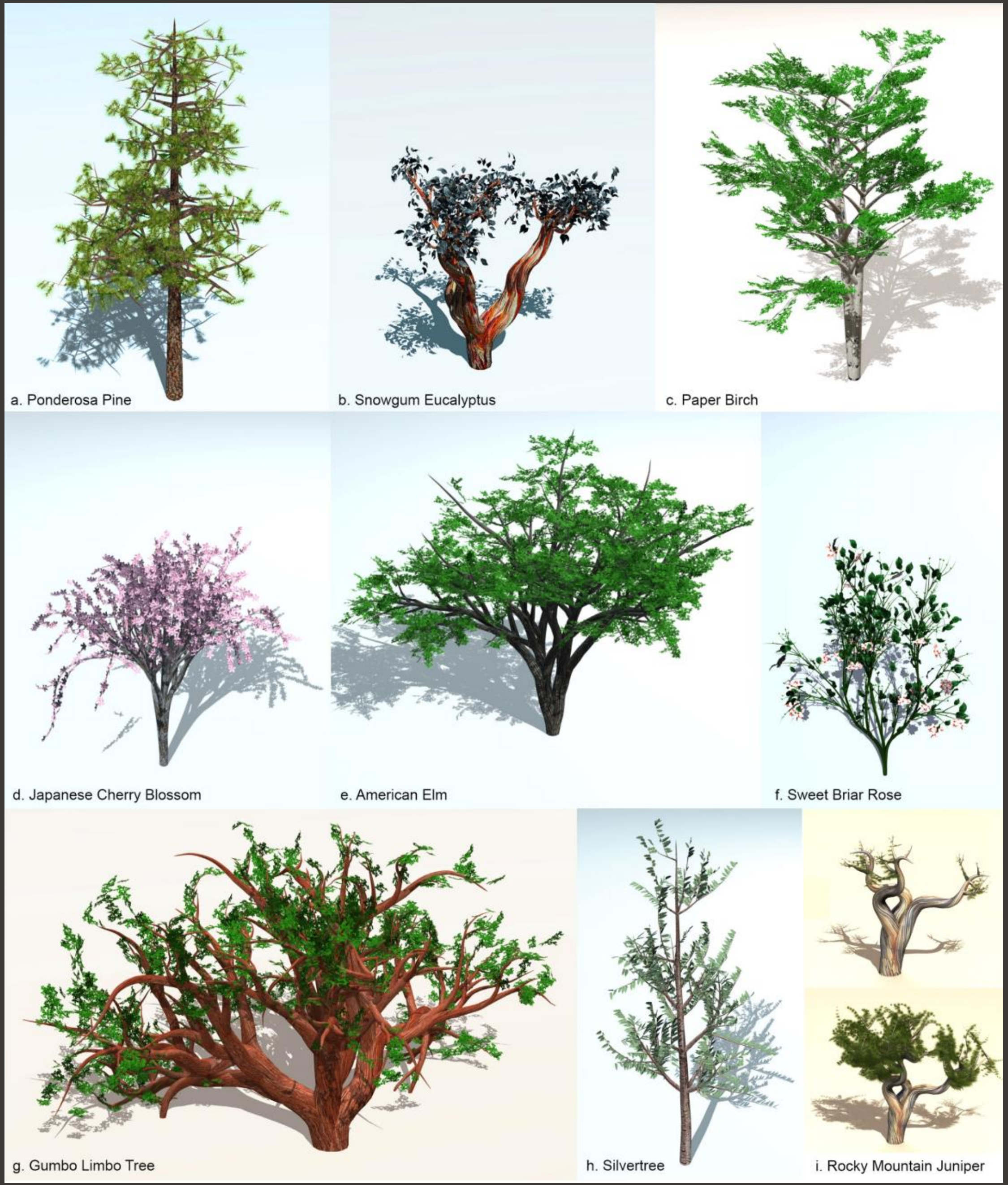
Results

SPECIFIC SPECIES

Specific species can still be modeled if desired.

A user interface could be designed for this model but was not the primary goal.

Parameters were individually tuned,
and 3D leaf parts & textures were provided.
These parts might also be procedurally generated in the future.



Results

EFFICIENT RENDERING

Loft meshes with bump/displacement and leaf parts with geometry instancing provide an efficient method for rendering on modern GPU hardware.

Input Model: 100 params, 120k for textures
Output Geometry: 1,773,852 triangles, 2044 inst.

TOP:
OpenGL, 15 msec + 11 msec (CSMs), 38 fps

BOTTOM:
NV OptiX, 860 msec/sample, 6.8 secs

Results

Results

Results

Results

Results

Conclusions

- A fully procedural model for natural vegetation, and especially branch shape.
- With.. efficient, GPU-based rendering by lofting and geometry instancing.
- Without.. massive data or computation, and without manually defined 3D spatial curves.
- Future:
  This model is a compact parametric latent space well suited to experiments in machine learning.

Exporting with Instancing

Are there any 3D model formats that support geometry instances?

Blender - Instancing Export & Import				Instancing	Instances	
Description	**File Format**	**File Size**	**Binary**	**in File Spec**	**Exported**	**Notes**
Original mesh, Wavefront obj	.obj	8	no	no	no	
Basic meshes, Standford ply	.ply	3	mixed	no	no	
Blender arrays	.blend	11	**yes**	**yes**	no	array instances do not export instanced
Blender instances (geometry nodes)	.blend	11	**yes**	**yes**	**yes**	instances can be exported
Blender instances (applied)	.blend	1043	**yes**	**yes**	no, flattened	no instances, flattened when applied
Collada	.dae	13	no	**yes**	no, only 1	no blender export / import
Khronos glFT, glb	.glb	2	**yes**	**yes**	no, only 1	no blender export / import
FBX	.fbx	3	**yes**	**yes**	no, only 1	no blender export / import
USD-A	.udsa	20	no	**yes**	**yes**	instancing export, file size unusually large
USD-C	.udsc	6	**yes**	**yes**	**yes**	yes, supports instancing export / import
Unreal 5 - Instancing Import						
Unreal 5 - Native formats FBX and OBJ	.fbx / .obj		mixed	**yes**	no	no instancing import
Unreal 5 - USD-C Importer (Beta, v1)	.udsc		**yes**	**yes**	**yes**	import seems to work correctly
Unreal 5 - Save Project	.uasset (.uproject)	3 (10)	**yes**		**yes**	unreal saves instances as actors (heavy)

Universal Scene Description

USD-A/C was the only format used by common modeling / gaming software that has good support for geometry instances.

THANK YOU!

A Procedural Model for
Diverse Tree Species

Rama Karl Hoetzlein