IPT & EGVE Workshop (2005)
R. Blach, E. Kjems (Editors)

Modeling of Trees with 3D Gestures and Growth Simulation

N. Murakami, K. Onishi, Y. Kitamura and F. Kishino

Graduate School of Information Science and Technology, Osaka University, Japan

Abstract

We propose a modeling system that enables users to create tree models with 3D gesture input. It generates tree models by using growth simulation based on the trunk or silhouette shapes of trees given by user gestures. The system carefully addresses the fragile balance and tradeoff between the freedom of user interaction and the autonomy of tree growth. Users intuitively and easily create tree models that have the exact features of branching structures or the silhouette shape of trees according to user intentions and imagination.

Categories and Subject Descriptors (according to ACM CCS): I.3.6 [Computer Graphics]: Methodology and Techniques - Interaction Techniques

1. Introduction

Bonsai, a dwarf tree, or a type of gardening, is attracting a great deal of attention because of the earth's environmental problems. Even in a virtual space, people try to cultivate plants or trees to simulate environmental assessments or education. One of the most promising methods to interactively generate models of trees is a sophisticated graphical user interface (GUI) in which a user draws or paints the shape of trees as if on paper. Here, there seem to be two major approaches. One generates tree models based on the shape of the trunk and branches, and the other is based on tree silhouette. The difference between these two methods reflects the cognitive model that users imagine firstly. For example, a user tends to employ the former when interested in the shape of a particular tree. On the other hand, the latter is generally used when interested in the global configuration of a virtual environment.

Much literature has been devoted to generating realistic tree models based on unique ideas [GGC*97, PL90, PMKL01, LD99, BPF*03]. Almost all of these ideas use procedural algorithms organized by procedural rules and/or numerical parameters; however, for ordinary users, they are not so intuitive. In addition, their branch structures depend on given parameters as initial conditions and production rules defined heuristically beforehand. Therefore, it is difficult for ordinary users to generate the shape of branches that completely correspond to imagination. Moreover, existing methods tend to rely on conventional 2D GUIs; however, this hampers the generation or interaction with trees in a 3D environment.

In this paper, we propose a modeling system that enables users to create the shape of tree models with 3D gesture input, as shown in Figure 1. Here we carefully address the fragile balance and tradeoff between the freedom of user interaction and the autonomy of tree growth by using an interactive L-system [OHKK03] as the tree's growth engine, enabling users to interactively control tree shape. Users intuitively and easily create tree models that have the exact branching structures or silhouette shape features of trees according to their intentions and imagination.

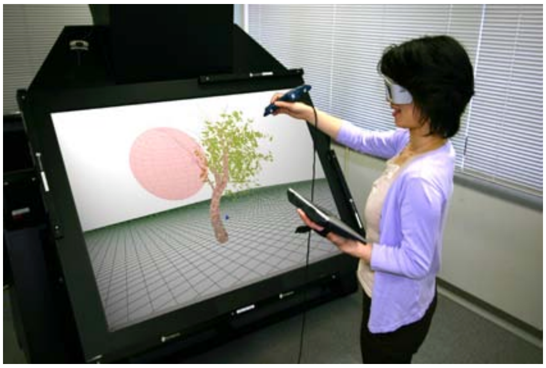


Figure 1: *Interactive modeling system for tree models*

2. Related Work

Much literature has been devoted to generating realistic tree models based on many unique simulation algorithms.

The L-system [PL90], a very famous algorithm for growth simulation, is a string rewriting system that operates on a set of rules. This approach is extended to allow tree models to adapt to environmental effects. *L-studio* [PMKL01] and *Xfrog* [LD99] are mentioned as tree generation modelers using L-system's algorithm. In these systems, by using numerical parameters and graphically-defined functions, users can control the angle and length of branch growth, the shape of leaves, etc. *Xfrog* generates tree models based on tree model components assembled hierarchically in a graphical user interface. This component consists of tree elements (leaves and a trunk) and an arrangement type. This allows users to change the geometric shape of tree models by changing the numerical values. *AMAP* [GGC*97] simulation software is another such system designed to generate realistic tree models by generating tree models using numerical parameters defined by measuring many tree shapes in the real world. Users input these parameters and run the simulation to produce the desired geometry of the tree model. Tree models from these systems are very realistic. But the silhouette of the entire tree and the imagined branching structure are not always generated, because the shape of the tree model depends on production rules defined beforehand and parameters as initial conditions. Also, because the production rules are heuristically defined, it is difficult to change them according to intentions for tree shape.

Another system, *ilsa* [PBPS99], can directly edit the shapes of plant models that are already created. It manipulates the bending of branches by using inverse kinematics technology. A method that generates new tree shapes by editing parts of existing tree shape might be a solution for intuitive tree shape modeling. However, users have to create models beforehand, and they can only manipulate branches. Therefore, it is difficult for users to reflect their intention in the branching structure.

Techniques that model 3D objects in virtual environments by using 3D gesture inputs have been proposed [SPS01, LKG*03]. Such modeling systems are intuitive and allow easy comprehension of the relation between input information and the created shape of objects. Therefore, they allow users to generate and modify the shape of objects according to their intention. However, it is difficult to create the complex shape of objects such as a tree model that has many component parts, because users must create the local shape of all leaves and branches. A system that models 3D tree models from 2D sketches is also proposed [OI03], but it is not easy to generate tree models that have characteristic branch structure in 3D.

We propose an Interactive L-system that enables users to directly control growth simulation results. The Interactive L-system enables users to control the generated tree models by introducing 3D spatial information to the L-system. In this paper, we control the Interactive L-system by using 3D gesture input to generate the complex shape of tree models according to user intentions.

Table 1: *Examples of Symbols*

Symbol	Order
F	Draw tube & move forward
+	Turn left
−	Turn right
&	Pitch down
^	Pitch up
\	Roll right
/	Roll left
[	Save state, start new branch
]	Restore state, end branch

3. Interactive L-system

In this section, we briefly introduce the Interactive L-system [OHKK03] that allows users to generate, manipulate and edit tree shape model based on the L-system by directly indicating its global information.

3.1. L-system

The L-system makes a string data of symbols, an L-string, by adapting production rules to the initial symbol, the axiom, for generating shapes of tree models. The system runs the modeling process by using this L-string as an instruction group. Examples of symbols are shown in Table 1. Moreover, the L-string is described using turtle geometry [Ad82].

A production rule has a format roughly as shown in equation (1).

$$pred : cond \rightarrow succ \quad (1)$$

pred is the strict predecessor symbols, *succ* is the successor symbols, and *cond* is the condition. The Lsystem process replaces the agreed symbols of *pred* within the L-string to the symbols of *succ*.

The numerical parameters of symbols included in the L-string are used to generate such complex shapes of tree models as weeping or branch thickness. In addition, the system controls these shapes by changing the number of times the production rules are adapted.

3.2. Interactive L-system Overview

The Interactive L-system hierarchically constructs the L-string structure to efficiently generate tree models. Figure 2 shows the L-string structure in the Interactive L-system.

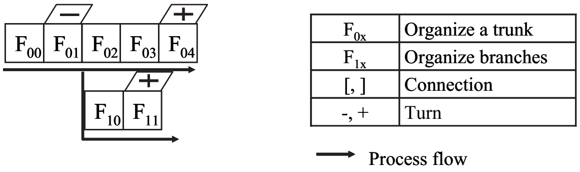


Figure 2: *Structure of L-string in Interactive L-system*

Table 2: *Grouping of symbols.*

Group	Symbols
Shape	F, f
Transformation	$+, -, \&, \wedge, /, \backslash$
Structure	[,]

The structure is constructed by classifying symbols of the L-string into three groups as shown in Table 2. Symbols classified into the "Shape" group are defined as geometric shapes of the model. Symbols classified into the "Transformation" group rotate these geometric shapes. Symbols classified into the "Structure" group are the structure of trees.

Attributes indicating the global and spatial information possessed by a part of the tree model are added into a symbol in the "Shape" group to interactively generate and edit the tree models. One of these attributes is the positional information of each symbol, $\vec{V}(x,y,z)$, defined by the 3D position of each symbol. The other is the relative rotation angle $(\vec{H},\vec{L},\vec{U})$, which is the angle of coordination of each symbol based on global coordination. Moreover, attribute parameter "**M**" that defines the variation of angle of each symbol is added.

Therefore, introducing angle and spatial information obtained from direct 3D inputs to production rules, the Interactive L-system allows users to interactively control the tree models with direct manipulations.

4. Modeling of Trees with Gestures and Growth Simulation

To generate the shape of tree models according to user images, we use hand gesture to determine the shape of tree models. Here, the tree model is generated based on two different concepts, as shown in Figure 3. One is "trunk-based modeling" that generates a tree model by defining the trunk shape with hand gestures. The other is "silhouette-based modeling" that generates a tree model with a silhouette defined by hand gestures. These two concepts are sometimes used individually, however, they can be used in combination according to the situation of modeling.

4.1. Trunk-Based Modeling

The trunk and main branches are components that define the branching structure and the entire shape of the tree model. The shape of such tree model components as trunk and branches is given by gestures. Then, the complex shape of tree models is created by using the Interactive L-system based on the given parts. To achieve this, we propose a method that translates the path of hand gestures to the simulation data (L-string) of the L-system. In the method, hand gesture paths are captured as point sets, and then the symbols that define trunk shape according to the paths are generated from the point sets. Trunk shapes are also corrected to avoid trunk collisions. Details are described below.

4.1.1. Acquisition of Point Sets

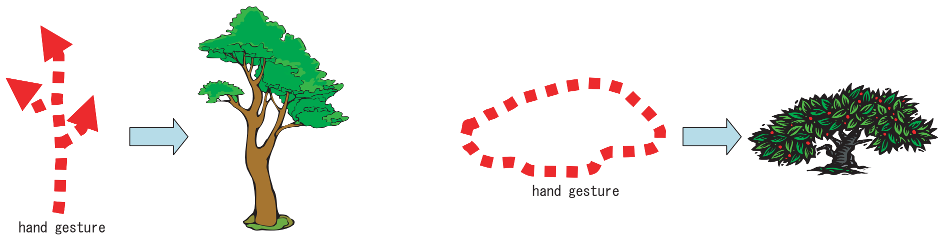
The trunk grows in the same direction as the stroke of a user's hand. Therefore, a user has a 3D tracker on his/her hand and move it as if drawing a line. The path of the hand gesture is captured as a set of 3D position information $\vec{V}(x,y,z)$ at even intervals of time/space, as shown in Figure 4(a).

4.1.2. Analysis of Point Sets

In the Interactive L-system, the L-string symbols have some attributes. As shown in Figure 4(b), the symbol and attribute values of the L-string are determined based on the captured 3D position information. In the Interactive L-system, trunk L-strings consist of two groups of symbols, i.e., Transformation and Shape. The symbols of the "Shape" group denote the form of any tree part. The symbols of the "Transformation" group denote the angle of rotation between the growth directions. Symbol "F" is defined as the form of part of the trunk. This symbol has some attributes, including 3D position information $\vec{V}(x,y,z)$, angle of relative rotation $(\vec{H},\vec{L},\vec{U})$, and a variation of angle **M**. The values of these attributes are given from the captured 3D position information. Symbol "F" is defined as a part of a trunk/branch. The length of the form given from symbol "F_n" is the distance for two continuous points $(\vec{V}_{n+1},\vec{V}_n)$. The 3D position information of symbol "F_n" is $\vec{V}_n$. And $\vec{H}_n$ is the direction in which the trunk is growing, defined as the vector from $\vec{V}_n$ to $\vec{V}_{n+1}$, as shown in equation (2):

$$\vec{H}_n = \vec{V}_{n+1} - \vec{V}_n \qquad (2)$$

$\vec{L}$ and $\vec{U}$ are defined as arbitrary vectors that exist on the plane whose normal vector is $\vec{H}$. But $\vec{L}$ is determined by rotating $\vec{U}$ 90 degree counterclockwise on the plane. **M** is derived from the "Transformation" symbols group determined by the method described below. Finally, these calculations are made with both continuous points of the captured 3D position information.



(a) Trunk-Based Modeling (b) Silhouette-Based Modeling

Figure 3: *Concepts of modeling a tree by hand gestures*

4.1.3. Generation of L-string

Here, we explain the determination of the "Transformation" group symbol, which is another L-string component.

The determination of the"Transformation" group symbol is based on the calculated attribute values about the angle of the relative rotation of a trunk. And the L-string of a trunk is generated from the created "Transformation" group symbols and "Shape" group symbols, as shown in Figure 4(c). The "Transformation" group symbols exist among the "Shape" group symbols. The "Transformation" symbol is determined from attribute values of the neighboring symbols of the "Shape" group. The symbol of the "Shape" group calculates the attribute values of the next "Shape" symbol by arbitrary rotation on the axes of its local coordinate system $(\vec{H}, \vec{L}, \vec{U})$. Also it may rotate two or more times on the same axis. Therefore, the degree of the angle and the order of axes rotation are underspecified by calculating only with vectors that define the angle of the relative rotation of trunk parts. However, generating the L-string in real-time is required to interactively create tree models. To simplify the derivation of "Shape" symbols, it is assumed that the attribute values about the angle of the relative rotation of the next symbol of the "Shape" group is calculated by rotating the axes in the order of $\vec{U}, \vec{L}, \vec{H}$ in the local coordinate system of the present symbol. Consequently, the relationship between the continuous "Shape" group symbols is defined as three symbols of the "Transformation" group.

Our method translates trunk shape given by gestures to the L-string. When the trunk is generated by gesture inputs, the 3D positional information input later is translated to an L-string as branches of an already generated trunk. Therefore, the method enables users to edit the tree models by the Interactive L-system without considering whether the trunk is given by gestures.

4.1.4. Generation of Growth Points

A trunk that defines branching structure is created by using gesture inputs. Branches and leaves and so on are generated by growth simulation based on the trunk. To achieve this, the growth points of branches are created on the trunk's generated L-string. Symbol "*A*" as a growth point is generated by L-system simulation with production rules of the growth points, as shown in Figure 5. The attributes (position, angle of shape, and diameter) of symbol "A_1" have the same values as the attributes of symbol "F_{01}" that draw part of a trunk as a root of a generating branch based on symbol "A_1". Branches are generated by the Interactive L-system with the attribute values of the growth points as initial conditions.

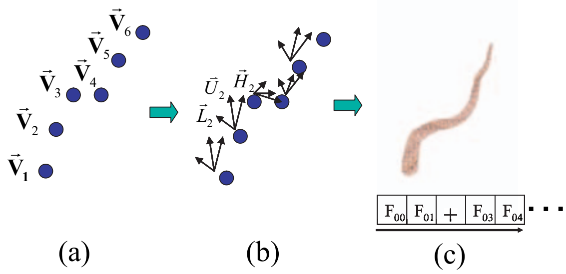


Figure 4: *Translation process from positional information to L-string. (a) Positional information. (b) Angles of relative rotation (H, L, U). (c) L-string.*

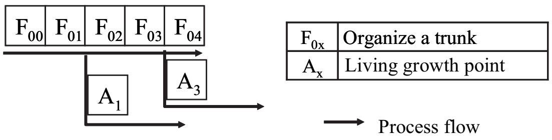


Figure 5: *Placement of growth points*

4.1.5. Collision Avoidance

3D gesture inputs enable users to freely and easily create trunk shapes. Therefore, users may create an "unnatural" tree shape that cannot exist botanically in the real world. As a typical example, a trunk pierces through other trunks or branches. Trunk shapes created by users are corrected to

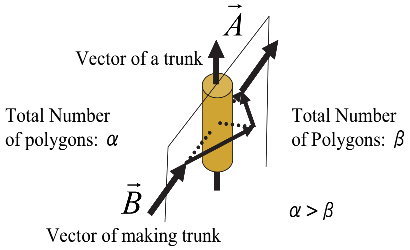


Figure 6: *Collision avoidance*

avoid trunk collisions. Since a real tree's trunk shape is often symmetrical, the bypass direction of the trunk is based on the shape of already created trunks and branches, as described below. Part of a trunk model is assumed to be a cylinder whose vectors are defined in Figure 6.

First, when a trunk collision occurs, the distance of vector ($\vec{A}$) of the already created trunk to vector ($\vec{B}$) of the "growing" trunk is calculated. If the distance is over a particular threshold, the bypass direction goes from $\vec{A}$ to $\vec{B}$. If the distance is equal to or smaller than a threshold, bypass direction is determined by the following method. The threshold is defined by the kind of tree. As shown in Figure 6, a plane, which bisects the space, contains both $\vec{A}$ and a certain vector generated by translating $\vec{B}$ until $\vec{B}$ intersects with $\vec{A}$. The number of polygons created in the tree modes in each space is calculated. To generate tree models whose branches grow uniformly, bypass is generated in spaces whose number of polygons is fewer than the other.

4.2. Silhouette-Based Modeling

Next, a method to generate a tree model based on silhouette shape is described.

First, silhouette shape is determined in two different ways. One simply determines the silhouette shape as a sphere, and the other determines the shape as a supertoroid, which is a special form of superquadrics. Users freely generate the region's shape as superquadrics by 3D gesture inputs. Therefore, the silhouette of the tree model is generated according to user imagination. Superquadrics formulas have some parameters. By adjusting them, a large variety of 3D shape can be generated easily. Two bounding contours of the silhouette are generated by hand gestures captured in the same way as generating trunks and input as the left and right sides of the region's silhouette. The feature points of the contour paths are given from sets of the 3D positions by Vector Tracer. The axes of the generating supertoroids are derived from those continuous feature points. Supertoroids as silhouette shape are generated based on those axes and feature points with a set of equations (3).

$$\begin{cases} x(u,v) &= a_x(\alpha_x + \cos^n u)\cos^e v + b_x \\ y(u,v) &= a_y \sin^n u + \alpha_y \\ z(u,v) &= a_z(\alpha_z + \cos^n u)\sin^e v + b_z \end{cases} \qquad (3)$$

$$-\pi \le u \le \pi, -\frac{\pi}{2} \le v \le \frac{\pi}{2}.$$

Two parameters (b_x, b_z) are introduced into the supertoroid formulas to reflect the slopes of those axes in the supertoroids.

Finally, tree models in this region are generated by using the Interactive L-system [OHKK03]. The growth rules of the Interactive L-system are applied only to the part of the symbols at the L-string, which is in the region. If no symbols exist in that region, the axiom at an arbitrary point in the region is applied to the growth rules of the Interactive L-system. Therefore, with the L-system using the generated region of interest, users can directly generate and edit the shape of tree models, whose silhouette reflects user images.

5. The System

5.1. System Summary

Figure 1 shows the interface and a screenshot of our system. The path of the hand gesture is captured as a set of 3D positions obtained in regular time intervals from a stylus in the user's hand. A 3D tracker is attached to the stylus and the region of interest (ROI) is given by the motion of the stylus. There are two ways to define ROI. One simply defines its shape as a sphere, and the position and the radius of the sphere are given by the stylus. The other defines its shape as a supertoroid, as described in 4.2, whose shape is determined from the trajectory of the stylus. After the shape of the ROI is defined, its position and size can be manipulated by the stylus. After the position and size of the ROI are determined, the system adapts the production rule to the parts of the model in the ROI with the Interactive L-system. Users can interactively observe the generated tree models by using stereoscopic LCD shatter glasses. One of the most important features of the system is that it allows users to see the actual L-strings of any part of the tree model. Figure 7 shows an example in which L-strings of branches are superposed on the tree models. Here, the symbols of the parts of the tree model in a ROI are displayed. Black strings show symbols of the "Shape" group defined as geometric shapes of the model, and red strings show symbols of the "Transformation" group that rotate the geometric shapes, as explained in 4.1.3.

Finally, the system is implemented on a personal computer (Xeon 2GHz, Mem 2GB, 3Dlabs WildcatII5110(TM),Windows 2000).

5.2. Example of Interactions

In this section, we describe the creation process of tree models. In our system, the hand gesture path is captured as a data

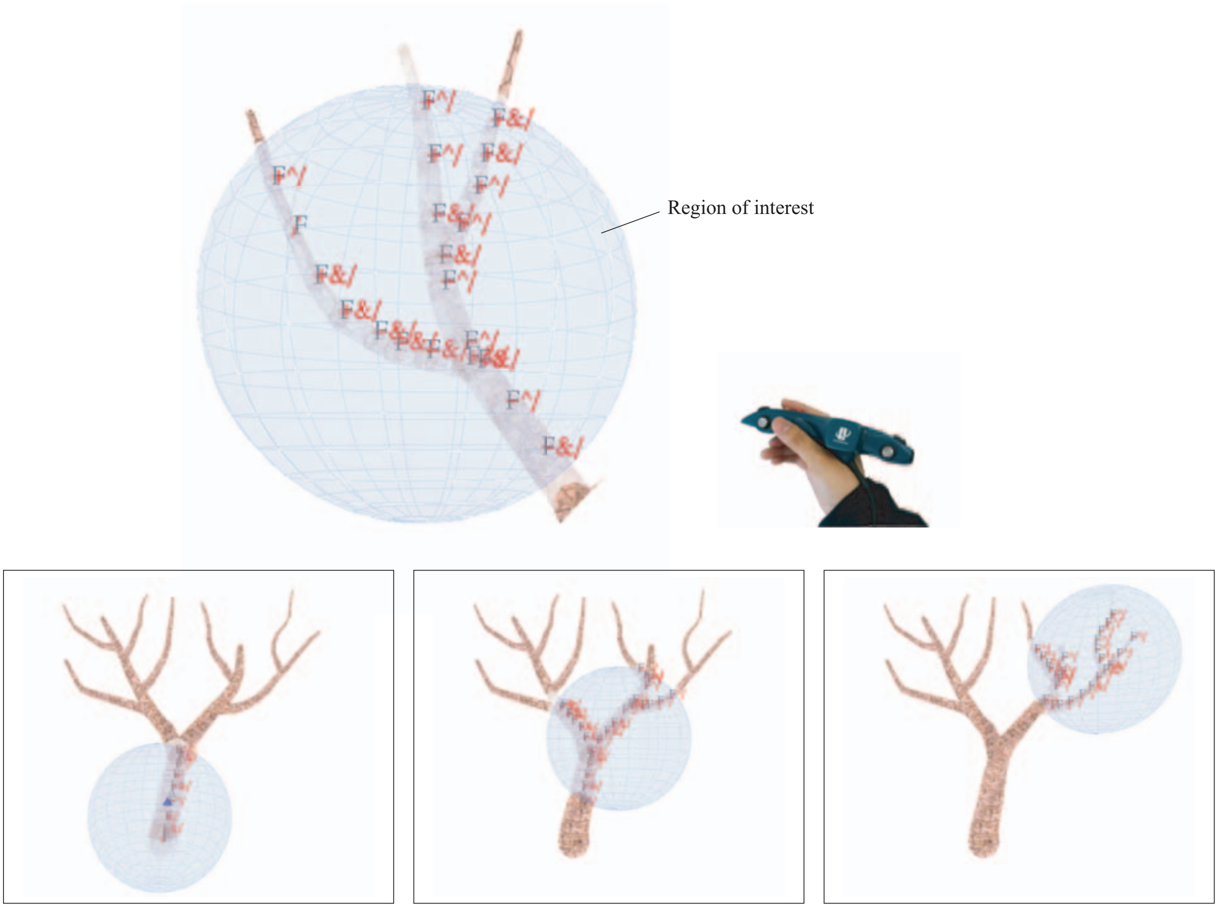


Figure 7: *L-string superposed on tree model*

set of 3D positions obtained from the stylus in the users' hand. As shown in Figure 8(a), L-string is generated based on the data set and our proposed method, and the system shows the trunk shape drawn based on L-string. As visual feedback, this interaction starts when user gesture input begins to enable users to recognize their gesture inputs. Also, collision detection of trunks to correct the data of the path is carried out immediately when the 3D position is obtained from the stylus.

When the shape of a trunk is generated from gesture inputs, the system requires users to designate the point on the created trunk where trunks are jointed, which is difficult to do correctly because our system uses 3D direct manipulation, and the displaced place of tree models and the gesture input place are separated. Therefore, our system calculates and shows a joint point on the created trunk closest to the pointer controlled by users. Trunk diameter increases in inverse proportion to the velocity of the stroke that generates the trunk.

Branches are generated by the Interactive L-system. Users determine size by controlling the region of interest that they themselves created freely. Branches are generated from the growth points on trunks in the region by adapting production rules for branches, as shown in Figure 8(b). Therefore, our system enables users to easily generate the shape of a tree model that has the silhouette and trunk shapes that reflect their demands.

Users can also directly edit the shape of the generated tree model with the Interactive L-system. The system edits part of the tree model in the region selected by the users, who can obtain various results by selecting the production rules defined beforehand for editing, as shown in Figure 8(c). In the production rules, our system defines generating/erasing blossoms and leaves, bending and pruning branches, and the trunk. As shown in Figure 9, tree models are generated by two trunk inputs and added blossoms and leaves. In Figure 10 the trunk is spiral, a shape easily generated by a beginner with this system. As shown in Figure 11, a branching struc-

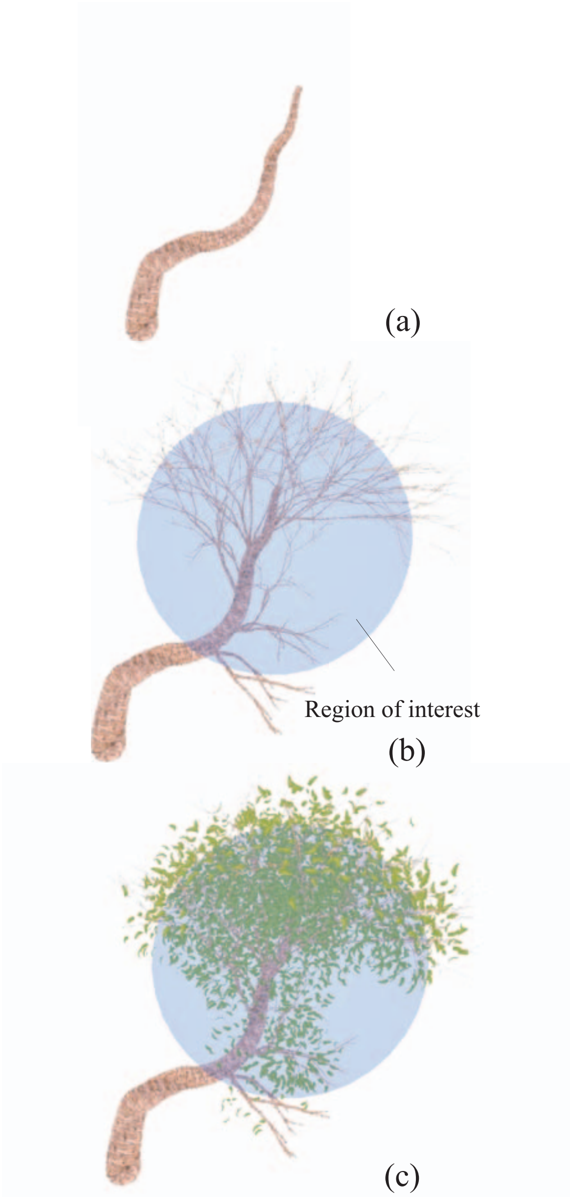


Figure 8: *Example of process to generate a tree model*

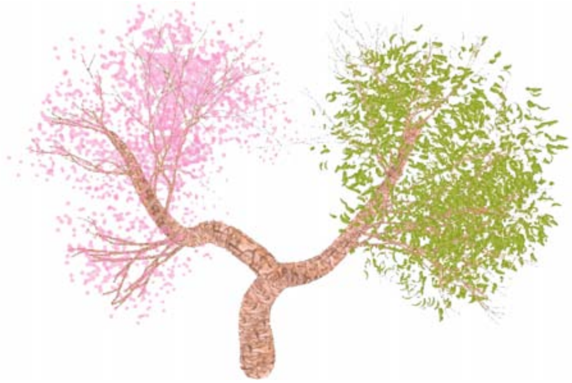


Figure 9: *Tree model generated by two trunk inputs and added blossoms and leaves*

Figure 10: *Tree model with a spiral trunk*

ture is generated from only inputting the region of interest as a silhouette of the entire tree.

6. Conclusions

In this paper, we proposed an interactive system that makes tree models with gesture input and growth simulation. This system enables users to make intuitively complex shapes of imagined tree models. In our method, trunk shapes and tree silhouettes are given by the path of hand gestures, and shapes are translated to growth simulation data. Also, our method corrects "unnatural" trunk shapes made by users to avoid trunk collisions. Using translated data as initial conditions, our method generates tree models by the Interactive L-system, enabling users to interactively control the shape of trees.

As future work, we are planning a method that defines production rules for L-system from shapes given by gesture and diversifies the interaction of generating tree models.

Acknowledgments

This research was supported in part by "The 21st Century Center of Excellence Program" of the Ministry of Education, Culture, Sports, Science and Technology, Japan.

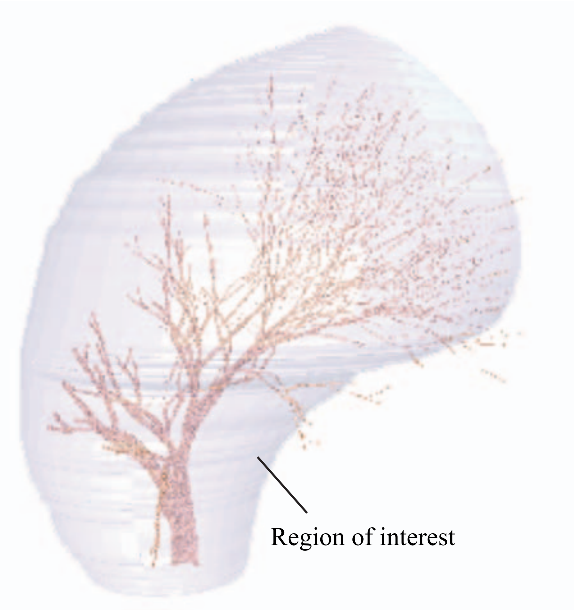


Figure 11: *Tree model generated by a silhouette input with a supertoroid ROI*

References

[Ad82] ABELSON H., DISESSA A.: *Turtle geometry*. MIT Press, Cambridge, 1982.

[BPF*03] BOUDON F., PRUSINKIEWICZ P., FEDERL P., GODIN C., KARWOWSKI R.: Interactive design of bonsai tree models. *Computer Graphics Forum 22*, 3 (2003), 591–599. (Proc. Eurographics'03) `http://diglib.eg.org/EG/CGF/volume22/issue3/cgf_707.html`.

[GGC*97] GORDIN C., GUEDON Y., COSTES E., GORDIN C., CARAGLIO Y.: Measuring and analyzing plants with the amapmod software. In *Plants to ecosystems-Advances in computational life sciences*. CISRO, 1997, pp. 54–84.

[LD99] LINTERMANN B., DEUSSENM O.: Interactive modeling of plants. *IEEE Computer Graphics & Application 19*, 1 (1999), 2–11.

[LKG*03] LLAMAS I., KIM B., GARGUS J., ROSSIGNAC J., SHAW C. D.: Twister: A space-warp operator for the two-handed editing of 3d shapes. In *Proc. of SIGGRAPH '03* (2003), pp. 663–668.

[OHKK03] ONISHI K., HASUIKE S., KITAMURA Y., KISHINO F.: Interactive modeling of trees by using growth simulation. In *Proc. of ACM VRST* (2003), pp. 66–72.

[OI03] OKABE M., IGARASHI T.: 3D modeling of trees from freehand sketches. In *SIGGRAPH '03 on Sketches & applications* (2003), p. 1.

[PBPS99] POWER J. L., BRUSH A. J. B., PRUSINKIEWICZ P., SALESIN D. H.: Interactive arrangement of botanical l-system models. In *Proc. of SI3D* (1999), pp. 175–182.

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

[PMKL01] PRUSINKIEWICZ P., MUNDERMANN L., KARWOWSKI R., LANE B.: The use of positional information in the modeling of plants. In *Proc. of SIGGRAPH '01* (2001), pp. 289–300.

[SPS01] SCHKOLNE S., PRUETT M., SCHRODER P.: Surface drawing: Creating organic 3d shapes with the hand and tangible tools. In *Proc. of CHI '01* (2001), pp. 261–268.