

# Efficient Procedural Modeling of Trees Based on Interactive Growth Volume Control

**Jinmo Kim<sup>1</sup>**

<sup>1</sup> Research Institute for Image & Cultural Content, Dongguk University  
Seoul, South Korea  
[e-mail: mythsjin@dongguk.edu]  
\*Corresponding author: Jinmo Kim

*Received April 30, 2013; revised August 16, 2013; accepted September 6, 2013; published September 30, 2013*

---

## **Abstract**

The present study proposes efficient procedural modeling methods for enabling the growth and creation of various trees with minimal user control. Growth volume algorithms are utilized in order to easily and effectively calculate many parameters that determine tree growth, including branch propagation. Procedural methods are designed so that users' interactive control structures can be applied to these algorithms to create unique tree models efficiently. First, through a two-line-based interactive growth volume control method, the growth information that determines the overall shape of the tree is intuitively adjusted. Thereafter, independent branch control methods designed to control individual branches are added to the growth deformation in order to enable the growth of unique trees. Whether the growth processes of desired trees can be easily and intuitively controlled by the proposed method is verified through experiments. Methods that can apply the proposed methods are also verified.

---

**Keywords:** Procedural modeling of trees, interactive control, growth volume, growth simulation

## 1. Introduction

To construct natural backgrounds, the creation of plants, including trees, is an important element. However, the creation of natural tree models requires special modeling methods because of the complexity of tree structures. Tree modeling methods are divided into rule-based procedural modeling methods and methods to reconstruct tree geometry information from input images or three-dimensional (3D) scan data. The reconstruction methods enable the creation of reliable tree models using only the images provided as inputs with minimum user control. However, to obtain good resultant tree models, high-quality images or scanned information taken under limited and defined conditions is necessary. Maintaining these prerequisites at all times is not easy, and outdoor terrain that can be made through these conditions is extremely limited in size and number of trees. Procedural modeling relieves these limitations. Nevertheless, to obtain botanically plausible results for diverse trees, artistic techniques are required.

The present paper proposes modeling methods that enable the independent and interactive control of branch growth based on procedural modeling methods using growth volume. The basic growth model involves using a growth volume algorithm which automatically calculates tree growth information, such as the length and width of branches [1]. Based on basic tree growth models, efficient procedural modeling methods that enable the intuitive creation of diverse trees with only simple control inputs from the user are proposed. Existing growth volume-based modeling methods are currently limited to affecting only the overall shape of a tree. In the present study, two novel methods are proposed to improve the existing growth volume-based modeling methods and to more easily and effectively create diverse trees. First, a method designed to interactively deal with the creation and addition of growth volumes through two-line inputs provides a function used to easily control tree growth processes through a simple operation by the user. Next, an independent branch growth method permitting individual control of the growth of branches selected by the user allows for the effective and intuitive creation of unique trees. Existing growth volume algorithms focus on the growth of multiple trees rather than a single tree and therefore do not provide the function of modifying growth volume once they are set. The proposed method complements this limitation and provides the function of generating and editing tree models quickly and efficiently. It can be differentiated from existing three-dimensional modeling tools and will likely become a method of presenting new user interface functions that are appropriate for a tree model. It will be usable as a method to alleviate the limitations that image-based tree modeling methods currently have. Whether or not diverse and unique trees can be easily and effectively created through the aforementioned methods is verified through experimentation.

A review of related studies is presented in Section 2, and the interactive tree modeling methods proposed in the present study are explained in Section 3. Diverse tree generation processes are checked and analyzed through experiments described in Section 4. Conclusions and the direction of future studies are presented in Section 5.

## 2. Related Work

Tree modeling has complex structures and various forms so it is impossible to express them fully using general modeling methods. Lindenmayer [2] proposed the L-system as a method of expressing these natural objects, and various tree modeling methods using this system have

been studied. The L-system divides the 3D tree growth forms into the length of branches, growth angles and width, etc., and assigns a symbolic character to each. Then, the form of the trees is determined in the next step by substituting the characters with another character string in one condition using the substitution rule [3,4]. Consequently, complex objects are created while continuing to substitute only simple individual components using this regional rule. This process of creating, one big phenomenon by analyzing regional phenomena is ideally suited to the method of expressing plants [3,4,5].

Based on this method of rule-based modeling, various attempts to create tree modeling methods have been made. Honda [6] configured trees with parameter characteristics such as the recursive structure of the generation of a specified branch, branch rotation angle, or the ratio in the reflective step of a tree structure. This perspective is used to form an overall appearance naturally by recursive reproduction based on the defined rules. In contrast, Ulam [7] configured trees by performing a self-organizing process that forms a branch pattern based on the competition of each individual unit in the space where the branch survives. He also granted a different destiny to each bud at the end of a branch, letting it generate a new branch each time, and made this generation depend only upon the results of competition in a given space for the branch propagation, not just on the results of the repetitive and reflexive processes [8]. Based on this, Prusinkiewicz [9] proposed a method of regional control of branch geometry involving the definition of rules for competition between buds and branches in a single space and internal competition and modeling through self-organization. Borchert and Slade [10] analyzed the key difference between the recursive generative structure proposed by Honda and the configuration of trees through self-organizing algorithm proposed by Ulam. They observed that the repetitive branch pattern has a constant size and the number of branch nodes increases geometrically as the depth of a reflexive structure increases. In addition, the reflexive tree generating structure or tree configuration method through self-organizing algorithm was developed in order to produce the methods to be used for image synthesis [9]. Studies have been carried out on controlling the balance between the leveled generation process and the user's control over a tree structure. Studies to create diverse trees by combining rule-based modeling methods using self-organizing algorithm with users' sketch inputs have also been conducted [11].

In addition, studies that enable more interactive control in the process of tree creation were also conducted, and the Xfrog software is an example of the results of these studies. This software is a system in which parameters of plants are controlled to create models. This enables the creation of diverse plants, but it can only be easily utilized by skilled users and thus cannot be widely used [12,13]. Studies to construct models with user control have also been actively conducted recently such as studies to create desired trees based on sketches [14], and studies to calculate 3D information from input images and reconstruct 3D trees from the 3D information [15,16,17].

These diverse studies related to tree and plant modeling have been utilized in establishing virtual ecosystems. Based on the studies mentioned above, studies of interactive, diverse, and efficient modeling for virtual plants and plant ecosystems have been conducted [18,19]. In fact, to establish realistic ecosystems, it is important to consider external environmental elements such as light, physical obstacles, and the presence of other plants and studies to efficiently calculate, apply and render these external factors have also been conducted [20].

Furthermore, studies have been conducted on the process of efficiently modeling trees; on adjusting the large numbers of vertices constituting trees, and replacing the vertices with leaf billboards to enhance efficiency in order to efficiently render virtual environments that consist of many trees and while processing lighting and shadows utilizing shaders in order to enhance

the reality of the leaf billboards [21] and on applying lighting processing suitable for tree structures so that many trees can be efficiently treated with high reality in real-time systems [22].

### 3. Design of Interactive Tree Modeling

In this study, procedural methods for more efficient modeling of diverse unique trees through interactive control are proposed. These methods are different from those that create tree models in detail using graphic tools, such as 3Ds Max. In these methods, users provide only the basic information of the desired tree shape, and the desired tree models are procedurally created through growth rules based on the basic information. To this end, a growth model is defined as a model in which branches propagate to make tree shapes based on the structures and growth parameters of trees. Growth volume methods are then used for intuitive growth control [1]. Growth volume components correspond to the user's input information so that the shape of the desired tree can be easily changed. In addition, unique tree growth is enabled by allowing for the selective control of branches.

#### 3.1 Growth Model

Tree structures consist of hierarchical relations of branches and have a growth model in which new branches propagate from current branches depending on growth steps. Branches consist of buds and leaves, and growth rules are defined as methods of creating new branches from current buds. The growth model for branch propagation is as follows. When the number of buds from the current branch has been determined, the position ( $p_i$ ) of the buds is determined. Then, the position of the candidate bud ( $p_c$ ) is made from the buds ( $p_i$ ) based on the growth direction ( $\vec{v}_{pr}$ ) of the parent branch and rotation angles ( $\theta_i, \phi_i$ ). Growth space with a certain width ( $r$ ) and length ( $l$ ) is generated from  $p_c$ . Here, growth space is a space that determines the randomness of branch directions within a certain range. Then, the growth direction ( $\vec{v}_{gr}$ ) of the current branch is determined from the arbitrary position in the growth space (Fig. 1).

The growth volume method allows for the easy and intuitive expression of tree shape while automatically calculating many growth parameters. The growth model in this study uses existing growth volume methods in order to simplify the user's input and control processes and effectively generate the desired tree model [1].

When overlapping growth volumes are created, the growth information of the current branch should be considered separately for each case. First, child branches that have propagated from parent branch inherit the growth volume information of the parent branch. If the growth volume of the bud of the current branch and the growth volume of the parent bud are different, the growth volume of the parent bud should take priority. However, when new branches are created from the trunk, the bud of the new branches is affected by the growth volume information of the current bud. For example, if three growth volumes overlap, three sub-trees will grow in one tree through different information.

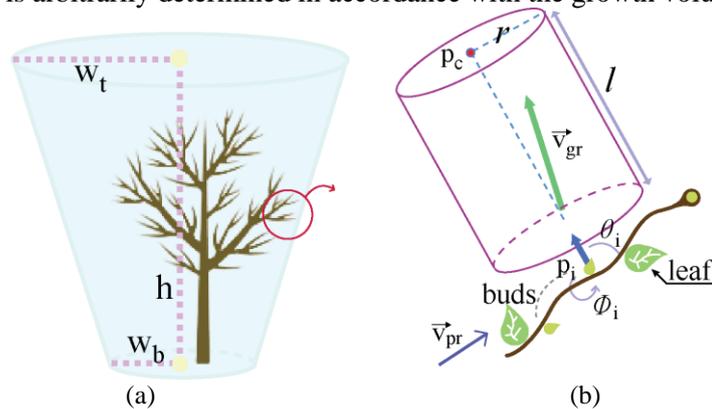
The following is an algorithm in which growth information is determined in accordance with the position of creation of the current bud.

```

if  $p_i \in \{p_{bud} \text{ in trunk}\}$  then
    Set initial values ( $r_{bud}, r_1, l_1, \dots$ ) from the current growth volume
else
    Calculate from growth information of parent branch
    
```

**Algorithm 1.** Method of determining growth information in accordance with bud positions

When a current bud has been created from the trunk, the initial values ( $r_{bud}, r_1, l_1, \dots$ ) of the growth parameters, such as width and height, are calculated from the growth volume to which the bud currently belongs. Growth volume with a cylinder form is comprised of the lengths of the top and bottom surfaces ( $w_t, w_b$ ) and height ( $h$ ) (Fig. 1 (a)). Here,  $r_{bud}$  represents the growth increase and decrease rates ( $w_t / w_b$ ), and  $r_1$  and  $l_1$  are the initial values of growth width and length, respectively, which are initial elements that determine the entire form of the tree. Otherwise, growth information is derived from the information of the parent branch. The number of buds is arbitrarily determined in accordance with the growth volume.



**Fig. 1.** Growth volume-based tree growth model. (a) Growth volume creation, (b) branch structure and growth elements

The growth volume algorithm has the advantage of automatically calculating many parameters that determine branch propagation. Therefore, if this characteristic is applied to interactive structures, a variety of unique trees could be created easily, intuitively, and efficiently with a little user control.

### 3.2 Interactive Control Method

Growth volumes can enable intuitive control of tree growth processes. However, only predefined growth volumes have been utilized to create trees. Therefore, in this study methods for interactive control of growth volumes are provided so that diverse forms of trees can be more easily and effectively created.

Firstly, a two-line-based interactive growth volume control method is proposed. Growth volume is configured in the form of cylinders designed to accept inputs of the diameters of the upper and lower surfaces of cylinders. Cylinder-shaped growth volumes are created by using input devices, such as a mouse or keyboard, to draw two lines that form the diameters of two circles, which are then connected. In this case, if the shapes of the two circles are irregular, the automatic calculation of growth parameters through growth volumes may not progress smoothly. Therefore, based on the growth volume input from the user, growth volume

components such as top or bottom internal angles ( $\theta_t, \theta_b$ ) are internally calculated after being aligned along the top line and bottom line (Fig. 2 (a)). The degree to which the growth volume is sloped is applied to the branch growth as an external force.

If multiple growth volumes are created, they should be consistently arranged. Otherwise, problems may occur in the calculation of the growth information necessary for branch propagation. Therefore, a method is designed to align the top surface of the previous growth volume with the bottom surface of the current growth volume. (1) Make a ray that connects the center of the bottom surface with the center of the top surface of the previous growth volume. (2) Match the center of the bottom surface of the current growth volume with the ray. (3) Separately, from the degree to which the current growth volume is sloped, calculate the movement vector between the top surface of the previous growth volume and the bottom surface of the current growth volume and determine the external force (Fig. 2 (b)). Through this process, different growth information is determined based on the growth volumes of the current buds, resulting in the creation of diverse tree models.

Fig. 2 shows the two-line-based growth volume editing process.

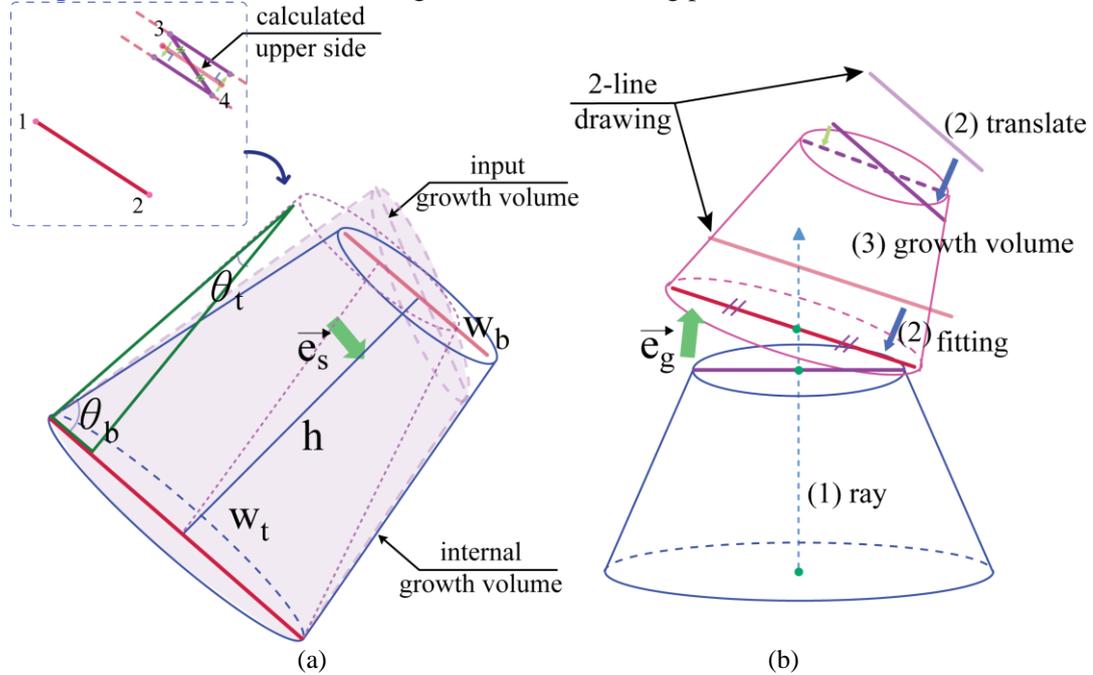


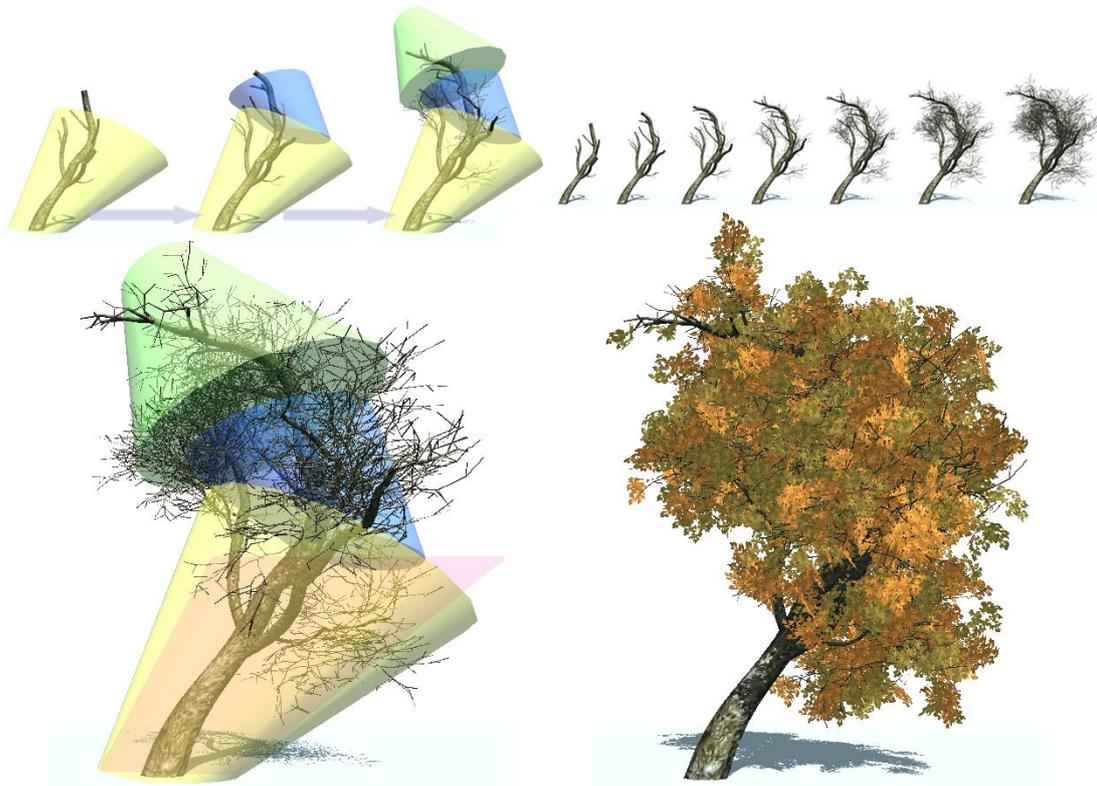
Fig. 2. Interactive growth volume control method. (a) Growth volume component determining method through two-line inputs, (b) multiple growth volume editing method

When a growth volume has been defined, the position( $p_c$ ) of candidate buds, which determines the growth direction of the branch is determined through the application of the growth elements in Fig. 1 and the external elements in Fig. 2. Equation (1) shows this process.

$$\begin{aligned}
 p_c^{apical} &= p_i + \omega(\vec{e}_s + \vec{e}_g), \\
 p_c^{lateral} &= p_i + R(\phi_i)R(\theta_i)\vec{v}_{pr} + \omega(\vec{e}_s + \vec{e}_g), \\
 \omega &= \frac{1}{a_0 + a_1 d_{trunk}}
 \end{aligned} \tag{1}$$

where whether to consider the rotation angle of each candidate bud ( $p_c$ ) depends on whether it is in the position of an apical ( $p_c^{apical}$ ) or lateral ( $p_c^{lateral}$ ) bud; and  $\omega$  is the weighted value of the external force, which is the highest for buds on the trunk and decreases the further buds that are from the trunk.  $\vec{e}_s$  is the external force set according to the inclination of each growth volume, and  $\vec{e}_g$  is the external force generated according to the differences in inclinations between growth volumes. In consideration of such external forces, the position of candidate bud determining growth space is calculated.

**Fig. 3** shows the process of tree growth through the interactive control of growth volumes and demonstrates the ease with which this can be accomplished. In this case, although the upper and lower surfaces of the growth volume input by the user are not parallel (**Fig. 3**, yellow growth volume), the components of the growth volume should be calculated and applied through the above calculation internally (**Fig. 3**, pink area).



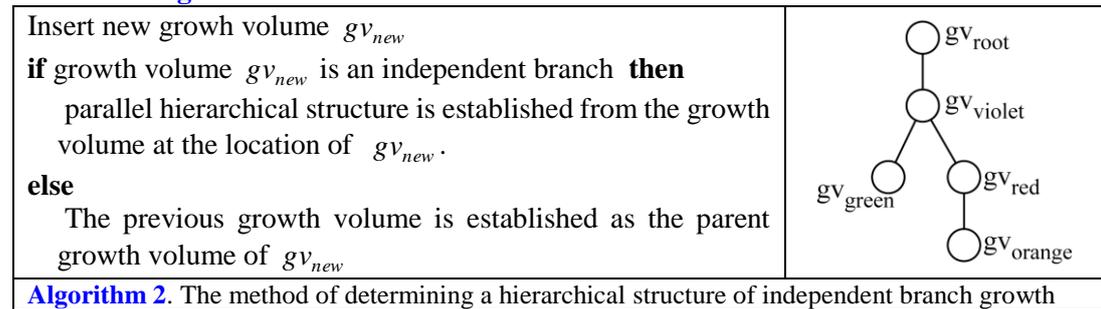
**Fig. 3.** Tree growth process through interactive growth volume control (pink area: calculated internal growth volume)

### 3.3 Independent Branch Growth

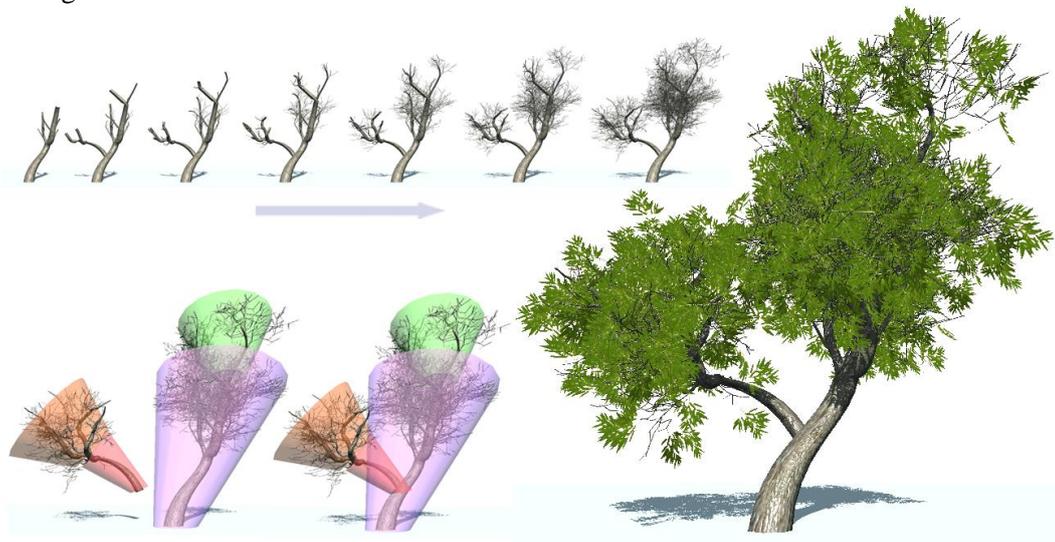
Interactive growth volume control is a method of editing the overall shape and growth process of the tree, but it is limited in terms of diversifying the tree shape. To overcome these limitations, a method is designed to select independent branches instead of the entire tree. The process is as follows.

Create a new growth volume centering on a starting point chosen by the user. In this case, use the interactive control method defined earlier for growth volume editing and addition.

Create branches on the basis of the newly added growth volume and grow the created branches by applying independent parameters that differ from the main growth volume. The following is an algorithm that shows how to determine a hierarchical structure among growth volumes in independent branch growth. If the newly generated growth volume is defined as an independent branch, a parallel hierarchical structure is established from the current location. If it is a general growth volume of overlapped structure, growth is processed using the method defined in Section 3.2, with the growth volume of the previous step as a parent growth volume. The tree structure in the right side of [algorithm 2](#) shows the growth volume hierarchical structure of [Fig. 4](#).



This process enables unique tree growth and creation. [Fig. 4](#) shows the process of creating unique trees through independent branch growth. This process has the advantage of easily creating trees.



**Fig. 4.** Tree growth process through independent branch growth volume control

The growth of diverse, unique trees can be easily and intuitively expressed with minimum user control using the proposed method. This tree editing is possible because tree growth processes can be effectively controlled through growth volumes.

Existing growth volume methods focus on multiple trees and fail to provide the ability to individually edit trees. This study proposes new functions that maintain the advantages of existing studies and may be used in a variety of directions.

## 4. Implementation and Results

For the tree modeling in this study, a proposed modeling system was made using Visual Studio 2008, DirectX SDK without any separate graphic tool, and a PC installed with an Intel Core i5-650 CPU, Geforce GT 320 GPU. In Section 4, examples of diverse tree growth using the proposed method will be presented along with application methods.

**Fig. 5** shows the results of tree creation using the interactive control method under the proposed procedural modeling method. The results demonstrate that various shapes of trees can be easily created through the control of a few growth volumes.



**Fig. 5.** Examples of tree creation using the proposed interactive control method

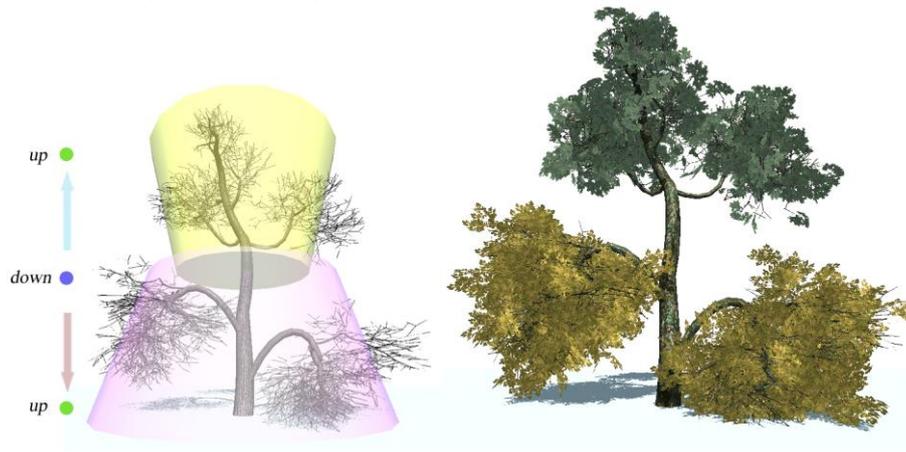
**Fig. 6** shows examples of tree growth made using interactive inputs and the independent branch control method, indicating that trees can be created to fit the user's intention. The editing process to create these shapes is easy and intuitive and can be used for games, animations, and movie synthesis.



**Fig. 6.** Examples of tree creation through the application of the proposed method

**Fig. 7** shows the resultant trees grown by setting different values of individual growth

volumes through the additional control of environmental elements. The lower growth volume was made by directing branches to grow downward by reducing gravitropism, and the upper growth volume was made by directing branches to grow upward by increasing phototropism. In addition, leaves were differently arranged to express two trees attached to one tree. This expression can easily be created by the user.



**Fig. 7.** Examples of tree creation through separate applications of environmental elements

Studies have recently been conducted on modeling methods for restructuring 3D trees based on input images. Although the proposed method is different from existing image-based tree modeling methods, if images are used as inputs and the user designates growth volumes to these tree images, trees with shapes similar to the input images can be made. **Fig. 8** shows examples of this application method. If inputs consist only of growth volumes, environmental elements, and the number of buds to be initially created (**Fig. 8**, colored points), all of which are taken from tree images, then the trees will grow with characteristics similar to the tree images. Although detailed branch patterns will be different from the input images, overall shapes can be made to be similar. With the foregoing, the limitations of existing image-based modeling can be overcome.





**Fig. 8.** Creation of a tree similar to the input image through growth volume control (the number of first buds (yellow), the number of second buds from parent branch (green) or trunk (pink)).

Finally, the efficiency of the proposed method was tested by measuring the average time to create a tree. The developed system automatically creates the vertex, index, and texture (uv) information of models while automatically modeling trees. The system also includes arranging leaf models designated in advance at arbitrary locations. Therefore, users only need to control growth volumes. **Table 1** shows the average times to obtain final results depending on growth volume control and growing trees. First, growth volume is edited through user inputs. As a major component of this study, the editing time in **Fig. 3, 4, and 6** was measured. It takes approximately 65 seconds for a user to plan the whole form of a desired tree and express it into a growth volume. **Fig. 6** shows the average time to generate four trees; this activity even took a little longer than it normally would because there were trees with many growth volumes. The next amount is the time when growth directions of branches are calculated based on the edited growth volumes, and the modeling of trees is performed. These are quickly calculable using the existing growth volume algorithm. The final time was measured in consideration of the distributions of leaves, the establishment of environmental factors, and the modification of work. Within an average of two to three minutes, a tree with a desired form may be generated, which verifies that the process is very efficient. However, this process is fundamentally different from those that employ graphic tools like 3D Max, and therefore, direct comparison is difficult. Thus, this study is advantageous in that it minimizes detailed modeling process using graphic tools and may generate a desired tree-formed model. The greater the number of the trees a user intends to produce, the higher time efficiency will be. Moreover, when this is applied to a real-time system, growth volumes are generated randomly, and if the growth volumes of desired parts are individually edited, the editing time will be considerably shortened, thus enabling the quick and efficient construction of a virtual ecosystem within 30 to 40 seconds. When the editing of growth volumes is complete, the modeling time does not take longer than 15 seconds; therefore, it may be applied to a large-scale system.

**Table 1.** Average time to create a tree using the proposed method (sec)

	<b>Edit growth volume</b>	<b>Growth and modeling of tree</b>	<b>Total time</b>
<b>Fig. 3</b>	60 sec	5 sec	112 sec
<b>Fig. 4</b>	72 sec	7 sec	146 sec
<b>Fig. 6</b>	102 sec	10 sec	265 sec

## 5. Conclusion

This study presents efficient procedural modeling methods using interactive growth volume control. Our method was developed to create diverse, unique trees through simple operations by the user and identify tree growth processes based on procedures. To this end, a growth volume algorithm was used. This method automatically calculates tree growth information that determines branch propagation based on intuitively structured processes. The growth and creation of diverse, unique trees were interactively controlled by the user. First, a function to easily control tree growth processes with a simple operation through the interactive control of two-line-based growth volumes was implemented, and an independent branch growth method for selected branches instead of the entire tree was designed. Diverse examples of the proposed methods were demonstrated through experiments.

Basically, this study includes all functions of existing growth volume algorithms. Growth volume editing and the independent growth of branches are proposed in this study, which are functions that existing methods do not have. The permitted range of overlapping the establishment of growth volumes was three or less, but at present, a user may establish as many as they want. While previously, overlapping methods were possible only in a series, diverse editing is now possible. Regarding the function of controlling environmental elements such as gravitropism, previously, partial control was possible through designating desired values to whole growth volumes. Now, however, individual designation of environmental elements only to desired growth volumes has become possible. Through this functional options, by adding the ability to individually edit a tree to existing methods of simultaneously generating multiple trees in an automatic, easy, and intuitive way, the basis of producing a user interface and system which make it possible to expand in diverse areas where three-dimensional tree models are necessary can exist.

The proposed procedural modeling is limited to determining branch propagation through growth volumes. The modeling does not separately provide functions to control diverse environmental elements that also affect the growth of plants. In future studies, methods for easily and intuitively controlling other elements that can affect tree growth will be examined. In addition, the interactive control of elements that should be included for realistic tree expression, such as leaves, flowers, and fruits, will be considered.

## References

- [1] J. Kim and H. Cho, "Efficient modeling of numerous trees by introducing growth volume for real-time virtual ecosystems," *Computer Animation and Virtual Worlds*, vol. 23, issue 3-4, pp. 155-165, May, 2012. [Article \(CrossRef Link\)](#).
- [2] A. Lindenmayer, "Mathematical models for cellular interaction in development, part I and II," *Journal of Theoretical Biology*, vol. 18, issue 3, pp. 280-315, March, 1968. [Article \(CrossRef Link\)](#).
- [3] F. Boudon, C. Pradal, T. Cokelaer, P. Prusinkiewicz and C. Godin, "L-py: an l-system simulation framework for modeling plant development based on dynamic language," *Frontiers in Plant Science*, vol. 3, no. 76, May, 2012. [Article \(CrossRef Link\)](#).

- [4] P. Prusinkiewicz, M. Hanmmel, J. Hann and R. Měch, "L-system: from the theory to visual models of plants," in *Proc. of the 2<sup>nd</sup> CSIRO Symposium on Computational Challenges in Life Sciences*, pp. 1-27, February 5-7, 1996. <http://algorithmicbotany.org/papers/l-sys.csiro96.html>
- [5] J. Power, A. J. Bernheim-Brush, P. Prusinkiewicz and D. Salesin, "Interactive arrangement of botanical l-system models," in *Proc. of the 1999 ACM Symposium on Interactive 3D Graphics*, pp. 175-182, April 26-29, 1999. [Article \(CrossRef Link\)](#).
- [6] H. Honda, "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*, vol. 31, issue 2, pp. 331-338, May, 1971. [Article \(CrossRef Link\)](#).
- [7] S. Ulam, "On some mathematical properties connected with patterns of growth of figures," in *Proc. of Symposia on Applied Mathematics*, vol. 14, pp. 215-224, December, 1962. <http://www.ams.org/bookstore?fn=20&arg1=psapmseries&ikey=PSAPM-14>
- [8] A. Runions, B. Lane and P. Prusinkiewicz, "Modeling trees with a space colonization algorithm," in *Proc. of the 2007 Eurographics Workshop on Natural Phenomena*, pp. 63-70, September 4, 2007. [Article \(CrossRef Link\)](#).
- [9] W. Palubicki, K. Horel, S. Longay, A. Runions, B. Lane, R. Měch and P. Prusinkiewicz, "Self-organizing tree models for image synthesis," *ACM Trans. Graph.*, vol. 28, issue 3. pp. 58:1-10, July, 2009. [Article \(CrossRef Link\)](#).
- [10] R. Borchert and N. Slade, "Bifurcation ratios and the adaptive geometry of trees," *Botanical Gazette*, vol. 142, no. 3, pp. 394-401, September, 1981. [Article \(CrossRef Link\)](#).
- [11] S. Longay, A. Runions, F. Boudon and P. Prusinkiewicz, "TreeSketch: interactive procedural modeling of trees on a tablet" in *Proc. of the Eurographics Symposium on Sketch-Based Interfaces and Modeling*, pp. 107-120, June 4-6, 2012. [Article \(CrossRef Link\)](#).
- [12] B. Lintermann and O. Deussen, "Interactive modeling and animation of branching botanical structures," in *Proc. of the Eurographics workshop on Computer animation and simulation '96*, pp. 139-151, August 31-September 1, 1996. [Article \(CrossRef Link\)](#).
- [13] B.lentermann and O. Deussen, "A modeling method and user interface for creating plants," *Computer Graphics Forum*, vol. 17, issue 1, pp. 73-82, March, 1998. [Article \(CrossRef Link\)](#).
- [14] X. Chen, B. Neubert, Y. Q. Xu, O. Deussen, S. B. Kang, "Sketch-based tree modeling using markov random field," *ACM Trans. Graph.*, vol. 27, issue 5, pp. 109:1-9, December, 2008. [Article \(CrossRef Link\)](#).
- [15] P. Tan, T. Fang, J. Xiao, P. Zhao and L. Quan, "Single image tree modeling," *ACM Trans. Graph.*, vol. 27, issue 5, pp. 108:1-7, December, 2008. [Article \(CrossRef Link\)](#).
- [16] Y. Livny, S. Pirk, Z. Cheng, F. Yan, O. Deussen, D. Cohen-Or and B. Chen, "Texture-lobes for tree modeling," *ACM Trans. Graph.*, vol. 30, issue 4, pp. 53:1-10, July, 2011. [Article \(CrossRef Link\)](#)
- [17] Y. Livny, F. Yan, M. Olson, B. Chen, H. Zhang and J. El-sana, "Automatic reconstruction of tree skeletal structures from point clouds," *ACM Trans. Graph.*, vol. 29, issue 7, pp. 151:1-8, December, 2010. [Article \(CrossRef Link\)](#).
- [18] B. Beneš, N. Andryscio and O. Št'ava, "Interactive modeling of virtual ecosystems," in *Proc. of the Fifth Eurographics conference on Natural Phenomena*, pp. 9-16, April 1, 2009.
- [19] B. Beneš, N. Andryscio and O. Št'ava, "Interactive modeling of virtual ecosystems," in *Proc. of the Fifth Eurographics conference on Natural Phenomena*, pp. 9-16, April 1, 2009. <http://dl.acm.org/citation.cm?id=2381694>
- [20] O. Deussen, P. Hanrahan, B. Lintermann, R. Měch, M. Pharr and P. Prusinkiewicz, "Realistic modeling and rendering of plant ecosystems," in *Proc. of the 25th annual conference on Computer graphics and interactive techniques*, pp. 275-286, July 19-24, 1998. [Article \(CrossRef Link\)](#).
- [21] W. Van Haevre, F. D. Fiore, P. Bekaert and F. Van Reeth, "A ray density estimation approach to take into account environment illumination in plant growth simulation," in *Proc. of the 20<sup>th</sup> spring conference on Computer Graphics*, pp. 121-131, April 22-24, 2004. [Article \(CrossRef Link\)](#).
- [22] E. Bruneton and F. Neyret, "Real-time Realistic Rendering and Lighting of Forests," *Computer Graphics Forum*, vol. 31, issue 2, pp. 373-382, May, 2012. [Article \(CrossRef Link\)](#).
- [23] K. Boulanger, S. Pattanaik, K and Bouaotuch, "Rendering grass in real-time with dynamic

lighting,” *IEEE Computer Graphics & Application Journal*, vol. 29, issue 1, pp. 32-41, January/February, 2009. [Article \(CrossRef Link\)](#).



**Jinmo Kim** is a full-time researcher at Research Institute for Image & Cultural Content, Dongguk University. He received B.E. degree in Multimedia Engineering from Dongguk University in 2006 and his M.E. & Ph.D. degrees in Multimedia from Dongguk University in 2008, 2012 respectively. His research interests include computer graphics, procedural modeling and natural phenomena.