

GeoMaTree : Geometric and Mathematical Model Based Digital Tree Authoring System

Seowon Jung¹, Daeyeoul Kim² and Jinmo Kim³

¹Department of Multimedia, Dongguk University

²Department of Mathematics and Institute of Pure and Applied Mathematics, Chonbuk National University

³Department of Software, Catholic University of Pusan

South Korea

[e-mail: jungseowon@dongguk.edu, kdaeyeoul@jbnu.ac.kr, jmkim11@cup.ac.kr]

*Corresponding author: Jinmo Kim

*Received March 30, 2017; revised July 23, 2017; accepted February 20, 2018;
published July 31, 2018*

Abstract

This study proposes a method to develop an authoring system(GeoMaTree) for diverse trees that constitute a virtual landscape. The GeoMaTree system enables the simple, intuitive production of an efficient structure, and supports real-time processing. The core of the proposed system is a procedural modeling based on a mathematical model and an application that supports digital content creation on diverse platforms. The procedural modeling allows users to control the complex pattern of branch propagation through an intuitive process. The application is a multi-resolution 3D model that supports appropriate optimization for a tree structure. The application and a compatible function, with commercial tools for supporting the creation of realistic synthetic images and virtual landscapes, are implemented, and the proposed system is applied to a variety of 3D image content.

Keywords: digital tree; procedural modeling; growth model; mathematical model; multiresolution 3D tree model; authoring tool; virtual landscape; photorealistic nature image

1. Introduction

In producing digital content such as movies and games, computer graphics related to plants are important elements in increasing the realism of content that has a natural background. Therefore, the effective generation and realistic expression of digital plants is an active area of research in a variety of fields.

New approaches and attempts aimed at realistically expressing digital plants under efficient structure are being made in diverse fields including computer graphics. Most of the studies are based on L-system, developed by Lindenmayer [1], and the botanical approach [2], and focus on realistic growth and expression. Later studies have attempted to generate digital trees more easily in an intuitive structure through the rule and grammar-based process [3]. Some studies have aimed at effectively reconstructing realistic digital plant by combining images with computer vision technique [4]. Because most of these studies focused on the realistic growth and modeling of plant, complexity and efficiency received relatively minor consideration. In creating a virtual landscape composed of multiple trees, simplification and optimization of the tree model structures is important, and studies should be made on real-time process that is appropriate for tree structures in order to further develop into real-time landscape.

This study presents a user-centered system that realistically models multiple diverse digital trees through a simple growth procedure in an efficient structure. The tree structure in our system is centered at the growth and shape of trunk and branch. For leaves, only the distribution is considered. The proposed GeoMaTree system has the following four main contributions:

1. The paper designs a geometric model based growth procedure for generating realistic tree model. This growth rule has an intuitive structure that can be predicted by the user, while simultaneously simplifying the generation structure. Furthermore, the user's satisfaction with the tree model's generation process, and the realism of the generation model can be verified experimentally.
2. The paper optimizes the branch pattern that appears in the growth procedure based on a mathematical model. The proposed method involves a novel approach to optimizing a mathematical model (based on its formula and grammar) through an algorithm and implementing it as a system. The real-time processing efficiency of the mathematical model, as well as the user's satisfaction, can be verified through comparisons with previous studies.
3. The paper describes the static level of detail (LOD) method that is optimized to the tree model so that it can be applied to real-time virtual landscape system. A novel simulation method has been designed, which applies the LOD method to the geometrical structure and mathematical model of a tree with the help of a growth model. Furthermore, its efficiency can be verified by creating a virtual landscape consisting of multiple trees, on which the LOD method has been applied.
4. The study implements compatible functions so that the generated tree model is applicable in diverse fields. By employing the compatible functions of the proposed authoring system, high quality rendering results and real-time applications can be directly produced to confirm the application's direction.

We verified the performance of the proposed system through diverse experiments where realistic expression of the tree model is checked by synthesizing with real image and

efficiency is verified by creating virtual landscape that is composed of multiple trees. **Fig. 1** shows our process and results of the proposed system.

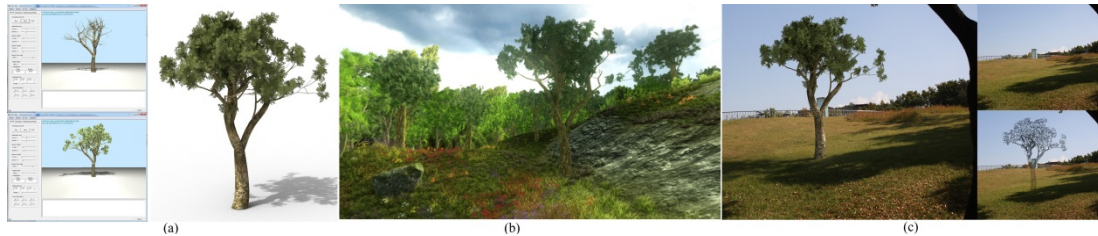


Fig. 1. GeoMaTree: geometric and mathematical model based digital tree authoring system.

- (a) Procedural modeling of tree based on the proposed growth and mathematical model,
 (b) Result of generating virtual landscape, (c) Synthetic image through our compatible functions.

Section 2 reviews previous works and Section 3 explains system structure of the digital tree authoring system that is proposed in this study. Section 4 explains the procedural modeling method based on a mathematical model, and Section 5 describes the application function that enhances the usability of the generated digital trees. Section 6 describes the analysis of the efficiency and usability of the proposed method obtained through a series of experiments. Finally, Section 7 draws together the conclusions of this study and offers some ideas for future research.

2. Previous Work

Studies on the expression of digital plants including trees progressed from the research on L-system by Lindenmayer [1]. This is a biological research, which examines L-system to describe the growth equation, growth pattern, and interaction of single cells such as yeast and mold or simple multicellular organisms. L-system later advanced from a tool for describing shapes and branch structure of complex plant to a tree modeling system. L-system sets a variety of factors that determine tree growth as parameters based on generation grammar that has recursive structure. After designating symbols, the system defines the process of transforming from current to next status as a production rule. Through this process, structure of creature is expressed as it grows and gets complicated. There have been diverse studies that attempt to express and create visual model of plant based on the L-system. There are studies that express smooth branch in curved structure [5] and define interaction between plant growth and environment or design modeling language by using the L-system [6]

Different from grammars, there are studies on modeling based on the analysis of geometric structure of trees [7] or on self-organizing methods that analyze botanical factors composing plants and reflect them in the growth [2,8]. There have been studies that attempted applying a botanical approach to the fate of a bud such as environmental and physical factors that affect the growth of plant [9] or that considered hormonal impact on growth [10-12]. Most of these methods have complicated rules in the procedural modeling method and either sometimes have too many parameters to control or their tree model process is not user-friendly due to unintuitive structure. Recently, studies have been conducted on mathematical approaches that are different from the existing digital plant modeling paradigms. One such study includes a modeling method that applies the CGA Shape grammar to the growth of digital plants based on the convolution sums of divisor functions [13]. Other examples include a study using hierarchical topology-preserving, blending, and morphing techniques for tree modeling

[14,15], another study involving a method for modeling trees using local convolution surface approximation, and yet another study based on simulating the growth of plant leaves that are composed of Voronoi diagrams [17]. However, these studies have shown limitations in the context of effectively generating various digital plants, since they lack in the aspect of intuitiveness from the user's perspective.

Facing these problems, interactive modeling methods have been developed that enable interactive control by users in the growth procedure of trees aiming at intuitive growth framework. For interactive control, Chen et al. [18] suggested a method that automatically generates 3D trees from a 2D input sketch. However, methods that make use of sketches require a design technique to create high-quality output. Recently, a method that reconstructs a 3D tree model from more than two images of the tree has been developed using computer vision techniques [4]. This method detects the features of the trees in the input image, and computes the 3D information from the detected point clouds. The point clouds are directly computed, but 3D scanners can also be used for the detection [19,20]. However, a previous tree model template is necessary, which introduces many requirements. Moreover, the results are often limited, as the generated trees are identical to the image. Meanwhile, there is a method that generates a tree model from an intuitive structure using a multi-touch tablet interface [21]. However, this has a rather difficult environment for general users to comprehend, and uses a method based on the L-system structure. Kim et al. [22] studied an interactive tree modeling method, which includes collision detection and collision avoidance using a sweep surface, but this method has the limitation that it requires the understanding of geometric model theory.

Most importantly, these studies are limited in their ability to generate multiple trees simultaneously, as they are focused on the realistic expression of a single-tree model. In addition, they do not consider the tree structure for real-time processing. Studies considering the expression of real-time virtual landscapes are mostly related to the rendering of areas involving illumination models that create realistic, real-time scenes [23-25]. Although there are several studies on the level of detail of tree model, they enhanced the performance by expressing 3D tree model in billboard of plane [26-28]. Recently, an LOD study of the optimal tree model for VR landscaping was carried out [29]. However, this method provides a simple model suitable for conventional commercial engines, but does not provide the variety of subset model data that are user-friendly and suitable to the system. So far, there is virtually no research on the LOD method of trees other than billboard.

On the contrary, there exists an effective procedural method that simultaneously creates a number of the diverse buildings and objects that compose a city. Vanegas et al. [30] proposed a framework that creates an effective 3D urban model through intuitive high-level control based on previously given urban information. In addition, diverse procedural modeling methods have been studied, and the authoring tools and systems that enable users to edit the results are currently under development [31]. However, as trees have more complex structures than box-shaped buildings, research on the multiple trees that constitute a virtual landscape is currently scarce.

3. System Overview

This study focuses on developing a user-oriented authoring system that can model and simulate many diverse trees, which are components of a virtual landscape or a natural background, in a realistic and efficient structure. The structure of the authoring system has

been designed in two sections (Procedural modeling, Simulation and applications). **Fig. 2** illustrates this aspect, and the authoring system has been constructed by dividing it into sub-modules, based on the two proposed core functions.

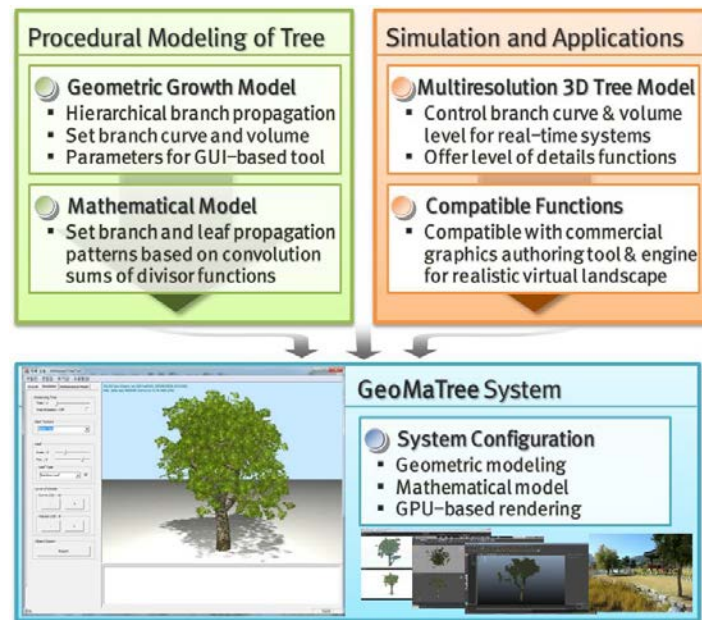


Fig. 2. System overview of the proposed GeoMaTree system.

The first function is procedural modeling that determines the growth model and growth pattern of branches comprising a tree. In this study, a predictable and realistic growth process with user-friendly intuitive rules has been designed which possesses an efficient branch structure that can be processed in real time, through a growth pattern based on a mathematical model. In Section 4, the procedural modeling method has been described, which takes into consideration both the growth model and growth pattern.

The second function comprises a data parsing feature, together with a simulation method that can be efficiently rendered in real-time systems, such as digital games. Such a function allows the tree models, created through the proposed authoring system, to be applied in various digital contents. First, a multi-resolution 3D tree model has been designed by applying a branch growth pattern based on a mathematical model and an LOD that adjusts the details of the geometric model structure. Subsequently, to increase the utility of the proposed authoring system, compatible functions have been implemented that are applicable to the creation of various digital content such as games, animations, and movies. Section 5 describes the simulation and application of the authoring system, in a step by step method.

4. Procedural Modeling

The process of growing trees can be divided into a growth model that determines the overall shape of the tree and a method that defines the distribution of the branches. This study proposes a procedural modeling method whereby the generated tree model has an efficient memory structure, but the user can effectively control the overall shape of the tree.

4.1 Geometric Growth Model Simplification

In the growth model of the proposed system, users can easily produce the desired trees, as the propagation of the current branch to the new one is predictable in an intuitive structure. This study removes unnecessary and ambiguous parameters based on the growth model proposed by Kim et al. [32] and only selects core parameters that are essential to realistic growth. Then, an algorithm is designed so that the proposed system can be implemented as a module.

First, the parameters needed to define the growth of child branches are computed using the growth information of the parent branch, such as the position and direction. The position of a bud is determined to generate a new child branch above the parent branch. The position of the child branch (p_i) can be obtained by controlling the length of the growth direction vector (\vec{v}_{pr}) from the position of the parent branch (p_{pr}). Here, whether the overall distribution of the branch is growing upward or hanging low depends on the length control parameter (r_{bud}). Next, the length (l_i) and direction (\vec{v}_i) of the next child branch are calculated. For the direction, a basis growth vector (\vec{v}_{gr}) is calculated by multiplying the rotation angle that determines the width (θ_i) and rotation angle that determines the volume (ϕ_i) using the parent branch as a local coordinate. Random seeds are then allocated around the basis direction, and one seed is chosen at random. The vector from the position of the bud (p_i) to the selected seed (p_{seed}) gives the final direction vector (\vec{v}_i). The random seed is an important factor that introduces some irregularity to the growth pattern of the branch. According to the number of seeds and their distribution, different trees can be generated from the same value (Fig. 3(a)). Environmental or physical phenomena can be applied to the growth direction of the branches to express more diverse and natural-looking growth. This can be simply addressed by adding an external vector (\vec{v}_{ext}) to the calculated growth direction vector, as expressed in Equation (1).

$$\begin{aligned} p_i &= p_{pr} + r_{bud} * \vec{v}_{pr}, \\ \vec{v}_{gr} &= l_i * R(\phi_i)R(\theta_i) * \vec{v}_{pr}, \\ \vec{v}_i &= (p_{seed} - p_i) + \vec{v}_{ext} \end{aligned} \quad (1)$$

Once the growth direction of the branch has been determined, the surface of the branch, including its volume and curvature, is calculated. To express the irregular surface of the branch at a fine scale using only the growth vector, the branch length must be sufficiently short such that the segments can be placed together end to end. However, this can be computationally expensive, and generating one tree can take a long time. In this study, the start and end points of the branch growth and two control points are chosen at random (Fig. 3(b)). A Bézier curve is then calculated using these four points to express the irregular curved surface. As shown in Fig. 4(a), the precision of the curvature of the branch differs according to the step of the curve. Finally, as for the volume of the curve, circles of radius (r_i^s, r_i^d) are centered normal to the curve, as in Fig. 4(b), and these are linked to determine the shape of the branch. The total number of vertices that constitutes the branch depends on the difference in the number of surface vertices around the circles (n_{surf}). Equation (2) describes the calculation of the control point of the branch curve in Fig. 3(b). The position of the surface vertex that composes the volume is computed using the same method as shown in Fig. 4.

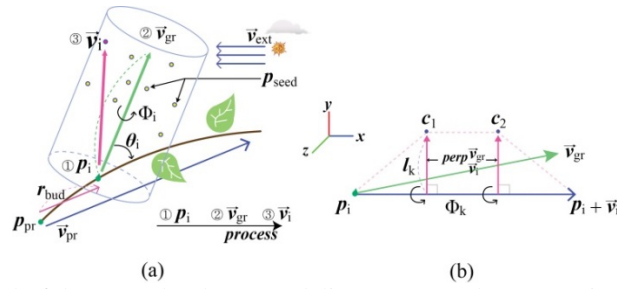


Fig. 3. Growth model of the procedural tree modeling. (a) Branch propagation process, (b) Control point calculation process for smooth branch curve.

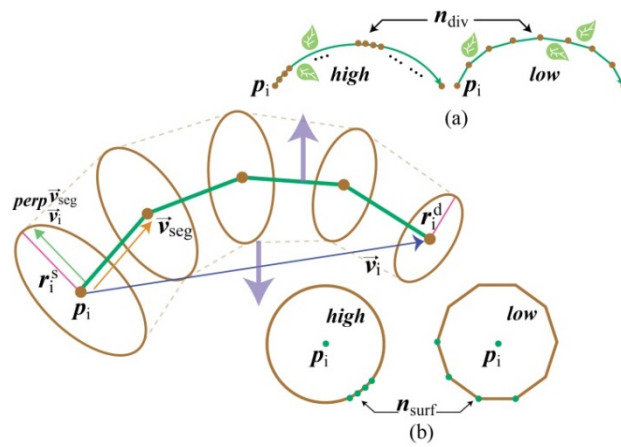


Fig. 4. Curve and volume step setting for geometric model of branch. (a) Setting of segment step of the branch curve using Bézier curve (n_{div}), (b) Setting the segment steps of the surface that composes the volume of the branch (n_{surf}).

$$c_k = R(\phi_k) \text{perp}_{\vec{v}_i}^{\vec{v}_{gr}} * l_i + \frac{k}{3} * \vec{v}_i + p_i, \\ \text{perp}_{\vec{v}_i}^{\vec{v}_{gr}} = \vec{v}_{gr} - \frac{\vec{v}_{gr} \cdot \vec{v}_i}{\|\vec{v}_i\|^2} \vec{v}_i \\ \text{where, } k = 1, 2 \quad (2)$$

Here, c_k represents a control point for a smooth branch's curve that can be calculated with an angle of rotation (θ_k) that determines the spatial location of the control point, the length value (l_k) that controls the degree of bending of the branch curve, and the position of the bud (p_i). The perpendicular vector ($\text{perp}_{\vec{v}_i}^{\vec{v}_{gr}}$) that determines the control point can be calculated by using the basis growth vector (\vec{v}_{gr}) and the final direction vector (\vec{v}_i).

The main parameters for the growth model are as follows:

- Value that controls the position of a bud, the starting point of each branch: r_{bud}
- Length and direction of branches that grow from a bud: l_i, θ_i
- Number of random seeds: n_{seed}
- External factor that affects the growth direction: \vec{v}_{ext}
- Initial thickness of the branch surface: r_o^s
- Curvature of the curved surface of the branch: ϕ_k, l_k

Algorithm 1 summarizes the process of determining the overall shape of a tree by growing branches based on the defined parameters. By organizing these parameters in an user interface, the proposed authoring system enables users to intuitively understand the impact of controlling these values on the overall shape of the tree. Here, the volume rotation angle of the branch (θ_i) and the step of the branch curve (n_{div}) are set to pre-processing threshold values.

Algorithm. 1. Procedural branch modeling using growth model.

```

 $p_{pr} \leftarrow$  position of bud of a parent branch.
 $\vec{v}_{pr} \leftarrow$  growth direction of a parent branch.
hierarchical branch structure
procedure CALCULATION OF GROWTH INFORMATION OF A NEW BRANCH{ }
    input growth parameter value.
    ① calculation of bud position  $p_i$  through controlling  $r_{bud}$  from information of the parent branch
    Equation (1).
    ② calculation of growth basis direction  $\vec{v}_{gr}$  of a child branch that will grow from  $p_i$  (Equation (1)).
    set growth space  $(l_i, \theta_i)$  using  $\vec{v}_{gr}$  as basis.
    allocate  $p_{seed}$  of the number of  $n_{seed}$  in random position inside the growth space.
    randomly choose one  $p_{seed}$ .
    ③ calculation of growth direction  $\vec{v}_i$  from the position  $p_i$  of the current bud to the selected  $p_{seed}$ 
    (Equation (1)).
    procedure GEOMETRIC MODELING OF A BRANCH ( $p_i, \vec{v}_i$ )
        calculation of direction vector normal to the growth direction  $\vec{v}_i$  using  $\vec{v}_{gr}$  (Equation (2)).
        creation of control point for Bézier curve from the position of bud  $p_i$  (Equation (2)).
        calculation of Bézier curve from the four control points of  $p_i, p_i + \vec{v}_i, c_1, c_2$ .
         $n_{div} \leftarrow$  segment step of the Bézier curve of a branch.
        control the smoothness of the branch curve using  $n_{div}$ .
         $n_{surf} \leftarrow$  segment step that composes branch surface.
        control the naturalness of the branch surface using  $n_{surf}$ .
    end procedure
end procedure

```

The parameters of the growth model are listed in the graphical user interface (GUI) of the proposed authoring system, enabling users to conveniently manipulate each menu expression when generating the desired trees. **Fig. 5** shows a screenshot of the system. The growth model can be checked in real-time through the viewer on the right-hand side.

The random seed of the growth model introduces irregularity into the tree growth process by selecting one seed at random. To simultaneously produce multiple trees of similar shape, users can control the growth direction of branches by selecting the same number of seeds as trees. This has the advantage that, if the random seeds change the final growth direction of each branch in an identical growth condition, a number of trees with similar overall shapes but different patterns can be simultaneously generated. Hence, the proposed system supports the production of multiple trees to compose a virtual landscape, rather than a single-tree model, in an easy and intuitive structure.

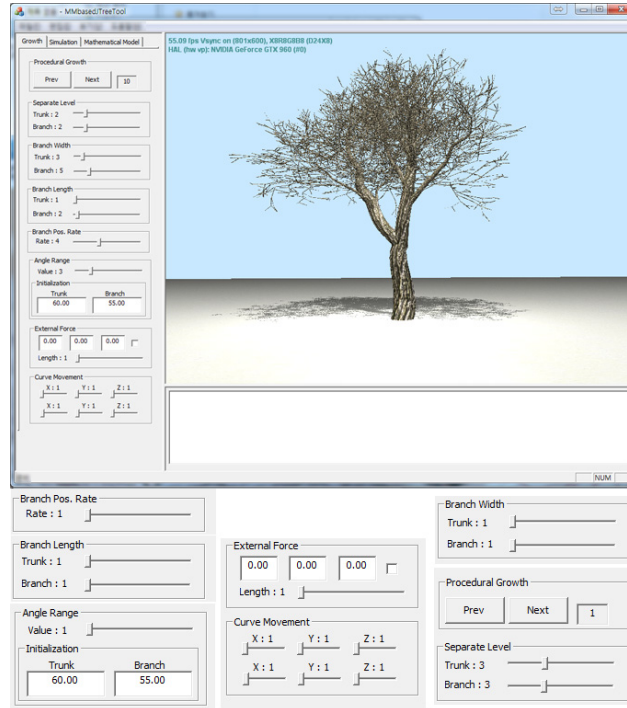


Fig. 5. Results of GUI-based system design with the proposed growth model.

4.2 Mathematical Model Based Growth Pattern

A method of determining the number of child branches generated from the parent branch is equally important as the growth model. However, it is inconvenient for users to state the desired number of branches at every time step. If the number of child branches is fixed at a constant value, the distribution of branches will increase exponentially.

This study applies a growth pattern based on a mathematical model using the convolution sum of the divisor function, as proposed by Kim et al [13]. Users can focus on the shape of the digital trees they want to produce, and the generated trees are provided in an efficient, real-time structure. However, existing studies have problem when directly applied to a system because they list growth pattern in a grammar form. Hence, growth pattern was defined as an algorithm using a mathematical model to effectively implement the distribution of branches and leaves.

The convolution sum of the divisor function is formed by a combination of divisor functions that have natural numbers as their solution according to a specified rule. Equation (3) shows the generalized equation for the convolution sum according to the digital tree growth, where k runs from 1 to $N-1$. The change in the calculated value is known to be constant (ex. $\sum_{k=1}^{N-1} \sigma_1(k)\sigma_1(N-k) = \frac{1}{12} \{5\sigma_3(N) + (1-6N)\sigma_1(N)\}$). Hence, if the design makes N growth steps corresponding to the growth pattern of a branch, the distribution of the branch can be maintained at a constant value, making the branch a structure that is predictable to the users. Distribution of leaves is calculated using the same approach. Although this study only accounts for leaf distribution, diverse expressions of tree growth is possible as creation and falling of leaves are determined according to the sign of the computed value (n_{lf}).

$$\begin{aligned}
& \sum_{k=1}^{N-1} D_b^i(B^i(x,y)k)D_l^i(L^i(x,y)(N-k)), \\
n_{dv} &= D_b^{g_s}(B^{g_s}(x,y)k) - D_b^{g_s-1}(B^{g_s-1}(x,y)k), \\
& \text{where, } n_{br} \in \{z | 1 \leq z \leq |n_{dv}| + 1\} \\
n_{lf} &= D_l^{g_s}(L^{g_s}(x,y)(N-k)) - D_l^{g_s-1}(L^{g_s-1}(x,y)(N-k))
\end{aligned} \tag{3}$$

Here, D_b^i, D_l^i are the i^{th} divisor function equation that determines the propagation of the branch and the leaves, and B^i, L^i are the constant multiples value of the divisor function. In this case, the pair (x,y) refers to x^y constants, and N is a growth step. Moreover, n_{br} is the number of propagated branches using the subtraction value of the convolution sum (n_{dv}). And the number of leaves is also determined by the calculated n_{lf} value. A study by Kim et al. [13] confirmed only the efficiency of tree branch distribution based on mathematical model, which was experimentally verified. Fig. 6 compares the branch distribution and the number of polygons of the actual tree model to reconfirm the efficiency.

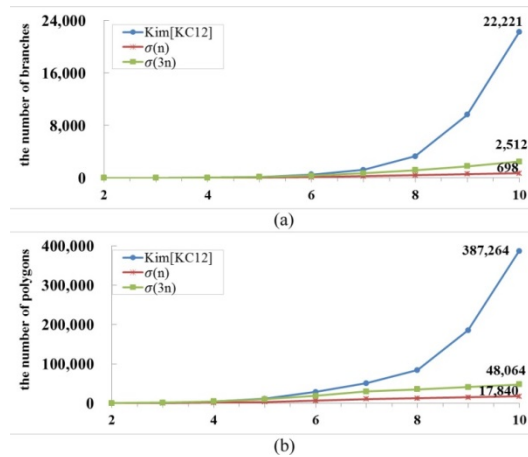


Fig. 6. Results of an efficiency test of the mathematical model-based branch propagation pattern.

- (a) Comparison of the increase in the number of branches through tree growth,
- (b) Comparison of the total number of polygons composing the trees.

Types of divisor functions that constitute a mathematical model are also diverse, making it appropriate for expressing irregularity of the branch propagation pattern. Equation (4) presents the types of divisor function (D_b^i, D_l^i) that are applied to the proposed system. Finally, constant of the divisor function (B^i, L^i) are the increasing coefficient of branches and leaves; and users can control the complexity of the branch propagation pattern based on the system where the generated trees are used. Through this, the usability of the generated tree model can be diversified.

$$\begin{aligned}
1 : \sigma_a(n) &= \sum_{d|N} d^a, \\
2 : \sigma_{a,b}(n; m) &= \sum_{\substack{d|N \\ d \equiv b \pmod{m}}} d^a,
\end{aligned} \tag{4}$$

$$3 : \tilde{\sigma}_a(n) = \sum_{d|N} (-1)^{d-1} d^a ,$$

$$4 : \sigma_a^*(n) = \sum_{\substack{d|N \\ n/d: odd}} d^a$$

Here, σ represents the sum of the a th powers of the divisors (d) of the natural number n that satisfies each condition.

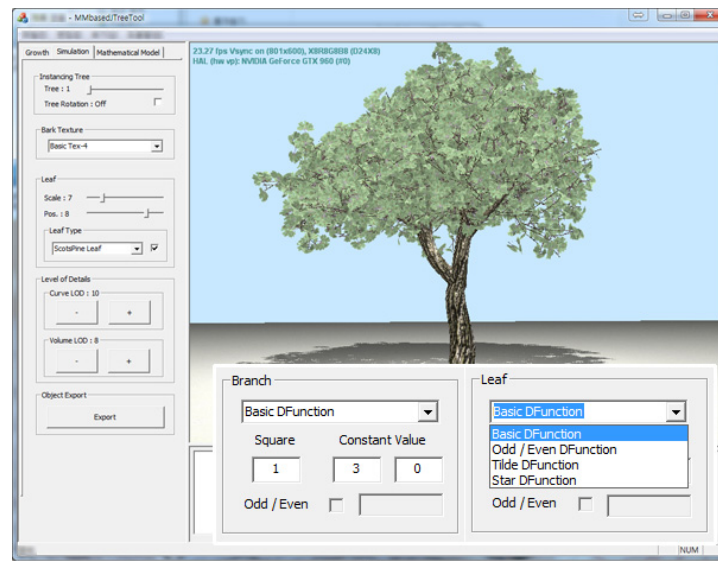
The proposed system enables users to select which divisor functions are applied to the branches and leaves. Moreover, our approach supports the production of trees via an efficient propagation structure and ensures natural presentation by providing an interface that controls the constant value determining the complexity of the branches. The mathematical model is intended to improve the branch propagation structure in order to increase the efficiency of generating virtual landscapes composed of trees. This process simultaneously addresses the irregularity of the growth process and the leaf processing scheme (**Fig. 7**). **Algorithm 2** defines the mathematical model-based branch and leaf propagation distribution in an applicable form to the system.

Algorithm. 2. Mathematical model based growth pattern.

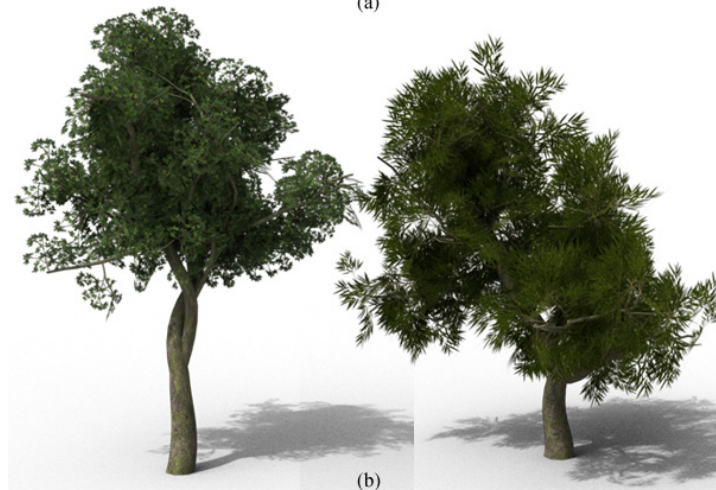
```

procedure BRANCH PROPAGATION PATTERN
   $D_b^i \leftarrow$  selects divisor function of the  $i^{\text{th}}$  branch (refer to Equation (4)).
   $B^i(x, y) \leftarrow$  selects increasing coefficient of the  $i^{\text{th}}$  branch.
  for  $k \leftarrow 1, N-1$  do
    if  $k = N-1$  then (in case of trunk)
       $n_{dv} = D_b^i(B^i(x, y)k)$ 
    else
       $n_{dv} = D_b^i(B^i(x, y)k) - D_b^i(B^{i-1}(x, y)k)$ 
    end if
     $n_{br} \in \{z | 1 \leq z \leq |n_{dv}| + 1\} \leftarrow$  calculates the number of branches that are to be newly added.
    creates  $n_{br}$  current branches from the parent branch.
    procedure LEAF PROPAGATION PATTERN ( $n_{br}$ )
       $D_l^i \leftarrow$  selects divisor function of the  $i^{\text{th}}$  leaf (refer to Equation (4)).
       $L^i(x, y) \leftarrow$  selects increasing coefficient of the  $i^{\text{th}}$  leaf.
      for  $j \leftarrow 1, n_{br}$  do
         $n_{lf} = D_l^i(L^i(x, y)(N - k)) - D_l^i(L^{i-1}(x, y)(N - k))$ 
        calculates location of the leaf based on the information of the current branch growth
        ( $p_i, \vec{v}_i$ ).
        if  $n_{lf} > 0$  then
          creates  $n_{lf}$  leaves of the current branch.
        else
          falls  $n_{lf}$  leaves from those created from the branch.
        end if
      end for
    end procedure
  end for
end procedure

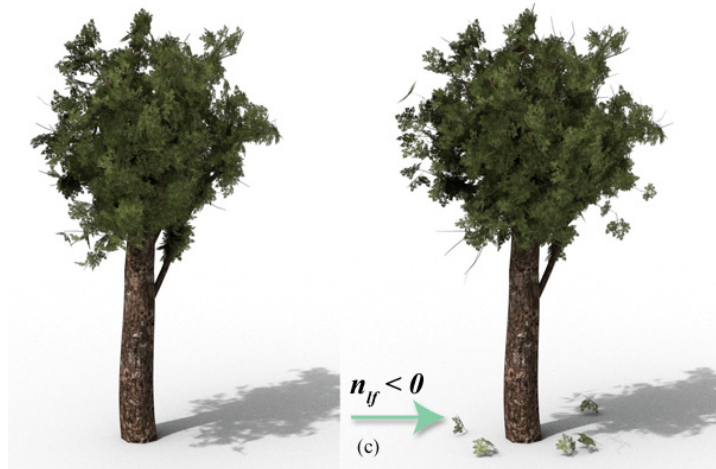
```



(a)



(b)



(c)

Fig. 7. Procedural modeling of trees based on a system tool using the proposed growth and mathematical model. (a) GUI-based system menu composition of the mathematical model, (b) Tree generation with the proposed authoring system, (c) Leaf expression based on a mathematical model.

5. Simulation and Applications

The digital tree models generated using the proposed authoring system are frequently used in applications such as interactive content that requires real-time processing and 3D animation. Hence, this study implements a multi-resolution 3D tree model production system, and offers compatibility with commercial tools to ensure its applicability in diverse content fields.

5.1 Multi-resolution 3D Tree Model

The mathematical model is intended to allow the efficient management of a number of branches composing a tree. However, as the vertices and surfaces within one branch model are complicated, controlling only the number of branches has limitations. This system was designed so that the number of vertices can be controlled during the process of modeling the branch surface based on Bézier curves.

The modeling of branch surfaces is divided into a curve step, which expresses the curve using the growth direction, and the surface vertex step, which expresses the branch volume. Smoothness between branches can change according to the curve step control, and the naturalness of the branches can change according to how the volume component is managed. The number of vertices that constitute the branch depends on the extent of both components, which eventually affects the total number of vertices in the tree. Fig. 8 shows the subset vertex data of branch model that was created by controlling two factors. We can see that vertex and polygon of the branch model changes according to the curve step (n_{dv}) and surface step (n_{surf}). It has an advantage in that the tree models that fit the system size can be used adequately whilst subset data consisting of diverse polygons are automatically generated in the process of creating one tree model.

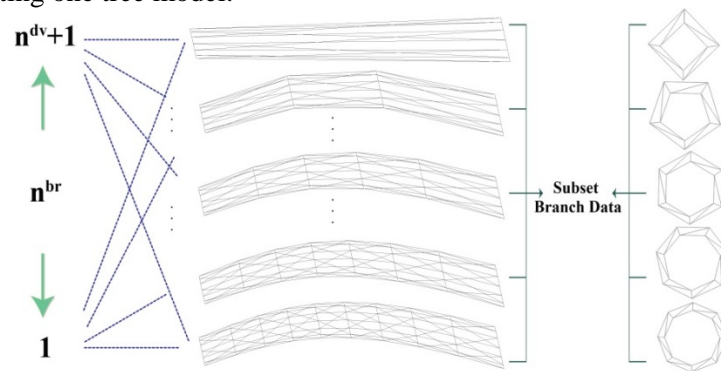


Fig. 8. Subset branch data for a multi-resolution 3D tree model.

The LOD method is used to enhance performance in systems that require real-time processing, such as games. LOD controls the subdivision of the model according to the distance between the camera and the model. Since tree models are mostly used in the background, they are generally relatively far from the camera. However, this does not mean that the tree models can be expressed using few vertices. This study has designed a multi-resolution model that can be used in the LOD method by providing high-resolution trees that have the same overall shape. Users can control the naturalness of the tree model according to the features of the system in which the tree model is to be used. For example, a model with few polygons can be provided for relatively low specification systems, such as mobile devices, whereas high-quality content can be generated by increasing the naturalness when there is no limit on the number of polygons. In Fig. 9, static LOD that is appropriate for the tree model

was determined by using subset data in Fig. 8. The proposed LOD method, by solving the limitation of previous studies [29] that provided only simple models suitable for commercial engines, now it provides the subset data authoring features suited to the system so that the user can control according to the purpose of the content.

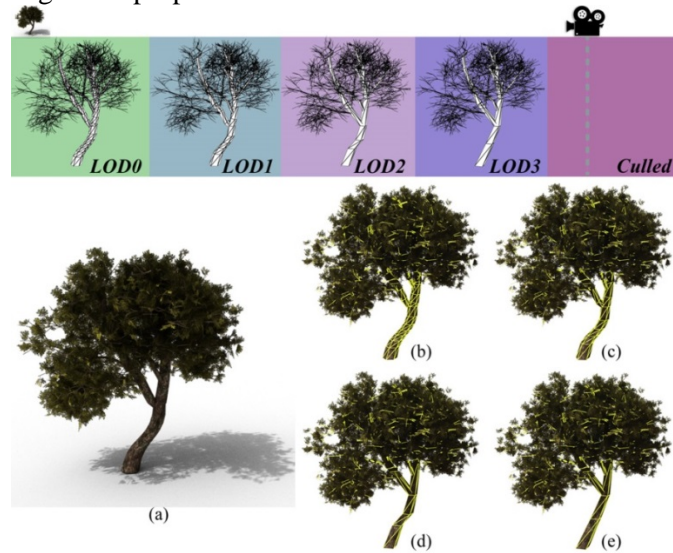


Fig. 9. Results of the branch model control of the multi-resolution 3D tree model (n_{div} , n_{surf}). (a) The generated tree model ($D_{b,l}^i : \sigma_l(3^m * k)$), (b) (10, 8), (c) (8, 7), (d) (6, 6), (e) (4, 5).

5.2 Compatible Functions

Our aim is to design a system that can be used for games, animation, and diverse 3D digital contents. Hence, for the generated tree model to be used in animation and image synthesis, it is important to be compatible with commercial graphics authoring tools such as 3ds Max and Maya. Moreover, if the produced models will be used in interactive content based on game engines, compatibility with the relevant environment is necessary. Hence, a compatibility function is implemented so that the tree model can be applied to graphics authoring tools or commercial engines without extra modification (Fig. 10).

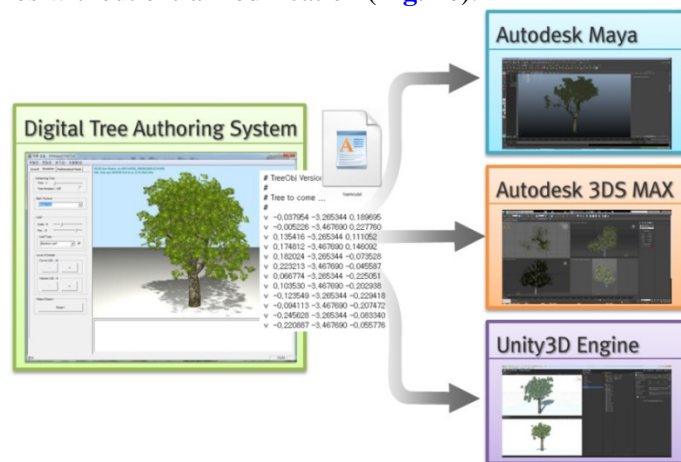


Fig. 10. Compatibility relationship between the proposed system and the other authoring tools.

In the case of the digital trees generated using the proposed system, all surface information is stored as a description of the curved surface. Moreover, the process of procedural growth can be checked in real-time using a viewer function. However, as other tools and engines have different drawing systems, the tree model should be transformed to a more general polygon-type structure prior to storage. The method of computing the vertex positions that constitute the curved surface of a branch was described in Section 4.1. The index information of each triangle is stored in a buffer. The texture coordinate is set such that the starting coordinate is $(0, 0)$ and the end point of the branch growth becomes $(1, 1)$ in every growth step. As a result, the texture can be applied repeatedly to the branches on each step (Fig. 11).

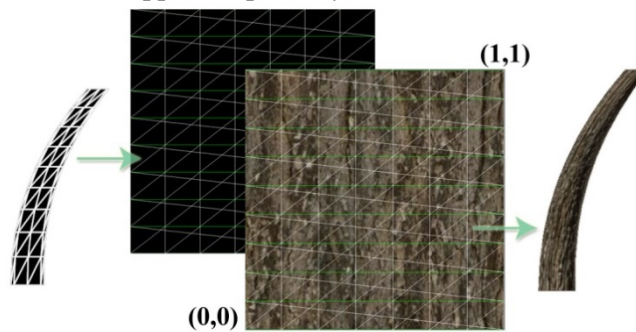


Fig. 11. The texture coordinate setting for texture mapping.

The proposed system has a compatibility function for the 3ds Max, Maya, and Unity3D that are currently used in graphic design and 3D interactive content generation, as shown in Fig. 10. Hence, the supported file formats include .fbx, .obj, and .dae; in this study, we use the .obj format to treat the polygon model.

5. Experimental Results and Analysis

The proposed GeoMaTree system was implemented using Visual Studio 2013 (Microsoft, Redmond, WA, USA), MFC, DirectX Software Development Kit (SDK), and the PC used in the test had an Intel® *core™* i7-4790 (Intel Corporation, Santa Clara, CA, USA) with 8 GB random access memory (RAM) and a GeForce GTX 960 GPU (NVIDIA, Santa Clara, CA, USA). The test of the compatibility function considered 3ds Max, Maya (Autodesk, San Rafael, CA, USA) and Unity3D 5.0 (Unity Technologies, San Francisco, CA, USA).

The test was divided into four steps. First, we tested whether digital trees could be effectively produced using the proposed authoring system. Users selected the necessary parameters for growth in the GUI menu, and selected the mathematical model to be applied to the trees. The growth factors were chosen once at the beginning of the tree generation process. Moreover, users could revise the growth factors if they wanted to change the tree shape during the growth process. Fig. 12 shows the results of the tree model generation using the authoring system. These results confirm that the tree model generated trees conforming to the diverse growth factors applied by the users. The impact of these growth factors on the trees can also be intuitively checked by comparing the growth values with the generated tree model.

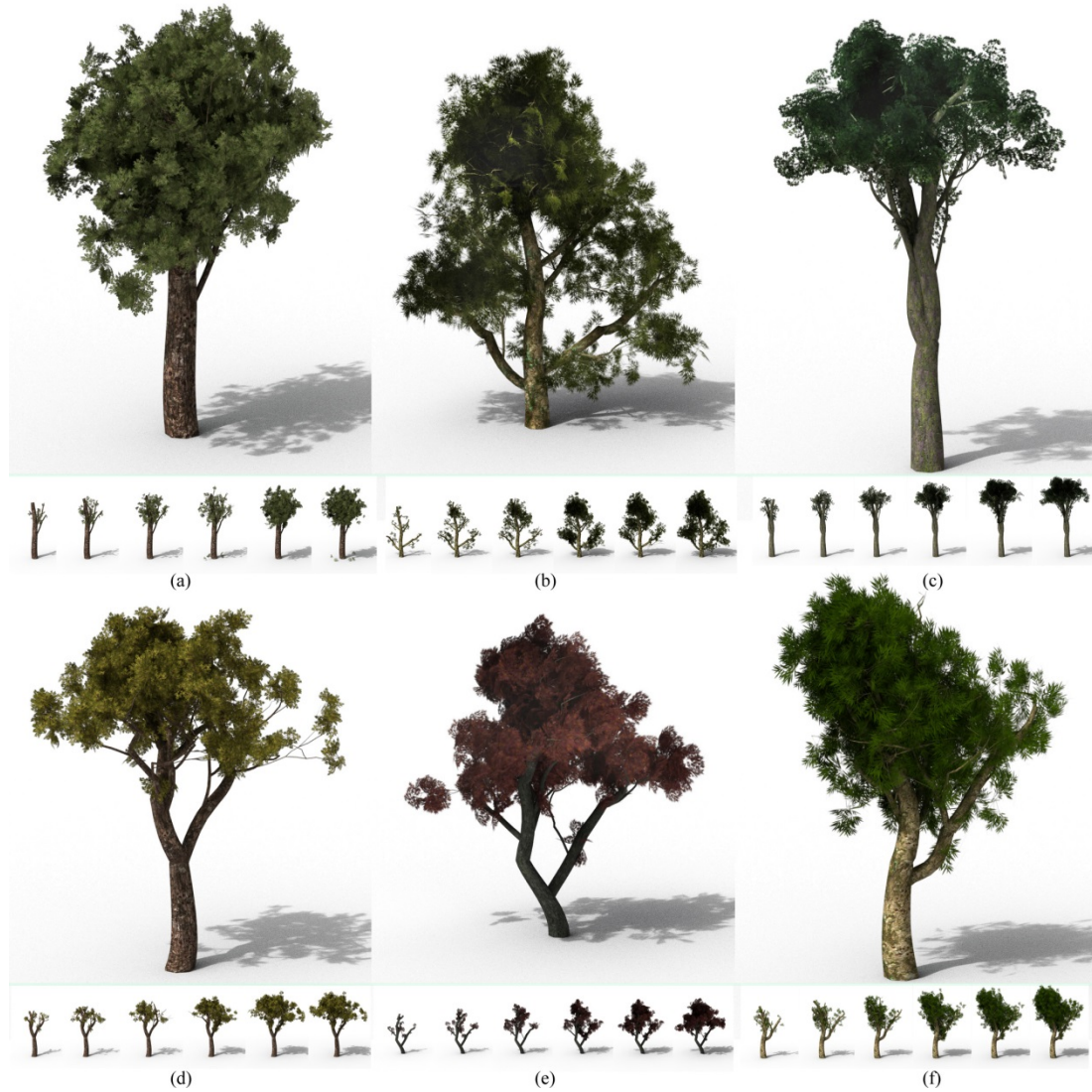


Fig. 12. Results of step-by-step tree generation using the proposed authoring system $(D_{b,l}^i(B^i(x,y)), r_{bud}, l_i, \theta_i)$. (a) $\sigma_1(2^m * k), 0.7, 0.7, 50^\circ \sim 60^\circ$, (b) $\sigma_{1,0}(2^m * k), 0.2, 0.6, 60^\circ \sim 80^\circ$, (c) $\sigma_{1,1}(2^m * k), 0.9, 0.5, 40^\circ \sim 55^\circ$, (d) $\sigma_1^*(k), 0.75, 0.65, 60^\circ \sim 70^\circ$, (e) $\tilde{\sigma}_1(k), 0.5, 0.55, 60^\circ \sim 75^\circ$, (f) $\sigma_1(3^m * k), 0.55, 0.8, 45^\circ \sim 65^\circ$.

Next, we tested the efficiency of the multi-resolution 3D tree model within the authoring system. We confirmed that the efficiency of the tree model in terms of the branch propagation pattern is basically higher than that of the general method, as shown in Fig. 6. The proposed system includes an additional method for the real-time processing of virtual landscapes consisting of multiple trees. As shown in Fig. 9, tree models with the same number of growth steps were produced at four different resolutions. Table 1 presents the efficiency results for each resolution. The number of frames per second (FPS) was measured by allocating 1500-6000 tree models of each resolution in the scene. The possibility of real-time processing was verified by recording at the rate of 25.44 fps in a virtual landscape comprising of 4,500 trees having the highest resolution (Fig. 9(b)), with the growth pattern based on a

mathematical model without applying the LOD. The model with the lowest resolution ([Fig. 9\(e\)](#)) faced no problems in terms of real-time processing, but the quality of the virtual landscape was found to be poor. Consequently, the tree model generated through applying the LOD provided the advantage of satisfactory performance while simultaneously ensuring quality. It was confirmed that users can effectively create a real-time virtual landscape as they effectively control the resolution of the tree model according to the complexity of the system.

Table 1. Efficiency test results of a multi-resolution 3D tree model.

Resolution-Figure	The number of polygon	FPS(number of trees)			
		1500	3000	4500	6000
high-9(b)	28,166	34.27	29.11	25.44	22.59
middle-9(c)	17,951	51.19	46.15	43.16	38.91
middle-9(d)	12,497	53.89	49.45	46.73	41.45
low-9(e)	8,507	81.16	75.56	66.46	59.19
LOD	-	49.18	46.70	42.51	39.29

In addition to the efficiency of the tree model created by the proposed system, generation of natural-looking virtual landscape is also an important experiment factor. [Fig. 13](#) shows the results of generating virtual landscape by using terrain editor in Unity3D where the LOD-applied tree models in this study were integrated in Unity3D engine by using a compatible function. Given that all trees have built-in LOD, efficient rendering according to the distance from the camera is possible. Moreover, we could confirm the naturalness of the generated virtual landscape.



Fig. 13. Results of generating virtual landscape composed of tree models created by the proposed system.

Another important goal of this study was to ensure that general users performing graphics tasks could make effective use of the proposed authoring system. Hence, we asked 20 people who were familiar with graphics authoring tools such as 3ds Max to generate one of the trees in Fig. 12, and measured the time taken to generate the tree model. Table 2 summarizes these results. The production of the same model using the proposed authoring system took a minimum time of 4.25 min, a maximum time of 7.15 min, and an average time of 5.74 min. These results were compared with the tree generation times using tree editor [33], which is a digital tree modeling tool that uses Unity3D. Consequently, it took a minimum time of 14.59 min, a maximum time of 23.14 min, and an average time of 19.76 min, thus revealing a time difference of ~14 min with the proposed system. In the case of tree editor with Unity3D, users must select inconvenient user interface and a number of complicated parameters, as this system provides diverse functions that precisely control and realistically express a single-tree model. On the contrary, the growth process of the single-tree model in the proposed authoring system is relatively simple, as it focuses on the effective generation of multiple trees. Hence, we confirmed the effectiveness of the proposed system in generating the desired trees. Subsequently, when the level of user's satisfaction was rated on a 5-point scale, the two authoring systems recorded average values of 4.12 and 4.09, respectively. Thus, both authoring systems have shown high satisfaction levels having values of over 4 points. Therefore, it has been demonstrated that they can both produce the desired tree models in similar production environments.

Table 2. Test results showing the usability and satisfaction levels of the proposed GeoMaTree

Fig	Average production time(min)	
	Unity3D tree editor	Our GeoMaTree system
12(a)	17.15	6.15
12(b)	22.11	4.99
12(c)	14.59	7.15
12(d)	19.46	4.25
12(e)	22.13	5.11
12(f)	23.14	6.78
Satisfaction level (1~5)	4.12	4.09

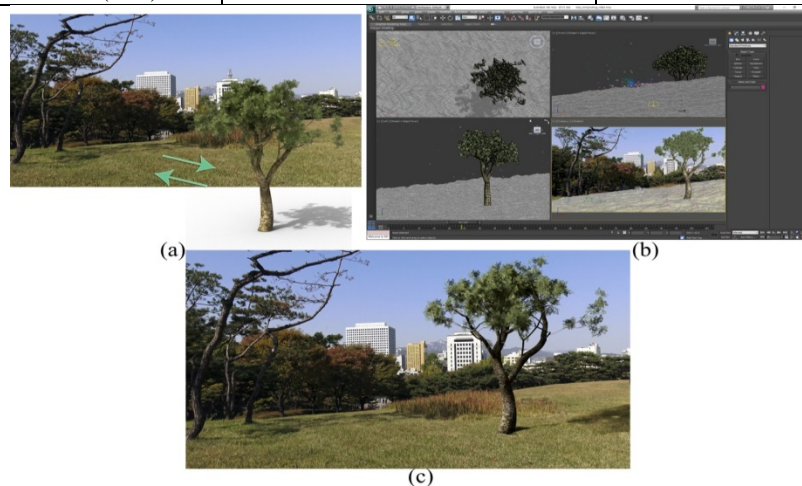


Fig. 14. Compatibility function for application of high-dynamic-range imaging (HDRI) and image-based lighting (IBL) method. (a) The generated tree model and a real input image, (b) Editing using 3ds Max based on our compatible function, (c) Synthetic image using real background image and the our tree model.

The subsequent experiment examined the realism of the generated digital trees. Realistic expression is more important than rapid generation for digital trees. However, the viewer in the authoring system does not include a realistic rendering method, as the proposed system concentrates on modeling. Hence, tree models were imported into 3ds Max through the compatibility function using high dynamic range imaging, and the final virtual landscape images were created using the real image (Fig. 14). Fig. 15 shows the results of this operation. The produced tree model was first imported into 3ds Max and rendered using the authoring tool based on a method used in real image synthesis. A high-quality synthetic image was created using image-based lighting from a similar environment, and then merged into the real image. It was confirmed that realistic expression was achieved when the real tree image and the tree model produced using the authoring system were harmonized.

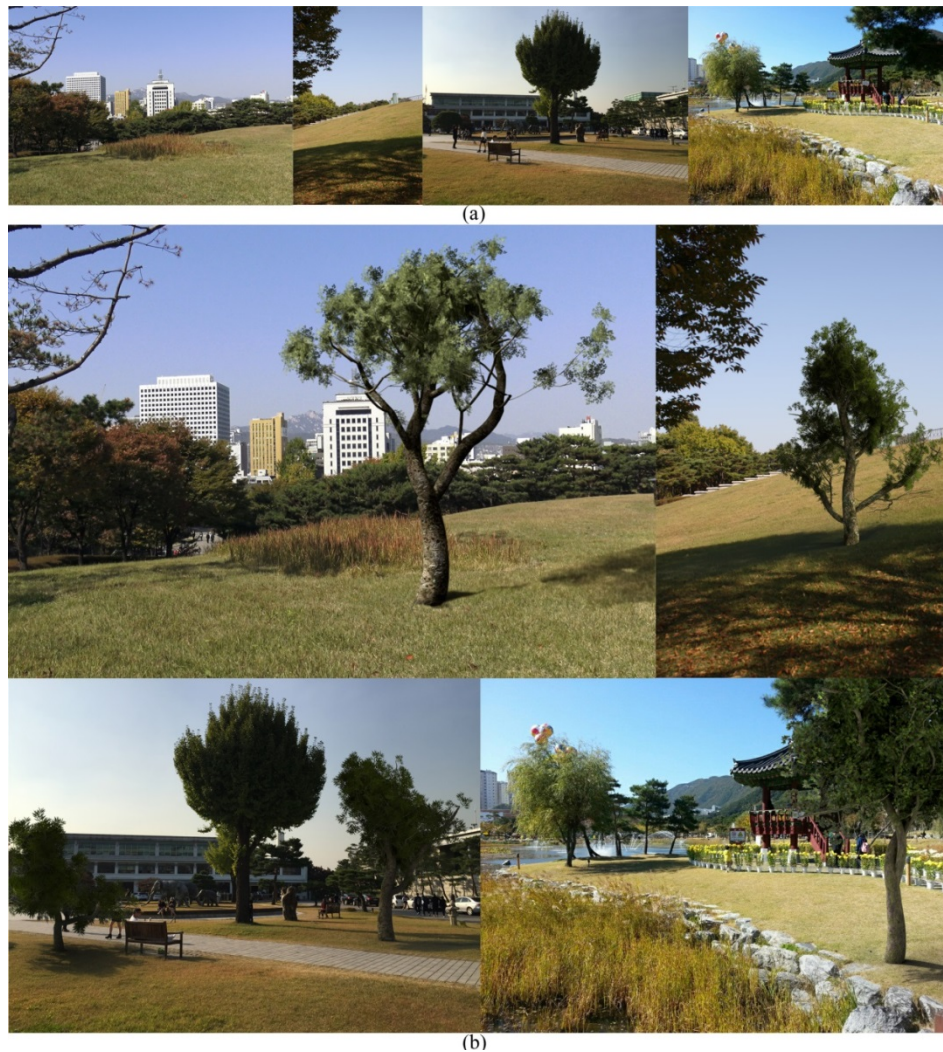


Fig. 15. Results of creating image synthesis for virtual landscape. (a) Real images taken with natural landscape as background, (b) The synthesis results of real images and tree models produced by the proposed authoring system.

Lastly, a comparative experiment with an existing study was performed. Kim et al. [29] designed an LOD method based on the procedural branch growth algorithm for effectively constructing an immersive virtual landscape. In order to design a user-oriented authoring system, the proposed growth model was simplified and an efficient growth pattern was designed based on a mathematical model before applying the LOD. Furthermore, the production of high-quality images was enabled through the application functions. Fig. 16 compares the results of the two studies, and a qualitative difference can be seen in similar tree models. In addition, tree models with similar shapes could be generated with fewer branches and polygons, since a growth pattern using a mathematical model was applied in this study. Therefore, the performance could be further improved when the LOD was applied. The comparison results in Table 3 show that it can be possible to express tree models with similar shapes having a smaller number of polygons. Furthermore, when the LOD was applied in the proposed system, the imaging rate had a value of 21.49 fps versus a value of 40.55 fps in the existing study involving 6000 trees, thereby confirming a higher real-time processing efficiency for the virtual landscape.



Fig. 16. Comparison results of the tree image generated by the proposed authoring system and the chosen existing study [29] (a) results of generating high quality image through the compatible functions of the proposed system (b) tree model results of existing study.

Table 3. Comparison results of the proposed authoring system and the chosen existing study [29]

LOD		Number of polygons	FPS(Number of trees)	
			3000 trees	6000 trees
Proposed System	Low	7,247	47.18fps	40.55fps
	Middle	13,252		
	High	25,433		
Existing System[29]	Low	11,917	30.12fps	21.49fps
	Middle	38,715		
	High	66,725		

5. Conclusion

This study has proposed an GeoMaTree system that can easily and intuitively produce diverse digital trees in an efficient structure for applications in real-time systems. The growth model was defined such that users can conveniently control the parameters of a procedural and hierarchical branch growth process. Moreover, a structure that can incorporate the simultaneous production of multiple trees that compose a virtual landscape is included. A

mathematical model based on the convolution sum of the divisor function was applied to effectively manage the complex branch propagation pattern, through which growth can be simultaneously realized in both the branch and the leaves. An application function was defined so that the digital tree produced by the proposed authoring system can be used effectively in real-time systems such as games as well as application fields that produce realistic images. A geometric model that is appropriate for tree structures was designed, and a multi-resolution 3D model was generated for use with the LOD method. The applicability of the proposed system to diverse 3D content fields was ensured by constructing a compatibility function that allows the digital image content to be exported to commercial game engines.

The proposed system accounts for the production of tree models only. Hence, future studies will involve not only the intuitive production of diverse digital plants, including leaves and flowers, but also the effective construction of large-scale virtual landscapes composed of such items. Moreover, we plan to conduct a statistical experiment to confirm whether general users can efficiently use the proposed authoring system.

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. NRF-2014R1A1A2055834, NRF-2017R1D1A1B03030286).

References

- [1] Lindenmayer, A., "Mathematical models for cellular interactions in development I. Filaments with one-sided inputs," *Journal of Theoretical Biology*, vol. 18, no. 3, pp. 280 – 299, March, 1968. [Article \(CrossRef Link\)](#).
- [2] Palubicki, W., Horel, K., Longay, S., Runions, A., Lane, B., Meřch, R., and Prusinkiewicz, P., "Self-organizing Tree Models for Image Synthesis," *ACM Trans. Graph.*, vol. 28, no. 3, 58:1–58:10, August, 2009. [Article \(CrossRef Link\)](#).
- [3] Boudon, F., Pradal, C., Cokelaer, T., Prusinkiewicz, P., and Godin, C., "L-Py: an L-System simulation framework for modeling plant development based on a dynamic language," *Frontiers in Plant Science*, vol. 3, no. 76, May, 2012. [Article \(CrossRef Link\)](#).
- [4] Zhang, D., Xie, N., Liang, S., and Jia, J., "3D tree skeletonization from multiple images based on PyrLK optical flow," *Pattern Recognition Letters*, vol. 76, no. 1, pp. 76, 49–58, June, 2015. [Article \(CrossRef Link\)](#).
- [5] Prusinkiewicz, P., Samavati, F., Smith, C., and Karwowski, R., "L-system description of subdivision curves," *International Journal of Shape Modeling*, vol. 9, no. 1, pp. 41–59, June, 2003. [Article \(CrossRef Link\)](#).
- [6] Karwowski, R., and Prusinkiewicz, P., "Design and Implementation of the L+C Modeling Language," *Electronic Notes in Theoretical Computer Science*, vol. 86, no. 2, pp. 134 – 152, September, 2003. [Article \(CrossRef Link\)](#).
- [7] Weber, J., and Penn, J., "Creation and Rendering of Realistic Trees," in *Proc. of Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques*, ACM: New York, NY, USA, SIGGRAPH '95, pp. 119–128, August 6-11, 1995. [Article \(CrossRef Link\)](#).
- [8] Prusinkiewicz, P., James, M., and Meřch, R., "Synthetic topiary," in *Proc. of Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, ACM: New York, NY, USA, SIGGRAPH '94, pp. 351–358, July 24-29, 1994. [Article \(CrossRef Link\)](#).
- [9] Takenaka, A., "A simulation model of tree architecture development based on growth response to local light environment," *Journal of Plant Research*, vol. 107, no. 3, pp. 321–330, September, 1994. [Article \(CrossRef Link\)](#).

- [10] Jirasek, C., Prusinkiewicz, P., and Moulia, B., "Integrating biomechanics into developmental plant models expressed using L-systems," in *Proc. of Plant biomechanics 2000, Proceedings of the 3rd Plant Biomechanics Conference*, Ch.; Speck, T., Eds.; Georg Thieme Verlag: Freiburg-Badenweiler, pp. 615–624, August 27–September 2, 2000. [Article \(CrossRef Link\)](#).
- [11] Mattingly, W.A., Chariker, J.H., Paris, R., Chang, D.j., and Pani, J.R., "3D Modeling of Branching Structures for Anatomical Instruction," *J. Vis. Lang. Comput.*, vol. 29, pp. 54–62, August, 2015. [Article \(CrossRef Link\)](#).
- [12] Runions, A., Lane, B., and Prusinkiewicz, P., "Modeling Trees with a Space Colonization Algorithm," in *Proc. of Proceedings of the Third Eurographics Conference on Natural Phenomena*, Eurographics Association: Aire-la-Ville, Switzerland, Switzerland, NPH'07, pp. 63–70, September 4, 2007. [Article \(CrossRef Link\)](#).
- [13] Kim, J., Kim, D., and Cho, H., "Procedural modeling of trees based on convolution sums of divisor functions for real-time virtual ecosystems," *Computer Animation and Virtual Worlds*, vol. 24, no. 3-4, pp. 237–246, May, 2013. [Article \(CrossRef Link\)](#).
- [14] Wang, Y., Xue, X., and Jin, X., "Creative Virtual Tree Modeling through Hierarchical Topology-preserving Blending," *IEEE Transaction on Visualization and Computer Graphics*, vol. 13, no. 12, December, 2017. [Article \(CrossRef Link\)](#).
- [15] Wang, Y., Wang, L., Deng, Z., and Jin, X., "Topologically consistent leafy tree morphing," *Computer Animation and Virtual Worlds*, vol. 28, no. 3-4, p.e1761, May, 2017. [Article \(CrossRef Link\)](#).
- [16] Zhu, X., Jin, X., and You, L., "High-quality tree structures modelling using local convolution surface approximation," *The Visual Computer: International Journal of Computer Graphics*, vol. 31, no. 1, pp.69-82, December, 2015. [Article \(CrossRef Link\)](#).
- [17] Alsweis, M., and Deussen, O., "Procedural Techniques for Simulating the Growth of Plant Leaves and Adapting Venation Patterns," in *Proc. of Proceedings of the 21st ACM Symposium on Virtual Reality Software and Technology*, Beijing, China, pp. 13-15, November 13-15, 2015. [Article \(CrossRef Link\)](#).
- [18] Chen, X., Neubert, B., Xu, Y.Q., Deussen, O., and Kang, S.B., "Sketch-based Tree Modeling Using Markov Random Field," *ACM Trans. Graph.*, vol. 27, no. 5, pp. 109:1–109:9, December, 2008. [Article \(CrossRef Link\)](#).
- [19] Li, C., Deussen, O., Song, Y.Z., Willis, P., and Hall, P., "Modeling and Generating Moving Trees from Video," *ACM Trans. Graph.*, vol. 30, no. 6, pp. 127:1–127:12, December, 2011. [Article \(CrossRef Link\)](#).
- [20] Tan, P., Fang, T., Xiao, J., Zhao, P., and Quan, L., "Single Image Tree Modeling," *ACM Trans. Graph.*, vol. 27, no. 5, pp. 108:1–108:7, December, 2008. [Article \(CrossRef Link\)](#).
- [21] Kim, Y., Woo, J., Kim, M., and Elber, G., "Interactive tree modeling and deformation with collision detection and avoidance," *Computer Animation and Virtual Worlds*, vol. 26, no. 3-4, pp. 423–432, May, 2015. [Article \(CrossRef Link\)](#).
- [22] Longay, S., Runions, A., Boudon, F., and Prusinkiewicz, P., "TreeSketch: Interactive Procedural Modeling of Trees on a Tablet," in *Proc. of Proceedings of the International Symposium on Sketch-Based Interfaces and Modeling*, Eurographics Association: Aire-la-Ville, Switzerland, Switzerland, SBIM '12, pp. 107–120, June 4-6, 2012. [Article \(CrossRef Link\)](#).
- [23] Bruneton, E., and Neyret, F., "Real-time Realistic Rendering and Lighting of Forests," *Computer Graphics Forum*, vol. 31, no. 2pt1, pp. 373–382. Special issue: Proceedings of Eurographics 2011, May, 2012. [Article \(CrossRef Link\)](#).
- [24] Qin, X., Nakamae, E., Tadamura, K., and Nagai, Y., "Fast Photo-Realistic Rendering of Trees in Daylight," in *Proc. of Computer Graphics Forum*, vol. 22, no. 3, pp. 243–252, September, 2003. [Article \(CrossRef Link\)](#).
- [25] Gumbau, J., Chover, M., Remolar, I., and Rebollo, C., "View-dependent pruning for real-time rendering of trees," *Computers & Graphics*, vol. 35, no. 2, pp. 364–374, April, 2011. [Article \(CrossRef Link\)](#).

- [26] Kratt, J., Cconum, L., Dapper, T., Schliep, J., Parr, P., and Deussen, O., “Adaptive Billboard Clouds for Botanical Tree Models,” *Proceedings of Digital Landscape Architecture 2014*, Wichmann Verlag: Berlin, Germany, pp. 274–282, May 21–23, 2014. [Article \(CrossRef Link\)](#)
- [27] Deussen, O., “A framework for geometry generation and rendering of plants with applications in landscape architecture,” *Landscape and Urban Planning*, vol. 64, no. 1–2, pp. 105–113, June, 2003. [Article \(CrossRef Link\)](#).
- [28] Behrendt, S., Colditz, C., Franzke, O., Kopf, J., and Deussen, O., “Realistic real-time rendering of landscapes using billboard clouds,” in *Proc. of Computer Graphics Forum*, vol. 24, no. 3, pp. 507–516, September, 2005. [Article \(CrossRef Link\)](#).
- [29] Kim, J., “Modeling and Optimization of a Tree Based on Virtual Reality for Immersive Virtual Landscape Generation,” *Symmetry*, vol. 8, no. 9, p. 93, September, 2016. [Article \(CrossRef Link\)](#).
- [30] Vanegas, C.A., Garcia-Dorado, I., Aliaga, D.G., Benes, B., and Waddell, P., “Inverse Design of Urban Procedural Models,” *ACM Trans. Graph.*, vol. 31, no. 6, pp. 168:1–168:11, November, 2012. [Article \(CrossRef Link\)](#).
- [31] Esri. CityEngine, Available online at [Article \(CrossRef Link\)](#), 2017.
- [32] Kim, J., and Cho, H., “Efficient modeling of numerous trees by introducing growth volume for real-time virtual ecosystems,” *Computer Animation and Virtual Worlds*, vol.23, no. 3–4, pp. 155–165, June, 2012. [Article \(CrossRef Link\)](#).
- [33] Unity3D. TreeEditor, Available online at [Article \(CrossRef Link\)](#), 2017.



Seowon Jung is Ph.D candidate in the Department of Multimedia at Dongguk University and CEO of Andromeda studio. She received B.E. degree in the Department of Game Multimedia from Dongguk University in 2011 and her M.E. degree in the Department of Multimedia from Dongguk University in 2013. Her research interests include computer graphics, virtual reality and real-time rendering.



Daeyeoul Kim is an assistant professor of Department of Mathematics and Institute of Pure and Applied Mathematics, Chonbuk National University. During 2007–2016, he worked department of Education & Culture, National Institute of Mathematical Sciences in Korea as a senior researcher. He received M.E. & Ph.D degrees from Chonbuk National University in 1994, 1998 respectively. His research interests include the graphic representation of ecosystem as a scenery of game, number theory.



Jinmo Kim is an assistant professor of Department of Software, Catholic University of Pusan. During 2012–2014, he worked Research Institute for Image & Cultural Content, Dongguk University as a full-time researcher. He received B.E. degree in Multimedia Engineering from Dongguk University in 2006 and his M.E. & Ph.D. degrees in the Department of Multimedia from Dongguk University in 2008, 2012 respectively. His research interests include computer graphics, VR/AR, procedural modeling and natural phenomena.