symmetry

MDPI

Article

Modeling and Optimization of a Tree Based on Virtual Reality for Immersive Virtual Landscape Generation

Jinmo Kim

Department of Software, Catholic University of Pusanm, Busan 46252, Korea;
jmkim11@cup.ac.kr; Tel.: +82-51-510-0645

Academic Editor: Angel Garrido
Received: 26 July 2016; Accepted: 30 August 2016; Published: 9 September 2016

Abstract: This study proposes a modeling method that can effectively generate multiple diverse digital trees for creating immersive virtual landscape based on virtual reality and an optimization method for real-time rendering. The proposed method simplifies a process of procedures from growth of tree models to the generation of the three-dimensional branch geometric model. Here, the procedural branch graph (PBG) algorithm is proposed, which simultaneously and effectively generates diverse trees that have a similar branch pattern. Moreover, the optimization method is designed in a polygon-based branch model which controls the resolution of tree models according to the distance from the camera to generate a tree model structure that is appropriate for an immersive system based on virtual reality. Finally, a virtual reality system is established based on the Oculus SDK (Software Development Kit) and Unity3D engine. In this process, the image processing-based pixel to tree (PTT) method is proposed as a technique for easily and efficiently generating a virtual landscape by allocating multiple trees on terrain. An immersive virtual landscape that has a stereoscopic perception and spatial impression is created through the proposed method and whether it can deliver experience of nature in virtual reality to the users was checked through an experiment.

Keywords: procedural modeling of tree; immersive virtual landscape; virtual reality; oculus HMD (Head Mounted Display)

1. Introduction

Virtual reality is a technology that enables users to come in and out of the boundary between reality and imagination by stimulating users' five senses in an imaginary environment or situation created by a computer and letting them have a spatial and temporal experience that is similar to reality. Recently, diverse virtual reality devices such as HMD (Head Mounted Display), motion input controller, and 360° camera have been popularized to general users, along with the advancement of computer graphics technology and hardware. As a result, a number of studies regarding virtual reality technology are under active progress in diverse fields including three-dimensional application graphics and digital contents such as games.

Studies related to virtual reality began from a study in the 1960s, which investigated an HMD system that delivers visual information with stereoscopic perception in virtual space to users [1]. Previously, there have been studies involving a tactile system for delivering material response of virtual world or hardware including a motion platform. Afterwards, a physical prop was used to enhance user immersion by overcoming the spatial limitation. Use of a physical prop in virtual space has been expanded from simply grabbing an object with the hands to description of actual walking movement of a human being [2,3]. In contrast with the technological studies regarding virtual reality technology from an aspect of experience and immersion, there have been studies that have analyzed motion sickness and psychological factors that interrupt user's immersion in virtual reality [4].

However, graphical factors consisting of the virtual space and methods for their modeling and rendering have been implemented based on the existing computer graphics theory. Here, optimization of scenes such as a limited number of polygons and lighting is important for the immersive virtual reality contents that are transferred through HMD, due to rendering through binocular disparity. If these are not satisfied, delay occurs between head tracking and rendering scene, which becomes a crucial factor that causes users' motion sickness. Moreover, this also causes occasions that are inappropriate for virtual reality contents. Nevertheless, there is no study aimed at immersive expression of virtual landscape composed of diverse multiple digital plants such as trees and flowers by using virtual reality technique, since it has to consider not only the realistic expression of digital plants, but also the optimization process. In particular, expressing diverse digital trees in virtual reality-based contents is very difficult due to the uniqueness and complexity of the structure in the tree models.

Most of the studies related to computer graphics of digital trees were aimed at realistic and efficient expression of the surface and shape of a digital tree, which has a complex structure. Rule-based modeling using the L-system is the representative study where the composition factor and growth parameters of digital plants are defined as a string and plants are procedurally created by substituting status through generation grammar in a recursive structure [5]. The L-system was mainly studied from a mathematical approach in applied mathematics field in the early days of research in this area. After converging with geometric modeling and computer graphic theory, it has been applied as a method for generating realistic trees [6]. Afterwards, the generation of realistic digital trees was studied through application with diverse technologies and research fields, including botanical approach and computer vision [7–9].

Most of these studies do not account for the vertex that composes the geometric model of digital tree or its complexity as they are focused on the realistic expression of digital trees. As a result, existing studies related to tree modeling have clear limitations when applied to the generation of a virtual landscape based on virtual reality. Hence, this study proposes a virtual reality-based tree modeling and optimization method to address this problem and to create an immersive virtual landscape. First, the growth rule in procedural structure is defined to effectively generate multiple diverse tree models that compose a virtual landscape. This is a method that procedurally implements realistic growth of digital trees based on one growth rule and simultaneously generates multiple tree models that have similar shape. In contrast to the existing simple method that duplicates a single tree model and allocates multiple tree models that compose a virtual landscape, this method can allocate tree models that differ with each other so that the generated virtual landscape can be similar to the actual natural environment.

This paper also designs an optimization method to generate an efficient tree model structure that is appropriate for the implementation of immersive virtual landscape based on virtual reality. Here, a method that simply decreases the number of polygon in tree models can make the system stably run. However, naturalness is very low in this case and this becomes an important factor that decreases user's immersion. Hence, this study designs a method that can maintain the naturalness of virtual landscape while increasing the efficiency of the system. This is a method that optimizes the polygon model in the process of modeling through the procedural growth rule and automatically generates tree models by making them level by using diverse resolution. For this purpose, this study proposes geometric modeling method that determines tree shape through the LOD (Level of Detail)-based growth rule. Finally, this paper conducts an experiment that verifies the efficiency and performance of the proposed method by actually creating immersive virtual landscape. In this study, virtual reality contents were generated by using Oculus Rift DK2 HMD where Unity3D engine was used. Hence, LOD method under Unity3D environment is also suggested so that the proposed LOD method for the tree models optimization can be compatible with the LOD function of Unity3D.

Section 2 describes studies related to digital plants that are required for the virtual reality technique and virtual landscape generation. Section 3 explains about the multiple diverse tree modeling and optimization method for the immersive virtual landscape based on virtual reality. Section 4 describes

a method of actually implementing immersive virtual landscape through the proposed method. Section 5 analyzes efficiency and usability of the proposed method through experiment. Section 6 presents conclusion of this study and direction for future research.

2. Related Works

2.1. Virtual Reality

To develop interactive contents with three-dimensional virtual space by using virtual reality techniques, a virtual environment and technique should be studied that can provides experiences that are identical to the actual reality by directly stimulating users' five senses. The first factor that should be considered to satisfy user's spatial impression is visual sense. In the 1960s, Sutherland [1] studied an HMD system where users can experience visual stereoscopic perception in virtual space. Afterward, diverse approaches and studies began regarding the virtual reality techniques. For the purpose of providing users with spatial impression, studies involving CAVE (Cave Automatic Virtual Environment) also progressed, which provides immersive virtual reality environment by creating room-size cubic space and reflecting image information that corresponds to the location and sight of the user to each side of a cube by using a projector [10,11]. Through this, users' activity range in the virtual space broadens, which consequently increased the user immersion. In addition to this, realism was enhanced by satisfying user's auditory experience in the space through an application of stereo or surround audio [12].

Tracking-based display system and audio system that transfer image information that corresponds to user's view and location in virtual environment are the most basic and important information that enhances user immersion. However, if the environment provided to users is limited to simply viewing and listening, the usable range can be extremely narrow. Hence, experiences obtained from interaction with virtual objects that exist in virtual space are also important factors of virtual reality. For this, there have been studies that enhance immersion by enabling users to directly touch and control virtual objects by using data input glove and to feel tactility through sensors [13–16]. Moreover, a motion platform or mechanical machineries such as exoskeletons were used to make the virtual space liberally move without spatial limitation by analyzing user's joint information or detecting movement as in motion capture device [2,17–19]. As the problem of spatial limitation was resolved, physical props became important factor that recreates virtual space. Simple motions such as grabbing a doorknob to open the door or pushing wall and diverse motions such as grabbing a virtual object and moving it to the desired location were implemented [17,20]. Cheng et al. [21] proposed TurkDeck, which is a human-oriented physical virtual reality system aimed at provision of new virtual environment to users every time. In their study, virtual props such as wall and room are recreated by human actuator in real-time according to the information of user's actual movement in reality. User immersion was maximized by considering not only simple props such as a wall and door, but also diverse physical objects and effects including ledges, steps, switches, portals, and wind. In addition, Li et al. [22] reflected change of expression by detecting user's face movement with a sensor attached to HMD and reflecting it on the virtual reality. In their study, change of expression was detected through a head-mounted RGB-D camera and movement of mouth was particularly computed in detail to be applied to the virtual character. However, most of the virtual reality-related studies were technical studies that involve hardware that can enhance user experience and immersion. Despite the recent situation where virtual reality devices are popularized, there are few approaches from software that can diversify the environment of user experience.

2.2. Procedural Modeling of Trees

This study attempts to provide users with new experience by constructing immersive virtual landscape that can diversify user's virtual reality experience environment. For this, multiple diverse digital trees that compose virtual landscape should be provided in optimized structure to fit the virtual

reality system. L-system proposed by Lindenmayer [5] is the representative study related to digital trees. This is a formal grammar which is an algorithm that defines growth process of plants in recursive structure. There was also a study regarding the parametric L-system that generates realistic digital plants by structuralizing parameters that are necessary for realistic growth of plants. There have been diverse application studies based on L-system including development of simulation framework where users can create desired plants in an intuitive structure [6,23]. In contrast, there were studies from a botanical approach such as self-organizing methods that generate plants by defining an algorithm of the factors that are necessary for the growth of plants including hormone and light [7,24,25]. However, the existing procedural modeling methods that are related to digital tree generation are mainly focused on the realistic expression of the shape of three-dimensional tree models that have complex structure. Hence, they do not consider editing and controlling process by users.

Afterwards, there have been studies aimed at applicability to actual contents production by making the tree models easier and intuitive, in addition to the academic perspective. For example, there was a sketch-based study where similar tree models are automatically generated by referring to the previously tree model template based on the user's input of simple branch structure of the desired tree model in sketch form. This was developed into a table interface where users can generated three-dimensional trees in interactive structure as if drawing [26–28]. Other than this, similar three-dimensional trees were automatically reconstructed by using computer vision technique when digital trees photo images are put in or three-dimensional tree model animation was expressed through digital tree video that reflect movements [9,29]. However, although this can more easily and intuitively generate a one-tree model than before, generating multiple diverse models that compose a virtual landscape one by one is practically impossible. A more important point is that digital tree models generated by the existing methods do not consider the complexity of tree models as a realistic expression that has the priority. In this case, it is difficult to apply to a virtual reality system where the number of polygons should be dealt with more carefully than the existing three-dimensional digital contents. Hence, this study proposes a virtual reality-based tree modeling and optimization method for the creation of immersive virtual landscape.

Studies related to tree modeling are used to guide the development of virtual landscapes composed of multiple trees. For effective generation of virtual ecosystems composed of plants such as trees, conversational modeling [30], realistic modeling, and rendering methods [31] have been studied. To aid creation of realistic ecosystems, some studies have examined the inclusion of external environmental factors such as light and physical obstacles in virtual landscape designs [24,32]. Kim et al. [33] designed a virtual landscape composed of multiple trees by using a mathematical model that yielded realism with real-time processing. However, since they do not fully account for the virtual reality environment, studies such as this do not solve the problem of motion sickness, which users may suffer in immersive virtual landscapes.

3. Modeling of Multiple Trees

Multiple diverse trees consisting of virtual landscape should be rapidly, effectively and realistically generated for the production of immersive virtual landscape. Moreover, the optimization process of the generated tree models is necessary for a real-time virtual landscape that is appropriate for the virtual reality. This study proposes a procedural branch graph (PBG) algorithm which simultaneously and effectively generates diverse trees with a similar branch pattern in a process of geometrically generating three-dimensional tree models where a single tree grows. Additionally, an optimization method for the tree model with highest complexity in the virtual landscape is proposed to maintain a graphics state required for the virtual reality contents (Figure 1).

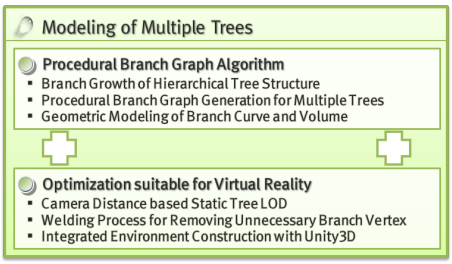


Figure 1. Overview of the proposed multiple tree modeling system.

3.1. Procedural Branch Graph Algorithm

Tree modeling method begins by defining a generalized rule for the process of procedural growth of branches that compose trees. Generalizing the process of growth from the current branch to the next branch can make the generation of tree models in a complex branch structure effective. This study designs a PBG algorithm that can simultaneously create multiple similar tree models and effectively create diverse tree models based on the tree structure proposed by Kim et al. [34].

First, parameters necessary for the growth of branch are defined. The start point is to determine the growth direction of the current branch (v_g^i) including the length (l_i). Here, branches have an overall hierarchy structure as the current branch is derived from the previously created parent branch. Hence, growth direction of the current branch is influenced by the growth direction of parent branch (v_{pr}). Current branches are largely divided into two types: an expansion of the parent branch and beginning of a new bud. If the current branch is expanded from the parent branch, its growth direction is the same as the previous direction of the growth of parent branch. However, if it is a newly growing branch from a bud, new growth direction should be computed using the growth direction of parent branch as a baseline (Figure 2a). Hence, the status of the current branch is defined by dividing into two.

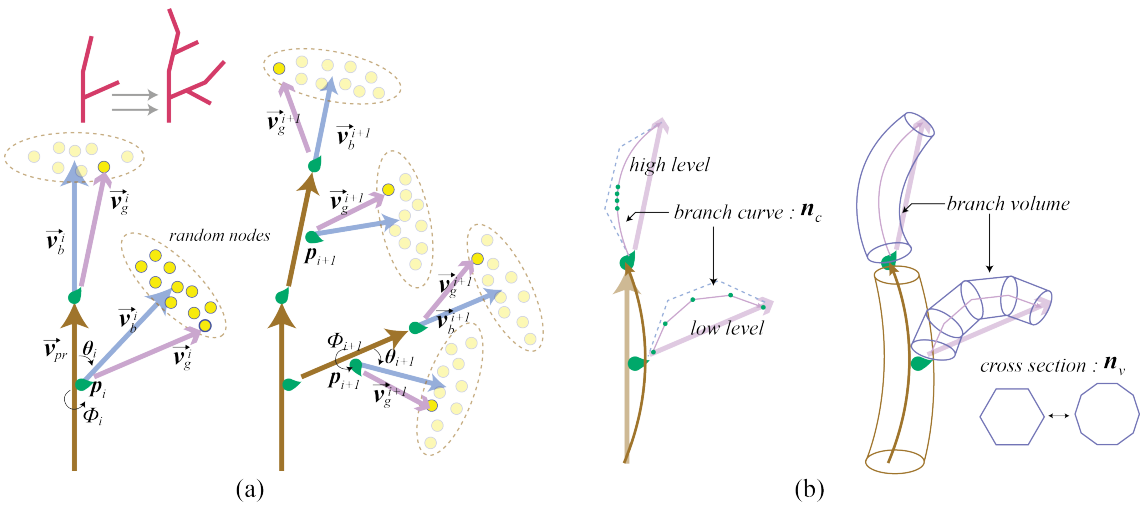


Figure 2. Procedural branch graph(PBG) algorithm process: (**a**) Branch growth in hierarchy structure for the calculation of branch graph; (**b**) Geometric modeling that composes branch curve and volume.

First, if the current branch is an expansion of the parent branch, basis vector (v_b^i) of the current branch is the same as the growth of parent branch vector (v_{pr}). Second, if current branch is a beginning of a new bud, location of the bud (p_i) is first determined and basis vector (v_b^i) is computed by applying orientation and rotation component (θ_i, ϕ_i) to the growth direction of parent branch. Once the basis vector is defined, growth process is identical. After multiplying length component of branch (l_i) with normalized basis vector, random nodes are allocated at the destination point of the basis vector. Then, one of the random nodes is selected. Branch growth is achieved by drawing a branch graph that

connects the selected nodes. Figure 2a shows the procedural growth of a branch where i-th node is connected to form a graph, based on which the branch graph is linked to $i+1$st node. Algorithm 1 summarizes this. Branch pattern differs every time according to which node is chosen at the current step. Hence, the irregularity of branch growth can be expressed.

Algorithm 1 Procedural Branch Graph.

$G_{\text{tree}} \leftarrow$ set of branch graph nodes.
$v_{pr} \leftarrow$ growth direction of parent branch.
$p_{pr} \leftarrow$ start node of parent branch.
procedure CALCULATION OF A NEW BRANCH GRAPH(G_{tree})
    $p_i \leftarrow$ start node of i-th current branch.
    $v_b^i \leftarrow$ i-th basis vector.
    $v_g^i \leftarrow$ growth vector of i-th current branch.
    setting of growth parameters (θ_i, ϕ_i, l_i)
    $S_{\text{nodes}} \leftarrow$ set of random nodes.
    $p_n \leftarrow$ node of selected branch graph.
    type $\leftarrow$ type of the current branch out of two types.
    if type = expansion of parent branch **then**
        $v_b^i = v_{pr}$ (direction of basis vector and growth direction of parent branch are identical.)
        computation of p_i from p_{pr} and v_{pr}.)
    else if type = growth from a new bud **then**
        $v_b^i = R(\phi_i)R(\theta_i)v_{pr}$ (application of rotation component to the growth direction of parent branch.)
        computation of a new bud location p_i from p_{pr} and v_{pr}.
    end if
    calculation of S_{nodes} by using p_i and v_b^i.
    $p_n \in S_{\text{nodes}}$: selection of one random point from S_{nodes} set.
    $v_g^i = p_n - p_i$: determination of growth direction of i-th current branch.
    store p_n in G_{tree}.
end procedure

After calculating the branch graph that expresses the overall structure of trees, the geometric modeling method for expressing the volume and curve of branches was designed. Three-dimensional branch modeling is a geometric modeling method using Bézier curve. First, nodes that compose the branch graph are set as each control point. Then, random control points are calculated between node and node. In addition, the Bézier curve is calculated by using a control point. Through this, an irregular and smooth branch curve can be expressed. To express volume to the branch curve, a vector that is vertical to the curve segment is computed. By adding vertices while rotating the vector by 360°, cylindrical branch shape is generated. Figure 2b shows that: the branch curve and volume that fit the current graph are automatically computed once the branch graph is procedurally calculated. Algorithm 2 summarizes this.

The final step is sequencing of a branch graph for the effective generation of multiple trees, which is the core goal of this study. In the process of determining the node of branch graph, the set of random nodes (S_{nodes}) is a factor that gives irregularity of branches. Moreover, selecting more than one random node and lining it by sequencing when determining the growth direction of the current branch can make it possible to simultaneously generate multiple trees that have similar growth pattern. That is, if the user selects the number of trees that are to be simultaneously generated, similar trees of a total number of trees (n_{total}) can be simultaneously generated by giving weight as the growth step (n_{gr}) increases. This is expressed in Equation (1). Figure 3 descries the sequencing process of the branch graph.

$$n_{\text{total}} = \sum_{i=1}^{n_{gr}} \varpi_i * (n_{seq}^i)^i \tag{1}$$

Algorithm 2 Branch Geometric Modeling.

$n_c \leftarrow$ segment step of branch curve.
$n_v \leftarrow$ number of polygon vertex in branch side.
procedure CALCULATION BRANCH SURFACE
    calculation of two random control points within two nodes of p_i and $p_i + v_g^i$.
    computation of Bézier curve based on four control points.
    calculation of segments of Bézier curve as much as n_c.
    for $i \leftarrow 1, n_c$ **do**
        setting of branch volume of width.
        calculation of vector that is vertical to v_g^i through Gram-Schmidt normalization by using v_g^i and v_b^i as basis.
        scalar product of vertical vector and the previously set width.
        for $j \leftarrow 1, n_v$ **do**
            $\alpha = (2 * \pi * j)/(n_v - 1)$: computation of rotation angle.
            calculation of vertex position by rotating vertical vector by an angle of α.
        end for
    end for
end procedure

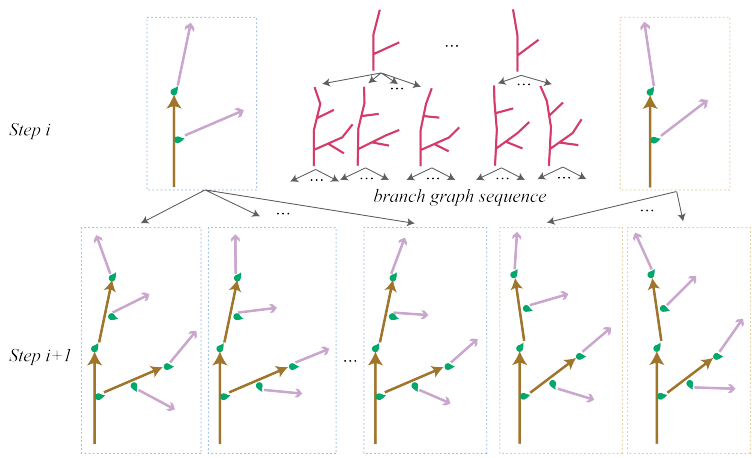


Figure 3. Automatic calculation of multiple tree growth pattern through sequencing of branch graph.

The PBG algorithm can generate diverse tree models by effectively controlling the irregular branch growth process with a small number of parameters from a tree model. Moreover, the algorithm has an advantage in that it can simultaneously generate multiple tree models that have similar shape by sequencing branch graphs.

3.2. Optimization Suitable for Virtual Reality

When generating trees by using the PBG algorithm, virtual landscape can be more easily and efficiently created through an effective generation of multiple diverse trees. However, the polygon of the branch model should be more seriously considered in case of application in the immersive system based on virtual reality. That is, whether to increase the number of vertex and polygon for the expression of smooth and realistic branch surface in the process of composing curve and volume of branch through Algorithm 2 determines its appropriateness for the virtual reality system. Decreasing the number of polygons can interrupt user immersion as the realism of tree models decrease and the overall quality of virtual landscape diminishes. In contrast, increasing the number of polygons can induce motion sickness due to delay between head tracking information, that is, user's gaze, and the actually rendered image because of complex computation in the rendering process. In this case, LOD method is generally used in case of real-time system such as games. This study proposes LOD-based optimization method that is capable of controlled rendering of polygon in the branch model that was generated by PBG algorithm according to the distance from camera.

LOD is a method whose expression generally simplifies models that have small contribution to the rendering image from an aspect that distant objects are not easy to see. There is static LOD method which computes and stores subset vertex data that correspond to LOD step in advance and uses model that fits the LOD step from the subset vertex data. There is a dynamic LOD method which computes and renders vertex data according to the current LOD step value. As for the method for dividing LOD steps, there is a method that is based on the distance between camera and model and the area-based method that sets the LOD step by using an area that was projected on a screen or bounding box.

When considering the virtual reality environment, dynamic LOD has high computation load and has a potential risk of bottle neck as device data pool should be locked and unlocked at every update. As this can be an important cause of motion sickness, it is inappropriate for a virtual reality system. Meanwhile, static LOD is capable of efficient processing as long as there is enough data storage space. Since a virtual reality-based system basically requires high specifications, there is no problem of applying static LOD. Hence, this study designs static LOD that is appropriate for tree models. As for the LOD step, distance-based LOD was used considering the small load on CPU and the compatibility with Unity3D that will be used later for the production of immersive virtual landscape contents.

The branch model generated by Algorithm 2 of PBG has geometric structure as shown in Figure 2b. Here, the number of vertex in the branch polygon model is determined by the parameters (n_c, n_v) that determine curve and volume of branch. Equation (2) expresses the process of determining current branch polygon number in one stage according to the curve and volume of branch. One branch is differentiated with curves in n_c step and it is composed of n_v number of volume vertex in each step.

$$n_{br} = n_c * n_v \tag{2}$$

Equation (3) expresses the method of calculation index information according to the curve and volume step based on the vertex coordinate of branch model calculated by Algorithm 2. Tree shape is modeled as partial branch models that are connected and are finally composed of the n_c step. However, the identical vertex can be redundantly created at a point where segments are connected in this process, as shown in Figure 4. Hence, an unnecessary vertex is not created through welding in this part.

$$\begin{aligned} &\text{Indices}(n_v * 6 * i + j) = n_v * 2 + 2 * i + k_j \\ &\text{where,} 1 \leq i \leq n_v, 0 \leq j \leq 5, k \in \{0, 1, 2, 1, 3, 2\} \end{aligned} \tag{3}$$

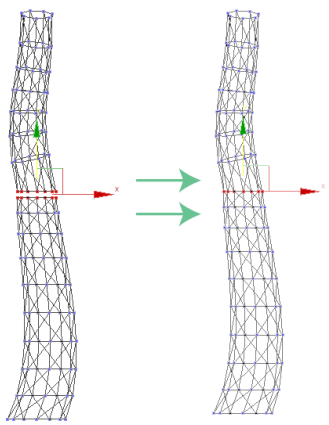


Figure 4. Welding process for the optimization of branch polygon model.

Parameters n_c and n_v not only basically have information required for the generation of geometric model, but are also important control variables for LOD. As shown in Equation (2), the parameters determine the number of vertexes of the current branch and eventually determine the overall resolution

of the tree model. Hence, tree models with resolution that is appropriate for the step are made to be automatically generated by substituting the LOD step defined in this study into the parameters. Once parameters n_c and n_v are defined, the vertex and index that compose the branch model are automatically computed. Hence, static object that is used in distance-based LOD in a process of generating a tree model can be simultaneously and automatically generated as much as users want. Moreover, the generated tree models also have no problem in maintaining naturalness due to identical shape as it uses a branch graph that has the same growth direction. Figure 5 shows the results of simultaneously creating tree models by dividing it into four steps from high resolution to low resolution through a control of parameters. In this study, Unity3D was used as a tool for producing immersive virtual landscape. Hence, the proposed optimization method should be directly applicable to the LOD function provided in Unity3D. Figure 5 shows LOD function in Unity3D, where registration of automatically generated tree models that have four step resolution in prefab can be promptly applicable to LOD.

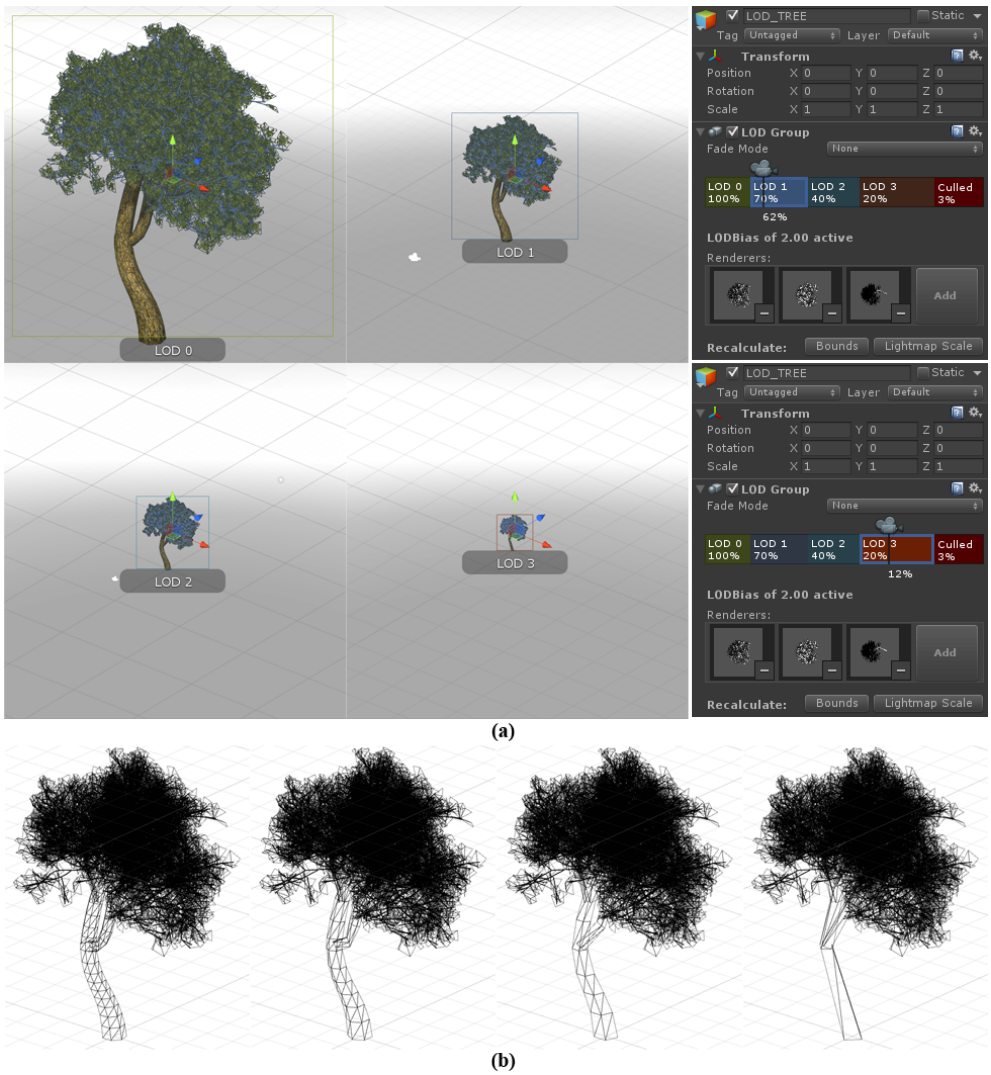


Figure 5. LOD (Level of Detail)-based tree optimization: (**a**) Process of applying the proposed LOD method to LOD function of Uinty3D; (**b**) Results of four step tree models generated by the proposed LOD method.

4. Immersive Virtual Landscape

It is necessary to first satisfy the visual sense to provide users with a natural virtual landscape composed of multiple trees in a virtual reality environment that has immersion. For this, scenes of virtual space are transferred to users by using Oculus HMD in this study. As a first step to specifically create virtual landscape, multiple digital trees that compose the landscape and terrain that contains all of them are necessary. Additionally, an optimized multiple diverse tree model that was previously proposed should be rendered in real-time. Hence, this study makes use of Unity3D that can integrate all of Oculus HMD, GPU processing, and create an interactive virtual landscape.

As the first step in creating an immersive virtual landscape, creation of a scene with stereoscopic perception of binocular disparity, user's head tracking, and user's motion in the virtual landscape are simultaneously processed by importing Oculus Development Kit to Unity3D engine. The next step is to create realistic terrain in an intuitive structure by using the terrain editor of Unity3D. Here, the generated tree models that were manufactured in this study are allocated on the generated terrain. Under the structure of the existing method of Unity3D, users should directly allocate trees on terrain one by one. Using this method for creating a forest that is composed of hundreds and thousands of trees is very inefficient. Hence, this study proposes the Pixel to Tree (PTT) allocation technique by using image processing technique that uses OpenCV on height map, which is provided in terrain editor of Unity3D. This is a method that allocates trees in accordance with the characteristics of the terrain by considering high and low like ridge or state of the terrain. Height map is an image that expresses height information consisting of colors between 0 and 255 in accordance with the minimum and maximum height of the current terrain. This makes the overall appearance of terrain intuitively understandable. Hence, PTT method searches for the location of a tree that fits the terrain through the height map-based 3 step image processing.

1. Edge and contour information of height map image is detected.
2. Location of tree is grouped by using edge and contour as basis.
3. Pixel within threshold value and height within area are determined as one tree location.

The area of edge and contour in terrain can be grouped according to the height and boundary. Here, the threshold value is set such that trees that are too high or a narrow area where trees cannot be allocated are excluded from the tree allocation. Here, threshold value is determined according to the resolution of the height map. Then, one pixel in the contour area of the finally set image is determined as the location of the tree. Here, distance between trees is controlled by selecting one pixel between 3×3 or 5×5 window pixel to manage the tree density. Multiple trees appropriate for the terrain can be automatically allocated through this process. Here, the number of trees that composes virtual landscape is automatically determined according to the resolution of the height map. For example, a height map of 100×100 resolution can have up to 10,000 trees as it has 10,000 pixels. Allocation of up to 400 trees is possible when applying a 5×5 window pixel as one tree is created for every 25 pixels. Through this, a number of trees that fit the scale of the system can also be conveniently controlled. Figure 6 shows a process of determining tree location by using the PTT method based on a height map created in Unity3D terrain editor.

Section 5 confirms that virtual landscape can be efficiently created by using the proposed PTT method in virtual reality environment using Oculus HMD through an experiment.

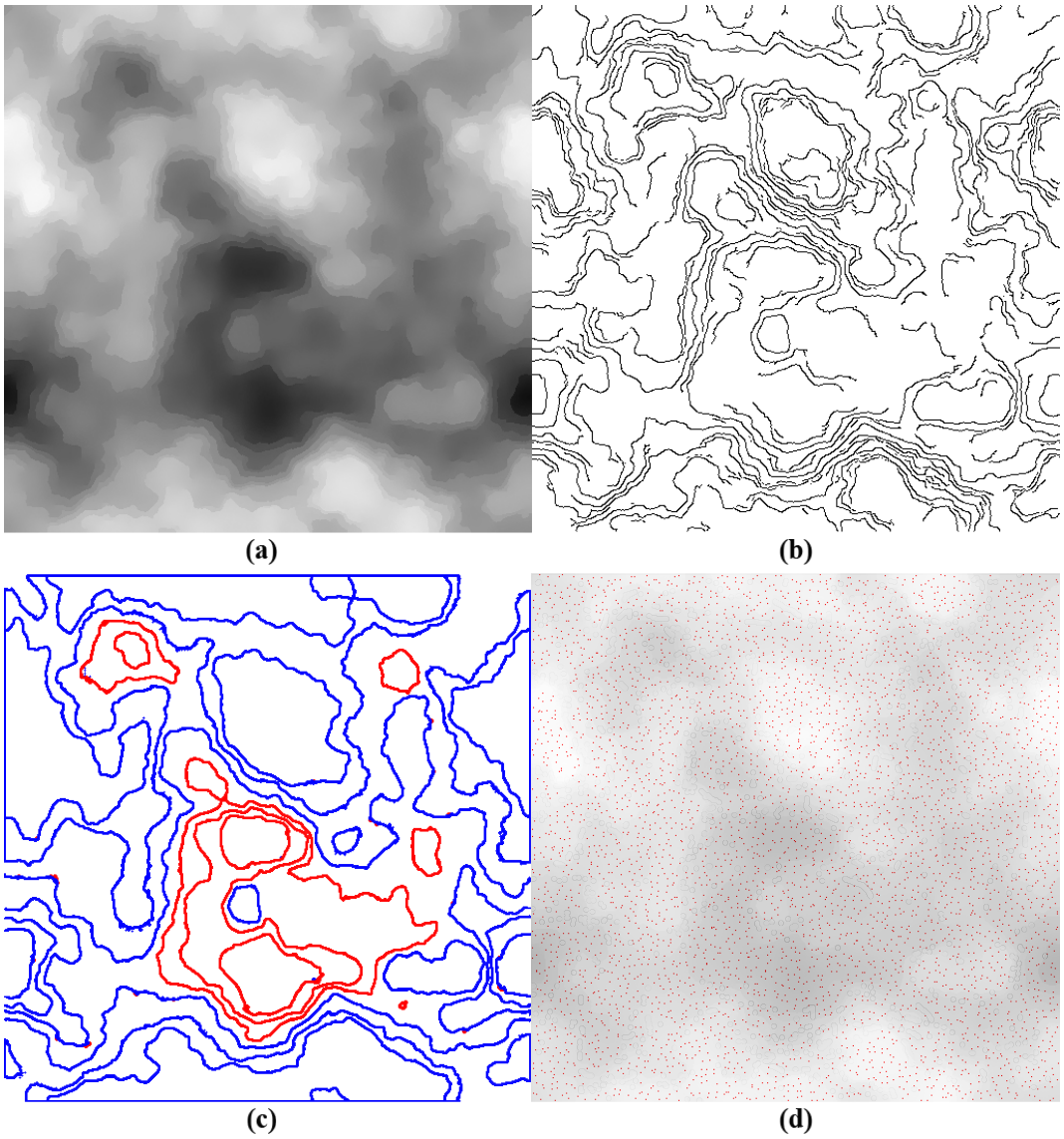


Figure 6. Process of determining the location of tree according to the terrain characteristics through PTT (Pixel To Tree) method: (**a**) Height map of virtual landscape; (**b**) Edge detection of terrain map; (**c**) Area division of terrain based on contour; (**d**) Determination of the location of tree by using 5 × 5 window pixel.

5. Experimental Results and Analysis

The proposed tree modeling and optimization method and the creation of immersive virtual landscape contents were implemented by using Visual Studio 2013, MFC, DirectX SDK, Unity3D 5.3, Oculus SDK. PC environment used for the experiment had built-in Intel(R) core^{TM} i7-4790, 8GB RAM, Geforce GTX 960 GPU.

The experiment was processed by being largely divided into three steps. First, the experiment checks whether multiple similar tree models can be simultaneously generated by the proposed tree modeling method. Multiple branch graphs were calculated through sequencing in PBG algorithm and similar tree models were automatically and simultaneously automatically generated. Figure 7 shows the results of automatic generation of similar tree models by the sequenced branch graph through the previously set growth parameters. In the results, we could confirm that multiple diverse tree models that compose virtual landscape were efficiently created.

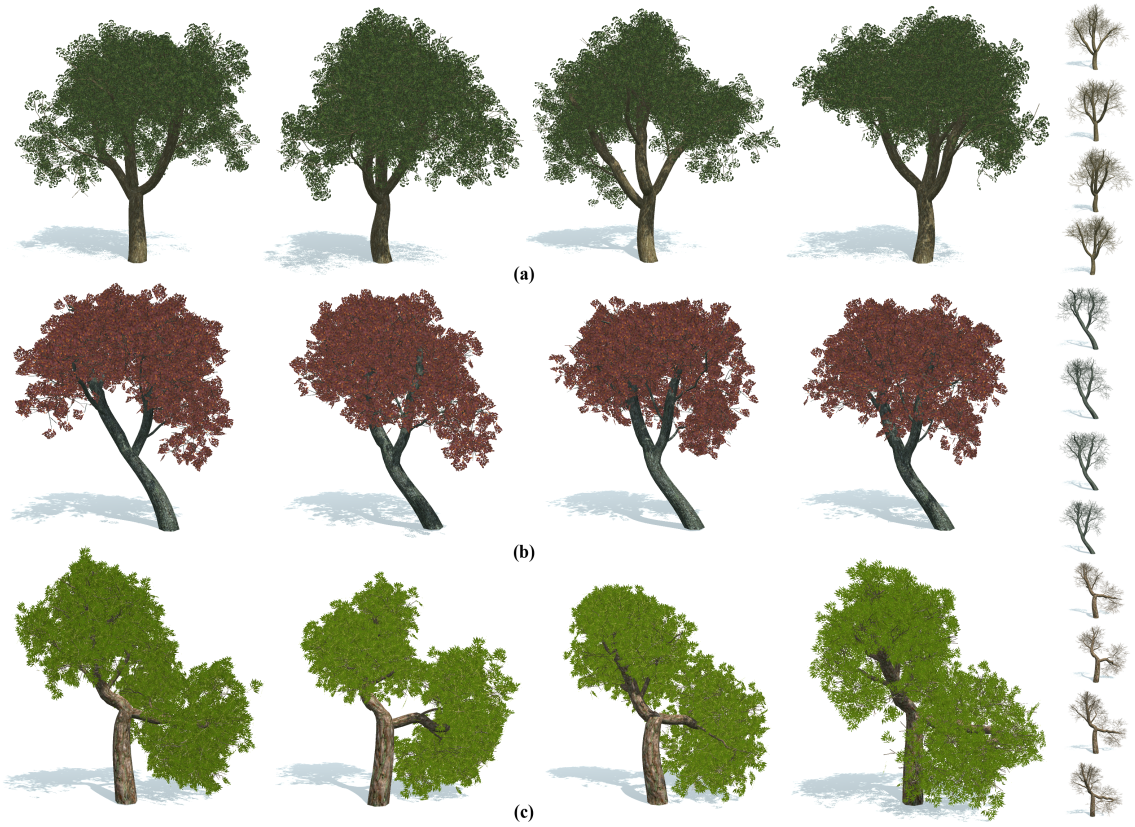


Figure 7. Results of generating multiple diverse tree models through the proposed PBG (Procedural Branch Graph) algorithm $(\theta, length, width)$: (**a**) $(60, 10, 9)$; (**b**) $(70, 12, 7)$; (**c**) $(80, 7, 10)$.

There follows an experiment on the optimization method of tree models. In general, a 90 hz refresh rate should be maintained or frame rate per second should be at least 75 fps in case of Oculus-based virtual reality contents. Users can experience virtual reality contents without feeling motion sickness only if these conditions are satisfied. Hence, tree models were created by setting four-step LOD according to the distance from the camera, as previously shown in Figure 5 to check whether the proposed method can be used in a virtual reality environment. Then, trees that fit the terrain were automatically created and allocated through PTT method. Here, the results of rendering without LOD application and the results of composition of prefab with LOD application were measured. Table 1 shows the results. Here, the total number of trees of virtual landscape in the same terrain was controlled by the resolution of the height map. The results were compared between a case of rendering without LOD and a case of rendering with LOD prefab application. We could find out how much the optimization method in this study can improve performance of the immersive virtual landscape based on virtual reality. A procedural tree modeling method has been developed based on the mathematical model proposed by Kim et al. [33] and has an appropriate structure for real-time processing of multiple trees in virtual landscapes. We verified the performance of the method by conducting a comparison experiment between the proposed LOD method and existing studies. Table 2 presents the comparison results, which suggest the LOD method that covers the virtual reality system performs better than do the methods reported in existing studies, which considered only real-time processes. Existing studies present no change in tree models following movement of the camera. In the LOD method, tree modeling is simplified according to the distance from the camera, which can result in a relatively less natural-looking outcome.

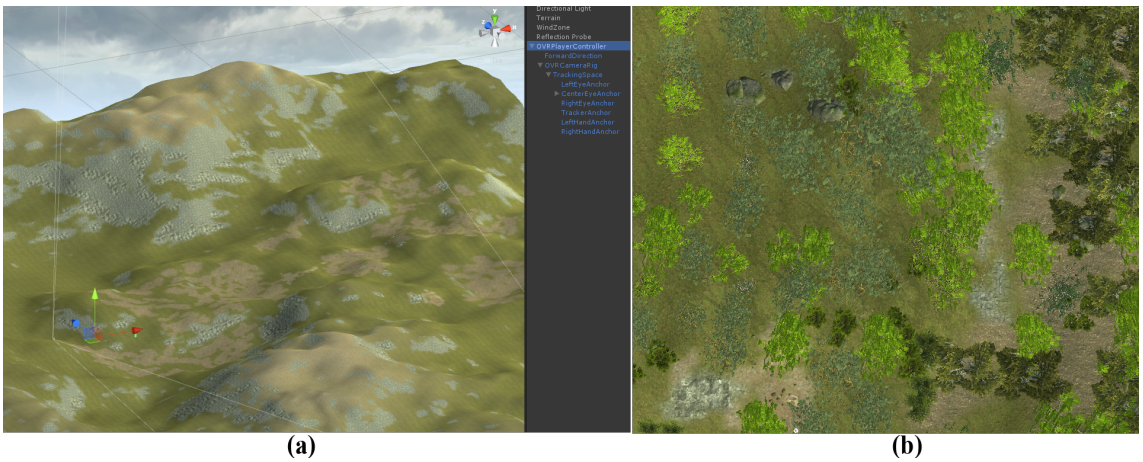
Table 1. Experiment of comparing the efficiency of the virtual reality system through the proposed LOD (Level of Detail)-based optimization method.

Resolution	Number of Trees	FPS (Frame Per Second)	
		Before Applying LOD	Applying LOD
256 × 256	1052	12.12	80.12
512 × 512	3958	5.09	73.44

Table 2. Experiment of comparing the creation of virtual landscape with the existing studies.

Number of Trees	Growth Step	FPS	
		Kim [33]	Our Method
1500	6	34.20	77.28
2000	5	24.17	76.11

The final step is an experiment in creating contents for immersive virtual landscape based on virtual reality. Immersive virtual landscape was composed through the proposed method and an environment was implemented that users can liberally experience by wearing HMD. As shown in Figure 8, multiple tree models that compose virtual landscape were created through the proposed method and expression of highly immersive virtual landscape was confirmed by additionally allocating diverse forest props. Then, we randomly selected 15 participants to compare created immersive virtual landscapes and respond to a survey comprised of two parts. The first part asked respondents to compare immersion into the natural background of the created virtual landscape presented on an ordinary display monitor with its presentation via a HMD. Figure 9a shows users' reported sense of immersion, with 5 as the maximum score. These results indicate the immersive virtual landscape presented in the HMD gave users a more realistic experience and fuller sense immersion into nature than did the monitor display. The second part of the survey required respondents to compare the impact of existing studies [33] and LOD optimization on their experience of motion sickness. Figure 9b illustrates perceived motion sickness, with 5 indicating the strongest feeling. Comparing motion sickness existing studies and the application of LOD technique revealed remarkable differences. In particular, the use of LOD with a virtual landscape composed of approximately 1000 trees rarely caused user motion sickness.



(a) (b)

Figure 8. *Cont.*

Figure 8. Results of contents creation of immersive virtual landscape based on virtual reality: (**a**) Unity3D terrain development environment with Oculus SDK (Software Development Kit); (**b**) Allocation of multiple tree models on terrain; (**c**) Results of virtual landscape creation; (**d**) Oculus rift DK2 HMD (Head Mounted Display); (**e**) Test environments construction using Oculus HMD.

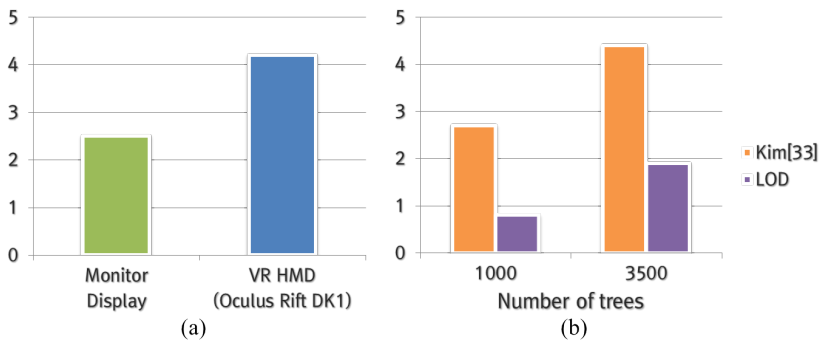


Figure 9. Results of the survey assessing users' perceived immersion and motion sickness in comparisons of two immersive virtual landscape conditions: (**a**) Comparison of users' sense of immersion when viewing the landscape on two different displays, monitor and HMD; (**b**) Comparison of the impact of the LOD-based optimization on users' reported motion sickness.

6. Conclusions

This study proposed a modeling and optimization method for generating diverse multiple tree models in an efficient structure for the immersive virtual landscape based on virtual reality. For this, the PBG algorithm was first designed. The PBG algorithm calculated a branch graph by using the information required for the growth of branches in hierarchical structure as nodes and generated three-dimensional tree models by applying a geometric model for modeling the curve and volume of the branches. Here, a method was designed that sequences the branch graph to simultaneously generate diverse trees effectively that have a similar branch pattern. Through this process, not only could the irregular growth of trees be controlled with a small number of parameters, but also multiple tree models that have similar shape could be simultaneously generated. Next, an optimization method was designed that optimizes tree models with complex structure for the immersive virtual landscape based on virtual reality. In this study, tree models with diverse resolution were simultaneously generated by controlling geometric model of branch based on camera-based static LOD. Through a test on the efficiency of the proposed LOD, it was confirmed that it has a tree model structure that is appropriate for a virtual reality environment. Moreover, the immersive system based on virtual reality could be more effectively created through compatibility with Unity3D engine. Virtual landscape based on virtual reality was created by using Oculus SDK and Unity3D engine based on the proposed tree modeling and optimization method. PTT method was proposed for effective allocation of multiple tree models on terrain. Through this process, we could confirm that high-quality immersive virtual landscape was actually created in an efficient structure in the test.

This study proposes a method for effectively creating and processing immersive virtual landscapes based on virtual reality in an efficient structure. The proposed method yields tree modeling that does not cause user motion sickness and at the same time optimizes the virtual reality environment. In contrast to the results of existing studies, this study finds the degree of realism within the tree model can be relatively low. It may be useful in general-purpose virtual reality situations such as games that require less realistic expression or that focus on providing educational content. Also, only visual sense using Oculus HMD was considered in this study related to virtual reality technology. Hence, we plan to study diverse user experience in virtual landscape in the future by using hand recognition device such as leap motion and treadmill-type motion recognition device such as Virtuix Omni. Moreover, we plan to conduct specific verification of the proposed method through a survey test regarding the psychological factors that users feel in immersive virtual landscape including immersion, reality, and motion sickness.

Acknowledgments: This paper was supported by Research Fund offered from Catholic University of Pusan.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Sutherland, I.E. A head-mounted three dimensional display. In Proceedings of the Fall Joint Computer Conference, Part I, San Francisco, CA, USA, 9–11 December 1968; ACM: New York, NY, USA, 1968; pp. 757–764.
2. Stewart, D. A platform with six degrees of freedom. In Proceedings of the Institution of Mechanical Engineers, London, UK, 13 April 1966; pp. 371–386.
3. Bergamasco, M. The GLAD-IN-ART project. In *Virtual Reality: Anwendungen und Trends*; Warnecke, H.J., Bullinger, H.J., Eds.; Springer: Berlin/Heidelberg, Germany, 1993; pp. 251–258.
4. Merhi, O.; Faugloire, E.; Flanagan, M.; Stoffregen, T. Motion sickness, console video games, and head-mounted displays. *Hum. Factors* **2007**, *49*, 920–934.
5. Lindenmayer, A. Mathematical models for cellular interaction in development, part I and II. *J. Theor. Biol.* **1968**, *18*, 280–315.
6. Karwowski, R.; Prusinkiewicz, P. Design and implementation of the L+C modeling language. *Electron. Notes Theor. Comput. Sci.* **2003**, *86*, 134–152.
7. Palubicki, W.; Horel, K.; Longay, S.; Runions, A.; Lane, B.; Měch, R.; Prusinkiewicz, P. Self-organizing tree models for image synthesis. In Proceedings of the ACM SIGGRAPH, New Orleans, LA, USA, 3–7 August 2009.
8. Tan, P.; Fang, T.; Xiao, J.; Zhao, P.; Quan, L. Single image tree modeling. In Proceedings of the ACM SIGGRAPH Asia 2008, Los Angeles, CA, USA, 11–15 August 2008.
9. Zhang, D.; Xie, N.; Liang, S.; Jia, J. 3D tree skeletonization from multiple images based on PyrLK optical flow. *Pattern Recognit. Lett.* **2015**, *76*, 49–58.
10. Cruz-Neira, C.; Sandin, D.J.; DeFanti, T.A.; Kenyon, R.V.; Hart, J.C. The CAVE: Audio visual experience automatic virtual environment. *Commun. ACM* **1992**, *35*, 64–72.
11. Cruz-Neira, C.; Sandin, D.J.; DeFanti, T.A. Surround-screen projection-based virtual reality: The design and implementation of the CAVE. In Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques, Anaheim, CA, USA, 2–6 August 1993; ACM: New York, NY, USA, 1993; pp. 135–142.
12. Alaraj, A.; Lemole, M.; Finkle, J.; Yudkowsky, R.; Wallace, A.; Luciano, C.; Banerjee, P.; Rizzi, S.; Charbel, F. Virtual reality training in neurosurgery: Review of current status and future applications. *Surg. Neurol. Int.* **2011**, *2*, 52.
13. Mine, M.R.; Brooks, F.P., Jr.; Sequin, C.H. Moving objects in space: Exploiting proprioception in virtual-environment interaction. In Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, Los Angeles, CA, USA, 3–8 August 1997; ACM Press/Addison-Wesley Publishing Co.: New York, NY, USA, 1997; pp. 19–26.
14. Insko, B.E. Passive Haptics Significantly Enhances Virtual Environments. Ph.D. Thesis, The University of North Carolina at Chapel Hill, Chapel Hill, NC, USA, March 2001.
15. Hoffman, H.G. Physically touching virtual objects using tactile augmentation enhances the realism of virtual environments. In Proceedings of the IEEE on Virtual Reality Annual International Symposium, Atlanta, GA, USA, 14–18 March 1998; pp. 59–63.
16. Szalavari, Z.; Gervautz, M. The personal interaction panel—A two-handed interface for augmented reality. *Comput. Graph. Forum* **1997**, *16*, C335–C346.
17. Cheng, L.P.; Lühne, P.; Lopes, P.; Sterz, C.; Baudisch, P. Haptic turk: A motion platform based on people. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Atlanta, GA, USA, 10–15 April 2010; ACM: New York, NY, USA, 2014; pp. 3463–3472.
18. Hollerbach, J.M.; Mills, R.; Tristano, D.; Christensen, R.R.; Thompson, W.B.; Xu, Y. Torso force feedback realistically simulates slope on treadmill-style locomotion interfaces. *Int. J. Robot. Res.* **2001**, *20*, 939–952.
19. Kohli, L.; Burns, E.; Miller, D.; Fuchs, H. Combining passive haptics with redirected walking. In Proceedings of the 2005 International Conference on Augmented Tele-Existence, Christchurch, New Zealand, 5–8 December 2005; ACM: New York, NY, USA, 2005; pp. 253–254.
20. Ortega, M.; Coquillart, S. Prop-based haptic interaction with co-location and immersion: An automotive application. In Proceedings of the IEEE International Workshop on Haptic Audio Visual Environments and Their Applications, Ottawa, ON, Canada, 1–2 October 2005; p. 6.

21. Cheng, L.P.; Roumen, T.; Rantzsch, H.; Köhler, S.; Schmidt, P.; Kovacs, R.; Jasper, J.; Kemper, J.; Baudisch, P. TurkDeck: Physical virtual reality based on people. In Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology, Charlotte, NC, USA, 11–15 November 2015; ACM: New York, NY, USA, 2015; pp. 417–426.
22. Li, H.; Trutoiu, L.; Olszewski, K.; Wei, L.; Trutna, T.; Hsieh, P.L.; Nicholls, A.; Ma, C. Facial performance sensing head-mounted display. *ACM Trans. Graph.* **2015**, *34*, 1–9.
23. Jirasek, C.; Prusinkiewicz, P.; Moulia, B. Integrating biomechanics into developmental plant models expressed using L-systems. In *Plant biomechanics 2000*, Proceedings of the 3rd Plant Biomechanics Conference, Freiburg-Badenweiler, Freiburg, Germany, 27 August–2 September 2000; Georg Thieme Verlag: Stuttgart, Germany, 2000; pp. 615–624.
24. Runions, A.; Lane, B.; Prusinkiewicz, P. Modeling trees with a space colonization algorithm. In Proceedings of the 2007 Eurographics Workshop on Natural Phenomena, Prague, Czech Republic, 4 September 2007; pp. 63–70.
25. Pirk, S.; Stava, O.; Kratt, J.; Said, M.A.M.; Neubert, B.; Měch, R.; Benes, B.; Deussen, O. Plastic trees: Interactive self-adapting botanical tree models. *ACM Trans. Graph.* **2012**, *31*, 1–10.
26. Boudon, F.; Pradal, C.; Cokelaer, T.; Prusinkiewicz, P.; Godin, C. L-Py: An L-system simulation framework for modeling plant development based on a dynamic language. *Front. Plant Sci.* **2012**, *3*, doi:10.3389/fpls.2012.00076.
27. Longay, S.; Runions, A.; Boudon, F.; Prusinkiewicz, P. TreeSketch: Interactive procedural modeling of trees on a tablet. In Proceedings of the International Symposium on Sketch-Based Interfaces and Modeling, Annecy, France, 4–6 June 2012; Eurographics Association: Aire-la-Ville, Switzerland, 2012; pp. 107–120.
28. Xu, L.; Mould, D. Procedural tree modeling with guiding vectors. *Comput. Graph. Forum* **2015**, *34*, 47–56.
29. Livny, Y.; Pirk, S.; Cheng, Z.; Yan, F.; Deussen, O.; Cohen-Or, D.; Chen, B. Texture-lobes for tree modeling. In Proceedings of the ACM SIGGRAPH 2011, Vancouver, BC, Canada, 7–11 August 2011; pp. 1–10.
30. Beneš, B.; Andrysco, N.; Štáva, O. Interactive Modeling of Virtual Ecosystems. In Proceedings of the Fifth Eurographics Conference on Natural Phenomena, Munich, Germany, 30 March–3 April 2009; pp. 9–16.
31. Deussen, O.; Hanrahan, P.; Lintermann, B.; Měch, R.; Pharr, M.; Prusinkiewicz, P. Realistic modeling and rendering of plant ecosystems. In Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques, Orlando, FL, USA, 19–24 July 1998; pp. 275–286.
32. Měch, R.; Prusinkiewicz, P. Visual models of plants interacting with their environment. In Proceedings of the 23rd International Conference on Computer Graphics and Interactive Techniques, New Orleans, LA, USA, 4–9 August 1996; pp. 397–410.
33. Kim, J.; Kim, D.; Cho, H. Procedural modeling of trees based on convolution sums of divisor functions for real-time virtual ecosystems. *Comput. Animat. Virtual Worlds* **2013**, *24*, 237–246.
34. Kim, J.; Cho, H. Efficient modeling of numerous trees by introducing growth volume for real-time virtual ecosystems. *Comput. Animat. Virtual Worlds* **2012**, *23*, 155–165.

© 2016 by the author; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (http://creativecommons.org/licenses/by/4.0/).