

3D Tree Modeling and Visualization Simulation Based on Fractal

Xudong Xu/ Associate professor
The College of Computer Science
Beijing University of technology
Beijing, China
xuxudong@bjut.edu.cn

Jingqing Ye/ Master
The College of Computer Science
Beijing University of technology
Beijing, China
ye.jing.qing@emails.bjut.edu.cn

Guanfeng Lv/ Lecturer
The College of Computer Science
Beijing University of technology
Beijing, China
lgf@bjut.edu.cn

Abstract—In this paper, we propose a 3D tree simulation method which is based on fractal theory and combined with the morphological characteristics of trees. By modifying a few simple parameters we can achieve the realization of different types of 3D tree automated generation. According to the tree structure fractal self-similar characteristic, combined with the morphological characteristics of the trees, we abstract identified class to realize the description of each organ's attributes and functions; Using random control function to improve the fractal iterative methods to make the generated trees more realistic; With the help of the developing platform of Visual C++ language and OpenGL, we can realize 3DTree's optimize modeling and visualization simulation. Experiments show that the method generates visually convincing results.

Keywords- morphology model, fractal, OpenGL, visualization

I. INTRODUCTION

Natural scenery is one of the indispensable important constituent parts in 3D games, digital entertainment, commercial advertising, city planning and virtual campus. At the same time, it remains a heated discussed issue as well as a difficult issue in computer graphics and virtual reality. As a kind of natural scenery, the trees have a wide range of species, great diversity of the shape, complicated structure and strong characteristics, which brings difficulty in the tree's modeling and visualization simulation.

In natural scenery aspects, many scholars have paid a strenuous efforts, put forward a lot of modeling methods, such as L-system, IFS (function iterative system), the parameters of axial technology, fractal method, particle system, etc [1], and they obtained certain achievements. Along with the computer science visualization technology continues to improve, people's demand to the authenticity of the 3DTree's visualization simulation is higher and higher. Therefore construct the more realistic 3DTree's modeling and visual simulation has important significance.

To overcome the difficulties of 3D tree modeling and visualization simulation, we provide a better method, basing on Oppenheimer's method that uses fractal technology to generate the tree, to make the generated tree more realistic. According to the tree structure fractal self-similar characteristic, combined with the morphological characteristics of the trees, we abstract different kinds of class to realize the description the attributes and functions of the trunk, branch and leaf; We propose a fractal iterative algorithm to generate 3D tree, adding random control

function into the algorithm, by using the random control function and setting the trees morphological parameters, make the generated branches more realistic; With the help of the developing platform of Visual C++ language and OpenGL, we can use fractal technology to realize 3D trees' simulation. This method to generate 3D tree is very simple, and just by modifying several parameters can we get different shapes of 3D tree, and it can be applied to 3D games and virtual campus.

II. RELATED WORK

Since the 1960s, people have achieved great development in the research of the plant modeling. In 1968, Lindenmayer put forward the L-system [2], using simple rules to generate complex plant morphology. Oppenheimer used fractal technology on trees modeling [3]. Reeves used particle system methods to simulate the forests [4]. Weber began with the overall geometry structure, using several parameters to control the trees modeling, and had discussed on forest mapping [5].

Our country about the research of the plant modeling in the 1990s, started relatively late in this area, but in recent years our country have obtained certain development in this field. Xiaoqin Hao and Tieying Song did the groundbreaking work in the trees visualization simulation [6]. Yang Xu used L-system to simulate the basic shapes of trees [7]. Qiong Wei used DDL as the basic algorithm to simulate trees morphology characteristics [8]. Lei Lei put forward a method based on energy model for leaf texture structure algorithm [9].

Early research on plant mainly on the modeling, and ignore the morphological characteristics of the trees. According to the morphological and structural features of trunk, branches, leaves, the topological relationship between trunk and branch, branch and branch, branch and leaves, we abstract identified class to describe each organ's attributes and functions, by constructing each organ class object, and call its methods, which can automatically generate 3D tree. Through comparing all kinds of the 3D tree algorithms, we propose a fractal iterative algorithm to generate 3D tree, adding random control function and tree morphological characteristics into the algorithm, make the generated 3D tree more realistic. In this way, we just need to modify a few simple parameters, without having to know the details of how to implement, thus the system can generate all shapes of the 3D tree automatically.

III. 3D TREE MODELING SELECTING

According to the biological characteristics of trees, we can know that the factors affecting trees morphology including the shape of the trunk, the shape of the branches, the shape of the leaves, as well as the topological relationships between trunk and branches, branch and branch, branch and leaves, the number of the branches on each level, the total level of the tree, and the number of the leaves on the last level of the branch. In order to make the tree morphological model more vivid, we make full use of the object-oriented design concept, combine with the biological characteristics of the trees to abstract identified class to realize the description of each kind organ's attributes and functions.

A. Basic Element Model

Most of the trunk and branches are not straight but by plurality of basic elements, we regard the basic element as a cylinder. Due to the rendering result of a cylinder is not very ideal, we adopt an octagonal prism to simulate a basic element. Experiments have verified that a polygon with an edge number greater than 8 can simulate smooth cylinder or frustum of a cone very well. By using this method, the generated tree more vivid, and it is beneficial to the tree's rendering. Therefore we construct a basic element class (CObject), in which the parameters decide to the morphological characteristics of the basic element, including its basic radius (basicRadius), top radius (topRadius) and the height (height). By calling this class object we can achieve the information which used to generate basic element model.

B. Trunk Model

We know that trunk are mostly straight, plump or curved, we can use multiple of basic element model to simulate the trunk. Therefore we construct a trunk class (CTrunk), in which the location of the trunk is decided by its location in the 3D space (the location based on viewpoint), the parameters decide to the morphological characteristics of the trunk, including the base radius of the trunk (baseRadius), the top radius of the trunk (topRadius), the height of the tree (height). By calling this class object we can achieve the information which used to generate trunk model. We can set the parameters of the trunk and the location of the tree by ourselves will to construct out different kinds of trunk model.

C. Branch Model

From tree's self-similarity on the statistical significance we know that each branch has similar morphological characteristics, and the branches are mostly curved, we can use multiple of basic element model to simulate the branch. Therefore we construct a branch class (CBranch), in which the parameters decide the location of branch, including the depth of the branch (depth), the angle of the branch (angel), the azimuth of the branch (azimuth) and the level it belongs (level); the parameters decide the morphological characteristics of the branch, including the base radius of the branch (baseRadius), the top radius of the branch (topRadius), the height of the branch (height); the

parameter which decides the number of the leaves on the last level branch is leaves number (numleaf); the parameter which saves the next level branches information is branches information (Child); the parameter which saves the leaves information is leaf information (leaf).

The depth of the branch is decided by the height of the branch it belongs to, and the number of the branches on that branch, we use uniform distribution law to get its depth. The angle of the branch is decided by the species of the trees, according to the growth of the plants deterministic, we can determine that the angle of the branch conform to the golden law, so we set the angle of the branch as 34.4° or 55.6° ; The azimuth of the branch also conform to the golden law, we set the azimuth on the adjacent branches at the same level as 225° ; The current level of the branch can be obtained by up one level plus one; The base radius of the branch, the top radius of the branch and the height of the branch are obtained by the up one level branch's related parameters using the certain attenuation coefficient, and the number of the leaves on the last level branch is decided by ourselves demand. Due to the strictly according to this method to generate the branches, which make the 3D tree too rule, therefore we add random control function on this method, so that the generated trees more realistic.

By calling this class object we can get the information which used to generate the branch, and the next level branches, then by judging whether it has got the last level we can get the information of generate the leaf. Using the improved fractal iterative method, and calling this class object, which we can construct different kinds of branch model.

D. Leaf Model

As the leaves are various and have complex edge information, if we want to describe the information of each leaf, it will reduce the operating efficiency of the procedure. By observing, we can see that the leaves are all in the same plane. In this paper we use quadrilateral to indicate leaves, through the image rendering we can get different kinds of leaves. Therefore we construct a leaf class (CLeaf), in which the parameters decide the location of the leaf, including the angle of the leaf (angel), the azimuth of the leaf (azimuth) and the depth of the leaf (depth); The parameters decide the morphological characteristics of the leaf, including the height of the leaf (height) and the wide of the leaf (wide).

The leaf on the branch have a variety of distribution, what we adapt to determine the location of the leaf is the height of the last level branch and the number of the leaves on this branch, we can use uniform distribution law to get it. The angle of the leaf is the angle between the leaf and the last level branch, we set the angle as 55.6° . The azimuth of the leaf is the angle between the adjacent leaves, we set the azimuth as 225° . The height of the leaf and the wide of the leaf are decided by the last level branch's basic radius, we get them through the certain proportion of the setting. To make the simulation more realistic, we add random control function in the process of generate related parameters, through calling this class object we can achieve the

information which used to generate leaf model.

IV. 3D TREE FRACTAL ALGORITHM

Self-similarity is widespread in plant morphology, precisely because of the regularity of the plant structure, which makes the fractal technology can be successfully applied to the plant computer modeling process. The branching of the tree in the nature has self-similar fractal structure on the statistical significance, so we can extract self-similar structure from the overall structure of the tree, and see it as a foundation to construct tree model. We propose a fractal iterative algorithm to generate branch, adding random control function into the algorithm, using it to achieve the generated branches information of all levels, providing data information for branches visualization simulation. With our common Yang, pine, fir, cypress as an example, we elaborate the fractal algorithm which uses to generate tree model.

The algorithm described as follows:

```
sector<Cbranch *> creatBranch()
{
    float azimuth=225;
    float angle=34.4;
    if (currentlevel<totalLevel)
    {
        // According to the current number of branches (numbranch), //we can use
        fractal iteration method to generate all the //information of branches on the next
        level.

        for (int i=0;i<numbranch;i++)
        {
            According to the activities of the braches i whose attributes baseRadius,
            topRadius,height, we can get the attributes (baseRadius,topRadius,height) of
            next level branch j
            // use random control function
            Using the variable azimuth to get the azimuth of the branch j(azimuthj);
            Cbranch *t=new Cbranch(baseRadius, topRadius,height, angle, azimuthj,
            numbranch-1, currentlevel+1, depth, totalLevel,numleaf);
            //construct this branch
            t->creatBranch();
            //generate the branches of next level
            child.push_back(t);
            // put branches information on the next level in the //member variables
            child
        }
    }
    if(currentlevel==totalLevel)
    {
        //According to the number of leaves, we set the basic //information of the leaf

        for(int j=0;j<numleaf;j++)
        {
            According to the activities of the braches i whose attribute baseRadius,
            getting the attributes(width, height, depth) of the leaf j ;
            // use random control function
            Cleaf *t=new Cleaf (height, width, angle, azimuth, depth);
            //construct this leaf
            leaf.push_back(t);
            // put the information of leaf in the member variables leaf
        }
    }
}
```

V. 3D TREE VISUALIZATION SIMULATION

According to the fractal theory, we see the process of the tree visualization simulation as a recursive process. The algorithm starts with generate the trunk of the tree, then the next level branches generated by the current branch, thus if the branch reach the maximum level of the tree, it begin to generate the leaves. Using the tree object that we defined, calling the class object method, getting all the information which used to generate 3D tree, and using the OpenGL method, we can realize the 3D tree visualization simulation.

The trees visualization simulation generate algorithm described as follows:

1. Determine the location of the tree in the 3D space, the height of the tree(height), the base radius of the trunk(baseRadius),the top radius of the trunk(topRadius) , the total level of tree(totallevel), the number of the primary branches(numbranch) N and the number of the leaves on a single branch(numleaf).
2. According to the location of the tree in the 3D space, the height of the tree(height), the base radius of the trunk(baseRadius) and the top radius of the trunk(topRadius) to draw trunk;
3. Judge whether N is 0,if it equals to 0 the drawing is completed, or continue to execute 4;
4. Search the primary branch to get the parameters of branch on the next level and the number of the branches m;
5. Use fractal iteration method to generate the branch we searched;
6. Judge whether the number of the iteration j has got the maximize of the branch level, if it has got the maximize of the branch level, then drawing the leaves, m--,or continue iterate;
7. Judge whether m is 0, which means judge whether we have drew all the branches on the next level of the active branch, if it not equals to 0 continue to execute 5, or continue to execute 8;
8. Judge whether N is 0, which means judge whether we have traversed all the primary branches, if it not equals to 0 to repeat 4, or we have finished drawing the tree.

The following pictures show the 3D tree visualization simulation using Visual C++ language and OpenGL technology in the Visual Studio 2008 platform. The figure 1 shows the tree which has 3 level branches, 6 branching on the branch and 8 leaves on the last level branch; The figure 2 shows the tree the same as the figure 1, but it looks at different perspectives; The figure 3 shows the tree which has 3 level branches, 7 branching on the branch and 8 leaves on the last level branch, and at the same time the viewpoint of it looks at is the same as the figure 1.



Figure 1. A 3D tree with 3 level branches, 6 primary branches and 8 leaves on the last level branch



Figure 2. The 3D tree is the same as figure1 at different perspectives

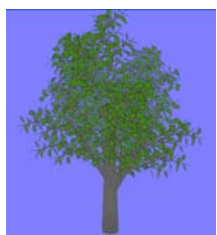


Figure 3. A 3D tree with 3 level branches, 7 primary branches and 8 leaves on the last level branch, having the same viewpoint as fig.1

VI. CONCLUSION

We make full use of object-oriented thinking, combined with the morphological characteristics of trees to construct each organ model of trees, in which we use random control function and simple parameter settings to improve the fractal iterative method, so that as to make the generated trees more realistic. Only by modifying a few simple

parameters can we let the system automatically generate a variety of trees, and experiments show that the generated 3D trees are more realistic. However, in our study we did not consider the affect of the light and the surrounding environment on the 3D tree rendering. In the future, we will make efforts to research the affect of the light on the 3D tree rendering.

ACKNOWLEDGMENT

We would like to thank everyone who has ever helped us. And we would like to thank the China seismological bureau geophysical institute who provides us support to research the project about digital earth.

REFERENCES

- [1] Weilong Ding, Fanguan Xiong, Youhua Zhang. Based on the component of plants to visual three dimensional structure model [J]. Small microcomputer system, 2004, 25 (9):1624-1627.
- [2] A. Lindenmayer. Mathematical models for cellular interaction development [J]. Journal of Theoretical Biology, 1968, 18:280-315.
- [3] Oppenheimer PE. Real time design and animation of fractal plants and trees . Computer Graphics.1986, 20(4):55-64
- [4] Reeves W. Approximate and probabilistic algorithms for shading and rendering structured particle systems. Proceedings of the SIGGRAPH' 85, San Fransico, California, 1985:313-322
- [5] Weber J, Penn J. Creation and rendering of realistic trees. Proceedings of the SIGGRAPH' 95, 1995, 119-127
- [6] Xiaoqin Hao. Forestry science and visualization [J]. Forestry science, 2001, 37(6) : 105 — 109.
- [7] Yang Xu, Lin Zhu, Ming Chang. A Tree-Dimensional Geometric Modeling Method for Botanical Trees [J]. Computer Engineering and Applications, 2001, (11): 141-143
- [8] Qiong Wei, Xiangning Jiang. Study on 3D visual tree model using DOL-system [J]. Journal of Beijing Forestry University, 2003, 25(3):64-47
- [9] Lei Lei, Xinyu Guo, Shuqiu Zhou. Construction of Leaf Veins Based Particle System [J]. Computer Engineering and Applications, 2004, (36): 218-219