

分形递归树的森林景观实时模拟方法

蒋丽涛, 孙博文

(哈尔滨理工大学 计算机科学与技术学院, 黑龙江 哈尔滨 150080)

摘 要:为真实地模拟自然界的森林景观,在利用分形递归算法构建参数化三维树木模型的基础上,结合 OpenGL 函数,将纹理映射、Alpha 测试和地面阴影应用到树木的真实感绘制中.并在算法中引入随机数,实现大小不同、形态各异的树木生成.绘制时,根据视点距离的远近,调整场景的精细程度,以符合视觉效果,且加快了图形生成速度.以此方法构造森林,可改变观察者的视点位置,实现对森林景观的实时模拟.

关键词:分形;递归算法;树木;森林景观;实时模拟

中图分类号: TP391.41 **文献标志码:** A **文章编号:** 1007-2683(2009)05-0039-04

Real-time Simulation of Forest Scenery Based on Fractal Recursion Trees

JIANGLi-*tao*, SUNBo-*wen*

(School of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China)

Abstract: To simulate the forest scenery of nature truly, a method of real-time simulation based on fractal recursion trees is presented in this paper. By constructing parameterized 3D trees model using recursion algorithm, and combining OpenGL function, texture-mapping, Alpha testing and ground shadow are applied to realistic rendering of trees. In addition, random numbers are introduced into algorithm, which generated trees that have different size and various shapes. According to the distance from view point, fine degree of scene can be adjusted in accord with visual effect and speedup generation of graphics when rendering. Forest is constructed by this method, and real-time simulation of forest scenery is realized through changing the position of observer's view point.

Key words: fractal; recursion algorithm; trees; forest scenery; real-time simulation

0 引 言

大规模自然场景的建模与绘制是图形学研究领域的前沿课题,它在虚拟现实、战场模拟、影视游戏制作等方面均有广泛应用.同时,对树木类的自然场景,特别是森林场景的实时模拟也一直是一个富有挑战性的课题.因为树木结构复杂,种类繁多,形态

各异,使其无论在造型、存储还是在绘制上都存在着一定困难.近年来,飞速发展起来的分形理论以自相似性和分数维为特点,突破了以往只能生成较规则图形的局限性,可将自然界中绝大多数的不规则图形,如树木、河流、山川等,真实地再现于计算机上.

目前,对森林景观的建模已做了大量工作,其中人们最为关注的是树木的模拟.在虚拟植物建模方面,最早采用的是美国生物学家 Aristid Lindenmayer

收稿日期: 2007-09-26

作者简介: 蒋丽涛(1983—),女,硕士研究生, E-mail: jlt11@126.com;

孙博文(1963—),男,副教授.

提出的L系统建模方法.随后又有学者提出了许多生成植物图形的方法和模型,例如各种改进的L系统^[1]、迭代函数系统(IFS)^[2]、粒子系统^[3]、几何设计法^[4]以及基于图像的造型法^[5]等.

本文提出了一种基于分形递归树的森林景观实时模拟方法.根据分形递归算法构建参数化的三维树木模型,结合OpenGL函数,采用图形学的编程技术生成具有真实感的树木.在分形递归树的基础上,实现对森林景观的实时模拟,绘制时利用LOD技术加快场景渲染的速度.将随机数引入树木生长算法,使森林场景中生成的树木大小不同、形态各异,树木之间的行、列间距离产生随机变化,提高了自然景物模拟的逼真度.

1 参数化的树木建模

1.1 树木结构模型的建立

树木在宏观结构上可以看成是由若干树枝和树叶构成的,树枝又是由若干枝段组成.因此,可以将一棵树递归地定义为:由若干枝段组成的树枝,而树叶则生长在树枝的顶端^[6],如图1所示.树木在整体结构上呈现出了自相似的分形结构,可通过递归变换由上一级分枝的形态结构得到下一级分枝结构,并利用参数传递来控制树木的整体形态结构模型.

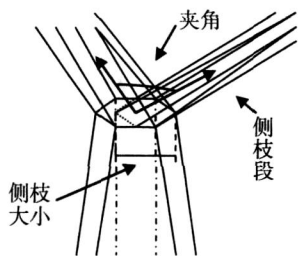


图2 分枝模型

在树木模型中,枝段和树叶是最基本的组成单元.枝段用一个可以弯折和扭转的六边形台体表示.每一个枝段用以下的数据结构来描述: Segment{ Expand, Twist, Length, Radius}.其中 Expand和 Twist为枝段弯折和扭转的程度,确定枝段生长的方向,体现了树枝生长的不规则性. Length和 Radius分别为该枝段的长度和半径,用于描述树枝生长越来越短、越来越细的特征.

树叶则用一个可以转动的平面四边形来表示.它可以用以下的数据结构来描述: Leaf{ Expand, Twist, Scale}.其中 Expand指绕X轴的转动量,

Twist指绕Y轴的转动量,两者控制树叶的生长方向, Scale为树叶的大小.

1.2 树木的生长算法

建模过程中,将树枝看作是由若干相似的枝段构成,这里将用迭代表达式来描述一条树枝各枝段的生成过程.假定前一个枝段的各参数分别为{ Expand, Twist, Length, Radius},那么次枝段的参数为

$$\left. \begin{aligned} \text{Expand}' &= \text{Expand}_{\text{Global}} \\ \text{Twist}' &= \text{Twist} + \text{Twist}_{\text{Global}} \\ \text{Length}' &= \text{Length} \times \text{Length}_{\text{Global}} \\ \text{Radius}' &= \text{Radius} \times \text{Radius}_{\text{Global}} \end{aligned} \right\} \quad (1)$$

下角标有 Global字样的都是全局变量. Expand_{Global}指定了树枝每一次弯折的程度, Twist_{Global}用来控制树枝逐段扭转的速度. Length_{Global}和 Radius_{Global}用来控制树枝变短变细的速度.

构造新的枝段时,次枝段将在前一枝段的坐标位置基础上,形成一个新的本地坐标系.分枝的弯折和扭转按照以下规则进行,分枝先在平面XZ上扭转T度,再绕向量[cos(T), 0, sin(T)]转动E度.模型中用A、B、C表示分枝旋转的方向,分枝的生长方向表示为[A', B', C'] = [A, B, C]R,其中R为3×3的旋转矩阵, R_A、R_B、R_C分别为A、B、C方向的旋转矩阵,由式(2)计算^[7]:

$$\begin{aligned} R_A &= \begin{bmatrix} \cos(T) & 0 & -\sin(T) \\ 0 & 1 & 0 \\ \sin(T) & 0 & \cos(T) \end{bmatrix} \\ R_B &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(E) & -\sin(E) \\ 0 & \sin(E) & \cos(E) \end{bmatrix} \\ R_C &= \begin{bmatrix} \cos(T) & 0 & \sin(T) \\ 0 & 1 & 0 \\ -\sin(T) & 0 & \cos(T) \end{bmatrix} \end{aligned} \quad (2)$$

则分枝的末端坐标由下式确定:

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & R' \\ 0 & 1 & 0 & L' \\ 0 & 0 & 1 & R' \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_A & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_B & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_C & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

其中, T'、E'、L'、R'分别表示 Twist'、Expand'、Length'、Radius'.

确定好枝段的各种属性计算式之后,就可以利用树木生长的自相似性构造树木的生长算法.尽管自然界中树木的种类繁多,形态复杂多变,但其构造的基本原则仍遵循自相似规则,同一株植物的各级

分枝也十分相似,复杂的结构可以描述为不断重复的基本结构元素.因此,可以用一个树干和两个分枝的二叉树作为树木的基本元,即分形元.对其进行迭代,算法思想如下:

- 1) 在主枝的顶端侧生两个分枝;
- 2) 将这两个分枝看成新的主枝;
- 3) 两个分枝中的一根为枝段模型,以式 (1) 的方式计算,另一根为侧枝段,是新树枝的初始枝段,如图 2 所示.侧枝段的参数计算公式如下:

$$\left. \begin{aligned} \text{Expand}_{\text{Side}} &= \text{Bifurcate}_{\text{Global}} - \text{Expand}_{\text{Base}} \\ \text{Twist}_{\text{Side}} &= \text{Twist}_{\text{Base}} + \text{Twist}_{\text{Global}} + 180^\circ \\ \text{Length}_{\text{Side}} &= \text{Length}_{\text{Base}} \times \text{Length}_{\text{Global}} \times \text{SideScale}_{\text{Global}} \\ \text{Radius}_{\text{Side}} &= \text{Radius}_{\text{Base}} \times \text{Radius}_{\text{Global}} \times \text{SideScale}_{\text{Global}} \end{aligned} \right\} \quad (3)$$

下角标有 Base 字样的是主枝段的参数,根据式 (1) 的公式计算. $\text{SideScale}_{\text{Global}}$ 为小于 1 的数,用来控制侧枝段的长度和半径,区别于另外一根树段.

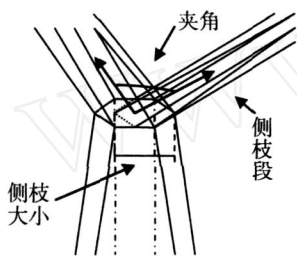


图 2 分枝模型

- 4) 在枝段的末梢,即树枝的顶端,产生树叶.

如此反复迭代改变了各树枝的大小或形态,再加上递归函数的作用,便可生成一棵枝叶茂盛的树木了.

综上所述,通过参数传递控制树木的整体形态结构模型,最终描述出植物各级枝干在粗细、长短和生长方向等方面的差异,植物模拟效果逼真、自然.

2 树木的绘制

本文以 VC++6.0 作为运行环境,结合 OpenGL 函数,采用了大量诸如纹理映射、反锯齿、光照和材质等图形学的编程技术,对上述建立好的树木结构模型进行可视化.绘制树木时,利用 OpenGL 的库函数设置树干材质,定义了环境光、漫反射光、镜面反射光和辐射光等颜色属性以及光亮度,并对枝干添加纹理,且利用 Alpha 测试来模拟有着不规则边缘的树叶.同时建立了由环境光和漫反射光构成的光

照模型,其中光源位置 GL_POSITION 缺省值为 (0, 0, 1, 0),定义了一个指向负 z 轴的方向型光源,然后对场景进行渲染,添加雾化的效果.

对于物体阴影的绘制,在工程图学中,一般都是基于三维直角坐标系,首先进行立体造型,然后根据光源模型和视点位置,结合复杂的技术进行投影得到的.这种方法以大计算量为代价,绘制丝毫不差的阴影.文中提出了一种简单的方法来获得三维树木的地面阴影.首先绕 y 轴旋转 90° ,使 z 轴指向光源的方向,并设置 z 轴的放大倍数为 0.再绕本地坐标系的 x 轴旋转 90° ,使 z 轴与地面垂直.然后启用混合功能,绘制另外一棵灰色的树.由于 z 轴的放大倍数为 0,因而树的整个三维图像似乎是被压扁粘贴在地面上,从而形成投影的平面.这样就可以沿着光照方向,当树木形态结构变化时而获得不同的地面树影.

此外,为了进一步描述植物同级枝干之间的差异,体现出植物的自然生长状态,在树木生长算法中引入随机数,如下式:

$$\left. \begin{aligned} \text{Expand}' &= \text{Expand}_{\text{Global}} + \text{rand}() \% 30 \\ \text{Twist}' &= \text{Twist} + \text{Twist}_{\text{Global}} + \text{rand}() \% 20 \\ \text{Length}' &= (\text{Length} + \text{rand}() \% 1) \times \text{Length}_{\text{Global}} \\ \text{Radius}' &= (\text{Radius} + \text{rand}() \% 1) \times \text{Radius}_{\text{Global}} \end{aligned} \right\} \quad (4)$$

即在枝段的弯折和扭转角度以及长度和半径中,添加上大小适当的随机函数,从而产生大小不同、形态各异的树木.如图 3 所示.



图 3 加入随机数后的树木形态

3 森林景观的实时模拟

在虚拟现实、动画制作等应用中,经常需要模拟一些森林景观的场景,这使得如何获得更具真实感森林场景的模拟得到了广泛关注.目前,森林景观的实时绘制方法主要有纹理映射法、粒子系统法以及细节简化法等.其中纹理映射法可以简化场景,消除几何面造成的混淆,使构造和准确绘制高度复杂的

自然场景成为可能,但该方法所绘制的植物群落缺乏动态效果,而且算法实现比较复杂.粒子系统则用很少的基本描述便可构造由百万个粒子构成的森林、草地等场景,但该方法构造的植物看起来有明显的人工痕迹,缺乏真实感.

本文采用了细节简化法,利用该方法中的 LOD 技术,在不影响画面视觉效果的前提下,通过逐次简化景物的表面细节来减少场景的几何复杂性,加快了场景渲染的速度,从而提高模拟效率.并在绘制三维树木的基础上,通过增加植物的绘制次数以及利用视图变换函数(即 glTranslate^* ()和 glRotate^* ())来改变观察者的视点位置和方向,实现对森林景观的实时模拟.

为进一步提高自然景物模拟的逼真度,通过调用随机函数,使树木之间的行、列间距离产生随机变化.同时,本文在绘制过程中,采用了上述模拟三维树木阴影的方法实现了动态阴影,使森林景观的模拟效果更加真实.在实验的测试平台(AMD 双核处理器 4000⁺ 2.1 GHz, 1.0GB 内存, NVIDIA GeForce 7300GT 显卡的 PC 机)上,图像分辨率为 $1\,024 \times 768$,绘制森林的平均总耗时为 1.5s,实验结果如图 4 所示.



图 4 森林场景

4 结 语

树木形态与生长的计算机模拟是一项非常有意义的研究.本文探讨了三维树木建模及其可视化的方法,并在绘制树木的基础上,采用细节简化法,实现了对森林景观的实时模拟.生成的树木和森林逼真、自然,可应用于虚拟现实、动画制作、园林设计等领域.下一步的工作是如何加入更多影响树木形态的参数,例如风力、重力等,以提高树木的模拟效果.同时,如何提高绘制效率和草地的真实感是今后值得研究的课题.

参 考 文 献:

- [1] PRUSNIEWSKI P, LINDENMAYER A. The Algorithmic Beauty of Plants [M]. New York: Springer-Verlag, 1990.
- [2] DEMKO S, HODGES L, NAYLOR B. Construction of Fractal Objects with Iterated Function System [J]. Computer Graphics, 1985, 19(3): 271 - 278.
- [3] 王小铭, 林 拉. 树木模拟的粒子系统模型及其实现 [J]. 华南师范大学学报:自然科学版, 2003(3): 49 - 53.
- [4] WEBER J, PENN J. Creation and Rendering of Realistic Trees [C]//Computer Graphics Proceedings Annual Conference Series New York: ACM, 1995: 119 - 128.
- [5] 严 涛, 吴恩华. 基于多幅图像的树木造型方法 [J]. 系统仿真学报, 2000, 12(5): 565 - 571.
- [6] BORIS J. WANG. Real-time Rendering of Photorealistic Natural Trees [EB/OL]. (2003.09.10) / [2007.09.20]. <http://www.lighthoughts.com/jpwang/old.site/etree.htm>.
- [7] 朱 岳, 邹北骥, 刘 洁, 等. 基于三维分枝模型的树木模拟 [J]. 计算机工程与科学, 2006, 28(7): 46 - 48.

(编辑:付长缨)