

文章编号: 1674-2869(2016)02-0204-05

基于OpenGL迭代函数植物建模算法的改进及实现

王志维, 江梦璇, 李 晖*

武汉工程大学计算机科学与工程学院, 湖北 武汉 430205

摘 要:提出了一种改进的基于OpenGL迭代函数植物几何建模方法,提高了植物模型的真实感.在传统方法中融合遗传算法对树木模型的构建过程进行优化,并且引入风场以实现对树木的动态模拟.结果显示该方法所构造的树木模型具有更加明显的分形姿态,随着迭代次数的增加,树木逼真度提高.在实验仿真条件下,通过绘点操作模拟逼真度最高能够达到96%.

关键词:植物几何建模;迭代函数系统;遗传算法;真实感

中图分类号:TP391.41 文献标识码:A doi:10.3969/j.issn.1674-2869.2016.01.019

Improvement and Implementation of Plant Modeling Algorithm of Iterative Function Based on OpenGL

WANG Zhiwei, JIANG Mengxuan, LI Hui

School of Computer Science and Engineering, Wuhan Institute of Technology, Wuhan 430205, China

Abstract: In this paper, an improved plant-geometric modeling method of iteration function based on OpenGL was proposed for improving the realistic feature of plant-model, in which the construction of the tree-model was optimized by adding genetic algorithm to the traditional method, and the dynamic simulation of the trees was achieved by introducing the wind field. The results show that the improved method produces much more obvious fractal shapes, and the realism of trees is improved with the iterations increasing. The highest fidelity can be up to 96% under the condition of experimental simulation by draw point operation.

Keywords: plants geometric modeling; iteration function system; genetic algorithm; realism.

1 引 言

近年来,虚拟现实技术(Virtual Reality Technology)的广泛应用为广大用户提供了更加逼真的三维视觉、听觉、嗅觉甚至味觉等各种感官享受,这给予广大体验者身临其境般的用户体验,也展现了计算机技术的独特魅力.

虚拟现实技术^[1]的发展与建模技术密不可分.建模技术是虚拟现实技术中最重要的技术之一,也是三维图形技术发展的关键技术之一.并且,建模技术也是一项涉及到多种计算机软件技术的综合型技术,并在模式识别、智能计算、虚拟

现实等学科中都有广泛应用.基于图形的建模与绘制技术和基于图像的建模与绘制技术^[2],是虚拟场景建模技术的两大支撑技术.按照所建立模型的类别来划分,虚拟场景建模主要可以划分为对自然景物模型的建模以及对人工景物模型的建模两大类.植物几何建模研究正是对自然景物模型的建模,主要用到几何建模^[3]法.

植物几何建模^[4]是一种运用计算机图形学理论与方法对植物的形态结构和生长过程进行模拟仿真的技术,其研究重点是植物的形态结构.运用该技术所构建的植物模型已经不仅仅局限于二维世界,向三维方向发展得也不错.

收稿日期:2015-11-02

基金项目:湖北省教育厅科研项目(Q20141507);武汉工程大学教学研究重点项目(X2014012);武汉工程大学计算机科学与工程学院院长基金

作者简介:王志维,本科生. Email:1563501222@qq.com.

*通讯作者:李 晖,博士,副教授. Email:lihui00317@163.com

在现实生活中,植物几何建模的应用领域十分广泛,在三维游戏开发、影视动画设计、虚拟社区等领域,它的应用都十分普遍.因此,它具有非常深远的研究意义.随着计算机软硬件技术的深入发展以及人类对植物更为深入的认识,植物几何建模技术必将会有更大的发展空间和应用空间,也将从一定层次上更多的提高人类的物质生活和精神生活的质量.

2 改进的迭代函数系统植物建模法

传统的迭代函数系统(Iterated Function System,简称IFS)植物建模方法所建立的植物几何模型并不够逼真,需要改进.下面主要讨论改进的迭代函数系统植物几何建模法,该方法对传统的IFS植物几何建模方法进行了两方面的改进:

1)在传统的IFS植物几何建模方法中加入遗传算法,使树木的结构体系更加稳定,也使模拟出的树木更加符合自然界中优胜劣汰的生存规律;

2)在传统的IFS植物几何建模方法中引入风场,实现树木的动态模拟.

改进的IFS植物几何建模方法的设计流程如图1所示.

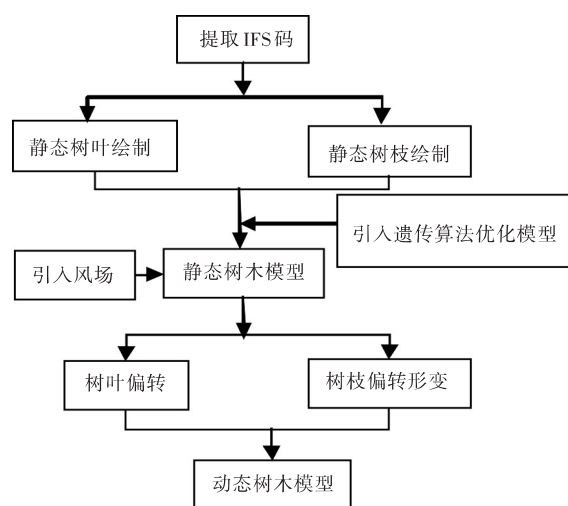


图1 改进的IFS植物建模设计流程

Fig. 1 Flowchart of improved IFS plant modeling

2.1 改进的迭代函数系统静态树木建模法

运用改进的IFS静态树木建模法进行树木建模,主要通过IFS码实现.每一组IFS码都有与之对应的仿射变换,即每一组IFS码都能唯一确定一幅分形图形,因此通过不同的IFS码,就可以重构出不同的树木模型.利用IFS码构造树木模型,主要分为构造树叶和构造树干两部分.

2.1.1 利用IFS码构造树叶 利用IFS码构造树叶,首先要获取树叶的IFS码.表1列举了一组树叶的IFS码,通过迭代函数系统模型进行多次迭代后,所得到的树叶模型就十分接近真实树叶.

表1 树叶的IFS码
Tab. 1 IFS-code of leaves

affine transformation	real variable						probability
	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>	
w_1	0	0	0	0.16	0	0	0.01
w_2	0.85	0.04	-0.04	0.85	0	1.6	0.85
w_3	0.2	-0.26	0.23	0.22	0	1.6	0.07
w_4	-0.15	0.28	0.46	0.24	0	0.44	0.07

下面对静态树叶模型的编号过程进行简单的描述.

第一步,定义一个二维双精度实型数组Code[[[]],将表1中的树叶IFS码存储在该数组中,方便之后调用.数组设定为4行7列.数组的第1行用于存储仿射变换 w_1 所对应的IFS码值;依次类推,数组的第2行、第3行、第4行分别用于存储仿射变换 w_2 、 w_3 、 w_4 所对应IFS码值.

第二步,定义函数IFSCode()用来实现静态树叶建模.函数IFSCode()的基本思想是循环迭代,即对于概率不同的各组IFS码值分别进行仿射变换模拟树叶形态,并为其赋予相应的颜色值.在经过多次迭代循环之后,树叶的形态会越来越逼近于真实树叶的形态.

2.1.2 树枝的构造 自然界中的树木千姿百态、形状各异,树枝是决定树木形态的重要因素.生物学上,植物的分枝方式主要有单轴分枝、合轴分枝以及假二叉分枝3种类型,其中单轴分枝是最常见的分枝方式,森林中有超过80%的植物采用单轴分枝方式,其最大的特点是有明显的树干.

建立树枝模型的主要思路是将树枝看成是粗细不一的圆柱体拼接而成.具体实现步骤可分为三步:

第一步,构造大小不一的若干圆柱体;

第二步,将构造出的圆柱体按单轴分枝的形式拼接成树枝的骨架;

第三步,对树枝的粗细程度、旋转角度、分枝长度等参数进行调整,使构造出的树枝模型更加贴近于现实景物.

2.1.3 引入遗传算法 运用传统的IFS植物几何建模法建立的树木模型具有丰富的纹理,但却不能体现树木之间的形态差异性.为了弥补IFS植物

几何建模法这一缺点,引入了遗传算法对传统的IFS植物几何建模法进行改进.在设计中设定的遗传算法的实现步骤如下:

- 1) 种群的设定. 初始种群随机产生,为简化操作流程,设计种群的规模为5;
- 2) 确定种群中优胜劣汰的最优适应度值;
- 3) 选择合适的概率对种群中的个体进行遗传变异操作,并计算新个体的适应度值. 此时存在的所有个体按适应度值的大小遵循从大到小的顺序排列,取最大的5个个体组成新的种群;
- 4) 若新种群中所有个体的最大适应度值接近或等于最优适应度值,则算法终止;
- 5) 若在出现最优个体前,预计的迭代次数已完成,则取所生成的适应度值最大的个体为算法的解.

在整个遗传算法执行的迭代过程中,难点在于适应度值的计算. 适应度函数的公式为:

$$F = \frac{aw_a + bw_b + cw_c + dw_d}{w_a + w_b + w_c + w_d} \quad (1)$$

式(1)中, a, b, c, d 4个参数分别表示影响树木生存的4个因素:叶片的采光性、树木结构的稳定性、叶片光合作用的能力、树木枝干的抗压能力;参量 w_a, w_b, w_c, w_d 分别表示 a, b, c, d 这4个参数的加权值.

2.2 改进的迭代函数系统动态树木建模法

2.2.1 虚拟风场建模 虚拟风场的建模一直是一个热门的研究领域,基于所选择的理论及方法的不同,现阶段比较热门的虚拟风场建模方法主要有基于分形布朗运动的风场建模、JunOhya风场模型、基于Perlin噪声值的风场建模3种. 本实验主要采用第2种模型,因此只介绍JunOhya风场模型.

JunOhya提出的风力 F 的近似表达式^[5]为

$$F = S\mu v \quad (2)$$

式(2)中, S 为树木受风影响的面积, v 为速度矢量, μ 为空气粘度系数,一般为常量.

但实际风场并不呈线性现象,为了使模拟出的风场与实际风场更加贴近,需要加入随机扰动因子使风力呈现非线性变化. 引入的随机扰动因子有扰动振幅 A 、扰动角速度 ω . 引入扰动因子后,风速表达式为

$$v = v_0 + A \sin \omega t \quad (3)$$

式(3)中,常量 v_0 表示初始风速;变量 t 表示时间.

那么将表达式(3)引入式(2)中,可以得到风力 F 的新表达式为

$$F = S\mu(v_0 + A \sin \omega t) \quad (4)$$

表达式(4)表示的则是非线性随机风场模型.

2.2.2 树木随风摇曳的实现过程 在分别对树枝与树叶在风场中的受力情况进行分析后,可以得出树枝的形变与偏转运动过程以及树叶的偏转运动过程. 但在实际生活中观察到的树木随风摇曳的场景是连续的,因此在实现过程中,需要对每幅图像进行处理,以达到模拟树木连续运动的效果. 具体的实现步骤如下:

- 1) 计算单位时间内树枝及树叶受到的风力;
- 2) 分析树叶的偏转运动方向及运动的幅度大小;
- 3) 分析树枝的形变及偏转运动情况;
- 4) 对树木的状态进行更新,并重新对树木进行渲染.

对动态树木的模拟主要分为树枝和树干两个部分,在不同的单位时间内,树木受力情况及运动情况会有差别,因此需要及时对树木运动状态进行更新,以达到模拟树木随风连续运动的效果.

3 结果及讨论

通过对改进的迭代函数系统植物几何建模方法的描述及执行,树木模型大致上能顺利建成. 本节将对改进的IFS植物几何建模法的实验结果进行分析,并对上述方法在实验过程中存在的缺陷及问题提出解决方案.

3.1 实验结果展示

由于动态树木是在静态树木的基础上增加风场实现的,因此对改进的IFS植物建模方法的结果展示直接使用动态树木. 为了简化计算及操作的步骤,实验过程中仅只考虑了树叶在风力作用下的偏转运动,而没有考虑树叶的形变. 同样的,场景中只模拟了微风对树木的影响. 经过相关步骤之后,最终生成的优质树木模型如图2所示. 场景中的树木以明显的分支形式呈现,仔细观察所构建的树木模型,也能发现模型在光照下的阴影.



图2 改进的IFS动态植物模型

Fig. 2 Dynamic plant modeling by improved IFS

在图2中所展示的树木具有明显的分形姿态,用改进的IFS植物建模方法构建的是树桩旁枝重新生长的树木景象,同时特地引入阴影图算法,为树木绘制了光照阴影,使树木模型更加真实.

3.2 实验结果分析

在利用改进的迭代函数系统植物几何建模法进行植物几何建模时,需要考虑三个问题:IFS码的选取、迭代次数的确定以及对内部资源的影响.

1) IFS码对植物形态的影响

下面分别对两组不同的IFS码构造的树木模型差异性进行说明.假设有两组不同的IFS码,如表2与表3所示.

表2 树木A的IFS码
Tab. 2 IFS-code of tree A

affine trounsfomation	real variable						probability
	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>	
w_1	0	0.04	0	0.16	0	0	0.01
w_2	0.85	0	-0.04	0.85	0	1.6	0.85
w_3	0.2	-0.26	0.23	0.22	0	1.3	0.07

表3 树木B的IFS码
Tab. 3 IFS-code of tree B

affine trounsfomation	real variable						probability
	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>	
w_{11}	0	0.04	0	0.16	0	0	0.13
w_{21}	0.85	0	-0.04	0.85	0	1.6	0.67
w_{31}	0.2	-0.26	0.23	0.22	0	1.3	0.21

将表2与表3中的这两组IFS码分别用于构建树木,确定两组IFS码的迭代次数均为2 000次,所生成的树木的草图如图3所示.

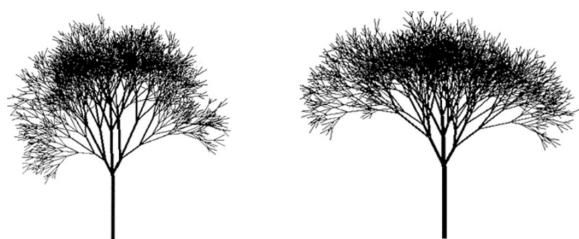


图3 两组IFS码对应的树木草图

Fig.3 Sketches of trees corresponding to two IFS-codes

在图3中,左侧的树木草图是表2中对应的IFS码迭代2 000次生成的,而右侧的树木草图是表3中对应的IFS码迭代2 000次生成的.两幅草图在树木形态上有着明显的差异,由此也可观察出IFS码对树木形态的影响.

2) 迭代次数对植物逼真度的影响

迭代函数系统对树木形态模拟的过程是通过一系列复杂的绘点操作来完成的.在对表2中的IFS码选取7组不同的迭代次数后,对比图像的逼真度,可以得到相对应的逼真度 R 与迭代次数 T 的关系如图4所示.

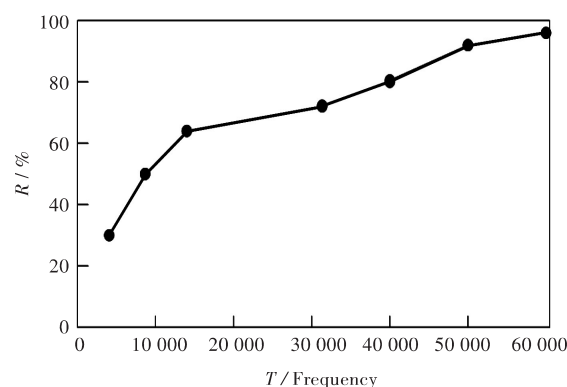


图4 迭代次数与模型逼真度的关系

Fig. 4 Relationship between iterations and fidelity of modeling

由图4中的数据对应关系可知,虽然迭代次数与模型的逼真度并无明显的线性关系,但随着迭代次数的增加,模型的逼真度也会随之相应增加.迭代次数越多,模型越接近于真实景物.

3) 树木复杂度对系统内存的影响

在利用改进的IFS植物几何建模法对树木进行模拟时,消耗系统的内部资源虽然比经典建模法小得多,但随着最终生成的树木模型的复杂度的增大以及形成的树木状态更新频率增大,系统内存消耗会越来越大.测试过程中,具体的图帧率 Y 与树木复杂度 O 之间的关系如图5所示,图5中横坐标的数值表示树木多边形面片的数量,多边形面片越多,对应的树木模型将越复杂;纵坐标表示对应生成的树木随风运动的动态画面的帧率.

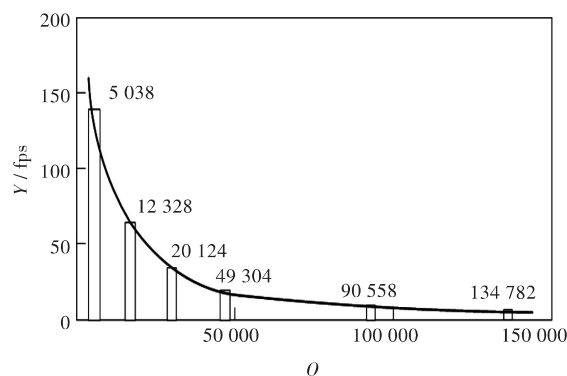


图5 帧率与树木复杂度的关系

Fig. 5 Relationship between frame rates and complexity of tree under test

通过图5中的数据可以得到结论:树木模型越复杂,则对应的帧率也就越低.帧率数值的大小影响着树木状态的更新速度,树木状态更新过快或过慢,都与现实场景不太符合.为此要严格控制风场模型中的噪声函数的频率,并通过频率控制来提高IFS动态树木模型的真实感.

4 结 语

以上着重讨论对改进的IFS植物的几何建模方法,主要为:对IFS静态植物建模过程进行设计,主要过程是利用IFS码构造树叶、结合圆柱体拼接法构造树干等.在树枝与树叶的构造过程中通过调用OpenGL库函数的方法以及三维坐标旋转法构造了树叶之间的差异性,增强了树木模型的真实感;引入遗传算法,对树木模型进行优化.在改进的迭代函数系统IFS植物几何建模方法中引入遗传算法,一方面,可以使构造出的树木模型更加贴近于现实;另一方面,使用该方法构造出的树木模型也是种群经过进化后产生的优质个体,这体现了自然界中优胜劣汰的生存规律;对IFS动态植物建模过程进行设计,模拟树木的动态过程主要通过引入风场来实现.在IFS动态植物建模过程中,风力对树木的影响主要分为树叶与树干两个

方面来考虑,在设计过程中考虑了树干受风力影响后的偏转运动及形变,但只考虑了树叶的偏转运动而忽略了树叶的形变.

参考文献:

- [1] PALMER I J, GRIMSDATE R L. Collision detection for animation using Sphere-Trees [J]. Computer graphics forum, 1995, 14 (2): 105-116.
- [2] 李自力. 虚拟现实基于图形与图像的混合建模技术[J]. 中国图形图像学报, 2001, 6 (1): 96-101.
LI Z L. Hybrid modeling technology based on graphics and image in virtual reality [J]. Journal of image and graphics, 2001, 6 (1): 96-101.
- [3] 孙长勇. 虚拟现实三维建模技术方法的分析与研究[D]. 郑州:解放军信息工程大学, 2004.
- [4] 刘倩. 树木三维可视化技术的应用研究[D]. 郑州:河南农业大学, 2008.
- [5] 李娜, 方恩浩, 滕飞, 等. 基于L-系统的分形植物模拟研究[J]. 计算机技术与发展, 2014, 24 (8): 199-205.
LI N, FANG E H, TENG F, et al. Research on fractal plant simulation based on L-system[J]. Computer technology and development, 2014, 24 (8): 199-205.

本文编辑:陈小平