

Study on fBm algorithm Generating 3D Realistic Terrain

Wu Yan

Department of Mathematics, LinYi Normal University, Shandong 276005
e-mail yanzi02801@sina.com

Abstract: Through the texture mapping technology provided by OpenGL, this article blended digital images and terrain model, puts forward a fBm algorithm based on texture mapping and gives the basic processes of the realistic terrain generation algorithm. In this algorithm, the outstanding advantage of generative terrain is efficient algorithm, faster graphics rendering and realistic scenes output.

Keywords: 3D realistic terrain; texture mapping; fBm algorithm

三维真实感地形生成的 fBm 算法

吴艳

临沂师范学院数学系, 山东 276000
e-mail yanzi02801@sina.com

【摘要】 本文通过 OpenGL 提供的纹理映射技术功能,把数字化图像与地形模型进行融合,提出了一种基于纹理映射技术的 fBm 算法,并给出了真实感地形生成算法的基本流程。应用该算法生成的地形,突出优点是算法效率高,图形渲染速度快,输出的场景真实感强。

【关键词】 三维真实感地形; 纹理映射; fBm 算法

1 引言

在真实感图形显示中,自然景物的模拟是最具挑战性的课题之一,因为自然景物不仅形状千姿百态,而且很多是实时变化的,其算法十分繁琐复杂,很难用一种常规的方法来描述。因此,自然景物的建模与绘制便成为计算机图形学中的一个研究热点和难点。与其它自然景物相比,地形自身的特殊性在于:其一,空间范围大、视野广。例如在飞行训练的模拟中尤其需要一个广视野的虚拟场景。其二,地形表面细节的复杂性。地表的起伏具有很大的随机性。三维真实感地形是可视化系统及虚拟仿真的基本组成部分,随着未来虚拟现实和仿真技术应用日益广泛,具有真实自然视觉效果的三维地形生成和建模技术显得越来越重要。

所谓真实感地形是通过对地形进行建模、透视投影、光照、消隐、纹理映射等操作后,生成像照片那样的黑白图像或彩色图像,产生仿真效果,生成的真实感地形图能够逼真地反映外部真实。为了提高地形模拟的真实感,国内外许多学者对此作了深入的研究。早在 1968 年, Mandelbrot 和 Van Ness 提出了一类一维高斯随机过程,称为“分维布朗运动”(fractional Brownian motion),简称 fBm。由于 fBm 很好地描绘了许多自然现象,所以,许多学者便开始尝试利用 fBm 方法来模拟真实感地形,其中,最为成

功的要属 Fournier 等人利用此法来模拟地域和各种星球表面的例子。但是,利用 fBm 方法模拟的地形与现实世界中的真实地形相比,其表面细节带有很大的的人工痕迹,真实感有待提高。为了更加逼真地模拟出高质量的地形,本文引入了纹理映射算法,提出了一种基于纹理映射的 fBm 算法,对 fBm 方法与纹理映射的有机结合作了有益的尝试。

2 地形模型构造

定义 fBm 分形模型的基本方法是递归,所以其计算量非常大,只有在计算机科学发展的今天才有可能付诸实施,也只有与计算机图形学的研究相结合,分形模型才得到广泛应用。

2.1 布朗运动^[1](fBm)

设 u 是 $(-\infty, +\infty)$ 中的一实数, w 是某一随机函数的值域,属于某个采样空间 W , 则普通的布朗运动 $B(u, w)$ 可定义为一个实随机函数。设 h 为一个参数,且 $0 \leq h \leq 1$, B_0 是任意实数,则参数为 h , 初值为 B_0 的 fBm $B_h(u, w)$ 为:

$$B_h(0, w) = b$$

$$B_h(u, w) - B_h(0, w) = 1/\Gamma(h+0.5) \int_{-\infty}^0 [(u-s)^{h-0.5}$$

$$-(s)^{h-0.5} \} d_{B(s,w)} + \int_0^u (u-s)^{h-0.5} d_{B(s,w)} \} \quad (1)$$

式(1)中,当 $h=0.5$ 时, $B_h(u,w)$ 就是普通的布朗运动 (fBm)。

利用 fBm 方法构造分形模型的主要步骤可概括为:

- (1) 给定初始值及初始条件;
- (2) 构造递归函数 Divide_Terrain(x[i],y[i]),
($i=0, 1, 2, 3, \dots$);

根据多边形的各顶点坐标计算各边的中点扰动后的坐标:

$$x_m = 0.5 * (x[i] + x[i+1]) + \delta * \text{RandG}();$$

$$y_m = 0.5 * (y[i] + y[i+1]) + \delta * \text{RandG}();$$

($i=0, 1, 2, 3, \dots$)

调用递归函数

```

.....
Divide_Terrain(x[i],y[i+1]);
.....

```

- (3) 连接各顶点和扰动后的点, 绘制分维曲面;

上述算法中, RandG() 是服从正态分布的 Gauss 随机函数, 可利用伪随机发生器 Rand() 生成。

2.2 分形模型的构造

在此, 我们利用上面的算法来建立地形的分形模型^[2]。图 1 所示, $P_1P_2P_3P_4$ 为一空间四边形, 首先对 $P_1P_2P_3P_4$ 上每条线段的中点产生扰动点 P_{12} 、 P_{13} 、 P_{14} 、 P_{11} , 再对线段

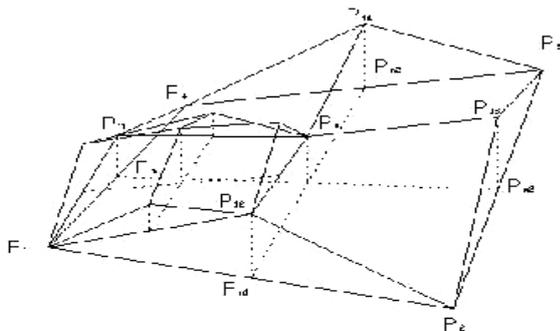


Figure 1. Sub-type model
图 1. 分形模型

$P_{m1}P_{m2}$ 和 $P_{n1}P_{n2}$ 使用同样的过程以生成它们的扰动点, 这两个扰动点的中点 P_0 便是细分后四个空间小四边形的公共点。四个小四边形为 $P_1P_{12}P_0P_{11}$ 、 $P_{12}P_2P_{13}P_0$ 、 $P_{13}P_3P_{14}P_0$ 、 $P_{14}P_4P_{11}P_0$, 然后再对这四个小四边形运用同样的方法进行细分, 这种细分过程一直进行到满足一定深度为止, 最后调用消隐函数:

```

glEnable(GL_DEPTH_TEST);/*深度测试*/
对生成的分形模型进行消隐。调用以下函数来设置光照模型:

```

```

.....
glEnable(GL_LIGHTING);/*启用光照*/
glLightModeli(GL_LIGHT_MODEL_TWO_SIZE, GL_TRUE);/*设置光照模型*/
glEnable(GL_COLOR_MATERIAL); /*启用颜色跟踪材质*/
.....

```

这样便生成了具有初步真实感的地形。

3 纹理映射

在 OpenGL 支持下, 经过分形模型构造、消隐、设置光照模型等一系列函数的调用, 即可模拟出具有初步真实感的地形, 在此基础上, 通过 OpenGL 提供的纹理映射技术功能把数字化图像与地形模型进行融合, 可进一步提高地形的表现力, 增强地形的真实感。

3.1 纹理的定义

纹理一般定义在单位正方形 ($0 \leq u \leq 1, 0 \leq v \leq 1$) 之上, 称为纹理空间^[3]。定义纹理一般采用两种方法:

- (1) 用连续法定义纹理函数。如:

$$f(u,w) = A[\cos(pu) + \cos(qw)] \quad (2)$$

在式(2)中, A 为 [0,1] 区域上的随机变量, p、q 为频率系数; u、w 为函数参数。该纹理函数可模拟粗布的纹理。

- (2) 用离散法来定义纹理模型^[4]。与函数纹理相比, 这种方法较为简便。此法的基本思想就是把纹理图像数据定义在一个二维数组中。本文就是采用离散法定义数字化图像, 图 2 所示为表现地形表面细节的数字化图像, 图 3 所示为天空的纹理图像。



Figure 2. Earth's surface texture image
图 2. 地表纹理图像



Figure 3. Sky texture image
图 3. 天空纹理图像

在有了数字化图像之后, 接下来就是调用纹理函数来生成符合纹理映射要求的纹理图像。

若数字化图像宽为 Width, 高为 Height, 每个像素 8bit, 图像数据信息存储在二维纹理数组 bits 中, 则调用以下函数即可生成二维纹理图像:

```

.....
gluBuild2Dmipmaps(type,3,Width,Height,GL_R
GB,GL_UNSIGNED_BYTE,bits);
.....
    
```

3.2 实现纹理映射

在定义纹理的基础上, 利用纹理映射把所定义的纹理图像与地形模型进行融合。本文采用纹理映射方式为 Mipmaps 方式, 其基本思想是以适当大小的正方形来近似表达每一像素在纹理平面上的映射区域。在 OpenGL 中, 通过调用以下函数来实现这一映射^[5]:

```

.....
gluBuild2Dmipmaps(type,3,Width,Height,GL_R
GB,GL_UNSIGNED_BYTE,bits);/*定义纹理数组*/
glEnable(GL_TEXTURE_2D); /*启用纹
理映射*/
glBindTexture(GL_TEXTURE_2D, filename);/*
实现纹理图像与模型的融合*/
.....
    
```

若要表现不同环境下的地形, 可以运用雾化效果来模拟雾化场景中的地形, 添加雾化效果将使场景显得更加真实。

4 真实感地形生成算法的基本流程

本节将给出计算机模拟真实感地形的一般步骤, 如图 4 所示:

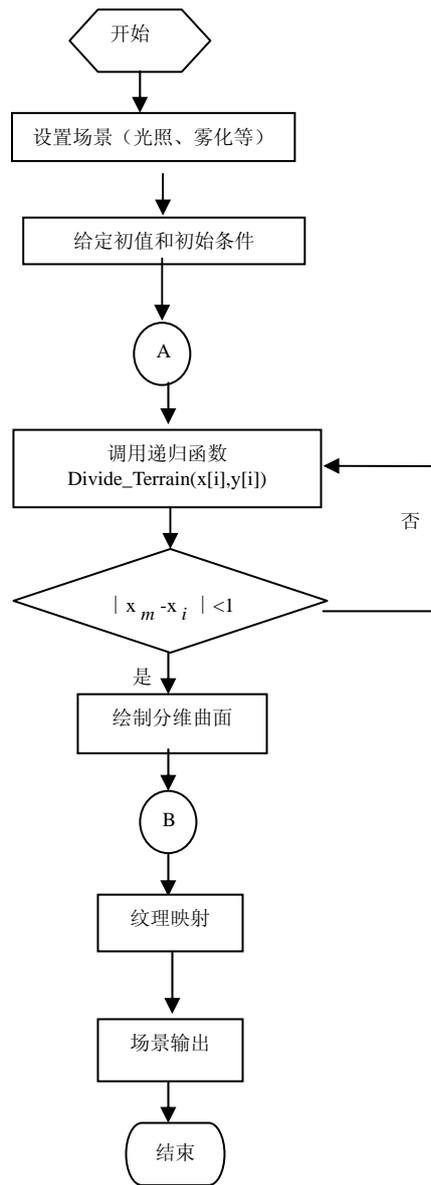


Figure 4 . Algorithm diagram of realistic terrain
图 4. 真实感地形算法框图

5 结束语

本文给出的基于纹理映射技术的 fBm 算法, 其突出优点是算法效率高, 图形渲染速度快, 输出的场景真实感强。

根据本文给出的算法, 我们在 CPU 为 PIII800, 内存 128M, Win98 环境下, 以 VC++6.0 为开发平台, OpenGL 为工具, 进行了真实感地形的模拟, 下面是两幅模拟的场景。图 5 为无雾化效果场景, 图 6 为雾化效果场景。从输出结果可以看出, 该算法模拟的地形真实感强, 能够满足军事训练模拟、飞行仿真等领域对地形的真实感要求。下一步, 我们将在场景中加入



Figure 5. Non-atomized scene
图 5. 无雾化效果场景



Figure 6. Atomization scene
图 6. 雾化效果场景

树木、湖泊、远山等景物，开发一个基于虚拟现实的环境仿真系统。

致谢

在论文的写作过程中，得到了许多朋友的宝贵建议，同时还得到同事和家人的支持和帮助，在此一并致以诚挚的谢意。

References (参考文献)

- [1] Tang rongxi, et al. Computer graphics tutorial[M]. Beijing: Science Press, 1990.
唐荣锡等. 计算机图形学教程[M]. 北京: 科学出版社, 1990.
- [2] Qu baiqing, Xu shejiao. Computer graphics tutorial[M]. Xi'an: Xidian University Press, 1998.
璩柏青, 许社教. 计算机图形学[M]. 西安: 西安电子科技大学, 1998.
- [3] Yan laibin, Li sikun, et al. Research of waves real-time generated technology[J], Journal of Computer-Aided Design and Graphics, 2000, (9): 715-719.
鄢来斌, 李思昆等, 海浪实时生成技术研究[J], 计算机辅助设计与图形学学报, 2000, (9): 715-719.
- [4] Zhang Qin, et al. Over view of modeling for irregular objects[J], Journal of Image and Graphics, 2000, 5(3): 186-190.
- [5] Xue an, Ma ainai, Li tianhong. A Study on the Representation of Terrain Realistic Image Based on OpenGL[J], Journal of image and graphics, 2001, 6 (8).
薛安, 马蔼乃, 李天宏, 基于OpenGL实现真实感地形表现的研究[J], 中国图像图形学报A辑2001, 6(8).