

虚拟场景中的实时降雪模拟

王顺领^{1,2}, 李学庆¹, 王善斌¹, 温英涛¹

(1. 山东大学 计算机科学与技术学院, 山东 济南 250061;

2. 蚌埠坦克学院 军事仿真训练中心, 安徽 蚌埠 233013)

摘要:降雪是一种重要的自然现象,对降雪的模拟可以大大提高虚拟场景的逼真度。文中在分析粒子系统建模方法的基础上,给出了一种雪粒子模型,该模型对雪的物理模型进行了适当简化,通过控制雪粒子的大小、形状、密度(数量)来实现雪强度的模拟;通过控制雪粒子的运动来模拟风对雪的影响,并对雪粒子的产生域进行了讨论,使其满足视点运动的要求。实践表明,该粒子系统模型可在复杂的三维场景中实时地模拟降雪现象,仿真效果逼真。

关键词:粒子系统;虚拟场景;降雪;模拟

中图分类号:TP391.9

文献标识码:A

文章编号:1005-3751(2006)03-0152-03

Real-Time Simulation of Snowfall in Virtual Scene

WANG Shun-ling^{1,2}, LI Xue-qing¹, WANG Shan-bin¹, WEN Ying-tao¹

(1. School of Computer Science & Technology, Shandong University, Jinan 250061, China;

2. Centre of Military Simulation Training, Bengbu Tank Institute, Bengbu 233013, China)

Abstract: Snowfall is an important phenomena of nature. Simulation of snowfall will improve lifelike qualities of virtual scene. In this paper, present a mode of snow based on analyzing particle systems. This mode simplified the attributes of real snow, this mode use size, shape and density of snows particle to simulate intensity of snow, and control movement of particle to simulate action of wind. At the same time, study the generation shape of snow particles and make it settle for the situation that the view is moved. The results of experimentation have proved that this method is more effective in the representation of snows in virtual scene.

Key words: particle systems; virtual scene; snowfall; simulation

0 引言

自然现象的模拟是虚拟现实领域的一个重要研究内容,对自然现象的模拟可以大大提高虚拟场景的逼真度。与规则几何体不同,自然景象往往包含有丰富或随机变化的形状,它们很难用传统的解析曲面来描述。1983年Reeves^[1]提出的粒子系统较好地解决了这个问题。它是采用许多形状简单的微小粒子作为基本元素来表示不规则模糊物体;充分体现了不规则模糊物体的动态性和随机性,被公认为模拟不规则物体最为成功的一种图形生成方法^[2]。降雪是很重要的一种自然现象,文中在分析粒子系统的基础上,从视景仿真的实时性要求出发,建立并实现了一种能够模拟雪现象的粒子系统。同时通过对粒子分布域的讨论,解决了由于视点运动带来的问题。该模型对雪现象的物理模型进行了适当的简化,在保证粒子系统实时性的前提下,达到了较高的逼真度。

1 粒子系统建模方法

粒子系统的基本思想是:采用大量的、具有一定生命和属性的微小粒子图元作为基本元素来描述不规则的模糊物体。景物被定义为由成千上万个不规则的、随机分布的粒子组成,随着时间的推移,已有的粒子不断改变形状、不断运动,而且不断有新粒子加入、旧粒子消失。为了模拟粒子的生长和死亡的过程,每个粒子均有一定的生命周期,使其经历出生、成长、衰老和死亡的过程。在每一时刻(即每一帧)只有其中的一部分是“生存”的,因此,在粒子系统的应用中,粒子的诞生、死亡的规律必须在一定程度上符合建模对象的物理规律,才能比较准确地模拟其特性。同时,为表达粒子系统的随机性,采用一些非常简化的随机过程来控制粒子在它所在系统的形状、特征及运动。对每一粒子参数均确定其变化范围,然后在该范围内随机地确定它的值,而其变化范围则由给定的平均期望值和最大方差来确定,基本表达式为^[3]:

$$\text{Parameter} = \text{MeanParameter} + \text{Rand}() \times \text{VarParameter}$$

(1)

其中Parameter代表粒子系统中的任一需随机确定的参数,Rand()为产生在-1到1之间的均匀随机函数,VarParameter为其方差。

收稿日期:2005-06-20

作者简介:王顺领(1975—),男,山东人,硕士研究生,讲师,研究方向为虚拟现实与视景仿真;李学庆,教授,博士生导师,研究方向为人机交互与虚拟现实。

由此可以得出以下粒子系统建模方法的特点:

(1)对物体的描述不是通过原始的具有边界的面元(如多边形集合)来描述,而是通过一组定义在空间中的原始粒子来描述。

(2)粒子系统不是一个静态实体,每个实体的属性都是时间的函数。

(3)由粒子系统描述的物体不是预先定义好的,其属性均用随机过程来描述。

实现粒子系统的基本步骤如图1所示。

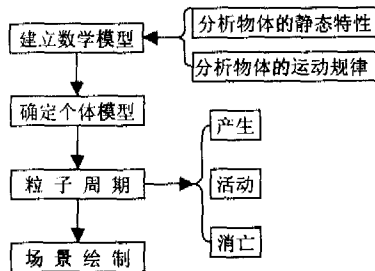


图1 实现粒子系统的基本步骤

2 基于粒子系统的降雪模型

从视景仿真的角度出发,采用粒子系统实现降雪现象的模拟,没有必要完全按照雪的物理模型来建模,那样是非常复杂和难以实现的。首先对现实中雪的物理特性进行适当的简化,忽略了雪粒子之间的相互作用,用一个粒子来代表一片雪花;通过控制雪粒子的大小、形状、密度(数量)来实现雪强度的模拟;通过控制雪粒子的运动来模拟风对雪的影响;通过对粒子的产生域的限制来减少粒子的数量,满足场景的实时性要求。具体模型如下。

2.1 分布域的确定

现实中的雪产生在天空中某一定高度的云层处,并且满天飞舞在天空中的每一处。与之相对应,在虚拟场景中,雪的理想产生域应该是覆盖整个场景的具有一定高度的平面,雪粒子应该分布在三维空间。然而计算机的资源是有限的,粒子的数量的多少直接影响三维场景的实时性。仅仅从视觉效果立足点出发,为了减少产生粒子的数量,人们对粒子的分布域进行简化。最典型的是文献[4]提出的将粒子系统的分布区域选在视锥内前的一个长方体内,利用粒子系统的分布区域即长方体随视点一起运动使起点始终在视点的前方来减少粒子的数量。文中没有讨论当视点运动时出现的问题:当视点转动时,如果粒子也跟着视点转动,就违反了现实中观察雪的规律;如果粒子不跟着分布域转动,当视点运动的速度较快时,将会出现部分区域的下半部分没有雪粒子;如果把分布域设的大些,就会失去减少粒子的意义。为了比较好地解决这个问题,提出了以下粒子分布域模型:将粒子分布域定义在视锥前的一个扇形环形体内,如图2所示。

产生域随视点一起运动,与视点的相对位置不变。为了解决视点转动的问题,将扇形的角度大于视场角角度,扇形角度 φ 与视场角 θ 、视点转动的最大角速度 $\maxRo-$

tate、粒子的平均下降速度 fallSpeed 、粒子分布域的高度 H 有关,计算公式如下:

$$\varphi = \theta + 2 \times (H \div \text{fallSpeed}) \times \max\text{Rotate} \quad (2)$$

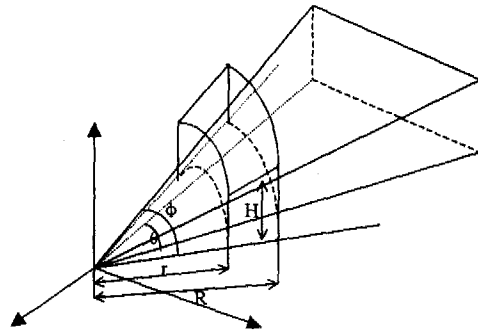


图2 简化的粒子分布域

使用该粒子分布域最大的优点是它的基本形状与视锥相似,大大减少了无效粒子的产生,同时该分布域利用极坐标比较容易计算粒子的产生位置和粒子的消亡。

在虚拟场景漫游系统中,经常会出现观察点的跳跃的情况,即视点变化的幅度比较大,例如允许用户自由选择观察点。这种情况下,可以使在雪粒子跟随视点一起运动,即每个活动的粒子产生与视点一样变换。这样就比较好地解决了视点运动的问题。

2.2 粒子的产生

2.2.1 新产生粒子数量的计算

在粒子系统中,活动的粒子不断消亡,同时新粒子不断产生,通过这一过程来描述降雪的动态过程。新产生的粒子数量是一个比较难解决的问题,通常的做法是每一帧产生固定数量的粒子,这种方法的优点是计算简单,但是由于虚拟场景在漫游过程中场景的复杂度是不一样的,因而两帧图像之间的时间间隔并不是固定的,从而会导致空间的粒子分布不均匀,严重影响效果。文中的思想是根据粒子的体密度(单位体积内的粒子数量 density)和粒子的平均下降速度 fallSpeed 来计算每一帧产生的粒子数量 newNumber ,以保证粒子在三维空间的分布的均匀性。

计算公式如下:

每一帧新产生的粒子数量 = 产生域的面积 × 粒子下降的平均速度 × 粒子系统的体密度 × 两帧图像间的时间间隔

具体到文中利用的粒子产生域,其计算公式如下:

$$\text{newNumber} = (\varphi \div 360) \times \text{PI} \times (R \times R - r \times r) \times \text{fallSpeed} \times \text{density} \times dt \quad (3)$$

其中 π 为圆周率, φ 是扇形的角度, R,r 是两扇形半径(见图2), dt 是这一帧图像与前一帧图像之间的时间间隔。

2.2.2 赋予新粒子的初始属性

新产生的粒子必须赋予新的属性,为了表示粒子的不规则性,在赋予粒子属性的时候分别加了扰动即方差,计算公式都是按照公式(1)来计算的。

a. 位置:粒子的初始位置 pos 是在分布域顶部的扇形环上,计算公式如下:

$$\text{pos. } x = (r + (R - r) \times 0.5 \times \text{Rand}()) \times \cos(\text{viewAngle} + \text{Rand}() \times \varphi \times 0.5)$$

$$\text{pos. } y = \text{分布域顶部的 } Y \text{ 值} + \text{Rand}() \times y\text{Disturb}$$

$$\text{pos. } z = (r + (R - r) \times 0.5 \times \text{Rand}()) \times \sin(\text{viewAngle} + \text{Rand}() \times \varphi \times 0.5)$$

其中 viewAngle 是视线的方向角, yDisturb 是产生新粒子位置在 Y 方向的方差。利用在 Y 轴方向的位置方差使新产生的粒子分布在一个三维空间,而不是在同一个平面上。

b. 角度:为了使初始的粒子形态各异,给每个粒子赋予绕 3 个坐标轴转动的角度 angle,计算公式如下:

$$\text{angle. } x = \text{Rand}() \times \pi$$

$$\text{angle. } y = \text{Rand}() \times \pi$$

$$\text{angle. } z = \text{Rand}() \times \pi$$

c. 速度:为了描述风对雪粒子的影响加入了风速 windSpeed 和风向 windAngle 参数;理想中雪粒子在重力的作用下应该做匀加速运动,但由于雪粒子比较轻,受到的空气阻力相对比较大,因此雪粒子在下降的过程中并不是做匀加速运动,此处把雪粒子简化为匀速下落,速度计算公式如下:

$$\text{velocity. } x = \text{windSpeed} \times \cos(\text{windAngle}) + \text{Rand}() \times v\text{Disturb. } x$$

$$\text{velocity. } y = \text{fallSpeed} + \text{Rand}() \times \text{velocityDisturb. } y$$

$$\text{velocity. } z = \text{windSpeed} \times \sin(\text{windAngle}) + \text{Rand}() \times v\text{Disturb. } z$$

其中 vDisturb 是粒子在 x, y, z 方向上的速度方差。

d. 大小:粒子的大小 size 计算公式如下:

$$\text{size} = \text{avSize} + \text{Rand}() \times s\text{Disturb}$$

其中 avSize 是粒子的平均大小, avSize 的取值和降雪的强度有关; sDisturb 是大小方差。

2.3 粒子属性的更新

只要粒子处于活动状态,每一帧都要对粒子的属性进行更新。雪粒子属性的更新主要包括:粒子的位置、角度更新。

雪粒子由于各种原因在空中并不是做直线运动,而是不规则地在空中漂浮,因此在位置进行更新的过程中,在 x, z 方向上除了加上了风力的影响,还加上了扰动。同时为了使下降中的雪花不在一个平面上,在 y 方向上对速度加了扰动。当前帧粒子的位置 pos 与上一帧的位置 ppos 的计算关系如下:

$$\text{pos. } x = \text{ppos. } x + (\text{velocity. } x + \text{Rand}() \times v\text{Disturb. } x) \times dt$$

$$\text{pos. } y = \text{ppos. } y + (\text{velocity. } y + \text{Rand}() \times v\text{Disturb. } y) \times dt$$

$$\text{pos. } z = \text{ppos. } z + (\text{velocity. } z + \text{Rand}() \times v\text{Disturb. } z) \times dt$$

由于空气阻力和风的作用,雪粒子在下降过程中总是做不规则的转动,因而在对雪粒子角度进行更新时转动速

度和方向应该是随机的,计算公式如下:

$$\text{angle. } x = \text{pangle. } x + \text{Rand}() \times r\text{Disturb. } x$$

$$\text{angle. } y = \text{pangle. } y + \text{Rand}() \times r\text{Disturb. } y$$

$$\text{angle. } z = \text{pangle. } z + \text{Rand}() \times r\text{Disturb. } z$$

其中 pangle 为粒子上一帧转动的角度; rDisturb 是转动的扰动角速度,注意 rDisturb 的值不能太大,并且在 x, y, z 方向上的值相等。

需要说明的是,由于对雪粒子的消亡的计算不是以粒子的寿命为标准,因而未对粒子的寿命进行讨论。

2.4 粒子的消亡

当粒子的存活时间超过粒子的生存期时,该粒子就应该消亡了。在实际的应用中考虑到系统的实时性,那些对视景效果没有贡献的粒子也应该消除,这样可以加快系统的绘制速度。文中给出的条件是当粒子不在分布域中或者粒子落在了地上,就认为该粒子对视景没有了贡献,应该消亡,从粒子系统中删除。

由于与场景中的景物相比,雪粒子本身的尺寸很小,因此粒子位置与粒子分布域关系的判断可简化为三维空间中的点与粒子分布域的关系。由前面的介绍可知,粒子分布域是一个扇形环形体,其位置相对视点的位置是不变的,因而粒子存在于分布域的条件是:粒子与视点在 X - Z 平面上的距离 distance 应该在两扇形半径之间,即 $r < \text{distance} < R$; 粒子与视线在 X - Z 平面上的夹角 η 的绝对值应该小于扇形角度的一半,即 $|\eta| < (\varphi / 2)$; 粒子在 Y 轴的坐标应该在分布域的最大和最小 Y 值之间。以上 3 个条件必须同时满足,有任一条件不满足,即可认为该粒子消亡。由于分布域是扇形环形体,角度和半径都是已知的,因而以上条件都比较好计算。

要判断粒子落在了地上或场景中的其它景物上,需要遍历虚拟空间中的模型的所有片元,并与粒子下落的轨迹求交。这样对任何复杂一点的场景,该项操作的时间开销都会很大,难以实时完成。因此将该条件简化为当粒子的垂向坐标值 Y 小于其所在位置的地平面垂向坐标值时,即认为该粒子消亡。

2.5 粒子的绘制

当完成粒子的状态设置后,需要根据粒子的状态对其进行绘制。文献[5]把每个粒子都视为一个带有透明纹理的四边形,绘制时进行单面绘制,为保证生命周期内的粒子都始终可见,所有粒子的法向量与视锥体远平面的法向量相同,即为视线方向的负方向。这种方法是把雪粒子简化成面粒子,对提高计算速度是不错的方法,但用面粒子来简化雪粒子,有些不够真实,尤其是对粒子进行旋转或者粒子距视点较近时,就明显地有些失真。针对这种问题,采用了体粒子和面粒子两种粒子模型来简化雪粒子:当粒子产生在距视点较远的位置时采用文献[5]的面粒子;当粒子产生在距视点较近的位置时用体粒子,即用两个垂直交叉的带有透明纹理的四边形,进行双面绘制。为

(下转第 157 页)

$e=1$ 时为 $G(39,15)$, $e=2$ 时为 $G(95,40)$ 。根据定理 1, 易得紧优 $G(N;s)$ 的无限族 $\{G(3t^2+4t;s):t=2e+1,s=6e^2+13e+6,e \geq 1\}$, $e=1$ 时为 $G(39,25)$, $e=2$ 时为 $G(95,56)$ 。

在文献[2]中列出了 69 类紧优和 33 类几乎紧优双环网络无限族,均可根据定理 1,列出对称的 69 类紧优和 33 类几乎紧优双环网络无限族。

徐俊明在文献[1]中研究了双环网络的另一种图论模型 $G(N;r,s)$,根据定理 2,文献[1]中的定理 1,2,3 也可推广。如由文献[1]中的定理 3 可得:

命题 A 设 Z 是非负整数无限集,则 $\{G(3t^2+4t-11;3,3t^2+t-6):t \in Z,t \geq 59\}$ 是一个紧优双环网络无限族,其直径为 $3t$;而 $\{G(21168e^2+30072e+10668;3,21168e^2+29820e+10496):e \in Z\}$ 是其中一个奇异的无限子族,其直径为 $252e+177$;它们的起始元素是 $G(10668;3,10496)$,其直径为 177;当 $e=1$ 时为 $G(61908;3,61484)$,其直径为 429。

徐俊明等人在文献[6]中研究了一类 4 紧优双环网络无限族,根据定理 1,文献[6]中的定理 3,4 也可推广。如由文献[6]中的定理 4 可得:

命题 B 设 $n(t) = 3t^2 + 6t - 26$, $t = t(g) = 14812g^2 + 3036g + 151$, $s(g) = 643877640g^4 + 263638788g^3 + 40175832g^2 + 2700207g + 67520$, 则 $\{G$

$(n(t(g));s(g):g = g(e) = 22 \cdot 85^2 e, e \in Z)\}$ 是 4 紧优双环网络无限族,直径为 $3t+5$,起始元素是 $G(69283;67520)$,直径为 458。

3 结束语

文中从双环网络的图论模型出发,研究了双环网络直径的对称性,若 $G(N;r,s)$ 的直径 $d(N;r,s) = d(N)$,则称 $G(N;r,s)$ 是最优的。因而搜寻最优或紧优 $G(N;r,s)$ 的范围可以减少一半。

参考文献:

- [1] 徐俊明. 计算机互连双环网络的最优设计[J]. 中国科学, E 辑, 1999, 29(3): 272-278.
- [2] 李乔, 徐俊明, 张忠良. 最优双环网络的无限族[J]. 中国科学, A 辑, 1993, 23(9): 979-992.
- [3] 徐俊明. 不含紧优和几乎紧优双环网络无限族[J]. 科学通报, 1999, 44(5): 486-490.
- [4] 徐俊明. 2 紧优双环网络无限族[J]. 高校应用数学学报, A 辑, 2000, 15(2): 148-152.
- [5] 徐俊明, 尹治军. 7 类新的 2 紧优双环网络无限族[J]. 中国科学技术大学学报, 2002, 32(4): 399-402.
- [6] 徐俊明, 刘琦. 一类 4 紧优双环网络无限族[J]. 中国科学, A 辑, 2003, 33(1): 71-74.

(上接第 154 页)

了表现雪粒子形状是不规则的,文中采用了 3 种纹理,在粒子产生时随机的选取。这样比较好地解决了绘制速度和真实性的矛盾,达到了系统需要的性能。

3 实验结果和结论

笔者把该降雪模型应用在坦克分队战术综合演练仿真系统中,通过设置粒子系统的密度、粒子的大小和粒子下降的速度来模拟下雪的强度,通过赋予粒子在 x, z 轴方向上一定的速度来模拟风对雪的影响。用链表结构来管理粒子系统,每一帧绘制粒子时先对其判断是否消亡,再对活动的粒子进行绘制,很好地保证了处于活动状态的粒子数量。实验证明,该降雪模型能够满足一定的需要。部分结果如图 3、图 4 所示。

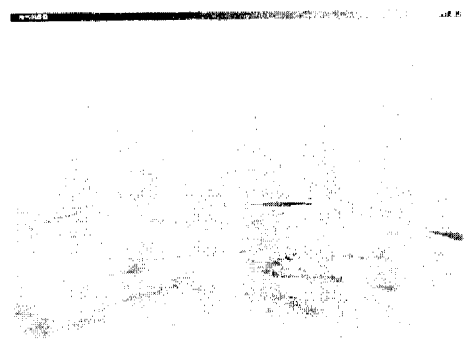


图 3 大雪效果

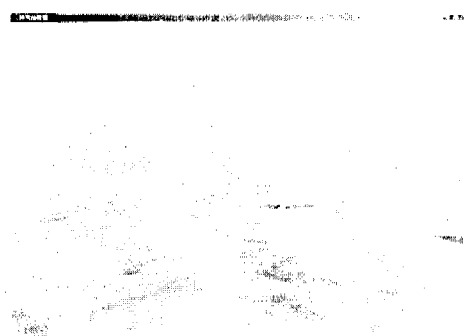


图 4 小雪效果

参考文献:

- [1] Reeves W T. Particle Systems - A Technique for Modelling a Class of Fuzzy Objects[J]. Computer Graphics, 1983, 17(3): 359-376.
- [2] 张芹. 基于粒子系统的建模方法研究[J]. 计算机科学, 2003, 30: 144-146.
- [3] 唐熔锡, 汪嘉业, 彭群生, 等. 计算机图形学教程[M]. 北京: 科学出版社, 2002.
- [4] 王润杰. 基于粒子系统的实时雨雪模拟[J]. 系统仿真学报, 2003, 15(4): 495-496.
- [5] 苏虎. 列车驾驶视景仿真中自然现象的模拟[J]. 铁道科学, 2003, 24(1): 19-22.