

# 基于 OGRE 粒子系统在烟花渲染中的研究

蔡政策, 魏 臻, 凌 勇, 程 磊

(合肥工业大学 计算机与信息学院, 安徽 合肥 230009)

**摘 要:**为了实现对自然现象和景物及日常生活进行真实模拟, 在游戏和虚拟生活中都要创造大量的虚拟场景, 使用虚拟渲染也节省开发成本, 加上粒子系统在计算机图形学中的地位十分重要。文中在 OGRE (Object-Oriented Graphics Rendering Engine) 引擎架构上, 对粒子系统的原理、结构及其脚本语言、粒子系统的烟花虚拟渲染流程进行了研究, 文中对以往烟花渲染进行了改进, 提出一种程序和脚本相结合的方法, 通过 OGRE 引擎平台进行实验论证, 实验结果取得了较好的视觉效果, 证明文中提出的方法可行, 而且节约开发时间和提高开发效率。

**关键词:**虚拟场景; 粒子系统; OGRE 引擎; 脚本语言

**中图分类号:** TP39

**文献标识码:** A

**文章编号:** 1673-629X(2011)10-0088-04

## Research Based on OGRE Particle System Rendering of Fireworks

CAI Zheng-ce, WEI Zhen, LING Yong, CHENG Lei

(Department of Computer and Information, Hefei University of Technology, Hefei 230009, China)

**Abstract:** In order to realize the scenery and natural phenomena and everyday life for the real simulation, both in the game and virtual life should create a large number of virtual scene, virtual rendering also saves development costs, and position of in the computer graphics Particle system is very important. The article in the OGRE engine architecture, the principle, structure and the virtual rendering process of fireworks particle system has been studied. Through the process achieves virtual particle system rendering fireworks, this paper has been improved in the past fireworks rendering, and propose a procedure and method of combining the script, using the OGRE engine's platform to make experimental demonstration, the results achieves good visual effects, demonstrate the proposed method is feasible, and saving development time and improving development efficiency.

**Key words:** virtual scene; particle system; OGRE engine; scripting language

## 0 引言

随着三维虚拟技术<sup>[1]</sup>的高速发展, 电影和游戏领域开始采用图形方式来完成对自然现象和景物的仿真渲染。特别是在高科技领域的应用如卫星发射、现代化的军事战争等<sup>[2]</sup>, 用来替代实体的演示, 这样更加简便、形象和直观, 而且节省了成本, 提高了效率, 同样可以取得视觉冲击的真实效果。文中主要研究了 OGRE 引擎下粒子和粒子系统的基本原理, 通过程序不断的使粒子发生器发射数量众多的粒子, 这样给人视觉上产生烟、火一样的场景。

OGRE 使用的以 C++ 语言为编程基础且面向自然界的对象 3D 引擎。开发者使用 C++ 可以十分容易或

不需要了解底层就能开发 3D 虚拟的应用场景<sup>[3]</sup>。3D 引擎的底层细节在类库基础上抽象, 自然界的真实对象的接口或者定义的类都可以由 3D 引擎提供。OGRE 是一个成熟、稳定、可靠、灵活、跨平台而且拥有丰富功能实时 3D 图形库。其中粒子系统既可实现对自然界的山、水、云、烟等景物的真实模拟<sup>[4]</sup>, 也可实现对烟花等日常娱乐生活的真实模拟。是一种基于物理模型来解决问题的方法, 其核心不是怎样去显示, 而是对微小物质模型的规则提取。粒子系统在三维显示中占有很重要的地位, 如 3D 中的雨、雪、喷泉、爆炸效果等都是粒子系统神奇魅力的表现。

## 1 OGRE 引擎的构架

OGRE 引擎就好像现实生活车辆的引擎, 渲染速度的快慢、能否产生真实的感觉、视觉感的好坏等都由其决定, 和车辆的发动机作用一样, 把所要虚拟演示的全部元素结合在一块, 在“人人看不到的后台”监控引导他们工作。通俗的讲, OGRE 的引擎可被称为“虚拟演示的场景控制的主程序, 不论是碰撞计算、物理引擎

收稿日期: 2011-03-25; 修回日期: 2011-06-13

基金项目: 安徽省科技攻关项目 (07010202045); 国家“863”计划资助项目 (2004AA001021)

作者简介: 蔡政策 (1983-), 男, 硕士, 研究方向为三维虚拟、无线网络; 魏 臻, 教授, 博士生导师, 研究方向为企业铁路智能交通、矿山运输安全控制、分布式控制与容错技术等。

系统的计算、物体在虚拟场景中的位置,还是用户的输入及能否输出用户满意的声音等”。系统负责初始化、更新、关闭这些子系统。可以具体分为如下几个子系统,如图 1 所示。

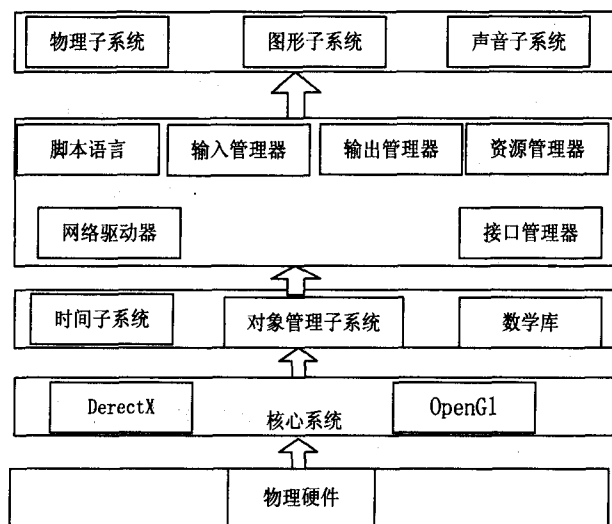


图 1 引擎构架

(1) 第一层物理硬件:引擎的构件主要在硬件上,主要操作硬件实体(独立显卡)。

(2) 第二层硬件的接口层,它是 OGRE 引擎的基础层,引擎的底层细节在类库基础上抽象,自然界的真实对象的接口或者定义的类都可以由引擎提供,有关硬件专用的驱动函数包也由其提供。需要的图形、外部设备、模拟的声音、提供给用户的视频、流媒体的接口等其他的部件也可由其提供。

(3) 第三层提供 OGRE 引擎的总体结构框架,并配有时间序列的帧控制,本类封装了基本的数学函数,大部分函数都是 C 运行库中相应函数的别名。使用这种方式的原因在于可以在这里提供额外的特性和级别较高的专门的数学函数集的功能。

(4) 第四层为引擎的控制层,使用的脚本语言、用户使用的输入输出系统、有关网络的虚拟驱动、资源使用方面的管理、不同种类的接口间的管理都是控制层的主要组成部分。使用的脚本语言可以让资源的配置变得灵活,根据虚拟的要求驱动对应的资源,当脚本需要更改时没有必要再次编码,输入和输出对应的子系统对外设进行管理,网络驱动可以为所要虚拟的系统提供所需要的网络连接与相应的平台搭建,接口管理和控制台相连接并适时做出不同的处理。

(5) 第五层主要由物理引擎系统、图形渲染系统、声音效果模拟系统。声音效果模拟系统主要完成每种声音的回放及声音的加载,物理引擎系统能够完成虚拟情形下的物理碰撞及算法的实时计算,图形渲染系统负责虚拟场景的全部图形和界面的相关驱动与效果功能。如水面的波纹、各种火花、浓烟和薄雾效果,其

中烟火是文中所要研究和实现的特殊效果之一。

## 2 OGRE 的粒子系统原理

主要工作原理是使用大量的形状相对简单的细微粒子用作基本单位的元素去虚拟渲染大自然中不规则及视觉看不清楚的场景。目前国内外用来模拟不规则的场景只有粒子系统是做的最完美的一种图形渲染算法,目前这个研究范围领域内,Fourier 的提出的分型算法及 Reeves 的粒子模拟系统原理是非常具有代表性的。

OGRE 的粒子系统虚拟渲染自然界众多不规则的对象主要就是使用大量的微粒作基本单位的元素聚集渲染的,这些微小粒子构成的粒子渲染系统是封闭的。在渲染模拟过程中,已经模拟使用过的粒子会消失或把自身的渲染状态及其它模拟属性进行改变。在旧的粒子渲染后新的粒子也会伴随连续不断的出现,渲染过程中所有的粒子都是有周期的,和人一样分为粒子的出生、成长、渲染过程后的衰竭甚至死亡。

### 2.1 粒子系统结构

模拟自然界的不规则的物体的引擎图形粒子原理是 William. T. Reeves 1983 年提出的。虽然是微小的粒子、但是在渲染中采用四边形表示,和四边形一样分为长和宽,因为渲染具有随机性,所以粒子系统还要有方向、渲染的数量、使用的材质和重量<sup>[5]</sup>。上面提到的属性封装在粒子属性变换器内,变换器里面含有颜色减弱变换器及线性决定变化器,粒子发射器发射模拟过程出现的连续不断的粒子。盒子形状发射器及点状粒子发射器是其主要发射器,这些就构成了渲染的脚本文件,.particle 是粒子渲染文件的扩展名。点状粒子发射器自由的从单独的一个地方发射粒子,而盒子形状发射器则是自由的从某个区域发射出渲染粒子。使用 Plugins 方法渲染者可以方便的在 OGRE 任意添加你所需要的发射器<sup>[6]</sup>。渲染过程中粒子会连续不断地发射出大量的渲染粒子,这样就能够模拟自然界的浓烟薄雾、火花及任何自然界的爆炸现象。OGRE 之所以加上脚本的粒子系统语言,主要因为是不需要改变程序而只需要去变更粒子的基本属性就可以渲染出不同的效果<sup>[7]</sup>,非常利于开发者的使用。

粒子系统结构图如图 2 所示。

### 2.2 烟花粒子渲染流程

OGRE 的粒子系统虚拟渲染自然界众多不规则的对象主要就是使用大量的微粒作基本单位的元素聚集渲染的,这些微小粒子构成的粒子渲染系统是封闭的<sup>[8]</sup>。且每一个粒子都有周期性,所以充分体现了所模拟物体的动态性和随机性。图 3 就是文中烟花粒子生成图像的过程图。

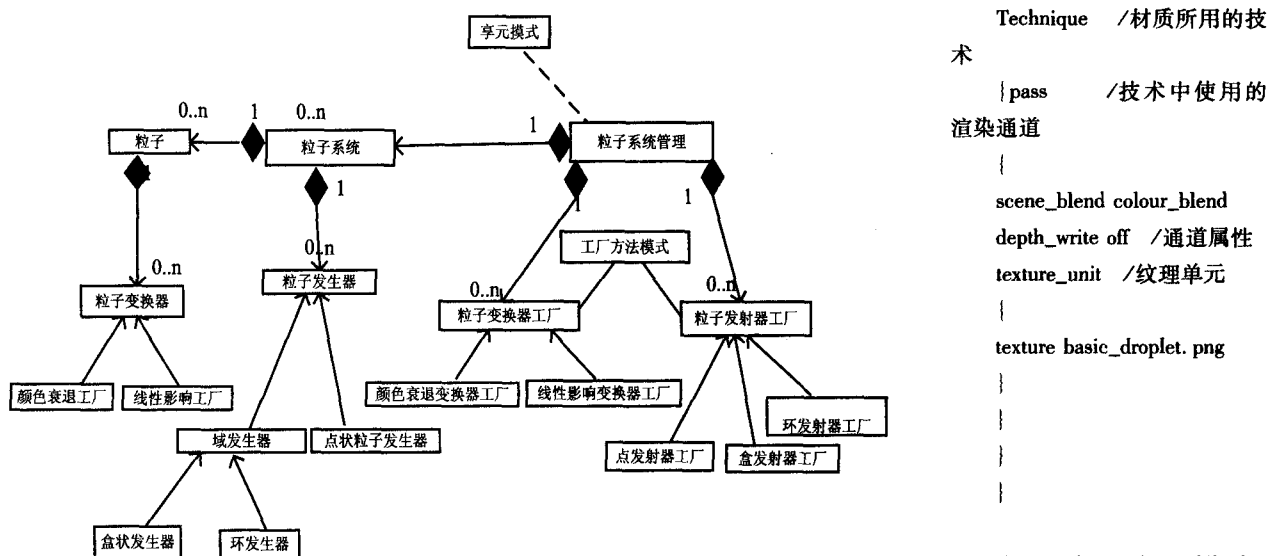


图 2 粒子系统结构图

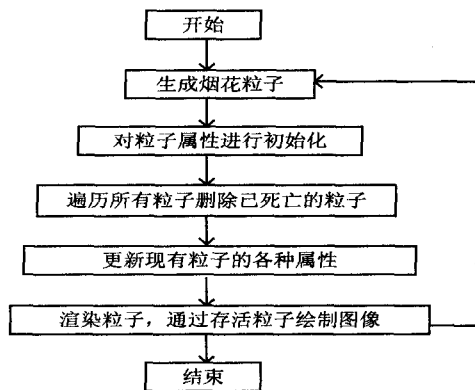


图 3 粒子生成图像的流程图

### 3 材质

OGRE 中材质主要用来说明渲染场景里面的对象是如何被描绘出的, 物体表面的属性是其决定的(如物体表面反射出的光的颜色、自发光颜色)。纹理不同表现出效果的不同。因材质定义了对光线反射的处理方法, 所以光源对材质影响很大。虽然 OGRE 中材质定义反射光线的方法, 但是事实并不会真的把光反射到周围的物体上, 而是利用局部辐射的算法去处理光照的效果<sup>[9]</sup>。材质不但能够通过编程来设置, 而且还可以像粒子系统一样通过脚本加载材质。渲染中使用的一个整体的材质一定要使用某种技术实现, 所使用的技术渲染实现中最低有一个渲染通路在内, 通路内至少包含一个纹理单元, 渲染需要的纹理则在纹理单元内被定义<sup>[10]</sup>。同样纹理单元里面中至少也要有一张 2D 纹理, 在 3D 物体表面采用纹理映射, 让虚拟物体表现得更加逼真。文中实现下烟花效果的材质脚本如下。

```
material Examples
```

```
{
```

```
Technique /材质所用的技术
```

```
{pass /技术中使用的
```

```
渲染通道
```

```
{
scene_blend colour_blend
depth_write off /通道属性
texture_unit /纹理单元
{
texture basic_droplet.png
}
}
```

### 4 粒子脚本主要模块

盒状粒子发射器。

```
emitter Box
```

```
{ angle 32 // 粒子发射时偏离 direction 的最大角度
emission_rate 32 // 发射速率(个/秒)
time_to_live 60 // 粒子生存时间(秒)
direction 11 // 粒子发射器的方向
velocity 14 // 速度的大小
colour-range-start 1 0 1 // 颜色值初始大小
colour-range-end 0.6 1 0.8 // 颜色渲染结束时的值大小
```

小

```
//定义粒子发射器 BOX 的大小
```

```
width 50
```

```
height 50
```

```
depth 50 }
```

```
// LinearForce:对运动中的粒子的加上一个外力,影响其运
```

动轨迹

```
affector LinearForce
```

```
{
force_vector 0 90 0 // 指定外力的影响(用向量表示)
force_application add // 粒子的运动向量加上外力的向量 }
```

```
// ColourFader:影响粒子中的颜色
```

```
// Fader affector ColourFader
```

```
//每秒衰减 0.28
```

```
red -0.28
```

```
green -0.28
```

```
blue -0.28 }
```

### 5 烟花粒子渲染效果与实现

实验程序创建三个粒子系统, 创建方法相同, 但模板不一样。粒子系统被创建好后, 可以将其绑定到场景节点上去, 便于运动控制。一旦粒子系统被绑定到场景节点上, 可以通过 FrameListener 的 FrameStarted 方法来控制节点运动, 实现粒子边运动边喷发了<sup>[11]</sup>。在该烟花粒子系统中, 主要由粒子发射器、效果器组

成。并通过纹理映射模拟烟花粒子的形状和颜色,非常逼真。效果图 4 是烟花在空中爆炸时渲染,帧刷新率达到了 163fps,画面渲染比较流畅,图 5 是渲染爆炸后粒子下落和消失时的渲染。两幅图基本对粒子系统周期进行了较好的渲染<sup>[12]</sup>。

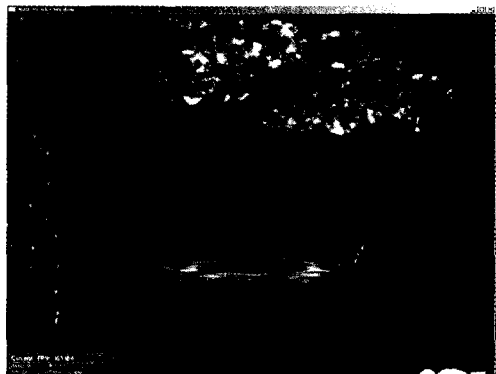


图 4 烟花爆炸效果(1)

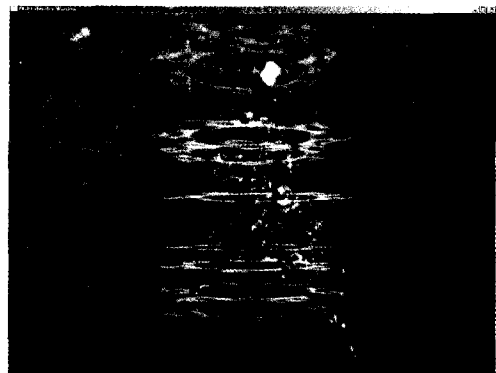


图 5 烟花爆炸效果(2)

## 6 结束语

文中利用 OGRE 图形引擎的粒子系统原理,对烟花进行了渲染,脚本和程序结合让渲染更加逼真。一方面充分对不规则模糊物体进行了很好的渲染,节省了成本,另一方面也具有较高性能的图形模拟表现,让

渲染画面清楚流畅。

由于烟花种类繁多,实现全部的烟花渲染是不可能的,这是文中不足之处。但是如果能够对粒子系统与场景管理进行很好的设置,将可以渲染出种类繁多的烟花渲染。

## 参考文献:

- [1] Admin. OGRE 图象引擎介绍[EB/OL]. 2010-06. <http://www.azure.com.cn/default.asp>.
- [2] 白建军,朱亚军,梁 辉,等. OpenGL 三维图形设计与制作[M]. 北京:清华大学出版社,1998:378-391.
- [3] 鲁 萌,刘建波. 三维地形显示中数据缓存与调度算法研究[J]. 微计算机信息,2010,4(1):210-212.
- [4] 杨子华,刘宏芳. 基于粒子系统模型的自然景物生成技术应用研究[J]. 计算技术与自动化,1998,17(3):20-23.
- [5] 王宏伟,刘 越,王涌天. 面向对象的通用粒子系统设计[J]. 系统仿真学报,2006,18(1):46-48.
- [6] Reeves W T, Blau R. Approximate and probabilistic algorithms for shading and rendering structured particle system[J]. Computer Graphics, 1985, 19(3):313-322.
- [7] 张尚华. 烟花燃放效果的仿真研究[D]. 广州:中山大学计算机与信息学院,2006.
- [8] Steven P, Reiss. An Engine for the 3D Visualization of Program Information[J]. Journal of Visual Languages & Computing, 1995, 6(3):127-129.
- [9] 张 璞,陶丽娜. 基于 OpenGL 的岩石楔形体边坡三维分析系统[J]. 计算机技术与发展,2009,19(3):53-56.
- [10] 张巧芳,李光耀. 基于单幅图像的三维浏览图生成算法[J]. 计算机技术与发展,2010,20(1):43-47.
- [11] Brundy A L, Saltzman D H, Emerson D, et al. Sonographic features associated with cleft palate[J]. Clin Ultrasound, 1986, 14: 486-489.
- [12] Monni G, Ibba RM, Olla G, et al. Colour Doppler Ultrasound and prenatal diagnosis of cleft palate[J]. Clin Ultrasound, 1995, 23:189-192.

(上接第 87 页)

- [7] 王少燕. 多媒体通信中的音视频同步问题研究与实现[D]. 西安:西安电子科技大学,2003.
- [8] 鲁 萍,马光四. 多媒体数据流实时传输速率研究及应用[D]. 西安:西安建筑科技大学,2005.
- [9] Moving Pictures Expert Group. MPEG-2 TextModel 5. DocISO/IEC JTC1/SC29/WG11/N0400[S]. 1993.
- [10] Fibush D. Timing and Synchronization Using MPEG-2 Transport Streams[J]. SMPTE Journal, 1996(7):395-405.
- [11] 求是科技. Vc++ 音视频编解码技术及实践[M]. 北京:人民邮电出版社,2006.
- [12] 陈 芳,沈晓军,陈 洁. 多媒体同步技术的研究[J]. 北京

工业大学学报,1996,22(4):110-114.

- [13] 程文青,陈云鹤,徐 晶. 一种适用于嵌入式媒体播放器的音视频同步方法[J]. 计算机与数字工程,2007,35(2):161-163.
- [14] 吴 进,贺 辉,洪 辉. 多媒体数据流实时传输技术的研究[J]. 通信技术,2009,42(1):342-344.
- [15] 刘 芳. 基于时间轴模型的音视频同步的研究与实现[D]. 南京:暨南大学,2008.
- [16] ZHANG Wei-ming, WU Ling-da, LAO Song-yang. Multimedia Information System[M]. Beijing:Publishing House of Electronic Industry, 2002.