

基于 Director 的交互式虚拟仿真系统

古成中, 吴新跃

(海军工程大学 船舶与动力学院, 湖北 武汉 430033)

摘 要: 针对虚拟仿真系统的开发, 设计了一套新方案. 该方案利用软件 Director MX 的多种插件及其集成语言 Lingo 实现虚拟仿真系统的交互式操作, 在满足一般性交互操作要求的同时避免使用 OpenGL 和 C 语言编程, 大大降低了系统开发的复杂度. 结合插件 Havok 研究了碰撞检测原理, 根据实际虚拟仿真系统精度要求演变了 BSP 树分割法.

关键词: 虚拟仿真; 交互; 碰撞检测; BSP 树

中图分类号: TP391.9

文献标识码: A

文章编号: 1006-754X(2007)04-0282-04

Interactive virtual reality system based on Director

GU Cheng-zhong, WU Xin-yue

(Academy of Shipping and Dynamics, University of Naval Engineering, Wuhan 430033, China)

Abstract: Aiming at the development of virtual reality system, a new scenario is presented. This scenario realizes interactive operation of virtual reality system by extra-programs of Director MX and Lingo. It supports usual interactive operation and simultaneously avoids OpenGL and C Programming, which greatly decreases developing complexity. Furthermore, research on collision inspection principle combining with the extra-program-Havok, is carried out. Finally, BSP portioning is evolved according to accuracy requirement of practical virtual reality system.

Key words: virtual reality; interactive; collision inspect; binary space portioning(BSP) tree

对于三维场景的虚拟仿真, 通常是建立在 DirectX 3D 和 OpenGL 等图形标准的基础上, 通过 DirectX 3D 和 OpenGL 等 API 函数库实现三维虚拟的交互操作. 然而, OpenGL 并不是一种语言, 它更像 C 语言运行时的一个函数库, 提供了一些预包装的功能^[1]. 因此, 要想运用好 OpenGL, 就必须在非常熟悉它本身编程技巧的同时掌握 C 语言等编程规范, 这对非专业人员来说是一件非常困难的事. 本文结合燃气轮机虚拟仿真系统的开发经验, 针对其对模型交互性操作要求不高的情况, 设计了一套系统开发方案, 巧妙地避开了 OpenGL, C 语言等程序编写.

1 交互场景的实现

1.1 开发平台

开发方案所涉及的软件平台有 3DS MAX,

Director MX 及其外挂程序, 在 3DS MAX 中建立三维场景, 并进行材质贴图渲染、机械仿真动画制作, 利用 Director MX 集成语言 Lingo 及插件 Shockwave 3D, 3DPI 和 Havok 实现三维场景的虚拟仿真, 可进行平移、旋转、缩放、壳体透明和播放仿真动画等操作.

3DS MAX 是一种优秀的三维动画造型软件, 与仿真软件相比, 它更强调造型的视觉效果, 其渲染的效果更能给观察者以临场感, 它也是目前第三方插件最多的 3D 造型工具之一^[2].

Director MX 是一套良好的界面开发工具, 其内部集成的 JavaScript, Lingo 语言具有良好的重复使用能力, 而且其自带的“行为程序库”也能实现强大的交互功能. 另外, Director MX 的外挂程序非常丰富, 基本能完成 C 语言所能实现的全部功能. 因此, 在界面编制方面, Director MX 已经成为 C 语言的替代品, 而且在实现相同功能的情况下, 它的编程

收稿日期: 2007-03-21.

基金项目: 军内科研项目.

作者简介: 古成中(1981-), 男, 安徽无为, 博士生, 从事机械自动化与仿真研究. E-mail: chaminggu@126.com.

量远小于 C 语言. 在三维场景方面, Director MX 主要应用于网络三维游戏的开发^[3], 具有良好的三维开发基础, 特别是外挂程序的引入更加完善了其三维制作能力. 作者别出心裁地将此种方法应用到机械结构的虚拟仿真, 既实现了较高的三维互动性, 又避免了复杂的程序编写.

1.2 场景建立

在 Director MX 中加入 Shockwave 3D 引擎. Shockwave 3D 的应用范围很广, 从简单的三维文字动画到交互式三维游戏环境, 都可以通过它被表现出来. Director 为 Shockwave 3D 加入了几百条控制 Lingo, 结合 Director 本身功能, 在交互能力上 Shockwave 3D 具有强大的优势. 但 Shockwave 3D 引擎只能浏览三维模型, 在场景建立方面必须要通过其他方式完成. Shockwave 3D 场景可以通过 Lingo 来创建, 但要想创建燃气轮机这样复杂的场景就必须借助 3DS MAX 等三维软件^[3].

首先利用 3DS MAX 完成燃气轮机各部件的三维造型, 为其添加恰当的材质纹理, 并制作相应的机械仿真动画, 如图 1 所示. 然后将燃气轮机模型转成 Shockwave 3D 格式文件, 导入 Director MX 中.

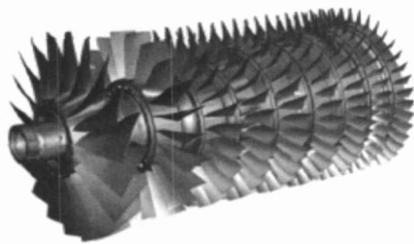


图 1 低压压气机
Fig. 1 Low pressure crusher

在 Director MX 中, Shockwave 3D 浏览器仅局限于模型的读入, 对 3D 模型的管理与编辑能力比较弱, 既没有编辑窗口, 又没有管理接口. 所以, 对于需要在 Director MX 中制作 3D 互动的用户来说, 管理 3D 模型时, 必须借助于外挂程序——3DPI.

由于篇幅的限制, 本文不详细阐述外挂程序 (Shockwave 3D, 3DPI, Havok 等) 的安装与设定. 进入 3DPI 之后, 就会看到十分详细的功能设定与格式的 3D 资源管理接口. 在 3DPI 这套外挂程序软件中, 可以任意设定 3D 对象的各种属性, 并且可以调整光源的设定, 纹理、阴影、运动、碰撞设定与摄影机的设定, 功能非常强大, 可以称之为 Shockwave 3D 的属性管理器 (property inspector). 甚至可以用 3DPI 来建立 3D 对象、光源、摄像机以及动画, 如图 2 所示.

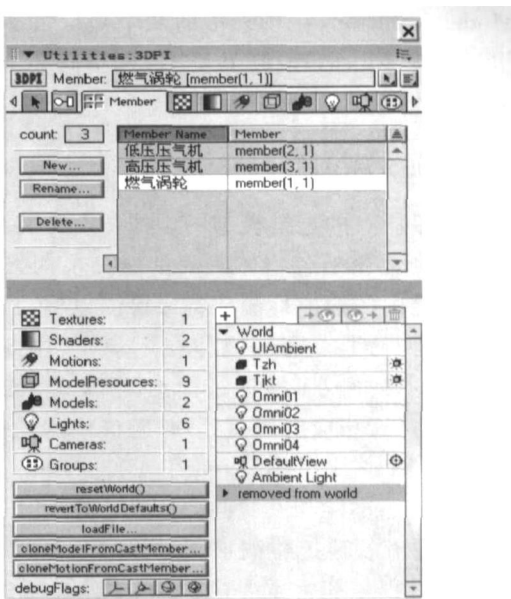


图 2 3DPI 管理界面
Fig. 2 Interface of 3DPI

1.3 交互功能的实现

场景的交互性是整个虚拟仿真系统的灵魂, 交互功能的强弱直接影响到虚拟仿真的最终效果. 本文所研究的燃气轮机虚拟仿真系统需要实现的三维空间的交互操作基本囊括了一般 PC 机上仿真的要求, 它可以利用鼠标自由拖动模型, 旋转模型, 实现模型的缩放、透明, 播放机械原理动画等. 要想实现这些功能, 利用 OpenGL 等语言编程的工作量是难以想象的. 利用 Lingo 也能很好地实现这些功能, 但同样有较大的程序量. Macromedia 公司为三维模型的交互操作提供了一套功能强大的工具包——3D Behaviors, 通过使用它的三维模型行为库可以在脱离编程的情况下创建和控制三维环境.

Director 包含了 2 种不同类型的三维行为库: Trigger (触发器) 和 Action (操作). 当虚拟仿真系统使用者的某一动作 (如单击鼠标左键) 发生时, Trigger 将向系统发出信号, 从而引起指定的 Action 的执行 (如透明所选部件).

在 Director 中, Action 又分成 3 种: Local Action (本地动作)、Public Action (全局动作) 和 Independent Action (独立动作). 当一个本地动作赋给一个燃气轮机部件后, 该动作只能被赋给同一部件的触发器所触发. 与本地动作不同的是: 公共动作能被任何触发器触发, 不管触发器是被赋给同一 Sprite 还是其他的 Sprite. 公共动作使用和本地动作一样的触发器, 而独立动作根本无需触发器触发即可执行. 例如: 可以使用独立动作中的 Automatic

Model Rotation, 让场景中的某个模型自动旋转; 利用 Model Rollover Cursor, 当鼠标指向某个模型时, 指针将改变形状。

2 碰撞检测的研究

几乎所有的三维模型的虚拟仿真都离不开碰撞检测——无论是各物体之间的碰撞检测, 还是物体与场景之间的碰撞检测。碰撞检测也称为干涉检测或者接触检测, 就是对物体在运动或装配过程中可能出现的碰撞或干涉进行检查, 避免碰撞或干涉的发生, 或进行相应的碰撞处理。碰撞检测是构造虚拟环境的一个重要组成部分。在真实世界中, 2 个机械零件很显然不能相互穿透, 所以在虚拟仿真系统制作时容易忽视碰撞检测这一过程的存在。然而, 碰撞检测的过程是非常重要的, 否则燃气轮机的转子就会在壳体内自由穿插, 零件之间也会相互穿身而过。DirectX 3D 和 OpenGL 中碰撞检测是在编程时实现的, 但本仿真系统是在 Director MX 中完成的, 并没有使用到 DirectX 3D 或 OpenGL 的开发平台, 这种情况下的碰撞检测应该如何实现呢? 本文采用 Director MX 的外挂程序——Havok, 很好地解决了三维物体的碰撞检测问题。

2.1 碰撞检测算法研究

碰撞检测的实现最重要的是研究选择碰撞检测算法的问题, 碰撞检测实际上就是由电脑计算判断多面体相交的问题, 场景中海量的姿态、位置各异的多面体使得上述计算是一项十分复杂和耗时的工作, 在简单的仿真中, 每一步都必须实现碰撞检测, 从而系统的运行代价随着场景的复杂性而急剧增加。目前常用的算法有包围球法(sphere)、沿坐标轴的包围盒 AABB(axis-aligned bounding boxes)、方向包围盒 OBB(oriented bounding box)、二叉空间分割 BSP(binary space partitioning) 树等。本系统所用模型是基于三角形的, 结合对上面几种算法原理的分析, 给出了一种适合本系统开发的算法, 这种算法是 BSP 树的一种改版^[4]。

首先, 根据物体的运动规律或用户的输入计算物体在即将渲染的一帧时的位置; 然后, 循环遍历场景中每一个三角形。在循环过程中, 找到当前三角形所在的平面 S , 沿平面法线方向对它作一个平移 d , 表示物体和平面之间所能接近的距离。判断物体在上一帧和当前帧的位置与平面 S 的关系: 如果上一帧在平面前, 而当前帧在平面后, 说明物体穿过了平面 S 。但因为平面是无边界的, 还需进一步判断物体是否在三角形之内穿过平面 S 。过三角形的三边, 做

垂直于三角形的平面 S_1, S_2, S_3 , 并且令它们的法线指向三角形的内部。为了体现物体和三角形所能接近的最短距离, 这 3 个平面也需要作一个平移 d , 只不过是沿着其法线负方向。依靠这 3 个平面就可以判断物体是否在三角形三边所界定的范围之内穿过平面 S 。但是, 如果三角形有很尖利的锐角, 就会使其界定的区域过大^[5]。所以需要另外 3 个与它们平行的平面 S_4, S_5, S_6 来削去产生的锐角, 如图 3 所示。通过判断物体现在的位置是否在这 6 个平面之内就可以判断物体与当前三角形是否发生碰撞。如果发生碰撞, 修正当前帧的位置, 使物体的运动沿着与平面 S 平行的方向, 否则, 继续计算下一个三角形。

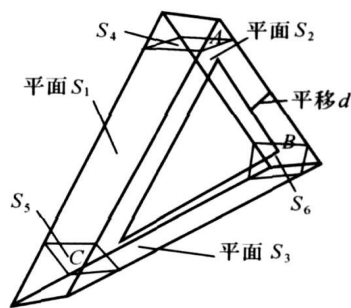


图 3 碰撞检测算法

Fig. 3 Arithmetic of collision inspect

总之, 该算法的原理就是: 给出一个三角形, 判断物体是否穿过了三角形所在的面, 再判断物体是否在三角型内部穿过。

2.2 碰撞检测的实现

碰撞检测的算法有很多, 算法只是一种理论方法, 最终的碰撞检测还需要通过程序来完成。针对本文开发环境, 选用 Director 的插件 Havok。该插件主要用于控制三维模型, 能够方便地对模型添加材质纹理、光照和阴影, 擅长于模型的碰撞检测; 其开发语言也可以为 Lingo 语言。下面以简单的实例说明利用 Havok 插件开发三维场景的基本方法以及碰撞检测的实现。

```
.....
havok = member("havok")
——初始化场景
.....
brickSh.ambient = rgb(200, 0, 0)
——添加纹理
.....
repeat with w = 1 to 12
    pengzhs[w] = []
    pengzhs[-h[w]] = []
```

```
repeat with l = 1 to 6
    pengzhs[w][1] = wrld.newModel("pengzh"&w&"_ "&l, pengzhRs)
    pengzhs[w][1].shaderList = pengzhSh
    if (w MOD 2) = 1 then
        .....
```

——碰撞检测

3 成果展示

根据本文所阐述的系统开发方案, 作者已成功地完成了军队科研项目“燃气轮机虚拟仿真训练教

学系统”的开发。利用 3DS MAX 软件建立燃气轮机模型, 并添加逼真的材质, 制作燃气轮机机械原理、结构的机械仿真模块; 利用 Director MX 编制了交互性能高、外形美观、使用方便的交互界面, Shock-wave 3D、3DPI 和 Havok 等外挂程序实现燃气轮机虚拟仿真。作者结合计算机辅助教学与虚拟仿真技术, 成功地开发出一套集图片文字、机械仿真动画、虚拟仿真等多维题材于一体的训练教学软件, 可以实现鼠标拖动、旋转、缩放、壳体透明、播放原理动画等三维空间的交互式操作。虚拟拆装界面如图 4 所示。

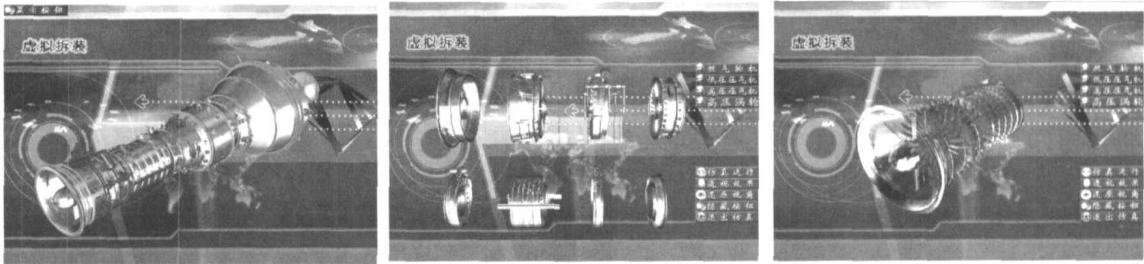


图 4 虚拟拆装界面
Fig. 4 Interface of dummy dismantle

4 小 结

本文结合燃气轮机虚拟仿真系统的开发经验, 针对其对模型交互性操作要求不高的情况, 设计了一套系统开发方案, 巧妙地避开了 OpenGL, C 语言等程序编写。成功地运用 Shockwave 3D、3DPI 和 Havok 等外挂程序实现燃气轮机虚拟仿真系统, 避免 OpenGL、CG 等复杂程序的编写。结合插件 Havok 研究了碰撞检测原理, 根据实际虚拟仿真系统精度要求演变了 BSP 树分割法。

由于计算机及其相关技术日新月异, 本文的研究也存在不足: 由于虚拟仿真模块脱离了 OpenGL 编程, 在降低制作难度的同时也降低了虚拟仿真操作的交互程度。对此可利用 Lingo 语言几乎可以调用其他所有语言程序的功能, 结合 OpenGL、CG 等专业函数库来弥补。先利用 OpenGL 编程, 读取模型的空间和材质数据, 实现在 3D 空间的互动; 再通过 CG 语言对模型进行着色、光照等渲染, 让模型具有完美的视觉效果; 最后通过 Director MX 的 Lingo 语言调用 OpenGL、CG 程序, 将高互动的虚拟仿真与其他模块整合。

参考文献:

[1] 徐 波. OpenGL 编程指南[M]. 北京: 机械工业出版社, 2006: 1-5.
XU Bo. OpenGL programming guide[M]. Beijing: China Machine Press, 2006: 1-5.

[2] 韩国安. 3ds max 7 渲染的艺术[M]. 北京: 中国青年出版社, 2005: 3.
HAN Guo-an. Rendering art of 3ds max 7[M]. Beijing: China Youth Press, 2006: 3.

[3] 颜志翰, 李雅玫. 精通 Director MX 2004 多媒体开发[M]. 北京: 清华大学出版社, 2005: 2-6.
YAN Zhi-han, LI Ya-wen. Master multi-media explorer of Director MX 2004[M]. Beijing: Qinghua University Press, 2005: 2-6.

[4] SAMUEL Ranta-Eskola. Binary space partitioning trees and polygon removal in real time 3D rendering [M/OL]. [2007-1-15]. <http://blog.csdn.net/FishCrazy/archive/2004/11/12/179243.aspx>.

[5] 孙家广. 计算机图形学[M]. 北京: 清华大学出版社, 1999.
SUN Jia-guang. Computer graphics[M]. Beijing: Qinghua University Press, 1999.