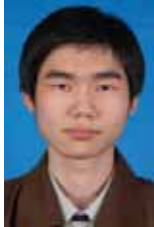


基于关节的虚拟场景建模及仿真方法

刘宇翔^{1,3}, 张珩², 鲁士文¹

(1.中国科学院计算技术研究所, 北京 100080; 2.中国科学院力学研究所, 北京 100080; 3.中国科学院研究生院, 北京 100039)



摘要: 针对虚拟现实技术中场景生成及仿真问题提出了一种基于关节的虚拟场景建模及仿真方法。该方法将场景中实体的位置状态及其基元间的连接关系用统一的关节描述, 通过建立场景连接图表述场景结构, 在场景更新过程中采用关节驱动基元的方式。在此基础上, 我们开发了通用的虚拟场景可视化软件平台, 测试证明提出的方法可以有效地提高场景建模及仿真处理的效率。

关键词: 虚拟现实; 场景建模; 关节; OpenGL

中图分类号: TP391.72 **文献标识码:** A **文章编号:** 1004-731X(2007)06-1316-04

Method for 3D Modeling and Simulation Based on Joint

LIU Yu-xiang^{1,3}, ZHANG Heng², LU Shi-wen¹

(1. Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080, China;

2. Institute of Mechanics, Chinese Academy of Sciences, Beijing 100080, China; 3. Graduate School of the Chinese Academy of Sciences, Beijing 100039, China)

Abstract: A method based on joint for 3D modeling and simulation was introduced. Both the position of entities and all kinds of connection between elements were described using uniform joints. Model constructs scene joint graph to reflect the structure of scene and carries out scene updating by adopting the scheme that drives entities by joints. Furthermore, a universal visualization platform was developed and tested in real system. The result shows that the proposed method has better efficiency in both scene modeling and simulation process.

Key words: virtual reality; scene modeling; joint; OpenGL

引言

虚拟现实(Virtual Reality), 是一种综合计算机图形处理、多传感交互、仿真等多种技术, 借助于多维输入输出设备生成逼真的三维虚拟环境的人机交互技术。它的核心内容是虚拟场景生成, 即获取实际环境中的三维数据, 并根据应用需要建立相应的虚拟环境模型, 其目的是将真实世界的对象在三维虚拟世界中重构。

目前, 大多数的虚拟场景建模技术通过层次结构组织和管理场景中的对象及其行为(如 VRML 和 Java3D 的场景结构图)。这种层次式结构化的建模方法能够较全面地描述场景对象的结构, 但对于较为复杂的场景往往导致模型描述开销巨大, 场景生成和实时仿真效率不高。针对基于结构的场景建模技术的不足, 本文提出了一种基于关节的场景建模技术及仿真方法。它以场景对象及其基元的连接关系为基础, 通过定义不同的连接方式反映场景元素在结构上的层次关系。该方法简化了场景描述规范, 统一场景元素处理方式, 有效地提高了虚拟场景建模的效率。

1 面向连接的虚拟场景建模

在基于连接方式的场景建模中, 用抽象的初始基元表示整个场景, 其坐标系与世界坐标系始终保持一致。将场景中

的实体拆分成具有独立几何属性的基元。场景中实体的位置状态及其基元间的连接关系用统一的关节模型描述。关节具有 6 个自由度 $(x, y, z, \theta_x, \theta_y, \theta_z)$ 。其中 x, y, z 是关节的位置坐标, $\theta_x, \theta_y, \theta_z$ 表示关节局部坐标系相对其依附体局部坐标系的转角。具体方法如下:

- 独立实体的位置及姿态。独立实体是可以自由移动的实体对象。在组成实体的基元中选取一个主体基元。主体基元连接于一个以初始基元为依附体的关节之上。该关节位置坐标及坐标轴旋转分量的实际意义是主体基元在世界坐标系中的位置和转角, 即实体的位置及姿态属性(以下简称“位姿”属性)。这种关节的 6 个自由度可以任意改变。

- 实体间的相对位置关系。场景中某些实体在空间位置上存在一定的约束(如太阳与地球)。建模时, 选取一核心实体, 同时非核心实体的关节依附于核心实体, 该关节的 6 个自由度变化必须满足约束条件。

- 基元之间的连接关系。常见的连接方式包括固连、滑动连接、转动连接以及它们的组合方式。我们同样可以采用设置约束条件的方式表示不同类型的连接。如固连关节参数不允许改变, 转动关节只有转角参数可変等等。

为了完整的描述场景, 还需定义光源节点和相机节点。

光源节点依据其类型可分为环境光源、平行光源、点光源和锥光源。

相机节点其功能在于在三维场景变化时实时记录相机节点的局部坐标系原点在世界坐标系中的坐标以及坐标轴的相对转角, 并将其提供给相机视窗绘制模块。

收稿日期: 2005-08-22

修回日期: 2006-08-18

作者简介: 刘宇翔(1980-), 男, 河北省唐山人, 硕士生, 研究方向为计算机网络技术; 张珩(1961-), 男, 研究员, 博导, 研究方向为遥科学及遥操作技术、鲁棒控制与智能系统; 鲁士文(1944-), 男, 教授, 博导, 研究方向为计算机网络协议。

经过上述抽象, 场景中存在着五类场景元素——初始基元、实体基元、光源节点、相机节点和关节。他们之间通过连接/依附关系联系在一起。为了提高处理效率, 建议遵循以下规则: 关节依附于非关节元素; 除初始基元外, 非关节元素必须连接于关节。至此任何场景元素的位姿属性可以完全由其连接的关节状态直接确定或递推求出。

2 虚拟场景生成模型

2.1 场景元素的统一化处理

在确定了场景元素及其连接关系的基础上, 我们构造树状的场景连接图。图中的节点对应场景元素, 边表示父节点和子节点间的连接/依附关系(见图 1)。树的根节点为初始基元。

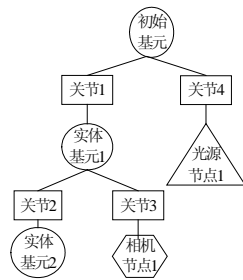


图 1 场景连接图

在场景生成阶段, 从初始基元开始, 依据连接关系深度优先遍历场景连接图中的元素。由于五类场景元素在结构上是无层次的平级关系, 对于各类元素可以采用统一的处理方式。处理节点时采用先根遍历的方式, 先绘制本节点, 再依次处理其子节点。其步骤如下:

- 1) 保存当前局部坐标系现场: 目前绝大多数图形库都直接提供此类接口, 在 OpenGL 实现中通过压入模型转换矩阵指令完成该功能;
- 2) 绘制节点;
- 3) 依次处理子节点;
- 4) 恢复局部坐标系现场: 在 OpenGL 实现中通过弹出模型转换矩阵指令实现。

2.2 场景元素绘制

初始基元的绘制过程就是设置场景全局环境变量(场景背景颜色、全局视点等)的过程。

实体基元绘制时先设置基元属性参量(颜色、材质、纹理等), 并依据基元形状绘制基元。

光源节点绘制时需要设置并开启场景光源。

绘制相机节点只需更新局部坐标系相对世界坐标系的位置及坐标轴旋转角度, OpenGL 实现中可以直接获得当前的模型转换矩阵, 从中可以推出相机节点当前的位置和视角。

关节的绘制是通过调用图形接口的平移和旋转函数将场景绘制的当前坐标系变换成关节局部坐标系的过程。

关节在虚拟场景模型中描述中表示基元间的位置和连接关系, 而其在计算机图形学中的实质是从关节依附体的局部坐标系到连接该关节的基元的局部坐标系的坐标系变换。因此在场景连接图中关节又称为变换节点(Transform Node); 相对于关节, 实体基元可以称为形体节点(Shape Node)。

2.3 相机视窗的生成与运算

用户可以在系统中设置相机, 其对应的相机视窗可以实

时生成相机节点观察到的场景。

绘制相机场景前先进行视图变换, 将视点置于相机节点在世界坐标系中的位置, 剩余步骤与绘制全局场景大致相同。

如何完成全局场景到相机场景的变换是绘制相机视窗的主要问题。利用现有的图形库提供的接口有两种方法实现该变换。

方法一: 在绘制全局场景时记录相机节点当前的模型转换矩阵, 从中推导出相机节点局部坐标系相对于世界坐标系的位置和转角。绘制相机视窗时直接设置视图变换矩阵, 将视点设置在世界坐标系的相应位置。

由计算机图形学原理知

$$\text{坐标原点平移的转换矩阵 } T \text{ 为: } \begin{bmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

依次绕 X,Y,Z 轴旋转 α, β, γ 角的转换矩阵 $R = R_x \times R_y \times R_z$ 为:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \beta & 0 & \sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \gamma & \sin \gamma & 0 & 0 \\ \sin \gamma & \cos \gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

最终的模型变换矩阵为 $T \times R$:

$$\begin{bmatrix} \cos \beta \cos \gamma & -\cos \beta \sin \gamma \\ \sin \alpha \sin \beta \cos \gamma + \cos \alpha \sin \gamma & -\sin \alpha \sin \beta \sin \gamma + \cos \alpha \cos \gamma \\ -\cos \alpha \sin \beta \cos \gamma + \sin \alpha \sin \gamma & \cos \alpha \sin \beta \sin \gamma + \sin \alpha \cos \gamma \\ 0 & 0 \\ \sin \beta & T'_x \\ -\sin \alpha \cos \beta & T'_y \\ \cos \alpha \cos \beta & T'_z \\ 0 & 1 \end{bmatrix}$$

(其中向量 (T'_x, T'_y, T'_z) 为向量 (T_x, T_y, T_z) 依次绕 X,Y,Z 轴旋转 α, β, γ 角所得)。

$$\text{所以若设相机节点的模型转换矩阵为: } \begin{bmatrix} m_{11} & m_{12} & m_{13} & v_x \\ m_{21} & m_{22} & m_{23} & v_y \\ m_{31} & m_{32} & m_{33} & v_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

则: $\beta = \arcsin(m_{13})$

$$\text{若 } \cos \beta \neq 0, \alpha = \arctan\left(-\frac{m_{23}}{m_{33}}\right), \gamma = \arctan\left(-\frac{m_{12}}{m_{11}}\right)$$

$$\text{若 } \cos \beta = 0, \alpha = 0, \gamma = \arctan\left(-\frac{m_{21}}{m_{22}}\right)$$

而相机节点位置可以将向量 (v_x, v_y, v_z) 依次绕 Z,Y,X 轴旋转 $-\gamma, -\beta, -\alpha$ 角得到。

方法二: 利用视图变换和模型变换的对偶性, 将相机节点记录的模型变换矩阵求逆, 并和当前场景的模型变换矩阵相乘, 这就相当于先对相机视窗的场景作逆变换, 将当前坐标系的原点平移至相机节点在世界坐标系的位置, 此时将视点设置于当前坐标系的原点, 而在后续的绘制中场景元素的坐标仍然按场景元素在世界坐标系中的数值进行处理, 生成

的场景视图就相当于将视点置于世界坐标系中相机节点所处位置观察到的场景视图。

3 虚拟场景的实时交互与更新

在实时仿真处理中，需要接收用户的指令并更新场景。场景更新时系统采用关节驱动基元的方式。其实现方法是用户直接修改对应关节的状态参量，通过关节的连带效应（见下文），连接于该关节的基元及相关后继场景元素的位姿属性也随之改变。

如图 1 所示，当关节 J1 的状态改变时，其局部坐标系 O_1 相对于世界坐标系的位姿发生改变，而连接于该关节的场景元素（关节连接体）E1 局部坐标系和 O_1 保持一致，所以 E1 的局部坐标系相对于世界坐标系的位姿状态与 O_1 发生同样的变化。而关节的状态参量是表示关节局部坐标系相对于其依附体局部坐标系的相对位姿，因此依附于 E1 的关节 J2 的局部坐标系 O_2 相对于世界坐标系的位姿也会发生相关变化。依次类推，关节连带的所有场景元素在世界坐标系中的位置属性都会发生改变。由于在场景绘制时先根遍历场景连接图，在处理当前节点时使用堆栈保存当前模型变换矩阵，处理完所有子节点后向祖先节点回溯时弹出栈顶矩阵恢复访问该节点前场景的状态，从而可以很方便的实现关节的连带效应。

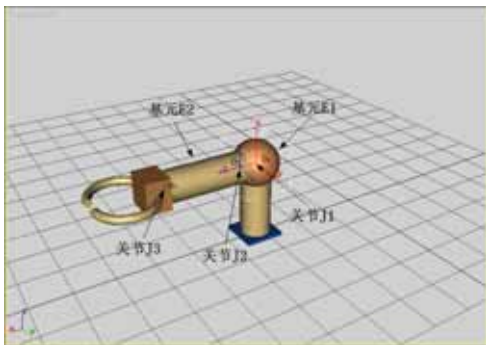


图 1 关节的连带效应

修改关节状态时根据关节的类型对状态参量的修改添加相应的约束条件，不满足约束条件的修改操作被取消。为关节定义适当的约束条件是规范场景元素行为、保证场景中实体的结构及其他关联的有效手段。

4 虚拟场景可视化软件的实现

在关节驱动的虚拟场景动态建模和实时仿真方法的基础上，我们开发了一套通用的虚拟场景可视化平台 U3DMS (Universal 3D Modeling & Simulation)。该平台支持用户自定义场景元素及其连接关系，并提供相应的场景交互接口方便用户实时控制场景的变化。

4.1 系统体系结构

U3DMS 采用层次式的设计思想，将系统划分为用户接口层、可视化控制层、场景绘制层、图形对象层和系统接口

层，如图 2。

用户接口层为用户提供了对虚拟场景进行动态建模及实时交互的接口；

可视化控制层负责初始化、场景连接图的生成及维护、场景的实时更新控制；

场景绘制层进行虚拟场景的生成及渲染，场景环境参量的设置，场景元素的绘制，包括场景模型矩阵堆栈的压入和弹出、关节坐标系的平移旋转变换、场景元素的属性设置、调用图形对象层的函数绘制实体基元；

图形对象层的功能是基本几何图形（长方体、球、柱体、锥体、台体等）的绘制；

系统接口层包括操作系统为程序员提供的 API（设备描述表、系统 I/O 接口、系统定时器等）、OpenGL 标准图形库以及常用的公共模块（如矢量和矩阵运算、数据类型转化、文件读写模块）



图 2 系统体系结构

4.2 系统运行流程

系统的运行分为场景建模和实时仿真两个阶段。

在场景建模阶段用户通过编辑场景数据库文件，对虚拟场景的结构、连接关系及行为建模。每一个场景元素对应场景数据库文件中一个数据项，数据项包含以下字段：

- 场景元素标识符 (ID)；
- 元素类型标识符 (Type)。Type 字段可以标识元素所属的种类（初始基元、关节、光源、相机和实体基元）；
- 实体基元具有形状、颜色、材质、纹理属性参量字段以及相应的扩展几何属性参量（如长方体型基元具有长、宽、高）字段；
- 实体基元、光源节点和相机节点具有记录其所连接关节的字段；
- 关节具有关节状态参量字段和记录关节依附体字段。

系统进行实时仿真时先导入场景数据库文件，生成树状的场景连接图。场景连接图的节点以场景元素对象的形式存储，不同类型的场景元素由元素类派生的不同子类实现。

初始化三维显示模型包括设置显示设备的像素格式、生成显示列表、开启深度缓存、启动刷新定时器等。

对于场景的更新用户可以通过用户接口层提供的接口直接通过关节 ID 修改场景连接图中相应关节的状态变量，也可以通过辅助的输入设备如空间留自由度鼠标控制相应关节的位置及转角。

绘制场景时依深度优先方式从根节点开始遍历绘制场景连接图中的场景元素。由于相机场景的生成依赖于相机节点记录的视点和视角，而这些信息通常在绘制全局场景时记录，因此刷新场景时先绘制全局场景再绘制相机场景。

4.3 性能实测

我们构造了一个空间机器人抓取飞行物的虚拟场景，其显示效果如图 3 所示。经过测试，软件可以稳定运行在

Pentium 4/1G RAM 的 PC 机平台上, 其对于存在数百个场景元素的虚拟场景进行实时仿真时可获得 30fps 以上的场景刷新率, 大大超过了人眼对流畅显示虚拟场景的要求。

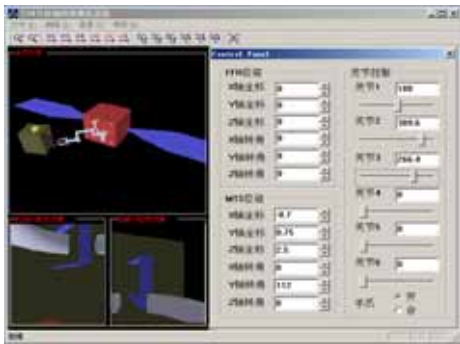


图 3 模拟空间机器人抓取飞行物

5 结论

本文提出的关节驱动的虚拟场景建模及仿真方法从场景对象及其组成基元的连接关系出发, 利用不同类型的连接方式反映场景元素在结构上的层次关系, 将场景中实体的位置状态及其基元间的连接关系用统一的关节模型描述。我们据此开发了通用的三维虚拟场景可视化软件。实践证明关节驱动的虚拟场景建模及实时性仿真方法可以简化模型描述

(上接第 1315 页)

不同, 距离对焦平面越远的物体清晰度越差, 且在对焦平面前后一定的范围内, 场景中的景物均可近似认为是清晰成像, 但在该景深范围外的物体则非常模糊。

5 结论

可编程图形处理器的应用大大提高了仿真算法的执行效率, 但是在顶点程序中保存深度信息时是对所有视锥体内的顶点进行处理, 所以程序执行速度快慢会受场景中顶点数量的多少影响。此外该透镜成像系统数学模型是对物理摄像机进行了一系列简化处理后得到的, 它忽略了很多比如镜头的畸变、衍射效应等等, 所以无法对真实摄像机的所有特性进行准确的仿真。

在今后的研究过程当中, 应该对所提出的数学模型进行进一步的完善与补充, 使其能更真实的仿真物理摄像机的成像效果和成像过程。充分利用可编程图形处理器的强大运算功能, 提高算法的执行效率, 使其能够应用于更复杂的场景当中。

开销, 提高实时仿真处理的效率, 降低高质量虚拟场景生成所需的计算资源。

参考文献:

- [1] Earnshaw R A, Gigante M A, Jones H. Virtual Reality System [M]. Carolina: Academic Press, 1993.
- [2] Wang Chengwei, Gao Wen, Wang Xingren. Theoretic Implementation and Application of Virtual Reality [M]. Beijing: Tsinghua University Press, 1996. (汪成为, 高文, 王行仁. 虚拟现实技术的理论实现及应用[M]. 北京: 清华大学出版社, 1996.)
- [3] Zeng Jianchao, Yu Zhihe. Virtual Reality Technology and Application [M]. Beijing: Tsinghua University Press, 1996. (曾建超, 俞志和. 虚拟现实的技术及其应用[M]. 北京: 清华大学出版社, 1996.)
- [4] Ji Qingge, Pan Zhigeng, Li Xiangchen. Overview of the Application of Virtual reality in Sports Simulation [J]. Journal of Computer-Aided Design & Computer Graphics, 2003, 15(11): 1333-1338. (纪庆革, 潘志庚, 李祥晨. 虚拟现实在体育仿真中的应用综述[J]. 计算机辅助设计与图形学学报, 2003, 15(11): 1333-1338.)
- [5] Hearn Donald, Baker M Pauline. Computer Graphics [M]. Beijing: Publish House of Electronics Industry, 1998. (Donald Hearn, M. Pauline Baker. 计算机图形学[M]. 北京: 电子工业出版社, 1998.)
- [6] Carey R, Bell G, Marrin C. ISO/IEC 14772-1:1997, The Virtual Reality Modeling Language- International Standard [S]. The VRML Consortium Incorporated, 1997.

参考文献:

- [1] 彭文彪, 戴树岭, 彭俊毅. 摄像机的计算仿真研究[J]. 系统仿真学报, 2004, 16(9): 2109-2111. (Peng Wen-Biao, Dai Shu-Ling, Peng Jun-Yi. Vidicon Computer Simulation [J]. Beijing: Journal of System Simulation, 2004, 16(9): 2109-2111.)
- [2] Fernando Randima, Kilgard Mark J. Cg 教程——可编程实时图形权威指南[M]. 北京: 人民邮电出版社, 2004.
- [3] Michal Potmesil, Indranil Chakravarty. A Lens and Aperture Camera Model for Synthetic Image Generation [C]//In: Computer Graphics Proceedings, Annual Conference Series, ACM SIGGRAPH, Dallas, 1981, 297-305.
- [4] Riguier Guennadi, Tatarchuk Natalya, Isidoro John. Real-Time Depth of Field Simulation [M]. ShaderX2—Shader Programming Tips and Tricks with DirectX9. Texas: Wordware Publishing, 2004.
- [5] Rokita Przemyslaw. Generating Depth-of-Field Effects in Virtual Reality Applications [J]. IEEE Computer Graphics and Applications (S0272-1716), 1996, 16(2): 18-21.
- [6] Kolb Craig, Mitchel Donl, Hanrahan Pat. A Realistic Camera Mmodel for Ccomputer Ggraphic [C]//In: Computer Graphics Proceedings, Annual Conference Series, ACM SIGGRAPH, Los Angeles, 1995, 317-324.
- [7] Jurriaan D, Robert van Liere. Fast Perception-Based Depth of Field Rendering [Z]. Association for Computing Machinery, 2000, 129-133.
- [8] 贺修桂, 肖绪珊. 摄影手册[M]. 北京: 中国摄影出版社, 1983.