

面向电子商城的虚拟现实引擎关键技术研究

鲁亚飞^{1,2}, 王正方¹

(1. 中国科学院沈阳自动化研究所 工业信息学实验室, 辽宁 沈阳 110016;
2. 中国科学院研究生院, 北京 100049)

摘要: 为了缓解目前电子商城在形象化和交互性方面的不足, 以及满足消费者的良好购物体验与购物需求, 设计并实现了虚拟现实引擎作为三维动态交互立体商城开发平台的核心构件。着重解析了引擎实现过程中的3ds模型导入、基于复合层次树的场景管理、AABB (axis-aligned bounding box) 和 Capsule (胶囊体) 相结合的碰撞检测和实时渲染等关键技术。实验结果表明, 这些技术在虚拟现实引擎中具有一定的通用性, 显著地提高了引擎的执行效率和实用性。

关键词: 虚拟现实引擎; 模型导入; 场景管理; 碰撞检测; 实时渲染

中图分类号: TP311.5 **文献标识码:** A **文章编号:** 1000-7024 (2013) 01-0318-05

Key technologies research on virtual reality engine for E-commerce

LU Ya-fei^{1,2}, WANG Zheng-fang¹

(1. Shenyang Institute of Automation, Chinese Academy of Sciences, Key Laboratory of Industrial Informatics, Shenyang 110016, China; 2. Graduate University, Chinese Academy of Sciences, Beijing 100049, China)

Abstract: To alleviate the shortcomings of current E-Commerce in visualization and interactivity, and for the purpose of meeting the good shopping experiences and shopping needs for consumers, the virtual reality engine as a core component of the three-dimensional dynamic interaction mall is designed and implemented. Several key techniques in the development of the virtual reality engine, including importing 3ds models, scene management based on complex hierarchical tree, collision detection of the combination of AABB (Axis-aligned bounding box) and Capsule, real-time rendering and so on, are discussed in depth. The results show that these techniques can be widely used in virtual reality engines, and can enhance their execution efficiency and practicality significantly.

Key words: virtual reality engine; importing models; scene management; collision detection; real-time rendering

0 引言

电子商城作为一种网上商品交易平台为广大消费者带来了便利, 网上购物已经被很多消费者所接受, 尤其是新崛起的年轻一代消费群体, 甚至成为其主要的消费方式。然而其单一、有限的展示方式却难以向消费者传递丰富的商品信息, 且与现实购物相比, 现有的电子商城, 不能满足消费者较高的购物体验与要求。为了使企业的品牌形象或产品形象获得更生动的展现, 进而增强消费者的购物体验、提高其对商品的认知程度, 三维动态交互立体商城便应运而生。尽管目前国内外已经有几家3D网上商城, 但由于其功能不够完善, 执行效率较低, 用户较少, 所以难以推广。而虚拟现实引擎作为3D网上商城开发的基础, 是决

定3D网上商城性能的重要因素, 为了缓解当前3D网上商城面临的问题, 并使其变得更具吸引力、影响力和生命力, 设计实现了面向电子商城的虚拟现实引擎, 以此作为三维动态交互立体商城的核心部件。文章结合虚拟现实引擎的开发过程, 深入探讨了三维动态交互立体商城开发平台中所应用到的几项关键技术——3ds模型导入、基于复合层次树的场景管理、AABB和Capsule包围盒相结合的碰撞检测、实时渲染等的实现方法。

1 虚拟现实引擎的设计

1.1 虚拟现实引擎的结构和开发环境

根据引擎开发过程中应用程序对相应数据信息的处理过程, 把整个引擎分为两部分: 接口部分和核心部分, 如

收稿日期: 2012-02-15; 修订日期: 2012-04-16

基金项目: 国家自然科学基金项目 (61174164、61003208、61105067)

作者简介: 鲁亚飞 (1984-), 女, 河南濮阳人, 硕士, 研究方向为集成技术和软件产品化; 王正方 (1968-), 男, 吉林白城人, 研究员, 研究方向为集成技术和软件产品化。E-mail: luyafei@sia.cn

图 1 所示, 这两部分是上下层的关系, 即通过接口部分输入的数据信息, 需要通过核心部分的处理, 才能实现三维动态交互式立体商城应有的效果和功能。其中 LWJGL (lightweight java game library) 为开发提供了简单易用的 API, 这些 API 用来访问 OpenGL (open graphics library), 实现场景、物体模型的实时重绘、渲染以及人机交互的实时控制。

虚拟现实引擎以开源软件 Eclipse 为开发平台, 以 LWJGL 为底层 API, 根据三维动态交互式立体商城应具有的功能, 运用 Java 相关知识构建开发应用程序。

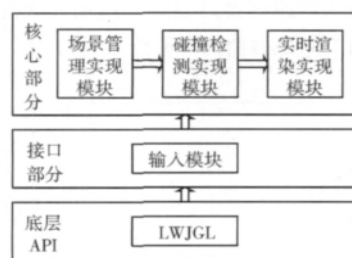


图 1 系统架构

1.2 虚拟现实引擎的功能设计

为了保证实用性及执行效率, 将虚拟现实引擎主要划分为 4 个模块: 输入模块、场景管理实现模块、碰撞检测实现模块和实时渲染实现模块。各模块的详细功能如下:

(1) 输入模块: 把现有的三维动态交互立体商城的 3D 模型 (用三维动画渲染和制作软件创建的三维模型) 及相关的资源信息加载到程序中, 且对它们进行统一管理以便在创建立体商城的过程中使用; 同时该模块也接收通过鼠标、键盘输入的信息。

(2) 场景管理实现模块: 基于 BSP (binary space partitioning) 树^[1]和场景图 (scene graph) 两种场景管理方法, 提出设计了复合层次树法来构建场景树, 以此实现对商城场景中所有对象的统一组织和管理。

(3) 碰撞检测实现模块: 通过计算要检测物体对象外面添加的包围盒的相交情况, 判断两个物体对象之间是否发生碰撞。该模块的实现可以使显示到屏幕上的场景更加真实、自然。

(4) 实时渲染实现模块: 利用视锥 (view frustum)、渲染队列和 Z-Order, 有效的实现商城中场景及场景中各个物体的实时渲染, 以使商城场景及其中的物体达到逼真的可视化效果。

2 3ds 模型导入

LWJGL 提供了访问 OpenGL 的 API, 在绘制立体商城中的模型和场景时使用。尽管 OpenGL 是高性能图形和交互性场景处理的行业标准^[2], 但它没有提供建立三维模型

的高级命令, 而是通过点、线及多边形等基本的几何图元来建立三维模型的^[3]。因此, 要利用 LWJGL 的 API 直接进行复杂物体的建模是不太现实的。而专业建模软件如 3DSMAX, 可以较方便的建立三维模型, 但是难以对建立的模型进行控制。所以, 综合 OpenGL 和用于建立 3D 模型的软件的优势, 首先采用常用的 3D 建模软件 3DSMAX 建立立体商城的模型, 然后在 Eclipse 开发环境下读取立体商城模型的数据信息, 并对其进行相应的控制操作, 最后在 LWJGL 的基础上进行绘制、渲染。该设计不仅可以借助 3DSMAX 建模软件的优势, 降低建立立体商城模型的时间, 减少代码量; 而且以 LWJGL 作为底层 API 实现对立体商城模型的实时重绘和人机的交互控制, 可以降低程序的复杂度, 提高虚拟现实引擎的开发速度。

2.1 3ds 模型文件结构

3ds 模型文件的基本组成单位是块, 且块层次结构非常复杂。其中, 块包括块头和主体内容两部分, 主体内容是块的数据信息, 块头又分为块 ID 和长度, 块 ID 表明该块中包含了哪类信息, 如 ID 为 0x4110 表示该块包含了对象中的顶点信息, ID 为 0x4120 表示该块包含了对象中的三角面信息。在整个立体商城模型中, 由于块中的信息不包含对象模型在整个立体商城模型中的位置, 因此, 在构建立体商城模型时, 用数字来表示每个对象模型在其中的位置顺序。

2.2 导入 3ds 模型的方法

由于立体商城的 3ds 模型结构复杂、数据量大, 所以为了降低开发过程中应用程序的复杂度, 且保证读取数据的准确率, 该虚拟现实引擎对模型文件实行了两次转化处理, 完成了 3ds 模型的导入, 导入过程如图 2 所示。

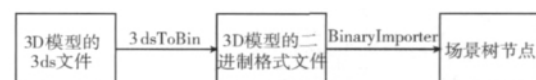


图 2 3ds 模型的导入过程

(1) 3ds 模型文件转化为二进制文件。

根据 3ds 模型文件的结构特点, 以 3dsToBin 类为主导, 通过 T3dsFile 类以及其它辅助类中的数据处理方法读取模型的每个块, 并根据其块头 ID 判断块中所包含的是哪类信息, 然后以二进制缓存数组为中介把 3ds 模型文件中的数据信息储存到二进制文件中, 相关的数据信息之间通过索引数组相关联。

(2) 二进制文件转化为场景树的几何节点。

几何节点所包含的对象是立体商城中的实体对象, 其中包含该对象的顶点、法向量、纹理、颜色信息及标识其在场景中位置顺序的数字, 这些信息均存储在缓存中, 且每个顶点通过数组列表与其对应的纹理相关联。以二进制文件为参数把模型数据加载到引擎中, 并根据几何节点对象相关信息的存储方式进行存储, 之后将根据所有几何节

点对象信息构建场景树。

3 基于复合层次树的场景管理

目前三维游戏中常用的场景管理方法有 BSP 树、八叉树 (Oct-Tree)、场景图等。其中 BSP 树法主要应用于室内场景的管理, 八叉树法主要适用于室外场景中静态物体的管理, 而场景图则是能有效管理动态室外场景中的物体的方法, 由此可见不同管理方法的应用场合不同。然而合理的场景管理, 能够从一定程度上提高虚拟现实引擎的渲染效率。由于要建立的三维动态交互立体商城为室内场景, 且只有商品模型和人物模型可以移动。因此, 文章根据面向三维动态交互立体商城特点, 并结合 BSP 树和场景图的方法

优点, 在虚拟现实引擎的开发过程中设计实现了一种构建场景树的新方法——复合层次树法。

该复合层次树法的主要特点为: ①将立体商城中的静态对象按照 BSP 树管理方法添加到场景树中, 以避免单一使用场景图方法时出现冗余的控制节点; ②为立体商城中的动态对象创建相应控制节点, 并以场景图方法添加到场景树中, 据此可以有效控制商城中店铺内外的动态物体, 增强立体商城的真实感。

3.1 场景树的构建过程

按照复合层次树法构建场景树的流程如图 3 所示, 其中, 选择分割面的原则是以走廊和店铺或店铺与店铺的交界面为主, 且尽量保持树的平衡性。

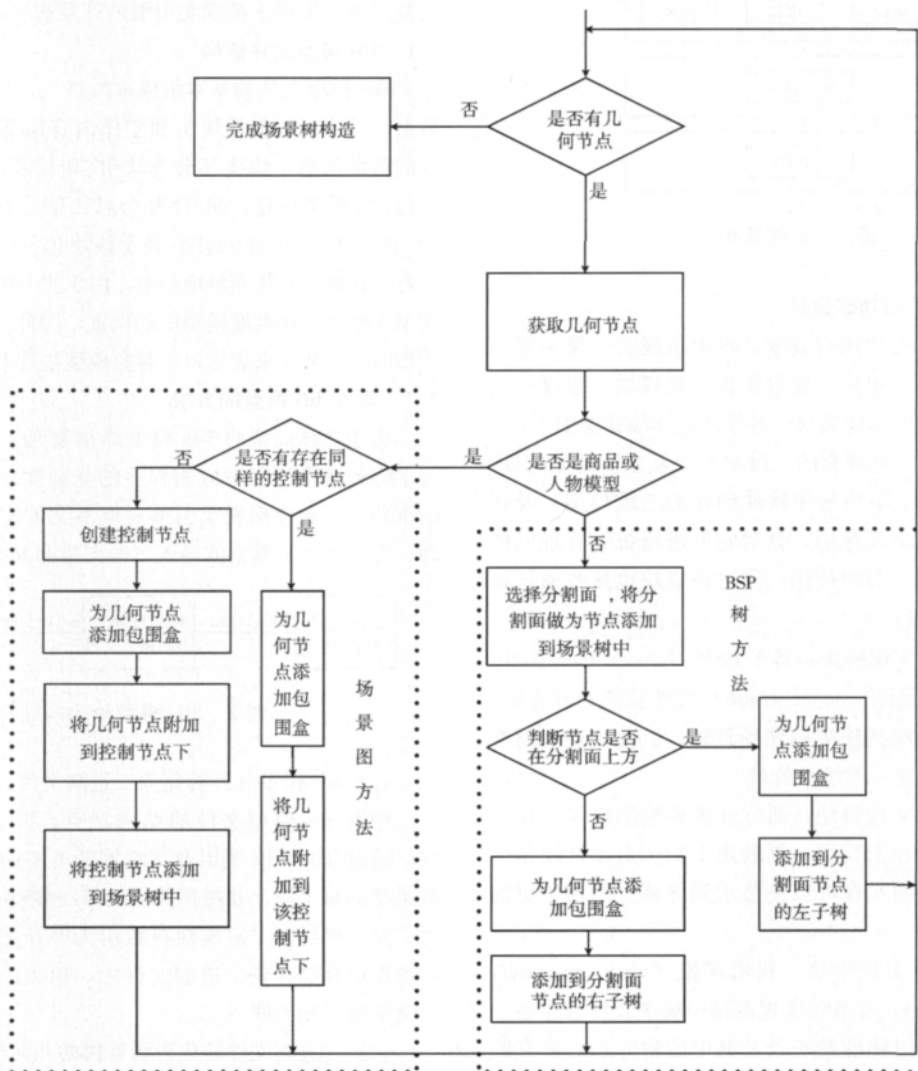


图 3 场景树的构造流程

根据以上方法构造的场景树简图, 如图 4 所示。

3.2 场景树中的操作方法

场景树是根据三维动态交互立体商城中场景和模型的空间位置关系来构建的, 其不仅包含与场景和模型相关的

位置坐标、纹理、渲染状态等静态属性信息, 而且还含有它们所用到的动态操作方法, 其中最主要的是: ①场景数据改变时的更新方法, 实现了人物或商品模型的移动、旋转及大小的改变。②更新包围盒的方法, 用来调整人物或

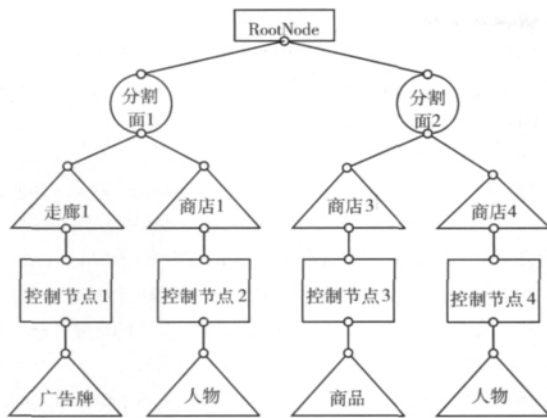


图 4 场景树

商品模型的包围盒大小或位置。两种方法相互配合,不仅可以加快引擎的执行速度,而且可以提高碰撞检测和渲染的精确度。

4 碰撞检测技术

人物模型在三维动态交互立体商城中走动观看的时候,应避免人物沿着墙壁走上去或从障碍物中穿过等不符合自然现象的情况出现。为此设计并实现了 AABB 和 Capsule 包围盒相结合的碰撞检测方法。即在构建的场景树中,为所有静止的场景、商品架及商品模型添加 AABB,为移动的人物模型添加 Capsule 包围盒。之后通过判断两个测试对象包围盒的相交情况来确定人物模型是否与另一个对象模型发生碰撞。若发生碰撞,则通过更新场景数据的方法改变人物模型的移动方向,同时需要通过更新包围盒的方法对其包围盒的位置方向做相应调整;否则,人物模型将继续前行,并随之调整其包围盒的位置。具体的实现过程如图 5 所示。

5 实时渲染

虚拟现实引擎的最终目的是把现有的 3D 模型经过处理后,渲染并投影到 2D 屏幕上。为此,先定义实现了几个主要功能:①创建视锥模型实现对渲染空间的剪裁;根据给定的位置、方向和视口设置等,计算出与渲染对象相关的数据信息。②通过视变换^[1]把渲染对象从世界空间转变到摄像机空间,然后再进行投影变换^[2]把摄像机空间中的对象转变到屏幕空间。③为渲染队列定义空间对象列表。④在渲染之前根据要渲染对象的透明程度,把场景元素分为透明、不透明以及正常颜色队列,然后对正常颜色队列中的对象用 Z-Order 方法进行渲染顺序排列操作,以避免渲染后出现不正常的遮挡现象。

在上述功能的基础上实现引擎实时渲染的具体步骤如下:

- (1) 调用 LWJGLRenderer 类中的 draw (Spatial s) 方

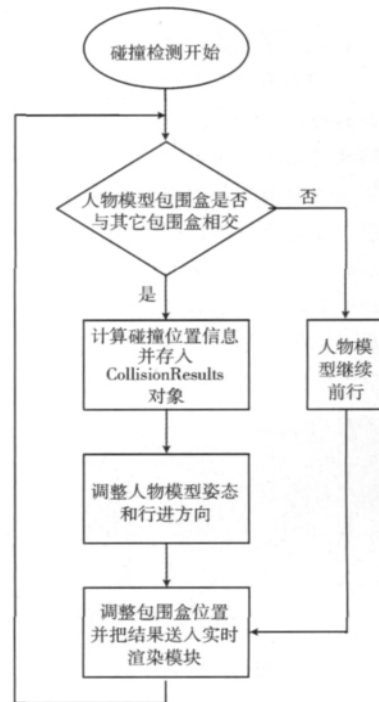


图 5 碰撞检测流程

法。即通过调用 Spatial 类中的 onDraw (LWJGLRenderer r) 方法确定要渲染的场景或模型。

(2) 判断要渲染的场景或模型是否可见。首先获得要渲染对象的剪裁属性 (Always、Dynamic、Inherit 或 Never); 然后创建摄像机类对象, 获取其视锥模型当前面的状态, 并通过 frustrumIntersects (用来表示视锥与面的相交情况) 属性判断是否对该对象进行剪裁操作; 最后重新设置视锥模型当前面的状态。

(3) 如果可见, 调用该场景或模型的 draw 方法。

(4) 如果要渲染的为 TriMesh 类对象, 则把这些数据渲染到 2D 屏幕, 该功能由 draw (TriMesh tris) 方法实现。其中 TriMesh 类是通过点、颜色、法向量和纹理的集合定义了一个 3D 对象的网状形式。

6 关键技术的应用效果

在虚拟现实引擎开发过程中, 通过对 3ds 模型文件的格式转换, 将三维动态交互立体商城的模型数据加载到引擎中, 不仅弥补了利用 LWJGL 的 API 不能创建复杂场景的缺陷, 而且通过专业建模软件 3DSMAX 构建立体商城模型, 也避免了通过大量程序代码创建相关模型的繁琐过程, 从而减少了代码量, 简化了虚拟现实引擎的开发过程, 因此加快了虚拟现实引擎的开发速度。由于本文提出的复合层次树构建方法, 仅仅是对可移动模型添加了相应控制节点, 所以在控制过程中将不需要考虑可移动模型节点是否继承父节点中的控制操作, 简化了场景管理中控制操作的

实现过程。

根据三维动态立体交互商城中场景、商品和人物模型的特点,采用 AABB 和 Capsule 包围盒相结合的方法,实现了非物理性的碰撞检测,在此情况下仅需计算结构简单的外包围盒的相交情况,便可以判断是否发生了碰撞,而不需要根据复杂的模型结构数据信息进行碰撞检测,从而减少了计算量,提高了碰撞检测的执行效率。视锥、渲染队列和 Z-Order 相结合实现的实时渲染技术,不仅减少了送入渲染通道中的场景模型数量,而且给定了合理的渲染顺序,其提高了渲染速度、增强了渲染效果真实性。经该引擎处理后的模型显示效果如图 6 所示。

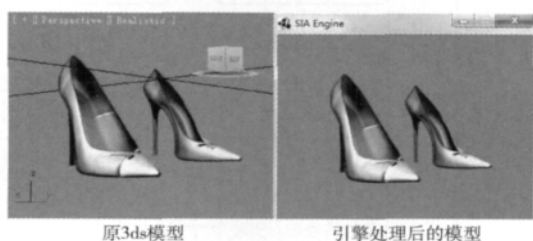


图 6 模型效果

实验证明,在虚拟现实引擎的实现过程中采用的这些关键技术,不仅可以从一定程度上减少开发时的工作量,节约了开发时间,而且还提高引擎的执行效率,增强了其实用性及准确性。

7 结束语

本文提出的复合层次树场景管理方法,降低了虚拟现实引擎执行过程中对物体模型实施控制操作的复杂度;而且利用 AABB 和 Capsule 包围盒相结合实现的碰撞检测方法,提高了人物模型参与碰撞检测时的精确度。与常用方法相比,文中设计实现的这两种方法比较适用于面向三维动态立体交互商城的虚拟现实引擎的开发。文章介绍的模型导入、场景管理、碰撞检测及实时渲染关键技术的实现方法,在虚拟现实引擎开发中得到了深入应用,这些技术在面向三维动态立体交互商城的虚拟现实引擎开发中具有通用性,且提高了虚拟现实引擎的开发效率。在今后的工作中,可进一步探索骨骼动画技术,使虚拟现实引擎的功能更完善,更具有通用性。

参考文献:

- [1] Fletch Dunn, Ian Parberry. 3D math primer for graphics and game development [M]. SHI Yinxue, CHEN Hong, WANG Rongjing, transl. Beijing: Tsinghua University Press, 2010: 85-368 (in Chinese). [Fletch Dunn, Ian Parberry. 3D 数学基础: 图形与游戏开发 [M]. 史银雪, 陈洪, 王荣静, 译. 北京: 清华大学出版社, 2010: 85-368.]
- [2] Dave Shreimer. The Khronos OpenGL ARB Working Group. OpenGL programming guide [M]. 7th ed. LI Jun, XU Bo, transl. Beijing: China Machine Press, 2010: 1-307 (in Chinese). [Dave Shreimer. The Khronos OpenGL ARB Working Group. OpenGL 编程指南 [M]. 7 版. 李军, 徐波, 译. 北京: 机械工业出版社, 2010: 1-307.]
- [3] ZHOU Feng, NI Junfang, ZENG Xianzheng, et al. Research on reading and redrawing of 3DS models based on OpenGL [J]. Journal of Soochow University (Engineering Science Edition), 2009, 29 (2): 53-56 (in Chinese). [周峰, 倪俊芳, 曾宪政, 等. 基于 OpenGL 对 3DS 模型的读取与重绘的研究 [J]. 苏州大学学报 (工科版), 2009, 29 (2): 53-56.]
- [4] David H Eberly. 3D game engine design: A practical approach to real-time computer graphics [M]. 2nd ed. Beijing: Posts & Telecom Press, 2009: 7-895 (in Chinese). [David H. Eberly. 3D 游戏引擎设计实时计算机图形学的应用方法 [M]. 2 版. 北京: 人民邮电出版社, 2009: 7-895.]
- [5] Christer Ericson. Real-time collision detection [M]. LIU Tianhui, transl. Beijing: Tsinghua University Press, 2010: 49-162 (in Chinese). [Christer Ericson. 实时碰撞检测算法技术 [M]. 刘天慧, 译. 北京: 清华大学出版社, 2010: 49-162.]
- [6] GAO Lina. Study on visibility culling in virtual reality engine [D]. Nanjing: Nanjing University of Technology, 2006: 11-29 (in Chinese). [高丽娜. 虚拟现实引擎中可见性裁减问题的研究 [D]. 南京: 南京工业大学, 2006: 11-29.]
- [7] CHEN Kai. Design and implementation of 3D game engine [D]. Hangzhou: Zhejiang University, 2007: 7-26 (in Chinese). [陈凯. 三维游戏引擎的设计与实现 [D]. 杭州: 浙江大学, 2007: 7-26.]
- [8] FAN Cui. Design of 3D game engine and implementation of key technologies [D]. Xi'an: Northwestern Polytechnical University, 2006: 27-36 (in Chinese). [樊翠. 三维游戏引擎的设计及关键技术的实现 [D]. 西安: 西北工业大学, 2006: 27-36.]
- [9] GENG Weidong, CHEN Kai, LI Xin. Implementation of 3D game engine [M]. Hangzhou: Zhejiang University Press, 2008: 8-47 (in Chinese). [耿卫东, 陈凯, 李鑫. 三维游戏引擎设计与实现 [M]. 杭州: 浙江大学出版社, 2008: 8-47.]
- [10] ZHANG Jing. Virtual reality technology and application [M]. Beijing: Tsinghua University Press, 2011: 1-66 (in Chinese). [张菁. 虚拟现实技术及应用 [M]. 北京: 清华大学出版社, 2011: 1-66.]
- [11] ZHOU Li, SU Honggen. Research on general file formats and algorithms of 3-D models and their implementation with OpenGL [J]. Computer Engineering and Design, 2009, 30 (2): 433-434 (in Chinese). [周莉, 苏鸿根. 通用 3D 模型文件格式和算法的研究及其 OpenGL 实现 [J]. 计算机工程与设计, 2009, 30 (2): 433-434.]
- [12] Andrew Davison. Java™ 6 3D game development Java 3D™, JOGL, Jinput, and JOAL APIs [M]. the United States of America: Apress Inc, 2007: 3-200.