

# Review of Three-Dimensional Graphics Engine

Xinliang Wei<sup>1,2</sup>, Wei Sun<sup>1,2</sup>

<sup>1</sup>School of Software, SUN Yat-sen University, Guangzhou Guangdong

<sup>2</sup>Key Laboratory of Information Technology (Ministry of Education), SUN Yat-sen University, Guangzhou Guangdong

Email: [weixinl@mail2.sysu.edu.cn](mailto:weixinl@mail2.sysu.edu.cn), [sunwei@mail.sysu.edu.cn](mailto:sunwei@mail.sysu.edu.cn)

Received: Sep. 5<sup>th</sup>, 2015; accepted: Sep. 26<sup>th</sup>, 2015; published: Sep. 30<sup>th</sup>, 2015

Copyright © 2015 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

---

## Abstract

**Objective:** Virtual Reality is the high-tech advanced technology developed in recent years, calculated using high-speed computer to simulate a virtual three-dimension space. It allows users to have sensory simulation with vision, hearing, touch and so on. Users feel that they are in the area, and can observe all the thing in the three-dimensional space in real time and without limitation. The three-dimensional graphics rendering technology is not only the basis of software technique in virtual reality, but also the core of the three-dimensional graphics engine. In order to make more researchers have a more comprehensive understanding in 3D graphics engine theory, structure and principle, a detailed introduction was reviewed. **Method:** A large number of 3D engine theory were summarized, classified and compared. This paper introduces the definition of the virtual reality briefly, as well as mainly introducing the research status, basic structure and draw principle of the three-dimensional graphics engine. **Result:** So far, there are so many 3D engine with many differences, but the basic structure and drawing principle are the same. The change is how to organize the structure of each module. In order to make the engine has maximum performance, we should consider the relationship of each module. **Conclusion:** The 3D engine technology is the basis of virtual reality, and there are a number of problems we should overcome. Some problems also need further research.

## Keywords

Virtual Reality, Three-Dimensional Graphics Engine, Basic Structure, Draw Principle

---

## 三维图形引擎综述

魏新亮<sup>1,2</sup>, 孙伟<sup>1,2</sup>

<sup>1</sup>中山大学软件学院, 广东 广州

<sup>2</sup>中山大学信息技术教育部重点实验室, 广东 广州

Email: [weixinl@mail2.sysu.edu.cn](mailto:weixinl@mail2.sysu.edu.cn), [sunwei@mail.sysu.edu.cn](mailto:sunwei@mail.sysu.edu.cn)

收稿日期: 2015年9月5日; 录用日期: 2015年9月26日; 发布日期: 2015年9月30日

## 摘要

**目的:** 虚拟现实是近年来出现的高新前沿技术, 利用电脑高速计算, 模拟产生一个虚拟的三维空间, 让使用者拥有视觉、听觉、触觉等感官的模拟, 如同亲历其境一般, 可以实时、无限制地观察和体验三度空间内的所有事物。虚拟现实的软件技术基础是三维图形渲染技术, 三维图形渲染技术也是三维图形引擎的核心。为使国内外三维图形研究者对三维图形引擎的理论, 架构和原理有一个比较全面的认识, 对其进行比较系统地综述。**方法:** 通过对国内外大量的虚拟现实, 三维引擎的知识进行总结, 分类及比较, 本文简要介绍了虚拟现实的定义, 主要论述了三维图形引擎的研究现状, 基本架构和绘制原理。**结果:** 目前存在很多三维引擎, 每个引擎的功能也许存在差异, 不过在基本架构和绘制原理上是没有变的, 改变的则是对架构中不同模块的组织, 如何合理组织各个模块的关系, 使引擎能发挥最大的性能。**结论:** 三维图形引擎技术是虚拟现实技术的基础, 目前仍然有许多需要克服的问题, 针对不同的问题还需要进行深入地研究与学习。

## 关键词

虚拟现实, 三维图形引擎, 基本架构, 绘制原理

## 1. 引言

虚拟现实(Virtual Reality, 简称 VR, 又称为虚拟实景), 是一种基于可计算信息的多通道的沉浸式交互环境, 它是以计算机技术为核心, 结合其他相关科学技术(如交互技术、人工智能等), 生成一个在视、听、触感等方面与逼真环境高度近似的数字化环境, 用户可以借助其他设备与该环境进行交互, 从而产生身临对应真实环境的感受和体验[1] [2]。虚拟现实的研究已经有 20 多年, 直到现在, 它仍是计算机图形学三大热门话题之一。它以“3I”的特性(见图 1) (Immersion 沉浸感、Interaction 交互式、Imagination 想象力), 应用于各个方向领域, 包括公共安全、工业设计、医疗辅助、军事模拟、航空航天、交通规划和文化教育等, 其应用又集中在培训演练、规划设计和展示娱乐三方面。

虚拟现实研究的主要内容包括实时三维图形生成技术、多传感器交互技术和高分辨率现实技术[2]。其软件技术基础则是三维图形渲染技术。目前三维图形技术的底层图形开发包主要有三类:

(1) 微软公司制定的 Direct3D, 并以此衍生出的基于 .Net 框架平台的 XNA 技术, 以及适用于浏览器的 Silverlight 技术;

(2) Khronos 工作组的 OpenGL, 以及用于移动 3D 图形开发的 OpenGL ES 和用于网络 3D 开发的 WebGL;

(3) Adobe 公司的 Flex, 可运行于装有 Flash 的系统或浏览器中;

三种图形开发包的优缺点本文暂不详细论述。但三种图形开发包都属于底层的图形开发包, 对于开发人员来说, 直接接触底层的技术将面临学习更多的专业知识和算法, 这使得开发变得困难, 也降低了开发效率。为了提高开发效率以及使得开发变得简便, 一种被称为“三维引擎”的中间件被开发出来。引擎, 从程序开发角度而言, 即是单独的一组功能框架, 其封装了三维图形算法和一系列的 API 函数。

引擎相当于程序员的开发工具包, 有了这个工具包, 开发人员就可以很快地搭建三维图形程序的雏形(见图 2)。其中专门负责三维图形渲染的模块或框架, 我们称之为“三维图形引擎”, 为了方便, 我们也常把“三维图形引擎”简称为“三维引擎”。三维图形绘制工具对比见表 1。

三维引擎最大的特点就是其“驱动性”, 即引擎仅仅在功能上支持某个方面的应用, 但具体的实现细节则依赖于应用本身。如 RPG 类型的游戏引擎只能用来制作 RPG 游戏, 而及时战略类的游戏引擎只能用来制作及时战略类的游戏。其次, 引擎具有“完整性”的特点, 即引擎是能完整实现某个方面功能的函数集合。再则, 引擎具有“独立性”的特点, 即引擎不依赖于具体的应用而独立存在。为了提高三

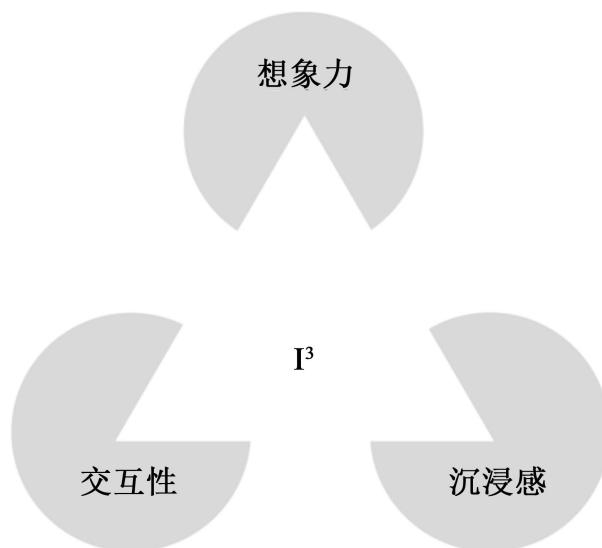


Figure 1. “3I” feature: immersion, interaction, imagination

图 1. “3I” 特征: 沉浸 - 交互 - 想象[1]

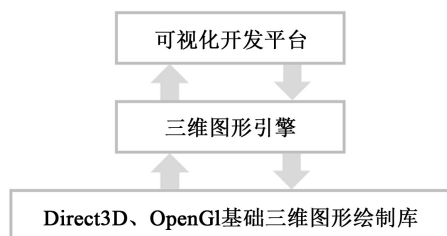


Figure 2. The development level of 3D drawing tools

图 2. 三维图形绘制工具开发层次

Table 1. The comparison of 3D drawing tools

表 1. 三维图形绘制工具对比

	基础三维图形绘制库	三维图形引擎	可视化开发平台
接口方式	API	IDE/API	可视化开发环境/API
封装方式	函数集合	面向对象/函数集合绘制程序架	面向对象/函数集合绘制程序框架
灵活性	高	较高, 绘制框架受限	较高, 开发环境受限
执行效率	高	高	不高
技术要求	高	中	低
系统开发量	大	中	较小

维图形程序开发者们的效率,在设计一个三维图形引擎时也要考虑充分三维引擎模块的高隐藏性和可扩展性。

三维引擎是虚拟现实技术的一部分,应用领域非常广泛。具体应用于军事训练、企业生产、科研方面、游戏娱乐、商业应用等等。三维引擎技术不仅仅在虚拟现实方面应用广泛,在计算机辅助设计与制造、CG制作、航空航天、地质勘探等方面也有着广泛应用[3]。其中,游戏娱乐是三维引擎发展的重要领域。2003年,国家863计划将三维游戏引擎的研发纳入了高技术发展计划。目前,我国的三维引擎开发比较落后,研发一款自己的三维引擎,研究三维引擎的关键技术是一次有意义的尝试,许多商业公司,学校,研究所加入了研究三维引擎的行列,这对我国的三维引擎技术的发展与进步起到了积极的推动作用。

本文在前人的基础上对三维图形引擎进行综述,首先介绍了三维图形引擎的发展现状,然后介绍三维图形引擎的基本架构和绘制原理,最后展望一下三维图形引擎的未来趋势以及指出三维图形引擎需要进一步需要解决与完善的若干技术问题。

## 2. 三维图形引擎研究现状

### 2.1. 国外研究现状

#### 2.1.1. 虚拟现实的发展

SRI研究中心建立了“视觉感知计划”,研究现有VR技术的进一步发展。1991年后,SRI进行了利用VR技术对军用飞机或车辆驾驶的训练研究。另外,SRI中心还利用遥控技术进行外科手术仿真的研究[4]。

美国北卡罗来纳大学(UNC)的计算机系是VR研究最早最著名的大学。他们主要研究分析建模、航空驾驶、外壳手术仿真、建筑仿真等技术。由Brooks教授领导的小组成功研制了第一个用于建筑设计的Walk-through虚拟建筑漫游系统,用户可以在虚拟的UNC计算机系大楼里面漫游[5]。

伊利诺斯州立大学研制出在车辆设计中,支持远程协作的分布式VR系统。不同国家、不同地区的工程师们可以通过计算机网络实时协作进行设计。研究人员在系统中采用了虚拟原型,从而减少了设计图像和新产品进入市场的时间,而且可以在新产品生产之前就能对其进行估算和测试,这样就大大地提高了产品质量[6]。

美国宇航局(NASA)的Ames实验室完善了HMD,并将VPL的数据手套工程化,使其称为可用性较高的产品。NASA研究的重点放在空间站操纵的实时仿真上,大多数研究是在NASA的约翰逊空间中心完成的。他们大量运用了面向座舱的飞行模拟技术。NASA完成的一项著名的工作是对哈勃太空望远镜的仿真[6]。

美国UC Berkeley漫游工作室在建筑漫游方面的工作颇具代表性。他们对Berkeley大学计算机系大楼Soda Hall进行了模拟漫游,用于及时发现并修正了建筑设计中存在的缺陷。Soda Hall模型由1418807个多边形构成,占据21.5M硬盘空间,使用了406中材质及58种不同纹理。由于研究小组开发了高效的漫游引擎,实现了Soda Hall在SGI Power Series 320平台上的实时漫游[6]。

麻省理工大学(MIT)是一个一直走在最新技术前沿的科学研究机构。MIT原先就是研究人工智能、机器人和计算机图形学以及动画的先锋。这些技术都是虚拟现实技术的基础。MIT在1985年成立了媒体实验室,进行虚拟环境的正规研究。媒体实验室建立了一个名叫BOLIO的测试环境,用于进行不同图形仿真技术的实验。利用这一环境,MIT建立了一个虚拟环境下的对象运动跟踪动态系统[6]。

#### 2.1.2. 三维图形引擎的发展

最早的一款图形引擎Space Rogue诞生于1990年,由Origin System公司开发,这个引擎的出现给以

后的引擎带来了一个惯例,就是用游戏名来命名引擎名,一直沿用至今。可惜受限于当时的物理运算能力,引擎的运算速度也被完全限制了,以至于这款引擎运行在 386 配置上的引擎,在 486 的机器上运行显得非常吃力。虽然这款引擎并未取得太大的成功,但是它却为游戏开发带来了春天,此后图形引擎如雨后春笋般一发不可收拾[7]。

1992 年,由 NovaLogic 公司开发的 Voxel 引擎问世,在此引擎基础上开发的作品有很多闻名于世的,包括命令与征服、红色警戒 2、三角洲等等。它在 3D 领域的尝试上迈出了重要的一步,结合体积和像素的概念,让 2D 的块拥有了高度,使图形更加平滑翔实。

1993 年, ID Software 公司开发了 ID tech 引擎,与此同时出现的就是大名鼎鼎的世界第一款三维第一人称射击游戏——DOOM,尽管它知识一款伪 3D 的引擎,但在当时主流 VGA 显卡的环境纹理映射渲染技术的支持下,让玩家在一台 386 机器上就可以领略到 DOOM 的风采[7]。同年,一款更加出名的游戏《毁灭公爵》让其开发引擎 Build 名声大噪。

1995 年,世界第一款真正的三维引擎 Xengine 终于横空出世,带来了著名的上古卷轴系列,创造出真实的游戏世界。一年以后由 ID Software 公司开发的 Quake 引擎,采用了大量技术包括 Z-buffer, 三维光源以及抗锯齿等,使雷神之锤占据了游戏的主流,也预示着三维图形引擎开始走向成熟。

在此之后,越来越多的商业三维图形引擎涌现出来,如 RenderWare, Quake II/ID Tech2, Cyan Plasma, GoldSRC, Unreal, Lithtech, Outcast, Quake III/ID Tech 3, GeoMod, Torque, Serious, Max-FX。几乎每一个引擎的名字都代表了新技术的发展,都代表着一个著名游戏系列的出现,都意味着一次图形学技术的进步,都使得图形引擎产业更加繁荣。

到现在为止,一个又一个的优秀引擎将游戏图形技术推向了更高峰,以 CryEngine 2 和 Unreal 3 为代表的优秀引擎更是在当前硬件条件支持下,将各种游戏图形发展到一个登峰造极的高度[7]。游戏开发者完全可以利用引擎所给出的 API 快速开发出极高质量的游戏,而不需要再编写大量冗余的代码,真正达到了所见即所玩的程度。

## 2.2. 国内研究现状

北京航空航天大学计算机系是国内最早研究 VR 技术,最有权威的单位之一。他们进行了一些基础研究,并开发了自己的虚拟现实运用系统平台,设计了虚拟环境漫游引擎,实现了诸如虚拟校园、飞行员模拟训练、房地产项目虚拟花园、虚拟珠峰漫游等项目[6]。

清华大学计算机系对虚拟现实和临场感方面进行了研究,在球面屏幕现实和图像随动、克服立体图闪烁的措施和深度感实验等方面都具有不少独特的方法。

浙江大学 CAD&CG 国家重点实验室开发出了一套桌面型虚拟建筑环境实时漫游系统。该系统采用了层面叠加的绘制技术和预消隐技术,实现了立体视觉,同时还提供了方便的交互工具,使整个系统的实时性和画面真实感都达到了较高的水平。另外,他们还改进了快速漫游算法和递进网格的快速生成算法,开发了虚拟紫禁城等虚拟环境漫游项目[4]。

四川大学计算机学院开发了一条基于 OpenGL 的三维图形引擎 Object-3D,该系统在微机上使用 Visual C++5.0 语言实现,其主要特征是:采用面向对象机制;与建模工具如 3D Max, MutiGen 相结合;对用户屏蔽一些底层图形操作;支持常用三维图形显示技术;支持动态裁剪技术等[5]。

中国地质大学分析了基于微机的三维应用程序的结构特点,提出了一个基于 OpenGL 和 Direct 3D 两种 3D API 的三维图形引擎结构框架。该框架已经成功应用到他们自己开发的系统“三维城市景观浏览器 Map3DViewer”中,并收到了较好的效果[5]。

2003 年,国家 863 计划正式将游戏引擎的研发纳入国家高科技发展计划,并将研发的任务交给了金

山公司。金山公司的负责人也表示,他们已经开发出一个比较完善的引擎并应用与游戏开发中,最具代表性的游戏就是《剑网3》。该引擎应用“植被渲染、地形渲染、Normal Map、Speed Tree、柔体程序算法”等先进技术,从而保证游戏的画面的真实感。

2007年涂鸦软件发布的Origo Engine(起点引擎)是国内第一款商用三维图形引擎,作为国内商用三维图形引擎的起点,其绘制质量和效率都有不错表现,初步具备与国外引擎竞争的实例。

2012年,畅游自主研发的Darkfire Engine(黑火引擎)发布,该引擎是专为次世代游戏开发而生的三维游戏引擎,其集成了主流的光影特效算法,在表现力上更上了一个台阶。

此后,由于各三维引擎的发展,出现了一大批游戏和影视特效公司,如网易、搜狐、巨人集团以及水晶石数字科技有限公司等等。并且很多大学也相继建立了图形学国家重点实验室,开始研究虚拟现实和三维图形引擎的理论知识,开发了自主的图形引擎并实现了一批又一批的应用系统。

中山大学信息技术教育部重点实验室自主研发了一款基于XNA/Sliverlight的三维图形引擎,该引擎能够很好地仿真出自然场景中的物体,包括天空、水、雨、树木等,并具有良好的柔和阴影贴图功能。目前基于该引擎,该实验室与各相关部门合作开发了大量的虚拟展示平台,效果良好。

### 2.3. 主流三维引擎介绍

一些极具影响力的公司开发了一批应用于游戏方面的三维图形引擎,下面将分别介绍主要的几个引擎[8]:

#### 2.3.1. Quake(雷神)系列引擎

Castel Wolfenstein 3D(1992)是第一款被游戏玩家广泛接受的3D第一人称射击类游戏(FPS)。这款游戏是由总部位于Texas的Id Software公司为PC平台开发的,它为游戏产业的发展指明了一个激动人心的新方向。Id Software公司随后又接连开发了Doom、Quake、Quake II和Quake III四款游戏,这四款游戏所使用的三维图形引擎架构类似,我们统称为Quake系列引擎。

Quake在三维图形引擎发展过程中具有里程碑的意义,它具备了现在三维图形引擎的雏形,并激励了后来者们的研发热情。许多基于Quake技术构建的游戏,并非是直接使用Quake引擎构建的,而是通过一些其他的游戏或平台间接地使用到Quake技术。事实上,Valve公司的Source引擎最初也根植于Quake技术[9]。

#### 2.3.2. Unreal(虚幻)系列引擎

Epic Games Inc.公司在1998年突然出现在第一人称射击类游戏领域,并带来了一个传奇性的游戏Unreal。从那之后,Unreal引擎就一直是Quake引擎在FPS领域最大的竞争对手。Unreal Engine 2(UE2)引擎是Unreal Tournament 2004的基础。大量的玩家和程序员据此开发出了数之不尽的“MOD”、大学项目和商业游戏。而Unreal Engine 3(UE3)引擎则宣传说它带来了业界最好的公爵和最丰富的游戏特性集,其中包括一个方便且强大的着色器构建图形用户界面以及一个被称为Kismet的方便且强大的游戏逻辑编程图形用户界面[10]。

Unreal引擎因其丰富的扩展性和易于使用的内置工具而广为人知。Unreal引擎并不是完美的,所以大多数开发者都会对其进行某些修改以使他们的游戏在特定的平台上能以最佳的方式运行。但Unreal引擎的确是一个异常强大的原型工具和商业游戏开发平台,几乎可以用它来构建任何第一人称和第三人称的游戏。

除了游戏,Unreal引擎还涵盖了教育、建筑等其他领域。Digital Design公司曾与联合国教科文组织的世界文化遗产分部合作采用Unreal引擎制作过巴黎圣母院的内部虚拟演示。Zen Tao公司采用Unreal

引擎为空手道选手制作过武术训练软件。另一家软件开发商 Vito Miliano 公司也采用 Unreal 引擎开发了一套名为“Unrealty”的建筑设计软件, 用于房地产的演示。

目前, Unreal 引擎已经发布了第四代引擎 Unreal 4, 它可以在不榨干硬件机能的情况下同时渲染数以百万的粒子, 并且包含了新的 Kismet 2 工具, 它让开发者在简单地操作界面下就能渲染出复杂的场景。这些特性使得虚幻 4 的 SDK 将更加简单易用, 从而允许更少的开发者完成更多的工作。

### 2.3.3. Source 引擎

Source 引擎是获得巨大成功的 Half-Life 2 及其续集 HL2: Episode One、HL2: Episode Two、Team Fortress 2 和 Portal 的三维引擎。Source 引擎是由 Valve 软件公司开发的一款高质量的引擎, 在图形性能和工具集方面可以与 Unreal Engine 3 引擎相匹敌, 且它对其他的游戏开发者开放授权。该引擎提供关于渲染、声效、动画、消锯齿、界面、网络、美工创意和物理模拟方面的支持。

### 2.3.4. XNA Game Studio

微软公司的 XNA Game Studio 是一款易于使用、非常易学的游戏开发平台。它鼓励玩家制作自己的游戏, 并在网上的社区中分享自己的作品, 就像 Youtube 鼓励用户制作并共享自拍视频一样。

XNA 基于微软的 C# 语言和公共语言运行时 (CLR) 技术。主要的开发环境是 Visual Studio, 或其免费版 Visual Studio Express。所有资源, 包括源代码和游戏资源等, 都由 Visual Studio 进行管理。开发者可以使用 XNA 为 PC 平台和微软的 Xbox 360 游戏机制作游戏。在付了一定费用后, 玩家可以把自己的作品上传到 Xbox Live 网络中与其他人共享。通过无偿地提供一些优秀的工具, 微软为普通用户打开了一扇制作新游戏的大门。

### 2.3.5. Unity 引擎

Unity 是由 Unity Technologies 开发的一个让玩家轻松创建诸如三维视频游戏、建筑可视化、实时三维动画等类型互动内容的多平台的综合型游戏开发工具, 是一个全面整合的专业游戏引擎。Unity 对 Direct X 和 OpenGL 拥有高度优化的图形渲染管道, 并支持所有文件格式, 能和大部分相关应用程序协同工作。Unity 的着色器系统整合了易用性、灵活性和高性能。Unity 也提供了具有柔和引擎和烘焙 lightmaps 的高度完善的光影渲染系统。

### 2.3.6. Gamebryo 引擎

Gamebryo 是一款便捷且易上手的开放式跨平台游戏开发引擎。Gamebryo 为当今主要的 DCC 工具 (3D Max, Maya, XSI) 都提供了游戏所需资源的导出插件。Gamebryo 拥有较完善的场景图结构、渲染系统、材质系统、多线程管理系统、几何模型系统、动画系统和物理系统等等。目前 Gamebryo 引擎正应用于国内的大型 3D 游戏, 如《御龙在天》、《轩辕传奇》等等。总结 Gamebryo 的优点就是, 灵活性、多平台、画面效果、负载能力。

### 2.3.7. Bigworld 引擎

Bigworld 是为大型多人在线游戏 (MMOG) 开发商提供的成熟的中间件平台。Bigworld 和其他游戏引擎不同, 它不是针对地图为单元, 将地图为单元分担到独立进程中去管理的。它是以人群数量去进行划分, 从理论上讲, Bigworld 支持无限大的游戏世界, 而且它完全支持无缝, 因为它完全没有从物理地图区域进行分割。Bigworld 引擎最具代表的作品就是《魔兽世界》。

### 2.3.8. CryEngine 引擎

CryEngine 由 Crytek 开发的一款著名的三维引擎。自微软退出了 Direct 10 之后, CryEngine 2 成为了

当时渲染质量最好的引擎,如今,CryEngine 对平台硬件条件要求变得十分高,导致了其使用并不如 Unreal 引擎广。目前,Crytek 发布了 CryEngine 3 引擎。

### 2.3.9. 其他商业引擎

2001 年之后,除了以上几款商业引擎外,还存在大量的商业三维引擎(如 V12 引擎、MAX-FX 引擎、Geo-Mod 引擎、Serious 引擎、夜光引擎、寒霜 3 引擎、Red Engine3 等等)。尽管独立开发者可以没有足够的预算购买一款商业引擎,但是这些商业引擎却提供了大量的在线文档和 wiki,这些文档和 wiki 中提供了大量有用的三维引擎信息和一般游戏编程知识[11]。

### 2.3.10. 专有内部引擎

许多公司都会构建和维护一些专有内部三维图形引擎。Electronic Arts 公司制作的许多即时战略游戏(RTS)都是基于 Westwood Studios 公司开发的一款称为 SAGE 的专有内部引擎开发的。Naughty Dog 公司制作的 Crash Bandicoot、Jak and Daxter 系列和最近的 Uncharted: Drake's Fortune 都在分别为 PlayStation、PlayStation 2 和 PlayStation 3 定制的三款内部引擎进行了构建[12]。当然,大多数的商业许可的三维图形引擎,例如 Quake、Source 和 Unreal Engine 最初都是专有内部引擎。

### 2.3.11. 开源引擎

开源三维图形引擎是由业余爱好者和专业游戏开发者构建的可以从网上免费获取的引擎。术语“开源”通常意味着可以免费获取引擎的源代码,同时还意味着引擎开发过程中使用了某种开源开发模式,而采用这种开发模式则意味着每个人都可以为引擎贡献代码。开源引擎通常会采用 GNU 公共许可(GPL)或宽松 GNU 公共许可(LGPL)。网上有大量的开源引擎,其中有些引擎非常不错,有些则比较一般,还有一些非常糟糕。

OGRE 3D 是一款架构良好、易于学习、易于使用的三维渲染引擎。它自称拥有一个功能完备的渲染器,包括高级光照和阴影功能、一个优秀的骨骼角色动画系统、一个用于平视现实(HUP)和图形用户接口的二维叠加系统和一个用于制作全屏效果(例如光晕效果)的后期处理系统。ORGE 并不是一个完整的三维引擎,但是它确实提供了大部分三维引擎都需要用到的一些基本组件。

下面列出了一些知名的开源三维引擎:

1) Irrlicht 是一个完全跨平台的 3D 游戏引擎。具有高效,实时等特点,使用 D3D、OpenGL 和它的自己的渲染程序。支持动态的阴影,粒子系统,角色动画,室内和室外技术以及碰撞检测等。提供 Anti-aliasing 支持,可运行于 Linux, Windows, MacOS。

2) Panda3D, 由迪斯尼 VR 工作室开发和维护的 3D 游戏引擎(BSD 授权)。是一款基于脚本的引擎。这一款引擎的主要接口是一种自定义 Python 脚本语言。它主要用于 3D 游戏原型和虚拟世界的快速制作。

3) Crystal Space 是 3D 构架,用 C++语言编写。开发者包括 Jorrit Tyberghein。首个公开版本发布于 1997 年 8 月 26 日,通常用于电子游戏目的。是使用 GNU Lesser General Public License 协议的自由软件,作为 SourceForge.net 2003 年 2 月的月度项目。可用 OpenGL(全平台),SDL(全 SDL 平台),X11(Unix 和 GNU/Linux)或 SVGALib (GNU/Linux)。

4) Genesis-3D 是国内游戏公司搜狐畅游研发并发布的一个可以让玩家及开发者轻松创建三维视频游戏等的首款国内外开源 3D 游戏引擎平台。它包括跨平台渲染引擎、2D 引擎、物理引擎、音效系统、粒子系统、动画系统、服务器引擎、游戏设计工具等,可适用于各种类型游戏研发的综合型游戏开发平台。

5) Yake 是一款在 OGRE 基础上构建的功能完备的三维图形引擎。它是一款组件化、面向对象化的引擎。



## 2.4. 主流三维引擎对比

由于大部分三维图形引擎工具没有公开体系结构、关键技术等技术细节, 不能对它们进行完整的对比分析, 下面就部分图形绘制功能进行对比, 如表 2~表 6。资料有限, 部分内容可能不够全面。以下选择目前行业内主流的 5 款三维引擎进行对比:

5 款引擎分别是 ORGE 引擎、Unreal 引擎、Unity 引擎、Gamebryo 引擎以及 Bigworld 引擎。其中 ORGE 引擎属于免费的开源引擎, 其余 4 款引擎属于收费的商业引擎。

Table 2. The comparison of market data

表 2. 各引擎市场资料对比

引擎名称	开发成功的作品
OGRE	搜狐畅游:《天龙八部》、《鹿鼎记》; 麒麟网:《成吉思汗》
Unreal	《细胞分裂》、《明日潘多拉》、《天堂 II》、《杀手 13 (XIII)》、《虚幻竞技场》
Unity	《捉迷藏 OL》、《蒸汽之城》
Gamebryo	腾讯:《御龙在天》、《轩辕传奇》、《QQ 飞车》; 盛大:《星辰变》; 上海烛龙:《古剑奇谭》; 久游:《宠物森林》; 金酷游戏:《魔界 2》; 成都梦工厂:《侠义道 III》
Bigworld	《魔兽世界》、《天下贰》、《创世西游》、《鬼吹灯 OL》、《三国群英传 OL2》、《侠客列传》

Table 3. The comparison of basic function

表 3. 各引擎基本功能对比

引擎名称	渲染系统	光源系统	场景管理	纹理系统	物理系统
OGRE	固定渲染通道、渲染纹理、字体、图形用户界面	着顶点、着像素、光照映射	普通方式、BSP、遮挡剔除、LOD、Octree	基本纹理、多重纹理、凹凸纹理、投影纹理、体纹理	碰撞检测和刚体
Unreal	固定渲染通道、可编程渲染、字体、图形用户界面	着顶点、着像素、光照映射	普通方式、BSP、基于入口的可见性判别、遮挡剔除、PVS、LOD	基本纹理、多重纹理、投影纹理	碰撞检测、刚体和车辆物理学
Unity	固定渲染通道、可编程渲染、字体、图形用户界面	高度完善的光影渲染系统	普通方式、BSP、LOD、遮挡剔除、PVS	基本纹理、多重纹理	内置 PhysX 物理引擎
Gamebryo	固定渲染通道、可编程渲染、字体、图形用户界面	着顶点、着像素	多线程管理系统 (Floodgate)、PVS	基本纹理、动态纹理	整合 PhysX 系统到 Gamebryo 中
Bigworld	固定渲染通道、可编程渲染、字体、图形用户界面	着顶点、着像素	四叉树、支持室内室外环境的无缝链接	基本纹理、多重纹理	简单的物理系统

Table 4. The comparison of animation interactive function

表 4. 各引擎动画交互功能对比

引擎名称	动画系统	人工智能系统
OGRE	逆向运动学、骨骼动画、Skeletal-Ani、动画混合、Blending	不支持
Unreal	关键帧动画、骨骼动画、动画混合	寻径、决策、有限状态机、脚本
Unity	支持 DCC 等工具导出生成的动画、脚本控制动画播放	不支持, 需要编写脚本 AI
Gamebryo	骨骼动画、面部动画、层级动画、动画序列、动画与物理模拟结果的混合、动画序列之间的混合	支持, 利用 xaitmentAI 中间件提供人工智能
Bigworld	支持四骨以上的复杂骨骼动画	支持

Table 5. The comparison of effective function

表 5. 各引擎特效功能对比

引擎名称	特效功能	音效	网络
OGRE	环境映射、镜头光斑、布告板、粒子系统、运动模糊、天空盒、水、雾	不支持	不支持
Unreal	镜头光斑、布告板、粒子系统、天空盒、水、爆炸效果、贴花、雾、镜头特效	2D、3D 音效	支持 CS 模式
Unity	粒子系统、曲面 fessellation 技术、凹凸纹理贴图、多边形曲面光照效果、环境光效果、水	实时三维图形混合音频流、视频流	从单人游戏到全实时多人游戏
Gamebryo	粒子系统、HDR、DOF、常见特效	使用 Audiokinetic 开发的 Wwise 音频中间件	模块化、可扩展的网络库
Bigworld	支持大量景深局部对比、增益、爆炸、烟雾	基于多普勒效应的音频处理	支持 CS 模式、服务器使用 RedHat 版的 Linux

Table 6. The comparison of basic information

表 6. 各引擎基本信息对比

引擎名称	公司/作者	图形 API	开发语言	是否开源	文档情况	特征情况
OGRE	Steve streeting	OpenGL、DirectX	C/C++	公开	充分	面向对象、插件式结构、可存取系统
Unreal	Epic Games	OpenGL、DirectX、Software	C/C++	不公开	充分	面向对象、插件式结构、可存取系统
Unity	Unity technology	OpenGL、DirectX	C/C++	不公开	提供逐步的指导、文档和实例方案	面向对象设计
Gamebryo	Emegent game technology	OpenGL、DirectX	C/C++	不公开	充分	面向对象、开放式跨平台游戏引擎、插件式结构、可存取系统
Bigworld	Big World	OpenGL、DirectX	C/C++	部分源代码公开	充分	面向对象、插件式结构、可存取系统

### 3. 三维图形引擎基本架构

目前，三维图形引擎应用最多就是三维游戏。有这么一句话，“游戏引擎就像汽车的引擎，决定着游戏的速度、真实感、吸引力等，玩家所体验到的剧情、关卡、美工、音乐、操作等内容都是由游戏的引擎直接控制的，它扮演着发动机的角色，把游戏中的所有元素捆绑起来，在后台指挥它的同时，有序地进行工作” [13]。

一个优秀的三维图形引擎需要提供动态灯光、曲面细分、体雾、镜面、天空盒、阴影、粒子系统、网格动画等功能。按照三维引擎所具有的功能，可以把三维引擎划分为几个功能模块：系统模块、底层渲染模块、控制台模块、数据存储模块、游戏接口模块与游戏插件模块。这些模块构成了三维引擎所具有的功能 [14]。

如图 3，三维应用程序可以调用三维引擎的函数接口来创建不同的场景。而三维引擎则封装了底层的图形应用程序接口，如 OpenGL 或 DirectX [15]。三维引擎实现的是各种高级的图形功能，其中主要由控制模块来实现各个模块之间的调用，包括对场景的渲染，程序的配置，用户的输入，场景的音效等等。OpenGL 和 DirectX 在工作时，会进一步调用底层的显示硬件，来完成渲染工作。三维引擎中各个模块之间的组织关系，会影响到引擎的性能，所以在引擎开发中要处理好各个模块之间的组织关系。

#### 3.1. 系统模块

系统模块是三维引擎中与计算机，准确来说是机器本身通讯的部分。判断一个引擎是否优秀，则要看其系统模块能否很好地进行不同平台的移植。一个优秀的三维引擎在进行平台移植时，它的系统模块

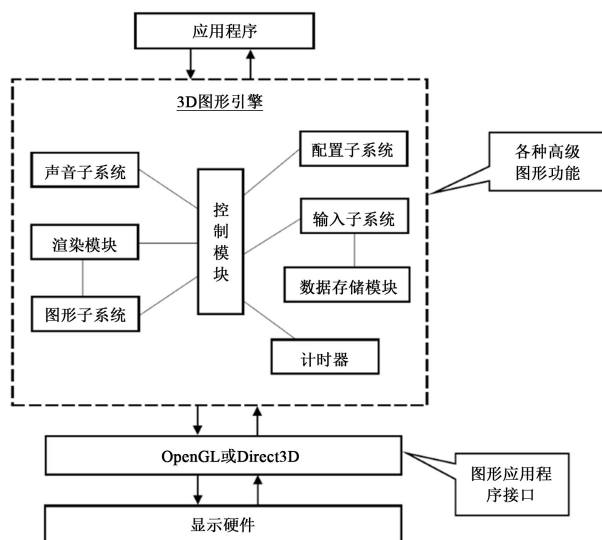


Figure 3. The structure of 3D engine

图 3. 三维引擎的架构[4]

则是唯一需要做主要更改的地方[16]。我们可以进一步将系统模块分为若干个子系统模块，分别是：图形子系统、输入子系统、声音子系统、时间子系统以及配置子系统。主系统模块负责初始化、更新以及关闭所有子系统[14]。

### 3.1.1. 图形子系统

图形子系统主要负责在屏幕上画点、线等图形操作，表现非常直观。图形子系统一般都是利用 OpenGL、Direct 3D、Java3D、Glide 等图形库来实现。每种图形库都有独自的一套 API 函数，并抽象出一个“图形层”置于实现的 API 函数之上，这样就方便了开发人员，提高了程序的开发效率[17]。

### 3.1.2. 输入子系统

输入子系统用来把各种不同的输入装置(键盘、鼠标、游戏摇杆等)的输入触发做成统一的控制接收处理，等同于将这些输入统一起来以允许控制的抽象化。通过输入子系统，开发人员不必一一检测每种输入设备，只需要向输入子系统发送请求以获取输入信息即可。玩家也可以方便地切换不同的输入设备[18]。

### 3.1.3. 声音子系统

声音子系统负责载入和播放声音等操作。目前大部分的游戏都支持 3D 声音，这让实现变得稍微复杂。

### 3.1.4. 时间子系统

时间子系统负责三维引擎中各部分组件的时间触发。引擎中很多功能包括对象等，都是基于时间进行的，一个优秀合理的时间子系统可以避免在开发过程中撰写许多相似的代码。因此在时间子系统中必须实现一些时间管理功能的代码[19]。

### 3.1.5. 配置子系统

配置子系统实际位于上述子系统之上。它负责读取配置文件、命令行参数、读取游戏进度、保存游戏进度、设置游戏参数等等。其他子系统在初始化和运行的时候都会向配置子系统查询相应的配置，并一直与配置子系统保持通信。配置子系统让整个引擎变得可配置化，这为调试和测试带来了很大的方便，并让玩家能更简单地按照自己喜欢的方式改变游戏设置，包括改变分辨率、键盘设置、声音设置等等。

## 3.2. 底层渲染模块

一款三维游戏真正吸引玩家的正是它逼真虚幻的场景画面和人物造型, 以及灯光照明、阴影效果、粒子系统特效等等, 这就需要一个高效的渲染引擎来完成。底层渲染模块是三维引擎最重要最核心的部分之一, 渲染模块最重要的就是提供基本的成像功能并且简洁干净, 这部分一般都是由 OpenGL 或 Direct 3D 图形库来实现[14]。

模块按照功能可以进一步细分为: 可见裁剪、摄影机、静态几何图形、动态几何图形、粒子系统、布告板、网格、天空盒、光照、雾化、顶点光影、输出等部分。其中每个部分应该各有一个界面, 并通过这个界面来改变设置、位置、方向、以及其他可能与系统相关的属性配置[29]。

目前底层渲染模块的各部分都有一些相应的成熟算法帮助开发者实现功能。这些算法都是前人通过实验测试得出来的, 运用好相应的算法, 可以避免三维引擎中出现大量的冗余代码[21]。

### 3.2.1. 物理系统

三维引擎还有一个重要的功能就是提供基本的物理系统, 即“控制物体运动模式的一套规则”。物理系统是渲染模块的一部分, 因为涉及到碰撞检测(collision detection)和碰撞反应(collision response)等功能[22]。在 3D 游戏的相关技术中, 首先要考虑的就是如何构建物体的实体模型以及如何真实地把它们呈现倒显示器上。一旦将这些技术应用到真实世界的仿真时, 我们还需要考虑自然法则的仿真, 如碰撞检测和碰撞反应。碰撞检测是物理系统的核心部分, 它可以探测游戏中各物体的物理边缘[23]。当两个 3D 物体撞在一起的时候, 这种技术可以防止他们互相穿过, 这就保证了当你撞在墙上的时候不会穿墙而过, 也不会把墙撞倒。

对于碰撞检测, 有很多优秀的算法已经实现出来。如基于 BSP 引擎的碰撞检测、基于 Voronoi 区域的碰撞检测方法等等[24]。目前有很多学者正在完善和研究更多更好的碰撞检测方法。

## 3.3. 控制台模块

控制台模块负责改变引擎和游戏的设置, 方便调试和测试。通过控制台的变量和函数, 可以在不重新激活游戏的情况下, 改变相应的设置[25]。该模块在游戏调试阶段具有非常重要的意义, 很多时间都需要测试一系列变量的值, 把这些值输出到控制台要比直接运行一个调试程序要方便快捷得多。如果在引擎运行期间发现一个错误, 可以不必退出程序, 通过控制台即可做出相应控制, 并打印出错误信息。另外, 在游戏正式运行的时候, 也可以通过控制台将调试信息隐藏屏蔽掉, 不必让玩家看到。

## 3.4. 数据存储模块

数据存储模块会被引擎的大部分组件所使用。它定义了游戏中所使用的数据格式以及他们的组织方式。该模块也包含了相应的数据读取和保存功能、以及数学程序代码(向量、平面、矩阵等等)、数据内存管理等等[26]。在游戏中会出现大量的数据结构的类库, 这些类库之间必然存在数据的传递和保存, 如何处理好这些数据的组织, 是该模块需要考虑的。

## 3.5. 游戏接口模块

游戏接口模块主要提供一个接口层, 方便游戏开发者使用三维引擎的各部分的功能, 这使得游戏开发变得简便快捷。游戏引擎中的每一部分都有动态的属性, 而该模块则提供了一个接口去修改这些动态的属性值。其中包含了摄像机、模型属性、光照、粒子系统、声音播放、输入处理、关卡变换、碰撞检测和相应、模型贴图的现实等等[27] [28]。一般该模块会以图形界面的形式呈现给游戏开发人员。游戏接口模块让引擎和游戏制作很好的分离开, 使代码设计变得更加清晰, 应用起来也会更方便, 它能使三维

引擎具有非常好的重用性。

### 3.6. 游戏插件模块

游戏插件模块负责提供一些插件供三维引擎与其他应用程序生成的内容进行融合, 如模型转换程序或插件等。每一个三维图形引擎都希望能制造出精美的虚拟场景和逼真的人物造型, 但三维引擎的主要任务不是场景和对象模型的制作, 即使提供了三维模型编辑器也是有限的, 这就需要编写一些模型转换程序或插件, 将其他专业的三维建模工具(3D Max、Maya 等)设计出的模型转换为该引擎所使用的格式。

## 4. 三维图形引擎绘制原理

### 4.1. 三维图形渲染流水线

对绘制过程的一种理解就是把它当作一系列的变换过程, 这些变换把对象从被编码的坐标系带入显示表面的坐标系[29]。这个过程有时被认为是渲染流水线, 被描述为一系列空间。对象通过这些空间走过了从数据库到显示屏幕的迁移之路。现有的基础三维图形绘制库, 如 OpenGL, Direct 3D 等, 都采用了传统的图形渲染流水线。这种基本的流水线设计在计算机图形学发展的早期就已经出现, 随着硬件技术不断发展而不断被增强和扩展, 但基本的原理至今却未改变。

### 4.2. 三维场景绘制过程

使用 OpenGL、Direct3D 等三维 API (即基础三维图形绘制库)绘制三维场景的整个流程如图 4:

下面将会简要介绍各部分的功能[29]。

#### (1) 数据预处理

数据预处理包括: 将建模完成后得到的场景中物体的几何模型数据转换成三维 API 可直接接受的基本图元的形式, 如点、线、面; 对影像数据如纹理图像进行预处理, 这种处理包括图像格式的转换、图像质量的改善、影响金字塔的生成等等。

#### (2) 参数设置

在对三维场景进行渲染前, 需要设置相关的场景参数值。这些参数一般包括光源性质(反射光、漫反射光、环境光)、光源位置(距离和方向)、明暗处理方式(平面处理或平滑处理)、纹理映射方式等。除此之外, 还需要设置视点位置和视线方向。

#### (3) 纹理映射

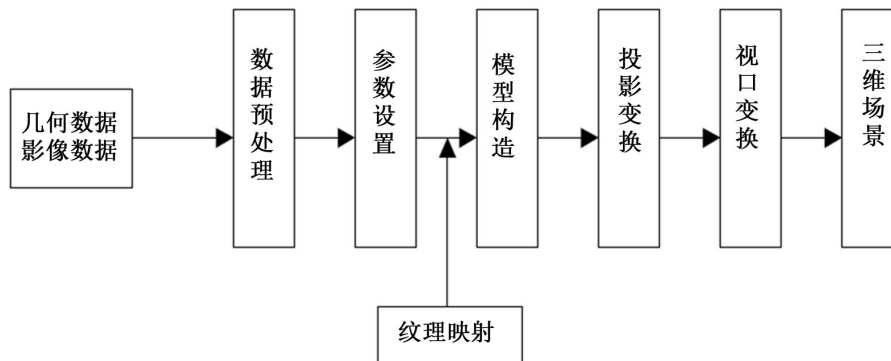


Figure4. The drawing flow chart of 3D scene

图 4. 三维场景绘制流程[29]

纹理映射的目的是把纹理图片粘贴到模型表面, 从而获得有真实感效果的场景。纹理映射在这里是指对纹理的定义和应用方式的指定。

#### (4) 模型构造

模型构造就是对整个场景模型进行几何重建, 即根据三维 API 中的有关命令函数将模型数据“装配”成场景模型。

#### (5) 投影变换

投影变换是将一种地图投影点的坐标变换为另一种地图投影点坐标的过程, 是生成三维场景的重要基础和关键步骤。投影变换一般分为透视投影变换和正射投影变换, 三维 API 中应用的一般都是透视投影变换。

#### (6) 视口变换

视口是指计算机屏幕中的矩形绘图区域。视口变换的目的就是将三维空间坐标映射为计算机屏幕上的二维屏幕坐标。

整个流程的最终结果就是在计算机屏幕上显示出的整个三维场景的二维图像。

## 5. 三维图形引擎未来趋势

经过多年的发展, 三维图形引擎技术取得了重要进展。特别是一些成功的案例, 包括游戏、商业虚拟系统等, 都让我们相信三维图形引擎在虚拟现实独特的、无可替代的地位和作用[30]。

目前, 三维图形引擎的主要应用市场仍是游戏开发。各个游戏公司都在开发并完善现有的游戏引擎。当一个游戏引擎框架基本确定之后, 未来的游戏竞争就是游戏可玩性的竞争。游戏的可玩性离不开人工智能, 如何让虚拟人物能和人类一样思考一直是人来研究的课题。包括通过人工智能技术对其他领域进行现实模拟, 比如人脸识别、神经网络、遗传算法等等。

随着科技的不断发展, 三维图形引擎正不断应用与各个领域的虚拟现实系统中, 如医疗卫生的虚拟手术训练、远程会诊、远程协作手术、虚拟手术规划等; 军事领域的军事仿真训练与演练、飞行模拟器等; 工业领域的产品设计、虚拟样机等; 教育文化领域的数字博物馆、虚拟展馆、仿真彩排等等。这些领域的虚拟现实系统仍然需要高效的三维图形引擎做支撑, 设计开发出更加生动逼真的虚拟环境供用户体验。

现有的三维图形引擎大部分都是基于 PC 端的引擎, 随着移动互联网的日益发展, 移动设备的不断更替, 基于移动设备的三维图形引擎的发展也将变得势在必行。如何在有限的硬件资源内, 在移动设备上呈现出生动逼真的虚拟场景, 将会是一个很大的挑战。

### 5.1. 需要进一步研究的问题

虽然三维图形引擎技术取得了很大的进展, 但仍然存在一系列的根本性理论问题和关键技术问题, 这些问题的突破将会使三维图形引擎技术得到更大的发展。

#### 问题 1: 复杂虚拟环境的构建与大数据的管理

目前虚拟环境的构造方法主要有数学建模和复制现实两种。虽然三维引擎不一定需要构建模型, 因为有专门的建模工具, 如 Maya, 3D MAX 等, 但是当需要渲染的物体的数量很多时, 如何保证虚拟环境的复杂性和逼真性, 这是值得继续研究的问题。另外, 复杂的虚拟环境拥有比较集中或分布存储的大量数据, 这些数据的有效调度、检索和维护是具有挑战性的问题。

#### 问题 2: 引擎的跨平台性

目前的三维引擎大都是应用于 PC 端的应用程序, 随着互联网和移动互联网的发展, 如何在网上高

效快捷地展现三维虚拟环境是值得探讨的问题。随着移动终端的普及, 基于移动终端的三维应用程序或者游戏也有待开发研究。如何开发一款引擎, 即满足 PC 端的要求, 又能满足移动终端的需要, 也是需要我们研究的问题。这可能需要良好的硬件环境支持, 以及显示技术的支撑。

### 问题 3: 渲染模型的可信度、复杂度

三维引擎的主要工作是渲染, 但是否所有渲染的物体的可信度都很高? 是否现实世界中的所有物体都可以模型化, 或者数字化建模? 这与可计算性与模型的复杂性有关。如真实海浪与气象的建模是很难的, 但我们可以渲染出与真实海浪相似的简化海浪, 我们可以模拟出与现实气象相似的简化气象。问题在于这简化后的模型的可信度是否仍然很高。可建模性、模型复杂性、模型简化与模型可信度分析等等, 都是我们需要继续学习研究的问题。

### 问题 4: 虚实景物的融合

虚拟景物和真实景物的融合, 包括在动、静态真实场景画面中融入虚拟景物和在虚拟环境中融入动、静态真实景物。这涉及到数据融合、模型融合等问题, 以及虚实景物的准确三维注册、虚实景物的空间遮挡关系、虚实光照效果的融合处理等关键技术。这些知识和技术都是需要我们去研究和突破的。

### 问题 5: 三维引擎性能评价标准的制定

目前判断一款三维引擎性能高低的方法就是看这款三维引擎渲染功能是否强大。但缺少一个统一的标准来衡量一款三维引擎的性能好坏。再加上各商业引擎的保密性, 针对不同游戏而设计的功能框架, 这使得在性能评价上更难统一。各个引擎都有它们各自的优点, 也有缺点。但针对三维引擎的各种性能, 如逼真性、交互性、实时性, 特别是虚拟场景中各个物体的可信度等, 建立合理的评价指标, 确定合适的评价标准, 是推动三维引擎技术进步的重要因素。目前这方面的研究相对较少, 本身也具有难度, 大多数性能难以定量描述。

## 6. 结束语

三维图形引擎, 或者说游戏引擎的发展代表了计算机硬件和软件开发的最新技术, 驱动型、完整性和独立性是其主要特点。三维图形引擎需要解决场景构造、对象处理、场景渲染、事件处理碰撞检测等问题。它其实就是把一个三维应用程序中可以重复利用的部分, 以高效的方式组织起来, 使其规格化和最佳化, 方便代码重用。

开发一款三维应用程序(游戏)需要耗费大量的人力物力, 因此开发三维图形引擎的目的就是为了简化开发的复杂度, 缩短开发时间, 提高开发效率, 降低开发成本。

本文通过对三维图形引擎的论述, 积累相关的理论知识, 希望对从事三维应用程序(游戏)的人员能够有一定的参考和帮助。本文下一步的研究工作将深入研究三维图形引擎的关键技术以及图形引擎的最新技术。

## 致 谢

首先很感谢我的导师, 孙伟老师。在本科以及研究生阶段, 孙伟老师不仅让我参与了实验室项目的开发, 在项目中锻炼自己, 而且还时常告诫我们要培养自己提出问题, 分析问题以及解决问题的能力。在日常学习生活中, 更以他严谨的治学态度教我们如何做人, 如何做学问。在此, 真心感谢孙伟老师。

其次, 我要感谢实验室的老师, 师兄师姐, 他们无论在学习上, 还是生活中, 都给了我很大的帮助, 他们都是我的良师益友。

最后, 我也要感谢其他文献的作者, 他们的文献资料对我有很大帮助。虚拟现实的研究在我国还不是特别优秀, 这正需要我们共同努力, 把我国虚拟现实的技术发展到更高的层次。

## 参考文献 (References)

- [1] 卞锋, 江漫清, 桑永英 (2007) 虚拟现实及其应用进展. *计算机仿真*, **06**, 1-4.
- [2] 赵沁平 (2009) 虚拟现实综述. *中国科学*, **1**, 2-46.
- [3] Shiratuddin, M.F. and Thabet, W. (2002) Virtual office walkthrough using a 3D game engine: Special issue on designing virtual worlds. *International Journal of Design Computing*, **4**, 1329-7147.
- [4] 张继开 (2004) 三维图形引擎技术的研究. 北方工业大学, 北京, 1-58.
- [5] 刘洋 (2006) 三维引擎技术研究. 郑州大学, 郑州, 1-65.
- [6] 左鲁梅 (2004) 三维图形引擎中的关键技术研究. 北方工业大学, 北京, 1-63.
- [7] 孙正 (2010) 三维图形引擎大规模场景实时渲染技术研究与应用. 电子科技大学, 武汉, 1-88.
- [8] Gregory, J. (2009) *Game engine architecture*. A K Peters, Natick, Massachusetts.
- [9] Eberly, D.H. (2001) *3D game engine design: A practical approach to real-time computer graphics*. Morgan Kaufmann, San Francisco.
- [10] Liu, Y.J., Chen, W.B. and He, X.M. (2010) 3D graphics engine technology research and implementation. *IEEE*, 697-700.
- [11] 樊翠, 王丽芳 (2006) 基于 D3D 的三维游戏引擎的设计与实现. *科学技术与工程*, **10**, 1431-1435.
- [12] 管琛琛, 李光, 赵富宽, 等 (2014) 基于 OpenGL 的三维可视化以及交互裁剪应用. *计算机应用*, **S2**, 240-242.
- [13] 万晓龙 (2013) 基于 Flash 的三维引擎的研究与实现. 硕士论文, 中山大学, 广州, 1-55.
- [14] 夏旺盛, 黄心渊 (2003) 3D 游戏引擎架构概述. *现代计算机*, **165**, 74-76.
- [15] 黄进 (2010) 基于 XNA 的游戏引擎架构设计. 2010 *Asia-Pacific Conference on Information Network and Digital Content Security (2010APCID)*.
- [16] 邱航, 陈雷霆 (2010) 面向对象三维图形引擎的设计与实现. *电子科技大学学报*, **1**, 123-127.
- [17] 李胜亮 (2007) 三维图形引擎关键技术研究. 硕士论文, 西北工业大学, 西安, 1-47.
- [18] 陈国军, 叶飞跃, 杨黎明 (2001) 三维图形生成支撑环境综述. *计算机应用研究*, **5**, 1-3.
- [19] 高兴, 郑智, 全宇 (2013) 三维游戏引擎的研究与设计. *计算机系统应用*, **8**, 29-33.
- [20] 靳柯 (2011) 图形引擎若干重要技术的研究. 硕士论文, 吉林大学, 长春, 1-47.
- [21] 李辉, 吴建国, 游志胜 (2001) 基于微机的面向对象三维图形引擎. *计算机工程*, **4**, 169-171.
- [22] 彭群生, 鲍虎军, 金小刚 (1999) *计算机真实感图形的算法基础*. 科学出版社, 北京.
- [23] 赵辉, 余云宜, 梁应宏, 等 (1999) 模拟训练视景系统、三维游戏开发工具 RGB Game Builder 的设计. *中国图像图形学报*, **1**, 61-67.
- [24] 刘祎玮, 张引, 叶修梓 (2006) 3D 游戏引擎渲染内核架构及其技术. *计算机应用研究*, **8**, 45-48.
- [25] 赵沁平, 郝爱民, 王莉莉 (2006) 实时三维图形平台 BH-Graph. *计算机研究与发展*, **9**, 1491-1497.
- [26] 罗冠, 郝重阳, 淮永建, 等 (2001) 虚拟现实引擎的设计与实现. *计算机学报*, **11**, 1263-1269.
- [27] Eberly, D.H. (2006) *3D game engine design*. Second Edition, Morgan Kaufmann, San Francisco.
- [28] Hook, B. (1995) *Building a 3D game engine in C++*. Wiley, Chichester.
- [29] 尚建嘎, 吴丹, 杨志强 (2002) 基于微机的三维图形引擎体系结构研究. *现代计算机*, **135**, 47-50.
- [30] 罗玉华, 左军, 李岩 (2010) 虚拟现实技术及其应用. *科技信息*, **1**, 63-64.