

6-6-2020

Development Trend of Next Generation 3D Graphics Engines

Zhao Wei

Aviation Institute of Naval Academy, Shanghai 200436, China;

Mao Ping

Aviation Institute of Naval Academy, Shanghai 200436, China;

Fanyu Shen

Aviation Institute of Naval Academy, Shanghai 200436, China;

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the [Artificial Intelligence and Robotics Commons](#), [Computer Engineering Commons](#), [Numerical Analysis and Scientific Computing Commons](#), [Operations Research](#), [Systems Engineering and Industrial Engineering Commons](#), and the [Systems Science Commons](#)

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

Development Trend of Next Generation 3D Graphics Engines

Abstract

Abstract: The development, relationship and differences between game engine and virtual reality engine were introduced, a joint name of 3D graphics engine was proposed. The bottlenecks and shortages of scene management, geometry model creating, physics simulation, lighting, shading, etc. were analyzed by synthetically considering the advancements in computer hardware, softwares and simulation theories, etc.. The development demands and trends were researched, *the Concept of adaptive multi-thread shading technology, advanced real-time dynamic deformation technology, integrated collision detection technology, independent of the collision process technology were proposed for the first time, and a brief features description of these concepts were give*, it can provided reference for developing the next generation 3D Graphics Engine.

Keywords

graphics engine, 3D simulation, virtual reality, game engine

Recommended Citation

Zhao Wei, Mao Ping, Shen Fanyu. Development Trend of Next Generation 3D Graphics Engines[J]. Journal of System Simulation, 2017, 29(12): 2935-2944.

下一代三维图形引擎发展趋势研究

赵维, 茅坪, 沈凡宇

(海军研究院空中所, 上海 200436)

摘要: 介绍了游戏引擎和虚拟现实引擎的发展现状、相互关系及区别, 将两种引擎统称为三维图形引擎。综合考虑计算机硬件、软件、仿真理论等的进步, 分析场景管理、几何模型构建、物理学仿真、光照和着色等技术目前存在的瓶颈及不足, 研究这些技术的后续发展需求及发展趋势; 提出了自适应多线程渲染技术、高级实时动态变形技术、一体化碰撞检测技术、自主化碰撞处理技术概念, 并对这些概念的特征进行了简单描述, 可为下一代三维图形引擎的开发提供参考和借鉴。

关键词: 图形引擎; 三维仿真; 虚拟现实; 游戏引擎

中图分类号: TP391.77

文献标识码: B

文章编号: 1004-731X (2017) 12-2935-10

DOI: 10.16182/j.issn1004731x.joss.201712001

Development Trend of the Next Generation 3D Graphics Engines

Zhao Wei, Mao Ping, Shen Fanyu

(Aviation Institute of Naval Academy, Shanghai 200436, China)

Abstract: The development, relationship and differences between game engine and virtual reality engine were introduced, a joint name of 3D graphics engine was proposed. The bottlenecks and shortages of scene management, geometry model creating, physics simulation, lighting, shading, etc. were analyzed by synthetically considering the advancements in computer hardware, softwares and simulation theories, etc.. The development demands and trends were researched, the Concept of adaptive multi-thread shading technology, advanced real-time dynamic deformation technology, integrated collision detection technology, independent of the collision process technology were proposed for the first time, and a brief features description of these concepts were give, it can provided reference for developing the next generation 3D Graphics Engine.

Keywords: graphics engine; 3D simulation; virtual reality; game engine

引言

在三维计算机游戏和基于虚拟现实技术的三维仿真中, 都需要根据各类数据描述模型控制计算机硬件生成复杂的图形画面, 其底层一般都基于 OpenGL 或 D3DX 驱动图形硬件实现, 但直接使

用 OpenGL 或 D3DX 进行游戏和三维仿真的开发工作量巨大、代码可重用度低, 部分公司为了提高项目的开发效率, 将一些常用、通用、可重复的代码进行封装, 逐步出现了大量的游戏引擎和虚拟现实引擎。这两类引擎都是用于完成对现实或虚拟世界的模拟仿真, 且最终都采用三维图形可视化手段展示仿真结果, 因此本文将这两种引擎统称为三维图形引擎, 三维图形引擎通过将很多可重复使用的功能进行封装, 仿真应用开发人员只需进行简单设置和编程, 即可调用相应函数快速实现同样的功能, 从而极大地降低仿真项目的开发难度、节省开



收稿日期: 2015-10-23

修回日期: 2016-01-23;

作者简介: 赵维(1981-), 男, 湖南长沙, 博士, 工程师, 研究方向为航空综合保障、虚拟现实技术、航空救生、计算机测控技术; 茅坪(1963-), 男, 浙江杭州, 高工, 博导, 研究方向为模拟器设计, 军用仿真等。

<http://www.china-simulation.com>

• 2935 •

发时间。三维图形引擎在军用、民用仿真领域都有非常广泛的应用,如游戏、大型军事仿真、飞行仿真、城市规划、虚拟展览系统等^[1]。

1 发展现状

目前游戏引擎与虚拟现实引擎还有比较明显的区别,虚拟现实引擎属于专业的图形渲染引擎,属于行业软件,它主要关注仿真效率,在大规模、超大规模仿真系统的图形渲染效率上有明显优势,在军事仿真(如飞行仿真、战场对抗、态势实时显示)、城市规划等领域应用广泛,游戏引擎很难胜任如此大规模的图形渲染任务,但虚拟现实引擎一般没有内置物理学仿真、人物角色仿真、人工智能、网络连接等模块,需要 Vortex、PhysX、DI-GUY、HLA 等第三方插件提供支持。而游戏引擎一般用于小规模仿真,它的功能小而全,主要关注仿真的视觉效果,一般内置了物理仿真、骨骼动画、人工智能、网络通信等模块,且其物理仿真、骨骼动画等方面效果和技术都非常先进,但其在超大规模仿真等方面与虚拟现实引擎相比还有较大差距。其实游戏引擎和虚拟现实引擎在基本原理和底层实现上是相同的,但受限于当前软、硬件水平,只能各有侧重点,近年来这两种引擎出现了相互融合的趋势^[2-3]。

目前三维图形引擎领域没有统一的行业标准,各引擎开发单位对图形引擎的功能模块、程序架构、扩展接口、模型格式、底层实现等的理解和定义各不相同,造成几何、高程、物理等模型的通用化程度低,相互移植困难;且每个引擎都有各自的侧重点,造成通用化引擎部分功能不足,专业化引擎应用范围受限,开源引擎稳定性、可靠性不高,商业引擎二次开发能力不足等问题,让广大游戏和虚拟现实开发人员在选择图形引擎时无所适从,只能根据项目特点频繁更换合适的图形引擎,大幅增加了项目开发的难度和工作量。总体来说,目前各类图形引擎普遍的问题是场景管理效率仍有待提高,模型精度及规模受限,物理学仿真模型不完善,光照和着色模型不够真实,光影处理效率低下等^[4]。

本文根据目前软、硬件技术和仿真实理论的最新发展成果,对在下一代三维图形引擎最有可能发生重大进步或革新的技术进行研究,期望起一个抛砖引玉的作用,更好地探索其未来发展方向^[1-2]。

2 场景管理技术

2.1 大规模动态场景高效管理技术

场景组织管理按其目的可分为面向交互与面向性能 2 类,前者主要采用场景图来描述和组织虚拟场景,场景图可以方便地进行更新修改,但进行碰撞检测、可见性剔除的效率较低;后者主要采用空间分割技术作为组织方式,包括二叉树、四叉树、八叉树等方法,其优点是可见性剔除和碰撞检测效率高,但其数据结构的建立和更新开销大,在处理动态或交互性要求高的虚拟场景时效率相对较差。为高效完成碰撞检测和可见性剔除工作,目前图形引擎大多采用面向性能的场景组织方式,下一代图形引擎在场景组织方式上可能很难有大的突破,但应研究新的动态场景组织管理策略,改进场景数据存储方式,优化场景图和空间结构划分技术,进一步提高场景数据结构的建立和更新效率,从而实现大规模动态场景的高效管理^[4-6]。

2.2 自适应多线程技术

目前图形引擎一般只是将“剔除”、“绘制”等任务作为单独的线程,整个图形渲染流程一般仅采用 1~3 个线程进行处理,目前主流 PC 机的 CPU 一般都具有 8 个以上计算核心,而中档图形工作站一般有几十个 CPU,线程过少将不能充分发挥多处理器、多计算核心的优势,因此下一代仿真引擎必须进一步分解仿真过程涉及的计算任务,根据 CPU 计算核心数目合理分配计算负载,自动启动与计算核心数目相匹配的线程,并将部分计算转移到 GPU 上进行,达到充分利用硬件资源、提高渲染效率的目的^[1,7]。

2.3 支持更大内存寻址空间

当前中档的图形工作站一般都配备 64GB 甚

至 128 GB 以上的内存, 32 位三维图形引擎开发的虚拟仿真系统只能利用不大于 4 GB 的内存, 不能充分发挥硬件进步带来的性能提升优势。64 位操作系统可以支持 2^{64} 的内存寻址空间, 最大支持的内存容量远超目前 32 位操作系统, 可以获得更高的性能, 但当前绝大多数用户仍使用 32 位操作系统, 因此即使有少数三维图形引擎已推出 64 位版本, 但因用户很少无法成为主流。随着各大软件厂商逐步转入 64 位程序的开发, 64 位操作系统应用生态将更完善, 届时 64 位的图形引擎也将逐步普及。

2.4 高级分布式渲染技术

单台计算机即使拥有多 CPU、多核心, 其计算能力仍有限, 分布式渲染才是解决大规模仿真渲染效率的最佳方案, 目前少数仿真引擎如 Mantis 已支持一定形式的分布式渲染技术, 但仍处于初期阶段, 存在一定的缺陷和不足, 下一代三维图形引擎的分布式渲染技术可以在网络通讯、任务分配调度、数据传输协议等方面进行革新, 从而提高超大规模仿真场景的渲染效率^[8-9]。

3 几何模型构建技术

3.1 超大规模几何元素支持

几何模型是三维仿真的基础, 直接决定仿真的真实性。受当前硬件性能的制约, 整个仿真中允许的多边形和顶点等几何元素数目有限, 因此目前实时三维仿真一般使用低精度模型, 得到的画面效果真实性与电影、广告等非实时渲染的画面效果相差甚远。电影属于非实时渲染, 画面渲染完成后将不再改变, 可以花费更多的渲染时间而得到更佳的视觉效果, 在电影中大都采用高精度模型, 如一架飞机的三角形数目可达到几百万个以上, 电影渲染计算机系统大都采用最高档的图形工作站, 但单帧画面的渲染时间一般仍需要几个小时甚至几天; 三维仿真和游戏需要根据操作者给出的指令实时生成场景画面, 且要求帧率不低于 24, 目前只能通过降低模型精度的方式来提高渲染效率和仿真帧率,

如一架飞机的三角形数目一般在 1 万个以下, 这样才能保证实时渲染帧率不影响操作者的视觉感受。

随着计算机硬件性能的革命性进步, 显卡 GPU、显存、计算能力等将有质的飞跃, 可以提供更多的计算机资源来加载和渲染高精度几何模型, 预计下一代三维图形引擎将可以提供超大规模的几何元素支持, 达到与电影相媲美的视觉效果^[10-12]。

3.2 高效的曲线曲面仿真

为了提高图形的渲染效率, 目前三维图形引擎大都采用多边形建立几何模型, 很少内置支持曲线曲面的仿真, 多边形模型在载入场景前已建好, 引擎只需读取模型中存储的顶点位置、线型、颜色、纹理等参数数据, 并根据该几何模型的格式定义构建正确的三维模型并载入场景即可, 加载速度较快, 易于控制模型精度, 便于对仿真规模和仿真效率进行预测和管理, 而且显卡等硬件也专门针对多边形渲染进行了特别优化。但采用多边形构建几何模型也有明显的缺陷, 部分简单的曲面如球体, 一般需要几千个三角形才能得到较平滑的表面, 造成系统负担较重。如果图形引擎支持曲线、曲面建模, 球体模型只需 1 个函数及简单的几个参数就可以非常精确的描述。曲线、曲面建模在三维电影、广告中应用非常广泛, 所建模型更加平滑自然, 但加载时需要根据曲线、曲面的参数, 实时构成三维模型, 增加了顶点位置、面片等数据的实时生成过程, 造成仿真效率明显下降, 因此曲线、曲面建模目前主要用于电影、广告等非实时渲染领域。随着硬件性能的不不断提升, 硬件资源将不再成为一个问题, 考虑到曲线、曲线仿真可以提供更加逼真的视觉效果、更加高效的建模手段, 预计下一代图形引擎将对曲线、曲面仿真提供更完美的解决方案^[13]。

3.3 高级实时动态变形技术

目前图形引擎大都仅支持几何模型实时简单变形, 较有代表性的是虚拟人仿真, 虚拟人仿真使用的技术有关节动画、渐变动画、骨骼动画等, 但这些动态变形仅能满足最基本的变形需求, 其变形

是可以预知的,在仿真开始前就已经建立固化的动作库,仿真中只是进行调用,很难实时完成其他不可预知的变形^[14-15]。除虚拟人仿真外,还有碰撞处理、软件动力学、流体仿真、布料仿真、植被等都需要用到动态变形技术,这些仿真首先需要进行动力学计算,获取计算结果后,采用动态变形技术将计算结果用图形可视化方式表现出来,这些动态变形的多边形运动没有规律可循,且需要根据受力情况提供不同的动态表现形式,因此其仿真难度很大,目前一般只在非实时渲染中进行。随着计算机硬件性能的提升,以及 GPU 编程技术的发展,可以在 GPU 的顶点处理器中对多边形顶点进行更复杂的控制,且相关算法日趋完善,下一代三维图形引擎有望支持更多的动态变形技术。图 1 为新版 Vortex 软件自带实例实现的动态钢索仿真效果,从图中可看出钢索外形随着受力情况的改变而产生的动态变化^[3]。

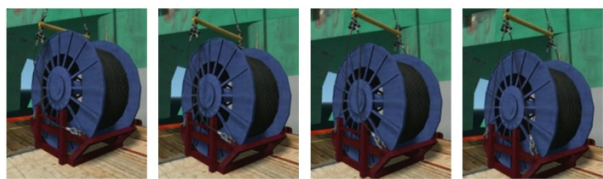


图 1 Vortex 实现的动态钢索仿真效果
Fig.1 Dynamic cable simulation developed by Vortex

3.4 先进纹理贴图技术

现有图形引擎主要使用二维纹理贴图,通过将二维纹理贴在三维模型的多边形上来提高模型真实度,且同一个多边形可以支持多层纹理,通过各层纹理的融合实现较复杂的效果。但常规二维纹理很难表现材料质地和局部的高低,光线从不同角度照射基本没有区别。为改善二维纹理贴图的效果,出现了凹凸贴图、环境贴图、视频纹理、双向纹理等新概念二维纹理,凹凸贴图用于表现诸如砖墙、轮胎花纹等较小的凹凸不平的细节效果,可以在不增加多边形数目的情况下提高模型的精细度;视频纹理通过将视频中的二维帧图像提取出来并贴在多边形表面,可以为虚拟场景中的虚拟电视机、广

告牌、喷泉、瀑布等提供动态模拟。但这些新概念纹理并没有从根本上解决二维纹理的不足^[16-18]。

三维纹理是纹理技术的革新,它内部包括整个体积的像素点,每个点在纹理内部空间中有三维的相对坐标,不再需要将图案贴在表面,而是将三维纹理部分地取舍和扭曲,把它变成最终物体的形状,但使用三维纹理需要消耗非常惊人的系统资源,其耗费的系统资源将随着纹理的分辨率呈立方级数增加。比如二维纹理目前最大分辨率可以达到 4096×4096 ;而三维纹理分辨率暂时只能达到 $256 \times 256 \times 256$ 的水平,即便如此低的分辨率就有 16 M 个体素点,如采用 32 位的色彩精度,需要 64 MB 的显存空间,目前主流 PC 机的显存一般在 512 MB 以内,很难支持在仿真中大规模使用三维纹理。除显存空间限制外,对三维纹理数据进行过滤、半透明混合和几何变形所需计算资源也非常惊人。但随着计算机硬件及三维纹理数据处理技术的进步,图形引擎有望逐步解决大规模应用三维纹理所面临的难题,为复杂物体的建模提供更有效的手段^[19-20]。

4 物理学仿真技术

物理学仿真主要涉及碰撞检测、碰撞处理和动力学仿真 3 个部分。

4.1 碰撞检测

碰撞检测用于检测对象之间是否发生相互作用,只有检测到发生了碰撞,才能进行后续的动力学仿真及碰撞处理。

物理学仿真包括刚体、软体和流体的仿真,其难点包括动力学建模和工程实现两个方面,动力学仿真目前已有较大突破,出现了较多专业的刚体、软体和流体仿真软件,但这些软件只能在其自身构建的简单物理环境中运行,碰撞检测需要进行大量的相交测试,虚拟仿真系统或游戏中一般存在大量的仿真对象,如果全部纳入同一个物理环境,目前计算机硬件根本无法承受,因此刚体、软体或流体仿真工具一般都是作为插件为图形引擎提供支持,

并构建各自独立的物理世界, 仿真时只将需要进行碰撞检测的对象加入该物理世界, 比如在使用 Vortex 进行刚体仿真时, 参与刚体仿真的对象需要根据自己的外形设定一个碰撞检测包围盒, 用包围盒代表该对象加入 Vortex 构建的刚体物理世界, 并用包围盒代表该物体与刚体物理世界中其他对象进行碰撞检测; 而如果该仿真同时使用 Physx 进行软体仿真, 也需要建立独立的软体物理世界和相关包围盒, 目前各个物理世界的包围盒并不能共用, 这样整个仿真系统就存在引擎本身、刚体和软体 3 个不同的物理世界, 极大地增加了仿真系统的复杂度。目前刚体、软体、流体等专业仿真软件一般都属于不同公司的产品, 集成度、通用度低, 下一代三维图形引擎应将相关产品进行整合, 做到各个产品的物理世界对象可以通用, 这样可以大大降低系统复杂度和开发工作量, 且大幅提高仿真效率^[3]。

4.2 碰撞处理

碰撞处理是系统完成碰撞检测后, 对发生碰撞对象的行为进行处理的过程, 比如当一个球从高处掉到地面时, 系统检测到球与地面碰撞后, 需要根据物理模型对球、地面的后续行为进行仿真, 计算结果决定球弹起的高度、变形、后续运动等情况。目前的图形引擎大都能提供简单的碰撞检测能力, 但一般都不具备碰撞处理功能, 必须由用户自己编程进行碰撞处理。碰撞处理涉及的内容非常多, 除受力情况分析外, 还涉及实时动态变形、特效仿真等技术。目前很多公司在开发第三方软件/插件, 为图形引擎提供碰撞检测、动力学仿真和碰撞处理工作。这种方式存在的问题在碰撞检测章节中已论述, 第三方软件必须构建自己的物理环境, 不能与图形引擎共用同一个物理世界, 造成系统资源浪费严重, 开发工作量巨大, 其发展趋势是实现碰撞检测、处理和动力学仿真功能集成、资源通用^[1,21]。

4.3 动力学仿真

动力学仿真用于计算对象的受力情况, 除用户定制开发的动力学模型外, 如飞行仿真中用户根据

机型特性定制的飞控模型, 还有很多第三方软件支持常见的动力学仿真工作, 包括刚体、软体、流体等, 它们一般同时具有碰撞检测、碰撞处理功能。

(1) 刚体动力学仿真

目前动力学仿真中的刚体仿真技术最成熟, 因为常用的刚体物理学模型已比较完善, 且刚体无需考虑自身形变, 因此实现难度相对较小, 比较先进的刚体动力学仿真引擎有 Vortex、PhysX、ODE 等。其中 Vortex 是最为成熟的商业化刚体动力学仿真引擎, 在工程机械使用培训系统开发中应用非常广泛。图 2 为 CMLabs 公司提供的基于 Vortex 开发的起重机操作训练系统视频截图。



图 2 Vortex 开发的起重机操作训练系统

Fig.2 Crane operation training system developed by Vortex

刚体动力学仿真的下一步发展方向是与软体、流体动力学仿真引擎等进行集成, 新版 Vortex 的绳索仿真就是该发展趋势的一个初步尝试, 因为严格意义上讲绳索并不属于刚体, 而是属于软体^[3,22]。

(2) 软体动力学仿真

软体动力学相比刚体动力学要复杂很多, 很难使用解析式表达其运动和受力情况。目前有少量引擎已开始支持简单的软体仿真, 图 3 为最新的 CryEngine3 引擎中实现的软体动力学技术, 可以看到它表现出了汽车撞墙后遭破坏的视觉效果, 但仅是视觉上的简单模拟, 破坏的具体位置、破坏程度等与真实情况还存在一定的差距, 其底层力学模型还有待进一步完善。

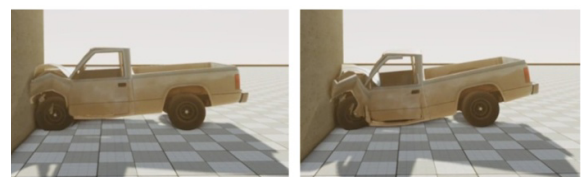


图 3 CryEngine3 引擎中软体动力学仿真

Fig.3 Softbody simulation in CryEngine3 Engine

软体仿真技术需要在如下方面加大研究力度:

软体几何建模技术:除少数软体仿真可以使用面模型外,大多数软体仿真需要为软体构建体模型,比如虚拟手术系统中为人体器官建模等,要求能用手术器械切开器官进行观察,器官的内部模型决定着仿真的真实性,应该运用三维纹理等技术,为软体建立逼真的体模型^[23-27]。

高效碰撞检测技术:碰撞检测是软体仿真最大的难点之一,除了考虑与其他对象的碰撞外,还需要考虑自身碰撞问题,且软体碰撞后会发生形变,其碰撞检测包围盒碰撞后也需要改变;当软体可破坏或碰撞后体模型无法完全复原时,现有包围盒技术将难于描述软体的碰撞检测问题,必须引入新的包围盒技术,比如可以考虑将 Vortex 刚体仿真软件中的 Mesh 包围盒建模技术引入软体动力学仿真,每次碰撞完成后,由引擎自动生成新的 Mesh 包围盒,包围盒与软体的外形基本保持一致。文献[28]提出了一种基于体纹理技术实现瀑布与地型的碰撞检测技术,也是一种很有借鉴意义的思路^[29-30]。

软体材质特性库:软体的动力学仿真与材质密切相关,软体发生碰撞后是否破坏、形变大小、恢复速度和恢复程度等,都与软体材质有非常密切的关系,需要通过大量的试验建立软体的材质特性库。

软体仿真方法:目前主要使用质点-弹簧系统进行软体仿真,在布料模拟、外科手术、毛发皮肤、火焰等仿真中应用广泛。但进行更加复杂的软体仿真时,将需要采用有限元法,有限元法的算法设计非常复杂,虽能获得逼真的仿真效果,但很难达到实时性要求,需要研究有限元法的优化技术^[31-34]。

(3)流体动力学仿真

目前图形引擎大多只是实现单一液体或气体单相流运动的简单模拟、采用快速傅里叶变换实现海面波浪的仿真等,表示流体的网格或粒子只是在其原地波动,无法表示真正的流动和复杂的效果^[2]。

研究流体动力学仿真时,除研究单相流的运动外,还需研究气-固、液-固、气-液和更多种相态的复杂混合物的运动,现有图形引擎提供的特效仿真

功能,如雨雪、爆炸、燃烧、烟雾、云彩、龙卷风、海面等,都属于流体动力学仿真的范畴。目前流体仿真实理论已相对完善,出现了很多专业的流体仿真软件,但因边界条件复杂、计算量过大等原因,一般只能进行小区域的非实时流体仿真,暂无法很好地与图形引擎整合^[35-36]。为提高下一代图形引擎中流体动力学仿真效果,应在如下方面加大研究力度:

流体仿真的细腻度:随着研究的深入,单流体新现象的模拟将逐渐减少,应集中研究改善局部细节处理精度,比如改善流固耦合或者多相界面的精度以获得更为细腻的模拟效果。图 4 为美军新型虚拟训练系统中实现的水花飞溅效果,较好地展现了虚拟人与水体的互动过程^[37-39]。



图 4 虚拟人溅起的水花效果
Fig.4 Splash effect of virtual human

流体仿真加速策略:利用已有流体动力学仿真成果,通过合理简化仿真模型,针对新硬件体系结构来设计高效的自适应混合表达算法,采用高精度的计算格式和快速迭代收敛方法,充分利用 GPU、专用硬件加速卡等提高流体动力学仿真效率,并优化计算和渲染流程的协同问题^[40-41]。

边界条件控制手段:流体动力学仿真要求解其自由表面,才能将其与周围的环境区分开来,对于小尺度液面仿真而言,其拓扑结构随时会发生变化,边界条件非常复杂。在图形引擎中实现流体仿真时,应考虑虚拟环境中的其它对象与流体仿真边界条件的统一问题,从而减少建模工作量并提高仿真效率^[36,42]。

流体仿真控制策略:除了提供对流体本身的属性、粘性系数、边界条件、障碍物等调整和设置功能外,还应给流体仿真开发人员提供更高级别的控

制, 如通过压强、速度、粒子等提高对流体仿真控制的灵活性。

预计下一代图形引擎的流体仿真模块将在如下方面取得进步:

水体仿真: 目前图形引擎已实现了较逼真的海面、浪花、船舶航迹效果仿真, 但水底仿真效果较差, 而在潜艇仿真、海底观光仿真等都需要用到水底效果, 下一代图形引擎有望实现更逼真的水底仿真和更细腻的碎浪、泡沫、飞溅等效果。

动态植被模拟: 目前 SpeedTree 等软件可以实现逼真的树木实时动态仿真, 下一代图形引擎中应支持草地、花卉等更复杂植被的实时动态仿真^[43]。

气象景观: 目前图形引擎一般使用粒子系统实现雨、雪、雾等的仿真, 一般没有考虑光线散射及受力情况, 形态仿真效果也一般。下一代图形引擎应采用物理仿真机制实现更多的气象景观的仿真, 建立较逼真的几何模型, 并考虑重力、风力、光线散射、统计特性等, 从而实现沙尘暴、龙卷风、海市蜃楼、宝光、彩虹等气象景观。图 5 为文献[44]中实现的海市蜃楼、雨后彩虹效果。



图 5 文献[10]中实现的海市蜃楼、雨后彩虹特效
Fig.5 Unreal city and rainbow effects in Paper [10]

爆炸模拟: 目前爆炸、碎片主要是采用粒子系统进行仿真, 破碎效果通过提前建好破坏后的模型实现, 仿真逼真度有限。下一代图形引擎应采用物理仿真机制实现更加真实的爆炸仿真, 根据爆炸物体的材质、受力情况, 自动将物体切分为恰当的碎片, 并以对应的初速度和方向飞出, 在重力、空气阻力作用下逐渐落下, 并提供火焰、烟雾等仿真。

天空景观仿真: 目前图形引擎已能对太阳、月亮等提供较真实的模拟, 未来应提供大气散射、折射、立体云彩、星空等的更真实模拟^[43-44]。

5 光照和着色技术

5.1 高级实时光照技术

光照用来表示材质和光源之间的相互作用, 也可以表示光源与所绘制几何对象之间的相互作用; 而着色处理则是计算光照并决定像素颜色的过程。目前在电影、广告等非实时渲染领域的高级光照技术已相对成熟, 但实时渲染仍主要使用简化的光照模型, 光照效果一般。

光照模型有局部光照模型和全局光照模型, 目前主流三维图形引擎一般只提供局部光照模型, 局部光照模型只考虑由光源引起的漫反射分量和镜面反射分量, 环境反射分量仅用常量来代替, 其真实性受到影响; 全局光照明模型能同时模拟光源和环境照明效果, 可提供真实感更强的光影效果。光线跟踪算法是最经典、应用最为广泛的全局光照模型, 大多数光照模型都以它为基础, 至今仍是图形学的研究热点, 其理论不断得到改进和完善。光线跟踪算法基于几何光学原理, 通过模拟光的传播路径来确定反射、折射、透射和阴影等, 对每个像素进行单独计算, 因此能很好地表现细节。该算法由于需跟踪每一条从视点发出的光线, 因而涉及到大量的光线与场景对象的求交运算, 所需的计算量十分惊人。一般生成一幅中等复杂程度的图形需要进行几百万次直线与场景对象的求交运算, 计算量巨大限制了该算法在实时三维图形引擎中应用^[45]。

下一代三维图形引擎可通过以下途径解决光线跟踪算法在实时仿真的应用瓶颈: 一是优化算法, 提高光线与场景对象的求交运算效率, 如采用光线跟踪参数曲面片、代数曲面片、分维曲面等, 针对曲面特点改善求交运算的数学方法, 减少求交计算量; 二是应用包围盒算法和空间剖分算法等技术, 快速确定光线与景物是否相交; 三是采用性能更好的硬件设备, 如多核心 CPU、专用计算加速硬件或 GPU 等, 尤其是 GPU 开始应用于通用计算领域, 传统光照模拟中大量由 CPU 完成的计算量可转移到 GPU 完成, GPU 天生支持矢量、矩阵计算, 对矢量、矩阵等的计算效率高于 CPU, 能

大幅提高计算效率,从硬件上提供更强的求交计算能力^[46-48]。

5.2 高级着色技术

除光照外,着色处理也是虚拟环境真实感的一个决定性因素,着色处理方法主要有 Flat、Gouraud 和 Phong 3 种,分别基于多边形、顶点和像素来计算光照效果。在高级着色语言出现前,程序员很难方便地控制顶点和像素,很难制作出精美的动画和逼真的光影效果。随着 GPU 技术的发展,目前主流图形引擎都直接或间接支持对顶点着色器和像素着色器的编程,可实现更加复杂的仿真效果,下一代图形引擎有望提供更加逼真、复杂的光影效果。

对国内图形引擎开发方向而言,目前的重点应解决红外和雷达仿真技术,西方国家至今仍对我国实施红外和雷达仿真技术封锁,比如国内使用最广泛的 Vega Prime 引擎就禁止向中国出售红外和雷达仿真模块,如果飞行模拟器不能实现雷达和前视红外设备的仿真,就难于实现训练流程的完整模拟。红外和雷达仿真的最终视觉效果也与实际装备的技术状态有关,即使能买到国外的相关模块,也不一定能很好地满足国产装备的仿真需求,只有掌握了自主的仿真技术,才能摆脱对国外的依赖。

国外的技术封锁也从侧面体现了红外、雷达仿真的技术难度,需要大量的基础研究进行支撑。以红外仿真为例,涉及探测器成像模型、大气辐射传输模型、红外材质库、热传递模型、着色模型等,需要进行长期深入的研究;进行红外仿真时,采用红外仿真模型计算出相关数据后,需要采用高级着色技术对场景中对象的片段颜色进行细微的控制,在高级着色语言出现前,其工程实现难度非常大。

虽然国内已对红外和雷达仿真技术进行了大量的研究,但主要集中于仿真实论、视觉效果仿真,很少进行大气传输模型、红外材质库的研究,该工作涉及大量、长期的试验和测量工作,需要大量的人力、物力,只有在国家层面支持下才可能完成。雷达仿真技术也需要建立仿真模型,并用高级着色技术表现最终仿真结果。图 6 是基于 Vega Prime、

采用红外热模型、高级着色技术实现的红外热像仪仿真效果,演示了随着发射炮弹数目的增加,炮管温度升高而导致炮管的灰度变化。该演示系统验证了技术可行性,但只建立了少数几种红外材质,大气传输采用美国 LOWTRAN 软件计算结果,还需做大量工作才可能开发出自主的红外仿真软件^[2]。



图 6 基于高级着色技术的红外热像仪仿真效果
Fig.6 Infrared Simulation with Advanced Shading Technology

6 结论

虚拟现实引擎与游戏引擎的基本理论基础完全一致,随着硬件性能及相关理论的进步,虚拟现实引擎与游戏引擎功能定位将逐渐重合,虚拟现实引擎将更注重仿真细节、特效的表现,游戏引擎将逐步提供对超大规模仿真的支持,二者有合二为一的趋势,对 GPU、多线程、多 CPU、分布式计算等支持更加完善,高级光照、流体仿真、软体仿真等以往只能用于非实时渲染的技术将逐步应用于下一代三维图形引擎,从而提供更加逼真的视觉仿真效果。

参考文献:

- [1] Ernest Adams, Andrew Rollings. Fundamentals of game design[M]. USA: Pearson Education, Inc. 2007: 90-102.
- [2] Presagis Canada Inc and/or Presagis USA Inc Developer Guider for Vega Prime 14[Z]. 2014.
- [3] CM Labs Simulations Inc. Technical Notes for Vortex 6. 4[Z]. 2015.
- [4] 刘腾飞. 大规模动态虚拟场景管理关键技术的研究[D]. 长沙: 中南大学, 2014: 7-47.
Liu Tengfei. Research on the Key Technologies of Large Scale Dynamic Virtual Scene Management[D]. Changsha: Central South University, 2014: 7-47.
- [5] 王永君, 龚健雅. 用于大范围虚拟地形环境的动态场景管理方法[J]. 测控学院学报, 2003, 20 (3): 231-234.
Wang Yongjun, Gong Jianya. Dynamic scene management method for a wide range of virtual terrain

- environments [J]. Journal of Institute of Surveying and Mapping, 2003, 20(3): 231-234.
- [6] 张宇. 一种基于八叉树的动态场景管理方式[J]. 电脑知识, 2015, 11 (14): 54-57.
- ZhangYu. An scene management method with octree[J]. Computer Knowledge, 2015, 11 (14): 54-57.
- [7] Randima Fernando. GPU Gems programming Techniques, Tips, and Tricks for Real-Time Graphics[M]. USA: Pearson Education, Inc., 2004.
- [8] 孙益辉, 陈福民, 王海峰. 一种基于 OGRE 图形引擎的实时分布式渲染系统[J]. 计算机工程与应用, 2008, 44(31): 102-103.
- Sun Yihui, Chen Fumin, Wang Haifeng. A real-time distributed rendering system based on OGRE graphics engine[J]. Computer Engineering and Applications, 2008, 44(31): 102-103.
- [9] Quantum3D, Inc. User Manual for Mantis2. 8. 3[Z]. 2009.
- [10] Hubert Nguyen. GPU Gems3[M]. USA: Pearson Education, Inc., 2008: 205-263.
- [11] Matt Pharr. GPU Gems 2: Programming Techniques for High-Performance Graphics and General-Purpose Computation[M]. USA: Pearson Education, Inc., 2006: 22-59.
- [12] Nvidia Corporation. Cuda Compute Unified Device Architecture Programming Guide[Z]. 2007.
- [13] Presagis Canada Inc and/or Presagis USA Inc. User Manual for Creator14[Z]. 2014.
- [14] 孙守迁, 吴群, 吴剑锋. 虚拟人技术及应用[M]. 北京: 高等教育出版社, 2010.
- Sun Shouqian, WuQun, Wu JianFeng. Virtual Human Technology and Applications[M]. Beijing: CHEP, 2010.
- [15] Junggon Kim, Nancy Pollard. Fast simulation of skeleton-driven deformable body characters [J]. Acm Transactions on Graphics (S0730-0301), 2011, 30(5): 1317-1329.
- [16] 成效, 万旺根, 余小清, 等. GPU 视频纹理技术在三维电影播放中的应用[J]. 计算机工程与设计, 2010, 31(18): 4020-4023.
- ChengXiao, Wan Wanggeng, Yu Xiaoqing, et al. Application of GPU Video Texture Technology in 3D Movie Playback[J]. Computer Engineering and Design, 2010, 31(18): 4020-4023.
- [17] Filip J. Bidirectional texture function modeling: a state of the art survey[J]. IEEE Transactions on pattern analysis and machine intelligence(S0162-8828), 2009, 31(11): 1921-1940.
- [18] Frank Suykens, Karl vom Berge, Ares Lagae, et al. Interactive rendering with bidirectional texture functions [C]. EUROGRAPHICS, 2003: 1-11.
- [19] 孟慧军, 袁修久, 张敬卓, 等. 基于三维纹理的雷达探测概率直接体可视化[J]. 空军工程大学学报(自然科学版), 2012, 13(1): 42-46.
- Meng Huijun, Yuan Xiujiu, Zhang Jingzhuo, et al. Radar Probing Probability Direct Body Visualization Based on 3D Texture[J]. Journal of Air Force Engineering University (Natural Science Edition), 2012, 13(1): 42-46.
- [20] Ludovic Paulhac, Pascal Makris, Jean-Yves Ramel, et al. A framework of perceptual features for the characterisation of 3D textured images [J]. Signal image and video processing (S1863-1703), 2014, 1(1): 1-29.
- [21] Ian Millington. Game Physics Engine Development [M]. USA: Morgan Kaufmann Publishers, 2007: 193-429.
- [22] Matthias Müller, Bruno Heidelberger, Marcus Hennix, et al. Position Based Dynamics[J]. Journal of Visual Communication & Image Representation (S1047-3203), 2007, 18(2): 109-118.
- [23] Joseph Teran, Neil Molino, Ronald Fedkiw, et al. Adaptive physics based tetrahedral mesh generation using level sets[J]. Engineering with Computers(S0177-0667), 2005, 21(1): 2-18.
- [24] Matthias Müller, Bruno Heidelberger, Matthias Teschner. Meshless deformations based on shape matching[J]. Acm Transactions on Graphics (S0730-0301), 2005, 24(3): 471-478.
- [25] 石义波. 虚拟手术中非线性软体组织形变研究[D]. 济南: 山东大学, 2013: 13-38.
- Shi Yibo. The Study on Nonlinear Softbody Tissue Deformation in Virtual Surgery[D]. Ji'nan: Shan Dong University, 2013: 13-38.
- [26] 郁松. 3D 虚拟现实内镜手术仿真技术研究[D]. 长沙: 中南大学, 2011: 24-97.
- Yu Song. The research on 3D virtual reality endoscopic surgery simulation technology[D]. Changsha: Central South University, 2011: 24-97.
- [27] Max Wardetzky, Mikl'os Bergou, David Harmon, et al. Discrete Quadratic Curvature Energies[J]. Computer Aided Geometric Design (S0167-8396), 2006, 24(8): 499-518.
- [28] 蹇旭. 基于体纹理技术的瀑布与地形碰撞检测[J]. 兰州理工大学学报, 2014, 40(3): 116-119.
- Jian Xu. Waterfall and Terrain Collision Detection Based on Body Texture Technology[J]. Journal of Lanzhou Technology University, 2014, 40(3): 116-119.
- [29] 张莉, 卢豫开, 段绍敏. 虚拟环境中软体的包围盒更新方法分析[J]. 微计算机信息, 2009, 25 (12): 32-33.
- Zhang Li, LuYukai, Duan Shaoming. Analysis of software bounding box updating method in virtual environment[J]. Microcomputer information, 2009, 25 (12): 32-33.
- [30] 淡科锋, 耿国华, 周明全. 虚拟手术中刚体和软体碰撞检测算法研究[J]. 计算机技术与发展, 2008, 18(9): 60-63.

- Dan Kefeng, Geng Guohua, Zhou Mingquan. Research on Collision Detection between Rigid Body and Software in Virtual Surgery[J]. Computer Technology and Development, 2008, 18 (9): 60-63.
- [31] 李健, 李鹏坤, 丁露露. 基于压力模型的柔性物体仿真[J]. 计算机工程与应用, 2010, 46 (31): 206-208.
- Li Jian, Li Pengkun, Ding Lulu. Flexible Object Simulation Based on Pressure Model [J]. Computer Engineering and Applications, 2010, 46 (31): 206-208.
- [32] 陆忠华, 张争艳, 陈定方. 基于有限元法的柔性体变形实时模拟[J]. 武汉理工大学学报(交通科学与工程版), 2012, 36 (4): 700-704.
- Lu Zhonghua, Zhang Zhengyan, Chen Dingfang. Real Time Simulation of Flexible Body Deformation Based on Finite Element Method[J]. Journal of Wuhan Technology University (Traffic Science and Engineering Edition). 2012, 36 (4): 700-704.
- [33] 曾亮, 吴亚刚. 软体多局部破碎特效研究[J]. 计算机学报, 2010, 33 (6): 1110-1114.
- ZengLiang, Wu Yagang. Research on Software Multi - local Breaking Effects[J]. Journal of Computer, 2010, 33 (6): 1110-1114.
- [34] Jonathan M Kaldor, Doug L James, Steve Marschner. Simulating knitted cloth at the yarn level[J]. Acm Transactions on Graphics(S0730-0301), 2008, 27(3): 15-19.
- [35] 柳有权, 王章野, 朱鉴, 等. 基于物理的流体动画加速技术的研究进展[J]. 计算机辅助设计与图形学学报, 2013, 25(3): 312-321.
- Liu Youquan, Wang Zhangye, Zhu Jian, et al. Research Progress of Physical Fluid Animation Acceleration Technology[J]. Journal of Computer-Aided Design and Graphics, 2013, 25(3): 312-321.
- [36] 柳有权, 刘学慧, 朱红斌, 等. 基于物理的流体模拟动画综述[J]. 计算机辅助设计与图形学学报, 2005, 17(12): 2581-2589.
- Liu Youquan, Liu Xuehui, Zhu Hongbing, et al. An Overview of Physical Fluid Simulation Animations[J]. Journal of Computer - Aided Design and Graphics, 2005, 17(12): 2581-2589.
- [37] 徐梦华, 莫海鸿. 大跨度梁式渡槽改进拟三维流固耦合模型的三维模拟效果分析[J]. 振动与冲击, 2010, 29(6): 207-210.
- Xu Menghua, Mo Haihong. Three Dimensional Simulation of Improved Three Dimensional Fluid - Structure Interaction Model for Long-Span Beam Aqueduct [J]. Vibration and Shock, 2010, 29(6): 207-210.
- [38] 曹源, 金先龙, 张晓敏, 等. 基于流固耦合的水力瞬变三维模拟及管壁动态应力分析[J]. 振动与冲击, 2009, 28(11): 70-72.
- Cao Yuan, Jin Xianlong, Zhang Xiaomin, et al. Hydraulic Transient 3D Simulation Based on Fluid - Solid Coupling and Analysis of Wall Dynamic Stress[J]. Vibration and Shock, 2009, 28(11): 70-72.
- [39] 杨志亮, 王章野, 柯晓棣, 等. 多相流灾害场景的真实感建模与绘制[J]. 计算机辅助设计与图形学学报, 2008, 20(8): 1023-1031.
- Yang Zhiliang, Wang Zhangye, Ke Xiaoli, et al. Realistic Modeling and Drawing of Multiphase Stream Disaster Scene [J]. Journal of Computer-Aided Design and Graphics, 2008, 20(8): 1023-1031.
- [40] Shiguang Liu, Qiguang Liu, Qunsheng Peng. Realistic simulation of mixing fluids[J]. The Visual Computer (S0178-2789), 27(2011): 241-248.
- [41] 谭捷. 基于物理的流体动画研究[D]. 上海: 上海交通大学, 2009.
- Tan Jie. Research on physics-based fluid animation [D]. Shanghai: Shanghai Jiao Tong University, 2009.
- [42] Dan S Bolintineanu, Gary S Grest, Jeremy B Lechman, et al. Particle dynamics modeling methods for colloidal suspensions [C]. Computational Particle Mechanics. 2014, 1(3): 321-356.
- [43] Presagis Canada Inc and/or Presagis USA Inc. Vega Prime Reference for SpeedTree [Z]. 2014.
- [44] 彭群生, 王长波, 刘世光. 自然景观的真实感模拟[M]. 浙江: 浙江大学出版社, 2013: 240-289.
- Peng Qunsheng, Wang Changbo, Liu Shiguang. Realistic simulation of the natural landscape[M]. ZheJiang: Zhejiang University Press, 2013: 240-289.
- [45] 李琳. 三维图形引擎设计及其光影特效的研究与实现 [D]. 成都: 电子科技大学, 2006: 57-84.
- Li Lin. Research and Implementation of 3D Graphic Engine Design and Its Light and Shadow Effects[D]. Chengdu: University of Electronic Science and Technology, 2006: 57-84.
- [46] 王世元. 基于 GPU 的光线跟踪算法的加速结构比较研究[D]. 武汉: 华中师范大学, 2008: 1-15.
- Wang Shiyuan. Accelerated Structure Comparison of GPU - based Ray Tracing Algorithm[D]. Wuhan: Central China Normal University, 2008: 1-15.
- [47] 邓志权. 基于 GPGPU 的实时高质量光线投射算法研究[D]. 广州: 华南理工大学, 2012: 5-26.
- Deng Zhiquan. Research on Real Time High Quality Light Projection Algorithm Based on GPGPU[D]. Guangzhou: South China University of Technology, 2012: 5-26.
- [48] 张怡. 实时光线投射算法加速技术研究[D]. 天津: 天津大学, 2006.
- Zhang Yi. Research on Acceleration Technology of Real Time Ray Casting Algorithm[D]. Tianjin: Tianjin University, 2006.