

# 海量模型实时交互可视化技术综述

高宇 吴玲达 魏迎梅

(国防科技大学多媒体研发中心,长沙 410073)

**摘要** 综述了海量模型实时交互可视化的最新研究进展,详细讨论了其中涉及的主要支撑技术,并对目前国内外的交互绘制系统进行比较和分析;最后,对今后的发展方向进行了展望。

**关键词** 海量模型可视化 实时绘制 可见性剔除 并行绘制 外存算法

中图法分类号:TP391 文献标识码:A 文章编号:1006-8961(2008)09-1633-08

## A Survey on Real-time, Interactive Visualization of Massive Models

GAO Yu, WU Ling-da, WEI Ying-mei

(Multimedia R&D Center, National University of Defense Technology, Changsha 410073)

**Abstract** In this paper we presents a survey on current techniques for real-time interactive visualization of massive models. Firstly, the related key technologies are summarized. Then some typical interactive visualization systems for massive models are analyzed and compared. Finally, some possible directions for further research are pointed out.

**Keywords** massive model visualization, real-time rendering, visibility culling, parallel rendering, out-of-core algorithm

## 1 引言

在计算机辅助设计、科学计算可视化和虚拟仿真等应用中,经常需要对复杂的3维模型进行交互浏览。然而随着近几年3维扫描和建模技术的快速发展,3维模型的数据量急剧增大,包含上亿多边形面片的海量模型在实际应用中越来越常见,例如北卡罗来纳大学的 Core-fired Power Plant 模型(13M Tris)和 Double Eagle Tanker 模型(80M Tris)<sup>[1]</sup>、波音公司 Boeing 777 的3维模型(350M Tris)<sup>[2]</sup>、斯坦福大学“数字米开朗基罗计划”(digital michelangelo project)的 David 模型(56M Tris)和 St. Matthew 模型(372M Tris)<sup>[3]</sup>等。尽管当前计算机硬件尤其是图形硬件的发展十分迅猛,目前主流图形硬件的多边形处理能力已经达到每秒钟百万量级甚至千万量

级。但是面对不断增大的模型数据,仅仅依赖目前硬件平台的图形处理能力和传统的实时绘制系统仍然无法满足实际应用对绘制速度的要求。因此,研发新的图形加速绘制技术以及高度优化的实时绘制系统受到了国内外研究者更多的关注,成为计算机图形学和虚拟现实领域的一个新的研究热点。

北卡罗来纳大学联合了美国其他各大著名大学成立了“漫游项目”(walk through project)<sup>[1]</sup>,致力于大规模场景绘制技术的研发。经过十多年的发展,开发出了多套基于 SGI 工作站和 PC 集群的实时绘制系统:MMR (massive model rendering)<sup>[4]</sup>、SHAPE<sup>[5]</sup>、GIGAWalk<sup>[6]</sup>和 SWITCH<sup>[7]</sup>。美国斯坦福大学与意大利 ISTI-CNR 合作,成立了著名的“数字米开朗基罗计划”<sup>[3]</sup>,通过一整套3维扫描硬件和3维重建软件完成了几尊大型雕塑的数字化过程,生成的3维模型都是由数以亿计的三角形组成,其数

基金项目:国家自然科学基金项目(60473117);武器装备预研基金项目(51404040105KG0135)

收稿日期:2006-11-13;改回日期:2007-05-25

第一作者简介:高宇(1978~),男,2007年获国防科技大学控制科学与工程专业博士学位。主要研究方向为计算机图形学、虚拟现实。E-mail:yugao@nudt.edu.cn

据量达到了 GByte 量级。随之,在计算机图形学领域掀起了围绕海量模型数据处理的研究热潮,包括外存简化<sup>[8]</sup>,外存多分辨率表示<sup>[9]</sup>、点绘制<sup>[10]</sup>、外存数据压缩<sup>[11]</sup>、流式传输<sup>[12]</sup>等。意大利 CRS4 视觉计算小组近年来也一直致力于海量数据可视化的研究,所面向的应用主要包括科学计算可视化和虚拟仿真产生的实验数据,并先后在 SIGGRAPH 2004 和 SIGGRAPH 2005 大会上发布了他们研制的 ATP (adaptive tetrapuzzles)<sup>[13]</sup>和 Far Voxels 系统<sup>[14]</sup>。德国 MPI Informatik 与萨尔河大学联合,主要研究海量模型的实时真实感绘制,并与美国的波音公司合作,成功开发了 Boeing 777 3 维模型的交互可视化系统<sup>[2]</sup>,为波音飞机的设计制造提供了很大帮助。

2006 年,在 EUROGRAPHICS 年会上,组委会专门组织了针对海量模型实时交互可视化的专题报告<sup>[15]</sup>,邀请了国际上一些主要研究机构和应用领域的知名学者和工程师,对该领域一些最新的研究成果进行了报道,并进一步讨论了理论研究与应用结合的途径,以及该领域未来的研究前景。

## 2 主要支撑技术

### 2.1 模型分割

处理海量模型的一般方法是将海量的模型数据分割成便于管理的数据块,然后按照空间层次的形式进行组织。这样,在交互绘制时,就可以只将需要的数据块调入内存进行处理。

目前常用的空间数据结构主要包括包围体层次(如包围球层次<sup>[10]</sup>、包围盒层次<sup>[5,6]</sup>)和空间分割层次(如 k-D 树<sup>[16]</sup>、二叉空间分割(BSP)树<sup>[2]</sup>和八叉树<sup>[4,17]</sup>等)。其中,包围体层次通常是按照自底向上的方式将模型中对象的包围体组织成层次的形式,它的优点是结构简单,而且保留了模型原有的对象划分,缺点是不能描述模型的空间连贯性,而且兄弟节点之间往往存在较多的重叠;空间分割层次则是按照自顶向下的方式对整个模型进行分割,它的优点是得到的层次结构较为平衡,而且兄弟节点之间很少有重叠,缺点是可能会破坏模型原有的对象划分。

利用空间层次数据结构,可以加快模型数据的搜索和遍历,快速定位到所需要的数据块;而且可以通过裁剪层次结构的某个分支来快速剔除那些不需要的数据,可以避免大量不必要的计算。

### 2.2 层次细节技术

层次细节技术是在不影响图像视觉效果的前提下,通过逐次简化模型表面的细节来减少模型的几何复杂性,从而提高绘制的效率。文献[18]给出了层次细节技术较为全面的综述。

通常,层次细节方法可以分为静态层次细节和动态层次细节。静态层次细节是一组具有不同复杂度和相似度的模型,这些模型可以由网格简化算法生成。在实时绘制时,根据模型对显示图像的贡献选择适当细节的简化模型<sup>[19]</sup>。动态层次细节是一种紧凑的模型表示方法,它没有显式的分层表示形式存在,而是在绘制的过程中,根据绘制算法所规定的某一准则(如基于视点的准则)自动生成适当分辨率的网格模型<sup>[20,21]</sup>。虽然动态层次细节增加了实时的计算开销,但是它能够保证简化性能和效率之间更好的平衡,而且在不同分辨率切换时还可以避免出现抖动现象。

此外,对于超过内存容量的海量模型,研究者们还相继提出了一些基于外存的多细节层次方法,包括分层层次细节模型<sup>[5]</sup>,聚类的累进网格层次<sup>[22]</sup>等。

### 2.3 可见性剔除

可见性剔除是在进行消隐之前,通过快速判断模型绘制元素的可见性,拒绝那些显然不可见的绘制元素,从而减少送入图形管道的模型的复杂度。可见性问题的综述参见文献[23]。

根据进行可见性评估的依据,可见性剔除可以分为视域剔除、背向剔除和遮挡剔除。其中前两类方法的判断比较简单,已经有较为成熟的算法,而遮挡剔除需要考虑同一场景中绘制元素组之间的相互关系,因此更为复杂,成为可见性研究的重点。

许多绘制系统预先计算基于视点区域的可见性,也就是将场景划分成多个单元,然后为每个单元预先计算单元内任意视点的可能可见集(PVS)<sup>[4,16]</sup>。这样多个视点就可以共用同一个 PVS,这种方法的缺点是需要较长的预处理时间,而且当视点跨越视点单元的边界时还可能会引起磁盘 I/O 的剧增。另一种遮挡剔除方法是基于视点的可见性算法,也就是在漫游时动态计算当前视点的可见性<sup>[17,24]</sup>。虽然这种方法实时的计算量很大,但是它的优点是几乎不需要预处理计算,而且还可以减少运行时刻磁盘 I/O 的剧增,因为从视点到视点的可见性变化要比从视点区域到视点区域的可见性变

化小得多。

遮挡剔除算法的另一种分类是根据进行遮挡计算的空间因素。实体空间的遮挡剔除是利用实体在3维空间中的位置关系来判断遮挡性<sup>[24]</sup>。而图像空间的遮挡剔除则是基于绘制元素光栅化后的离散表示,在图像空间进行可见性判断<sup>[6, 25]</sup>。

## 2.4 几何缓存和预取

在交互浏览时,帧与帧之间视点参数的变化一般很小,也就是说帧与帧之间存在着时间一致性。因此,可以在内存中创建一个缓存区,保存那些最近经常使用的数据。当视点变化时,只需要动态地更新这个几何缓存,通常采用最近最少使用的替代策略<sup>[26]</sup>,保留最近最多可见的模型。

虽然帧与帧之间存在时间一致性,但是有时可见性的变化还是会很大,这是因为即使很小的视点变化也可能引起较大的可见性变化。这种情况下会因为突然增加的磁盘 I/O,造成系统的延迟,使得结果帧率变得非常不稳定。解决这一问题的一种技术途径是投机预取,也就是试图将用户将来可能看到的模型提前调入内存中。预取技术并不是一个新思想,它在操作系统中已经使用了数十年<sup>[27]</sup>。投机预取的关键问题是确定哪些物体将会是可见的。一方面,希望在对模型进行绘制时,这些模型已经在内存中;另一方面也希望避免调入过多最终未被使用的数据,减少不必要的开销。通过预取技术,可以有效地将磁盘 I/O 操作分摊到每一帧,从而得到平滑稳定的帧率。在传统上预取技术都是和基于视点区域的可见性算法一起使用<sup>[4, 16]</sup>。最近,Correa 等人<sup>[17]</sup>又提出了基于视点可见性的预取方法。

## 2.5 基于图像的绘制

基于图像的绘制(image-based rendering, IBR)技术可以用来加速场景的绘制。IBR 技术通过预先计算的图像来生成不同视点的场景画面。因为 IBR 方法的绘制元素是屏幕空间的像素,因此绘制元素的数目与场景复杂度无关。有关 IBR 技术的综述可以参见文献<sup>[28]</sup>。

在海量模型交互绘制中,最多使用的是纹理映射的替代物(texture-mapped impostor, TMI)<sup>[29, 30]</sup>。TMI 可以在预处理时生成,也可以在实时绘制时生成,它的基本思想是对模型中的一些对象,用它们当前视点的图像来替代。TMI 也可以作为层次细节模型(LOD)的一个特殊的细节层次使用。TMI 的进一步扩展包括基于图像的对象 IBO (image-based

object)<sup>[31]</sup>、布告版云<sup>[32]</sup>、多层替代物<sup>[33]</sup>等。纹理深度网格<sup>[4]</sup>也可以看作是对 TMI 的一种改进,纹理深度网格提供了透视的校正,可以用来替代远处的模型。

## 2.6 点绘制

点绘制是随着3维扫描的广泛应用而兴起的绘制技术。一方面随着模型三角形数目的增加,许多三角形在屏幕上的投影小于一个像素;另一方面使用“点元”作为绘制元素可以直接利用扫描得到的原始采样点数据,而且造型灵活,不需要考虑点与点之间的连接关系,LOD 层次的建立和可见性判定也相对简单。点绘制的缺点是数据冗余大,并且缺少专门硬件加速支持。

QSplat<sup>[10]</sup>是点绘制系统的代表,它用一个包围球层次组织点数据并建立 LOD 树,绘制时根据对绘制速度的要求决定 LOD 树的遍历深度,但是当叶结点在屏幕上投影较大时,绘制图像可能产生空洞。因此,还有研究者试图结合点绘制和多变形绘制的特点,研究基于点和多边形的混合绘制方法<sup>[34, 35]</sup>。

## 2.7 并行绘制

采用并行技术是实现海量模型实时交互可视化的一种有效途径。并行技术不仅可以通过多个 CPU 来分担绘制、可见性计算和磁盘 I/O,而且还可以通过多个图形处理器(GPU)实现多边形绘制的并行性。按照对图形绘制流水线的组织方式,并行绘制可以分为 sort-first、sort-middle 和 sort-last 3 种策略。这 3 种并行策略各自有其难点问题,其中 sort-first 的难点是负载平衡问题,而 sort-last 的关键是图像合成,sort-middle 方法虽然最为直观和常用,但也存在这负载平衡和解决几何处理和光栅化之间通信的问题。关于并行绘制的详细介绍可以参见文献<sup>[36]</sup>。

近几年,随着 PC 图形卡性能的提高和网络技术的发展,出现了一种新的并行绘制体系结构,建立在以高速网络连接的 PC 集群上的并行绘制系统<sup>[37, 38]</sup>。集群式的并行绘制系统具有诱人的性价比和良好的可扩展性,已广泛引起人们的研究兴趣。

## 2.8 外存技术

对于海量模型,其数据量往往超过了计算机的内存容量,使得系统无法将整个模型一次性调入,需要访问速度较慢的硬盘。内存到图形卡的数据传输



MMR 系统<sup>[4]</sup>是北卡罗来纳大学在 1999 年研制的一个海量模型绘制系统。MMR 系统采用了多种加速技术,包括基于图像的替代物、LOD 管理、遮挡剔除、基于视点区域的预取等。MMR 系统是首个正式提出可以支持超过上千万多边形模型的交互绘制系统。但是,MMR 系统需要很长的预处理时间和昂贵的高端图形工作站。

QSplat 系统<sup>[10]</sup>是斯坦福大学开发的一个基于“点元”的海量模型交互绘制系统。QSplat 将海量模型表示成顶点层次的形式,并以“点元”替代三角形网格对进行绘制。该系统能够以接近三角形网格的图像质量对 David 和 St. Matthew 模型进行交互显示。QSplat 的最新扩展<sup>[12]</sup>还支持流式传输,可以在普通的网络环境中实现远程绘制。

SHAPE 系统<sup>[5]</sup>是北卡罗来纳大学研制的一个支持大规模静态和动态场景的交互绘制系统。SHAPE 系统采用场景图表示场景,并自动计算场景图节点的 HLOD 模型。节点的简化模型可以对应场景图的整个分支,因此可以得到较高保真度的剧烈简化。此外,SHAPE 系统还通过在漫游时渐进地更新场景 HLOD 模型的子集来支持动态场景,并且系统还支持两种绘制模式:图像质量绘制模式和目标帧率绘制模式。

GigaWalk 系统<sup>[6]</sup>也是由北卡罗来纳大学研制的。该系统通过两个 GPU 来分担遮挡剔除和几何绘制,能够在一台具有 3 个 300MHz 的 R12000 处理器和两个 Infinite Reality 2 图形处理器的工作站上,对 Double Eagle Tanker 模型进行交互绘制,绘制帧率在每秒 4 ~ 5 帧左右。GigaWalk 后来还在基于两台装有高端 NVIDIA GeForce4 显卡的 PC 集群系统上实现,集群中通过 TCP/IP 实现网络连接和通信。

SWITCH 系统(GPU 置换绘制系统)<sup>[7]</sup>是在 GigaWalk 系统的基础上研制的,它不仅利用可计算机中央处理器的并行性,同时还利用了图形处理器的并行性,而且系统还集成了遮挡剔除技术,能够在 3 台 PC 的集群系统上实现海量模型的交互绘制。与 GigaWalk 相比,SWITCH 在增加一台 PC 的基础上,性能增加了一倍,绘制 Double Eagle Tanker 模型的帧率在每秒 10 ~ 11 帧左右。

iWalk 系统<sup>[17]</sup>是由 Princeton 大学的 Correa 等人研制的,它是一个能够在普通 PC 上运行的大规模场景交互绘制系统。iWalk 系统主要包括两个部

分:外存预处理和外存交互绘制。在预处理阶段,通过场景分割和 LOD 计算,为输入场景生成外存的分层多分辨率表示;在实时绘制时,采用多线程技术,通过计算基于视点的可见性,并利用几何缓存和基于视点的预取策略,可以在普通图形硬件上实现大规模外存场景的交互绘制。此外,Correa 等人还采用基于 PC 集群的 sort-first 并行绘制技术对 iWalk 系统进行了扩展<sup>[37]</sup>,能够在 8 个 PC 的绘制集群上以交互式帧率得到高分辨率(4 096 × 3 072)的显示图像。

德国萨尔河大学的 Wald 等人与美国波音公司合作,开发了一个 Boeing 777 模型的交互可视化系统<sup>[2]</sup>,该系统结合了实时光线跟踪、外存管理和数据调度策略,并通过一种类似于光域方法的近似来表示还未调入内存的几何数据,可以大大提高外存绘制的效率,能够在普通 PC 上实现 Boeing 777 模型的交互浏览。

Quick-VDR 系统<sup>[22]</sup>由北卡罗来纳大学的 Yoon 等人研制。该系统结合了视点相关的简化、遮挡剔除和外存绘制。预处理时将模型表示为聚类的累进网格层次(CHPM);在实时绘制时,利用聚类层次进行全局粗的选择细化,利用累进网格进行局部细的选择细化,从而减少了模型动态细化的开销,在保证图像质量的前提下能够在普通 PC 上对超过千万多边形的海量模型进行视点相关的交互绘制。

ATP 系统<sup>[13]</sup>是意大利 ISTI-CNR 和 CRS4 视觉计算小组在 2004 年 ACM SIGGRAPH 大会发布的一个视点相关的海量模型外存可视化系统。该系统采用等角的四面体层次结构对海量模型进行空间分割,其中每个四面体包含了预先生成的简化网格。在实时绘制时,由粗到细遍历四面体层次,根据视点参数选择适当分辨率的四面体进行绘制。ATP 系统充分利用了现代图形硬件的批绘制能力,在目前普通的图形平台上绘制速度超过了每秒 70M 个三角形。

Far Voxels 系统<sup>[14]</sup>同样是由意大利 CRS4 视觉计算小组在 2005 年 ACM SIGGRAPH 大会上发布的。该系统使用了多种混合的绘制元素,其中场景层次的叶节点用三角形网格表示,而内部节点则使用基于光照模拟的“体元”表示。系统还集成了可见性剔除、层次细节模型和外存数据管理,能够在普通的 PC 上实时绘制超过 1G 三角形面片的海量模型。

目前,国内的研究人员也逐步展开了针对海量

模型实时交互可视化的研究工作。北京大学的孟放和冯洁等分别对基于点元和基于三角形网格的大型 3 维模型的交互绘制进行了研究<sup>[43,44]</sup>, 通过结合基于视点的 LOD 控制和内外存调度, 实现了大型 3 维点云数据和大型 3 维网格模型在一般配置 PC 上的实时交互浏览。在 2005 年的 CCVRV 大会上, 浙江大学 CAD&CG 国家重点实验室的鲍虎军教授作了题为“TYR: 大规模场景实时绘制系统”的大会报告, 详细介绍了他们多年的研究成果。TYR 系统是一个基于 PC 集群的大规模复杂动态场景高分辨率实时绘制系统, 该系统集成了外存数据管理、可见性剔除、HLOD 等技术, 并通过多投影画面拼接, 实现了大数据量场景在超高分辨率屏幕上的实时显示。TYR 系统的相关文献可以参见文献[25, 45, 46]。

## 4 结 语

海量模型的实时交互可视化是伴随着近几年 3 维模型数据量的不断增大以及计算机图形硬件的飞速发展而提出来的, 是计算机图形学和虚拟现实领域的一个新兴的研究方向。本文回顾了海量模型实时交互可视化领域的研究进展, 总结了其中的主要支撑技术, 并对国内外一些典型的绘制系统进行了分析和比较。从目前的研究进展情况来看, 虽然取得了不少的研究成果, 也提出了一些实际的应用系统, 但是依然还存在许多问题需要进一步深入研究, 主要集中在以下几个方面:

(1) 处理动态模型: 目前的海量模型交互绘制系统大多只支持静态模型, 即使个别的系统支持动态模型, 往往也假设模型中只存在较少的运动实体。这主要是因为运动实体不仅会造成整个模型空间结构的变化, 而且还会造成可见性变得非常不稳定, 使得预处理的结果变得无效。但是, 在实际应用中, 场景模型往往是动态的, 在漫游或交互的过程中, 模型实体不仅会发生空间位置的变化, 也可能发生形状的改变, 因此如何有效地处理动态模型成为了海量模型交互绘制的一个新的挑战。

(2) 并行绘制: 虽然单个 GPU 的处理能力在过去几年得到了很大的提高, 但是在处理海量模型时依然存在很多的局限性。而并行技术是构建高性能复杂模型绘制系统的最有效的手段。特别是随着近几年 PC 集群技术的发展, 已经逐步取代了昂贵的图形工作站而成为并行绘制的理想平台, 因此如何

合理地利用现代图形硬件所提供的新的并行支持, 更好地解决各个 PC 之间的负载平衡和结果图像的投影合成问题, 都是非常值得深入研究的。

(3) 基于 GPU 的绘制: 近年来, 图形硬件以令人惊异的速度迅猛发展, 其发展的速度甚至是 CPU 更迭速度的 3 倍之多。特别是随着 GPU 的可编程特性的发展, 许多传统上由 CPU 完成的工作都可以在 GPU 上实现了, 如何在海量模型实时交互可视化中合理地利用 GPU 的新特性, 也是计算机图形学当前研究的一个热点方向。

(4) 远程绘制: 随着 Internet 的兴起和网络技术的不断发展, 基于 3 维几何模型的远程绘制技术成为计算机图形学的一个新兴的研究方向。但是对于大规模复杂模型, 其庞大的数据量是模型网络传输面临的一个难点问题。为了降低数据传输的网络带宽, 提高远程绘制的性能, 需要对传输的模型数据进行压缩, 并以流的方式进行传输(先传输粗糙的模型, 后传输精确模型的渐进式传输方式), 也可以通过定义用户的感兴趣区域, 有选择性地传输和绘制用户感兴趣区域的部分信息。

(5) 端到端的显示: 为了实现高质量的绘制, 提高交互性, 目前的海量模型交互绘制系统大多需要较长的预处理时间。例如 Quick-VDR 系统<sup>[10]</sup>在对包含 372M 面片的 St. Matthew 模型进行预处理时, 大概需要 10 个小时的预处理时间, 这在实际应用中往往是无法接受的。因此, 进一步的工作还需要研究新的数据处理和组织策略, 以及探索实时计算与离线处理之间的平衡问题, 尽可能地减少系统的预处理时间, 最终目的是期望提供给用户一种端到端的交互显示方式。

## 参考文献 (References)

- 1 Brooks F P, Jr. The UNC Walkthrough Project [EB/OL]. <http://www.cs.unc.edu/~walk>, March 2001.
- 2 Wald I. An interactive out-of-core rendering framework for visualizing massively complex models [A]. In: Proceedings of Eurographics Symposium on Rendering [C], Norrköping, Sweden, 2004: 81 ~ 92.
- 3 Levoy M. The digital michelangelo project: 3D scanning of large statues [A]. In: Proceedings of ACM siggraph [C], New Orleans, Louisiana, USA, 2000: 131 ~ 144.
- 4 Aliaga D. MMR: An interactive massive model rendering system using geometric and image-based acceleration [A]. In: Proceedings of 1999 ACM Symposium on Interactive 3D Graphics [C], Los Angeles, CA, USA, 1999: 199 ~ 206.
- 5 Erikson C, Manocha D, Baxter W. HLODS for fast display of large

- static and dynamic environments [A]. In: Proceedings of ACM Symposium on Interactive 3D Graphics [C], New York, USA: ACM Press, 2001: 113 ~ 117.
- 6 Baxter W, Sud A, Govindaraju N, *et al.* GigaWalk: interactive walkthrough of complex environments [A]. In: Proceedings of 2002 Eurographics Rendering Workshop [C], Saarbruecken, Germany, 2002: 203 ~ 214.
- 7 Govindaraju N. Interactive Visibility culling in complex environments using occlusion-switches [A]. In: Proceedings of ACM Siggraph Symposium on Interactive 3D Graphics [C], New York: ACM Press, 2003: 103 ~ 112.
- 8 Cai Kang-ying, Fei Guang-zheng. Balanced tiling based out-of-core simplification [J]. Chinese Journal of Computers, 2002, **25**(9): 936 ~ 944. [蔡康颖, 费广正. 一遍完成的平衡布点外存模型简化算法 [J]. 计算机学报, 2002, **25**(9): 936 ~ 944.]
- 9 Shaffer E, Garland M. A multiresolution representation for massive meshes [J]. IEEE Transactions on Visualization and Computer Graphics, 2005, **11**(2): 1 ~ 10.
- 10 Rusinkiewicz S, Levoy M. QSplat: A multiresolution point rendering system for large meshes [A]. In: Proceedings of ACM Siggraph 2000 [C]. New York: ACM Press, 2000: 343 ~ 352.
- 11 Isenburg M, Gumhold S. Out-of-core compression for gigantic polygon meshes [J]. ACM Transactions on Graphics, 2003, **22**(3): 935 ~ 942.
- 12 Rusinkiewicz S, Levoy M. Streaming QSplat: A viewer for networked visualization of large, dense models [A]. In: Proceedings of ACM Symposium on Interactive 3D Graphics 2001 [C], New York: ACM Press, 2001: 63 ~ 68.
- 13 Cignoni P. Adaptive TetraPuzzles-efficient out-of-core construction and visualization of gigantic polygonal models [J]. ACM Transactions on Graphics, 2004, **23**(3): 796 ~ 803.
- 14 Gobbetti E, Marton F. Far Voxels-A multiresolution framework for interactive rendering of huge complex 3D models on commodity graphics platforms [J]. ACM Transactions on Graphics, 2005, **24**(3): 878 ~ 885.
- 15 Kasik D, Manocha D, Yoon S E, *et al.* Real Time Interactive Massive Model Visualization [EB/OL]. <http://www.sci.utah.edu/~abe/massive06/>, 2006.
- 16 Funkhouser T A, Sequin C H. Adaptive display algorithm for interactive frame rates during visualization of complex virtual environments [A]. In: Proceedings of ACM Siggraph 1993 [C], New York, NY, USA, 1993: 247 ~ 254.
- 17 Correa W T, Klosowski J T, Silva C. Iwalk: Interactive out-of-core rendering of large models [R]. Technical Report TR-653-02, Princeton, NJ, USA, Princeton University, 2002.
- 18 Luebke D P. A developer's survey of polygonal simplification algorithms [J]. IEEE Computer Graphics and Applications, 2001, **21**(3): 24 ~ 35.
- 19 Garland M, Heckbert P. Surface simplification using quadric error metrics [A]. In: Proceedings of ACM Siggraph [C], Los Angeles, CA, USA, 1997: 209 ~ 216
- 20 Hoppe H. View-dependent refinement of progressive meshes [A]. In: Proceedings of ACM Siggraph [C], Los Angeles, CA, USA, 1997: 189 ~ 198.
- 21 Pajarola R, DeCoro C. Efficient implementation of real-time view-dependent multiresolution meshing [J]. IEEE Transactions on Visualization and Computer Graphics, 2004, **10**(3): 353 ~ 368.
- 22 Yoon S E. Quick-VDR: out-of-core view-dependent rendering of gigantic models [J]. IEEE Transactions on Visualization and Computer Graphics, 2005, **11**(4): 369 ~ 382.
- 23 Cohen-Or D. A survey of visibility for walkthrough applications [J]. IEEE Transactions on Visualization and Computer Graphics, 2003, **9**(3): 412 ~ 431.
- 24 Klosowski J T, Silva C T. The prioritized-layered projection algorithm for visible set estimation [J]. IEEE Transactions on Visualization and Computer Graphics, 2002, **6**(2): 108 ~ 123.
- 25 Hua Wei. Fast Rendering Techniques for Large-Scale Complex Scene [D]. Hangzhou: Zhejiang University, 2002. [华炜. 大规模场景快速绘制技术 [D]. 杭州: 浙江大学, 2002.]
- 26 Tanenbaum A S, Woodhull A S. Operating Systems: Design and implementation [M]. (2nd Edition) Upper Saddle River, NJ, USA: Prentice Hall, 1997.
- 27 Gindele B S. Buffer block prefetching method [J]. IBM Technical Disclosure Bulletin, 1977, **20**(2): 696 ~ 697.
- 28 Shum H Y, Kang S. A review of image-based rendering techniques [A]. In: Proceedings of IEEE/ SPIE Visual Communications and Image Processing [C], Perth, Australia, 2001: 2 ~ 13.
- 29 Aliaga D G, Lastra A A. Architectural walkthroughs using portal textures [A]. In: Proceedings of IEEE Visualization [C], Phoenix, Arizona, USA, 1997: 355 ~ 362.
- 30 Darsa L, Costa B, Varshney A. Navigating static environments using image-space simplification and morphing [A]. In: Proceedings of Symposium on Interactive 3D Graphics [C], Providence, Rhode Island, USA, 1997: 28 ~ 30.
- 31 Song Han-chen. Research on Image-based Object Modeling and Rendering in Virtual Environment [D]. Changsha: National University of Defense Technology, 2004. [宋汉辰. 虚拟环境中图像对象的建模与表现技术研究 [D]. 长沙: 国防科技大学, 2004.]
- 32 Decoret X. Billboard clouds for extreme model simplification [A]. In: Proceedings of ACM Siggraph [C], New York, NY, USA: ACM Press, 2003: 689 ~ 696.
- 33 Decoret X. Multi-layered impostors for accelerated rendering [J]. Computer Graphics Forum, 1999, **18**(3): 61 ~ 73.
- 34 Cohen J D, Aliaga D, Zhang W. Hybrid Simplification: combining multi-resolution polygon and point rendering [A]. In: Proceedings of the IEEE Visualization [C], San Diego, CA, USA, 2001: 37 ~ 44.
- 35 Ji Jun-feng, Li Sheng. Parameter space based multiresolution structure for hybrid rendering [J]. Journal of Software, 2004, **15**(10): 1515 ~ 1521. [冀俊峰, 李胜. 基于参数空间的混合多分辨率绘制 [J]. 软件学报, 2004, **15**(10): 1515 ~ 1521.]

- 36 Shi Jiao-ying Jin Zhe-fan. A survey on parallel polygon rendering [J]. *Journal of Computer Aided Design & Computer Graphics*, 2003, **15**(6): 637 ~ 642. [石教英, 金哲凡. 并行多边形绘制技术综述 [J]. *计算机辅助设计与图形学学报*, 2003, **15**(6): 637 ~ 642.]
- 37 Correa W T. *New Techniques for out-of-core visualization of large Datasets* [D]. Princeton NJ, USA: University of Princeton, 2004.
- 38 Gouranton V. *Interactive rendering of massive terrains on PC cluster* [R]. Technical Report RR-2004-07, Orleans, France, Computer Science Laboratory of Orleans, 2004.
- 39 Silva C T, Chiang Y J, El-Sana J, *et al.* Out-of-core algorithms for scientific visualization and computer graphics [EB/OL]. <http://www.sci.utah.edu/~csiva/papers/sliva-et-al-2003.pdf>, 2003.
- 40 Lawder J K. *The application of space-filling curves to the storage and retrieval of multi-dimensional data* [D]. London: Birkbeck College, University of London, 2000.
- 41 Yoon S E, Lindstrom P, Pascucci V, *et al.* Cache-oblivious mesh layouts [J]. *ACM Transactions on Graphics*, 2005, **24**(3): 886 ~ 893.
- 42 Yoon S E, Manocha D. Cache-efficient layouts of bounding volume hierarchies [J]. *Computer Graphics Forum*, September, 2006, **25**(3): 507 ~ 516.
- 43 Meng Fang, Zha Hong-bin. Rendering of huge point-sampled geometry based on LOD control and out-of-core techniques [J]. *Journal of Computer Aided Design & Computer Graphics*, 2006, **18**(1): 1 ~ 8. [孟放, 查红彬. 基于 LOD 控制与内外存调度的大型三维点云数据绘制 [J]. *计算机辅助设计与图形学学报*, 2006, **18**(1): 1 ~ 8.]
- 44 Feng Jie, Zha Hong-bin. Simplification and view-dependent LOD control for large 3D mesh models [J]. *Journal of Computer Aided Design & Computer Graphics*. 2006, **18**(2): 186 ~ 193. [冯洁, 查红彬. 大型三维网格模型的简化及基于视点的 LOD 控制 [J]. *计算机辅助设计与图形学学报*, 2006, **18**(2): 186 ~ 193.]
- 45 Lu Yan-qing. *Study of the Real-time Rendering for Large-scale Terrain Dataset* [D]. Hangzhou: Zhejiang University, 2003. [陆艳青. 海量地形数据实时绘制技术研究 [D]. 杭州: 浙江大学, 2003.]
- 46 Jin Zhe-fan, Lin Hai, Shi Jiao-ying. Research and Implementation of a data distributed sort-first parallel rendering system [J]. *Journal of Computer Research and Development*, 2004, **41**(2): 376 ~ 382. [金哲凡, 林海, 石教英. 数据分布型 Sort-first 并行图形绘制系统的研究与实现 [J]. *计算机研究与发展*, 2004, **41**(2): 376 ~ 382.]