

面向移动设备的远程渲染技术综述

徐婵婵, 费广正*, 韩红雷

(中国传媒大学动画与数字艺术学院 北京 100024)
(gzfei@cuc.edu.cn)

摘要: 移动设备以其轻便、便携、易联网等特点成为人们生活的主宰并极大的影响着人们的生活方式, 却因其内存、硬件处理能力、分辨率、电池续航等方面的弱势限制了移动应用程序的开发和普及. 因而, 面向这些移动设备的远程渲染技术是计算机图形学领域的一个热门研究课题, 具有很大研究意义的同时也极具挑战性. 文中回顾了远程渲染技术的研究进展, 对其关键技术以及应用进行概括和对比分析, 最后对未来工作中的研究难点和发展趋势进行了展望.

关键词: 移动设备; 远程渲染技术; 云游戏; 图像变形
中图分类号: TP391.41

A Review of Remote Rendering on Mobile Devices

Xu Chanchan, Fei Guangzheng*, and Han Honglei

(School of Animation and Digital Arts, Communication University of China, Beijing 100024)

Abstract: Mobile device has become more and more popular because it provides an easier way to gain access to internet. However, its application is significantly limited by the storage capacity, processing power, screen resolution and battery life. In computer graphics area, this presents a new challenge for research on remote rendering technique. In this paper, we present an overview of classic remote rendering techniques on mobile devices. Firstly, we analyze the strengths and weaknesses of the techniques. We also compare the performance of their state-of-art implementation. Finally, we provide some insight into the future development of remote rendering.

Key words: mobile devices; remote rendering; cloud gaming; image warping

移动设备的出现给人们的生活带来了巨大的变革. 手机、掌上电脑等移动设备以其轻巧、便携及随时随地可联网等特点成为人们生活的必需品. 近年来, 移动设备在图形处理能力上的革新和多媒体应用上的出色表现给人们带来了全新的娱乐体验. 然而, 随着图像捕获(如光探测与测量技术)、三维建模、三维渲染技术的发展演变, 产生的图形数据愈加庞大, 所需的计算资源仍非当前高端移动设备所能具备. 在本地运行高质量的三维渲染程序对移动设备来说依旧是一个异常艰

巨的任务. 面向移动设备的远程渲染技术则成为当前解决这一问题的有效手段.

远程渲染技术的思想早期出现于 20 世纪 90 年代^[1], 其使用专有的、高性能的设备(称为服务器端)负担渲染工作, 而把显示和用户交互的程序放在网络远端设备(称为客户端)执行. 该处理方式免去了显示终端的计算量, 使得一些大型的应用程序能够运行于处理能力低的终端设备上, 故而能够解决在移动设备这种资源受限的终端机上渲染复杂几何和呈现精美画面的难题. 应用面向移动

收稿日期: 2015-06-09; 修回日期: 2015-12-25. 基金项目: 国家科技支撑计划(2012BAH62F00). 徐婵婵(1988—), 女, 博士研究生, CCF 学生会员, 主要研究方向为计算机图形学; 费广正(1973—), 男, 博士, 教授, 博士生导师, 论文通讯作者, 主要研究方向为计算机图形学、虚拟现实; 韩红雷(1980—), 男, 博士, 讲师, 主要研究方向为计算机图形学、游戏设计.

设备的远程渲染技术弱化了复杂的应用程序对移动设备硬件计算性能的依赖,更深层次地挖掘了移动应用市场的潜在价值,从而刺激了远程可视化、远程漫游、远程几何模型展示、云游戏等应用领域在移动终端的发展和普及。

1 概述

1.1 移动设备的构成及不足

移动设备亦被称为手持设备或者行动设备,常见的有智能手机、掌上电脑、平板电脑、掌上游戏机等。一款移动设备主要包括硬件体系、软件体系以及网络模式三部分,如图 1 所示。其中,中央处理器、存储器以及图形处理器共同决定着移动设备数据处理能力的优劣;而图形处理器是计算机显卡的处理核心,在移动设备中则作为一枚图形芯片和 CPU 封装在一起,主要负责处理视频解码和三维图形渲染工作。表 1 列出了最新图形处理器附带的 GPU 参数。

由表 1 可以看出,当前领先的移动设备图形处理器性能在 GPU 频率上可能会优于 2006 年的中高端显卡,但是在像素填充率以及浮点运算性能上就明显处于劣势(如表 1 中 GeForce 8800GTX),与最新的显卡性能相比更是相差甚远(如表 1 中 GeForce GTX960)。并且,现今的图形图像处理技术产生的几何数据往往以 10^{12} 为数量级,远超出移动设备所能承载的容量。在运行大型或超大型三维场景以及计算复杂数据逻辑上,移动设备因尺寸带来的固有缺陷显得愈发明显且难以在短期内有所突破。除此之外,移动设备使用的无线网络传输速率不高(如图 2 所示),且连接不稳定,易断开。这些均限制了原面向台式机的应用程序在移动设备上的实现,远程渲染的思想可以很好的解决这一难题。

1.2 远程渲染技术的核心思想及革新

远程渲染的思想起源于分布式渲染。借鉴于分布式渲染中将用户交互流程和渲染流程交由不同 CPU 执行的思想,远程渲染引入网络传输,

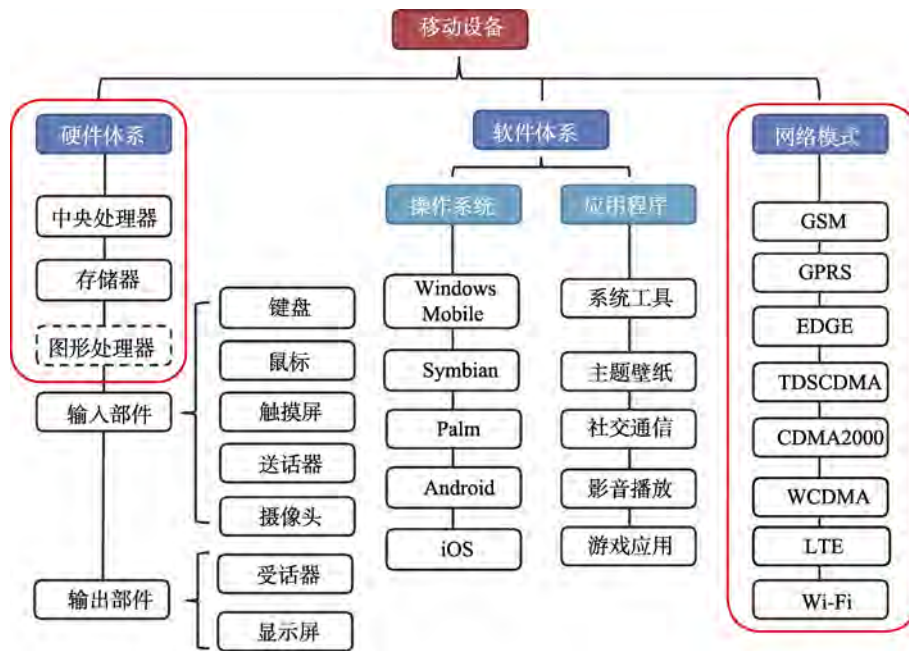


图 1 移动设备的构成

表 1 图形处理器参数对比

型号	上市年份	类型	GPU	生产厂商	频率 /MHz	像素填充率 /(G 像素/s)	32 位浮点运算 /(GFLOPS)	显存		DirectX	OpenGL
								频率 /MHz	容量 /MB		
Exynos 7420	2015	移动	Mali-T760 MP8	ARM Holdings	772	5.6	155			11.1	ES 3.1
Atom Z3590	2015	移动	PowerVR G6430	Intel	640	4.8	155			11.1	ES 3.0
GeForce 8800GTX	2006	台式	GeForce 8800GTX	NVIDIA	575	13.8	500	1 800	768	10.0	2.0
GeForce GTX 960	2015	台式	GeForce GTX 960	NVIDIA	1 126	36.1	2 300	7 010	2 000	11.2	4.2

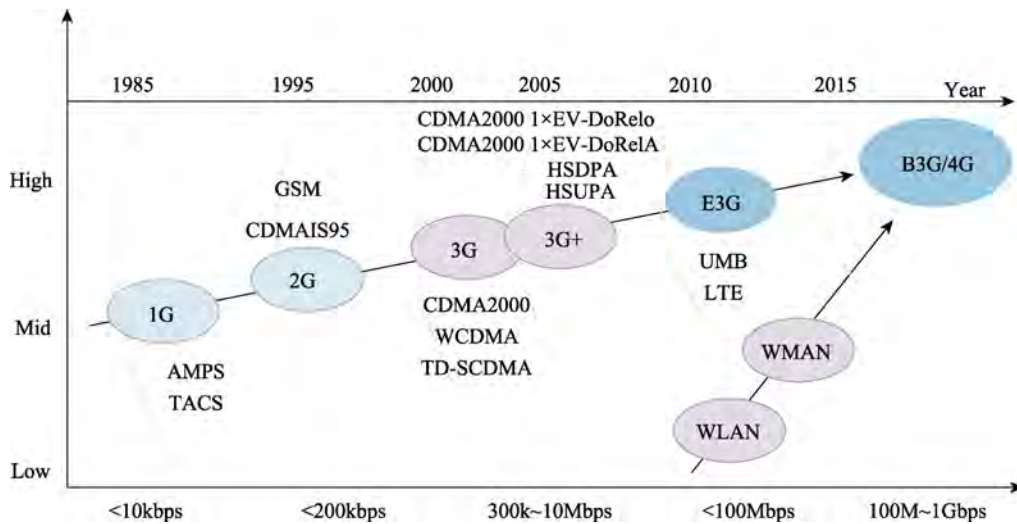


图 2 常用移动网络模式

将用户交互和渲染放于经由网络连接的不同设备上, 负责渲染的设备用于给用户渲染服务, 称为服务器端, 而负责给用户显示以及接收用户直接操作的设备, 称为客户端。

远程渲染有以下特点: 1) 必须使用网络进行通信, 通信的内容可以是交互请求或者场景数据(几何/图像); 2) 用户在客户端操作并浏览内容, 服务器只提供服务而不提供显示; 3) 计算重心一般落在服务器端; 4) 服务器会使用渲染集群或硬件加速方法给渲染流程提速。由此, 利用远程渲染的方法能够利用服务器端的高计算性能经由网络传输数据, 从而在低端的设备如智能手机上显示原本超出其计算负荷的渲染结果。

远程渲染自被提出后经历了一段时间的缓慢发展过程, 其主要应用是电影、动画的离线渲染服务(如渲染农场), 该服务并不能交互, 只提供渲染完毕后的浏览。近年来, 大数据、可视化等带有大型数据库应用的兴起, 带动远程渲染技术的快速更迭。3G/4G 网络的普及和智能移动设备的大众化, 使得远程渲染技术和移动设备的结合更加紧密。云游戏、在线装修等新型的基于云渲染的服务实现了在普通电脑甚至网页上运行复杂图形应用, 这也表明在低端化、移动化的设备上实现复杂图形应用开发已经成为一种趋势。移动设备对基于移动 GPU 开发的图形框架 Metal 和 Vulkan 的支持以及 OpenGL/ES 相关渲染标准的合并将给移动应用开发格局带来新的可能, 面向移动设备的远程渲染技术已成为一个颇具前景、颇具潜力的研究课题。

1.3 面向移动设备的远程渲染体系结构及问题

本文总结面向移动设备的远程渲染体系结构如图 3 所示。可以看出, 服务器端响应客户端请求的流程可分为数据生成、数据传输和数据显示 3 个阶段。服务器端一般包含用户模块和应用程序模块 2 个模块。用户模块用于处理用户的请求信息, 而应用程序模块用于数据的生成、处理和压缩。远程渲染的整个流程可描述如下: 服务器端调用数据库中的数据生成模型; 之后根据渲染系统的方法设定以及用户请求信息确定需要传递的数据并经由无线网络传输到移动客户端上; 移动客户端接收数据, 并根据数据类型选择直接显示、三维渲染或者变形等方法将最终渲染结果呈现给用户。由此, 整个渲染过程受原始模型的复杂度、服务器端硬件处理能力、网络传输能力和客户端硬件的处理能力这四方面的影响。

面向移动设备的远程渲染技术的核心在于处理复杂几何数据在资源受限的移动设备上高质量渲染的问题, 而目的则在于优化基于远程渲染的移动应用的表现和能耗。为此, 相关研究方法主要从以下方面入手: 1) 交互延迟。流畅的交互是所有应用的目的, 探究如何保证交互延迟在用户的容许范围之内; 2) 显示质量。如何保证移动端在有限的屏幕分辨率上给用户可提供可接受渲染质量; 3) 负载均衡。如何平衡网络负载, 以及如何平衡客户端和服务器的计算负载; 4) 数据压缩。如何简化数据, 减少表示、存储、传输的数据量。

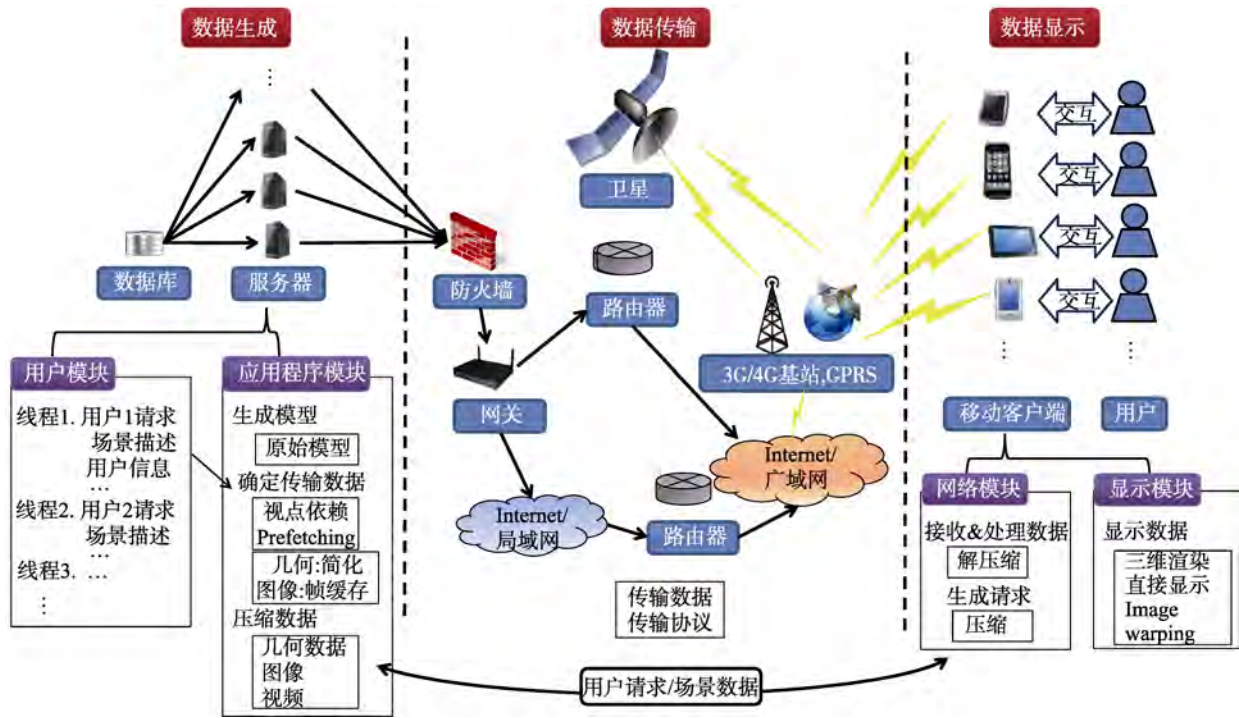


图 3 面向移动设备的远程渲染体系结构

2 面向移动设备的远程渲染关键技术

根据渲染端使用的数据类型, 本文把面向移动设备的远程渲染技术划分为基于模型的渲染方法、基于图像的渲染方法和混合渲染方法 3 种进行讨论.

2.1 基于模型的渲染方法

在该类方法中, 客户端使用从服务器端下载的几何模型(三维网格、点云模型等)在本地渲染并把渲染结果呈现给用户. 这些几何模型可以一开始就已经被客户端完整下载; 也可以在用户交互过程中逐步被传输到客户端上. 服务器负责几何数据的生成和存储、用户请求处理、几何模型的选择拷贝、几何数据的压缩传输等任务; 而客户端负责几何数据的解压缩、本地渲染和显示. 该类方法需要客户端具备足够的硬件计算和存储能力, 以承担相当的计算量和存储量, 同时还需要充足的网路带宽以及及时输送大量的几何数据, 而在移动设备和无线网络下这两点均难以满足. 其改进方法是对处理模型做简化处理. 按照简化的方式和程度, 该类方法还可进一步划分为基于原始模型、基于提取模型和基于抽象模型的渲染方法 3 种子类.

2.1.1 基于原始模型的渲染方法

考虑到服务器端在处理并发请求、场景渲染以及视频图像的压缩和网络传输可能造成的交互延迟, 基于原始模型的渲染方法选择将几何模型传

输到客户端进行本地渲染. 简化的方法是进行几何压缩, 其几何编码标准有 X3D^[2]、三维图形压缩模型^[3]等. 客户端负责解压缩几何数据并渲染输出, 其渲染大多能够使用系统图形库完成^[4]. 在模型传递完毕之后, 网络带宽以及网络拥堵对交互造成的影响即被弱化, 其交互延迟则主要取决于客户端的硬件性能. Games@Large^[4-6]利用客户端渲染原始模型的方法, 实现了在局域网环境下(家庭或宾馆)给用户提供更便捷的游戏体验. 其中, 文献[4]利用指令缓存及混合编码方法, 节省了 80%的图形数据码率; 文献[5-6]则介绍了 Games@Large 的体系结构. 该类方法适用于中小型几何模型的交互, 针对大型场景则需要使用提取或者抽象重构的方法简化几何, 以降低移动终端需要处理的数据量.

2.1.2 基于提取模型的渲染方法

基于提取模型的渲染方法依靠传输部分或简化几何模型的策略实现大规模几何场景在移动终端的本地渲染, 其特点是场景模型过于庞大而需要使用简化或删减的方法减少几何数据量. 层次细节(level of detail, LOD)的方法能够获得不同几何分辨率的简化模型, 从而实现数据压缩^[7]、渐进式渲染^[8]和自适应分辨率显示^[9]等. 经典的背面剔除、视域剔除、遮挡剔除等方法可以省去不在可视

范围内的几何模型的渲染工作量。基于视点依赖的策略可以根据场景几何距离观察者的远近选择不同的细节层次模型显示,也可以选择在视域内的部分模型传输。在用户交互操作上,采取受限的导航机制约束用户的漫游路径和操作,根据用户的历史行为记录^[7,10]预测用户的下一动向;之后使用预取(prefetching)策略^[11]预先下载模型,减少模型传输的等待过程。其中,Deb等^[7]使用前5帧的行为平均值,而Lazem等^[10]使用用户的行为存储库以及前向(look-ahead)策略。文献^[12]利用用户学习方法证明用户的交互操作行为可以被预测并呈现出倾斜访问的特性,Shi等^[13]则总结了漫游过程中常见的一些行为模式,可以据此作为用户行为预测的凭据。基于提取模型的方法给客户端提供的是部分模型或者粗糙网络,所以可以在带有数字安全的模型展示中用于交互操作。通常情况下,该类方法需要辅以基于参考图像以保证最终的渲染质量^[14]。

2.1.3 基于抽象模型的渲染方法

基于抽象模型的渲染方法使用艺术化的方式重构几何模型的轮廓形态,主要应用于非真实感绘制(non-photorealistic rendering, NPR)中。使用的表示基元是线基元^[15-16]或者点云^[17-19]。使用线基元表示的关键步骤即是提取特征线,Diepstraten等^[15]直接从几何模型中提取几何模型的边界线、轮廓线、山脊线、山谷线作为特征线。Quillet等^[16]对二维图像使用Canny边界检测器,并用像素链接方法获得几何特征线。文献^[20]使用分布式风格化机制以加快特征线提取流程。至于点云模型,其脱离了几何模型线面的拓扑关系,而以密度表示模型的艺术色调。获得点云的方法有He等^[21]利用基于体素划分的规则采样方法, Ji等^[18]使用不同密度的点勾勒模型的轮廓和等值面的方法。Liang等^[19]根据用户兴趣属性和网络带宽,给出基于点云模型的层次选择和传输控制方法。Yan等^[22]使用LOD点云模型和误差控制机制以降低模型数据传输频率,最小化网络传输的数据量。线基元和点云模型由原始模型衍生变化而来,在数字博物馆的展示中使用带有颜色和法线的点云模型用于交互,既能保证数字模型的安全性又能增加用户交互的流畅度^[17]。在提高NPR的绘制效率和绘制效果中,朱标等^[23]设计一种可交互像素块的数据存储标准,并在基于MapReduce并行框架上实现了在线的三维模型的高分辨率表意性绘制,满足了百兆、千兆的几何模型的绘制及交互需求。

2.1.4 小结

基于模型的渲染方法从模型的完整性和表示方式上入手。使用完整模型的方法的优点在于对模型的交互操作受网络的影响大大弱化,交互延迟则主要和移动终端的硬件性能有关。但是受限于移动设备的有限的存储空间,该方法仅适用于中小型场景的本地渲染,以及几何模型的计算生成比模型渲染需花费更多计算资源的应用程序。模型提取方法和模型抽象表示方法是针对大型几何模型在移动设备上难以存储和计算的问题提出来的,“模型提取”方法抽取原始几何模型的部分数据(蕴含合理的拓扑结构)到移动端渲染,而“模型抽象”方法是使用另一种表示方法描述几何模型的物理特征,这2种方法以一种削减几何信息的方式减少需要渲染的几何模型数据量,同时力图保证用户交互的流畅,可应用的场景更为广泛。然而,信息量的缺失意味着渲染输出结果的失真,并且这2种方法给服务器端额外增加了计算工作,造成了不小的计算压力。

2.2 基于图像的渲染方法

与基于模型的渲染方法相比,基于图像的渲染方法由服务器完成渲染任务,根据从客户端获得的用户数据(位置、相机参数)渲染场景,并将渲染得到的数据以二维图像或者视频流的方式发送给客户端呈现。由于大部分计算都已经在服务器端完成,因而该方法不需要客户端设备具备图形处理功能,在低配置的平台,如资源受限的移动设备或者移动浏览器上也能呈现出高质量的渲染效果,可用于几何复杂的、追求画面质量的应用程序在移动平台上的实现。按照图像使用的方式,该方法包括基于图片替换(impostor)和基于图像变形的渲染方法2种。

2.2.1 基于图片替换的渲染方法

图片替换的思想是远程渲染的一个经典方法,客户端将服务器在用户当前视角下渲染得到的图像直接替换成当前帧,用户无交互操作时传递单张图片,有交互操作时传递视频流片段,客户端不参与场景的生成和渲染而只负责操作指令的发送和图像数据的解压缩;其显示效果仅和移动设备的屏幕分辨率有关,广泛用于远程游戏^[24]、三维可视化^[25]、场景漫游^[26]等对画面质量要求较高的应用程序中。基于图片替换的渲染方法在数据压缩和减少图片的传输频率两方面进行改进。经典的图像压缩方法为JPEG,能将BMP图片压缩至原来

的 0.8%~2.1%^[17], 而 MPEG 和 H.26X 系列为最常用的视频编码方法, 此外, Deb 等^[7]使用基于小波的逐步转换编码器用于图像压缩, Giesen 等^[27]利用相机参数及运动估计实现视频压缩增强. Ghiletiuc 等^[28]提出多层次天空盒全景替换图片的方法, 证明能达到 0 延迟, 60 帧/s 的渲染速度, 并能适应于不同分辨率的低端终端机上^[29]. Pazzi 等^[30]提出渐进全景图的传输协议, 使用关键部分全景图和按需传递的策略降低交互延迟. 除此之外, 使用图像变形方法也能减少替换图片的传输频率.

2.2.2 基于图像变形的渲染方法

不同于直接图片/视频替换的渲染方法, 基于图像变形的渲染方法利用接收到的图片(通常为深度图)中的深度信息对图片变形以生成新视点下的图像. 深度图包含一幅颜色图像和深度图像, 颜色图像用作无交互下的帧图像, 而深度图像记录了每个像素点的深度信息, 用于交互操作中的新视点图像的生成. Pajak 等^[31]将深度图和视频流结合, 提出增强视频流的概念. Shi 等^[32]亦将三维图像变形方法和视频结合, 由参考帧生成中间帧, 解决了交互延迟的问题, 并实现移动设备上云游戏的视频增强^[33]. Vázquez 等^[34]引入了视点预测和渐进式传输的策略. Bao 等^[35-36]提出基于视点补偿机制的图像渲染方法, 利用图像变形求得稀疏的差值图像, 极大减少了网络传输数据量. Hsia^[37]提出一种针对深度图缺失信息的视点补偿. Tizon 等^[38]和 Makhinya^[39]提出基于深度图的兴趣区域划分的自适应视频编码方法, 提升了视频压缩率. 图像变形产生的新视点图像不可避免会有信息缺失, 文献[36,40]提出利用超视点的超尺寸参考图像以消除视觉间隙, Shi 等^[13]利用多深度图以尽量减少孔洞的产生, 针对已经存在的图像缺失, 使用超采样进行补偿^[41]. Hsia 等^[37]使用对称和不对称高斯过滤器减少图像变形过程中产生的孔洞.

服务器端的硬件性能远比移动终端的强, 同时服务器端可以使用分布式集群渲染、使用基于 GPU 的硬件加速方法^[42]、使用云计算的方式利用所有可以使用的服务器, 所以基于图像的渲染方法一直颇受高性能的场景渲染应用的青睐. 随着图形硬件的飞速发展, 这种方式所能处理的几何场景也越来越复杂, 能承担的计算职能也愈加强大; 其缺点是系统在处理并发请求的能力比较弱, 而用户数量的增加会导致计算量成倍增加, 所以并不能有效处理多个用户的并发操作, 若有即是

各个设备共享同一个显示视角^[43]. 并且这种方法忽略了移动设备本身具有的计算能力.

2.2.3 小结

基于图像的渲染方法把模型计算和渲染均交由服务器处理, 并不要求客户端硬件具备高性能图形处理器, 所以能适用于很多低端配置的终端设备上. 具体说来, 使用图片替换的方法给客户端的计算压力是最小的, 客户端只需要具备图片/视频的解压缩能力和操作指令的发送接受能力即可, 该类方法针对静态场景或者操作偏少的交互式场景很有优势, 但是处理动态的、交互式强的应用程序往往会带来交互卡顿和画面跳跃等不自然的交互体验. 一个原因是从用户操作命令从客户端发出到服务器根据操作命令渲染图像发回客户端这个流程需要经过 1 个往返时间, 且其传输速率和网络带宽以及网络状况有关; 第二个原因则是大量视频/图片数据极易造成无线网路拥堵. 改进的方法除了经典的图像压缩和视频压缩标准外, 使用基于深度图的图像变形方法从减少图片的传输频率的角度, 通过增加客户端一点计算压力进而缓解网络拥堵情况并极大的减少了交互延迟, 该方法在以几何旋转为主要交互操作的应用如几何模型展示方面颇具优势, 也能应用于交互简单的三维场景漫游中.

2.3 混合的渲染方法

混合渲染的方法是基于模型的渲染方法和基于图像的渲染方法的结合, 具体说是基于提取/抽象模型的渲染方法和基于图片替换的渲染方法的结合. 当前的研究方法有 2 个方向: 一是划分操作过程, 针对用户的交互操作和简单查看 2 种操作过程使用不同数据流(几何数据和图像数据); 二是划分场景几何, 即将部分场景(通常是静态的)的渲染图像作为简单的几何模型(如平面)的纹理, 如背景, 而前景则使用场景模型.

2.3.1 基于操作划分的渲染方法

该类方法以一种面向过程的方式整合基于模型的渲染方法和基于图像的渲染方法, 将用户操作看作动态交互过程和静态浏览过程 2 个流程. 在交互过程中, 使用粗糙的几何模型^[44]、稀疏三维网格面片^[45]或者是点云模型^[17]达到实时交互; 而在静态浏览时, 使用高质量的渲染图像替换当前帧. 该方法非常适用于对单个几何模型的交互, 在数字文化遗产的模型显示^[17,45]中具有很高的应用价值.

2.3.2 基于场景划分的渲染方法

相对于近景而言, 场景的远景并不能给人带来强烈的空间感. 基于场景划分的方法即基于这种观察思想, 将远处的几何模型用单张图像代替而近处的场景仍使用几何模型表示. 这种处理方法省去了远处场景的几何模型的存储和渲染, 同时保证用户的空间操作自由和视觉一致性. 存在的问题是图像和模型的接壤处会产生视觉断裂, 而且平面的二维图像没有纵深感. 改进的方法是使用全景图^[14,46]以增加背景图像的空间感, 主要应用于远程场景漫游中.

2.3.3 小结

混合的渲染方法力图充分利用服务器端和客户端的硬件计算能力, 在交互延迟和视觉效果上取得平衡. 其中, 基于过程划分的渲染方法利用不同类型的数据流应对不同的操作状态, 优点是既可以保证用户交互过程的流畅性, 亦能保证在交互结束后渲染效果的质量, 缺点是不同的数据流会造成交互中和最终查看时视觉上的差异, 给用户造成不自然的体验, 并且根据几何模型的重建原理, 仍有发生数据泄露的可能; 基于场景划分的渲染方法利用视点依赖的策略, 仅将用户附近的几何模型传递到客户端渲染, 该方法在远程场景漫游中非常有优势, 结合全景图的方法, 用户在固定位置的旋转查看几乎不需要传输新的数据, 问题是当用户发生位置移动的时候, 新的场景几何模型以及背景的全景图就需要重新获得, 这时会

产生较大的交互延迟.

2.4 分析讨论

本文对基于模型、基于图像以及混合的渲染方法做了分析比较, 结果如表 2 所示. 可以看出, 这 3 种渲染方法各有千秋, 但是秉承着“能者多劳”的理念, 其共同点均是由服务器端分担大部分的计算压力; 其共同目的是追求在尽可能减少交互延迟的前提下给移动终端带来流畅、自然、高质量的用户体验; 其共同面临的主要矛盾是无线网络受限的数据传输能力和用户对高质量低延迟应用性能的需求之间的矛盾; 其考虑的共同的解决途径是尽可能减少经由网络传输的数据量, 比如视频/图像/几何模型/交互命令的数据压缩、模型简化方法和按需传递策略等; 以及最大限度的利用本地已有数据, 比如图像变形方法、使用全景图和用户行为预测策略等. 究其不同之处: 1) 基于模型的渲染方法中模型计算部分发生在服务器端而模型的渲染部分发生在客户端, 要求终端设备具备一定的计算能力和存储能力. 其优点在于交互延迟受网络的影响要比基于图像的渲染方法弱, 适用于对交互速率的需求远比显示质量高的应用; 而受限于移动设备本身的处理能力, 只能处理中小型几何模型的远程显示, 并且容易发生数据泄露. 其优化方法着重在简化模型表达以及删减不需要计算的几何面片上, 比如 LOD 方法、特征线表示

表 2 3 种渲染方法的分析比较

子类	基于模型的渲染方法			基于图像的渲染方法		混合渲染方法	
	完整模型	提取模型	抽象模型	图片替换	图像变形	操作划分	场景划分
渲染端	客户端	客户端	客户端	服务器端	服务器端	客户端/服务器端	客户端/服务器端
传输数据	完整几何	部分几何 简化几何	线基元 点云	参考图像 视频流	深度图 视频/视点信息	简化/部分几何 参考图像	部分几何 参考图像
占用带宽	高	中等	中等	高	低	低	低
主要方法	几何压缩	LOD 预取策略 视点依赖	特征线提取 点云构建 表意绘制	视频压缩 图像压缩 多层次全景图	图像变形 视点补偿	LOD 视频压缩 图像压缩	视点依赖 图像压缩
目的	交互流畅	删减几何 交互流畅	减少数据量 交互流畅	保证质量 交互流畅	保证质量 降低传输频率 交互流畅	交互流畅	减少数据量 交互流畅
文献	[2-6]	[7-12]	[15-23]	[17, 24-30]	[13, 31-41]	[17, 44-45]	[14, 46]
优势应用	中小型几何	复杂几何场景 模型展示	NPR	强调高质量远程游戏 可视化偏静态场景	旋转操作场景 模型展示	强调交互流畅 模型显示	强调交互流畅 场景漫游
不足	移动端存储 数据有限	效果失真; 移动端存储有限	效果失真; 服务器端需 做预处理	高清图片造成网络拥 堵; 较难处理并发请求	图像变形易产 生信息缺失, 影 响视觉效果	交互完毕需等待 较长时间查看最 终图像	用户位置移动时 会造成较大延迟

方法、点云表示方法和视点依赖策略等。2) 基于图像的方法中的渲染和计算工作由服务器全权代劳, 因而非常适用于配置较低的终端设备, 但是也因此给服务器端造成极大的渲染压力, 尤其是在处理并发用户请求上的劣势相当明显。渲染结果经由视频流或图像的媒体形式发送到移动终端, 可以保证在客户端获得相对较高的渲染结果, 适用于强调渲染质量的应用程序, 但是传输大量清晰图片也会造成网络拥堵, 故而在即时性强的应用程序, 比如射击游戏中面临的挑战很大, 使用深度图并结合图像变形的方法可以减少图片的传输频率, 从而减少交互延迟。3) 混合渲染方法结合了基于图像替换的渲染方法和基于提取模型的渲染方法。应用于大规模三维场景渲染中时, 应用全景图片替换作为场景远景, 而使用几何模型作为近景用于用户交互; 应用于高精度几何模型展示时, 粗糙网格可以给用户提供低延迟的交互, 图片替换的方式用于用户交互结束之后的最终效果呈现, 这样既保证了交互速率又保证了最终的渲染质量。所以, 混合渲染也是当下较为流行的渲染方式。

本文对近年来面向移动设备的远程渲染方法的方法研究、方法性能、方法应用这几方面进行统计, 如图 4 所示。可以看出, 自 2010 年以后, 智能手机和平板电脑等移动设备开始占领移动设备市场, 面向移动设备的远程渲染技术迅速发展起来。早期文献主要面向方法探索, 之后应用实现和性能分析的文献开始增多, 说明人们对现有方法陷入一种性能评测、思索和再改进的阶段。根据这几年的论文发表情况可以明显看出, 面向移动设备的远程渲染技术现处于国际前沿的关注热点之一。

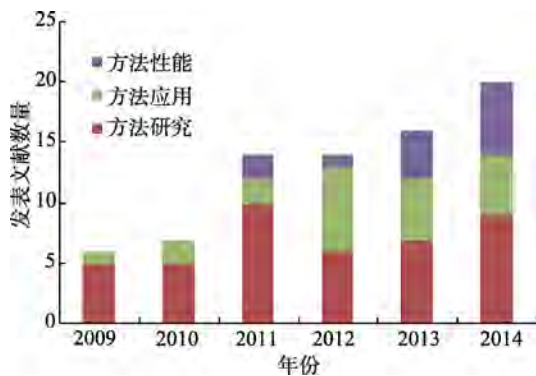


图 4 相关论文发表情况统计

3 面向移动设备的远程渲染的应用分析

本文统计了近 10 年面向移动设备的远程渲染的相关文献, 分析、抽取并统计文献中涉及的应用领域, 如图 5 所示。

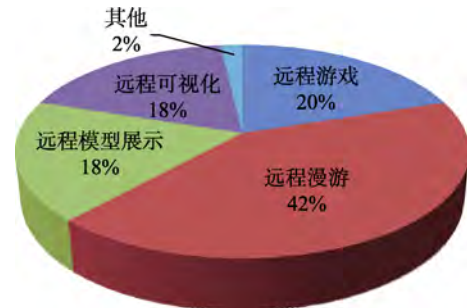


图 5 应用领域统计

1) 远程漫游

远程漫游是一款偏向于规划和旅游的应用, 着重于给用户三维虚拟场景的巡视、简单交互等服务, 以给用户有关场景的整体印象, 从而评估场景建设的好坏。其场景规模偏向于中型或者大型。面向移动设备的解决方法一是采用混合渲染的方式, 另一种方法是使用基于深度图的图像变形方法。

2) 远程游戏

远程游戏一直颇受人们的关注, 尤其是近年来云渲染的兴起, 云游戏这一服务吸引了大多数人的眼球。远程游戏一般使用面向图像的渲染方法向客户端发送视频流, 以实现用户在游戏中的交互。低延迟是远程游戏中一个重要需求, 如在第一人称射击类游戏中, 容许的延迟限制在 100 ms 以内。使用数据压缩(几何压缩、图像压缩和视频压缩)^[33,47]和提升网络带宽可以减少交互延迟。随着网络技术的发展和硬件性能的改进, 现今的远程游戏亦着重于画面质量的提高。已有的云游戏平台有 Rhizome^[48], Gaminganywhere^[24]。

3) 远程可视化

可视化是利用计算机图形学和图像处理技术将数据转换成可见的图像或者图形显示出来, 以揭示数据之间的潜在的逻辑关系, 是一种由无形到有形的方法。由于涉及大量的数据库, 远程可视化一般也采用基于图像的渲染方法。地形可视化是远程可视化一个重要的方面, 能帮助用户准确定位。WebGL 和远程可视化推动可视化在 web 浏览器的实现。Glob3 是一款用于地形可视化的开源框架, 能够在移动设备上和 WebGL 兼容的浏览器上实现

虚拟地球的渲染^[49]。Google Earth 是一款比较成熟的虚拟地球仪软件,使用粗糙几何地形和高质量纹理能达到实时交互。文献[25]使用混合的渲染方法 ParaViewWeb 用于远程渲染, WebGL 用于本地渲染以实现分子模型在移动端的可视化。

4) 远程模型展示

与远程漫游不太相同,远程模型展示通常是单个复杂且精密的几何模型的展示,且该类应用通常涉及数字安全的模型,如文化遗产模型的展示。其矛盾就是既要保证原始几何模型的数据不被恶意窃取,又要为用户提供模型的高清显示,以普及中国文化遗产并方便一些业内学者的学术研究。这种情况下,通常使用脱离原始几何拓扑结构的粗糙网格或点云模型^[17]用于用户交互,从而保证用户交互速度的同时也保护原始几何的数据安全,交互结束后,再给用户当前视角下的高质量的渲染图像,以保证最终的呈现质量。

4 结论和展望

本文从面向移动设备的远程渲染技术的相关概念入手,详细剖析了移动设备的构成和特点,并阐述了远程渲染的基本概念以及面向移动设备的远程渲染的体系结构。与此同时,文中还列举了当前的前沿技术,根据渲染端使用到的数据类型将这些技术按照基于模型的渲染方法、基于图像的渲染方法和混合渲染方法分别讨论,根据分类方法的特性讨论其适应的应用场景,并从分类方法的技术特点、技术研究内容以及方法的应用领域等角度对近几年的研究工作做了深入阐释和分析。

移动设备将在今后相当长一段时间盘踞人们的闲暇时间,并且其发展势头只增不减。随着软硬件技术的发展和网络技术的进步,移动设备可以给人们提供的服务越来越多,具备有高度精美画面、高度复杂场景的应用也不仅仅只在台式机甚至是高性能工作站上才能体验,利用远程渲染等技术,这类应用也将在移动设备上呈现,给用户带来功能更便捷、视觉更华丽的盛宴。

随着云计算和云渲染的兴起,以及移动互联网的加速发展,未来对远程渲染技术的探索还将继续受到重视。下面4个方面可能是未来10年内的研究重点及难点:

1) 硬件方面。移动图形硬件的开发会是图形硬件厂家的一个继续钻研的课题,以期望有一个重

大突破。

2) 渲染方式。移动设备开始配备有 GPU,具备了处理中小型图形场景的功能。面向图像的渲染方式由于在处理并发请求上具有劣势,面向移动设备的远程渲染的渲染工作重心会逐渐移到客户端上,基于服务器的渲染方法被基于客户端和基于混合两端的渲染方法所替代。

3) 应用方向。云游戏给在普通电脑上也能运行大型游戏提供了可能,会在今后一段时间仍是一个应用热点。远程可视化和远程漫游是一个经久不衰的应用方向。远程教育、远程监控、远程支持的应用随着移动设备性能的提升、无线网络的改进、虚拟现实技术的发展和穿戴式设备的演进也会有新的呈现形式。

4) 着重解决的难点。交互延迟和渲染质量仍是面向移动设备的渲染技术一直追求实现的目的。移动图形硬件的进一步发展,探索如何从密集数据中直接渲染——即使用 CPU 和 GPU 配合直接从未解压缩数据中直接渲染场景——或成为一种趋势和技术难点。在远程渲染过程中降低能量消耗从而延长移动设备的电池续航时间也是要着重解决的技术难点之一,除了涉及移动电源上的改进还有远程渲染方法性能的提升。

参考文献(References):

- [1] Schmalstieg D. The remote rendering pipeline-managing geometry and bandwidth in distributed virtual environments[D]. Vienna: Vienna University of Technology, 1997
- [2] ISO/IEC 19775-1—2013 Information technology—computer graphics, image processing and environmental data representation—Extensible 3D (X3D)[S]
- [3] Jovanova B, Preda M, Preteux F. MPEG-4 part 25: a generic model for 3D graphics compression[C] //Proceedings of 3DTV Conference: The True Vision-Capture, Transmission and Display of 3D Video. Los Alamitos: IEEE Computer Society Press, 2008: 101-104
- [4] Eisert P, Fechteler P. Low delay streaming of computer graphics[C] //Proceedings of IEEE International Conference on Image Processing. Los Alamitos: IEEE Computer Society Press, 2008: 2704-2707
- [5] Nave I, David H, Shani A, *et al.* Games@Large graphics streaming architecture[C] //Proceedings of IEEE International Symposium on Consumer Electronics. Los Alamitos: IEEE Computer Society Press, 2008: 1-4
- [6] Laikari A, Fechteler P, Eisert P, *et al.* Games@Large distributed gaming system[OL]. [2015-06-09]. http://iphome.hhi.de/eisert/papers/nem_summit09.pdf
- [7] Deb S, Bhattacharjee S, Patidar S, *et al.* Real-time streaming

- and rendering of terrains[M] //Lecture Notes in Computer Science. Heidelberg: Springer, 2006, 4338: 276-288
- [8] Callahan S P, Bavoil L, Pascucci V, *et al.* Progressive volume rendering of large unstructured grids[J]. *IEEE Transactions on Visualization and Computer Graphics*, 2006, 12(5): 1307-1314
- [9] Lluch J, Gaitán R, Escrivá M, *et al.* Multiresolution 3D rendering on mobile devices[M] //Lecture Notes in Computer Science. Heidelberg: Springer, 2006, 3992: 287-294
- [10] Lazem S, Elteir M, Abdel-Hamid A, *et al.* Prediction-based prefetching for remote rendering streaming in mobile virtual environments[C] //Proceedings of IEEE International Symposium on Signal Processing and Information Technology. Los Alamitos: IEEE Computer Society Press, 2007: 760-765
- [11] Li F W B, Lau R W H, Kilis D, *et al.* Game-on-demand: an on-line game engine based on geometry streaming[J]. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 2011, 7(3): Article No.19
- [12] De Silva R N, Cheng W, Liu D, *et al.* Towards characterizing user interaction with progressively transmitted 3D meshes[C] //Proceedings of the 17th ACM International Conference on Multimedia. New York: ACM Press, 2009: 881-884
- [13] Shi S, Nahrstedt K, Campbell R. A real-time remote rendering system for interactive mobile graphics[J]. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 2012, 8(3s): Article No.46
- [14] Noguera J M, Segura R J, Ogáyar C J, *et al.* A scalable architecture for 3D map navigation on mobile devices[J]. *Personal and Ubiquitous Computing*, 2013, 17(7): 1487-1502
- [15] Diepstraten J, Gorke M, Ertl T. Remote line rendering for mobile devices[C] //Proceedings of the Computer Graphics International. Los Alamitos: IEEE Computer Society Press, 2004: 454-461
- [16] Quillet J C, Thomas G, Granier X, *et al.* Using expressive rendering for remote visualization of large city models[C] //Proceedings of the 11th International Conference on 3D Web Technology. New York: ACM Press, 2006: 27-35
- [17] Su C, Ping J, Yue Q, *et al.* Protected-3DMPS: remote-rendering based 3D model publishing system in digital museum[J]. *Journal of Computational Information Systems*, 2006, 2(1): 277-283
- [18] Ji G F, Shen H W, Gao J. Interactive exploration of remote iso-surfaces with point-based non-photorealistic rendering[C] //Proceedings of IEEE Pacific Visualization Symposium. Los Alamitos: IEEE Computer Society Press, 2008: 25-32
- [19] Liang X H, Zhao Q P, He Z Y, *et al.* A point-based rendering approach for real-time interaction on mobile devices[J]. *Science in China Series F: Information Sciences*, 2009, 52(8): 1335-1345
- [20] Lim M, Kim H S, Lee Y. A distributed stylization mechanism for line extraction process of a 2D image[J]. *Applied Mathematics & Information Sciences*, 2015, 9(21): 471-481
- [21] He Z Y, Liang X H. A point-based rendering approach for mobile devices[C] //Proceedings of the 16th International Conference on Artificial Reality and Telexistence Workshops. Los Alamitos: IEEE Computer Society Press, 2006: 26-30
- [22] Yan Y, Liang X H, Xie K, *et al.* ASEHM: a new transmission control mechanism for remote rendering system[J]. *Multimedia Tools and Applications*, 2014, 69(3): 585-603
- [23] Zhu Biao, Zhang Hongxin, Chen Wei. High resolution expressive rendering[J]. *Journal of Computer-Aided Design & Computer Graphics*, 2015, 27(8): 1398-1406(in Chinese)
(朱 标, 张宏鑫, 陈 为. 高分辨率表意性绘制[J]. *计算机辅助设计与图形学学报*, 2015, 27(8): 1398-1406)
- [24] Huang C Y, Chen K T, Chen D Y, *et al.* Gaminganywhere: the first open source cloud gaming system[J]. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 2014, 10(1s): Article No.10
- [25] Marion C, Pouderoux J, Jomier J, *et al.* A hybrid visualization system for molecular models[C] //Proceedings of the 18th International Conference on 3D Web Technology. New York: ACM Press, 2013: 117-120
- [26] Neven D M. Interactive remote rendering[D]. Delft: Delft University of Technology, 2014
- [27] Giesen F, Schnabel R, Klein R. Augmented compression for server-side rendering[C] //Proceedings of Vision Modeling and Visualization. Boca Raton: CRC Press, 2008: 207-216
- [28] Ghiletiuc J, Färber M, Brüderlin B. Real-time remote rendering of large 3D models on smartphones using multi-layered impostors[C] //Proceedings of the 6th International Conference on Computer Vision/Computer Graphics Collaboration Techniques and Applications. New York: ACM Press, 2013: Article No.14
- [29] Ghiletiuc J, Färber M, Brüderlin B. A highly scalable image-based remote rendering framework[OL]. [2015-06-09]. http://www.transmechatronic.de/uploads/tx_vitramemberadmin/literature/A_Highly_Scalable_Image-Based_Remote_Rendering_Framework.pdf
- [30] Pazzi R W, Boukerche A. PROPANE: a progressive panorama streaming protocol to support interactive 3D virtual environment exploration on graphics-constrained devices[J]. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 2014, 11(1): Article No.5
- [31] Pajak D, Herzog R, Eisemann E, *et al.* Scalable remote rendering with depth and motion-flow augmented streaming[J]. *Computer Graphics Forum*, 2011, 30(2): 415-424
- [32] Shi S, Jeon W J, Nahrstedt K, *et al.* Real-time remote rendering of 3D video for mobile devices[C] //Proceedings of the 17th ACM International Conference on Multimedia. New York: ACM Press, 2009: 391-400
- [33] Shi S, Hsu C H, Nahrstedt K, *et al.* Using graphics rendering contexts to enhance the real-time video coding for mobile cloud gaming[C] //Proceedings of the 19th ACM International Conference on Multimedia. New York: ACM Press, 2011: 103-112
- [34] Vázquez P P, Sbert M. Bandwidth reduction for remote navigation systems through view prediction and progressive transmission[J]. *Future Generation Computer Systems*, 2004, 20(8): 1251-1262
- [35] Bao P, Gourlay D. Low bandwidth remote rendering using 3D image warping[C] //Proceedings of International Conference on Visual Information Engineering. Stevenage: IET Press, 2003: 61-64
- [36] Bao P, Gourlay D. A framework for remote rendering of 3D scenes on limited mobile devices[J]. *IEEE Transactions on Multimedia*, 2006, 8(2): 382-389
- [37] Hsia C H. Improved depth image-based rendering using an adaptive compensation method on an autostereoscopic 3-D display for a Kinect sensor[J]. *IEEE Sensors Journal*, 2015, 15(2):

- 994-1002
- [38] Tizon N, Moreno C, Cernea M, *et al.* MPEG4-based adaptive remote rendering for video games[C] //Proceedings of the 16th International Conference on 3D Web Technology. New York: ACM Press, 2011: 45-50
- [39] Makhinya M. Performance challenges in distributed rendering systems[D]. Zürich: University of Zürich, 2012
- [40] Bao P, Gourlay D. Remote walkthrough over mobile networks using 3-D image warping and streaming[J]. IEE Proceedings Vision, Image and Signal Processing, 2004, 151(4): 329-336
- [41] Shi S, Kamali M, Nahrstedt K, *et al.* A high-quality low-delay remote rendering system for 3D video[C] //Proceedings of the 18th ACM International Conference on Multimedia. New York: ACM Press, 2010: 601-610
- [42] Yoo W, Shi S, Jeon W J, *et al.* Real-time parallel remote rendering for mobile devices using graphics processing units[C] //Proceedings of IEEE International Conference on Multimedia and Expo. Los Alamitos: IEEE Computer Society Press, 2010: 902-907
- [43] Nijdam N A, Tisserand Y, Magnenat T N. Refurbish a single user 3D application into a multi-user distributed service: a case study[C] //Proceedings of the 19th ACM Symposium on Virtual Reality Software and Technology. New York: ACM Press, 2013: 193-200
- [44] Koller D, Turitzin M, Levoy M, *et al.* Protected interactive 3D graphics via remote rendering[J]. ACM Transactions on Graphics, 2004, 23(3): 695-703
- [45] Okamoto Y, Oishi T, Ikeuchi K. Image-based network rendering of large meshes for cloud computing[J]. International Journal of Computer Vision, 2011, 94(1): 12-22
- [46] Noguera J M, Segura R J, Ogáyar C J, *et al.* A hybrid rendering technique to navigate in large terrains using mobile devices[OL]. [2015-06-09]. <http://cgi2010.miralab.unige.ch/short/SP20/SP20.pdf>
- [47] Chuah S P, Cheung N M. Layered coding for mobile cloud gaming[C] //Proceedings of International Workshop on Massively Multiuser Virtual Environments. New York: ACM Press, 2014: 1-6
- [48] Shea R, Fu D, Liu J C. Rhizome: utilizing the public cloud to provide 3D gaming infrastructure[C] //Proceedings of the 6th ACM Multimedia Systems Conference. New York: ACM Press, 2015: 97-100
- [49] Suárez J P, Trujillo A, de la Calle M, *et al.* An open source virtual globe framework for iOS, android and WebGL compliant browser[C] //Proceedings of the 3rd International Conference on Computing for Geospatial Research and Applications. New York: ACM Press, 2012: Article No.22