

基于移动设备的三维图像重建架构及其应用

宋亮, 聂生东

上海理工大学医学影像工程研究所, 上海 200093

【摘要】随着移动互联网的兴起,各种移动设备诸如平板电脑和智能手持设备乃至智能手机已渐渐在医疗行业中崭露头角。目前随着移动设备的快速发展,其硬件性能和软件平台越来越强大,在放射科以及其它临床科室中移动设备也逐渐得到了应用,使真正意义上的移动阅片、移动报告成为可能。三维图像重建是影像诊断和临床手术规划中的重要高级功能,以往需要借助专业工作站,目前在移动设备上也已经可以实现。本文主要综述了在移动设备上实现三维图像重建的几种架构方法,包括基于移动设备硬件进行重建、基于Web技术的服务器的远程重建以及基于移动设备的渐进式重建和体绘制。对上述几种方案的关键技术进行了具体描述,并针对几种架构具体分析了其应用场景,为基于移动设备的开发和应用提供借鉴。

【关键词】移动设备;图像重建;渐进式;体绘制;综述

【中图分类号】R318.5

【文献标志码】A

【文章编号】1005-202X(2016)09-0903-06

Three-dimensional image reconstruction architecture based on mobile device

SONG Liang, NIE Sheng-dong

Institute of Medical Imaging Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China

Abstract: With the development of mobile internet, many kinds of mobile devices, such as tablet and personal digital assistant even smart phone, are more and more popular in healthcare industry. Due to the increasing powerful hardware performance and mature software platform, the mobile device is gradually used in the department of radiology and other clinical departments. Mobile diagnosis and mobile report are becoming possible. The three-dimensional (3D) image reconstruction is an advanced function in imaging diagnosis and clinical surgery planning. The professional desktop workstation was needed for 3D image reconstruction previously, but now 3D image reconstruction is possible on mobile devices. Some kinds of architectures of 3D image reconstruction on mobile devices are briefly introduced in the paper, including reconstruction with mobile device hardware, reconstruction remotely on server and transport on web, and progressive image reconstruction and rendering based on mobile devices. The key technologies are introduced in details and the application scenarios are analyzed to provide reference for the development and application based on mobile devices.

Key words: mobile device; image reconstruction; progressive; volume rendering; review

前言

近年来随着技术进步,IT行业的增长点已经从PC转移到移动设备上。对于医疗行业而言,移动设备的引入可以提高医生的工作效率,改善治疗结果,同时因为设备的可移动性,对医生的工作地点要求大大降低,可以创造很多新的医疗工作模式。

在影像领域,移动设备的应用也在逐渐变为现实。通过移动设备的辅助,医生可以不局限于放射科的固定工作地点参与影像诊断,实现随时随地远程阅片和参与会诊。目前,普通的二维影像在移动设备上的显示已完全可行。通用移动设备受限于屏幕大小,分辨率和灰阶度,以前并没有大量用于诊断应用。2011年,美国食品和药物管理局(FDA)首次批准了苹果iPad/iPhone作为医疗影像诊断放射学的辅助产品使用^[1],为移动设备在放射科诊断的应用打开了大门。而三维图像重建作为影像诊断的高级应用,在诊断和手术规划中都有着重要的作用,因此在移动设备实现三维重建功能具有很重要的意义。本文较为详尽地论述了几种在移动设备上实现三维重

【收稿日期】2016-06-02

【基金项目】国家自然科学基金(60972122);上海市自然科学基金(14ZR1427900);上海市研究生创新基金项目(JW-CXSL1402)

【作者简介】宋亮,硕士研究生,E-mail:2119691@qq.com

【通信作者】聂生东,E-mail:nsd4647@163.com

建的实现架构,同时总结了几种方法的应用场景,提出了今后研究方向的展望。

1 三维图像重建的原理

1.1 体数据

通过CT和MR等影像设备扫描而形成的数据为二维图像数据。以CT为例,CT扫描后形成相应层厚的CT二维图像,将这些图像按顺序叠加起来即形成体数据。在二维图像上的一个网格点我们称为像素,而在体数据三维空间内,每一个网格即为一个小立方体,称为体素(Voxel)^[2]。

1.2 体绘制

体绘制(Volume Rendering)是以体素为基本单元,直接由体数据集生成三维物体的图像^[2],可分为间接体绘制和直接体绘制。直接体绘制算法包括错切-变形算法(Shear-Warp)、抛雪球算法(Splatting)和光线投射算法(Ray-Casting)等。目前最为常用的体绘制算法是光线投射算法,其方法是从每一个体素,沿视角方向发射一条光线,对经过的体素进行插值采样获取采样点的颜色和透明度,同时根据光线吸收模型累加颜色值和透明度,直至光线穿过体数据。光线投射算法中进行颜色累加的公式如下^[3]:

$$C_i = C_{i-1} + (1 - A_{i-1})C_i \quad (1)$$

$$A_i = A_{i-1} + (1 - A_{i-1})A_i \quad (2)$$

其中, A_{i-1} 为前一个位置的透明度, C_i 和 A_i 为当前位置的颜色和透明度。 C_i 和 A_i 为根据当前位置计算出的新的颜色和透明度累加值。而如何将体数据中的体素值或灰度值转化为颜色和透明度是靠一个光线函数进行映射,可以根据实际需要调整(比如可以分别设置一系列的参数值用于在光线投射算法中显示骨骼,造影增强的血管和皮肤等)。光线投射算法的体绘制效果较好,细节清楚,概念容易理解,并且适合于使用GPU(Graphics Processing Unit)来进行加速,因此得到了较为广泛的应用。本文中所描述的三维重建体绘制方法均使用光线投射算法。典型的基于GPU计算的光线投射算法着色器(Shader)实现思路如图1所示。图中的Compositing部分即为光线投射进行颜色累加的关键部分。

在早期,基于光线投射的体绘制都只能基于专用的图像工作站才能进行,国内外研究人员对于提升运算性能和渲染效果进行了不懈的努力。真正使体绘制技术在绝大多数工作站普及得益于计算机图形显卡技术的进步。如前所述,光线投射可适用于GPU计算,但其中很关键的步骤是根据光线路径查

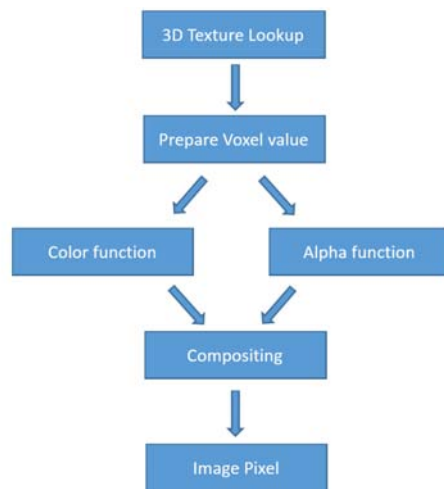


图1 典型光线投射GPU算法实现思路

Fig.1 Workflow of typical ray-casting graphics processing unit algorithm

找所经过的像素点。在GPU未引入三维纹理特性时,这个过程只能通过二维纹理模拟,性能不高。近年来随着计算机硬件的提升和图形显卡的技术进步,尤其是近年来的GPU相继引入了三维纹理特性,使得体绘制性能大大提升,主流PC机配备高性能图形加速卡也可以进行很好的体绘制运算,渲染帧数已经达到实时应用的要求。

2 三维图像重建在移动设备上的实现方法

2.1 基于移动设备的本地图像重建和体绘制

如前所述,想要得到效果满意的本地图像重建和体绘制在早期必须在专业的工作站和带有图形加速器的PC上才能完成,而目前想得到满意的绘制速度使用GPU加速计算更是必须条件。随着目前移动设备上硬件的发展,移动设备GPU其性能和编程API(Application Programming Interface)已渐渐向传统桌面图形加速卡靠拢。但目前移动设备上的GPU编程和传统桌面平台上仍有所区别,目前被广泛支持的是OpenGL ES(OpenGL for Embedded Systems)^[4], OpenGL ES是OpenGL API的子集,针对手机、PDA和游戏主机等嵌入式设备而设计。受限于移动设备的功耗和性能,其所支持的功能相对于同版本的OpenGL有所缩减,但随着OpenGL ES版本的进步,在3.0版本中针对体绘制而言很重要的三维纹理功能终于得到支持^[5]。就具体设备而言,目前苹果iPhone5S之后的设备以及目前市面上的绝大多数Android设备都已经对OpenGL ES3.0提供了良好的支持,因此在GPU编程层面移动设备进行实时体绘制已完全成为可能。

目前最新版本的开源可视化框架VTK 7.0已经

支持在 iOS 和 Android 平台利用 OpenGL ES3.0 进行基于光线投射的体绘制渲染^[6]。以 VTK (Visualiza-

tion Toolkit) 在 iOS 上的体绘制为例, 其绘制流水线如图 2 所示。

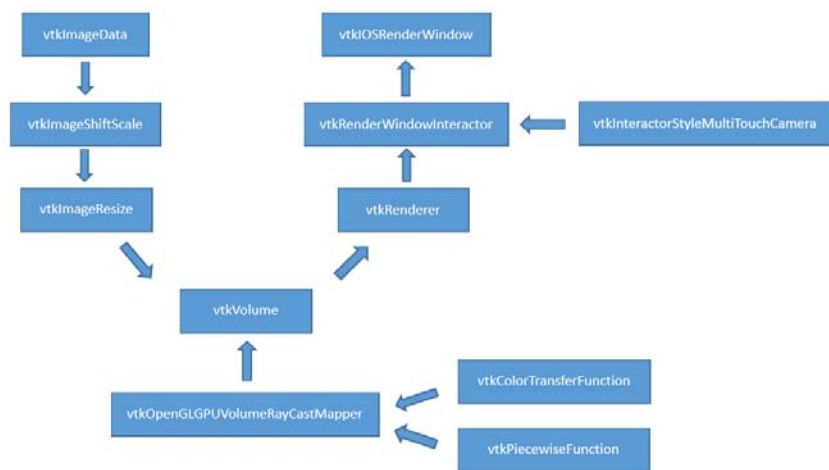


图 2 iOS 下使用 VTK 进行体绘制的流水线
Fig.2 Pipeline of volume rendering on iOS with visualization toolkit

从图中可以看到, 基于 VTK 的体绘制算法流水线基于一系列的过滤器实现, 其中第二步和第三步的 `vtkImageShiftScale` 和 `vtkImageResize` 这两个过滤器分别是用来将原始的体数据进行降位处理(由 CT 原始数据 16 位降为 8 位)和降低分辨率(由 512×512 降为 256×256)重采样处理。这是因为尽管目前移动设备 GPU 的性能比起以前已经有了革命性提升, 但比起桌面 GPU 相比还是有非常大的差距, 同时因为移动设备的屏幕面积相对较小, 屏幕颜色位深比起专业诊断屏幕仍有差距, 作为辅助诊断应用可以为了达到实时性的要求适当对数据大小和位数进行裁剪。另外, 移动设备的内存容量通常小于桌面 PC, GPU 也通常不配备独立显存而和 CPU 共享内存, 所以为节约内存使用, 数据裁剪对大多数设备来说也是一种相对现实的解决方案。

2.2 基于 Web 技术的服务器远程重建

基于 Web 技术的远程渲染是将渲染计算全部放在服务器端实现, 客户端仅仅负责交互操作并获取从服务器得到的渲染数据, 因此这种架构不仅仅应用于移动设备, 在传统的 PC 领域同样采用并大量应用于基于浏览器的解决方案。采用这种方式, 服务器负责进行三维图像重建和体绘制, 而后将绘制完成的渲染结果返回给客户端。这种方式的客户端与服务器交互流程如图 3 所示。

可以看到, 移动设备发起重建请求之后, 服务器会针对客户端实例创建相应的会话, 用于支持之后的操作。移动设备客户端实际上是基于对服务器返回的伪三维图像进行操作。针对客户端的每一次改

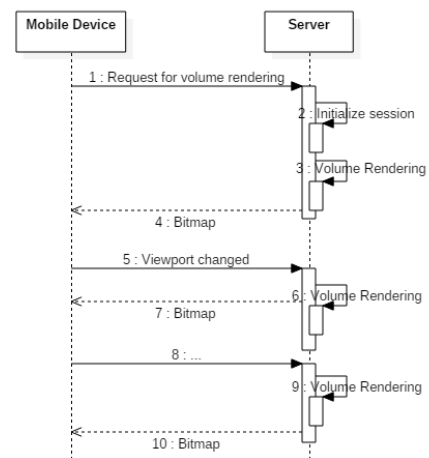


图 3 基于 Web 技术的服务器渲染的通常流程
Fig.3 Typical workflow of web-based server rendering

变视角、状态改变或放大缩小等操作, 都会向服务器发出请求, 服务器会根据客户端要求的参数进行一次体绘制, 然后再将渲染结果以 JPEG 或 PNG 等位图格式返回移动设备客户端, 而移动设备在每次发出请求后需要等待服务器的处理结果并渲染后才能得到更新后的结果。这样的模式将图像重建和体绘制的压力完全转移到了服务器端, 客户端只需要性能很简单的设备即可实现。

实时三维重建和体绘制是一种占用计算资源很多的运算, 和通常意义上的 Web 服务有着很大的不同。并且由于将图像重建的工作搬到服务器端之后, 服务器必然会面对多个用户, 而不同的用户需要进行操作的体数据都相对非常庞大, 因此单个服务器所能承载的用户量非常有限, 有必要对体绘制

服务器架构进行针对性的特殊设计。

如图4所示,目前很多服务器渲染方案如Paraview Web^[7]采用的是由一个调度服务器和多个渲染服务器集群的方式,在移动设备进行第一次请求时,调度服务器会根据当前服务器集群的负载情况分配一个渲染服务器给客户端,并在渲染服务器为该客户端创建独占的渲染实例,同时调度服务器在会话中保存该客户的信息以保证客户端下一次连接到调度服务器之后可以找到其专有渲染实例位于哪台渲染服务器。一旦客户端和渲染实例的直接通信建立,客户端即可向渲染实例发起请求进行图像重建和体绘制渲染。待客户的操作最终结束,渲染服务器即可关闭该次渲染实例,以腾出计算资源给其它的客户使用。这种架构方便灵活,渲染实例可基本复用原来在客户端工作站上已实现的渲染方法,同时可根据当前用户数量,随时动态增减渲染服务器,以满足用户的访问要求。

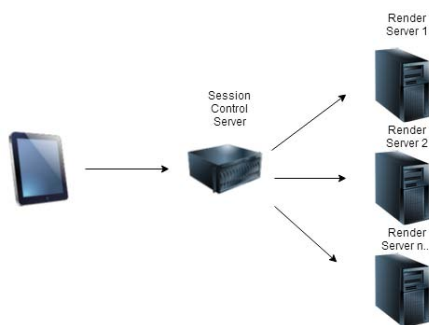


图4 可进行动态调度分配的服务器渲染架构

Fig.4 Architecture of rendering server managed by proxy server

如前文中所述,要提高体绘制性能,需要使用GPU加速,而目前通常的服务器硬件配置和PC工作站不同,一般侧重于CPU和内存,而并不重视图形加速卡的配置。以目前大量采用的刀片式服务器为例,由于体积原因很难配置强大的GPU。另外目前主流的Web服务器大量使用Linux和Unix操作系统,一般只会保留命令行操作方式,而图形化界面又是GPU绘图所必须的,所以这些都会为我们采用GPU加速体绘制带来障碍。因此我们在进行服务器体绘制时需要综合考虑几种体绘制加速方案。以Paraview Web为例,其支持几种典型的应用场景。在国外常见的亚马逊EC2云平台中部署时,通常可以使用Nvidia GRID^[8]这样的虚拟化GPU加速方案,而在更多的没有配备图形加速卡的普通服务器进行部署时,也可以考虑使用OSMesa^[9]。OSMesa是一种基于软件的OpenGL实现,目的是在没有GPU的情况下

使用CPU指令来模拟GPU实现一定程度的GPU加速和离线渲染,VTK和Paraview Web都在编译层面对它提供了比较好的支持。

对于服务器可以独立配置GPU而又需要脱离Linux或Unix X Window图形化环境的情况可以采用EGL规避服务器必须使用图形化界面的问题。EGL (Native Platform Interface)^[10]是位于OpenGL和操作系统窗口系统之间的接口,只要GPU驱动程序支持,它可向我们提供直接离线渲染而不需要图形化界面的能力,VTK在编译层面目前已经对其提供了相应的支持。

传统的基于Web技术的服务器渲染基本都是用基于HTTP的通信方式,所以大多受限于HTTP的GET和POST两种操作方式。这两种方式由于都由客户端主动发起,同时随着服务器返回数据而结束。由于三维重建交互性和实时性的要求较高,仅仅基于HTTP通讯不能满足高实时性的需要。目前有些基于服务器渲染的架构使用了基于Websocket^[11]的通信方式。Websocket是HTML5标准提供的一种通讯协议,它可以实现全双工的通信方式,相比传统的HTTP方案而言,它的通讯开销更小,一旦建立连接即可传输实时数据而不必等某一方发起请求,因此传输实时内容性能提高很多。对比传统的基于HTTP的服务器端绘制方案,采用Websocket协议之后客户端从服务器获取到的渲染结果会更快,更连续,交互体验更出色。

2.3 渐进式和可变层厚的图像重建和体绘制

传统的图像重建和体绘制,不管是基于工作站的体绘制还是前文中所述的基于服务器的重建方案,都是在获取到所有的二维图像之后得到完整的体数据,进行一次性的数据载入后再进行体绘制。但是移动设备的实际应用环境不仅仅是在医院内的局域网,更多情况是在移动互联网。在不确定的网络环境下进行完整体数据下载可能需要耗费相当长的时间,而在数据下载完成之前用户无法看到任何重建结果。另外,由于移动设备屏幕大小和硬件性能参差不齐,对于同一份体数据不同种类的移动设备所最终体现的性能差距可能是非常巨大的。所以有必要研究一种可根据网络环境和硬件性能应用不同策略的图像重建和体绘制方式。

渐进式和可变层厚的体绘制的方法:移动客户端首先评估网络情况和当前的硬件情况,确定渐进绘制的策略,然后分步骤请求策略对应参数的图像来完成几次体绘制,流程如图5所示。以一个包含320幅图像,层厚为1 mm的CT序列为例,制定的绘

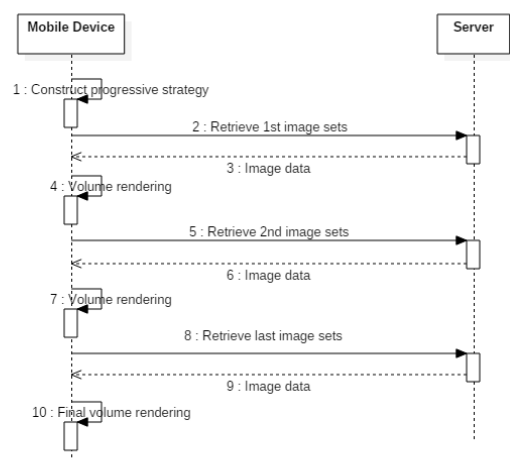


图5 渐进式体绘制流程

Fig.5 Workflow of progressive volume rendering

制策略如表1所示。

移动客户端首先从服务器请求层厚为8 mm的40幅图像,目的是为了尽快完成第一次重建,让用户能看到粗略的渲染结果,之后再次请求层厚为4的80幅图像,完成第2次重建,这次的渲染结果比第1次就有了很大的质量提升,最后一次渲染获取层厚为2的160幅图像完成最终的体绘制渲染结果。

整个渐进式体绘制的显示效果与广泛采用的类似渐进式JPEG的效果类似,由大概轮廓逐渐清晰,直至达到最终的结果。在整个例子中值得注意的是,第3次重建仍然没有获得完整的原始层厚的图像,图像位数和图像分辨率也是降采样过的,这是因为在渐进式体绘制过程中,重建策略是根据设备的

表1 层厚为1 mm、图像数目为320的CT图像序列渐进式重建策略

Tab.1 Progressive image reconstruction strategy for a CT series with 320 images and 1 mm slice thickness

Item	1 st	2 nd	3 rd
Slice thickness	8 mm	4 mm	2 mm
Images	40	80	160
Resolution	256×256	256×256	256×256
Bit of Pixel	8	8	8
Compression	JPEG2000 Lossy	JPEG2000 Lossy	JPEG2000 Lossless

具体情况所制定的,对于大部分移动设备,硬件性能不足以支撑完整数据的体绘制。在本文中的例子中,第3次所获取的图像大小已达到当前设备的处理能力上限,所以第3次体绘制结果就作为最终的体绘制结果。

为实现客户端的渐进式体绘制,服务器必须支持向客户端提供不同分辨率,不同图像位数,和不同层厚的图像。服务器与客户端的通讯协议可以采用目前已广泛使用的WADO^[12]协议,WADO作为DICOM协议的一部分可支持根据HTTP请求的相应参数提供不同的图像格式和分辨率。在实现层面,JPEG2000^[13]作为最新一代的JPEG编码标准提供了在一副图像中保存多个分辨率的能力,同时可以以比较快的速度解码这些不同分辨率的图像,因此也可以作为服务器的实现技术之一。同时因渐进式体绘制的策略包含了不同的层厚,而客户端需要获取的图像层厚实际上有可能和原始的图像数据不同,因此服务器可能需要将相应层厚的图像进行合并和重采样后才能返回客户端,这样额外的算法操作会相应增加服务器的实现复杂性。

3 基于移动设备的图像重建及体绘制的临床应用场景

目前像iPad Pro这样的高端移动设备其本身的性能已基本可以满足本地重建和体绘制的需要,但是一方面由于绝大部分移动设备的屏幕相比桌面工作站仍然较小,另一方面移动设备目前大多都是基于触摸为主的操作方式,这些都决定了移动设备并不适合进行精细操作,所以并不能替代目前高端工作站进行图像分割处理等工作。

基于移动设备本地渲染的性能和渲染效果最接近工作站,交互体验最好,同时由于其可移动的特性,适合给专家级医生用于关键病例的治疗和手术决策用途。专家级医生通常工作忙碌,经常穿梭于手术台办公室、病床和各种会议之间,移动设备契合其使用场景,可在会议,病床旁等各种院内移动场合使用。例如对于肝移植、肝癌或肺癌手术切除以及目前正日益发展的精准介入治疗(如射频和微波消融),基于CT图像的三维重建和器官病灶分割并创建相应手术计划目前已被广泛采用。通过配备性能

强大的移动设备,手术医生可以在院内任何地点使用移动设备浏览完整的图像重建结果,在此基础上可以显示由专业工作站上完成的分割结果等信息,方便医生进行手术计划的调整并对最终手术做出调整和评估,提供完整的临床治疗决策功能。

基于服务器体绘制的一大优点是对设备本身的性能依赖大大降低了,同时对体数据下载的步骤也可省略,因此适合于医院外的工作场景,尤其适合在没有WIFI覆盖,只有移动网络的情况。目前随着移动设备的发展,医生的工作地点可以不局限于医院内部。对于离开医院的医生,非常有可能需要临时参与医院内病人的诊断和治疗决策。对于放射科医生来说,可以使用这种方案在医院外使用移动网络参与诊断,临床医生也可以在院外临时根据需要利用移动网络根据此方案提供的体绘制结果进行治疗评估和临床决策。

渐进式体绘制方案实质上是针对移动设备本地体绘制的一种增强,其提供了根据移动设备的性能和网络状况动态调整体绘制结果的机制,所以可适用于绝大多数移动设备和各种网络状况。例如在基于移动设备的远程会诊、协同读片等协作场合,对于实时性的要求较高,同时需要在图像和基于重建的结果上进行一系列的操作。但在会诊时双方所使用的移动设备种类可能并不相同,所以可使用渐进式的方案来针对性满足,从而加强使用体验。渐进式体绘制的更多应用场景还需要在产品开发和实际应用尝试中进一步完善。

总体来说,以上几种方案在实际应用中可以根据移动设备的性能和实际应用场景采用不同的方案,更多的应用于具体的临床实践中还需要进一步研究。

【参考文献】

- [1] FDA: FDA clears first diagnostic radiology application for mobile devices. Provides wireless access to medical images for iPhone, iPad users [EB/OL]. (2011-02-04) [2015-11-04]. <http://www.fda.gov/NewsEvents/Newsroom/PressAnnouncements/ucm242295.html>.
- [2] 罗述谦,周果宏. 医学图像处理与分析[M]. 2版. 北京: 科学出版社, 2010: 180-190.
LUO S Q, ZHOU G H. Medical image processing and analysis [M]. 2nd ed. Beijing: Science Press, 2010: 180-190.
- [3] MARKUS H, PATRIC L, CHRISTOF R S, et al. Advanced illumination techniques for GPU- Based volume raycasting [C]. ACM SIGGRAPH ASIA 2008 courses, 2008: 10-19.
- [4] Khronos™ Group. The standard for embedded accelerated 3D graphics [EB/OL]. [2015-12-09]. <https://www.khronos.org/opengles/>.
- [5] Khronos™ Group. OpenGL ES 3.0 specification [EB/OL]. [2015-12-09]. <https://www.khronos.org/registry/gles/#specs3>.
- [6] MARTIN K, GOODLETT C, CHAUDHARY A, et al. Hanwell and jean-christophe fillion-robi: volume rendering on iOS and android [EB/OL]. [2015-12-18]. <https://blog.kitware.com/volume-rendering-on-ios-and-android>.
- [7] Kitware Inc. The paraview web homepage [EB/OL]. [2015-12-08]. <http://paraviewweb.kitware.com/>.
- [8] Nvidia Inc. Nvidia GRID virtual GPU technology [EB/OL]. [2015-12-12]. <http://www.nvidia.com/object/grid-technology.html>.
- [9] Mesa3d. org. The mesa 3D graphics library [EB/OL]. [2015-12-10]. <http://www.mesa3d.org/osmesa.html>.
- [10] Khronos™ Group. Native platform interface [EB/OL]. [2015-12-11]. <https://www.khronos.org/egl>.
- [11] Internet Engineering Task Force. The websocket protocol [EB/OL]. [2015-12-13]. <https://tools.ietf.org/html/rfc6455>.
- [12] National Electrical Manufacturers Association. DICOM PS3.18 2016b-web services [EB/OL]. [2015-12-12]. <http://dicom.nema.org/medical/dicom/current/output/html/part18.html>.
- [13] Joint Photographic Experts Group. Jpeg2000 [S/OL]. [2015-12-13]. <https://jpeg.org/jpeg2000/>.
- [14] 诸葛斌. 医学图像三维重建[M]. 杭州: 浙江工商大学出版社, 2008: 6-30.
ZHU G B. 3D reconstruction of medical image [M]. Hanzhou: Zhejiang Gongshang University Press, 2008: 6-30.
- [15] 高鹏, 刘鹏, 苏红森, 等. 基于HTML5与可视化工具包的医学影像三维重建及交互方法研究[J]. 生物医学工程学杂志, 2015(2): 336-342.
GAO P, LIU P, SU H S, et al. Research on three-dimensional medical based on HTML5 and image reconstruction and interaction visualization toolkit [J]. Journal of Biomedical Engineering, 2015(2): 336-342.
- [16] 王召伟, 陈雷霆, 房春兰. 基于Web模式的医学图像三维重建与切割[J]. 计算机应用研究, 2009(2): 789-791.
WANG Z W, CHEN L T, FANG C L. Medical image 3D-reconstruction and volume-clipping based on web [J]. Application Research of Computers, 2009(2): 789-791.
- [17] FERNANDO R. GPU gems: programming techniques, tips and tricks for real-time graphics [M]. Addison-Wesley Professional, 2004.
- [18] THIRION T. Building VTK for mobile architectures [EB/OL]. [2015-12-17]. <https://blog.kitware.com/building-vtk-for-mobile-architectures>.
- [19] Kitware Inc. Visualization homepage [EB/OL]. [2015-12-08]. www.vtk.org.
- [20] 王冠宇, 杨弋. 基于三维纹理的体绘制研究与实现[J]. 电子科技, 2012(25): 12-14.
WANG G Y, YANG Y. Research and Realization of Three-dimensional Texture Volume Rendering [J]. Electronic Science and Technology, 2012(25): 12-14.
- [21] Khronos™ Group. Khronos releases OpenGL ES 3.0 specification to bring mobile 3D graphics to the next level [EB/OL]. [2015-1-12]. <https://www.khronos.org/news/press/khronos-releases-opengl-es-3.0-specification>.

(编辑: 薛泽玲)