

Web-Based Medical Volume Visualization System

Shuai Yang, Zhaoming Liu, Yongzhen Ke

School of Computer Science & Software Engineering, Tianjin Polytechnic University, Tianjin
Email: yangshuai@tjpu.edu.cn

Received: Jun. 4th, 2018; accepted: Jun. 20th, 2018; published: Jun. 28th, 2018

Abstract

The client-based medical image 3D visualization system can only work on a specified platform, which limits the promotion of the medical image visualization system and raises the cost of using the system. Based on WebGL technology, we implement a plug-in-free medical image visualization system. The ray casting algorithm is one of volume rendering methods. But traditional ray casting algorithm requires a large number of interpolation operations, and the three-dimensional reconstruction speed is slow. We improve the traditional ray casting algorithm by selecting appropriate sampling points, optimizing the interpolation operation and early ray termination, which have increased the speed of 3D reconstruction and meet the needs of real-time interaction.

Keywords

Medical Volume, Visualization, Web, Ray Casting

基于Web的医学体数据可视化系统

杨 帅, 刘兆明, 柯永振

天津工业大学计算机科学与软件学院, 天津
Email: yangshuai@tjpu.edu.cn

收稿日期: 2018年6月4日; 录用日期: 2018年6月20日; 发布日期: 2018年6月28日

摘 要

基于客户端的医学图像三维可视化系统仅能运行于指定的单一平台,限制了医学图像可视化系统的推广,提高了系统的使用门槛。本文基于WebGL技术,实现了一个无插件的医学图像可视化系统。体数据渲染常用的一种方法为光线投射算法,传统的光线投射算法需要进行大量的插值运算,三维重建速度较慢。

本文通过对传统的光线投射算法进行改进,从选取采样点、插值运算、提前光线截止等方面进行改进,从而提高了三维重建的速度,基本满足实时交互的需要。

关键词

医学体数据, 可视化, Web, 光线投射算法

Copyright © 2018 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

1. 引言

随着计算机技术的飞速发展,计算机可视化成为一门新兴的计算机图形学方面重要的应用学科,它将抽象的数学符号转化成肉眼能够直接观察到的图像,使研究学者更加方便观察结构。医学图像三维重建[1]在医学诊断中扮演着越来越重要的角色,利用计算机图形学技术将CT、MRI等数据图像成像技术所获取的人体信息利用计算机重构为三维图像,并在计算机中显示出来,在临床治疗,手术和医学教学方面等有很大的应用前景。现阶段医学体数据可视化方法较多,国际上应用比较成熟且广泛的主要包括:

1) 基于商业数据可视化软件二次开发,利用其提供的接口实现,如Amira [2]、Mimics [3]等。其软件十分成熟,能够快速、高质量地实现对医疗体数据的可视化;但是商业软件的开发与使用成本非常昂贵,客户端必须安装相应的软件环境才能实现对应的功能。

2) 基于开源软件包开发,如VTK [4]、ITK [5]、MITK [6]或OpenGL [7]等,配合C语言等编程语言在桌面程序环境中实现建模与绘制。这种方法具有开源免费的优点,能够灵活实现医疗体数据的可视化;但开发的桌面程序不具备跨平台性,软件的升级与维护困难,提高了用户的使用成本。

可见传统的医疗体数据可视化方法依赖于桌面环境,开发、维护和使用成本较高。随着HTML5 [8]标准的提出与大规模应用,基于浏览器的WebGL [9]图形技术提供了优良的三维可视化途径。与传统客户端方法相比,WebGL具有以下优势:

1) 平台兼容性。网络平台支持各种操作系统,在代码上实现了统一化开发,用户可以自由选择使用环境,特别是便携式移动终端技术的发展(如智能手机和平板电脑等),跨平台性的优势更加明显。

2) 开发灵活性。WebGL是免费开源的技术标准,减低了开发成本,对代码的升级和系统的维护都是在服务器端统一处理,减少了客户端的负担,并且网络平台很容易集成到其他的一些业务系统中,可以无缝地为用户提供更加强大的功能。

3) 用户体验性。网络平台拥有良好的用户体验模式,用户像浏览网页一样使用可视化系统,并且用户可以在移动终端上随时随地使用可视化系统,极大地方便了用户。

体数据可视化有光线投射法、抛雪球方法和频域绘制法[10],其中光线投射法[10]是应用最广泛的方法。本研究基于WebGL,使用光线投射算法实现医疗体数据的可视化。然而医疗体数据的数据量比较大,并且光线投射算法计算复杂度较高,传统方法无法实现实时绘制。因此本文基于WebGL提出一种改进的光线投射算法,并且以此为基础提出一套医学体数据可视化系统。

2. 系统框架和实现

WebGL的绘制流程延续OpenGL ES2.0的绘制流程[11]。图形中的任意一个顶点都包含多个属性例

如：颜色、纹理坐标、法向量等，这些属性称之为顶点属性。这些数据流依次经过顶点着色器、元图组光栅化、片元着色器、单片元操作最后到达图形帧缓存，用于在 Canvas 元素中显示，如图 1 所示。其中顶点着色器和片元着色器在 GPU 中运行。

本系统基于 WebGL 开发，是一个轻量级的可视化应用。用户可以上传医学体数据进行可视化，系统支持的体数据格式为 DICOM (Digital Imaging and Communications in Medicine)格式。系统包含 5 个子视图，其中有两个为三维视图，三个二维视图。三个二维视图分别是：横截面视图、矢状面视图和冠状面视图，每个二维视图中都有一个十字形的坐标指示器，用于标识当前各个二维面所处的位置。两个三维视图分别是：多层面视图和三维可视化视图，多层面视图主要展示三个二维视图之间的空间位置，三维可视化视图用于展示使用光线投射算法得到的可视化结果。这些视图的上方有一个悬浮层，提供一些简单的设置功能。系统的 UI 框架如图 2 所示。

该系统的大部分可视化逻辑和行为都是有 viewerjs 库提供的。这个库依赖于一个底层的数据可视化组件 XTK (The X Toolkit: WebGL for Scientific Visualization)和一个医学图像处理库 AMI (Medical Imaging JavaScript ToolKit) [12]。底层的绘制和交互由 threejs 和 WebGL 提供支持。系统的逻辑框架如图 3 所示。

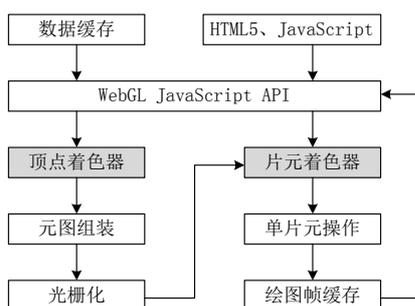


Figure 1. WebGL rendering process

图 1. WebGL 图形渲染流程



Figure 2. UI framework of the system

图 2. 系统 UI 框架

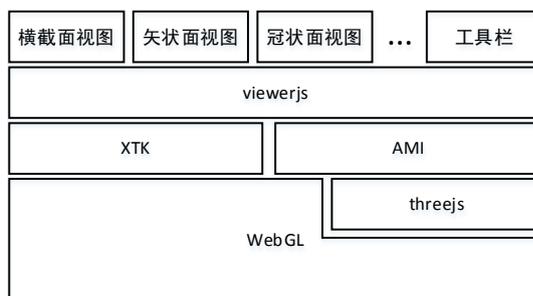


Figure 3. Logical framework of the system

图 3. 系统逻辑框架

3. 改进的光线投射算法

为了解决传统的光线投射算法三维重建速度较慢这个问题，本文提出了一种改进的光线投射算法，从选取采样点、插值运算、提前光线截止等三个方面进行改进。首先本文在选取采样点时通过使用加速度步长采样方法确定采样点并通过改进插值算法计算采样点值，最后采用光线提前截止法进一步提高三维重建速度。通过 WebGL 在片段着色器中实现上述改进算法，对比传统光线投射算法起到一定加速作用，绘制速度明显提高，图 4 为基于 WebGL 改进之后的光线投射算法流程图。

3.1. 加速度步长采样

传统的光线投射算法，延射线方向向体数据场进行等距离步长采样。运用这种取样方法，会取到许多无用的体素，同时也得对这些无用采样点进行插值运算，这些无用的插值运算，严重影响三维重建速度。一条射线上，越靠近射线起点的采样点对图像重建贡献越大，越靠后的采样点对图像重建贡献越小。所以在射线上可以适当多取靠近前侧的采样点，少取靠后的采样点。使用加速度步长采样方法正好迎合了这种选取采样点思想，减少选取部分采样点，以此来提高运行速度，加速度步长采样点选取如图 5 所示。

本文采用从前到后合成方式，在一条射线上靠近前端的体素对三维重建作用大，远离前端的体素对三维重建作用小，本文提出采用加速度方式进行采样如图 6 所示，越靠近射线后端，采样步长越大，通过采用此方法减少重采样点的选取来提高重建速度。 L_1 为初始采样间距， d 为加速度步长变化因子。

第 k 个体素与第 $(k + 1)$ 个体素之间距离为 $L_1 + (k - 1)d$ ，假设一条射线上有 560 个重采样点，初始采样间距为 L_1 ，则总长度为 $560 L_1$ 。当 $d = 0.004$ 时，运用加速度算法计算知：需要选取 505 的重采样点，相对于等距离采样要少选取 55 个采样点，少取约 9.8% 的采样点。

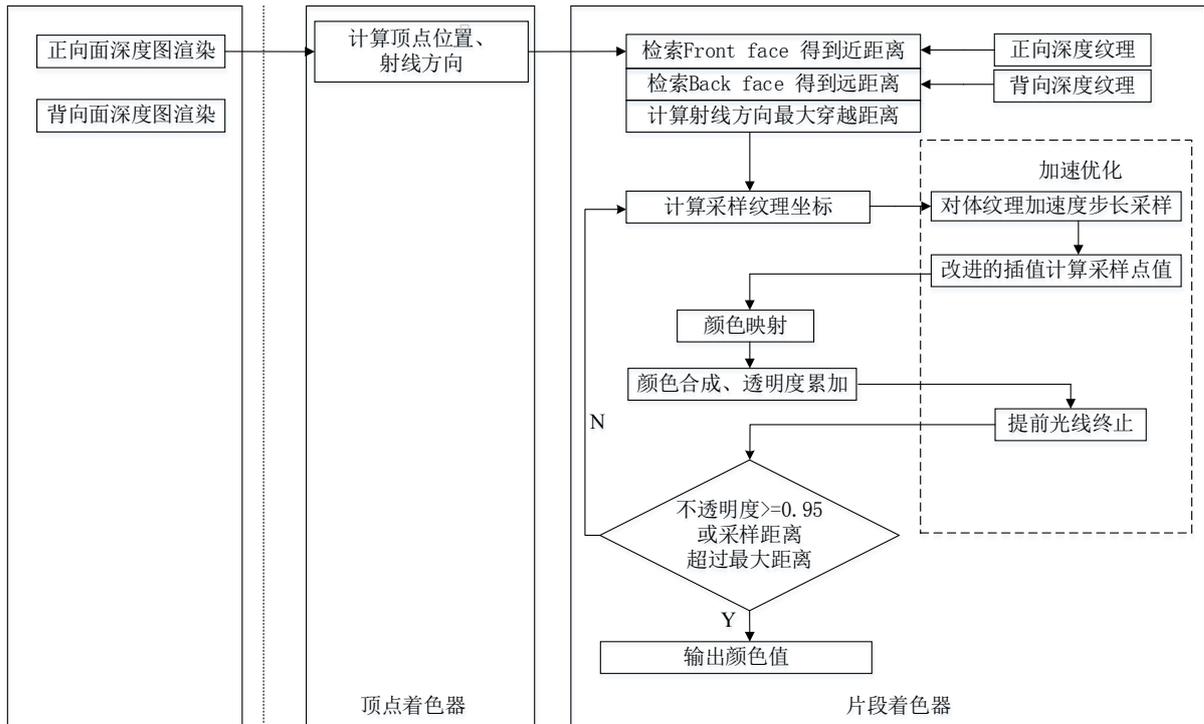


Figure 4. Improved ray casting algorithm flow
图 4. 改进的光线投射算法流程

3.2. 改进插值算法实现

在获得到采样点位置之后, 需要通过采用三线性插值计算获取重采样点的值。首先将一条射线上的所有采样点分为若干组, 每 2 个采样点分为一组, 对每一组前 1 个采样点采用三线性插值计算采样点的值, 由于相邻的两个采样点不透明度值和颜色值差别很小, 可以分别将这两个采样点的不透明度值和颜色值设为相同的数值, 即将每一组中的前一个采样点的值, 赋给后一个采样点, 以此来减少计算量, 提高插值运算速度。例: 将采样点 M 和 X_0 分为一组, X_1, X_2 分为一个组, 对采样点 M, X_1 采用三线性插值计算, 将采样点 M, X_1 计算得到的数值分别赋值给采样点 X_0, X_2 。按照此种方法可以得出一条射线上只有 1/2 的采样点进行了三线性插值运算, 剩余 1/2 的采样点不需要进行三线性插值运算, 这大大减少了数据运算量。

3.3. 光线投射提前截止法

在像素混合迭代求解的过程中, 不透明度 α 的值是逐渐增大的, 当 α 的值接近于 1 时, 表示该光线对像素值的计算结果趋近于饱和, 即使继续采样, 最终的像素值也几乎不会受到影响。因此, 可以采用提前终止光线计算的方法, 这样可以提高整体绘制的时间。设置一个阈值, 只要 α 的值超过这个阈值, 就终止光线, 本文设置的阈值为 0.95。

4. 实验结果与分析

本文实现了一个医学体数据可视化系统, 该系统能够可视化 DICOM 格式的医学体数据, 可以通过鼠标进行放大/缩小、旋转、平移和更改切片位置等操作。该系统在 PC 和平板电脑上流畅地运行(测试使用的浏览器为 Google Chrome, 版本号为 64.0.3282.186), 如图 7 所示。

为了验证本系统的性能, 本文使用 A、B、C 三个人体头部 CT 数据作为输入数据。数据的分辨率分别为 $364 \times 364 \times 300$ 、 $480 \times 480 \times 265$ 和 $536 \times 536 \times 300$ 。本文从三维重绘效果和重建速度两个方面进行评价。

基于 WebGL 传统光线投射算法三维重建效果如图 8 所示。

本文算法的重建效果如图 9 所示。

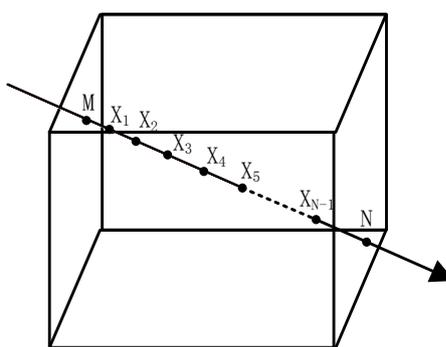


Figure 5. Acceleration step sampling

图 5. 加速度步长采样

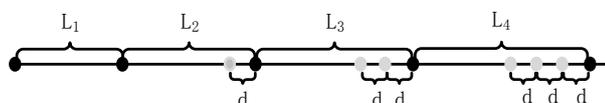


Figure 6. Acceleration sampling distance

图 6. 加速度采样距离

从图 8 和图 9 可以看出, 本文改进算法所实现的三维重建效果与文献[12]算法的重建效果相似, 都可以实现高质量的三维重建图像。

从表 1 中数据可知, 两种重建算法所需的时间与数据分辨率有关。从表中数据可以明显看出, 本文算法所需的时间更短, 并且在数据量变大时, 这种优势更加明显。

从以上两个实验可以看出, 本系统在绘制效果基本不变的情况下, 提高了绘制的速度, 基本满足实时交互的需要。



Figure 7. The result of the system running on multiple platforms
图 7. 系统在多平台运行的结果

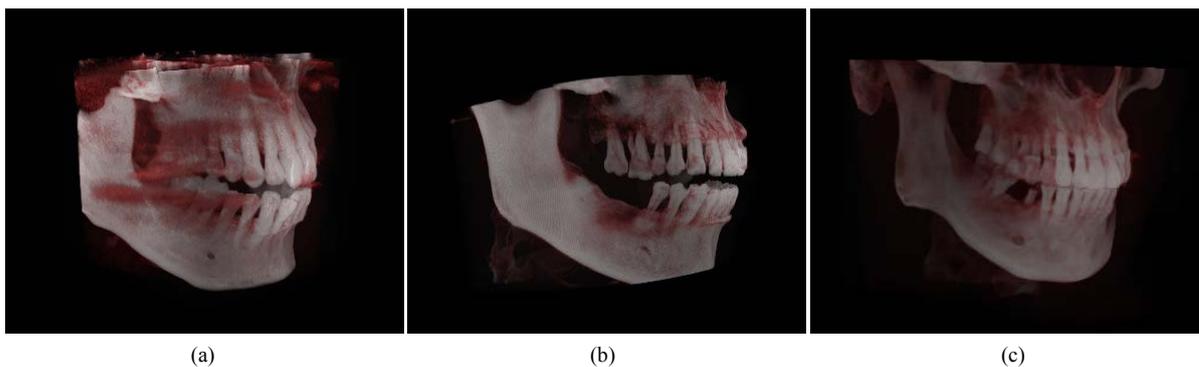


Figure 8. 3D reconstruction based on WebGL traditional ray casting algorithm
图 8. 基于 WebGL 传统光线投射算法三维重建效果

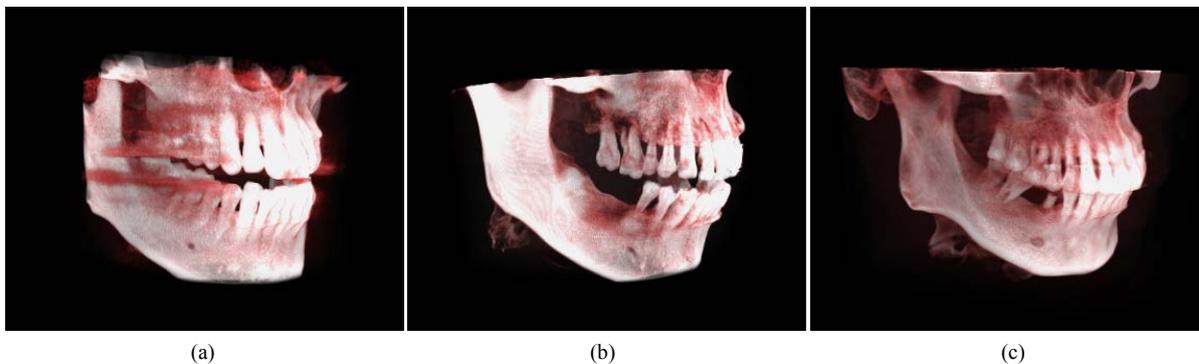


Figure 9. 3D reconstruction based on our method
图 9. 基于本文算法三维重建效果

Table 1. 3D reconstruction time
表 1. 三维重建时间对比

数据集	数据分辨率	切片数量/张	绘制时间/s	
			文献[12]的算法	本文的算法
A	364 × 364 × 300	336	2.74	2.01
B	480 × 480 × 265	336	3.90	2.73
C	536 × 536 × 300	336	5.04	3.36

5. 实验结果与分析

本文基于 WebGL 和一些开源组件, 实现了 WEB 端的实时医学体数据可视化系统的开发。该系统能够运行于 PC 和 PAD 等移动终端平台, 实现了一次开发多平台运行的目的。针对 web 低效率的问题, 本文提出一种改进的光线投射算法, 从选取采样点、插值运算、提前光线截止等方面进行改进。该算法在保证绘制质量的同时提高了绘制效率, 保证了系统可实现基本的实时交互。

然而本系统功能仍然有限, 今后可以考虑添加其它体数据可视化方法, 为系统添加实时沟通讨论功能等。

基金项目

天津市自然科学基金项目(17JCTPJC46800)。

参考文献

- [1] Zheng, L., Li, G. and Sha, J. (2007) The Survey of Medical Image 3D Reconstruction. *Proceedings SPIE 6534, 5th International Conference on Photonics and Imaging in Biology and Medicine*, Wuhan, 65342K.
- [2] Stalling, D., Westerhoff, M. and Hege, H.C. (2005) Amira: A Highly Interactive System for Visual Data Analysis. *Visualization Handbook*, 27, 749-767. <https://doi.org/10.1016/B978-012387582-2/50040-X>
- [3] Materialise (2018) 3D Medical Image Processing Software|Materialise Mimics. <http://www.materialise.com/zh-hans/medical/software/mimics>
- [4] Putanowicz, R. (2007) Visualization of Large Data Sets by Mixing Tcl and C++ Interfaces to the VTK Library. Pergamon Press, Inc., Oxford.
- [5] Yoo, T.S., Ackerman, M.J., Lorensen, W.E., Schroeder, W., Chalana, V., Aylward, S., *et al.* (2002) Engineering and Algorithm Design for an Image Processing Api: A Technical Report on Itk—The Insight TOOLKIT. *Stud Health Technol Inform*, 85, 586-592.
- [6] German Cancer Research Center (2018) The Medical Imaging Interaction Toolkit (MITK). <http://mitk.org/wiki/MITK>
- [7] Khronos Group (2018) OpenGL—The Industry Standard for High Performance Graphics. <https://www.opengl.org/>
- [8] W3C (2018) HTML5: A Vocabulary and Associated APIs for HTML and XHTML. <http://www.w3.org/TR/html5/>
- [9] Khronos Group (2018) WebGL-Opengles 2.0 for the Web. <http://www.khronos.org/webgl/>
- [10] Zhang, Q., Eagleson, R. and Peters, T.M. (2011) Volume Visualization: A Technical Overview with a Focus on Medical Applications. *Journal of Digital Imaging*, 24, 640-664. <https://doi.org/10.1007/s10278-010-9321-6>
- [11] Ginsburg and Dan (2009) The OpenGL ES 2.0 Programming Guide. Addison-Wesley, Boston.
- [12] Arbelaz, A., Moreno, A., Kabongo, L. and Garcia-Alonso, A. (2017) Volume Visualization Tools for Medical Applications in Ubiquitous Platforms. *eHealth 360°*. Springer International Publishing, Berlin.

知网检索的两种方式：

1. 打开知网页面 <http://kns.cnki.net/kns/brief/result.aspx?dbPrefix=WWJD>
下拉列表框选择：[ISSN]，输入期刊 ISSN：2161-8801，即可查询
2. 打开知网首页 <http://cnki.net/>
左侧“国际文献总库”进入，输入文章标题，即可查询

投稿请点击：<http://www.hanspub.org/Submission.aspx>

期刊邮箱：csa@hanspub.org