

基于 WebGL 的三维 WebGIS 场景实现

朱丽萍^{1,2}, 李洪奇^{1,2+}, 杜萌萌^{1,2}, 王莹^{1,2}

(1. 中国石油大学(北京)地球物理与信息工程学院, 北京 102249; 2. 中国石油大学(北京)石油数据挖掘北京市重点实验室, 北京 102249)

摘要: 寻求一种不需要组件加载来实现三维 WebGIS 场景的解决方案, 对突破目前三维 WebGIS 缺乏高效前端显示技术的瓶颈具有重要意义。将 WebGL 作为图形引擎, 采用真实空间数据作为验证和测试数据, 使用 JavaScript 编写一套较完整的三维 WebGIS 场景及图层元素类库, 设计简单的数据服务, 形成一个简单的 GIS 系统。实验验证了 WebGL 是一种可行的实现零组件三维 WebGIS 场景构建的技术手段。

关键词: 三维; 地理信息系统; 网页地理信息系统; 网页三维技术; 网页开放图形库

中图分类号: TP311.5 **文献标识码:** A **文章编号:** 1000-7024 (2014) 10-3645-06

Implementation of WebGL-based 3D WebGIS scene

ZHU Li-ping^{1,2}, LI Hong-qi^{1,2+}, DU Meng-meng^{1,2}, WANG Ying^{1,2}(1. College of Geophysics and Information Engineering, China University of Petroleum (Beijing), Beijing 102249, China;
2. Beijing Key Laboratory of Petroleum Data Mining, China University of Petroleum (Beijing), Beijing 102249, China)

Abstract: Finding a solution to realize the rendering of the three-dimensional WebGIS scene with none plug-in has great significance in breaking through the bottleneck of the three-dimensional WebGIS front-end graphic technology. A more complete class library for the three-dimensional scene and layer items of the GIS was programmed by taking the WebGL as the graphics engine. A series of services were designed to meet the needs on reading space data and a simple geographic information system was formed at last. The test using real space data as check and test data proves that WebGL is a feasible and effective technology in building the three-dimensional WebGIS scene with none plug-in.

Key words: 3D; geographic information system; WebGIS; Web3D; WebGL

0 引言

受制于 Web3D 技术的瓶颈, 传统的用以实现三维 WebGIS 场景的 Web3D 技术, 包括 Flash (Flex)^[1]、Java3D^[2]、X3D^[3] 等技术, 都需要加载组件或者预先安装客户端 (或插件), 而且具有显示效果不佳、兼容性差等缺陷。WebGL 作为新一代 Web3D 图形标准, 具有不需要组件加载、渲染效果好、兼容性好等优点并且取得了众多浏览器生产厂商的支持, 具有以往技术无法比拟的优势。本文在前人所做的众多探索的基础上, 采用 WebGL 作为图形引擎来展开无组件三维 WebGIS 场景系统的实现, 比较了各种技术之间的优缺点, 实验验证了 WebGL 是一种非常有效的实现三维 WebGIS 场景渲染的技术手段。

HTML5 标准发布以来, 已经有不少学者开展了基于此标准的 GIS 应用研究^[4], 但是多是以 canvas 的 2D 环境为图形基础, 无法满足三维 WebGIS 的应用要求, 作为 canvas 的 3D 环境, WebGL 具有能够直接调用硬件图形引擎, 兼容性好等多种优点。

展开这一研究有着较大的现实意义和长远的工程学意义。首先进行这项研究具有短期的现实意义, 如果能够找到一种不需要预先安装客户端或者加载组件的方法, 可以避免大组件的安装, 用户不再需要等待较长时间, 大大提高了用户体验。其次进行这项研究有深远的工程意义。人们一直以来都在进行三维 WebGIS 解决方案的研究和探索。目前 WebGL 已经展现出了强大的生命力, 国内外包括谷歌、诺基亚在内的多家企业都已经展开了基于 WebGL 的

收稿日期: 2013-10-29; 修订日期: 2014-01-21

基金项目: 十二五国家重大专项基金项目 (2011ZX05020-007-007); 中国石油大学 (北京) 基金项目 (KYJJ2012-05-25)

作者简介: 朱丽萍 (1973-), 女, 山东青岛人, 硕士, 副教授, 研究方向为软件集成; +通讯作者: 李洪奇 (1960-), 男, 辽宁铁岭人, 博士, 教授, 研究方向为软件集成; 杜萌萌 (1986-), 河南三门峡人, 硕士研究生, 研究方向为地理信息系统; 王莹 (1990-), 女, 安徽淮北人, 硕士研究生, 研究方向为软件集成。E-mail: hq_li@cup.edu.cn

GIS 展示端研究工作, 有些公司甚至已经推出了相关 GIS 案例展示。如果本研究能够证实 WebGL 是一个有效的实现 Web3D 场景构建的方式, 将给 WebGIS 这一领域找到一种新的实现浏览器端三维场景构建的途径, 而且摒弃传统技术的弊端, 具有较高的商业和工程意义。

1 需求定义

本文以三维 WebGIS 的展示功能作为主要的功能性需求, 所设计的三维 WebGIS 场景展示系统应具有如下功能:

(1) 展示一个 Web 端的三维 GIS 基本场景, 所有的 GIS 图层元素将加载在基本场景上。

(2) 场景能够进行 360° 的全方位展示和移动, 视角能够进行放大和缩小。

(3) 系统应该能够实现所有基本的 GIS 图层元素的生成、加载、卸载, 按照一般 GIS 系统的要求, 所能加载的 GIS 元素包括如下几种: 中英文文字标注, 主要用于地名、兴趣点等的标注; 图形标注, 主要用于与文字标注相结合进行地点标示; 矢量线, 主要用于道路、河流、轨迹、境界等的标示; 矢量面, 主要用于区块、行政区域、湖泊等的标示; 三维模型, 主要用于在场景中加载建筑、植被、公共设施、地形等; 影像切片, 主要用于基本地表的绘制。

本文的出发点是为了解决现有的三维 WebGIS 场景都需要加载或者预先安装组件来实现的问题, 以提高用户体验和交互性, 因此系统应该具有如下的非功能性需求:

- (1) 不能加载任何组件或者插件;
- (2) 具有较好的用户体验。

2 系统设计

目前业界已经有几个较为成熟的基于 WebGL 的 JavaScript 三维图形框架, 较为流行的有 Three.js、Oak3D^[5] 等。为更加适用于 WebGIS 开发需求和减小脚本体积, 本人决定抛弃这些庞大的框架, 按照面向对象设计的理论, 以 JavaScript 为编程语言^[6], 建立一套应用与网页的三维图形库。

2.1 渲染流程

WebGL 编程通过调用内嵌于浏览器中的 WebGL 上下文环境中的图形接口进行图形绘制, 因此不再需要任何组件的加载。浏览器直接通过 GPU 进行硬件图形加速。使用 WebGL 进行图形绘制的流程如图 1 所示。首先, 获取 WebGL 上下文环境, 根据着色器代码片段创建着色器, 一般包括顶点着色器 (VertexShader) 和片元着色器 (FragmentShader), 将着色器绑定给 WebGL 上下文, 链接之后生成 WebGL program。接着为需要绘制的图形对象创建顶点数组与颜色数组, 如果使用索引方式构建面的话还要使用顶点索引数组, 如果使用贴图的话还要生成纹理对象以及顶点纹理坐标数组等。然后为 WebGL 上下文绑定着色器

需要的这些数组以及纹理对象, 同时要为 WebGL 上下文设置着色器需要的变量, 包括投影矩阵、模型视图矩阵等。最后调用 WebGL 的基本图形绘制方法进行图形对象的绘制。

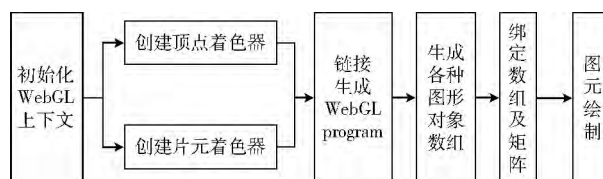


图 1 WebGL 绘制流程

2.2 模块划分

为使图 1 中的绘制流程适用于 WebGIS 场景的绘制, 系统需要设计相应的顶点着色器以及片元着色器, 以及用于数学向量矩阵运算的函数库, 和能够进行各种 GIS 场景元素创建和绘制的 GIS 图层元素类库。因此系统将包括数学运算模块、着色器、图层元素模块等 3 个主要部分。

3 详细设计

3.1 数学运算模块

图 2 是提供了基本图形计算功能的数学运算模块的类图。该模块包括经纬度坐标类 Point2D 和三维空间坐标类 Point3D, 它们提供了经纬度坐标以及三维空间坐标的相互转换。也包括四维向量类 Vector4f, 提供向量乘法和加法运算。四阶矩阵类 Matrix44f, 提供了四阶矩阵的乘法运算以及逆运算。同时该模块包含了透视投影矩阵类 ProjPerspective 和正投影矩阵 ProjOrtho, 用以满足系统对三维空间投影的需求。

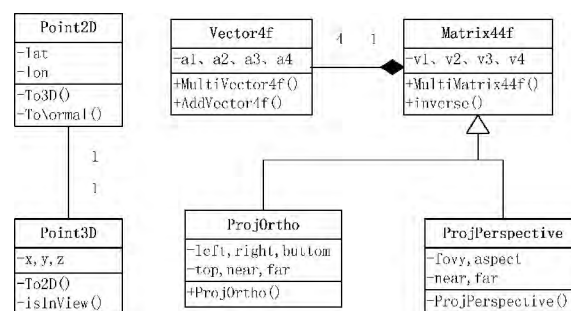


图 2 数学运算模块类图

3.2 图层元素模块

根据图 1 中 WebGL 的渲染流程, 我为所有的场景元素设计了一个统一的创建和绘制流程, 如图 3 所示。首先根据图层元素的需要进行对象初始化, 例如获取标注类型的文字标注信息、图标路径, 矢量类型的顶点数据等。接着根据获取到的数据进行各种 buffer 的创建, 例如顶点数组、面的顶点索引数组、贴图的纹理坐标数组、颜色数组等。接着根据需要设置元素对象的透视投影矩阵和模型视图矩

阵。通常以场景的基本透视矩阵为元素的投影矩阵, 场景的模型试图矩乘以元素对象的模型变换矩阵作为元素对象最终的模型试图矩阵。绘制的时候给着色器设置相关变量后, 依次绑定前步生成的各种 buffer 以及投影矩阵和模型变换矩阵。最后调用 WebGL 的基本图元绘制方法进行图层元素的绘制。

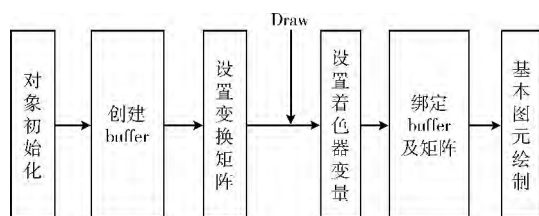


图 3 场景元素创建与绘制流程

统一的创建和绘制流程有利于进行场景元素对象的创

建和统一绘制调用, 同时有利于系统的扩展, 因为图层元素不能仅仅局限于上述的几种。为使所有的场景元素能够实现统一的创建和绘制流程, 系统通过 JavaScript 原型继承方法实现了图层元素的面向对象设计。首先设计了一个 BaseClass 类作为所有类的基类, 它提供了一个 extend 方法来实现 JavaScript 的原型继承, 该方法在执行的时候会将没有经过重声名的父类函数成员继承给子类, 同时为子类的原型添加新的函数成员。接着设计了一个所有场景元素类的基类 RenderableObject, 这个类相当于一个抽象类, 包含了获取 program、buffer 创建、设置矩阵和 Draw 方法等。这些方法都将在继承了 RenderableObject 类的场景元素类中实现。Draw 方法包含了绑定 buffer、绑定矩阵、设置贴图、调用图元绘制等子流程。这些类之间的关系如图 4 所示, 其中 RenderableObjectList 是为具有多个子对象的场景对象设计的场景元素类, 通常用来表示整个图层或者一组元素。

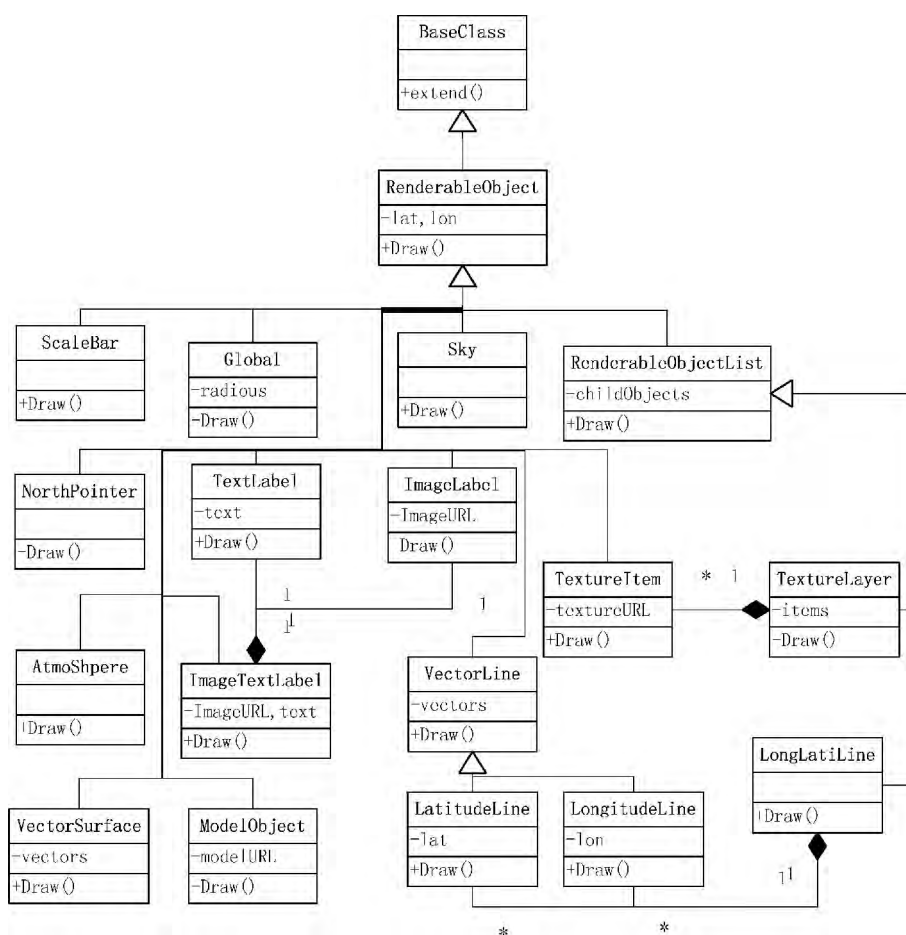


图 4 图层创建模块类图

3.3 场景渲染入口

系统设计了一个名为 World 的场景类, 这个类中包含了一个 WorldObjectList 属性, 它是一个 RenderableObjectList 对象, 场景的所有图层元素对象都将加入到

WorldObjectList 中。另外 World 类实现了初始化 WebGL 上下文、初始化着色器、初始化世界矩阵、绘制场景等方法。绘制的时候将叠代的调用 WorldObjectList 中所有成员的 Draw 方法进行图层元素绘制。由于上节所述的继承关系

的存在,各元素将自动调用自身所重写的方法或者调用其父类中实现的方法。

4 系统实现

4.1 系统界面

系统使用 HTML5 canvas 作为渲染容器,由于无法同时获取 canvas 得二维绘图环境和 WebGL 三维绘图环境,因此系统使用 2 个叠加的 canvas 标签元素。其中 canvas 一个负责获取 WebGL 的上下文环境进行基本场景、大气、天空、经纬线、指南针、矢量线、矢量面以及模型元素的绘制。由于 WebGL 没有提供文字的渲染接口,因此我们使用另外一个 canvas 获取二维绘图环境,用以进行比例尺、文字标注、图形标注、图文标注的绘制。

4.2 着色器

着色器是用来替代固定渲染管线的可编辑图像渲染程序,由 WebGL 基于 OpenGL ES 2.0,着色器是系统必不可少的一部分。系统设计实现了简单的顶点着色器和片元着色器,顶点着色器主要负责根据场景元素对象的顶点数组 buffer 以及模型视图矩阵和投影矩阵生成顶点位置,并将获取到的纹理坐标和顶点颜色传递给片元着色器,片元着色器根据这些信息生成各片元的颜色。

4.3 其它关键技术实现

4.3.1 多边形的三角切分

在进行矢量面图层元素 VectorSurface 的顶点 buffer 创建的时候,需要根据矢量面所提供的顺序顶点序列将矢量面进行三角拆分。由于本文采用拆分算法的是环形追踪算法^[7],使用它进行三角拆分的运算时间复杂度比较高,在浏览器端使用 JavaScript 执行的时候所需要的时间更多,众所周知 JavaScript 脚本的执行是单线程的,如果长时间的执行一个方法,将阻塞 UI 线程的执行,造成用户界面卡死的现象,这会大大降低用户体验。

为了解决这一问题,系统采用了 HTML5 的新特性之一 WebWorker 来进行多线程的开启。对多边形的拆运算将在开启的新线程中执行,执行的结果即拆分好的三角形列表将作为对象返回到主线程中。这样做的好处是大大提高了多边形的拆分效率,减少了 UI 线程的等待时间,提高了用户体验。另外由于浏览器所能开启的 WebWorker 的数量是有限的,即浏览器所能开启的线程数量是有限的,一次开启过多的线程会造成浏览器崩溃。因此系统设计了一个类似于线程池的 WorkerPool 类来进行 WebWorker 的管理。

4.3.2 贴图管理

图形标注以及地表影像等需要使用到大量的纹理贴图,这些贴图经常出现大量重复的现象。为了优化影响切片加载,以及一般的图片纹理的生成,系统作了一些优化,构建了一个 ImageManager 类来管理所有的图片纹理,将所有已经加载的图片和根据其生成的贴图纹理保存在一个队列

中,当下次再重新使用到这个图片的时候直接从队列中取出相应的贴图,系统将不再需要重新下载和重新生成纹理,一定程度上增加了渲染效率,提升了用户体验。

4.3.3 大气层模拟

一般来说大气层的绘制需要使用雾化、离子系统^[8]等对图形引擎性能要求较高的图形技术。但是考虑到 WebGL 是运行在浏览器端,渲染性能有一定的限制,同时也为了降低浏览器的内存消耗,这里使用了一个小技巧,将一幅具有渐变颜色的图片作为纹理贴在地球的边缘外侧,这样视觉上将为地球添加了一个漂亮的大气层。

5 实验与分析

为充分验证系统的有效性和可用性,本文共搜集整理出 25 组真实空间数据用于测试。测试数据按照用途可以分为基础地理数据和油田地理数据 2 个类别,按照地理信息元素类型可以分为文本标注数据 2 组,图文标注数据 7 组,矢量线数据 9 组,矢量面数据 5 组,模型数据 1 组,影像栅格数据 1 组。测试数据列表见表 1。

表 1 测试数据

名称	用途	测试目的	数量
国家名称	基础地理	文本标注	248
国家首都	基础地理	图文标注	210
省市名称	基础地理	文本标注	34
国境线	基础地理	矢量线	408
省市边界	基础地理	矢量线	816
沙漠边界	基础地理	矢量线	50
盆地边界	基础地理	矢量线	65
河流	基础地理	矢量线	743
湖泊	基础地理	矢量面	5
一级构造	基础地理	矢量面	20
二级构造	基础地理	矢量面	129
建筑物	/	模型	568
地球影像	基础地理	影像切片	160
探井	油田地理	图文标注	3677
评价井	油田地理	图文标注	1575
预探井	油田地理	图文标注	1482
正钻井	油田地理	图文标注	18
正试井	油田地理	图文标注	19
区域探井	油田地理	图文标注	14
集气支线	油田地理	矢量线	291
集气干线	油田地理	矢量线	806
集油支线	油田地理	矢量线	5110
集油干线	油田地理	矢量线	2177
气田	油田地理	矢量面	85
油田	油田地理	矢量面	30
总数量	—	—	18740

经过测试,所有空间数据均能够被系统创建并加载在

三维场景当中。满足了需求分析阶段所提出的功能需求, 表明 WebGL 能够满足三维地理信息系统所有元素类型的创建。图 5 和图 6 分别展示本位所构建的基本三维地球场景和加载了文本标注以及图形标注的测试结果。图 6 的场景中加载了国家首都、国家名称、省市名称、影像、大气层等元素。

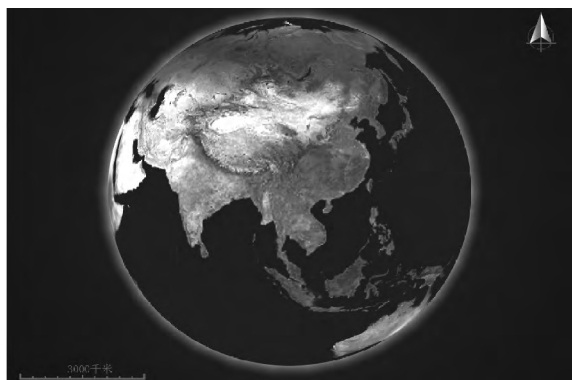


图 5 基本场景

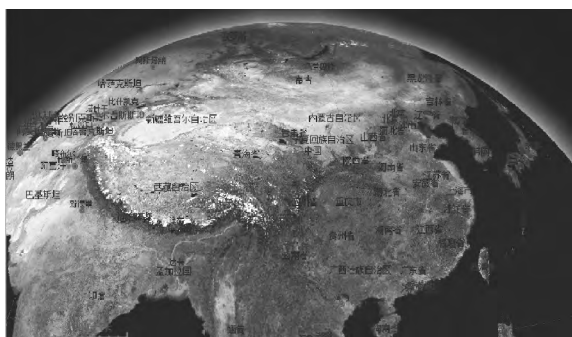


图 6 标注

传统的 Web3D 技术, Java3D 技术需要用户安装 Java 运行环境, 同时需要实时下载 Applet 小程序, Flash 需要为浏览器安装 FlashPlayer 插件, 同时要加载 swf 文件。本系统所采用了 WebGL 技术, 不再需要加载任何组件或者插件便能够实现三维 GIS 场景在 Web 页面上的展示, 这是传统 Web3D 技术无法比拟的优势。

从性能方面来说, 传统的 WebGIS 显示技术通常使用的是 CPU 来尽心图形计算。一方面 CPU 相对于 GPU 的计算速度要低两到三个数量级, 因此传统 Web3D 在性能上完全无法匹敌基于 GPU 计算的 WebGL。另一方面, 传统技术无使用 GPU 这一极佳的运算器, WebGL 则可以充分利用, 将 CPU 从复杂费力的图形运算中解放出来。经测试, 本系统的动画帧数能够达到 50 帧以上, 接近于最佳帧数 60, 而传统的 Flash 技术帧数通常在 30-40 左右, 一些基于网页 DOM 元素的技术帧数甚至更低。

从兼容性方面来看, 目前应用较广的 java 和 flash 技术

在浏览器端都具有较好的兼容性。但是, 随着移动互联网技术的不断发展和智能手机的不断普及。移动 Web 领域已经不可避免的成为地理信息系统所要进军的另一个领域。传统技术在手机浏览器上的无法执行的, IOS 甚至完全不支持 Flash 技术。WebGL 则不然, 目前已经有多家手机和手机浏览器生产厂商宣布将对 WebGL 提供软硬件方面的支持。这是未来 WebGL 的一个优势。

6 结束语

三维 WebGIS 场景能够提供相对于二维 WebGIS 更友好的用户体验和更加丰富有效的交互手段, 在网页上实现三维 GIS, 将能够三维 GIS 带入 Web 这一更加灵活、包容、丰富、用户群更加庞大的领域。为地理信息系统产业的发展带来强劲的推动作用^[9,10]。本文所设计的基于 WebGL 的三维场景系统在大量真实空间数据的测试中表现了良好的渲染效果。由此表明使用 WebGL 可以实现浏览器端三维场景的无组件绘制, WebGL 作为一个新的 WebGIS 前端展示层的实现手段是完全可以胜任的。

参考文献:

- [1] XU Peifeng. Flash-based Web3D development method [J]. Value Engineering, 2010, 29 (32): 179-180 (in Chinese). [徐佩峰. 基于 Flash 的 Web3D 开发方法研究 [J]. 价值工程, 2010, 29 (32): 179-180.]
- [2] YANG Xiaodong, HU Litang, TANG Zhonghua. 3D modeling and visualization of stratum based on Java/Java 3D [J]. ACTA Geodaetica ET Cartographica Sinica, 2006, 35 (2): 166-169 (in Chinese). [杨小冬, 胡立堂, 唐仲华. 基于 Java/Java3D 的地层三维建模与可视化 [J]. 测绘学报, 2006, 35 (2): 166-169.]
- [3] QIU Wei. Research and application of rich client WEBGIS based on X3D [J]. Microelectronics & Computer, 2011, 28 (1): 157-160 (in Chinese). [丘威. 基于 X3D 的富客户端 WEBGIS 应用研究 [J]. 微电子学与计算机, 2011, 28 (1): 157-160.]
- [4] XU Sha, YANG Fan, XU Changqing. Research and application of WebGIS based on HTML5 canvas [J]. Information Technology, 2012, 36 (4): 149-151 (in Chinese). [徐莎, 杨帆, 徐昌庆. 基于 HTML5 的 WEBGIS 的研究与应用 [J]. 信息技术, 2012, 36 (4): 149-151.]
- [5] HAN Yi. The new development trend in Web3D and Web visualization—taking WebGL and O3D as example [J]. Science Mosaic, 2010 (5): 81-86 (in Chinese). [韩义. Web3D 及 Web 三维可视化新发展——以 WebGL 和 O3D 为例 [J]. 科技广场, 2010 (5): 81-86.]
- [6] YANG Jun, LI Yanmei. Probe & analysis on JavaScript object-oriented programming [R]. Beijing: OA International Symposium, 2010: 21-24 (in Chinese). [杨俊, 李艳梅.

- JavaScript 面向对象编程探析 [R]. 北京: OA International Symposium, 2010: 21-24.]
- [7] ZHAI Renjian, WU Fang. An optimal triangulation algorithm for general polygon based on adaptive partitioning [J]. Journal of Geometrics Science and Technology, 2010, 27 (1): 122-125 (in Chinese). [翟仁建, 武芳. 利用自适应分块的任意多边形三角剖分算法 [J]. 测绘科学技术学报, 2010, 27 (1): 122-125.]
- [8] GUO Yemin. The construction of cloud model with particle system [J]. Journal of Changchun Normal University (Natural Science), 2012, 31 (12): 26-28 (in Chinese). [郭焰旻. 应用粒子系统构建云雾模型 [J]. 长春师范学院学报, 2012, 31 (12): 26-28.]
- [9] LV Zhihan, FENG Shengzhong. Multi-dimensional WebGIS based 3D interactive network virtual community [J]. Journal of System Simulation, 2013, 25 (9): 2109-2114 (in Chinese). [吕智涵, 冯圣中. 基于多维 WebGIS 的三维互动网络虚拟社区 [J]. 系统仿真学报, 2013, 25 (9): 2109-2114.]
- [10] TAN Wenwen, DING Shiyong, LI Guiying. Web-3D animation design and implementation based on WebGL and HTML5 [J]. Computer Knowledge and Technology, 2011, 7 (28): 6981-6983 (in Chinese). [谭文文, 丁世勇, 李桂英. 基于 WebGL 和 HTML5 的网页 3D 动画的设计与实现 [J]. 电脑知识与技术, 2011, 7 (28): 6981-6983.]

(上接第 3593 页)

- [2] Ma L, Li H, Lian S, et al. Expression and application of geo-spatial relation ontology [C] //Geoinformatics, 21st International Conference on IEEE, 2013: 1-5.
- [3] Mossakowski T, Moratz R. Qualitative reasoning about relative direction of oriented points [J]. Artificial Intelligence, 2012, 180: 34-45.
- [4] Tang Xuehua, Qin Kun. Direction-relation similarity model based on fuzzy close-degree [C] //Proceedings of the IEEE International Conference on Progress in Informatics and Computing, 2010.
- [5] Goyal R, Egenhofer M J. The direction-relation matrix: A representation for directions relations between extended spatial objects [C] //The Annual Assembly and the Summer Retreat of University Consortium for Geographic Information Systems Science, 1997.
- [6] Goyal R, Egenhofer M J. Cardinal directions between extended spatial objects [J]. IEEE Trans Data Knowledge Engineering, 2000.
- [7] Li S, Cohn A G. Reasoning with topological and directional spatial information [J]. Computational Intelligence, 2012, 28 (4): 579-616.
- [8] Wang S, Liu D. Knowledge representation and reasoning for qualitative spatial change [J]. Knowledge-Based Systems, 2012, 30: 161-171.
- [9] Skiadopoulos S, Koubarakis M. Composing cardinal direction relations [J]. Artificial Intelligence, 2004, 152 (2): 143-171.
- [10] WANG Miao, HAO Zhongxiao. Location representation and cardinal direction relation reasoning based on qualitative coordinates [J]. Academic Journal of Xi'an Jiaotong University, 2010, 44 (8): 36-41 (in Chinese). [王淼, 郝忠孝. 采用定性坐标的位置表达及主方向关系推理 [J]. 西安交通大学学报, 2010, 44 (8): 36-41.]
- [11] OUYANG Jihong, SUN Wei, LIU Dayou, et al. Composition of direction relation matrix [J]. Journal of Jilin University, 2010, 40 (4): 1048-1053 (in Chinese). [欧阳继红, 孙伟, 刘大有, 等. 方向关系矩阵的复合 [J]. 吉林大学学报(工学版), 2010, 40 (4): 1048-1053.]