

# WebGL 技术的三维 WebGIS 平台研究与应用

王星捷<sup>1</sup>, 卫守林<sup>2</sup>

(1. 成都理工大学 工程技术学院, 四川 乐山 614007;  
2. 昆明理工大学 信息工程与自动化学院, 昆明 650093)

**摘要:**针对 WebGIS 平台上的三维技术存在着开发平台成本高、需要额外插件、扩展性差、运行速度慢, 以及二、三维数据坐标易偏移和模型 UV 变形等问题, 采用 WebGL 的三维引擎技术和地理信息服务技术结合构建了三维 WebGIS 平台, 通过对 WebGL 场景坐标的分析, 设计了基于 WebGL 场景坐标转换算法, 实现了地理坐标转 WebGL 场景坐标, 解决了二、三维数据坐标易偏移的问题。通过对 WebGL 坐标与 UV 面的空间关系的研究, 设计了模型 UV 算法, 建立重构面的 UV 计算方程, 解决了模型 UV 变形问题。实验结果表明, 本文实现的三维 WebGIS 平台无需产品平台依靠和额外插件, 扩展性好, 响应速度快, 二、三维坐标数据准确, 解决了坐标偏移和模型 UV 变形的问题, 为三维 WebGIS 平台的开发提供了一种新的技术平台。

**关键词:** WebGL; 地理信息服务技术; 三维 WebGIS 平台; 场景坐标转换; 模型 UV 算法

doi: 10.3969/j.issn.1000-3177.2019.03.021

中图分类号: TP319 文献标志码: A 文章编号: 1000-3177(2019)163-0134-05

## Research and Application of WebGL Technology on 3D WebGIS Platform

WANG Xingjie<sup>1</sup>, WEI Shoulin<sup>2</sup>

(1. The Engineering Technical College, Chengdu University of Technology,  
Leshan, Sichuan 614007, China;  
2. The Faculty of Information Engineering and Automation, Kunming University of  
Science and Technology, Kunming 650093, China)

**Abstract:** For the 3D technology on WebGIS platform, there are some problems such as high cost of development platform, extra plug-in, poor scalability, slow operation speed, easy offset of 2D-3D data coordinates and UV deformation of the model. In this paper, WebGL 3D engine technology and geographic information service technology are combined to build a 3D WebGIS platform. Through the analysis of WebGL scene coordinates, the WebGL scene coordinate transformation algorithm is designed to realize the transformation from geographic coordinates to WebGL scene coordinates and to solve the problem of easy offset of the 2D-3D data coordinates. By studying the spatial relationship between WebGL coordinates and the UV surface, the UV algorithm of the model is designed, and the UV calculation equation of the reconstructed surface is established to solve the UV deformation problem of the model. Experimental results show that the 3D WebGIS platform implemented in this paper does not need to rely on product platform and additional plug-ins; it has good scalability, fast response, and the accurate 2D-3D coordinate data, solves the problems of coordinate offset and model UV deformation, and provides a new technology platform for the development of 3D WebGIS platform.

**Key words:** WebGL; geographic information service technology; 3D WebGIS platform; scene coordinate transformation; Model UV algorithm

收稿日期: 2017-11-27 修订日期: 2018-02-01

基金项目: 国家自然科学基金(11403009)。

作者简介: 王星捷(1980—), 男, 硕士, 副教授, 主要研究方向为数字城市、智慧城市、三维空间数据处理、大数据应用。

E-mail: stars\_80810@163.com

## 0 引言

WebGIS 让越来越多的用户通过 Internet 来获取和使用地理信息服务,实现了地理信息系统的大众化服务。但是三维 WebGIS 技术<sup>[1]</sup>一直是业界的难点,目前市面上成熟的三维 WebGIS 产品屈指可数,发展较好的国内有 SuperMap 和 MapGIS 平台;国外有 ArcGIS、SkyLine 平台。这些公司在 WebGIS 的三维可视化的研究及相关产品处于业界的领先水平。但是目前实现三维 WebGIS 的手段还是需要借助用户下载额外的插件程序来辅助实现,不能浏览太多的三维模型,不同的平台需要分别开发对应的插件,开发平台的成本价格较高。

本文基于 WebGIS(web graphics library)<sup>[2]</sup>结合地理信息服务(geographic informatica service)<sup>[3]</sup>对 WebGIS 的三维可视化分析技术进行探索和研究。使用地理信息服务发布地理系统的数据和分析服务,通过 Web 客户端调用这些服务,对场景的坐标转换和模型 UV 设计了算法优化,将大量的二、三维数据呈现在 Web 三维场景里。WebGL 独有的硬件级加速优势<sup>[4]</sup>与本文研究的坐标转换、模型 UV 优化算法的结合使场景浏览更加流畅,响应速度快,界面友好。实现了二三维联动、放大、缩小、数据查询定位、模型替换编辑等三维 WebGIS 功能。

## 1 三维 WebGIS 平台设计

系统主要分为数据存储层、网络服务层、平台业务层、前端表现层 4 层。数据层主要分为文件系统存储和数据库存储,网络服务层主要是由 ArcServer 发布的 GIS 服务和 IIS 提供的网页服务构成。平台业务层根据不同的业务而进行相应的业务处理。前端表现层即是用户最终在浏览器中看到效果,可以实现平台与用户的交互。如图 1 所示。

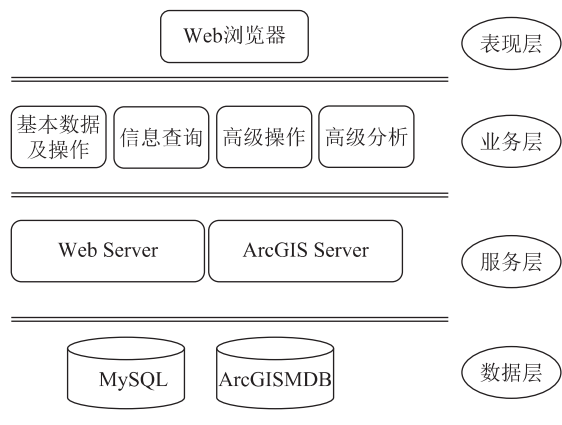


图 1 系统架构图

数据存储层主要分为文件系统存储和数据库存储 2 种类型,文件系统存储的数据主要包括平台所需的模型及贴图文件和属性数据。数据库存储又分为非空间(属性)数据和空间数据存储,其中空间数据使用 ArcGIS 的 SDE 工具将数据存储到 SQL Server 中,非空间数据直接使用 SQL Server 数据库进行存储。网络服务层的所有服务都必须基于数据存储层的支持,所以数据存储层是作为整个平台的基础支撑存在。

服务层由 ArcGIS Server 提供的 GIS 服务和 IIS 服务共同组成,ArcGIS Server 将数据库中的数据发布为 GIS 数据服务,并为平台提供地理信息系统的地理查询、缓冲区分析、网络路径分析等 GIS 服务。IIS 为平台提供模型和贴图文件数据,使平台在三维模型的创建和加载速度大大提升。网络服务是平台的核心处理中心存在。

平台业务层主要工作就是根据不同的业务需求将网络服务层的网络服务与前端响应部分进行搭配组合不同的功能,例如将缓冲区分析与空间查询服务结合成邻近设施查询功能。

前端表现层可以理解为用户最终在浏览其中看到的效果,主要目的就是实现与用户的交互。前端采用 EasyUI.js 插件库作为平台的 UI 驱动,二维地图使用 ArcGIS API For JavaScript 的二次开发接口进行开发实现,三维 GIS 场景使用 Three.js 插件库作为 WebGL 的驱动<sup>[5]</sup>。界面简洁,功能强大。

## 2 场景坐标转换

GIS 矢量数据的坐标不能直接用于 WebGL 场景中,需要进行的坐标转换,转换为场景中的坐标点,以便 WebGL 使用这些矢量数据。在参考托球体中,范围较小的区域可以近似于一个平面,因此只需当作平面坐标向平面坐标的转换进行处理。

面积小的区域的坐标可以直接转换到 WebGL 场景中,在三维场景中,习惯于在点(0,0,0)附近进行操作。地理坐标的数值长度和精度较高,所以需要找到一个标准点作为原点,其他坐标就可以根据标准点坐标进行校正。本文把这个点叫做校正点  $J(lon_j, lat_j)$ 。本文需要将地理坐标点  $m$  转换为以校正点为原点的平面坐标  $M$ 。

假如地理坐标使用的比例尺为 1:100 000。即图上一个单位长度对应现实中 100 000 m,在 WebGL 坐标系中,需要将经纬度坐标分别乘以 100 000。

$$M(lon_M, lat_M) = (m(lon_m, lat_m) - J(lon_j, lat_j)) \times 100\,000 \quad (1)$$

WebGL 场景使用的是右手坐标系,  $Y$  轴向上。为了方便, 将 WebGL 场景中的  $XZ$  坐标作为经纬度坐标的映射。即最终得到场景中的坐标为:  $P(lon_M, 0, -lat_M)$ 。从而实现地理坐标向 WebGL 场景坐标的转换, 过程如图 2 所示。

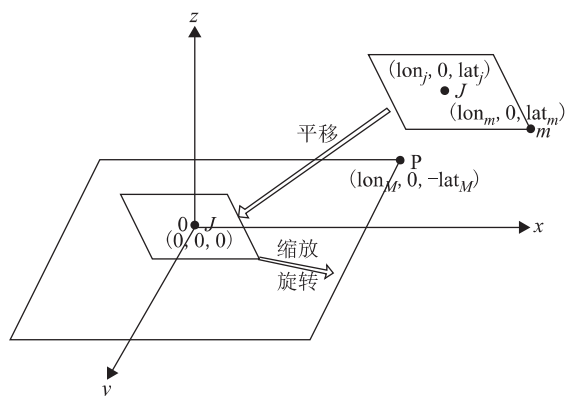


图 2 地理坐标在 WebGL 场景中的转换图

将准备好的影像图的范围作为场景建模范围, 将影像图的中心点坐标作为算法中提到的校正点  $J$ , 然后将影像图的左下角坐标和右上角坐标分别按照上面的坐标转换算法进行转换, 再将影像图根据转换后的坐标导入到场景中。本文测试数据为  $10 \text{ km}^2$ , 对于较大的数据可以进行数据切片<sup>[6]</sup>处理后再进行组合。

### 3 模型 UV 算法

在完成了场景中影像图的添加, 需要在场景中创建建筑模型<sup>[7]</sup>, 由于考虑到涉及数据量较大, 大部分的建筑模型将直接把二维 GIS 数据进行三维可视化处理<sup>[8]</sup>。具体实现方法是将制作好的建筑矢量面数据进行坐标转换处理得到平台适用的 Shape 矢量面, 然后再将这些建筑模型的 Shape 矢量面按照建筑物的高程字段进行拉伸, 从而获得三维立体建筑模型<sup>[9]</sup>。这也是大多数 GIS 平台将二维 GIS 数据进行三维可视化的常用手段<sup>[10]</sup>。但这种方式很容易出现侧面的贴图变形, 原因是在计算拉伸面的  $UV$  坐标时没有考虑  $Y$  坐标轴方向的坐标。因此拉伸出来的侧面  $UV$  贴图坐标是错乱的。因此需要根据每一个顶点的三维坐标对拉伸出来的面的  $UV$  坐标使用  $UV$  算法重新计算, 使三角面的  $UV$  坐标分布比例与三角面顶点在对应  $UV$  面的投影点分布比例保持一致。

空间直角坐标系中的三角面顶点分布如图 3 所示。需要在三角面  $ABC$  所在的平面创建二维平面坐标系作为  $UV$  面。假设  $O$  点在三角面所在的平

面的投影点为  $P$ , 所得到的向量  $OP$  即为三角面的法向量。将空间直角坐标系中的  $Y$  坐标轴与  $OP$  向量形成的平面与三角面  $ABC$  所在的平面的相交线 (即  $Y$  轴在三角面  $ABC$  上的投影线) 作为二维平面坐标系的  $Y$  轴, 投影线的方向为二维平面坐标系  $Y$  轴的正方向。设  $P$  点为三角面  $ABC$  所在平面的二维平面坐标系的原点, 则可以将以  $P$  点出发的射线作为二维坐标系的  $X$  轴的正方向。

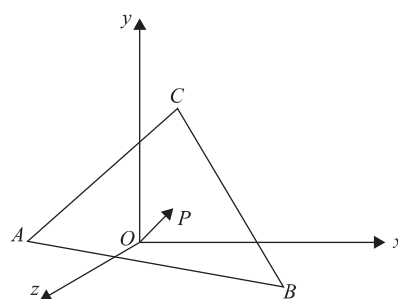


图 3 三角面顶点在立体空间的分布

已知坐标  $A(Xa, Ya, Za)$ ,  $B(Xb, Yb, Zb)$ ,  $C(Xc, Yc, Zc)$ , 三维场景坐标系中  $Y$  轴所在的向量为  $\vec{e}(0, 1, 0)$ ,  $\vec{AB}$  向量的计算公式如下:

$$\vec{AB} = \vec{b}(Xb, Yb, Zb) - \vec{a}(Xa, Ya, Za) \quad (2)$$

$\vec{AC}$  向量的计算公式如下:

$$\vec{AC} = \vec{c}(Xc, Yc, Zc) - \vec{a}(Xa, Ya, Za) \quad (3)$$

$OP$  向量的计算公式如下:

$$\vec{p} = \vec{AB} \times \vec{AC} \quad (4)$$

平面坐标系的  $U$  轴所在的向量的计算公式如下:

$$\vec{m} = \vec{p} \times \vec{e} \quad (5)$$

平面坐标系的  $V$  轴所在的向量的计算公式如下:

$$\vec{n} = \vec{m} \times \vec{p} \quad (6)$$

有了  $\vec{m}$  和  $\vec{n}$  向量, 那么空间直角坐标系中任意三角面顶点  $K(Xk, Yk, Zk)$  的平面  $UV$  坐标  $(Uk, Vk)$  的坐标值就可以定义为到  $K$  点对应的向量  $\vec{k}(Xk, Yk, Zk)$  分别在  $\vec{m}$  和  $\vec{n}$  向量上的投影距离。  $(Uk, Vk)$  的计算公式如下:

$$Uk = \frac{\vec{k} \cdot \vec{m}}{|\vec{m}|} \quad (7)$$

$$Vk = \frac{\vec{k} \cdot \vec{n}}{|\vec{n}|} \quad (8)$$

最终计算得到顶点  $K$  在对应三角面中重新定义的  $UV$  坐标的计算公式如下:

$$K\left(\frac{\vec{k} \cdot \vec{m}}{|\vec{m}|}, \frac{\vec{k} \cdot \vec{n}}{|\vec{n}|}\right) \quad (9)$$

各个向量之间的关系如图 4 所示。

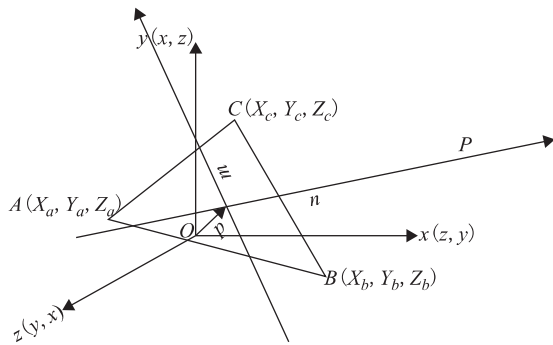


图 4 UV 处理算法中的向量图解

#### 4 系统实验与分析

本文测试的数据是一个德国小镇的数据,范围为  $10 \text{ km}^2$ 。模型数量超过 5 000 个,其中矢量面拉伸生成的模型 1 421 个,导入外部模型 3 600 多个,按平台设计需要还包含每个建筑物的高程数据、模型参数(路径、大小、旋转)、道路宽度、建筑物名称等必要属性数据。

通过场景坐标转换后,将影像图根据转换后的坐标导入到场景中,效果如图 5 所示。



图 5 影像图导入效果

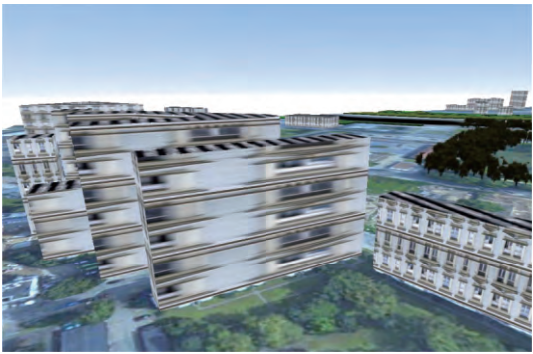
可以看到影像图导入到场景中几乎没有变形,证明了本文研究的坐标转换算法的正确和可行性。

经过模型 UV 处理算法处理之后,场景中三角面的 UV 贴图坐标的比例与三角面顶点在空间直角坐标系中的坐标分布比例保持一致。当模型被赋予贴图时不会发生变形。变换前后效果如图 6 所示。

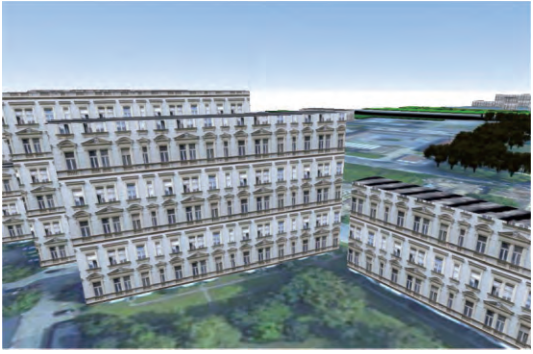
从 UV 算法处理前后的对比效果图,可以得出本文 UV 算法的正确性。

将数据和三维模型导入后,实现的三维 WebGIS 平台的效果如图 7 所示。

为了验证平台的实用价值,现将本平台与市场其他商业的三维 WebGIS 平台进行了分析比较,结果如表 1 所示。



(a) UV 算法处理前



(b) UV 算法处理后

图 6 UV 算法处理前后的对比

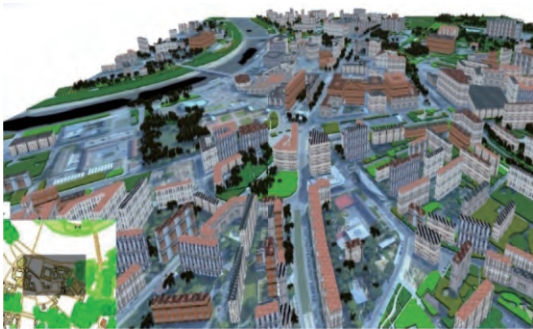


图 7 三维 WebGIS 平台

表 1 三维 WebGIS 开发平台的比较

开发平台	ArcGIS	SkyLine	SuperMap	MapGIS	本平台
插件	否	是	否	是	否
开源	否	否	否	否	是
收费	是	是	是	是	否
跨平台	支持	不支持	支持	不支持	支持
拓展性	难	难	难	难	易

从对比结果可以看出大多数三维 WebGIS 产品化的开发平台,需要安装插件辅助浏览三维场景,支持跨平台使用产品较少,都是非开源产品,扩展性差、收费较贵、与这些平台的产品相比之下,本平台具有跨平台支持、高性能、易扩展、低成本等优点。



## 5 结束语

本文研究了基于 WebGL 搭建三维 WebGIS 平台的总体思路,以及场景坐标转换和模型 UV 处理算法,并充分结合了 WebGL 硬件级三维渲染加速的优势,实现了三维 WebGIS 平台。其优点:①本平台基于开源,无需任何额外的平台费用,成本低;②采用了 three.js 轻量级技术与客户端交互,保证了客户端数据交互更快捷、高效;③设计了特有的场

景坐标转换和模型 UV 处理算法,解决了坐标偏移和模型贴图变形的问题;④在客户端进行浏览无需任何插件,并且支持跨平台;⑤本平台从上至下,都是基于开源模式设计,无需依靠任何产品,平台的扩展性强;⑥本平台以小区域化进行设计,支持大区域数据切片和组合,模型加载方便。本文实现的三维 WebGIS 平台具有实际的应用价值,实现的技术为三维 WebGIS 平台的开发提供了一种新的技术模式。

## 参考文献

- [1] 欧阳峰,龚桂荣,何列松.面向 WebGL 的矢量数据三维绘制技术[J].测绘科学技术学报,2016,33(6):635-638.
- [2] 黄若思,李传荣,冯磊,等.基于几何的 WebGL 矢量数据三维渲染技术研究[J].遥感技术与应用,2014,29(3):463-468.
- [3] 李兴田,张丽萍.基于 WebGL 的工程制图网络虚拟模型库的开发[J].图学学报,2016,37(6):836-841.
- [4] 杨帆,杜凯.基于 WebGL 的三维虚拟地球系统设计与实现[J].地理信息世界,2016,23(2):113-118.
- [5] 郑华,宿景芳.面向 Web 的三维模型生成与处理技术[J].现代电子技术,2015,38(24):83-86.
- [6] 陈举平,丁建勋.矢量瓦片地图关键技术研究[J].地理空间信息,2017,15(8):44-47.
- [7] 朱丽萍,李洪奇,杜萌萌,等.基于 WebGL 的三维 WebGIS 场景实现[J].计算机工程与设计,2014(10):3645-3650.
- [8] JENNY B, AVRIÅ B Å, LIEM J. Real-time raster projection for web maps[J]. International Journal of Digital Earth, 2016,9(3):215-229.
- [9] TU W, DING S, WU L, et al. SynBioEcoli: a comprehensive metabolism network of engineered E. coli in three dimensional visualization[J]. Quantitative Biology, 2017, 5(1):99-104.
- [10] LI P, YU XQ, WANG J. Progressive compression and transmission of 3D model with WebGL[J]. International Conference on Audio, 2017:170-173.