修订日期:2018年9月13日接受日期:2018年11月20日

DOI: 10.4218 / etrij.2018 - 0352

**原文**

IMAGE

**基于WebGL的三维空间信息开放平台的实现**

**Ahyun Lee |Insung Jang**

超级量连接通信

研究实验室，电子和电信研究所，大田，韩国代表

**对应**

李爱贤，超连接通信研究实验室，电子与电信研究所，大田，韩国代表。电子邮件:ahyun@etri.re.kr

**资金信息**

这项工作得到了部里的支持

根据韩国基础设施技术进步署(KAIA)的城市规划与建筑(UPA)研究支持计划(批准13项城市规划与建筑02)，韩国城市规划与建筑(MOLIT)。它还得到了一笔拨款(18DRMS-B147287-01)，用于根据消费者需求开发定制的三维(3D)地理空间信息更新和利用技术，资金来自韩国政府土地、基础设施和交通部。

VWorld由韩国国土交通运输部运营，提供国家空间信息，如航空图像、数字高程模型和3D结构模型。提出了一种基于WebGL的基于VWorld空间信息的三维空间信息开放平台。WebGL是一个基于web的图形库，它的优点是与各种web浏览器兼容。我们的开放平台也兼容各种web浏览器。因此，它很容易通过VWorld网站访问，并使用三维(3D)地图程序。在本研究中，我们描述了所提出的平台配置，以及VWorld空间信息数据的请求、管理和可视化方法。我们的目标是建立一种方法，即使在没有基于四叉树结构的图形处理单元的低端个人计算机上，也能提供稳定的渲染速度。我们期望用户能够通过VWorld开放平台来可视化三维空间信息，并且所提出的平台将成为各种应用的基础。

**关键字**

三维地图，地理信息系统，空间信息，VWorld, WebGL

**1 |介绍**

VWorld数据中心由韩国国土交通运输部(MOLIT)运营，提供最新的、高质量的三维(3D)国家空间信息数据[1,2]。它还提供了基于各种元数据的搜索功能，以及连接到二维(2D)/三维地图的信息查询功能。提供的元数据包括三维结构模型、航空图像、数字高程模型(DEMs)和行政区域信息。本研究描述了一种基于WebGL的基于VWorld数据的三维空间信息开放平台的实现方法，如图1所示。

WebGL是一个基于web的图形库，它使用JavaScript编程语言[4-9]。WebGL使用基于OpenGL ES 2.0和文档对象模型接口的HTML5 canvas元素，这是W3C的官方标准。W3C表示与平台或语言[10]无关的结构化文档。我们基于WebGL的平台可以在各种浏览器上运行，如谷歌Chrome、Microsoft Internet Explorer 11、Edge、Mozilla Firefox、Safari和Opera。因此，任何人都可以在各种计算环境中访问开放平台的VWorld站点[11]。开放平台是应用MOLIT提供的国家空间信息的基础平台。

这是一篇根据韩国开放政府许可(KOGL)条款发布的开放获取文章，类型4:源指示+商业使用禁止+更改禁止(http://www.kogl.or.kr/info/licenseTypeEn.do)。

1225-6463/$©2019 ETRI

| 277

wileyonlinelibrary.com/journal/etrij

*电子杂志。2019;41:277 - 288。*

**278 |**

**图1基于VWorld数据的三维(3D)空间信息开放平台**

OpenGlobus通过网络浏览器提供基于WebGL的3D地图。然而，如果对地形数据的请求没有完成，那么在没有纹理图像的情况下，就不能正确地呈现地形数据，如图2A[12]所示。基于WebGL‐的铯三维地图在球面上表示二维地形纹理图像

(B)

(一)

李和张成泽

表示全球[13]。因此，地面高程信息不予说明，如图2B所示。

使用最广泛的3D地图是谷歌Earth[14](图2C)。谷歌地球是基于数字表面模型(DSM)，因此它比基于DEM的方法更快。DSM方法使用单个‐纹理图像，结合地形和结构信息[15-18]。图像尺寸小于DEM方法中使用的图像尺寸。然而，如图2D所示，这是图2C的放大图，其中建筑物体并不比用DEM渲染的图像更清晰，地形和建筑图像是独立使用的。此外，建筑模型不能被视为独立的对象。因此，很难根据3D模型选择建筑对象或表示与建筑相关的额外信息。相比之下，我们基于DEM的开放平台可以用DEM来表示济州岛，如图2E所示，与图2C不同。此外，所选建筑可以用不同的方式表示，如图2F所示。

首尔的3D地理信息系统(GIS)平台与我们的平台相似，它使用VWorld数据，是一个基于WebGL的[19]平台。然而，它只处理首尔市的区域，而外部城市区域(除了内部城市区域)用黑色高亮显示，如图2G所示。为了加速三维建筑的渲染，所有建筑的纹理图像被合并并保存在一个单独的文件中。这是有利的，因为请求的文件数量和文件的大小都很小。

(D)

(F)

(4 0)

(0,0)

(4、9)

(0,9)

(C)

(E)

(一)

（G）

**图2的例子3 D地图平台:(A) OpenGlobus [12], (B)在韩国济州岛插图使用铯[13]没有高程数据,谷歌地球(C) [14], (D) (C)的放大图像显示不均匀的建筑物表面,(E)济州岛了高程数据,(F)用户优先选择建筑显示在红色在我们提出平台[11],和三维地理信息系统(G)首尔平台[20]**

(9,0)

(9日19)

(0,0)

(0,19)

(B)

**图3瓷砖结构的例子:(A)第0级的50块瓷砖和(B)第1级的200块瓷砖**



李和张成泽

**| 279**

但是分辨率和质量都低于VWorld数据中心提供的原始文件。

在本研究中，我们描述了在VWorld数据中心中作为三维空间信息使用的开放平台的实现细节。我们的目标是引入一种方法，在不使用图形处理单元(GPU)的情况下，使用低端个人计算机(PC)有效地呈现VWorld空间信息(总计30 TB或更多)。提出了一种基于四叉树结构的海量VWorld空间信息数据管理、请求和绘制方法。我们希望这项研究能够对基于我们所提议的开放平台的VWorld空间信息数据的应用有所帮助。

本研究的其余部分结构如下。在第二部分，我们描述了VWorld的空间信息数据。在第3节中，我们描述了三维空间信息开放平台的实现细节。在第4节中，我们描述了平台的转换和渲染过程。最后，在第5部分，我们概述了结束语。

**2 | VWORLD空间信息数据**

韩国国土交通运输部VWorld数据中心[1]提供的空间信息数据由航空图像、DEMs、3D建筑数据、景点、混合图像和其他元素组成，这些元素的总大小可超过30 TB。本节介绍VWorld空间信息数据的结构和特点。我们还描述了用于3D结构模型的Oracle BI publisher资源跟踪文件(XDO)的独特3D数据格式。

**2.1 |瓷砖**

地砖是包含与地球相关的空间表面图像信息的单元。地砖由具有地形高度的航空图像(256像素×256像素)和DEM图像(64像素×64像素)组成。表示地球球面形状的磁瓦由50个零级磁瓦组成，在WGS84坐标系[20]中分为36°纬度和36°经度线，如图3所示。第0级的瓦片被分成4个相等的部分，包括第1级的瓦片，它们依次被18°纬度和18°经度线划分。因此，对于0-15范围内的级别，tile数据包括50×4个tile level /level。

平铺ID定义函数h(l, r, c)定义为(1)从底部‐左侧部分到第一条水平线。此外，l、r和c表示水平面、行和列。



50×4 + 10×2×r + c,否则。*l*−1 *l*

(1)

**VWorld三维空间信息的位置由三维坐标系中的纬度(lat)和经度(lon)坐标定义。**经纬度(lat- lon)位置基于WGS84坐标，三维位置基于ECEF左手坐标系，如图4A所示。x轴通过(0,0)和(0,180)或(0，- 180)的拉特隆坐标。此外，y轴通过(0,90)和(0,90)的拉特隆坐标，而z轴通过南北两极。WGS84和ECEF变换方程定义在式(2)和式(3)中，地球的半径R为6,378,137 (m)，因此，由弧度角组成的latlon坐标，ECEF矢量的值为x、y、z。

⎡R∗因为纬度∗因为经度⎤

⎢R∗因为纬度∗罪经度⎥ECEF向量=⎢⎥,

⎢⎥



⎥⎣1⎦

·000年

(2)

��√

⎡罪−1 z x2 + y2 + z2⎤

�

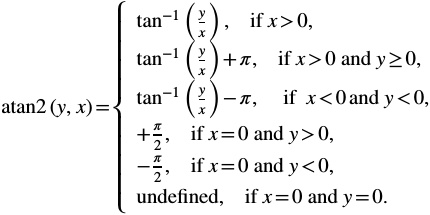
⎢⎥WGS84向量=⎢−量化⎥(y、x)。⎢⎥⎣1⎦

(3)

基于3D网络的图形库WebGL使用右手坐标系，如图4B所示。在第4节中，我们将描述如何将VWorld数据的XDO转换为WebGL坐标。

��

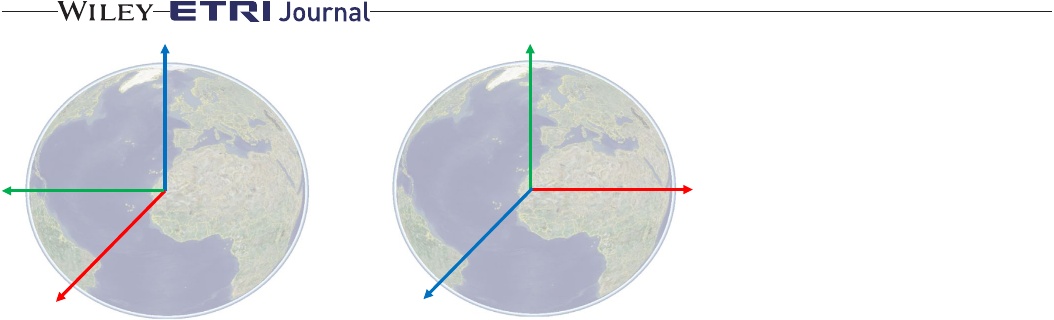
*y*



(4)

**2.3 | XDO**

将VWorld 3D结构数据细分为XDO和XDO的索引数据。块具有索引数据。它的数据对象与objectCount的值一样多，如表1所示。它还包含与object- Count的值一样多的XDO数据对象。XDO是一种表示VWorld结构模型的新格式。如表2所示，一个XDO包含一个独立建筑的3D结构模型的数据。因此，顶点是一个3D点，index表示顶点数据数组对三角形网格进行合成的索引，uv表示三角形网格在纹理图像中的2D位置。



**280 |**

李和张成泽

*z*

*y*

**图4坐标系统:(A) Oracle BI publisher资源跟踪文件(XDO)和(B) WebGL**

*y*

*x*

*x*

*z*

(一)

例如，图5A显示了一个呈现的建筑物。因此，一个黄色网格由三个顶点，和表示，如图5B所示，它们用图5A的字符串构成呈现结果。*v*0*v*1*v*2

在本例中，index是顶点数据数组中的位置值(0,1,2)。另外，uv有一个2D坐标值，在这个值上纹理图像分辨率大小被归一化为1，如图5C所示。顶点和uv的值都是4字节长。因此，顶点三维坐标的内存大小为vertexCnt×12(字节)，uv二维坐标的内存大小为vertexCnt×8(字节)。在索引的情况下，相同的顶点可以包含在几个索引中。因此，索引的数量与顶点的数量不成比例。

VWorld XDO只提供主要城市和设施的三维结构模型，而不是整个国家。当前可用的VWorld XDO数据只包含在13-15层的块中。第14层有桥梁，第15层有建筑物。特别值得一提的是，第13层包含了韩国独岛的地形信息，而独岛并不是一座建筑。

**3|开放的三维空间信息平台**

本节描述VWorld开放平台的实现细节。根据虚拟世界的空间信息特征，描述了三维数据的请求和表示方法。此外，鉴于我们的平台处理大量的高流量数据，也提出了一种有效管理这些数据的方法。图6显示了所提议的开放平台的任务流程图。一系列进程以每帧一次的速度重复。实现的平台由一个称为扇区的基本单元组成。使用用户界面控制摄像机的位置，通过改变摄像机的位置来搜索绘制请求块(扇区)。接下来的步骤是请求和绘制磁贴(扇区)并执行基于WebGL的渲染过程。

(B)

**表1名称级别IDX IDY objectCount**

XDO的VWorld索引数据

**4 4 4 4 4对象数据4 1 1 keyLen 16 4 48 1 1 dataFileLen 1 imgFileNameLen**

版本类型keyLen关键字海拔box imgLevel数据文件imgenamelen imgFileName

**经度标识(ID)纬度标识号对象版本类型大小key key中心坐标高度边界图像级别数据文件大小数据文件大小图像文件名图像文件名**

|  |  |  |
| --- | --- | --- |
| **表2** | VWorld XDO数据 |  |
| **的名字** | **大小** | **内容** |
| vertexCnt | 4 | 顶点的大小 |
| 顶点 | *vertexCnt 12*× | 顶点 |
| 紫外线 | *vertexCnt×8* | 紫外线 |
| indexCnt | 4 | 指数的大小 |
| 指数 | *indexCnt* | 指数 |
| 颜色 | 4 | 物体的颜色 |
| 盒子 | 48 | 边界 |



李和张成泽

**| 281**

**3.1 |行业**

地砖是一种地形图像与航空图像和DEM的结合。在我们提出的平台中，将平铺与3D结构模型相结合的结构称为扇形。扇区包括航拍图像、DEM、三维结构模型(XDO)、瓦片的大小和位置等信息。与平铺类似，扇区包含不同的级别(0-15)，显示扇区的级别是根据相机与扇区中心的距离确定的。扇区标识(ID)的确定方式与第(1)中的tile ID相同。

扇区是管理平台中数据的基本单元，如图7所示。扇区经理决定按每个扇区请求、存储和显示的数据。开放地图应用程序编程接口(API)提供地图操作功能，开放数据API提供数据请求和管理功能。WebGL库用于以3D方式显示扇区数据。

**3.2 |四叉树结构**

本节介绍图6中所示的draw-request扇区搜索功能。扇区搜索功能基于四叉树结构[21,22]。0级的50个扇区各有4个子扇区，每个子扇区包含另外4个子扇区。因此，15个层次由50×4个扇区组成。level 我们的平台具有四叉树结构，可以有效地实时呈现多级扇区。

如果父扇区不是四叉树结构中的呈现目标，那么其子扇区也不是呈现目标。例如，如果一个零级的父扇区不是呈现

作为目标，没有必要对它的所有子扇区执行筛选测试(见表3)。也就是说，1-15范围内有1,073,741,824个扇区。我们的主要目标是提出一种方法，以有效地渲染VWorld空间信息(大小达30 TB或以上)，使用低端PC，没有GPU，实时。

根据相机的位置和方向，给出了各个层次的扇区，如图7所示。要显示的扇区的级别是通过计算ECEF坐标系中摄像机位姿与扇区中心位姿之间的距离来计算的。***c s***

√

(5)

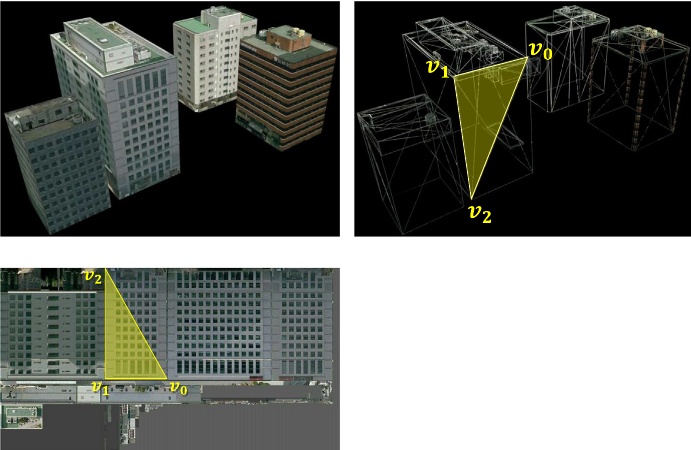


(6)

如果显示扇区的级别不同，则绘制重叠扇区。此外，如果有一个扇区的数据请求不完整，则可能发生屏幕闪烁。例如，级别10和8的扇区确实存在，但是级别9的扇区不存在。当所提供的级别从8更改为10时，由于相关的web浏览器功能(不能保证首先下载请求)，级别9的扇区无法显示。

我们基于四叉树的平台克服了上述限制。首先，它在0级时对所有50个扇区进行筛选测试，如表3所示。此外,绘画水平的部门通过了筛选测试是基于(6)决定的。如果一个行业的水平小于画水平,它分为四个孩子,和扑杀测试进行之前的部门,就等于画的水平。

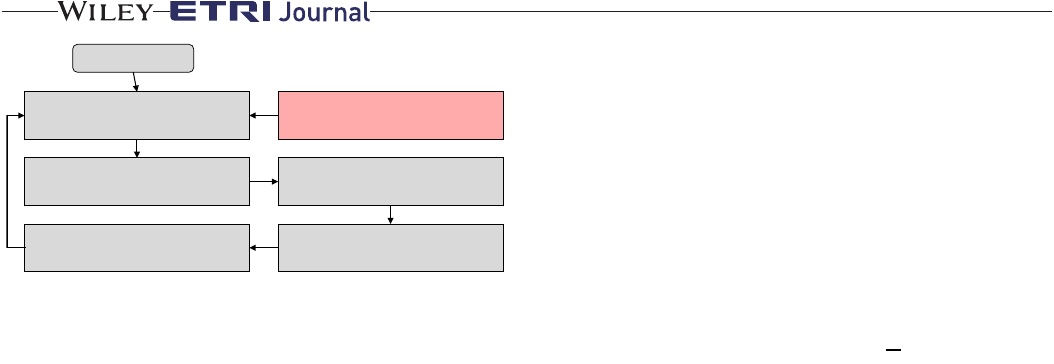
**图5 XDO 3D建筑渲染示例:(A)渲染建筑，(B)用字符串表示的渲染建筑中的三角形网格，(C)纹理图像中的三角形区域**



(一)

(B)

(C)



**282 |**

李和张成泽

⌊中心=(180 +朗)∗2列。*lv*

36

⌋

(8)

开始计算相机姿态

找到draw-request行业

用户界面请求扇区

WebGL呈现

画领域

**图6拟议平台的任务流程:红色的块表示用户界面的事件处理程序，灰色的块表示呈现循环流程**

所有六架截锥体飞机，这个扇区必须位于视图‐截锥体内，并代表渲染目标。如果红色区域的尺寸太大，所有的角落都位于view‐frustum平面之外，尽管该区域必须被渲染，如图10所示。当这种问题出现在这个平台上时，目标扇区是中央扇区。中央扇形是包含照相机的主线与地球表面的交点的扇形。每一层中央部门的行数和列数可以计算如下:



应用程序

WebGL部门

用户界面

部门经理

开放数据API

打开地图API

结构的VWorld开放的空间平台

**图7的信息**

当绘制级别和扇区级别相同时，执行检查以评估数据请求是否已完成。如果子扇区的请求已经完成，则将其放置在draw数组中;否则，父扇区将放置在draw数组中，子扇区将放置在请求数组中。请求数组中的扇区数据请求是通过请求扇区函数执行的，如图6所示。图8显示了根据每个扇区的绘制级别呈现的结果示例。

**3.2.1 |剔除试验**

该平台根据三个步骤进行了筛选测试。基本的后置面和视截头筛选试验是经典的方法。两次筛选测试都使用了四个扇形角。角是基于XDO坐标系的三维矢量，如图9所示。**s***i* 在背面剔除的情况下，如果角点与摄像机之间的角度小于90°，则目标为渲染目标，位于前方[23-25]。

)



(7)

(

**姐姐面前。***i*

视图‐截头剔除仅使用，这已被确定在前面。**s***i* 如果其中一个角在的法向量的方向上

行、列和级别可用于计算(1)中的扇区ID，以确定该扇区是否是中心扇区。在中央区域的情况下，后面的面和视图的截锥体剔除试验是不适用的。除中央航段外，所有其他航段都进行了背面和视截锥剔除试验。

**表3基于四叉树结构的draw-request扇区搜索函数伪代码**

**函数:为:i = 0到49 DO查找DRAW-REQUEST扇区()**

**IF: CULLING TEST(扇区[i]) drawLevel =设置绘制级别(扇区[i])。IF: drawLevel >扇区[i]。***级别查找子扇区(扇区[i])。***其他:**

**如果:检查加载的数据(扇区[i])。**在drawArray保存。

**其他:**

在requestArray保存。

**结束**

**功能:扑杀测试(年代)**

**如果:Is中心扇区(s)**

**返回true。**

**如果:面后剔除和视截距剔除返回true。**

**其他:**

**返回false。**

**结束**

**功能:查找子扇区。***子元素[0]= h (s。等级+ 1,s。行×2,s。col×2). s。子元素[1]= h (s。等级+ 1,s。行×2,s。坳×2 + 1)。年代。子元素[2]= h (s。等级+ 1,s。行×2 + 1,s。col×2). s。子元素[3]= h (s。等级+ 1,s。行×2 + 1,s。坳×2 + 1)。***FOR: i = 0 to 3 DO**

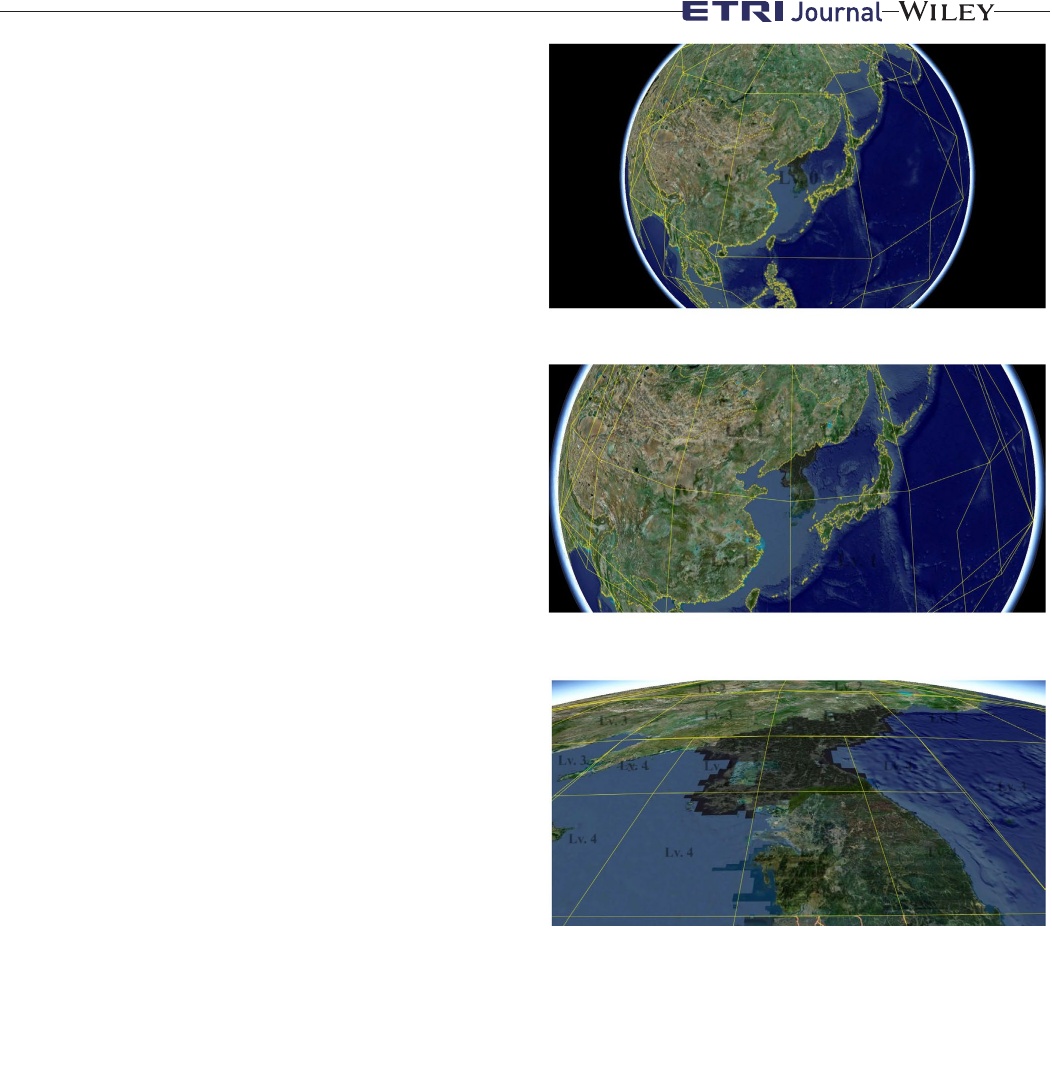
**IF: CULLING TEST (.child[i]) drawLevel =设置绘制级别(.child[i])。IF: drawLevel > .child[i]。***level查找子扇区(s.child[i])*

**IF:检查加载的数据(.child[i])。**在drawArray保存。

**其他:**

保存s.child[我]。*在drawArray父母。*保存年代。*孩子在requestArray[我]。*

**结束**



李和张成泽

**| 283**

**3.2.2 |数据请求**

在通过筛选测试的扇区中，未被请求的扇区被放置在请求数组中。当基于四叉树结构的搜索过程完成后，它通过对请求数组中的扇区进行优先排序来请求数据，如表4所示。首先，请求数组中的扇区根据其级别按升序排序。然后在每个阵列级计算相机与扇区中心之间的距离，并根据距离对扇区按升序排序。

根据所使用的web浏览器，每个服务器/代理的并发、默认和持久连接的最大数量设置为2-8。我们的开放平台针对谷歌Chrome进行了优化，最多只能有6个连接。因此，在执行表4的两个升序排序操作时，在结果的每一帧只请求前两个扇区数据。此时，请求的数据包括XDO的航空图像、DEM和索引数据。航空图像以256×256像素的矩阵形式保存，每个单元的大小大约为40kb到100kb。DEM的大小也小于约10kb，并不大。该平台在一帧中呈现大约10到70个扇区，并为XDO请求航空图像、DEM和索引数据，同时逐步划分扇区。因此，在一般的网络环境下，可以方便地进行三维地形数据请求和管理。

但是，对于XDO数据，由于大小从100 kB到10 MB不等，因此实时请求大量3D结构模型是有限制的。XDO不会立即启动请求，但只将其放入请求XDO数组中，如表4所示。XDO数据在不需要航空图像和dem的帧中被请求。比大型XDO数据集更小的地形数据优先表示。从而为用户提供一个稳定的实时[3]显示环境。

**Lv。0**

(一)

**Lv。**

**Lv。**

**Lv。**

**Lv。**

(B)

**Lv。**

**Lv。**

**3、3、4**

**Lv。**

**Lv。**

**Lv。4**

**1 . Lv. 2 Lv. 3**

**Lv。4**

**Lv。4**

**Lv。**

**Lv。4**

**Lv。4**

**Lv。4**

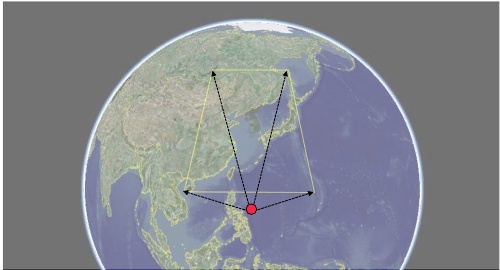
**Lv。4**

(C)

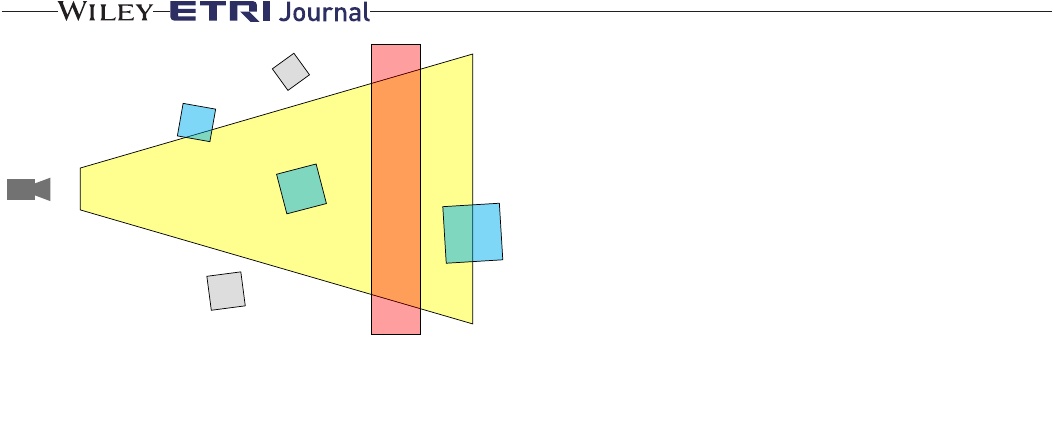
**图8渲染结果示例:(A) level 0， (B) level 0扇区在相机接近扇区时，在level 1将扇区划分为4个子扇区，(C)多个级别扇区的视图(“Lv”)。**”表示所属行业的水平)

**3.2.3 |四叉树结构**

只有13到15级的部门有三维结构模型的XDO数据。第14层包含桥梁，第15层包含建筑物，如图11A和11B所示。具体来说，level 13包含韩国独岛的地形信息，而不是建筑信息，如图11C所示。在韩国的3358个岛屿中，独岛(面积相对较小)在国家领海中占有重要位置。这个地区一直与日本的持续外交冲突联系在一起，尽管它曾被用作韩国人的居住区。因此，Dokdo被详细说明



**图9扇形角向量:红点为地球中心点，四个黑色箭头表示扇形角向量**



**284 |**

李和张成泽

初始的3D对象包含在XDO坐标系中，如图4A所示。然而，WebGL坐标系使用如图4B所示的右手坐标系。从XDO到WebGL坐标系的转换包括在模型‐视图矩阵中**M**.

(

)



。(9)

**图10 A扇区(红色)大于view‐frustum范围:蓝色块位于view‐frustum范围内(或部分位于view‐frustum范围内)，灰色块位于view‐frustum范围外**

模型‐视图矩阵根据用户界面操作而变化。**M** 中心点表示相机的主线与地球表面的交点。

中心点运动的变化与绕地球运行的卫星的运动相同。的

**表4请求扇区函数的伪代码**

**函数:请求扇区()lvArray =按级别排序(requestArray)。***requestCount = 0。*

**FOR: i = 0 to lvArray.eachLv。***长度做distArray =排序与距离(lvArray.eachLv[i])。***对于:j = 0到distArray。***长度请求扇区数据(distArray[j])。***distArray[j]有XDO**

请求XDO的索引数据。将请求的XDO数据放入requestXDOArray中。*requestCount + +。*

**IF: requestCount≥maxRequestNumber返回**

**结束**

使用两个XDO数据对象而不是DEMs，这与其他地形不同。

如果只在显示带有XDO的扇形区域时才表示3D结构模型，则由于摄像机位置的变化，通常不会显示建筑物，如图12A所示。为了在绘制三维结构模型时获得自然的显示，该平台绘制了XDO，如表5和表6所示。即使一个部门没有XDO数据，3D模型也可以显示在12-15层的部门。渲染结果如图12B所示。

**4 |转换和渲染**

虚拟相机在开放平台上的平移和旋转通过用户界面进行。根据摄像机位置确定所代表的扇区，使用WebGL库进行3D渲染。本节描述WebGL转换的过程[25,26]。基于WebGL转换管道，将XDO坐标系中的3D对象转换为屏幕坐标系中的2D对象，如图13[25]所示。

相机围绕地球旋转，同时指向地球中心。在XDO坐标系中，纬度和经度坐标的变化分别沿着z轴和y轴旋转。其中，是每个轴的旋转矩阵，是一个模型视图矩阵，该矩阵排除了倾斜变换，只适用于每个轴的旋转。**R M**′



(10)

倾斜变换强加一个翻译,检测R沿着x轴方向,通过一个旋转的倾角𝜃沿着y轴是应承担的实现首先紧随其后的是一个翻译的R沿着x轴方向,如图14所示。当加入倾斜变换后，相机将不会对准地球中心。因此，只有当倾斜角为0°时，相机的方向才会对准地球中心。

(11)



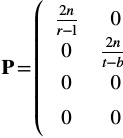
XDO坐标与WebGL坐标系之间的变换矩阵表示为(12)。



(12)

投影矩阵定义了截锥体的可视体积，即3D对象数据如何投影到屏幕上。投影矩阵取决于虚拟摄像机的设置。如果虚拟相机的属性不变，

使用原来的设置。宽、参数l=b heightare =near，宽在(13)。



l + r

*l r−*

*b + t*

*t−b*

−f + n

*f−n*

0 0⎞⎟⎟f−−n2fn⎟⎟。0⎟⎠

(13)

−1



李和张成泽

**| 285**

利用投影矩阵对剪辑坐标进行变换，最后根据视口变换矩阵V变换到屏幕坐标系中。**i**screen

·000年

⎜V =⎜⎜⎜⎝

宽度

2

0

0

0

0

heigth

2

0

0

0

0

0

0

heightwidth 2⎞⎟02⎟⎟,

⎟1⎠

(14)

**iscreen = V⋅P⋅iwebgl = V⋅P⋅M�⋅L⋅ixdo。**

(15)

(一)

如果不包含倾斜变换矩阵，则可以计算总和矩阵的逆矩阵。**L** 但是，如果旋转和平移变换矩阵重叠，则不能建立逆关系。要将WebGL坐标转换为XDO坐标，首先必须根据式(16)计算倾斜变换的逆矩阵。**L**−1 另外，根据式(17)计算逆变换。

(B)

**L−1 = M�−1 t (−R) xM�−1 R(𝜃)zM评选�−1 t (R) x,**

**我��xdo = L−1−1 iwebgl。**

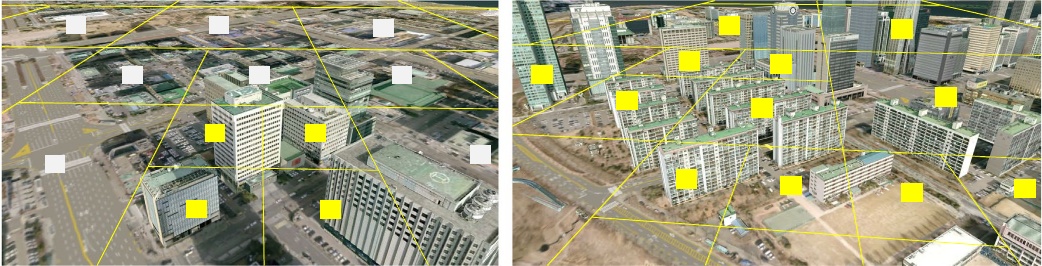
(16)(17)

**5 |实验**

(C)

**图11 XDO模型的示例。**(A)汝矣岛地区的建筑物有15层，(B)麻浦大桥有14层，以及(C)独岛有13层。

实验使用了两种类型的计算机。第一款是配备Intel®4.20 GHz Core™i7 CPU、NVIDIA GeForce GTX 1080Ti GPU和有线网络的台式电脑。另一款是配备Intel®2.70 GHz Core™i7 CPU、Intel HD 620图形处理单元(GPU)和无线网络的笔记本电脑。台式机配备了外部的GPU，而笔记本则没有。实验使用的浏览器是谷歌Chrome(版本67.0.3396.99,64位)。



**13**

**13**

**13**

**13**

**13**

**14**

**14**

**14**

**14**

**13**

**14**

**14**

**14**

**13**

**15**

**15**

**14**

**14**

**15**

**15**

**15**

**14**

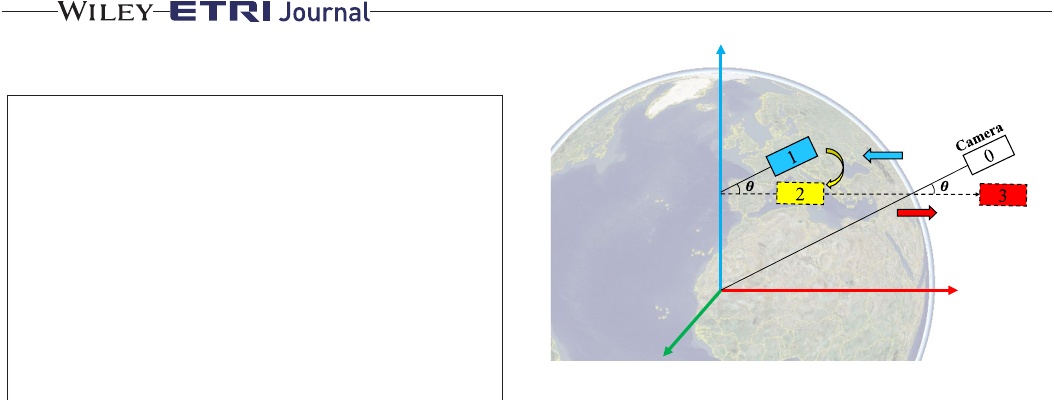
**15**

**15**

(一)

(B)

**图12 XDO的效果图示例:(A)仅在15层呈现建筑，(B)在12 - 15层呈现建筑**



**286 |**

李和张成泽

**结束**

**表5用于表示12-15层的三维结构模型的伪代码**

**功能:绘制建筑(s)**

**如果:s。***= 15级,*

绘制s的XDO数据。

**如果:s。***= 14级,*

**FOR: i= 0 to 3 DO**

绘制s.child[i]的XDO数据。

**如果:s。***水平= 13*

**FOR: i= 0 to 3DO**

**FOR: k = 0 to 3 DO**

绘制s.child[i].child[k]的XDO数据。

**如果:s。**水平= 12

**FOR: I = 0 to 3 DO**

**FOR: k = 0 to 3 DO**

**FOR: l = 0 to 3 DO**

绘制s.child[i].child[k].child[l]的XDO数据。

*Z*

**倾斜相机**

*Y*

*X*

**图14 XDO坐标系中的倾斜变换**

**表6用于表示12-15层的3D桥梁和Dokdo的伪代码**

**功能:画桥(s)如果:s。***水平= 12*

**FOR: I = 0 to 3DO**

**FOR: k = 0 to 3 DO**

绘制s.child[i].child[k]的XDO数据。**如果:s。***= 13级,*

**FOR: I = 0 to 3，**

绘制s.child[i]的XDO数据。**如果:s。***= 14级,*

绘制s的XDO数据。

**如果:s。**等级= 15

绘制s.parent的XDO数据。**结束**

**函数:绘制DOKDO(s)如果:s。***= 12级,*

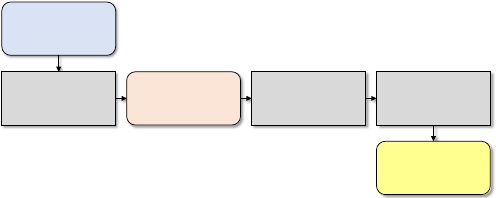
**FOR: i= 0 to 3DO**

绘制s.child[i]的XDO数据。**如果:s。***= 13级,*

绘制s的XDO数据。水平= 14

绘制s.parent的XDO数据。**如果:s。**等级= 15

绘制s.parent.parent的XDO数据。



XDO坐标

ModelView矩阵

WebGL坐标

投影矩阵

视口矩阵屏幕坐标

**图13 WebGL转换管道[25]**

表7显示了屏幕更新所在帧中的每个步骤的处理时间。它有效地表示了在106,496帧时使用2,000帧时测量的处理时间的平均值

多边形和208纹理文件被渲染。呈现相同数量的数据,价值观纬度,经度,高度,和α的值,β,γ,(10)所示,设置为相同的值。具体来说，在使用GPU的情况下，请求扇区进程仅在146帧中被请求，在不使用GPU的情况下，请求扇区进程仅在713帧中被请求。因此，平均需要146帧和713帧。执行“查找‐扇区”过程以识别要显示的扇区。渲染过程包括使用航空图像和DEM数据创建3D地形模型并将3D模型数据输入到WebGL缓冲区的步骤。每秒帧数(fps)的上限为60，具体取决于web浏览器环境。因此，使用GPU实现的最大速度是60fps。

实验结果表明，即使在不使用GPU的情况下，所提出的基于四叉树的开放3D空间信息平台的速度至少为40帧/秒。因此，我们的平台只有在使用支持WebGL的web浏览器的情况下才能增强3D空间信息地图的使用，而不考虑计算环境。

我们使用VWorld数据将处理时间与基于WebGL的首尔3D GIS[20]进行了比较。与我们处理全球区域的平台不同，首尔3D GIS只处理首尔市的区域，如图2G所示。为了加速三维建筑的渲染，所有建筑的纹理图像在一个单独的平铺块中被合并并保存在一个单独的文件中。因此，其分辨率和质量都低于VWorld数据中心提供的原始文件。

为了测量首尔三维地理信息系统的处理时间，我们使用了Chrome开发工具中的FPS表。在没有GPU的环境中，记录的速度约为34.8 fps。但是，很难将绝对速度与我们的平台进行比较，因为我们无法测量



李和张成泽

**| 287**

**表7每个步骤的处理时间(表示为2000帧使用的平均值，实验环境中web浏览器的最大帧速率为60帧每秒)**

**我们的平台**

**首尔3 d GIS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **GPU(女士)** | **没有GPU(女士)** | **GPU(女士)** | **没有GPU(女士)** |
| 找到行业 | 5.114 | 18.194 | 60 fps | 34.8帧/秒 |
| 要求部门 | 0.342 | 0.344 | 60 fps | 34.8帧/秒 |
| 呈现 | 1.469 | 6.517 | 60 fps | 34.8帧/秒 |
| 总计 | 6.925 (60 fps) | 25.055(39.9帧/秒) | 60 fps | 34.8帧/秒 |

三维数据量，如多边形或纹理文件，在首尔三维地理信息系统的情况下，在单一帧中呈现。

**6 |结论**

在本研究中，我们描述了一个基于WebGL的开放的三维空间信息平台的研究和实现方法，该平台基于韩国地形的VWorld数据。我们还描述了如何使用WebGL为3D结构模型呈现XDO数据。使用该平台，可以在不使用GPU的情况下，使用低端PC基于四叉树扇区有效地呈现高质量的、总大小≥30 TB的三维空间信息。实验表明，在非GPU计算环境中，至少可以保持40帧/秒的渲染速度。拟议的平台的性能与现有的商业平台类似，但很难披露其内部算法或方法。我们希望在开放平台的基础上开发与三维空间信息相关的各种应用。

在未来的研究工作中，我们将研究先进三维空间信息的开放平台。针对各种计算环境下的稳定性问题，提出了一种基于计算环境的优化方法。我们还打算为开发人员提供OpenAPI，以支持使用VWorld空间信息平台的各种应用程序。此外，我们打算开发一个基于Unity3D或Unity3D资产的VWorld平台，用于游戏、城市模拟或MRs.我们预计当用户购买VWorld开放平台的Unity3D资产时，他们将能够简单地创建一个VWorld 3D地图。

**ORCID**

https://orcid.org/0000-0002-7788-1921

*Ahyun李*

**参考文献**

1.VWorld数据中心由韩国国土交通和海事部运营(2017年)，网址为http://data.vworld.kr/data/v4dc\_usrmain.do。

2.张海生等，CDN方法作为空间信息开放平台服务在V - World web服务中的性能评估。Inf. Res. 24(2016)， 355-364。

3.M. S. Kim和I. S. Jang，对大量异构地理传感器数据的高效内存处理，啐。Inf. Res. 24(2016)， 313-322。

4.C. Marrin, Webgl规范，Khronos Webgl工作组，2011。

5.安吉尔和施瑞纳，交互计算机图形学:与WebGL的自顶向下方法，第七版，艾迪生‐韦斯利，波士顿，2015。

6.G. Lavoue等人，使用JavaScript和WebGL在web上流式压缩3D数据，在第18版3D web技术上。2013年6月，圣塞巴斯蒂安，第19-27页。

7.一个开源的虚拟地球框架，适用于iOS, Android和WebGL兼容的浏览器，在Proc. Int. Conf. Comput。地理空间研究应用。,华盛顿特区。，美国，2012年，第22页1 - 10。

8.一个基于浏览器的多人在线游戏框架，使用WebGL和WebSocket，在Int, Conf. multime。抛光工艺。，中国杭州，2011年，第471-474页。

9.陈文华等，基于webgl的实时容积数据交互可视化，载《三维网络技术研究进展》，2011，第137-146页。

10.吴立伟，“网络编程:W3C DOM规范”，《IEEE互联网计算》，1999年第3期，第48-54页。

11.VWorld，基于WebGL的三维空间信息开放平台，由韩国国土交通和海事部运营(2017)，http://map.vworld.kr/map/wcmaps提供。做的。

12.OpenGlobus: A.交互式3D地图的JavaScript库，2016年，http://www.openglobus.org。

13.Cesium:一个开放源码的JavaScript库，用于世界级的3D地球仪和地图、分析图形公司(AGI)和宾利系统(2012)，可在https://cesiumjs.org上获得。

14.谷歌地球，谷歌. 2017，可在https://earth.google.com/网站获得。

15.闵国强等，地图空中更新导航服务系统框架，ETRI J. 33(2011)， 476-486。



**288 |**

李和张成泽

16.唐建华，龚国雄，大型地形数据库建模与实时绘制技术，北京:中国地质大学出版社。Simul. 18(2006)， 453-456。

17.虚拟地理信息系统:一个实时的三维地理信息系统，载《可视化研究进展》95年，亚特兰大，美国，1995年，第94-100页。

18.m . Krivokuća et al ., 3 d网格几何的渐进压缩利用冗余帧字典的稀疏近似,ETRI j . 39 (2017), 1 - 12。

19.首尔3D, GIS (2018)， http://3dgis.seoul.go.kr。

20.R. Stanaway, GDA94, ITRF, WGS84:有什么区别吗?2007年，澳大利亚，与动态数据合作。

21.吴建华，“四叉树及相关层次数据结构”，中国计算机学会计算调查(CSUR)，第16期(1984)，第184 - 260页。

22.J. Kim和I. K. Jeong，基于单图像的三维树和生长模型重建，ETRI J. 36(2014)， 450-459。

23.Broggi等人，使用有理B样条曲面和立体视觉的越野自动地面车辆地形测绘，IEEE Intell，车辆信号。，《黄金海岸》，澳大利亚，2013年，第648-653页。

24.李泽彬等，基于数字航空影像的三维地图生成，ISPRS，摄影。遥感。60(2006)，413-427。

25.吴敏等，OpenGL编程指南:学习OpenGL 1.2版的官方指南，第三版，Addison‐Wesley, 1999。

26.《计算机图形学导论》，艾迪生‐韦斯利，雷丁，MA, 1994。

**作者传记**

**李爱贤博士毕业于韩国大田科技大学计算机科学专业。**2011年至2012年，他在LG电子(LG Electronics Inc.)担任工程师。平泽，韩国众议员。他已经

曾在电子和电信研究所担任研究员。



从2016年开始。他的研究兴趣包括计算机视觉、增强现实、机器人和空间信息。

**张茵星分别于1999年和2001年获得釜山国立大学计算机工程学士学位和硕士学位。**自2001年以来，他一直是电子和信息技术公司的高级研究员

大韩民国大田电信研究所，他一直在那里工作



在釜山国立大学攻读计算机工程博士学位。他的主要研究领域包括地理传感器、导航和定位服务平台。