

Research on Visualization of 3D Container Ship Map Based on Web

Mingzheng Yu, Bin Xu, Jia Chen

Dalian Maritime University, Dalian Liaoning
Email: 843480700@qq.com

Received: Aug. 24th, 2018; accepted: Sep. 7th, 2018; published: Sep. 14th, 2018

Abstract

The use of container ship maps can effectively organize port container loading and unloading operations, ensure the safe use of ships and cargo, rationally utilize port resources and improve work efficiency. This paper mainly uses 3D visualization technology to present 3D container ship map on the Web, and combines port container management system to realize the sharing and deployment of port container information resources. WebGL combined with JavaScript is used to develop 3D container ship map based on J2EE framework and data interaction is used to realize the dynamic effect of container ship chart on web page, which are different from most client port management systems, ignoring the constraints of time, environment, personnel and other conditions in the port business. It is lightweight and flexible, which is relatively rare in today's development tools. Besides, the Web-based features can extend the development of 3D container ship map visualization technology to the Web mobile terminal, and realize a leap in the port information management platform, which is of great significance for the entire port business management.

Keywords

3D Visualization, Container Ship Map, WebGL, Web Mobile

基于Web的三维集装箱船图可视化研究

于明正, 徐 斌, 陈 佳

大连海事大学, 辽宁 大连
Email: 843480700@qq.com

收稿日期: 2018年8月24日; 录用日期: 2018年9月7日; 发布日期: 2018年9月14日

摘 要

使用集装箱船图可以有效组织港口集装箱装卸作业, 保证船舶及货物的安全使用, 合理利用港口资源并

提高工作效率。本文主要利用三维可视化技术在Web上呈现三维集装箱船图,并结合港口集装箱管理系统实现港口集装箱信息资源的共享和调配。使用WebGL结合JavaScript基于J2EE框架进行三维集装箱船图的开发并利用数据交互实现网页上集装箱船图的动态功能,这有别于大部分客户端港口管理系统,忽略了港口业务中时间、环境、人员等条件的约束,具有轻便性和灵活性,这在如今的开发手段中是比较少见的。不仅如此,基于Web的特性可以让三维集装箱船图可视化技术的开发拓展到Web移动端,实现港口信息管理平台的一次飞跃,这对于整个港口业务管理来说都具有重要的意义。

关键词

三维可视化, 集装箱船图, WebGL, Web移动端

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

对于现阶段的海上货轮运输业来说,传统的集装箱港口管理已经不能满足各个码头现代化管理的需求,港口业务的数据管理与三维可视化场景相结合能够成为提高核心竞争力的有效途径,直观的三维可视化展示可以使整个业务流程更加核心化、规范化、标准化,进一步提高码头生产的专业深度,有效加快港口业务速度,提高船舶与集装箱业务操作能力,使船舶运营、码头装卸和堆场管理更加协调、科学和高效。而如今有许多的成熟的技术可以为我们提供实现数据可视化的平台开发基础。HTML5 标准及其所包含或所关联的 Canvas、SVG、WebGL 等相关技术的出现将数据可视化带到了 Web 平台上。因此,在 Web 端完成三维场景的展示甚至通过移动端来查看三维场景可以成为一种新的手段。假如港口集装箱的三维场景可以直观的在浏览器上展示,甚至这种三维场景可以与码头集装箱管理系统相结合,完成港口集装箱数据的关联和集成,快速更新集装箱的相关信息,实现港口资源的有效利用和共享,那么对于集装箱装卸流程和整个港口的业务管理来说都是十分有价值的事情。

2. 系统设计原理

本文的设计原理是通过 JavaScript 来创建和管理数据并借助 HTML5 的技术实现三维可视化的目的,这种技术在多年的完善和发展后是可以得到充分的验证和实现的。在 HTML5 中增加了 canvas 这个基于 JavaScript 绘图的 API。为实现 Web 平台的数据三维动态可视化提供了这样一个思路,可以利用 HTML5 的 canvas 将一张画布放置于 Web 前端,使用 WebGL 提供的 JavaScript 编程接口在画布上进行 3D 绘图,通过 canvas 来获取 WebGL 特定的绘制上下文,这里使用 WebGL 框架中的 Three.js 来进行绘制,这样就可以在 Web 平台上显示三维数据模型。当然还要实现数据的动态交互,这里用 JSON (JavaScript Object Notation)作为平台间交互的数据传输格式,在 J2EE 架构的基础上利用 JQuery 结合 Ajax 技术,实现用 HTTP Get 和 HTTP Post 方法从远程服务器上发送或请求 JSON,到了后台 Servlet 里实现与数据库的交互 [1]。最终呈现的结果是 Web 上的三维对象可以与数据库实现动态交互。不论是绘制三维数据模型还是与数据进行动态交互,在框架的整体构建上都共同用到了 JavaScript (本文所用的 JQuery 是 JavaScript 的简化版本)来开发,JavaScript 成为了联系两个平台之间的纽带,这使数据的动态三维可视化成为可能。

3. 三维集装箱船图的动态可视化设计

3.1. 船箱位在三维空间中的抽象化处理

三维集装箱船图的核心内容就是对集装箱在船舶上的船箱位仿真成三维场景并能利用数据化的思想进行集中管理，考虑到三维集装箱船图在生成方法及渲染等因素的复杂性，为了船箱位的数据在处理时更加容易操作和理解，所以采用贴合语言标准的设计方式以直观三维坐标系来转换船箱位坐标，抽象出现实与计算机数据之间的中介。在设计中忽略集装箱配载时实际因素的影响，按照船箱位的国际标准进行三维抽象化，用抽象后的数据与系统数据库进行交互，在三维场景中集装箱的定位采用 x, y, z 三维坐标系来表示，定位的结果是一个三维坐标，具体的标准如下所示[2]。

x 轴：以船舶纵向(前后方向)为 x 轴，船首至船尾为 x 轴正方向。

y 轴：以船舶横向(左右方向)为 y 轴，右舷至左舷为 y 轴正方向。

z 轴：以船舶垂直向(上下方向)为 z 轴，甲板层向上为 z 轴正方向。

如图 1 所示对标准船箱位进行三维抽象化处理，得到在三维场景中集装箱的定位坐标结果，其中 20 ft 箱在三个坐标轴上用自然数表示，40 ft 箱在 x 轴上用介于两个自然数之间的平均数表示。

3.2. 集装箱的 3D 建模方法

3D 建模核心的内容是对模型位置的计算与确立，目前集装箱在船舶上的定位是按照国际 ISO9711-1:1990 标准规定的，这是一种对集装箱(分为 20 ft、40 ft)在船上标准纵向布置并以 6 为数字对箱位进行表示的方法[3]。具体的规则如下：

行号(Bay)：指船舶纵向序列坐标，用两位数字表示，编号规则是从船首至船尾依次：01, 02, 03, 04, 05, …，其中 20 ft 箱用奇数表示，40 ft 箱用介于两个奇数之间的偶数表示。

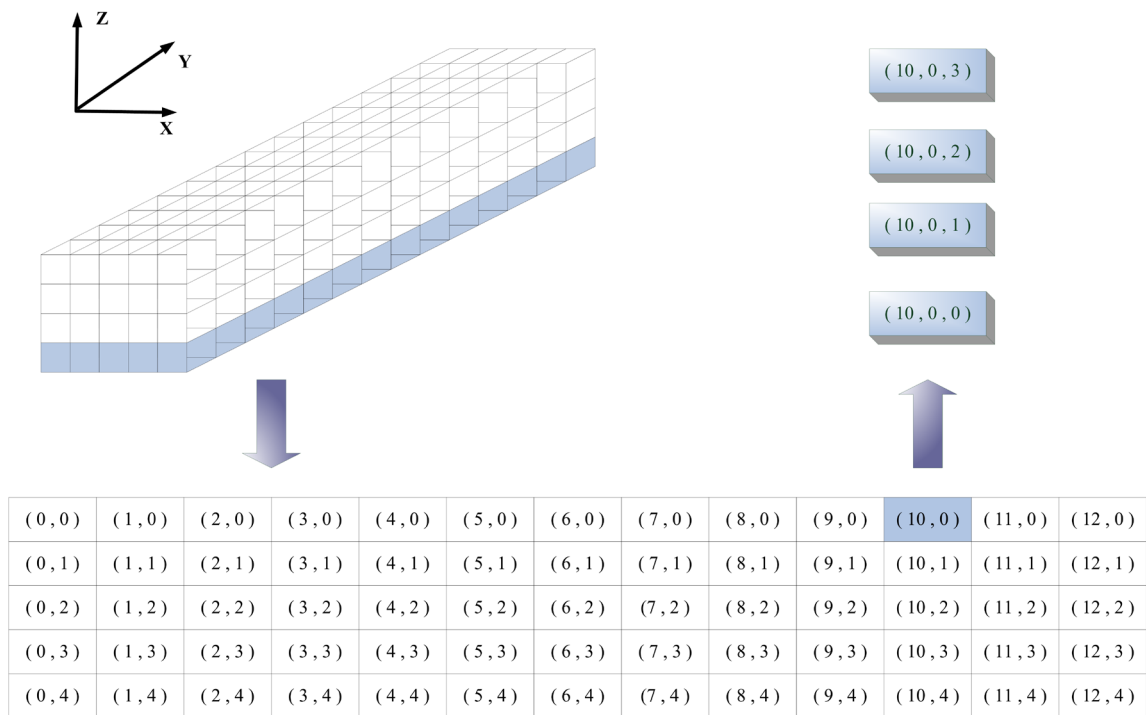


Figure 1. Method on boat box abstraction

图 1. 船箱位抽象化处理方法

列号(Row): 指船舶横向序列坐标, 用两位数字表示, 编号规则是从船舶中间开始依次向左右两边排序, 按照“左偶右奇”的原则, 中间向左舷依次 02, 04, 06, ..., 中间向右舷依次 01, 03, 05, ..., 若总列数为奇数时中间列号为 00, 总列数为偶数时则没有中间列号。

层号(Tier): 甲板与舱内分开表示, 舱内自下而上依次 02, 04, 06, ..., 甲板从底层向上依次 82, 84, 86, ... [4], 在本次论文里只研究甲板层的集装箱配载。

由于在现实生活中, 每条船实际的船箱位排布虽然不同, 但是集装箱定位标准是相同的, 只要能够获得一些基本的船舶基本参数信息, 就能够准确定位集装箱在船舶上的位置。所以重要的是确立标准船箱位与三维空间中的模型坐标之间联系。在 3D 建模中系统能够利用标准船箱位的 6 位数字迅速转化为三维空间的模型坐标参数, 确定集装箱模型在三维空间的位置, 具体过程和参数说明如下所示:

Bay: 船箱位行号

Row: 船箱位列号

Tier: 船箱位层号

TotBay: 船舶行位总数

TotRow: 船舶列位总数

TotTier: 船舶层位总数

x: 三维模型 x 向坐标

y: 三维模型 y 向坐标

z: 三维模型 z 向坐标

mod(a, b): 用于判断 a 是否能被 b 整除的函数, 返回 1 代表可以整除, 返回 0 则不能。

三维坐标 x 的计算公式:

$$x = \begin{cases} \frac{Bay-1}{2}, \text{mod}(Bay, 2) = 0 \\ \frac{Bay}{2} - 0.5, \text{mod}(Bay, 2) = 1 \end{cases}$$

三维坐标 y 的计算公式:

$$y = \begin{cases} \frac{TotRow-1}{2} - \frac{Row+1}{2}, \text{mod}(Row, 2) = 0 \text{ mod}(TotRow, 2) = 0 \\ \frac{TotRow-1}{2} + \frac{Row}{2}, \text{mod}(Row, 2) = 1 \text{ mod}(TotRow, 2) = 0 \\ \frac{TotRow}{2} - \frac{Row+1}{2}, \text{mod}(Row, 2) = 0 \text{ mod}(TotRow, 2) = 1 \\ \frac{TotRow}{2} + \frac{Row}{2} - 1, \text{mod}(Row, 2) = 1 \text{ mod}(TotRow, 2) = 1 \end{cases}$$

三维坐标 z 的计算公式:

$$z = \frac{Tier - 82}{2}$$

通过对已知参数进行上述公式的计算得出集装箱模型在三维空间的坐标为(x, y, z), 用于三维空间的构建。针对集装箱进行 3D 建模是使用 Three.js 技术结合 JQuery 的相关语言实现的, 集装箱 3D 模型拥有一定程度的复用性和灵活性, 3D 集装箱模型在技术层面上主要包括两个方面, 一方面是基础模型数据的处理, 它是指集装箱建模时对 Three.js 标准里的形状(Geometry)对象和材质(Material)对象的设置, 之后在

此基础上实现网格(Mesh)化处理;另一方面是纹理数据的处理,它是指将处理过的外部图片文件(在预先采集好的集装箱相关图片上使用 Photoshop 进行亮度、色彩、清晰度和人为痕迹消除的调整以达到浏览器呈现的要求)借助 Texture 功能进行模型材质的自动贴图功能,除此之外还要对三维模型进行更多细节性的处理,具体过程和伪代码如下:

使用 Texture 功能导入处理过的外部图片文件生成纹理数据,将纹理数据应用于材质对象中。

```
var texture=THREE.ImageUtils.loadTexture(imageurl)
```

```
var cubeMaterial=new THREE.MeshMaterial(map:texture)}
```

使用图形库提供的标准几何图形建立长方体形状对象并设置合适的大小,与材质对象一起应用 Mesh (网格)化处理生成三维模型。

```
var cubeGeometry=new THREE.CubeGeometry(length, width, height)
```

```
var cube=new THREE.Mesh(cubeGeometry, cubeMaterial)
```

此时的三维模型已经存在于内存中,这里规定三维模型的 position 属性以决定模型在场景中的位置,最后将它添加到场景中去。

```
cube.position.x,y,z=(x, y, z)
```

```
scene.add(cube)
```

集装箱 3D 建模效果如图 2 所示。

3.3. 三维集装箱船图的动态交互原理

3.3.1. 数据可视化动态交互原理

可视化数据交互是基于 J2EE 框架实现的,以全局的角度来看,数据是从数据库到服务端再到前端接口最后经过三维渲染效果显示在浏览器上的。进行可视化数据交互的核心由三部分组成,第一部分是浏览器将封装好的 JSON 数据利用 Ajax 发送给服务器端接收回调函数并判断操作是否成功(回调函数也可以收到从服务器端返回的参数);第二部分是在服务器端接收 JSON 数据,并在此基础上与数据库进行交互,

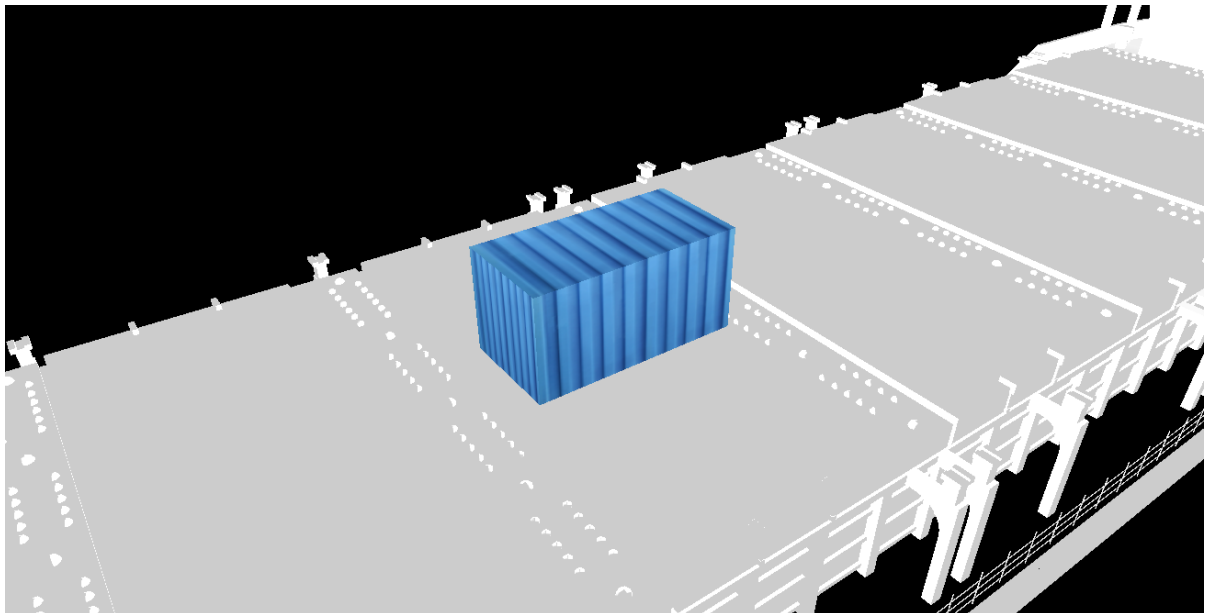


Figure 2. Container 3D modeling

图 2. 集装箱 3D 建模效果图

交互结束后返回结果给浏览器；第三部分是浏览器在交互数据成功后基于回调函数的进一步操作。三维集装箱船图的基础是通过浏览器与数据库的数据交互，完成依据数据实现三维可视化的功能。也就是说要做到浏览器不仅能宏观的展现集装箱船图，还能够根据集装箱货船的作业流程实时的反应集装箱船图的变化情况，展示动态的效果，核心伪代码如下：

```
function RefreshMessage()
var json = {'messageID': message}
$.get(URL, {data: JSON.stringify(json)}, function(rsfn))
.....
end function
end function
```

3.3.2. 三维集装箱船图的基本操作原理

三维集装箱船图的基本操作有鼠标悬停(mouseover)、鼠标单击(click)和鼠标双击(dblick)等，这些操作都可以达到人机交互的目的。用 Raycaster 类向鼠标锁定的集装箱发射射线后，可以接收被射中集装箱的所有结果集并计算出集装箱的 position 坐标，该坐标与数据库里的坐标对应，从而可以完成对集装箱的定位进行接下来的操作，核心伪代码如下：

```
var standardVector = new THREE.Vector3(x, y, z)
var worldVector = standardVector.unproject(camera)
var ray = new worldVector.sub(camera.position).normalize()
var raycaster = new THREE.Raycaster(camera.position, ray)
var intersects = raycaster.intersectObjects(scene.children)
```

为交互操作设立事件监听器，当指定事件发生时系统利用事件监听触发对应的方法，经过射线相交返回 intersects 对象，intersects 对象是一个代表所有关联该事件的对象集数组，一般选用 intersects [0]作为事件触发的第一个集装箱对象，取出已锁定集装箱的 position 系数坐标进行接下来的数据交互行为，核心伪代码如下：

```
addEventListener('mouseover'/'click'/'dblick',Rayfunc)
function Rayfunc
if intersects.length>0 then
cofunc(intersects[0].object.position.x,intersects[0].object.position.y,intersects[0].object.position.z)
.....
endif
end function
```

除了一些人机交互的基本操作外，三维集装箱船图还包括了视角旋转、视角远近调节、视图切换、集装箱拖动、集装箱的手动配载等多个操作，本文不做详细的阐述。

4. 集装箱船图的三维动态可视化实现

4.1. 集装箱船图三维可视化的结果

利用上述技术手段可以对港口船舶的集装箱船箱位进行全面的的管理，将集装箱通过可视化的方法显示在 Web 页上面。在网页上可以看到一个小规格的集装箱货船，甲板上的集装箱可以通过数据库实时更新并反馈集装箱详细信息给用户。假设每个集装箱之间的特殊差异忽略不计，为了可视化的效果能够直

观的展示且不影响其他的系统功能，我们让每一个集装箱材质纹理的颜色随机产生，这种方式可以让浏览器上的效果特殊化且对于用户体验来说也是极为有必要的。如图 3 所示，利用上述技术可以在 Web 页上直观的显示三维集装箱船图的整体分布情况。

不仅如此，Web 页上的集装箱分布可视化情况可以与数据库进行实时的动态交互，这样一来集装箱船在进行装船或卸船的操作时，系统能够实时对数据库的数据进行更新，假如集装箱船图连接在可更新的数据库上，那么数据库的更新结果会直观的显示在 Web 页上并实时的做出调整，通过数据库的更新可以看到在集装箱货轮装船过程中实时反映的三维集装箱船图的动态可视化分布情况。同时在业务活动中也能详细了解集装箱的资源利用和实时动态，这一定程度的简化了港口业务活动提高了港口管理的工作效率。

4.2. 三维集装箱船图向移动端的拓展

基于 Web 的三维集装箱船图也可以向移动端进行拓展，因为 Web 开发的技术可以兼容移动端的开发，所以在选择上可以脱离 App 应用开发转向移动端 Web 开发或者混合开发，在不需要变动主体的情况下简单的给移动端一个来自服务器端的“视图”。比如这里有一个移动端开发的新方法，利用手机扫码工具扫描已经准备好的二维码，这样在移动端就可以浏览与 Web 上大致相同的内容。整个操作过程比较简单，而且数据交互地实时性比较好，经过实验发现如果网络不是非常阻塞，移动端的展示与 Web 上的实现并无明显差异。在移动端的三维集装箱可视化实现如图 4 所示。



Figure 3. 3D container ship map renderings

图 3. 三维集装箱船图效果

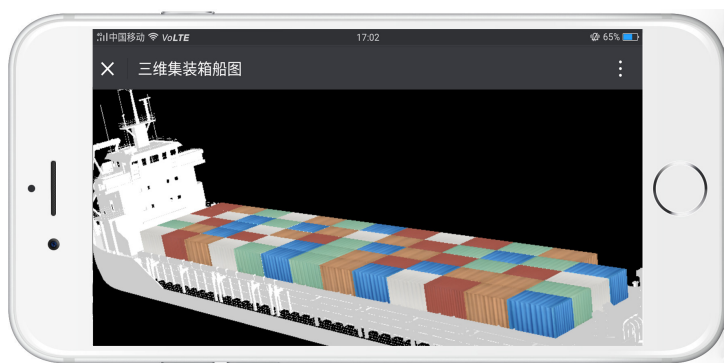


Figure 4. 3D container ship map on the mobile side

图 4. 移动端的三维集装箱船图

因为二维码的生成是由网页的地址直接决定的,所以只要将二维码放在手机或者移动设备可以扫到的地方,就能实现移动端的三维集装箱船图可视化,工作人员的工作环境也可以依据实际情况而进行相应的调整。对于整个港口管理来说,港口各个工作人员之间可以实现信息资源的共享,这种操作推动了整个港口管理系统的现代化进程,也可以提高港口管理人员的工作素质。

5. 总结

本文主要研究并实现了基于 Web 的三维集装箱船图的可视化设计,并将它拓展到了移动端来应用,提高了港口业务管理的整体水平。对于整个港口管理来说,三维集装箱船图可以使码头装卸作业的资源利用更加合理,工作人员通过直观的可视化结果对装卸作业进行调整。同时它还具备港口装卸评估预警能力,利用现场的港口业务信息预测出码头的整体装卸能力,评估预测整个港口的集装箱装卸流程,当评估中出现问题时可以及时提醒和纠正装卸策略,以确保货轮和货物的安全。这种方式满足了港口业务管理的现代化需求,丰富了管理过程的技术手段,对未来港口管理的发展有着重要的意义。

参考文献

- [1] 刘娜. 济南市技师学院教务管理系统的设计与实现[D]: [硕士学位论文]. 山东: 山东大学, 2016.
- [2] 崔健. 规则波中箱型驳船顶浪航行波浪力特征的研究[D]: [硕士学位论文]. 大连: 大连海事大学, 2016.
- [3] 张维英. 集装箱船全航线配载智能优化研究[D]: [博士学位论文]. 大连: 大连理工大学, 2006.
- [4] 马云崙. 集装箱配载数据差异对比及配载图校正[D]: [硕士学位论文]. 大连: 大连海事大学, 2012.

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: 2326-3431, 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: ojtt@hanspub.org