

# 基于 WebGL 技术三维模型呈现研究

□ 聂帅 秦爽 麦文

**摘要:**应用WebGL技术可以让浏览器免去下载其他插件情况下渲染出三维图像,封装了WebGL一些底层API图形接口的Three.js引擎,使得在网页上高效地进行创建三维模型提供了可能。本文先对WebGL的3D绘图功能进行简单的介绍,再通过对Three.js开源框架进行研究,实现了基于网页的虚拟三维模型的呈现,从而发挥出WebGL免浏览器插件、经过OpenGL接口加速图形渲染的优势。

**关键词:** WebGL; Three.js; 三维模型; 网页

随着互联网技术的不断发展,Web 信息技术研究核心逐渐转向 Web3D 的应用。近些年来,在前端领域内 HTML5 的发展逐渐的趋向成熟,在制作 3D 网站上 WebGL 扮演了愈加重要的角色,使得在网页上呈现虚拟三维模型成为可能。

在生活中当我们计划去某些博物馆、停车场等一些地方,我们有时想提前体验一下它们的大体的结构,这时就需要一个网络上的展示互动平台,借助于此平台用户可以在线观察展品虚拟模型的三维效果。为了将虚拟三维模型渲染出,现在许多 3D 可视化图像制作软件对操作系统配置要求较高,且实现过程中工程量较大。

本文致力于研究一种开源化的 3D 引擎-Three.js,它是采用 JavaScript 编写封装了 WebGL 一些功能的 3D 库,编写少量的代码,即可将三维模型渲染出来。

## 一、基于 WebGL 的 3D 绘图技术简要分析

(1) WebGL 通过联合 JavaScript 与 OpenGL ES 2.0,在 3D 绘图技术中变的愈加标准,其中 OpenGL ES 在制作三维图形时使用的功能接口是属于 OpenGL 的 API 子集,许多嵌入式设施均是采用 OpenGL ES 来设计的,H5 中绘图元素 Canvas 只能绘制一些 2D 图形,为了让 H5 的 Canvas 元素对 3D 图形进行渲染,这时可以增加一个 JavaScript 绑定到 OpenGL ES 2.0 中。从而使用系统显卡即可加速浏览器中 3D 网页的渲染。

(2) 当前 Web 三维动画存在交互上的两个问题:第一,需要在浏览器端安装大量的插件才可对一些三维模型进行渲染;第二,有些三维动画只能在指定的平台上才可运行,不可跨平台;而 WebGL 很好的克服了以上两个局限性,它是经过 HTML 脚本对三维动画渲染,有着较好的兼容性,且跨平台性较强。利用 WebGL 进行绘制 3D 图形可大致分为以下六个步骤:

①创建画布。在进行绘制 3D 图形时首先我们的创建出一个图形容器,以供用户在其中开展绘图操作,一般用 Canvas 标签。

②初始化 WebGL 的上下文。在创建

完画布之后,接下来就要在此画布基础之上运用一定的方法创建出上下文对象,以后所有的绘图操作均在此对象中展开。

③构建顶点数组。我们在下文中讲到的网格对象又称为顶点数组,作为顶点数据存储在 WebGL 缓冲区,以供着色器来做相应处理。

④矩阵变换。通过对矩阵做出一定的运算处理,从而实现矩阵对应的模型对象做出相应的变换。

⑤着色器创建。启动程序将着色器连接到 CPU,通过操作构建顶点数组数据,进而使得创建场景速度大大提升。

⑥绘制出面元。将创建出的着色器在画布上渲染出来。以上用 WebGL 直接进行 3D 图像的绘制虽然取得了良好的效果,但在实现过程中需要编写大量的代码,为了更加高效快捷地将 3D 图形渲染出来,我们可以采用 WebGL 进行简化与封装形成的

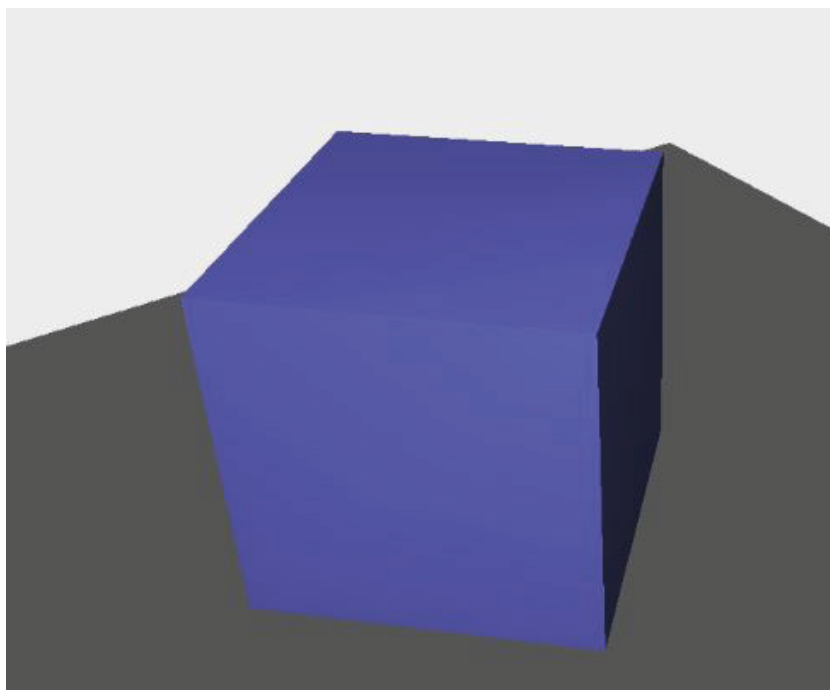


图1 在Three.js场景中构建的正方体网格



图2 博物馆模型



```
//创建场景
var scene=new THREE.Scene();
//创建摄像机
var camera=new THREE.PerspectiveCamera(45,600/400,0.1,1000);
//创建出渲染器并设置一定的大小,并设置出阴影效果
var renderer=new THREE.WebGLRenderer();
renderer.setClearColor(new THREE.Color(0xEEEEEE,1.0));
renderer.setSize(600,400);
renderer.shadowMapEnabled=true;
//向场景中添加立方体网格
var cubeGeometry=new THREE.CubeGeometry(10,10,10);
var cubeMaterial=new THREE.MeshLambertMaterial({color:0x00ffff});
cube=new THREE.Mesh(cubeGeometry, cubeMaterial);
scene.add(cube);
//若是向场景中添加外部软件建模模型时,如博物馆模型时,代码如下
var mtlLoader=new THREE.MTLLoader();
mtlLoader.setPath('/asset/obj/'); //加载文件的路径
mtlLoader.load('博物馆.mtl',function(mtl)//这里的mtl指的是加载的模型材质
{
    mtl.preload(); //预加载
    var objModel=new THREE.OBJLoader();
    objModel.setMaterials(mtl);
    objModel.setPath('/asset/obj/');
    objModel.load('博物馆.obj',function(object) //object是加载模型几何体
    {
        for(k in object.children){//对加载的几何体赋予阴影效果
            object.children[k].castShadow=true;
            object.children[k].receiveShadow=true;
        }

        object.position.y=50;
        object.rotation.y=0.5;
        //object.scale.set(15,15,15);
        scene.add(object);
    });
});
```

图3

```
//定位摄像机的位置并指向场景的中心
camera.position.x=-30;
camera.position.y=40;
camera.position.z=30;
camera.lookAt(scene.position);
//在场景中添加环境光
var ambientLight=new THREE.AmbientLight(0x0c0c0c);
scene.add(ambientLight);
//添加聚光灯并赋予其阴影效果
var spotLight=new THREE.SpotLight(0xfffff);
spotLight.position.set(-40,60,20);
spotLight.castShadow=true;
scene.add(spotLight);
//将渲染器的输出添加到指定的div元素中
document.getElementById("WebGL-output").appendChild(renderer.domElement);
```

图4

一个3D图形库Three.js,它作为一个轻量级的前端框架,运行在浏览器中,用于绘制计算机3D图形图像,对Web开发人员来说,不须学习较为复杂的计算机3D图形学的操作,通过调用相应的Three.js一些API接口就可完成3D场景的渲染。

## 二、基于Three.js虚拟三维模型的展示

(1) Three.js创建三维模型对象是基于网页端形式来实现的,能够以三维形式在网页上渲染并展示给用户,它封装了场景、摄像机、加载器、光源、材质、渲染器、网格、粒子等。从而使得用户更加简单、直观地在网页中呈现三维图形。通过Three.js完成3D模型的渲染并呈现,大致分为以下几步:

①三维场景的创建。该场景为一个对象数组,Three.js中

所有的对象,包括摄像机、网格、渲染器等对象,均保存到此数组中。

②设置相机。Three.js场景中相机决定了图形展示的角度,其中透视投影相机将每个点都投影到三维立体空间中,而正交投影相机投影时是把物体投射到二维平面。

③设置光源的光照。在三维场景中经过增加一些光源的光照,可以给场景增添几份美感,在Three.js场景中根据光照类型不同可把光源分为:聚光灯、点光源、面光源、半球光等。

④加入渲染器。渲染器决定了渲染的输出结果是什么样子,同时又将三维物体绘制到浏览器页面上。

⑤加载文件。首先是加载Three.js中的库文件,即引入three.js文件,当我们想要在场景中加入其他的模型时我们就要进行引入相应的库文件,比如加载文件为obj模型类型,就引入OBJLoader.js库文件。

⑥模型的构建与导入。以上几个步骤主要构造出了一个三维网页大致框架,随后就可以在其中绘制网格模型或导入其他3D软件制作的三维模型了,使用Three.js自带的一些方法可以创建如正方体、球等简易的网格体,但是对于博物馆、车辆等复杂模型,采用Three.js自身一些方法来构建会比较困难,这时可以借助于其他3D软件来制作出并转化为指定的格式,再通过指定的加载器加载出对应的模型对象,常见的加载器有:JSONLoader、OBJLoader、MTLLoader等加载器。以下图1与图2分别是展示的网格体与复杂三维模型的效果图。

(2) 以下展示的是场景中正方体网格效果图,如图1所示。

另外通过加载其他软件制作的模型同样可单独的呈现在Three.js场景中,图2是使用Blender建模软件建出的博物馆三维模型,效果如图2。

图3与图4是展示的呈现模型关键部分代码,在HBuilder软件中进行编辑:

我们所展示的是渲染场景的核心代码,且向场景中添加的元素均为静态形式的,若我们想要让元素动起来,就要向场景中添加动画效果,目前一般使用requestAnimationFrame()方法来完成,通过不断更新屏幕动画,达到动画的效果。

在three.js中,也能够应用动画引擎来完成一些动画特效,Tween.js是一款可产生平滑过渡动画效果的js库,利用它能够完成某个对象的指定属性的属性值在规定时间内不断变化来完成动画效果,以下代码展示了一个立方体网格经过2000ms在水平方向上移动到-1000的位置上的动画。

```
var tweenBack=new TWEEN.Tween(cube.position).
```





to({x:-1000}, 2000).start()。

### 三、结束语

本文旨在采用 WebGL 封装好的一种 3D 库 Three.js, 快速地在浏览器页面中呈现出虚拟的三维模型, 并对其中的如何添加动画效果做了简要的分析, 提出了一种无需下载其他插件的渲染方法, 达到了很好的一种展示效果。将来 WebGL 技术会与其他前端技术相结合来完成 3D 网站的制作, 值得我们进一步的研究和学习。

### 参考文献

[1] 任宏康, 祝若鑫, 李风光. 基于 Three.js 的真实三维地形可视化设计与实现 [J]. 测绘与空间地理信息, 2015, 38 (10): 51-54.

#### (上接第 15 页)

自身的理论知识, 提高计算机操作技术能力, 增加培训次数, 确保培训的质量。

并且, 为了能够和现代化运用中的计算机科学技术设备相互配合, 企业就要积极的引导员工在是市场上使用最为普遍的计算机科学与技术。

#### (二) 加强信息保障, 完善网络完全

在信息化发展的今天, 网络信息发展非常的泛滥, 所以, 用户首先就要对自身的计算机信息安全加强保护, 提高自身安全意识, 在使用计算机时, 要严格的按照操作步骤执行, 正确的使用计算机。

国家相关部门在此方面也要加强监管力度, 对一些非法盗取机密的行为要严厉的打击惩罚<sup>[6]</sup>。企业针对内部工作人员在使用计算机方面也要制定相应的规章制度, 建立专门的网络信息安全管理监督部门, 完善信息安全体系, 保障网络信息系统可以在安全的环境下稳定发展。

#### (三) 提升监管力度

国家相关部门对于信息泄露问题要高度的重视, 对于违法犯罪现象要严厉打击。并且要求相关单位组织建立和完善监督管理机制, 制定防范监督措施, 在计算机技术的应用方面要加大监管力度, 拓宽管理范围, 利用真实有效的法律保障信息的安全, 同时在加强违法犯罪行为打击力度的过程中, 要不断的提升管理能力。

#### (上接第 37 页)

随着病毒种类和数量的不断增多, 传输速度和破坏力也在逐渐增强, 如何去应对和防御也变得越来越重要。因此, 我们需要强化病毒的防御技术, 提高对网络病毒的防控。首先, 我们需要科学的认识和分析各种病毒, 了解各种病毒对计算机网络的破坏。然后我们需要定期的利用病毒的防御技术对病毒进行检测监控。最后, 我们需要用科学的手段强化病毒防御技术, 提高病毒防御能力, 从而更好地保障计算机的网络安全, 杜绝网络信息的泄露等问题。

#### (三) 做好网络安全性能评估

一个好的网络环境离不开网络的安全性能评估, 为降低风险网络安全评估尤为重要。安全性能评估不仅能够提高网络的安全, 还能加强对数据通讯的维护。定期的网络安全评估能很好的预防风险, 还能更全面更细致的对整个网络系统进行掌握, 真正实现网络安全。

#### (四) 推动加密技术升级提升身份验证技术

出现网络安全问题, 一般是在信息的传输过程中发现发生泄漏, 使犯罪分子有可乘之机, 根据这些漏洞来编造信息, 使发生损失。为了保证信息传输的安全性, 能从两方面加以解决, 一方面是提高加密技术, 另一方面是开启身份验证。加密技术可以使信息在传输过程中更加安全。开启身份验证, 在信息传输中只有特定身份的人能进行信息的浏览与获取, 从而杜绝不法分子利用漏洞获得信息从而非法获利的行为。

#### (五) 个人使用者防范意识的提高

[2] 荣艳冬. 基于 WebGL 的 3D 技术在网页中的运用 [J]. 信息安全与技术, 2015 (8): 90-92.

[3] 郑华, 刘洋. 基于 WebGL 的三维模型及其信息化技术研究 [J]. 石家庄铁路职业技术学院学报, 2017, 16 (1): 64-70.

[4] 曹焱. 虚拟现实的技术瓶颈 [J]. 科技导报, 2016, 34 (15): 94.

(作者单位: 四川师范大学)

作者简介: 聂帅 (1993 ~), 男, 在读研究生, 研究方向为基于 web 前端三维地图开。

基金项目: 四川省高等学校科技创新重大培育项目; 项目编号: 15CZ004

### 四、结语

总之, 加强计算机科学与技术的研究与应用, 不仅有利于推动社会发展和国家经济水平的提高, 对于我国人民自日常生活中的需求能够有效的满足, 推动社会稳定和谐发展, 当然现在我国计算机科学与技术的应用中还是存在部分问题, 但是只有认真的去分析了解问题源头, 就可以有效的解决这些问题, 促进我国计算机科学与技术水平稳定安全发展<sup>[7]</sup>。

### 参考文献

[1] 孔令华. 计算机科学与技术的现代化应用 [J]. 电子技术与软件工程, 2017 (17): 145.

[2] 高泽寒. 试论计算机科学与技术的现代化运用 [J]. 电脑迷, 2017 (08): 41.

[3] 邓达烈. 浅谈计算机科学与技术的现代化运用 [J]. 科技展望, 2016, 26 (29): 9.

[4] 叶津凌, 王小龙. 计算机科学与技术的现代化运用浅析 [J]. 信息化建设, 2016 (02): 52.

[5] 胡志颖, 崔航, 金子翔. 计算机科学与技术的现代化运用分析 [J]. 通讯世界, 2016 (02): 84.

[6] 顾亚娜. 浅谈计算机科学与技术的现代化运用 [J]. 科技创新与应用, 2015 (12): 88.

[7] 赵跃, 季伟达. 解析计算机科学与技术的现代化应用 [J]. 电子技术与软件工程, 2014 (23): 185.

(作者单位: 中国人民大学)

首先, 做到上网安全, 使用电脑上网, 最起码安装上杀毒软件和防火墙软件, 定时进行网络病毒的查杀, 营造一个好的上网环境; 软件提示的不安全网站千万不要点入; 然后不要在网吧等公共网络区域登录个人的银行账号和各种有用的保密性强的账号, 登录完成后要记得随时退出。对自己所使用的网络设备进行密码加密保护, 去正规渠道去购买网络系统和网络设备。保障好个人和社会的网络信息安全。

总之, 数据通信网络为人们的生活提供了巨大的便利, 但在运行的同时, 也存在着一些不容忽视的安全问题。在网络维护上个人要提高安全意识, 对网络系统及时进行杀毒和日常维护, 企业 and 国家要做好安全性评估, 对整个网络系统进行监测, 杜绝网络安全问题的发生。这样才能更好地保障网络信息安全, 推动国家经济与科技的快速发展。

### 参考文献

[1] 白艳英. 论数据通信网络维护及网络安全 [J]. 数字化用户, 2018, 24 (49): 5.

[2] 张国平. 关于数据通信网络维护与网络安全问题研究 [J]. 数字化用户, 2018, 24 (40): 18+119.

[3] 薛皓宝. 数据通信网络维护及网络安全问题研究 [J]. 电子元器件与信息技术, 2018 (8): 6-8+17.

[4] 山崎. 数据通信网络维护与网络安全问题思考 [J]. 数字技术与应用, 2017 (5): 215.

(作者单位: 中移铁通有限公司湖南分公司)