

Project 2: 真实感静态景物

16307130169 康子健

一. 项目介绍

本项目绘制的是太阳系中太阳和八大行星的静态景物图，因为在真实太阳系比例模型中，比例尺过大，难以在同一模型中看到所有行星，所以采用了一定的解决策略：

缩小太阳的大小，将太阳的大小缩小，因为太阳相对于行星过大，所以将太阳的大小缩小到和行星差不多的尺度。

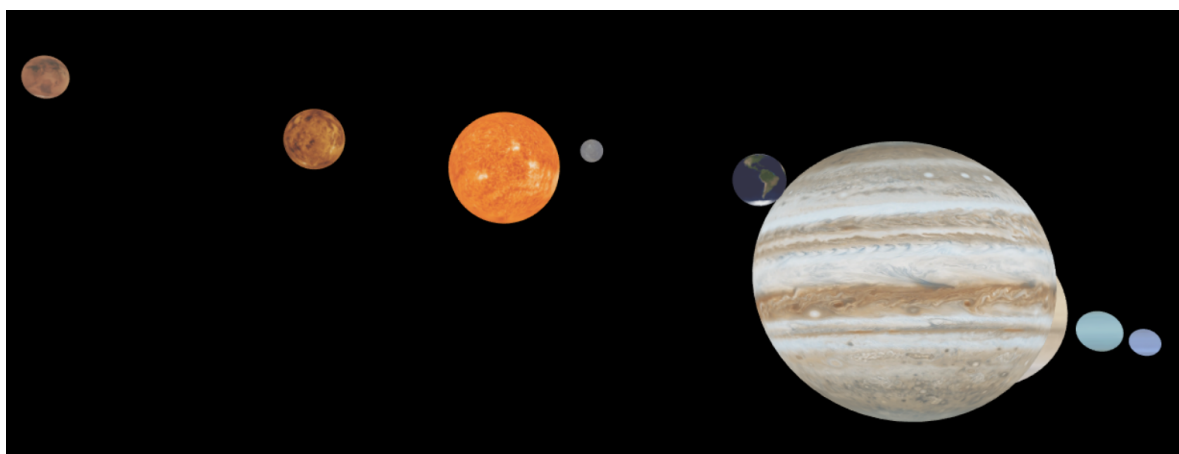
行星和太阳的距离和行星本身的大小采用不同的量纲，因为距离相对于本身的大小过大，如果想看到所有行星，那么行星本身就看不到，所以采用的方法是仅按照大小的比例显示大小，按照距离的比例显示距离。

对于行星的自转和公转，因为真实的周期太长，所以缩小到肉眼可观测的程度，并保持时间的比例真实。

二. 使用方法

因为three.js中图片加载存在跨域请求的问题，因此直接打开index.html是看不到贴图的，所以需要在源文件目录中利用 `http-server -c-1` 搭建本地服务器，访问本地服务器查看绘制出的3D模型

在页面中添加了OrbitControls控制，可以使用鼠标左键旋转、鼠标右键拖拽、鼠标中键缩放



拉到一定的角度可以正好看到太阳系中的全部行星如上图所示

三. 实现方法

本项目采用的是WebGL构建3D模型，利用到three.js，即利用javascript编写的WebGL开源框架。

绘制准备

three.js的图形绘制包括以下准备工作

- 创建渲染器

```
renderer = new THREE.WebGLRenderer({antialias: true})
```

- 创建相机

```
function initCamera() {  
  camera = new THREE.PerspectiveCamera(45, window.innerWidth /  
    window.innerHeight, 0.1, 1000);  
  ...  
  camera.lookAt(new THREE.Vector3(0, 0, 0));  
}
```

PerspectiveCamera为创建透视相机，参数分别为视角、宽高比、近平面距离、远平面距离。再加上相机的位置和相机照相的方向（lookAt）就可以在三维空间中唯一确定一个可视空间，出现在该空间内的物体最后才能够被渲染出。

- 创建场景

```
function initScene() {  
  scene = new THREE.Scene();  
}
```

搭建简单场景用来放置绘制的景物。

- 创建光源

```
ambientLight = new THREE.AmbientLight("#bbbbbb");
```

光源是一定的背景光，然后再通过星球自带一些发光特性，让星球更清晰的被观察。

模型绘制

模型绘制分成球体绘制，自转公转形成，贴图三部分完成

球体绘制采用了 `THREE.SphereGeometry(r, w, h)` 函数，并通过给几何体附加属性的方法增加了散发光。

自转公转形成采用的是 `rotation` 方法，通过在y轴上的旋转产生自转，通过设定在原点的轴并绕其旋转形成公转。

贴图采用了给几何体附加属性的方法完成，图片存放在src文件夹中。

四. 参考资料

<https://threejs.org/>