

计算机图形学 PJ3

13302010023

何天成

运行环境

- OSX El Capitan 10.11.2 (15C50)
- Google Chrome Version 46.0.2490.86 (64-bit)

开发环境

- IntelliJ Idea 15.0.2
- webpack 1.12.9

运行方式

代码在 `./src` 目录下，使用 `webpack` 打包。首先需要安装 `webpack` 并打包：

```
npm install -g webpack
npm install -g webpack-dev-server
webpack
```

然后可以用 `webpack-dev-server` 运行于 `http://localhost:8080`：

```
webpack-dev-server --inline -d
```

或者也可以用其它 `http server` 打开 `index.html`。在线打开：<https://htc.fdu13ss.org/public/PJ/index.html>

实现的部分

- 贴图物品的加载，分层贴图（mipmap）
- Obj模型的加载
- 开启 CULL_FACE
- 场景漫游
- 天空贴图
- 平行光照（附带阴影效果）
- 点光源光照（无阴影效果）
- fps 计数

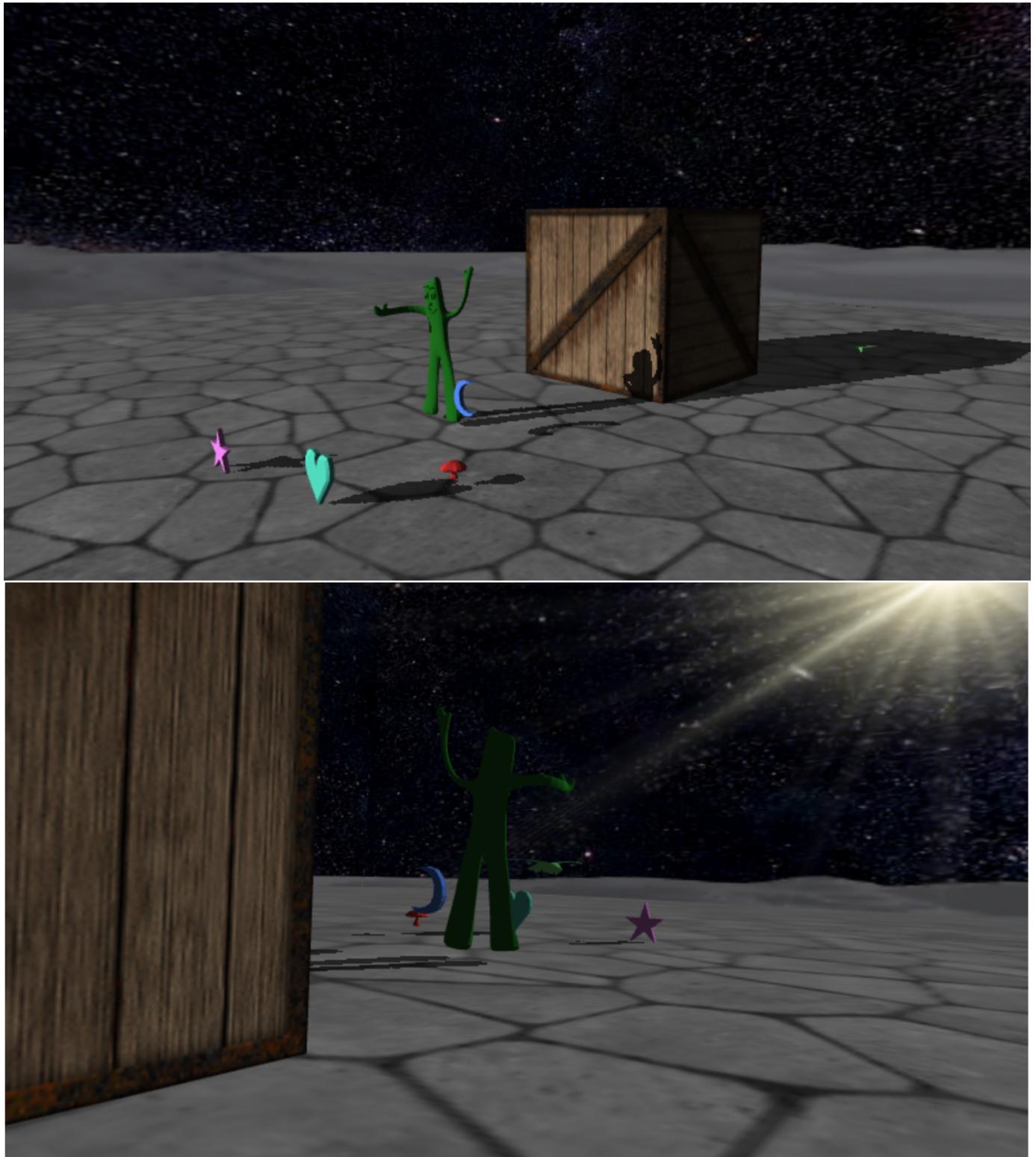
关于天空贴图的实现

见 `src/SkyboxEntity.js`。天空贴图是一个 `cubemap`，由六张图组成。将六张图加载到一个 `cubemap texture` 之后，根据相机的位置和方向进行渲染即可。渲染天空时要关闭深度测试和面切除。

关于光照和阴影的实现

见 `src/ShaderProgram.js`，统一处理了光照效果。阴影的做法是，首先从光照位置渲染一张光照深度图到一个 `Texture` 上，然后正常渲染，在 `FragmentShader` 中读取 `Texture` 并与当前点的光照深度进行比对，如果当前点的光照深度更深，则判断为在阴影中，取消该处的平行光效果。点光源光照则简单的按照距离计算强度。

效果图



由于时间原因本打算实现而尚未实现的部分

- 点光源阴影，在点光源处渲染一个 `cubemap` 的深度贴图，其余部分步骤同平行光阴影。
- 阴影过滤与反锯齿，使用 VSM 过滤及反锯齿算法使得阴影看起来更加真实。
- 水面反射折射(只能想想...)，渲染场景的反射部分贴图和折射部分贴图，然后计算水面的物理变化，并根据折射和反射的物理性质将贴图变为水面的部分。
- 计算大气折射，使得天空看起来更加真实。在视线到天空的路径上进行积分即可。

