

# 个人信息和实验环境

---

杨雅儒, 2017011071, 计85

Ubuntu 18.04, gcc 7.5.0

## 问题回答

---

### 光线投射算法逻辑

---

#### 整体逻辑

整体逻辑同下发的文档一致。

- 利用命令行参数获取输入文件和输出文件名；
- 利用 SceneParser 对输入文件进行解析，获得光源、物体、相机等信息；
- 枚举相机的每一个像素，通过相机模型计算得到 Ray 类型的投射光线；
- 枚举场景中的所有物体，找到与投射原点最近的交点；
- 若找到交点，则对该交点利用 Phong 模型计算所有光源对其的影响，并得到最终的颜色值；若没有找到交点，则设置为背景颜色；
- 最终将图片用 BMP 格式存储到输出文件中。

#### 求交点

我所使用的方法均用向量来进行运算，会相当方便。注意求交之后需要判断  $t$  值是否合法。

#### 射线与球体求交

代码中假定了射线起点在球体之外。

令  $A$  点为射线起点， $AB$  为射线单位向量， $O$  点为球心， $C$  点为  $O$  点到  $AB$  的垂足。

- 首先利用点积求出  $AC$  的长度进而得到  $AC$  向量，又已知  $AO$  向量，故可以得到  $OC$  向量；
- 分类讨论 0、1、2 个交点的情况，对于 1 个交点的情况， $C$  即为交点；对于 2 个交点的情况，令更近的交点为  $D$ ，此时由于已知  $OD$  和  $OC$  的长度，利用勾股定理即可求出  $CD$  长度；
- 又因为  $AC$  已知，故可以求出  $AD$  向量，进而得到交点  $D$  点。

#### 射线与平面求交

利用平面上任意一点+单位法向量的方式来表示平面。令射线起点为  $A$ ，射线单位向量为  $AB$ ， $A$  到平面的垂足为  $C$ ，平面上某点为  $P$ ，射线所在直线与平面交点为  $T$ 。

- 首先利用点积可以求出  $AC$  长度，进而乘上单位法向量得到  $AC$  向量；
- 此时特判一下射线与平面平行的情况；
- 对非平行的情况，利用点积求出  $AB$  与  $AC$  夹角的  $\cos$  值，又由于知道  $AC$  长度，可以得到  $AT$  的长度；
- 最后利用  $AB$  向量乘上  $AT$  长度即得到  $AT$  向量，进而得到交点  $T$  点。

#### 射线与三角形求交

- 首先与平面求交，无交点或  $t$  值不合法则直接返回 `false`；

- 令 T 为射线与平面交点，利用叉积和点积判断两次——判断 AT 是否在 AB 与 AC 之间，判断 BT 是否在 BA 与 BC 之间，若均为是，则 T 点为射线与三角形交点，否则返回 false。

## Group 类

直接使用 vector 即可。addObject 时利用 vector 的 insert 函数插入；调用 getGroupSize 时利用 vector 的 size 方法获得大小；求交时枚举其中的所有对象，调用其 intersect 函数，并取它们返回值的或值作为返回值即可。

## 透视相机模型及Phong模型

这两个内容同下发的文档上一致，不再详细描述。

唯一需要说明的是，透视相机模型中，传入的 angle 值是用于计算 fx 和 fy 值：

$$fx = \frac{imgW}{2 \tan(angle/2)}$$
$$fy = \frac{imgH}{2 \tan(angle/2)}$$

## 遇到的问题

遇到的唯一一个问题就是一个我自己实现的，用于给定射线和射线上一点 P 时求出 t 值的 `calcT()` 函数。

令射线原点为 A，一开始我使用 AP 向量的 x 值除以方向向量的 x 值得到 t，但是实际上会在方向向量 x 为 0 时出现问题。

后来改为找到方向向量上绝对值最大的一维，在这一维上作除法即可。

## 交流/讨论

并未与任何同学进行交流讨论，也没有借鉴任何网上或其它同学的代码。

## 如何编译并渲染

在 `code/` 目录下使用如下命令：

```
bash run_all.sh
```

结果将存放在 `code/output/` 目录下。

## 未解决 bug

就目前测试结果来看，并没有任何未解决的 bug。