**计算机图形学**

# 图形学作业4-2: 光线投射Ray Casting

姓名:杨杰 班级:111172 学号:20171002157

**问题描述：**

本次作业中，你要实现基本的光线投射。这是后面许多作业的基础，因此，合适的代码设计是很重要的。正如在课堂中所讲的，光线投射器从每个象素发送一条光线，与场景中的所有物体求交。你需要对球实现正交光线投射。你将使用非常基本的光照模型：物体具有常数颜色。你还要实现显示每个象素到像机的距离t。

需要使用面向对象的设计，来使光线投射易扩展。一个抽象的Object3D 类作为所有3D图元的父类。由此导出子类如Sphere, 并在其中实现其特殊属性。在后面的作业中，将扩展更多的基本图元如平面planes和多边形polygons。类似的，此作业需要实现一个通用的Camera类，由此导出OrthographicCamera子类。下次作业中，将导出一般的透视投影相机类perspective camera。

我们提供了光线类（Ray class）和交点类（Hit class） 来管理由照相产生的光线和其交点, 以及材质类（Material class）的框架。

**问题解决：**

1. **编写一个虚类Object3D class**

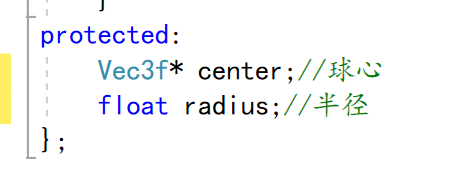
按照文档提示，编写类：

图片包含 屏幕截图

描述已自动生成

1. **由Object3D导出子类Sphere**

按照文档提示，编写类：

需要的属性：

构造函数:

图片包含 屏幕截图

描述已自动生成

在求交intersect函数中，需要查找沿光线Ray最近的交点，距离由参数t给定。tmin 用于限制交点的范围。如果求得的交点满足 t > tmin 且 t 小于当前交点相应t值（该值存储在Hit中），需要更新Hit。如果新的交点比前面求的交点更近，t（距离）和Material（材质）都要更新 。需要注意在求交函数（intersection）中检验 t >= tmin. tmin 依赖于相机camera的类型，求交函数不能修改它。

图片包含 屏幕截图

描述已自动生成

1. **由Object3D导出子类Group**

按照文档提示，编写类:

包括的参数有：

图片包含 屏幕截图

描述已自动生成

需要编写Group求交intersect算法，该求交算法对所有的实例，循环调用其求交算法。Group 构造函数以此group所函数对象个数为输入参数。group 应包含增加对象的方法：void addObject(int index, Object3D \*obj):

图片包含 屏幕截图

描述已自动生成

1. **编写纯虚的Camera 类和OrthographicCamera子类**

Camera 类有两个纯虚方法:

virtual Ray generateRay(Vec2f point) = 0;

virtual float getTMin() const = 0;

图片包含 屏幕截图

描述已自动生成

前者用于对每个屏幕座标产生一条光线ray, 屏幕坐标用Vec2f描述。由orthographic相机产生的光线的方向direction为常数, 但原点origin是变化的.getTMin()方法会在光线穿过屏幕时使用。 对orthographic相机, 光线总是始于无限远处，因此tmin 应为一个很大的负数。当然，在下一个作业中，需要实现透视perspective相机，而tmin的值应为零，以便裁剪视点后面的物体，而不显示。

正交orthographic相机由一个正交基orthonormal basis (一个点，三个向量)和图像大小(一个浮点数)给定。构造函数的输入包括图像中心, 方向向量, 向上向量（up vector）, 和图像大小。输入的方向direction可能不是单位向量，此时需要单位化（normalized）。输入的向上up向量可能不是单位向量，或者不与方向direction向量垂直。必须进行修改，以实现与方向向量direction正交。第三个基向量，图像平面的水平向量，由方向direction和向上up向量导出(提示: 回忆向量代数和叉积cross)。由相机的屏幕座标产生的光线原点，变化范围为(0,0) -> (1,1), 其规律为：

center - (size\*up)/2 - (size\*horizontal)/2 -> center + (size\*up)/2 + (size\*horizontal)/2

相机不知道屏幕分辩率。图像分辩率应该在主程序main的循环中处理。对于非正方形图像比例, 剪切掉相应的屏幕座标。

图片包含 屏幕截图

描述已自动生成

1. **使用输入文件分析（parsing）代码来装入相机camera, 背景色和场景对象**
2. **编写主程序main，其功能包括读入场景(使用所提供的分析parsing代码)** **对图像平面的每个象素循环, 使用你的OrthographicCamera 类产生光线, 与高层的Group类求交，在该类中存储了场景的对象, 并将最近交点的颜色写到图像文件中。**
3. **实现第二种渲染模式，来可视化显示场景中物体的深度*t*。两个输入的深度值指定深度值的范围，该范围内的值对应到灰度值来进行可视化。深度范围外的值可简单地剪掉（clamped）。**

对于每一个输入数据，生成两张图片，仅包含轮廓信息的图片，开始就设置其背景为输入的 background，对于深度图则将背景设为黑色。根据输入的分辨率，我们将图片坐标（从(0,0)到(width,high)）转化为横纵坐标从‐1到1的输入坐标，这个转化做简单的平移伸缩变换。

如果有交点，对于轮廓图我们只需要在相应的图片坐标的位置填充上输入的材质的颜色 即可；而对于深度图，我们根据输入的深度值的范围，以及交点对应的 t 值，进行下列 比值的运算。然后将对应图片坐标位置的RGB值设置为（α,α,α）。

**测试截图：**

-input scene1\_01.txt -size 200 200 -output output1\_01.tga -depth 9 10 depth1\_01.tga

图片包含 台球

描述已自动生成

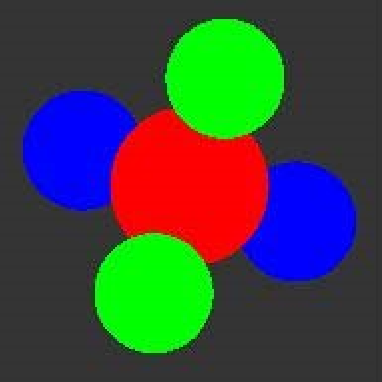
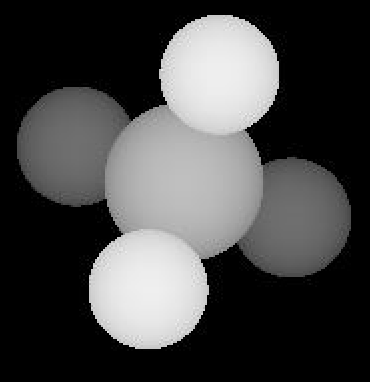
-input scene1\_02.txt -size 200 200 -output output1\_02.tga -depth 8 12 depth1\_02.tga

图片包含 矢量图形

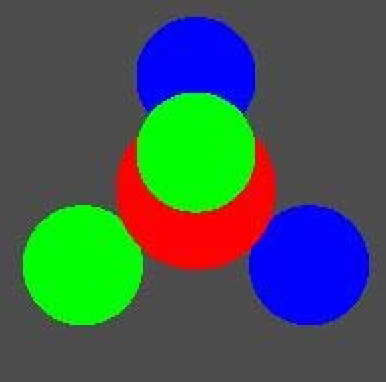
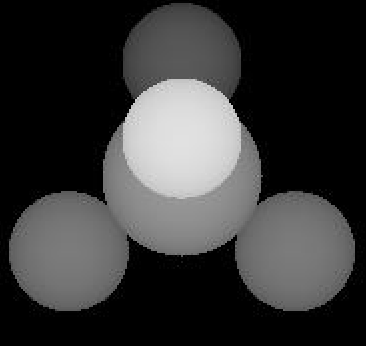
描述已自动生成图片包含 室内, 就坐, 蛋糕, 顶部

描述已自动生成

-input scene1\_03.txt -size 200 200 -output output1\_03.tga -depth 8 12 depth1\_03.tga

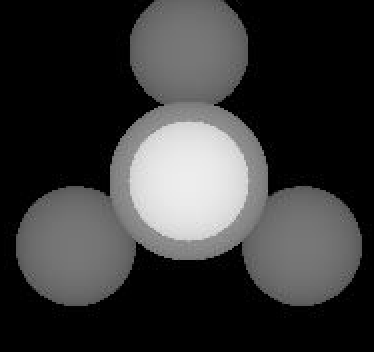
 

-input scene1\_04.txt -size 200 200 -output output1\_04.tga -depth 12 17 depth1\_04.tga

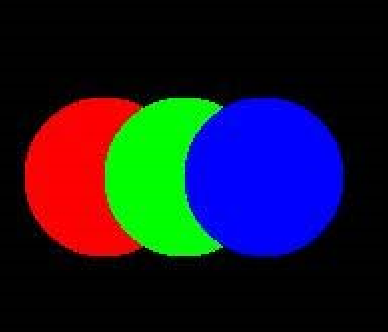
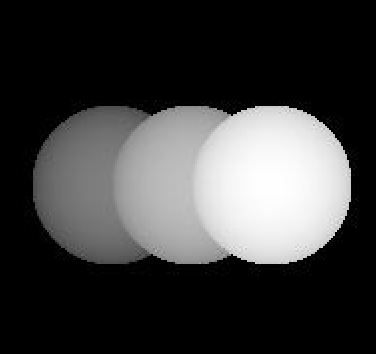
 

-input scene1\_05.txt -size 200 200 -output output1\_05.tga -depth 14.5 19.5 depth1\_05.tga

图片包含 矢量图形

描述已自动生成 

-input scene1\_06.txt -size 200 200 -output output1\_06.tga -depth 3 7 depth1\_06.tga

-input scene1\_07.txt -size 200 200 -output output1\_07.tga -depth -2 2 depth1\_07.tga

