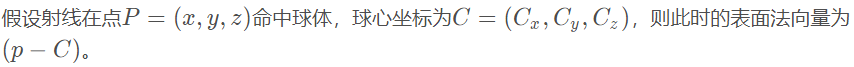
README

作业4 3-3

变换Transformations & 更多基本图元Primitives

修改Sphere类

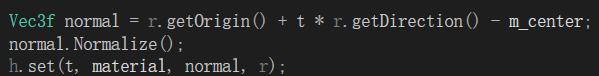
由于本题目提供的hit类与上一次作业中初始化接受的参数不同，新增加了法向量。所以在球类进行求交算法完成之后，还要求出球面上法向量，求解球类法向量的公式为：



p表示的是与球面交点的射线，C球心坐标。

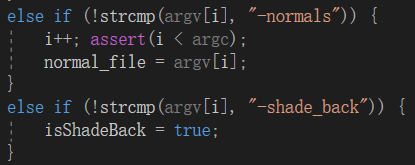
射线的公式就是将求出的t带入光线方程之中得出结果。

则增加的代码为：



增加命令行命令项

题目测试项增加了两项normals（法向量），shade\_back（法向量反转）。增加代码部分为：

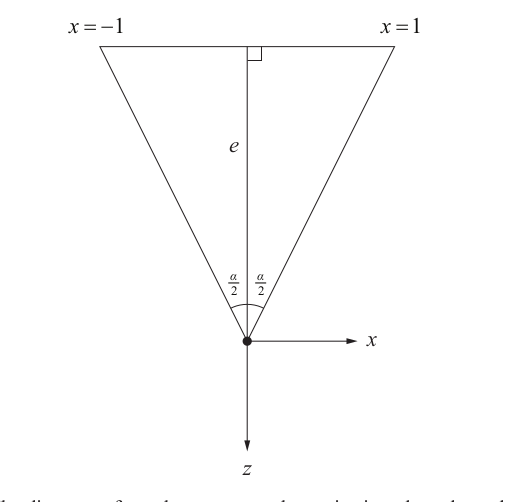


增加类PerspectiveCamera

和上一次作业中的正交相机相同，都是继承于camera类，并且实现对于任意位置产生的实现向量算法。

和OrthographicCamera类似，传进来的坐标已经是从屏幕坐标转换到了坐标从(0,0)到(1,1)的矩形空间内。然而和平行投影不同的是，透视投影需要不同的参数来决定，透视相机还多了角度和焦距，还有长宽比的成员变量。

透视相机的二维示意图如下图所示：



如上图的集合图可得知焦距公式为：d=(screenHeight\2)\tan(alpha\2)

而屏幕的长宽高是由分辨率决定的，但是不能预先确定分辨率，所以预先判断分辨率为1：1。

如果当宽大于高时，选择高为2，而宽为2倍的分辨率，反之相同。

我们知道焦距是视点到显示区域平面的距离，所以显示平面上的点相当于是以视点为原点在mDirection向量方向上的坐标，现在还差mUp和mHorizontal的方向上的坐标就可以得到显示区域上点的完整坐标。

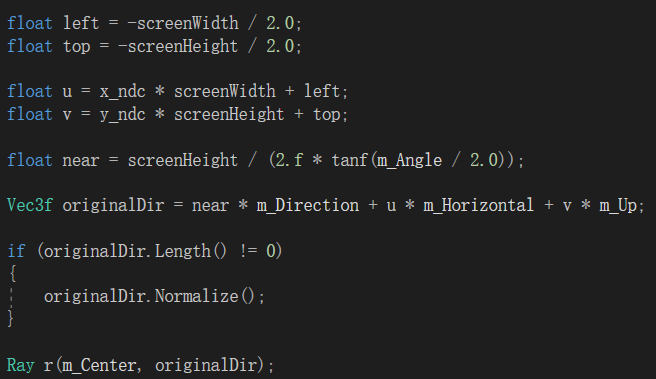
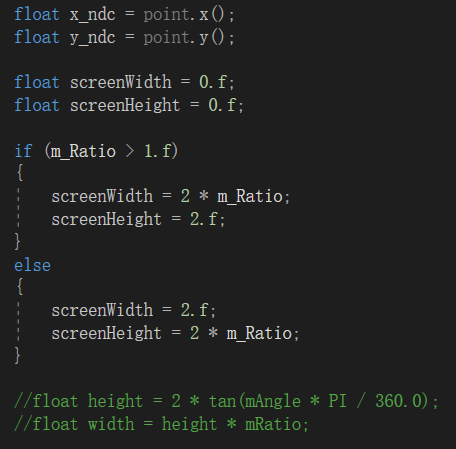
和平行投影一样，屏幕坐标映射到了(0,0)到(1,1)的矩形后，还要映射至(-screenWidth/2, -screenHeight/2)，(screenWidth/2, screenHeight/2)的矩形，最后得到在mHorizontal和mUp上的坐标。

矩形中任意一点的坐标为：



法向量的求法和上次作业的求发相同，两个相互垂直的向量来决定另一条相互垂直的正交向量。

上述全部代码表示为：

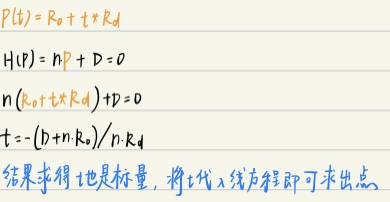


添加Plane类

计算与平面的交点需要的变量有法向量和平面点的偏移量。在平面内的任意点都必须要满足式子 n\*p+D = 0 （n为法向量 D为平面具体 为交点）。

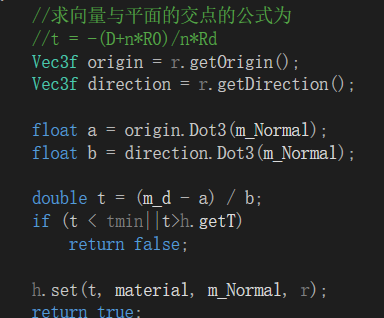
求解线面交点，就是同时满足线与面的两个方程，光线显示方程带入平面隐式方程，求解t。

推导过程为：



求出最近点的方法即t的值必须大于t最小值，小于t的最大值。

代码表示为：

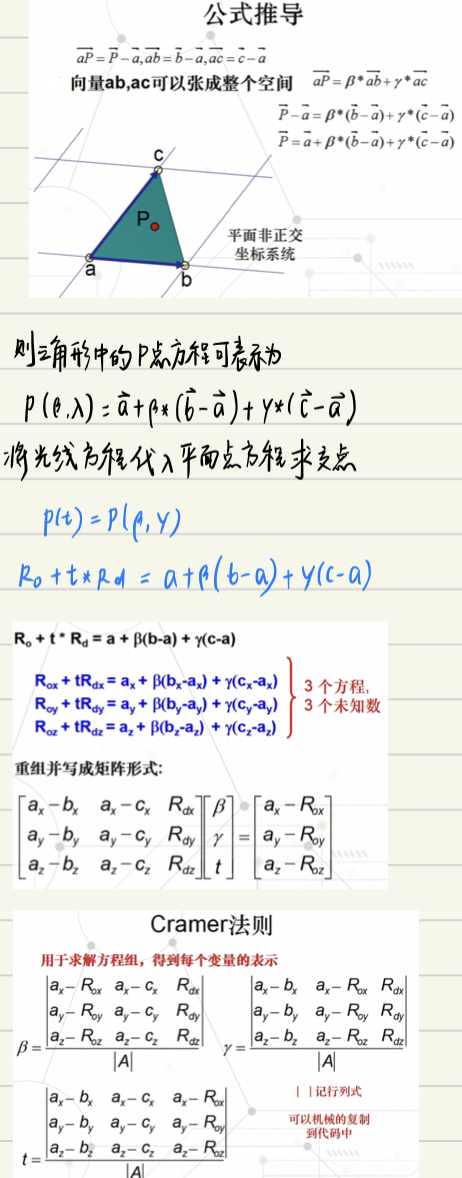


添加Triangle类

Triangle传入的参数有三个点和材质。则类Triangle成员变量有三个点和法向量，法向量可以由三个点任意组成的两个向量叉乘法向量决定。

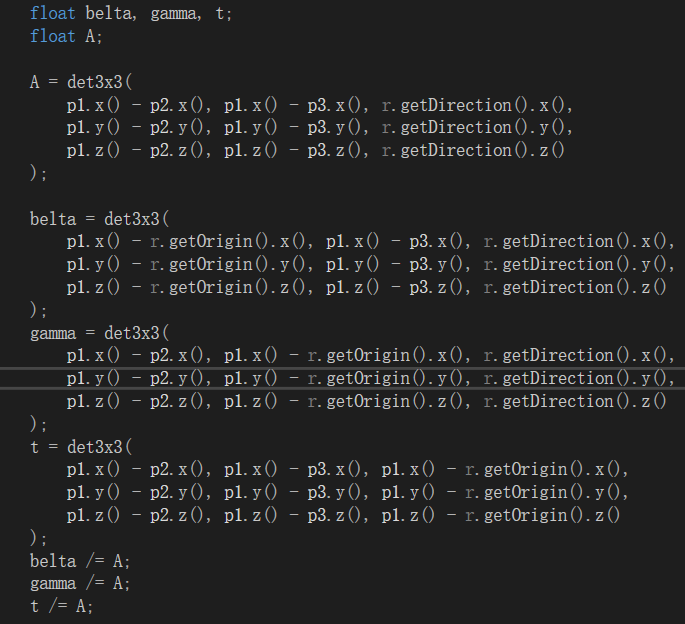
进行交点的计算，利用了三角形三个点插值的方法。

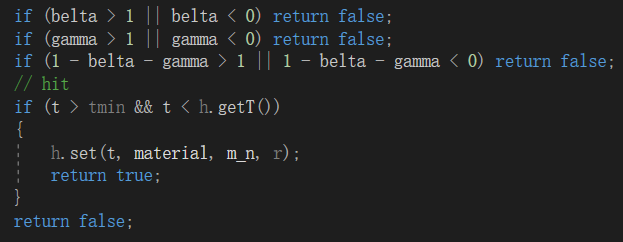
推导过程为：



计算最近点的方法和平面求最近点相同。

代码表示为：



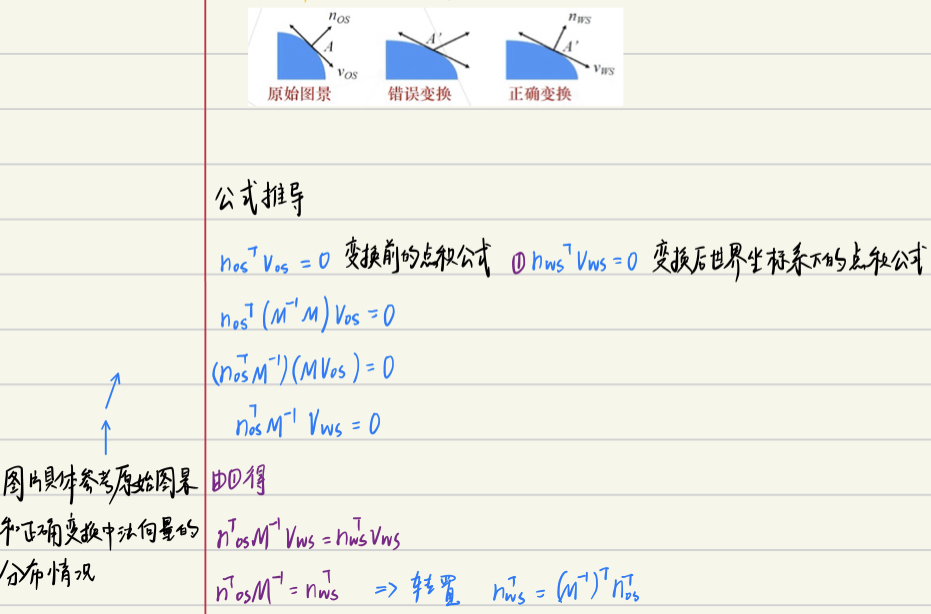


添加Transform类

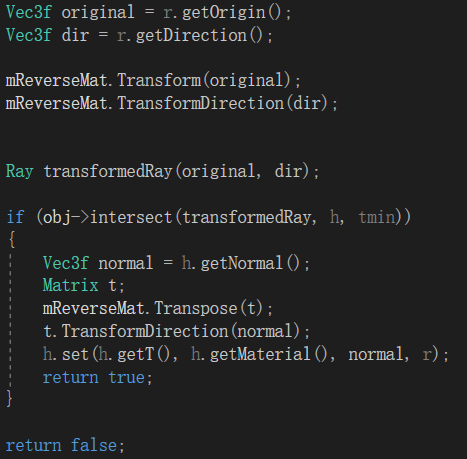
传入的参数有变换的矩阵，还有物体对象。

因为在变换之后，法向量不能用相同的方法利用相同的变换矩阵进行变换，要利用其他的方法进行计算。

公式推导过程：



代码表示为：



计算像素颜色的值

计算颜色的之后，需要考虑多个光源进行叠加，包括周围自然光和其它光源的影响，所以在计算物体颜色的时候需要考虑到物体本身材质、周围自然光、其它光源和球体法向量。

相应的计算公式为：



计算深度像素值

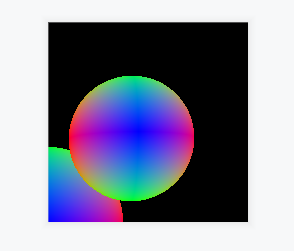
深度值的计算公式为 depthMax-t/depthMax-depthMin。

计算法向量像素值

写入法向量的向量取绝对值之后的值即可。

实习中遇到的困难和解决方案

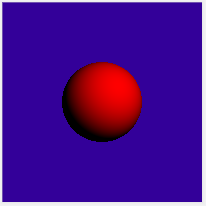
在进行图像显示的时，屏幕坐标映射到了(0,0)到(1,1)的矩形时出现比例的错误，导致只显示了一部分的物体。



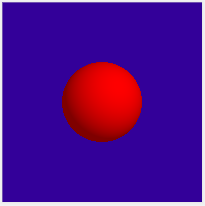
利用了几何关系，调整比例。

实验结果展示

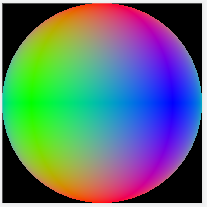
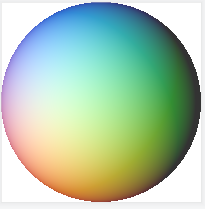
Assignment2 -input scene2\_01.txt -size 200 200 -output output2\_01.tga



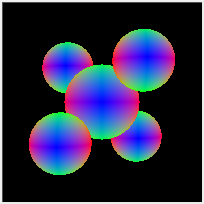
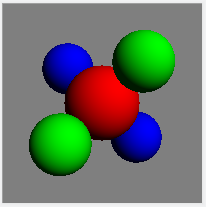
Assignment2 -input scene2\_02.txt -size 200 200 -output output2\_02.tga



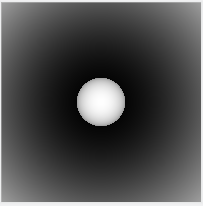
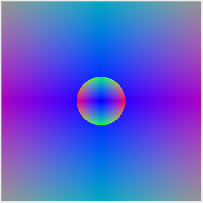
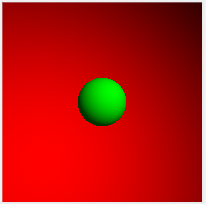
Assignment2 -input scene2\_03.txt -size 200 200 -output output2\_03.tga -normals normals2\_03.tga



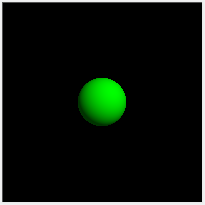
Assignment2 -input scene2\_04.txt -size 200 200 -output output2\_04.tga -normals normals2\_04.tga



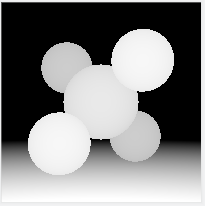
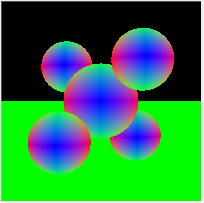
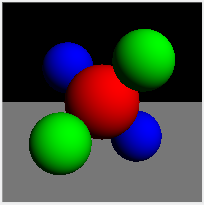
Assignment2 -input scene2\_05.txt -size 200 200 -output output2\_05.tga -depth 9 11 depth2\_05.tga -normals normals2\_05.tga -shade\_back



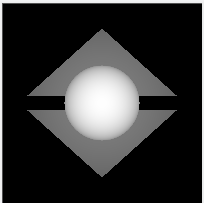
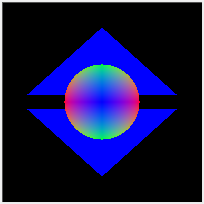
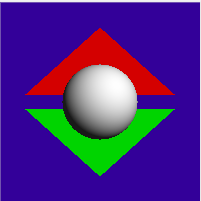
Assignment2 -input scene2\_05.txt -size 200 200 -output output2\_05\_no\_back.tga



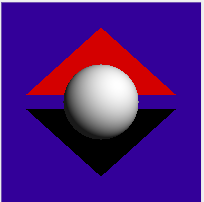
Assignment2 -input scene2\_06.txt -size 200 200 -output output2\_06.tga -depth 8 20 depth2\_06.tga -normals normals2\_06.tga



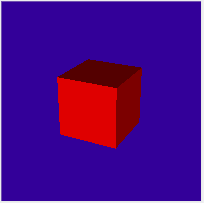
Assignment2 -input scene2\_07.txt -size 200 200 -output output2\_07.tga -depth 9 11 depth2\_07.tga -normals normals2\_07.tga -shade\_back



Assignment2 -input scene2\_07.txt -size 200 200 -output output2\_07\_no\_back.tga



Assignment2 -input scene2\_08.txt -size 200 200 -output output2\_08.tga



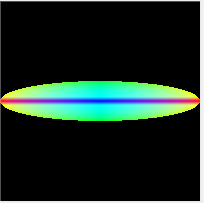
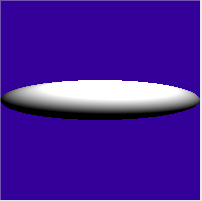
Assignment2 -input scene2\_09.txt -size 200 200 -output output2\_09.tga



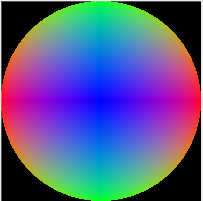
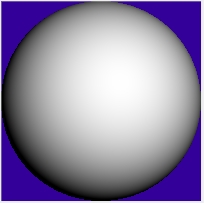
Assignment2 -input scene2\_10.txt -size 200 200 -output output2\_10.tga



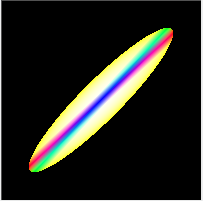
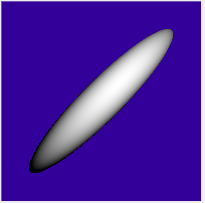
Assignment2 -input scene2\_11.txt -size 200 200 -output output2\_11.tga -normals normals2\_11.tga



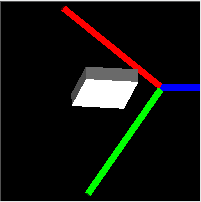
Assignment2 -input scene2\_12.txt -size 200 200 -output output2\_12.tga -normals normals2\_12.tga



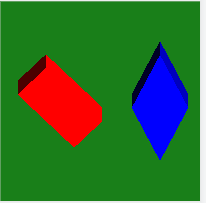
Assignment2 -input scene2\_13.txt -size 200 200 -output output2\_13.tga -normals normals2\_13.tga



Assignment2 -input scene2\_14.txt -size 200 200 -output output2\_14.tga



Assignment2 -input scene2\_15.txt -size 200 200 -output output2\_15.tga



Assignment2 -input scene2\_16.txt -size 200 200 -output output2\_16.tga -depth 2 7 depth2\_16.tga

