作业报告

F1403701 5137319014 张雨婷

1. **调用OpenGL库函数绘制一个球面**

使用了glutSolidSphere函数绘制完成。由于是第一次接触opengl，我花了挺多时间安装各种库并理解画图的流程，做完这个作业后，感觉对opengl熟练了很多。在这个作业中，为了支持滚轮操作，我下载了不同版本的glut库。

1. **可改变视角从各个方向观察该模型**

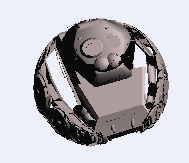
这个使用了myMouse和myMotion函数，看了网上的教程后顺便尝试了一下myKeyboard函数，w和s代表远和近。a和d代表左右。鼠标左键拖动可旋转。鼠标滚轮可放大缩小。

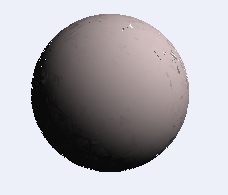
1. **写一个Loader读入三维模型（.max、.3ds或.obj格式）并绘制**

一开始我想偷懒使用库函数。下载了一堆库，比如glfw、glew等。配完环境发现自己也不是很会用，网上教程里的东西都跑不起来，反而是浪费了很多时间来配环境。nehe教程也看不太懂，主要因为代码不完整导致跑不起来。最后开始自己解析obj格式。v代表顶点，vt是二维纹理，vn是法向量，f是面。读入这些数据然后不停的画三角形或者四边形就可以了。关于具体的纹理绘制，由于我只读入了.obj，对于模型的贴图.jpg和.bmp文件等没有读入，所以并没有载入obj中的纹理和材质。

1. **建立球面顶点与导入模型顶点之间的映射，实现球面向任意模型的变形过程动画**

把模型上的点按照法向量投射到单位球体上，设置步数，一步步dx,dy,dz增加到原来的位置。我尝试了两种方式，根据顶点所在的位置的法向量来映射。这样不规则的obj会导致映射出来的圆不太圆。镂空后效果真的就跟投射在圆上一样很奇怪。没有obj点的地方就是空的。效果如图：

【图一】

空的地方就是这个图这个向量上本身没有obj点的顶点。这样的圆是有明显缺陷的。然后我采用了按照反光normal的方向来确定法向量。效果如图：  
【图二】

虽然看起来有点圆，但是这个圆表面还是有缺陷，并且圆内部的线条很复杂。并不是一个真正的圆。

本来还想尝试寻找单位圆上距离最近的点来映射。只有最近的相邻的映射点才有可能建立属于圆的face数组。但是就算找到了，obj加载的面的存在使得这个圆也会有缺陷，如图一，物体是分开的时候，圆会断开，除非动画变换到一半突然把代表面关系的faces数组从圆替换成obj，可是这样也不算连贯的动画变换。同时， 寻找最近点最大的挑战就是算法效率。一个点可能先找到最近的点，然后把这个点从圆中剔除，然而事实上存在另一个点更适合映射这个被剔除的点。如果要反复计算计较的话太消耗计算资源，而我又没有优秀的算法。

最终没有优秀算法的我选择了有缺陷的图二算法，至少比图一正常一些。最后为了方便批作业，我把所有题放在一个.cpp和.h中，开始会有几秒第一题的球展示，然后进入第四题的变换，除了变换途中没法很正常的使用鼠标键盘滚轮操作，其他时候都能用，这是第二题的内容，变换完就是三题的内容，即可以载入obj。感觉第一次作业对我来说有点非常地难TATTTT。