**上海电力学院**

**虚拟现实技术**

**课程设计报告**

****

**题目 人脸的3D渲染**

**姓名 杨佐栋**

**专业 电子信息工程**

**班级 2015071**

**学号 20150761**

**一、三维坐标系统**

　　如果你熟悉二维坐标系，如Windows桌面应用或者IOS手机应用采用的坐标系，你一定知道x、y轴。二维坐标可以表示子窗体或者UI控件摆放的位置，当调用绘图API是可以定义画笔和画刷的绘制点。与二维坐标类似，三维坐标系统只是多了一个z轴，这个方向用来描述深度信息（一个物体距离屏幕的远近），如果你已经了解二维坐标系的概念，那么转换到三维坐标系就很简单了。需要注意的是OpengGL通常也是采用的右手坐标系。

**二、网格、多边形、顶点**

　　绘制3D图形有许多方法，用的最多的是用网格绘制。一个网格由一个或多个多边形组成，这些多边形的顶点都是三维空间中的点，它们具有x、y、z三个坐标值。网格中通常采用三角形和四边形，这些基本面片可以围成网格，从而形成了模型。

**三、材质、贴图、光照**

　　除了x、y、z坐标以外，网格的表面采用另外的属性表示。表面属性可以非常简单地采用单色，也可以采用复杂的方法，比如它的反光效果怎么样或者它看起来是否有光泽。网格表面还可以采用一个或多个位图，一个我们叫贴图，多个我们叫图集。贴图可以是文字效果（例如T恤上面的图案），也可以是复杂的粗糙效果或彩虹效果。大多数的图形系统会将网格的表面属性统一用材质来表示，而材质最终表现出来的效果会受环境中的光照影响。

1. **部分实验代码：**

|  |
| --- |
| int main(int argc, char\* argv[]) |
|  | { |
|  | // GLUT初始化 |
|  | glutInit(&argc, argv); |
|  | glutInitDisplayMode(GLUT\_DOUBLE | GLUT\_RGBA); |
|  | glutInitWindowPosition(20, 20); |
|  | glutInitWindowSize(WindowWidth, WindowHeight); |
|  | glutCreateWindow(WindowTitle); |
|  | glEnable(GL\_DEPTH\_TEST); |
|  |  |
|  | glEnable(GL\_TEXTURE\_2D); // 启用纹理 |
|  | //texGround = load\_texture("2.bmp"); //加载纹理 |
|  | //texWall = load\_texture("lby.bmp"); |
|  | texWall = load\_texture("ARface.bmp"); |
|  | glutDisplayFunc(&display); //注册函数 |
|  | glutReshapeFunc(reshape); |
|  | glutMouseFunc(mouse); |
|  | glutKeyboardFunc(keyboard); |
|  | //glutIdleFunc(&myIdle); |
|  | glutMainLoop(); //循环调用 |
|  | return 0; |
|  | } |

**五、人脸渲染结果：**



1. **小结**

通过这次的虚拟现实课程设计，我基本上掌握了使用Linux系统进行人脸的3D渲染的原理，并成功得出一个较为正确的结果。