

真实感静态景物报告

沈钊蕾 16307130285

在本项目中，我选择的工具是 Python3+OpenGL。

为了更好地理解 OpenGL 提供的各个函数，并设置合适的参数，我仔细阅读了 OpenGL 教程，并根据其中的指示进行了绘制。

绘制真实感静态景物，需要考虑的因素有：背景色彩、物体的大小、位置、色彩、色彩过渡、光源的类型、位置，以及窗口大小、物体在窗口大小变换时如何保持比例不变等因素。为此，我把每一个函数的参数设置以及考虑因素都写在了注释中。

代码与具体解释如下：

```
from OpenGL.GL import *
from OpenGL.GLU import *
from OpenGL.GLUT import *

def display():
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT)
    #清除当前帧的颜色、把所有像素的深度值设置为最大值
    glutSolidSphere(1.2, 50, 50)
    #绘制一个球体。
    #第一个参数表示半径，且为 double 型数据。这里设为 1.2
    #第二、第三个参数表示经线、纬线数量。数值越大，越精确。
    glFlush()
    #清空缓冲区

def reshape (w, h):
    glViewport(0, 0, w, h);
    #定义视口大小
    #函数原型为 glViewport(GLint x,GLint y,GLsizei width,GLsizei height)
    #x、y 以像素为单位，指定了视口左下角在窗口里的坐标位置。
    #width,height 表示视口矩形的宽度和高度，根据窗口的实时变化重绘窗口。
    glMatrixMode(GL_PROJECTION);
    #将当前矩阵指定为投影矩阵
    glLoadIdentity();
```

```

        #重置当前指定的矩阵为单位矩阵
    if w <= h:
        glOrtho (-2, 2, -2 * h / w, 2 * h / w, -10.0, 10.0 )
        #创建一个正交平行的视景体
        #glOrtho(left, right, bottom, top, near, far)
        #此函数的参数设置为使用一个和窗体一样比例的视景体来截取图像
    else:
        glOrtho (-2 * w / h, 2 * w / h, -2, 2, -10.0, 10.0)
    glMatrixMode(GL_MODELVIEW)
    #表示模型视图矩阵
    glLoadIdentity()

glutInit()
glutInitDisplayMode(GLUT_RGBA | GLUT_SINGLE)
    #设置初始显示模式
    #这里的参数分别对应了指定 RGBA 颜色模式的窗口、指定单缓存窗口
glutInitWindowSize(300, 300)
    #设置窗口大小
glutCreateWindow("Shere")
    #给窗口命名
glClearColor(0.95, 1.0, 1.0, 1.0)
    #glClearColor 用于指定窗口的背景颜色。
glShadeModel(GL_SMOOTH)
    #glShadeModel 函数用于控制颜色的过渡模式
    #参数一般为 GL_SMOOTH 与 GL_FLAT
    #opengl 在指定的两点之间进行插值， 绘制其他点。
    #当两点颜色不同时， GL_SMOOTH 将出现过渡效果
    #而 GL_FLAT 只是以指定的某一点的单一色绘制其他所有点。
    #由于本项目需要绘制静态实感的实物
    #若使用 GL_FLAT 则会出现颜色断层的和谐情况
    #因此采用 GL_SMOOTH 参数。
Diffuse = [0.5, 0.8, 0.9, 1.0]
glMaterialfv(GL_FRONT_AND_BACK, GL_AMBIENT_AND_DIFFUSE, Diffuse)
    #glMaterialfv 函数可以指定材质对漫射光的反射率。
    #第一个参数一般可以取 GL_FRONT、GL_BACK、GL_FRONT_AND_BACK 等，
    #分别表示材质属性运用于对象的正面、反面、或是正反两面。
    #我选取的是 GL_FRONT_AND_BACK
    #第二个参数表示对何种光进行设置。
    #我选用的参数为 GL_AMBIENT_AND_DIFFUSE，
    #表示对环境光和漫射光反射率进行设置。
    #第三个参数是一个四维数组， 描述了反光率的 RGBA 值，
    #每一项取值都为 0-1 之间。这里我选取了一个我相对喜欢的颜色。
Position = [1.0, 1.0, 1.0, 0.0]
glLightfv(GL_LIGHT0, GL_POSITION, Position)

```

```
#glLightfv 函数用于创建指定光源
#第一个参数表示选择 0 号光源
#第二个参数用于指定光源的属性，这里需要指定的属性是位置
#第三个参数表示第二个参数所对应属性的具体的值
glEnable(GL_LIGHTING)
    #光源在默认情况下是关闭状态。由于在后续的渲染中需要，因此设置开启。
glEnable(GL_LIGHT0)
    #为了使用第 0 号光源，需要指定 GL_LIGHT0。
glEnable(GL_DEPTH_TEST)
    #启用深度测试，只绘制最靠前的一层。
glutDisplayFunc(display)
    #绘制窗口
glutReshapeFunc(reshape)
    #自适应屏幕窗口大小的改变
glutMainLoop()
```

生成图像如 demo.png 所示。

参考网站：

https://blog.csdn.net/ivan_ljf/article/details/8726672

<https://learnopengl-cn.github.io/>