

# 浙江大学实验报告

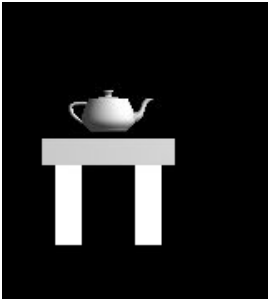
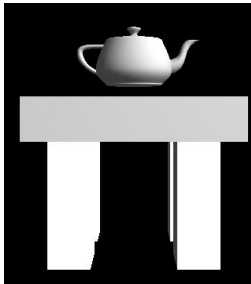
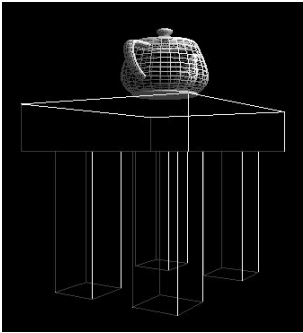
课程名称： 计算机图形学 指导老师： 成绩：  
实验名称： OpenGL 三维观察 实验类型： 基础实验 同组学生姓名：

## 一、实验目的和要求

在模型变换实验的基础上，通过实现下述实验内容，掌握 OpenGL 中三维观察、透视投影、正交投影的参数设置，并能使用键盘移动观察相机，在透视投影和正交投影间切换，验证课程中三维观察的内容；进一步加深对 OpenGL 三维坐标和矩阵变换的理解和应用。

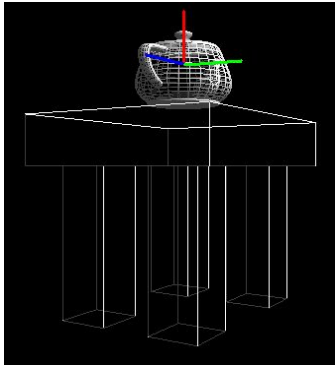
## 二、实验内容和原理

使用 Visual Studio C++编译已有项目工程，并修改代码生成以下图形：

 正投影	 透视投影
	使用键盘改变 camera 位置与观察方向 (按键为 asdwzc，也可以自行设定)

添加键盘对茶壶的控制，主要是茶壶沿着桌面的平移操作（如下图中绿色和蓝色标示）和茶壶绕自身轴（如下图中红色标示）的旋转操作；按键为：l, j, I, k, e。

装  
订  
线



### 三、主要仪器设备

Visual Studio C++

glut-3.7.6-bin.zip

glutEx1-vs2005 工程

### 四、操作方法和实验步骤

#### 1) 设置正投影

```
glOrtho(-3, 3, -3, 3, -100, 100);
```

#### 2) 设置透视投影

```
if (bPersp)
{
    gluPerspective(45.0f, whRatio, 0.1f, 100.0f);
}

.....
switch(k)
{
    case 'p': {bPersp = !bPersp; updateView(wHeight, wWidth);break; }
    .....
}
```

#### 3) 使用键盘移动观察相机

```
float eye[] = {0, 0, 8}; // 观察点位置
float center[] = {0, 0, 0}; // 观察中心位置

.....
switch(k)
{
    .....
    // a, d, w, s分别对应左右上下
    case 'a':
```

```

        //todo, hint: eye[] and center[] are the keys to solve the problems
        eye[0] -= 0.2f;
        center[0] -= 0.2f;
        break;
    }
case 'd':
    //todo
    eye[0] += 0.2f;
    center[0] += 0.2f;
    break;
}
case 'w':
    //todo
    eye[1] -= 0.2f;
    center[1] -= 0.2f;
    break;
}
case 's':
    //todo
    eye[1] += 0.2f;
    center[1] += 0.2f;
    break;
}
case 'z':
    //todo
    eye[2] -= 0.2f;
    center[2] -= 0.2f;
    break;
}
case 'c':
    //todo
    eye[2] += 0.2f;
    center[2] += 0.2f;
    break;
}
.....
}

.....
gluLookAt(eye[0], eye[1], eye[2],
          center[0], center[1], center[2],
          0, 1, 0);    // 设置观察相机，向上方向为(0, 1, 0)

```

4) 添加键盘对茶壶的控制，主要是茶壶沿着桌面的平移操作和茶壶绕自身轴的旋转操作

茶壶沿桌面的平移操作和绕自身轴的旋转可以通过每次重绘图形时重新设置茶壶相对桌面的位置来实现，绕自身为此增加以下全局变量：

```
// 对茶壶进行移动的参数
bool bTeaPotAnim = false; // 是否开启动画
float fX = 0.0, fZ = 0.0, fAngle = 0.0; // 茶壶相对桌面在 X 轴和 Z 轴的平移量及相对自身轴的旋转角度
```

接着在茶壶的绘图代码前添加平移的代码：

```
glRotatef(fAngle, 0.0f, 1.0f, 0.0f); // 相对自身轴旋转
glTranslatef(fX, 0.0f, fZ); // 相对桌面平移
```

添加相应的键盘响应代码以改变平移量和响应绕自身轴旋转：

```
switch(k)
{
    .....
//茶壶相关操作
    case 'l':
        {
            //todo, hint:use the ARRAY that you defined, and notice the teapot can NOT be moved
            out the range of the table.
            if (fX >= 2.5) fX = 2.5;
            else fX += 0.25;
            break;
        }
    case 'j':
        {
            //todo
            if (fX <= -2.5) fX = -2.5;
            else fX -= 0.25;
            break;
        }
    case 'i':
        {
            //todo
            if (fZ >= 2.0) fZ = 2.0;
            else fZ += 0.2;
            break;
        }
    case 'k':
        {
            //todo
            if (fZ <= -2.0) fZ = -2.0;
            else fZ -= 0.2;
            break;
        }
    case 'e':
        {
            //todo

```

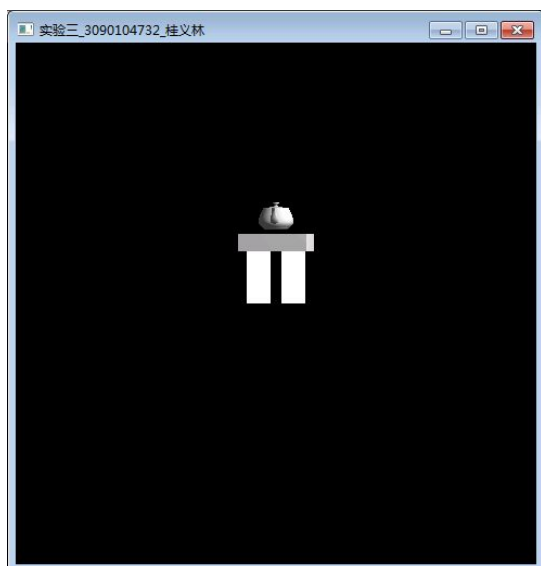
```
        bTeaPotAnim = !bTeaPotAnim;  
        break;  
    }  
    .....  
}
```

## 五、实验数据记录和处理

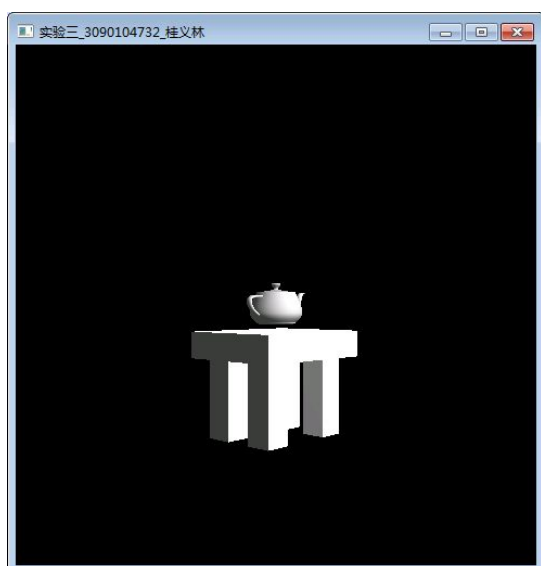
无

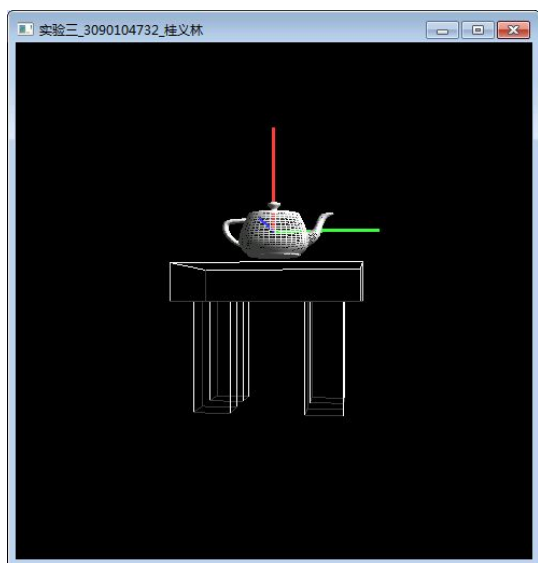
## 六、实验结果与分析

正投影：

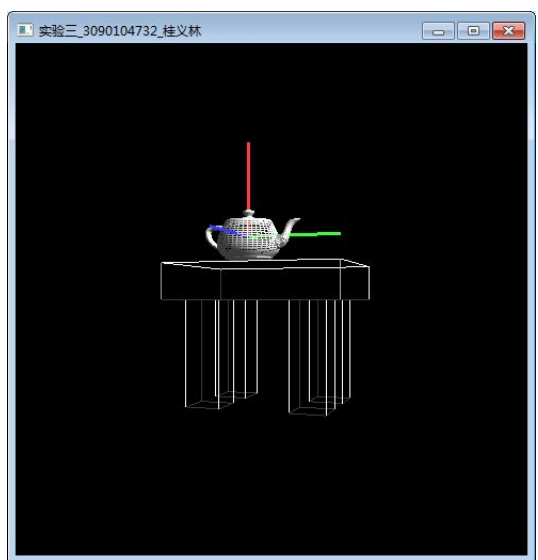
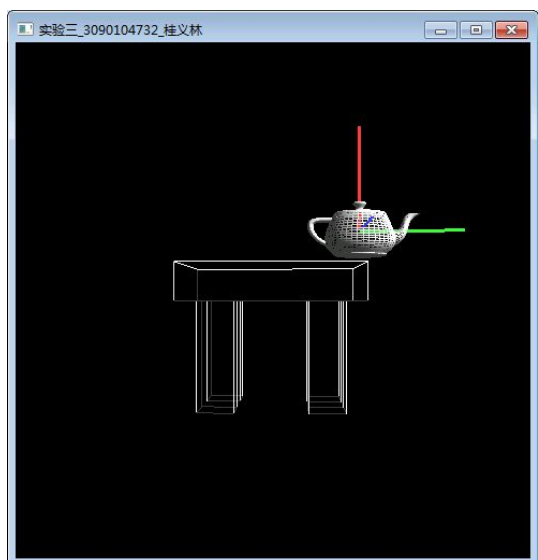


透视投影：（按 p 键切换）

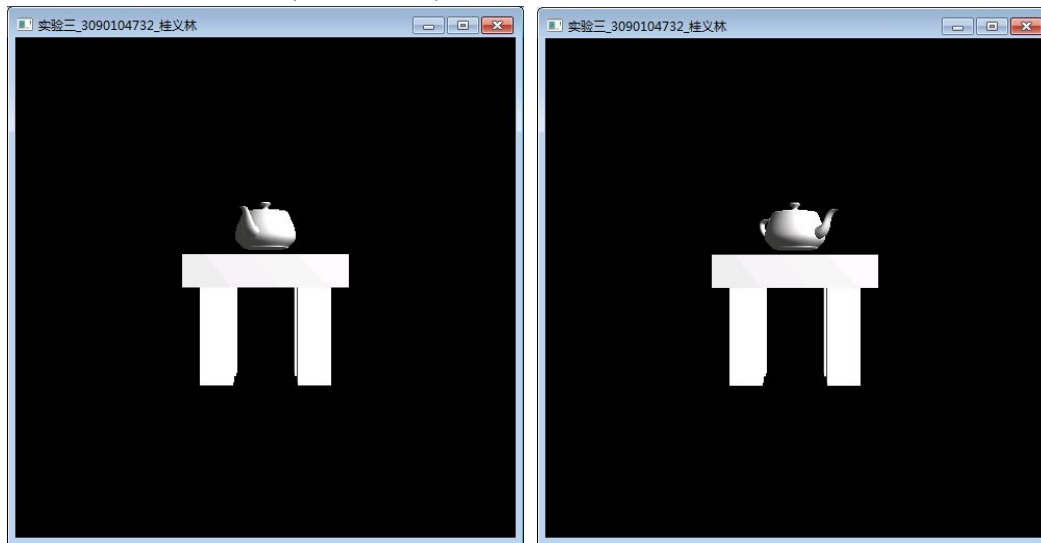




茶壶沿着桌面的平移:  $(j,l,k)$



茶壶绕自身轴的旋转：(按 e 键切换)



## 七、讨论、心得

通过本次实验，我理解并掌握了 OpenGL 中三维观察、透视投影、正交投影的参数设置，并能使用键盘移动观察相机，在透视投影和正交投影间切换，验证了课程中三维观察的内容；在实验中，我感觉到 OpenGL 提供的函数十分方便，编程者不需要了解变换矩阵的形式，只需了解各个参数的意义即可以轻松地完成观察相机的摆放；此外，在实现茶壶相对桌面运动和绕自身轴旋转的过程中，我进一步巩固了 OpenGL 中模型变换的内容，加深了对相对参考系的了解。