专业：计算机科学与技术
姓名：余启航
学号：3190103324
日期：2021.12.11
地点：宿舍

浙江大学实验报告

课程名称：计算机图形学 指导老师：童若锋 成绩：
实验名称：OpenGL 显示列表 实验类型：基础实验 同组学生姓名：

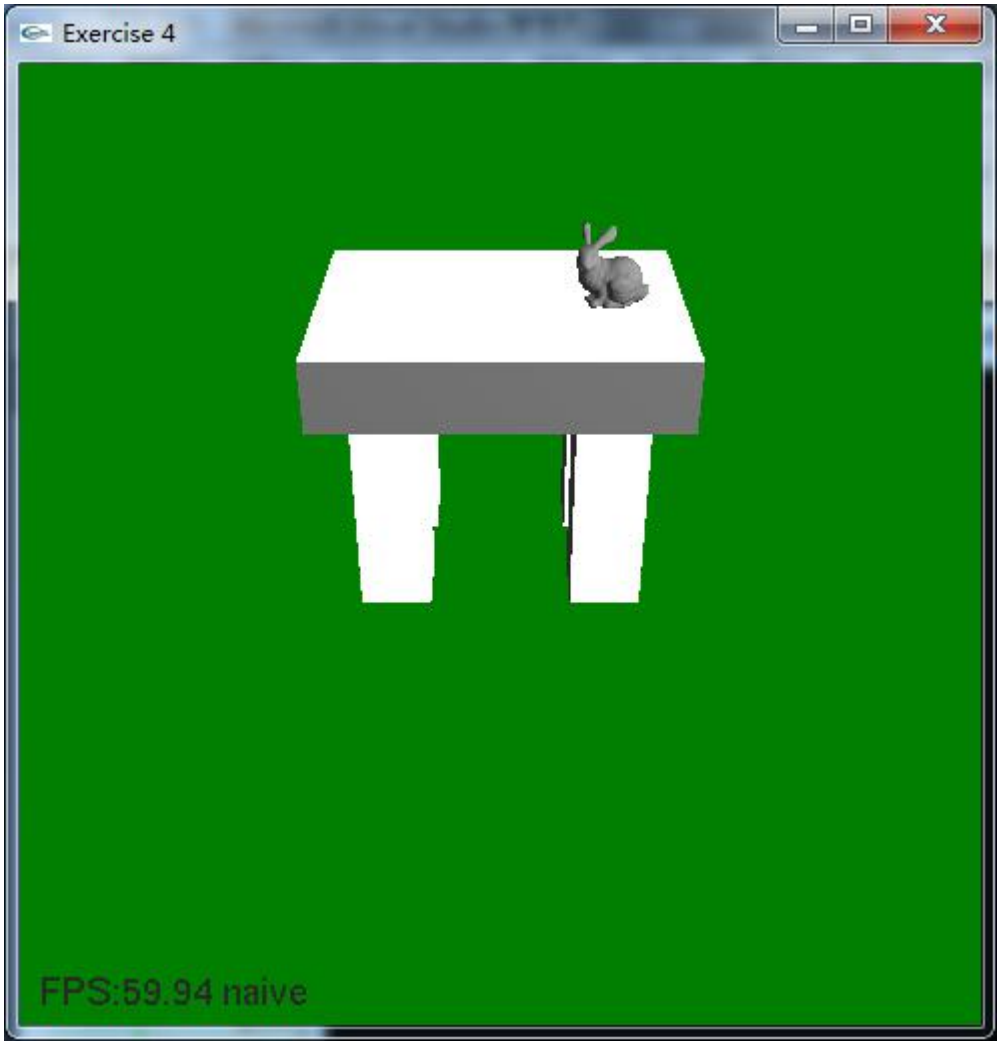
装 订 线

一、实验目的和要求

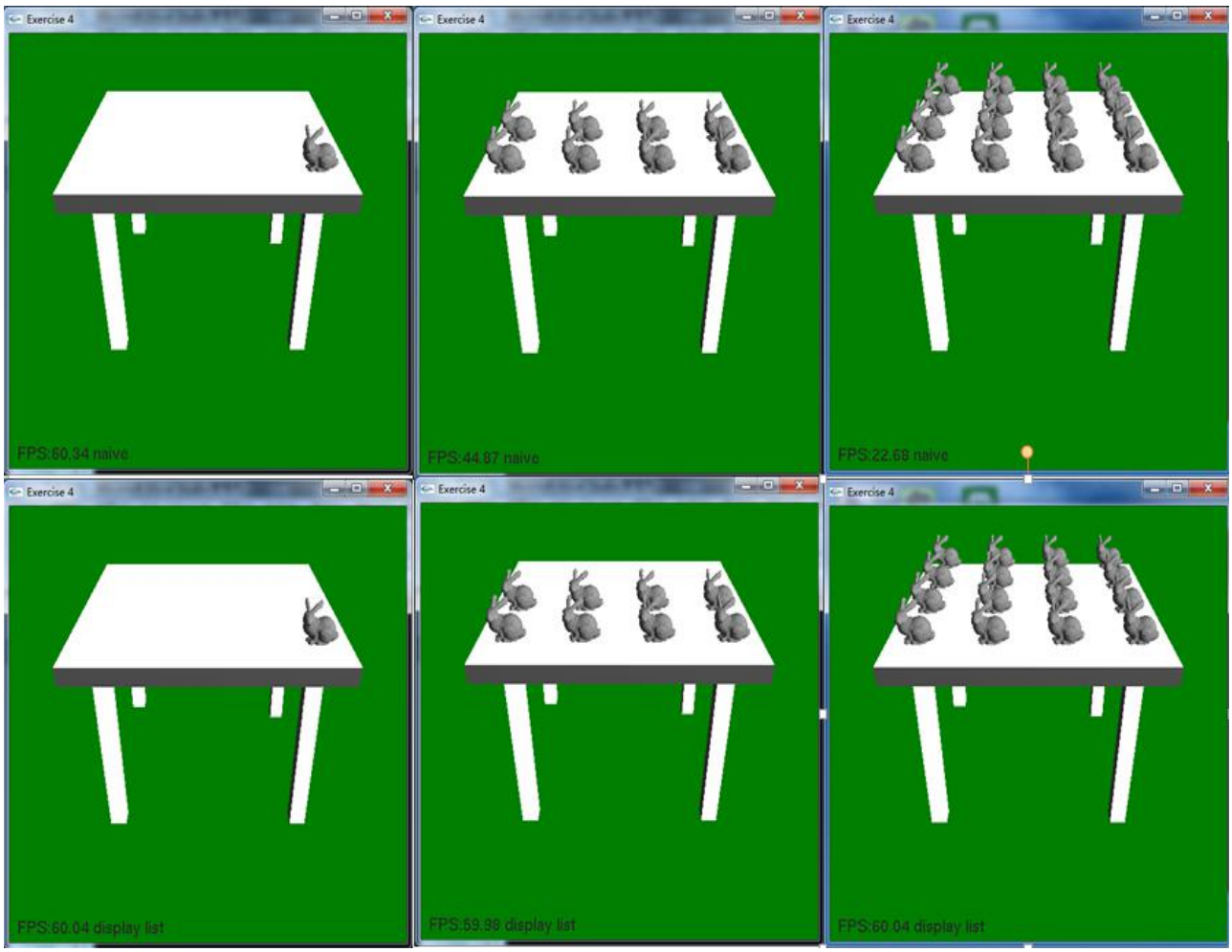
在三维观察实验的基础上，通过实现下述实验内容，掌握 OpenGL 中显示列表的作用和使用方法。

二、实验内容和原理

使用 Visual Studio C++编译已有项目工程，



修改代码，通过**键盘按键**，控制兔子的数量（1 至 16 个）以及整个场景的渲染模式，生成以下图形：



用按键 I、K 添加兔子数量增减（所有兔子均摆放着在桌面上，兔子间不要有交叉，桌面不够大可自行调整），按键 L 来切换显示列表和非显示列表绘制方式。WASDZC 控制上下左右前后移动,空格键控制整体旋转。

通过动画以及对 FPS 的理解和分析显示列表对程序绘制性能的影响。

三、主要仪器设备

Visual Studio 2017

glut.zip

Ex4-vs2010 工程

四、操作方法和实验步骤

1．创建显示列表

显示列表是一组存储在一起的 OpenGL 函数，可以在以后执行。调用一个显示列表时，它所存储的函数就会按照顺序执行。当一个显示列表被创建之后创建显示列表以 glNewList 开始，以 glEndList 结束。通过 glCallList 可以调用显示列表。在这次实验中，使用两个显示列表 tableList 和 rabbitList 分别绘制桌子和兔子。

```
GLint tableList = 0;
GLint rabbitList = 0;//兔子列表
GLint GenTableList(){
    GLint lid = glGenLists(2);//2 个空显示列表
    glNewList(lid, GL_COMPILE);//创建显示列表
    DrawTable();
    glEndList();
    glNewList(lid + 1, GL_COMPILE);//兔子显示列表
    DrawBunny();
    glEndList();
    return lid;
}
```

2．利用显示列表进行绘制

利用显示列表的绘制方式和普通绘制步骤完全一直，不同的是，绘制函数从原先的绘制改为 glCallList

```
void DrawScene_List()
{
    glPushMatrix();
    glTranslatef(2.2, 4.5, 1.5);
    glScalef(2, 2, 2);
    //画兔子
    for (int i = 1; i <= NumberOfRabbit && i <= 16; i++) {
        glCallList(rabbitList);
        if (i % 4 == 0) //换行
            glTranslatef(-0.7f, 0.0f, 1.5f);
        else
            glTranslatef(0.0f, 0.0f, -0.5f);
    }
    glPopMatrix();
    glCallList(tableList);
}
```

由于每一行只能容纳四只兔子，所以每次遇到四的倍数都需要进行调整，使兔子去下一行。

3．兔子数量控制

通过 i 和 k 可以控制兔子数量，要求是 1-16 只兔子的显示，所以不能跳出此范围

```
case 'i'://兔子数量增加
{
    if(NumberOfRabbit<16)
        NumberOfRabbit++;
    break;
}
case 'k'://兔子数量减少
{
    if (NumberOfRabbit > 1)
        NumberOfRabbit --;
    break;
}
```

4．fps 的计算

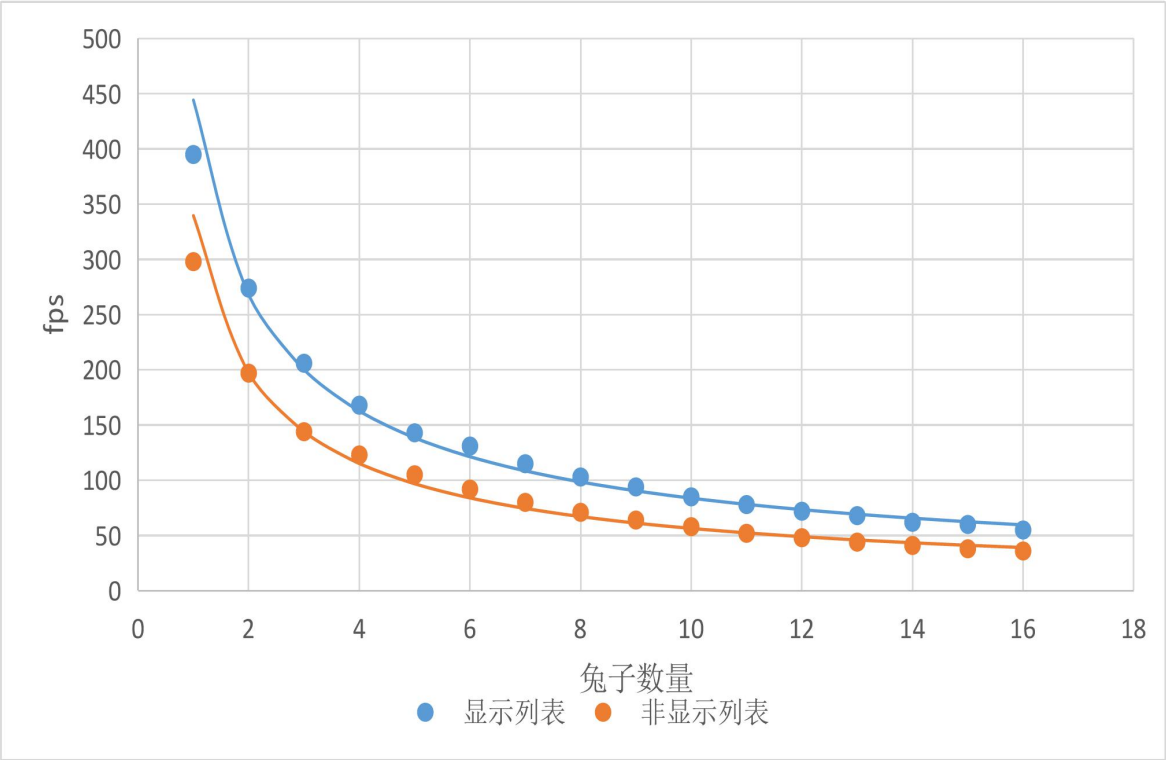
fps 是衡量动画信息储存、显示量大小的指标。每一帧都是静态图像，将每一帧快速、连续的显示出来，就产生了运动的错觉。这也是 Opengl 的执行流程。fps 直观的反应了动画的流畅程度。要计算 fps 需要计算单位时间内调用 redraw() 函数的次数，使用一个 frame 变量记录帧数，使用 glutGet(GLUT_ELAPSED_TIME) 返回两次调用之间的时间间隔，当时间达到 1s 时，更新 fps 的值，并通过 glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18, *c) 显示在窗口中。

五、实验数据记录和处理

1．表格

兔子数量	显示列表	非显示列表
1	395	298
2	274	197
3	206	144
4	168	123
5	143	105
6	131	92
7	115	80
8	103	71
9	94	64
10	85	58
11	78	52
12	72	48
13	68	44
14	62	41
15	60	38
16	55	36

2．图像



六、实验结果与分析

从实验数据结果可以看到，相同模式下，兔子数量越多，fps 越低，不同模式下，显示列表明显高于非现实列表。而显示列表的原理是提前将顶点和像素数据计算、编译并储存在内存中，之后可以直接调用而不需要再次计算、传输。因此显示列表非常适合绘制静态数据的图像，特别是时间开销的大的操作如：光照、纹理等。但缺点也是非常明显的，就是一旦编译后，数据就不能再修改。

七、讨论、心得

本次实验学习了显示列表的使用，明白了显示方式的多样性，针对不同的情况采取不同的显示方式有利于提高显示效率，减少消耗。同时，对于 fps 也有了初步的了解。