

计算机图形学实验

实验五

学 号： 20191130038

姓 名： 郑广敏

专 业： 智能科学与技术

年 级： 2019级

任课教师： 吴昊

2020 年 11 月 27 日

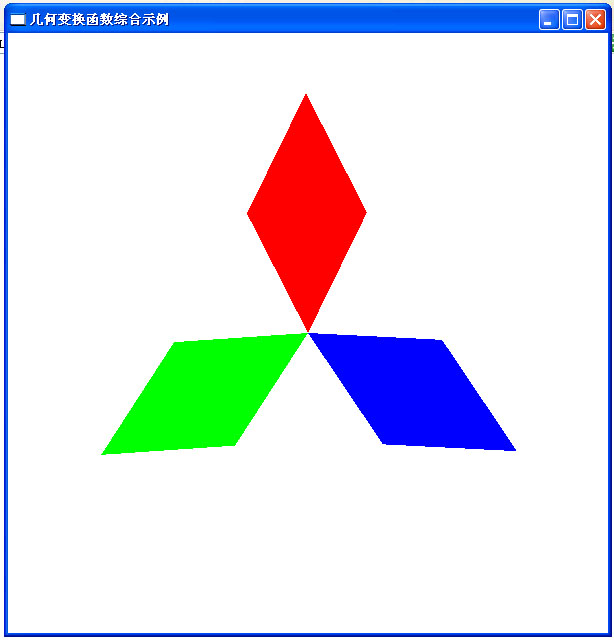
**一、目的和要求：**

通过实验掌握下列知识:

1. 理解变换的原理；
2. 掌握平移、旋转、缩放变换的方法；
3. 掌握以上方法的组合变换。

**二、实验内容：**

绘制如下图形：



提示：

（1）写一个绘制菱形的函数drawDiamond(void)；

void drawDiamond(void) //绘制中心在原点的菱形

{

glBegin (GL\_POLYGON); //顶点指定需要按逆时针方向

glVertex2f (0.0f,-1.0f);//下点

glVertex2f (2.0f,0.0f);//右点

glVertex2f (0.0f, 1.0f);//上点

glVertex2f (-2.0f,0.0f);//左点

glEnd ( );

}

（2）用几何变换绘制三个不同位置、旋转角度、颜色的菱形。

额外的任务：

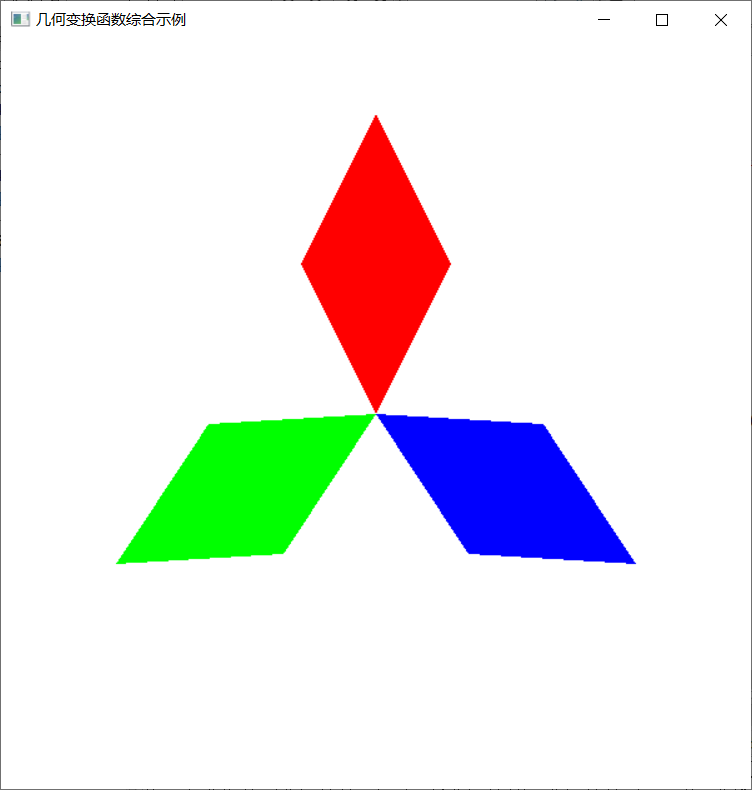
1. 为组合图形整体添加交互的平移、旋转、缩放功能。
2. 为单个图元添加交互的平移、旋转、缩放功能。

**三 关键步骤的源码和注释**

1. **float** theta[4];
2. **float** ud[4],lr[4];
3. **float** s[4]={1.0,1.0,1.0,1.0};
4. **int** ctr=0;//0整体，1红色，2绿色，3蓝色
6. **void** display(){
7. glClear (GL\_COLOR\_BUFFER\_BIT); //清空
8. glLoadIdentity();        //将当前矩阵设为单位矩阵
9. glScalef(s[0],s[0],s[0]);
10. glScalef(s[1],s[1],s[1]);
11. glTranslatef(lr[0],ud[0],0.0);
12. glTranslatef(lr[1],ud[1],0.0);
13. glRotatef(90.0+theta[0]+theta[1],0.0,0.0,1.0); //顺时针旋转90角度
14. glTranslatef(2.0,0.0,0.0);
15. glColor3f(1.0f,0.0f,0.0f);//红色
16. drawDiamond();
18. glLoadIdentity();        //将当前矩阵设为单位矩阵
19. glScalef(s[0],s[0],s[0]);
20. glScalef(s[2],s[2],s[2]);
21. glTranslatef(lr[0],ud[0],0.0);
22. glTranslatef(lr[2],ud[2],0.0);
23. glRotatef(30.0+theta[0]+theta[2],0.0,0.0,1.0);
24. glTranslatef(-2.0,0.0,0.0);
25. glColor3f(0.0f,1.0f,0.0f);//绿色
26. drawDiamond();
28. glLoadIdentity();        //将当前矩阵设为单位矩阵
29. glScalef(s[0],s[0],s[0]);
30. glScalef(s[3],s[3],s[3]);
31. glTranslatef(lr[0],ud[0],0.0);
32. glTranslatef(lr[3],ud[3],0.0);
33. glRotatef(-30.0+theta[0]+theta[3],0.0,0.0,1.0);
34. glTranslatef(2.0,0.0,0.0);
35. glColor3f(0.0f,0.0f,1.0f);//蓝色
36. drawDiamond();
38. glFlush();
39. }
41. **void** myKeyboard(unsigned **char** key,  **int** x, **int** y)//ad控制旋转
42. {
43. **if**(key>='0'&& key<='3') ctr=key-'0';
44. **if**(key == 'a' || key == 'A')
45. theta[ctr] += 5.0;
46. **if**(key == 'd' || key == 'D')
47. theta[ctr] -= 5.0;
48. **if**(key == 'c' || key == 'C')
49. exit(0);
50. **if** (theta[ctr]>360) theta[ctr] -=360;
51. **if** (theta[ctr]<0) theta[ctr] +=360;
52. glutPostRedisplay(); //重新调用绘制函数
53. }
54. **void** mySpecialKeyboard(**int** key,  **int** x, **int** y)//方向控制移动
55. {
57. **if**(key == GLUT\_KEY\_UP)   ud[ctr]+=0.1;//上
58. **if**(key == GLUT\_KEY\_DOWN)   ud[ctr]-=0.1;//下
59. **if**(key == GLUT\_KEY\_LEFT)   lr[ctr]-=0.1;//左
60. **if**(key == GLUT\_KEY\_RIGHT)   lr[ctr]+=0.1;//右
61. glutPostRedisplay(); //重新调用绘制函数
62. }
63. **void** myMouse(**int** button, **int** state, **int** x, **int** y){//滑轮控制缩放
64. **if**(button== 3 ){
65. s[ctr] += 0.05;
66. }
67. **if**(button== 4 && s[ctr]>0.1){
68. s[ctr]-=0.05;
69. }
70. glutPostRedisplay(); //重新调用绘制函数
71. }

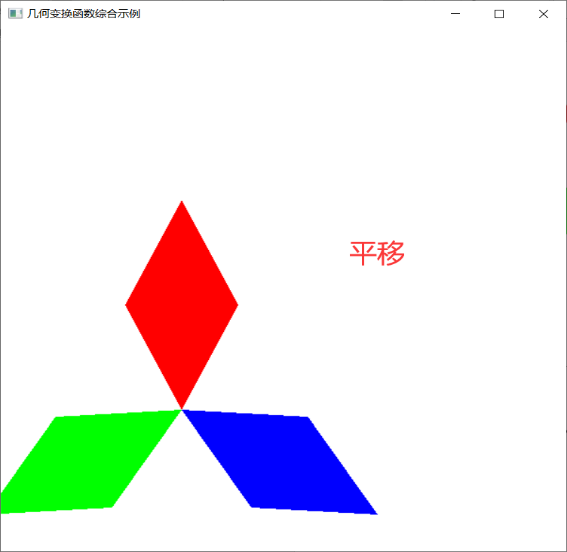
四 **实验结果（截图）**

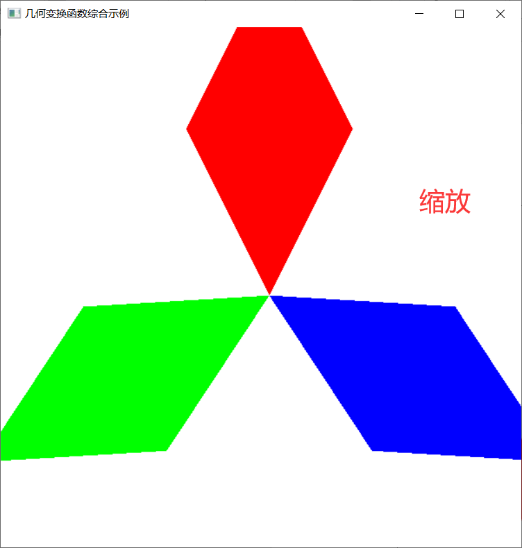
**1.图形绘制**



**2.额外的功能**

**（1）整体的旋转、平移、缩放**





**（2）对单个图元的操作**



**五、实验分析**

要注意平移、旋转、缩放操作的顺序，不同顺序执行同样地操作，结果可能天差地别，另外该题功能单调，可以尝试增加交互的操作对其改进，考虑到该图是一个组合图形，因此操作包括对整体的操作以及对单个图形的操作，考虑设置数组theta[4]表示旋转角度，u[4]d表示上下平移量，lr[4]表示左右平移量，s[4]表示缩放比例，用来存储对整体和单个图元的操作量，整体的操作量作用于所有图元，而单个的操作量作用于自身即可。