图形交互, 缩放+移动+错切+旋转

学号:15307130083, 姓名:刘瑞林

算法说明

新建一个python类:

```
class interact():
    centre = 图像中心点
    points = 四个端点
    OperatorRec = 外切该图像的矩形
    MouseDown = 鼠标是否点下
    interactType = 交互的类型
    shear_direction = 错切的方向
    InteractMatrix = 获取矩阵函数
    # 4种变换的矩阵
    def shift(self, x, y):
    def zoom(self, x, y):
    def rotate(self, x, y):
    def shear(self, x, y):
    # 获取外切矩形
    def GetOperatorRec(self, points):
    # 绘制图形, 外接矩形, 外接矩形的外接圆
    def Draw(self, points):
    # 鼠标移动的回调函数
    def ActiveMotion(self, x, y):
    def Display(self):
    # 鼠标点击的回调函数
    def MouseClick(self, button, state, x, y):
    def Reshape(self, w, h):
```

初始化OpenGL

```
    ob = interact()
    glutMouseFunc(ob.MouseClick)    # 鼠标按下
    glutMotionFunc(ob.ActiveMotion) # 鼠标按下后, 移动
    glutDisplayFunc(ob.Display)     # 鼠标未按下, 或鼠标按下不移动
    glutIdleFunc(ob.Display)

    #glutReshapeFunc(ob.Reshape())  # 缩放窗口, 未实现
    glutMainLoop()
```

流程说明:

1. 鼠标左键按下时,
    1. 记录x,y坐标
    2. 判断鼠标 在 顶点附近, 设置变幻类型为缩放
    3. 判断鼠标 在 边附近, 设置变幻类型为错切
    4. 判断鼠标 在 圆附近, 设置变幻类型为旋转
    5. 判断鼠标 在 中间, 设置变幻类型为平移
    6. 否则不发生变换
2. 鼠标移动时(active motion),
    1. 获取坐标 x, y
    2. 与鼠标按下时的坐标作计算
    3. 获取变换类型对应的矩阵
    4. 平移到(0, 0)
    5. 左乘变换矩阵
    6. 平移回去
    7. 绘制图形
3. 鼠标松开:
    1. 更新points, centre
4. Display:
    1. 若鼠标按下且静止, 计算变换后的位置, 绘制
    2. 若鼠标未按下, 直接绘制

绘制说明:

1. 图形使用glBegin(GL_POLYGON)绘制
2. 矩形使用glBegin(GL_LINE_LOOP)绘制
3. 圆使用glBegin(GL_LINE_LOOP)绘制, 一共绘制圆上连续的100个点

程序使用说明

运行interact.py
中间的矩形为被操作的图形
外接的矩形与圆 指示鼠标可以操作的位置:
    1. 在外接矩形内部操作平移
    2. 在外接矩形顶点操作缩放
    3. 在外接矩形边 操作错切
    4. 在外接圆 操作旋转