Report-pj1

一、项目目录及文件说明

```
├─20302010040-于康
   |--report.pdf
   |
   └─src
         --config.js
         --scanConversion.html
```

> report.pdf：项目文档
>
> src：代码部分
>
> - config.js：项目的功能实现文件
> - scanConversion.html：前端显示的html文件

二、代码结构

函数名	作用
drawPoint	绘制一个点
drawLine	绘制一条路径
drawCircle	为顶点增加圆形手柄
getMouseLocation	判断鼠标位置是否在画布内且圆形手柄内，若是则返回圆形手柄下标
doMousedown	鼠标按下事件
doMousemove	鼠标移动事件
doMouseup	鼠标松开事件
drawQuad	绘制四边形网格
fillPolygon	实现了多边形扫描转化，包含以下四个函数： - **initNET**：初始化新边表 - **insert**：添加新的活性边到链表 - **remove**：移除链表中非活性边 - **update**：更新活性边表每项x_i，并重新排序

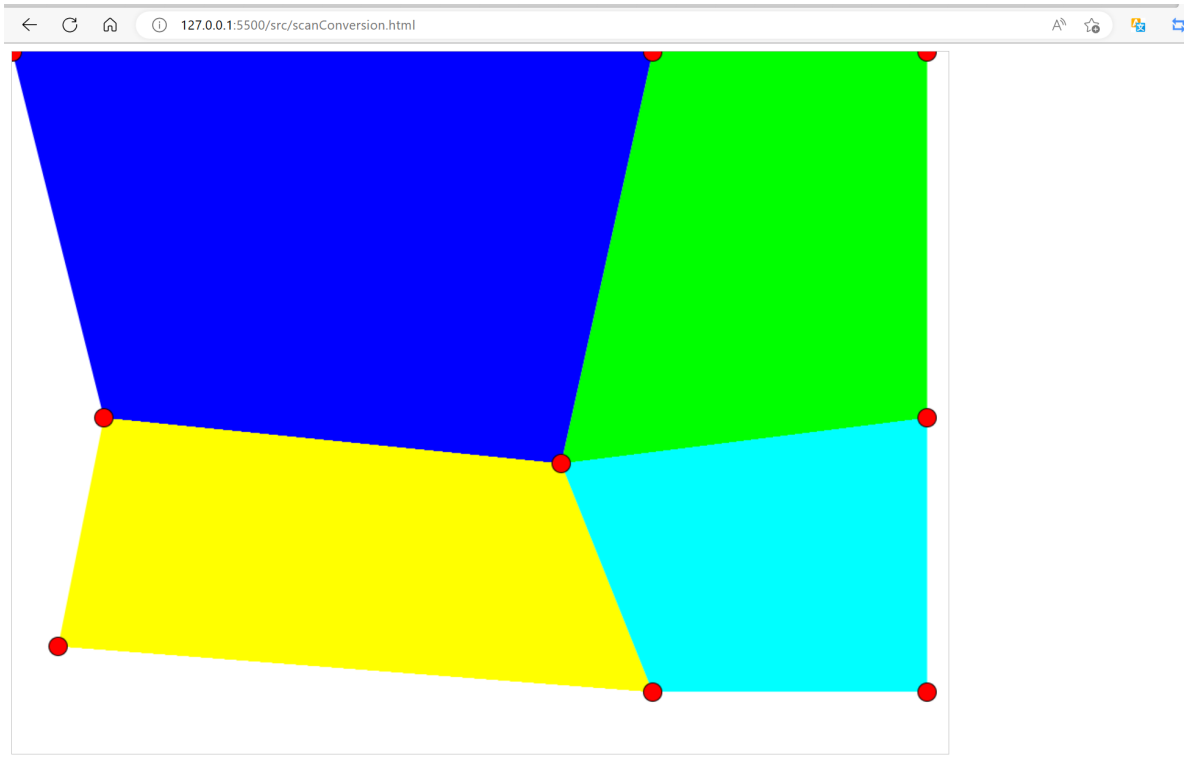
三、开发及运行环境

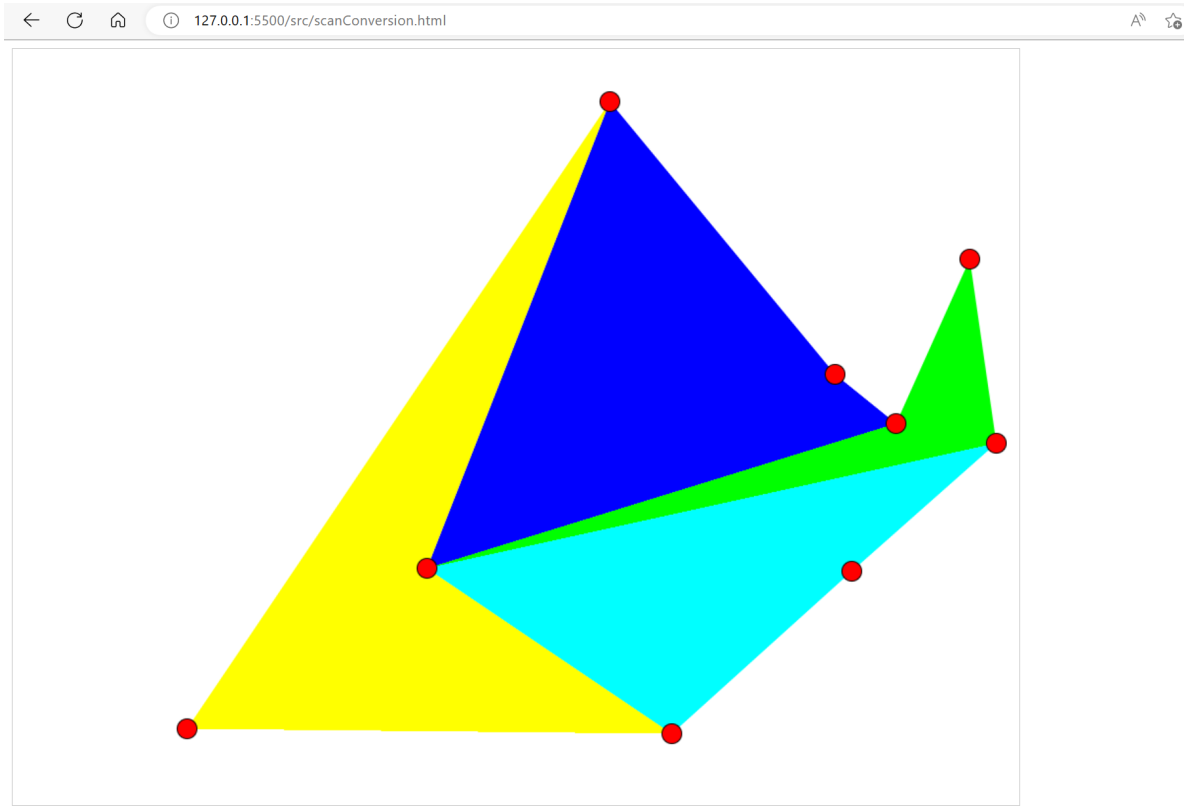
开发环境：Visual Studio Code

运行环境：Microsoft Edge，版本 111.0.1661.51 (正式版本) (64 位)

四、运行及使用方法

在src目录下直接打开*scanConversion.html*即可，画面上会显示出四边形网格；在画布内拖动圆形手柄即可改变四边形顶点位置，重新进行扫描绘制。





五、项目亮点

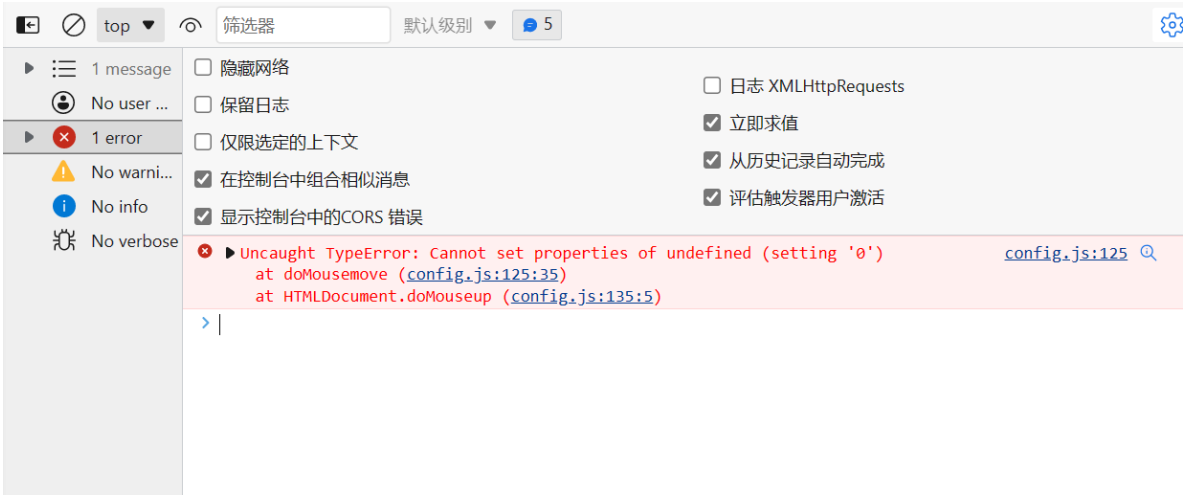
实现了多边形扫描转化（不只是支持四边形），采用增量的思想，借助活性边表的数据结构，避免了直线的求交运算。

```
    class Edge {
        constructor() {
            this.xi = 0;
            this.dx = 0;
            this.ymax = 0;
            this.id = 0;
        }
    }
```

六、开发过程中遇到的问题

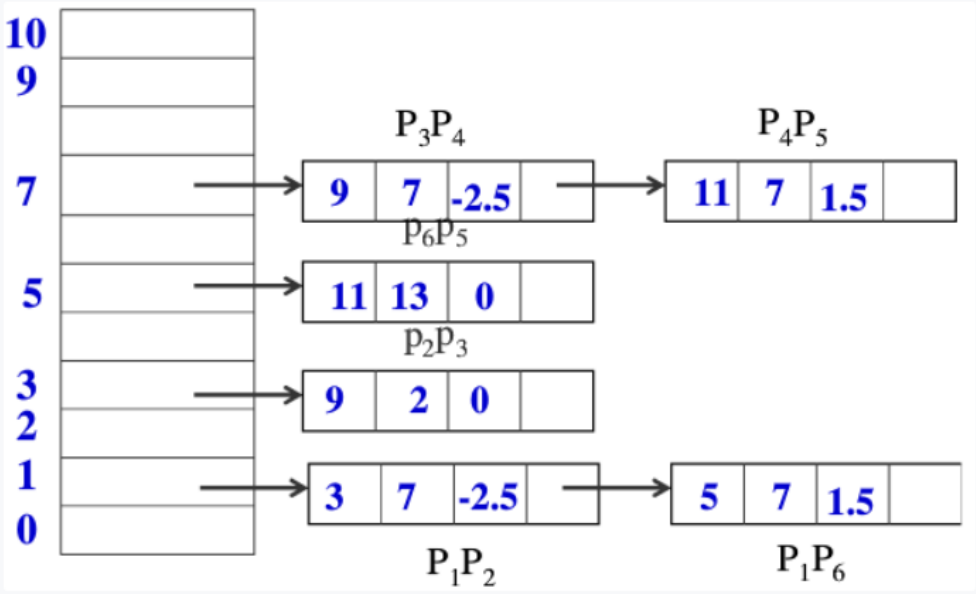
首先遇到的问题就是如何设计适用更广泛的多边形扫描算法，通过该文章[计算机图形学——多边形的扫描转换（基本光栅图形算法） - 王陆 - 博客园 (cnblogs.com)](), 学习到采用增量的思想进行扫描，将整个扫描过程转化成对于数据结构的维护。

在鼠标松开事件处理函数中，起初加入了一次重新绘制，导致出现bug。因为松开事件可以单独在顶点外发生，由于被选中点是在有效的鼠标按下事件后更新的，此时无被选中顶点。



七、项目缺陷及思考

在多边形扫描算法中，借助新边表来存放多边形边的信息，放入对应的y值上：



而对于本pj，图形简单但边也较长：

```
var vertex_pos = [
    [0, 0, 0],
    [700, 0, 0],
    [1000, 0, 0],
    [100, 400, 0],
    [600, 450, 0],
    [1000, 400, 0],
    [50, 650, 0],
    [700, 700, 0],
    [1000, 700, 0]
];
```

此时开辟(ymax-ymin+1)个数组来存放所有边有些浪费，可能能够使用更少的数组来实现。