

Weiler-Atherton算法实现

实验环境

- Windows11
- Python3.8
- PyQt5.15

实验目的

实现一个以Weiler-Atherton算法为核心的多边形裁剪程序。

实验方法

实验主要有两个核心部分：多边形绘制，WA算法。

由于有内环存在，多边形(目标多边形、裁剪多边形)的顶点设计了二重列表来保存， $list[i][j]$ 即为第*i*个多边形的第*j*个点。

多边形顶点读入后，需要判断是外环还是内环，这点通过遍历多边形的环判断顶点是否在其他多边形环内部即可。由于题目要求多边形不会自交，因此只需判断每个环上的一个点即可。判断点在环内还是环外只需计算由此点到各顶点连线形成的夹角大小是否为 2π 即可。

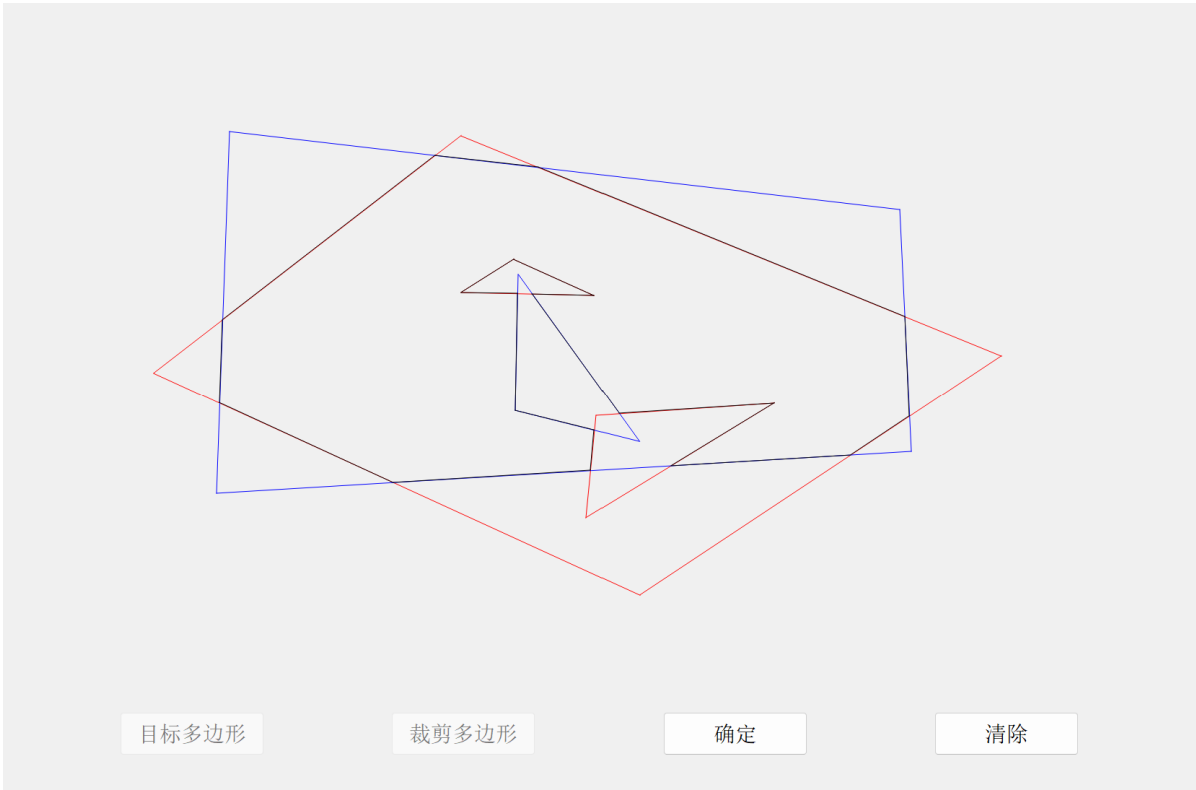
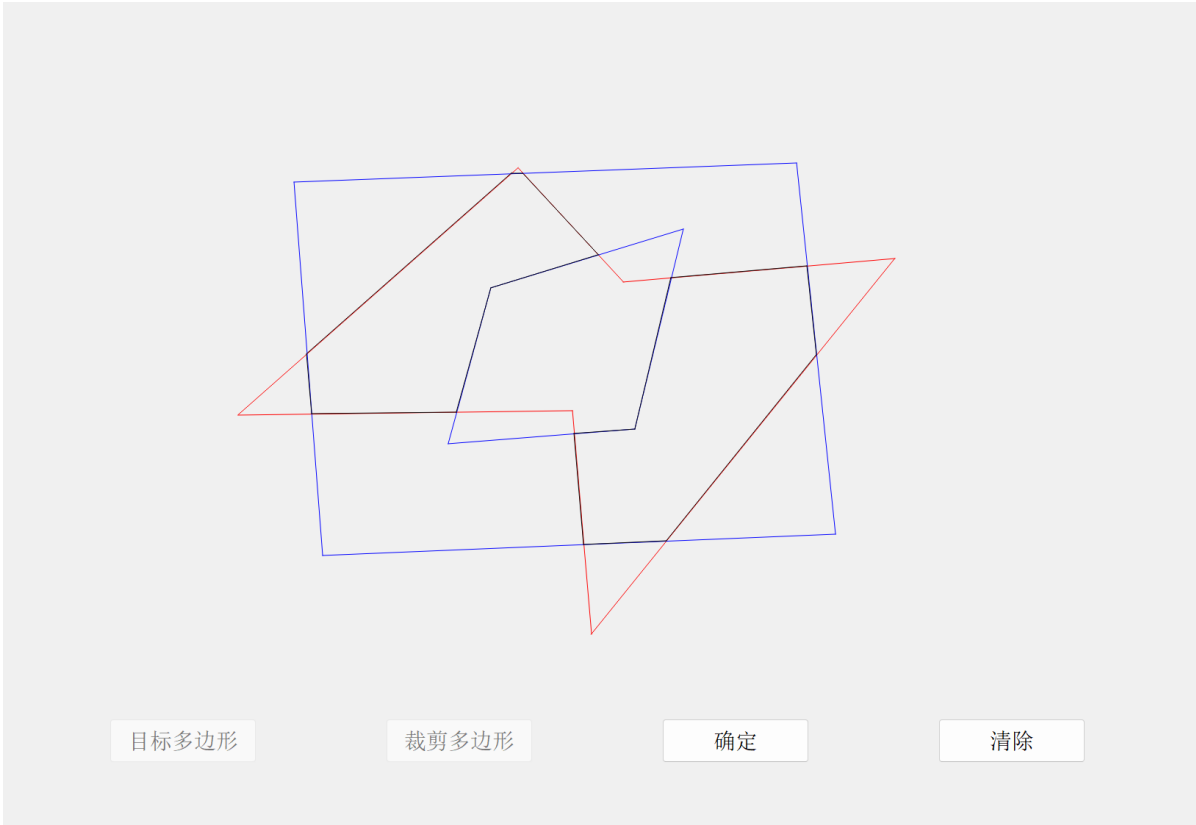
WA算法之前，还要求出各线段之间的交点。在已知两点的情况下，可用两点表示出直线的一般式方程。两直线的交点有相应的相应的数学公式求解。求出交点后判断交点是否在4个顶点限制的范围内即可。

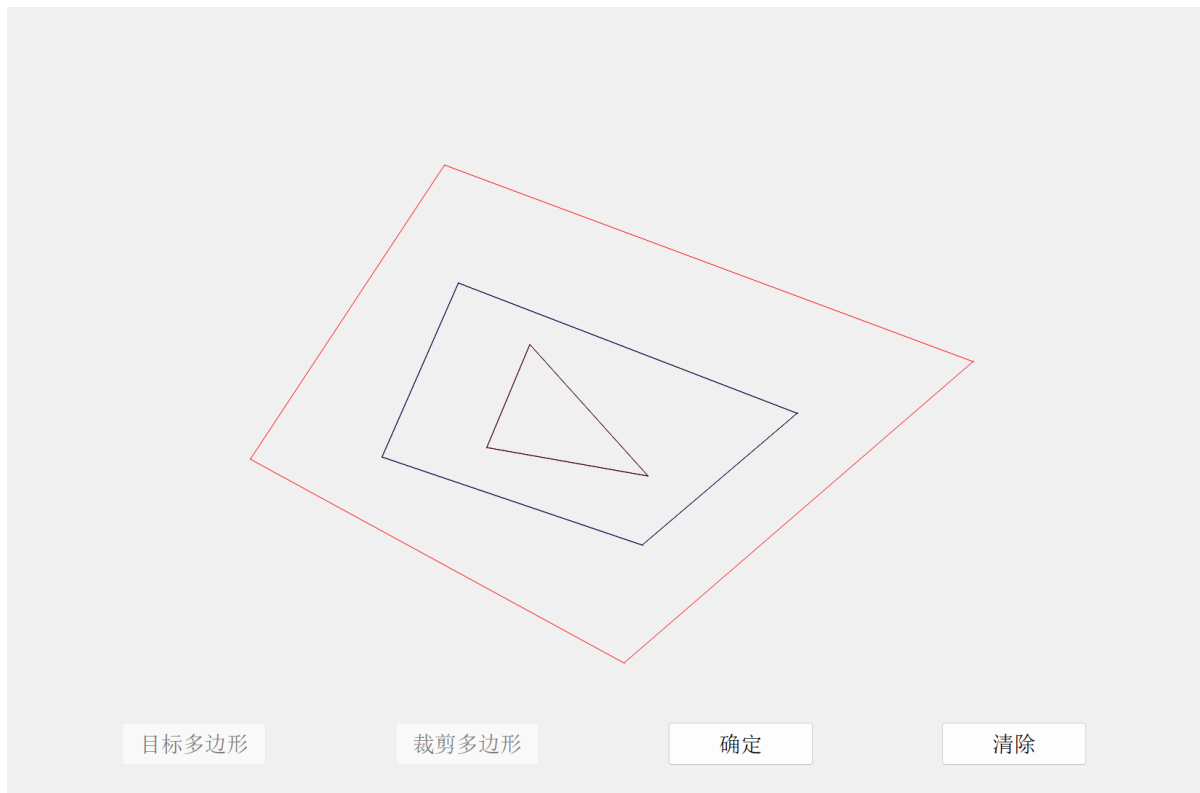
对于没有交点的多边形，直接利用类似之前判断内外环的方法，判断目标多边形与裁剪多边形的内外关系，将符合的环加入结果。

WA算法按课件描述，在两个链表之间不断跳转，直到所有节点都被打上visited标签。

最后，将结果集中的多边形绘制在图上。

实验结果





交互方式

运行可执行文件，进入交互界面，默认选中“目标多边形”绘制状态。在程序窗口内左键按下选点，右键闭合，绘制出多边形（红色）。

点击“裁剪多边形”，切换至绘制裁剪多边形状态（蓝色）。

点击“确定”，运行WA算法求结果多边形（黑色）。

点击“清除”，清除当前所有内容，整个算法从零开始，重新开始输入点序列。

注：多边形至少要有三边才能闭合；多边形不能自交，包括各环本身以及各环之间；内环不能套内环

编译方式

src目录下运行 `python main.py` 即可，但是系统需要提前安装pyqt相关的包(可以使用 `pip install -r requirements.txt`)。