C 组计算几何项目报告

带孔多边形三角剖分算法在 2.5 D 地图渲染中的应用

小组成员: 国孟昊 陈浩翔 陈铮

2021 年 6 月 18 日

目 录

1 问题描述

三角剖分是计算几何中的一种常见的算法，在可视化、计算机图形学以及地理信息系统中有着广泛的应用。本次项目选取了一个三角剖分在可视化领域的一个应用-2.5D 地图可视化。2.5D 地图是指在二维地图基础上，对所有建筑物来说，额外增加了高度信息，可以给人的带来更直观的感受。每一个建筑物用若干个（带孔）多边形表示，假设地图已被切分成若干个多边形，我们需要将所有的多边形（带孔）进行三角剖分算法，然后利用 OpenGL 对所有的三角面片进行着色、渲染，最后通过交互界面展现渲染后的 2.5D 地图。

2 运行环境

本系统运行在 ubuntu 系统中，需安装 python3.6 及以上版本。
python 依赖：

- numpy
- opencv
- prioq (优先队列)
- dendroid (包含红黑树的实现)
- reprit (方便直接查看对象变量的内容, 与算法主体无关)
- symba (Python 定义抽象类所需的库，与算法主体无关)

C++ 依赖：

- boost
- opencv >= 3.4.0
- openGL
- Pangolin
- Eigen3

3 算法步骤

本章节将会对算法的每一步进行介绍，每一步的执行结果如图.1所示。

3.1 对所有点集进行 delaunay 三角剖分

我们使用分治算法实现 delaunay 三角剖分。首先，我们将所有点按照从小到大对 **x** 进行排序。排完序后，我们可以对点集进行不断左右分治，直到某个区域内点的个数小于等于 3，此时我们就可以把这个区域内的点剖分成三角形或者线段，并结束向下递归，然后向上进行回溯。在回溯的过程中，我们需要将已经剖分好的左右子集进行合并。合并的第一步是插入底部的公切线，不妨将这条线记为 L_t, 其左端点记为 P_l，右端点记为 P_r。然后，我们可以先考虑左侧点集 S_l，对于每个与 L_t 相连的顶点 P_i，都有可能与 P_r 构成新的边。因为我们是进行 delaunay 三角剖分，因此要对每个点进行如下的检测:

- P_l, P_r 和 P_i 三点构成的圆的内部不包含其它的点。
- P_l 和 P_i 构成的线段与 L_t 的夹角小于 180 度。

将满足上述条件的 P_i 和 P_r 进行连线，并做为新的 L_t。依次类推，可以将左侧点集添加完毕。同理，我们可以对右侧点集进行相同的处理。

加完边之后，我们还要考虑加上的边是否与已有的边有所冲突。我们需要删除两类边：

- 与新边相交的老边
- 判断左边新加的点所对应圆是否包含右边点，若包含则删除，右侧同理。

通过上述方法，不断合并，然后得到最终的 delaunay 三角剖分。

3.2 对已有的三角剖分进行内边约束

在做完无约束的三角剖分之后，我们在此基础上加上边的约束。这些边的约束是指两种边约束。一种是外环上的边约束，另一种是内孔上的边约束。首先对于内孔上的约束边，对于每一条约束边，我们要都要执行以下操作。首先，判断约束边是否已经在已有的三角剖分里面，如已经在已有的三角剖分里面，该边判断完毕，判断下一条边。若该边不在已有的三角剖分内部，则查找已有的三角剖分和该边是否存在相交的边，若存在，则把相交的边去掉，加上带约束的新边。对新加入边的相邻的三角形进行 delaunay 约束检测，根据约束更新三角剖分的内边。

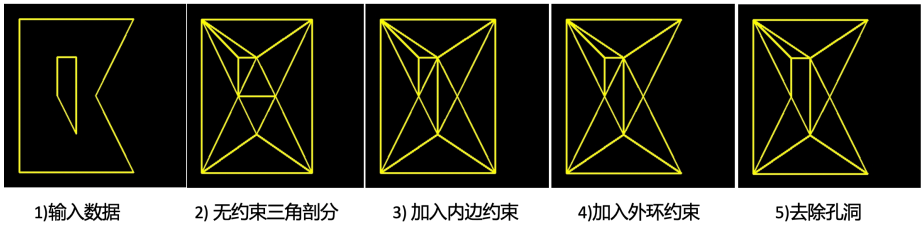

图 1: 算法执行每一步后的效果展示

3.3 对已有的三角剖分进行外环约束

对于外环上的约束边，我们使用 QuadEdge 结构，找到已得到的三角剖分最外围的、并且不在 outer polygon 里的所有边 outer delaunay。然后，利用广度优先搜索 outer delaunay，可以将多边形外环 outer polygon 以外的边找到，并将其剔除。

3.4 去除孔洞

我们使用了扫描线算法对进行去除孔洞。首先创建事件优先队列 EventQueue Eq。然后找到三角剖分的所有内边，并且将内边在事件队列 Eq 里进行注册。最后使用扫描线算法扫描队列中的每一个事件（每一条内边），如果这条边 xx 且 xx, 则删除该边，以此类推直到扫描结束。

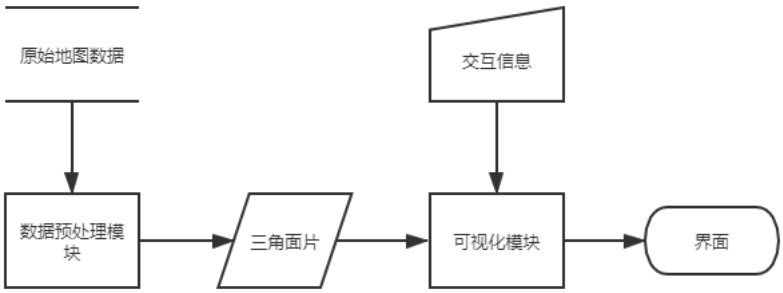

图 2: 总体数据流图

4 数据结构与模块设计

本应用系统由两大模块组成，数据预处理模块和可视化模块。数据预处理模块离线地读入地图数据，对其进行三角剖分，输出点、线、面。可视化

模块读入点线面信息，向用户提供可交互的 3 维 2.5D 地图。系统总体数据流图如图2所示。

4.1 数据预处理

4.2 可视化

可视化模块有数据存储与交互模块，界面模块，渲染模块和用户交互处理模块共 4 大模块。可视化模块调用图见图3。

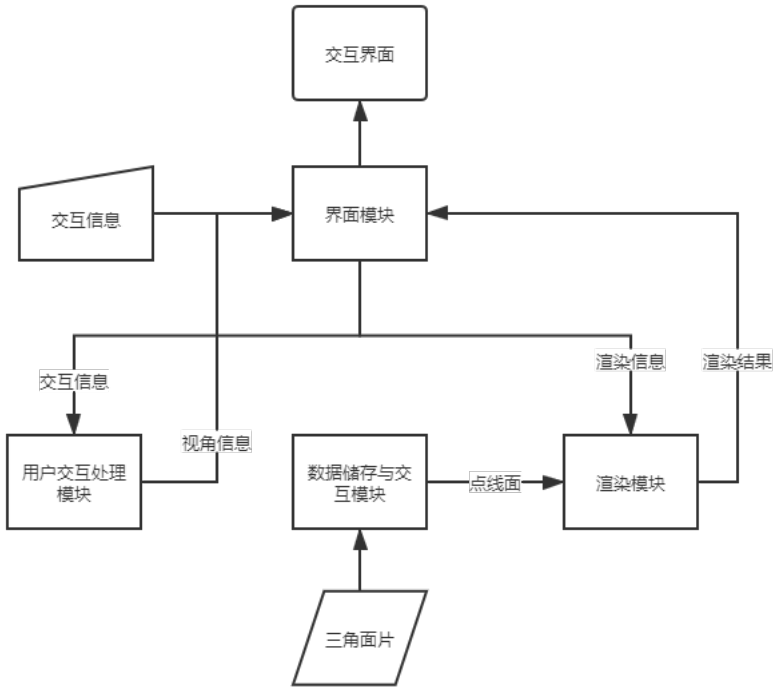

图 3: 可视化模块数据流图

4.2.1 数据存储与交互模块

该模块负责数据的读入、存储以及用 OpenGL 将点线面上传至 GPU 的功能。该模块由 MyPoint 类、MyBox 类和 MyBoxList 类构成，主要功能类为 MyBoxList 类。该类会从文件中读入数据预处理模块生成的三角面片信息，并将其上传至 GPU 显存中。

4.2.2 界面模块

该模块负责生成用户交互界面,获取用户交互信息。该模块由 MyViewer 类构成，该类定义了用户交互界面的各个按钮和拖拽条。该类还定义了交互处理句柄和渲染模块实例用于与用户交互处理模块和渲染模块交互。

4.2.3 渲染模块

该模块负责调用 OpenGL 渲染出某个视角下的三维地图，传入界面模块以供用户观看。该模块由 MyDrawer 类、MySensor 类和 MyShader 类构成。主要功能模块为 MyDrawer 类，该类定义了数据储存模块实例，实现了地图各部分各自的渲染函数。MyShader 类在程序内定义了地图各部分的着色器，为 OpenGL 渲染必须的组件。MySensor 类与用户交互模块进行交互，定义了用户输入的视角，用于渲染。

4.2.4 用户交互处理模块

该模块负责处理用户的键鼠输入，产生对应的视角。该模块由 MyHandler 类构成，该类继承了 Pangolin 中的 Handler3D 类，实现了根据鼠标左右键拖拽和滚轮滚动来控制视角的移动变化和旋转。

5 鲁棒性测试

5.1 不带孔情况

1. 我们对三点共线情况进行测试，对于输入点 (0, 0), (3, 0), (1, 2), (0, 3)。我们的程序得到的三角剖分是正确的，答案是 [(0, 0), (1, 2), (3, 0)] 和 [(0, 0), (1, 2), (0, 3)]。

2. 我们对存在重复的点的情况进行了测试。对于输入点 (0, 0), (3, 0), (1, 1), (1, 1), (0, 3)。我们的程序会自动进行去重复,得到正确的结果,运行得到的答案为:[(0, 0), (3, 0), (1, 1)], [(0, 0), (1, 1), (0, 3)], [(0, 3), (1, 1), (3, 0)]。

5.2 带孔情况

1. 我们对内环和外环两个环存在重合点的情况进行了测试，对于输入外环 (0, 4), (0, 0), (4, 0)，输入内环 (1, 2), (0, 0), (2, 1)，重合点为 (0,0)，程序得到的结果是正确的，运行结果为 [(0, 0), (4, 0), (2, 1)], [(1, 2), (2, 1),

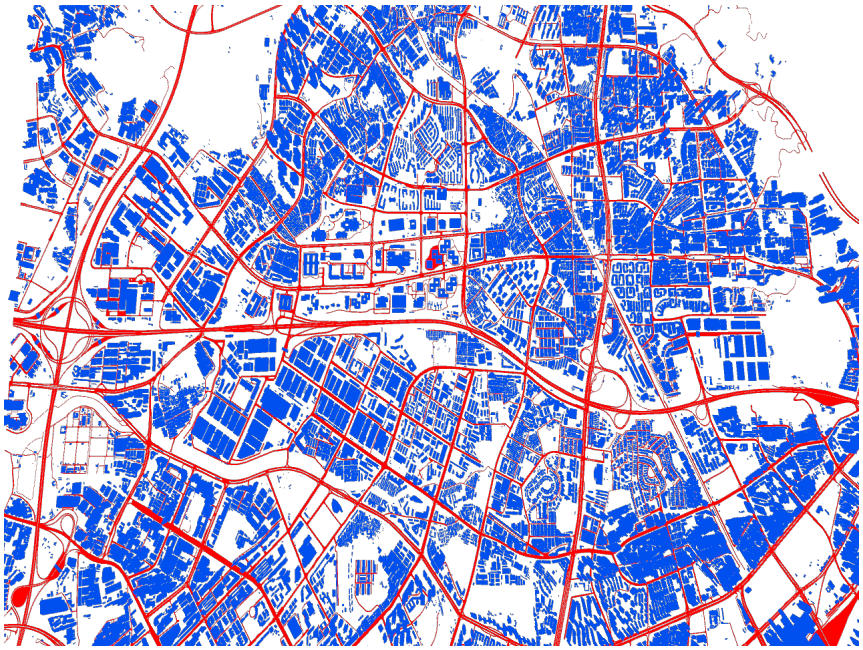

图 4: 全局地图俯瞰图。红色为道路，蓝色为建筑。

(4, 0)], [(0, 4), (1, 2), (4, 0)], [(0, 0), (1, 2), (0, 4)]。

2. 我们对外环中存在重合点，以及内环和外环两个环存在重合点的情况进行了测试，对输入外环 (0, 4), (0, 0), (0, 0), (4, 0)，输入内环 (1, 2), (0, 0), (2, 1)。外环重合点为 (0,0)，内外环重合点为 (0,0)。我们的程序的输出结果是正确的的，运行结果为 [(0, 0), (4, 0), (2, 1)], [(1, 2), (2, 1), (4, 0)], [(0, 4), (1, 2), (4, 0)], [(0, 0), (1, 2), (0, 4)]。

6 性能评估

7 地图渲染结果展示

本章节会展示运用上述带孔多边形三角剖分算法对 2.5D 地图进行三角剖分后的渲染结果。

由于本环节使用的地图数据为实验室所有，仅能用于报告和 demo 展示，包中 data 文件夹中为原始地图数据各个多边形独立去除位置信息后的数据，可用来测试带孔三角剖分算法的正确性，无法形成地图。如需数据，请联系陈浩翔，邮箱为 chx20@mails.tsinghua.edu.cn。

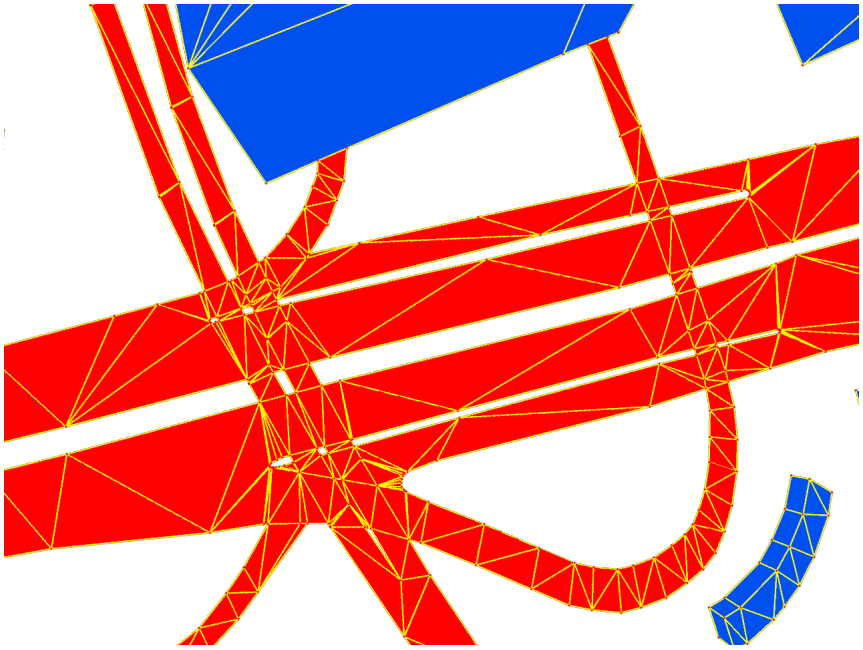

图 5: 带孔路面三角剖分结果。红色部分为道路面，也是三角面片，黄色线条为三角形的边，红色道路中间空出的部分即为孔洞。

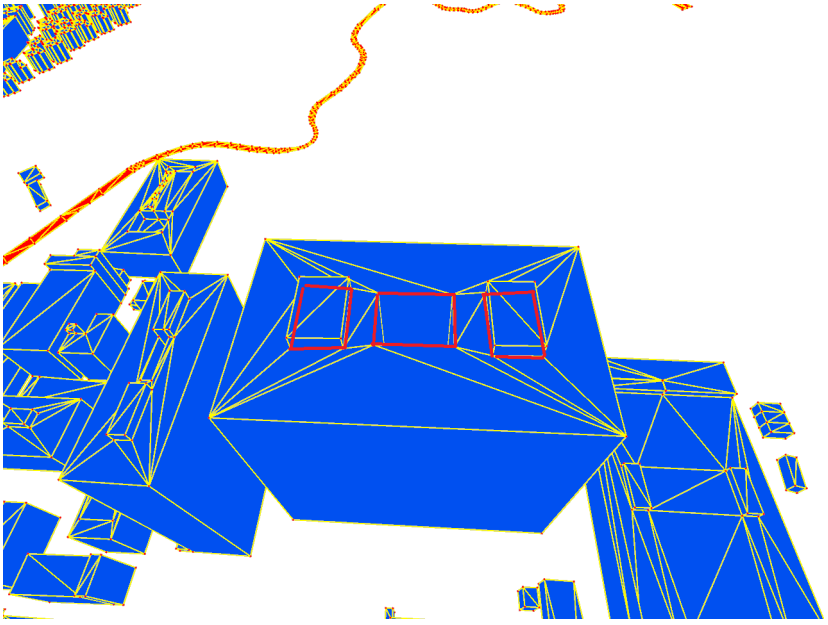

图 6: 带孔建筑三角剖分结果。红框圈起部分为顶层多边形的三个孔，由于三维遮掩，中间的孔被墙挡住，左右两边的孔被更高的方柱填补，黄色线为三角边。

人员	工作
国孟昊	算法步骤二添加边的约束、对三角剖分所有模块的调试等，并且编写相关工作的文档。
陈浩翔	算法步骤三利用外边界约束模块和 OpenGL 可视化等，并且编写相关工作的文档。
陈铮	算法步骤一 delaunay 三角剖分和算法步骤四扫描线去孔，并且编写相关工作的文档。

表 1: 分工情况

8 小组分工

分工情况如表. **??**所示。

9 实验总结

参考文献