

计算机视觉课程

——基于图像的三维模型重建（上）



主讲人 隋博士



课程内容

✓ 基于图像的三维模型重建

- ✓ 基本流程
- ✓ 应用场景

✓ 三维模型的表述方式

- ✓ 边界表述方法
- ✓ 空间划分法

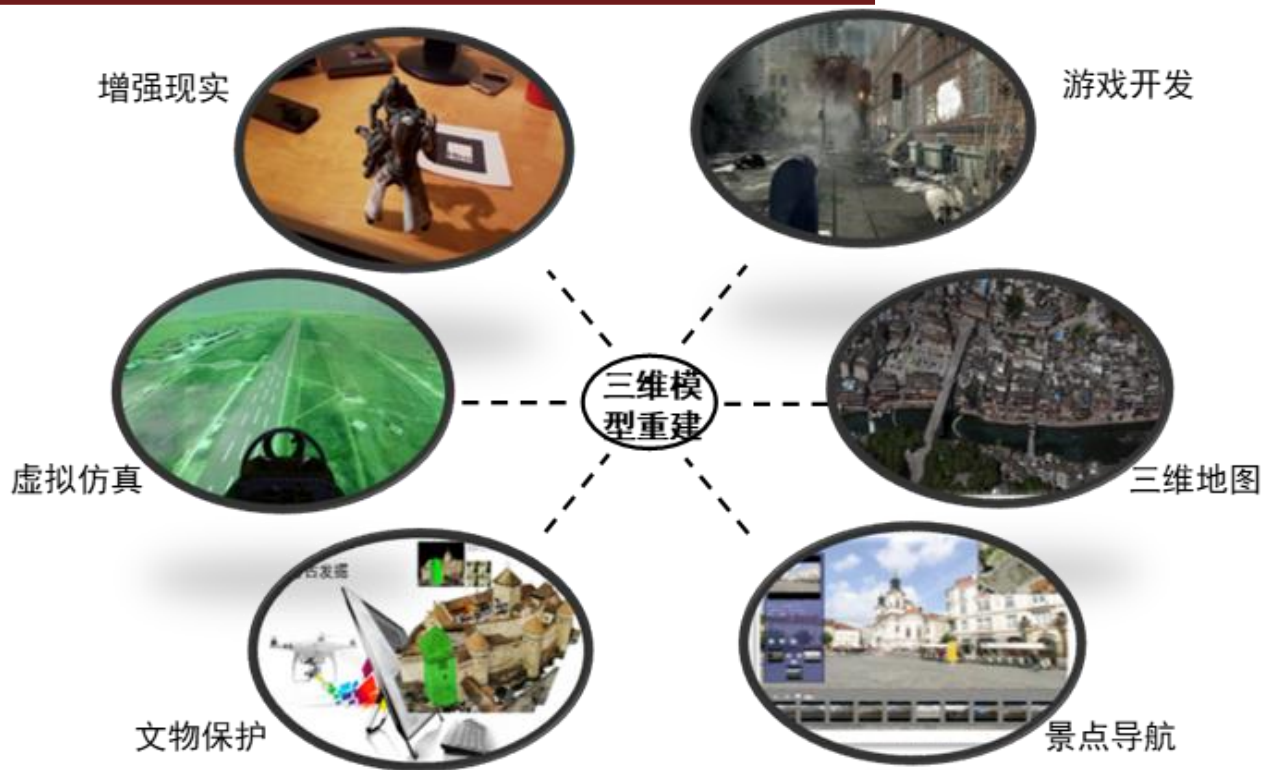
✓ 从点云到网格模型

- ✓ 德劳内三角剖分
- ✓ 基于体素的重建方法
- ✓ 基于隐函数的重建方法

✓ 城市建筑的三维模型重建

- ✓ 基于基本元素检测与排列的方法
- ✓ 基于特征保持的隐函数重建方法

三维模型重建应用广泛

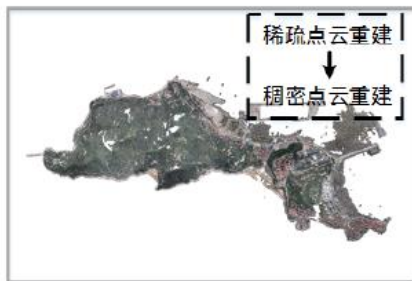


基于图像的三维模型重建流程

输入图像



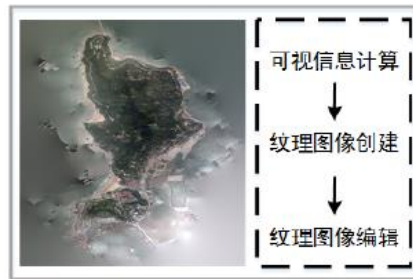
基于图像的点云生成



点云到模型的重建

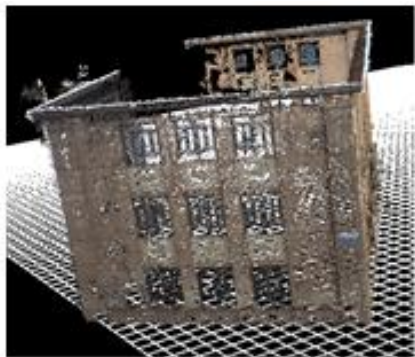


纹理模型

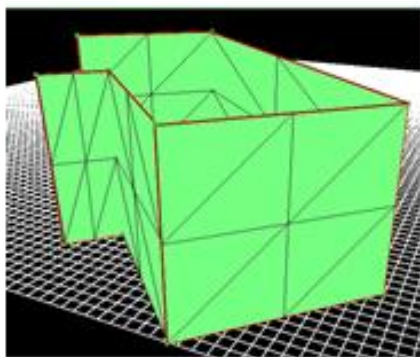


纹理图像的创建与编辑

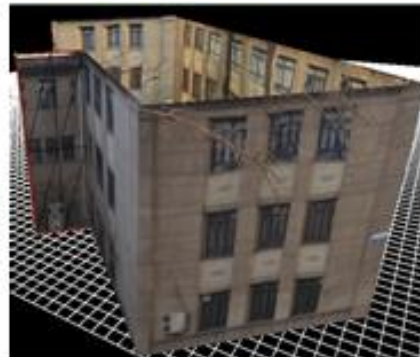
基于图像的模型重建基本流程



三维点云获取



几何结构恢复



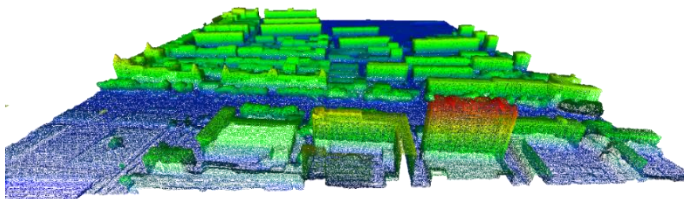
场景绘制

- 三维点云不具有拓扑结构，不能进行高质量的渲染
- 几何结构的恢复重建点云的拓扑结构，是进行场景渲染的基础
- 三维点云的获取和场景绘制都已经有了较为成熟的技术，但是建筑几何结构的恢复仍然没有得到解决

三维数据的获取方式

激光雷达扫描

高效、精度高、成本高

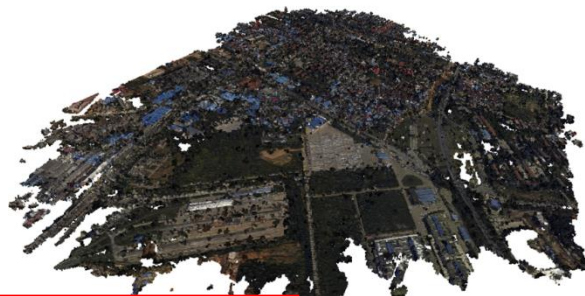
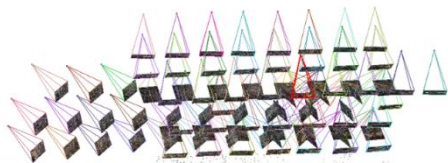


Kinect扫描

便捷、快速、有效范围受限



基于多视角图像的方法



经济、简捷、真实感强

基于图像的模型重建

图像获取便捷

航拍图像



地面图像



互联网搜索



真实感强烈

真三维纹理模型

复现城市的原貌



(a) 真三维城市场景



(b) 二维卫星地图

(c) 全景浏览地图

(d) 三维仿真地图

精度满足渲染需求

比激光雷达扫描
精度低，但能够
满足真实感需求

课程内容

✓ 基于图像的三维模型重建

- ✓ 基本流程
- ✓ 应用场景

✓ 三维模型的表述方式

- ✓ 边界表述方法
- ✓ 空间划分法

✓ 从点云到网格模型

- ✓ 德劳内三角剖分
- ✓ 基于体素的重建方法
- ✓ 基于隐函数的重建方法

✓ 城市建筑的三维模型重建

- ✓ 基于基本元素检测与排列的方法
- ✓ 基于特征保持的隐函数重建方法

三维模型的表述方式

边界表述方法（B-reps）

物体被描述成一组表面，这组表面将物体的内部和外部分离开

- 多面体表述法
- 曲面表述法

空间划分法

将物体内部空间区域划分成细小、不重叠的连续实体（如正方体）来描述物体形状

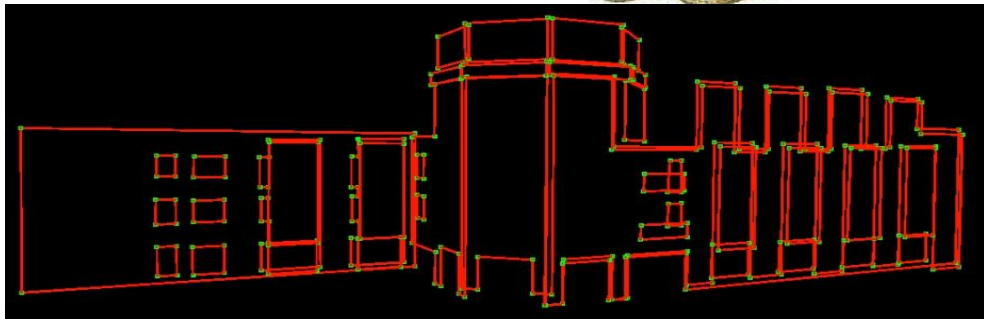
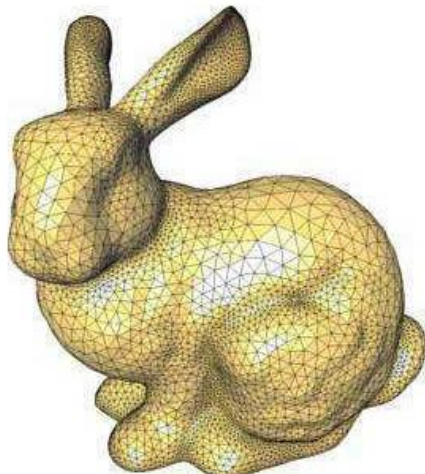
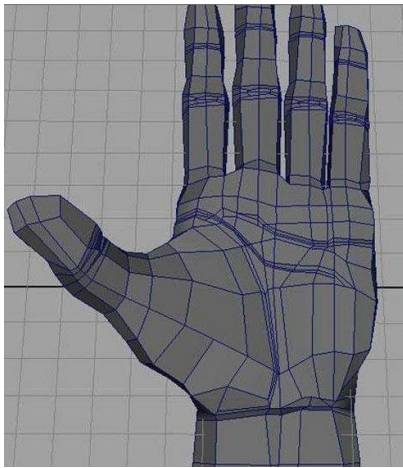
- 构造体素法
- 八叉树法
- 二分空间法

边界表述方法 (B-reps)

多面体表述法

将物体表面表述成一组封闭的多边形，最常用的是三角形

- 稳定性高
- 能够恢复模型表面细节
- 要求点云稠密分布均匀

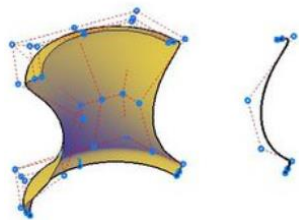


边界表述方法 (B-reps)

曲面表述法 将物体的表面表述成一组参数或者非参数化的曲面
精确性高，方便操控和变形

参数平面 $z = f(x, y)$

- ✓ B-spline 曲面
- ✓ NURBS 曲面



- ✓ 球面，椭球面

隐式平面 $\{(x, y, z) | f(x, y, z) = 0\}$

- ✓ 符号距离函数(Signed Distance Function)

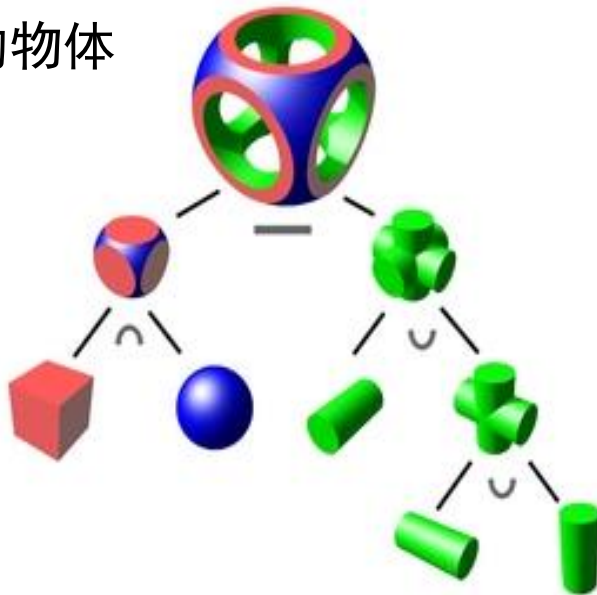
$$f(x) = \begin{cases} d(x, \partial\Omega) & \text{if } x \in \Omega \\ -d(x, \partial\Omega) & \text{if } x \in \Omega^c \end{cases}$$

- ✓ 指示函数(Indicator Function)

$$1_A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases}$$

构造体素法 (Constructive Solid-Geometry)

该方法通过对一些基本元素(圆柱、圆锥或者球体)等进行加、减、并集和交集等运算组合成新的物体



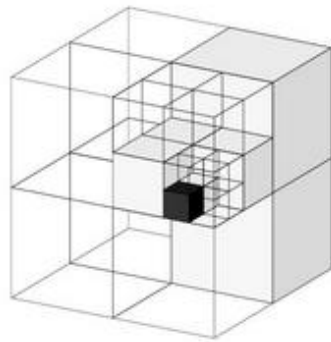
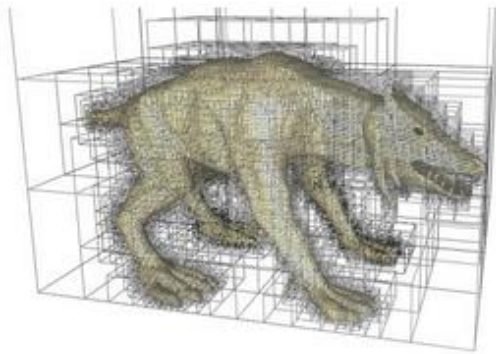
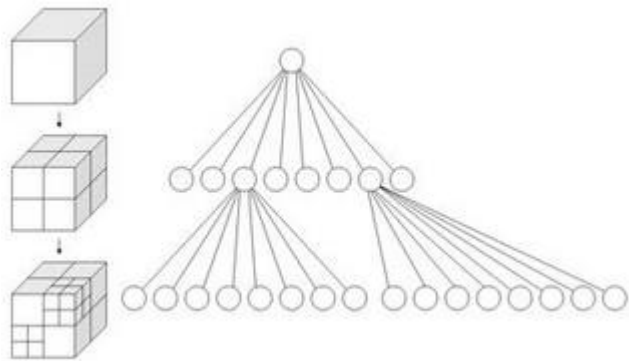
操作简单，便于实现

只能表述结构较为简单的实体

空间划分法

八叉树法(Octree)

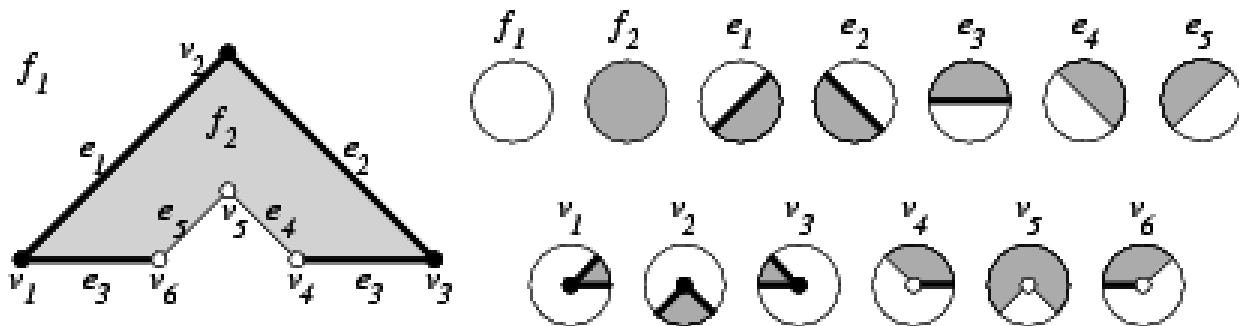
从根节点开始，将包含物体的节点进行均匀划分，知道满足条件



可以方便的进行节点删除或者合并操作，表述精度受到空间单位分辨率的影响

二分空间法(Binary Space-partitioning)

与八叉树类似，对空间进行逐步划分，不同的是每次划分成两部分，且划分的位置和方向根据空间布局随时调整



表述精度受到空间单位分辨率的影响

课程内容

✓ 基于图像的三维模型重建

- ✓ 基本流程
- ✓ 应用场景

✓ 三维模型的表述方式

- ✓ 边界表述方法
- ✓ 空间划分法

✓ 从点云到网格模型

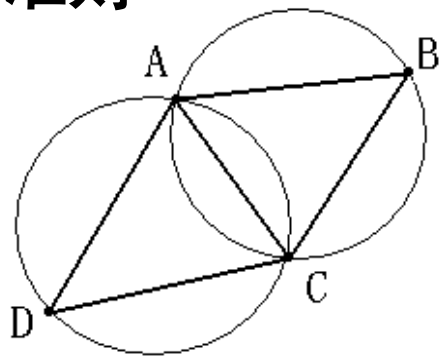
- ✓ 德劳内三角剖分
- ✓ 基于体素的重建方法
- ✓ 基于隐函数的重建方法

✓ 城市建筑的三维模型重建

- ✓ 基于基本元素检测与排列的方法
- ✓ 基于特征保持的隐函数重建方法

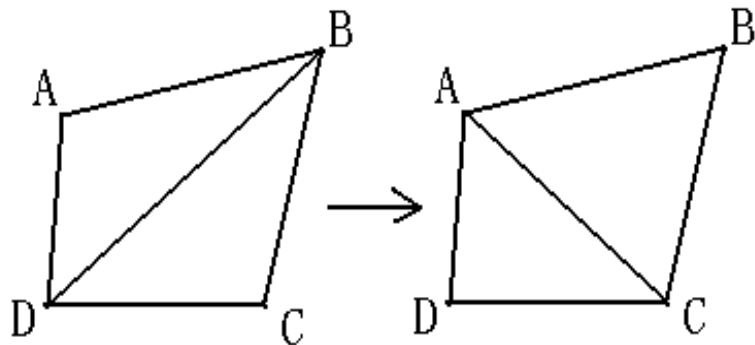
德劳内三角剖分

德劳内准则



空圆特性

在Delaunay三角形网中，任意三角形外接圆不包含其它点



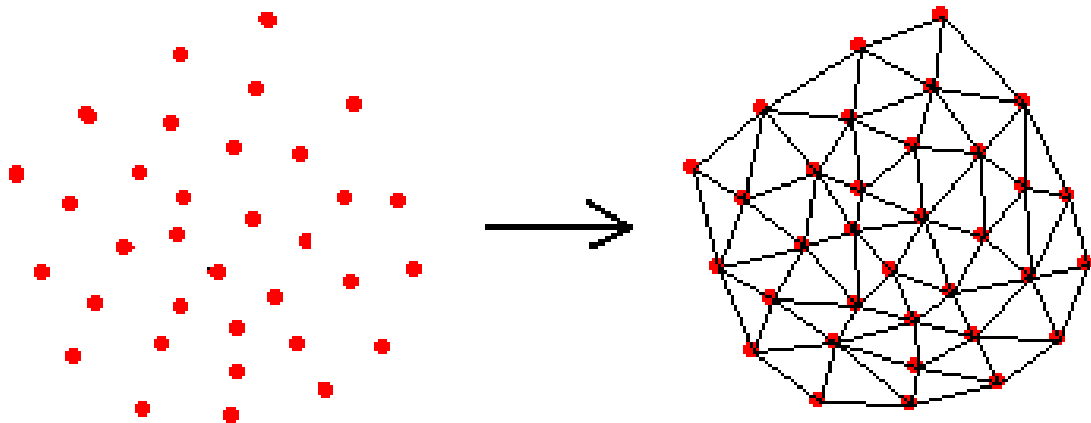
最大化最小角

在散点集可能形成的三角剖分中，Delaunay三角剖分所形成的三角形的最小角最大

德劳内三角剖分

二维点的三角剖分

唯一性，最优性，区域性



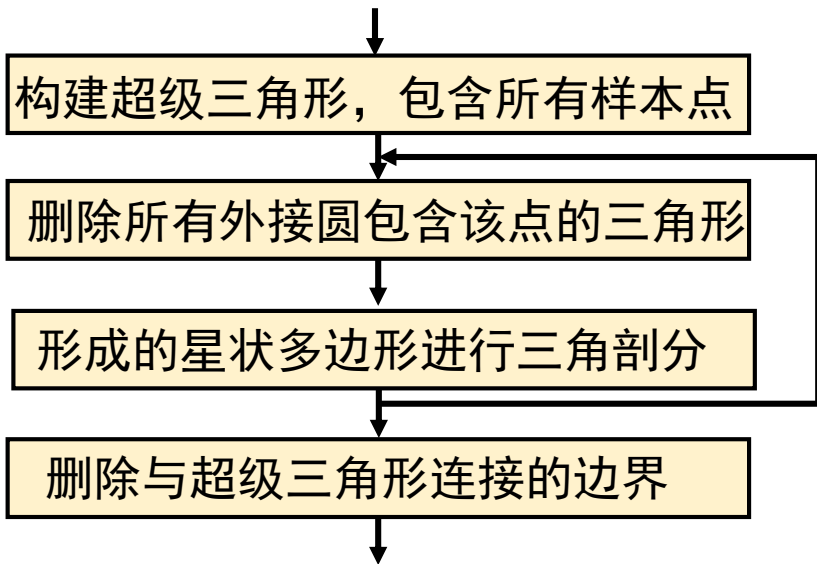
Lawson算法

Bowyer-Watson算法(推荐)

德劳内三角剖分

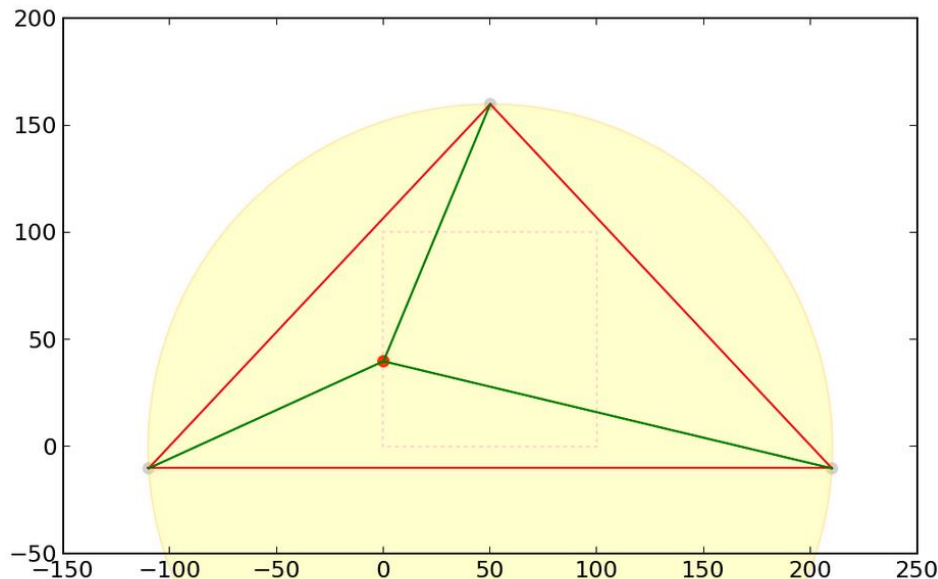
Bowyer-Watson算法(推荐)

增量的算法，每次插入一个点时，删除不满足德劳内准则的三角形，重新进行三角剖分



德劳内三角剖分

Bowyer-Watson算法(推荐)



1. 超级三角形中插入一个顶点

构建超级三角形，包含所有样本点

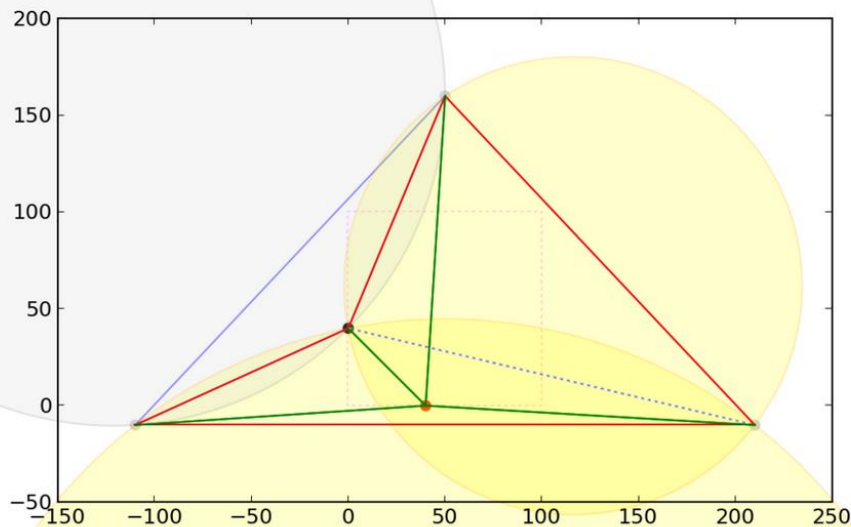
删除所有外接圆包含该点的三角形

形成的星状多边形进行三角剖分

删除与超级三角形连接的边界

德劳内三角剖分

Bowyer-Watson算法(推荐)



2. 插入第二个顶点

构建超级三角形，包含所有样本点

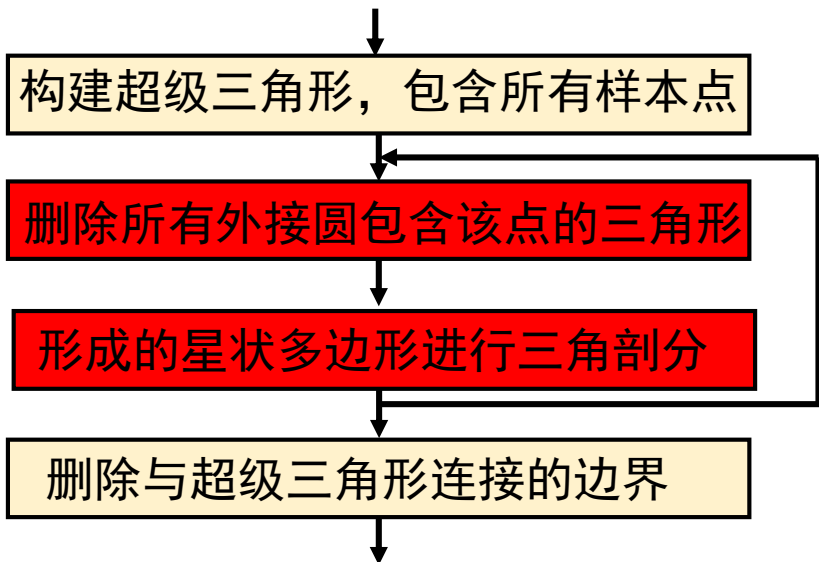
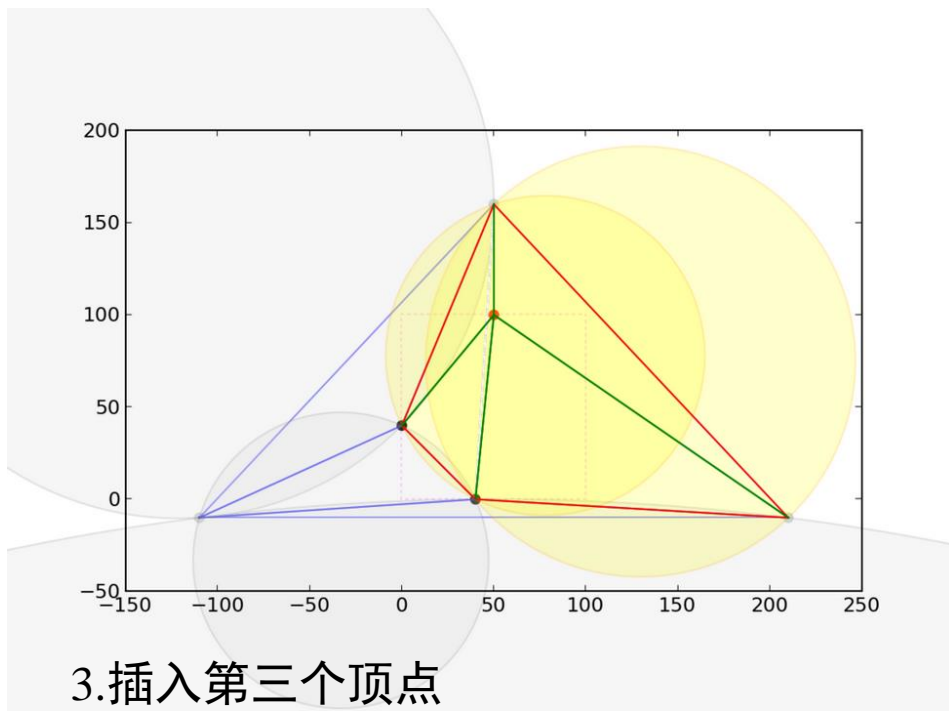
删除所有外接圆包含该点的三角形

形成的星状多边形进行三角剖分

删除与超级三角形连接的边界

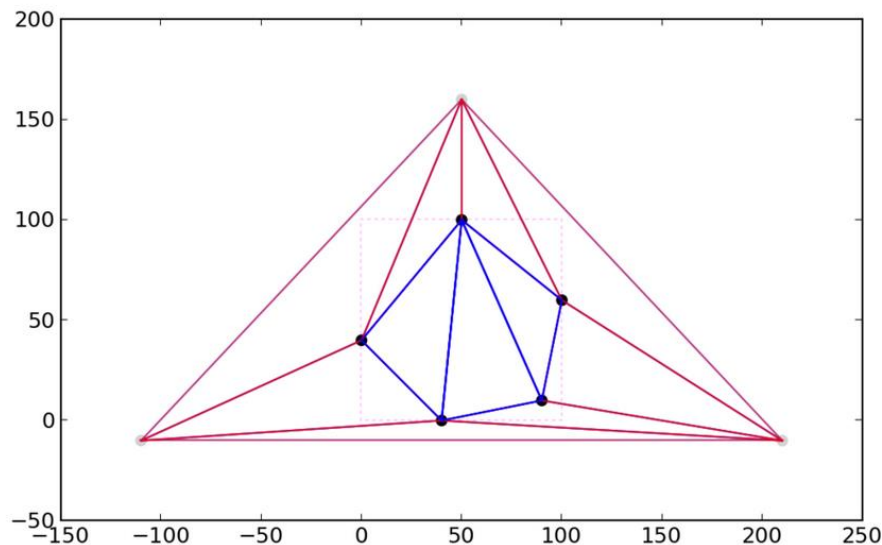
德劳内三角剖分

Bowyer-Watson算法(推荐)



德劳内三角剖分

Bowyer-Watson算法(推荐)



4.插入5个顶点之后

构建超级三角形，包含所有样本点

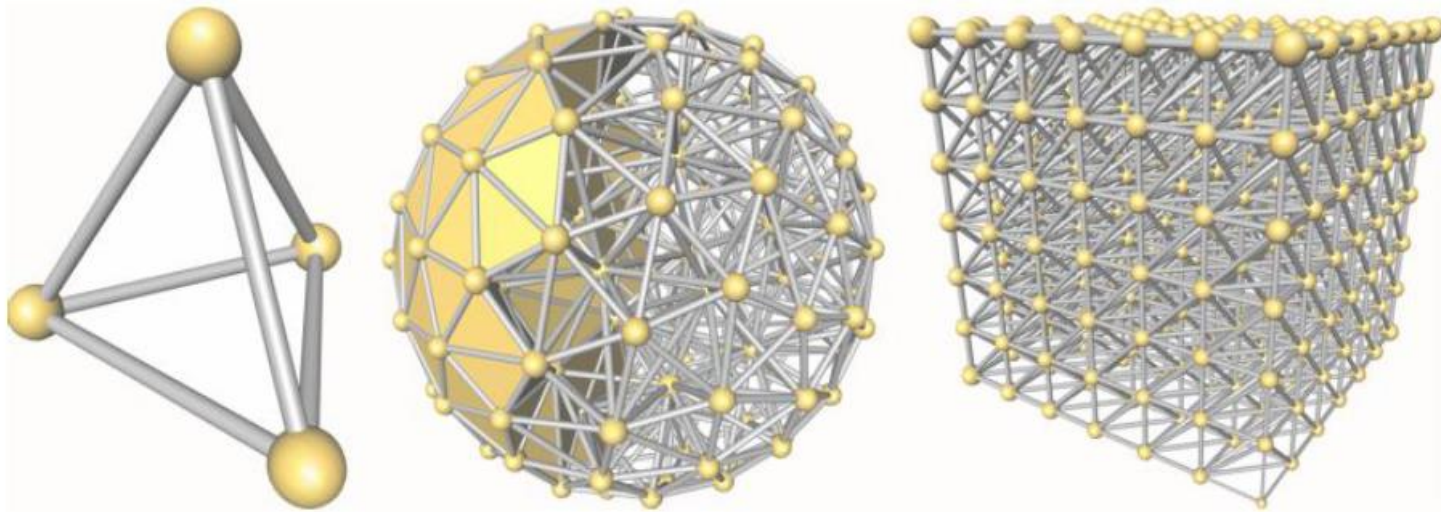
删除所有外接圆包含该点的三角形

形成的星状多边形进行三角剖分

删除与超级三角形连接的边界

德劳内三角剖分

三维正四面体剖分



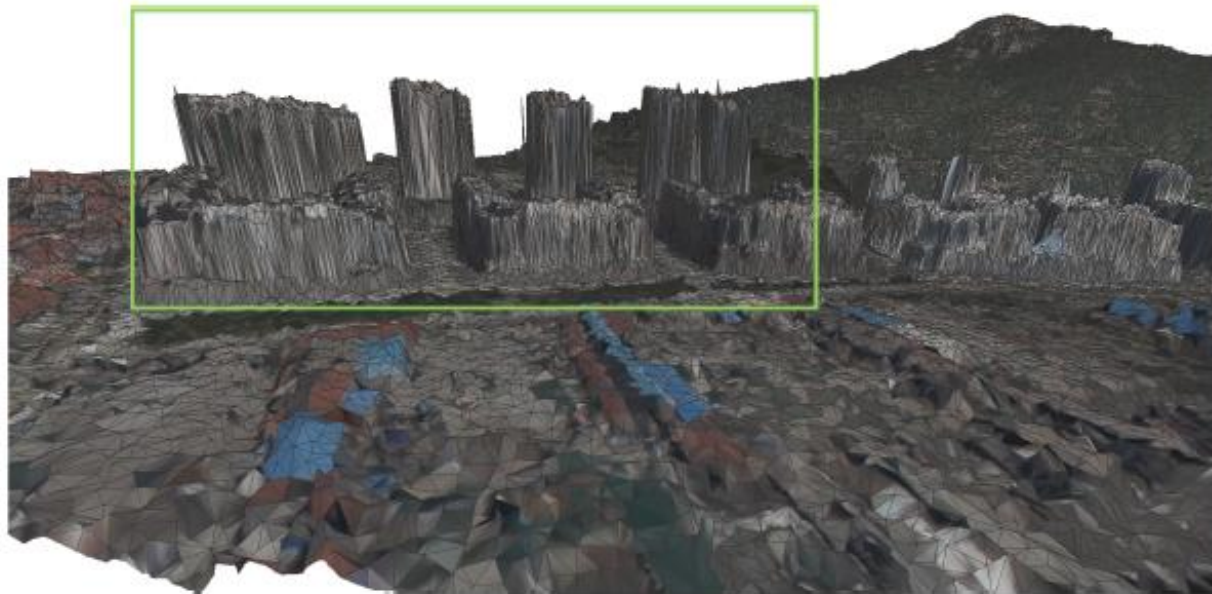
算法简单，容易实现，对噪声敏感，不能处理空洞

德劳内三角剖分

2.5D 三角剖分

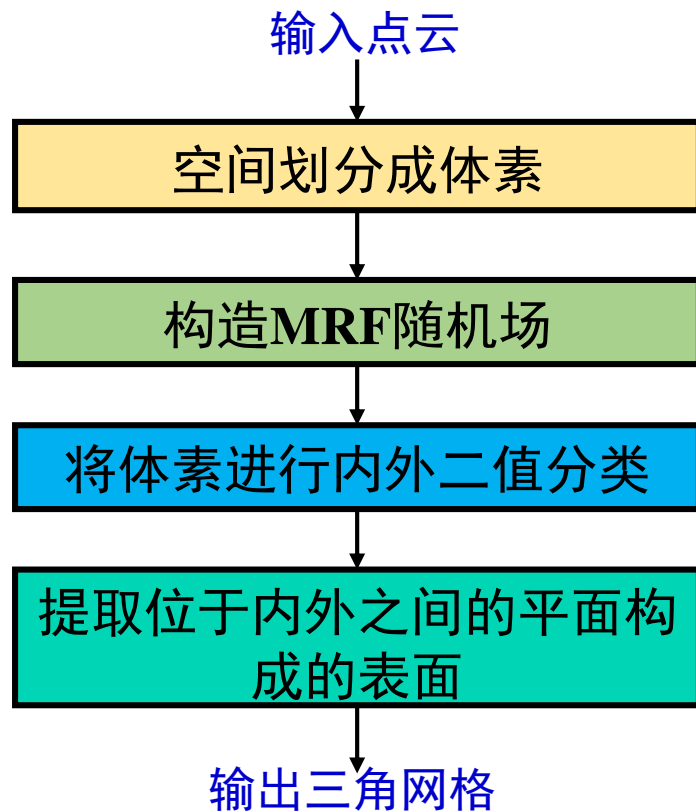
将三维点投影到平面上，进行德劳内三角剖分

不能保证德劳内特性



基于体素的重建方法

重建流程

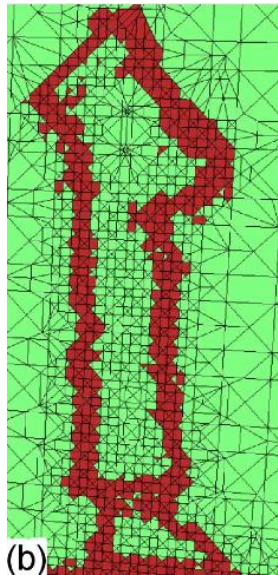


基于体素的重建方法

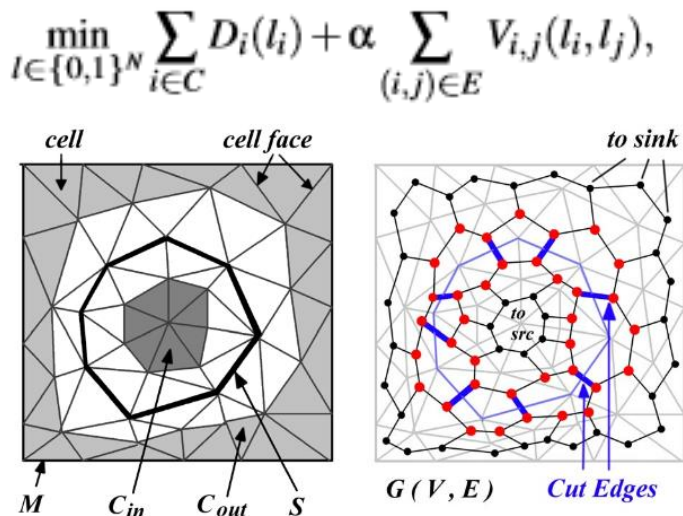
自适应四面体



输入图像



点云正四面体化



构造MRF随机场



生成模型

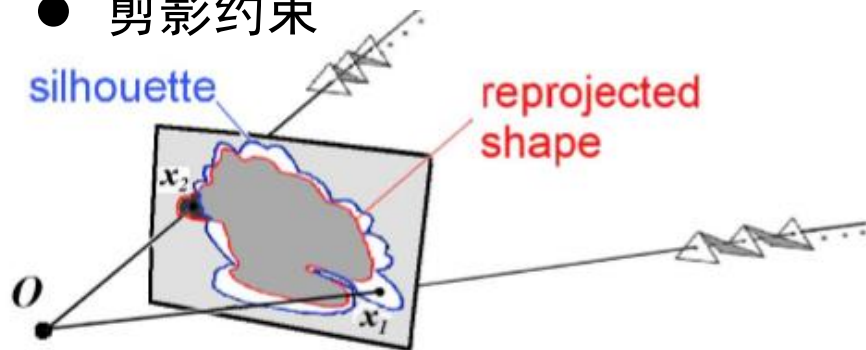
基于体素的重建方法

约束条件

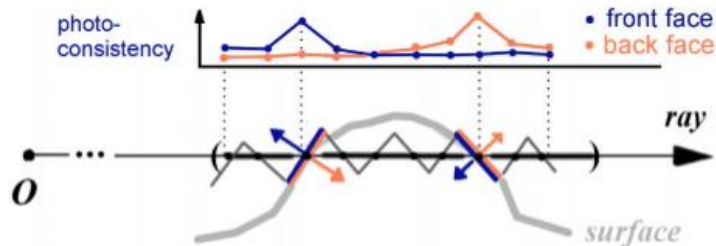
- 光度一致性假设



- 剪影约束

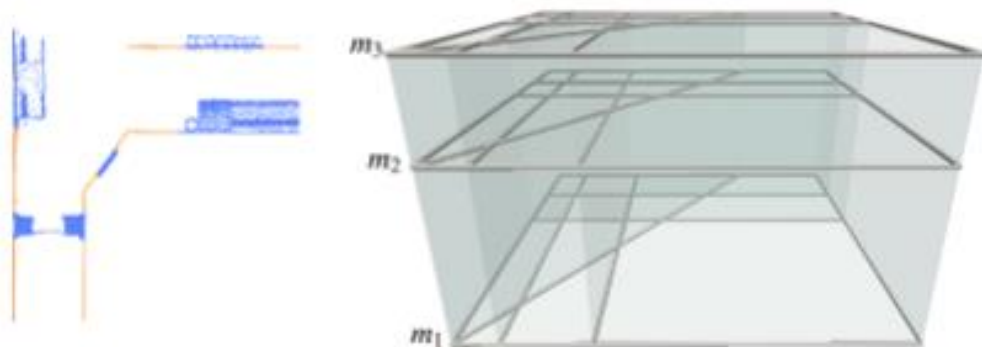


- 可视性约束



基于体素的重建方法

规则建筑重建



点云进行多面体重建

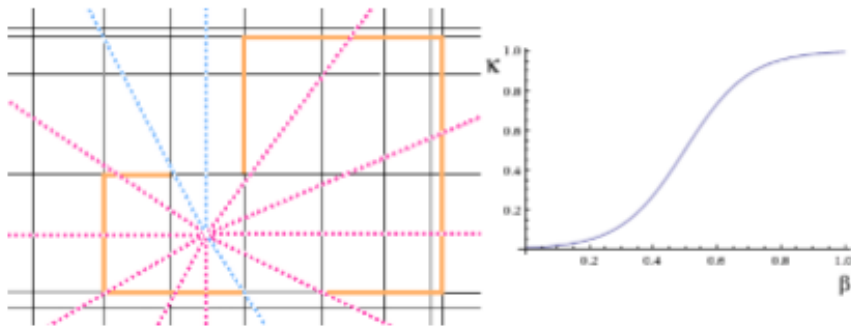
$$\min_{l \in \{0,1\}^N} \sum_{i \in C} D_i(l_i) + \alpha \sum_{(i,j) \in E} V_{i,j}(l_i, l_j),$$

构建MRF随机场

基于体素的重建方法

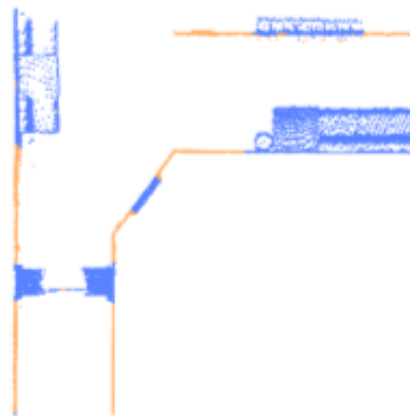
- 数据项-通过射线穿越次数判断位于内部还是外部

$$D_i(l_i) = \begin{cases} (1 - \beta) \cdot \rho, & l_i = 0 \\ \beta \cdot \rho, & l_i = 1 \end{cases}$$

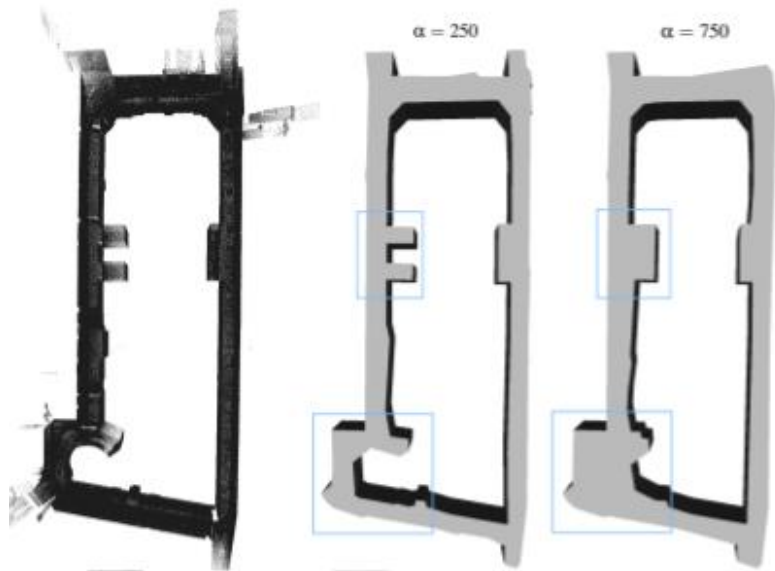
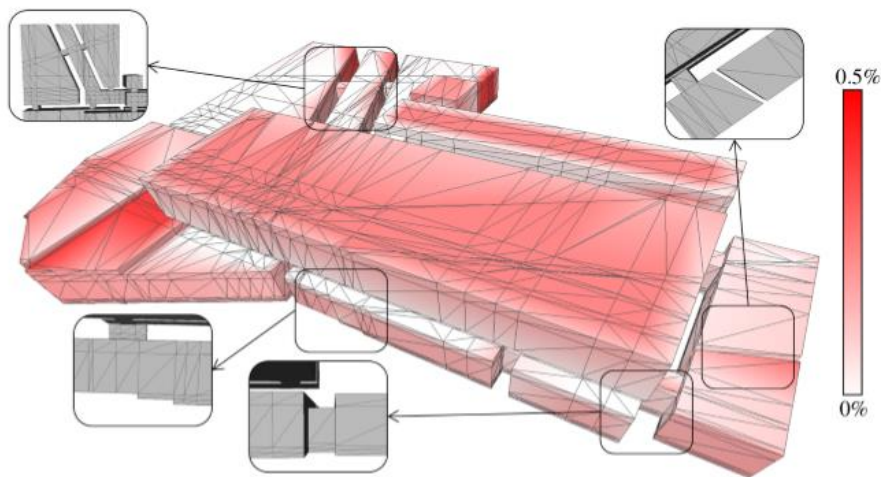


- 平滑项-两个多面体的墙面上点云覆盖面积

$$V_{i,j}(l_i, l_j) = \begin{cases} 0, & l_i = l_j \\ (1 - \omega_{i,j}) \cdot \gamma, & l_i \neq l_j \end{cases}$$

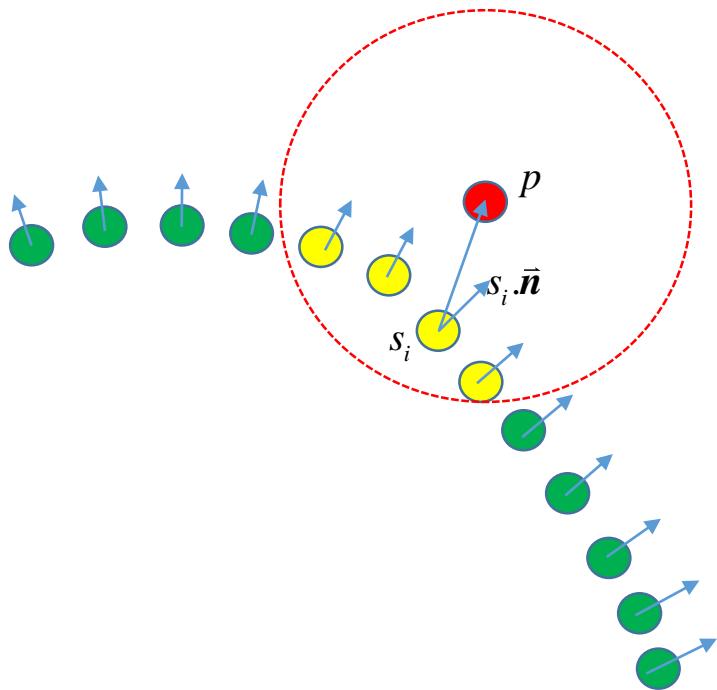


基于体素的重建方法



基于隐函数的重建方法

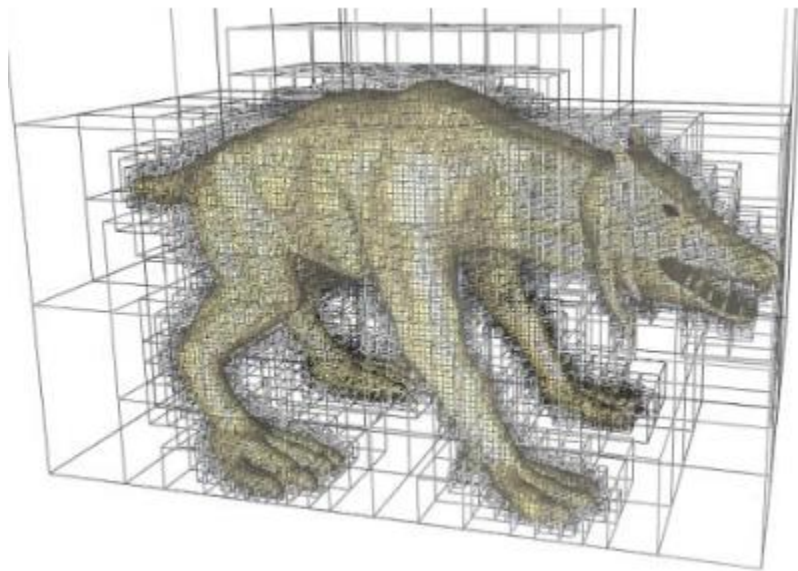
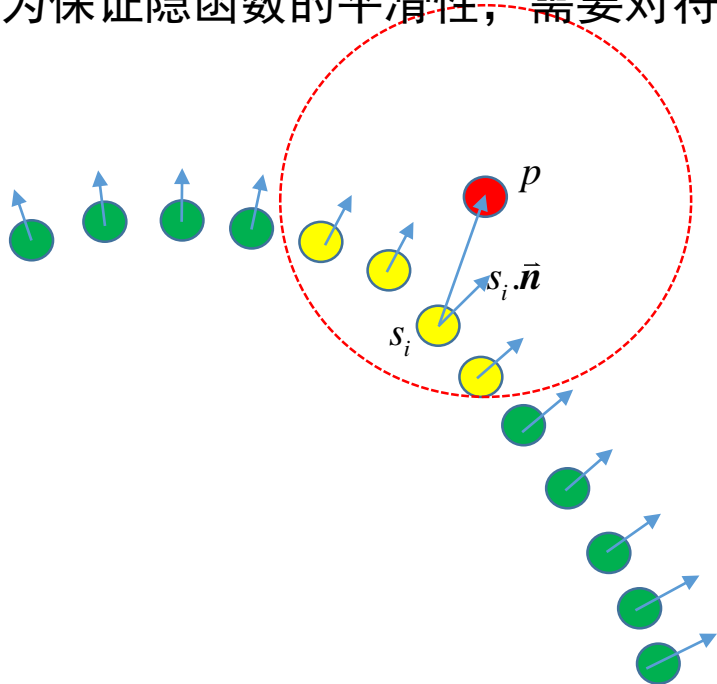
局部隐函数重建-符号距离场(Signed Distance Field)



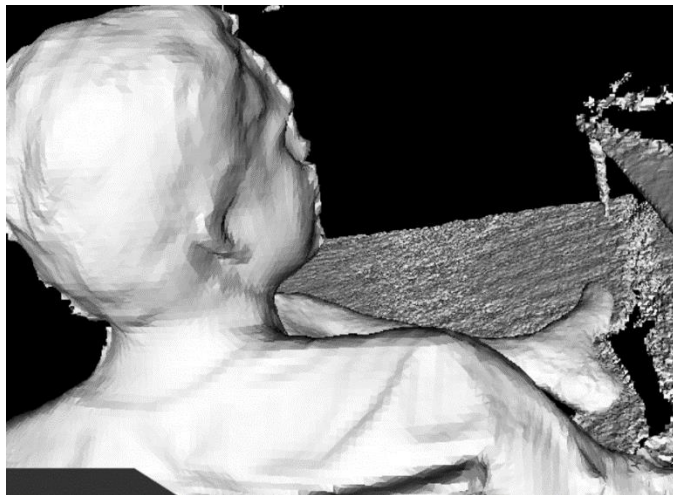
$$sdf(p) = \frac{\sum_i w_i (s_i \cdot \mathbf{c} - p \cdot \mathbf{c}) \bullet s_i \bar{\mathbf{n}}}{\sum_i w_i}$$

基于隐函数的重建方法

- 八叉树的每一个节点上计算符号距离
- 为保证隐函数的平滑性，需要对符号距离进行混合



基于隐函数的重建方法




算法原理简单，计算速度快，能够反映物体细节
对点云质量要求高，无法处理空洞，容易受到噪声影响


基于隐函数的重建方法


全局隐函数重建


将隐函数表示成建立在以输入点为中心的径向基函数(RBF)之和

$$f(x) = \sum_i \alpha_i B_i(x, p_i, c)$$


$$B_1(x, p_1, c) = B_1(\|x - p_1, c\|)$$

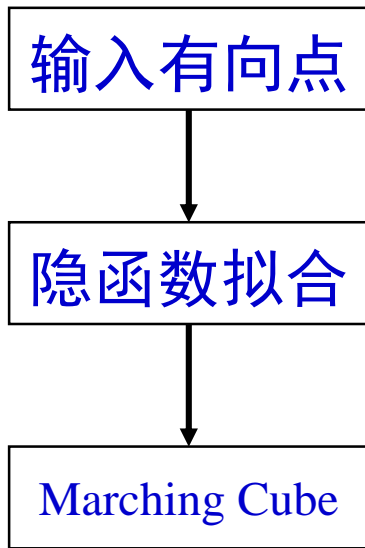

$$B_2(x, p_2, c) = B_2(\|x - p_2, c\|)$$


$$B_4(x, p_4, c) = B_4(\|x - p_4, c\|)$$

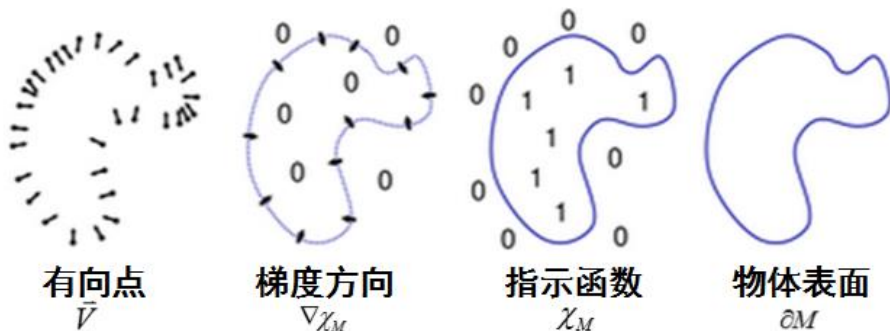

$$B_3(x, p_3, c) = B_3(\|x - p_3, c\|)$$

泊松表面重建算法

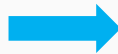
算法流程



算法原理



$$\nabla \chi_M = \vec{V}$$

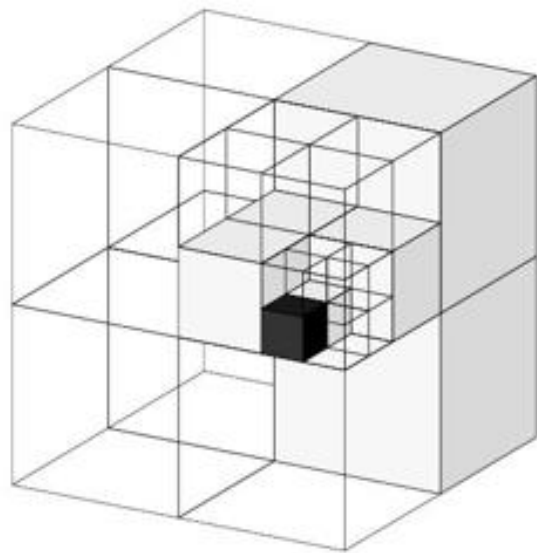


$$\Delta \chi_M = \nabla \cdot \vec{V}$$

泊松方程

泊松表面重建算法

指示函数 $\chi(p)$



八叉树上有 $|O|$ 个有效节点

每个节点上 o_i 建立基函数 F_i

$$\chi(p) = \sum_{i=1}^{|O|} x_i F_i(p) \quad \{x_1, \dots, x_{|O|}\}$$

$$\|Lx - v\|^2$$

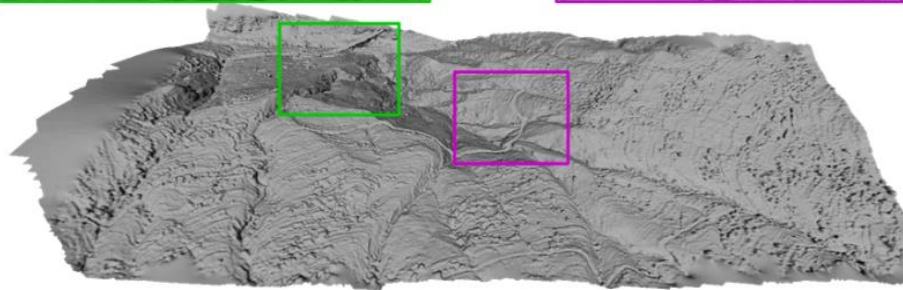
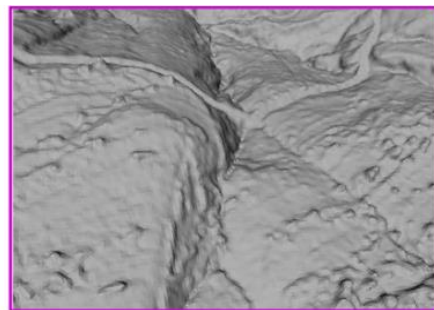
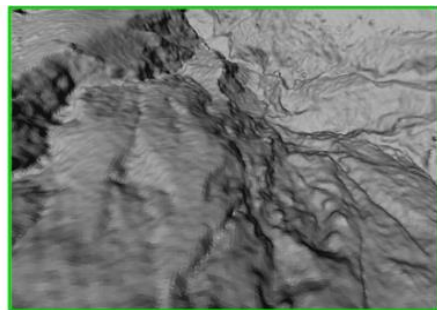
$$L \in R^{|O| \times |O|} \quad L_{o,o'} = \langle \Delta F_o, \Delta F_{o'} \rangle \quad v_o = \langle \nabla V_o, F_o \rangle \quad x = [x_1, \dots, x_{|O|}]^T$$

引导梯度场 $\vec{V}(q)$ 的构建

$$\begin{aligned}\vec{V}(q) &\equiv \sum_{s \in S} \sum_{o \in \text{Ngbr}_D(s)} \alpha_{o,s} F_o(q) s \cdot \vec{N} \\ &\equiv \sum_o \sum_{s \in \text{Ngbr}(o)} \alpha_{o,s} F_o(q) s \cdot \vec{N} \\ &\equiv \sum_o F_o(q) \sum_{s \in \text{Ngbr}(o)} \alpha_{o,s} s \cdot \vec{N} \\ &\equiv \sum_o F_o(q) \vec{N}_o\end{aligned}$$

泊松表面重建算法

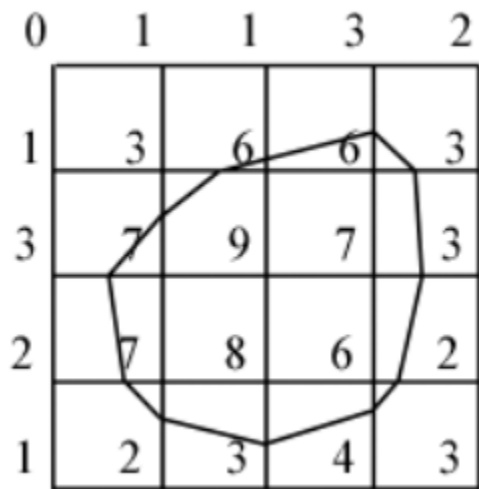
算法鲁棒，能够处理噪声和空洞
计算量大，对局部细节容易造成过平滑



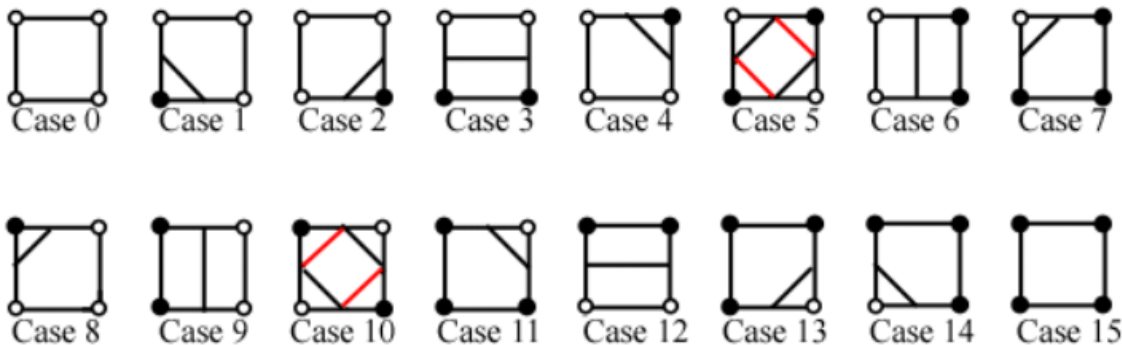
基于隐函数的重建方法

Marching Square

将隐函数或者符号距离长转化成分段线段



Iso val=5时，提取等值线



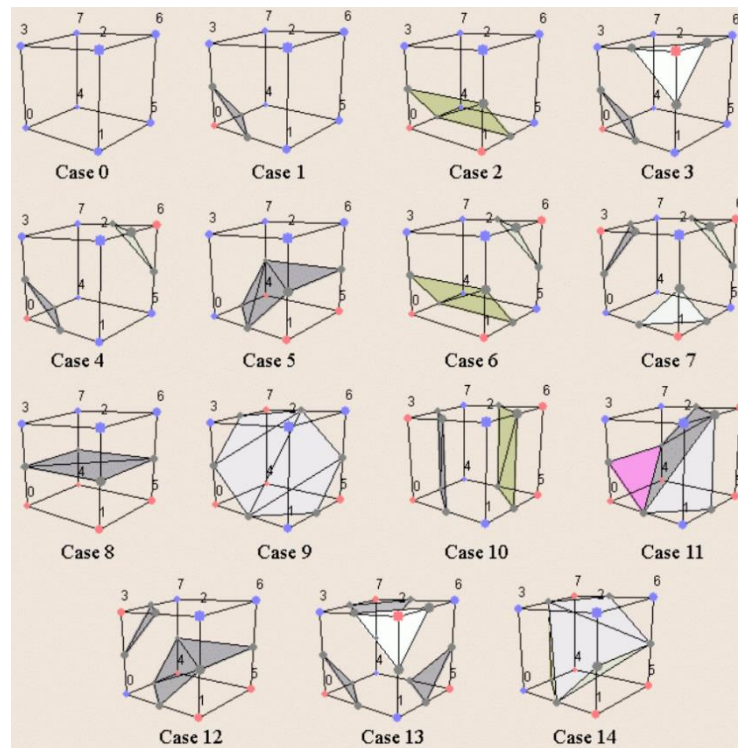
16种不同的取值情况

基于隐函数的重建方法

Marching Cube

将隐函数或者符号距离场转化成三角面片

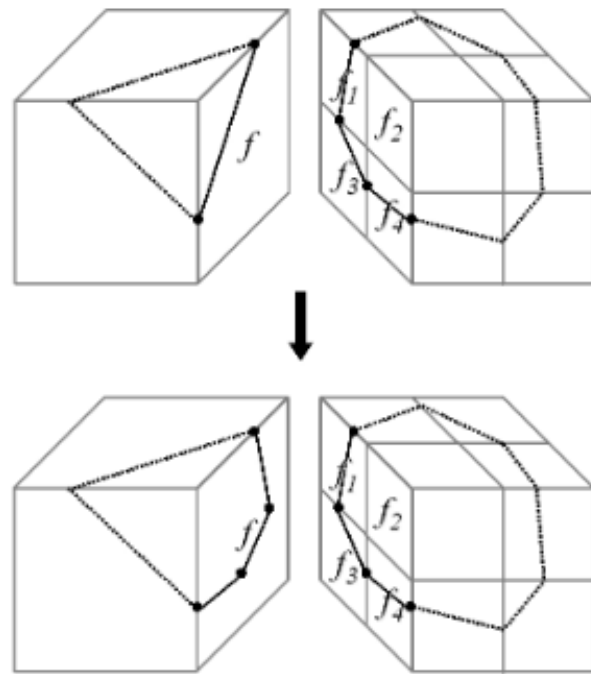
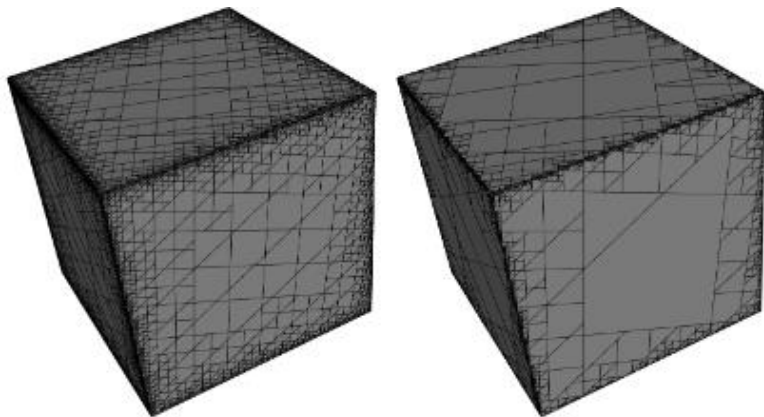
一共有256种模式，经过适当的旋转，很多状态是一致的，这样可以归纳出15种状态



基于隐函数的重建方法

Octree Based Marching Cube

在八叉树上进行Marching Cube,解决了不同分辨率之间Cell生成面片一致性的问题



课程内容

✓ 基于图像的三维模型重建

- ✓ 基本流程
- ✓ 应用场景

✓ 三维模型的表述方式

- ✓ 边界表述方法
- ✓ 空间划分法

✓ 从点云到网格模型

- ✓ 德劳内三角剖分
- ✓ 基于体素的重建方法
- ✓ 基于隐函数的重建方法

✓ 城市建筑的三维模型重建

- ✓ 基于基本元素检测与排列的方法
- ✓ 基于特征保持的隐函数重建方法

城市建筑的三维模型重建

城市建筑的特征

含有大量规则的基本元素

四面体



平面元素



重复结构



城市建筑类型多样结构复杂

曼哈顿结构



分片平面

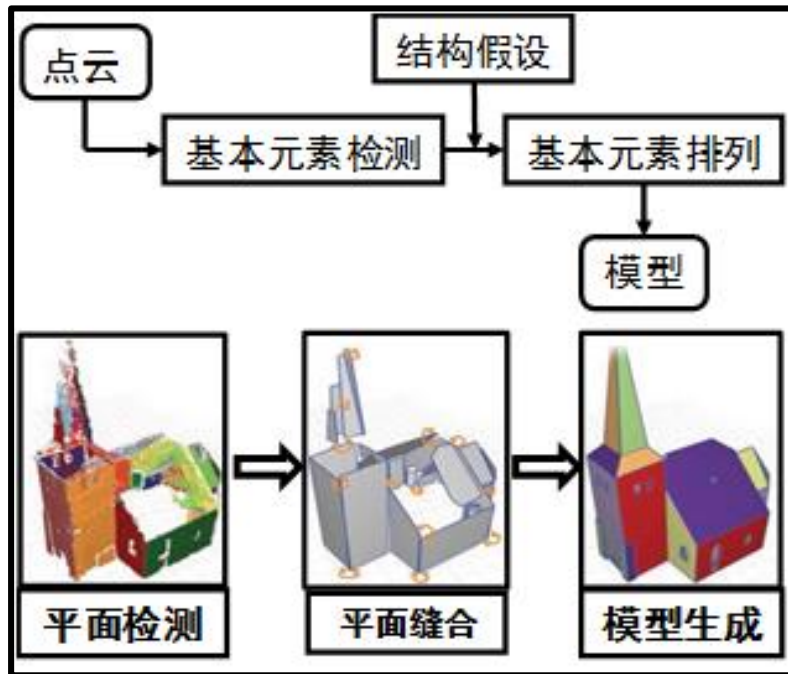


更加复杂的结构



城市建筑的三维模型重建

基于基本元素检测与排列的方法



优点:

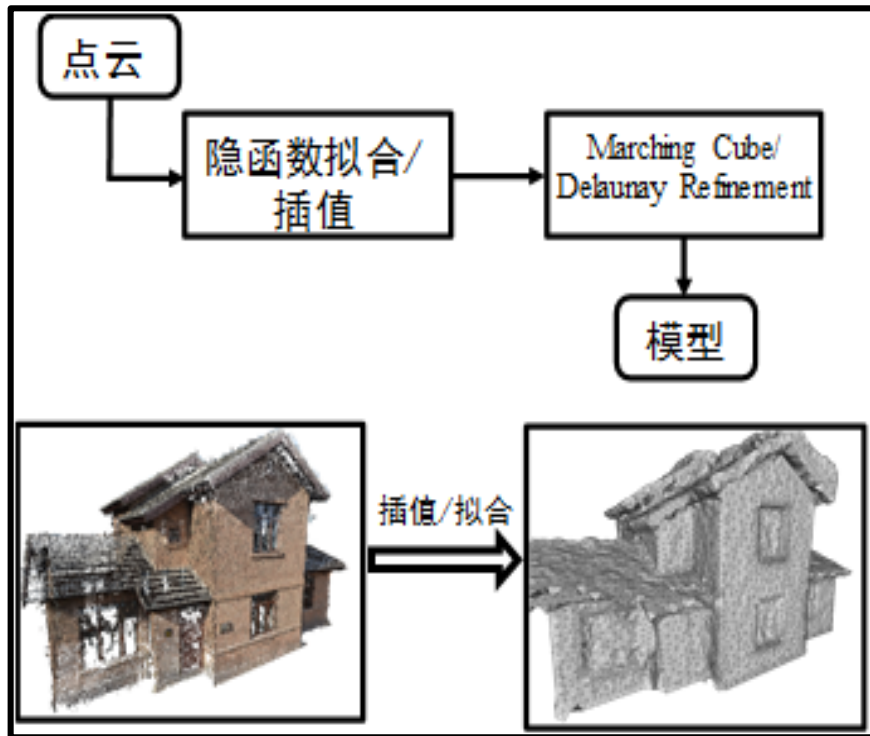
- 保持建筑尖锐结构
- 模型简洁
- 效果逼真

缺陷:

- 基本元素检测困难
- 空间关系难以表述

城市建筑的三维模型重建

基于表面重建的方法



优点:

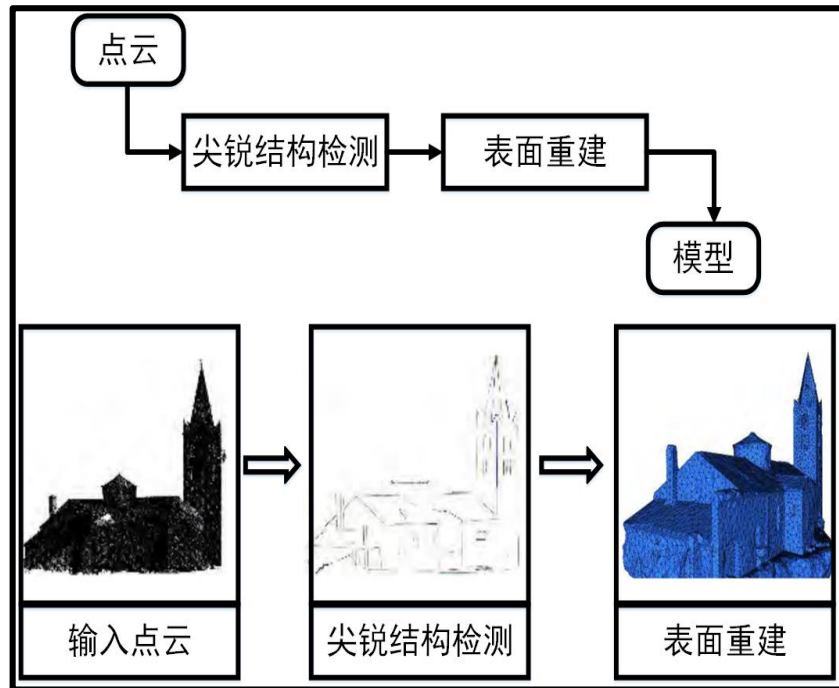
- 全自动
- 适用各种类型的数据

缺陷:

- 无法保持尖锐特征
- 生成模型冗余
- 重建表面受噪声影响

城市建筑的三维模型重建

基于特征保持的表面重建方法



优点:

- 全自动
- 适用于各种数据
- 一定程度保持尖锐特征

缺陷:

- 依赖于尖锐结构检测
- 存在冗余
- 只能保持部分尖锐结构

城市建筑的三维模型重建

定义一系列基本形状，采用特定语法规则对这些基本形状进行排布，以生成复杂的建筑物三维模型



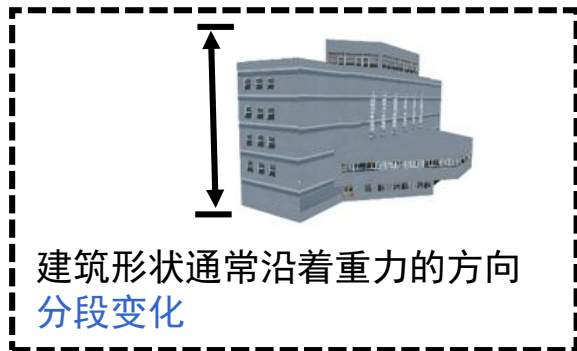
十分快速高效

关键在于语法规则的获取

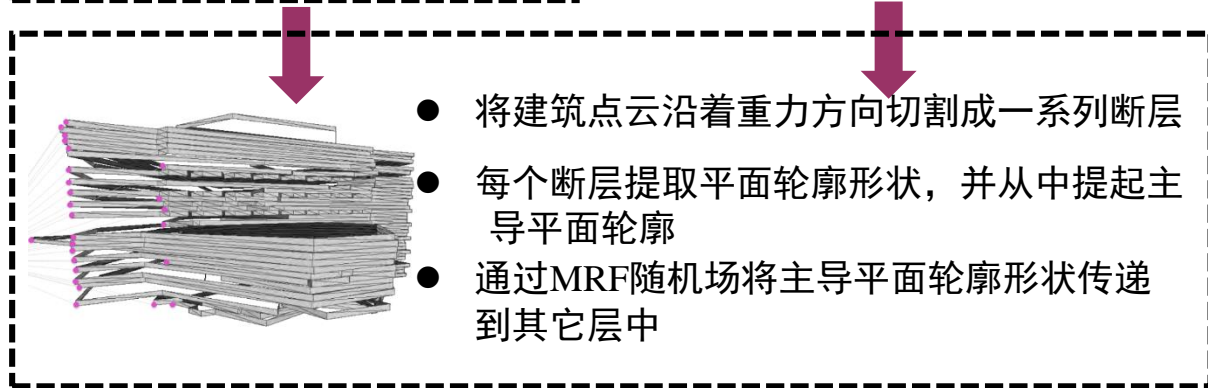
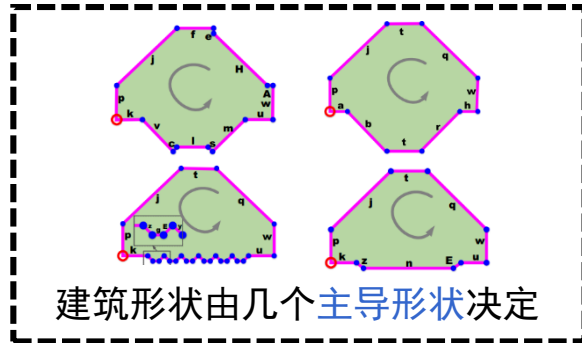
城市建筑的三维模型重建

基于断层的建筑模型重建

观察一

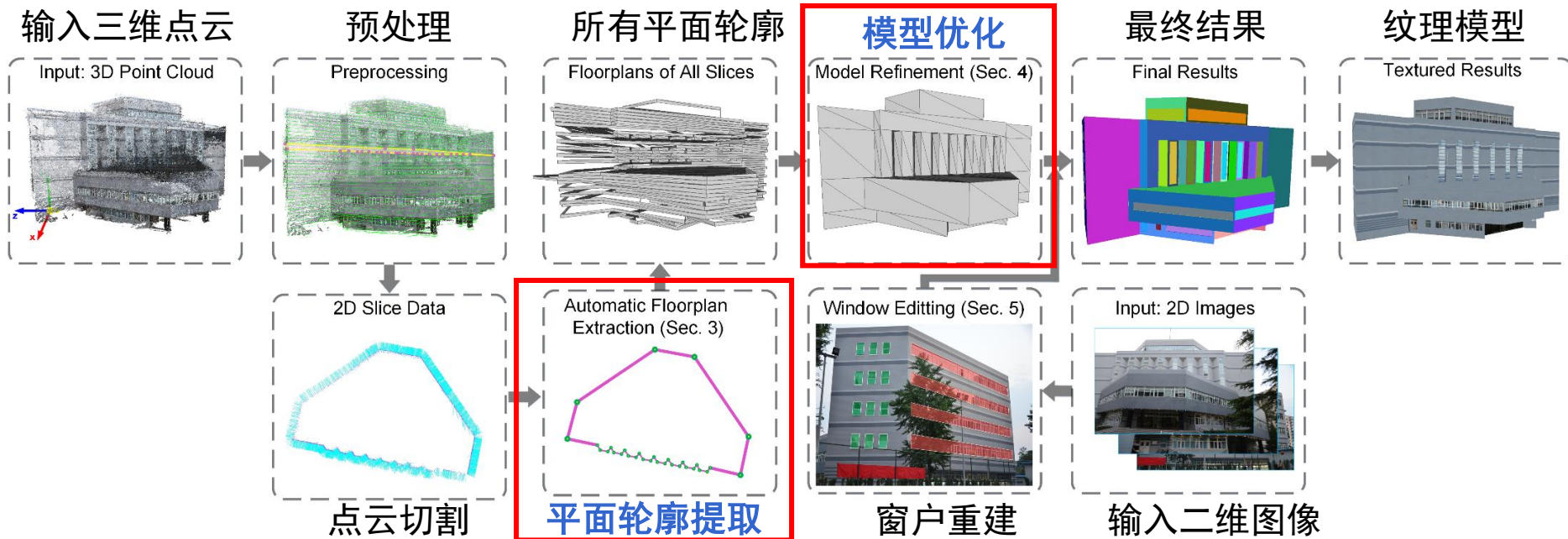


观察二



城市建筑的三维模型重建

算法框架



城市建筑的三维模型重建

基于图像的交互式窗户重建

简单的结构假设

窗户是三维的立方体

窗户重建流程

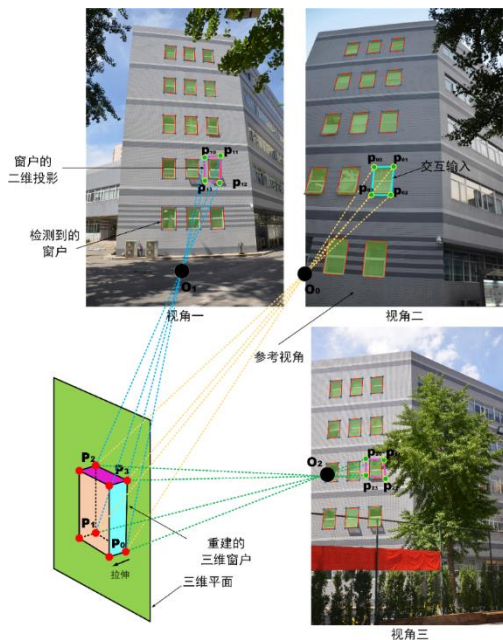
输入三维平面

⊕
窗户模板

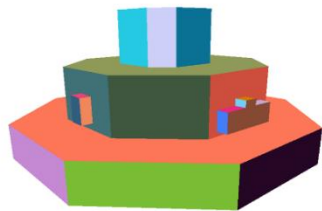
2D重复结构检测

多视角约束

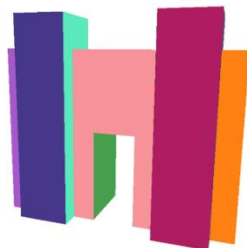
3D窗户



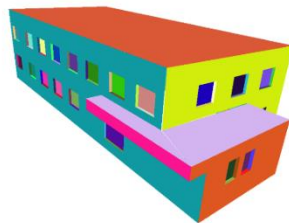
城市建筑的三维模型重建



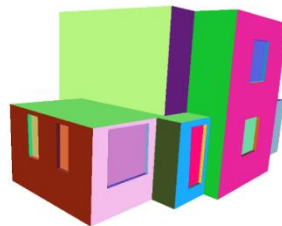
(a) Algubbat



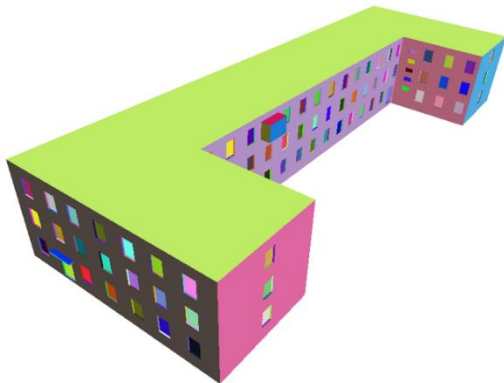
(b) Babylon



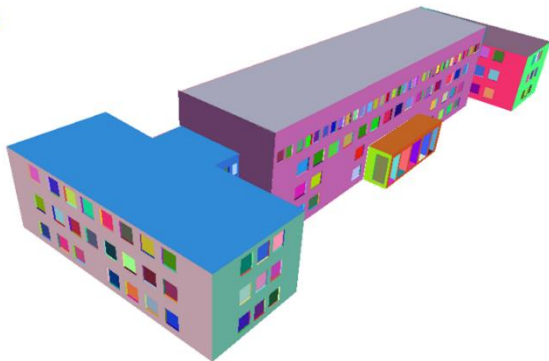
(d) House3



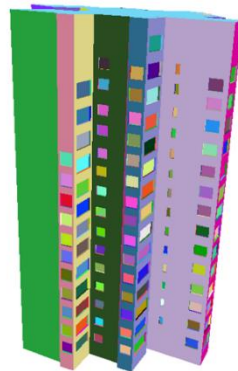
(c) House4



(d) House1

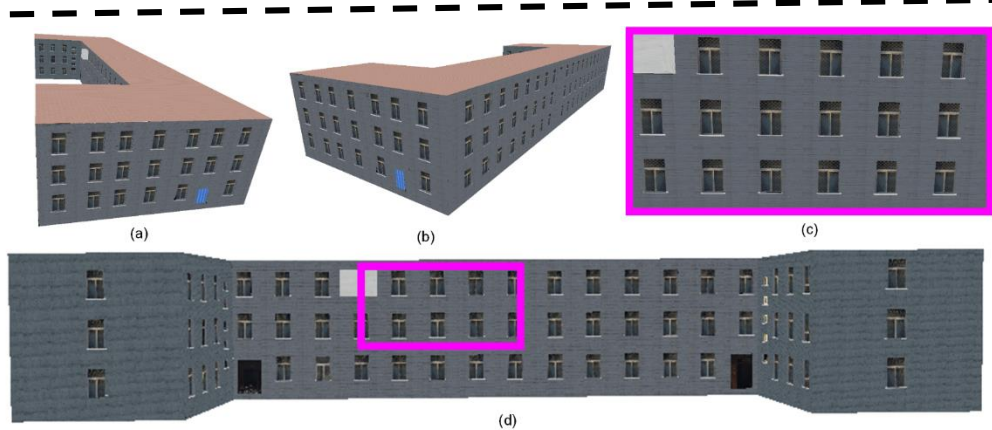
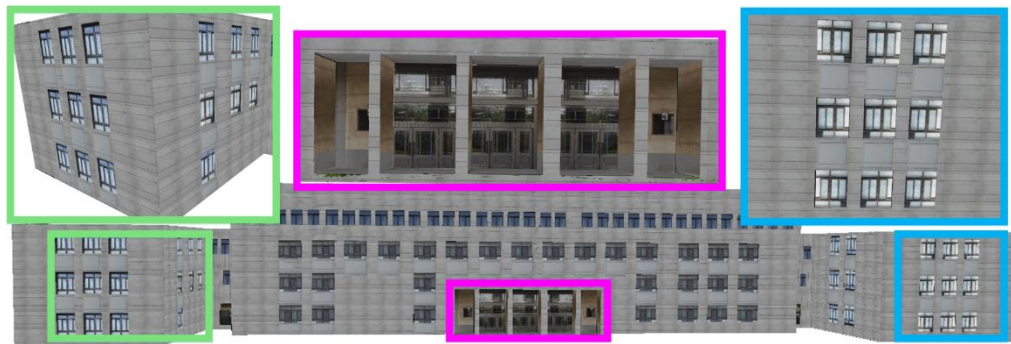
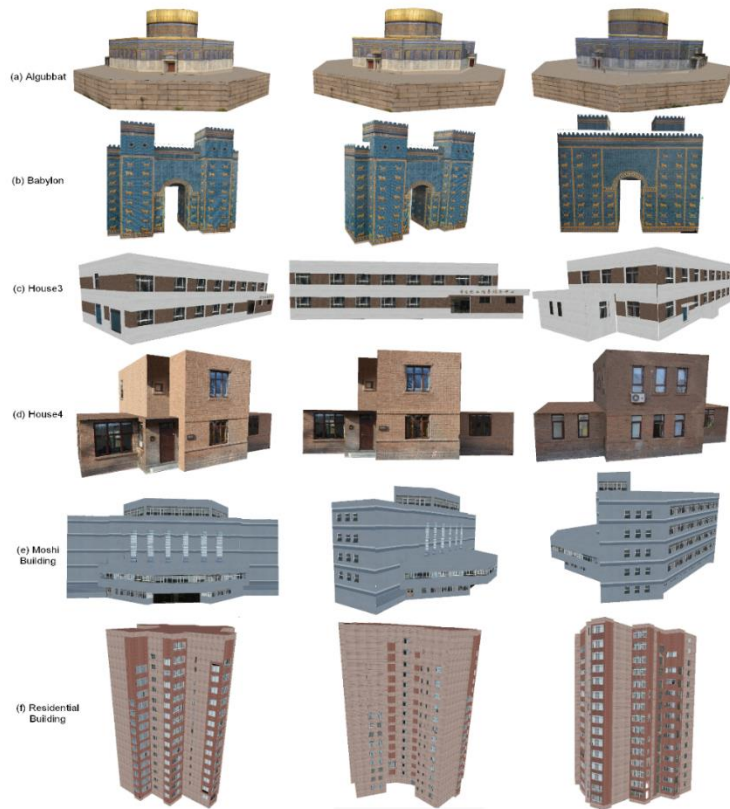


(e) House2



(f) Residential building

城市建筑的三维模型重建





感谢各位聆听 !
Thanks for Listening ●