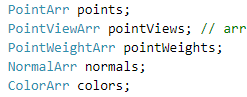
# OpenMVS算法笔记文档

## 稠密点云生成

### 1.1稠密点云生成点云包含的数据结构



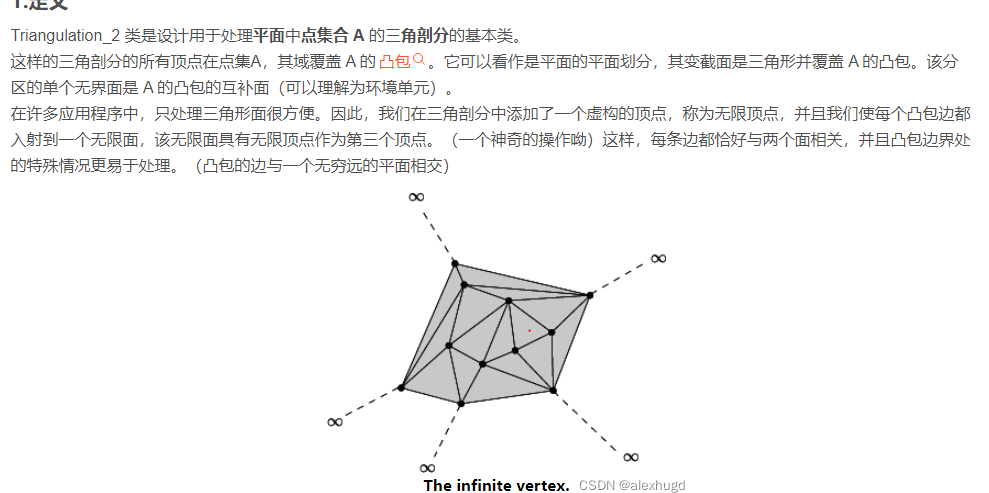
### 1.2

## 生成mesh

生成mesh的任务对应的输入是上一节输出的稠密点云，需要根据这些稠密点云生成mesh，即三角网格。

对应的输入命令：ReconstructMesh scene\_dense.mvs -p scene\_dense.ply

### 2.1 cgal中的infinite顶点

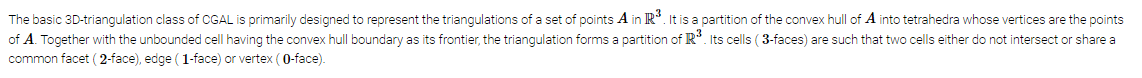


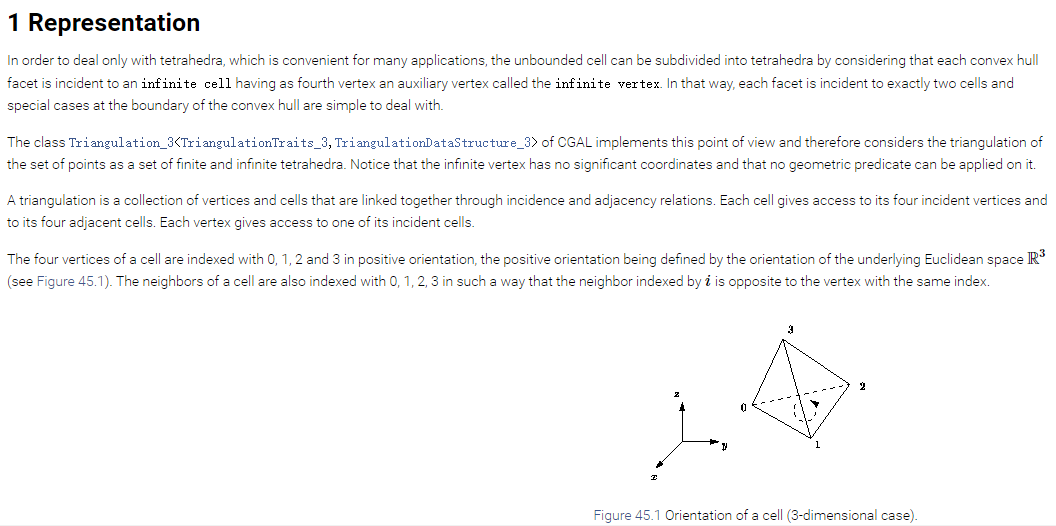
### 2.2 orientation函数

作用：根据输入的三个顶点所形成的面片信息，判断输入的点p是否在这平面上，在这个平面外部，还是在平面内部。输入的点是带有顺序的。

### 2.3 delaunay.number\_of\_cells

参考文献：<https://doc.cgal.org/latest/Triangulation_3/index.html#chapterTriangulation3>





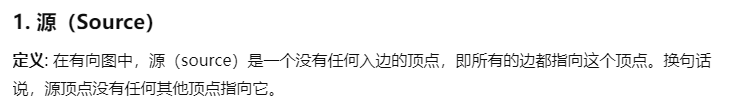
### 2.4 IsDepthSimilar

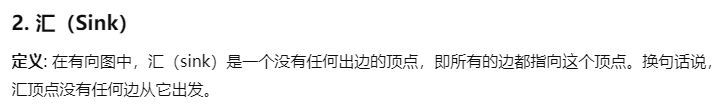
判断两个点在相机坐标系下的深度是否接近，作者使用了如下的公式进行判断：

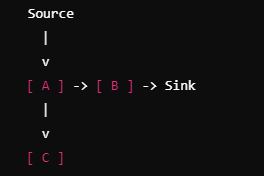
### 2.5 intersect

### 2.5 graph的source和sink

在进行图割算法时，需要对图论的基本知识进行了解。其中node分为source和sink类型，定义如下：







在进行图割时，每个顶点加入到IBFS图割算法时，需要设定这个顶点属于source和sink的权重。

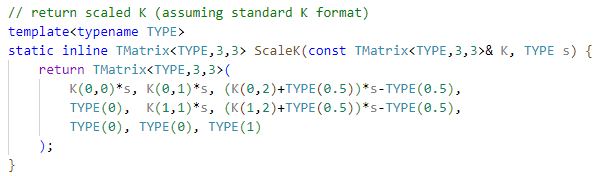
### 2.6 UpdateCamera

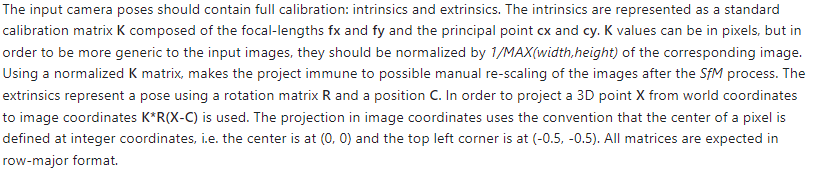
**相机内参的归一化操作：**

参考文档：

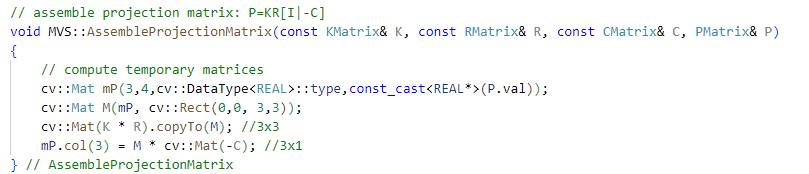
<https://dsp.stackexchange.com/questions/6055/how-does-resizing-an-image-affect-the-intrinsic-camera-matrix#comment71853_6098>

https://github.com/cdcseacave/openMVS/wiki/Interface#output





**相机投影矩阵：**

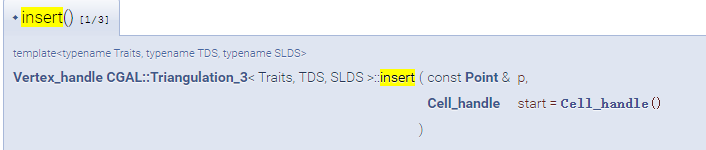


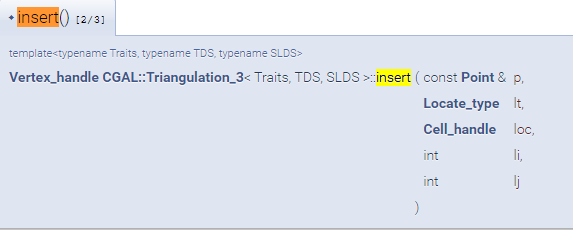
### 2.7 SelectNeighborViews

### 2.8 Export3Dprojections

### 2.9 德劳内对象的insert函数

在向德劳内对象中插入顶点时，cgal库提供了两个函数入口：





明明就只是插入一个顶点，不是主要顶点的坐标不就行了吗？这是因为，德劳内对象每次插入点都需要形成新的四面体结构，根据顶点和现有四面体的关系会犹如下的情况：

如果顶点恰好等于现有对象中的顶点坐标，那么德劳内对象整体结构不发生任何变化，然后返回这个顶点在德劳内顶点索引；

如果顶点在凸包内：如果在现有四面体内部，那么这个四面体内部被分裂成四个小四面体；如果在某个四面体的面上，那么公用这个面的两个四面体分别分裂成3个四面体；如果在边上那么公用这个边的所有四面体分别分裂成2个四面体；

如果顶点不在凸包内，但是在仿射包内：会构建一个新的三角面片；

如果顶点不在仿射包内：则

可以看出来插入一个顶点需要知道和现有的四面体的关系，因此需要进行一次搜索，那么到底从哪个cell开始搜索呢？start就是起始搜索的cell，会大大提高搜索效率。

### 2.10 InsertViews

## 三、Mesh Refine

优先级别低！！！！！！！！！

## 四、纹理优化