
algorithm 1 Generate Octree from Point cloud

Input: *point* Point cloud**Output:** *Octree* Octree structure

```
1: function GENOCTREE(point)
2:   mcode, Lmax  $\leftarrow$  MORTON(point)                                //Generate morton code and max level
3:   proot.childPoint add {all pointID}
4:   Octree1.node add proot
5:   for l={1,2,...,Lmax} do
6:     for node={OctreeL.node} do
7:       for pointID={node.childPoint} do                                //All pointIDs in a node voxel
8:         idn  $\leftarrow$  mcode(pointID)
9:         for i=0,1,...,7 do
10:          node.childPoint(i) add pointID if idn = i
11:          node.occupancyCode  $\leftarrow$  ISMEMBER(8 : -1 : 1, idn + 1)
12:          OctreeL+1.node add node
13:        end for
14:      end for
15:    end for
16:  end for
17:  return Octree2:L+1
18: end function
```

八叉树几何编码/解码算法

4.1 几何坐标的莫顿码表示

通过莫顿 (Morton) 码构建几何八叉树。点云中点的几何位置用三维笛卡尔坐标 (X, Y, Z) 表示。用 N 比特来表示每个坐标值, k -th 点的坐标 (X^k, Y^k, Z^k) 可以表示成

$$X^k = (x_{N-1}^k x_{N-2}^k \dots x_1^k x_0^k)$$

$$Y^k = (y_{N-1}^k y_{N-2}^k \dots y_1^k y_0^k)$$

$$Z^k = (z_{N-1}^k z_{N-2}^k \dots z_1^k z_0^k)$$

k -th 点对应的莫顿码可以表示如下:

$$M^k = (x_{N-1}^k y_{N-1}^k z_{N-1}^k, x_{N-2}^k y_{N-2}^k z_{N-2}^k, \dots, x_1^k y_1^k z_1^k, x_0^k y_0^k z_0^k)$$

将每三个比特用八进制数表示 $m_n^k = (x_n^k y_n^k z_n^k), n = 0, 1, \dots, N-1$, 则 k -th 点对应的莫顿码可以表示成

$$M^k = (m_{N-1}^k m_{N-2}^k \dots m_1^k m_0^k)$$

4.2 依据莫顿码构建八叉树

依据莫顿码, 按照广度优先从根节点 N_0^0 (第 0 层) 构建几何八叉树。

Step 1: 首先将所有的点根据莫顿码的第 0 个八进制数, m_{N-1}^k 分到八个子节点中:

- 所有 $m_{N-1}^k = 0$ 的点分到第 0 个子节点 N_0^1 中,
- 所有 $m_{N-1}^k = 1$ 的点分到第 1 个子节点 N_1^1 中
- ...:
- 所有 $m_{N-1}^k = 7$ 的点分到第 7 个子节点 N_7^1 中

则, 八叉树第一层的节点由此八个节点构成。

Step 2: 八比特 $B_0^0 = (b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0)$ 表示根节点 N_0^0 的八个子节点是否被占用。如果 $N_k^1, k = 0, 1, \dots, 7$ 至少包含点云中的一个点, 其对应的比特 $b_k = 1$; 如果该子节点不包含任何点, $b_k = 0$ 。

Step 3: 根据几何位置的莫顿码的第 1 个八进制数 m_{N-2}^k , 对第一层中被占用的

节点 $N_{l_n}^1$ 进一步划分成八个子节点；并用八个比特 $B_{l_n}^1$ 表示其子节点的占用信息
(l_n 是被占用节点的序号, $n = 0, \dots, N^1 - 1$, N^1 表示第一层被占用的节点数。)

Step 4: $N_{l_n}^t$ 再根据几何位置的莫顿码中的第 t 个八进制数 m_{N-1-t}^k , 对第 $t = 2, 3, \dots, N - 2$ 层中被占用的节点进一步划分成八个子节点；并用八个比特 $B_{l_n}^t$ 表示其子节点的占用信息. (l_n 是被占用节点的序号, $n = 0, \dots, N^t - 1$, N^t 表示第 t 层被占用的节点数。)

Step 5: 对第 $t=N-1$ 层, 所有的节点成为叶节点。如果编码器配置允许重复点, 需要在码流中记录每个被占用的叶节点上的重复点数。