# VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection

## 论文地址

<https://arxiv.org/pdf/1711.06396.pdf>

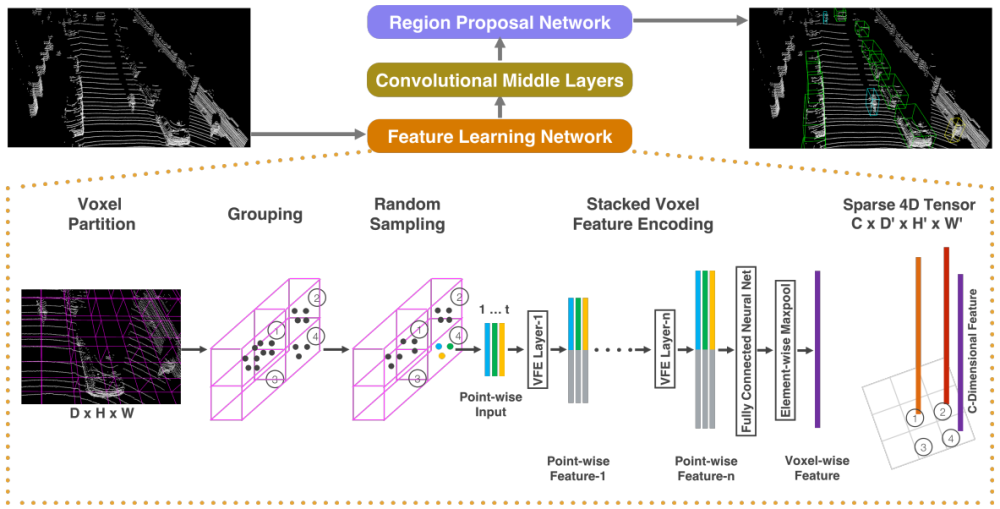
## 概括

点云处理的困难: 点云稀疏、点云密度变化、点云数量比较多

该论文提出一个直接处理点云，不使用人工特征的点云网络VoxelNet

## 模型架构与细节

### 模型架构



模型由三个部分组成，(1) Feature learning network, (2) Convolutional middle layers, and (3) Region proposal network

#### Feature learning network

* **Voxel Partition**：对于输入点云所在的空间大小使用D、H、W表示，等距分割成很多小的Voxel，每个Voxel大小为VD、VH、VW

* **Grouping**：根据点所在的Voxel进行分组，每组的点后续一起操作

* **Random Sampling**：从点数多于T的Voxel内采样出T个点，使得每个voxel内的点数都不超过T，降低Voxel之间的点数不平衡同时可以降低计算量

* **Stacked Voxel Feature Encoding**：将VFE layers级联，提取每个voxel的特征。VFE网络结构如下图所示，对于每一个Voxel来说，对内部的点均是执行如下操作，如点Pi(Xi，Yi，Zi，Ri)，其中ri表达的是反射率

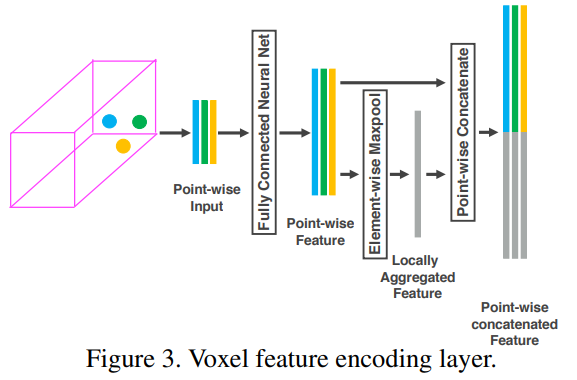
* + 计算所有点的均值，记为（Vx，Vy，Vz）

* + 将点Pi表示为{Xi，Yi，Zi，Ri，Xi-Vx，Yi-Vy，Zi-Vz}，增加了全局特征和残差特征

* + 对上述步骤的点通过FCN层转换到特征空间，FCN由一个线性层、一个BN、一个RELU组成，即可得到Point-wise Feature

* + 再通过Element-wise maxpooling层，结合周围的所有的点得到Locally Aggregated Feature

* + 最后将Point-wise Feature和Locally Aggregated Feature两种特征Concat在一起



* 经过n个VFE Layers之后就得到了上图中的Point-wise Feature-n，然后输入到FC再MP，就得到了描述该Voxel的特征Voxel-wise Feature，其中FC层的参数共享

* Sparse Tensor Representation：最终的点云可以用4D Tensor表示（尺寸C\*D'\*H'\*W'），C是每个Voxel的特征向量长度，D'是Z轴方向上的Voxels的个数，H'(Y轴)、W'(X轴)与D'意义相同，其中XOY平面就是BEV平面。超过90%的Voxel都是空的

#### Convolutional Middle Layers

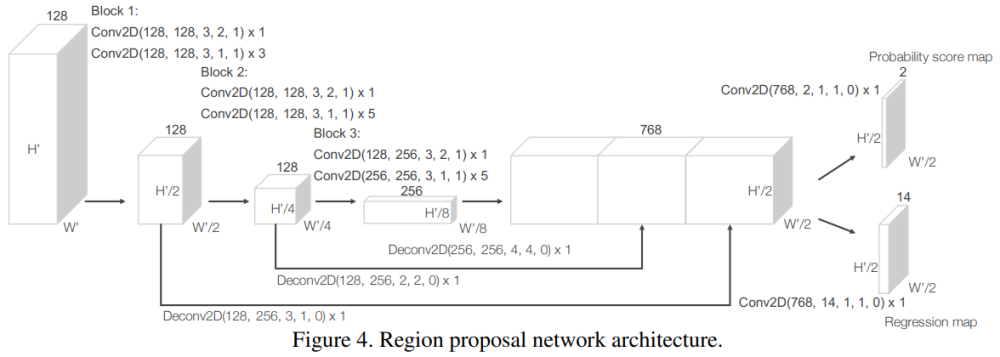
* 使用3D卷积对4D tensor进行卷积，并进行reshape到3D tensor

* + 举例:输入尺寸是4D Tensor是(128\*10\*400\*352)，输出（经过Convolutional Middle Layers之后）是（64\*2\*400\*352）,然后reshape到128\*400\*352,变成3D tensor（注意到400\*352 正是BEV视图上的栅格尺寸）

* + - 3D卷积可以聚合Voxel周边关联的Voxel的信息

* + - Reshape之后可以使用RPN网络进行2D卷积

#### Region Proposal Network



* 多尺度采样，上采样特征图再Concat在一起

* 学习目标：Probability score map 和 Regression map

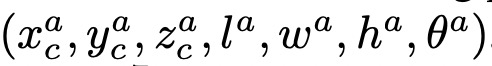
* 最终的卷积是在类似BEV的视图上进行卷积，但是输出的回归量是3D的

### Loss Function

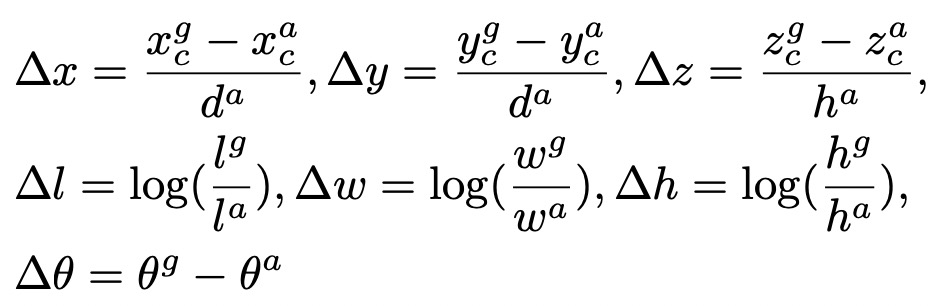
Scores通道为2，分别对应positive和negative分数，regression map存在14维，对于每一个回归的BBox都用7维来表示



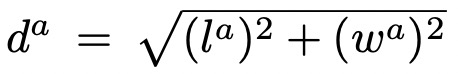
即中心位置+长宽高+yaw轴，默认地面是水平的，因此只需要7个参数既可以进行表示。假设要预测的anchor为:



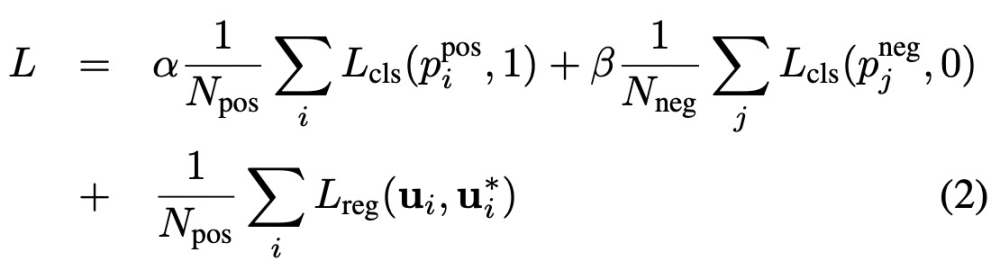
此时残差表示如下:



其中：



损失函数定义为:



损失为常见的交叉熵损失函数和L1 Loss损失函数，主要学习的是与anchor-gt之间的残差，借鉴了Faster RCNN模型。

anchor的大小是跟具体检测的目标有关系的，如汽车是需要使用专门设计的anchor来进行检测，假设预测出的anchor在鸟瞰图中具有和gt最大的IOU或则它与gt的IOU值大于0.6就为positive.如果和所有的gt都小于0.45，就是negtative，处于0.45~0.6之间的不做处理。

## 简单总结

* 该论文当时直接在点云上得到SOTA的结果，比很多直接使用多模融合效果要好，原因可能是

* + 3D下的目标检测，可能跟RGB信息相关性不那么大，更容易通过形状信息来学习到

* + 多模交互的方式存在问题？