ChipNet: Real-Time LiDAR Processing for Drivable Region Segmentation on an FPGA

Abstract:

本文是Real-Time Road Segmentation Using LiDAR Data Processing on an FPGA 的扩展。仍然是使用CNN通过FPGA对road detection获得比较好的效果，并且速度有了提升。

本文作者提出了一个CNN网络称为ChipNet并在FPGA对处理LiDAR数据获得了实时的较好的效果。总结如下：

1. We introduce a new data organizing and sampling method in spherical coordinate that improves the usage of LiDAR points and creates a dense input tensor for CNN.
2. We propose an efficient convolution block for CNN that is both hardware friendly and extendable.
3. The proposed approach of drivable region segmentation results the state-of- art accuracy when evaluated using Ford dataset and KITTI benchmark. We also labelled the Ford dataset for training and evaluation. （为Ford数据集进行了标注）
4. An efficient and flexible 3D convolution module is designed and implemented on an FPGA, which can achieve real-time processing speed with limited hardware resource and power usage.

Related work：

Lidar data arrangement

文献[6] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *IEEE CVPR*, volume 1, page 3, 2017. 见point cloud数据转化到top view和image view并用CNN将其与camera数据相结合。VoxelNet和3D-FCN在世界坐标系中用CNN直接处理points数据。LoDNN则将point cloud投影到top-view中，再送到CNN中。

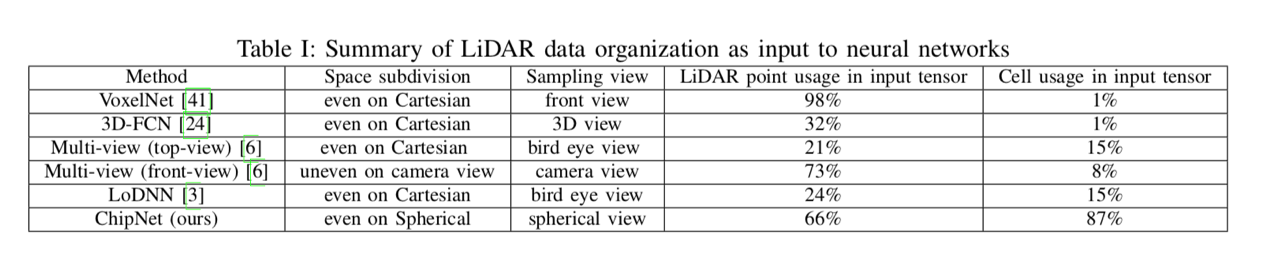
CNN for road perception

主要是两种FCN和encoder-decoder结构

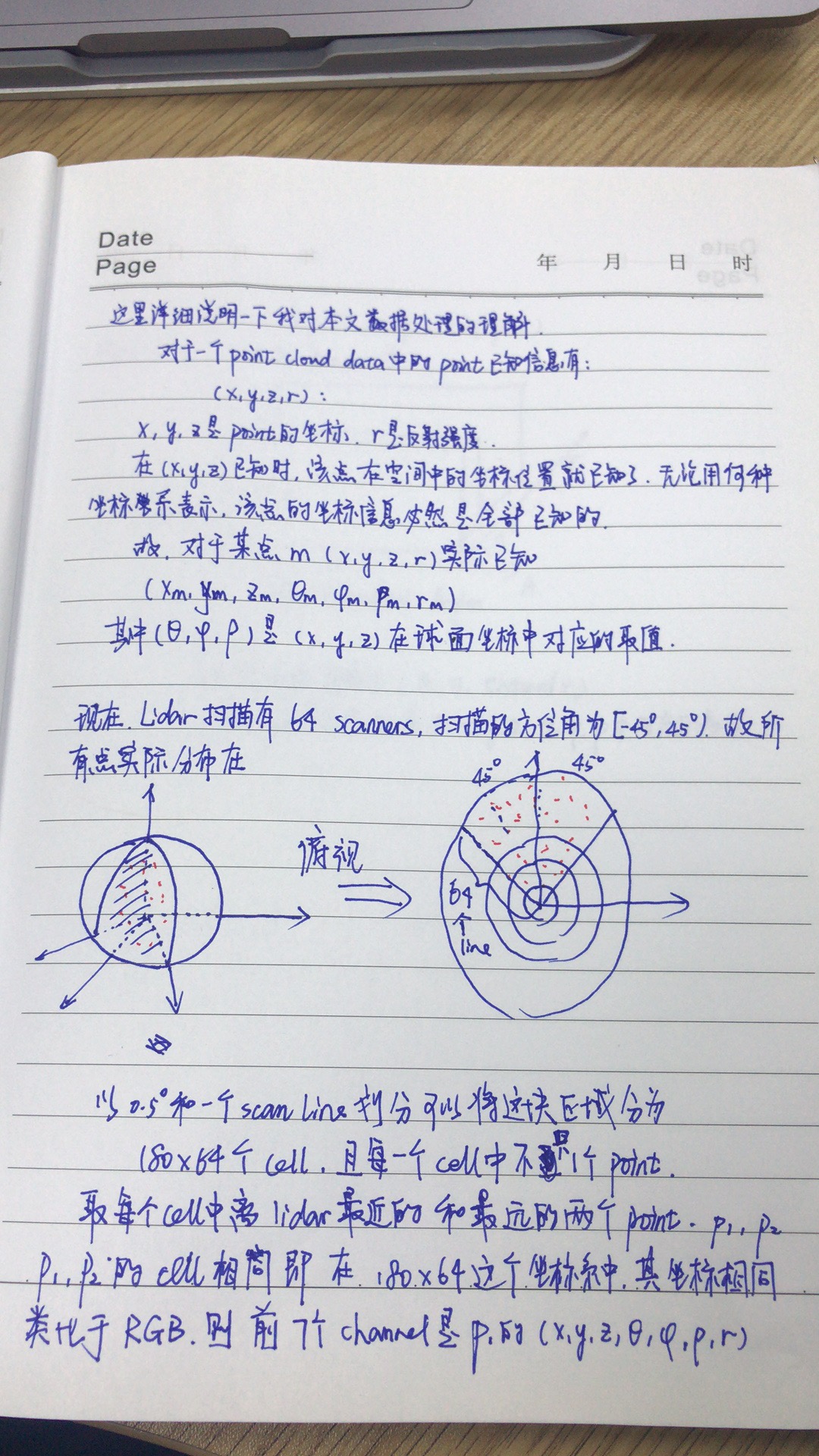
Algorithm design

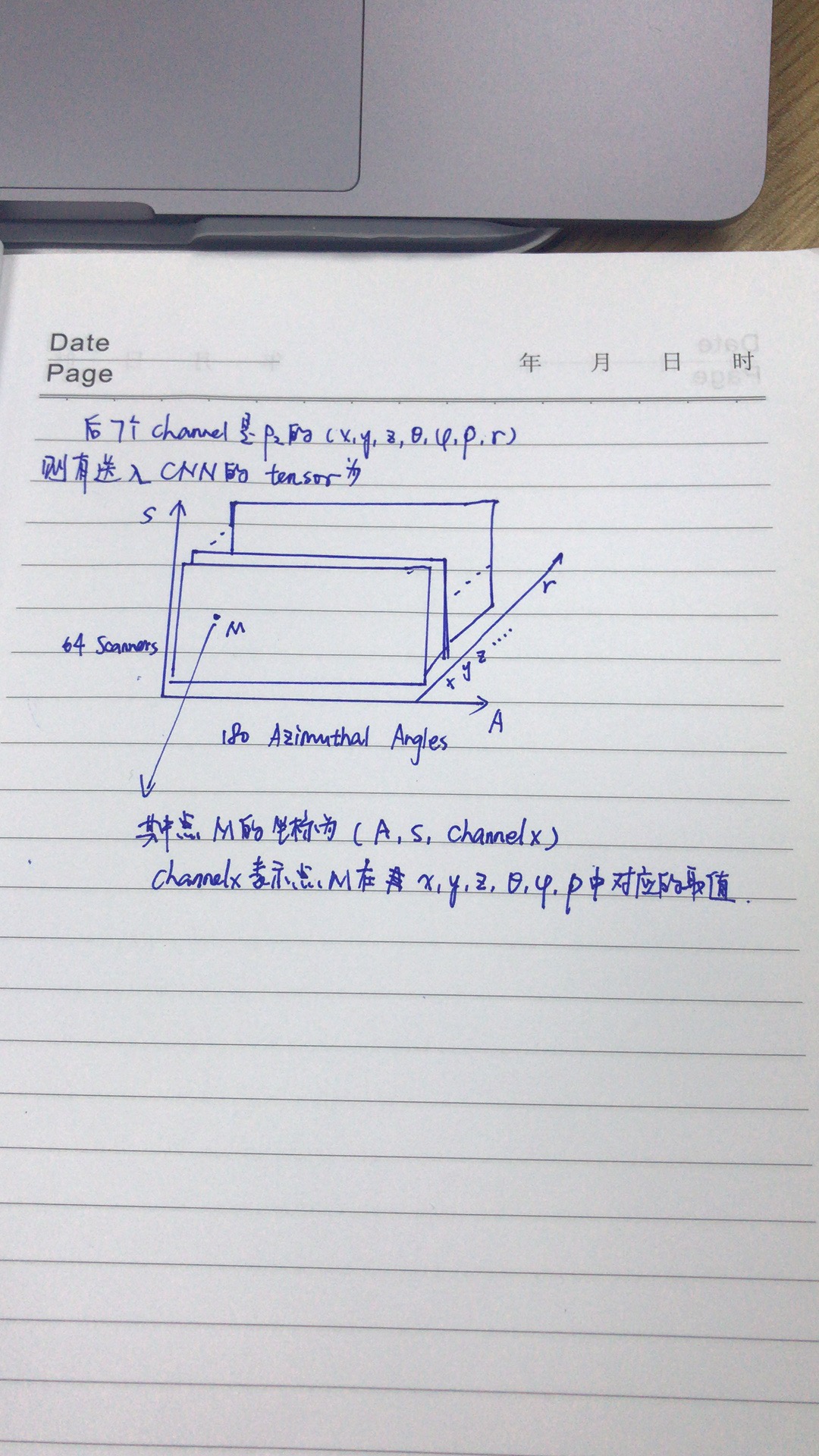
1. LiDAR data preparation

一般的Lidar表示为(x,y,z,r)其中(x,y,z)是该point在lidar坐标系上的坐标，r是该点在反射平面上的反射强度。下表总结了以不同形式组织LiDAR数据的几项研究工作。



预处理过程如下：



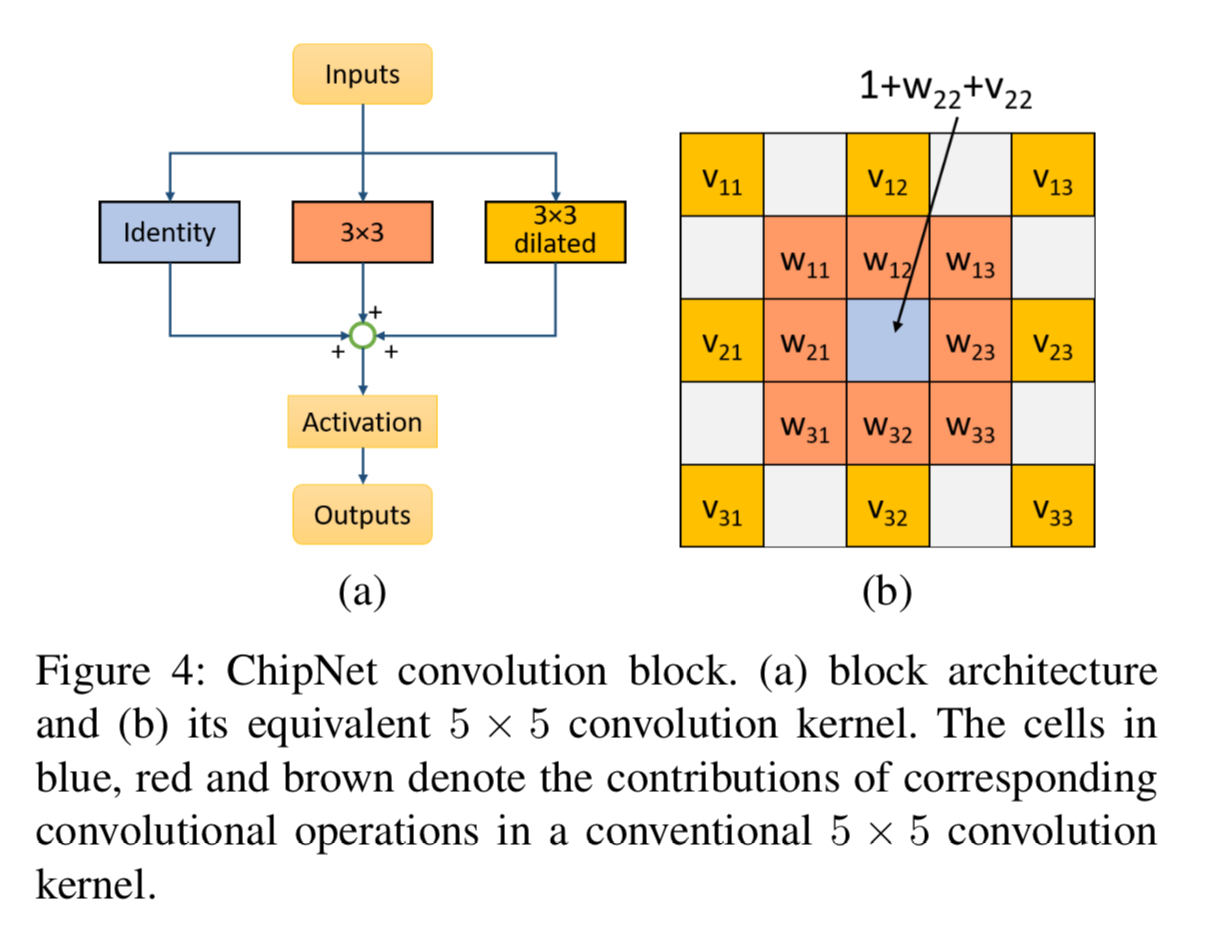


1. ChipNet

作者提出了一个新的CNN网络(实际上是空洞卷积)

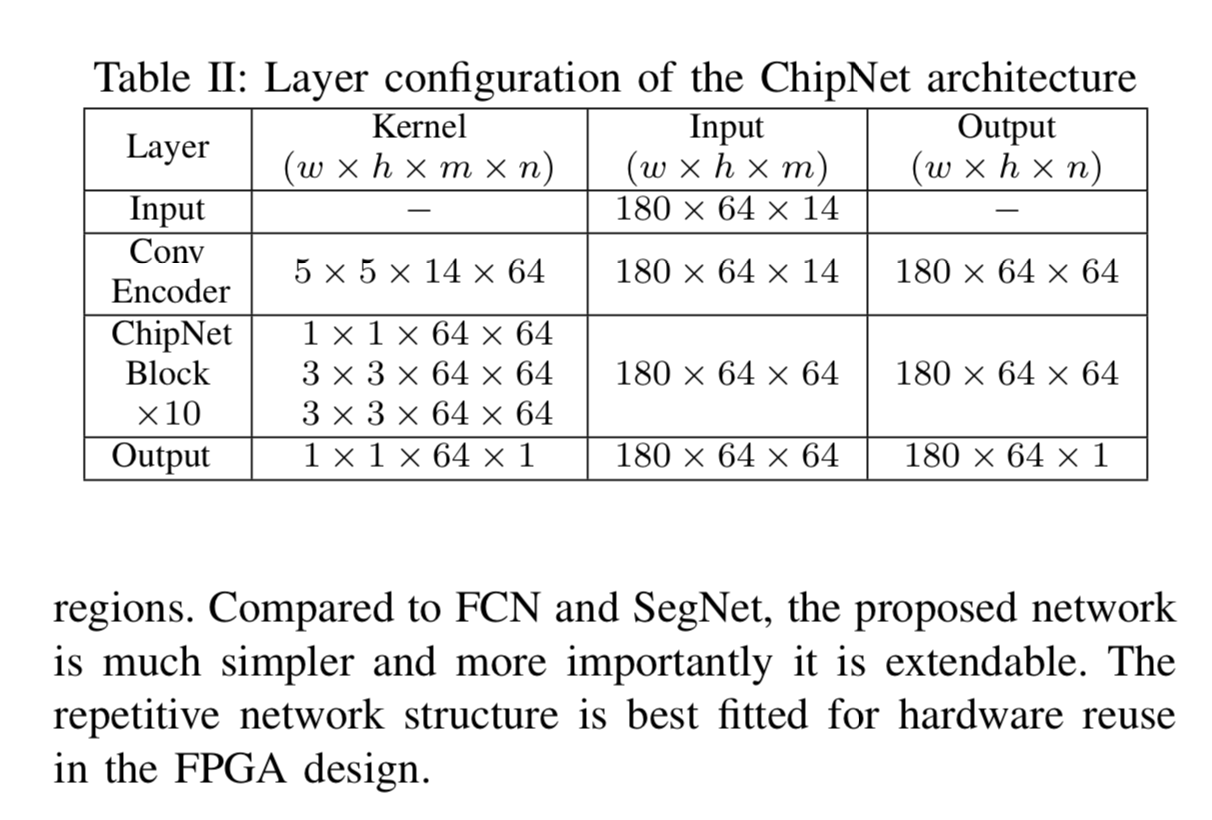
1. 设计了一个相当于5×5卷积核的扩张块（dilated block），但是减少了参数量。
2. 用dilated block设计了一个扩张的CNN结构。
3. 提出模拟量化算法为硬件获取定点参数。
4. ChipNet卷积block：  
    每个network block有三个branches:
5. Identity branch:直接将inputs复制给outputs:防止梯度消失和梯度爆炸
6. 第二个branch：3x3卷积，输出channels是64：目的是对特征编码
7. 第三个branch：3x3空洞卷积

Block结构如下所示：



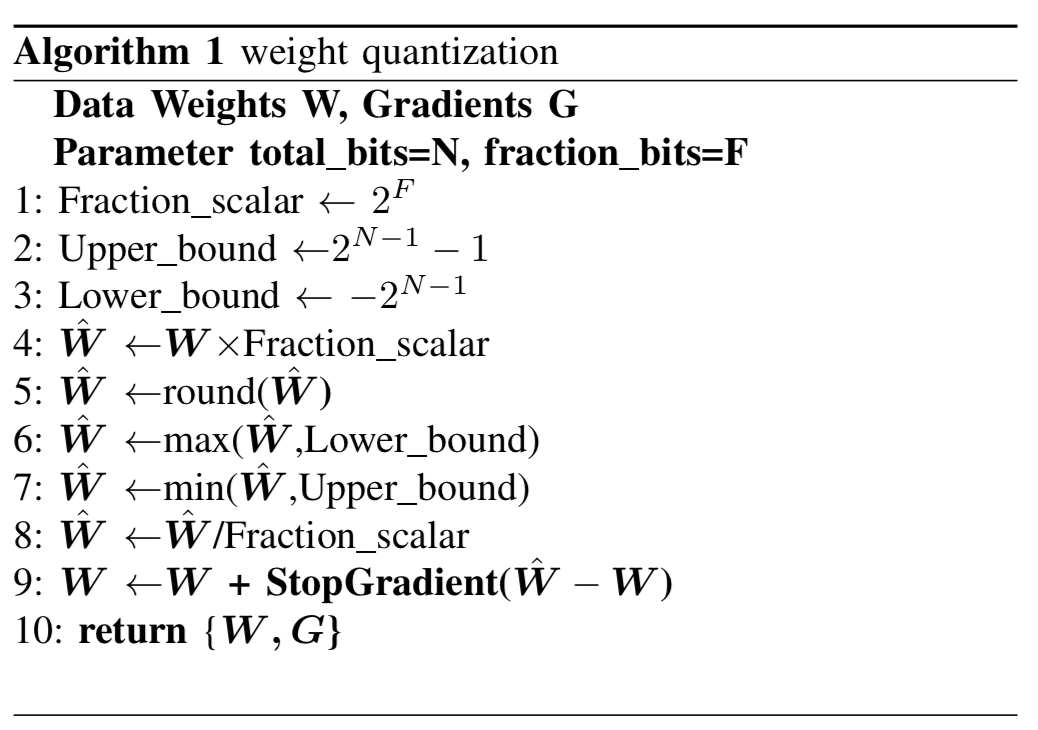
1. ChipNet的结构：

1. 第一层是局部特征编码器，旨在将输入的LiDAR数据编码为64通道特征张量。
2. 在编码之后，在网络中重复地实现ChipNet卷积block以执行附加编码和解码。
3. 对于输出层，使用channel-wise mapping来生成最终的decision mapping，显示相应可通行区域的概率。



1. Simulated quantization：
   1. Simulated quatization of weight：

在我们的工作中，定义了一个新的权重调节器并将其添加到现有网络中。算法1中描述了调节器。关键的创新是调节器在训练期间量化权重，并且在前向操作期间使用定点数。同时，浮点权重也存储在存储器中，当在反向传播期间计算梯度时使用浮点权重。然而，量化函数是不可微分的。因此，作者引入StopGradient函数。 StopGradient函数是TensorFlow中的内置函数，它对给定输入强制其梯度为零。通过应用该函数，梯度保持与浮点反向传播相同，同时权重被量化。因此，所提出的量化算法作为插入式调节器导入TensorFlow平台。与[21]中的模拟量化相比，所提出的量化算法和数据流如图5所示。

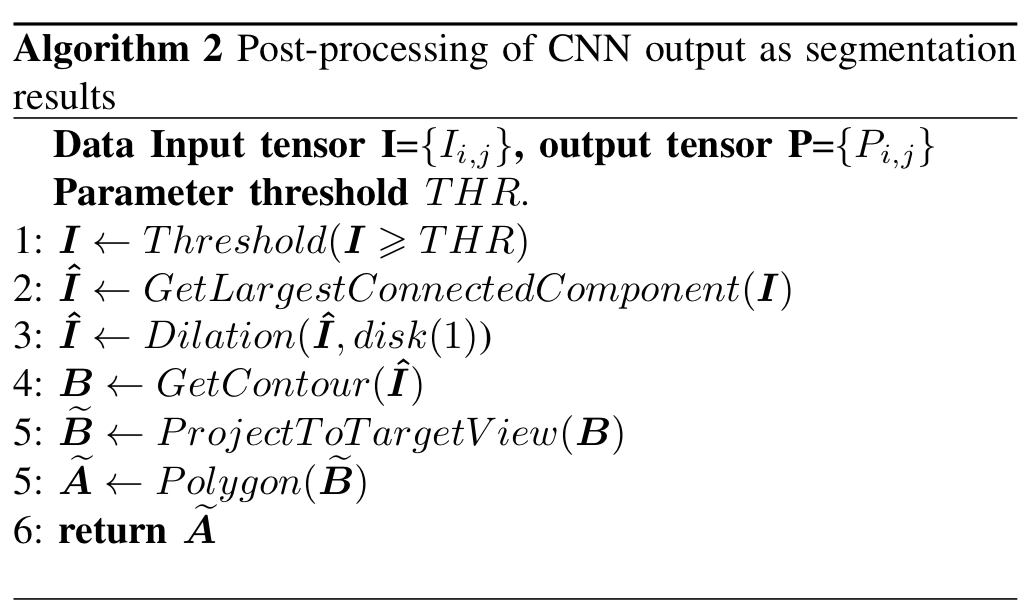


* 1. Simulated quantization of variables

和weight的量化基本一致，具体要看源代码

1. View of drivable region

网络的输出表示了每一个cell在球面视图中时可通行区域的可能性。在post-processing中，我们将output投影到车前方20m\*40m的长条区域中，使用top-view来表示。Post-processing算法如下所示：



说明：

* + 1. 可通行区域可能性的阈值设为THR

2）then the reference point in each column j in the network output Pi,j is determined by the nearest LiDAR point in group {P|col=j,p<THR}. 在生成eference point之后，可通行区域的轮廓变为包含所有参考点作为顶点的多边形。

3）在后处理中，我们初始化一个网格图实例，其范围设置为x坐标为[6,46]米，y坐标为[-10,10]米。 分辨率设置为每个单元0.05米，使得网格图在x坐标中具有800个单元，在y坐标中具有400个单元。 当后处理节点接收网络输出帧时，它将帧存储为特征矩阵。