**3D 目标检测与跟踪**

**一.** 流程图：

3D 目标检测输入的是3D 障碍物的观测数据，针对apollo，其数据即是激光雷达的点云数据。其输出为检测到障碍物的信息，一般包含位置和速度。期处理流程如下：



二. HDMap Region of Interest (ROI) Filter：

（1） 功能：

从hd map中过滤出可以驾驶的区域。并且过滤掉ROI区域以外的树和建筑物等。

（2） 输入：

1） hd map：一个多边形的集合，每个多边形是一些点的有序集合，没个点都是在世 界坐标系下标记的，即世界坐标系下的坐标；事先做好的。

2） 3d 点云：从激光传感器接收到的。

（3） 输出：

输入点的索引，这些输入点是在ROI中的，ROI来源于HD map。

（4） 组件：

1） 坐标转换：

在使用点云数据和hd map做查询时，需要将两者转化到同一个坐标系下，点云 数据应该是基于lidar的，hd map是基于世界坐标系的，apollo将两者转化为一个本 地坐标系，以便进行查询（对比）。

2） 构造ROI LUT（查询表）：

为了确定输入点是否在ROI区域，apollo将HD map 量化为一个2D grid：A， 在其中根据实际环境标定ROI之后2D grid：B，根据 是否在ROI，把grid中的cell 标记为1/0，然后根据激光雷达的扫描范围，生成 一个2d grid：C，求出B与C的交 集即可。（使用线扫描算法和位图编码）

3） 使用LUT查询点云：

点是否在矩形区域；使用LUT判断点云是否属于ROI；收集所有属于ROI的点， 输出其index。矩形区域即是认为规定激光扫描的有效范围，与激光雷达的器件相关。

三. 基于CNN的障碍物分割：

（1） 功能：

检测出障碍物以及类型，例如轿车、卡车、自行车以及行人等。

（2） 输入：

1） 点云：同ROI。

2） 在ROI中点的索引，ROI filter的输出。

（3） 输出：

ROI中障碍物的数据集合，应该还是点云。

（4） 组件：

1） 通道特征提取：

对于每一帧的点云，apollo根据传感器（激光雷达）位置为原点构建一个2D 栅 格图坐标系，每个点根据其XY坐标值分配到对应的cell中，针对每个cell中的点，计 算出八个统计值，作为下一个步骤的输入。八个统计值分别为：

a) 单元格中点的最大高度。

b) 单元格最高点的强度。

c) 单元格中点的平均高度。

d) 单元格中点的平均强度。

e) 单元格中的点的个数。

f) 单元格中心与原点的角度。

g) 单元格中心与原点的距离。

h) 表示单元格是否为空的二进制值。

疑惑：2D栅格图坐标系如何生成？点的强度是什么意思？

2） 基于CNN的障碍物预测：

根据通道特征提取结果，使用深度完全卷积神经网络（FCNN）算法预测障碍物的属性，包括中心偏移、对象性、积极性（啥意思）以及障碍物的高度。

输入为栅格的列号、栅格的行号以及通道特征数；输出为中心偏移、对象性、积极性（啥意思）以及障碍物的高度。

FCNN分为三层：下游编码层，上游解码层和对象属性预测层。是算法和核心部分。通道特征图片作为输入，首先经过编码层以卷积为手段下采样，然后经过解码层以退化为手段上采样只有，进入障碍物预测层，进而预测给出物体具体的属性。

对于CNN还不太了解，这里没有深入。先搭架构，在研究理论。

3） 障碍物聚集：

根据预测结果，产生候选障碍物对象集群。输入是中心偏移、对象性、积极性和对象高度，输出为候选障碍物对象集群。apollo以中心偏移为中心，采用压缩的联合查找算法构建有向图，每个cell为一个node。

对象性表示是否为障碍物的概率，超过0.5为是，否则不是。此步结束，即获取障碍物信息。

4） 后端优化：

为每个候选障碍物集群打分，输出最终的障碍物集群或者分段。

四. MinBox 障碍物边框重建：

（1） 功能：

为给定的障碍物构建一个边框，目的扩展激光雷达所获取的障碍物面积，由于激光雷达本身的局限，可能使得障碍物不够大，这样就相当于增加了一层膨胀层，为了安全吧。

（2）输入：

障碍物信息。

（3）输出：

带边框的障碍物。

（4） 组件：

1） MinBox算法。

2） 构建6边形。

五. HM对象跟踪：

（1） 功能：

根据分割出来的障碍物，估计出其运动轨迹。

（2） 输入：

分割出来的障碍物信息。

（3） 输出：

障碍物的运动轨迹。

（4）组件：

1） 检测跟踪关联：分两步，第一步构建距离关联矩阵，根据关联特征计算给定检 测和一条轨迹之间的距离。第二步，求出距离关联矩阵之后，根据匈牙利算法构建一个 二分图通过最小化距离成本找到最佳的监测跟踪匹配。

2） 跟踪运动估计：

在检测跟踪关联之后，HM 对象跟踪器使用鲁棒卡尔曼滤波评估当前轨迹列表中 轨迹的运动状态，采用不变速度运动模型。运动状态包含3D 障碍物的位置和速度， 为降低检测时的干扰影响，在滤波器算法中引入鲁棒统计技术。

a) 冗余观察：

速度的选取来源一系列冗余观察值，包括锚点、边框中心、边框角点等，速度 量也是过滤算法的输入。因为多观察值的误差小于单个观察值，所以冗余观察提 高了鲁棒性。

b) 阈值：

高斯滤波算法假设噪声属于高斯分布，然而在运动估计中，噪声通常属于肥尾 分布，这会造成更新补偿过高，所以在过滤算法中需要设置一个阈值。

c) 更新关联质量：

原始卡尔曼滤波器更新其状态不区分其测量的质量。然而，质量是滤波噪声 的有益提示，可以估计。例如，在关联步骤中计算的距离可以是一个合理的测量 质量估计。根据关联质量更新过滤算法的状态，增强了运动估计问题的鲁棒性和 平滑度。

（5） 运动估计小结：

1）构造跟踪对象并将其转换为世界坐标。

2）预测现有跟踪列表的状态，并对其匹配检测。

3）在更新后的跟踪列表中更新运动状态，并收集跟踪结果。