**障碍物层栅格地图构建算法流程(不包含freespace信息)**

**一.方案一:采用BOF滤波器的粒子实现进行每个栅格占用情况的后验概率的更新,使用融合之后的boundingbox附带的速度信息作为先验速度；**

结合freespace信息的话就是boundingbox对应的是观测占用区域,freespace对应的是观测空闲区域,其余的就是未观测区域,作为*m*\_*grid*\_𝑐𝑒𝑙𝑙\_𝑎𝑟𝑟𝑎𝑟𝑦用于计算后验概率

**1.具体流程**

* **(1).超参数:栅格数量,栅格尺寸,每个网格生成粒子数(在每个boundingbox对应的栅格生成粒子的时候就把速度给粒子)**
* **(2).输入:融合后的3dboundingbox列表和速度和Odometry信息**
* **(3)具体流程:**

1.初始化栅格地图*grid*\_𝑐𝑒𝑙𝑙\_𝑎𝑟𝑟𝑎𝑟𝑦*(0)*,和车身的初始位姿P(0)

第i个栅格在t(0)时刻的状态表示为M*i,j*(0)=( O , F , Vx , Vy){占用率,空闲率,X方向速度,Y方向速度};

* t(0)时刻的车身位置为P(0)=(X , Y , θ){X, Y, 偏航θ};
* 根据第一帧出现的boundingbox信息生成一定数量的粒子*particles\_𝑐𝑒𝑙𝑙\_𝑎𝑟𝑟𝑎𝑟𝑦*,第n
* 个粒子Ln(Ix,Iy,Vx,Vy){第几行,第几列,初始X方向速度,初始Y方向速度};
* 每个栅格中粒子的数量乘以权重表示网格的占用状态,平均速度表示网格的速度

2.while k时刻 List=3Dboundingbox列表 ,odom=里程计信息 ,*grid*\_𝑐𝑒𝑙𝑙\_𝑎𝑟𝑟𝑎𝑟𝑦*(K-1)*

3. 根据t(k)和t(k-1)时刻位姿计算ΔP=(ΔX,ΔY,Δθ),得到旋转和平移矩阵R(k)和T(K)

4. 根据R(k)和T(k) 更新局部栅格地图让栅格坐标系跟随车身坐标系变化

5. 将3Dboundingbox投影到2D,计算得到观测被占用的栅格列表*m*\_*grid*\_𝑐𝑒𝑙𝑙\_𝑎𝑟𝑟𝑎𝑟𝑦

6. *grid*\_𝑐𝑒𝑙𝑙\_𝑎𝑟𝑟𝑎𝑟𝑦*(K-1)中的粒子*速度预测t(k)时刻粒子属性Lj(k)

7. 根据预测之后的粒子群属性计算得到预测的栅格地图列表*p*\_*grid*\_𝑐𝑒𝑙𝑙\_𝑎𝑟𝑟𝑎𝑟𝑦

8. 用*m*\_*grid*\_𝑐𝑒𝑙𝑙\_𝑎𝑟𝑟𝑎𝑟𝑦和*p*\_*grid*\_𝑐𝑒𝑙𝑙\_𝑎𝑟𝑟𝑎𝑟𝑦得到更新后的后验𝑔𝑟𝑖𝑑\_𝑐𝑒𝑙𝑙\_𝑎𝑟𝑟𝑎𝑟𝑦*(k)*

9. 对粒子更新(重采样,生成,消失)

9. 𝑔𝑟𝑖𝑑\_𝑐𝑒𝑙𝑙\_𝑎𝑟𝑟𝑎𝑟𝑦*(k)* 作为𝑔𝑟𝑖𝑑\_𝑐𝑒𝑙𝑙\_𝑎𝑟𝑟𝑎𝑟𝑦*(k-1)的先验概率*

**(4)在kitti数据集上测试一下效果,不行就换方案二**

**2.估计**

1.先尝试用BOF滤波器的粒子实现去做

2.如果这个boundingbox的速度好用的话应该理论上可以提供一个先验速度,但是不知道这个速度能不能好用,如果不好用的话就只能用方案二.

**二.方案二:只使用融合之后的boundingbox的位置信息,不用速度信息,用DS-PHD/MIB滤波器进行预测和更新栅格的占用状态**

https://github.com/TheCodez/dynamic-occupancy-grid-map改为Gpu版本,结合里程计信息,输入改为3Dboundingbox列表