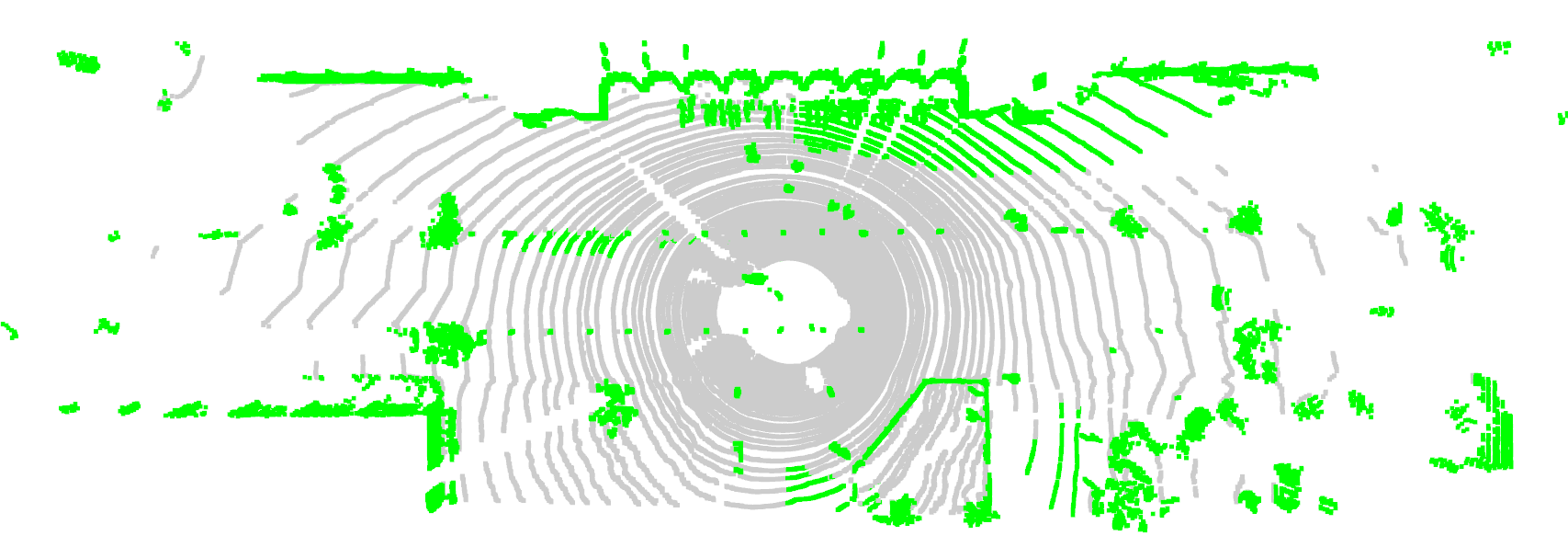
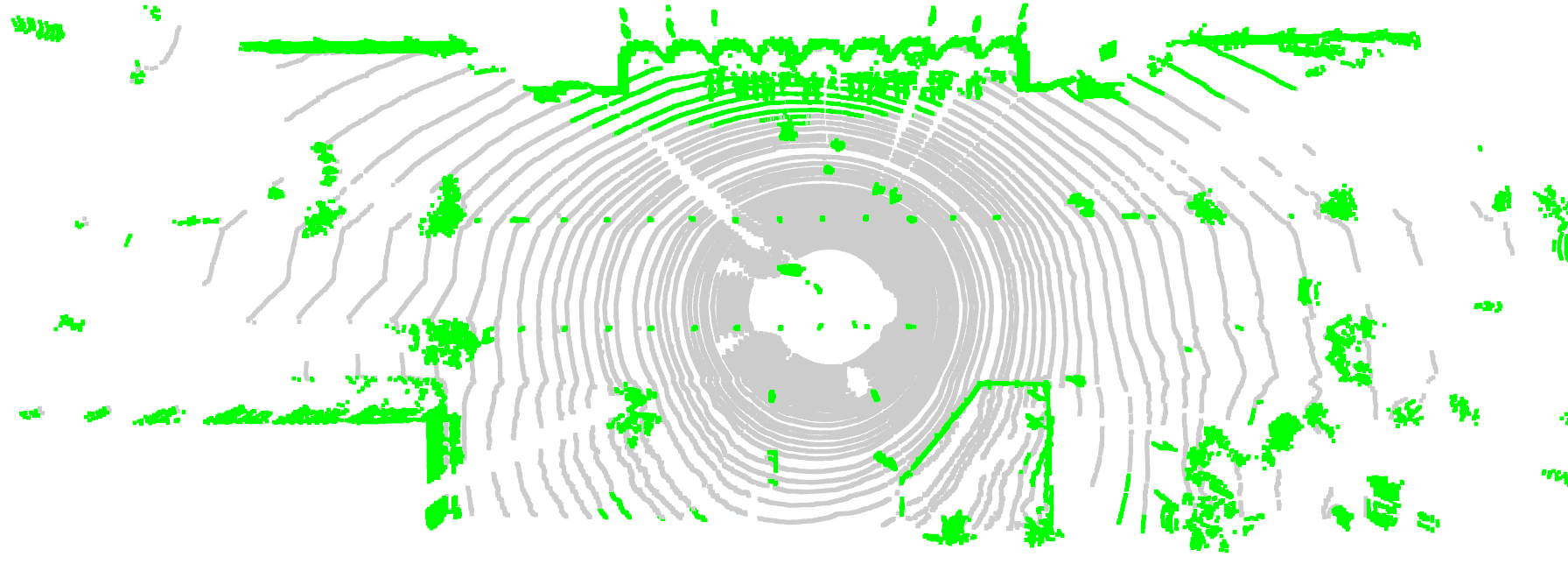
1. Ground detection test on /test/000111.bin

参考：Fast Segmentation of 3D Point Clouds: A Paradigm on LiDAR Data for Autonomous Vehicle applications

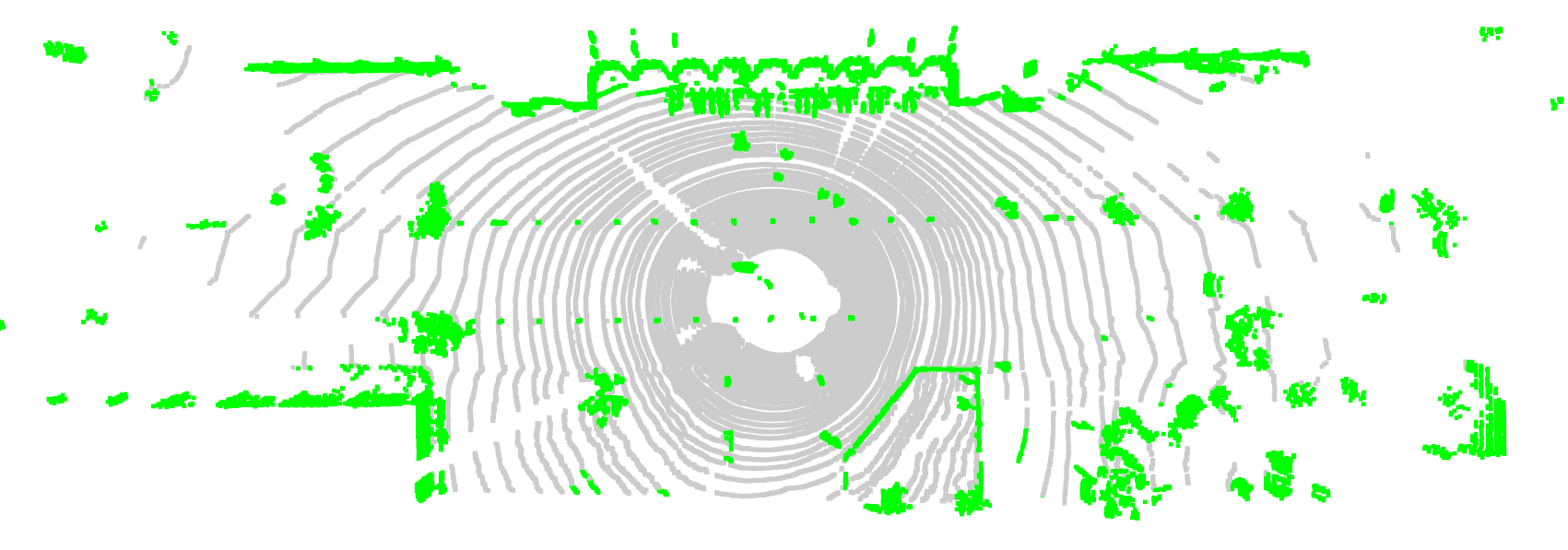
RANSAC on 2 segments: runtime ~750ms in python



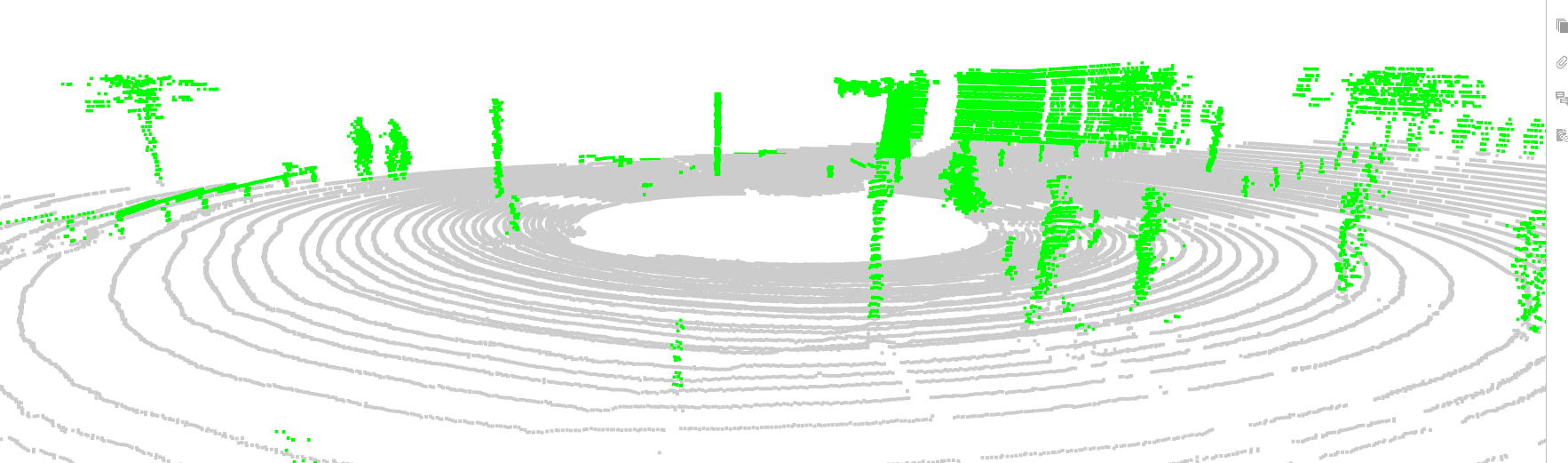
RANSAC on 5 segments: runtime ~750ms in python



Iterative SVD Method on 3 segments with initial ground points extraction: runtime ~100ms in python

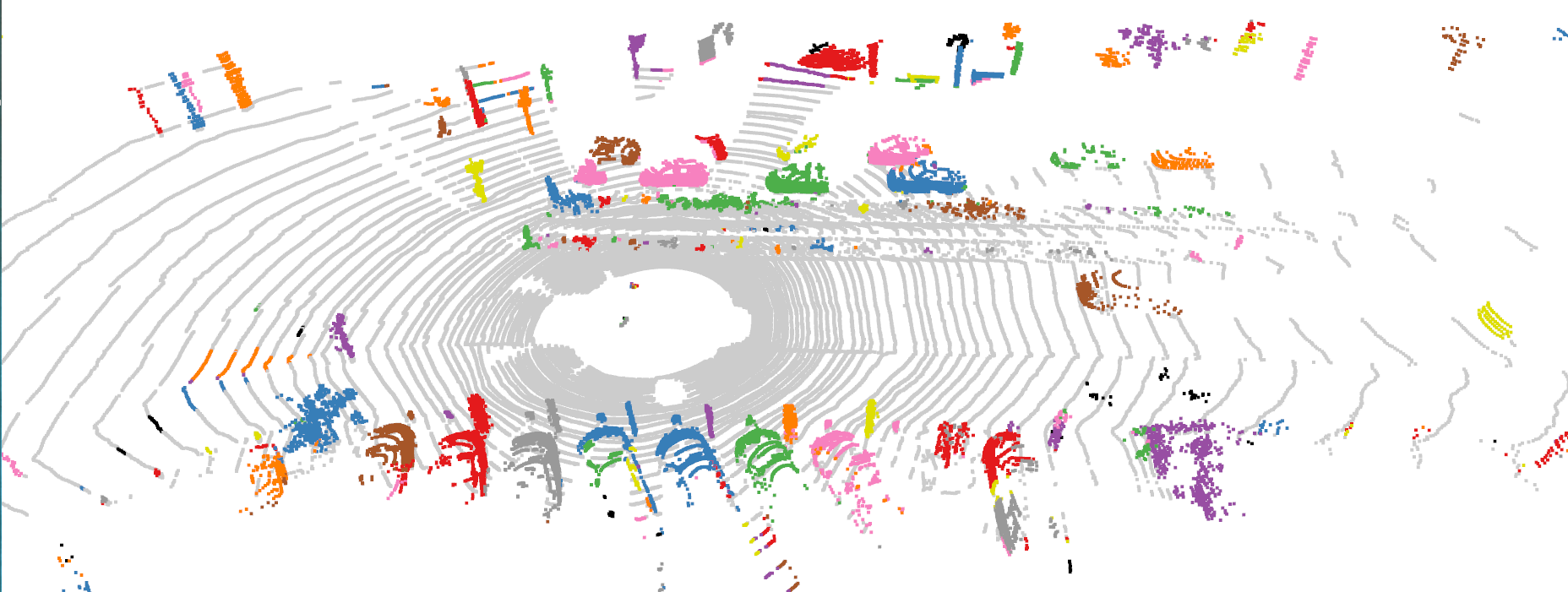


行人和地面上结构的下部分被保留的很好。

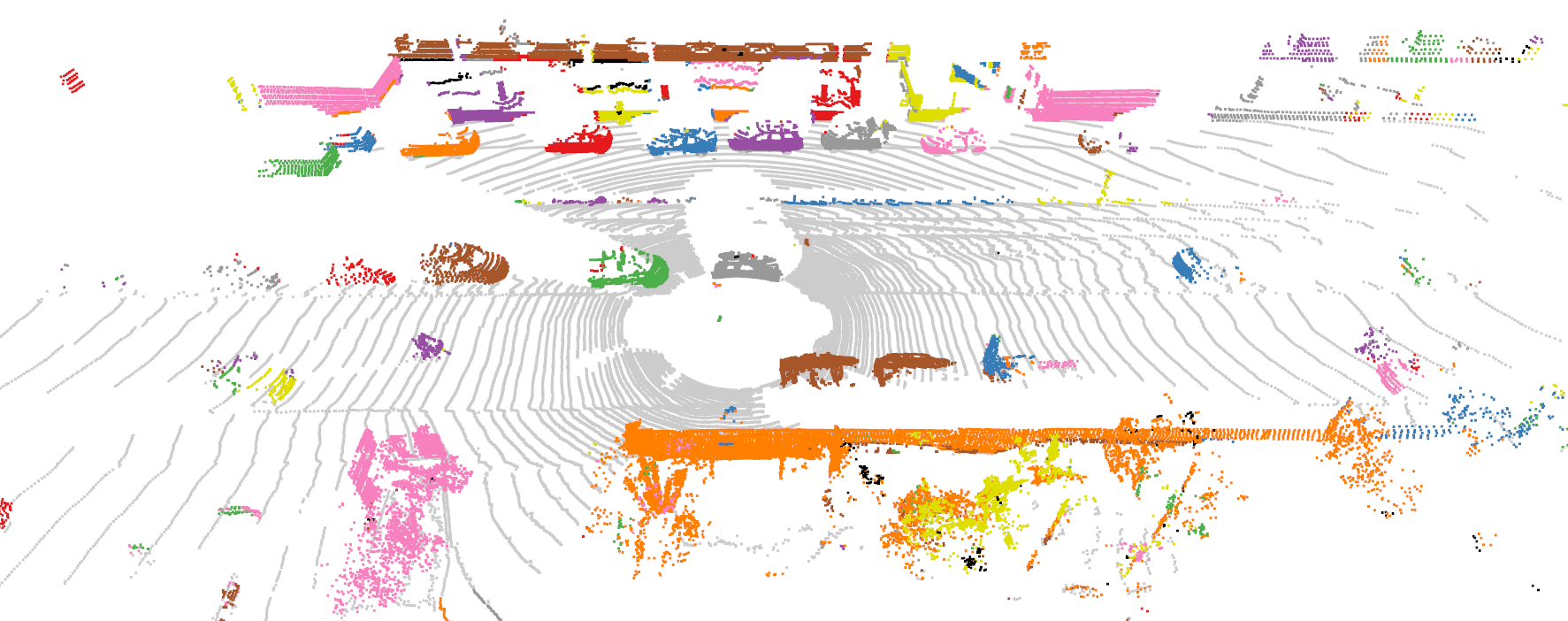


1. Clustering on range image:

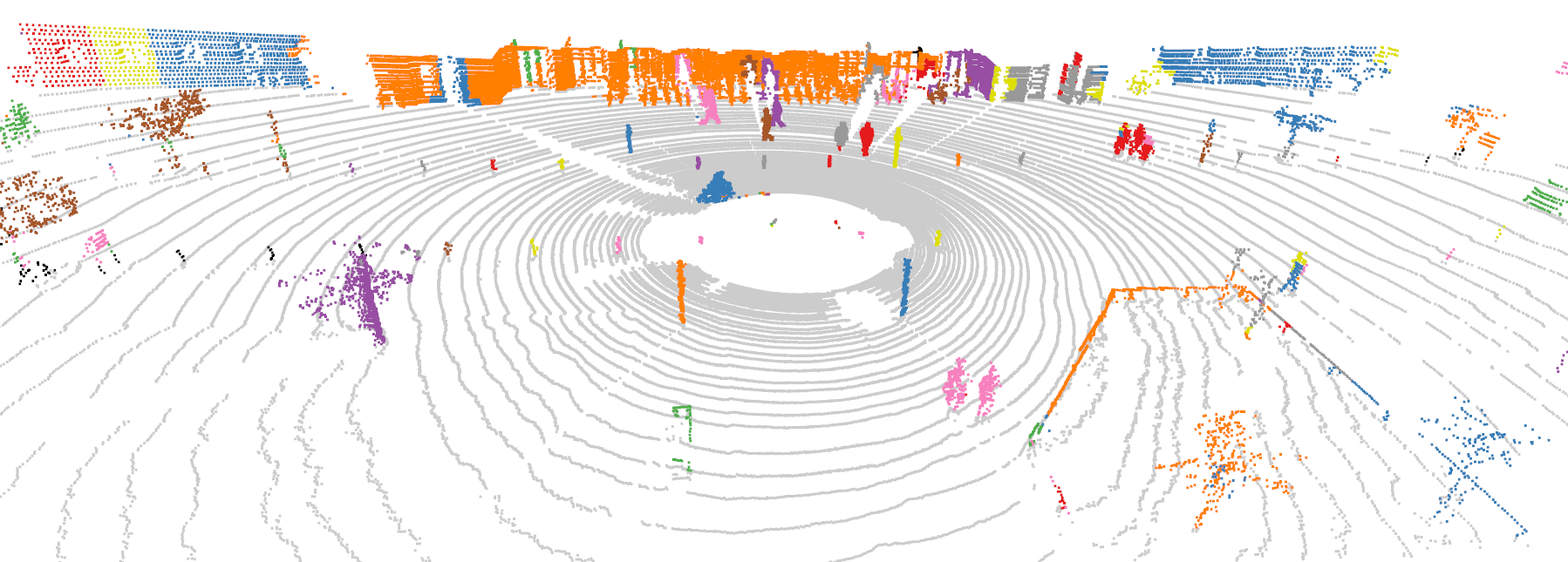
/test/000077.bin



/test/000099.bin



/test/000111.bin



该方法参考了Efficient Online Segmentation for Sparse 3D Laser Scans，将3D问题转换成2D，在2D深度图中利用BFS检测不同目标的连通性，检测思想为两个pixel对应两个三维点的连线和较远点对应laser beam之间的夹角是否过小，以及两个pixel的深度值之差是否过大。该方法优势在于检测速度快，不用进行radius neighbour search。劣势在于：

1. 该方法受range image的精度影响较大：当精度较小时，属于同一物体的pixel的连通性较好，但多个三维点被分配到了一个同一个pixel，造成误判; 当精度较小时，属于同一物体的pixel的连通性较差，容易over segment，我的解决方法是用morphology operation中closing的方法处理range image（见foreground\_clustering.py 159行），增加连通性，但物体边缘会受到影响，边缘处over segment依然存在。
2. 对于过laser beam的平面，检测思想中提到的夹角几乎为0，该情况也会出现over segment。

综上，将3D转换2D做clustering，相比于DBSCAN带来了运行速度的提升，但牺牲了一些三维欧式空间的信息，聚类准确性的下降。得出的结论是：没有一个聚类算法是万能的，根据应用场景采用具体算法吧:(