



深蓝学院
shenlanxueyuan.com

三维点云处理第三期

——第四章作业讲评



主讲人 郝爽





Homework



Object detection pipeline for lidar

- Use KITTI 3D object detection dataset, select 3 point clouds, do the followings.
- Step 1. Remove the ground from the lidar points. Visualize ground as blue.
 - Any method you want – LSQ, Hough, RANSAC
- Step 2. Clustering over the remaining points. Visualize the clusters with random colors.
 - Any method you want
- Step 3. Classification over the clusters
 - Homework of Lecture 5
- Step 4. Report the detection precision-recall for three categories: vehicle, pedestrian, cyclist
 - Homework of Lecture 5

- 数据分析：kitti地面点不是严格的平面，高度差最大可能超过0.5m，拟合完平面后需要结合策略去做精细分割。一帧点云大概有12万个点，处理时间可能很长。kitti数据包含大量噪点。
- 思路分析：首先是地面点的提取，地面点提取的好坏决定了后面聚类的结果好坏。地面提取遗漏的小部分地面点对聚类影响很大。
- 预处理：Voxel Grid降采样
- 地面点分割：LSQ、Hough Transform、RANSAC

解题思路：地面提取



If we know the inlier points

- Least Square



What if there is small amount of outliers?

- Robust Least Square, e.g., robust loss function
- Hough Transform
- RANSAC



What if there are lots of outliers / more than one models in data?

- Hough Transform
- RANSAC

解题思路：地面提取



Hough Transform – Summary



Advantage

- Robust to noise
- Robust to missing points of the shape
- Can be extended to lots of models



Disadvantage

- Doesn't scale well with complicated models
 - Usually works for models with less than 3 unknown parameters



RANSAC - Summary



Advantages

- Simple and general
- Usually works well in practice, even with low inlier ratio like 10%



Disadvantages

- Need to determine the inlier threshold τ
- Need large number of samples when inlier ratio is low

RANSAC最适合点云地面提取

RANSAC平面拟合

- 1 确定迭代次数 N 、inlier ratio r 和阈值 τ
- 2 对每一次迭代
 - 2.1 随机选取三个点，构建平面模型
 - 2.2 遍历所有点，计算点到平面距离
 - 2.2.1 距离小于阈值 τ 为内点
 - 2.2.2 距离大于阈值 τ 为外点
 - 2.3 内点比例达到 r 停止迭代，否则返回2继续迭代
- 3 确定使得内点数量最多的模型

Attention

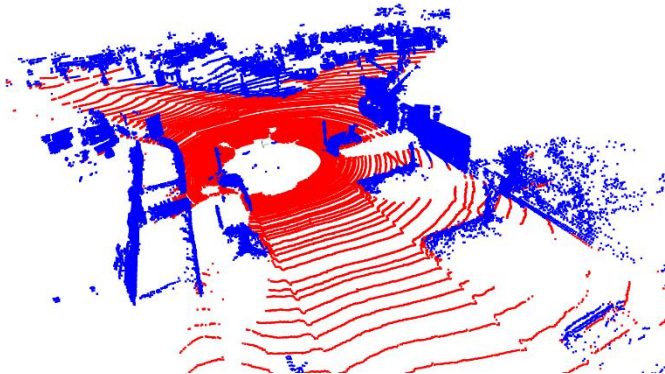
- 1 随机取三个点，共线判断
非满秩，两两组成的向量共线
- 2 平面模型
PCA，叉乘，解一元三次方程组
- 3 根据高度滤波
根据kitti的lidar安装高度，将地面范围内的所有点先提取出来，排除非地面点的干扰
- 4 平面法向量和竖直方向的夹角
三个点确定的法向量与竖直方向的夹角比较小

解题思路：地面提取

Tricks: RANSAC不能保证地面提取的很彻底

按照xyz分治，按照距离，八叉树

```
#分区
x_filter_1 = voxel_filtered_pc[:,0] >= 0.0
y_filter_1 = voxel_filtered_pc[:,1] >= 0.0
x_filter_2 = voxel_filtered_pc[:,0] < 0.0
y_filter_2 = voxel_filtered_pc[:,1] < 0.0
filter_1 = np.logical_and(x_filter_1,y_filter_1)
filter_2 = np.logical_and(x_filter_1,y_filter_2)
filter_3 = np.logical_and(x_filter_2,y_filter_2)
filter_4 = np.logical_and(x_filter_2,y_filter_1)
#分区做ransac
segmented_points_1 = ground_segmentation(data=voxel_filtered_pc[filter_1,:])
segmented_points_2 = ground_segmentation(data=voxel_filtered_pc[filter_2,:])
segmented_points_3 = ground_segmentation(data=voxel_filtered_pc[filter_3,:])
segmented_points_4 = ground_segmentation(data=voxel_filtered_pc[filter_4,:])
#合并
segmented_points = np.vstack((segmented_points_1,segmented_points_2,\
    segmented_points_3,segmented_points_4))
```



By zhp曾

解题思路：目标聚类

方法比较

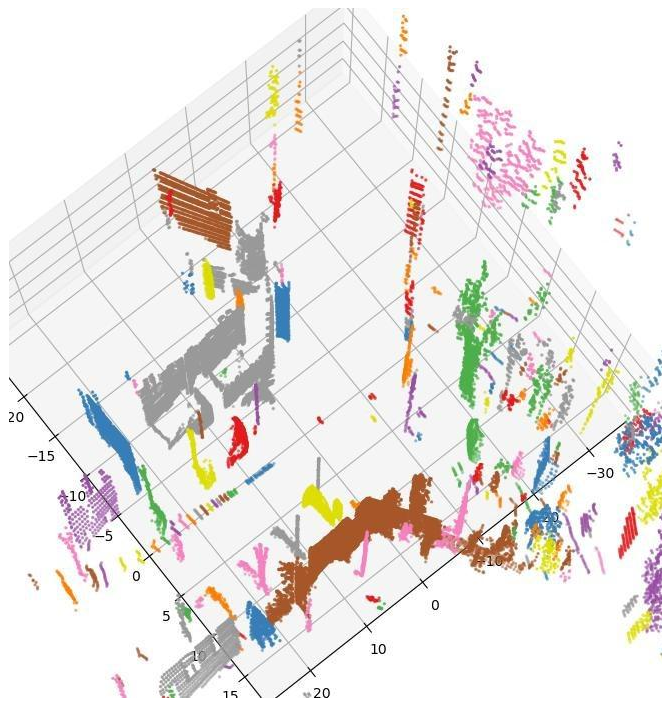
	K-Means	GMM	Spectral	Mean Shift	DBSCAN
Metric	Euclidean	Euclidean	Similarity	Density / Euclidean	Density / Euclidean
# of clusters	Pre-defined	Pre-defined	Heuristic	Automatic	Automatic
Robustness to outlier	Bad	Medium	Good	Good	Good
High dimension data	Medium	Medium	Good	Bad	Bad
Complexity	$O(t \cdot k \cdot n \cdot d)$ t: iteration k: # of clusters n: # of data d: dimension	$O(t \cdot k \cdot n \cdot d)$ t: iteration k: # of clusters n: # of data d: dimension	$O(n^3)$ n: # of data	$O(Tn \log(n))$ n: # of data T: # of centers	$O(n \cdot \log(n))$ n: # of data

解题思路：目标聚类DBSCAN

1. 将所有的点都标记为未被访问
2. 构造Kd-Tree, 确定radius和min_samples两个参数
3. 从未访问点集合中随机取一个点p, 标记p为被访问, radius-NN查找所有邻居
 - 3.1 若邻居数小于min_samples, 标记p为噪点;
 - 3.2 若邻居数大于等于min_samples, 则p为core point, 创建新簇C, 转步骤4
4. 遍历p的所有邻居n, 若n未被访问, 将n的类别标记为C, 若邻居n也为core point, 重复步骤4
5. 重复步骤3和4, 直到所有点都被访问

解题思路：目标聚类

- 1 radius和min_samples的取值, 可以通过分析kitti点云来决定
- 2 一般的同学radius设置为0.5到1.0, min_samples设置为4到10左右

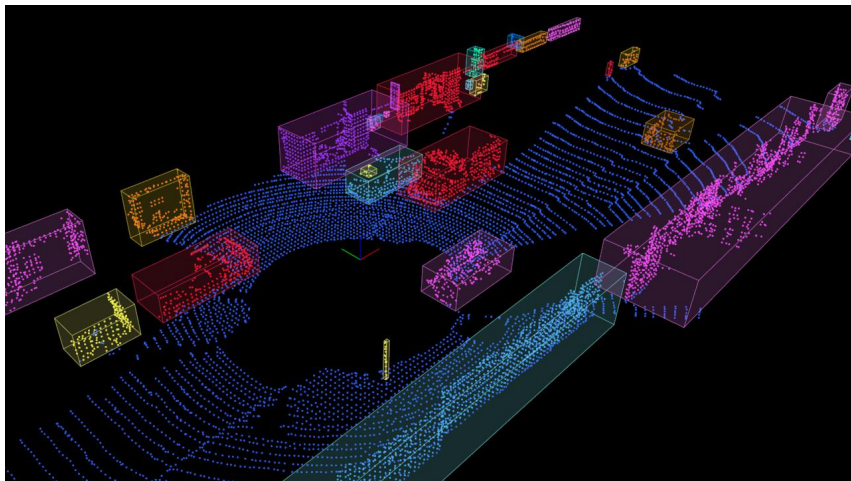


By zhp曾

解题思路：目标聚类

1 radius和min_samples的取值，可以通过分析kitti点云来决定

2 一般的同学radius设置为0.5到1.0， min_samples设置为4到10左右



Process Pipeline

1. Read the raw Lidar point-cloud from the KITTI dataset
 2. Crop the main Region of Interest (ROI) for further processing
(Centre as the Origin, $L = 60$ m, $W = 20$ m, $H = 3$ m)
 3. Down-sample the cropped point-cloud using Voxel Grid Down-sampling method, with leaf size = 0.2 m
 4. Segment the ground plane using the RANSAC, with a max number of iterations of 100 and a threshold of 0.2 m
 5. Cluster the foreground points into a series of small clusters and assign them an unique ID & a random color
 6. Draw a Bounding Box for each of the clusters
- **Additional: Track the same object across frame and keep its color consistent****
7. Use a simple Hungarian method with Bounding Box IOU as a gauge to associate objects between two adjacent time frames (e.g. $t = k$, $t = k+1$)
 8. For the new objects clustered by ****Step 6**** at $t = k+1$, if it has a corresponding object at $t = k$, it will be assigned with that ID and painted with the same color as its counterpart

Q&A



感谢各位聆听 !
Thanks for Listening

