OpenDriveLab

上海人工智能实验室
Shanghai Artificial Intelligence Laboratory

Perception - LiDAR

Dr. Hongyang Li

Shanghai AI Lab
Mar 27 2024

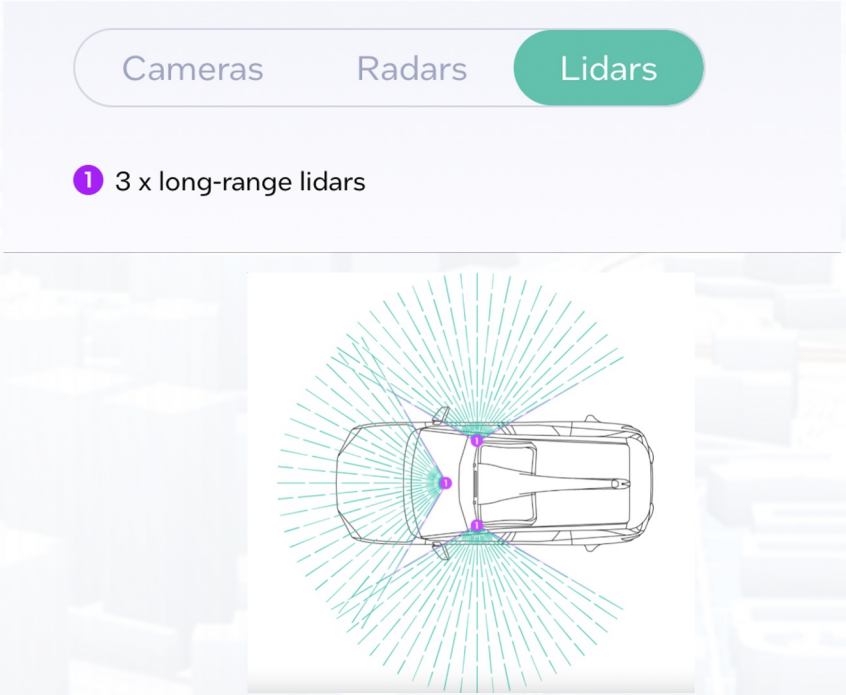
Slides credit to Cheng'en and Zetong,
members from OpenDriveLab

Outline

- **概述**
  - 什么是激光雷达 & 为什么要用激光雷达
  - 什么是点云
- **如何利用雷达点云**
  - 基于体素的点云感知算法
  - 基于点表征的点云感知算法
  - 基于Range-view的感知算法
- **如何进一步解决雷达点云缺陷**
  - 多帧点云算法
  - 融合技术概述

https://www.mobileye.com/ces-2024/

回顾 | LiDAR



LiDAR全称**Light Detection and Ranging。**激光雷达通过各个方向发射激光脉冲，波长在 **纳米** 范围，这些脉冲在抵达物体表面之后反射回来，并被接收器接收。

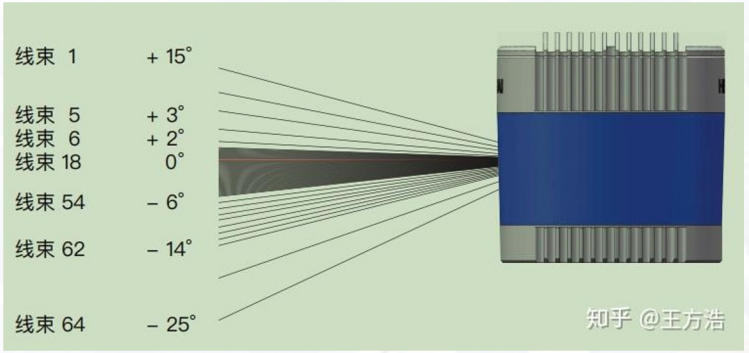
- LiDAR波长较短，相比Radar更加精准，可以检测**更小的**物体
- 同时由于波长较短，很容易收到空气介质中的各种杂质干扰，**恶劣天气下效果影响很大**

回顾 | LiDAR

LiDAR在实际应用中，有两个常见的参数：

- **垂直方向上**，激光收发模块的数量被称为线数。**线数越高，扫描结果越精确。**常用的线数有16线，32线，64线。
- **水平方向上**，由于激光雷达在旋转扫描，扫描的点数和激光雷达的扫描频率有一定的关系，这个参数一般被称为水平分辨率。

禾赛64线激光雷达光束分布



Credit to FangHao Wang @ Zhihu

	Velodyne					Hesai		Ouster		RoboSense
	VLS-128*	HDL-64S2	HDL-32E	VLP-32c	VLP-16	Pandar64	Pandar40p	OS1-64	OS1-16	RS-LiDAR-32
Channels	128	64	32	32	16	64	40	64	16	32
FPS[Hz]	5-20	5-20	5-20	5-20	5-20	10,20	10,20	10,20	10,20	5,10,20
Precision[m]	±0.03	±0.02[a]	±0.02	±0.03	±0.03	±0.02[c]	±0.02[c]	±0.03[d]	±0.03[d]	±0.03[c]
Max.Range[m]	300	120	100	200	100	200	200	120	120	200
Min.Range[m]	N/A	3	2	1	1	0.3	0.3	0.8	0.8	0.4
vFOV[deg]	40	26.9	41.33	40	30	40	40	33.2	33.2	40
vRes[deg]	0.11[b]	0.33[b]	1.33	0.33[b]	2.0	0.167[b]	0.33[b]	0.53	0.53	0.33[b]
hRes[deg]10hz	0.2	0.16	0.16	0.2	0.2	0.2	0.2	0.35	0.35	0.2
λ[nm]	903	903	903	903	903	905	905	850	850	905
d[mm]	165.5	223.5	85.3	103	103.3	116	116	85	85	114
Weight(kg)	3.5	13.5	1.0	0.925	0.830	1.52	1.52	0.425	0.425	1.17
Firmware Ver.	—[e]	4.07	2.1.7.1	N/A	3.0.29.0	5.10	4.29	—[f]	1.12.0	1.12.0
Price[g][USD]	$$$$	$$$	$$	$$	$	$$$	$$$	$	$	$$

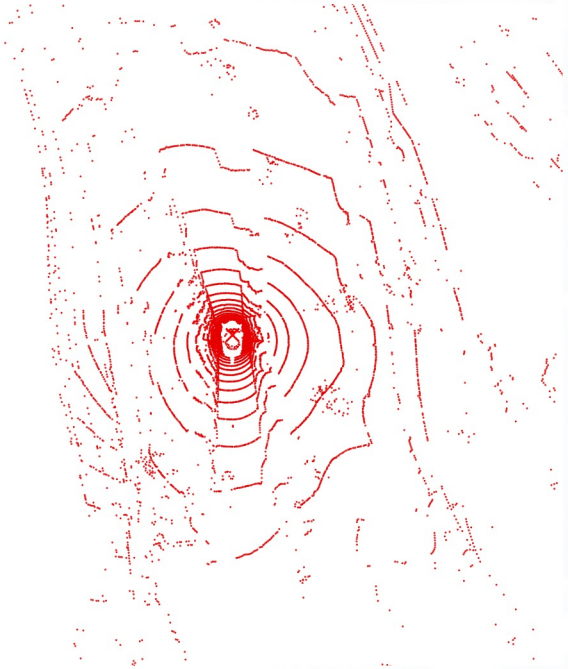
10款主流的旋转式机械激光雷达在自驾场景中的对比

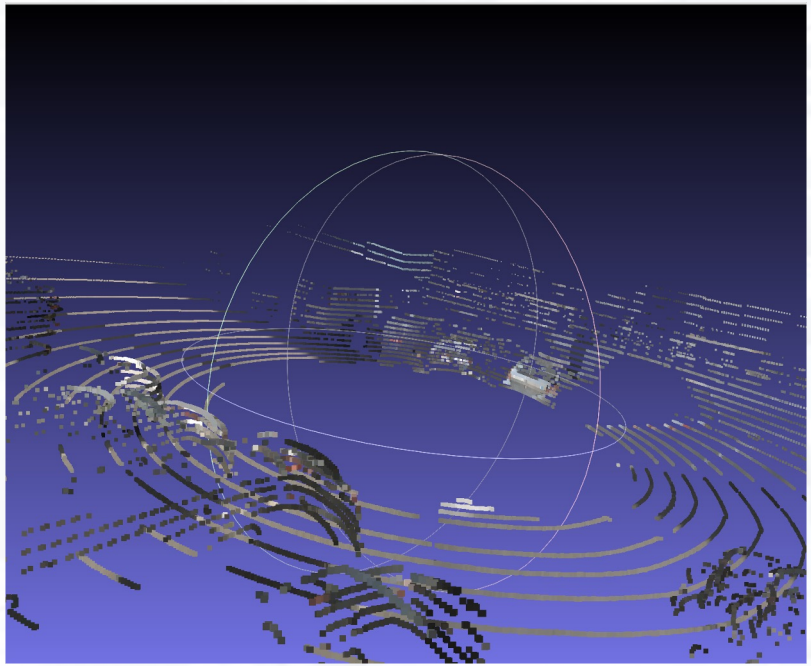
https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9142208

LiDAR Point Cloud

点云中每个点的参数，一般为X，Y，Z，R，G，B

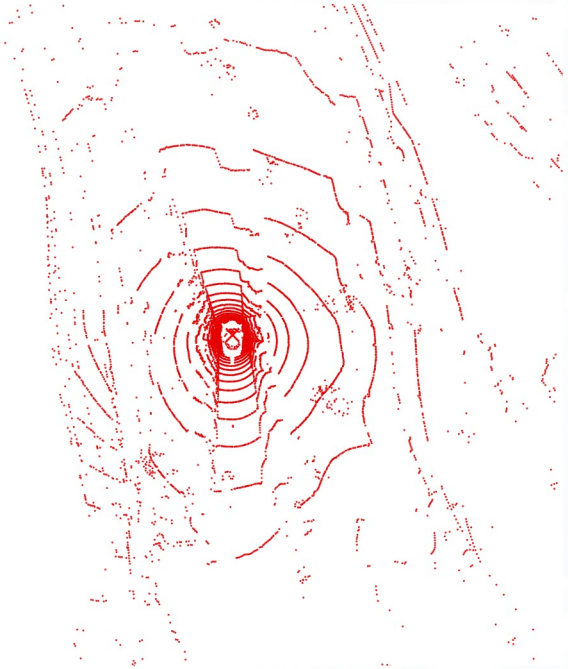
，（Refelctance）

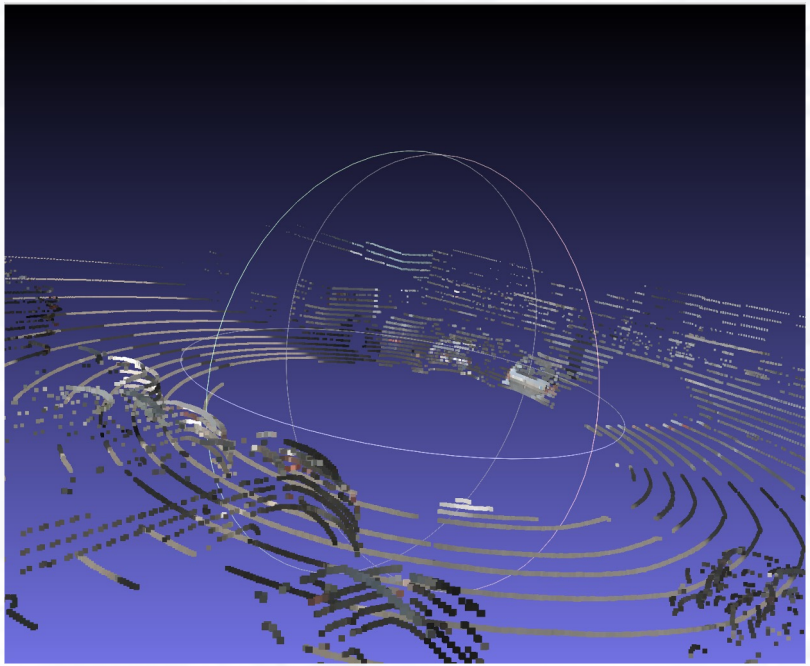




LiDAR Point Cloud

点云数据与图像数据相比具有**稀疏性，不规则性，无序性**的特点。





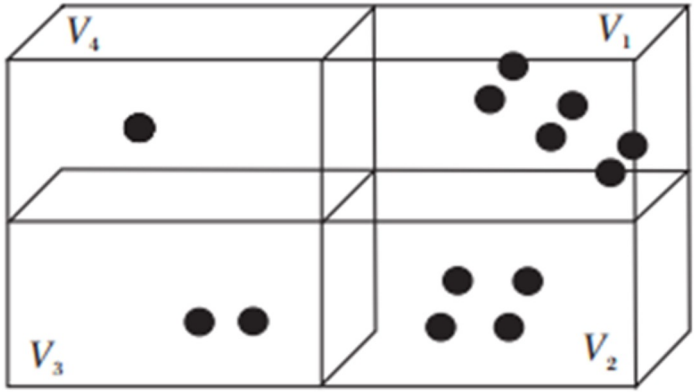
LiDAR Perception Methods

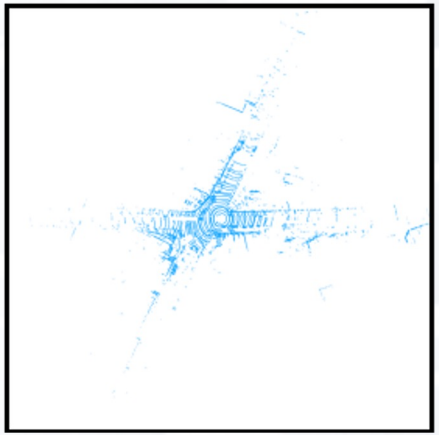
根据以上点云特点，有三种主要的研究思路：

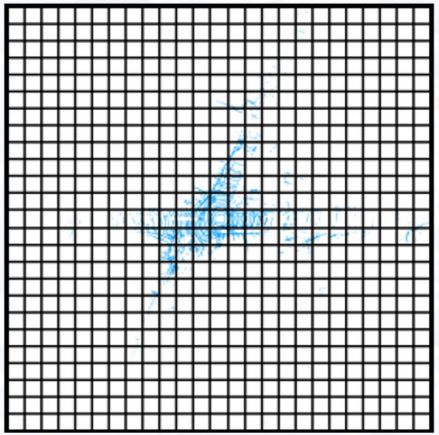
- 基于体素的感知
- 基于点的感知
- 基于Range-view的感知

基于体素的感知方法

为了解决点云的稀疏性，无序性，最简单的策略就是让点云数据变得密集且有序。







基于体素的感知方法 (VoxNet)

https://graphics.stanford.edu/courses/cs233-21-spring/ReferencedPapers/voxnet_07353481.pdf

Congress Center Hamburg
Sept 28 - Oct 2, 2015. Hamburg, Germany

VoxNet: A 3D Convolutional Neural Network for Real-Time Object Recognition

Daniel Maturana and Sebastian Scherer

基于体素的感知方法 (VoxNet)

https://graphics.stanford.edu/courses/cs233-21-spring/ReferencedPapers/voxnet_07353481.pdf

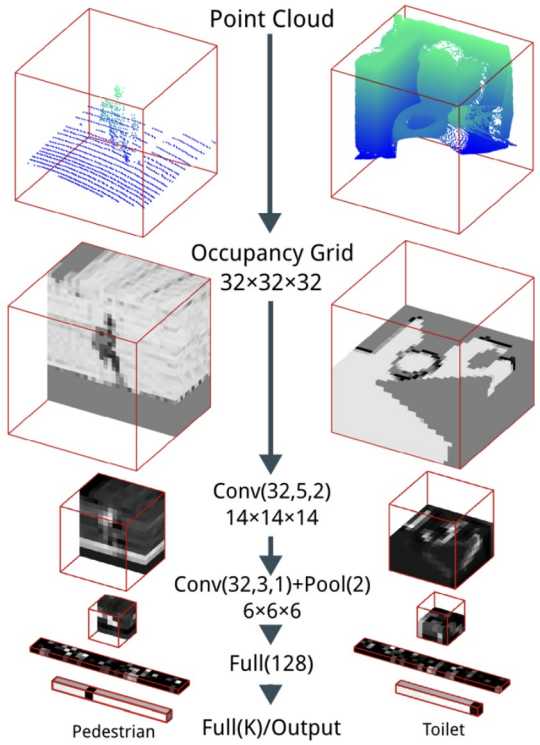


TABLE IV

COMPARISONS WITH SHAPENET IN MODELNET (AVG ACC)

Dataset	ShapeNet	VoxNet
ModelNet10	0.84	**0.92**
ModelNet40	0.77	**0.83**

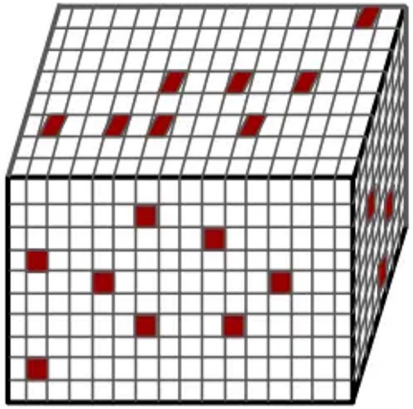
TABLE V

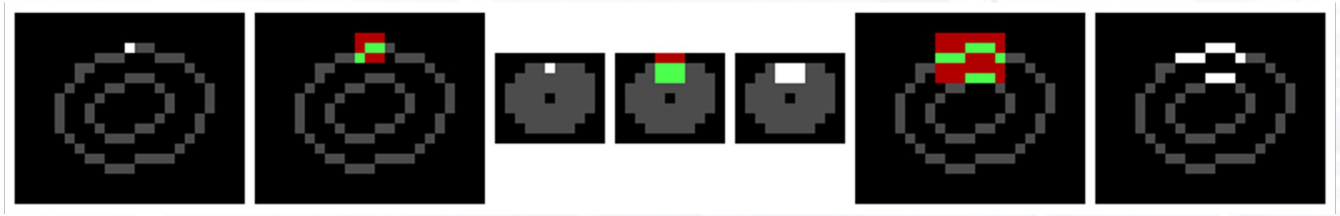
COMPARISON WITH SHAPENET IN NYUV2 (AVG ACC)

Dataset	ShapeNet	VoxNet	VoxNet Hit
NYU	0.58	**0.71**	0.70
ModelNet10→NYU	**0.44**	0.34	0.25

基于体素的感知方法 (SparseConv)

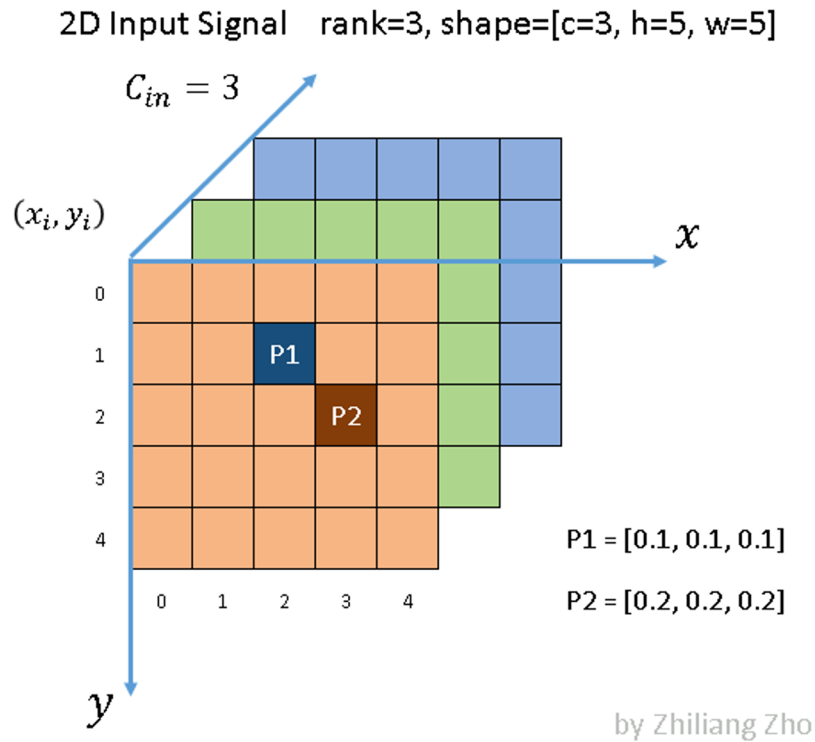
由于点云稀疏性的特点，是否能有效地计算稀疏数据的卷积，而不是扫描所有的图像像素或空间体素？否则空白区域会带来大量冗余的计算量。





基于体素的感知方法 (SparseConv)

An Example



以左侧的稀疏图像作为输入，除了P1和 P2两点外，所有像素都是(0,0,0)。这种非零的元素也称为**active input sites**。

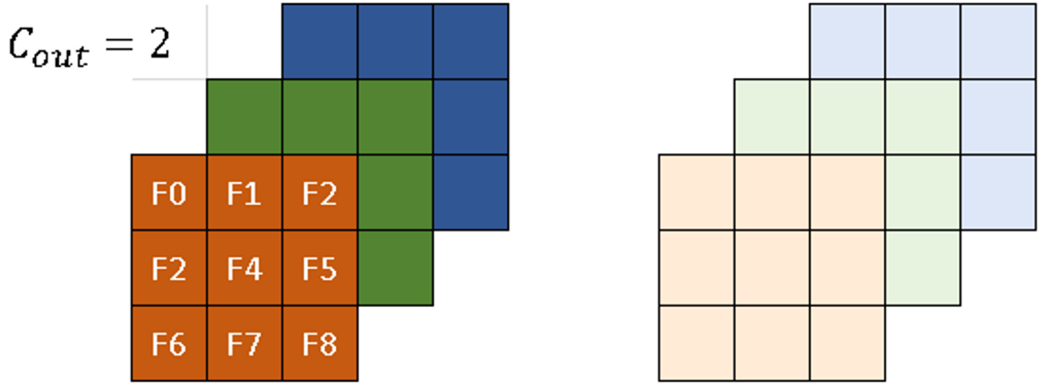
在稀疏格式中，数据列表是[[0.1,0.1,0.1]，[0.2,0.2,0.2]，索引列表是[1,2]，[2,3]，按照YX 顺序。

基于体素的感知方法 (SparseConv)

An Example

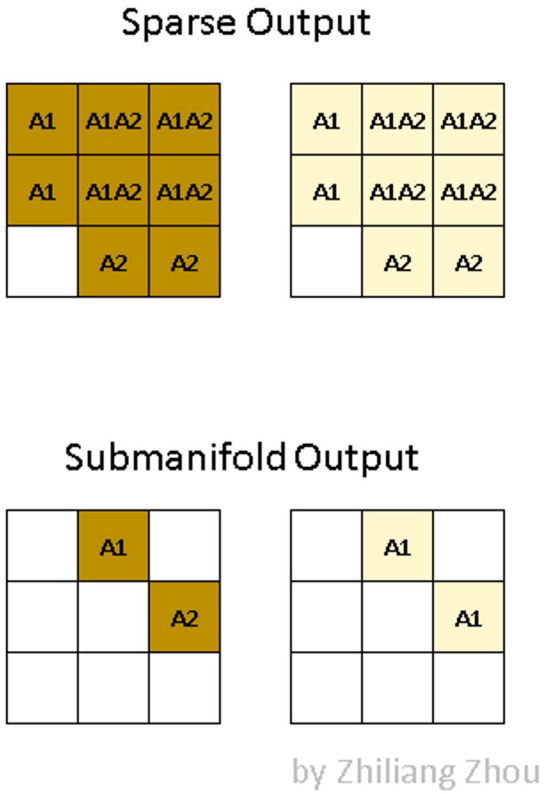
稀疏卷积的卷积核与传统的卷积核相同。本例中我们使用的卷积核，其内核大小为3x3

$Kernel \rightarrow C_{in} \times 3 \times 3 \times C_{out}$ with stride=1 and pad=0



基于体素的感知方法 (SparseConv)

An Example



稀疏卷积的输出与传统的卷积有很大的不同。一般用两种常用的输出：

- 第一种是 regular output definition，就像普通的卷积一样，只要kernel 覆盖一个 active input site，就计算出 output site。
- 第二种是submanifold output definition。只有当kernel的中心覆盖一个 active input site时，卷积输出才会被计算。

基于体素的感知方法 (VoxelNet)

https://arxiv.org/pdf/1711.06396.pdf

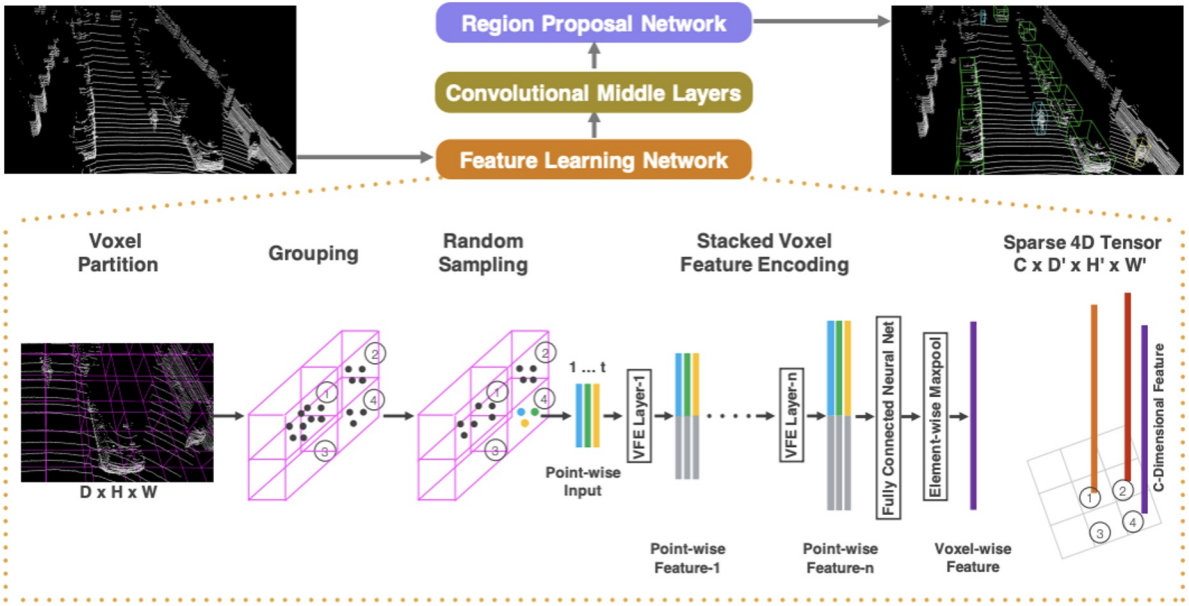
VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection

Yin Zhou
Apple Inc
yzhou3@apple.com

Oncel Tuzel
Apple Inc
otuzel@apple.com

基于体素的感知方法 (VoxelNet)

https://arxiv.org/pdf/1711.06396.pdf

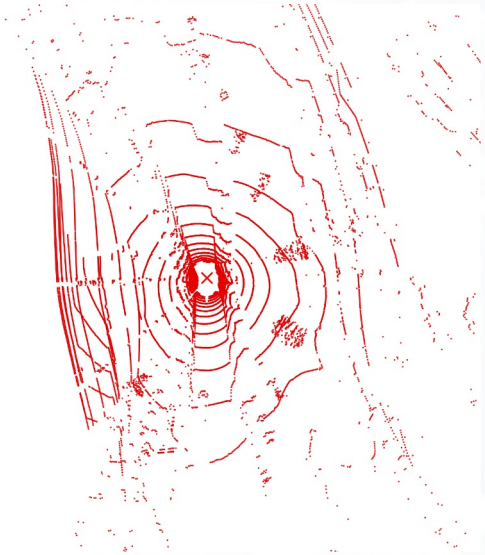


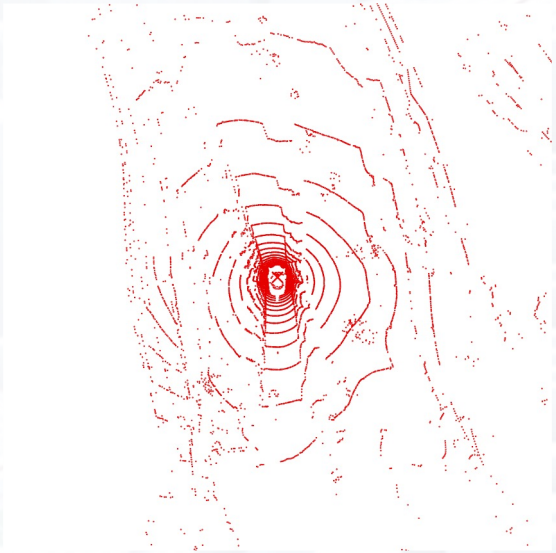
整体Pipeline分为三个部分：

- 将点云数据Voxelize之后，对每一个非空Voxel使用若干个**VFE(Voxel Feature Encoding)**层进行局部特征提取
- 然后经过3D Convolutional Middle Layers进一步抽象特征
- 最后使用RPN(Region Proposal Network)对物体进行分类检测与位置回归。

基于点的感知方法

基于体素的方法不仅性能较优，计算速度也较可观，尤其是稀疏卷积的发展，促进体素方法的应用。但是，基于体素的方法受设置参数的影响，不可避免地丢失一部分点云信息





基于点的感知方法 (PointNet)

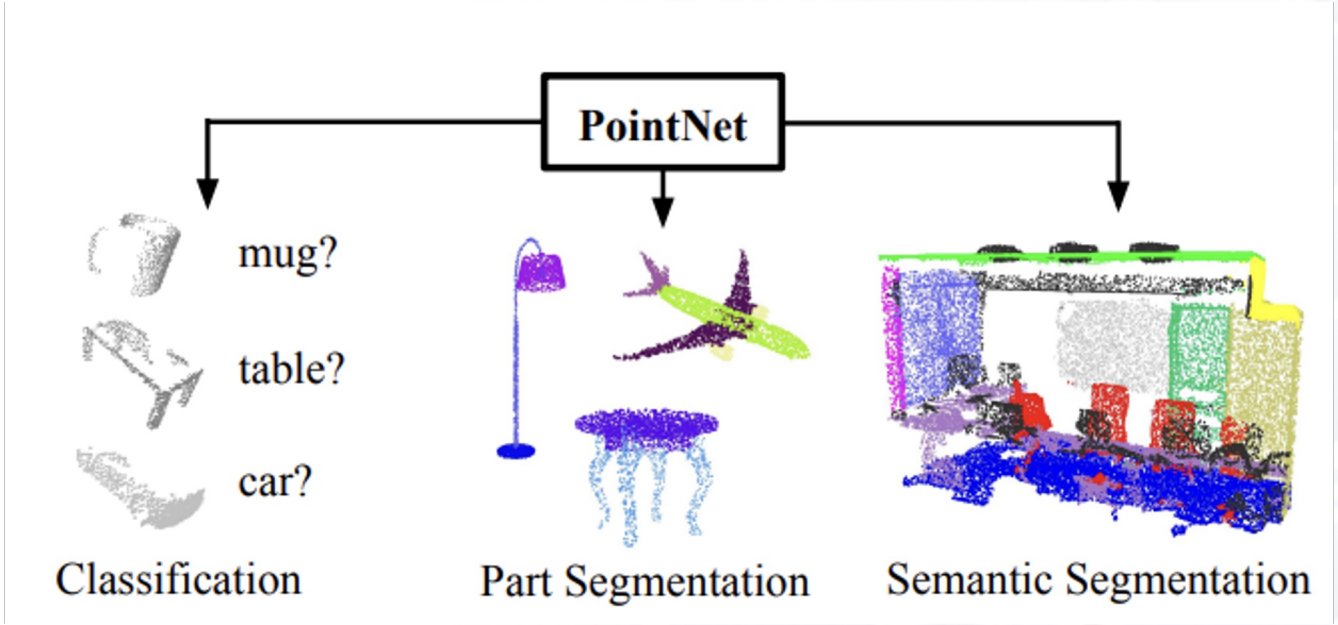
https://arxiv.org/abs/1612.00593

PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation

Charles R. Qi* Hao Su* Kaichun Mo Leonidas J. Guibas
Stanford University

基于点的感知方法 (PointNet)

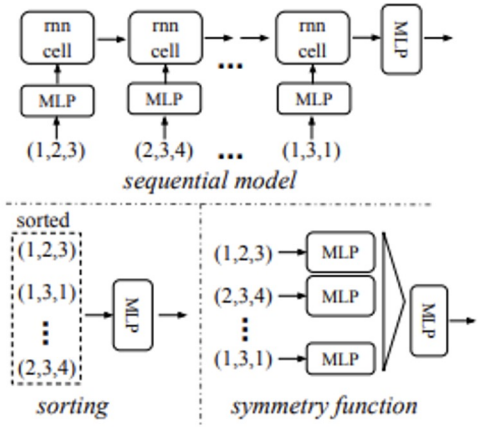
https://arxiv.org/abs/1612.00593



PointNet直接将纯点云作为输入，输出为整个点云的类标签（classification）或者对每个点的标签（part/ semantic segmentation）。

基于点的感知方法 (PointNet)

https://arxiv.org/abs/1612.00593



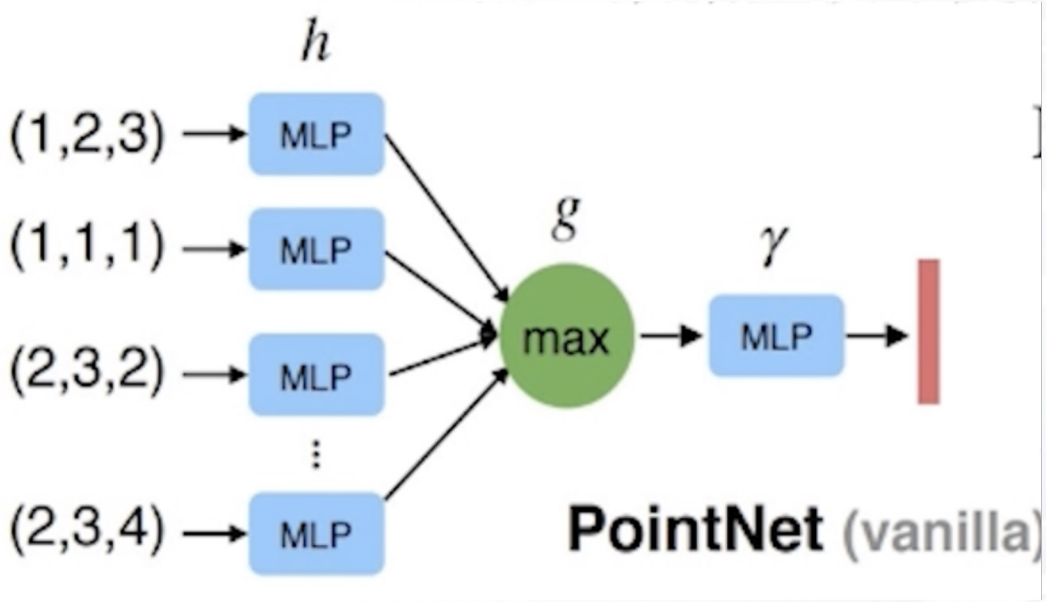
	accuracy
MLP (unsorted input)	24.2
MLP (sorted input)	45.0
LSTM	78.5
Attention sum	83.0
Average pooling	83.8
Max pooling	**87.1**

直接输入原始点云数据存在一个问题：**点云顺序不同结果应该一致。**一般有三种解决的方式：

- 按照一定规则对点集进行排序
- 将点集的所有排列作为增强数据，训练一个循环网络
- 利用一个对称函数将所有信息进行聚合

基于点的感知方法 (PointNet)

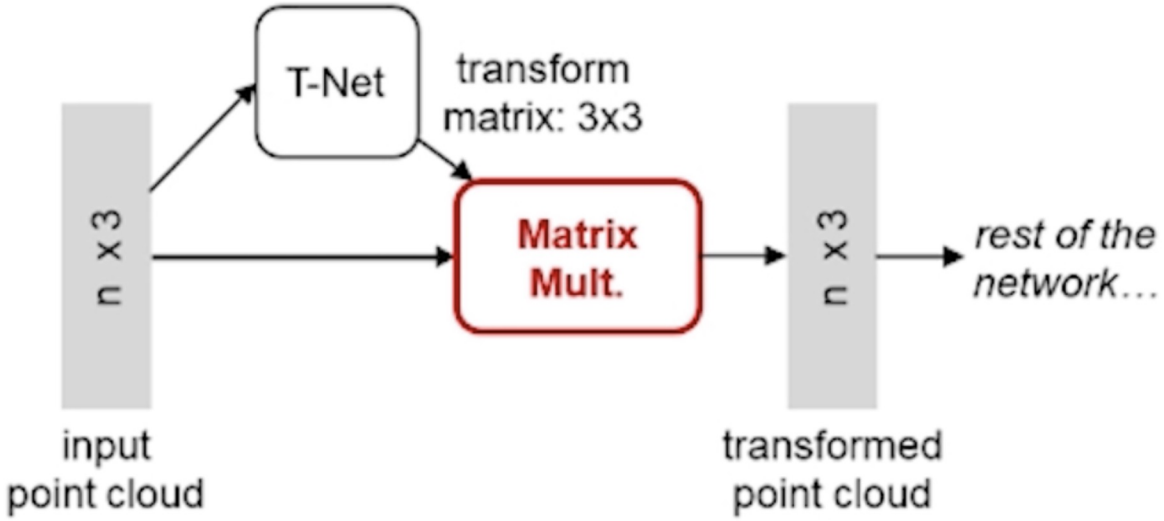
https://arxiv.org/abs/1612.00593



PointNet最终选择采用多层感知机（MLP）和最大池化（Max Pooling）

基于点的感知方法 (PointNet)

https://arxiv.org/abs/1612.00593

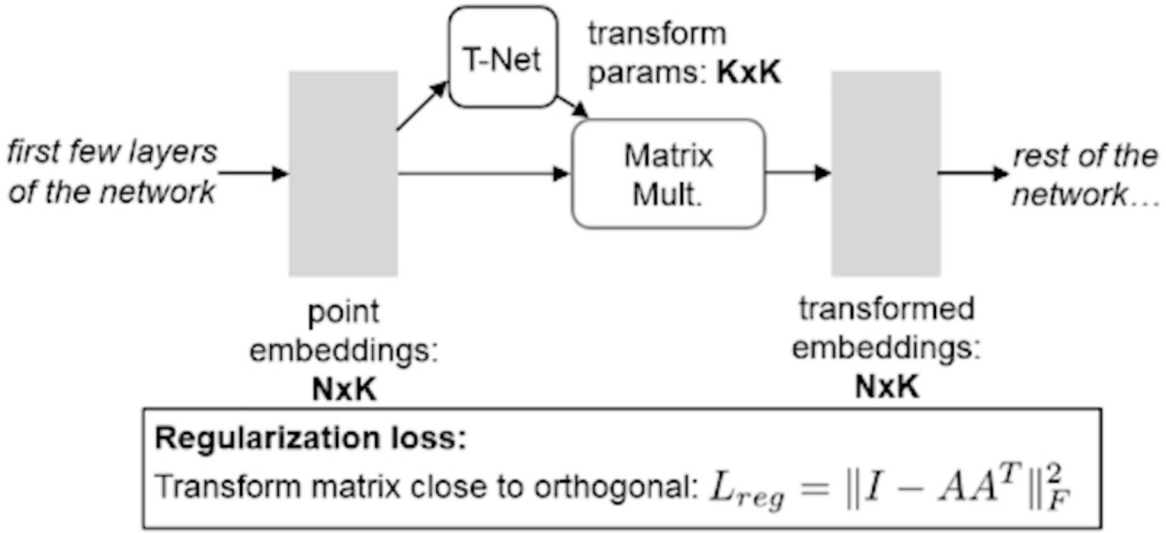


直接输入原始点云数据的另一个问题是：**旋转后分类结果应该一致。**

因此PointNet引入T-Net得到一个旋转矩阵，对输入特征进行自动对齐。

基于点的感知方法 (PointNet)

https://arxiv.org/abs/1612.00593

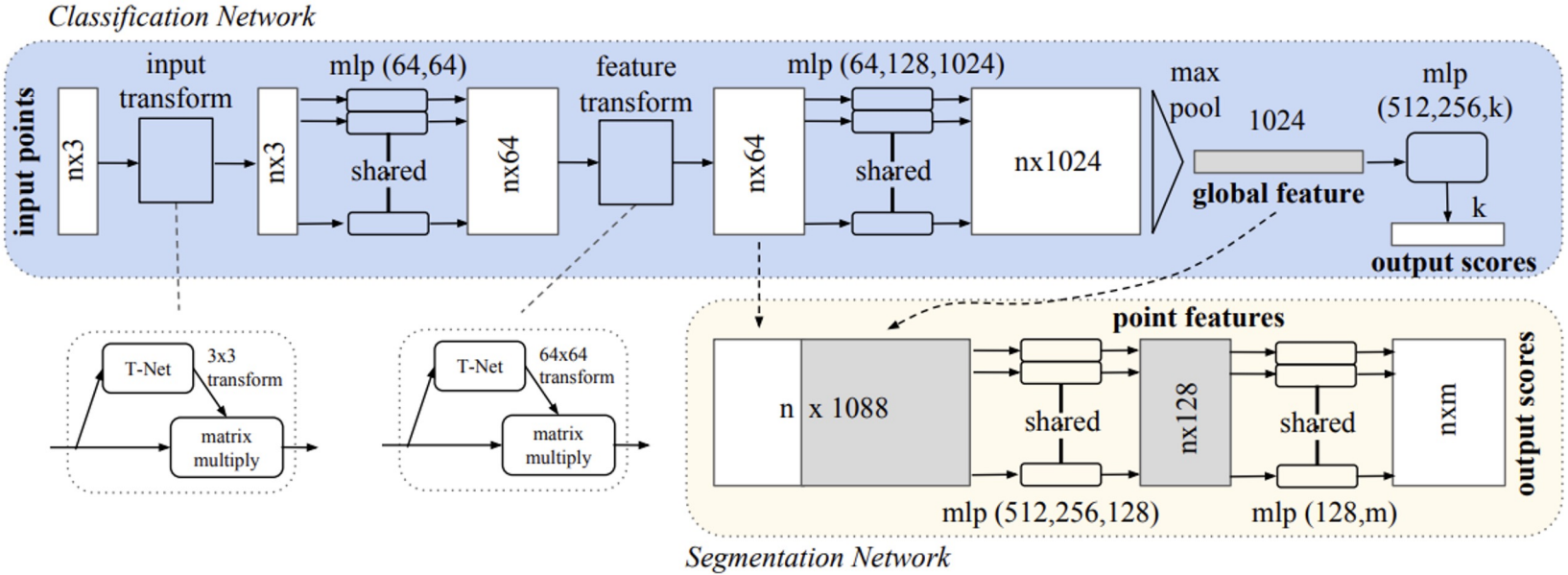


并且，PointNet将正则化项添加到最终的softmax训练损失中，即将特征变换矩阵约束为接近正交矩阵

基于点的感知方法 (PointNet)

https://arxiv.org/abs/1612.00593

PointNet整体结构



基于点的感知方法 (PointNet)

https://arxiv.org/abs/1612.00593

PointNet分类结果

	input	#views	accuracy avg. class	accuracy overall
SPH [11]	mesh	-	68.2	-
3DShapeNets [28]	volume	1	77.3	84.7
VoxNet [17]	volume	12	83.0	85.9
Subvolume [18]	volume	20	86.0	**89.2**
LFD [28]	image	10	75.5	-
MVCNN [23]	image	80	**90.1**	-
Ours baseline	point	-	72.6	77.4
Ours PointNet	point	1	86.2	**89.2**

基于点的感知方法 (PointNet)

https://arxiv.org/abs/1612.00593

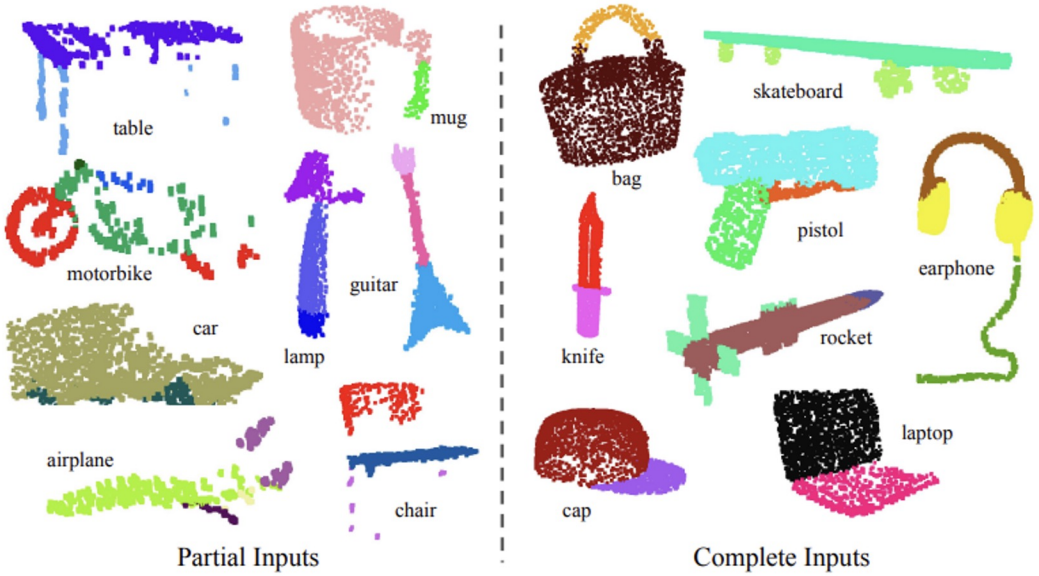
PointNet部件分割结果

	mean	aero	bag	cap	car	chair	ear phone	guitar	knife	lamp	laptop	motor	mug	pistol	rocket	skate board	table
# shapes		2690	76	55	898	3758	69	787	392	1547	451	202	184	283	66	152	5271
Wu [27]	-	63.2	-	-	-	73.5	-	-	-	74.4	-	-	-	-	-	-	74.8
Yi [29]	81.4	81.0	78.4	77.7	**75.7**	87.6	61.9	**92.0**	85.4	**82.5**	**95.7**	**70.6**	91.9	**85.9**	53.1	69.8	75.3
3DCNN	79.4	75.1	72.8	73.3	70.0	87.2	63.5	88.4	79.6	74.4	93.9	58.7	91.8	76.4	51.2	65.3	77.1
Ours	**83.7**	**83.4**	**78.7**	**82.5**	74.9	**89.6**	**73.0**	91.5	**85.9**	80.8	95.3	65.2	**93.0**	81.2	**57.9**	**72.8**	**80.6**

基于点的感知方法 (PointNet)

https://arxiv.org/abs/1612.00593

PointNet部件分割结果



基于点的感知方法 (PointNet++)

http://stanford.edu/~rqi/pointnet2/

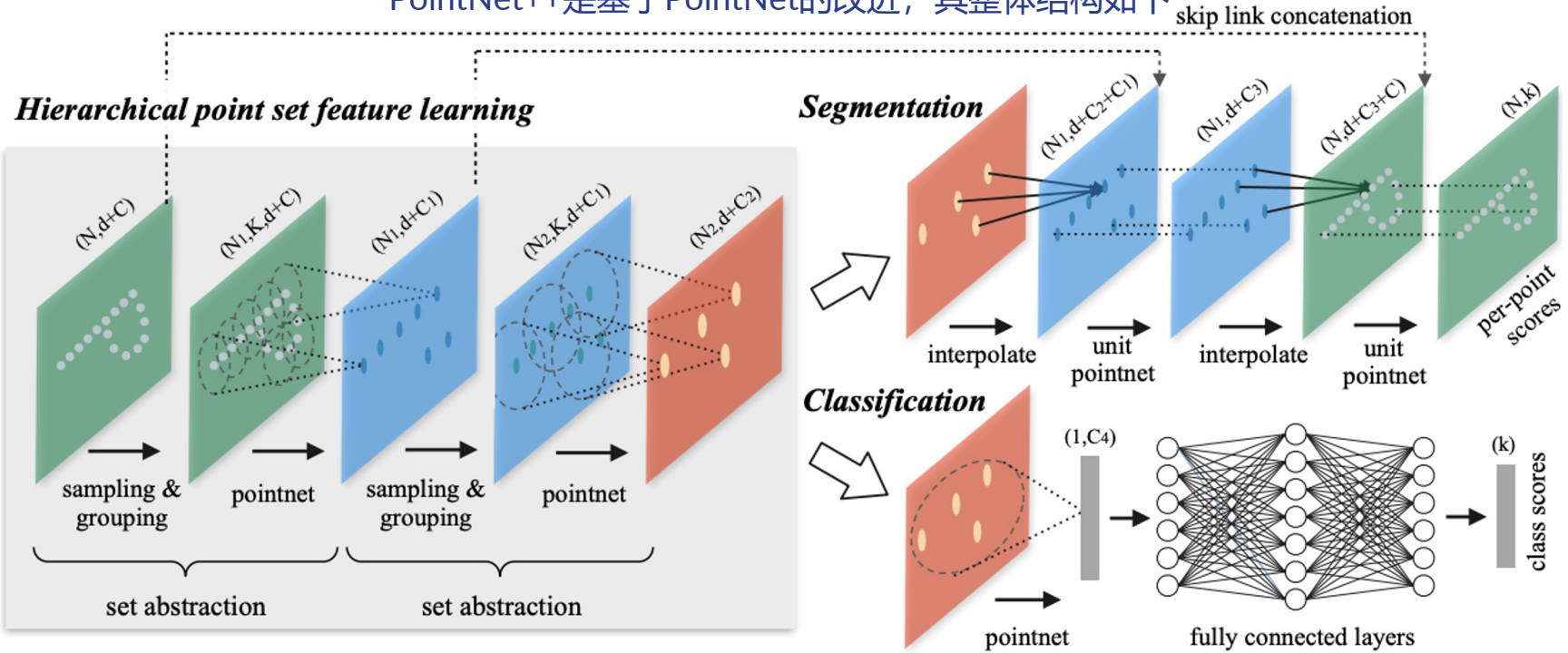
PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space

Charles R. Qi Li Yi Hao Su Leonidas J. Guibas
Stanford University

基于点的感知方法 (PointNet++)

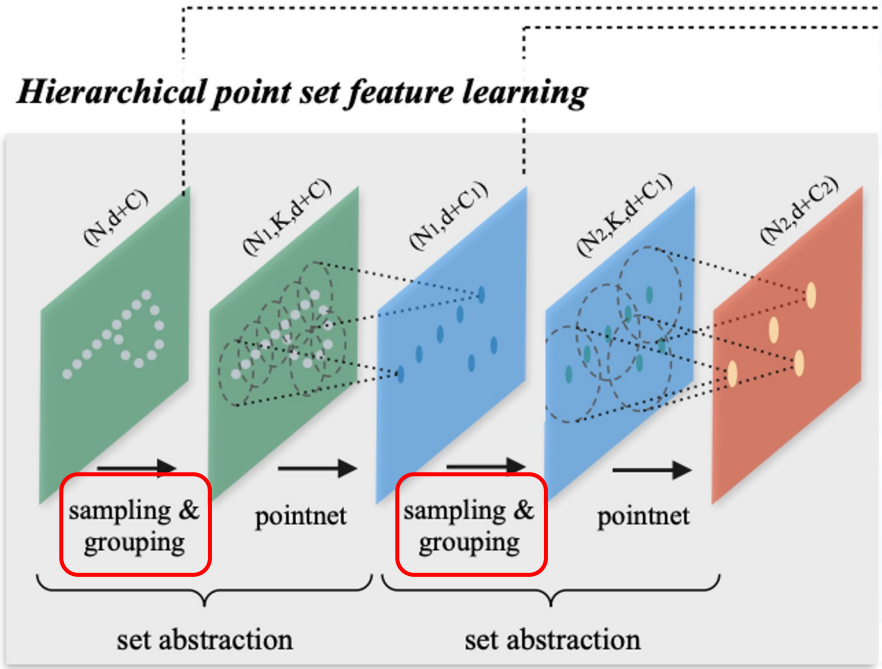
http://stanford.edu/~rqi/pointnet2/

PointNet++是基于PointNet的改进，其整体结构如下



基于点的感知方法 (PointNet++)

http://stanford.edu/~rqi/pointnet2/

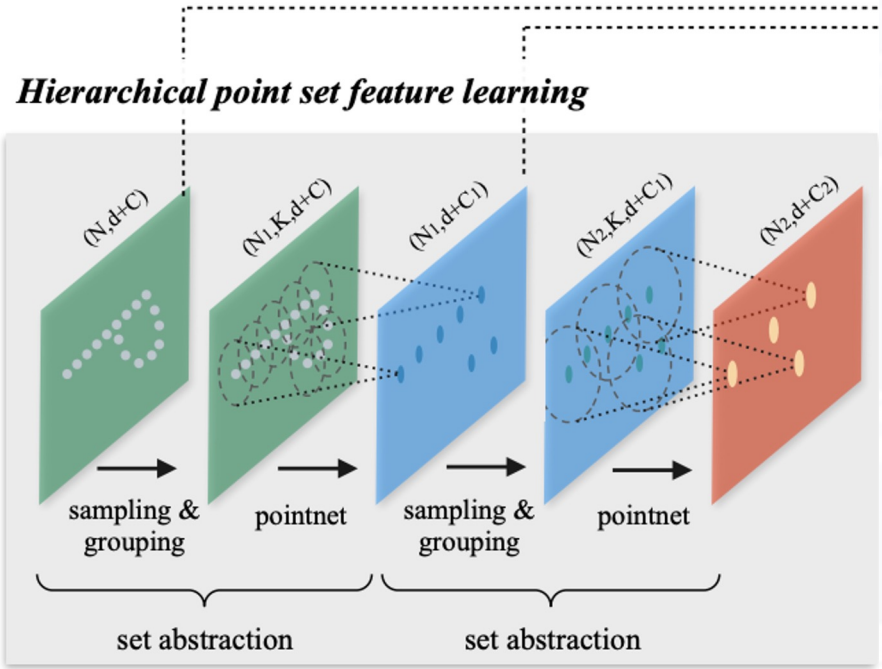


PointNet特征提取时只考虑单点，不能很好的表示局部结构

因此，PointNet++引入了**sampling & grouping**，考虑局部领域特征

基于点的感知方法 (PointNet++)

http://stanford.edu/~rqi/pointnet2/

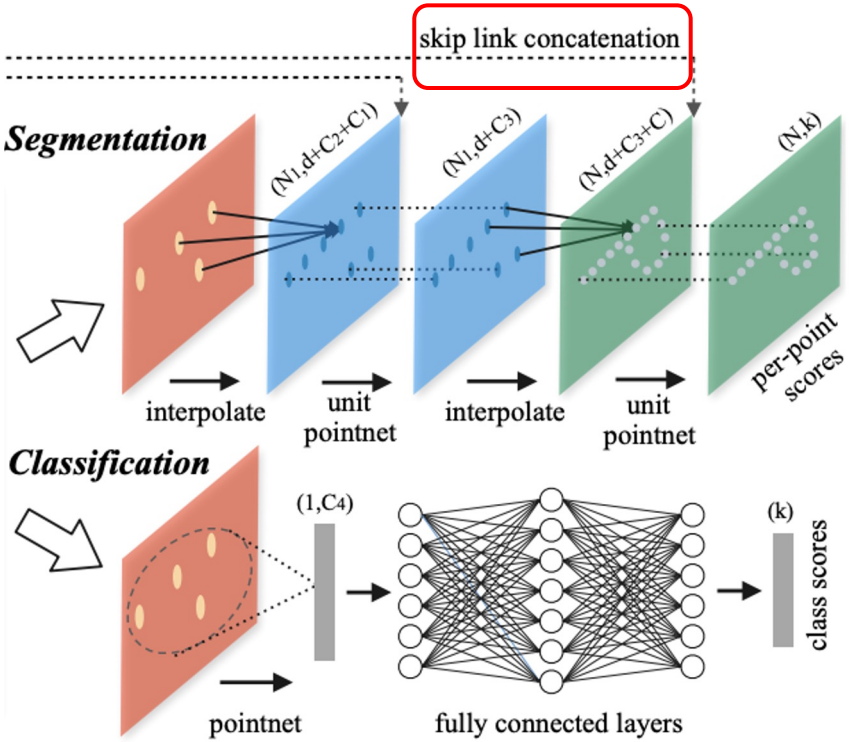


PointNet中global feature直接由max pool得到，容易造成信息丢失 。

因此PointNet++采用层级结构，可以有效的依据不同的感受野大小来提取不同区域的局部特征。

基于点的感知方法 (PointNet++)

http://stanford.edu/~rqi/pointnet2/



PointNet分割任务的全局特征global feature是直接复制与local feature拼接，生成discriminative feature能力有限。

因此，PointNet++在分割任务中设计了encoder-decoder结构，先降采样再上采样，使用skip connection将对应层的local-global feature拼接

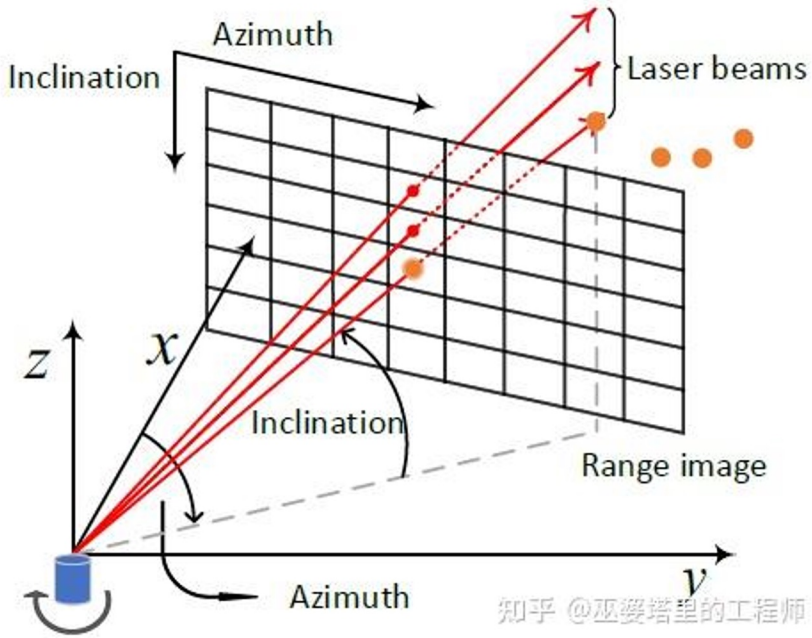
基于Range-view的感知方法

Range-view方法是运用点云在2D图像上的表示形式，可以轻松使用一些2D方法。

(a) Range of LiDAR points (RV)

(b) Intensity of LiDAR points (RV)

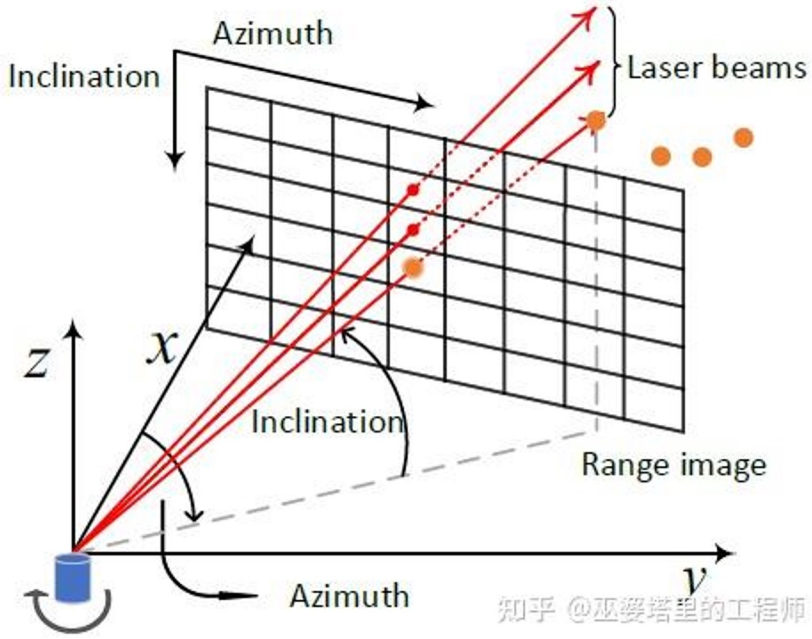
基于Range-view的感知方法



激光雷达的点云来自于多条激光扫描线。例如64线的激光雷达，那么在**垂直方向（Inclination）**上就有64个离散的角度。

激光雷达在FOV内扫描一遍，会有多个**水平方向（Azimuth）**的角度。比如说水平分辨率是0.2°，那么扫描360°就会产生1800个离散的角度。

基于Range-view的感知方法



这里也可以粗略把Inclination和Azimuth类比为地球上的纬度和经度。把水平和垂直方向的角度值作为X-Y坐标，就可以得到一个二维图像。

图像中的像素值是相应角度下的反射点的特性，比如距离，反射强度等。这些特性可以作为图像的channel，类似于可见光图像中的RGB。

基于Range-view的感知方法 (RSN)

https://arxiv.org/abs/2106.13365

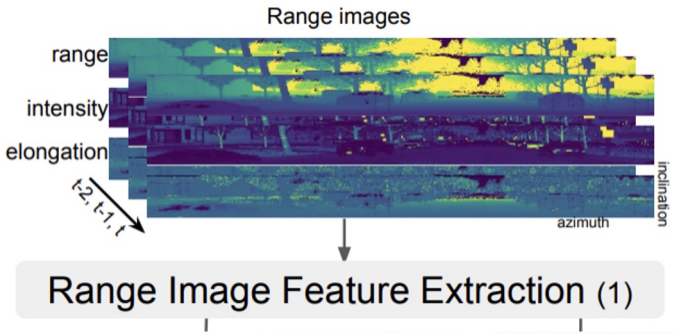
RSN: Range Sparse Net for Efficient, Accurate LiDAR 3D Object Detection

Pei Sun[1] Weiyue Wang[1] Yuning Chai[1] Gamaleldin Elsayed[2]
Alex Bewley[2] Xiao Zhang[1] Cristian Sminchisescu[2] Dragomir Anguelov[1]

[1]Waymo LLC, [2]Google
peis@waymo.com

基于Range-view的感知方法 (RSN)

https://arxiv.org/abs/2106.13365

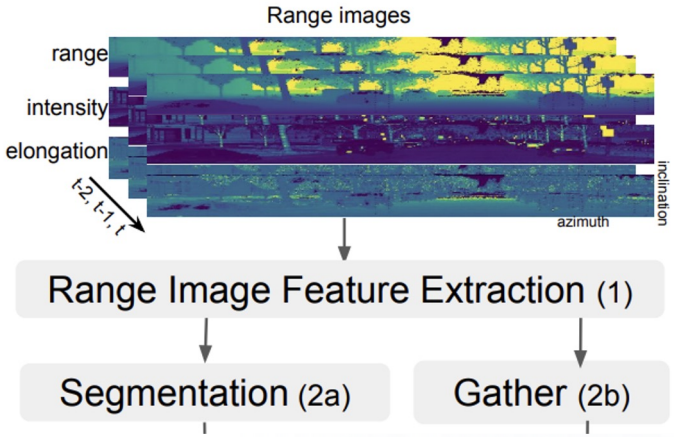


RSN整体结构分为五个部分：

1. **Range image feature extraction**: 在Range-view图像上使用二维卷积网来提取相关的图像特征。

基于Range-view的感知方法 (RSN)

https://arxiv.org/abs/2106.13365

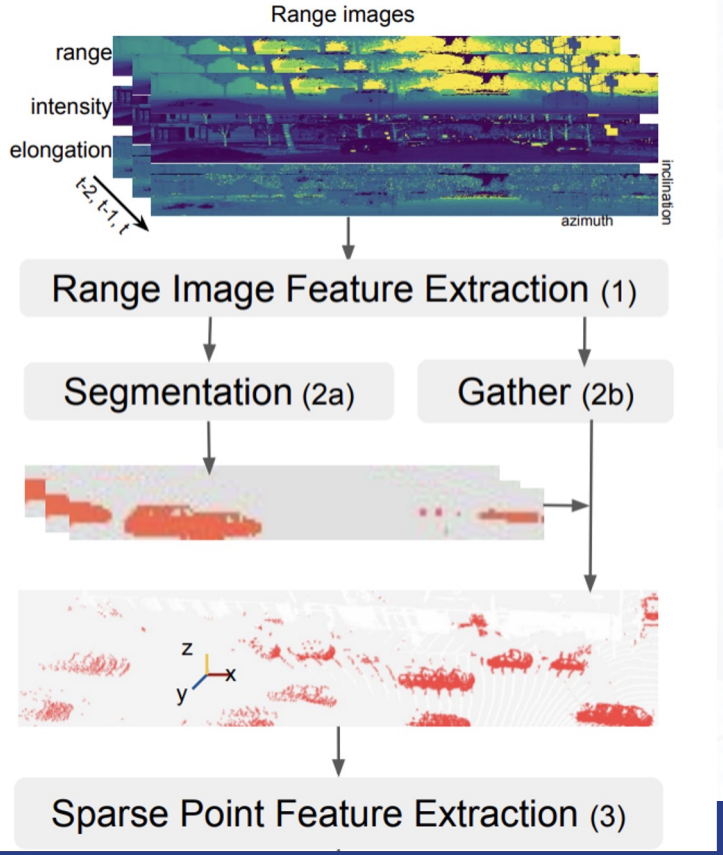


RSN整体结构分为五个部分：

1. **Range image feature extraction**: 在Range-view图像上使用二维卷积网来提取相关的图像特征
2. **Foreground point selection**：在2a) 中对Range-view图像上的前景点进行分割；将前景点与学习到的Range-view图像图像特征一起在2b) 中汇聚成稀疏点。

基于Range-view的感知方法 (RSN)

https://arxiv.org/abs/2106.13365

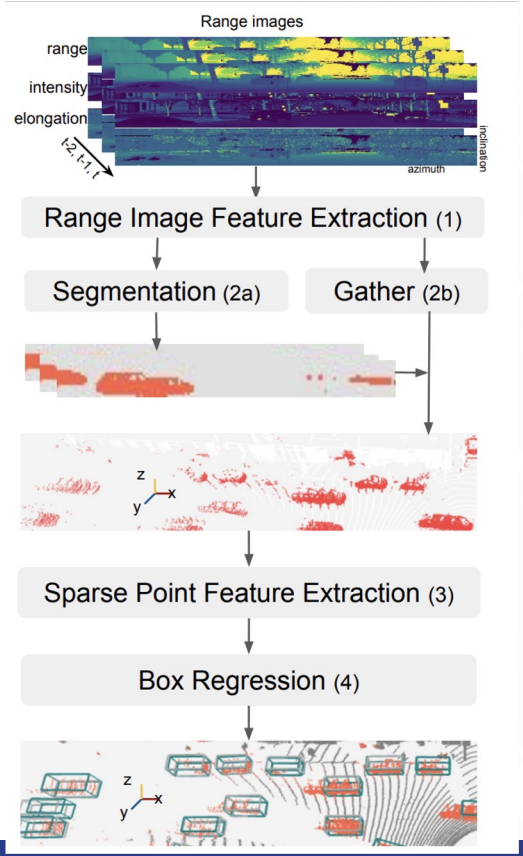


RSN整体结构分为五个部分：

1. **Range image feature extraction**: 在Range-view图像上使用二维卷积网来提取相关的图像特征
2. **Foreground point selection**：在2a) 中对Range-view图像上的前景点进行分割；将前景点与学习到的Range-view图像图像特征一起在2b) 中汇聚成稀疏点。
3. **Sparse point feature extraction**: 通过稀疏卷积在选定的前景点上提取每个点特征。

基于Range-view的感知方法 (RSN)

https://arxiv.org/abs/2106.13365



RSN整体结构分为五个部分：

1. **Range image feature extraction**: 在Range-view图像上使用二维卷积网来提取相关的图像特征
2. **Foreground point selection**：在 **2a)** 中对Range-view图像上的前景点进行分割；将前景点与学习到的Range-view图像图像特征一起在 **2b)** 中汇聚成稀疏点。
3. **Sparse point feature extraction**: 通过稀疏卷积在选定的前景点上提取每个点特征。
4. **A sparse CenterNet head to regress boxes.**

基于Range-view的感知方法 (RSN)

https://arxiv.org/abs/2106.13365

Performance comparisons on the Waymo Open Dataset validation set for **vehicle detection**

Method	Latency (ms)	AP/APH L1 BEV	AP/APH L1 3D	AP/APH L2 BEV	AP/APH L2 3D	AP/APH L1 3D by distance <30m	30-50m	> 50m
LaserNet CVPR'19 [21] *	64.3	71.2/67.7	52.1/50.1	-	-	70.9/68.7	52.9/51.4	29.6/28.6
P.Pillars CVPR'19[16] †	49.0	82.5/81.5	63.3/62.7	73.9/72.9	55.2/54.7	84.9/84.4	59.2/58.6	35.8/35.2
PillarMultiView ECCV'20[35]	66.7‡	87.1/-	69.8/-	-	-	88.5/-	66.5/-	42.9/-
PVRCNN CVPR'20[30]	-	83.0/82.1	70.3/69.7	77.5/76.6	65.4/64.8	91.9/91.3	69.2/68.5	42.2/41.3
PVRCNN WOD'20[31]	300 ¶	-	77.5/76.9	-	68.7/68.2	-	-	-
RCD CORL'20 [3]	-	82.1/83.4	69.0/68.5	-	-	87.2/86.8	66.5/66.1	44.5/44.0
RSN CarS_1f (Ours)	-	86.7/86.0	70.5/70.0	77.5/76.8	63.0/62.6	90.8/90.4	67.8/67.3	45.4/44.9
RSN CarS_3f (Ours)	**15.5**	88.1/87.4	74.8/74.4	80.8/80.2	65.8/65.4	92.0/91.6	73.0/72.5	51.8/51.2
RSN CarL_1f (Ours)	-	88.5/87.9	75.1/74.6	81.2/80.6	66.0/65.5	91.8/91.4	73.5/73.1	53.1/52.5
RSN CarL_3f (Ours)	25.4	91.0/90.3	75.7/75.4	82.1/81.6	68.6/68.1	92.1/91.7	74.6/74.1	56.1/55.4
RSN CarXL_3f (Ours)	67.5	**91.3/90.8**	**78.4/78.1**	**82.6/82.2**	**69.5/69.1**	92.1/91.7	77.0/76.6	57.5/57.1

基于Range-view的感知方法 (RSN)

https://arxiv.org/abs/2106.13365

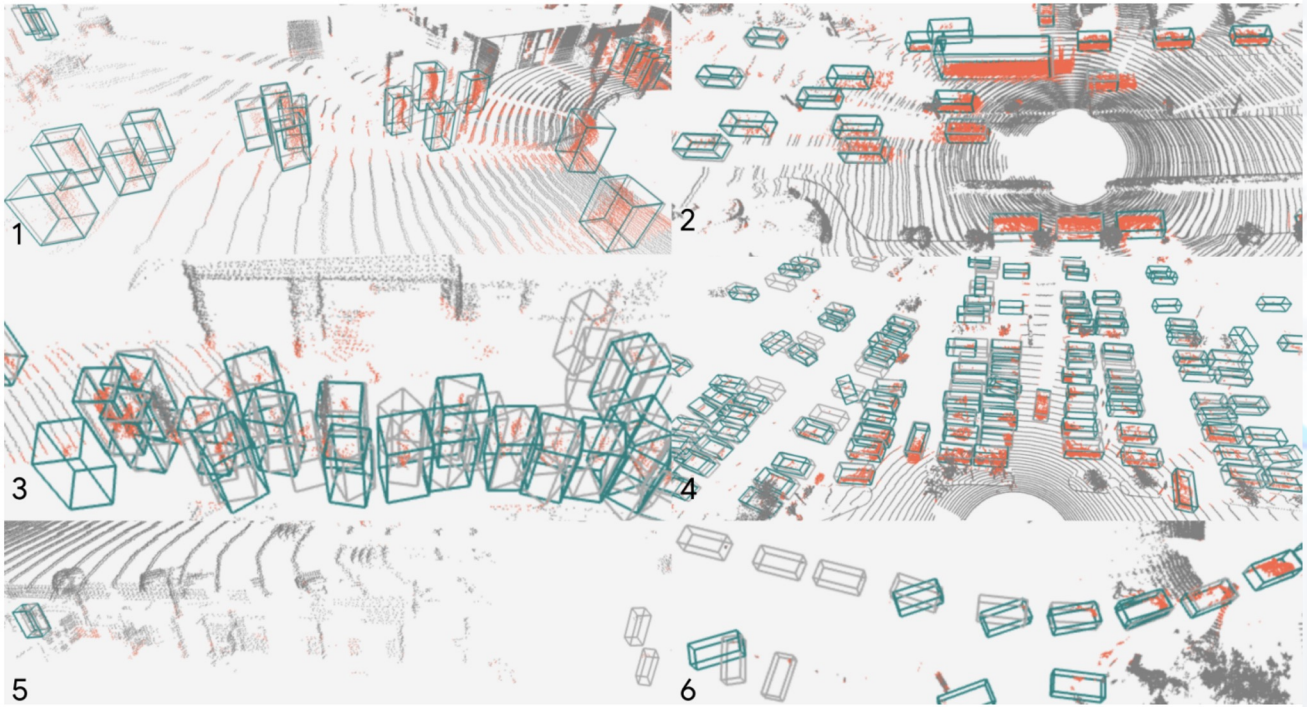
Performance comparisons on the Waymo Open Dataset validation set for **pedestrian detection**

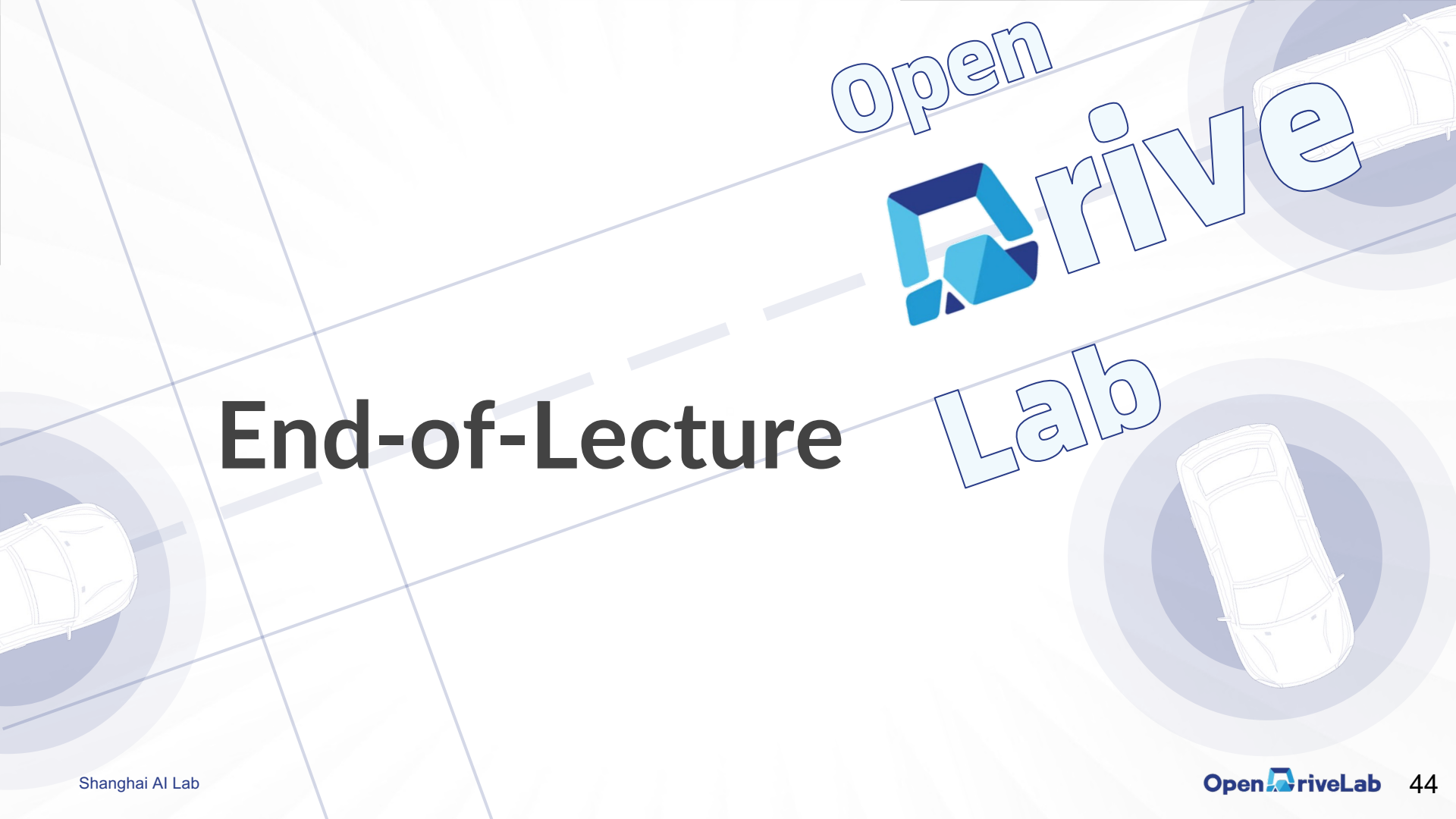
Method	Latency (ms)	AP/APH L1 BEV	AP/APH L1 3D	AP/APH L2 BEV	AP/APH L2 3D	AP/APH L1 3D by distance < 30m	30-50m	> 50m
LaserNet CVPR'19[21]*	64.3	70.0/-	63.4/-	-	-	73.5/-	61.6/-	42.7/-
P.Pillars CVPR'19[16]†	49.0	76.0/62.0	68.9/56.6	67.2/54.6	60.0/49.1	76.7/64.3	66.9/54.3	52.9/40.5
PillarMultiView ECCV'20[35]	66.7‡	78.5/-	72.5/-	-	-	79.3/—	72.1/—	56.8/—
PVRCNN WOD'20[31]	300 ¶	-	78.9/75.1	-	69.8/66.4	-	-	-
RSN PedS_1f (Ours)	-	80.7/74.9	74.8/69.6	71.2/65.9	65.4/60.7	81.4/77.4	72.8/66.8	59.0/50.6
RSN PedS_3f (Ours)	**14.4**	84.2/80.7	78.3/75.2	74.8/71.6	68.9/66.1	81.7/78.8	74.4/71.3	64.9/61.5
RSN PedL_1f (Ours)	-	83.4/77.6	77.8/72.7	73.9/68.6	68.3/63.7	83.9/79.7	74.1/68.2	62.1/54.1
RSN PedL_3f (Ours)	28.2	**85.0/81.4**	**79.4/76.2**	**75.5/72.2**	**69.9/67.0**	84.5/81.5	78.1/74.7	68.5/65.0

基于Range-view的感知方法 (RSN)

https://arxiv.org/abs/2106.13365

Example pedestrian and vehicle detection results on the Waymo Open Dataset validation set.





End-of-Lecture