3D Object Detection Using Temporal LiDAR Data

Scott McCrae and Avideh Zakhor

Video and Image Processing Lab

Department of Electrical Engineering and Computer Sciences

UC Berkeley

Berkeley
UNIVERSITY OF CALIFORNIA

Related Work

- Feature Extraction
  - PointNet[1], PointNet++[2]
- Voxelization
  - VoxelNet[3]
- Bird's Eye View
  - Fast and Furious[4], YOLO3D[5]
- Direct on point cloud
  - Frustum ConvNet[6]

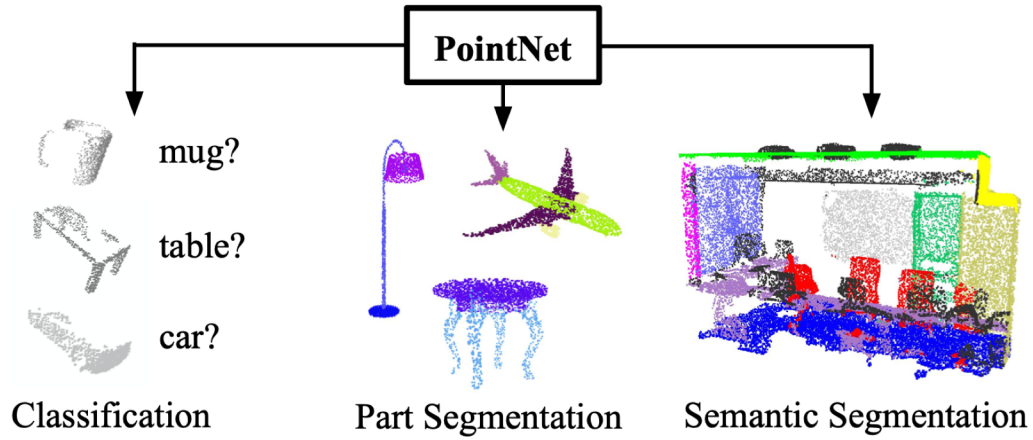
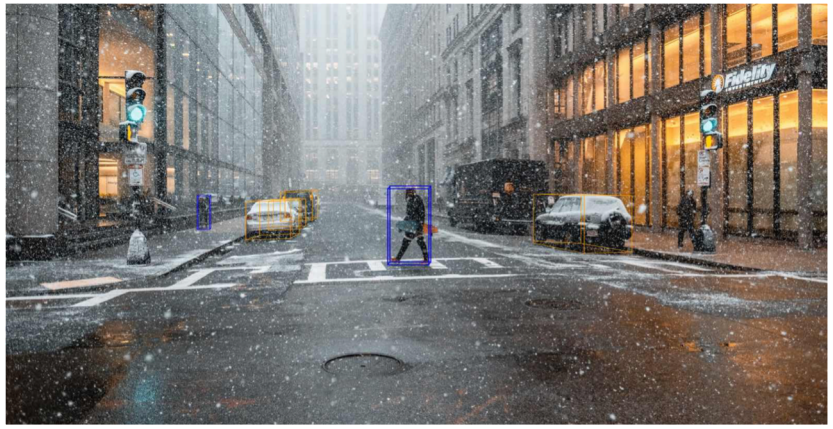


Figure from [3] demonstrates many applications of deep learning on point clouds

[1] Qi et al., "Pointnet: Deep learning on point sets for 3d classification and segmentation," *CVPR*, 2017.
[2] Qi et al., "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *NeurIPS*, 2017.
[3] Zhou et al., "Voxelnet: End-to-end learning for point cloud based 3d object detection," *arXiv preprint*, 2017.
[4] Luo et al., "Fast and furious: Real time end-to-end 3d detection, tracking and motion forecasting with a single convolutional net," *CVPR*, 2018.
[5] Ali et al., "Yolo3d: End-to-end real-time 3d oriented object bounding box detection from lidar point cloud," *arXiv preprint*, 2018.
[6] Wang et al., "Frustum convnet: Slid- ing frustums to aggregate local point-wise features for amodal 3d object detection," *arXiv preprint*, 2019.

Introduction

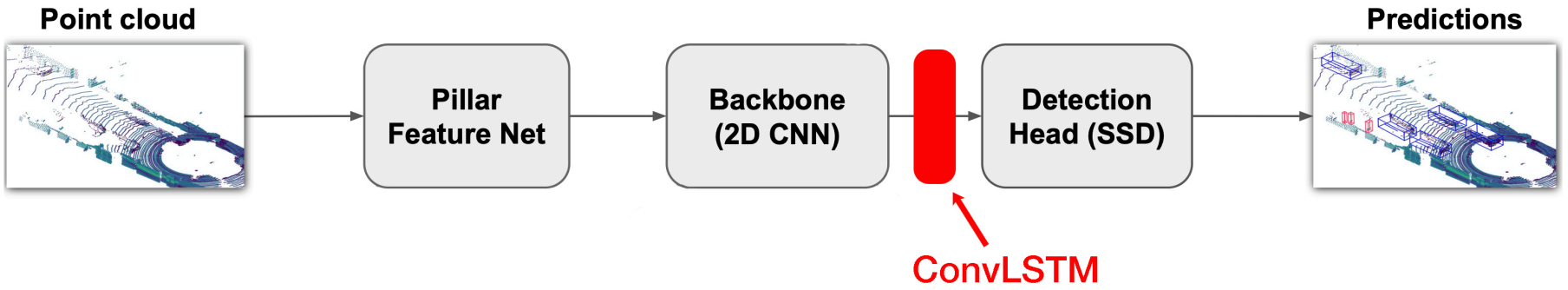
- 3D object detection is fundamental for autonomous driving
- New datasets (nuScenes, Waymo, Lyft Level5) from mid-2019 include sequential LiDAR data
- Investigate the efficacy of recurrent neural networks for processing sequential point cloud data



nuScenes data allows for testing recurrent architectures

Overview of Proposed Method

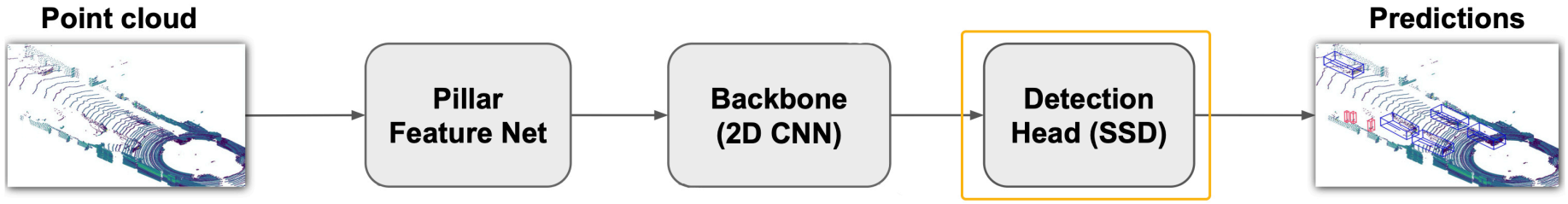
- Use PointPillars (PP)[7] as a benchmark non-recurrent object detection network
- Modify original architecture to use Convolutional Long Short-term Memory (ConvLSTM) recurrent structure ➔ call it PP-REC



[7] Lang et al., "PointPillars: Fast encoders for object detection from point clouds," *CVPR*, 2019.

PointPillar Architecture

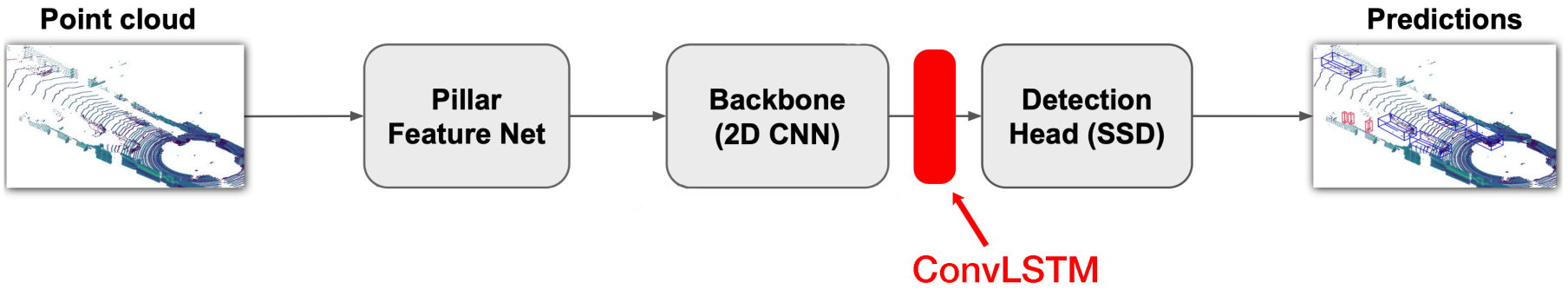
- PP[7] has three main stages:
  - Point cloud feature extraction, via PointNet[1] on pillar-shaped voxels
    - Maps point cloud into 2D pseudo-image
  - 2D convolutional neural network backbone
  - SSD detection head for producing 3D bounding boxes



[1] Qi et al., “Pointnet: Deep learning on point sets for 3d classification and segmentation,” *CVPR*, 2017.
[7] Lang et al., “PointPillars: Fast encoders for object detection from point clouds,” *CVPR*, 2019.

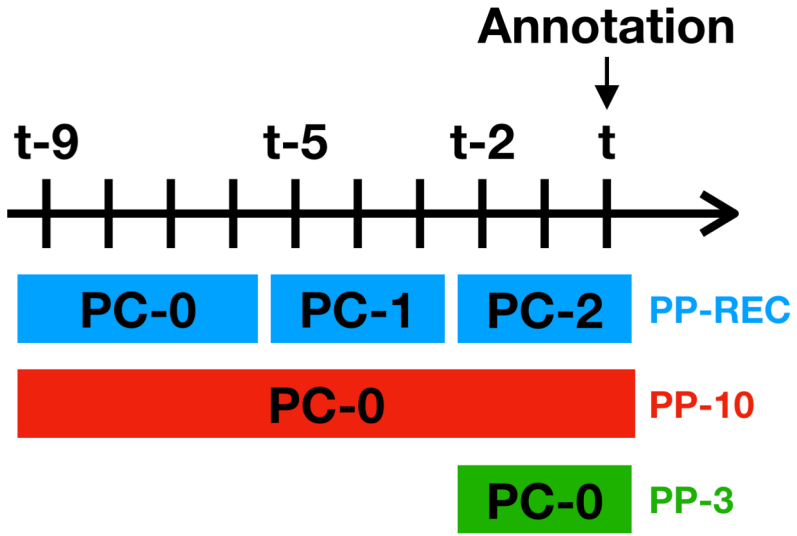
Proposed Architecture

- Added ConvLSTM layer, shown in red, takes input from the 2D CNN
- ConvLSTM output is directly used by the Detection Head
- Use recurrent structure to process sequences



Data Processing: Time Frame

- Original PP[7] models combine LiDAR data from multiple time steps into one Group of Frames (GoF)
  - e.g. 10 LiDAR frames combined into one GoF
- We create multiple smaller LiDAR GoF's for proposed recurrent model
  - e.g. 10 LiDAR frames split into three GoF's
  - Allows for build-up of recurrent memory
  - Less point cloud data per prediction

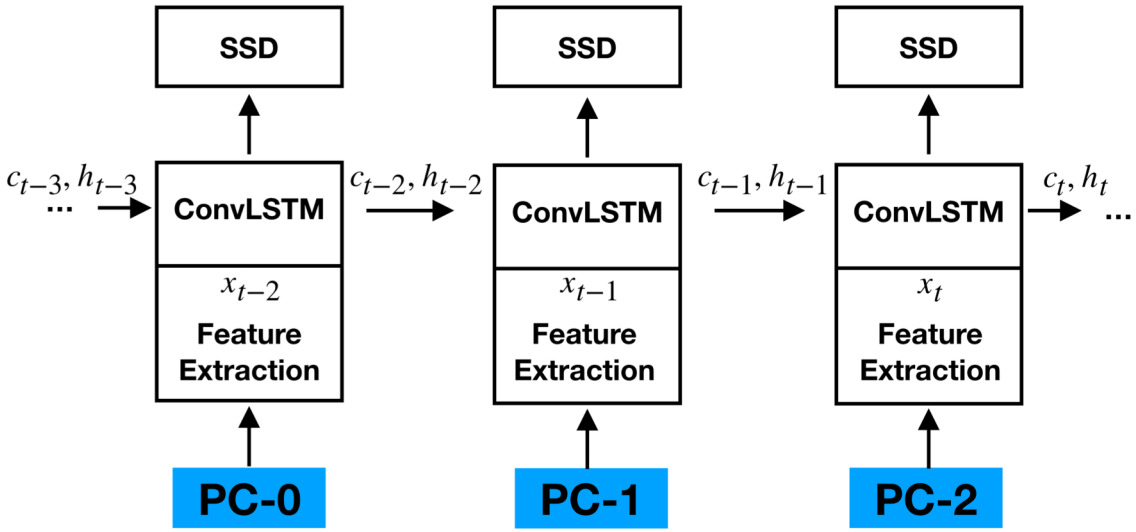


LiDAR data split into several Groups of Frames (PC-0, PC-1, PC-2), or a single Group of Frames (PC-0)

[7] Lang et al., "PointPillars: Fast encoders for object detection from point clouds," *CVPR*, 2019.

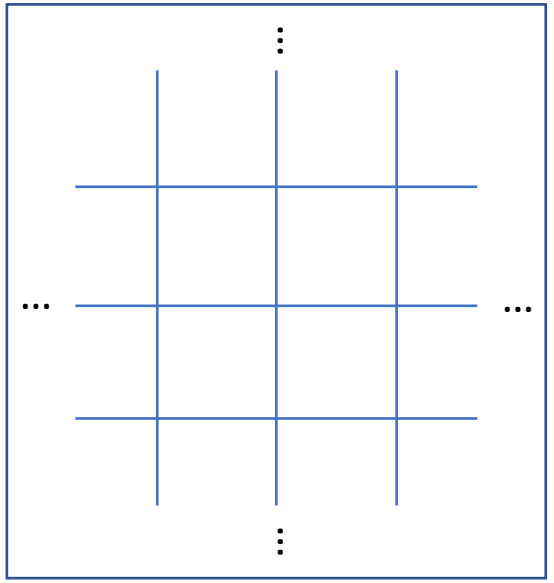
Data Processing: Recurrent Architecture

- ConvLSTM takes 2D featurized point cloud as input
- Hidden dimension is propagated through time, then used for detection
- ConvLSTM integrates well with CNN feature extraction



Experimental Setup: Voxelization

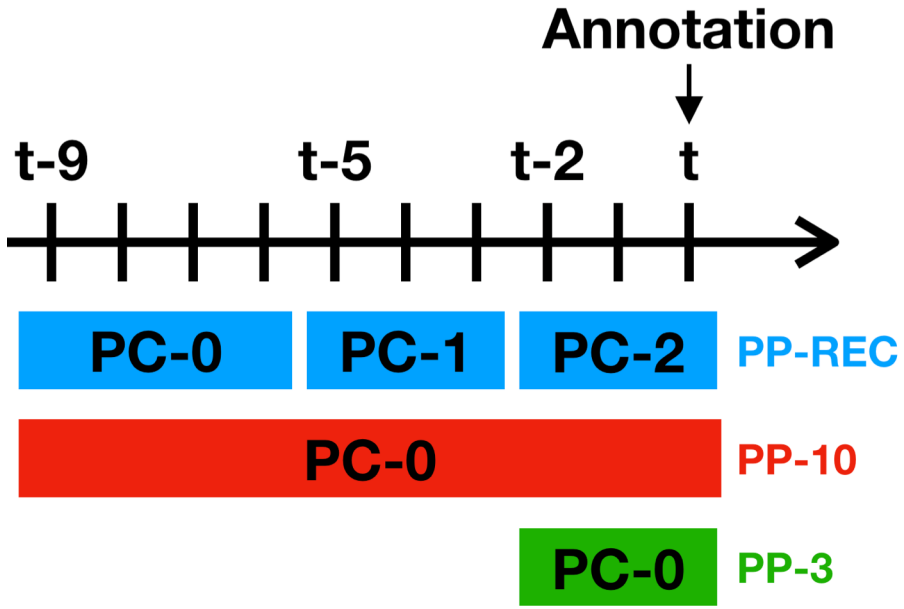
- Voxelization is controlled by varying the dimensions of pillars
- Train “coarse” and “fine” levels of voxelization
  - Coarse models use base dimension 0.3125m x 0.3125m
  - Fine models use base dimension 0.25m x 0.25m



Pillar bases are drawn on the ground plane

Experimental Setup: Time Frame

- Training data is annotated with bounding boxes at 2Hz, LiDAR recorded at 20Hz
- Models given variable number of LiDAR frames leading up to an annotation
- PP[7] is trained with up to 3, 10, and 20 frames before annotation
  - PP combines all frames into one GoF to produce detection results
- PP-REC (ours) is trained with up to 10 and 20 frames before annotation
  - Split into a sequence of GoF's with 3-4 frames each
  - Sequence is treated as a data stream
  - PP-REC run on data stream, detection results taken from most recent GoF
  - PP-REC-S uses GoF sequences incremented by one time step, rather than 3-4



[7] Lang et al., "PointPillars: Fast encoders for object detection from point clouds," *CVPR*, 2019.

Experimental Results (1)

Object Detection mAP (%)					
		Coarse		Fine	
	# of Frames	Car	Pedestrian	Car	Pedestrian
PP-20	20	74.81	52.32	**75.44**	59.44
PP-10	10	69.43	44.27	74.68	**60.26**
PP-3	3	65.79	30.51	71.64	50.82
PP-REC, 20 (Ours)	20	70.26	**53.01**	–	–
PP-REC, 10 (Ours)	10	67.04	52.46	67.97	56.87
PP-REC-S, 7 (Ours)	7	**75.79**	49.41	–	–

- PP-REC (ours) compares favorably to PP-3[7]
  - Our model has higher car and pedestrian mAP
  - Both use 3 LiDAR frames for detection; ours also has recurrent memory
  - PP-REC between PP-3 and PP-10 for pedestrian detection in fine case
- PP-10[7] and PP-20[7] maintain an advantage in fine case

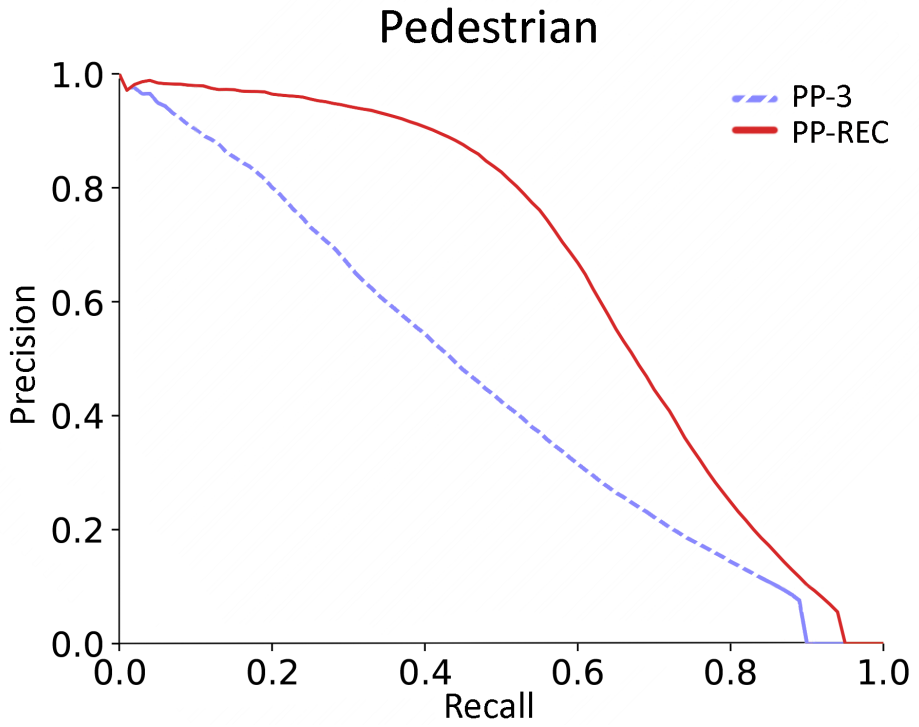
[7] Lang et al., "PointPillars: Fast encoders for object detection from point clouds," *CVPR*, 2019.

Experimental Results (2)

Object Detection mAP (%)					
		Coarse		Fine	
	# of Frames	Car	Pedestrian	Car	Pedestrian
PP-20	20	74.81	52.32	**75.44**	59.44
PP-10	10	69.43	44.27	74.68	**60.26**
PP-3	3	65.79	30.51	71.64	50.82
PP-REC, 20 (Ours)	20	70.26	**53.01**	–	–
PP-REC, 10 (Ours)	10	67.04	52.46	67.97	56.87
PP-REC-S, 7 (Ours)	7	**75.79**	49.41	–	–

- PP-REC-S 7 outperforms PP-10 in coarse case
  - Outperforms PP-20 in coarse car detection, with many fewer LiDAR frames for detection
- Accuracy increases with more data
  - More data alleviates issues from reduced resolution
  - Our model can build long-term memory to use at runtime without increased overhead

Experimental Results (3)



- PP-REC (ours) maintains higher positive predictive value as the recall rate increases compared to PP-3[7]
  - This explains the reason PP-REC outperforms PP-3 in pedestrian detection mAP

[7] Lang et al., "PointPillars: Fast encoders for object detection from point clouds," *CVPR*, 2019.

Conclusion and Future Work

- Recurrent models are a promising avenue for processing newly available sequential LiDAR data
- Leverage multi-sensor fusion in a recurrent fashion
- RGB-based recurrent networks
  - 2D video object detection, monocular depth estimation

Thank you!

Berkeley
UNIVERSITY OF CALIFORNIA