PERCEPTION AND PREDICTION IN MULTI-AGENT URBAN TRAFFIC SCENARIOS FOR AUTONOMOUS DRIVING

Prarthana Bhattacharyya

PhD Thesis Seminar

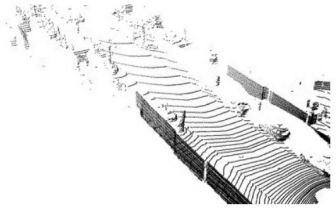
September 5, 2023

Committee Members:

Krzysztof Czarnecki, Zhou Wang, Mark Crowley

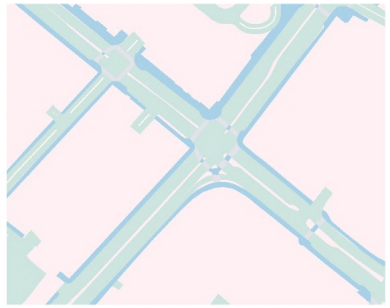
INTRODUCTION

TRADITIONAL AUTONOMY STACK



LiDAR

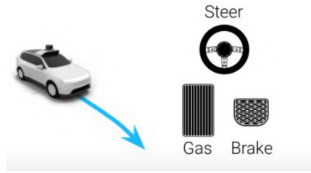
RGB



Map

Inputs

3D Object Detection → 3D Multi-Object Tracking → Multi-Agent Forecasting



Outputs

UNIVERSITY OF WATERLOO

CONTRIBUTIONS

Part II: SA-Det3D
Bhattacharyya*, Huang, Czarnecki, ECCVW'20, ICCVW'21

Part I: FANTrack
Baser, Balasubramanian*, **Bhattacharyya***, Czarnecki, IV'19

Part III(a): SSL-Lanes
Bhattacharyya*, Huang, Czarnecki, CoRL'22

Part III(b): SSL-Interactions
Bhattacharyya*, Huang, Czarnecki, Preprint'23

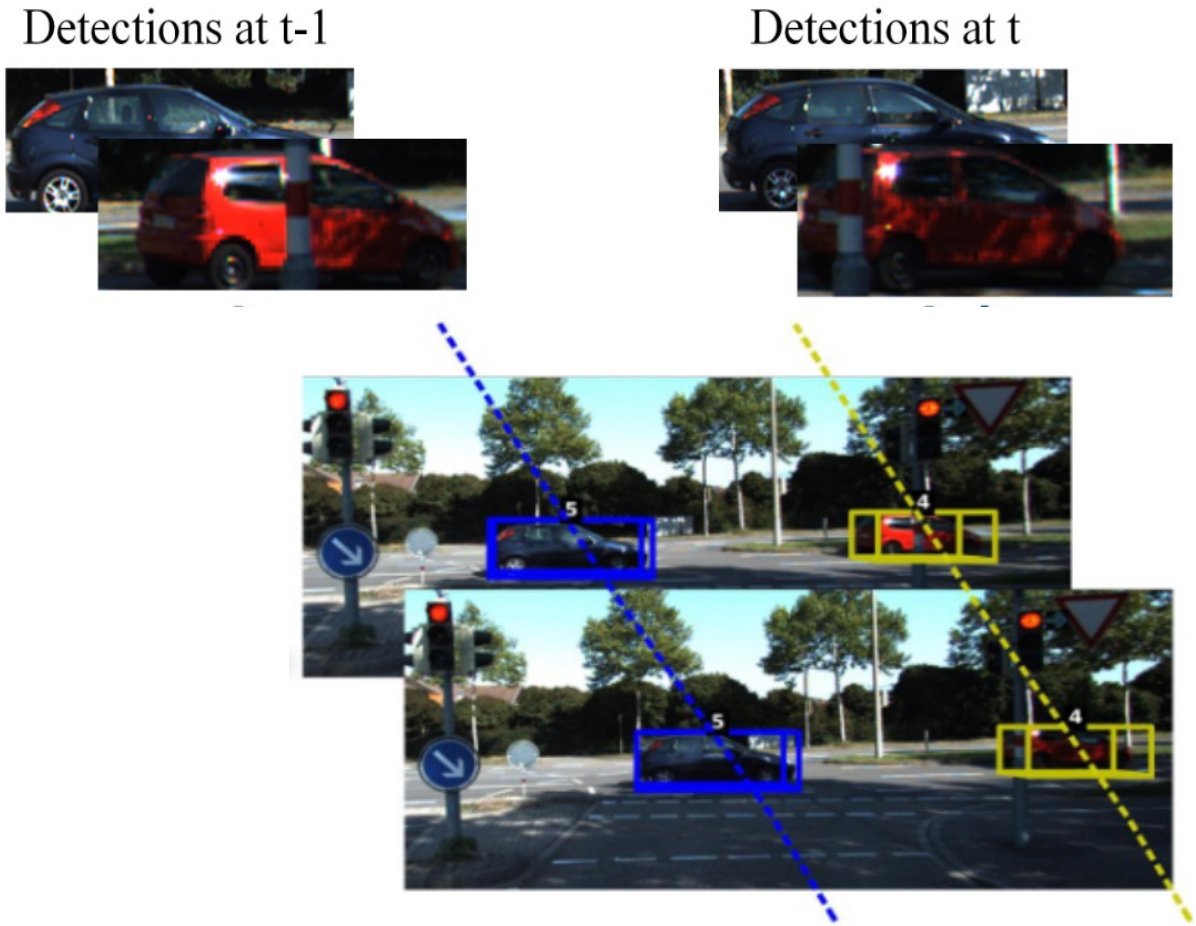
UNIVERSITY OF WATERLOO

3D MULTI-OBJECT TRACKING

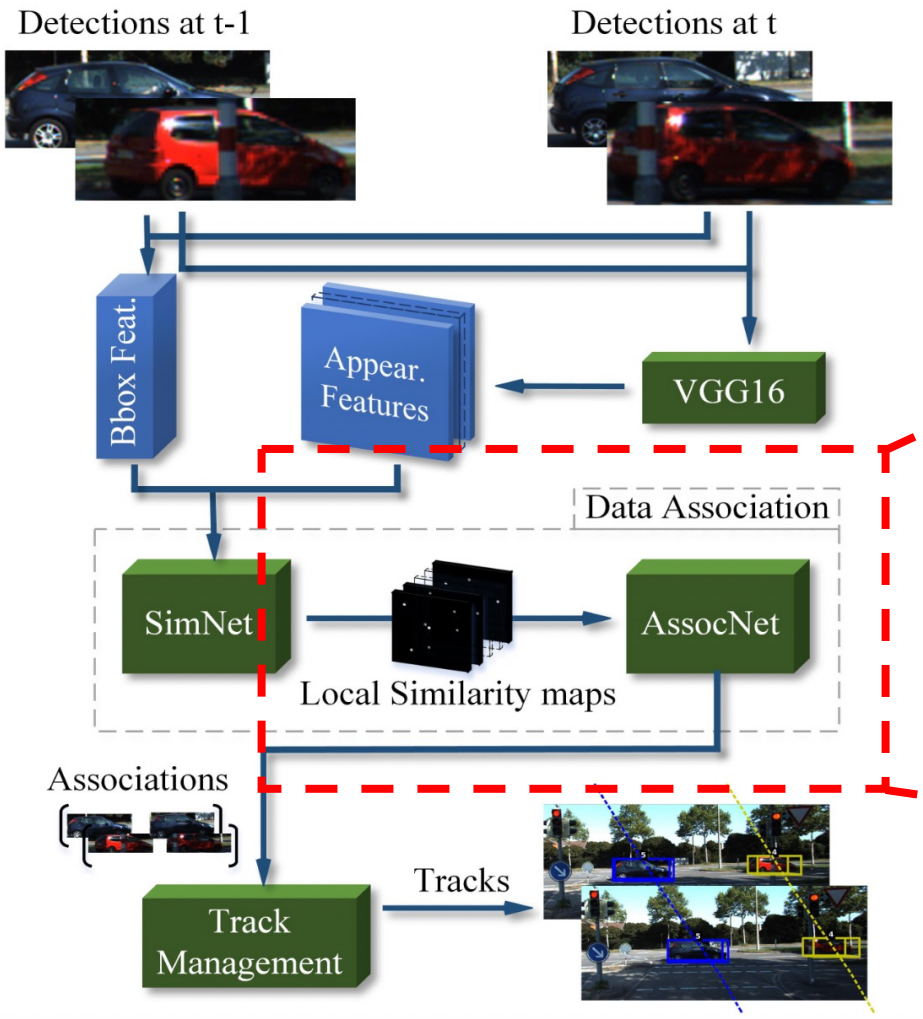
Erkan Baser, Venkatesh Balasubramanian*, **Prarthana Bhattacharyya***, Krzysztof Czarnecki
"FANTrack: 3d multi-object tracking with feature association network." IEEE Intelligent Vehicles Symposium (IV), 2019.

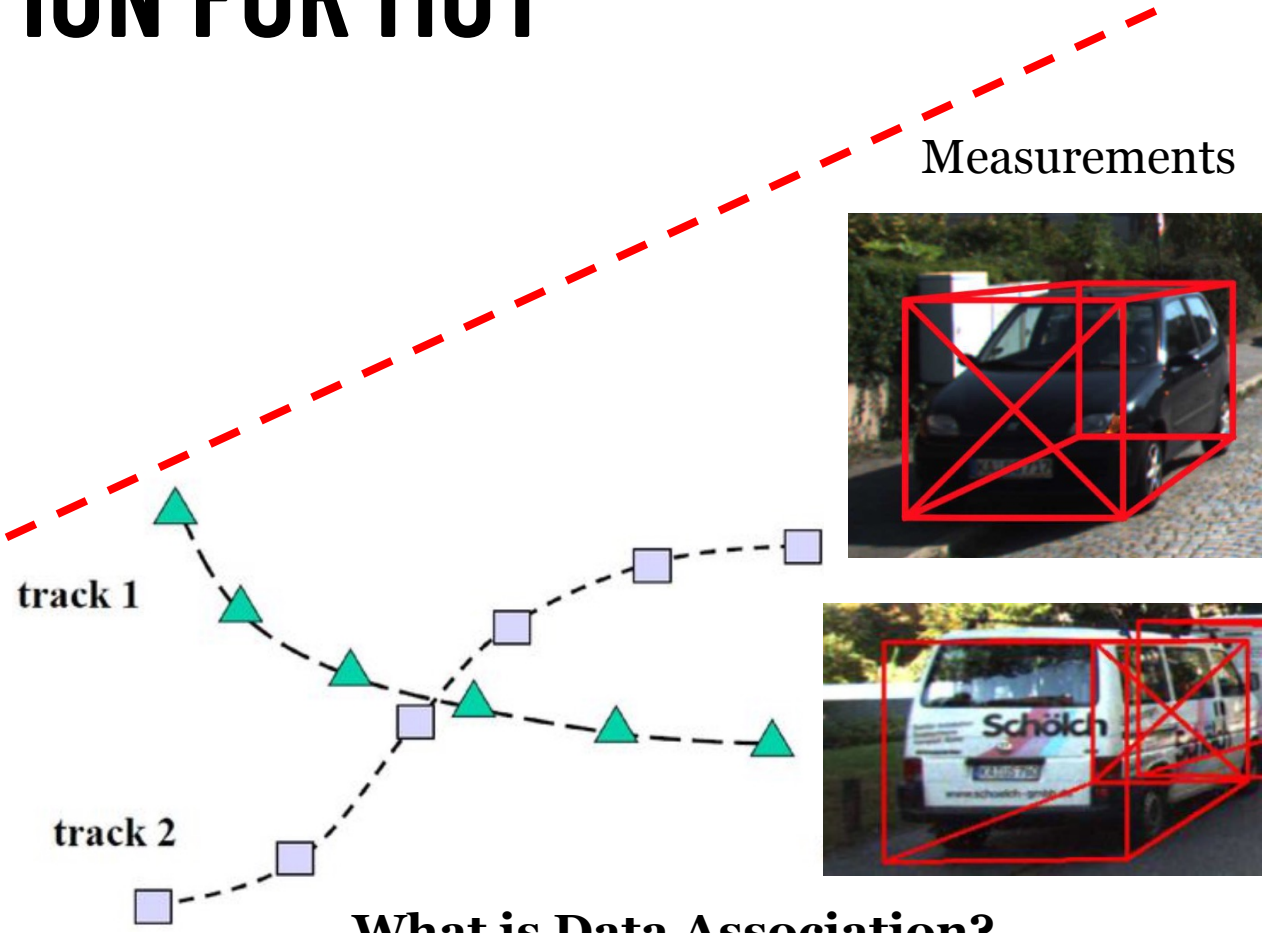
GOAL: MULTI-OBJECT TRACKING (MOT)

- 3D multi-object tracking: determine location and orientation of multiple objects in a three-dimensional environment over time.
- 'Tracking-by-detection': detecting objects in each frame of a sequence and then associating these detections across frames.



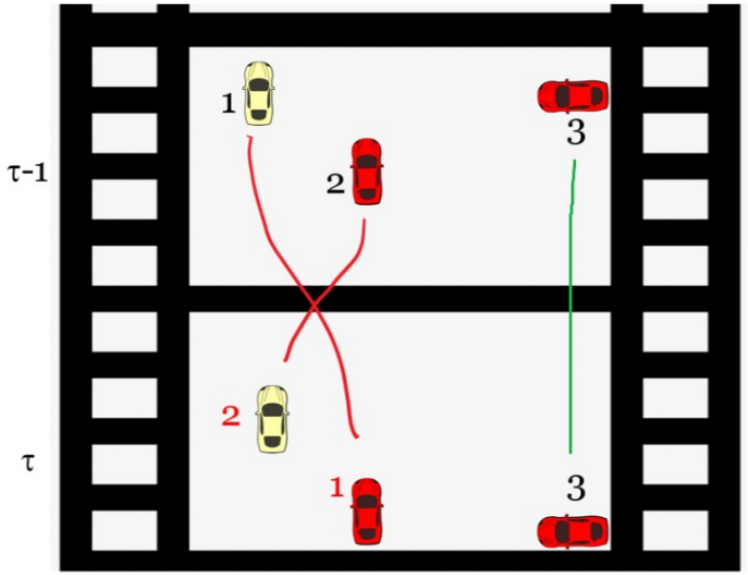
PROBLEM: DATA ASSOCIATION FOR MOT



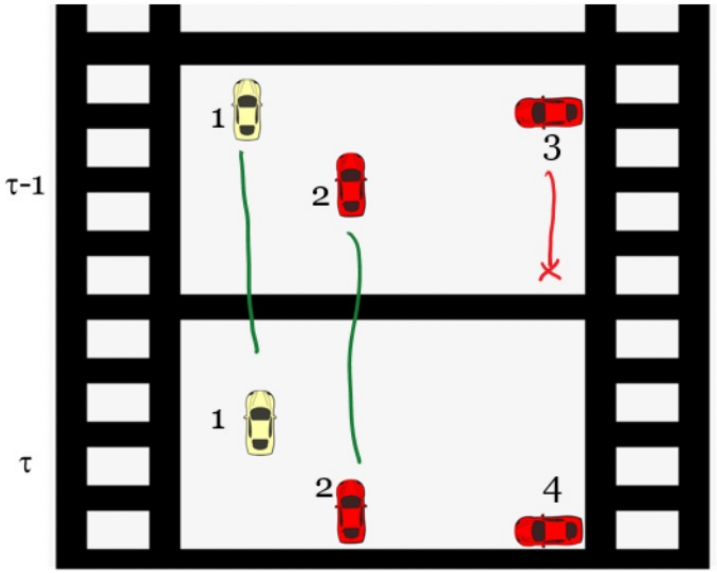


CHALLENGES: DATA ASSOCIATION FOR MOT

- Clutter
- Occlusion
- Missed detections
- False Detections



(a) ID Switching scenario



(b) Fragmentation scenario

LITERATURE: LINEAR ASSIGNMENT FOR DATA ASSOCIATION

We have *N* tracks in previous frame and *M* objects in current frame, a table of match scores c(i, j) for i=1…N and j=1…M.

Measurements (M)

Tracks (N)	1	2	3	4	5
1	0.95	0.76	0.62	0.41	0.06
2	0.23	0.46	0.79	0.94	0.35
3	0.61	0.02	0.92	0.92	0.81
4	0.49	0.82	0.74	0.41	0.01
5	0.89	0.44	0.18	0.89	0.14

Cost Matrix

Problem: choose a 1-1 correspondence that maximizes sum of match scores.

$$\min_{x_{i,j}} \sum c_{i,j} x_{i,j}$$

$$s.t. \sum_{i:i>0} x_{i,j} = 1$$

$$\sum_{j:j>0} x_{i,j} = 1$$

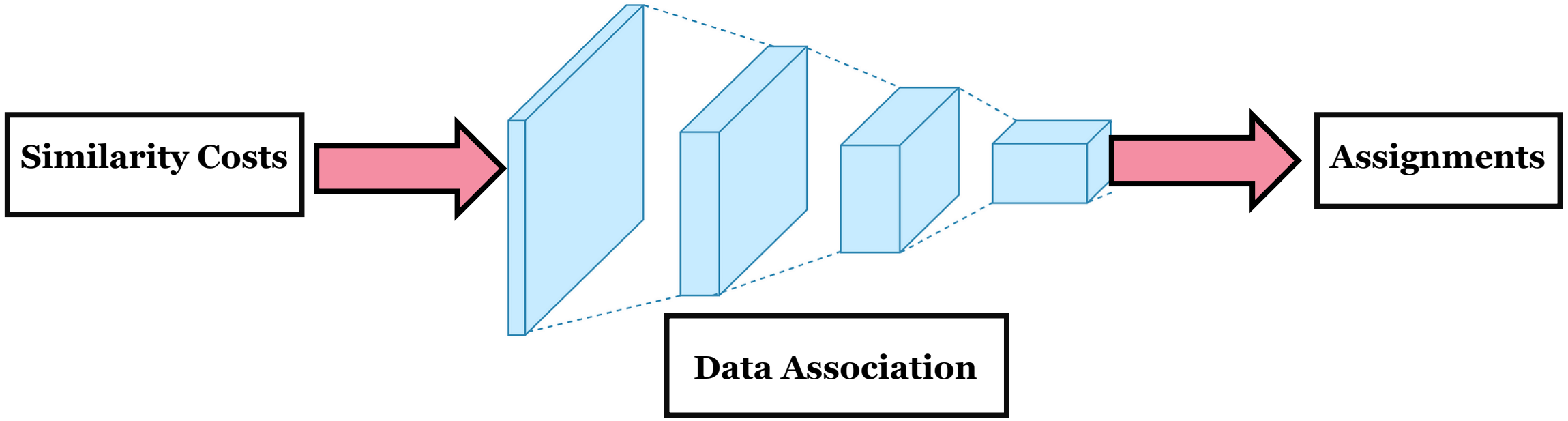
$$x_{i,j} \in \{0, 1\}$$

Hungarian Algorithm

$$\begin{bmatrix} \underline{\mathbf{0.9}} & \begin{bmatrix} \underline{\mathbf{1}} & 0 & 0 & 0 & 0 \\ 0 & 0 & \underline{\mathbf{1}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \underline{\mathbf{1}} \\ 0 & \underline{\mathbf{1}} & 0 & 0 & 0 \\ 0 & 0 & 0 & \underline{\mathbf{1}} & 0 \end{bmatrix} & .41 & 0.06 \\ 0.2 & & .94 & 0.35 \\ 0.6 & & .92 & \underline{\mathbf{0.81}} \\ 0.4 & & .41 & 0.01 \\ 0.8 & & \underline{\mathbf{.89}} & 0.14 \end{bmatrix}$$

Final Assignment

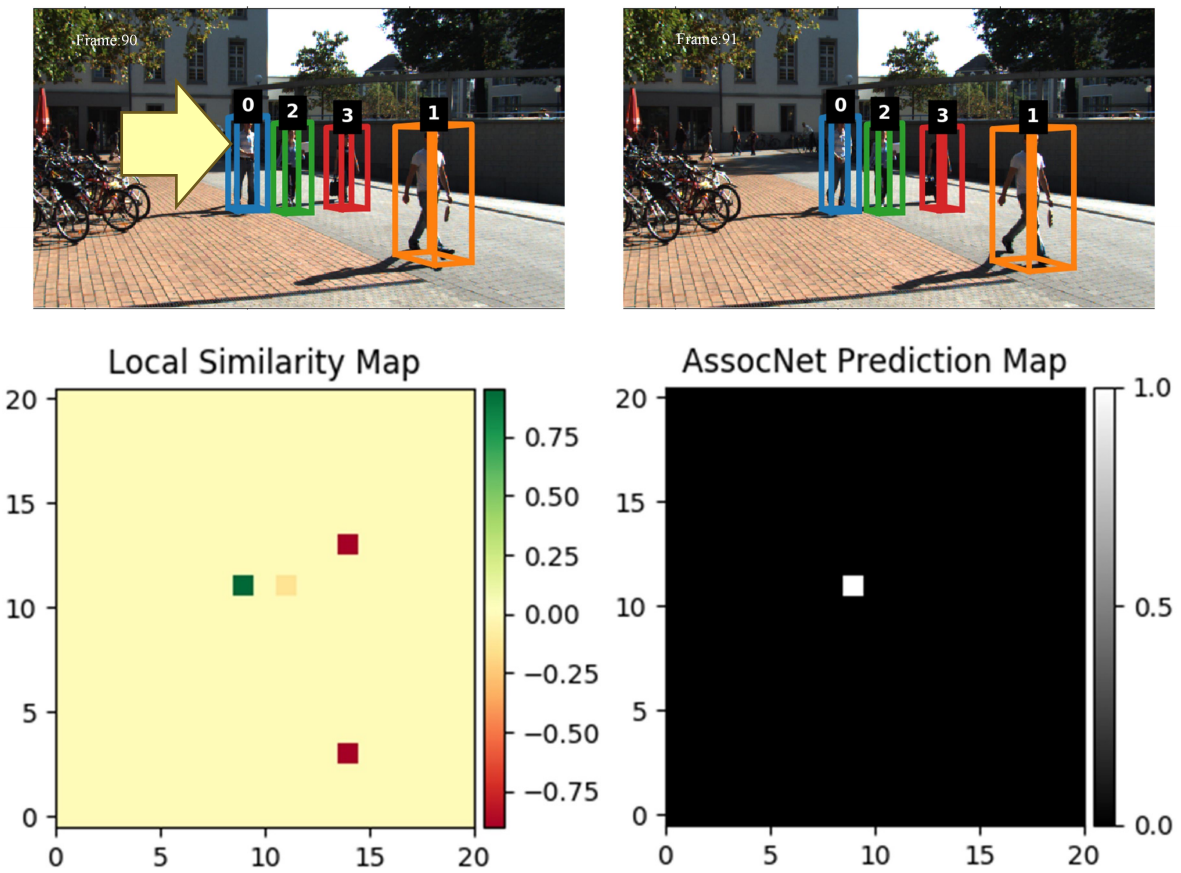
FANTRACK: DATA-DRIVEN ASSOCIATION



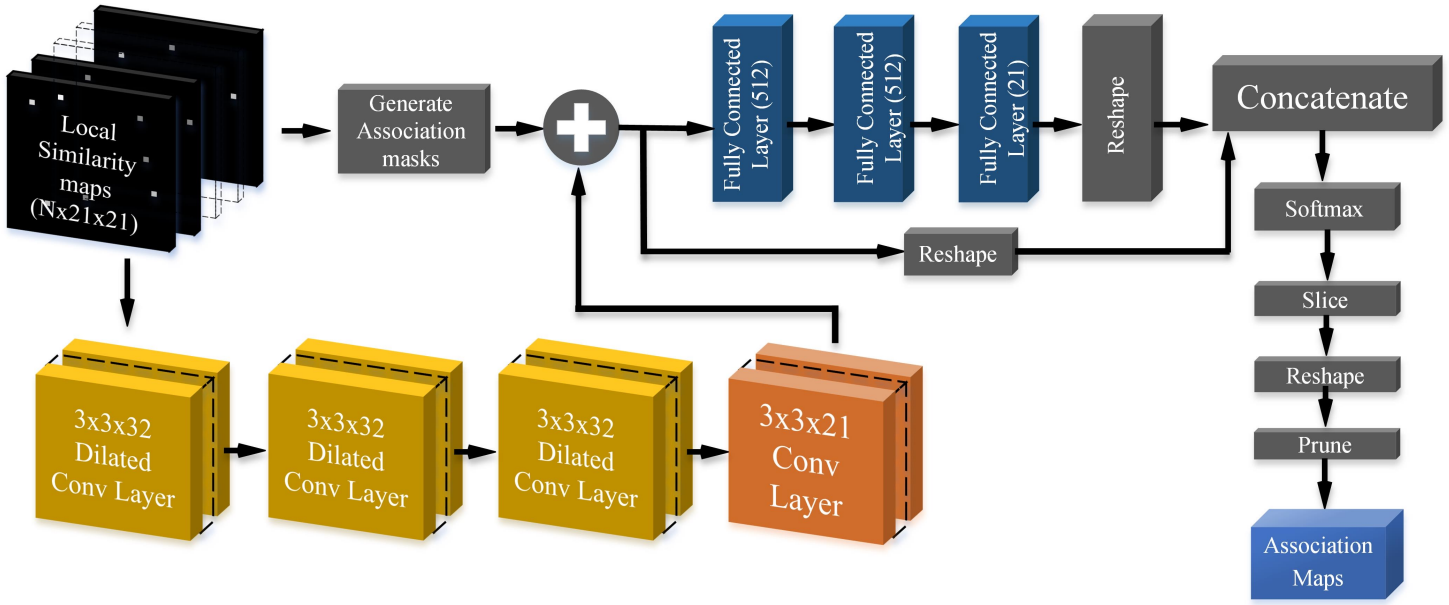
- *Core idea: Learn the matching step using a CNN*
- *Advantages: Learn matchable features for data association end-to-end*

GENERAL IDEA FOR DATA-DRIVEN ASSOCIATION (ASSOCNET)

- We have four tracks in frame *t-1* and four measurements in frame *t*.
- For simplicity, we only consider the target belonging to track 0.
- The local similarity map is constructed, and we see that the most probable measurement that is like the target is given a higher similarity score.
- Output prediction by AssocNet corresponds to correct assignment for track 0.



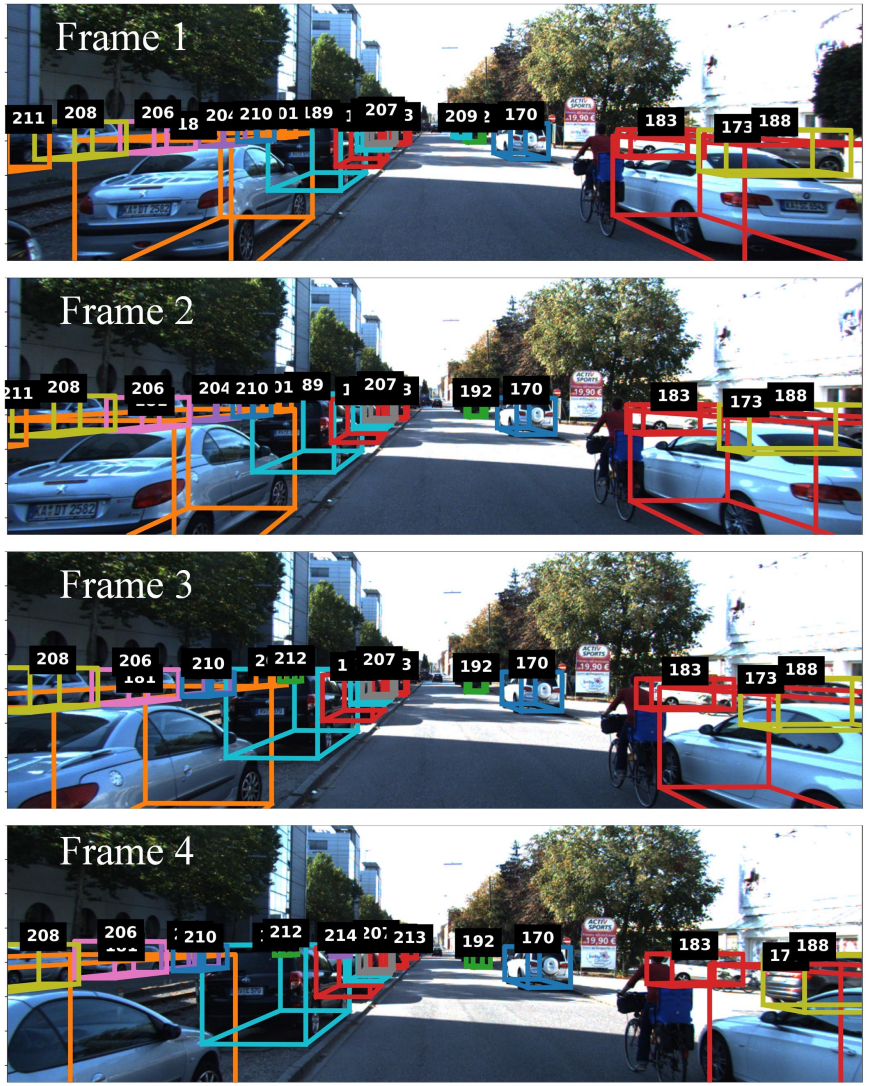
PROPOSED: ASSOCNET



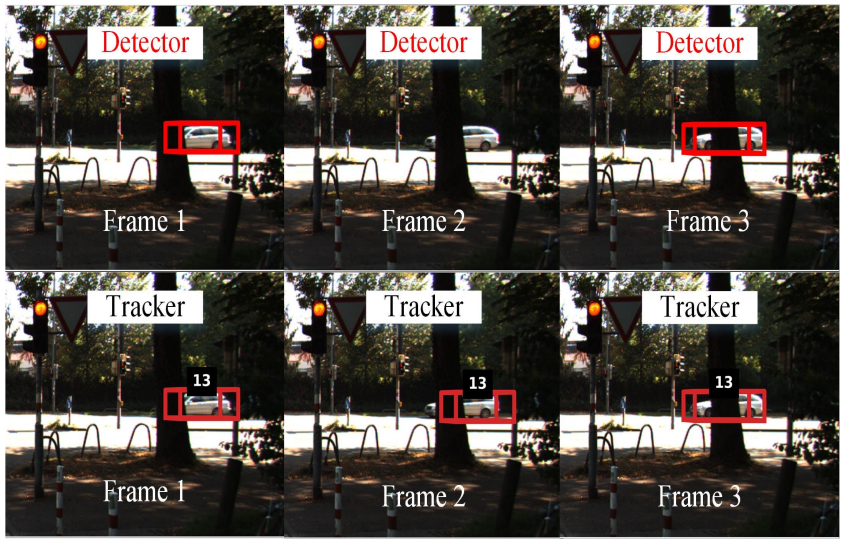
- AssocNet performs the actual data association between the targets (tracks) and measurements using the similarity scores.
- The outputs from the network are the *probability distributions for each target (track) of being associated to any measurement or to none*.

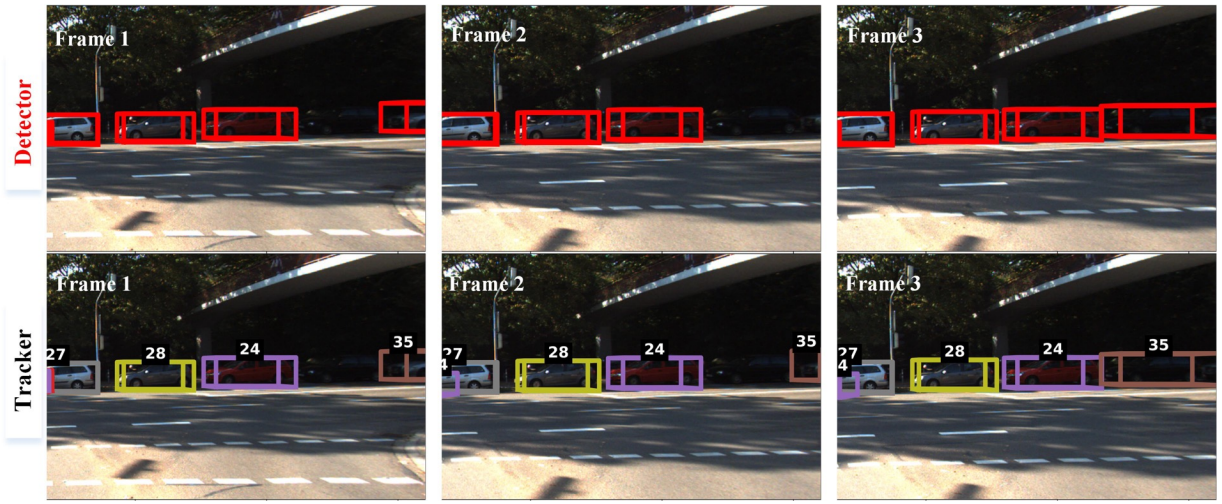
FANTRACK PERFORMANCE

- Scenario: Crowded with parked cars
- The tracker can *perform well despite the clutter* from closely parked cars



ROBUSTNESS OF FAN-TRACK





- Proposed methods can track through missed detections (*left*) and low-lit conditions (*right*)

HIGHER PERFORMANCE COMPARED TO HUNGARIAN ALGORITHM

Method	MOTA ↑	MOTP ↑	MT ↑	PT ↑	ML ↓	IDS ↓	FRAG ↓
Euclidean+*AssocNet*	56.16 %	84.84 %	72.22 %	18.51 %	9.25 %	269	320
Manhattan+*AssocNet*	56.75 %	84.83 %	**73.14 %**	17.59 %	9.25 %	265	319
Bhattacharyya+*AssocNet*	56.69 %	84.81 %	72.22 %	18.51 %	9.25 %	256	307
ChiSquare+*AssocNet*	57.17 %	84.81 %	**73.14 %**	18.51 %	**8.33 %**	262	311
SimNet+Hungarian	74.59 %	**84.92 %**	65.74 %	**23.14 %**	11.11 %	26	93
SimNet ImgOnly+*AssocNet*	74.30 %	84.75 %	73.14 %	17.59 %	9.25 %	29	82
SimNet BboxOnly+*AssocNet*	75.51 %	84.74 %	72.22 %	18.51 %	9.25 %	15	70
SimNet+*AssocNet*	**76.52 %**	84.81 %	**73.14 %**	17.59 %	9.25 %	**1**	**54**

(↑ denotes higher values are better. ↓ denotes lower values are better)

- AssocNet has *less fragmentation, fewer ID switches* and *higher tracking accuracy* as compared to Hungarian based data association on KITTI 'Car' Val Tracking Split

FANTRACK COMPARED TO STATE-OF-THE-ART

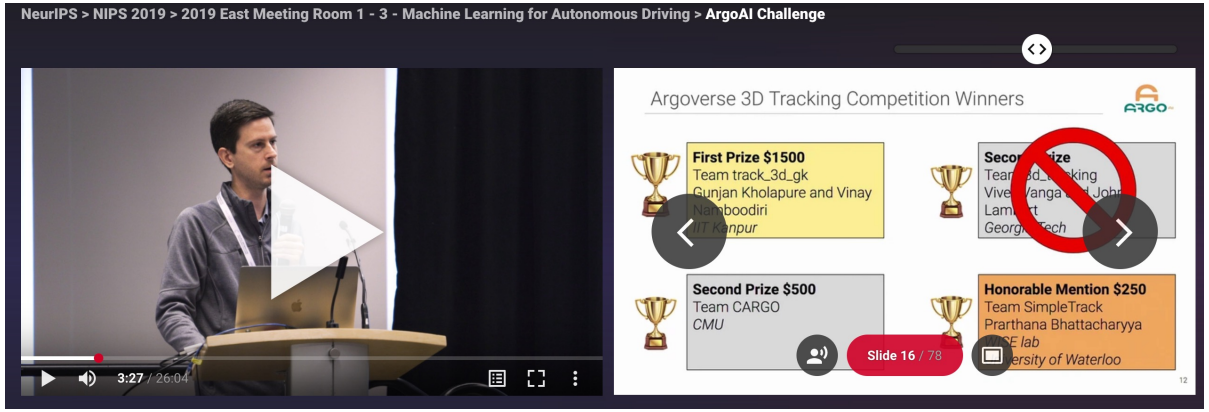
Method	MOTA ↑	MOTP ↑	MT ↑	ML ↓	IDS ↓	FRAG ↓
MOTBeyondPixels [73]	**84.24 %**	**85.73 %**	**73.23 %**	**2.77 %**	468	944
JCSTD [87]	80.57 %	81.81 %	56.77 %	7.38 %	**61**	643
3D-CNN/PMBM [72]	80.39 %	81.26 %	62.77 %	6.15 %	121	613
extraCK [29]	79.99 %	82.46 %	62.15 %	5.54 %	343	938
MDP [86]	76.59 %	82.10 %	52.15 %	13.38 %	130	**387**
FANTrack (Ours)	77.72 %	82.32 %	62.61 %	8.76 %	150	812

(↑ denotes higher values are better. ↓ denotes lower values are better)

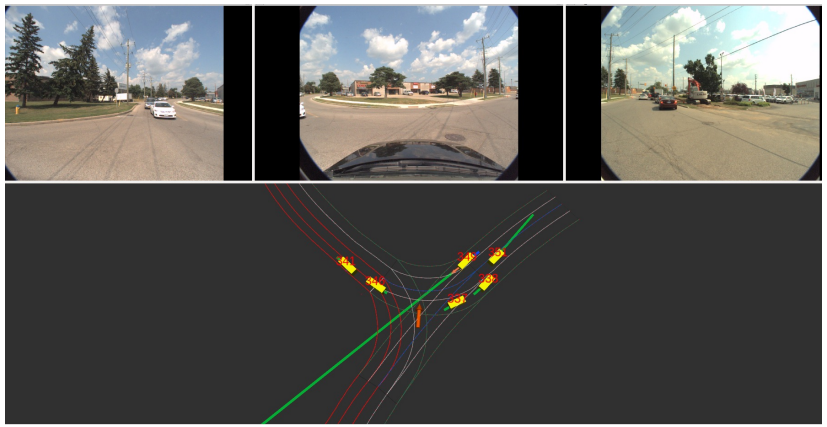
- MOTBeyondPixels uses Hungarian algorithm for data association
- Our proposed method has 3x fewer ID switches on KITTI 'Car' Test Tracking Split
- ID switches can be extremely costly to an ADS - interrupted or fragmented trajectories that weaken prediction and planning capabilities

EXTENDING FANTRACK

Fantrack: 3d multi-object tracking with feature association network 104 2019
E Baser, V Balasubramanian*, P Bhattacharyya*, K Czarnecki
2019 IEEE Intelligent Vehicles Symposium (IV), 1426-1433



https://slideslive.com/38923162/argoai-challenge



Balasubramanian et al., UW Space 2019

PERFORMANCE OF CAR ON THE KITTI VAL SET USING THE PROPOSED **3D** MOT EVALUATION TOOL WITH NEW METRICS.

Method	Input Data	Matching criteria	**sAMOTA**↑	AMOTA↑	AMOTP↑	MOTA↑	MOTP↑	IDS↓	FRAG↓	FPS↑
mmMOT [30] (ICCV'19)	2D + 3D	IoU_{thres} = 0.25	70.61	33.08	72.45	74.07	78.16	10	55	4.8 (GPU)
		IoU_{thres} = 0.5	69.14	32.81	72.22	73.53	78.51	10	64	
		IoU_{thres} = 0.7	63.91	24.91	67.32	51.91	80.71	24	141	
FANTrack [15] (IV'20)	2D + 3D	IoU_{thres} = 0.25	82.97	40.03	75.01	74.30	75.24	35	202	25.0 (GPU)
		IoU_{thres} = 0.5	80.14	38.16	73.62	72.71	74.91	36	211	
		IoU_{thres} = 0.7	62.72	24.71	66.06	49.19	79.01	38	406	

Weng et al., IROS 2020

TAKEAWAYS

- Current 3D object tracking methods do not consider data-driven methods for data-association, formulating the task as a complex optimization problem that can be solved effectively.

- We present FANTrack and exploit the power of deep learning to formulate the data association problem as inference in a CNN.

- FANTrack improves 3D tracking accuracy and enables competitive tracking across challenging scenes.

3D MULTI-OBJECT DETECTION

1. **Prarthana Bhattacharyya**, Krzysztof Czarnecki. “Deformable PV-RCNN: Improving 3D Object Detection with Learned Deformations”. ECCVW, 2020.

2. **Prarthana Bhattacharyya**, Chengjie Huang, Krzysztof Czarnecki. “SA-Det3D: Self-Attention based Context-Aware 3D Object Detection”. ICCVW, 2021.

UNIVERSITY OF WATERLOO

GOAL: TO IMPROVE TRACKING, IMPROVE DETECTIONS

$$\text{MOTA} = 1 - \frac{\text{FP} + \text{FN} + \text{IDS}}{\text{num}_{\text{gt}}}, \quad (6)$$

where num_{gt} is the number of ground truth objects in all frames. The AMOTA is then defined as follows:

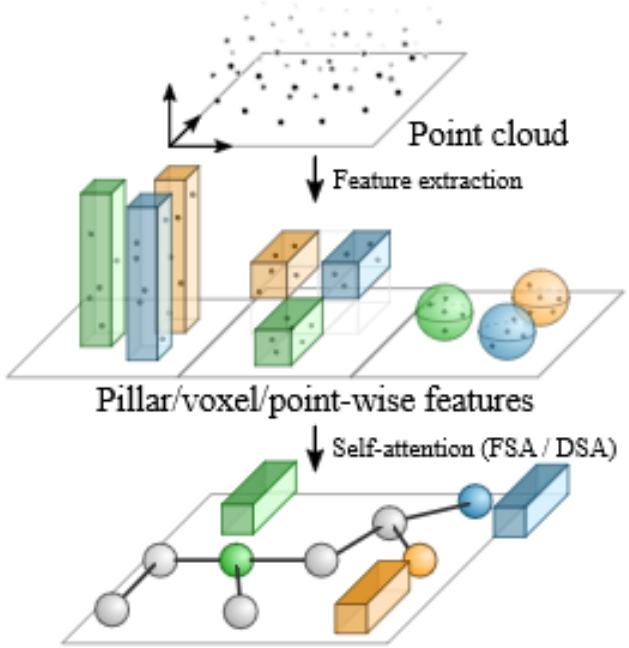
$$\text{AMOTA} = \frac{1}{L} \sum_{r \in \{\frac{1}{L}, \frac{2}{L}, \cdots, 1\}} (1 - \frac{\text{FP}_r + \text{FN}_r + \text{IDS}_r}{\text{num}_{\text{gt}}}), \quad (7)$$

where FP_r, FN_r and IDS_r are the number of false positives, false negatives and identity switches computed at a specific recall value r. Also, L is the number of recall values (number of confidence thresholds for integration). The higher L is,

- One of the key tracking metrics: AMOTA
- We have improved ID-switches in the previous section
- *In this part, we propose to improve detector performance*

SA-DET3D

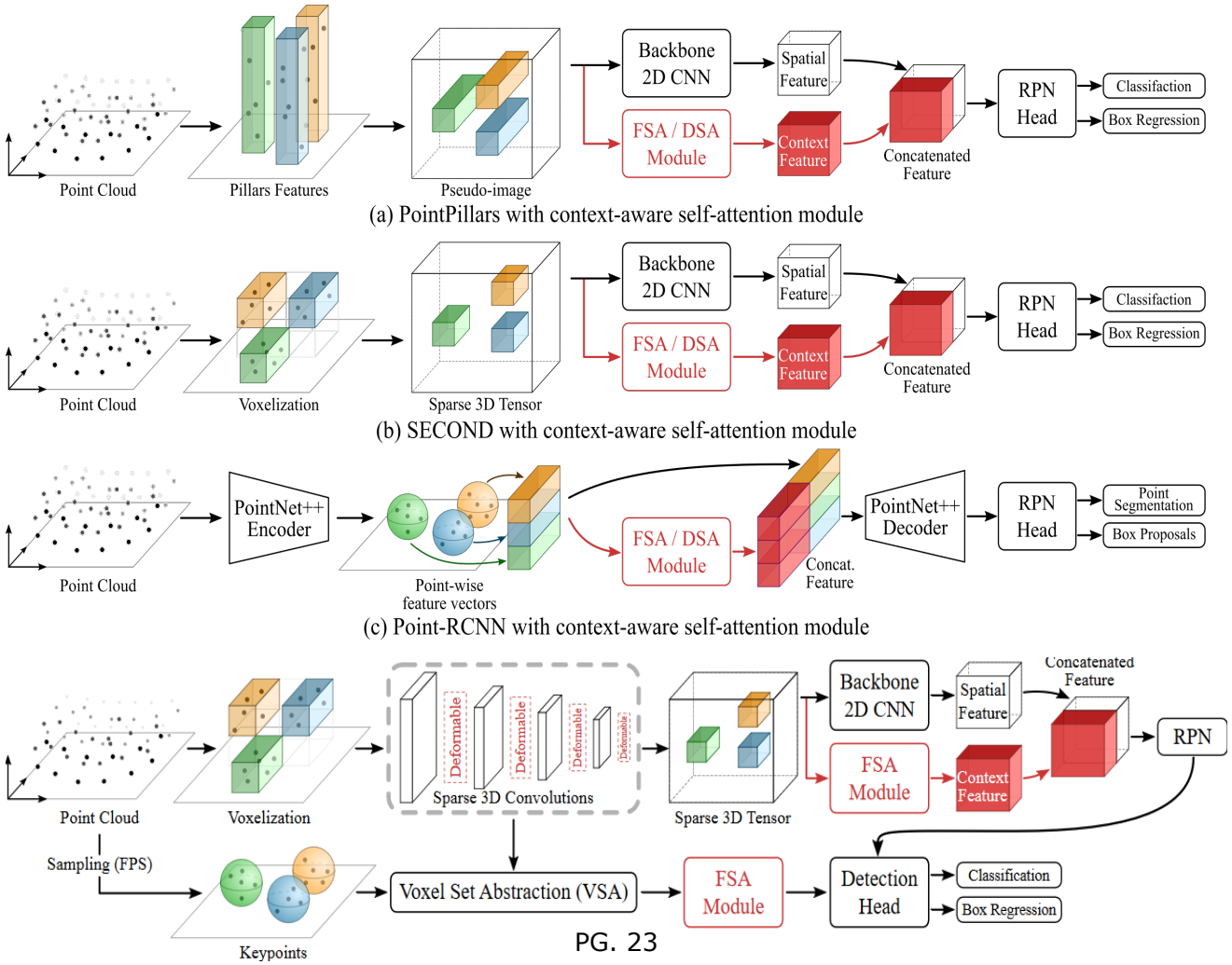
- CNN-like feature extractors for current 3D detectors have several limitations
  - Number of parameters scale poorly with increase in receptive field
  - Learned filters stationary across all locations
- Global correlation awareness can produce more powerful features
  - Missing/noisy point-cloud data
  - Large imbalance between points in nearby and far objects
  - Strong correlation between orientation of cars in same lane
  - High-confidence false positives can be eliminated by acquiring context in larger resolutions
- Non-local neural networks and self-attention for 2D-vision have been quite effective



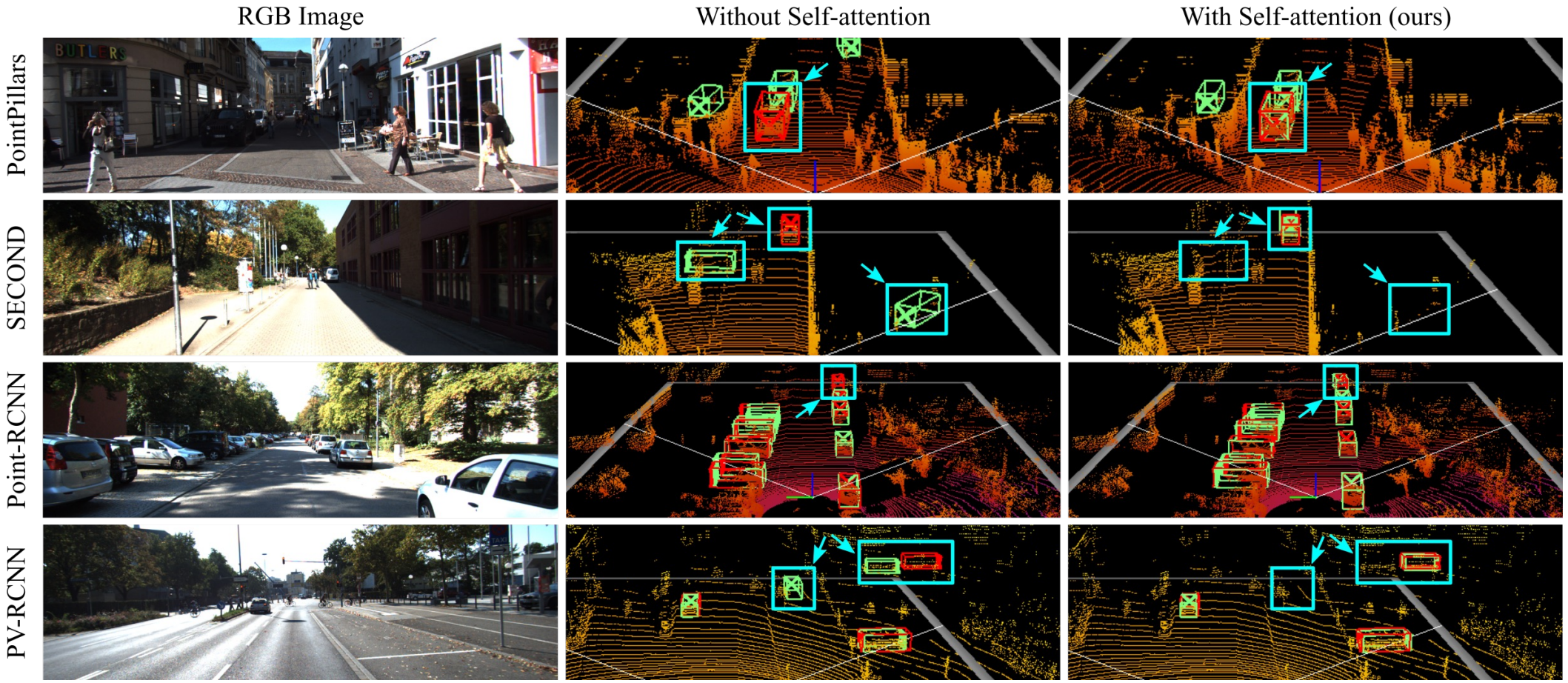
LITERATURE

Method	Task	Modality	Context	Scalability	Attention + Convolution Combination	Stage Added
HG-Net [3]	detection	points	global-static	-	gating	Attention modules are added at the end.
PCAN [55]	place-recognition	points	local-adaptive	-	gating	
Point-GNN [33]	detection	points	local-adaptive	-	-	Attention modules fully replace convolution and set-abstraction layers.
GAC [40]	segmentation	points	local-adaptive	-	-	
PAT [49]	classification	points	global-adaptive	randomly sample points subset	-	
ASCN [47]	segmentation	points	global-adaptive	randomly sample points subset	-	
Pointformer [22]	detection	points	global-adaptive	sample points subset and refine	-	
MLCVNet [46]	detection	points	global-static	-	residual addition	Attention modules are inserted into the backbone.
TANet [17]	detection	voxels	local-adaptive	-	gating	
PMPNet [51]	detection	pillars	local-adaptive	-	gated-recurrent-unit	
SCANet [18]	detection	BEV	global-static	-	gating	
A-PointNet [21]	detection	points	global-adaptive	attend sequentially to small regions	gating	
Ours (FSA/DSA)	detection	points, voxels, pillars, hybrid	global-adaptive	attend to salient regions using learned deformations	residual addition	Attention modules are inserted into the backbone.

ARCHITECTURES

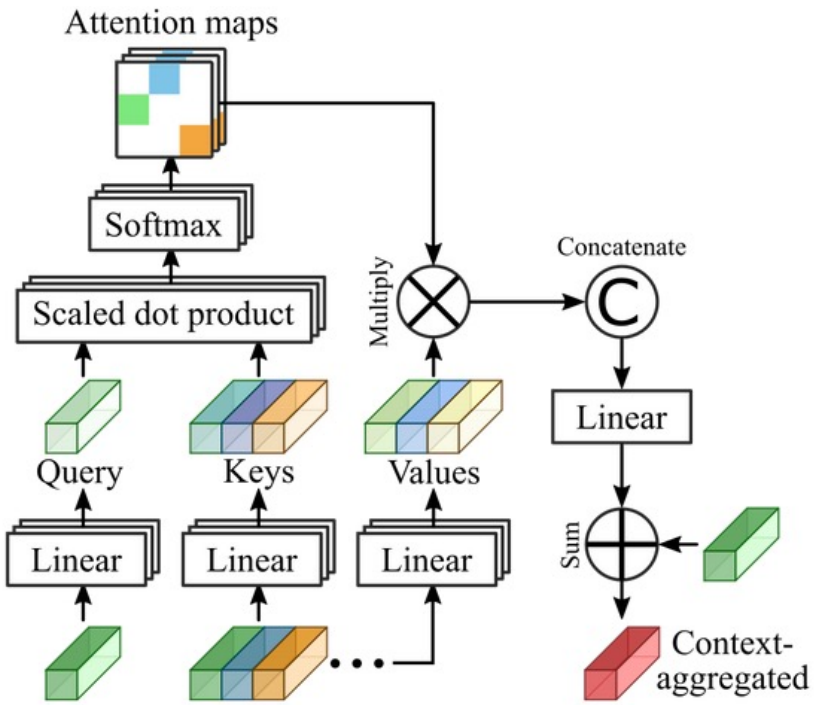


SA-DET3D: QUALITATIVE DETECTION PERFORMANCE



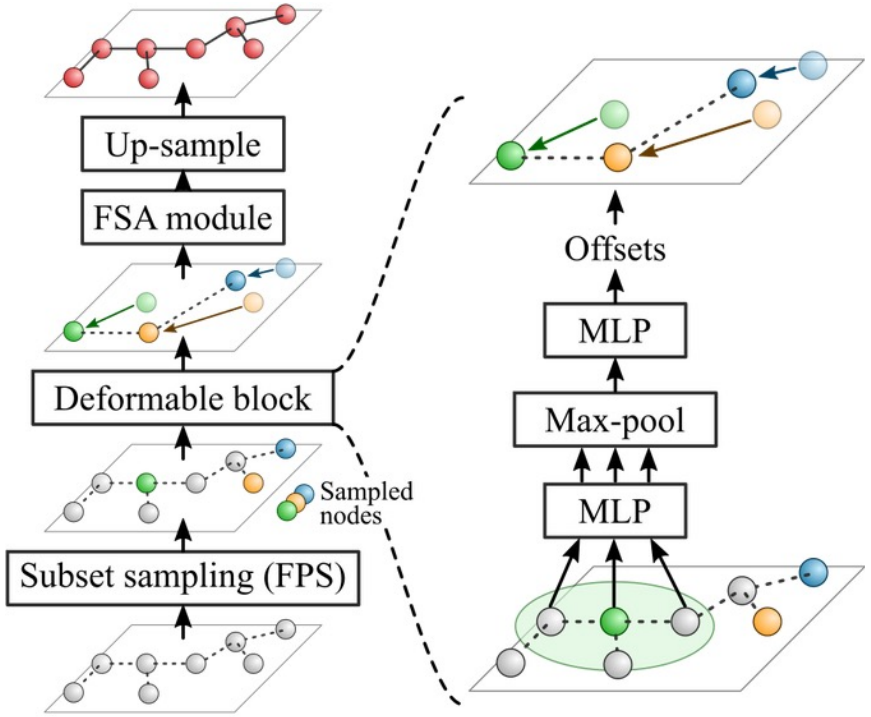
PROPOSED: FSA

- ***Advantages:***
  - Resolution at which context is gathered is independent of the number of parameters – replace a fraction of convolution with FSAs
  - Permutation invariant inductive bias which is necessary for aggregating point-cloud set context
  - Content-adaptive global context
- ***Disadvantages:***
  - Pairwise similarity compute is $O(n^2d)$
  - FSA works because of sparsity in point-clouds but will not scale for global context computation in case of more dense point-cloud/smaller resolutions etc.



PROPOSED: DEFORMABLE SELF-ATTENTION (DSA)

- Compute self-attention context only over $m << n$ of key-points so that the computation is O(*mnd*)
- Sample the key-points differentiably to cover representative locations inspired by the idea of deformable convolutions
- **Advantages**:
  - Can scalably aggregate global context for pillar/voxel/points O(*mnd*) computational complexity
  - Can cover representative regions of the point-cloud improving the feature descriptors



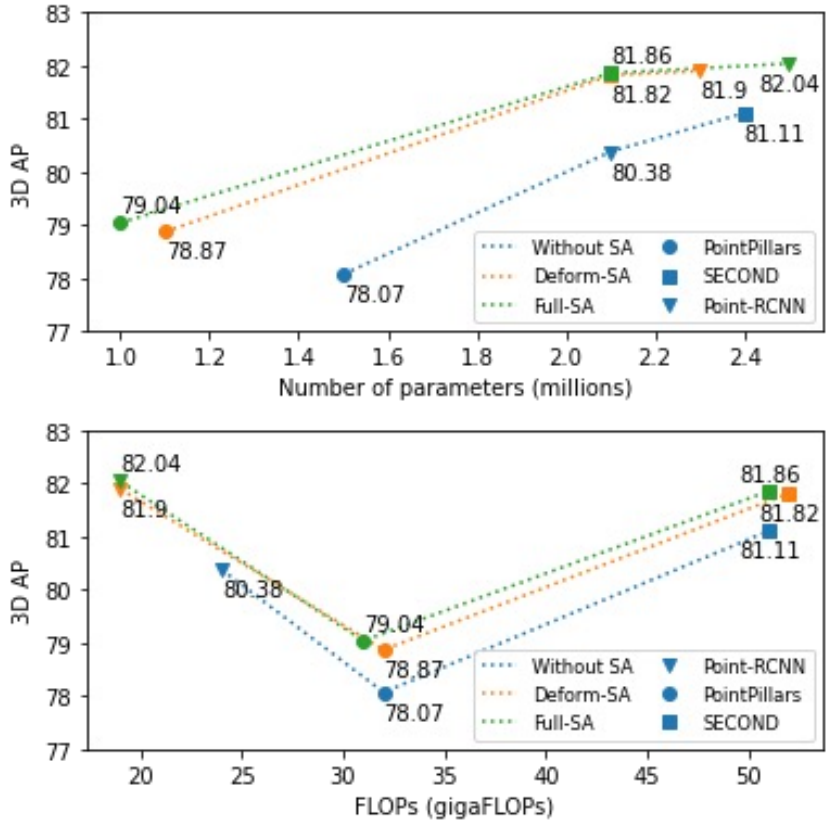
SA-DET3D: KITTI VAL PERFORMANCE

Method	PointPillars [18]				SECOND [48]				Point-RCNN [35]				PV-RCNN [34]			
	3D	BEV	Param	FLOPs	3D	BEV	Param	FLOPs	3D	BEV	Param	FLOPs	3D	BEV	Param	FLOPs
Baseline	78.39	88.06	4.8 M	63.4 G	81.61	88.55	4.6 M	76.9 G	80.52	**88.80**	4.0 M	27.4 G	84.83	**91.11**	12 M	89 G
DSA	78.94	88.39	1.1 M	32.4 G	**82.03**	89.82	2.2 M	52.6 G	81.80	88.14	2.3 M	19.3 G	84.71	90.72	10 M	64 G
FSA	**79.04**	**88.47**	1.0 M	31.7 G	81.86	**90.01**	2.2 M	51.9 G	**82.10**	88.37	2.5 M	19.8 G	**84.95**	90.92	10 M	64.3 G
Improve.	+0.65	+0.41	-79%	-50%	+0.42	+1.46	-52%	-32%	+1.58	-	-37%	-38%	+0.12	-	-16%	-27%

Table 1: Performance comparison for moderate difficulty Car class on KITTI *val* split with 40 recall positions

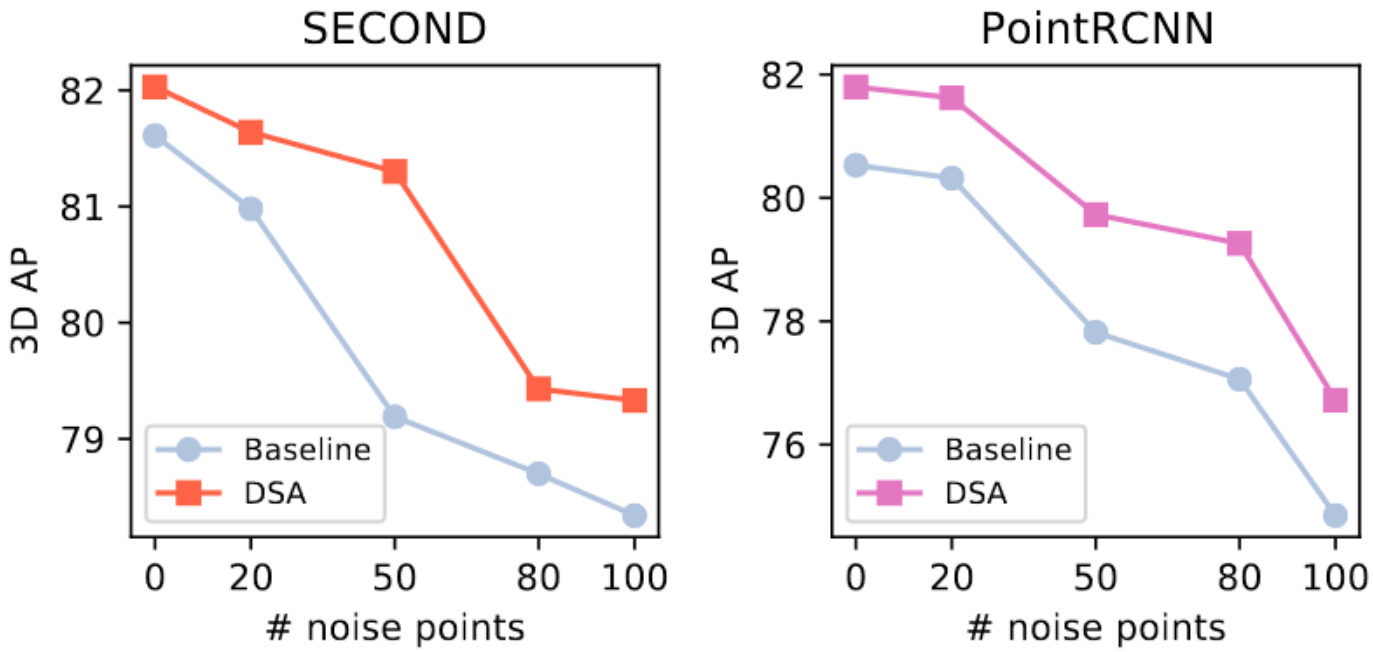
FSA/DSA consistently improves performance for all backbones

SA-DET3D: COMPUTE AND PARAMETER EFFICIENCY



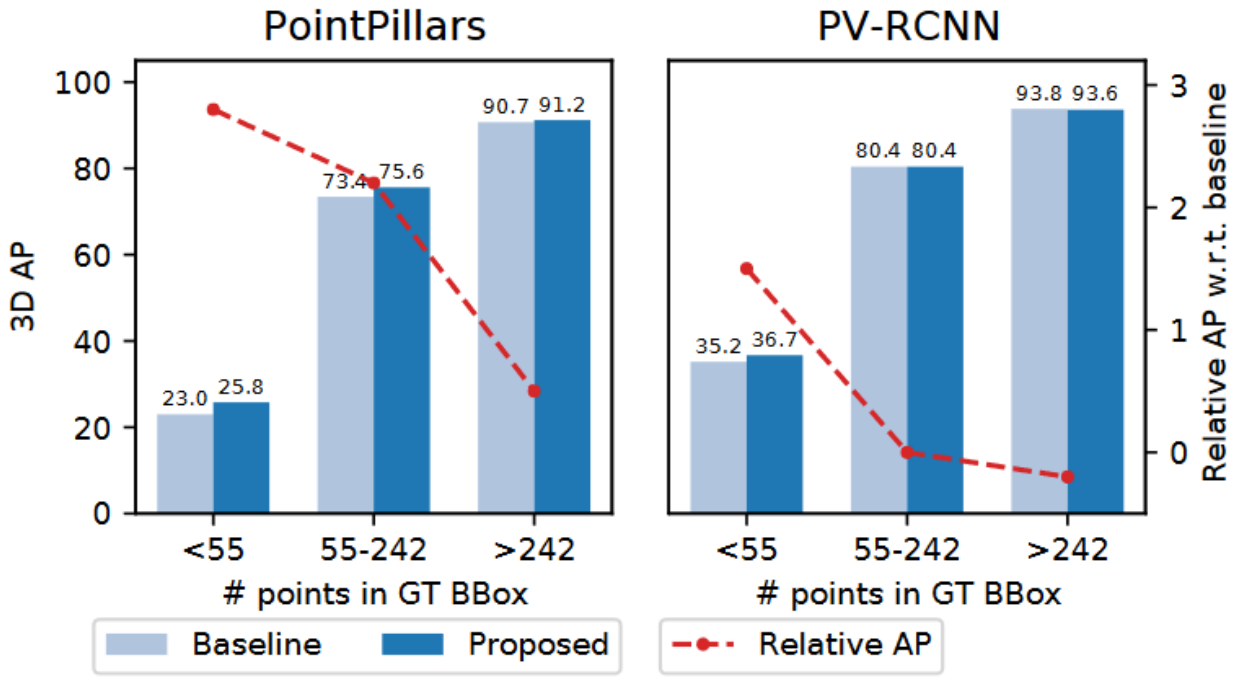
FSA/DSA shows consistent gains in parameter and computation budget across backbones

SA-DET3D: ROBUSTNESS



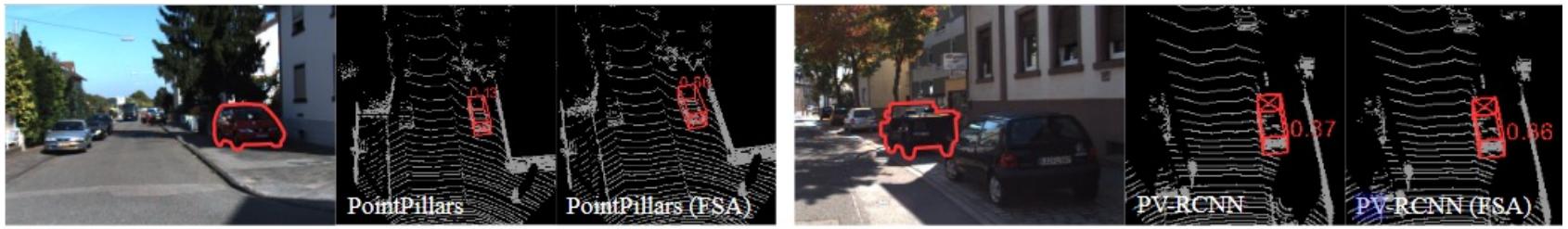
- Add noise uniformly to each bounding box
- Self-attention-augmented models are more robust to noise than the baselines

SA-DET3D: DIFFICULT DETECTIONS

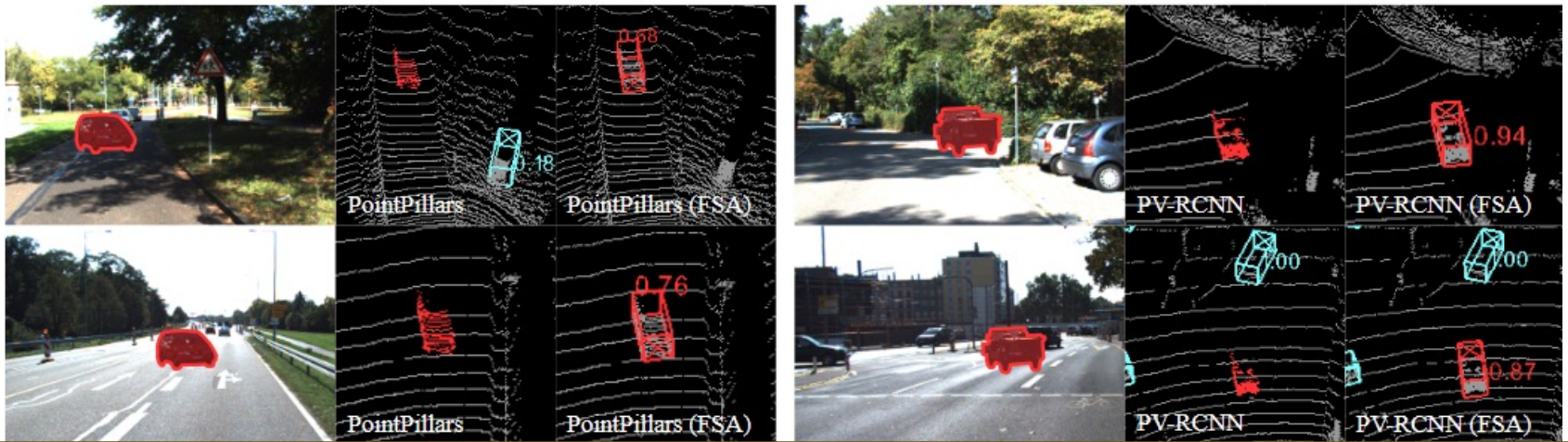


- Sort the cars based on the numbers of points in them in increasing order
- Context seems to be more important for detecting difficult cases

SA-DET3D: COPY-PASTING TO OTHER SCENES



(a) Object in the original scene



(b) Object inserted into another scene

EXTENDING SA-DET3D

SA-Det3D Public

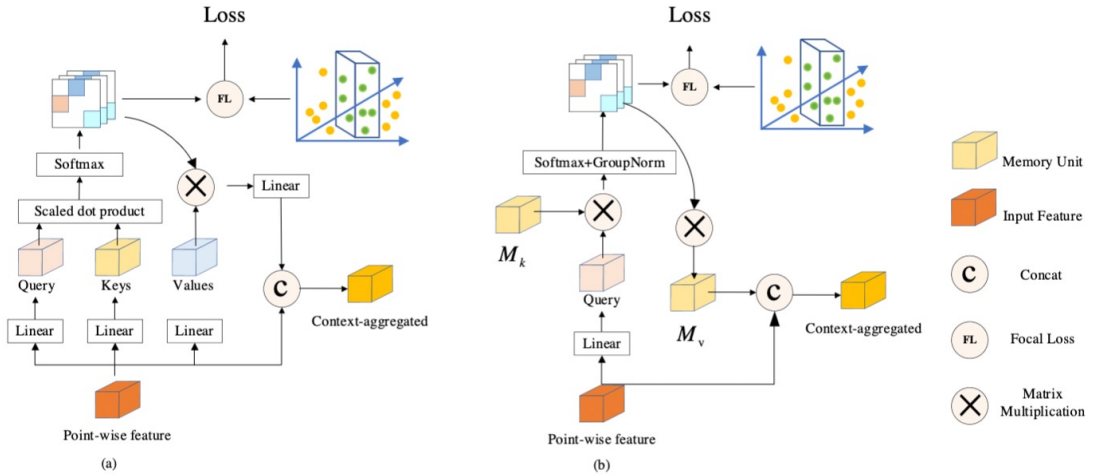
[ICCVW-2021] SA-Det3D: Self-attention based Context-Aware 3D Object Detection

Python ☆ 160 31

Deformable-PV-RCNN Public

[ECCVW-2020] Deformable PV-RCNN: Improving 3D Object Detection with Learned Deformations

Python ☆ 96 12



Qian
202

Complexity. The complexity of the self-attention calculation is $O(N^2d)$; the complexity of the external attention module is $O(Nsd)$, where s is the dimension of the memory block. Based on the number of input point clouds, N, in the square relationship, a smaller value of s was selected in this study to achieve a similar effect as SA. In this study, s was set to 64, and the complexity of PEA was only 1/4 or 1/8 of that of SA for input $N = 512$ or 256. The inherent sparsity of the point cloud and efficient pairwise computing based on matrix multiplication make PSA a feasible feature extractor in the current 3D detection architecture.

TAKEAWAYS

- We explore self-attention for context aggregation in 3D object detection.

- SA-Det3D consistently outperforms strong 3D baselines across different backbones.

- SA-Det3D proposes a variant to exploit deformations to spatially sample key points for relation modeling.

MULTI-AGENT MOTION FORECASTING

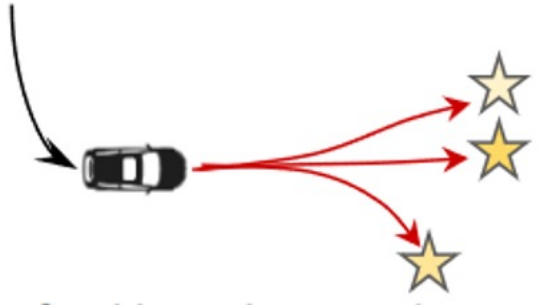
1. **Prarthana Bhattacharyya**, Chengjie Huang, Krzysztof Czarnecki. “SSL-Lanes: Self-Supervised Learning for Motion Forecasting in Autonomous Driving”. Conference on Robot Learning (CoRL), 2022.

2. **Prarthana Bhattacharyya**, Chengjie Huang, Krzysztof Czarnecki. “SSL-Interactions: Pretext Tasks for Interactive Trajectory Prediction”. Preprint, 2023.

UNIVERSITY OF WATERLOO

GOAL: MULTI-AGENT MOTION FORECASTING

- Given:
  - Tracking outputs for visible agents for $t = 1,\ldots t_{obs}$
    - $X_i = \{(x^t_i, y^t_i) \in R^2 |\ t = 1,\ldots,t_{obs}\}$ for $\forall\ i \in \{1,\ldots,N\}$; $X=\{X_1, X_2$
  - Ground truth future trajectory
    - $Y_i = \{(x^t_i, y^t_i) \in R^2 | t = t_{obs}+ 1, \ldots, t_{pred}\}$ for $\forall\ i \in \{1,\ldots,N\}$; $Y=\{Y_1, Y_2,\ldots Y_N\}$
  - Scene information (I_t)
  - Map information (M)



- Our goal is to approximate the underlying (and potentially, multimodal) distribution $P(Y|X,I_t,M)$ which can generate feasible samples for their future trajectories, i.e., $\hat{Y}_i$ for $\forall\ i \in \{1,\ldots,N\}$.

CURRENT DEEP-LEARNING APPROACHES: RNNS

Class	Advantages/Disadvantages	Work	Summary of Prediction Method
Recurrent Neural Networks	- Good at processing temporal dependencies. Single RNN: - Requires additional mechanism to model interaction and contextual features.	[19], [18], [23]	Single RNN: Multi-layer LSTM network is used as a sequence classifier.
		[30]	Single RNN: Two-layer LSTM is used to predict the parameters of acceleration distribution.
		[28]	Single RNN: Single-layer LSTM is used to predict future x-y position of the TV.
		[22]	Single RNN: An encoder-decoder LSTM is used to predict the probability of the occupancy on a grid BEV.
		[27]	Multiple RNNs: A group of GRUs is used to model the pairwise interaction between the TV and each of the SVs.
		[25]	Multiple RNNs: One group of LSTMs is used to model individual vehicles' trajectory, another group is used to model pairwise interaction.
		[21]	Multiple RNNs: One LSTM is used to estimate the target lane, another LSTM is used to predict the trajectory based on estimated target lane.
		[20]	Multiple RNNs: Multi-layer LSTM are used to predict mixtures of Gaussian distribution.
		[24]	Multiple RNNs: One LSTM encoder is applied to the input sequence. The hidden state is fed to six LSTM decoders (one per manoeuvre). Another LSTM encoder is used to predict the probability of each manoeuvre.
		[32]	Multiple RNNs: multiple LSTMs are grouped as two layers: instance layer and category layer. The former learns instance movement and their interactions, while the latter reason about the similarities of the instance in the same category.

Mozaffari et al., Deep Learning-based Behavior Prediction for Autonomous Driving Applications: a Review (2020)

CURRENT DEEP-LEARNING APPROACHES: CNNS AND GNNS

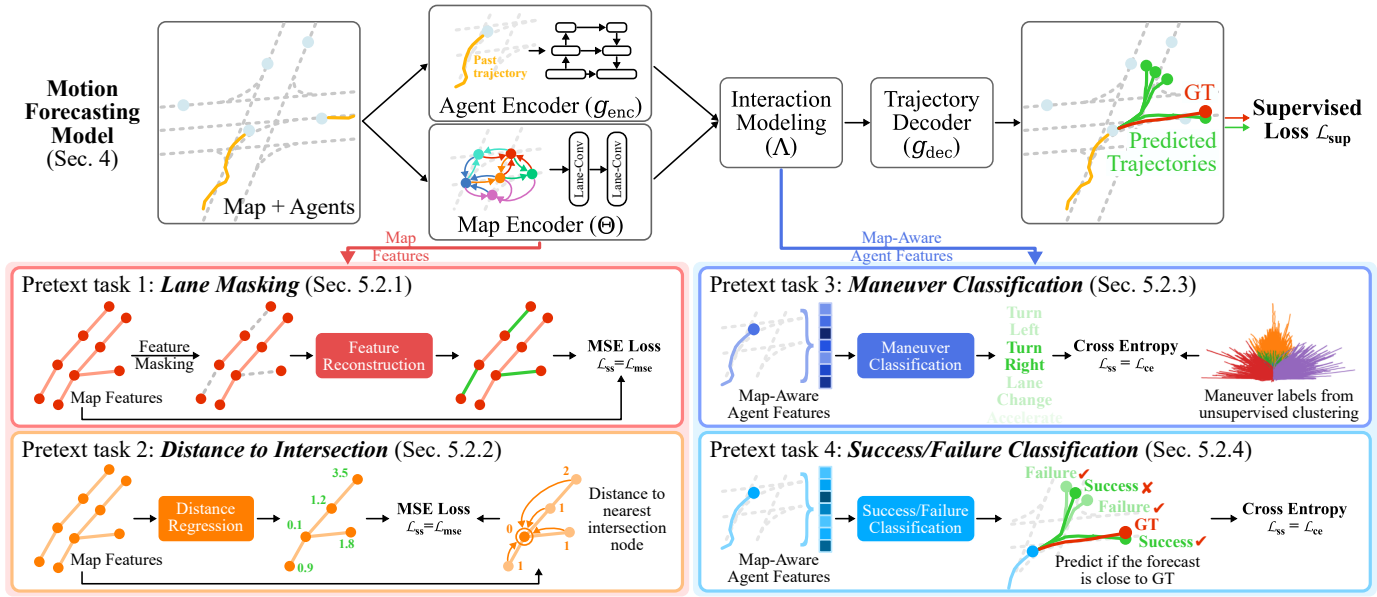
Convolutional Neural Networks	- Good at processing spatial dependencies. - 2D CNNs lack a mechanism to model data series.	[34]	Six layer CNN with convolution and fully connected layers are used to predict the intention of surrounding vehicles.
		[35], [36]	MobileNetV2 [48] is used as feature extractor.
		[41]	A convolution-deconvolution architecture, introduced in [49], is used to predict vehicle behaviour.
		[43]	First, 3D convolutions are applied to the temporal dimension of input data. Then, a series of 2D convolution is used to capture spatial features. Finally, two branches of convolution layers are used to find the probability of being a vehicle and predict the bounding box over current and future frames.
		[44]	First, two backbone CNNs are used to extract the features of lidar data and rasterized map separately. Then three different networks are applied to the concatenation of extracted features to detect vehicles and predict their future intention and trajectory.
Graph Neural Networks: - Comply with graph structure of traffic. - Static scene context is usually neglected.		[31]	Graph Convolutional Network (GCN[50]) and Graph Attention Network (GAT[51]) are used with some adaptations.
		[29]	Graph Convolutional Model is used which consists of several convolutional and graph operation layers.

Mozaffari et al., Deep Learning-based Behavior Prediction for Autonomous Driving Applications: a Review (2020)

LIMITATIONS OF CURRENT METHODS

- Self-supervised learning (SSL) is an emerging technique that has been successfully employed to train convolutional neural networks (CNNs) and graph neural networks (GNNs) for more transferable, generalizable, and robust representation learning.
- However, its potential in motion forecasting for autonomous driving has rarely been explored.
- In this study, *we report the first systematic exploration and assessment of incorporating self-supervision into motion forecasting*.

SSL-LANES: SELF-SUPERVISION + MOTION FORECASTING



- Self- supervision has seen huge interest in both natural language processing and computer vision to make use of freely available data without the need for annotations.

- *Research question: Can self-supervised learning in motion forecasting to improve its accuracy and generalizability, without sacrificing inference speed or architectural simplicity?*

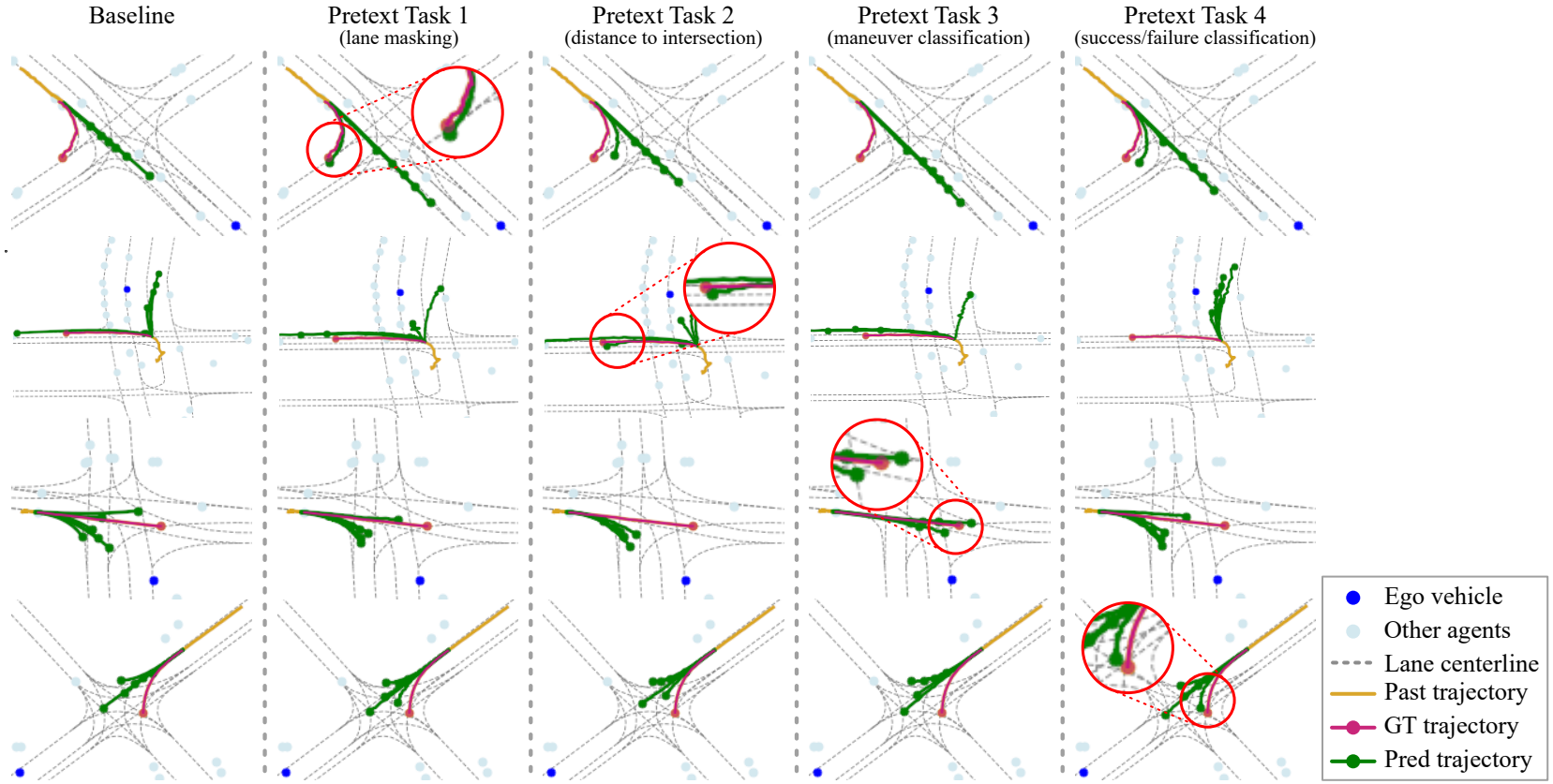
PROPOSED PRETEXT TASKS FOR SSL-LANES

SSL Task	Property Level	Primary Assumption	Type
Lane-Masking	Map features	Local map structure	Aux. auto-encoder
Distance to Intersection		Global map structure	Aux. regression
Maneuver Classification	Map-aware agent features	Agent feature similarity	Aux. classification
Success/Failure Classification		Distance to success state	

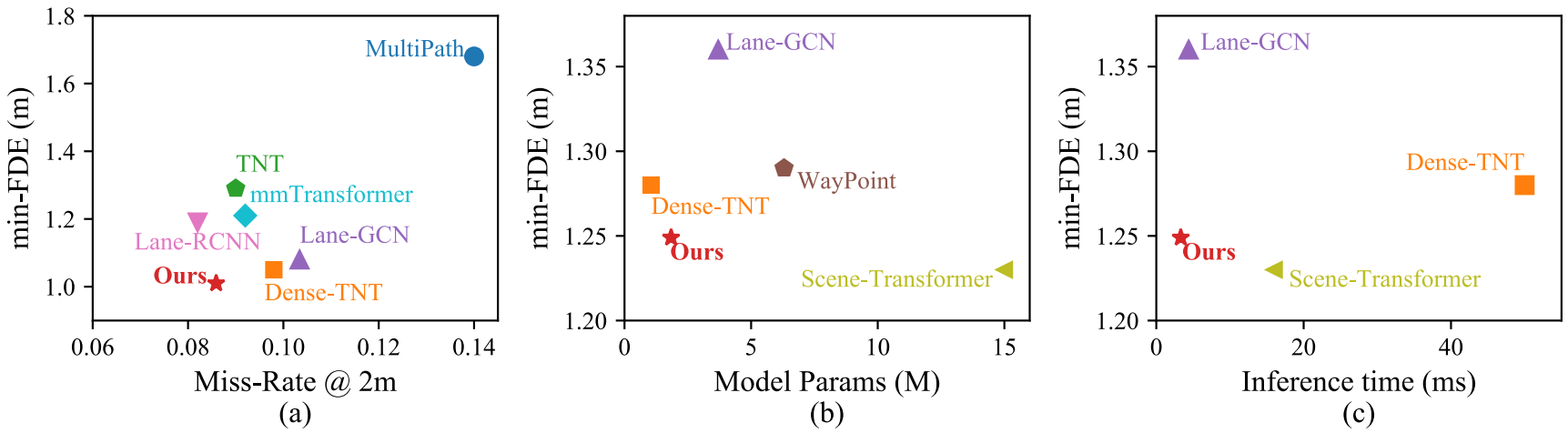
- At the core of our SSL-Lanes approach is defining pretext tasks based upon self-supervised information from the underlying map structure and the overall temporal prediction problem itself.
- Our proposed prediction-specific self- supervised tasks assign different pseudo-labels from unannotated data. The self-supervised task acts as the regularizer learned from unlabeled data under the minor guidance of human prior (design of pretext task).

SSL-LANES: QUALITATIVE PERFORMANCE

- Motion forecasting on Argoverse validation.
- We show four challenging scenarios at intersections.
- The baseline misses all the predictions.
- Our proposed pretext tasks are successful at predicting the future trajectory.



SSL-LANES: COMPARISON TO STATE-OF-THE-ART



- We optimize both min-FDE and miss-rate successfully in comparison to other popular approaches on Argoverse validation dataset.
- Our work achieves the best trade-off (lower-left) with respect to accuracy, computation and parameters.

DOES SSL BENEFIT FORECASTING?

Description	Experimental Setup		Method	minADE_6	minFDE_6	MR_6
	Training	Validation				
Effects of limited training data	25% of train	All	Baseline	0.82	1.33	14.66
			Ours	**0.78**	**1.22**	**12.63**
Effects of new domain	100% PIT + 20% MIA	MIA val	Baseline	0.88	1.46	17.21
			Ours	**0.85**	**1.34**	**14.96**
Performance on difficult maneuvers	All	Turning & lane changing	Baseline	0.90	1.53	19.90
			Ours	**0.84**	**1.34**	**14.93**
Effects of imbalanced data	2x straight 1x other maneuvers	Turning & lane changing	Baseline	0.94	1.65	21.53
			Ours	**0.90**	**1.49**	**17.97**
Effects of noisy data	All	Gaussian noise ($\sigma = 0.2$) with $p = 0.25$	Baseline	1.01	1.37	15.59
			Ours	**0.96**	**1.24**	**11.98**
Effects of noisy data	All	Gaussian noise ($\sigma = 0.2$) with $p = 0.5$	Baseline	1.19	1.56	20.64
			Ours	**1.13**	**1.40**	**15.65**

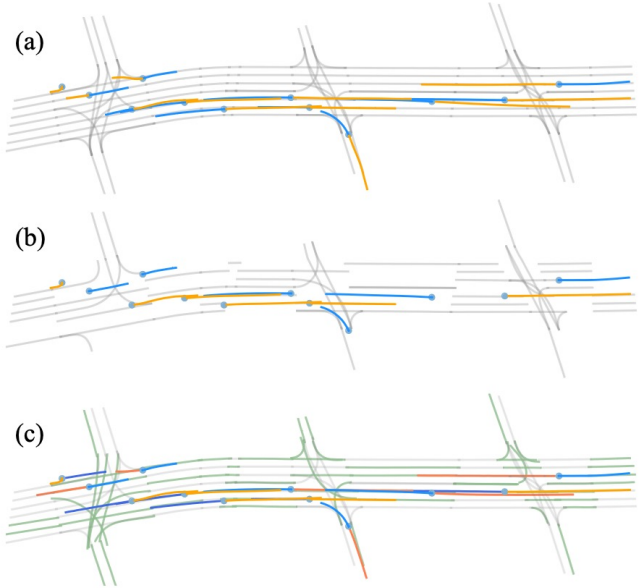
- We further design experiments to explore why forecasting benefits from SSL.
- We provide results to hypothesize that SSL-Lanes learns richer features as compared to a model trained with vanilla supervised learning.

EXTENDING SSL-LANES

SSL-Lanes Public

[CoRL-2022] SSL-Lanes: Self-Supervised Learning for Motion Forecasting in Autonomous Driving

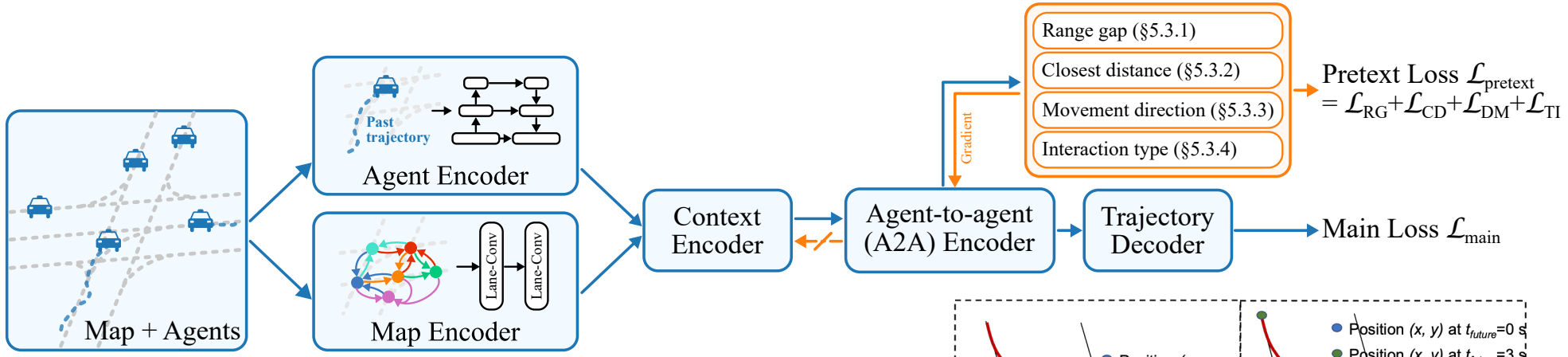
Jupyter Notebook ☆ 91 14



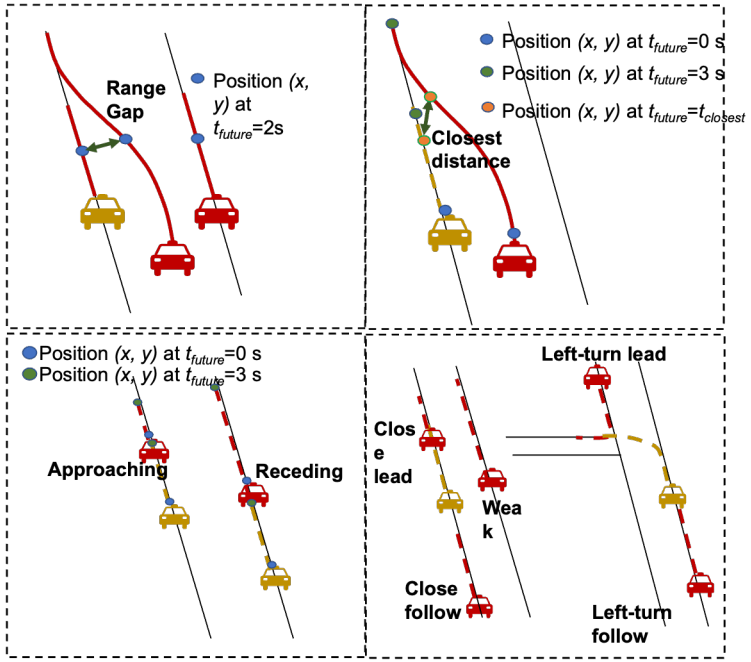
recent vector-based or graph-based models. However, another pioneering work, SSL-Lanes [1], demonstrated that carefully designed pretext tasks can significantly enhance performance without using extra data by learning richer features. In this paper, we follow this approach to learn better and more generalized features using the existing dataset.

Cheng et al., Forecast-MAE: Self-supervised Pre-training for Motion Forecasting with Masked Autoencoders, ICCV 2023

SSL-INTERACTIONS: PRETEXT TASKS FOR INTERACTIVE TRAJECTORY PREDICTION



Model	A2A	Pretext Task	i-minFDE$_6$ ↓ All-Interactive	i-minFDE$_6$ ↓ Strong-Interactive
Baseline	✓	✗	1.279	1.320
Ours	✓	Range-gap	1.202 (+6.0%)	1.242 (+5.9%)
Ours	✓	Closest-distance	1.192 (+6.8%)	1.235 (+6.4%)
Ours	✓	Direction of Movement	**1.175** (+8.1%)	**1.216** (+7.9%)
Ours	✓	Type Of Interaction	1.183 (+7.5%)	1.226 (+7.1%)



TAKEAWAYS

- We explore self-supervised learning for motion forecasting by proposing pretext tasks.

- We improve future prediction in challenging scenes at intersections and showing better generalization.

- We improve future prediction performance on interaction-heavy scenes.

SUMMARY AND FUTURE WORK

LEARNINGS AND OPEN QUESTIONS

- 3D Multi-Object Tracking improves with end-to-end data-driven training
  - Best way to integrate tracking and its associated uncertainty within the autonomy stack
  - Generative models may improve occluded, lowly lit cases for data association
- 3D Multi-Object Detection improves with context-based aggregation
  - Leveraging unlabeled data to improve to out-of-distribution cases
- Self-supervision improves and generalizes motion forecasting performance
  - Investigation of more effective metrics for evaluating motion forecasting
  - Intermediate representations to get best of both worlds for forecasting and planning
- Exploration of learning pre-trained representations from cross-domains

THANK YOU FOR YOUR TIME!