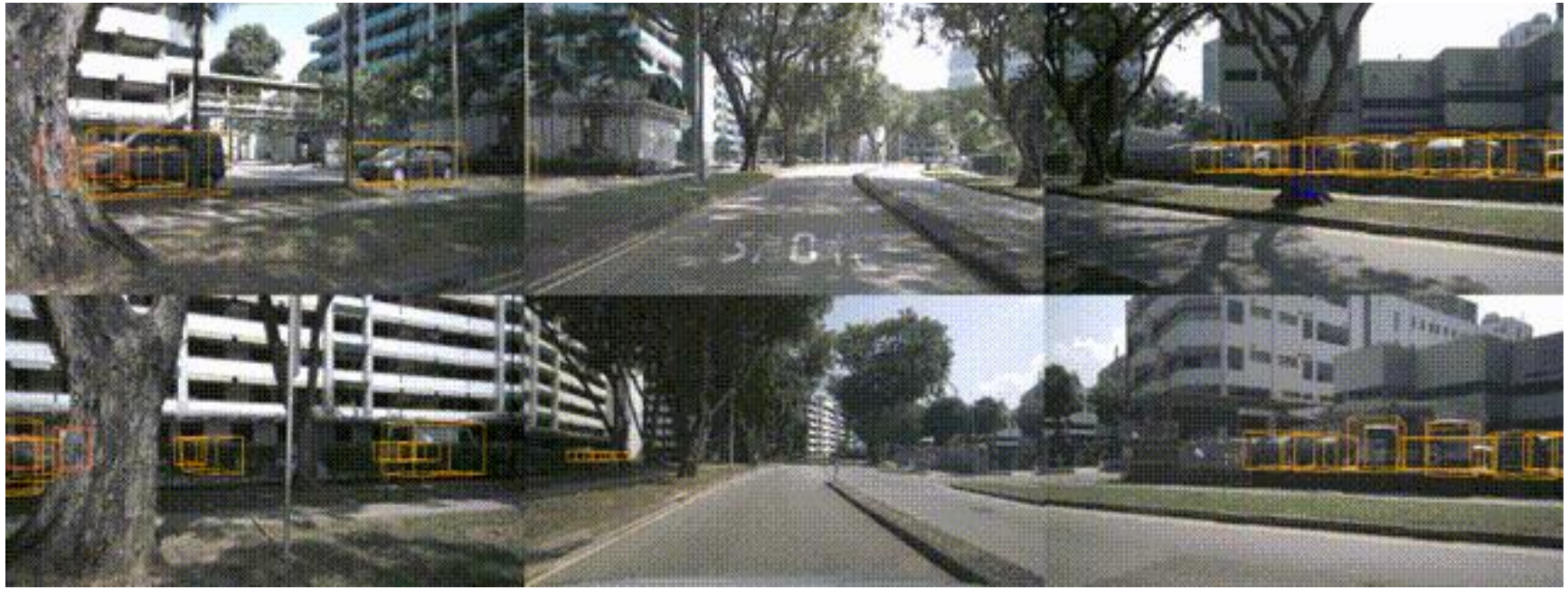
Query-based Temporal Fusion with Explicit Motion for 3D Object Detection

Jinghua Hou, Zhe Liu, Dingkang Liang, Zhikang Zou, Xiaoqing Ye, Xiang Bai

Temporal information is important for 3D perception in autonomous driving.



Motivation

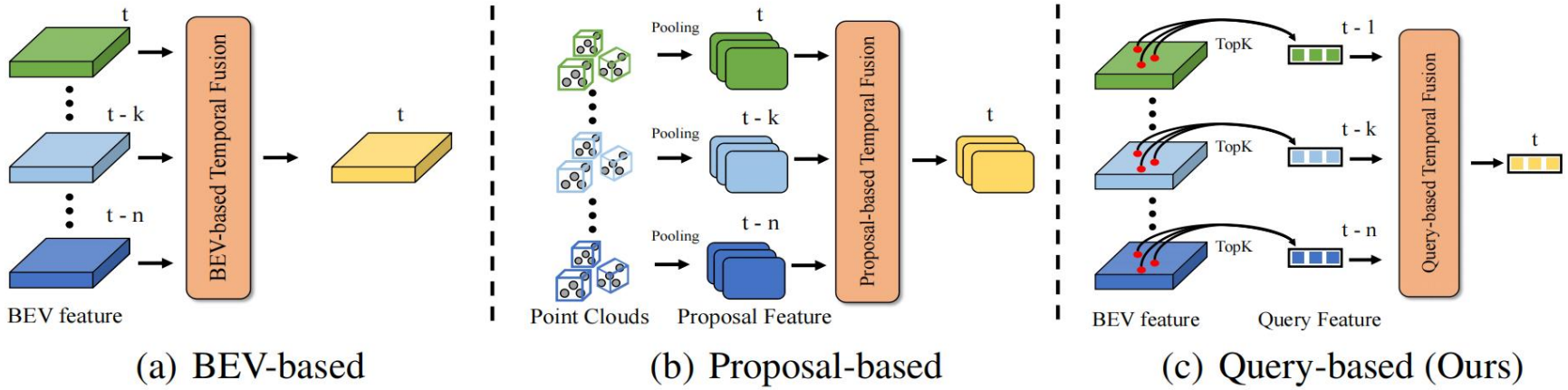
1. Existing methods either conduct temporal fusion based on the dense BEV features or sparse 3D proposal features.

2. BEV-based methods do not pay more attention to foreground objects, leading to more computation costs and sub-optimal performance.

3. Proposal-based methods implement time-consuming operations to generate sparse 3D proposal features, and the performance is limited by the quality of 3D proposals.

How to do?

Query is a better representation for temporal modeling.



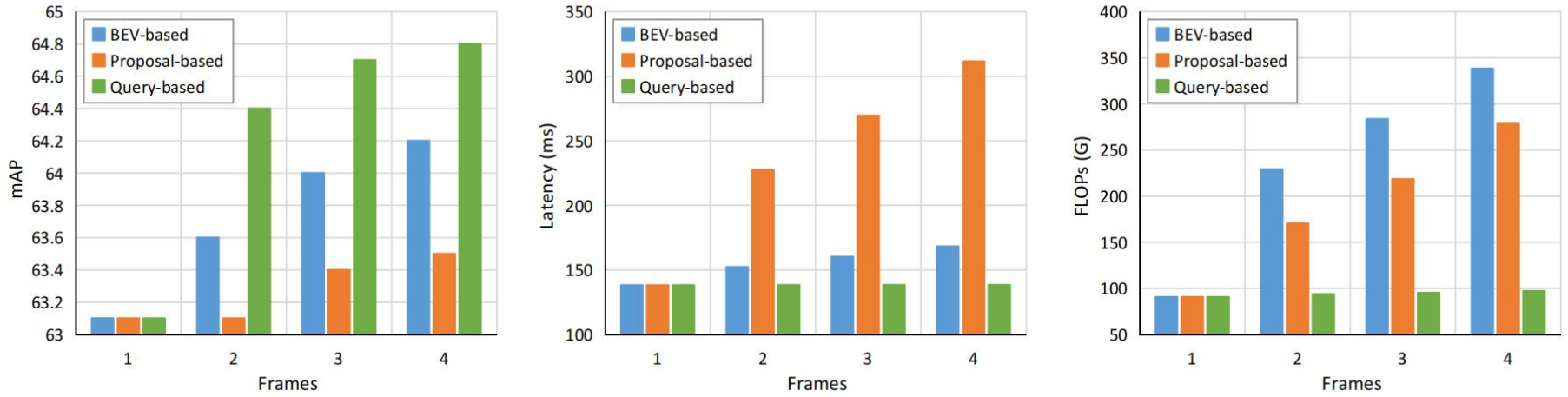
Why?

1. On the one hand, the query-based representation is sparse and can effectively aggregate the foreground object information by attention operation.

2. On the other hand, the query-based format is more efficient due to getting rid of complex 3D RoI operations and is less sensitive to 3D object of size and orientation than proposal-based representation.

Contribution

1. We propose a new temporal fusion method called QTNet for 3D object detection based on sparse query-based feature representation, which is more effective and efficient than BEV-based and proposal-based manners.
2. We propose the MTM module, which can be plugged into LiDAR-only or multi-modality 3D detectors and boost their performance with negligible computation cost and latency.



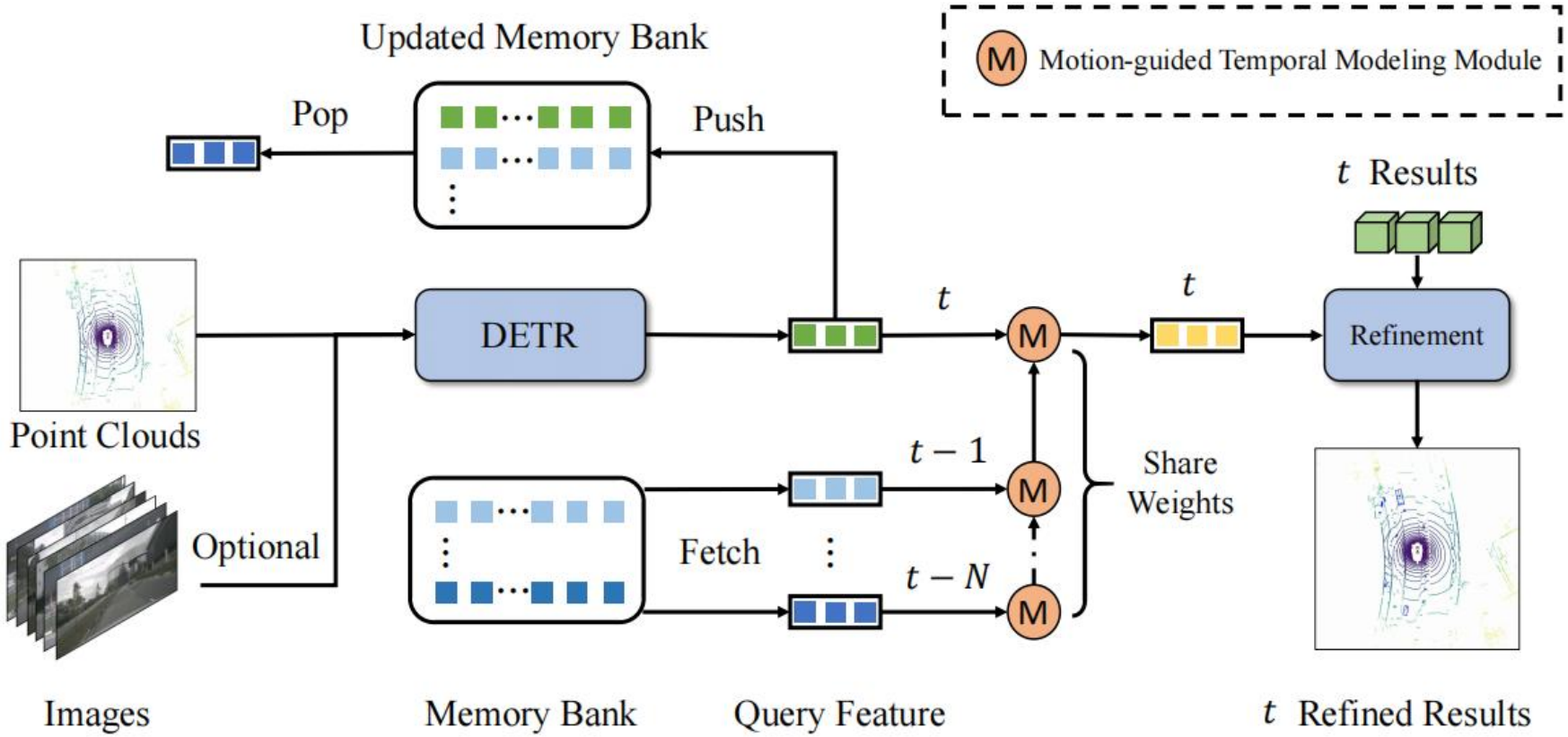
Problem

However, the temporal modeling based on query representation is non-trivial to work well.

Problem: The standard cross attention with position embedding is not work well.

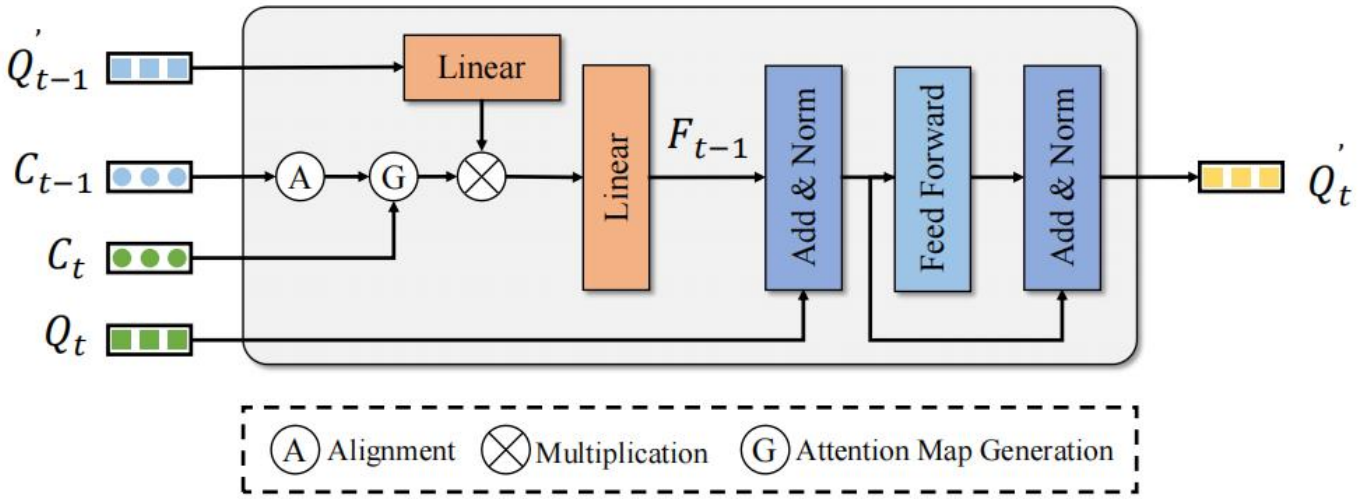
Solution: Adopt the explicit motion of objects for temporal modeling.

Architecture



QTNet consist of a detector, a memory bank, and a MTM module.

MTM



$$R_{t-1}^{t} = R_{w}^{t} \cdot \mathrm{inv}(R_{w}^{t-1})$$

$$C'_{t-1} = (C_{t-1} + V_{t-1} \cdot \Delta t) \cdot (R_{t-1}^{t})^{T}$$

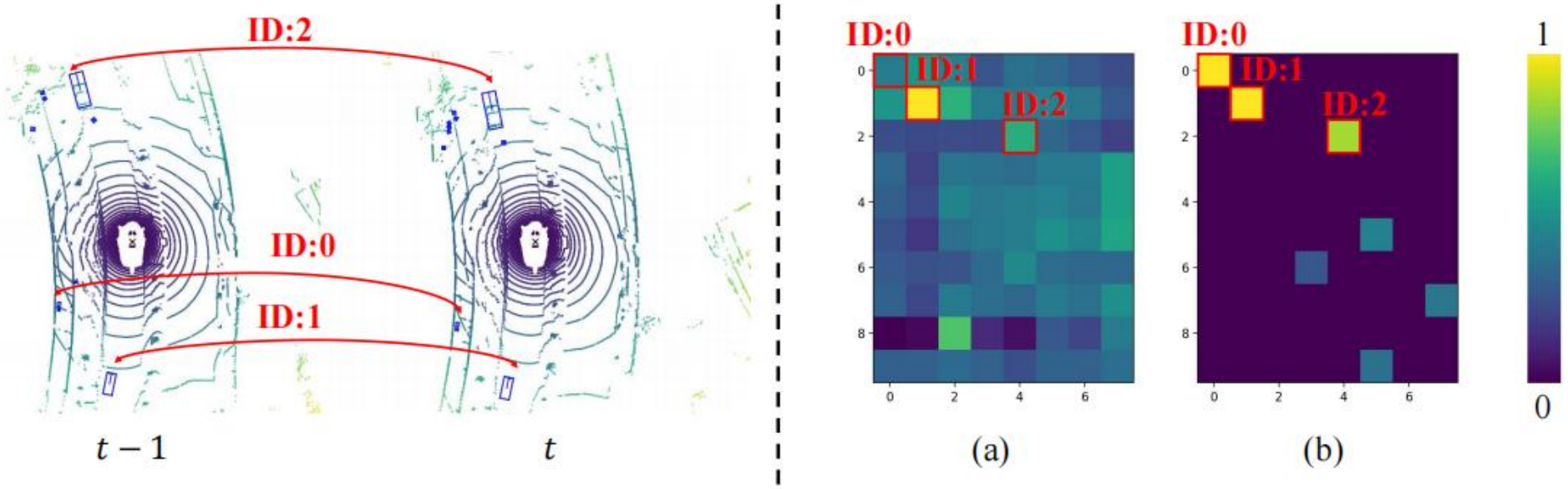
$$O_{t-1}^{t} = \mathrm{L}_2(C_t - C'_{t-1})$$

$$M = \begin{cases} 0, & O_{t-1}^{t} \leq \gamma \quad and \quad s_t = s_{t-1} \\ 1e^{8}, & O_{t-1}^{t} > \gamma \quad \text{or} \quad s_t \neq s_{t-1} \end{cases}$$

$$A = \mathrm{softmax}(-1 \cdot O_{t-1}^{t} + M)$$

We utilize the aligned location of objects in previous frame and current frames for caculating the euclidean distance.

Cross Attention vs MTM



The attention map (a) generated by cross attention is ambiguous. The attention map (b) generated by MTM is more discriminative than cross attention.

Results

Table 1: Comparison with state-of-the-art methods on the nuScenes validation set. The L and C represent LiDAR and camera, respectively. The column of Frames denotes the number of key frame. * denotes our reproduced results. † denotes future information is used.

Method	Year	Modality	Frames	Backbones	mAP↑	NDS↑
MVP [51]	NeurIPS 2021	LC	1	DLA34 & VoxelNet	67.1	70.8
AutoAlignV2 [8]	ECCV 2022	LC	1	CSPNet & VoxelNet	67.1	71.2
TransFusion [2]	CVPR 2022	LC	1	R50 & VoxelNet	67.5	71.3
BEVFusion [22]	NeurIPS 2022	LC	1	Swin-Tiny & VoxelNet	67.9	71.0
DeepInteraction [47]	NeurIPS 2022	LC	1	R50 & VoxelNet	69.9	72.6
BEVFusion [28]	ICRA 2023	LC	1	Swin-Tiny & VoxelNet	68.5	71.4
MSMDFusion [15]	CVPR 2023	LC	1	R50 & VoxelNet	69.3	72.1
CMT [44]	ICCV 2023	LC	1	V2-99 & VoxelNet	70.3	72.9
TransFusion* [2]	CVPR 2022	LC	1	R50 & VoxelNet	67.1	70.7
+QTNet	-	LC	4	R50 & VoxelNet	68.5	71.6
DeepInteraction* [47]	NeurIPS 2022	LC	1	R50 & VoxelNet	69.9	72.6
+QTNet	-	LC	4	R50 & VoxelNet	**70.3**	**73.1**
CenterPoint [50]	CVPR 2021	L	1	VoxelNet	59.6	66.8
TransFusion-L [2]	CVPR 2022	L	1	VoxelNet	65.1	70.1
INT [43]	ECCV 2022	L	10	VoxelNet	61.8	67.3
LidarMultiNet [48]	AAAI 2023	L	1	VoxelNet	63.8	69.5
VoxelNeXt [7]	CVPR 2023	L	1	VoxelNet	60.0	67.1
LargeKernel3D [6]	CVPR 2023	L	1	Voxel-LargeKernel3D	63.3	69.1
LinK [30]	CVPR 2023	L	1	Voxel-LinK	63.6	69.5
MGTANet [16]	AAAI 2023	L	3	VoxelNet	62.9	68.7
MGTANet† [16]	AAAI 2023	L	3	VoxelNet	64.8	70.6
TransFusion-L* [2]	CVPR 2022	L	1	VoxelNet	65.0	70.0
+QTNet	-	L	3	VoxelNet	**66.3**	**70.8**
+QTNet	-	L	4	VoxelNet	**66.5**	**70.9**

Table 2: Comparison with state-of-the-art methods on the nuScenes test set. † denotes future information is used.

Method	Year	Modality	Frames	Backbones	mAP↑	NDS↑
CenterPoint [50]	CVPR 2021	L	1	VoxelNet	60.3	67.3
TransFusion-L [2]	CVPR 2022	L	1	VoxelNet	65.5	70.2
VISTA [10]	CVPR 2022	L	1	VoxelNet	63.7	70.4
LidarMultiNet [48]	AAAI 2023	L	1	VoxelNet	67.0	71.6
VoxelNeXt [7]	CVPR 2023	L	1	VoxelNet	64.5	70.0
LargeKernel3D [6]	CVPR 2023	L	1	Voxel-LargeKernel3D	65.3	70.5
LinK [30]	CVPR 2023	L	1	Voxel-LinK	66.3	71.0
3DVID† [49]	TPAMI 2021	L	3	VoxelNet	65.4	71.4
MGTANet† [16]	AAAI 2023	L	3	VoxelNet	65.4	71.2
QTNet	-	L	3	VoxelNet	68.2	72.0
QTNet	-	L	4	VoxelNet	**68.4**	**72.2**

Results

Table 3: Comparison of computation cost and latency on the nuScenes validation set. * denotes our reproduced results. The FLOPs and Latency are tested on a single NVIDIA RTX 4090 GPU with the batch size of 1.

Method	Modality	mAP (%)↑	NDS (%)↑	FLOPs (G)↓	Latency (ms)↓	Params (M)
TransFusion* [2]	LC	67.1	70.7	445.9	201.2	37.0
+QTNet	LC	68.5	71.6	446.4	207.7	37.7
DeepInteraction* [47]	LC	69.9	72.6	499.2	355.0	57.8
+QTNet	LC	**70.3**	**73.1**	499.7	361.5	58.5
TransFusion-L* [2]	L	65.0	70.0	90.7	138.2	8.3
+QTNet	L	**66.5**	**70.9**	90.8	144.7	8.6

Method	Year	Modality	Frames	Resolution	Backbone	mAP	NDS	mATE	mASE	mAOE	mAVE	mAAE
CAPE* [42]	CVPR 2023	C	1	704 × 256	R50	27.5	35.9	0.794	0.286	0.642	0.847	0.215
+MTM	-	C	2	704 × 256	R50	**31.6**	**43.8**	**0.752**	**0.277**	**0.558**	**0.438**	**0.182**
CAPE* [42]	CVPR 2023	C	1	800 × 320	V2-99	39.7	46.3	0.693	0.270	0.438	0.747	0.206
+MTM	-	C	2	800 × 320	V2-99	**43.9**	**53.6**	**0.656**	**0.266**	**0.380**	**0.350**	**0.183**

Visualization

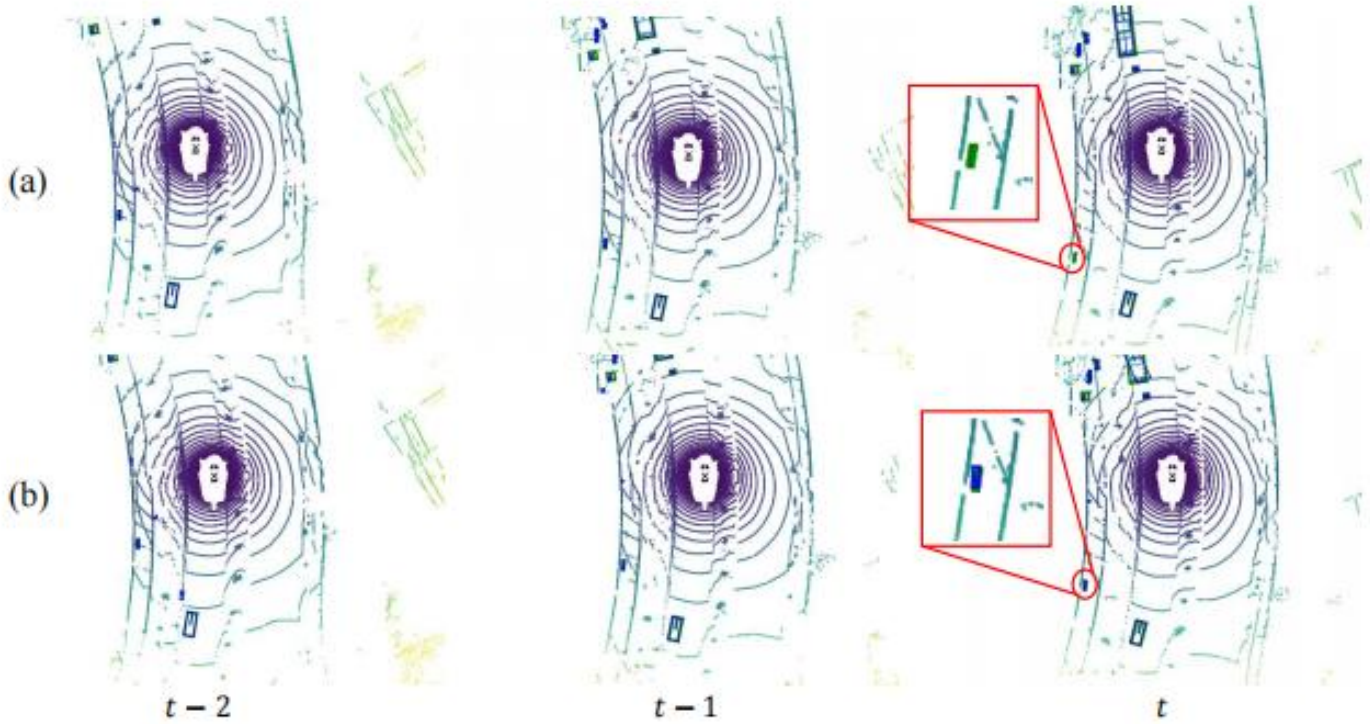


Figure 8: Comparison of LiDAR-only baseline TransFusion-L (a) and QTNet (b) along the temporal dimension. The ego vehicle is moving from bottom to top.

Thanks