A Supplementary Material

In the supplementary material, we present ablative studies regarding augmentation strategies in Section A.1. Further, we list additional quantitative results including class-wise evaluation results in Section A.2. To highlight the advantage of a radar-camera fusion, we provide results of a filtered evaluation for rain and night scenes in Section A.3, in which radar is especially useful. Finally, additional qualitative examples are provided in Section A.4.

A.1 Ablations

In Table 4 we show the impact of different augmentation strategies on our RC-BEVFusion with BEVFeatureNet and BEVDet. As in [9], we use a two-stage augmentation. First, 2D augmentation is applied by rotating, resizing and horizontally flipping the images. In the BEV view transformer, the 2D augmentations are reversed so that the original orientation is retained. Then, 3D augmentations are applied to the BEV features, radar points and ground truth boundings boxes. They include scaling, rotating, as well as horizontal and vertical flipping. Our results show that 3D augmentation is very important, while 2D augmentation helps to improve the results only slightly. This is due to the camera encoder branch being shared across the six cameras, leading to more variety in the images than in the BEV plane. Therefore, the BEV encoder is more prone to overfitting and requires more augmentation [9].

Table 4: Ablation study for 2D and 3D augmentations. All experiments conducted with the model based on BEVFeatureNet and BEVDet.

3D	2D	mAP↑	NDS↑	mATE↓	mAVE↓
		0.380	0.453	0.595	0.676
✓		0.419	0.513	0.520	0.478
✓	✓	**0.434**	**0.525**	**0.511**	**0.421**

Further, we examine two potential augmentation strategies for our RadarGridMap encoder, which have been proposed based on a similar setup in [39]. To combat the sparsity of the radar grid, a blurring filter spreads values from the reference cell to neighboring cells depending on the number of detections per reference cell. Further, the authors find that the distribution of compensated Doppler values is heavy-sided towards zero and introduce a Doppler skewing function to spread the distribution for values close to zero. The results are shown in Table 5. The grid mapping variant without extra augmentation works quite well, confirming the effectiveness of the approach. The blurring filter has little impact on the results and thus is not deemed necessary. The Doppler skewing function leads to higher velocity errors and should therefore be omitted.

Table 5: Ablation study for augmentation strategies of the RadarGridMap encoder: a blurring filter (BF) and a Doppler skewing (DS) technique. All experiments conducted with the model based on BEVDet.

BF	DS	mAP↑	NDS↑	mATE↓	mAVE↓
		0.429	**0.525**	0.523	**0.412**
✓		0.424	0.524	0.517	0.439
	✓	0.425	0.508	0.523	0.482
✓	✓	0.423	0.487	**0.516**	0.650

Table 6: Experimental class-wise results for our radar-camera fusion used in different architectures on the three most common dynamic classes of the nuScenes val split. *We use the implementation of BEVDet-Tiny with a heavier view transformer from [21]. †We list the results as reported by the authors.

	Cam. model	Radar model	Class		AP↑	ATE↓	ASE↓	AOE↓	AVE↓	AAE↓
[9]	BEVDet*	None	car		0.538	0.529	0.157	0.128	0.992	0.232
			ped.		0.377	0.700	0.304	1.383	0.872	0.759
			truck		0.290	0.679	0.209	0.165	0.911	0.252
Ours	BEVDet*	BEVFeatureNet	car		0.700	0.315	0.156	0.106	0.395	0.196
				Δ_r	30%	-40%	-1%	-17%	-60%	-16%
			ped.		0.468	0.528	0.300	1.016	0.701	0.329
				Δ_r	24%	-25%	-1%	-27%	-20%	-57%
			truck		0.405	0.493	0.206	0.131	0.329	0.202
				Δ_r	40%	-27%	-1%	-21%	-64%	-20%
[17]†	BEVDepth	None	car		0.559	0.475	0.157	0.112	0.370	0.205
			ped.		0.363	0.690	0.297	0.831	0.491	0.244
			truck		0.270	0.659	0.196	0.103	0.356	0.181
Ours	BEVDepth	BEVFeatureNet	car		0.661	0.356	0.162	0.134	0.289	0.193
				Δ_r	18%	-25%	3%	20%	-22%	-6%
			ped.		0.410	0.585	0.295	0.732	0.434	0.208
				Δ_r	13%	-15%	-1%	-12%	-12%	-15%
			truck		0.332	0.563	0.214	0.162	0.254	0.198
				Δ_r	23%	-15%	9%	57%	-29%	9%
[16]†	BEVStereo	None	car		0.567	0.457	0.156	0.104	0.343	0.204
			ped.		0.402	0.653	0.297	0.803	0.479	0.249
			truck		0.299	0.650	0.205	0.103	0.321	0.197
Ours	BEVStereo	BEVFeatureNet	car		0.687	0.324	0.159	0.106	0.250	0.192
				Δ_r	21%	-29%	2%	2%	-27%	-6%
			ped.		0.469	0.530	0.295	0.694	0.413	0.197
				Δ_r	17%	-19%	-1%	-14%	-14%	-21%
			truck		0.364	0.516	0.208	0.106	0.214	0.184
				Δ_r	22%	-21%	1%	3%	-33%	-7%
[54]	MatrixVT	None	car		0.517	0.529	0.162	0.155	1.049	0.221
			ped.		0.309	0.746	0.300	1.204	0.813	0.465
			truck		0.244	0.713	0.213	0.154	0.917	0.219
Ours	MatrixVT	BEVFeatureNet	car		0.658	0.346	0.162	0.141	0.400	0.190
				Δ_r	27%	-35%	0%	-9%	-62%	-14%
			ped.		0.386	0.618	0.298	1.071	0.695	0.335
				Δ_r	25%	-17%	-1%	-11%	-15%	-28%
			truck		0.320	0.547	0.214	0.133	0.337	0.201
				Δ_r	31%	-23%	0%	-14%	-63%	-8%

A.2 Class-wise results

In addition to the quantitative evaluation in Section 4.2 and Table 2, we present some more detailed, class-wise results in Table 6. To reduce the amount of data, we only list values of the three most common dynamic classes: car, pedestrian, and truck. The results show that the average precision for important classes increased across the board, with an even greater increase for the larger classes that are more easily detectable by radar: car and especially truck. Translation errors decreased most notably for cars, while scale errors remained relatively unchanged, except for a slight decrease for our model based on BEVDepth [17], which may be due to statistical effects. Orientation errors improved for models without temporal fusion, but increased for our model based on BEVDepth, possibly indicating difficulty in estimating the orientation of some additionally detected cars and trucks. Velocity errors saw a significant reduction, especially for cars and trucks and for models without temporal fusion. Even with temporal fusion, there was a considerable decrease in velocity error. Finally, the attribute error decreased most for pedestrians, with radar making it easier to determine if a pedestrian is moving or standing.

A.3 Results for rain and night scenes

In this section, we want to evaluate the performance of our radar-camera fusion in adverse conditions for the camera. We therefore filter the scene descriptions in the nuScenes [1] validation set for the terms "rain" and "night", respectively, to obtain 27 rain and 15 night scenes, on which we run the evaluation for BEVDet [9] and our corresponding RC-BEVFusion algorithm with BEVFeatureNet. Since not all classes are represented in the rain and night scenes, the averaged metrics across all classes are less meaningful. We therefore again present class-wise results of the three most common dynamic classes: car, pedestrian, and truck. The results in Table 7 show that compared to the overall AP increase across all scenes listed in Table 6, there were higher improvements for rain and especially for night scenes. We conclude that the camera-only model struggles in these adverse conditions, particularly in detecting pedestrians and trucks at night. This is where the proposed radar-camera fusion can add the most value. Note that the true positive metrics for pedestrians and trucks at night should be treated with caution due to the low number of matches for the camera-only model. As discussed above, we again observe significant decreases in translation, velocity, and attribute errors, while the scale error remains relatively unchanged. This time, there is also a considerable improvement in orientation error. This finding suggests that the camera-only model struggles to accurately predict orientation in these adverse conditions, and further emphasizes the potential of radar-camera fusion in such scenarios.

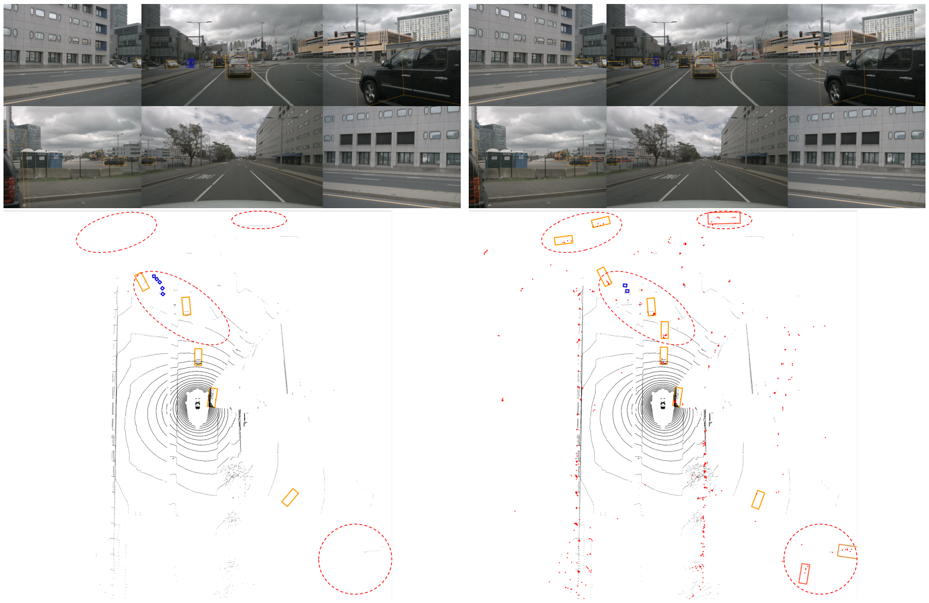
A.4 Additional qualitative evaluation

We provide additional selected qualitative examples for the camera-only baseline BEVDet [9] in comparison with our proposed radar-camera fusion model with

Table 7: Experimental class-wise results for our radar-camera fusion based on BEVDet on the three most common dynamic classes of the nuScenes val split, filtered by rain and night scenes, respectively. *We use the implementation of BEVDet-Tiny with a heavier view transformer from [21].

Split		Cam. model	Radar model	Class		AP↑	ATE↓	ASE↓	AOE↓	AVE↓	AAE↓
Rain	[9]	BEVDet*	None	car		0.517	0.548	0.158	0.133	0.693	0.151
				ped.		0.218	0.748	0.360	1.679	1.043	0.725
				truck		0.276	0.751	0.216	0.153	0.479	0.120
	Ours	BEVDet*	BEVFeatureNet	car		0.723	0.304	0.160	0.121	0.277	0.141
					Δ_r	40%	-45%	1%	-9%	-60%	-7%
				ped.		0.338	0.516	0.360	1.011	0.849	0.332
					Δ_r	55%	-31%	0%	-40%	-19%	-54%
				truck		0.449	0.539	0.200	0.120	0.235	0.108
					Δ_r	63%	-28%	-7%	-22%	-51%	-10%
Night	[9]	BEVDet*	None	car		0.403	0.527	0.137	0.111	1.619	0.485
				ped.		0.045	0.664	0.296	1.509	0.675	0.469
				truck		0.057	0.630	0.221	0.151	2.795	0.582
	Ours	BEVDet*	BEVFeatureNet	car		0.611	0.310	0.137	0.092	0.538	0.469
					Δ_r	52%	-41%	0%	-17%	-67%	-3%
				ped.		0.191	0.262	0.283	0.810	0.592	0.037
					Δ_r	324%	-61%	-4%	-46%	-12%	-92%
				truck		0.265	0.304	0.181	0.127	0.616	0.697
					Δ_r	365%	-52%	-18%	-16%	-78%	20%

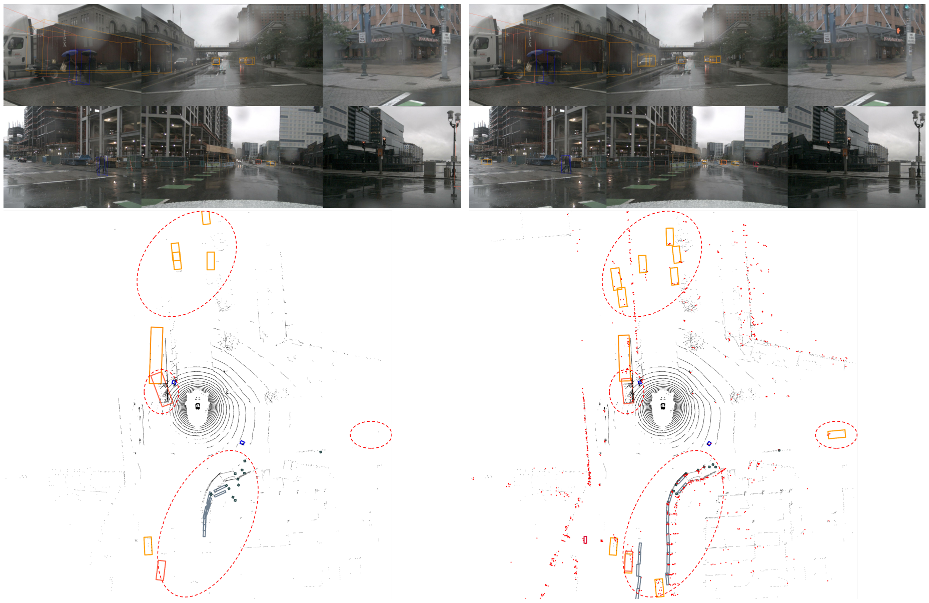
BEVFeatureNet and BEVDet. Figure 5 shows another example at daytime. Our fusion network achieves better performance for long ranges as can be seen with the distant cars in the front and back right area. It also detects an occluded car two vehicles ahead of the ego-vehicle. Figure 6 shows an example during rain. As indicated by the quantitative evaluation in the previous section, the network shows a much better overall scene understanding with the barriers in the back area as well as the vehicles in the front area. In addition, the orientation estimation for the truck on the left is improved and an additional vehicle is detected on the right. However, this frame also shows some failure cases that still exist. The two pedestrians on the left are only detected as one by both networks, which are possibly confused by the umbrellas. Also, the fusion network detects an additional parked car in the front right due to matching radar detections, which is a false positive. Figure 7 shows another example at night. The fusion network again shows better performance for the distant vehicles in the front and back. In addition, it shows better orientation estimation for the motorcycle and truck on the right and does not detect a false positive pedestrian on the left.



(a) BEVDet

(b) Proposed RC-BEVFusion

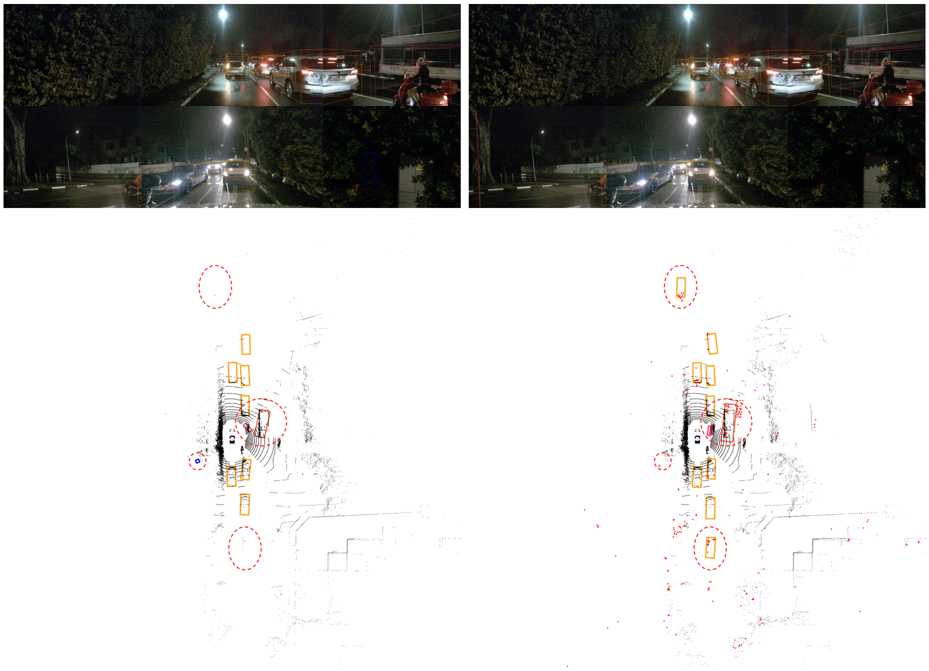
Fig. 5: Inference example at daytime. Our network more accurately detects distant cars in the front and back right area as well as an occluded car directly in front.



(a) BEVDet

(b) Proposed RC-BEVFusion

Fig. 6: Inference example during rain. Our network has much better overall scene understanding with the road barriers in the back and the cars in front and on the right.



(a) BEVDet

(b) Proposed RC-BEVFusion

Fig. 7: Inference example at night. Our network more accurately detects distant cars in the front and back, has better orientation estimation for the motorcycle and truck on the right and no false positive pedestrian on the left.