V2U4Real: A Real-world Large-scale Dataset for Vehicle-to-UAV Cooperative Perception

Supplementary Material

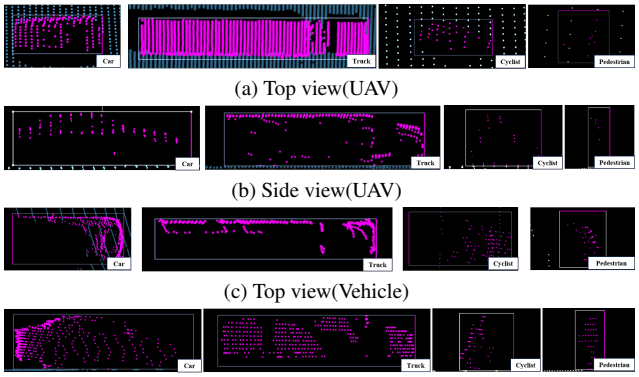
A. Appendix

This supplementary document is organized as follows:

- LiDAR appearance of different object classes are provided in Sec. B.
- More details about V2U4Real are provided in Sec. C.
- Implementation details of the evaluated models are covered in Sec. D.
- More experimental results are discussed in Sec. E.
- Limitations and future work are discussed in Sec. F.

B. Dataset Visualization

We present the LiDAR appearance of each object class from the UAV and the ego vehicle in Fig. 7.



(a) Top view(UAV)

(b) Side view(UAV)

(c) Top view(Vehicle)

(d) Side view(Vehicle)

Figure 7. **LiDAR appearance of different object classes in V2U4Real.** The first and second rows show the top and side views captured by the UAV-mounted LiDAR, while the third and fourth rows present the corresponding views from the vehicle-mounted LiDAR.

C. More Details about V2U4Real

C.1. Explanation of "Vehicle-to-UAV"

The term "Vehicle-to-UAV" in our paper denotes communication connectivity rather than the direction of data flow. In our experiments, all data are transmitted from the UAV to the ego vehicle for fusion.

C.2. Sensor Cost Reference

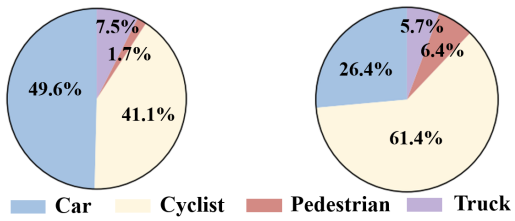
We include an approximate sensor cost below for reference.

Table 6. **Sensor Cost of V2U4Real.**

Sensor	Model	Cost ($)
LiDAR	OS-128 / RS-128 / M1-Plus	12K / 9K / 3K
Camera	HikRobot MV-CA023-10GC	700
GPS/IMU	Quectel / Sigma100 / DETA100	35 / 1K / 200

C.3. Time Synchronization

For all cooperative perception datasets, achieving temporal synchronization between sensors is crucial. To ensure a unified temporal reference, all onboard computing units are first synchronised to the GPS time standard. Subsequently, hardware-level synchronisation of LiDAR and camera sensors is accomplished through the combined use of the Precision Time Protocol (PTP) and Pulse Per Second (PPS) signals. Afterward, the temporally closest LiDAR frames from the ego vehicle and UAV are paired, and the corresponding camera data are temporally aligned with each LiDAR frame to construct coherent multi-modal data samples. The inter-sensor temporal discrepancy across the two agents is maintained within 20 milliseconds for each recorded sample.



(a) UAV-side (b) Vehicle-side

Figure 8. **Category distributions of V2U4Real.**

C.4. Dataset Statistics

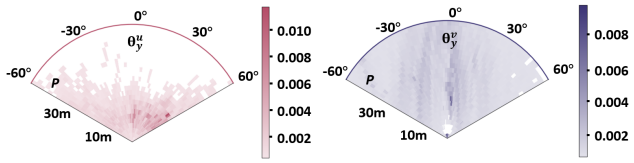
To enable comprehensive evaluation and fair comparison across various cooperative perception tasks, V2U4Real dataset is split into train/val/test sets at the sequence level. This design prevents frame-level overlap and ensures that models evaluated on the validation and test sets are not exposed to temporally adjacent frames from the training set. The samples are evenly distributed across the three splits to maintain a balanced representation of scene types, object densities, and motion patterns. Tab. 7 summarizes the frame-level statistics for each split, including the number of data frames and total 3D annotations. The test split is deliberately made more challenging, with greater scene diversity and denser traffic interactions, allowing for a robust assessment of model generalization in real-world cooperative perception scenarios.

Table 7. **Dataset statistics of V2U4Real.**

Split	Sequence	Scene	Frames	Vehicles	Cyclists	Pedestrians	Track IDs	Duration (s)
Train	2025-07-17-16-12-1	Urban	1200	7140	1036	0	112	30.0
	2025-07-17-16-12-2	Urban	1244	9255	1386	0	138	31.1
	2025-07-17-16-12-3	Urban	1116	13865	1116	0	145	27.9
	2025-07-17-16-12-4	Urban	1208	8890	1551	0	129	30.2
	2025-07-17-16-35-3	Urban	3112	10175	18342	2730	241	77.8
	2025-07-17-16-50-1	Urban	1528	7247	14733	1953	207	38.2
	2025-07-17-16-50-2	Urban	1532	5748	25348	1299	324	38.3
	2025-07-17-16-50-3	Urban	1748	4526	15532	616	171	43.7
	2025-07-17-17-07-2	Campus	1744	1986	8592	0	61	43.6
	2025-07-17-17-07-6	Campus	1588	13805	19806	1291	242	39.7
	2025-07-17-17-42-1	Campus	1264	8743	12596	3695	154	31.6
	2025-07-17-17-42-2	Campus	1244	1566	6077	1728	88	31.1
	2025-07-17-17-42-3	Campus	884	2457	3405	1930	83	22.1
	2025-07-17-17-42-4	Campus	1120	4673	8144	2791	127	28.0
	2025-07-17-17-42-5	Campus	732	1097	12328	1275	153	18.3
	2025-07-18-12-10-1	Campus	1224	1387	23034	1670	266	30.6
	2025-07-18-12-10-2	Campus	1284	1154	16701	3340	226	32.1
	2025-07-18-12-10-3	Campus	1768	1228	29856	655	360	44.2
	2025-07-18-12-37-1	Rural	1240	8141	2379	311	74	31
	2025-07-18-12-37-2	Rural	1936	15650	6899	1395	185	48.4
	2025-07-18-12-37-3	Rural	1400	6209	1800	0	76	35
	2025-07-18-12-53-1	Rural	1212	2162	1622	130	26	30.3
	2025-07-18-12-53-2	Rural	876	2021	875	0	21	21.9
	2025-07-18-12-53-3	Rural	1340	4735	1027	0	42	33.5
Val	2025-07-17-16-12-7	Urban	364	2460	364	0	65	9.1
	2025-07-17-16-12-8	Urban	544	3686	375	0	56	13.6
	2025-07-17-16-35-2	Urban	784	932	19775	116	309	19.6
	2025-07-17-16-50-6	Urban	2128	3880	27245	3398	342	53.2
	2025-07-17-17-07-1	Campus	1524	7355	10731	1343	134	38.1
	2025-07-17-17-42-7	Campus	364	552	2738	121	67	9.1
	2025-07-18-12-10-4	Campus	1280	1843	15853	663	228	32.0
	2025-07-18-12-37-5	Rural	1400	3160	1682	0	51	35.0
	2025-07-18-12-53-4	Rural	1508	8417	2471	8	185	37.7
Test	2025-07-17-16-12-6	Urban	1036	7453	1016	0	135	25.9
	2025-07-17-16-35-1	Urban	1096	8431	6063	410	166	27.4
	2025-07-17-16-50-4	Urban	1460	1094	26474	149	303	36.5
	2025-07-17-16-50-5	Urban	1156	867	16620	772	327	28.9
	2025-07-17-17-07-3	Campus	1108	1108	21746	1110	240	27.7
	2025-07-17-17-42-6	Campus	700	802	12820	0	140	17.5
	2025-07-17-17-42-8	Campus	476	1273	4859	826	96	11.9
	2025-07-18-12-10-5	Rural	1896	15997	16860	4962	413	47.4
	2025-07-18-12-53-5	Rural	1616	12559	1654	533	126	40.4

C.5. Annotation Process and Labeler Details

We use a two-round annotation pipeline. (1) **44** professional annotators are responsible for specific scenarios. (2) **11** quality-control annotators refine annotations across four scenarios.



(a) UAV-side (b) Vehicle-side

Figure 9. **Target bounding box distributions of V2U4Real.**

C.6. Category Distributions

Fig. 8 shows the distribution of object types in V2U4Real across different agents. On the UAV side, Car constitutes the majority at 49.6%, followed by Cyclist at 41.1% and Trucks at 7.5%, while Pedestrian makes up only 1.7%. On the vehicle side, Cyclist account for the largest proportion at 61.4%, followed by Car at 26.4%, Trucks at 5.7%, and Pedestrian at 6.4%.

C.7. Target Bounding Box Distributions

Variations in sensing altitude and observation geometry lead to markedly different BEV distribution patterns across agents. As shown in Fig. 9, UAV-side and vehicle-side annotations exhibit distinct spatial and angular characteris-

Table 8. **Single-agent 3D object detection benchmarks on V2U4Real val set.** The vehicle category includes car and truck.

LiDAR Type	Method	Vehicle (3D AP@IoU)		Cyclist (3D AP@IoU)		Pedestrian (3D AP@IoU)	
		0.5	0.7	0.25	0.5	0.25	0.5
Ruby-128	PointPillars [16]	60.57	29.57	59.62	45.30	38.08	27.65
	SECOND [35]	56.61	32.01	52.73	42.29	48.15	36.97
	CenterPoint [36]	58.24	26.93	56.39	42.56	**58.79**	**43.64**
	PV-RCNN [24]	**65.91**	**41.99**	**67.11**	**53.35**	45.06	37.92
M1-Plus	PointPillars [16]	41.37	20.81	53.28	41.40	15.06	9.09
	SECOND [35]	34.73	18.04	54.67	42.09	24.20	16.03
	CenterPoint [36]	**41.84**	19.75	**60.62**	42.54	**24.81**	**16.35**
	PV-RCNN [24]	41.38	**22.27**	59.12	**45.89**	12.57	7.58

Table 9. **Cooperative 3D object detection benchmarks for vehicle category on V2U4Real test set.** Sync. means synchronous setup ignoring communication delays. Async. implies asynchronous setup with a $(0, 1000]$ ms delay.

Paradigm	Method	Year	Sync. (3D AP@IoU = 0.5 / 0.7)			Async. (3D AP@IoU = 0.5 / 0.7)			AM
			Overall	0–50m	50–100m	Overall	0–50m	50–100m	(MB)
Vehicle to UAV	Early Fusion	-	**45.44/20.76**	**49.59/22.38**	**20.56/13.16**	24.63/9.63	26.29/9.71	16.53/11.01	3.18
	Late Fusion	-	36.76/13.20	39.89/14.06	11.53/8.29	23.48/8.14	24.62/8.38	10.22/7.37	0.009
	Where2comm [10]	2022	33.14/12.14	36.31/12.94	16.69/12.45	30.68/10.81	33.50/11.26	16.58/12.44	0.65
	V2X-ViT [32]	2022	29.69/11.65	33.24/12.63	10.92/7.93	29.72/11.62	33.23/12.76	10.87/7.80	0.65
	AttFuse [33]	2022	38.75/17.22	43.17/18.95	17.57/12.23	29.76/11.65	33.00/11.87	17.06/11.87	0.65
	CoBEVT [31]	2022	31.52/14.45	36.26/16.59	13.85/8.72	25.27/9.77	28.50/10.53	13.41/8.41	0.65
	CoAlign [20]	2023	41.78/18.79	46.06/20.17	17.75/13.42	**36.41/15.58**	**39.88/16.98**	**17.15/13.04**	0.65
	ERMVP [39]	2024	36.48/14.28	40.97/15.76	13.22/8.21	24.46/8.07	27.29/8.44	10.87/7.51	0.65
	DSRC [41]	2025	41.88/17.27	45.73/18.48	18.32/12.88	35.38/13.38	38.27/13.91	16.75/12.63	0.65
Vehicle only	No Fusion	-	28.25/10.05	31.13/12.05	10.11/6.01	28.25/10.05	31.13/12.05	10.11/6.01	0
UAV only	No Fusion	-	31.47/11.06	33.21/12.42	12.52/8.13	31.47/11.06	33.21/12.42	12.52/8.13	0

tics. On the UAV side, targets occupy a broad and relatively sparse BEV region, with higher counts around the central downward viewing direction. This arises from the UAV's elevated bird's-eye viewpoint: ground objects are projected across a wide radial range, resulting in larger dispersion and higher variance in BEV distances. Moreover, the extended visibility from aerial perspectives leads to targets appearing within a wide horizontal angular span $\theta_y^u \in [-60°, 60°]$, further contributing to the irregular and non-uniform distribution. In contrast, the vehicle-side distribution is significantly more compact. Due to the vehicle's low and stable sensing height, detections are concentrated in a narrower, forward-looking region. Targets mainly appear near the frontal direction, with shorter BEV distances and much smaller angular variation $\theta_y^v \in [-10°, 10°]$. The limited elevation also reduces the likelihood of observing distant objects, yielding lower variance and a higher density of detections around the ego origin.

D. Implementation Details

D.1. Detection Model Settings

We provide additional implementation details of the baseline methods used in our experiments to facilitate accurate reproduction of the benchmark results. For the single-agent 3D object detection task, we set the maximum number of points per voxel to 32 and the maximum number of voxels to 160,000. The models are trained for 40 epochs with an initial learning rate of 0.002. Additional optimization and preprocessing configurations are summarised in Tab. 10. For the cooperative 3D object detection task, we similarly set the maximum points per voxel to 32, while the maximum voxel count is reduced to 32,000 due to the multi-agent fusion input. The same training schedule with single-agent object detection task and a voxel size of [0.4, 0.4, 8] is adopted. The remaining settings can be found in Tab. 11.

D.2. Asynchronous Communication Modeling

Following [32, 34], we simulated asynchronous communication by temporally misaligning sensor frames and modeling delay as the sum of system overhead, transmission time, and backbone computation latency.

E. More Experimental Results

E.1. Results on Ruby-128 and M1-Plus LiDAR

Tab. 8 shows the single-agent 3D object detection performance on the V2U4Real val set using two heteroge-

Table 10. **Implementation details of single-agent 3D object detection task.**

Method	Epoch	Batch	LR	LR Scheduler	Voxel Size
PointPillar[16]	40	4	0.002	OneCycle	$[0.2, 0.2, 8]$
SECOND[35]	40	4	0.002	OneCycle	$[0.1, 0.1, 0.2]$
CenterPoint[36]	40	4	0.002	OneCycle	$[0.1, 0.1, 0.2]$
PV-RCNN[24]	40	4	0.002	OneCycle	$[0.1, 0.1, 0.2]$

neous LiDAR sensors: Ruby-128 and M1-Plus. We evaluate four representative baseline detectors, including PointPillars [16], SECOND [35], CenterPoint [36], and PV-RCNN [24]. These results reveal the varying robustness of different 3D detectors under heterogeneous LiDAR sensors.

E.2. Results on the Test Set

Tab. 9 shows the cooperative 3D object detection results for the vehicle category on the V2U4Real test set, evaluated under both synchronous and asynchronous settings. Compared with the validation results reported in the main paper (see Tab. 3), the test split presents significantly greater challenges, including more frequent pose variations, denser traffic, and more complex occlusion patterns. Under the synchronous setting, Early Fusion achieves the best performance, indicating that existing intermediate fusion methods still have substantial room for improvement when dealing with more complex real-world scenarios. However, under the asynchronous setting, Early Fusion fails to effectively handle communication delays (consistent with the observations in Tab. 3), leading to a significant performance drop, while CoAlign [20] achieves the best performance.

E.3. Visualization of Cooperative 3D Detection

We present additional qualitative results of cooperative 3D detection comparisons in Fig. 10, 11, and 12, all evaluated under the synchronous setting. In these scenes, Intermediate Fusion methods demonstrate significantly better detection performance compared to Late Fusion and Early Fusion. CoAlign [20] achieves the closest alignment between predicted and ground-truth bounding boxes among all compared methods. These observations are consistent with the numerical results reported in Tab. 3 and reflect the internal mechanisms of each algorithm. We further analyze typical failure cases across methods. Specifically, we observe misalignment between **predicted** and **ground-truth** bounding boxes, which is mainly caused by localization noise arising from different agent motion patterns (Fig. 2 in the main paper). In addition, missed detections are frequently observed under viewpoint inconsistency, where heterogeneous sensing perspectives lead to discrepancies in cross-agent point cloud distributions (Fig. 6(b) in the main paper).

Table 11. **Implementation details of cooperative 3D object detection task.** CAW stands for CosineAnnealWarm.

Method	Batch	LR Scheduler	LiDAR Range
No Fusion	4	OneCycle	$[-100, -80, 100, 80]$
Early Fusion	4	MultiStep	$[-100, -80, 100, 80]$
Late Fusion	4	MultiStep	$[-100, -80, 100, 80]$
Where2comm[10]	4	CAW	$[-100, -80, 100, 80]$
V2X-VIT[32]	2	CAW	$[-89.6, -76.8, 89.6, 76.8]$
AttFuse[33]	4	MultiStep	$[-100, -80, 100, 80]$
CoBEVT[31]	2	CAW	$[-102.4, -80, 102.4, 80]$
CoAlign[20]	4	MultiStep	$[-100, -80, 100, 80]$
ERMVP[40]	2	CAW	$[-89.6, -76.8, 89.6, 76.8]$
DSRC[41]	4	CAW	$[-100, -80, 100, 80]$

E.4. Quantitative Analysis of Localization Noise

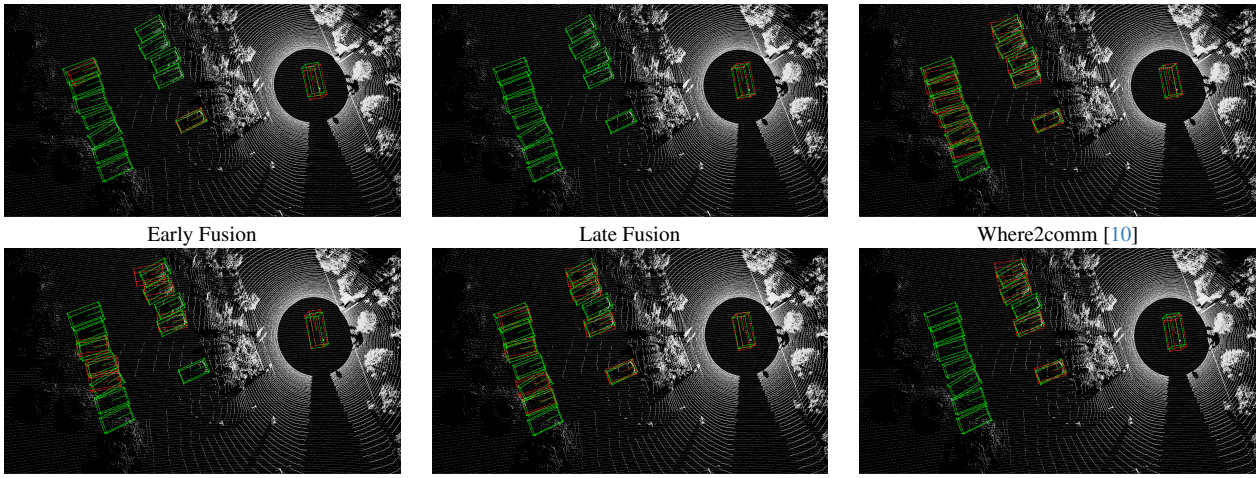
In cooperative perception, the accuracy of relative poses between agents is more critical than absolute pose estimation. Therefore, our evaluation focuses on the consistency of relative transformations rather than absolute pose errors. We leverage the annotated 3D bounding boxes of the ego vehicle and the UAV, which are defined within a shared global coordinate system. Based on these annotations, we derive the deviations of the same target across different agents in each dimension, which indirectly reflect the relative localization errors of GPS. We report the mean (μ) and standard deviation (σ) of the errors in X, Y, Z, Yaw, and IoU. As shown in Table 12, our dataset achieves comparable or lower noise levels than V2V4Real [34] in most metrics.

Table 12. **Quantitative results of localization noise.**

Dataset	Localization Noise (μ / σ)				
	X (m)	Y (m)	Z (m)	Yaw (°)	IoU (%)
V2V4Real	7.0/48.4	0.6/20.3	0.1/1.4	0.05/1.5	2.4/11.0
Ours	2.4/6.8	1.6/2.9	1.0/1.1	0.1/1.5	4.6/11.6

F. Limitations and Future Work

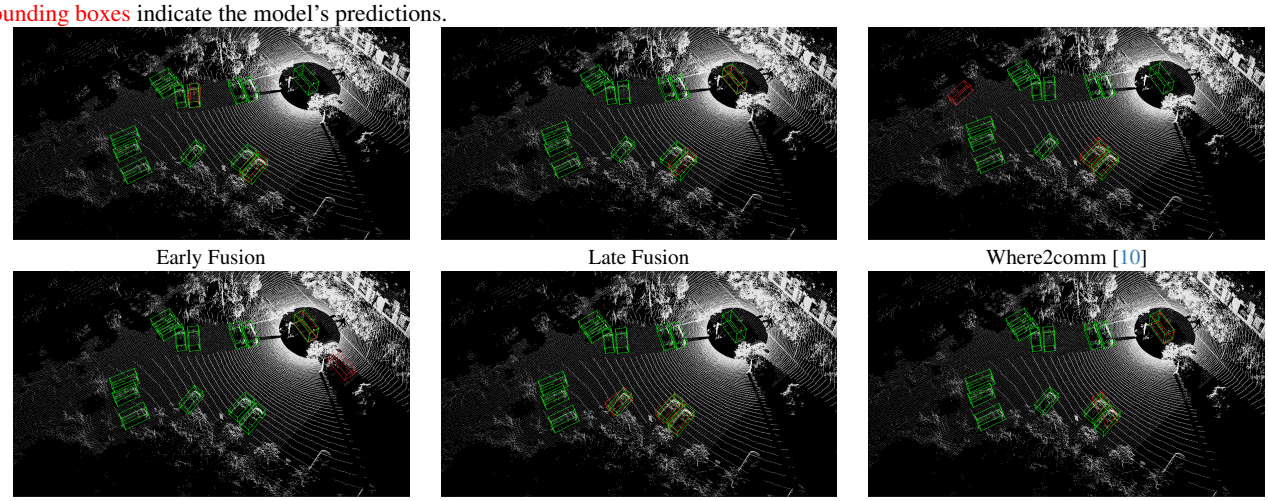
While our experiments on the V2U4Real dataset demonstrate the promise of 3D V2U cooperative perception, several important limitations remain. The current approach primarily focuses on LiDAR-based V2U cooperation, leaving multi-modal sensing underexplored, and multi-view cooperative strategies have not been fully investigated. The scale and heterogeneity of collaborating agents are also limited, whereas real-world deployments typically involve dynamic networks of diverse vehicles and UAVs with varying sensing and communication capabilities. Moreover, due to airspace management policies and safety regulations, UAV data collection is restricted to designated regions, which may limit the diversity and coverage of real-world scenarios. In future work, we will further expand the dataset to more challenging scenarios, including off-road environments, nighttime conditions, adverse weather, varying flight altitudes, and broader operational ranges.



Early Fusion | Late Fusion | Where2comm [10]

AttFuse [33] | CoAlign [20] | DSRC [41]

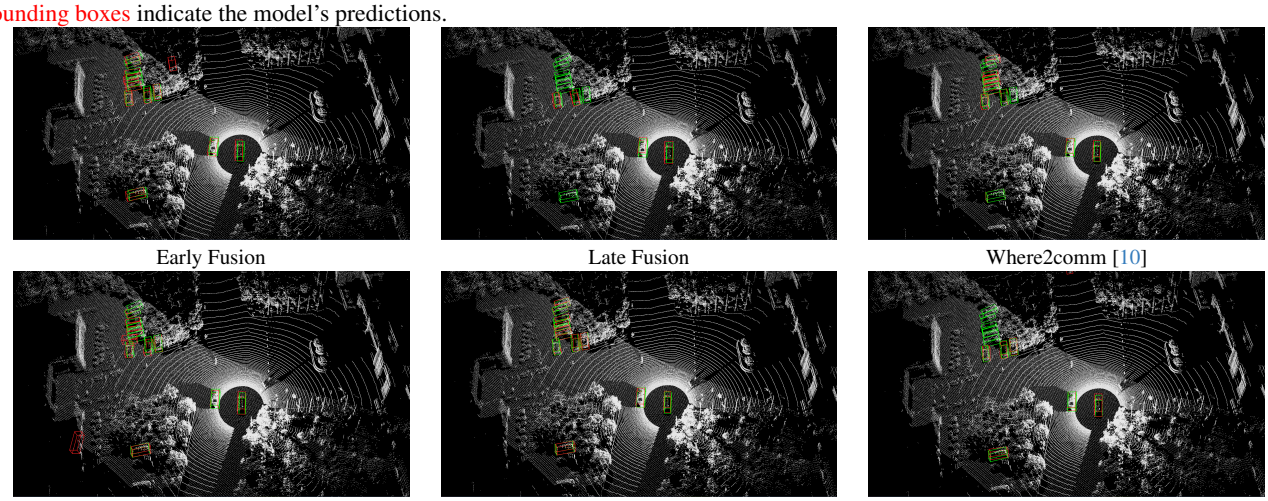
Figure 10. **Visualization of 3D cooperative detection results in scene 1.** Green bounding boxes indicate ground-truth annotations. Red bounding boxes indicate the model's predictions.



Early Fusion | Late Fusion | Where2comm [10]

AttFuse [33] | CoAlign [20] | DSRC [41]

Figure 11. **Visualization of 3D cooperative detection results scene 2.** Green bounding boxes indicate ground-truth annotations. Red bounding boxes indicate the model's predictions.



Early Fusion | Late Fusion | Where2comm [10]

AttFuse [33] | CoAlign [20] | DSRC [41]

Figure 12. **Visualization of 3D cooperative detection results scene 3.** Green bounding boxes indicate ground-truth annotations. Red bounding boxes indicate the model's predictions.