RadarMOSEVE: A Spatial-Temporal Transformer Network for Radar-Only Moving Object Segmentation and Ego-Velocity Estimation Supplementary Material

A Supplementary Overview

To better comprehend our proposed Radar-MOSEVE network, more details will be provided in this supplementary material. In Sec. B, we describe the details of our proposed Radar-MOSEVE network, including the proposed radar spatial-temporal transformer backbone, the architecture of the MOS module, and the EVE module. The details of our self-collected dataset and the public dataset, View-of-Delft (VoD) (Palffy et al. 2022), are presented in Sec. C. We comprehensively introduce each sensor setup of our dataset and show more data examples. Additionally, we show our re-annotation of the VoD dataset. In Sec. D, the evaluation metrics are further explained for MOS and EVE. In Sec. E, we further demonstrate more qualitative results in both datasets to understand our methods' advantages and limitations. And extra ablation studies are also provided to validate the necessity of our proposed module, parameter choice, and loss function.

B Network Architecture Details

Our EVE module is a feature extraction network with the proposed radar transformer-based encoder backbone. Our MOS module is an encoder-decoder network based on our proposed radar transformer network. Several MLPs are used to connect them and generate the final output results.

Encoder backbone for MOS and EVE. Our proposed radar transformer is used to construct the encoder backbone for our MOS and EVE modules. The backbone details are present in Tab. 1. It includes the radar self-attention module and the radar cross-attention module. The input for the backbone is a feature vector $\boldsymbol{F}$ of size $4 \times 512 \times 32$, generated by an MLP using the original point cloud $\boldsymbol{P}$ of size $4 \times 512 \times 4$, which will be used as the input to MOS and EVE modules. For object attention, we sample 16 points within a radius of 2 m around each point. For each layer of the radar self-attention module, The input features are transformed into three embedding $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$ by different MLPs. The final dimensions of $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$ for each point are 512. The radar cross-attention module takes the output of radar self-attention modules from two frames to obtain the spatial and temporal features $\boldsymbol{F}'$. For radar cross attention, $\boldsymbol{Q}$ is a feature vector of 512 dimensional generated by an MLP using the current frame feature, $\boldsymbol{K}$ and $\boldsymbol{V}$ are generated by two MLPs using the previous frame feature, which have the same shape as $\boldsymbol{Q}$. They are all generated by MLPs. The output of the radar cross-attention module is a feature vector $\boldsymbol{F}'$ of size $4 \times 32 \times 128$.

Operator	Sample dis.	Sample inter.	Output
MLP	-	-	$4 \times 512 \times 32$
Obj. Att.	2	-	$4 \times 512 \times 32$
Downsample	-	-	$4 \times 128 \times 64$
Obj. Att.	2	-	$4 \times 128 \times 64$
Downsample	-	-	$4 \times 32 \times 128$
Sce. Att.	-	2	$4 \times 32 \times 128$
Cro. Att.	5	-	$4 \times 32 \times 128$

Obj. Att.: object attention module; Sce. Att.: scenario attention module; Cro. Att.: radar cross attention module. Sample dis: the distance between sampling points and source points; Sample inter: the sampling interval in the scenario attention module;

Table 1: Attention-based Backbone

Operator	Sample dis	Sample inter	Output
Sce. Att.	-	2	$4 \times 32 \times 128$
Upsample	-	-	$4 \times 128 \times 64$
Obj. Att.	2	-	$4 \times 128 \times 64$
Upsample	-	-	$4 \times 512 \times 32$
Obj. Att.	2	-	$4 \times 512 \times 32$
MLP.	-	-	$4 \times 512 \times 2$

Obj. Att.: object attention module; Sce. Att.: scenario attention module; Cro. Att.: radar cross attention module. Sample dis: the distance between sampling points and source points; Sample inter: the sampling interval in the scenario attention module;

Table 2: Attention-based decoder of MOS module

Decoder of MOS module. The decoder architecture for MOS is presented in Tab. 2. It takes the feature $\boldsymbol{F}'$ and outputs the MOS feature $\boldsymbol{F}''$ of size $4 \times 512 \times 32$, and it will be transformed to the moving semantic $\boldsymbol{S}$ of size $4 \times 512 \times 2$ by an MLP structure.

C Dataset

Due to the lack of MOSEVE data, we collected 13,654 frames of radar point clouds for evaluating MOSEVE performance and other synchronized sensor data to label the

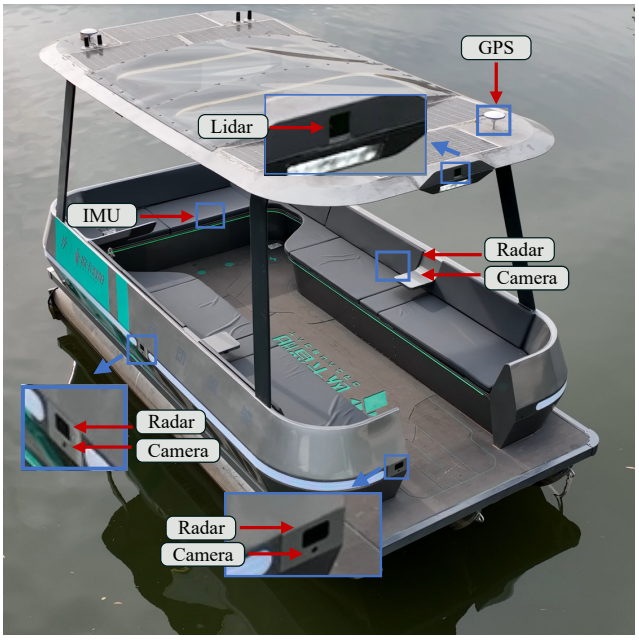

Figure 1: Data acquisition platform in water scenarios. (Some sensors are only marked with their location because the ship's hull obstructs them.)

Performance	Value
Range resolution	0.127m
Maximum unambiguous range	52m
Velocity resolution	0.188m/s
Maximum unambiguous velocity	3.011m/s
Azimuth angular resolution	14.3°
Elevation angular resolution	57.2°

Table 3: The performance parameters of the mmWave radar point clouds for the water platform.

Performance	Value
Range resolution	0.045m
Maximum unambiguous range	18.89m
Velocity resolution	0.13m/s
Maximum unambiguous velocity	2.03m/s
Azimuth angular resolution	14.3°
Elevation angular resolution	57.2°

Table 4: The performance parameters of the mmWave radar point clouds for the road platform.

radar points for MOS. This section introduces the details of our dataset, including the acquisition platform at Sec. C.1 and dataset details at Sec. C.2. In addition, we test the performance in a public dataset, Viewd-of-Delft (VoD), whose details are provided at Sec. C.3.

C.1 Dataset Acquisition Platform

We equipped our two platforms with radar, LiDAR, camera, IMU, and RTK-GPS for the data acquisition. The sensors are annotated for the water and road platforms in Fig. 1 and Fig. 2. Because the sampling frequency of radar is 10 HZ, we synchronize each sensor to a sampled frequency of 10 Hz. Adjacent frames replace data missing from sensors with a frequency lower than 10 Hz. We provide more details about each sensor as follows.

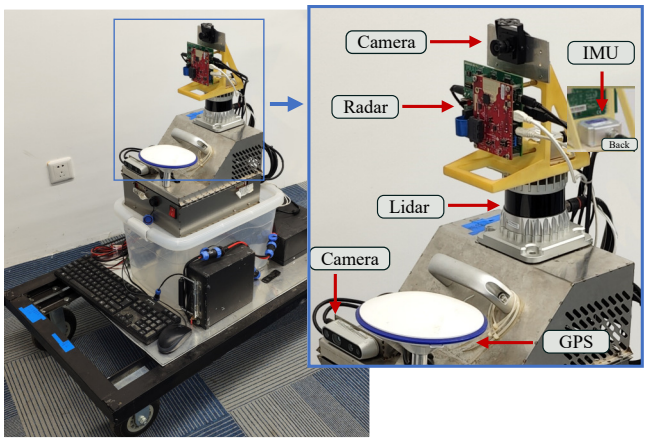

Figure 2: Data acquisition platform in road scenarios.

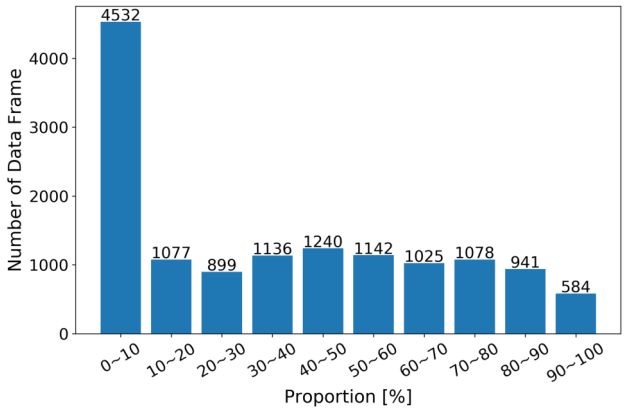

Figure 3: Moving point cloud distribution of our dataset

Radar We choose the radars from Texas Instruments (TI) because of their low price and good performance. We use three radars for our water platform and one radar for our road platform. Each radar contains three transmitting antennas and four receiving antennas. By designing the radar waveform, we obtain the radar point clouds. The performance parameters of radar point clouds on the water platform are listed in Tab. 3, and the parameters on the road platform are listed in Tab. 4.

LiDAR We use a Livox Mid-70 LiDAR on the platform of water scenarios to acquire the point clouds to complete the MOS label generation. This LiDAR sensor scans a conical area with a length of 100 m and an angle of 35 degrees. For road scenarios, an Ouster-os1 32-beams LiDAR is used to obtain a circular perception area with a radius of 90 m and a vertical angle of 45 degrees.

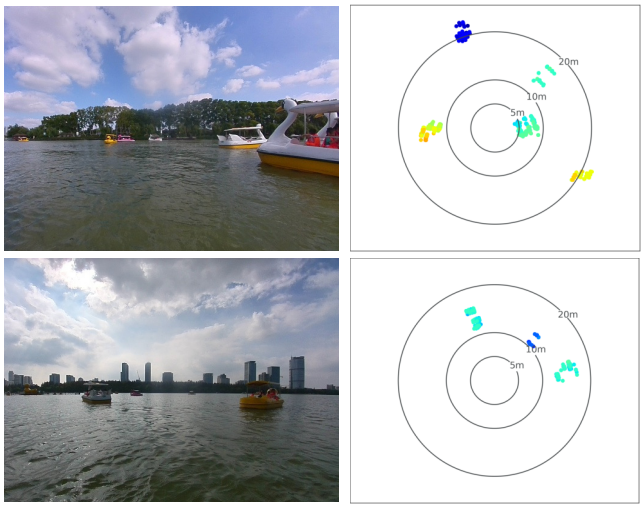

Figure 4: Scenarios with lots of moving objects but a few static objects.

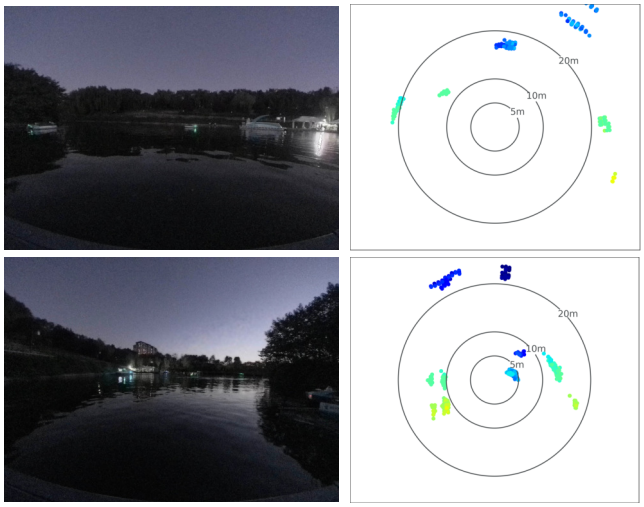

Figure 5: Some scenarios at night.

Cameras The image data is used for visualization and verifying the MOS labels. For the platform of water scenarios, We assemble a 120-degree camera on the front of the platform and two 187-degree cameras on the sides of the platform to acquire the image data around the platform. An Intel Realsense D435i camera is mounted on the road platform with a field of view of 70 degrees.

Other sensors We also equip our platforms with an inertial navigation system (INS), which includes an LPMS-IG1-CAN IMU and a U-Blox ZED-F9 GPS. They are utilized to compensate for the coordinates of point clouds to promote the performance of MOS label generating. They are also used to generate the ground truth of the EVE task.

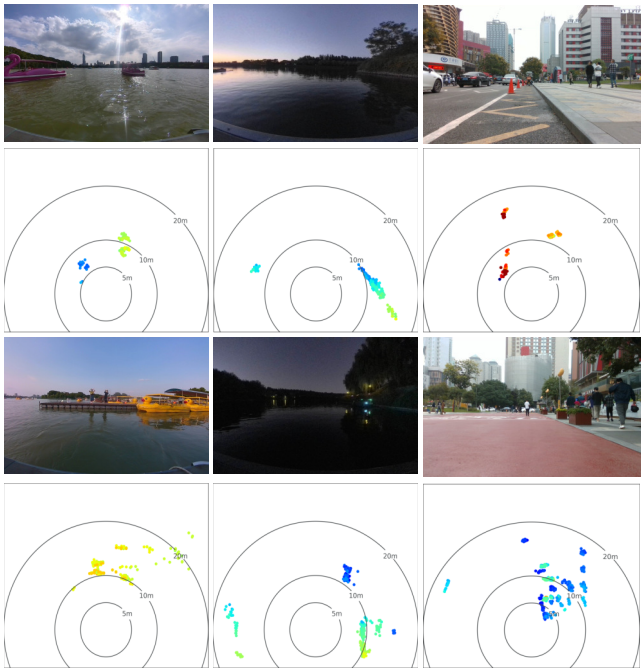

Figure 6: More examples of our dataset.

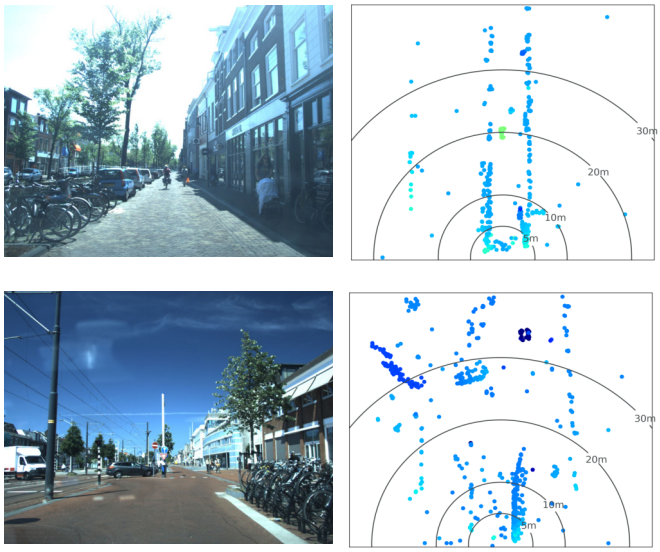

Figure 7: Some examples of VoD dataset.

C.2 Dataset details

Data Annotation In certain scenarios, we can obtain the speed and acceleration of surrounding vehicles to label the corresponding points based on their motion characteristics. In non-collaborative scenarios, we first perform moving object segmentation using image or LiDAR data and then label the corresponding radar point clouds. To ensure accuracy, we manually verify and correct the labels as necessary.

Data Distribution We employ our platform equipped with a single-chip MWR sensor to capture radar data in both wa-

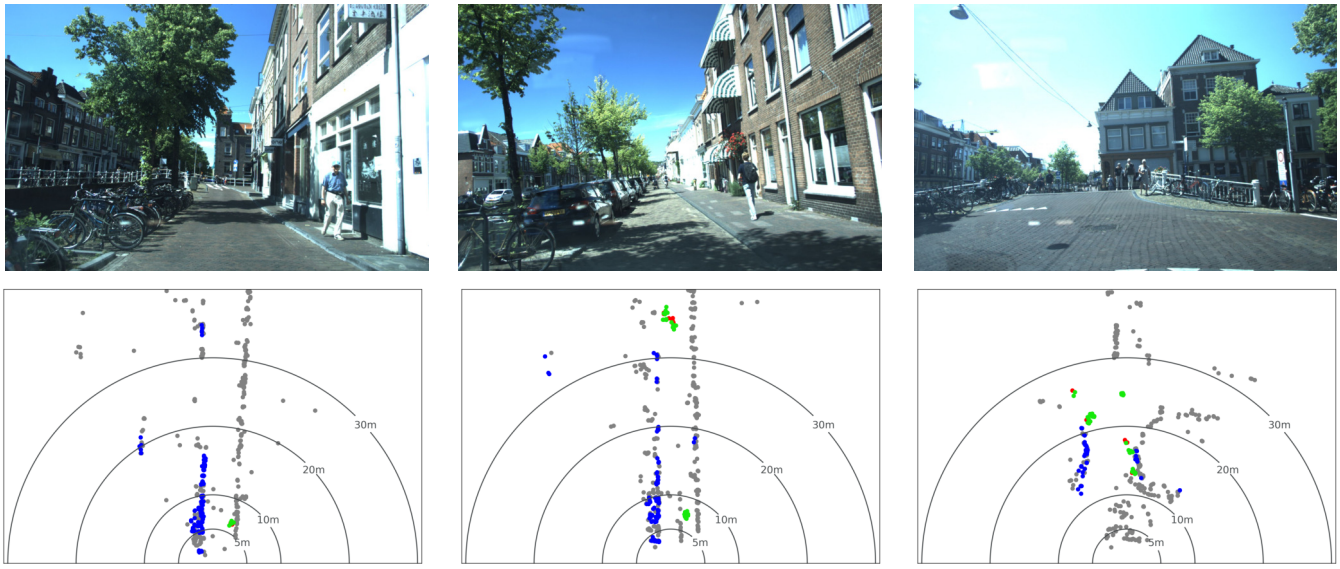

Figure 8: Some wrong annotation of VoD dataset. The red points and gray points, respectively, represent the moving points and static points. the green points and blue points are the wrong moving annotations and static annotations.

ter and road scenes. The water scene data was collected during the day and at night on a park lake, where most moving objects were boats. On the other hand, we recorded the road scene in a crowded walking street with numerous pedestrians.

We collected and annotated a total of 13,654 radar point cloud data, and we calculated the proportion of moving points in each point cloud after annotating them. The resulting statistics are presented in Fig. 3, indicating a significant presence of moving objects in our data scenes, which poses a considerable challenge for MOS and EVE. Finally, we split the data into training, validation, and testing sets in a ratio of 4:1:5. More dataset examples are provided in the following subsection.

More images and point clouds examples of our dataset Our dataset has some challenging scenarios for MOS and EVE, such as a scenario with many moving objects and few static objects. The images and point clouds of these scenarios data are shown in Fig. 4. We also show some data at night in Fig. 5, which is challenging for visual-based MOS showing the benefits of our employed radar sensors. More data examples of water scenarios and road scenarios are shown in Fig. 6.

C.3 Details of VoD dataset

In recent years, several radar datasets have been published for a more robust perception, but only a few for radar MOSEVE. The VoD dataset contains about 8,600 frames of radar points and also other modality data, including LiDAR, camera, and odometry data. To explicitly understand the VoD dataset, some data examples are provided in Fig. 7. There are two major scenarios, roads and streets. The difference between the two scenarios is the velocity of the moving targets. In both scenarios, many static objects are certain to appear, which is different from our dataset.

Existing method (Ding et al. 2023) utilizes the compensated radar doppler velocity to generate the MOS label. However, the label contains many errors, as shown in Fig. 8. We can often observe that some static-labeled points are submerged into the moving object points and vice versa. Therefore, we annotate the radar points again in the same way as labeling our dataset. Finally, we mark about 2,900 frames of radar points in the VoD dataset, which can be divided into 12 sequences. We use six of them, about 1,600 frames, to train the MOSEVE network and one, about 1,00 frames, to validate the model performance. The rest sequences, about 1,200 frames, are used to compare the performance to other methods.

D Evaluation Metrics

D.1 Metrics for MOS

For the MOS task, the mIoU, F1-score, and accuracy are used to evaluate the performance of different methods.

mIoU. Before calculating the mIoU of MOS, we need to calculate the true positive(TP), false positive(FP), true negative(TN), and false negative(FN) of moving and static classes. Then, the mean intersection of union, mIoU, is formulated as follows:

$$\text{mIoU} = \frac{1}{N}\sum_{i=1}^{N}\frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i + \text{FP}_i}, \tag{1}$$

where N is the number of classes. In the MOS task, N equals 2 for moving and static classes.

F1 score. F1 score is the harmonic average of Precision and Recall. Precision describes the ratio of true prediction in all predictions, which can be formulated as Precsion $= \frac{\text{TP}}{\text{TP+FP}}$. Recall describes the ratio of true prediction in all truth,

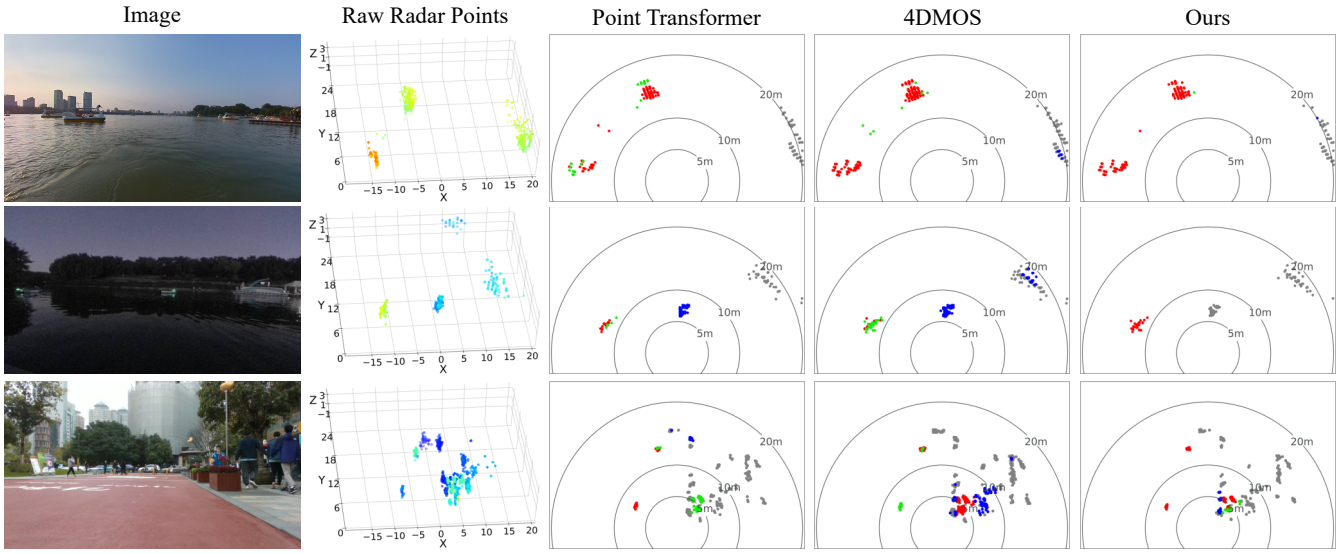

Figure 9: More qualitative results of Point Transformer, 4DMOS, and ours. The red points are the true moving points, the gray points are the true static points, the green points are the false moving points, and the blue points are the false static points.

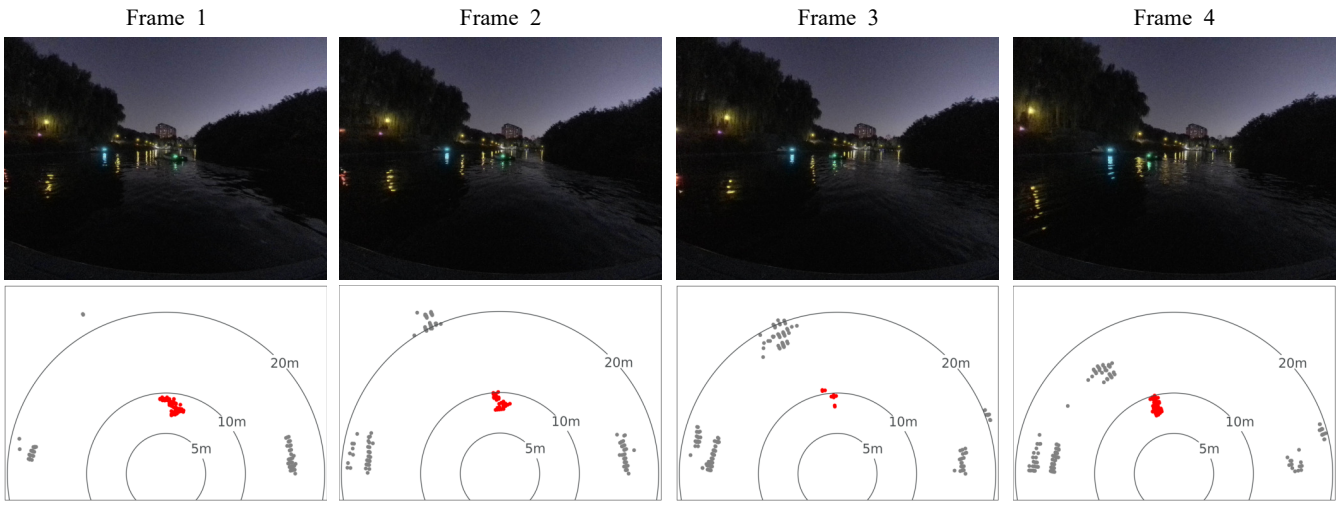

Figure 10: Quantization results of consecutive frames. The frame interval for these data is about 2 seconds.

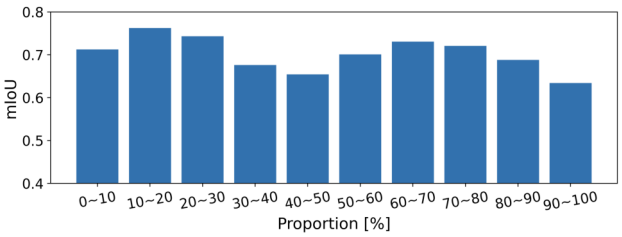

Figure 11: MOS performance with different ratios of moving points

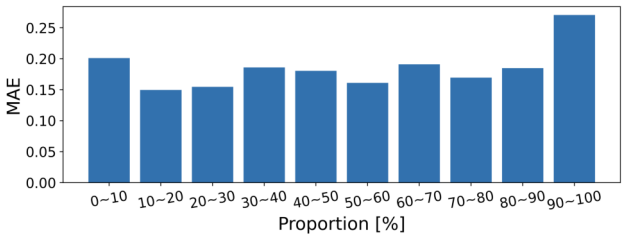

Figure 12: EVE performance with different ratios of moving points

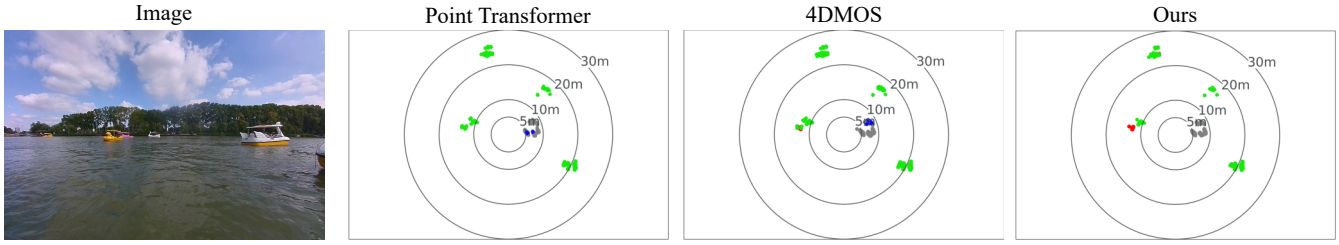

Figure 13: Failure case. The existing methods, including ours, cannot perform well when there are many moving objects but a few static objects.

which can be formulated as Recall $= \frac{\text{TP}}{\text{TP+FN}}$. The F1 score measures recall and precision at the same time. When both of them are higher, the F1 score will be higher. F1 score can be expressed as follows:

$$\text{F1 score} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \tag{2}$$

Accuracy. Accuracy metric refers to the correction ratio of every class prediction. And we also evaluate the average of all class accuracy.

D.2 Metrics for EVE

For the EVE task, we leverage MAE of velocity, MSE of velocity, and the velocity precision at different thresholds to compare different methods.

MAE. MAE of velocity can be formulated as follows:

$$\text{MAE} = \frac{1}{N_p} \sum_{i=1}^{N_p} |v_i - \hat{v}_i|, \tag{3}$$

where N_p denotes the frame number for velocity estimation, v_i denotes the ground truth velocity of acquisition platform, and $\hat{v}_i$ denotes the predictive velocity of the platform. MAE can directly describe the accuracy of velocity estimation.

MSE. We also use the MSE of velocity to measure the stability of the estimated velocity. It can be formulated as follows:

$$\text{MSE} = \frac{1}{N_p} \sum_{i=1}^{N_p} (v_i - \hat{v}_i)^2. \tag{4}$$

Precision. In addition, we found that some of the estimated velocities deviated significantly from the ground truth, but the MAE and MSE of velocity maintained a reasonable range. It is caused by a large number of data. When these velocities were used to compensate for the radial velocity of radar point clouds, it would produce some detrimental features to the MOS task. Hence, we construct the precision metric, which measures the error ratio between the estimated velocities and ground truth velocities at the different thresholds. We can determine which methods can avoid large velocity deviations by comparing the error ratio of different thresholds. For expressing the precision of velocity, we apply $\text{f}(i)$ to describe whether the velocity deviation of frame i is less than the velocity threshold, which can be formulated as follows:

$$\text{f}(i) = \begin{cases} 1 & |v_i - \hat{v}_i| < h \\ 0 & else \end{cases}, i = 1, 2, ..., N_p, \tag{5}$$

where h denotes the threshold of velocity deviation, so the precision of velocity can be expressed as follows:

$$\text{precision}_h = \frac{1}{N_p} \sum_{i=1}^{N_p} \text{f}(i). \tag{6}$$

E More Qualitative Results

E.1 Results on our dataset

More qualitative results in different scenarios are provided to demonstrate the advantage of our methods. We visualized the results of our approach with Point Transformer (Zhao et al. 2021) and 4DMOS (Mersch et al. 2022). The results are shown in Fig. 9. Compared to other methods, our approach performs well in handling object edges, resulting in fewer errors in the local segmentation of the target. Moreover, our approach also performs better in objects' overall moving semantic judgment. In addition, Fig. 10 shows the results of moving object segmentation in continuous data, with a time interval of about 2 seconds for the data. More results are shown in the attached video.

Considering about unbalance distribution of moving points in our dataset, we compare the mIoU for the MOS task and MAE for the EVE task in different ratios of moving points. The MOS and EVE results are shown in Fig. 11 and Fig. 12. The performance drop is not apparent when increasing the ratio of moving points since our method takes individual points as inputs, which are relatively independent of the distribution of moving points in the entire scan.

For a scenario with many moving objects but a few static objects, all of the methods cannot perform well, and their results are shown in Fig. 13.

E.2 Results on VoD dataset

To evaluate the generalization of our methods, we compare our methods to other methods. The quantitative results are present in Tab. 5. The results demonstrate our method achieves state-of-art performance in the VoD dataset. We also provide more qualitative results in the VoD dataset, as shown in Fig. 14. Except for the method compared in our dataset, we also visualized the results of radar-based methods, RaFlow (Ding et al. 2022) and CMFlow (Ding et al. 2023). In all scenarios, our approach demonstrates superior performance in the MOS task. Notably, in Scene 2, our method exhibited no errors. Furthermore, in Scene 3, our

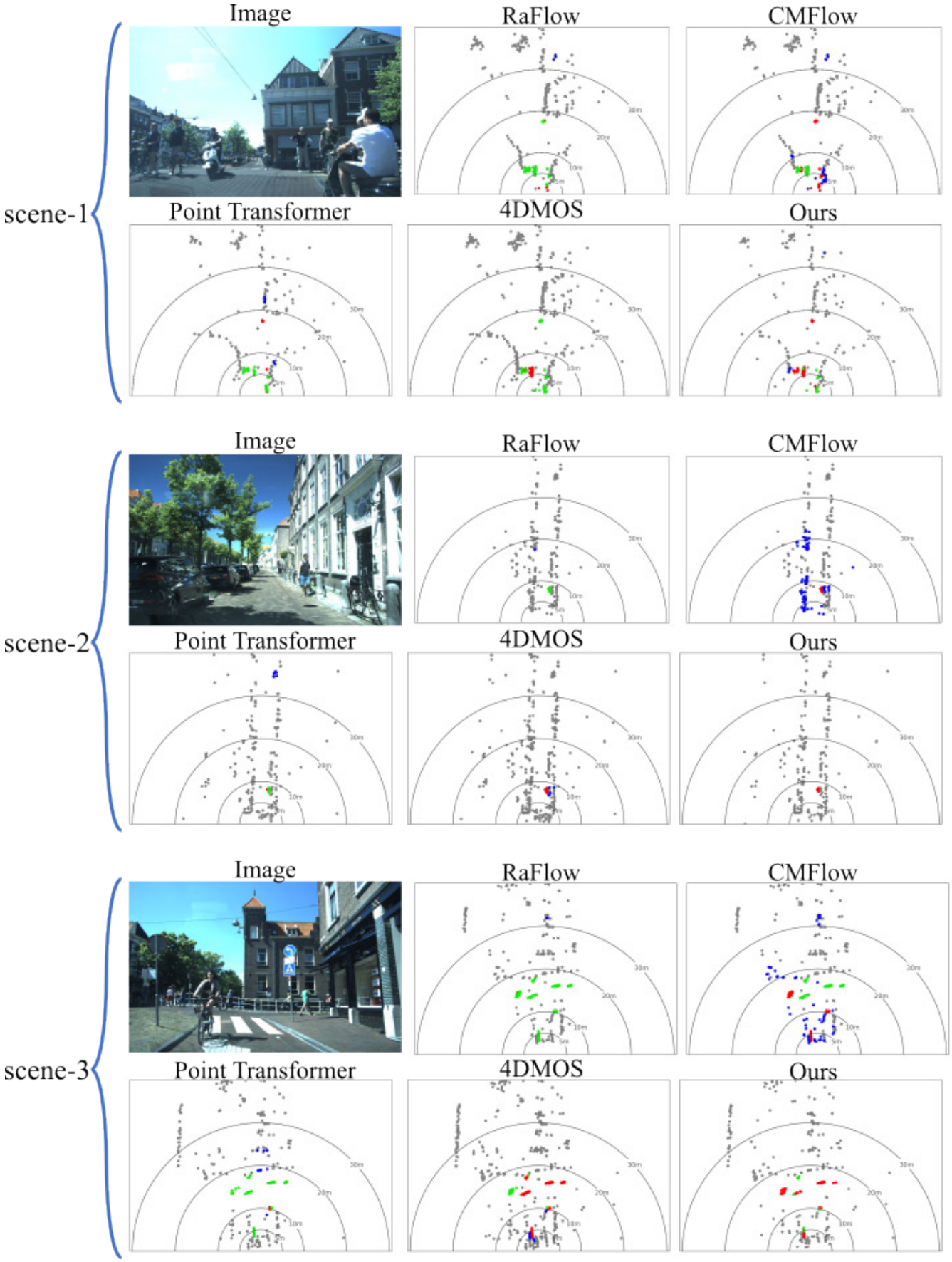

Figure 14: More qualitative results of Point Transformer, 4DMOS, and ours. The red points are the true moving points, the gray points are the true static points, the green points are the false moving points, and the blue points are the false static points.

Method	MOS			EVE	
	mIoU↑	**F1↑**	**mAcc↑**	**MAE↓**	**MSE↓**
RANSAC (2013)	40.7	50.7	58.5	0.713	0.624
ICP (1992)	14.1	23.0	49.9	0.964	0.978
PT (2021)	63.9	69.7	77.5	0.783	1.480
4DMOS (2022)	73.3	79.1	81.9	-	-
RaFlow (2022)	54.1	58.0	63.7	0.460	0.442
CMFlow (2023)	43.9	54.7	75.1	0.416	0.315
PT+V	75.3	80.7	86.3	-	-
4DMOS+V	75.6	81.7	83.8	-	-
Ours	**79.3**	**84.1**	**91.2**	**0.341**	**0.300**

Table 5: The MOSEVE evaluation results on VoD dataset

With Doploss	MAE ↓	MSE ↓	Dop. Err. ↓
Without doppler loss	0.212	0.091	0.329
with doppler loss	**0.182**	**0.065**	**0.283**

Dop. Err.: the mean absolute error between the actual radial velocity of static radar points and radial velocity estimated from ego-velocity;

Table 6: Ablation study on EVE with Dop Loss

Ball radius	MOS			EVE	
	mIoU↑	**F1↑**	**mAcc↑**	**MAE↓**	**MSE↓**
0.5	66.0	73.3	78.2	0.312	0.142
1	68.7	75.7	80.2	0.232	0.091
2	**70.2**	**76.5**	**81.9**	**0.182**	**0.065**
3	67.8	74.5	78.2	0.224	0.088
5	63.7	70.6	77.2	0.351	0.162

Table 7: Ablation study of ball query radius on MOS and EVE

approach maintains a high level of accuracy compared to other methods, with only 4DMOS closely approaching our method's performance. Additionally, other radar-based approaches show substantial disparities when contrasted with our method. Many moving points are misclassified as static points in RaFlow, while numerous static points are inaccurately predicted as moving points in CMFlow. More results are shown in the attached video.

E.3 Ablation study

Study on Doppler Loss. We investigate the impact of the proposed Doppler loss for EVE shown in Tab. 6. The table includes MAE and MSE results with and without the Doppler loss, indicating that training with Doppler loss in the EVE task leads to better performance than training without it. Furthermore, to comprehend the role of the doppler loss, we compare the mean absolute error between the actual radial velocity of static radar points and radial velocity estimated from ego velocity in both studies. Training with doppler loss can make radial velocity calculated from ego velocity more related to the actual radial velocity. This indicates doppler loss can enhance the network's capability to estimate ego velocity from actual static points.

Strategy	MOS			EVE	
	mIoU↑	**F1↑**	**mAcc↑**	**MAE↓**	**MSE↓**
[a] simultaneous	48.5	57.5	70.0	0.406	0.256
[b] sequential	57.9	65.7	76.3	**0.182**	**0.065**
[c] ours	**70.2**	**76.5**	**81.9**	**0.182**	**0.065**

Table 8: MOS and EVE with different training strategies

Study on Ball radius. We conducted a series of studies to select the radius of the ball query. As shown in Tab. 7, we tested the performance of ball radius of 0.5 m, 1 m, 2 m, 3 m, and 5 m, respectively, with the same configurations. The results indicate that the study on a 2 m ball radius achieves the best performance in both the MOS and EVE tasks. Therefore, we adopt a 2 m ball query radius as our final choice.

Training strategy. Finally, we explore different training strategies of multi-task multi-head networks for our radar MOSEVE tasks. These strategies include sharing the EVE backbone as the encoder for MOS, training two heads simultaneously [a], sequentially [b], and our final choice training two heads sequentially using two backbones [c]. The results are presented in Tab. 8. The results show that it is challenging for EVE and MOS to perform well using one multi-head network. Therefore, we adopt a two-stage method that performs EVE and MOS sequentially.

References

Besl, P. J.; and McKay, N. D. 1992. Method for registration of 3-D shapes. In *Sensor Fusion*, volume 1611, 586–606.

Ding, F.; Palffy, A.; Gavrila, D. M.; and Lu, C. X. 2023. Hidden Gems: 4D Radar Scene Flow Learning Using Cross-Modal Supervision. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 9340–9349.

Ding, F.; Pan, Z.; Deng, Y.; Deng, J.; and Lu, C. X. 2022. Self-Supervised Scene Flow Estimation With 4-D Automotive Radar. *IEEE Robotics and Automation Letters (RA-L)*, 1–8.

Kellner, D.; Barjenbruch, M.; Klappstein, J.; Dickmann, J.; and Dietmayer, K. 2013. Instantaneous ego-motion estimation using doppler radar. In *Proc. of the IEEE Intl. Conf. on Intelligent Transportation Systems (ITSC)*, 869–874.

Mersch, B.; Chen, X.; Vizzo, I.; Nunes, L.; Behley, J.; and Stachniss, C. 2022. Receding moving object segmentation in 3d lidar data using sparse 4d convolutions. *IEEE Robotics and Automation Letters (RA-L)*, 7(3): 7503–7510.

Palffy, A.; Pool, E.; Baratam, S.; Kooij, J. F.; and Gavrila, D. M. 2022. Multi-class road user detection with 3+ 1D radar in the View-of-Delft dataset. *IEEE Robotics and Automation Letters (RA-L)*, 7(2): 4961–4968.

Zhao, H.; Jiang, L.; Jia, J.; Torr, P. H.; and Koltun, V. 2021. Point transformer. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 16259–16268.