

國立臺灣大學電機資訊學院資訊網路與多媒體研究所

碩士論文

Graduate Institute of Networking and Multimedia

College of Electrical Engineering and Computer Science

National Taiwan University

Master Thesis

惡劣天氣條件下基於關聯注意力機制融合雷達和光達
進行物件偵測

Fusion of Radar and LiDAR Using Associative Mechanism
for Object Detection in Adverse Weather Conditions

陳柏維

Bo-Wei Chen

指導教授: 李明穗 博士

Advisor: Ming-Sui Lee, Ph.D.

中華民國 112 年 7 月

July, 2023

國立臺灣大學碩士學位論文
口試委員會審定書

惡劣天氣條件下基於關聯注意力機制融合雷達和光達進行物件偵測

Fusion of Radar and LiDAR Using Associative Mechanism
for Object Detection in Adverse Weather Conditions

本論文係陳柏維君（學號R10944044）在國立臺灣大學資訊網路與多媒體研究所完成之碩士學位論文，於民國一百一十二年七月三十一日承下列考試委員審查通過及口試及格，特此證明。

口試委員：

李明禮

（簽名）

（指導教授）

李福珍

李福珍

所長：

鄭卜壬



Acknowledgements

感謝李明穗教授讓我加入 IVlab 這個大家庭，這兩年間也教會我怎麼做研究，以及研究的時候該注意的事項，這兩年獲益良多，謝謝教授。



摘要

隨著深度學習技術的不斷發展，物件偵測的準確性也日益提高。自動駕駛 Level 5 的實現已經近在眼前。在良好的天氣條件下，物件偵測的平均精確度可以高達百分之八十五以上。然而，天氣並非時時都理想，有時候會下雨、起霧，甚至下雪，這種惡劣天氣會大幅降低物件偵測的準確性。

傳統的感測器，如攝像頭和 LiDAR，都容易受到惡劣天氣的影響。因此，我們採用 RADAR 和 LiDAR 的融合來進行物件偵測。RADAR 在惡劣環境下不受影響，但會產生許多噪點雲。因此，我們需要使用 LiDAR 作為輔助，因為 LiDAR 能提供精確的環境點雲信息，有助於減少虛擬偵測。

我們使用注意力機制來融合 LiDAR 和 RADAR 的特徵。同時，我們提出了特徵選取模塊 (Feature Selection Module)，解決了注意力機制中關注權重的問題。此外，我們還提出了關聯融合模塊 (Associative Feature Fusion Module)，充分利用注意力機制選取的特徵。通過實驗證明，我們提出的模型優於目前最先進的 RADAR 和 LiDAR 模型。

關鍵字：深度學習、多模態物件偵測、基於注意力機制進行特徵融合



Abstract

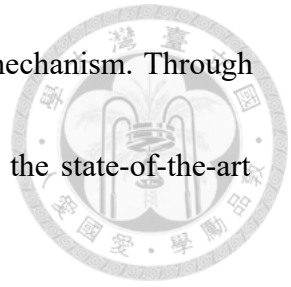
With the continuous development of deep learning technology, the accuracy of object detection has been steadily improving. The realization of Level 5 autonomous driving is within reach. In favorable weather conditions, the average accuracy of object detection can reach over 85 percent. However, the weather is not always ideal, and conditions such as rain, fog, and even snow can significantly reduce the accuracy of object detection.

Traditional sensors like cameras and LiDAR are susceptible to the influence of harsh weather conditions. Therefore, we adopt a fusion of RADAR and LiDAR for object detection. RADAR is unaffected by adverse environmental conditions but introduces a lot of noisy point clouds. Hence, we utilize LiDAR as an auxiliary sensor because it provides accurate environmental point cloud information, which helps mitigate ghost detection.

We employ an attention mechanism to fuse the features from LiDAR and RADAR. Additionally, we propose a Feature Selection Module to address the issue of attention weights in the attention mechanism. Furthermore, we introduce an Associative Feature

Fusion Module to fully utilize the selected features from the attention mechanism. Through experiments, we demonstrate that our proposed model outperforms the state-of-the-art RADAR and LiDAR models.

Keywords: Deep learning, multimodal object detection, feature fusion based on attention mechanism





Contents

	Page
Verification Letter from the Oral Examination Committee	i
Acknowledgements	ii
摘要	iii
Abstract	iv
Contents	vi
List of Figures	viii
List of Tables	x
Chapter 1 Introduction	1
Chapter 2 Related Work	6
2.1 Unimodal Sensor Detection	6
2.1.1 Radar-Only Dtection	7
2.1.2 LiDAR-Only Detection	8
2.2 Multimodal Sensor Fusion Detection	9
2.2.1 Lidar and Camera Fusion Detection	10
2.2.2 Lidar and Radar Fusion Detection	10
Chapter 3 Method	12
3.1 Framework Overview	13

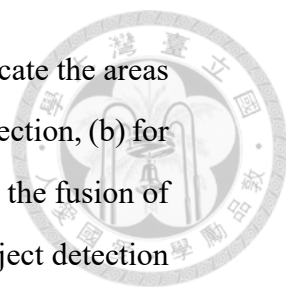
3.2	Feature Selection Module	14
3.2.1	Feature Spatial Selection	14
3.2.2	Feature Channel Selection	14
3.3	Associative Feature Fusion Module	15
Chapter 4	Experiments	17
4.1	Dataset Description	17
4.2	Implement Detail	18
4.2.1	Evaluation Metrics	18
4.3	Comparison with Only Radar	19
4.4	Comparison with Only LiDAR	19
4.5	Comparison with Radar and LiDAR	20
4.6	Ablation Study	21
4.6.1	Comparison with Other Feature Fusion Operation	21
4.6.2	Ablation of Model Components	22
4.7	Discussion	23
Chapter 5	Conclusion	29
	References	31





List of Figures

1.1	The blue boxes represent the ground truth, and it can be observed that in snowy weather, the camera fails to capture any objects.	2
1.2	The blue boxes represent the ground truth, and it can be observed that in snowy weather, lidar fails to capture any objects.	3
1.3	Samples data from the night case, and The yellow circles indicate the areas that differ from the ground truth, (a) for radar-only object detection, (b) for lidar-only object detection, and (c) for object detection with the fusion of radar and lidar. (d) represents the ground truth	5
2.1	Model architecture of Faster R-CNN	7
2.2	Model architecture of TRL	8
2.3	Model architecture of PointPillars	9
2.4	Model architecture of MVDNet	11
3.1	Architecture of RLANet	13
3.2	Feature Spatial Selection	15
3.3	Channel Feature Selection	15
3.4	Associative Feature Fusion Module	16
4.1	Samples data from the snow case, and The yellow circles indicate the areas that differ from the ground truth, (a) for radar-only object detection, (b) for lidar-only object detection, and (c) for object detection with the fusion of radar and lidar. (e) represents the ground truth,(f) shows object detection using only camera data trained in Deformable DETR.	25



4.2 Samples data from the fog case, and The yellow circles indicate the areas that differ from the ground truth, (a) for radar-only object detection, (b) for lidar-only object detection, and (c) for object detection with the fusion of radar and lidar. (e) represents the ground truth,(f) shows object detection using only camera data trained in Deformable DETR. 26

4.3 Samples data from the sunny case, and The yellow circles indicate the areas that differ from the ground truth, (a) for radar-only object detection, (b) for lidar-only object detection, and (c) for object detection with the fusion of radar and lidar. 27

4.4 Samples data from the sunny case, and The yellow circles indicate the areas that differ from the ground truth, (b) represents the ground truth, (c) shows object detection using only camera data trained in Deformable DETR. 28



List of Tables

4.1	The amount of each adverse weather condition in the RADIATE dataset. .	18
4.2	Comparison of state-of-the-art only radar methods on RADIATE test splits. The best result is highlighted in bold.	19
4.3	Comparison of state-of-the-art only LiDAR methods on RADIATE test splits. The best result is highlighted in bold.	20
4.4	Comparison of state-of-the-art radar and LiDAR methods on RADIATE test splits. The best result is highlighted in bold.	20
4.5	Comparison of other fusion methods on RADIATE test splits. The best result are highlighted in bold.	21
4.6	Comparison of convergence speed with self-attention.	22
4.7	Ablation study of model components. The best result is highlighted in bold.	23



Chapter 1 Introduction

With the rapid advancement of technology and the breakthroughs in deep learning techniques, achieving fully autonomous driving (Level 5)[6] has become highly possible. However, to achieve fully autonomous driving, we must overcome the challenges of degraded object detection performance in adverse weather conditions.

Currently, common sensors deployed in vehicles include camera, lidar, and radar. Camera provide rich semantic information, aiding in the detection of small objects. It is also cost-effective, helping to reduce the overall cost of vehicles. However, camera is not sufficiently robust in adverse weather conditions such as low light, fog, rain, or snow (as shown as Figure 1.1), which results in the loss of accurate visual information. Additionally, objects may appear differently in size when captured by camera at varying distances. These limitations severely impact the detection performance.

Lidar addresses some of the limitations of camera. It provides rich geometric information and operates effectively even in low light conditions. However, lidar still struggles to function properly in adverse weather conditions(as shown as Figure 1.2 such as rain, snow, or fog. The laser beams emitted by lidar can be scattered by fog, snowflakes, or raindrops, leading to inaccurate object localization.

Radar, due to its reliance on radio waves, remains operational in adverse weather



Figure 1.1: The blue boxes represent the ground truth, and it can be observed that in snowy weather, the camera fails to capture any objects.

conditions. It is not affected by fog, snow, or raindrops. However, radar-generated point cloud data may contain noise, resulting in ghost detection[39] and decreased detection performance.

In unimodal object detection, camera-based detection[23, 30, 32, 34, 35] and lidar-based detection[3, 11, 17, 29, 37, 40] are widely studied as primary directions. Camera-based detection[23, 30, 32, 34, 35] allows for cost-effective implementation in self-driving vehicles and provides rich visual information with a perspective similar to that of humans, making it easy to interpret. However, camera-based detection[23, 30, 32, 34, 35] faces challenges in terms of lacking depth and geometric information, and its performance is heavily influenced by environmental factors such as lighting conditions, limiting its capability to achieve higher levels of performance.

Lidar-based detection[3, 11, 17, 29, 37, 40] provides rich geometric information and accurate localization, maintaining excellent performance even in low-light environments. However, the point cloud imaging provided by Lidar is sparser compared to RGB images[5]., and direct utilization of point clouds for deep learning requires substantial mem-

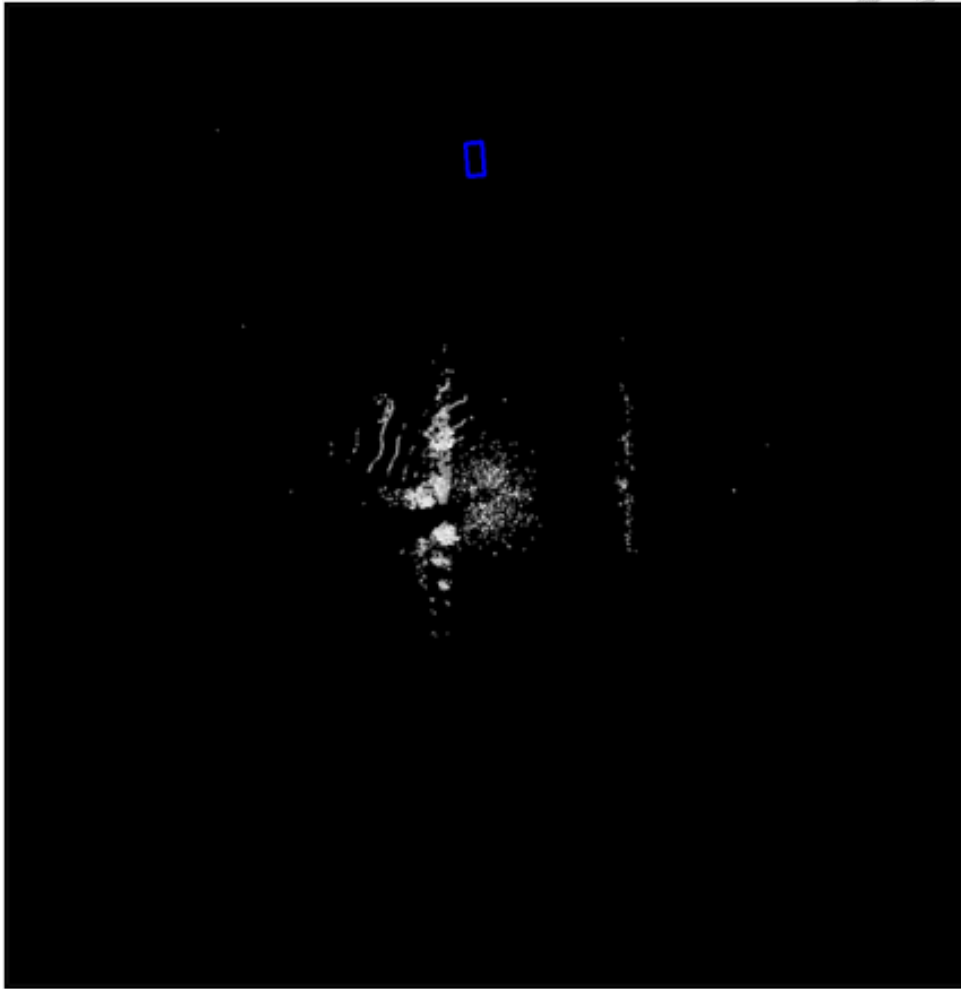
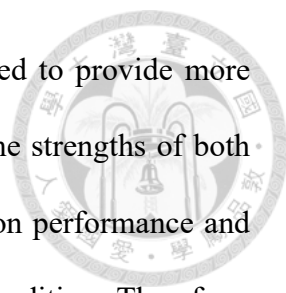


Figure 1.2: The blue boxes represent the ground truth, and it can be observed that in snowy weather, lidar fails to capture any objects.

ory and computational resources.

In recent years, there has been growing interest in multimodal object detection to leverage the unique characteristics of different sensors. In multimodal object detection, the fusion of camera and lidar[1, 10, 13, 14, 36] has become a mainstream research direction. This fusion allows for the simultaneous utilization of visual information and dense features from cameras, as well as geometric information and precise object positioning from lidar. It helps overcome the performance degradation caused by insufficient lighting conditions. However, the fusion of camera and lidar[1, 10, 13, 14, 36] still fails to provide robust performance in adverse weather conditions.



Considering these factors, combining radar and lidar is believed to provide more robust performance in adverse weather conditions. By leveraging the strengths of both sensors, overcome the limitations of each and improve the perception performance and safety of fully autonomous driving systems, even in adverse weather conditions. Therefore, a novel end-to-end radar and lidar fusion model is proposed, and model that leverages attention mechanism for feature fusion. The following are the contributions of this paper:

1. A novel end-to-end multimodal object detection model is proposed, which includes Feature Selection Module and Associative Feature Fusion Module, and the model outperforms the current state-of-the-art by 4.68% in adverse weather conditions.
2. Feature Selection Module is introduced to address the limitation of attention mechanism.
3. Associative Feature Fusion Module captures the surrounding features of the query and attends to channel features to improve performance.

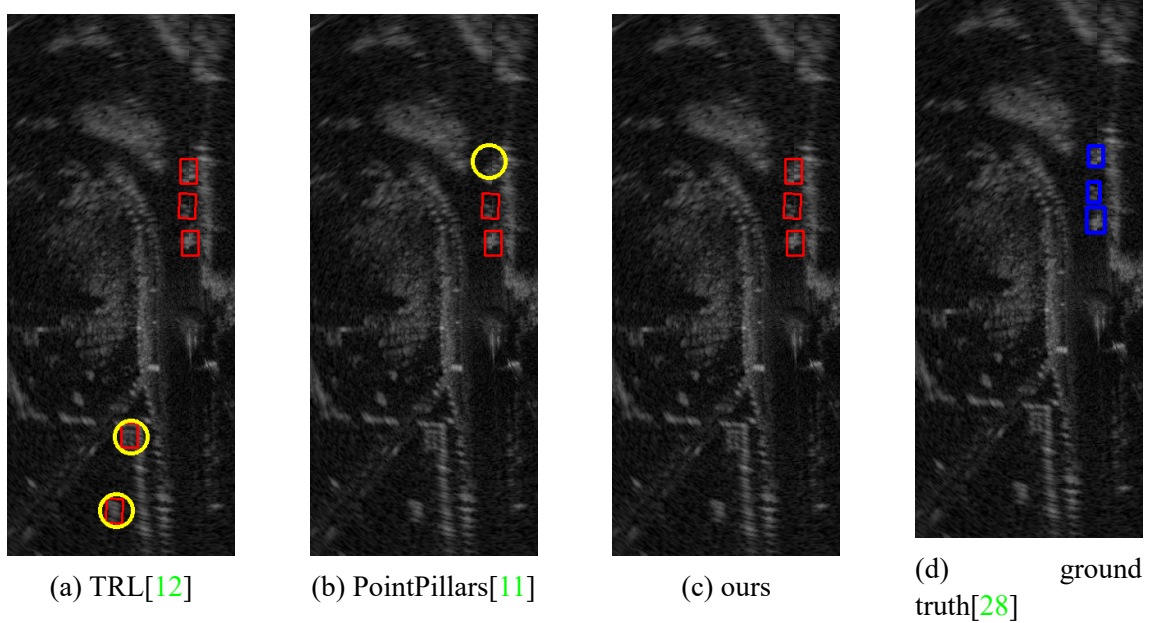


Figure 1.3: Samples data from the night case, and The yellow circles indicate the areas that differ from the ground truth, (a) for radar-only object detection, (b) for lidar-only object detection, and (c) for object detection with the fusion of radar and lidar. (d) represents the ground truth



Chapter 2 Related Work

In the field of autonomous driving, researchers are studying various methods for object detection using different sensors. These sensors include popular ones like cameras, radars, and lidars, as well as less common ones like gated cameras and infrared cameras. This article will focus on methods relevant to our research. Firstly, there is Radar-only detection, which focuses on object detection using radar sensors. Next, there is Lidar-only detection, which utilizes lidar sensors for object detection. Additionally, there are methods for Lidar-camera fusion, where information from lidar and cameras is combined to enhance object detection performance. Similarly, there are methods for Lidar-radar fusion, which combine information from lidar and radar for object detection.

2.1 Unimodal Sensor Detection

Unimodal sensor detection refers to the task of object detection using a single type of sensor or modality. In the context of computer vision, this typically involves using a single sensor, such as a camera or a lidar, to detect and localize objects in a scene.



2.1.1 Radar-Only Dtection

There is relatively less research on using radar alone for object detection. This is mainly due to the limitations of radar data provided by most autonomous driving datasets, which typically use mmWave radar. The low resolution of radar data makes it challenging to use radar alone for object detection. However, the RADIATE dataset provides high-resolution radar data that can be used for standalone radar-based object detection. RADIATE also offers a baseline model based on the Faster R-CNN[27](Figure 2.1) algorithm for detection. Another method[12](Figure 2.2)utilizes radar alone for object detection. [12]observes that vehicles exhibit similar features in radar images, and objects in consecutive frames also share similar features. Therefore, [12]leverages attention mechanisms to establish temporal relationships between objects in current and previous frames, aiming to enhance detection performance.

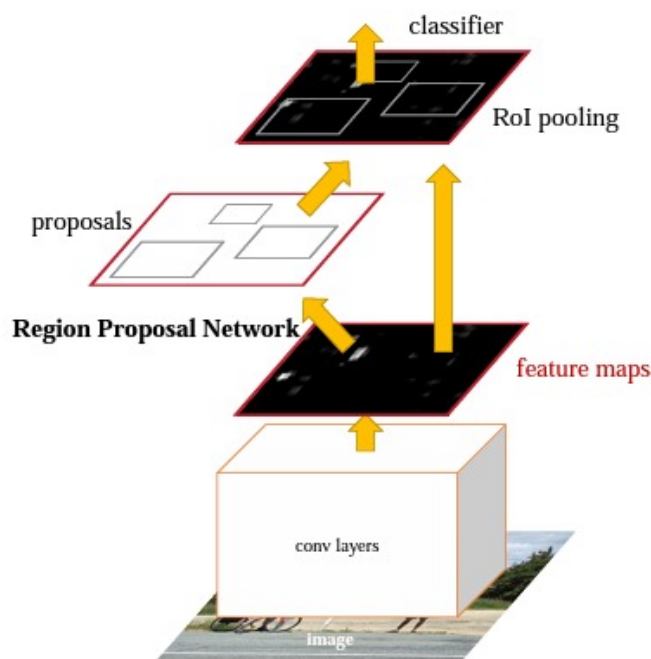


Figure 2.1: Model architecture of Faster R-CNN [27].

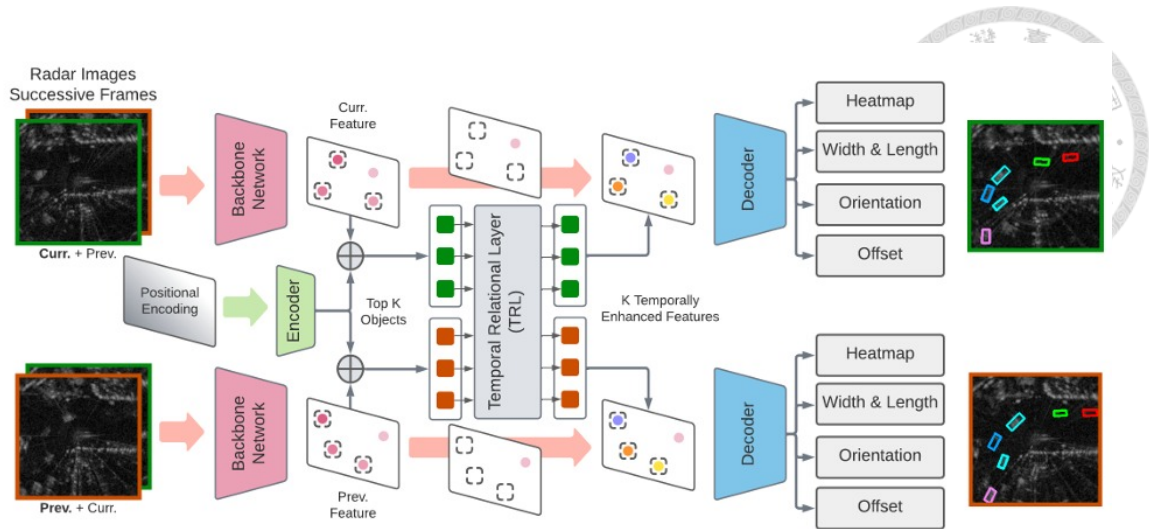


Figure 2.2: Model architecture of TRL [12].

2.1.2 LiDAR-Only Detection

”Using LiDAR alone for object detection is a classic and popular research field. However, before utilizing LiDAR for object detection, it is necessary to determine how to process LiDAR point cloud data. The common approach is to directly process the point cloud, such as with the PointNet series [19–21]. However, this method has high GPU requirements, so some studies convert the point cloud into voxel data. The advantage of voxel conversion is to avoid the sparsity of the point cloud and treat it as a three-dimensional feature map. Methods[17, 40] employ this processing technique. However, even though converting to voxels reduces the computational resource demand compared to directly processing the point cloud, it still requires a certain level of GPU capability.

Therefore, PointPillars[11](Figure 2.3)encodes the point cloud into a two-dimensional feature map. The method involves projecting the point cloud onto a 2D plane and then voxelizing it. Unlike VoxelNet, VoxelTransformer[17, 40], PointPillars[11] performs voxelization after projection, resulting in only two-dimensional features. This enables the utilization of common models such as CNN and VGG for feature extraction. PointPil-

lars[11] requires fewer computational resources and achieves faster computation speed than other methods that solely use LiDAR for detection. As a result, some approaches like TransFusion[1] utilize PointPillars[11] as their backbone network.

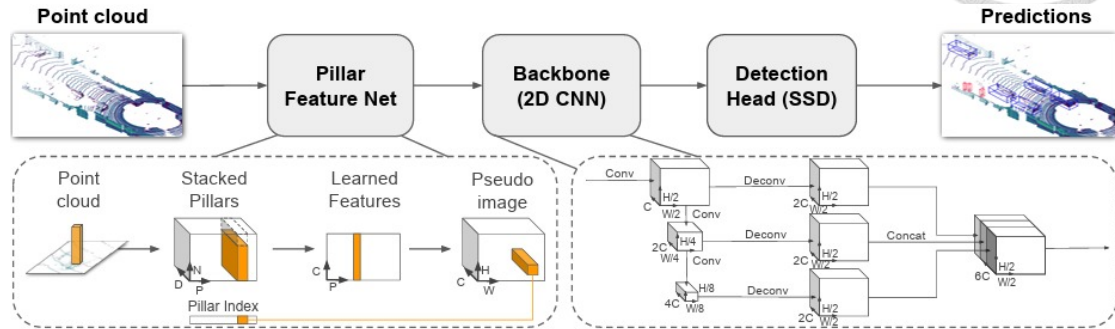
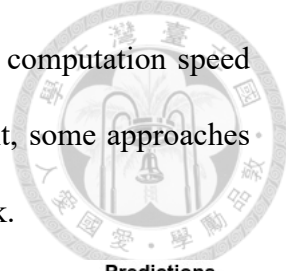


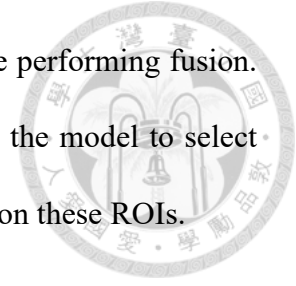
Figure 2.3: Model architecture of PointPillars [11].

2.2 Multimodal Sensor Fusion Detection

Multimodal sensor fusion object detection is an approach that combines multiple sensors to obtain richer and more comprehensive object detection information. In this approach, instead of relying on a single sensor, multiple different types of sensors, such as cameras, radar, and lidar, are integrated to achieve a higher level of perception. Through multimodal sensor fusion, the limitations and shortcomings of individual sensors can be overcome, while leveraging the strengths of multiple sensors to improve the accuracy of object detection.

The challenge of multimodal sensor fusion detection lies in how to fuse features from different modalities. Existing fusion methods can be categorized into **early fusion**, **middle fusion**, and **late fusion**. **Early fusion** involves fusing the raw data before inputting it into the model, with a focus on integrating the raw data. For example, some methods project LiDAR point clouds onto RGB images and perform subsequent processing. **Middle fusion** entails feeding the raw data into the model to extract feature maps, which are

then fused. These methods extract features from the raw data before performing fusion. **Late fusion**, also known as decision fusion, involves first allowing the model to select regions of interest (ROI) and then making the final prediction based on these ROIs.



2.2.1 Lidar and Camera Fusion Detection

Lidar-Camera Fusion Detection is a perceptual technique that combines the capabilities of both lidar and camera sensors for object detection and environmental perception. This approach leverages the information from both lidar and camera to synergistically enhance the perception and object recognition capabilities in the surrounding environment. Several methods[1, 13, 14] have achieved remarkable results in Lidar-Camera fusion detection in the field of object detection, however, Lidar-Camera Fusion Detection is still subject to performance degradation in adverse weather due to inherent limitations of the sensors.

2.2.2 Lidar and Radar Fusion Detection

Lidar-Radar fusion is a relatively less explored research area, primarily due to the low resolution of radar, which makes it challenging to establish meaningful features for detection models. However, radar offers certain advantages over camera and lidar in terms of stability, especially in adverse environmental conditions. Additionally, radar shares the same coordinate system as lidar, eliminating the need for coordinate transformation and potential loss of features in camera-lidar fusion approaches.

With the release of datasets like RADIATE dataset and Oxford radar robot car dataset, there has been a growing interest in Lidar-Radar fusion research. One notable example

is the MVDNet[22](Figure 2.4), which leverages radar and lidar fusion. It demonstrates excellent performance in both clear weather and foggy weather conditions.

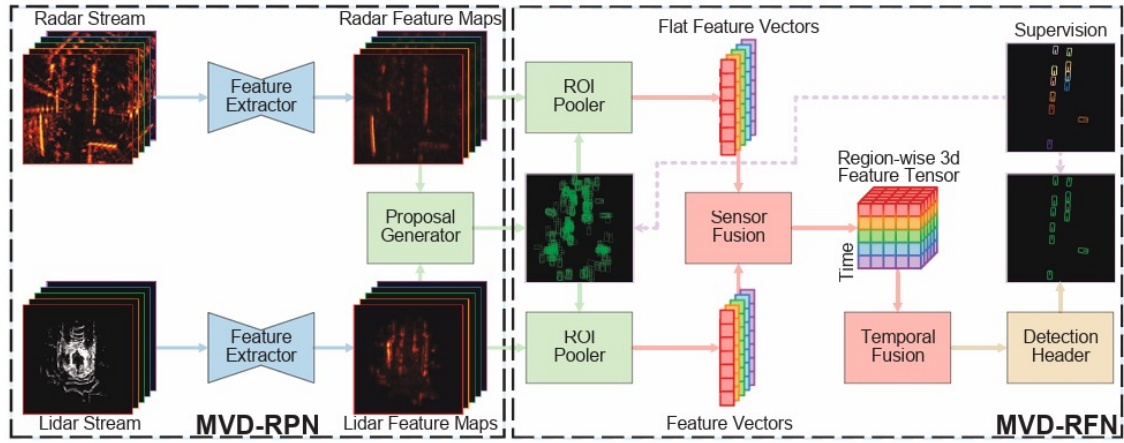


Figure 2.4: Model architecture of MVDNet [22].

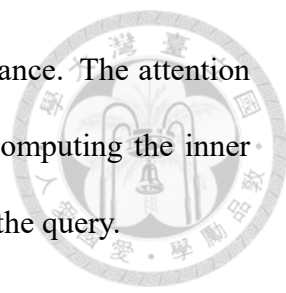


Chapter 3 Method

A novel end-to-end radar and lidar fusion model called RLANet is proposed, same as [7, 41], adopts the DETR architecture. Unlike the faster-rcnn series [9, 27] or the YOLO series [8, 24–26], the DETR architecture does not require additional hand-crafted design. According to [38], the final performance of a model is influenced by hand-crafted design. Therefore, choosing the DETR architecture avoids the cumbersome hand-crafted design and does not compromise the performance of model.

There are three main methods for feature fusion: element-wise operation, concatenation, and attention mechanism[33]. Element-wise operation combines features through simple arithmetic operations. However, if there are many noisy features in the fusion process, the fused feature will also be heavily influenced by the noise. Concatenation can address the limitations of element-wise operation, but it may introduce data alignment issues as the fused features come from different sensors. High-quality calibration is required to align the data, but there may still be errors. Therefore, we adopt the attention mechanism for feature fusion, because it can learn relationship between different data, so it solve data alignment problem.

However, the attention mechanism[33] has a drawback in that it tends to focus too much on irrelevant regions at the beginning, leading to longer convergence time and higher



computational cost, and it may result in poor overall model performance. The attention mechanism calculates the similarity between queries and keys by computing the inner product of isolated key-query pairs, which neglects the features near the query.

Therefore, the proposed Feature Selection Module and Associative Feature Fusion Module are introduced to address these issues. Experimental results on RADIATE demonstrate that RLANet outperforms current state-of-the-art methods.

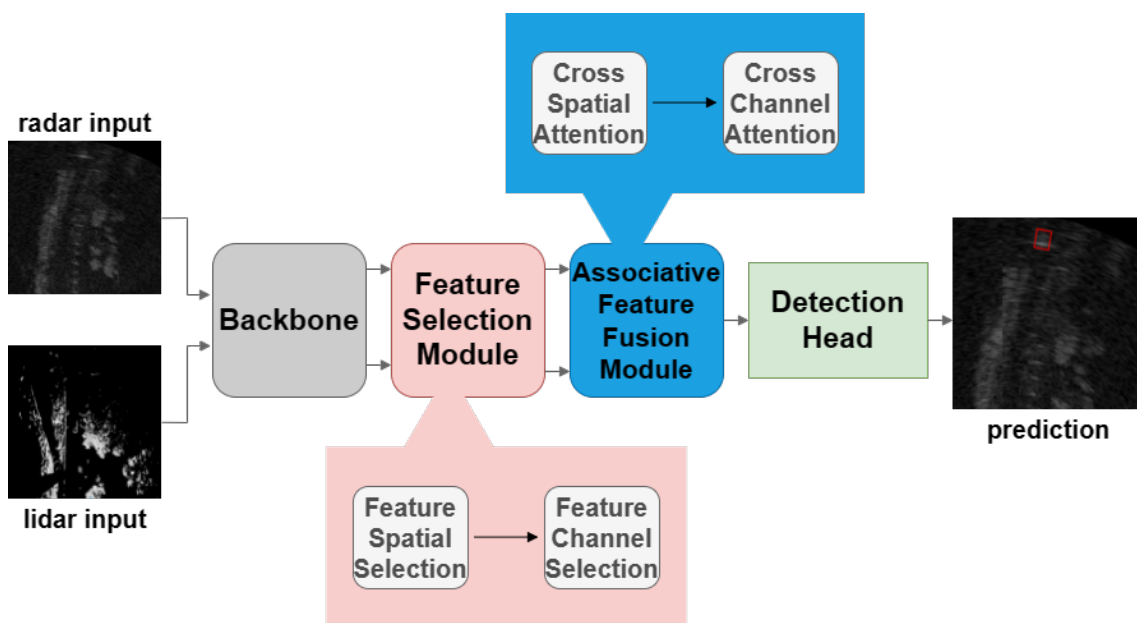
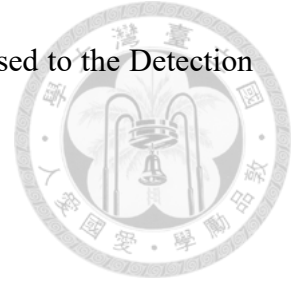


Figure 3.1: Architecture of RLANet.

3.1 Framework Overview

Figure. 3.1 illustrates the architecture of RLANet. The processing workflow is as follows: Firstly, the radar image and lidar image are separately fed into the backbone network to extract feature maps. Then, the feature maps are passed through the Select Feature Module for the first round of feature map weight calculation. Next, the feature maps are forwarded to the Associative Feature Fusion Module for feature fusion. In this stage, spatial attention and channel attention are computed to obtain the fused features.

Finally, the fused features are refined by the Refine Module and passed to the Detection Head to obtain the final detection results.



3.2 Feature Selection Module

The main purpose of designing this module is to address the issue with the attention mechanism, where it tends to excessively focus on unimportant areas of the feature map during the initial stages. This leads to prolonged model convergence time and excessive computational burden.

3.2.1 Feature Spatial Selection

This module (as shown in Figure 3.2) is primarily designed for selecting spatial features. The operation process of this module is as follows: first, it performs max and mean pooling along the channel dimension, compressing the channel dimension into a 1-dimensional vector. Then, it concatenates the max and mean pooling results and utilizes point-wise convolution to further reduce the channel dimension to 1. Next, a series of operations including layer normalization, GELU activation, and depth-wise convolution are applied. Finally, a sigmoid function is used to obtain the spatial attention weight, which is multiplied with the input feature to obtain the selected spatial feature.

3.2.2 Feature Channel Selection

This module (as shown in Figure 3.3) is primarily designed for selecting channel features. The operation process of this module is as follows: it performs average pooling and max pooling and then adds these two results together to increase the feature richness.

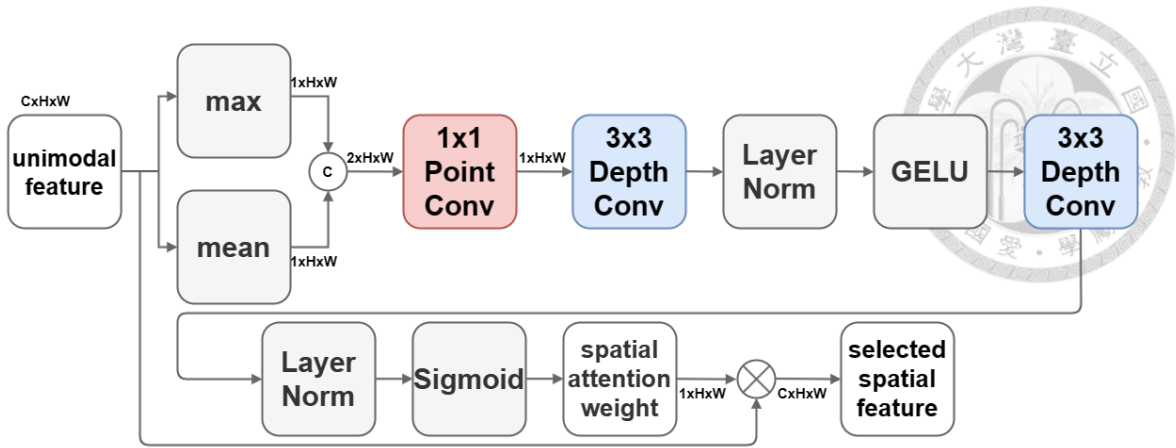


Figure 3.2: Feature Spatial Selection

Next, a series of operations including point-wise convolution, layer normalization, and ReLU activation are applied. Finally, a sigmoid function is used to obtain the channel attention weight, which is multiplied with the input feature to obtain the final selected channel feature.

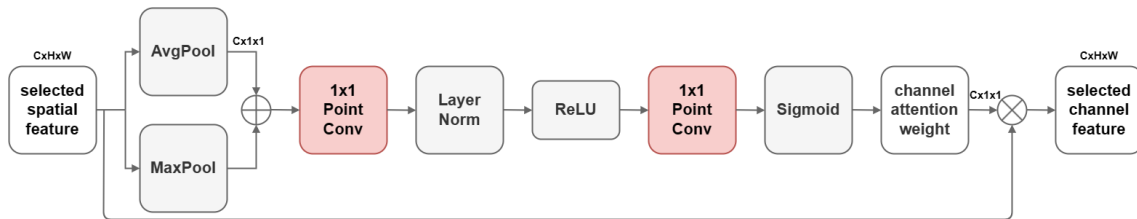


Figure 3.3: Channel Feature Selection

3.3 Associative Feature Fusion Module

The purpose of this module is to improve the traditional attention mechanism, which calculates similarity only between isolated key-query pairs. However, there is a high correlation between features near the query. Therefore, considering the surrounding features together when calculating the similarity between the query and the key can enhance performance.

Figure 3.4 illustrates the architecture of the Associative Feature Fusion Module,



which consists of the Cross Spatial Attention Module and the Cross Channel Attention Module.

The Cross Spatial Attention Module aims to extract features near the query in a comprehensive manner. It can be observed that the key undergoes a 3x3 convolution operation and is then concatenated with the query to obtain the key-query feature. Subsequently, two 1x1 convolution operations are performed to generate the weight.

Unlike the conventional attention mechanism that calculates similarity using vector dot products to obtain the weight matrix, here, the two 1x1 convolutions are utilized to enable the model to learn the weights, thereby reducing computational complexity. Once the weight matrix is obtained, it is multiplied with the value through matrix multiplication to obtain the final result.

The Cross Channel Attention Module combines the output of the Cross Spatial Attention Module with the query through channel attention fusion. It can be observed that the Cross Channel Attention Module has two branches. The lower branch performs global average pooling to obtain global channel features. The results from the upper and lower branches are then added together and passed through the sigmoid function to obtain the weights. These weights are multiplied with the results obtained from the query and the Cross Spatial Attention Module, and then added together to obtain the final fused feature.

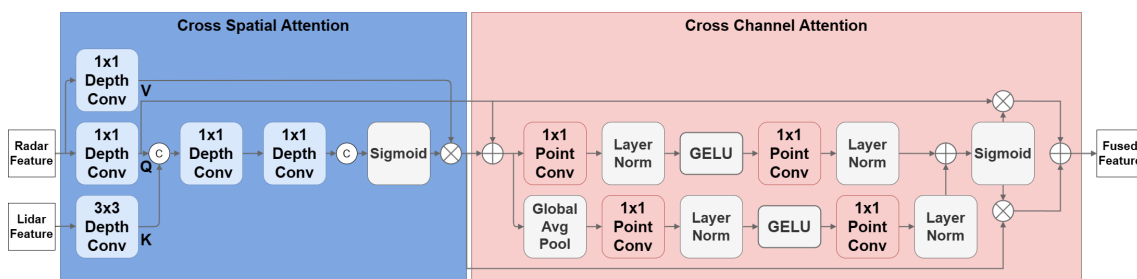


Figure 3.4: Associative Feature Fusion Module



Chapter 4 Experiments

4.1 Dataset Description

RADIATE[28] provides high-resolution radar data, which is different from other datasets. For example, nuScenes[4] collects radar data using millimeter-wave radar, but its resolution is low, resulting in the model's inability to fully utilize its features. Although Waymo[31] provides high-resolution lidar data and camera data, it does not include radar data. Additionally, although ORR[2] uses the same radar as RADIATE[28] and provides high-resolution radar data, it does not provide annotations, making it unsuitable for object detection. Therefore, we conducted experiments on the RADIATE[28] dataset.

RADIATE[28] is a large-scale dataset that consists of five hours of data. The dataset captures scenes under different weather conditions, including sunny, overcast, urban, night, rainy, foggy, and snowy conditions. In the field of autonomous driving, it is crucial to have stable performance of object detection in challenging environments. Currently, apart from RADIATE, there is no other dataset that provides such diverse weather conditions. At most, datasets offer variations in different scenes. Therefore, RADIATE dataset can bridge the gap between ideal and practical applications.

Table 4.1: The amount of each adverse weather condition in the RADIATE dataset.

	Total	Sunny	Overcast	Urban	Night	Rain	Fog	Snow
Training	23540	9611	1806	8438	890	2513	282	0
Testing	8335	2384	726	1325	1779	689	766	666

4.2 Implement Detail

Processing raw LiDAR point clouds requires a significant amount of GPU memory. Therefore, the LiDAR point clouds are projected onto a 2D plane, converting them into image data. Although this approach reduces the richness of the features, it significantly decreases the GPU memory requirement and training time compared to the direct processing of lidar point clouds.

The experimental environment was built using Python 3.8, PyTorch 12.1[18], CUDA 11.1, and mmrotate 0.1.0. The training parameters are set as follows: the input image resolution is 1152x1152, the batch size was 3, and the learning rate is set to 0.00001. The AdamW optimizer[16] is used to optimize the model, and the focal loss function[15] is employed to encourage model convergence. The training process is conducted on a single NVIDIA RTX A6000.

4.2.1 Evaluation Metrics

The evaluation metric employed to gauge the model’s performance is average precision (AP). Specifically, AP_{50} is computed, where a detection is considered successful if the Intersection over Union (IOU) between the predicted bounding box and the ground truth bounding box surpasses 0.5.



4.3 Comparison with Only Radar

In Table 4.2, "Baseline" refers to a model introduced in the RADIATE dataset[28], utilizing the Faster R-CNN architecture[27]. TRL[12] utilizes both the current frame and the previous frame data, divided into two branches, resulting in a total of four frames of feature information. Despite using only a single frame for object detection, RLANet model's performance is still slightly better than TRL[12].

Table 4.2: Comparison of state-of-the-art only radar methods on RADIATE test splits. The best result is highlighted in bold.

Model	Overall	Sunny	Overcast	Urban	Night	Rain	Fog	Snow
Baseline	45.84	78.88	41.91	30.36	40.49	29.18	48.30	15.16
TRL	54.00	80.53	46.62	50.34	60.32	39.76	61.15	34.85
RLANet(Ours)	54.81	81.20	47.81	53.37	60.75	40.31	60.28	36.14

4.4 Comparison with Only LiDAR

We also conducted a comparison with PointPillars[11]. Although both PointPillars[11] and the approach being considered utilize lidar data, they employ distinct processing methods. PointPillars[11] initially voxelizes the lidar data before projecting it onto a 2D plane, creating a bird eye view feature map with dimensions $C \times H \times W$. In contrast, the lidar data is directly projected into an image using a projection matrix in the current method, resulting in dimensions of $3 \times H \times W$, with C being significantly larger than 3. This design grants PointPillars[11] a more enriched point cloud feature representation compared to the current method. However, the current approach holds the advantage of faster data loading due to its point cloud processing methodology."

While PointPillars[11] possesses at least 10 times more features than the method

under consideration (PointPillars[11] features consist of a minimum of 30 channels), its performance does not exhibit a significant superiority over the our method. As a result, the outcome is indeed quite satisfactory.

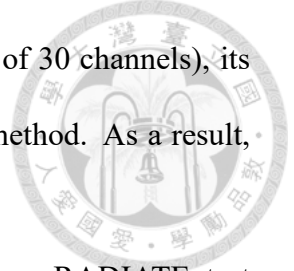


Table 4.3: Comparison of state-of-the-art only LiDAR methods on RADIATE test splits. The best result is highlighted in bold.

Model	Overall	Sunny	Overcast	Urban	Night	Rain	Fog	Snow
PointPillars	51.34	81.03	45.89	40.19	55.27	30.16	47.86	13.84
RLANet(Ours)	47.27	79.17	44.07	36.79	53.34	27.10	44.45	10.66

4.5 Comparison with Radar and LiDAR

In this section, a comparison is made between the approach and MVDNet[22]. MVDNet[22] utilizes a similar architecture to Faster-RCNN[27] and employs a similar approach to PointPillars for handling lidar data, enabling the preservation of a greater amount of feature information. Furthermore, MVDNet[22] incorporates traditional attention mechanisms[33] for feature fusion to effectively integrate the features from both radar and lidar modalities. Currently, MVDNet[22] stands as the state-of-the-art method for Radar and lidar fusion.

Compared to MVDNet[22], RLANet not only eliminates the need for additional handcrafted designs but also addresses the issues caused by traditional attention mechanisms. As shown in Table 4.4, RLANet outperforms MVDNet in terms of performance.

Table 4.4: Comparison of state-of-the-art radar and LiDAR methods on RADIATE test splits. The best result is highlighted in bold.

Model	Overall	Sunny	Overcast	Urban	Night	Rain	Fog	Snow
MVDNet	54.92	81.71	48.51	54.45	60.90	40.56	62.25	38.64
RLANet(Ours)	57.49	83.31	50.12	56.04	63.14	43.51	66.92	41.43



4.6 Ablation Study

In the ablation study section, the results of utilizing the Associative Feature Fusion Module for feature fusion will be analyzed in comparison with three other feature fusion operations: element-wise operation, concatenate operation, and the traditional attention mechanism[33].

Additionally, experiments will be conducted to demonstrate the effectiveness of the two key components in the model, namely the Feature Selection Module and the Associative Feature Fusion Module.

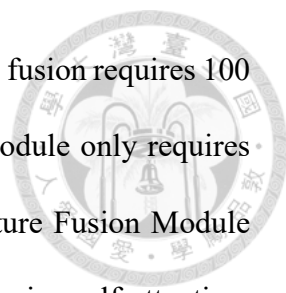
4.6.1 Comparison with Other Feature Fusion Operation

In Table 4.5 present the results of using the Associative Feature Fusion Module for feature fusion. Compared to the other three feature fusion operation, the Associative Feature Fusion Module exhibits superior performance. This is because the Associative Feature Fusion Module does not mix noisy features into the new features like element-wise operations, it does not cause data alignment issues like concatenate operations, and it avoids excessive focus on irrelevant regions as seen in traditional attention mechanisms. Therefore, using the Associative Feature Fusion Module leads to better overall performance.

Table 4.5: Comparison of other fusion methods on RADIATE test splits. The best result are highlighted in bold.

Model	Overall	Sunny	Overcast	Urban	Night	Rain	Fog	Snow
Concatenate	50.91	79.28	45.05	47.69	54.11	32.27	52.41	30.25
Self-attention[33]	52.30	79.75	44.92	48.14	55.32	33.71	54.19	34.67
RLANet(Ours)	57.49	83.31	50.12	56.04	63.14	43.51	66.92	41.43

In Table 4.6, "Self-attention" denotes the feature fusion method using self-attention.



It can be observed that the model trained with self-attention for feature fusion requires 100 epochs, while the model utilizing the Associative Feature Fusion Module only requires 36 epochs. Furthermore, the model employing the Associative Feature Fusion Module for feature fusion exhibits better performance compared to the model using self-attention. This improvement primarily stems from the fact that the Associative Feature Fusion Module addresses the issue of self-attention focusing excessively on irrelevant features. Additionally, the module effectively leverages features surrounding the query, enhancing the model’s learning capability.

Table 4.6: Comparison of convergence speed with self-attention.

Model	Overall	Epoch
Self-attention[33]	52.30	100
RLANet(Ours)	57.49	36

4.6.2 Ablation of Model Components

In this section, the effectiveness of model components will be validated. Table 4.7 shows the performance of different configurations. "Base" represents using only the backbone and the self-attention mechanism[33] to fuse radar and lidar features for object detection. "w/o F.S.M." denotes experiments without utilizing the Feature Selection Module, and "w/o A.F.M." represents experiments conducted without using the Associative Feature Fusion Module. Instead, traditional attention[33] was employed for feature fusion.

The results presented in Table 4.7 demonstrate the contributions of each component to the model’s performance. It is evident that each component significantly impacts the model’s performance, and none of them can be omitted.

Table 4.7: Ablation study of model components. The best result is highlighted in bold.

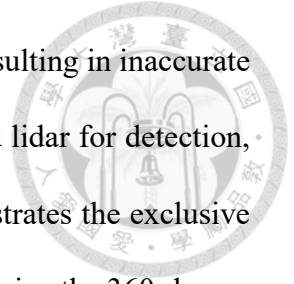
Model	Overall	Sunny	Overcast	Urban	Night	Rain	Fog	Snow
Base	48.73	76.91	42.62	43.44	52.84	30.51	50.35	30.94
w/o F.S.M.	53.71	80.51	47.19	50.69	59.11	39.27	60.14	37.25
w/o A.F.M.	52.30	79.75	44.92	48.14	55.32	33.71	54.19	34.67
Full model	57.49	83.31	50.12	56.04	63.14	43.51	66.92	41.43

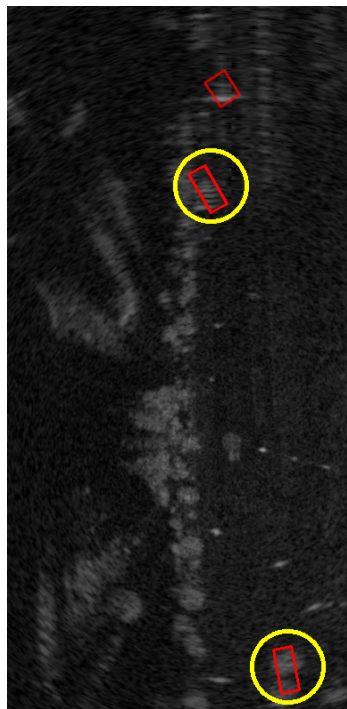
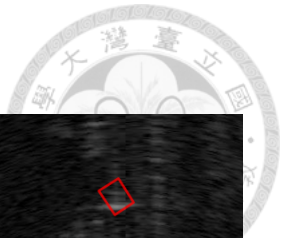
4.7 Discussion

Figures 4.1 and 4.2 illustrate two scenarios: snowy weather and foggy weather, respectively. Due to the weather resilience of radar, TRL(Figures 4.1(a) and Figures 4.2(a)) demonstrates accurate vehicle predictions even in adverse conditions, utilizing radar alone. Nevertheless, radar-generated point clouds contain a significant amount of noise, resulting in numerous false detections within TRL’s outcomes. PointPillars(Figures 4.1(b) and Figures 4.2(b)), which relies solely on lidar, exhibits a plethora of erroneous detections. This is primarily attributed to the phenomenon where the laser beams emitted by the lidar may refract when passing through snow or fog particles, leading to imprecise vehicle localization. Similarly, the fusion of radar and lidar in MVDNet(Figures 4.1(c) and Figures 4.2(c)) also encounters instances of false detections. Deformable DETR(Figures 4.1(f) and Figures 4.2(f)) showcases exclusive camera-based detection, completely failing to identify vehicles. This further verifies the inadequate performance of cameras under harsh weather conditions. However, RLANet(Figures 4.1(d) and Figures 4.2(d)) precisely detects vehicle.

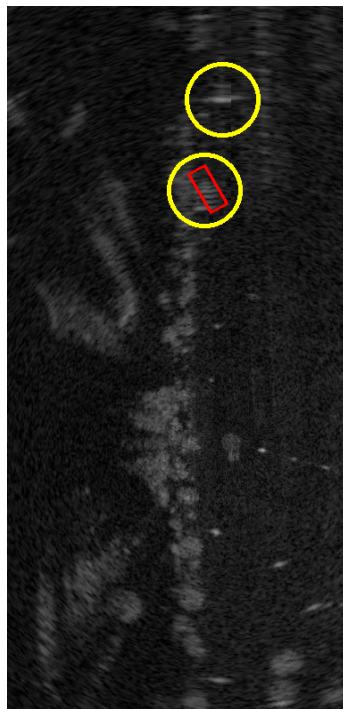
Both Figure 4.3 and Figure 4.4 depict scenarios under sunny conditions. In Figure 4.3(a), the approach utilizing radar-only detection exhibits numerous instances of ghost detection. Conversely, Figure 4.3(b), which solely employs lidar for detection, misses two vehicles. This outcome is primarily attributed to the lidar’s laser beams encountering

obstacles that prevent accurate reflection back to the lidar, thereby resulting in inaccurate vehicle detection. Figure 4.3(c), representing the fusion of radar and lidar for detection, displays a single missed vehicle. Furthermore, Figure 4.4(c) demonstrates the exclusive camera-based object detection method, which lacks the ability to perceive the 360-degree scene as effectively as the other methods. This underscores the comparative deficiency of environmental perception in the camera-only approach.”





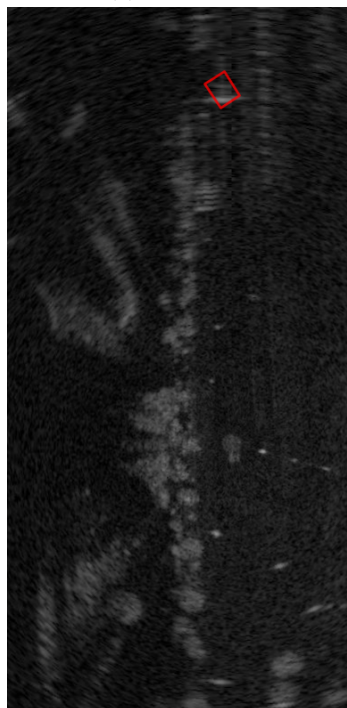
(a) TRL[12]



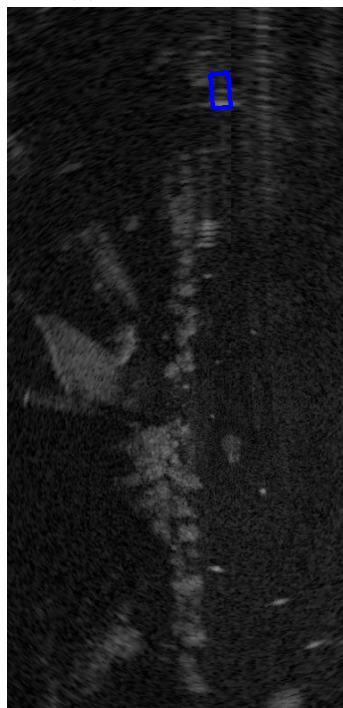
(b) PointPillars[11]



(c) MVDNet[22]



(d) RLANet (ours)

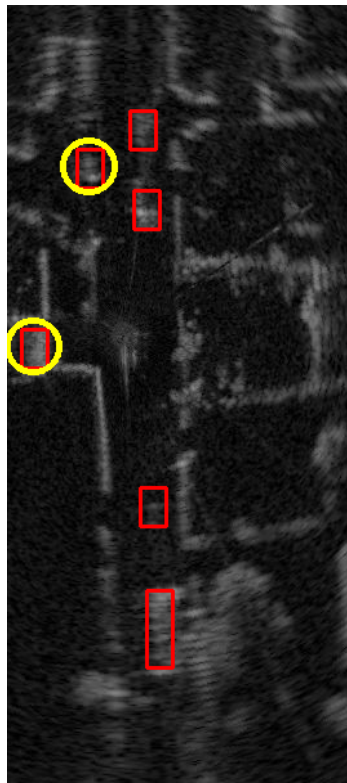


(e) ground-truth[28]

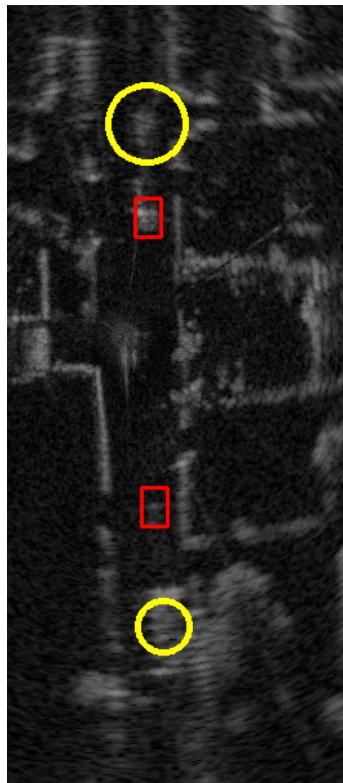


(f) Deformable DETR[41]

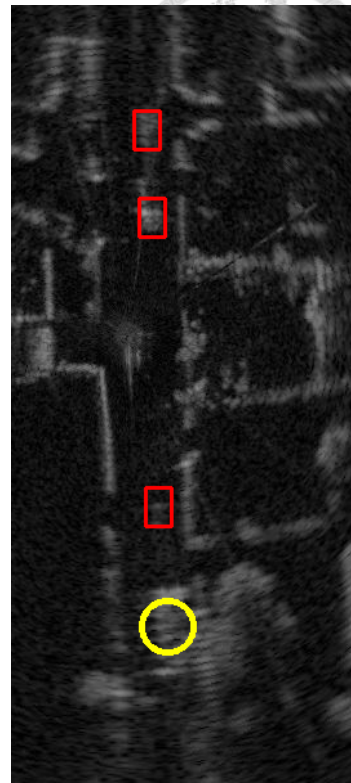
Figure 4.1: Samples data from the snow case, and The yellow circles indicate the areas that differ from the ground truth, (a) for radar-only object detection, (b) for lidar-only object detection, and (c) for object detection with the fusion of radar and lidar. (e) represents the ground truth,(f) shows object detection using only camera data trained in Deformable DETR.



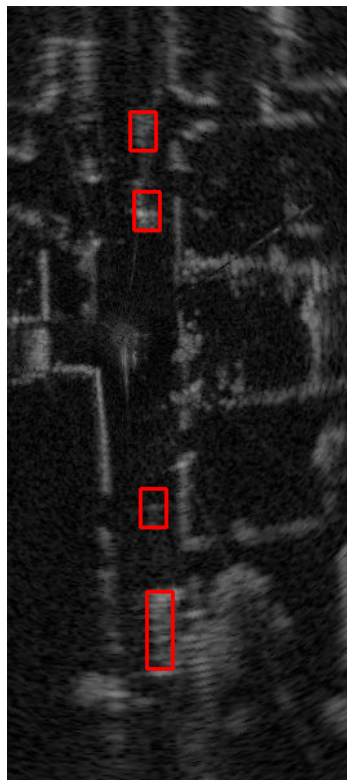
(a) TRL[12]



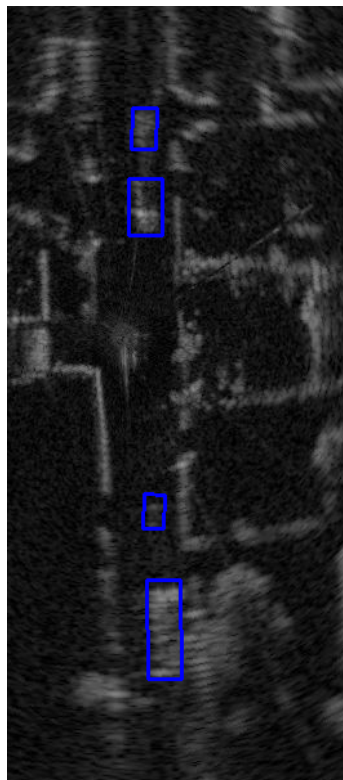
(b) PointPillars[11]



(c) MVDNet[22]



(d) RLANet (ours)



(e) ground-truth[28]



(f) Deformable DETR[41]

Figure 4.2: Samples data from the fog case, and The yellow circles indicate the areas that differ from the ground truth, (a) for radar-only object detection, (b) for lidar-only object detection, and (c) for object detection with the fusion of radar and lidar. (e) represents the ground truth,(f) shows object detection using only camera data trained in Deformable DETR.

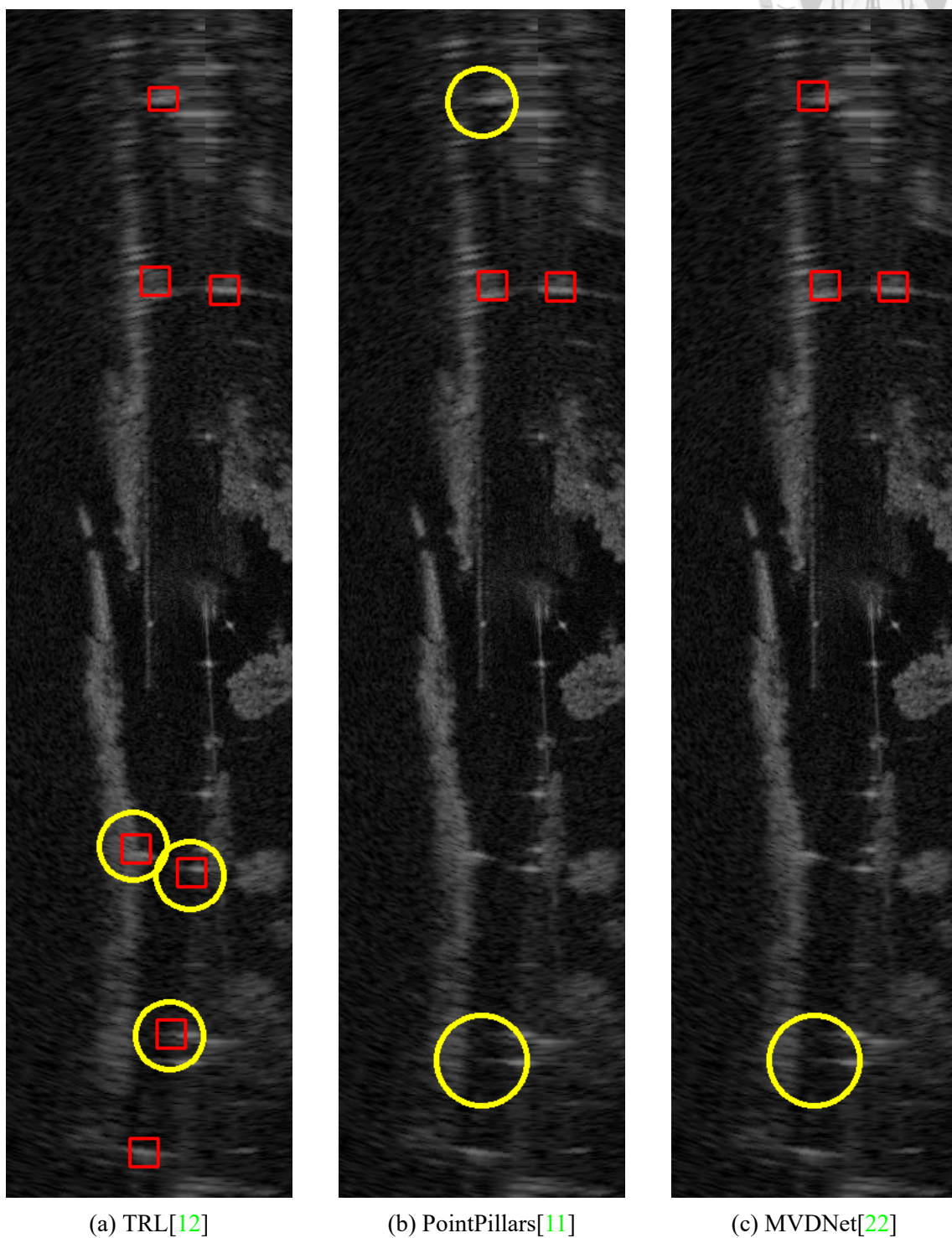
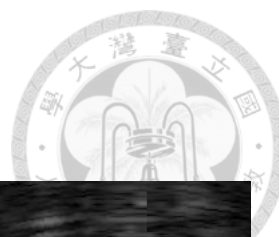
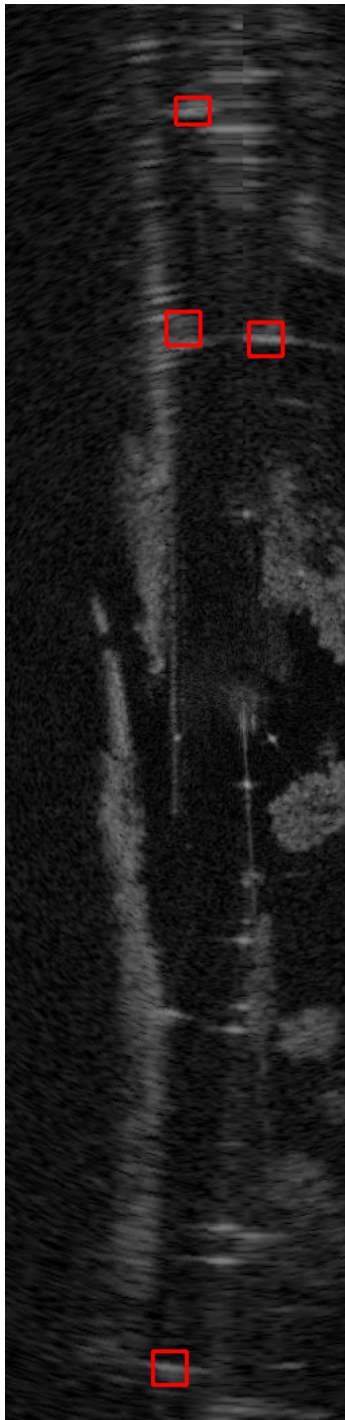
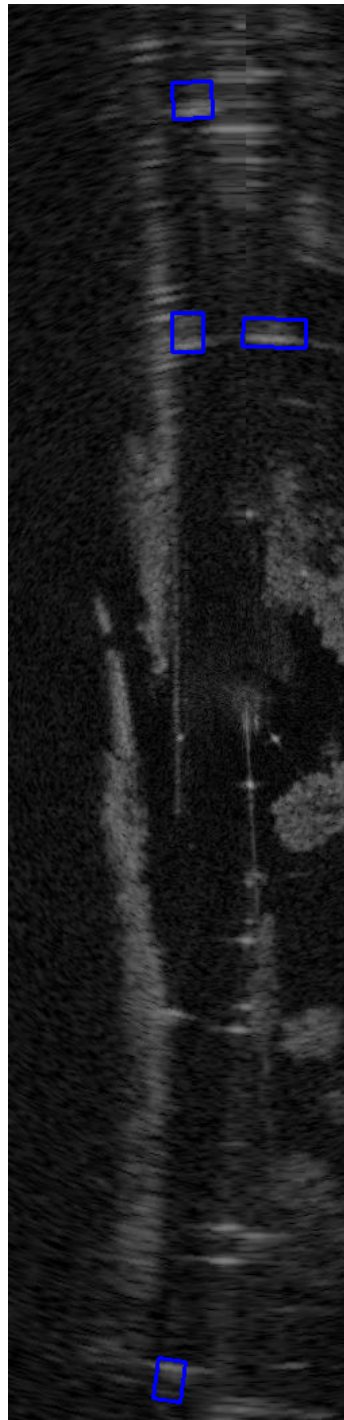


Figure 4.3: Samples data from the sunny case, and The yellow circles indicate the areas that differ from the ground truth, (a) for radar-only object detection, (b) for lidar-only object detection, and (c) for object detection with the fusion of radar and lidar.



(a) RLANet (ours)



(b) ground-truth[28]



(c) Deformable DETR[41]

Figure 4.4: Samples data from the sunny case, and The yellow circles indicate the areas that differ from the ground truth, (b) represents the ground truth, (c) shows object detection using only camera data trained in Deformable DETR.



Chapter 5 Conclusion

In this paper, a novel end-to-end lidar and radar fusion model (RLANet) is proposed. Unlike one-stage and two-stage models that involve extensive handcrafted designs, this model eliminates the need for such manual interventions. Therefore, the performance of the model is independent of hyperparameters, eliminating the need for tedious tuning. Additionally, a Feature Selection Module is introduced to address the limitation of self attention mechanism[33] that initially focus on the entire feature map with equal attention weights. This module enhances the performance of the model. Furthermore, an Associative Feature Fusion Module is designed that fully leverages the features near the query. Unlike conventional attention mechanisms that compute similarity based on isolated query-key pairs, this module considers the features surrounding the query to calculate similarity. Moreover, the attention weights are designed as learnable parameters, and the computationally expensive inner product operation is eliminated, reducing the overall computational requirements. In summary, the model not only obviates the need for hyperparameter tuning but also surpasses the state-of-the-art in radar and lidar fusion, providing more robust performance in adverse weather conditions.

However, the work still has limitations. The conversion of the lidar point cloud into an image can result in a loss of features. In the future, it would be beneficial to explore more suitable preprocessing techniques for lidar point clouds, such as voxeliza-


tion or leveraging the PointNet series, which directly process point clouds. This would theoretically further improve the performance of the model.

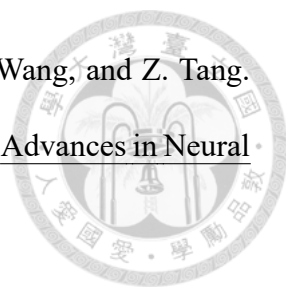


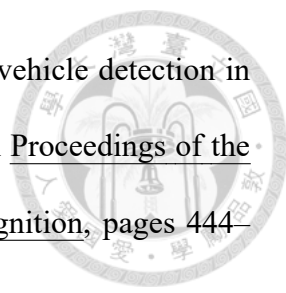


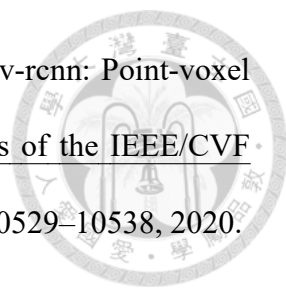
References

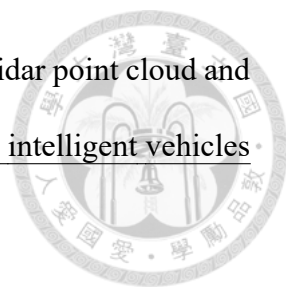
- [1] X. Bai, Z. Hu, X. Zhu, Q. Huang, Y. Chen, H. Fu, and C.-L. Tai. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 1090–1099, 2022.
- [2] D. Barnes, M. Gadd, P. Murcutt, P. Newman, and I. Posner. The oxford radar robotcar dataset: A radar extension to the oxford robotcar dataset. In 2020 IEEE International Conference on Robotics and Automation (ICRA), pages 6433–6438. IEEE, 2020.
- [3] A. Barrera, C. Guindel, J. Beltrán, and F. García. Birdnet+: End-to-end 3d object detection in lidar bird’ s eye view. In 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), pages 1–6. IEEE, 2020.
- [4] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. nuscenes: A multimodal dataset for autonomous driving. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 11621–11631, 2020.
- [5] S.-Y. Chu and M.-S. Lee. Mt-detr: Robust end-to-end multimodal detection with confidence fusion. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pages 5252–5261, 2023.

- 
- [6] O.-R. A. D. O. Committee. Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles, sep 2016.
- [7] L. Dai, H. Liu, H. Tang, Z. Wu, and P. Song. Ao2-detr: Arbitrary-oriented object detection transformer. IEEE Transactions on Circuits and Systems for Video Technology, 2022.
- [8] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun. Yolox: Exceeding yolo series in 2021. arXiv preprint arXiv:2107.08430, 2021.
- [9] R. Girshick. Fast r-cnn. In Proceedings of the IEEE international conference on computer vision, pages 1440–1448, 2015.
- [10] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. L. Waslander. Joint 3d proposal generation and object detection from view aggregation. In 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 1–8. IEEE, 2018.
- [11] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 12697–12705, 2019.
- [12] P. Li, P. Wang, K. Berntorp, and H. Liu. Exploiting temporal relations on radar perception for autonomous driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 17071–17080, 2022.
- [13] Y. Li, A. W. Yu, T. Meng, B. Caine, J. Ngiam, D. Peng, J. Shen, Y. Lu, D. Zhou, Q. V. Le, et al. Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 17182–17191, 2022.

- 
- [14] T. Liang, H. Xie, K. Yu, Z. Xia, Z. Lin, Y. Wang, T. Tang, B. Wang, and Z. Tang. Bevfusion: A simple and robust lidar-camera fusion framework. Advances in Neural Information Processing Systems, 35:10421–10434, 2022.
- [15] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In Proceedings of the IEEE international conference on computer vision, pages 2980–2988, 2017.
- [16] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101, 2017.
- [17] J. Mao, Y. Xue, M. Niu, H. Bai, J. Feng, X. Liang, H. Xu, and C. Xu. Voxel transformer for 3d object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 3164–3173, 2021.
- [18] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. Advances in neural information processing systems, 32, 2019.
- [19] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 652–660, 2017.
- [20] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems, 30, 2017.
- [21] G. Qian, Y. Li, H. Peng, J. Mai, H. Hammoud, M. Elhoseiny, and B. Ghanem. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. Advances in Neural Information Processing Systems, 35:23192–23204, 2022.

- 
- [22] K. Qian, S. Zhu, X. Zhang, and L. E. Li. Robust multimodal vehicle detection in foggy weather using complementary lidar and radar signals. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 444–453, 2021.
- [23] C. Reading, A. Harakeh, J. Chae, and S. L. Waslander. Categorical depth distribution network for monocular 3d object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 8555–8564, 2021.
- [24] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 779–788, 2016.
- [25] J. Redmon and A. Farhadi. Yolo9000: better, faster, stronger. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 7263–7271, 2017.
- [26] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767, 2018.
- [27] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems, 28, 2015.
- [28] M. Sheeny, E. De Pellegrin, S. Mukherjee, A. Ahrabian, S. Wang, and A. Wallace. Radiate: A radar dataset for automotive perception in bad weather. In 2021 IEEE International Conference on Robotics and Automation (ICRA), pages 1–7. IEEE, 2021.

- 
- [29] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 10529–10538, 2020.
- [30] X. Shi, Q. Ye, X. Chen, C. Chen, Z. Chen, and T.-K. Kim. Geometry-based distance decomposition for monocular 3d object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 15172–15181, 2021.
- [31] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 2446–2454, 2020.
- [32] Z. Tian, C. Shen, H. Chen, and T. He. Fcos: Fully convolutional one-stage object detection. In Proceedings of the IEEE/CVF international conference on computer vision, pages 9627–9636, 2019.
- [33] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. Advances in neural information processing systems, 30, 2017.
- [34] T. Wang, X. Zhu, J. Pang, and D. Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 913–922, 2021.
- [35] Y. Wang, V. C. Guizilini, T. Zhang, Y. Wang, H. Zhao, and J. Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In Conference on Robot Learning, pages 180–191. PMLR, 2022.

- 
- [36] Z. Wang, W. Zhan, and M. Tomizuka. Fusing bird' s eye view lidar point cloud and front view camera image for 3d object detection. In 2018 IEEE intelligent vehicles symposium (IV), pages 1–6. IEEE, 2018.
- [37] Z. Yang, Y. Sun, S. Liu, and J. Jia. 3dssd: Point-based 3d single stage object detector. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 11040–11048, 2020.
- [38] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 9759–9768, 2020.
- [39] Y. Zhou, L. Liu, H. Zhao, M. López-Benítez, L. Yu, and Y. Yue. Towards deep radar perception for autonomous driving: Datasets, methods, and challenges. Sensors, 22(11):4208, 2022.
- [40] Y. Zhou and O. Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 4490–4499, 2018.
- [41] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai. Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159, 2020.