Mask-RadarNet: Enhancing Radar Object Detection With Spatio-Temporal Context

Yuzhi Wu, Jun Liu, *Senior Member, IEEE*, Guangfeng Jiang, Weijian Liu, *Senior Member, IEEE*, Danilo Orlando, *Senior Member, IEEE*, and Li Xiao, *Member, IEEE*

***Abstract*—As a cost-effective and robust technology, automotive radar has seen steady improvement during the last years. Radio frequency (RF) images, serving as a radar data format with rich semantic information, have attracted considerable interest in radar object detection. Previous RF-based models heavily rely on convolutional neural networks, leading to the high computational cost. To solve this problem, we propose a model called Mask-RadarNet to fully utilize the hierarchical semantic features from the RF image sequences. Mask-RadarNet exploits the combination of interleaved convolution and attention operations in the encoder. In addition, patch shift is introduced to Mask-RadarNet for efficient spatial-temporal feature learning. By shifting part of patches with a specific mosaic pattern in the temporal dimension, Mask-RadarNet achieves competitive performance while reducing the computational burden of the spatial-temporal modeling. In order to capture the spatial-temporal semantic contextual information, we design the class masking attention module (CMAM) in our encoder. Moreover, a lightweight auxiliary decoder is added to our model to aggregate prior maps generated from the CMAM. Experiments on the CRUW dataset demonstrate that the proposed Mask-RadarNet achieves state-of-the-art performance with relatively lower computational complexity and fewer parameters.**

***Index Terms*—Environment awareness, radar object detection, frequency-modulated continuous-wave radar, convolutional neural network, transformer.**

I. Introduction

LAST decades have witnessed growing interests in environmental perception for safe autonomous driving [1], [2], [3], [4], [5], [6], and object detection is one of the most fundamental task for practical deployment. To better capture the surrounding objects, sensors equipped in autonomous cars are receiving increasing attention. Among the commonly used sensors, millimeter-wave (MMW) frequency-modulated continuous-wave (FMCW) radar has the following unique advantages: 1) the narrow MMW band allows radar signal to penetrate through fog and smoke, which is crucial in extreme weather conditions; 2) FMCW radar has better acquisition capabilities for detecting longer ranges; 3) FMCW radar is robust to lighting while being cheap. However, because of the difficulties in deciphering significant clues for semantic understanding, radar is frequently regarded as a complement sensor for RGB cameras and LiDARs. Comparatively, the RGB images and point cloud data from cameras and LiDARs are relatively easy for human to understand since the semantic information they convey is obvious [7]. For example, Fig. 1 shows some RGB images and their corresponding radio frequency (RF) images which represent the same scene. RF images are 2D representations generated via a series of Fast Fourier Transforms on raw radar signals, which are typically formatted in the radar range-azimuth coordinates. In recent work [8], [9], [10], FMCW radar is merely processed to provide location and speed information for the detected objects without fully exploiting the semantic information. In other words, the development of object detection with FMCW radar is still in its early stages, making it worthwhile to explore further.

Radar data are usually represented in two different formats, i.e., RF images and radar point clouds. Considering that the current 3D radar point clouds are too sparse to detect objects accurately [11], [12], [13], [14], many researchers start to take advantage of RF images [15], [16]. In the field of traditional radar signal processing [17], peak detection algorithms, such as those with a constant false alarm rate (CFAR), are utilized in RF images to determine the object's location. Subsequently, a classifier is employed to identify the object's category [18]. However, conventional radar systems often struggle with the high number of false positives generated by the CFAR algorithm, which significantly degrades detection precision [13]. With the emergence of deep learning, the focus of research is naturally shifted to extract RF image features via neural networks. It has to be pointed out that many labeled data are required for training neural networks. However, it is more difficult to annotate RF images than RGB images due to the abstract semantic information, especially for object detection task. Zhang et al. [19] proposed an

Received 14 March 2024; revised 23 January 2025; accepted 30 October 2025. Date of publication 18 November 2025; date of current version 26 December 2025. This work was supported in part by the National Natural Science Foundation of China under Contract 62471450, Contract 62202442, Contract 62471485, and Contract 62071482; in part by the Natural Science FoundationHubei Province under Grant 2025AFB873; and in part by the USTC Global Cooperation Expansion and Cultivation Fund under Grant MS-A-2025-02-050. The work of Danilo Orlando was supported in part by Italian Ministry of Education and Research (MUR) in the Framework of the FoReLab Project (Departments of Excellence) and in part by the European Union in the NextGenerationEU Plan through Italian Program "Bando PRIN 2022," D.D. 104/2022 (PE7, Project "CIRCE") under Grant H53D23000420006. The Associate Editor for this article was Y. Zhang. *(Corresponding author: Jun Liu.)*

Yuzhi Wu, Jun Liu, Guangfeng Jiang, and Li Xiao are with the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei 230027, China (e-mail: yuzhiwu1105@mail.ustc.edu.cn; junliu@ustc.edu.cn; jgf1998@mail.ustc.edu.cn; xiaoli11@ustc.edu.cn).

Weijian Liu is with Wuhan Electronic Information Institute, Wuhan 430019, China (e-mail: liuvjian@163.com).

Danilo Orlando is with the Dipartimento di Ingegneria dell'Informazione, Universitá di Pisa, 56122 Pisa, Italy (e-mail: danilo.orlando@unipi.it).

Digital Object Identifier 10.1109/TITS.2025.3629140

1558-0016 © 2025 IEEE. All rights reserved, including rights for text and data mining, and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: University of Science & Technology of China. Downloaded on January 26,2026 at 08:17:20 UTC from IEEE Xplore. Restrictions apply.

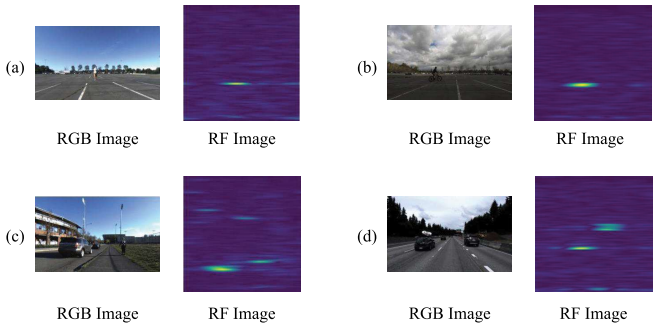


Fig. 1. Examples of RGB images and their corresponding RF images which represent the same scene. RF images are in range-azimuth coordinates.

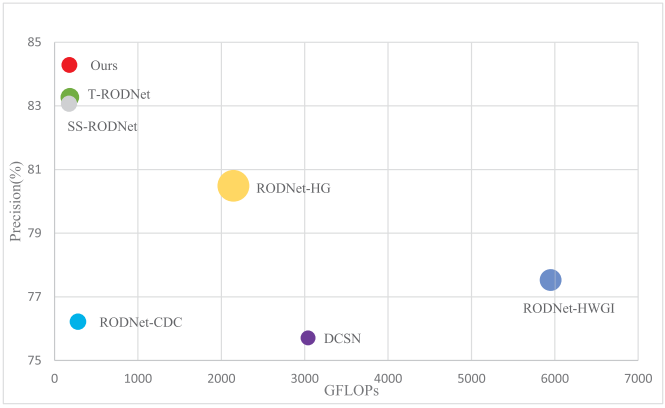


Fig. 2. Comparisons of Mask-RadarNet with other SOTA models on the CRUW dataset. Different models are represented by marks of different colors. Moreover, a smaller mark means a smaller model size.

instance-wise auto-annotation method to build a new radar dataset called RADDet. But the size of this dataset is small. Ouaknine et al. [20] presented a semi-automatic annotation approach and proposed a dataset of synchronized camera and radar recordings with annotations. Recently, Wang et al. [13] developed a cross-modal supervision framework to annotate object labels on RF images automatically with a camera-radar fusion (CRF) strategy, and built a new benchmark for radar object detection task. During the training stage, annotations of object are processed into confidence maps as the ground truth. To test the models, the output is post-processed like [21] to generate final results.

With the open access of radar RF image datasets, much work [13], [22], [23] uses 3D convolutional neural network (CNN) to extract semantic and velocity information from multi-frame RF images. While these models perform well in certain tasks, the extensive 3D convolutions come with a high computational cost, which may be inappropriate for real-time applications. Besides, 3D CNN shows poor performance in extracting global features and it cannot acquire the dependencies between multi-frame radar RF images well. The work in [24] was the first one to introduce the transformer-based model into radar object detection. The model is a U-shaped one containing convolution and attention operations. Although the architecture facilitates the extraction of multiscale features, it overlooks the significance of the spatial-temporal semantic context of the attention maps, leading to some misclassified results.

To solve the issues mentioned above, we propose a novel model called Mask-RadarNet, a 3D transformer for radar object detection. The Mask-RadarNet exploits the combination of interleaved convolution and self-attention operations. The hybrid architecture enables the encoder of Mask-RadarNet to extract local and global features effectively. We utilize a simple but effective method called patch shift [25] for efficient spatial-temporal modeling in the 3D transformer. This attempt enhances spatial-temporal feature learning efficiently for our model. Moreover, we design a class masking attention module (CMAM) in our encoder to capture the spatial-temporal contextual information. Although RF images are non-intuitive compared with RGB images, and much more difficult for human eyes to understand, we still hold the belief that the spatial-temporal semantic context contained in RF image sequences is crucial for radar object detection. With the supplement of the spatial-temporal semantic context, the CMAM enhances the global feature acquired by attention operations. It also generates prior maps for our task and guides the model to update during the training stage. Besides, we add a semantic auxiliary decoder to aggregate prior maps from different stages. As shown in Fig. 2, our model achieves the state-of-the-art (SOTA) performance on the CRUW dataset. In brief, our work has made the following contributions:

1) We propose Mask-RadarNet that is a 3D Transformer combining interleaved convolution and self-attention operations for radar object detection. The proposed Mask-RadarNet achieves significant improvements in object detection performance over previous models on the CRUW dataset.

2) We introduce patch shift in the Mask-RadarNet for efficient spatial-temporal feature learning. Our model achieves competitive performance with other methods while reducing the computational burden of spatial-temporal modeling.

3) We design a specific module called CMAM to capture the spatial-temporal contextual information and enhance the global feature with spatial-temporal semantic context. Besides, we add an auxiliary decoder to generate prior maps during the training stage.

The rest of this article is structured as follows. In Section II, we introduce some related work. Section III describes the design of Mask-RadarNet in detail. The experimental results and ablation studies are included in Section IV. Section V concludes with conclusions.

II. Related Work

A. Radar-Based Perception Methods for Environmental Perception

There are mainly two representations of radar data: one is the dense raw RF images, and the other is the sparse radar point clouds. In Sections II-A, we review related work on perception for autonomous driving from the perspective of two different formats.

1) RF Image: With the advancement in deep learning, a series of research explores neural networks to extract features from raw RF images. Capobianco et al. [26] employed CNN to recognize vehicle categories from range-doppler images.

Authorized licensed use limited to: University of Science & Technology of China. Downloaded on January 26,2026 at 08:17:20 UTC from IEEE Xplore. Restrictions apply.

Angelov et al. [18] considered a modular pipelined framework on raw radar data, and explored three distinct kinds of neural networks including convolution-based ones to classify radar objects. Gao et al. [27] used a modified complex-valued convolutional neural network to enhance radar imaging. Zhang et al. [19] utilized a residual network as the backbone and proposed a dual detection head for more accurate predictions. In order to extract semantic and velocity information from multi-frame radar images, some work was proposed to use 3D convolution. Hazra and Santra [28] employed a model based on 3D CNN to acquire embedding features from radar data with a distance-based triplet-loss similarity metric. Wang et al. [13] proposed a stacked-hourglass model on multiframe RF images to generate predictions. Hsu et al. [22] further adopted dilated convolution in the backbone network to achieve a larger receptive field. The work in [23] used the squeeze-and-excitation network to predict the location and category of the object. T-RODNet [24] introduced a transformer-based model in radar object detection that contains convolutions and attention operations, with the intention of utilizing the ability of both to acquire local and global features simultaneously. SS-RODNet [29] further proposed a lightweight model by pretraining radar spatial-temporal information.

Most of aforementioned approaches focus on leveraging CNN for extracting local features. However, in order to further extract global features, multiple layers of CNNs are required, which increases the computational complexity, especially for multi-frame inputs. While some of the aforementioned work has utilized attention mechanisms to extract global features, they leave out the spatial-temporal semantic context during the encoding stage, leading to some misclassified results.

2) Radar Point Cloud: In the current real-world automotive application, radar suppliers commonly provide radar point clouds for environment awareness. As a lightweight data representation, point clouds provide an intuitive spatial structure of the surroundings. Liu et al. [30] believed that radar points with diverse semantic information rarely belong to the same object, and designed a clustering method based on semantic segmentation. Xiong et al. [31] proposed a contrastive learning method to address the problem of insufficient annotation of radar points, and designed a model that performs well with limited labeled radar points. Kernel density estimation branch is added to the pillar-based backbone for feature encoding in SMURF [32], alleviating the impact of sparsity in radar point clouds. Some work attempts to integrate radar point clouds and corresponding RGB images. RCFusion [33] utilized orthographic feature transform for transforming the image perspective view (PV) features into the bird's-eye-view (BEV) domain, and then fused image BEV features and radar BEV features using interactive attention module. LXL [34] generated radar occupancy grids and predicted image depth distribution maps separately, which both assist in converting image PV features to BEV features, so that the image features can be aligned with radar BEV features. Although radar point clouds have advantages in being a lightweight data representation, they suffer from the inevitable loss of potential information in raw radar tensors during signal processing [15], which may cause the failure in detecting small objects.

In summary, although radar point clouds are a lightweight data representation, peak detection during the post processing inevitably leads to information loss. In addition, radar point clouds are very sparse and often need to be combined with RGB images, which can result in data alignment issues.

B. Transformer-Based Methods for Various Computer Vision Tasks

After being developed in the field of natural language processing (NLP) [35], transformer-based methods have gained popularity for various computer vision tasks following vision transformer (ViT) [36]. The transformer-based methods have produced outstanding results on semantic segmentation [37], [38], object detection [39], [40], video segmentation [41], [42] and other computer vision tasks [43], [44], [45]. Liu et al. [46] adopted the shifted window-based approach in ViT architectures, which greatly enhances performance. Other work followed this approach. Cao et al. [47] built a U-shape transformer-based model that employs the hierarchical transformer architecture with shifted windows as the backbone for feature extraction.

However, in comparison to CNNs, vision transformers still experience the drawback of image-specific inductive bias, leading to inefficiency in extracting local information. Some researchers try to investigate ways to improve the local feature modeling capabilities of ViTs. To model the relationships between tokens at different scales, Xu et al. [48] adopted a hybrid architecture that contains depthwise convolutions and cross-attention operations. Chu et al. [49] built a model upon pyramid vision transformer [50] by combining depth-wise separable convolutions and relative position embedding. Tu et al. [51] proposed a model which involves MBConv-based convolution followed by block-wise attention operations and grid-wise attention operations.

To sum up, the combination of CNNs and attention mechanisms has been proven to have advantages in various computer vision tasks, yet it has not been fully exploited in RF image-based object detection.

C. Semantic Context for Various Computer Vision Tasks

Computer vision tasks require semantic context information to get high-quality results. Chen et al. [52] proposed a module which employs atrous convolution to efficiently broaden the field of view of filters in order to include more context information. They further augmented this module with global average pooling in [53]. Zhao et al. [54] proposed a pyramid pooling module which can aggregate context from different region to leverage global context information. Yu et al. [55] added the global pooling on the top of the U-shape model with the purpose of encoding the global context. Zhang et al. [56] designed a new context encoding module that, by introducing prior information, improves the model's performance. Jain et al. [57] incorporated the semantic context of RGB images into the backbone of a hierarchical transformer model. Zhang et al. [58] explored the impact of global contextual information in semantic segmentation. Jin et al. [59] advocated enhancing pixel representations by combining the image-level and semantic-level contextual information.

Authorized licensed use limited to: University of Science & Technology of China. Downloaded on January 26,2026 at 08:17:20 UTC from IEEE Xplore. Restrictions apply.

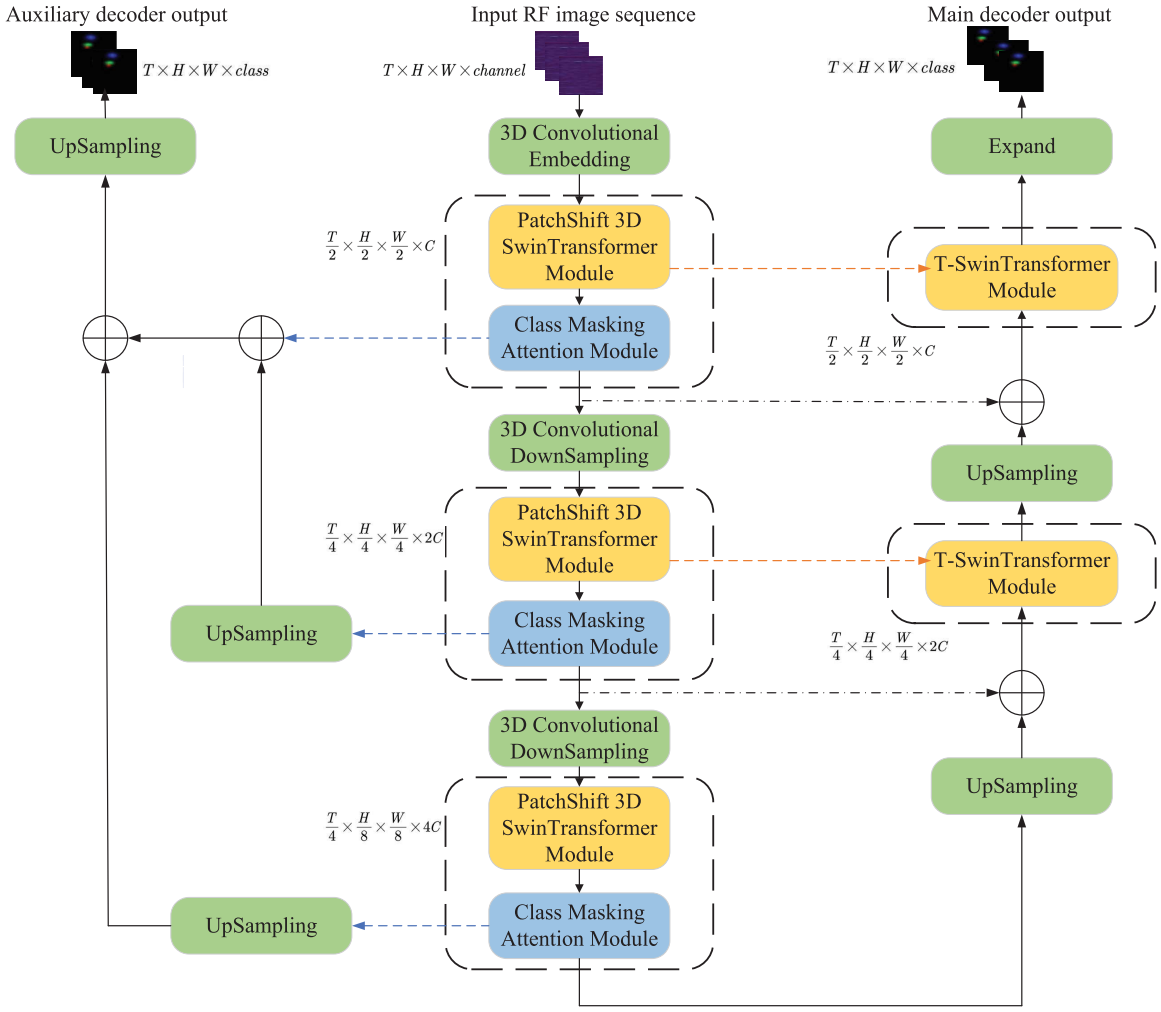


Fig. 3. Overview of proposed Mask-RadarNet. The encoder is in the middle, and the two decoders are on the left and right. The encoder is a hierarchical hybrid structure of convolution and self-attention mechanisms, which consists of the PatchShift 3D SwinTransformer module and the CMAM. The right decoder is the main decoder including the T-SwinTransformer module. The left decoder is the auxiliary decoder which generates the final prior maps. The orange lines represent the movement of query and key features from the PatchShift 3D SwinTransformer module to the main decoder. The blue lines represent the movement of query features from the CMAM to the auxiliary decoder.

So far, the value of semantic context has not been emphasized in RF image-based object detection, mainly because RF images are non-intuitive compared to RGB images. In this work, we believe that the semantic context contained in RF image sequences is crucial for radar object detection.

III. Methodology

A. Task Definition

In this study, we consider the task of radar object detection with continuous multi-frame RF image sequences. Given one RF image sequence denoted as $I \in \mathbb{R}^{T_o \times H_o \times W_o \times channel}$, our model is designed to predict the confidence maps $\hat{\mathcal{C}} \in \mathbb{R}^{T_o \times H_o \times W_o \times class}$. Here, T_o denotes the frames of the original RF image sequence, and H_o, W_o denote the height, width of an RF image, respectively. We treat the real and imaginary values as two channels in an RF image, so *channel* equals two. *class* represents the number of categories we want to detect. After obtaining the confidence map, we employ a post-processing method following [24] to get the final prediction results.

B. Overall

The overall architecture of our Mask-RadarNet is presented in Fig. 3, which maintains one encoder and two different decoders: main decoder and semantic auxiliary decoder. Specifically, the encoder is a hierarchical hybrid structure of convolution and self-attention mechanisms for exploiting both local and global representations. We specially design the PatchShift 3D SwinTransfomer module and the CMAM to model and capture the spatial-temporal contextual information. Symmetrically, the main decoder involves two T-SwinTransformer modules, two upsampling layers and the last expanding layer for mask predictions. It not only adds skip connections between corresponding feature pyramids of the encoder and decoder, but also utilizes cross attention to fuse the features from the encoder and the inherent features from the decoder. Besides, we use a lightweight semantic auxiliary decoder during training to generate prior maps. The network architecture will be described in detail in the following sections.

Authorized licensed use limited to: University of Science & Technology of China. Downloaded on January 26,2026 at 08:17:20 UTC from IEEE Xplore. Restrictions apply.

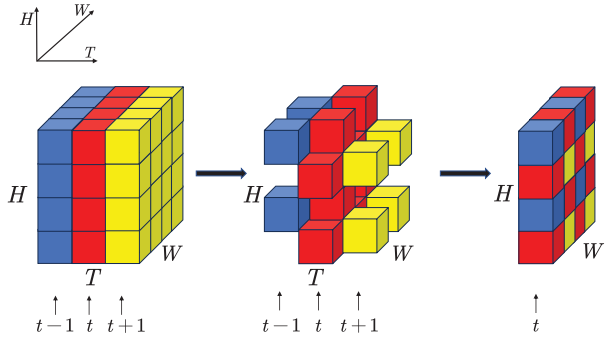


Fig. 4. An example of patch shift for three neighboring frames. The current frame t aggregates information from neighboring frames $t-1$ and $t+1$.

0	1	0	1
-1	0	-1	0
0	1	0	1
-1	0	-1	0

Pattern A

0	1	0	1
-1	2	-1	2
0	1	0	1
-1	2	-1	2

Pattern B

-4	1	2	-4
-1	0	3	-1
-2	-3	4	-2
-4	1	2	-4

Pattern C

Fig. 5. Three typical shift patterns. Pattern A only shifts patches within 3 neighboring frames, while Pattern B has a temporal of 4 and Pattern C has a temporal field of 9.

C. Encoder

1) PatchShift 3D SwinTransformer Module: Considering that the inter-frame details within RF image sequences play an indispensable role in assisting the network to accurately recognize targets, we propose PatchShift 3D SwinTransformer Module for spatial-temporal feature extraction. Transformer model with patch shift operation was first proposed by [25] for action recognition. It was originally designed for RGB image sequences. Our model for the first time introduces it in RF image sequences for efficient spatial-temporal fusion. Generally, patch shift is an effective way for temporal modeling, which shifts the patches of input features along the temporal dimension following specific patterns. Fig. 4 shows an example of patch shift for three neighboring frames, where the symbols H, W, and T denote the height, width, and temporal dimension, respectively, and the blue, red and yellow colors represent the frames $t-1$, t, and $t+1$, respectively. Part of patches in red frame are replaced by patches from blue and yellow frames. This means the current frame t aggregates information from other frames $t-1$ and $t+1$ with a specific pattern. Patch shift can be carried out with different patterns. Fig. 5 depicts some typical shift patterns. The numbers denote the frame indices from which the patches are taken. The symbols "0", "-", "+" represent the current, previous, and next frames, respectively. To cover all patches, we continually apply shift pattern in a sliding window way [46]. After patch shift operation, the spatial features mingle with the temporal feature in a zero-computation way. Hence we can directly exploit volume-based 3D transformer module.

Patch shift operation learns temporal information by moving part of patches from other frames, thus keeping the full channel information of each patch. This means that patch shift is sparse in the spatial domain but dense in the channel domain. Channel shift is also a method for temporal modeling, which is just the opposite to patch shift. It replaces a constant proportion of channels in the current frame with other frames along the temporal dimension. As shown in Fig. 6, part of channels are shifted forward by one frame, while another part of channels are shifted backward by one frame, with the rest remaining unchanged. Fig. 7 illustrates the differences between patch shift and channel shift. We describe a tensor with flattened spatial dimension HW, temporal dimension T and channel dimension C. The green, blue, red, yellow colors represent four successive frames. The features at different channel stamps in one frame are denoted as different shades of the same color. We can see from Fig. 7 that the two shift methods perform shifting operations in orthogonal directions.Previous work [25] has revealed that patch shift and channel shift have a certain amount of complementary to each other, and the ability of temporal modeling can be enhanced by alternating them. We follow this idea and add channel shift to our model as a supplement for patch shift.

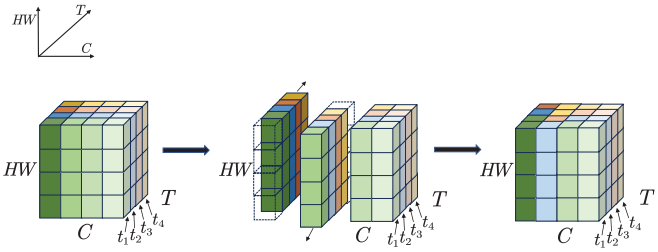


Fig. 6. An example of channel shift for four neighbouring frames. The first channels of frames t_2, t_3 and t_4 are replaced by those of frames t_1, t_2 and t_3. The second channels of frames t_1, t_2 and t_3 are replaced by those of t_2, t_3 and t_4. The rest remains unchanged.

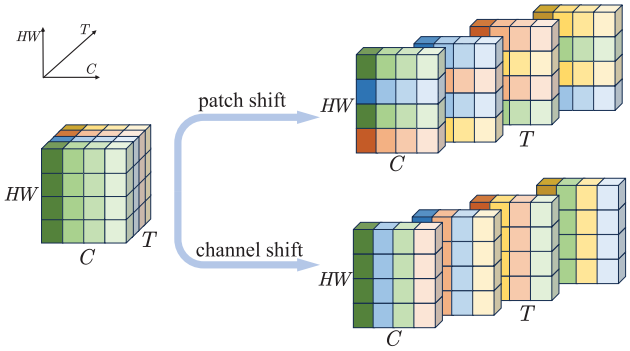


Fig. 7. Illustration of patch shift and channel shift. We can see that they perform shifting operations in orthogonal direction.

Fig. 8 depicts two consecutive PatchShift 3D SwinTransformer blocks. Specifically, given the input feature $\mathcal{X}^{l-1} \in \mathbb{R}^{T\times H\times W\times C}$ from the previous $(l-1)^{th}$ block, we first replace part of the channels of the current frame with neighboring frames following [60], which can be formulated as:

$$\mathcal{X}^{l}_{cs} = \text{ChannelShift}\left(\text{LN}\left(\mathcal{X}^{l-1}\right)\right) \tag{1}$$

where $\mathcal{X}^{l}_{cs}$ represents the output spatial-temporal mixed feature after channel shift, and LN denotes layer normalization. The shift operation enables the integration of spatial information

Authorized licensed use limited to: University of Science & Technology of China. Downloaded on January 26,2026 at 08:17:20 UTC from IEEE Xplore. Restrictions apply.

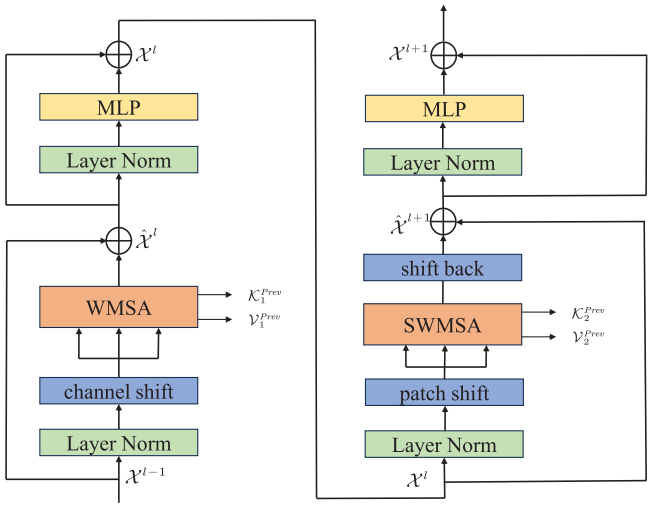


Fig. 8. Overview of two consecutive PatchShift 3D SwinTransformer blocks in the encoder.

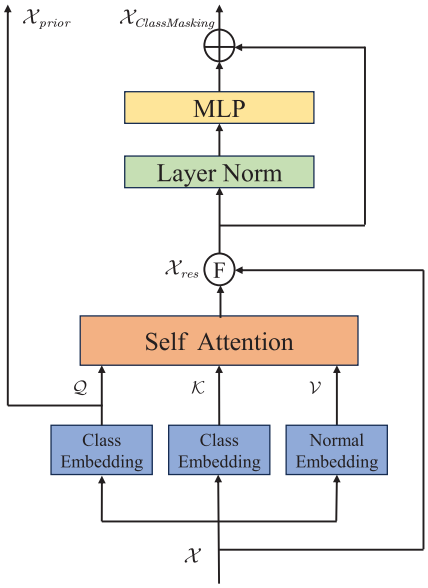


Fig. 9. Overview of CMAM in the encoder.

with temporal information in a zero-computation way, which reduces the computational complexity.

Then we set the size of each 3D window to $P_T \times P_H \times P_W$ and arrange the windows in a non-overlapping way to evenly split feature $\mathcal{X}_{cs}^{l}$ for efficient computation cost. Thus, feature $\mathcal{X}_{cs}^{l}$ is reshaped as $\mathcal{X}_{cs}^{l} \in \mathbb{R}^{N\times P\times C}$, where $N = \frac{THW}{P_T P_H P_W}$ denotes the number of windows, and $P = P_T P_H P_W$ represents the flattened window size. Afterwards we apply the window-based multi-head self attention (WMSA) module [46]. Limiting attention computation in non-overlapping windows can bring the locality of convolution operations whiling saving computational resources. Finally we get the output $\mathcal{X}^{l}$ of the l^{th} block through a feed forward network (FFN) and a shortcut connection like a standard transformer architecture. The process can be formulated as:

$$\hat{\mathcal{X}}^{l} = \text{WMSA}\left(\mathcal{X}_{cs}^{l}\right) + \mathcal{X}^{l-1} \tag{2}$$

$$\mathcal{X}^{l} = \text{FFN}\left(\text{LN}\left(\hat{\mathcal{X}}^{l}\right)\right) + \hat{\mathcal{X}}^{l} \tag{3}$$

where $\hat{\mathcal{X}}^{l}$ represents the output feature after WMSA. It should be noted that the l^{th} block also outputs key feature $\mathcal{K}_1^{Prev}$ and value feature $\mathcal{V}_1^{Prev}$ from the WMSA module preparing for feature fusion in decoders.

After obtaining $\mathcal{X}^{l}$ from the l^{th} block, we first shift the patches of each frame along the temporal dimension with the specific pattern as:

$$\mathcal{X}_{ps}^{l+1} = \text{PatchShift}\left(\text{LN}\left(\mathcal{X}^{l}\right)\right) \tag{4}$$

where $\mathcal{X}_{ps}^{l+1}$ denotes the output feature after patch shift. Previous work [25] has revealed that patch shift and channel shift are complementary to each other, so we adopt patch shift in the $(l+1)^{th}$ block and channel shift in the l^{th} block in order to enhance the ability of temporal modeling.

Then we exploit the shifted window multi-head self-attention (SWMSA) module for cross-window connections. After the SWMSA, we follow [25] and shift patches from different frames back to their original locations to keep the frame structure complete. Finally the output feature map $\mathcal{X}^{l+1}$ of the $(l+1)^{th}$ block is generated by a FFN and a shortcut connection. The process can be formulated as:

$$\hat{\mathcal{X}}_{ps}^{l+1} = \text{SWMSA}\left(\mathcal{X}_{ps}^{l+1}\right) \tag{5}$$

$$\hat{\mathcal{X}}^{l+1} = \text{ShiftBack}\left(\hat{\mathcal{X}}_{ps}^{l+1}\right) + \mathcal{X}^{l} \tag{6}$$

$$\mathcal{X}^{l+1} = \text{FFN}\left(\text{LN}\left(\hat{\mathcal{X}}^{l+1}\right)\right) + \hat{\mathcal{X}}^{l+1} \tag{7}$$

where $\hat{\mathcal{X}}_{ps}^{l+1}$ and $\hat{\mathcal{X}}^{l+1}$ denote the output features after the SWMSA and shift back, respectively. Note that the $(l+1)^{th}$ block also outputs the key feature $\mathcal{K}_2^{Prev}$ and value feature $\mathcal{V}_2^{Prev}$ preparing for feature fusion in decoders. By using shift operations, we extract the spatial-temporal feature in RF image sequences without high computation cost.

2) Class Masking Attention Module (CMAM): As illustrated in Fig. 9, the CMAM is proposed to capture the spatial-temporal contextual information from the perspective of the entire RF image sequence and generate enhanced feature maps with class-dependent semantic context information. It is designed to follow the PatchShift 3D SwinTransformer module at each stage of our model.

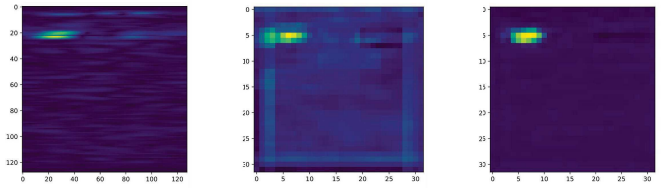
Given the output feature $\mathcal{X} \in \mathbb{R}^{T\times H\times W\times C}$ from previous 3D SwinTransformer module, we first utilize a class embedding layer to query feature $\mathcal{Q}$ and key feature $\mathcal{K}$. Class embedding layer is a linear layer that converts the channel C to *class*, where *class* denotes the number of categories. This operation means projecting the feature $\mathcal{X}$ into semantic space.

$$\mathcal{Q} = \text{ClassEmbedding}\left(\mathcal{X}\right) \tag{8}$$

$$\mathcal{K} = \text{ClassEmbedding}\left(\mathcal{X}\right) \tag{9}$$

where $\mathcal{Q}$ and $\mathcal{K}$ are both tensors of size $T \times H \times W \times class$. This strategy is inspired by some pioneering work [57], [58]. Our intuition is: the output $\mathcal{Q}$ after class embedding layer contains class-dependent RF image semantic information to some extent, which can serve as the prior representation of current stage. Then $\mathcal{Q}$ is sent to auxiliary decoder for further ground truth supervision.

Authorized licensed use limited to: University of Science & Technology of China. Downloaded on January 26,2026 at 08:17:20 UTC from IEEE Xplore. Restrictions apply.



(a) input RF image (b) feature map before CMAM (c) feature map after CMAM

Fig. 10. Visualization of the feature maps before and after the CMAM module. The input RF image has the size of 128×128. We select the feature maps with the size of 32×32.

We also adopt a normal embedding layer to get value feature $\mathcal{V}$. Normal embedding layer is a linear layer which can convert the channel C to the embedding dimension C, playing the same role as that in standard transformer architecture.

$$\mathcal{V} = \text{NormalEmbedding}\left(\mathcal{X}\right) \tag{10}$$

where $\mathcal{V}$ is a tensor of size $T \times H \times W \times C$.

Next we perform reshape operations and calculate the similarities between $\mathcal{Q}$ and $\mathcal{K}$:

$$\mathbf{Q} = \text{Reshape1}\left(\mathcal{Q}\right) \tag{11}$$

$$\mathbf{K} = \text{Reshape1}\left(\mathcal{K}\right) \tag{12}$$

$$\mathbf{S} = \text{Softmax}\left(\mathbf{Q} \otimes \mathbf{K}\right) \tag{13}$$

where Reshape1 is used to transform tensors $\mathcal{Q}$ and $\mathcal{K}$ to matrices $\mathbf{Q}$ and $\mathbf{K}$ of size $THW \times class$, respectively, $\otimes$ stands for matrix multiplication, and $\mathbf{S}$ is the similarity score between $\mathcal{Q}$ and $\mathcal{K}$. Considering that $\mathcal{Q}$ serves as the prior representation of current stage and is indirectly supervised by ground truth in auxiliary decoder, the score values in $\mathbf{S}$ contain semantic context information, which guide $\mathcal{V}$ to update as:

$$\mathbf{V} = \text{Reshape2}\left(\mathcal{V}\right) \tag{14}$$

$$\mathcal{R} = \text{Reshape3}\left(\mathbf{S} \otimes \mathbf{V}\right) \tag{15}$$

where Reshape2 is used to make $\mathcal{V}$ a matrix of size $THW \times C$ and Reshape3 is used to make $\mathcal{R}$ a tensor of size $T \times H \times W \times C$.

To mitigate the vanishing gradient problem, we add a shortcut connection and multiply $\mathcal{R}$ with a learnable scalar constant β additionally for smooth finetuning. Finally we follow the traditional transformer architecture and get the final feature maps $\mathcal{X}_{ClassMasking}$ through a FFN and a shortcut connection. The process can be formulated as:

$$\mathcal{X}_{res} = \beta\mathcal{R} + \mathcal{X} \tag{16}$$

$$\mathcal{X}_{ClassMasking} = \text{FFN}\left(\text{LN}\left(\mathcal{X}_{res}\right)\right) + \mathcal{X}_{res} \tag{17}$$

where $\mathcal{X}_{res}$ is the output after attention operation.

The output $\mathcal{X}_{ClassMasking}$ contains more spatial-temporal contextual information compared with the input $\mathcal{X}$. As shown in Fig.10, the output feature map after the CMAM module focuses more on object areas. The results of the subsequent ablation experiments reveal that the CMAM enhances the performance of our model, particularly on average precision.

D. Decoder

In order to integrate the features and prior maps obtained from different encoder stages respectively, we utilize two different decoders.

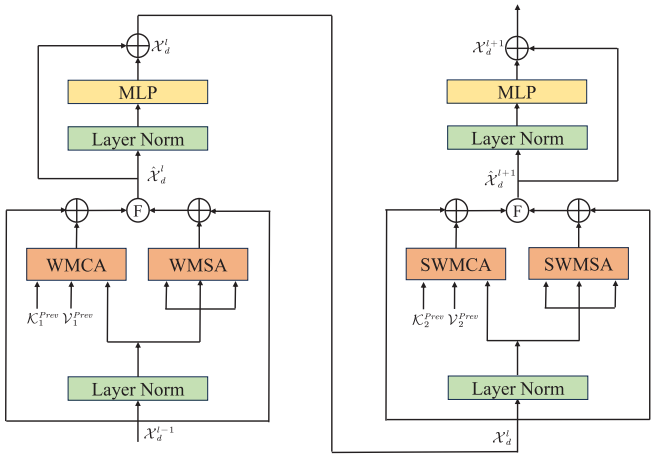


Fig. 11. Overview of two consecutive T-SwinTransformer blocks in the main decoder.

1) Main Decoder: We employ the T-SwinTransformer blocks from [24] as the main decoder in our work. Two consecutive T-SwinTransformer blocks are shown in Fig. 11. Inspired by the original transformer model [35] in NLP, we believe that better prediction results can be obtained by integrating the feature from the encoder into the intrinsic feature of the decoder via cross attention operations.

Specifically, given the features $\mathcal{K}_1^{Prev}$, $\mathcal{V}_1^{Prev}$ from the corresponding PatchShift 3D SwinTransformer module and the inherent feature $\mathcal{X}_d^{l-1}$ from the previous $(l-1)^{th}$ decoder block, we first process them with the WMSA and the WMCA for cost-efficient computation:

$$\mathcal{SA}_l = \text{WMSA}\left(\text{LN}\left(\mathcal{X}_d^{l-1}\right)\right) + \mathcal{X}_d^{l-1} \tag{18}$$

$$\mathcal{CA}_l = \text{WMCA}\left(\text{LN}\left(\mathcal{X}_d^{l-1}\right), \mathcal{K}_1^{Prev}, \mathcal{V}_1^{Prev}\right) + \mathcal{X}_d^{l-1} \tag{19}$$

where WMCA represents window based multi-head cross-attention using regular window partitioning configurations, $\mathcal{SA}_l$ denotes the inherent feature from the main decoder, and $\mathcal{CA}_l$ represents the feature obtained by interacting with the encoder. Then $\mathcal{SA}_l$ and $\mathcal{CA}_l$ are fused as:

$$\hat{\mathcal{X}}_d^l = \gamma\mathcal{CA}_l + (1-\gamma)\,\mathcal{SA}_l \tag{20}$$

where γ stands for a scaling factor that can be learned to compare the significance of the two outputs. After a FFN and a residual structure, the output feature map $\mathcal{X}_d^l$ of the l^{th} block is obtained as:

$$\mathcal{X}_d^l = \text{FFN}\left(\text{LN}\left(\hat{\mathcal{X}}_d^l\right)\right) + \hat{\mathcal{X}}_d^l \tag{21}$$

Next, we repeat the whole procedures above and shift windows during attention operation for cross-window connections:

$$\mathcal{SA}_{l+1} = \text{SWMSA}\left(\text{LN}\left(\mathcal{X}_d^l\right)\right) + \mathcal{X}_d^l \tag{22}$$

$$\mathcal{CA}_{l+1} = \text{SWMCA}\left(\text{LN}\left(\mathcal{X}_d^l\right), \mathcal{K}_2^{Prev}, \mathcal{V}_2^{Prev}\right) + \mathcal{X}_d^l \tag{23}$$

$$\hat{\mathcal{X}}_d^{l+1} = \gamma\mathcal{CA}_{l+1} + (1-\gamma)\,\mathcal{SA}_{l+1} \tag{24}$$

$$\mathcal{X}_d^{l+1} = \text{FFN}\left(\text{LN}\left(\hat{\mathcal{X}}_d^{l+1}\right)\right) + \hat{\mathcal{X}}_d^{l+1} \tag{25}$$

where SWMCA represents window based multi-head cross-attention using shifted window partitioning configurations. Overall, the main decoder fuses encoder and decoder features in a learnable way in the T-SwinTransformer blocks.

Authorized licensed use limited to: University of Science & Technology of China. Downloaded on January 26,2026 at 08:17:20 UTC from IEEE Xplore. Restrictions apply.

2) Auxiliary Decoder: During the training stage, a lightweight FPN-like semantic decoder is used to provide ground truth supervision to the prior maps generated from the CMAM at each stage of the encoder. Considering that all prior maps from different stage have the same channel dimension of *class*, we aggregate them only with some upsampling and summation operations. More details are displayed in Fig. 3.

E. Loss Function

In this paper, we utilize the auxiliary loss function to supervise the training of our proposed method. After passing through the network, confidence maps (ConfMaps) $\hat{\mathcal{C}} \in \mathbb{R}^{T_o \times H_o \times W_o \times class}$ and prior maps $\hat{\mathcal{P}} \in \mathbb{R}^{T_o \times H_o \times W_o \times class}$ are predicted from main decoder and auxiliary decoder, respectively. We use binary cross entropy loss to supervise the output of the main decoder, and the main loss function is defined as:

$$l_{main} = -\sum_{class}\sum_{i,j}\left\{\mathcal{GT}_{i,j}^{class}\log\hat{\mathcal{C}}_{i,j}^{class} + \left(1-\mathcal{GT}_{i,j}^{class}\right)\log\left(1-\hat{\mathcal{C}}_{i,j}^{class}\right)\right\} \tag{26}$$

where $\mathcal{GT}_{i,j}^{class}$ indicates the ground truth generated by CRF at coordinate (i, j) for category label *class*, and $\hat{\mathcal{C}}_{i,j}^{class}$ indicates the predictions generated by the main decoder at coordinate (i, j) for category label *class*.

Then we add a specific auxiliary loss function to supervise the output of auxiliary decoder. The auxiliary loss function is consistent with the main loss function, which can be defined as:

$$l_{aux} = -\sum_{class}\sum_{i,j}\left\{\mathcal{GT}_{i,j}^{class}\log\hat{\mathcal{P}}_{i,j}^{class} + \left(1-\mathcal{GT}_{i,j}^{class}\right)\log\left(1-\hat{\mathcal{P}}_{i,j}^{class}\right)\right\} \tag{27}$$

where $\hat{\mathcal{P}}_{i,j}^{class}$ indicates the prior maps generated by the auxiliary decoder at coordinate (i, j) for category label *class*.

Furthermore, we use the parameter α to balance the weight of the main loss and auxiliary loss, i.e.,

$$l = l_{main} + \alpha l_{aux} \tag{28}$$

It should be noted that we only use the auxiliary loss in the training phase.

IV. Experiments

In this section, we present the experimental evaluation of our model. Firstly we describe the dataset and the evaluation metric that we utilize. Then we give details concerning the experiments. Next we compare our model with the SOTA and analyse the results quantitatively and qualitatively. Finally we perform some ablation studies of Mask-RadarNet.

A. Dataset

We train our Mask-RadarNet with the training data of the CRUW dataset [13]. The CRUW dataset was collected by the University of Washington, containing approximately 400 K synchronized camera-radar frames from various driving scenarios such as parking lots, campus roads, city streets and highway. These data were acquired by a sensor platform including a pair of stereo cameras and two 77 GHz FMCW MMW radar antenna arrays at 30 FPS, and the high frame rate makes the CRUW dataset being appropriate for evaluating the temporal models. In this dataset, RGB images captured by the stereo cameras have a resolution of 1440×864, and radar data is processed and presented as Range-Azimuth (RA) heatmaps, depicting a bird-eye-view of the scene seen from the ego-vehicle. RA heatmaps can be described as images with a resolution of 128×128 and the intensity depicts the magnitude of the RF signal. The cross-modal supervision framework in [13] labels the collected objects with camera-radar locators, which makes full use of FMCW radar and offers an appropriate capability for range estimation free from any systematic bias. Generally, there are around 2.6×10^5 objects in the CRUW dataset, 92 percent of which are utilized for training and 8 percent for testing. Besides, this dataset contains some vision-fail scenarios, allowing the model to be tested in extreme environments.

B. Evaluation Metrics

To evaluate model on the CRUW dataset, a new metric called object location similarity (OLS) [13] is defined. This method depicts the correlation between two point-based detections while taking into account their distance, classes, and scale characteristics. Specifically, the OLS can be written as:

$$\text{OLS} = \exp\left\{\frac{-d^2}{2\left(S \cdot K_{cls}\right)^2}\right\} \tag{29}$$

where d is the distance between the two points in an RF image; S is the object distance from sensors, indicating object scale information; and K_{cls} is a per-class constant representing the error tolerance for class *cls*, which can be determined by the object average size of the corresponding class.

The evaluation approach employed in our work is consistent with that of [24]. Here is a synopsis of the procedure:

1) Get all the 8-neighbor peaks in all *class* channels in ConfMaps within the 3×3 window as a peak set P.

2) Pick the peak $p^* \in P$ with the highest confidence score, and remove it from the set P to the final peak set P^*.

3) Calculate the OLS with each of the rest peaks $p_i \in P$. If the OLS between p^* and p_i is greater than a threshold, we remove p_i from the set P.

4) Repeat steps 2 and 3 until the set P becomes empty.

The whole procedure is similar to previous work for pose estimation [61]. Finally, average precision (AP) and average recall (AR) are calculated through the variation of OLS threshold between 0.5 to 0.9 with steps of 0.05. The AP and AR are the main evaluation metrics for object detection.

C. Implementation Details

We run all experiments on Python 3.8, PyTorch 1.10.1 and Ubuntu 18.04. All training procedures have been performed on RTX 3080 GPU. With the purpose of comparing the performance of every model, we split the 40 sequences in the CRUW dataset into two parts: 36 sequences for training and 4 sequences for testing, which is the same in [24]. We input 16 consecutive RF frames at one time and treat the real and imaginary values as two channels in an RF image. So the input

Authorized licensed use limited to: University of Science & Technology of China. Downloaded on January 26,2026 at 08:17:20 UTC from IEEE Xplore. Restrictions apply.

TABLE I
RESULTS OF DIFFERENT MODELS ON THE CRUW DATASET

Model	All		Pedestrian		Cyclist		Car		GFLOPs	Parameter(M)	Time(H)
	AP(%)	AR(%)	AP(%)	AR(%)	AP(%)	AR(%)	AP(%)	AR(%)			
RODNet-CDC [13]	76.33	79.28	77.11	79.64	69.39	70.02	82.91	89.13	280.03	34.52	**9**
RODNet-HG [13]	79.43	83.59	78.90	83.81	76.69	78.85	83.36	88.55	2144.86	129.19	65
RODNet-HGWI [13]	78.06	81.07	79.47	81.85	70.35	71.40	84.39	90.05	5949.68	61.29	330
DCSN [22]	75.30	79.92	76.70	81.50	66.78	69.04	82.56	89.52	3039.89	**28.10**	204
T-RODNet [24]	83.27	86.98	82.19	85.41	82.28	84.30	**86.22**	**92.53**	182.53	44.31	15
SS-RODNet [29]	83.07	86.43	81.37	84.61	83.34	85.11	85.55	90.86	**172.80**	33.10	15
Mask-RadarNet (Ours)	**84.29**	**87.36**	**82.74**	**85.80**	**85.06**	**86.67**	85.96	90.66	176.91	32.12	14

size is $2 \times 16 \times 128 \times 128$. For the encoder, we set the size of convolutional kernels as $9 \times 5 \times 5$ for 3D Convolutional Embedding and 3D Convolutional DownSampling. The work in [62] demonstrates that employing the larger convolutional kernel rather than a series of small kernels might be a better option. For the PatchShift 3D SwinTransformer module, we carry out patch shift with Pattern C in Fig. 5 and apply channel shift with a shift ratio of 0.25, identical to that described in [60]. Furthermore, we use a 3D window of $4 \times 4 \times 4$ for window partition. The number of heads of multi-head self-attention used in different encoder stages is [2, 4, 8]. The corresponding part in the main decoder has the same settings. For the loss function, we empirically set the auxiliary loss weight α as 0.4. The Adam optimizer is utilized for optimization, with the starting learning rate set to 0.0001.

D. Comparisons With SOTA

We perform some experimental analysis of several previous algorithms on the CRUW dataset, including the SOTA T-RODNet [24]. All models are tested under the same conditions without using any data enhancement or pretrained models. Aside from AP and AR, we compare models in terms of model efficiency, where we adopt two efficiency-related evaluation metrics, i.e., the size of model parameters and GFLOPs. We also document the time required for model training.

1) Quantitative Results: To facilitate a qualitative comparison, the numerical results on the CRUW dataset are presented in Table I, revealing that the Mask-RadarNet outperforms other models in general. The AP and AR of Mask-RadarNet in all categories are 1.02% and 0.38%, respectively, higher than the T-RODNet (SOTA). Notably, Mask-RadarNet significantly improves detection performance on small objects such as pedestrians and cyclists. In addition, its GFLOPs and parameters are less than those of the T-RODNet. It means that the Mask-RadarNet achieves stronger detection results with lower computational complexity. Compared to previous convolution-based temporal modeling methods utilized in radar object detection, such as temporal deformable convolution network in [13] and dimensional apart module in [24], the shift operation we employ can significantly reduce the computation cost. As a result, our model exhibits lower computational complexity. In addition, the reason for stronger detection results lies in the fact that the CMAM we design is capable of merging semantic spatial-temporal context in the feature map acquired by the encoder and generating semantic prior maps. Simultaneously, our auxiliary decoder is capable of aggregating the semantic prior maps generated by the CMAM at various stages. The final semantic prior map, produced after aggregation, is supervised by the ground truth, resulting in the auxiliary loss. Our subsequent ablation experiments demonstrate that the model performance can be enhanced by incorporating auxiliary loss during training.

2) Qualitative Results: To make a qualitative comparison, Fig. 12 shows some visual examples of different models. It is evident that the Mask-RadarNet is able to generate better predictions than other methods. We can observe that the Mask-RadarNet predicts the location and the category of objects accurately, while some other models can predict exact location but mix up categories, such as (1st row, 5th col), (1st row, 8th col) and (6th row, 9th col) in Fig. 12. Our Mask-RadarNet has better ability of class-dependent semantic feature modeling, thus improving detection performance. Overall, the predictions from our Mask-RadarNet resemble the ground truth for the data.

E. Ablation Studies

In this section, we perform some ablation studies of our Mask-RadarNet on the CRUW dataset. It should be noted that all experiments are conducted under identical conditions to achieve fair comparisons.

1) The Effectiveness of Patch Shift: To evaluate patch shift operation, we conduct experiments with different shift pattern settings. Note that all experiments in this section adopt the CMAM and set the auxiliary loss weight α as 0.4. We only change patch shift patterns and remain other components unchanged. First we remove all shift operations. Table II indicates that the Mask-RadarNet without shift operation performs worse than any Mask-RadarNet with shift operation, which demonstrates significance of patch shift. In other words, the Mask-RadarNet profits from the shift operation to efficiently learn the spatial-temporal feature. Then we test different kinds of patch shift patterns in Fig. 5. Pattern A only shifts patches within 3 neighboring frames, while Pattern B has a temporal of 4 and Pattern C has a temporal field of 9. Table II shows

Authorized licensed use limited to: University of Science & Technology of China. Downloaded on January 26,2026 at 08:17:20 UTC from IEEE Xplore. Restrictions apply.

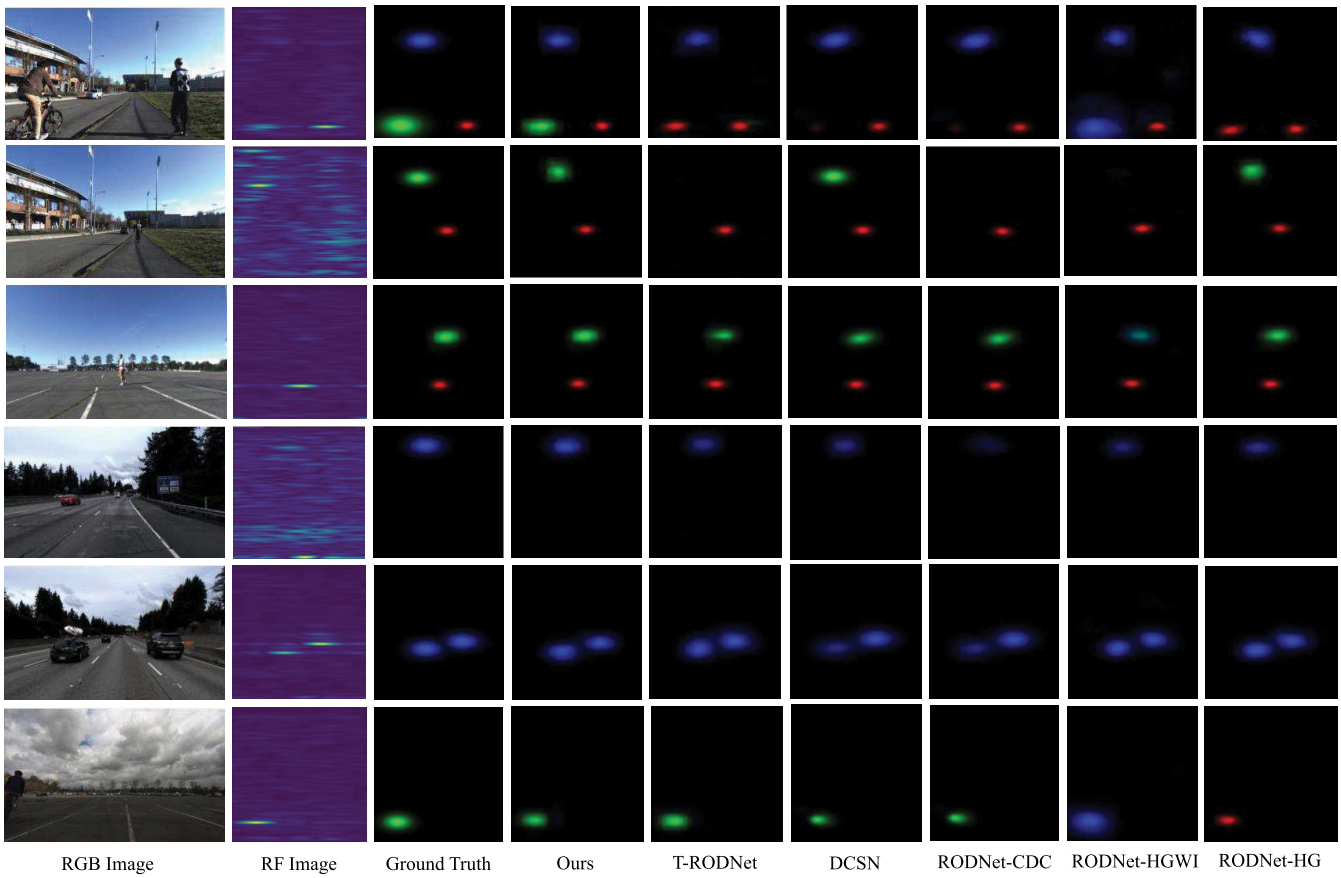


Fig. 12. Visual comparison with other models on the CRUW dataset. Mask-RadarNet outperforms others in all scenarios. The colors red, blue, and green correspond to different categories of objects: pedestrian, car, and cyclist, respectively.

TABLE II
RESULTS OF ABLATION EXPERIMENTS FOR PATCHSHIFT

Temporal Modeling Methods		All		GFLOPs
		AP(%)	AR(%)	
PatchShift	No shift	81.35	85.44	**176.91**
	Pattern A	82.06	85.83	**176.91**
	Pattern B	82.20	85.97	**176.91**
	Pattern C	**84.29**	**87.36**	**176.91**
Dimensional apart module		83.69	87.22	239.16
Temporal deformable convolution		82.84	86.59	290.86

TABLE III
RESULTS OF ABLATION EXPERIMENTS FOR CMAM

Module	AP(%)	AR(%)
None	81.69	85.92
CMAM	**84.29**	**87.36**
Traditional Transformer Module	81.72	85.59

that the performance of Mask-RadarNet steadily improves as the temporal field expands. Besides, we have conducted additional experiments to compare patch shift operation with other temporal modeling methods. We have replaced patch shift operation with other convolution-based temporal modeling methods, such as dimensional apart module in [24] and temporal deformable convolution [13]. From Table II we can see that patch shift operation in Pattern C outperforms other temporal modeling methods. Notably, compared to convolution-based methods for temporal modeling, patch shift operation does not require additional computation cost, which is the main reason why Mask-RadarNet has lower GFLOPs.

2) The Effectiveness of CMAM: We conduct three groups of experiments to explore the impact of CMAM on our model. In the first group, we remove the CMAM module while remaining other components unchanged. In the second group, we keep the CMAM module. In the last group, the CMAM module is replaced with the traditional transformer encoder module [36]. The experimental results are presented in Table III. To facilitate comparisons, we utilize the AP and AR of all categories as indicators. It is observed that the AP and AR obtained by the Mask-RadarNet without the CMAM decrease remarkably. And the detection results have not been significantly improved when the CMAM module is substituted by the standard transformer encoder module. This reveals that the simple attention operations cannot provide the effectiveness of the CMAM module, and the CMAM module can better capture the spatial-temporal semantic context.

Authorized licensed use limited to: University of Science & Technology of China. Downloaded on January 26,2026 at 08:17:20 UTC from IEEE Xplore. Restrictions apply.

TABLE IV
RESULTS OF DIFFERENT AUXILIARY LOSS WEIGHT

Auxiliary Loss Weight α	AP(%)	AR(%)
$\alpha = 0$	81.71	85.18
$\alpha = 0.1$	82.34	86.13
$\alpha = 0.2$	82.94	86.00
$\alpha = 0.3$	82.89	86.82
$\alpha = 0.4$	**84.29**	87.36
$\alpha = 0.5$	83.65	87.11
$\alpha = 0.6$	83.44	**87.53**
$\alpha = 0.7$	82.73	86.47
$\alpha = 0.8$	82.32	86.23
$\alpha = 0.9$	82.36	87.03

3) The Effectiveness of Auxiliary Loss Weight: It is essential to set an appropriate loss weight in the auxiliary decoder, so that the model can regard the spatial-temporal semantic context as a supplemental signal rather than the main prediction. We conduct several experiments with setting the auxiliary loss weight α between 0 and 1. The numerical results are shown in Table IV. To facilitate comparisons, we utilize the AP and AR of all categories as indicators. In general, the introduced auxiliary loss aids in optimizing the training process while not influencing the inference process. We can comprehend auxiliary loss from the perspective of gradient descent. When the model incorporates an auxiliary branch, the gradient of the parameters in the model originates from both the main branch and the auxiliary branch. The auxiliary loss utilized in our Mask-RadarNet is aligned with the main loss, ensuring that the gradient it provides aligns with the direction of the main branch, which benefits the training of the main branch. We can observe from Table IV that $\alpha = 0.4$ yields the best performance which achieves a good balance between modeling feature context and semantic context in RF image sequence.

F. Robustness Testing

To test the performance of our model under different conditions, we have added Gaussian noise with a mean of 0 and variance of 1 to the RF image sequences in the test set of CRUW dataset. Specifically, the test set is regarded as the original signal, and Gaussian noise serves as the noise signal, simulating the random disturbances that may occur in real-world scenarios. The signal-to-noise ratio (SNR), which is a crucial metric for evaluating the quality of the signal in the presence of noise, is set in the range of $[-10, 20]$ dB. We train the models on the training set without adding any noise, and conduct testing on the test set with the aforementioned Gaussian noise added. The results are shown in AP-SNR curve in Fig. 13. It can be observed that the AP of all models exhibits a downward trend as the SNR decreases, which is due to the added Gaussian noise. However, the decline in AP for Mask-RadarNet is slower compared to the other networks. Notably, the performance of our model on small objects such as pedestrians and cyclists remains superior to other models under different SNR conditions. The reason behind this can

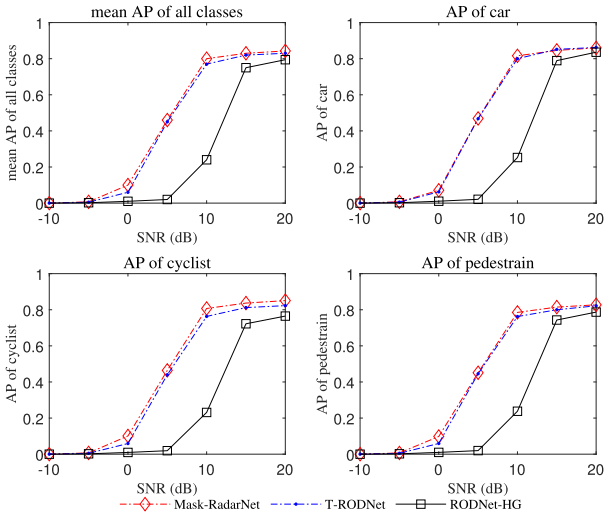


Fig. 13. Comparisons of different models on different SNR.

be attributed to the incorporation of the semantic context in attention mechanism, which enables the model to allocate more focus to the significant regions of the input RF image sequences that are the most relevant for accurate detection and classification. By doing so, Mask-RadarNet effectively reduces the impact of noise interference.

G. Fusion With RGB Images

To integrate our model with RGB images, we have applied polar aligned attention (PAA) from EchoFusion [14]. Namely, we input RF image sequences represented in range-azimuth coordinates into the encoder of Mask-RadarNet, and output the semantically enhanced RF image features $\mathcal{X}_{radar} = \left[F_r^1,\ F_r^2,\ \ldots,\ F_r^T\right] \in \mathbb{R}^{T\times R\times A\times C}$. Here, T, R, A and C represent the shape of temporal, range, azimuth, and channel dimensions, respectively. Simultaneously, the corresponding RGB image sequences are fed into a ResNet-50 [63] with pre-trained weights provided by torchvision, and temporal aggregation is carried out through 3D convolutional layers to obtain the RGB image features $\mathcal{X}_{camera} = \left[F_c^1,\ F_c^2,\ \ldots,\ F_c^T\right] \in \mathbb{R}^{T\times H\times W\times C}$. Here, H and W denote the shape of row and column of RGB image features, respectively.

After obtaining RF image features $\mathcal{X}_{radar}$ and RGB image features $\mathcal{X}_{camera}$, we first initialize queries $Query = \left[Q^1,\ Q^2,\ \ldots,\ Q^T\right] \in \mathbb{R}^{T\times R\times A\times C}$ uniformly in polar space. For Q^i, F_c^i and F_r^i (where i ranges from 1 to T), we directly employ PAA from [14]. Specifically, according to [14], queries of the same azimuth can be associated with the same column of the RGB image. As a result, we use all the queries with the same azimuth as the query matrix, and all the features in the corresponding column in the RGB image as the key and value matrices for the cross attention operation. Given a query $q \in \mathbb{R}^C$ located at (φ, ϕ) from Q^i, this process can be expressed as follows:

$$F_c^i(x,\cdot) = \text{Stack}\left(\left[F_c^i(x,y)\right]\right) \in \mathbb{R}^{H\times C},\ \forall y \in [0,\ H-1] \quad (30)$$

$$\hat{q}(\varphi,\phi) = \text{CrossAttention}\left(q(\varphi,\phi),\ F_c^i(x,\cdot),\ F_c^i(x,\cdot)\right) \quad (31)$$

where $F_c^i(x,y)$ is the RGB image feature located at (x,y).

Authorized licensed use limited to: University of Science & Technology of China. Downloaded on January 26,2026 at 08:17:20 UTC from IEEE Xplore. Restrictions apply.

TABLE V
RESULTS OF THE FUSION MODEL ON THE CRUW DATASET

Method	All		Pedestrian		Cyclist		Car	
	AP(%)	AR(%)	AP(%)	AR(%)	AP(%)	AR(%)	AP(%)	AR(%)
Mask-RadarNet	84.29	87.36	82.74	85.80	85.06	86.67	85.96	90.66
Mask-RadarNet + PAA	**84.93**	**88.22**	**83.82**	**86.81**	**86.17**	**89.94**	**86.18**	**91.12**

All $\hat{q}$ at different locations form the updated query $\hat{Q}^i$. Then we use $\hat{Q}^i$ to integrate features from F_r^i with cross attention operation. This step can be formulated as:

$$\tilde{Q}^i = CrossAttention\left(\hat{Q}^i,\ F_r^i,\ F_r^i\right) \tag{32}$$

The final fused feature $\mathcal{X}_{fuse}$ can be represented as $\mathcal{X}_{fuse} = \left[\tilde{Q}^1,\ \tilde{Q}^2,\ \ldots,\ \tilde{Q}^T\right] \in \mathbb{R}^{T\times R\times A\times C}$, which is then fed into the main decoder of Mask-RadarNet for feature decoding. The other parts of the fusion model are consistent with the original Mask-RadarNet.

The results of the fusion model are shown in Table V. It can be observed that the fusion method improves the detection performance of all categories of objects. Notably, thanks to the high resolution and fine-grained features provided by RGB images, the fusion method achieves a more significant performance improvement for small objects, with an AP increase of 1.08% for pedestrians and an AP increase of 1.11% for cyclists. These improvements reveal the huge benefit of fusing radar and RGB images, and we will conduct more research in this area in the future.

V. CONCLUSION

In this paper, we proposed a transformer-based model called Mask-RadarNet for radar object detection, which addresses the issue of leaving out spatial-temporal semantic context during encoding. Specifically, our Mask-RadarNet exploits the combination of interleaved convolution and attention operations on multiframe RF images to extract both local and global features. Patch shift was introduced to attention operations as an efficient method for temporal modeling. Moreover, we designed a simple yet effective module called CMAM to capture the spatial-temporal contextual information in our encoder. Besides, a lightweight auxiliary decoder was proposed to aggregate prior maps that guide the model in fine-tuning the features it captures, thereby enhancing the network's performance. The experimental results showed that our Mask-RadarNet achieves SOTA performance with lower GFLOPs and fewer parameters on the CRUW dataset.

Although our model has demonstrated promising performance in many aspects, we find that detecting low-speed targets, such as pedestrians and bicycles, remains a significant challenge compared to detecting cars. This is due to the fact that the Doppler characteristics of low-speed targets captured by radar are not pronounced. In future work, we will continue to explore the integration of radar and cameras to improve the detection performance of low-speed targets.

REFERENCES

[1] J. Karangwa, J. Liu, and Z. Zeng, "Vehicle detection for autonomous driving: A review of algorithms and datasets," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 11, pp. 11568–11594, Nov. 2023.

[2] B. R. Kiran et al., "Deep reinforcement learning for autonomous driving: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 6, pp. 4909–4926, Jun. 2022.

[3] K. Muhammad, A. Ullah, J. Lloret, J. D. Ser, and V. H. C. de Albuquerque, "Deep learning for safe autonomous driving: Current challenges and future directions," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4316–4336, Jul. 2021.

[4] J. Levinson et al., "Towards fully autonomous driving: Systems and algorithms," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2011, pp. 163–168.

[5] X. Huang et al., "The ApolloScape dataset for autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 1067–10676.

[6] X. Wang, K. Li, and A. Chehri, "Multi-sensor fusion technology for 3D object detection in autonomous driving: A review," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 2, pp. 1148–1165, Feb. 2024.

[7] Y. Wang, G. Wang, H.-M. Hsu, H. Liu, and J.-N. Hwang, "Rethinking of radar's role: A camera-radar dataset and systematic annotator via coordinate alignment," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 2809–2818.

[8] X. Dong, B. Zhuang, Y. Mao, and L. Liu, "Radar camera fusion via representation learning in autonomous driving," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 1672–1681.

[9] R. Nabati and H. Qi, "RRPN: Radar region proposal network for object detection in autonomous vehicles," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 3093–3097.

[10] M. Meyer and G. Kuschk, "Deep learning based 3D object detection for automotive radar and camera," in *Proc. 16th Eur. Radar Conf. (EuRAD)*, Oct. 2019, pp. 133–136.

[11] O. Schumann, M. Hahn, J. Dickmann, and C. Wöhler, "Semantic segmentation on radar point clouds," in *Proc. 21st Int. Conf. Inf. Fusion (FUSION)*, Jul. 2018, pp. 2179–2186.

[12] N. Scheiner et al., "Seeing around street corners: Non-line-of-sight detection and tracking in-the-wild using Doppler radar," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2065–2074.

[13] Y. Wang, Z. Jiang, Y. Li, J.-N. Hwang, G. Xing, and H. Liu, "RODNet: A real-time radar object detection network cross-supervised by camera-radar fused object 3D localization," *IEEE J. Sel. Topics Signal Process.*, vol. 15, no. 4, pp. 954–967, Jun. 2021.

[14] Y. Liu, F. Wang, N. Wang, and Z. Zhang, "Echoes beyond points: Unleashing the power of raw radar data in multi-modality fusion," in *Proc. Adv. Neural Inf. Process. Syst.*, 2023, pp. 1–17.

[15] B. Major et al., "Vehicle detection with automotive radar using deep learning on Range-Azimuth-Doppler tensors," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 924–932.

[16] X. Dong, P. Wang, P. Zhang, and L. Liu, "Probabilistic oriented object detection in automotive radar," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 458–467.

[17] R. Miller, "Fundamentals of radar signal processing (Richards, M.A.; 2005) [book review]," *IEEE Signal Process. Mag.*, vol. 26, no. 3, pp. 100–101, May 2009.

[18] A. Angelov, A. Robertson, R. Murray-Smith, and F. Fioranelli, "Practical classification of different moving targets using automotive radar and deep neural networks," *IET Radar, Sonar Navigat.*, vol. 12, no. 10, pp. 1082–1089, Oct. 2018.

[19] A. Zhang, F. E. Nowruzi, and R. Laganiere, "RADDet: Range-Azimuth-Doppler based radar object detection for dynamic road users," in *Proc. 18th Conf. Robots Vis. (CRV)*, May 2021, pp. 95–102.

Authorized licensed use limited to: University of Science & Technology of China. Downloaded on January 26,2026 at 08:17:20 UTC from IEEE Xplore. Restrictions apply.

[20] A. Ouaknine, A. Newson, J. Rebut, F. Tupin, and P. Pérez, "CARRADA dataset: Camera and automotive radar with range{-} angle{-} Doppler annotations," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 5068–5075.

[21] A. Neubeck and L. Van Gool, "Efficient non-maximum suppression," in *Proc. 18th Int. Conf. Pattern Recognit. (ICPR)*, 2006, pp. 850–855.

[22] C.-C. Hsu, C. Lee, L. Chen, M.-K. Hung, Y.-L. Lin, and X.-Y. Wang, "Efficient-ROD: Efficient radar object detection based on densely connected residual network," in *Proc. Int. Conf. Multimedia Retr.*, Aug. 2021, pp. 526–532.

[23] P. Sun, X. Niu, P. Sun, and K. Xu, "Squeeze-and-excitation network-based radar object detection with weighted location fusion," in *Proc. Int. Conf. Multimedia Retr.*, Aug. 2021, pp. 545–552.

[24] T. Jiang, L. Zhuang, Q. An, J. Wang, K. Xiao, and A. Wang, "T-RODNet: Transformer for vehicular millimeter-wave radar object detection," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–12, 2023.

[25] W. Xiang, C. Li, B. Wang, X. Wei, X.-S. Hua, and L. Zhang, "Spatiotemporal self-attention modeling with temporal patch shift for action recognition," 2022, *arXiv:2207.13259*.

[26] S. Capobianco et al., "Vehicle classification based on convolutional networks applied to FMCW radar signals," in *Proc. Traffic Mining Appl. Police Activities*, 2018, pp. 115–128.

[27] J. Gao, B. Deng, Y. Qin, H. Wang, and X. Li, "Enhanced radar imaging using a complex-valued convolutional neural network," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 1, pp. 35–39, Jan. 2019.

[28] S. Hazra and A. Santra, "Short-range radar-based gesture recognition system using 3D CNN with triplet loss," *IEEE Access*, vol. 7, pp. 125623–125633, 2019.

[29] L. Zhuang, T. Jiang, J. Wang, Q. An, K. Xiao, and A. Wang, "Effective mmWave radar object detection pretraining based on masked image modeling," *IEEE Sensors J.*, vol. 24, no. 3, pp. 3999–4010, Feb. 2024.

[30] J. Liu et al., "Deep instance segmentation with automotive radar detection points," *IEEE Trans. Intell. Vehicles*, vol. 8, no. 1, pp. 84–94, Jan. 2023.

[31] W. Xiong, J. Liu, Y. Xia, T. Huang, B. Zhu, and W. Xiang, "Contrastive learning for automotive mmWave radar detection points based instance segmentation," in *Proc. IEEE 25th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2022, pp. 1255–1261.

[32] J. Liu, Q. Zhao, W. Xiong, T. Huang, Q.-L. Han, and B. Zhu, "SMURF: Spatial multi-representation fusion for 3D object detection with 4D imaging radar," *IEEE Trans. Intell. Vehicles*, vol. 9, no. 1, pp. 1–14, Jan. 2024.

[33] L. Zheng et al., "RCFusion: Fusing 4-D radar and camera with bird's-eye view features for 3-D object detection," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–14, 2023.

[34] W. Xiong, J. Liu, T. Huang, Q.-L. Han, Y. Xia, and B. Zhu, "LXL: LiDAR excluded lean 3D object detection with 4D imaging radar and camera fusion," *IEEE Trans. Intell. Vehicles*, vol. 9, no. 1, pp. 79–92, Jan. 2024.

[35] A. Vaswani et al., "Attention is all you need," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, vol. 30. Red Hook, NY, USA: Curran Associates Inc., 2022, pp. 5998–6008.

[36] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. 9th Int. Conf. Learn. Represent.*, May 2021, pp. 1–8.

[37] J. Fan, B. Gao, Q. Ge, Y. Ran, J. Zhang, and H. Chu, "SegTransConv: Transformer and CNN hybrid method for real-time semantic segmentation of autonomous vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 2, pp. 1586–1601, Feb. 2024.

[38] H. Wang, Y. Zhu, H. Adam, A. Yuille, and L.-C. Chen, "MaX-DeepLab: End-to-end panoptic segmentation with mask transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 5459–5470.

[39] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," 2020, *arXiv:2005.12872*.

[40] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," 2020, *arXiv:2010.04159*.

[41] G. Bertasius, H. Wang, and L. Torresani, "Is space-time attention all you need for video understanding?," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 1–11.

[42] Z. Liu et al., "Video Swin transformer," 2021, *arXiv:2106.13230*.

[43] M. Pu, Y. Huang, Y. Liu, Q. Guan, and H. Ling, "EDTER: Edge detection with transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 1392–1402.

[44] W. Wang, C. Chen, M. Ding, H. Yu, S. Zha, and J. Li, "TransBTS: Multimodal brain tumor segmentation using transformer," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2021, pp. 109–119.

[45] B. Xie, G. Milam, B. Ning, J. Cha, and C. H. Park, "DXM-TransFuse U-Net: Dual cross-modal transformer fusion U-Net for automated nerve identification," *Computerized Med. Imag. Graph.*, vol. 99, Jul. 2022, Art. no. 102090.

[46] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9992–10002.

[47] H. Cao et al., "Swin-UNet: UNet-like pure transformer for medical image segmentation," in *Proc. Eur. Conf. Comput. Vis. Workshops (ECCVW)*, 2022, pp. 205–218.

[48] W. Xu, Y. Xu, T. Chang, and Z. Tu, "Co-scale conv-attentional image transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 9961–9970.

[49] X. Chu et al., "Twins: Revisiting the design of spatial attention in vision transformers," in *Proc. NeurIPS*, 2021, pp. 1–9.

[50] W. Wang et al., "Pyramid vision transformer: A versatile backbone for dense prediction without convolutions," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 548–558.

[51] Z. Tu et al., "MaxViT: Multi-axis vision transformer," 2022, *arXiv:2204.01697*.

[52] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[53] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking Atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*.

[54] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6230–6239.

[55] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Learning a discriminative feature network for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 1857–1866.

[56] H. Zhang et al., "Context encoding for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Mar. 2018, pp. 7151–7160.

[57] J. Jain et al., "SeMask: Semantically masked transformers for semantic segmentation," 2021, *arXiv:2112.12782*.

[58] H. Zhang et al., "Context encoding for semantic segmentation," 2018, *arXiv:1803.08904*.

[59] Z. Jin, B. Liu, Q. Chu, and N. Yu, "ISNet: Integrate image-level and semantic-level context for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 7169–7178.

[60] J. Lin, C. Gan, and S. Han, "TSM: Temporal shift module for efficient video understanding," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7082–7092.

[61] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[62] X. Ding, X. Zhang, J. Han, and G. Ding, "Scaling up your kernels to 31×31: Revisiting large kernel design in CNNs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 11953–11965.

[63] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

Authorized licensed use limited to: University of Science & Technology of China. Downloaded on January 26,2026 at 08:17:20 UTC from IEEE Xplore. Restrictions apply.