Hindawi
Complexity
Volume 2021, Article ID 6791882, 12 pages
https://doi.org/10.1155/2021/6791882

Research Article

Multiscale Efficient Channel Attention for Fusion Lane Line Segmentation

Kang Liu and Xin Gao

School of Mechanical Electronic & Information Engineering, China University of Mining & Technology, Beijing, Beijing 100 083, China

Correspondence should be addressed to Xin Gao; bqt2000405024@student.cumtb.edu.cn

Received 10 August 2021; Revised 1 November 2021; Accepted 23 November 2021; Published 7 December 2021

Academic Editor: Chao Zeng

The use of multimodal sensors for lane line segmentation has become a growing trend. To achieve robust multimodal fusion, we introduced a new multimodal fusion method and proved its effectiveness in an improved fusion network. Specifically, a multiscale fusion module is proposed to extract effective features from data of different modalities, and a channel attention module is used to adaptively calculate the contribution of the fused feature channels. We verified the effect of multimodal fusion on the KITTI benchmark dataset and A2D2 dataset and proved the effectiveness of the proposed method on the enhanced KITTI dataset. Our method achieves robust lane line segmentation, which is 4.53% higher than the direct fusion on the precision index, and obtains the highest F2 score of 79.72%. We believe that our method introduces an optimization idea of modal data structure level for multimodal fusion.

1. Introduction

Reliable and robust lane line segmentation is one of the basic requirements of autonomous driving. After all, in order to ensure that unmanned vehicles drive on the correct and reasonable roads, the vehicle must be able to detect the lane line the first time. The driving assistance system provides a decision-making basis for the autonomous driving control module through the results of lane line detection [1]. In this article, we focus on lane line segmentation based on multiple sensor fusion.

Existing algorithms rely heavily on the camera, which provides a rich visual description of the environment [2, 3]. The camera image has the original high-resolution and efficient array storage structure. It can provide long-distance dense information under good light and sunny weather conditions, and it is efficient in storage and calculation. However, when perceiving the surrounding environment, the performance of the camera is easily affected by the light intensity and sharp changes in light [4, 5]. Unlike cameras, LiDAR retains an accurate three-dimensional point cloud of the surrounding environment and directly provides accurate distance measurement. Although the depth information is very accurate, the LiDAR usually has a measurement range of only 10 to 100 meters and can only provide sparse and irregular point cloud data. The empty voxels caused by the sparse point cloud bring the accuracy requirements of lane line detection. Here comes the challenge.

At present, most of the sensing sensors of vehicles on the road work independently, which means that they hardly exchange information with each other. Instead, their respective sensing modules process the data of a single sensor and then deliver the sensing results to the decision-making module. This method increases the number of perception modules and imposes a great burden on the calculation efficiency of onboard computing resources and decision-making modules [6, 7]. The fusion of information from multiple sensors is a growing trend and the key to efficient autonomous driving. Multimodal fusion can take advantage of the complementarity of different sensor information and use feature-level fusion to promote semantic segmentation, thereby improving the accuracy and efficiency of lane line

segmentation and ensuring the correctness and timeliness of decision-making.

Some recent work has explored the use of camera images and LiDAR point clouds for lane line segmentation tasks in autonomous driving. Due to the perspective transformation in imaging, the camera image cannot describe the accurate distance information, and the method of directly using the two-dimensional camera image for lane line segmentation is unreliable [8]. Although the depth information of the LiDAR point cloud is already available, so far, the main success of the fusion method is to use the advantages of multimodal data to supplement the camera image with the precise depth information of the LiDAR. Previous studies put multimodal fusion in a two-dimensional space, usually using a direct stacking method to fuse the depth information of point cloud data with the camera image with a fixed weight. Another idea is to fuse multimodality [9, 10]; they put it in the three-dimensional space, make full use of the accurate representation of the distance information of the point cloud data, and fuse the data in the three-dimensional space. However, the camera image and the LiDAR point cloud are data of different modalities and have great differences[11]. The direct stacking fusion method ignores the characteristics of multimodal data and will inhibit the respective advantages of multimodal data, and may even appear the effective fusion information is misjudged as the negative effect of noise. While placing multimodal fusion in a high-dimensional space, algorithms based on 3D detection often require large computing resources, which are difficult to meet the needs of lightweight and real time in autonomous driving[12]. For this reason, we propose a novel multimodal fusion lane line segmentation method based on multiscale convolution and channel attention mechanisms. We believe that multimodal fusion should focus on the fusion feature space, and use reasonable methods and weights to guide multimodal fusion.

In order to make full use of multisource data for reasonable control and use, we need to explore a question: what method should be used to promote semantic segmentation to obtain better lane line segmentation results. To this end, we first analyzed the benchmark dataset for lane line segmentation. In the KITTI dataset, the area occupied by lane lines in the image is only 1.5% to 2%, and the problem of class imbalance is quite serious. In this article, we hope that when the deep learning network is extracting features, it can more effectively focus on the characteristics of lane lines, thereby improving the quality of the segmentation results. For this reason, we use multiscale convolution for feature fusion in multimodal fusion and introduce the channel attention mechanism to modify the fusion weight. The results are shown in Figure 1, and we believe that the task of lane line segmentation should find a way to maximize the effect of multimodal data under the premise of ensuring the quality of the data.

This article is organized as follows: in Section 2, we separately analyzed the current lane line segmentation algorithms based on camera images and point clouds and introduced the current status of the fusion method; in Section 3, we carried out the proposed method and network structure in detail; Section 4 discussed the processing of the dataset, as well as the experimental results and performance evaluation obtained after applying the proposed method; in Section 5, an ablation experiment was used to measure the contribution of each module in the proposed method; and in Section 6, the proposed methods are summarized and future directions are provided.

In conclusion, the main contributions of the article are as follows: (1) an idea of using multiscale convolution for multimodal fusion lane line segmentation is proposed; (2) ECANet[13] is used for the weight correction of the fusion feature channel, which effectively improves the accuracy of the lane line segmentation model; and (3) the proposed multiscale efficient channel attention(MS-ECA) can be widely used in the field of multimodal fusion and has good mobility.

2. Related Work

2.1. Lane Line Segmentation. The traditional lane line segmentation uses the canny operator to detect the sharp change in brightness[14], which is defined as an edge under a given threshold, and then uses the Hough transform to find the lane line. In recent years, the emergence of machine learning has promoted the development of artificial intelligence, and the wide application of deep learning has made feature-level lane line segmentation algorithms gradually mature [15, 16]. Wenjie Song et al. [17] designed an adaptive traffic lane model in the Hough space. The model has a maximum likelihood angle and a dynamic rod detection area (ROI) of interest. This model can also be improved through geographic information systems or electronic maps to obtain more accurate results. To get more accurate results. Xingang Pan et al.[18] proposed spatial CNN(SCNN), which extended the traditional layer-by-layer convolution to the slice-by-slice convolution in the feature map, thereby enabling message passing between pixels between rows and columns in a layer. Bei He et al. [19] designed a DVCNN network that optimizes both the front view and the top view. The front view image is used to eliminate false detections, the top view image is used to remove nonclub-shaped structures, such as ground arrows and text, and a large number of complex constraints are used. Conditions improve the quality of lane line detection. However, due to the photosensitivity of the camera, lane line detection based on pure vision still has great challenges in terms of performance and robustness.

Some recent work has explored the use of multimodal fusion for detection and segmentation tasks in autonomous driving [17, 20, 21]. Andreas Eitel introduced a multistage training method that effectively encodes the depth information of CNN[22], so that learning does not require large depth datasets, through the data enhancement scheme of robust learning of the depth image, it is corroded with the real noise mode [23]. Hyunggi Cho et al. [20] redesigned the sensor configuration and installed multiple LiDAR pairs and vision sensors. Based on the combination of measurement models of multiple sensors, they proposed a new moving target detection and tracking system. Reference [24] explored all aspects of pedestrian detection by fusing LiDAR

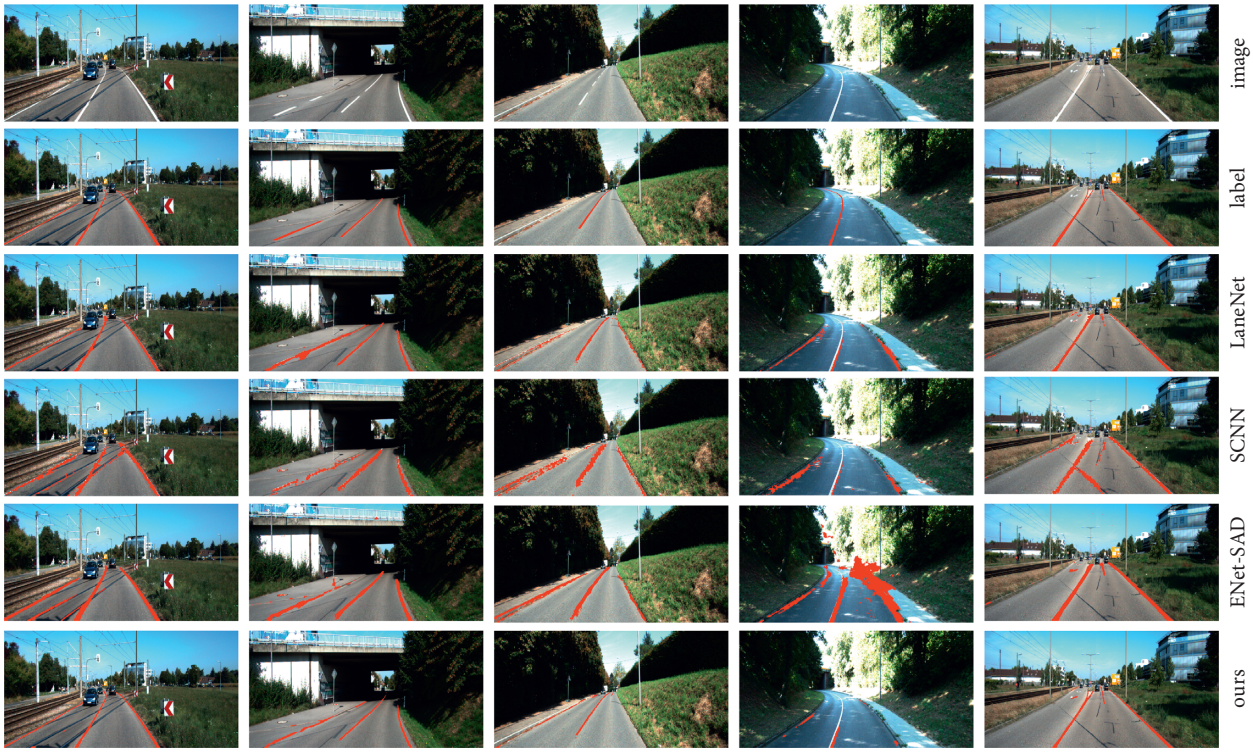

Figure 1: Test results on KITTI-aug dataset. The rows from top to bottom are input images, ground truth, output from LaneNet, SCNN, ENet-SAD, and ours.

and color images in the context of convolutional neural networks. This work samples the point cloud into a dense depth map, then extracts three features representing different aspects of the 3D scene, and use LiDAR as an additional image channel for training. However, current fusion algorithms pay more attention to data quality and network structure, and the characteristics of multimodal data and the representation of fusion data have not been paid attention to. The difference is that our proposed method naturally selects the fusion weight and the fusion channel adaptively in the fusion and effectively shows the advantages of multimodal data.

2.2. Attention Mechanism. The attention mechanism has recently been widely used to learn the weight distribution [25], and the neural network is used to focus on different parts of the input data or feature maps, so that the attention module is designed to weight the input data or feature maps. Jianlong Fu et al. [26] used a classification network and a network to generate attention proposal on each target scale of concern, defined a rank loss to train the attention proposal, and forced the final scale to obtain a classification result that was better than the previous one, so that the attention proposal extracts the target part that is more conducive to fine classification [27]. In the classification network, an attention module composed of two branches is added [28]: one is a traditional convolution operation, and the other is two downsampling plus two upsampling operations; the purpose is to obtain the larger receptive field serves as an attention map. High-level information is more important in classification problems; they use an attention map to improve the receptive field of low-level features and highlight features that are more beneficial to classification. Liang-Chieh Chen et al. [29] constructed multiple scales by scaling the scale of the input picture. The traditional method is to use average pooling or max pooling to fuse features of different scales, and they constructed an attention model composed of two convolutional layers to automatically learn the weights of different scales for fusion. We have empirically found that due to the small proportion of lane lines in the image, the overall attention of the spatial attention mechanism may interfere with segmentation. Therefore, our work pays more attention to the effect of the channel attention mechanism on multimodal fusion.

3. Methods

In this part, we introduce the basic structure of our network and introduce the proposed multiscale convolution fusion module, and related experiments are completed based on this part of the network.

3.1. Baseline for Multimodel Fusion. Lane line segmentation is a typical pixel-level segmentation task. We established a baseline fusion model based on Unet [30]. As shown in Figure 2, its input is two modal data, which is the same as most current fusion methods. Multimodal data are concat-fused together after a convolution. The baseline model is trained end-to-end by an encoder and a decoder, and the size of the convolution kernel of all convolution blocks is $3 * 3$.

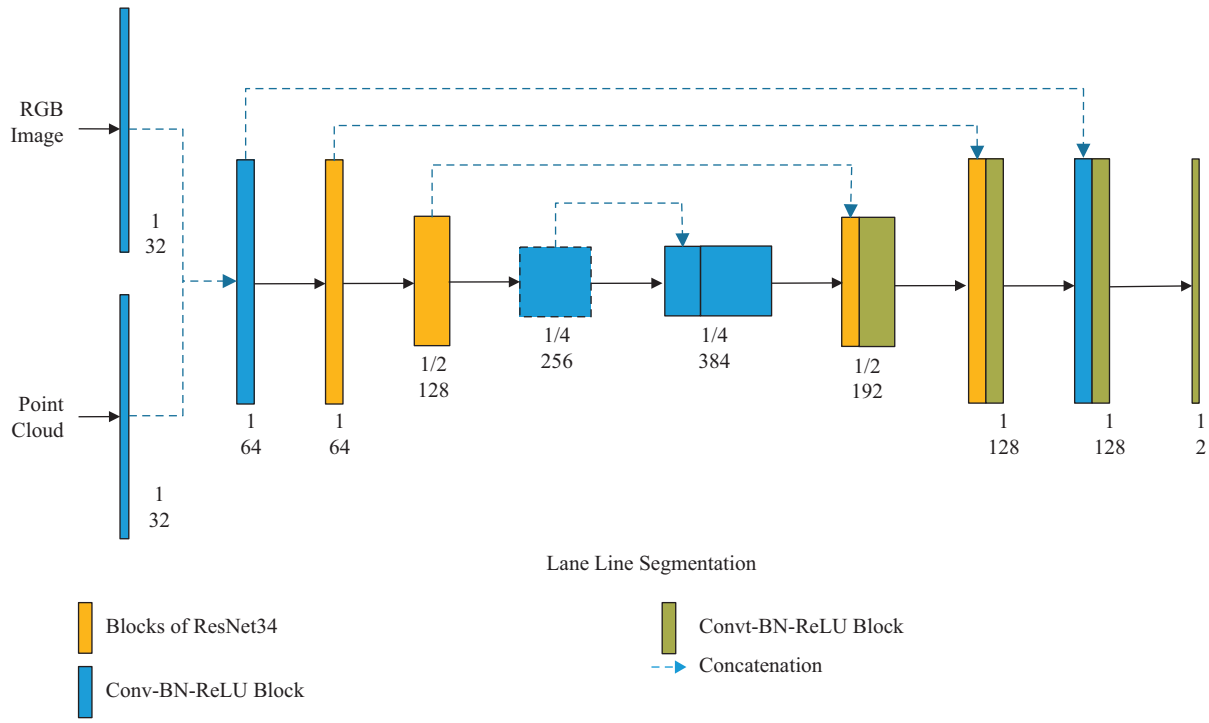

Figure 2: Our fusion modal baseline: it takes an RGB image and a point cloud as input and outputs a 512 ∗ 256 binary map.

Based on Unet's skip connection, we link the output of each block in the encoder to a block of the corresponding size in the decoder and use different levels of feature map semantic information through concatenating.

3.2. Multiscale Convolution Fusion. Generally, for a given task model, the size of the convolution kernel is determined, and the convolution kernel of uniform size can be easily calculated. However, studies have shown that for a given input, if the network can adaptively adjust the size of the receptive field according to the multiple scales of the input information, extract the features under the multiscale receptive field, and finally, use the "selection" mechanism to fuse multiscale features, the performance of the model can be effectively improved. For the camera image and LiDAR point cloud data, although they are not the same input data, they are aligned to describe the same scene. We creatively use multiscale convolution to extract features for these two modalities. Obtain multimodal features under different sizes of receptive fields, and finally fuse them to obtain multiscale multimodal fusion features.

Based on SKNet's[31] dynamic selection strategy, we also choose 3 ∗ 3 and 5 ∗ 5 size convolution kernels as multiscale convolution kernels. Generally speaking, a camera image will have millions of pixels. In contrast, the performance of LiDAR for the same scene is often only tens of thousands of effective points. Even after point cloud completion processing, it still looks sparse compared to the camera image. Therefore, as shown in Figure 3, we use the 5 ∗ 5 size convolution kernel for the point cloud branch and use the 3 ∗ 3 convolution kernel for the camera image branch, which will be more conducive to the extraction of the original effective information. In order to further improve the efficiency, the conventional convolution of the 5 ∗ 5 convolution kernel is replaced with a 3 ∗ 3 convolution kernel and an expanded convolution with an expansion size of 2.

We naturally use the Fuse and Select operations in SKNet to calculate fusion multiscale features. We embed global information by simply using global average pooling to generate channel-level statistics. Specifically, the c-th element of s is calculated by reducing the spatial dimension H ∗ W:

$$s_c = \mathscr{F}_{gp}(U_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} U_c(i, j), \tag{1}$$

and then, a simple fully connected layer is used to realize the guidance of accurate and adaptive selection, and reduce the dimension to improve the efficiency:

$$z = \mathscr{F}_{fc}(s) = \delta(\mathscr{B}(Ws)), \tag{2}$$

where δ is the ReLU function and $\mathscr{B}$ is the batch normalization, and $W \in \mathbb{R}^{d \times C}$. Finally, we adaptively choose different spatial scales to obtain cross-channel attention weight. Specifically, the softmax operator is applied to the channel-wise digits; in (3), z is the compact feature descriptor, and a, b denote the soft attention vectors:

$$a_c = \frac{e^{A_c z}}{e^{A_c z} + e^{B_c z}}, b_c = \frac{e^{B_c z}}{e^{A_c z} + e^{B_c z}}. \tag{3}$$

In this process, convolution kernels of different sizes provide multiscale receptive fields for the two modes of data, and large convolution kernels can extract the features of

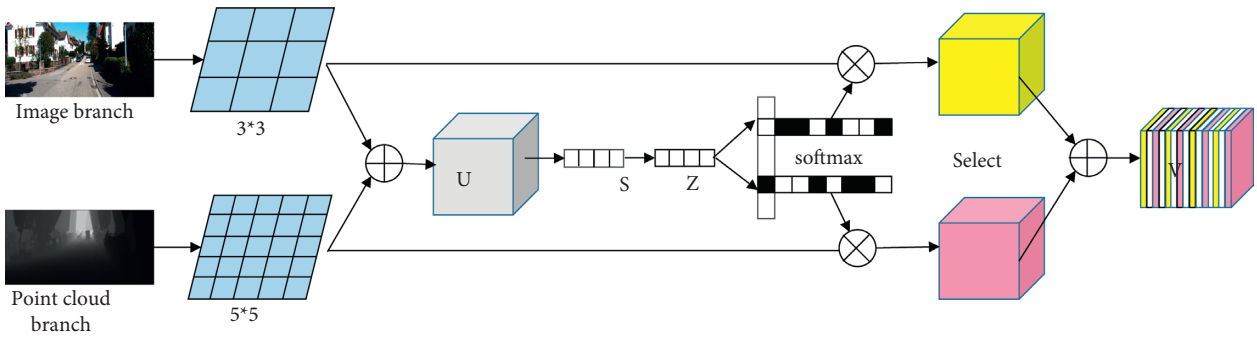

Figure 3: Structure of multimodal multiscale fusion. We use a 3 ∗ 3 convolution kernel for the image branch and a 5 ∗ 5 convolution kernel for the point cloud branch.

sparse point cloud data more effectively, which is very helpful for multiscale fusion. In addition, in the process of using channel-level statistical information to embed global information, the nonlinear learning in the network is increased, which alleviates the negative impact of rough conversion of multimodal data to the same feature space to a certain extent, and improves the learning ability of the network. After weighting the features of multimodal branches with the weights of channel level, the expression of lane line features of each modal branch can be increased better, so that lane line features can be extracted more effectively after fusion.

3.3. Local Interaction of Fusion Feature Channels. In the task of lane line segmentation, the proportion of the lane line area in the image is very small, and it is a serious challenge to efficiently extract the lane line features from a large amount of background or noise. In this kind of unbalanced data, in order to allow the network structure to adaptively pay attention to the lane line features, we use an efficient attention mechanism. It can be seen from the figure that in the process of extracting features from the network, due to the difference of filters, the focus of feature extraction from different feature channels is different. In this process, some feature channels can extract rich features. Information and some feature channels contain a lot of noise information. In a neural network, these feature channels will be stacked in sequence to act on the segmentation task. Naturally, how to enhance this part of the efficient feature channel becomes a problem.

At the same time, considering that the lane line segmentation task is a prerequisite for unmanned driving decision planning and has high real-time requirements, we use the lightweight channel attention mechanism model ECANet for the fusion features after multimodal fusion. Note that we only discuss the effect of lightweight attention mechanism on multiscale and multimodal fusion. Through the channel attention mechanism, we calculate the importance of each feature channel of the fusion feature in the network and let the network adaptively learn the contribution of each feature channel to the lane line segmentation task, and the feature channels that make a positive contribution to the segmentation will be adaptively enhanced; otherwise, they will be suppressed.

As shown in Figure 4, in the idea of ECANet, the importance of each feature channel will be represented by modeling, and the neighboring channels are correlated, and the weight of each feature channel will be calculated by its neighboring neighbor channels, so that it can avoid dimensional loss while capturing local cross-channel interactive information. We integrate ECANet into the task of multimodal fusion lane line segmentation and obtain a model with lower model complexity and smaller network parameters. The network structure of ECANet is shown in Figure 5.

Without dimensionality reduction, ECANet calculates the nearby k channels of each feature channel centered on itself and uses the correlation between adjacent channels to interact with local information. In this channel weight calculation, the lane feature channels that perform well in the effective features of the line will receive greater attention, which will lead to positive positive contributions to the feature channels nearby. When the channel dimension C is given, the value of k can be determined adaptively according to the following formula:

$$C = \phi(k) = 2^{(\gamma * k - b)}. \tag{4}$$

$$k = \psi(C) = \left| \frac{\log_2(C)}{\gamma} + \frac{b}{\gamma} \right|_{o\,dd}, \tag{5}$$

where $|T|_{\text{odd}}$ indicates the nearest odd number of t. Same as ECANet, we set γ and b to 2 and 1. In the experiment, the calculation result of k is an odd number not exceeding 9.

We embed the channel attention module after the fusion module to perform channel-level weight correction on the fusion features after multimodal fusion. The fusion features obtained through fusion between the image branch and the point cloud branch are used as the input of the channel attention module, and the output of the channel attention module is used as the input of the next layer of the network structure in the original baseline. The network structure is shown in Figure 5, and the whole of the multimodal fusion module and the channel attention module is called MS-ECA (Table 1).

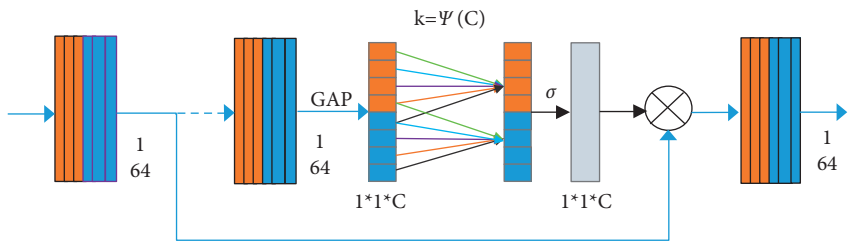

Figure 4: The structure of efficient channel attention. The input channel here is 64.

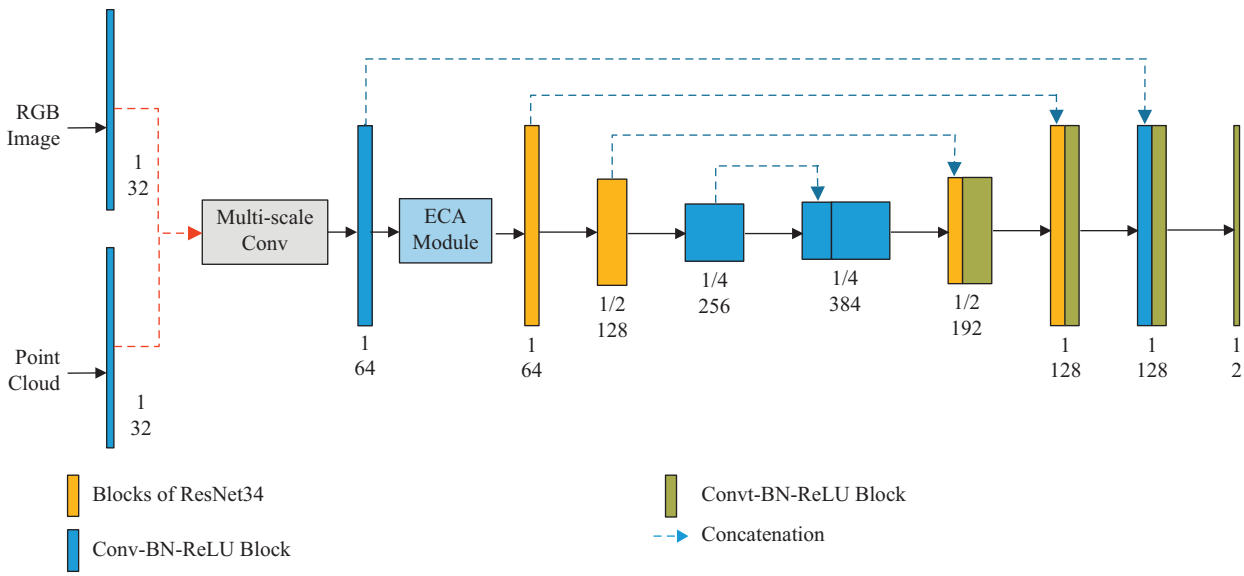

Figure 5: The structure of our method(multiscale efficient channel attention) embedded in the baseline.

Table 1: Information of the lane line segmentation datasets.

Name	Frame	Train	Validation	Test	Resolution
KITTI	383	228	40	115	1242 × 375
KITTI-aug	3331	2736	480	115	1242 × 375
A2D2	470	282	47	141	1920 × 1208

4. Experiment

4.1. Dataset Preparation. The current multimodal lane line segmentation dataset is relatively lacking. To verify the proposed method, we conducted extensive experiments on the benchmark datasets KITTI-Road[32] and A2D2[33]. As shown in Figure 6, the KITTI-Road and A2D2 datasets include synchronized camera images and LiDAR point clouds with calibration parameters and ground truth values. We filter out complex cross-lines or forward lines in the dataset and use the remaining data to validate our proposed method and model.

In the processing of the dataset, we also filtered out confusing lane lines, such as markings on the sidewalk and signs outside the lane lines to better meet the task requirements of lane line segmentation. Compared with the TuSimple dataset, in order to extract the lane line features more accurately, we only use the visible lane line pixels on the image and ignore the part of the lane line behind obstacles or other invisible lane lines to ensure that the network learns completely the characteristics of the lane line. Finally, the dataset annotations are redone as pixel-level lane line labels. In training, we use the same feature extraction module to extract features of the camera image and the point cloud. As for the network input, the initial size of the original camera image and the corresponding point cloud is 1242 ∗ 375, in order to reduce the calculation overhead, in the data preprocessing, we reshape them to the size of (256, 512) in the same way and then input them into the network.

The KITTI and A2D2 datasets have limited samples. In order to conduct experiments better, we need to carry out reasonable data enhancement. In the acquisition of the A2D2 dataset, only one 8-line and two 16-line LiDARs are used to collect point cloud data. The point cloud is very sparse and contains little information. In contrast, KITTI uses a 64-line LiDAR to complete the point cloud collection, and the resulting point cloud has a richer description of the entire space. Therefore, we use the KITTI dataset as the main verification dataset. In addition, we performed strategies such as cropping, brightness conversion, and adding noise to the KITTI data and obtained a dataset 12 times the original KITTI data, which is represented by KITTI-AUG. All experiments use 60% of the data as the training set, 30% for the test model, and the remaining data for verification in training. The dataset information we use is shown in Table 1.

Figure 6: Examples of KITTI(top row) and A2D2(bottom row). The columns from left to right are camera image, lane line label, and point cloud.

4.2. Training Procedure. In order to ensure fairness, all experiments are implemented on a standard training platform, with only differences in the methods in the neural network. Our hardware platform has the following: 8 GB of RAM, a three-core E5 series CPU, and an NVIDIA TiTan XP GPU with 12G memory, and the operating system is Ubuntu 16.04. All the codes are based on the PyTorch framework. We have implemented end-to-end network training. In order to speed up the convergence, we use the Adam optimization algorithm, and the parameters in Adam are default values. In order to prevent the difficulty of finding the optimal solution during training, we also use a learning rate LR with periodic decay:

$$\text{nllloss} = -\sum_{n=1}^{N} y_n \text{logprob}(x_n), \tag{6}$$

and we use the Adam optimization algorithm [34] to train the network end-to-end, using a periodic decay learning rate LR:

$$lr_0 = 0.0001lr = 2^{\lfloor epoch/50 \rfloor} \times 0.8^{\lfloor epoch/1 \rfloor} \times lr_0, \tag{7}$$

where $lr_0 = 0.0001lr_0$ is the initial learning rate. The training rounds and batch sizes of all experiments are set to 200 and 4, respectively. In the training process, we use the strategy of using the validation set to verify the current model while training. Specifically, we will use the current model parameters to perform a performance evaluation on the validation set every 5 epochs of training. If the current model parameters have achieved performance upgrade, the corresponding weight file and related verification results will be automatically saved.

In semantic segmentation tasks, recall and accuracy are both important indicators to measure model performance. For lane line segmentation, the recall rate reflects the proportion of lane line pixels correctly predicted by the model in all positive samples, and the accuracy rate reflects the proportion of real lane line pixels in the result of the model prediction. The formula is as follows:

$$\text{precision} = \frac{tp}{tp + fp}. \tag{8}$$

$$\text{recall} = \frac{tp}{tp + fn}. \tag{9}$$

In addition, in order to make a clearer comparison, we also used F-measure(including F1 and F2) and calculated the accuracy of the overall prediction as "acc." Finally, in order to verify the real-time performance of the proposed method, we calculated the FPS of the lane line segmentation in the test for some experimental models.

4.3. Experimental Results. The experiment in Table 2 compares the performance of single-mode and multimodal fusion for lane line segmentation in the two datasets of KITTI and A2D2. It can be seen that in the task of lane line segmentation, using only camera data has a slight advantage over using only LiDAR data, and multimodal fusion has obvious advantages over single-modal data. Lane line segmentation is a pixel classification problem. Camera data with good pixel continuity are more suitable for lane line segmentation. The data structure of the LiDAR point cloud is discrete points, and the accurate description of the edge of the lane line is not as good as the camera image. This is also an important factor for us to project the point cloud data to the camera plane for multimodal fusion. From the comparison of the experimental results of KITTI and A2D2, it can be seen that the lane line detection effect of the KITTI data is better, and the detection results reflect the data quality and the difficulty of the scene. We can see that KITTI data is more universal, therefore, in subsequent experiments, we will mainly use KITTI dataset and the data-enhenced KITTI-aug. In the experiment, we will mainly use the KITTI dataset and the data-enhanced KITTI-aug.

We have conducted extensive experiments on the KITTI dataset and KITTI-aug. As shown in Table 3, we have conducted experimental comparisons between the single-modal, multimodal direct fusion and the proposed method.

Table 2: Comparison of single-modal and multimodal fusion in KITTI and A2D2. "REC" denotes "Recall" and "PRE" denotes "Precision," and we use the same abbreviation in the following sections.

Dataset	Mode	REC	PRE	F1	F2	Acc
KITTI	LiDAR	85.43	25.82	37.84	54.41	95.19
	Camera	82.30	45.53	55.59	67.06	97.84
	Fusion	93.34	45.89	57.89	71.22	98.16
A2D2	LiDAR	84.44	22.60	35.51	48.54	94.45
	Camera	87.47	24.36	36.27	54.65	95.22
	Fusion	87.06	32.09	45.06	61.57	98.87

Table 3: Comparision of different methods in KITTI and KITTI-aug. F-MS-ECA denotes fusion with our method(multiscale efficient channel attention).

Dataset	Mode	REC	PRE	F1	F2	Acc
KITTI	LiDAR	85.43	25.82	37.84	54.41	95.19
	Camera	82.30	45.53	55.59	67.06	97.84
	Fusion	93.34	45.89	57.89	71.22	98.16
	F-MS-ECA	93.66	48.16	62.81	77.68	98.50
KITTI-aug	LiDAR	85.19	29.40	42.57	60.83	96.22
	Camera	82.44	47.65	57.31	71.19	97.95
	Fusion	92.24	48.57	62.89	77.12	98.44
	F-MS-ECA	92.23	51.38	65.25	78.57	98.65

After data enhancement, the overall test effect of the direct fusion of single modal and multimodality has been significantly improved. Among them, the F2 score of direct fusion of multimodality on KITTI-aug is 5.9% higher than that on KITTI, which shows that the used data enhancement can improve the robustness of the model. After using the proposed MS-ECA fusion method, the overall performance of the model has been further improved, and the F2 scores on KITTI and KITTI-aug have been improved by 6.46% and 1.45%, respectively. It can be seen that the precision index of the main factor of performance improvement has been significantly improved, which shows that the proposed multimodal fusion method MS-ECA can effectively reduce the false detection rate of the model to the lane line. The proposed fusion method is of great benefit to the detection of actual lane lines.

We compared our model with the current advanced models SCNN [18], LaneNet [3], and ENet-SAD [35]. All models are trained from scratch, except SCNN, and LaneNet load pretrained VGG-16 [36] weights to accelerate learning. To be fair, we train SCNN and LaneNet for 60 000 iterations (equivalent to 175 epochs). They stopped optimization after 3000 iterations. For ENet-SAD, we added the SAD strategy at the 40 000 iterations. Our model was trained for 200 epochs, and they almost converged after about 150 epochs. The experimental results are shown in Table 4. It can be seen that our model has the characteristics of lightweight and is in the same order of magnitude as the lightest ENet-SAD. Compared with the current state-of-the-art model, our model has obvious overall performance advantages and at the same time has a very high FPS, reaching 59.5 frames per second.

Table 4: Performance of different SOTA algorithms on KITTI-aug testing set.

Algorithm	Size (M)	REC	PRE	F2	Acc	FPS
LaneNet	285.7	80.97	32.81	60.97	96.87	69.1
SCNN	270.3	88.61	30.37	63.06	97.07	14.4
ENet-SAD	11.0	91.21	33.96	66.90	97.44	22.5
Ours	25.0	92.23	51.38	78.57	98.65	59.5

4.4. Ablation Study. In order to verify the contribution of each structure in the proposed method to the performance of the model, we conducted extensive ablation experiments under different backbones, and the loss curve of using ResNet34 is shown in Figure 7. As shown in Table 5, we conducted experiments on the performance of the multiscale fusion module and ECA module in the proposed method, named, F-MS and F-ECA, respectively, and the visualization results are shown in Figure 8. It can be seen that the impact of multiscale module and the ECA module on the method is mainly to improve the precision index. The multiscale module has a slight advantage in the gain of precision. When ResNet50 is used as the backbone, the gain of multiscale to precision reached 3.27%. It can be seen from the FPS that the frame rate of all models is maintained above 50, and the amount of calculation required to multiscale fusion module is greater, which leads to a more significant increase in the reasoning time of the model. Our method guarantees a high frame rate, while the overall performance of the model is excellent. It can be seen that as the network deepens, the accuracy of all models is gradually improving. When ResNet50 is used as the backbone, our method improves the precision index by 4.53% compared to the direct fusion. It is worth noting that the actual vehicle-mounted autonomous driving platform needs to carry multiple deep learning models. The deeper the network parameters, the greater the number of network parameters. Although when ResNet50 is used as the backbone, our model still has at least 50 FPS on the current test platform. In order to ensure sufficient accuracy and lightness, we still recommend using ResNet18 or ResNet34 as the backbone for actual use.

In order to verify the role of point cloud data in the lane line segmentation task, we split the information in the point cloud and fused the depth, height, and tensity with the camera image for experiments. Note that this experiment used ResNet34's pretraining parameters, and the results are shown in Table 6. It can be seen that the three types of information contribute differently to the fusion. Among them, precision is increased by 0.72 when using height information, and recall is slightly reduced when using tensity and depth information, but precision has been greatly improved with 1.42 and 1.37, respectively. This shows that the tensity and depth information in the fusion is more important than the height. It is worth noting that the tensity has a better effect in the task of lane line segmentation; however, in other fusion tasks, we suggest to pay more attention to the depth information in the point cloud, which can make up for the lack of depth information for two-dimensional images.

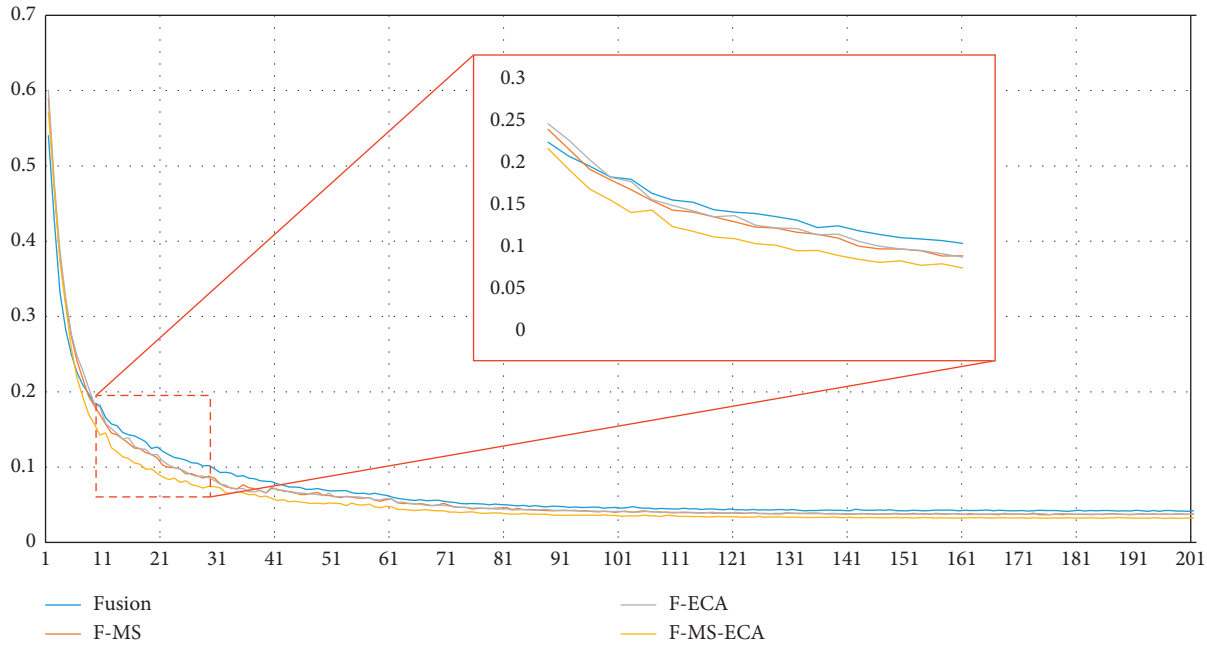

Figure 7: Loss curve during training; all training uses the pretraining parameters of ResNet34.

Table 5: The performance of using different pretrained parameters in fusion. F-MS denotes fusion only with multiscale strategy, and F-ECA denotes that fusion only with ECA.

Backbone	Method	REC	PRE	F2	Acc	FPS
	Fusion	92.24	48.57	77.12	98.44	79.0
ResNet18	F-MS	92.17	50.52	77.97	98.41	66.5
	F-ECA	92.13	50.66	78.03	98.57	78.3
	F-MS-ECA	93.08	51.21	78.34	98.62	60.2
	Fusion	92.75	49.37	78.23	98.54	75.3
ResNet34	F-MS	92.08	50.77	78.47	98.61	65.2
	F-ECA	92.12	50.36	78.46	98.61	73.9
	F-MS-ECA	92.23	51.38	78.57	98.65	59.5
	Fusion	93.09	49.84	78.53	98.58	65.9
ResNet50	F-MS	92.35	53.11	79.15	98.73	57.1
	F-ECA	92.27	52.97	78.83	98.69	62.8
	F-MS-ECA	93.17	54.37	79.72	98.81	53.6

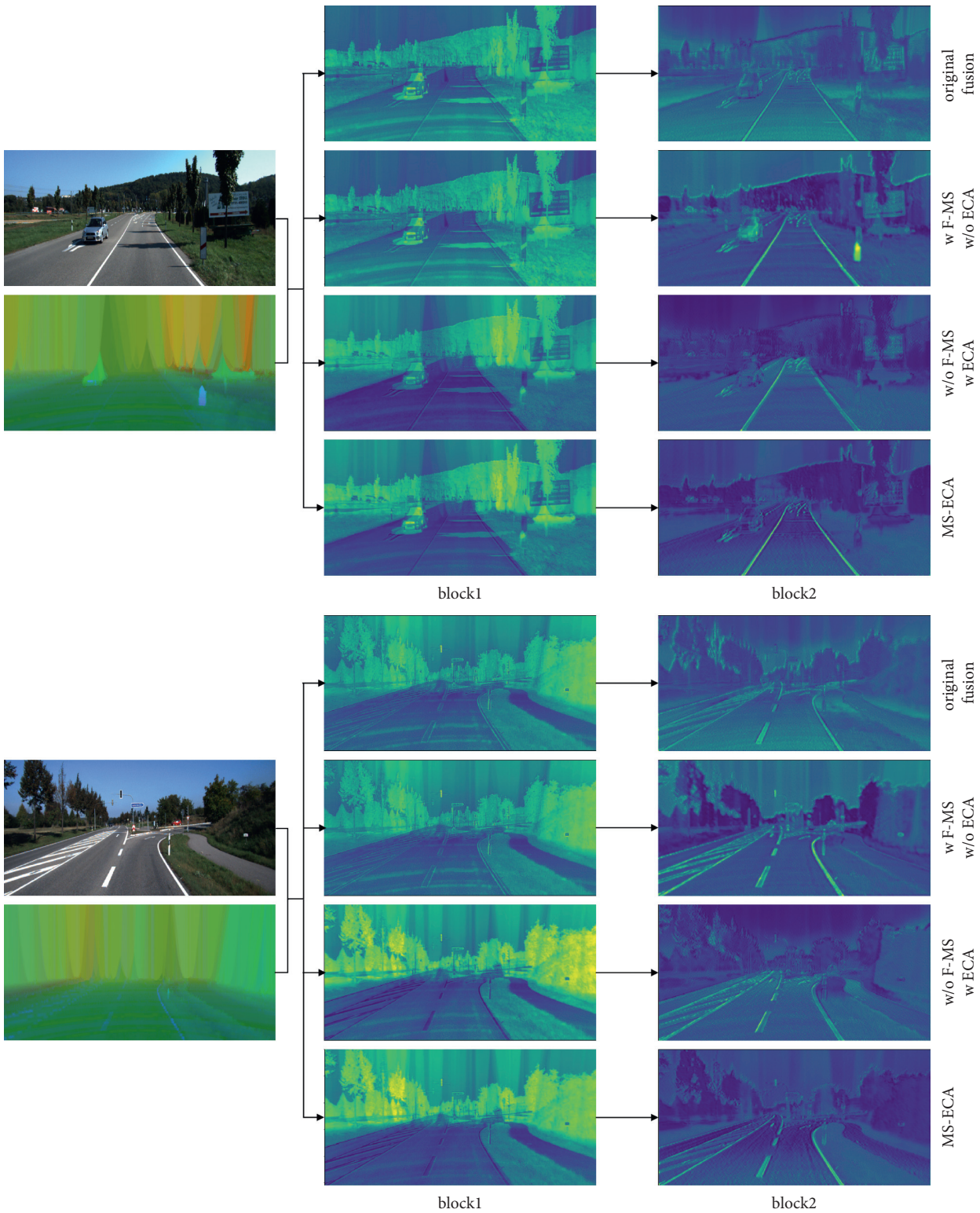

Figure 8: Visualization results of the fusion with and without MS/ECA in the feature channels of two different convolution blocks in the fusion.

Table 6: Performance comparison when only partial point cloud information is used. For example, F-depth means that only the depth information in the point cloud is used.

Method	REC	PRE	F2	Acc
Fusion	92.75	49.37	78.23	98.54
F-height	92.68	50.09	78.39	98.55
F-tensity	92.31	50.79	78.54	98.61
F-depth	92.44	50.74	78.43	98.60
F-all	92.23	51.38	78.57	98.65

5. Conclusion

This article proposes to optimize the multimodal fusion by using the multiscale fusion and ECA module for the task of lane line segmentation. By extracting features of different scales from camera images and LiDAR point clouds, and using the channel attention mechanism to calculate the weights of fusion features, we have achieved excellent results in a multimodal fusion network. In the test on the KITTI-aug dataset, we obtained the best performance model when using ResNet50 as the backbone, with the highest F2 score of 79.72%. At the same time, our method can maintain excellent test speed in actual tests. The structural difference between the modalities is one of the main problems that make the current multimodal fusion difficult. In the future, we will explore the fusion of different modalities in high-dimensional space and analyze the differences and differences between the modalities from the structure of the data and achieve more robust fusion.

Data Availability

All the data generated or analyzed during this study are included within this article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This research was funded by the Natural Science Foundation of Shanxi Province under Grant No. 201901D111467.

References

[1] N. Garnett, R. Cohen, T. Pe'er, R. Lahav, and D. Levi, "3d-lanenet: end-to-end 3d multiple lane detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2921–2930, Seoul, Korea (South), November 2019.

[2] J. Zhang, Yi Xu, B. Ni, and Z. Duan, "Geometric constrained joint lane segmentation and lane boundary detection," in *Proceedings of the european conference on computer vision (ECCV)*, pp. 486–502, Munich, Germany, October 2018.

[3] Ze Wang, W. Ren, and Q. Qiu, "Lanenet: real-time lane detection networks for autonomous driving," https://arxiv.org/abs/1807.01726.

[4] X. Xu, G. Li, G. Xie, J. Ren, and X. Xie, "Weakly supervised deep semantic segmentation using cnn and elm with semantic candidate regions," *Complexity*, vol. 2019, 2019.

[5] An Feng-Ping and J.-e Liu, "Medical image segmentation algorithm based on optimized convolutional neural network-adaptive dropout depth calculation," *Complexity*, vol. 2020, 2020.

[6] R. Yin, Y. Cheng, H. Wu, Y. Song, B. Yu, and R. Niu, "Fusionlane: multi-sensor fusion for lane marking semantic segmentation using deep neural networks," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2020.

[7] L. Tong, Z. Chen, Yi Yang, Z. Wu, and H. Li, "Lane detection in low-light conditions using an efficient data enhancement: light conditions style transfer," in *Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1394–1399, IEEE, Las Vegas, NV, USA, November 2020.

[8] E. Arnold, O. Y. Al-Jarrah, M. Dianati, S. Fallah, D. Oxtoby, and A. Mouzakitis, "A survey on 3d object detection methods for autonomous driving applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 10, pp. 3782–3795, 2019.

[9] Y. Cui, R. Chen, W. Chu et al., "Deep learning for image and point cloud fusion in autonomous driving: a review," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–18, 2021.

[10] K. Zhu, R. Guo, W. Hu, Z. Li, and Y. Li, "Legal judgment prediction based on multiclass information fusion," *Complexity*, vol. 2020, 2020.

[11] S. Gu, Y. Zhang, J. Tang, J. Yang, and H. Kong, "Road detection through crf based lidar-camera fusion," in *Proceedings of the 2019 International Conference on Robotics and Automation (ICRA)*, pp. 3832–3838, IEEE, Montreal, QC, Canada, May 2019.

[12] L. Caltagirone, M. Bellone, L. Svensson, and M. Wahde, "LIDAR-camera fusion for road detection using fully convolutional neural networks," *Robotics and Autonomous Systems*, vol. 111, pp. 125–131, 2019.

[13] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "Eca-net: efficient channel attention for deep convolutional neural networks," in *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, June 2020.

[14] P. M. Daigavane and P. R. Bajaj, "Road lane detection with improved canny edges using ant colony optimization," in *Proceedings of the 2010 3rd International Conference on Emerging Trends in Engineering and Technology*, pp. 76–80, IEEE, Goa, India, November 2010.

[15] W. Van Gansbeke, B. De Brabandere, D. Neven, M. Proesmans, and L. Van Gool, "End-to-end lane detection through differentiable least-squares fitting," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, Seoul, Korea (South), October 2019.

[16] Y. Ko, Y. Lee, S. Azam, F. Munir, M. Jeon, and W. Pedrycz, "Key points estimation and point instance segmentation approach for lane detection," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2021.

[17] W. Song, Y. Yang, M. Fu, Y. Li, and M. Wang, "Lane detection and classification for forward collision warning system based on stereo vision," *IEEE Sensors Journal*, vol. 18, no. 12, pp. 5151–5163, 2018.

[18] X. Pan, J. Shi, P. Luo, X. Wang, and X. Tang, "Spatial as deep: spatial cnn for traffic scene understanding," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, Louisiana, USA, February 2018.

[19] B. He, Ai Rui, Y. Yang, and X. Lang, "Accurate and robust lane detection based on dual-view convolutional neutral network," in *Proceedings of the 2016 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1041–1046, IEEE, Gothenburg, Sweden, June 2016.

[20] H. Cho, Y.-W. Seo, B. V. K. Vijaya Kumar, and R. Raj Rajkumar, "A multi-sensor fusion system for moving object detection and tracking in urban driving environments," in *Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1836–1843, IEEE, Hong Kong, China, June 2014.

[21] Ma Han, Y. Ma, J. Jiao et al., "Multiple lane detection algorithm based on optimised dense disparity map estimation," in *Proceedings of the 2018 IEEE International Conference on Imaging Systems and Techniques (IST)*, pp. 1–5, IEEE, Krakow, Poland, October 2018.

[22] A. Eitel, J. Tobias Springenberg, L. Spinello, M. Riedmiller, and W. Burgard, "Multimodal deep learning for robust rgb-d object recognition," in *Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 681–687, IEEE, Hamburg, Germany, October 2015.

[23] H. Wang, F. Zhou, W. Zhou, and L. Chen, "Human pose recognition based on depth image multifeature fusion," *Complexity*, vol. 2018, 2018.

[24] Joel Schlosser, C. K. Chow, and Z. Kira, "Fusing lidar and images for pedestrian detection using convolutional neural networks," in *Proceedings of the2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2198–2205, IEEE, Stockholm, Sweden, May 2016.

[25] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," in *Advances in Neural Information Processing Systems*, pp. 5998–6008, MIT Press, Cambridge, MA, 2017.

[26] J. Fu, H. Zheng, and T. Mei, "Look closer to see better: recurrent attention convolutional neural network for fine-grained image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4438–4446, Honolulu, HI, USA, May 2017.

[27] X. Cheng, H. Wang, J. Zhou, H. Chang, X. Zhao, and Y. Jia, "Dtfa-net: dynamic and texture features fusion attention network for face antispoofing," *Complexity*, vol. 2020, 2020.

[28] F. Wang, M. Jiang, C. Qian et al., "Residual attention network for image classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3156–3164, Honolulu, HI, USA, July 2017.

[29] L.-C. Chen, Yi Yang, J. Wang, W. Xu, and A. L. Yuille, "Attention to scale: scale-aware semantic image segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3640–3649, Las Vegas, NV, USA, June 2016.

[30] O. Ronneberger, P. Fischer, and T. Brox, "U-net: convolutional networks for biomedical image segmentation," in *Proceedings of the International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Springer, Munich, Germany, May 2015.

[31] Li Xiang, W. Wang, X. Hu, and J. Yang, "Selective kernel networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 510–519, Long Beach, CA, USA, June 2019.

[32] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Proceedings of the 2012 IEEE conference on computer vision and pattern recognition*, pp. 3354–3361, IEEE, Providence, RI, USA, June 2012.

[33] J. Geyer, Y. Kassahun, M. Mahmudi et al., "A2d2: audi autonomous driving dataset," 2020, https://arxiv.org/abs/2004.06320.

[34] D. P. Kingma and Ba Jimmy, "Adam: a method for stochastic optimization," 2014, https://arxiv.org/abs/1412.6980.

[35] Y. Hou, M. Zheng, C. Liu, and C. C. Loy, "Learning lightweight lane detection cnns by self attention distillation," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1013–1021, Seoul, Korea (South), November 2019.

[36] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, https://arxiv.org/abs/1409.1556.