ABSSNet: Attention Based Spatial Segmentation Network for Traffic Scene Understanding

Xuelong Li, *Fellow, IEEE,* Zhiyuan Zhao, and Qi Wang, *Member, IEEE*

***Abstract*—The location information of road and lane lines is the supremely important thing for the automatic drive and auxiliary drive. The detection accuracy of these two elements dramatically affects the reliability and practicality of the whole system. In real applications, the traffic scene can be very complicated, which makes it particularly challenging to get the precise location of road and lane lines. Commonly used deep-learning-based object detection models perform pretty well on the lane line and road detection tasks, but they still encounter false detection and missing detection frequently. Besides, existing Convolution Neural Network (CNN) structures only pay attention to the information flow between layers, while it can not fully utilize the spatial information inside the layers. To address those problems, we propose an attention-based spatial segmentation network for traffic scene understanding. We use the convolutional attention module to improve the network's understanding capacity of spatial location distribution. Spatial Convolution Neural Network (SCNN) gets through the information flow within one single convolutional layer and improves the spatial relationship modeling ability of the network. Experimental results demonstrate that this method effectively improves the neural network's application ability of the spatial information, thereby improving the effect of traffic scene understanding. Furthermore, a pixel-level road segmentation dataset named NWPU Road Dataset is built to help improve the process of traffic scene understanding.**

***Index Terms*—Traffic scenes understanding, Spatial Convolution Neural Networks, Attention Model, Road Detection, Lane Lines Detection.**

I. Introduction

TRAFFIC accident brings enormous damage to life and property. The development of social science and technology has brought the possibility of using an auxiliary method to avoid accidents. Autopilot and assisted driving are two common ways to help avoid accidents. The development of these technologies is inseparable from traffic scene understanding assignments, which includes computer vision tasks such as lane detection [1], [2], road detection [3], or road marking detection [4]. All those three tasks can be helpful for navigation of vehicles in autopilot and assisted driving systems [5], [6].

However, in real applications, considering the complexity of traffic scenarios, these tasks can become challenging. As shown in Fig. 1, changing weather, dazzle lighting conditions, and complex road scenes cause significant challenges for understanding the scene, especially in lane and road detection task. The challenges can be summarized as follows:

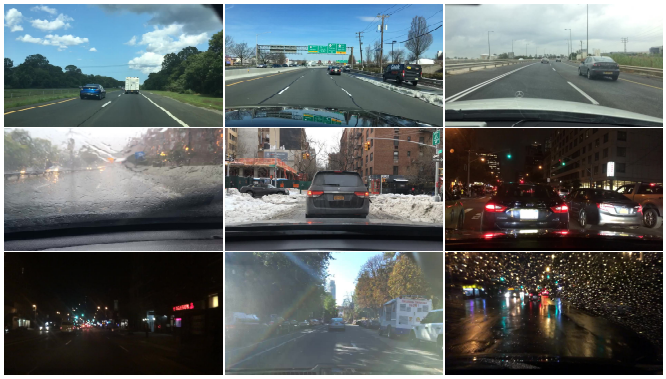

Fig. 1. This figure shows several examples of traffic scenes under different external conditions. The first row shows three simple scenes, while it is more involved in the second and third.

- Lots of factors could reduce the visibility of lane lines and roads, such as complicated weather, disappointing light conditions, occlusion of pedestrians or vehicles, and the wear of lane lines.
- Without regard to visibility, lane lines have structural features of elongation and continuity, but it will be truncated into something with entirely different characteristics when occlusion occurs.
- Even in the same scene, for a particular lane line, the characteristics it exhibits at the near end, and the far end are quite different. In this case, proximal and distal targets appear on imparity scales, so it is not easy to use a straightforward detection model to detect the whole lane line.

It is hard to overcome these challenges with simple image processing or traditional machine learning methods. In recent years, deep features extracted by deep neural networks show strong representational ability in image classification, object detection, and other tasks in the computer vision field. It performs much better than traditional handcrafted feature-based machine learning methods among lots of tasks.

Although lane lines are artificially designed with specific shapes, it is still hard to design effective patterns to match all complex scenes. To get rid of those influence factors, some researchers use convolutional neural networks (CNNs) to extract high order semantic information, and then use those features to detect lane lines directly. However, CNN's operation principle

This work was supported by the National Key Research and Development Program of China under Grant 2018AAA0102200 and the National Natural Science Foundation of China under Grant 61871470, U1801262, U1864204 and 61773316.

X. Li, Z. Zhao and Q. Wang are with the School of Computer Science and the Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an 710072, China (e-mail: li@nwpu.edu.cn; tuzixini@gmail.com; crabwq@gmail.com).

Q. Wang is the corresponding author.

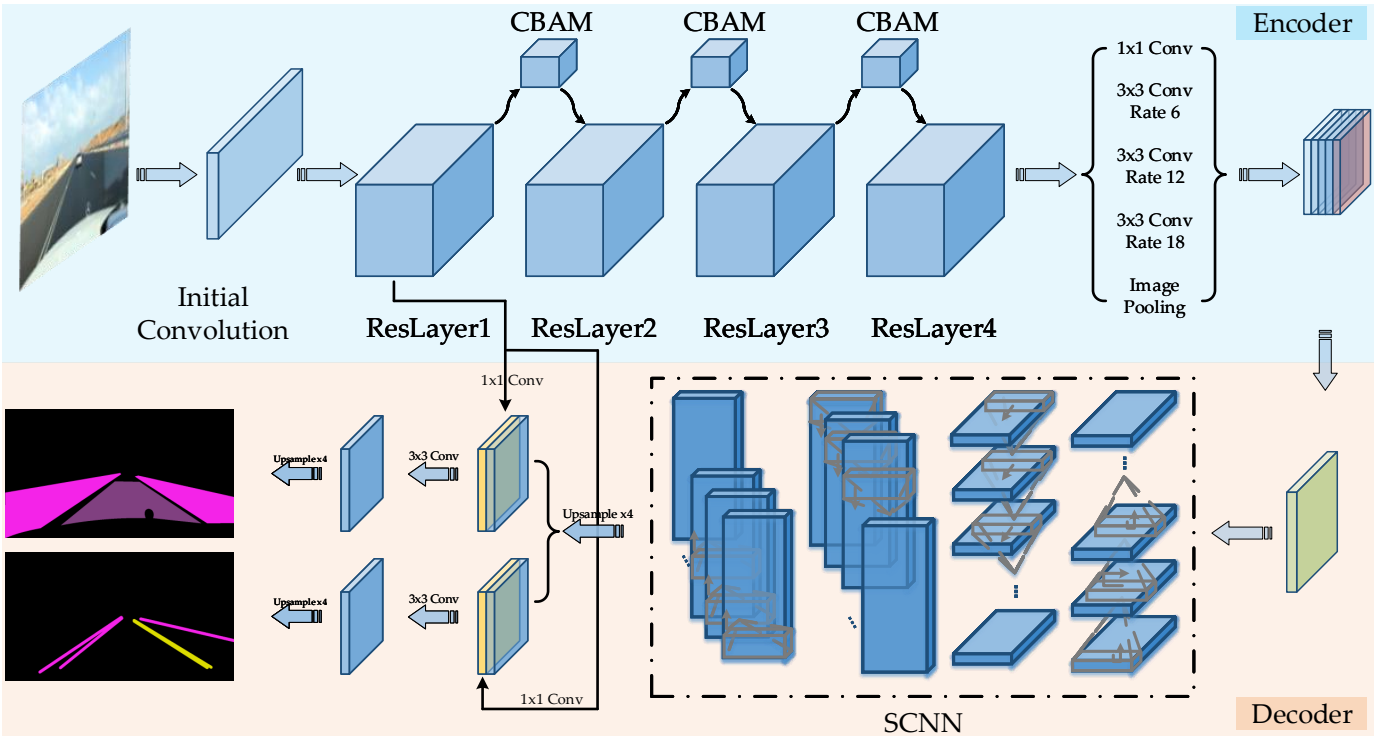

Fig. 2. The schematic diagram of the proposed attention-based spatial segmentation network. Firstly, given an input traffic scene image, the initial convolution layer generates low_level convolutional features. Then the features pass through dilated ResLayers with CBAM attention modules. The ASPP structure uses different dilatation coefficients to fuse multi-scale features and extends the horizon of the model. Then, a spatial convolutional network is presented at the top of the decoder to parse the spatial information of the features. Finally, multiple decoders are utilized to complete the segmentation target of the road and lane line.

and the down-sampling layers in it lead to the loss of spatial information when extracting features. For lane lines that are easily occluded and have large scale changes, spatial location information becomes vital. Therefore, it is necessary to make some changes to the existing network structure to improve the network's perception of spatial information and further increase the accuracy of lane detection.

Owing to the above reasons, our framework focuses on how to extract features with sufficient spatial information to improve lane line detection and road detection performance.

A. Overview of Our Approach

In this paper, we propose Attention Based Spatial Segmentation Network for Traffic Scene Understanding (ABSSNet) to conduct traffic scene understanding from the perspective of image semantic segmentation and make better use of spatial information. As shown in Fig. 2, DeepLabV3+ is used as the backbone, while lane line and road detection, spatial attention mechanism is leveraged in the convolutional layer and the final output layer, which effectively improves the detection accuracy of networks.

1) Encoder with convolution attention modules : For the input traffic scene images, the encoder aims to extract powerful features. In this stage, a dilated ResNet is used as the backbone, and the initial convolution module remains unchanged, while convolutional attention mechanisms are inserted into different positions of Res-Blocks. The attention structure assigns different weights to different spatial locations of features, which forces the model to have different concerns in different places. Thus the attention-based convolutional encoder can learn more rich features to describe the spatial location information of lane lines and roads.

2) Multi-task decoder with Spatial Convolutional Neural Network : The decoder is used to construct the mapping between the feature vectors encoded by the encoder and the desired segmentation results. The spatial convolution neural network is added to the top level of the decoder to analyze and model the spatial structure information inside the input feature vectors. The information flow in four directions within the feature vector effectively improves the analysis and understanding ability of the decoder. Besides, the decoder of DeepLabV3+ is upgraded to a multi-task version. The two sub-decoding modules share one SCNN spatial modeling process and then perform up-sampling and convolutional decoding independently. One branch focuses on the lane line decoding, and the other focuses on road decoding. The results of road segmentation and lane segmentation will be output independently.

B. Contributions

Different from previous methods, ABSSNet implements lane line detection and road detection through segmentation. A robust semantic segmentation network provides the possibility of more sophisticated detection in complex traffic scenes. Rich high dimensional features can be extracted due to the excellent representation ability of Deep Convolutional Neural Networks

(DCNN). Here, the lane line detector and road detector share one specific DCNN encoder. By removing excess interference class from the target list, the segmentation accuracy of the interested class can be improved effectively. Therefore, the detection stage consists of two single-target detectors, which depend on different decoders. In summary, the encoder encodes the input image into convolutional features, and the two decoders decode the same feature to get different output results.

To further enhance the image spatial information extraction ability of the entire model, the convolution attention mechanism is included in the existing encoder structure. Through employ the attention mechanism, different parts of the feature obtain different weights, so that the network can better focus on the part we care about, instead of treating the entire image feature equally. Also, we introduce the spatial convolutional layer at the top of the decoder. This layer drives information flow inside the extracted features in different directions. The modeling and extraction ability of the encoder and decoder is improved by combining these two structures.

A road segmentation dataset containing 3,949 images (extracted from 11 traffic scenes videos) with accurate pixel-level labels is collected and finally released by this paper. This dataset contains a verify of fluent or congested traffic scenes. Both urban and highway roads, simple or complex scenes are accommodated inside the dataset. This high-resolution road segmentation dataset will improve the current road segmentation method.

The following parts of this paper are organized as follows: Section II focuses on review related works. The NWPU Road Dataset will be introduced in SectionIV. Detailed implementation of the framework is given in Section III. Section V evaluates the performance of the proposed method. Finally, we conclude the proposed method and present future works in Section VI.

II. Related Works

This section focuses on introducing some work related to the proposed method. This paper covers several technology fields, including lane line detection, road detection, semantic segmentation, attention mechanism, and spatial information modeling methods. Some related works from different perspectives are introduced here.

A. Lane line detection and road detection methods.

Many traditional machine learning and image processing methods do lane detection base on handcrafted low-level features [7], [8]. These methods are quite useful in simple traffic scenes, while in complex and changeable conditions, their effectiveness decreases significantly. Huval *et al.* [9] make the first attempt to use the neural networks to tackle lane detection problems. However, they did not have relatively large datasets for network training, so the final result is not satisfactory. Tao *et al.* [10] also try to use the deep learning method in lane line detection and discuss the importance of spatial information in the convolutional neural network. As for road detection He *et al.* [11] firstly estimate the boundaries based on the intensity image, then road areas are subsequently detected base on the full-color image. Kong *et al.* [12] propose a method attempts to estimate the vanishing point associate with the central part of the road followed by the segmentation of the corresponding road area upon the detected vanishing point.

PASCAL VOC dataset [13] proposed by Everingham *et al.* publish a publicly available dataset of images and annotations, together with standardized evaluation software. Fritsch *et al.* [14] release a novel open-access dataset and benchmark for road area and ego-lane detection, which contains 600 annotated training and test images of high variability from the KITTI autonomous driving project. Yu *et al.* [15] build a new driving dataset comprised of over 100K videos with diverse annotations including image-level tagging, object bounding boxes, drivable areas, lane markings, and full-frame instance segmentation. The emergence of all these large datasets provides a solid foundation for data-driven machine learning methods, thereby promoting the development and progress of a large number of deep learning methods.

B. Image semantic segmentation.

Image segmentation is an old but meaningful computer vision problem. There are many related studies even before 1985. Haralick, Robert M, and Shapiro, Linda G [16] summarize major classes of image segmentation techniques and give several algorithms corresponding to each class. Shi and Malik [17] propose the normalized cuts aim to extract the global impression of an image and realize the image segmentation task. Image segmentation methods using mathematical morphology based on watershed transform and homotopy modification are presented by Meyer *et al.* [18]. Grady [19] performs multi-label, interactive image segmentation by random walks. Those methods mentioned above are relatively efficient image segmentation algorithms when they are first presented. However, it is much worse than the current deep learning-based segmentation method.

Since the Fully Convolution Network (FCN) [20], more and more researchers have been inspired to apply a neural network to image segmentation tasks. Noh *et al.* [21] propose the deconvolution network composed of deconvolution and unpooling layers used for image semantic segmentation. Then SegNet [22] introduces the encoder/decoder structure into the segmentation framework. Meanwhile, U-Net [23] tries to do segmentation in the case of small data set size, and Kai *et al.* [24] combine shape information and convolution features to train the classifier and speed up the segmentation. Wang *et al.* [25] propose a joint method of priori convolutional neural networks at the super-pixel level and soft restricted context transfer. DeepLab [26]–[29] series effectively improves the segmentation accuracy step by step. So far, image segmentation methods based on deep learning have made great progress.

Due to the nature of the image segmentation task, the training process of neural network models requires a large amount of pixel-level annotated data. That is, each pixel in the image needs to be manually labeled with its corresponding category. Labeling such data is extremely time-consuming

for high-quality images. In order to reduce the cost of data, researchers make efforts from different angles. Richter *et al.* [30] and Ros *et al.* [31] construct the synthetic scene image semantic segmentation datasets using different software respectively. Since it is a computer-generated virtual picture, the category corresponding to each pixel in the figure can be directly generated by the computer, which saves a lot of labeling work. Besides synthetic datasets, Wang *et al.* [32] and Li *et al.* [33]–[35] propose weakly supervised and few-shot methods to reduce data dependence of models.

C. Attention mechanism.

Demitry *et al.* [36] propose the attention concepts for the first time, and the experimental results show that the employ of the attention mechanism is quite effective in improving the neural network's understanding of spatial information. Although attention mechanism is initially devoted to machine translation for processing text sequences [36], [37], just like other feature embedding methods [38], [39], currently it is extended to various fields including image Q&A [40], speech recognition [41], image captioning [42], and image classification [43], [44] *et al.*

III. Attention Based Spatial Segmentation Network

As shown in Fig. 2, ABSSNet is built above DeepLab V3+ [29] and consists of two key improvements: convolutional attention block inside the encoder backbone and spatial CNN among multi-task decoder. In this section, we firstly have a quick description of the DeepLab V3+ baseline and then introduce the above two improvements in detail separately.

A. Baseline:DeepLab V3+

Since DeepLab V1 [26], four versions of DeepLab [27]–[29] have been proposed. Different adaptation of DeepLab has quite distinct architecture. Here we are only going to cover the fourth version (more detailed information can be found through the references). DeepLab V3+ is composed of an encoder-decoder structure. The encoder contains atrous convolutional feature extractor and Atrous Spatial Pyramid Pooling (ASPP) operations. Once the complete image is inputted, the atrous convolution extracts CNN features, and ASPP tries to integrate the deep features through different scales. Then the encoder outputs the encoded representation of the input image and a low-level feature separated from the primary layers. Thus, the decoder is carried out with features extracted by the encoder. Simple bilinear upsampling and convolution are used to make the pixel-wise classification. At the same time, the feature is recovered to the original image size. The entire DeepLab V3+ framework can be end-to-end trained with cross-entropy loss. For clear representation, we use ADCNN, ASPP, $\mathscr{F}_{level}$ represent atrous Deep Convolutional Neural Network, Atrous Spatial Pyramid Pooling, and different levels of features. The whole segmentation structure can be described as:

$$\mathscr{F}_{ll}, \mathscr{F}_{ADCNN} = ADCNN(img), \tag{1}$$

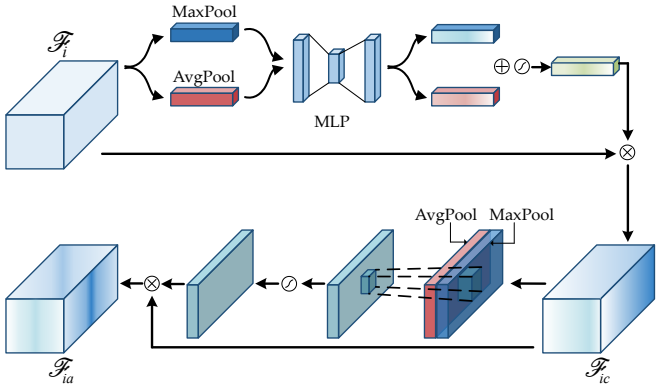

Fig. 3. The process diagrams of channel attention and spatial attention operation in series. Cuboids represent various feature vectors. The blue cuboid comes from MaxPooling, and the red one comes from AveragePooling. The different shapes result from operations in different directions.

$$\mathscr{F}_{ASPP} = ASPP(\mathscr{F}_{ADCNN}), \tag{2}$$

$$ScoreMap = Decoder(\mathscr{F}_{ll}, \mathscr{F}_{ASPP}), \tag{3}$$

where img is the input image, $\mathscr{F}_{ll}$ means low-level features separated from ADCNN's primary layer, and $\mathscr{F}_{ADCNN}$ represents the output of ADCNN, $\mathscr{F}_{ASPP}$ is the feature extracted by ASPP. The output of the whole framework is called the score map, which is a single channel figure of the same size as the input image, and each pixel contains the value of the category it belongs to. Eq. (1) and Eq. (2) together form the encoder. Here we mainly make improvements to the structure distributed in Eq. (1) and Eq. (3).

B. Multi-layer Attention Convolution

Unlike general object detection, lane and road detection need to resolve the information contained in near and far scenarios. Thus spatial clue becomes extremely important. Although the atrous convolution in the original framework has been an exploration for the utilization of spatial information, we still try to use other methods to further improve the understanding ability of the neural network for spatial information. The attention mechanism makes neurons in different locations have various responses to the input. Usually, the attention mechanism is a two-dimensional or even one-dimensional operation. To apply it inside convolutional neural networks, the three-dimensional attention mechanism CBAM [44] appears. It means to apply attention to both spatial and channel degrees. Simultaneously, considering its ease of use in other network structures, it is designed as a plug-in light-weighted module with the same input and output shape.

As shown in Fig. 3, CBAM has two kinds of modes, that is channel attention and spatial attention. Two modes can be used alone or together according to the application requirements. Here we use two CBAM modes in series, which means for an input feature $\mathscr{F}_i$, it will first pass through channel attention structure and get $\mathscr{F}_{ic}$. Then the spatial attention CBAM will continue to focus on spatial information and generate $\mathscr{F}_{ia}$, which represents attention-based processed features. The detailed working process can be expressed by the following formulas:

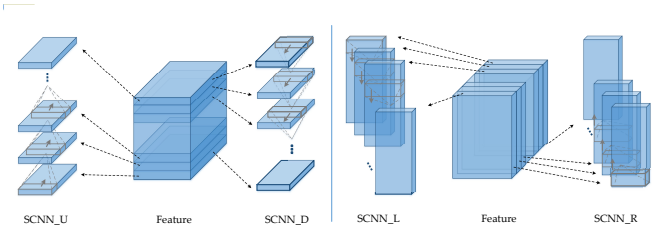

Fig. 4. Four kinds of SCNNs in different directions. The input features can be divided into many slices with different indications.

$$\mathscr{F}_{ic} = \mathscr{F}_i \otimes (\sigma(MLP(AP(\mathscr{F}_i)) + MLP(MP(\mathscr{F}_i))), \quad (4)$$

$$\mathscr{F}_{ia} = \mathscr{F}_{ic} \otimes (\sigma(f^{7\times7}([AP(\mathscr{F}_{ic}); MP(\mathscr{F}_{ic})]))), \quad (5)$$

where $\mathscr{F}_{ic}$ means intermediate channel attention features inside each CBAM block, $\otimes$ denotes element-wise multiplication, σ is the sigmoid function, $f^{7\times7}$ represents a convolution operation with kernel size 7×7 and MLP, AP, MP represent multi-layer perceptron, average pooling, max pooling operation respectively.

Due to the shape keep characteristic (input and output have the same shape) of CBAM, it can be inserted into anywhere of the networks easily. Different quantities and positions led to various combinations with distinct modeling abilities. The detailed analysis and experiments will be shown in Section V.

C. Spatial Convolution Neural Networks

Since lane and road detection requires precise prediction of specific curves. It is quite natural and effective to use semantic segmentation models to generate probability maps of input images, and then separate lane lines and roads from the probability maps. In order to obtain accurate probability maps, spatial relationship analysis becomes noteworthy. The traditional convolutional neural network only transmits information between layers. Thus there is no flow of massage inside the convolutional layer. Those methods always modeling spatial relationships based on Markov Random Fields (MRF) or Conditional Random Fields [45] (CRF), which means simply connect an MRF or CRF on top of a DCNN. Such methods work, but not sufficient enough. SCNN proposed by Pan *et al.* [46] generalizes traditional deep layer-to-layer convolutions to slice-by-slice convolutions within feature maps, thus enabling message passing between pixels across rows and columns in a layer. This internal flow of information helps a lot in modeling spatial relationships, finally contributes to lane and road detection.

As shown in Fig. 4, there are four kinds of SCNN operations, the suffix 'D,' 'U,' 'R,' 'L,' who donate SCNN towards down, up, right, left direction respectively. The detailed implementation of information flow inside the feature vector is introduced below. Firstly, select the flow direction (here we use SCNN_U as an example). Then, slice the input feature vector along the selected direction, a small convolution is applied to the bottom wafer, and convolution results will be added to the upper one. After that, the same convolution is applied to the second slice then add to the third slice. Repeat this operation until it reaches the top one. Finally, information flows from the bottom to the top, which means all slices contain information from all previous slices. As for the other three kinds of SCNNs, it is just operating in different directions. We apply $SCNN\_D, U, L, R$ sequentially to the encoder's output $\mathscr{F}_{ASPP}$ and get the refined features, which is specially designed for the spatial relationship information and improves the performance of traffic scene understanding.

D. Attention Based Spatial Segmentation Network

Considering the spatial modeling ability of CBAM and SCNN, this approach tries to combine two flexible structures with a powerful semantic segmentation network to achieve traffic scene understanding. Firstly, the atrous deep convolutional neural network integrates with CBAMs is used to extract convolutional features from the original input images. Then the atrous spatial pyramid pooling is used to reconstruct new features with different scaling levels. After passing through the encoder, the encoded features are sent to SCNNs with four different directions for the sake of internal information flow within the high-dimensional feature tensors. When the internal information flow finishes, two separate multi-task decodes share the same feature vector and decode different score maps for roads and lane lines.

As shown in Fig. 2, according to experimental results, the final full model chooses to use four CBAMs right after four ResLayers and integrate four directions of SCNNs to the decoder. This combination achieves relatively good results.

IV. A Road Segmentation Dataset: NWPU Road Dataset

Besides, we build a simple single task dataset that aims to improve road segmentation methods. This dataset contains 11 videos, and each video lasts 3 minutes with a frame rate of 30 fps. All videos are shot by one specific automobile data recorder with a size of 1280x720 and a resolution of 96 dpi. The dataset includes a variety of real driving scenes. It is impossible and also meaningless to label all frames in the videos. So we divide the videos into discontinued image sequences. Through sampling images every 15 frames, each video can be split into 359 images, which means extract the video frames each half-second.

Totally $11 \times 359 = 3,949$ images are generated from the 11 videos. All images are labeled using a web-based annotation tool LabelMe [47]. Multiple points are drawn manually and then joined together to form polygons encircle the labeled road area. Some examples of the final label results are shown in Fig. 5.

V. Experiments

We validate the effectiveness of the proposed ABSSNet for road and lane detection by comparing with the bottleneck or partial improvement version and other popular traffic scene understanding methods on related datasets.

Section V-A introduces the datasets and our modification on it. Section V-B shows the experiment's implementation details.

Fig. 5. This figure shows different traffic scene samples from the released dataset and their corresponding visualized road masks. The dataset contains a variety of driving scenes. The first row demonstrates a relative emptiness situation, while the third row shows congestion scenes.

Then the results and ablations studies are given in Section V-C. Section V-D try to discover the internal working principle of CBAM and SCNN.

A. Datasets and Protocols

BDD100K [15] is a newly constructed large-scale driving dataset. It contains over 100K videos with various annotations, including lane markings, road areas, object bounding boxes, and full-frame instance segmentation *et al.*.

Since here we mainly focus on lane lines and road detection, some trade-offs have been made to the BDD100K dataset. Considering the actual functionality of the road area and following the labeling method of BDD100K, the road area (so-called drivable area) is divided into two different categories: "directly drivable area" and "alternatively drivable area". The former one means the area that the driver is currently driving on, and the latter one is a lane the driver is currently not driven on but could do so through changing lanes.

The original BDD100K's lane markings are annotated by polygon areas and classified into eight main categories with different attributes of continuity and direction. In order to fit semantic segmentation networks, we reconstruct lane marking annotation data using our tools and following the new rules. The new data has pixel-level annotation, and the corresponding label convert rules are shown in Tab. I.

Finally, we get a modified version of BDD100K called BDD100K_M. All our train and test process are based on this variant. The dataset contains 70000 training images, 10000 validation images, and 20000 test images.

B. Implementation Details

We use ResNet152 [48] as the encoder's backbone network with dilate rate [1, 1, 1, 2] for four Res-Layers. The whole network can be end-to-end trained with Adam optimizer and cross-entropy loss function. Part of the parameters is initialized by the ImageNet [49] pre-trained model. The optimizer is created with an initial learning rate 0.0001, weight decay of 0.005, and we manually adjust the learning rate every epoch according to the following formula:

$$LR = LR \times (1 - \frac{1 \times epoch}{MAX\_EPOCH}) \times 0.9, \tag{6}$$

where $epoch$ is the serial number of the current one, and MAX_EPOCH means the upper limit of the epochs, which we set to 100 here. All training process are implemented with three NVIDIA 1080Ti GPU, Intel(R) Core(TM) i7-6800K CPU @ 3.4GHz and input batch size 32. All code is developed by python language, based on the Pytorch [50] framework.

In the encoder part, each attention module consists of a channel CBAM and a spatial CBAM in series. An attention module is placed between every two Res-Layers. For the decoder, the spatial convolutional neural network is placed behind the encoder, which is the top layer of the decoder. SCNNs in four different directions are used serially. Two simple convolutional multi-task decoders share the same spatial feature vector and focus on segment road and lane lines while restoring image resolution at the same time.

TABLE I
LABEL CONVERT RULES. FROM ORIGINAL BDD100K TO BDD100K_M

Original Labels	New Labels
double white - parallel - solid	white - solid
single white - parallel - solid	white - solid
double white - parallel - dashed	white - dashed
single white - parallel - dashed	white - dashed
double yellow - parallel - solid	yellow - solid
single yellow - parallel - solid	yellow - solid
double yellow - parallel - dashed	yellow - dashed
single yellow - parallel - dashed	yellow - dashed
all others	background

C. Evaluation Results and Ablations Studies

We firstly train the original DeepLab V3+ as the baseline on the modified BDD100K datasets (Of course, simple adjustments are made, such as modifying the number of prediction categories and using one encoder with two decoders to achieve the effect of multi-task segmentation). Then we combine baseline with CBAM and SCNN separately, test and analyze how these two parts bring improvements to the whole model. Finally, evaluate the proposed Attention Based Spatial Segmentation Network. As for the measure of model effect, just like the common segmentation measurements, according to the evaluation criterion of Pascal VOC [13], detection accuracy is measured by Intersection over Union(IoU) and Mean Intersection over Union(mIoU) defined by formula 7 and 8:

$$IoU = \frac{p_{ij}}{\sum_{j=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ij} - p_{ii}}, \tag{7}$$

$$mIoU = \frac{1}{k+1} \sum_{i=0}^{k} IoU, \tag{8}$$

where k is the number of significative labels (Usually, a zero label means a useless or unlabeled point). and p_{ij} represent

TABLE II
DETAIL CLASS IOU OF THE TEST RESULTS, COMPARE DIFFERENT STEPWISE MODELS (THE BASELINE DEEPLABV3+, BASELINE WITH CBAM, BASELINE WITH SCNN AND FULL MODEL) ON MODIFIED BDD100K DATASET'S VAL SET AND NWPU ROAD (IN %).

Method	Road_Now	Road_Drivable	Lane Line_ws	Lane Line_wd	Lane Line_ys	Lane Line_yd	NWPU Road
DeepLabV3+	80.21	64.72	43.78	46.53	50.92	23.17	81.14
DeepLabV3+ & CBAM	81.98	69.00	44.75	48.11	53.22	25.81	83.57
DeepLabV3+ & SCNN	81.44	66.78	44.27	45.81	52.07	25.78	82.39
Full Model	**82.73**	**70.13**	**45.99**	**48.78**	**55.80**	**28.10**	**85.36**

TABLE III
COMPARISON OF DIFFERENT STEPWISE MODELS (THE BASELINE DEEPLABV3+, BASELINE WITH CBAM, BASELINE WITH SCNN AND FULL MODEL) ON MODIFIED BDD100K DATASET'S VAL SET (IN %).

Method	Road mIoU	Lane Lines mIoU
DeepLabV3+	79.80	52.51
DeepLabV3+ & CBAM	81.04	54.02
DeepLabV3+ & SCNN	81.12	53.24
Full Model	**82.71**	**55.38**

for one point p, its real label is i, and the predicted result is j. If i equals j, that means for this specific point, the model gets the right prediction results. Otherwise, it gives the wrong predictions. This means we can also express IoU by equation 9:

$$IoU = \frac{TP}{FN + FP + TP}, \tag{9}$$

where TP means the number of points which we get right predictions, FN and FP contain all points with different predicted results and labels.

As shown in Tab. III, by contrast, only using baseline DeepLabV3+ alone has the worst effect. The addition of two different modules improves the overall effect of the model, while the combination of the two parts brings comprehensive improvement, and the final complete model brings the most noticeable improvement. The experimental results show the same trend in both lane line detection and road detection. The proposed method brings mIoU from 52.51% to 55.38% for lane line detection task and improves road detection accuracy by 2.91% (from 79.80% to 82.71%). Tab. II shows detail test class IoU scores. The full model outperforms all other models in all road and lane classes. The visualized test results for different models are shown in Fig 7. The leverage of spatial modeling ability effectively improves the detection result.

In principle, CBAM is quite different from SCNN, and there are many different ways to combine them with baseline. We explore the difference between various combinations through experiments, and analyze the causes of such results.

1) The Effects of CBAM: From the very beginning, CBAM is designed as a plug-in module with the same input and output shapes. Its relatively simple structure allowed it to have very few parameters and can be easily plugged into most of the existing network structure when needed. In our case, DeepLabV3+ mainly consists of Encoder, ASPP, and Decoder. These three parts contain lots of convolution layers. Most of these convolutional layers are theoretically integratable with CBAM modules. Obviously, we cannot append CBAM to all convolutional layers (even CBAM only have a small number of parameters, its abuse still increases the computing burden of the network). Therefore, we design to add an appropriate amount of CBAM modules in different places, and finally determine the insertion mode we use through experimental analysis of the actual effects.

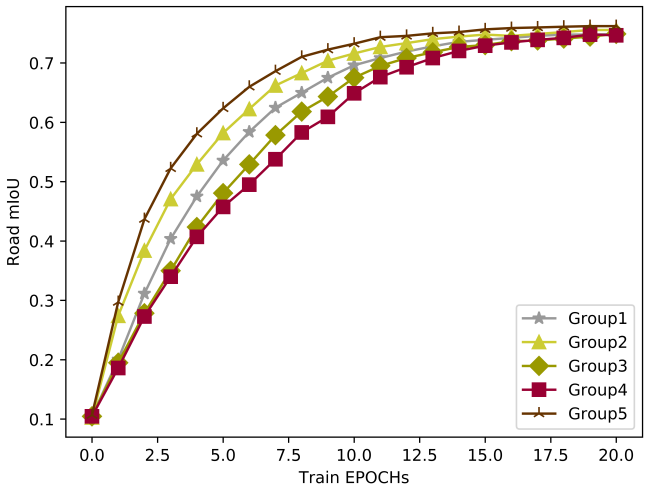

Fig. 6. The comparison of different insert methods and different insert quantity of CBAMs. It can be seen that different settings will have a significant impact on the convergence speed at the early stage, and the final convergence results are slightly different from limited training cycles. In general, the increase in the number of CBAM will improve the final result but reduce the convergence speed. Furthermore, the fifth group showed relatively good performance both in convergence rate and detection accuracy. So the final insert methods of the proposed methods use the fifth setting.

ResNet_152 mainly consists of four Res_Layers, and each Res_Layer contains lots of Res_Blocks. We try to add CBAM modules to the bottom of all the blocks in each Res_Layer from the beginning. That is, the first group adds a CBAM module after all the Res_Blocks in Res_Layer1. The second group adds the CBAM module after all the Res_Block in Res_Layer1 and Res_Layer2, and so on. Finally, one more group that only inserts CBAM modules between each Res_Layers is added into the experiment plan. A total of five groups are trained, and then the effect of different combinations is tested and compared. The test result is shown in Fig. 6.

According to the experiment results, in terms of the overall trend, the effect of the model is getting better as the number of CBAM increases. However, after exceeding a certain number, the improvement becomes weaker. Moreover, due to the rapid increase in the number of Res_Block modules, which leads to the explosive growth of the number of CBAM modules,

TABLE IV
THE INFLUENCE OF SCNN'S DIRECTION (MIOU IN %).

Method	Road mIoU	Lane Lines mIoU
SCNN_U	80.67	52.87
SCNN_D	80.93	53.22
SCNN_L	80.15	53.17
SCNN_R	80.28	52.68
SCNN_UDLR	**82.71**	**55.38**

a large amount of computation is introduced in the case of limited model improvement. In a restricted number of tests, we can see that the introduction of too many CBAM modules brings limited improvement, which also increases the training difficulty of the network to some extent and slows down its convergence speed. So after careful consideration, we finally choose the last set of solutions. That is, the CBAM module is inserted into the bottom of each Res_Layer. In this way, only four CBAM modules are used, which can effectively improve the performance of the model without introducing too much computation.

2) The Effects of Spatial CNN : SCNN makes a significant contribution to the information flow inside the feature tensors extracted by the neural networks. Since the tensors are high-dimensional information carriers, its internal information flow will have directional characteristics, and the information flow along different directions will undoubtedly lead to different results. As shown in Fig. 4, there are four different directions. To verify the effect of different directions, we firstly use four SCNNs with different directions alone, then integrate all four choices. Finally, analyze the actual effect of different combinations through experimental results.

According to Tab. IV, four kinds of SCNN that only use a single direction do not bring significant improvement to the network, while the method of four directions combined brings relative improvements to the model. Considering its relatively small amount of parameters, we finally chose to improve the framework in a way that uses all directions.

As for the insert position of SCNN, since they also have the same input and output size, the optional insert positions are similar to CBAM. Combine different positions and amounts can make an infinite number of structure groups. The experiments developed by Pan *et al.* [46] conclusively prove that SCNN can achieve a better effect as long as it is inserted at the top of the original network structure. Therefore, we will not repeat that work here and choose to directly place SCNN at the top, which is the position immediately after ASPP as shown in Fig. 2.

3) The Effects of Atrous Convolution: The central variable part of the dilation rates lies in the Res-Layers of the encoder. In the original DeepLabV3+ framework, the dilation rates of four Res-Layers have been set to [1, 1, 1, 2] and [1, 1, 2, 4] separately. Of course, the structure of the neural network then needs to be slightly adjusted and changed with the shape of the features. Here we set experiments with different dilatation rates and analyze the detailed influence of the atrous convolution.

Three sets of different experiments are set up. The only difference between the different groups is that the expansion coefficients used in the atrous convolution, and all other settings are consistent. All training processes use the same hardware equipment as before, but due to the time limit, the number of epochs is reduced, and we only train 25 complete epochs in each group. The first group uses ResNet without dilation as the backbone. The second and third groups use ResNet with dilation rates of four Res-Layers set to [1, 1, 1, 2] and [1, 1, 2, 4] separately. The results of the comparison experiment are given by Tab. V. A large dilation rate brings improvement to the detection accuracy. Thus, the dilation group 2 is used for the encoder backbone base on the experiment result.

TABLE V
THE INFLUENCE OF ATROUS CONVOLUTION (MIOU IN %).

Method	Road mIoU	Lane Lines mIoU
Without dilation	79.21	52.65
Dilation Group 1	79.54	52.41
Dilation Group 2	**80.58**	**53.14**

TABLE VI
THE INFLUENCE OF MULTI-TASK DECODER (MIOU IN %).

Decoder Type	Road mIoU	Lane Lines mIoU
Single Road	80.37	-
Single Lane	-	52.86
Multi Task	**80.58**	**53.14**

4) The Effects of Multi-Task Decoder: To verify the influence of the multi-task decoder. We design experiments use the full model with dual-task decoder or single road/lane decoder for comparison. The max training epoch is also set to 25 like in Section V-C3.

Tab. VI shows the impact of multi-task decoder on the final results. We can see that different decoder branches promote each other, which can slightly speed up the model's learning process. The biggest change brought by the dual-task decoder is that multiple results can be obtained by only one encoding process.

D. Discussion

According to the experimental results, the CBAM and SCNN modules bring improvements to the understanding ability of the spatial information. However, how exactly do they work, and why can they make such a difference based on what we already have? Here We design different analysis methods for these two structures and try to explore the in-depth principles.

1) Analysis of CBAM modules: The original purpose of the CBAM module is to make the input feature vector's different positions have different weights to achieve the effect of the attention mechanism. This section extracts CBAM modules from the well-trained frameworks with different training progress and visualizes the effect of the attention.

To verify the effect of training on CBAM, we select different training stages to verify and visualize its impact. Here we

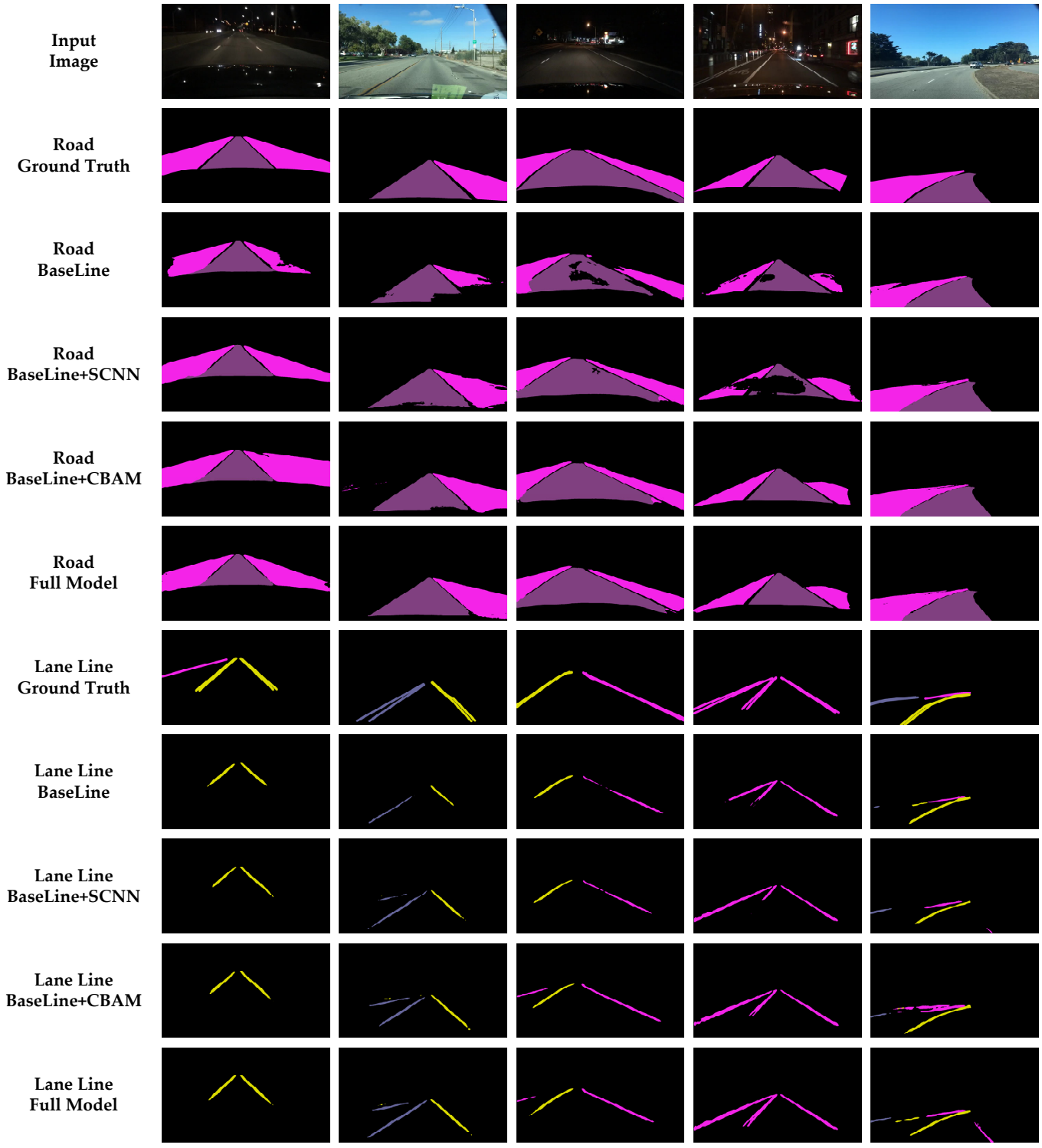

Fig. 7. Exemplar results on modified BDD100K traffic scenes understanding datasets. We show four comparative results here, namely baseline, baseline with SCNN, baseline with CBAM, and full model (baseline with SCNN and CBAM).

Fig. 8. Visualizations of the attention effects of CBAM at different training stages(random initialization, 10 epochs, 30 epochs, 50 epochs).

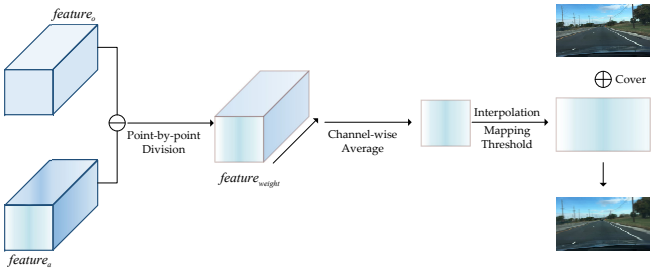

Fig. 9. Schematic diagram of the CBAM visualization process. The feature change weight coefficients after dimensionality reduction can easily overlay on the original input image and show attention distribution intuitively.

chose random initialization, train 10 epochs, 30 epochs, and 50 epochs for comparison. For a trained model under a specific training phase, a randomly chosen traffic scene is used as the input. Base on this particular input, the model first obtains a high-dimensional convolution feature $\mathscr{F}_o$ through the convolutional network, and then after the feature passes the attention module, it becomes a new convolution feature $\mathscr{F}_a$. We use $\mathscr{F}_a$ to do a point-by-point division of $\mathscr{F}_o$ and calculate the weight value $\mathscr{F}_{weight}$ that attention gives to each point.

Fig. 9 shows the visualization process of attention effect. The feature change weight coefficient obtained by point-by-point division is a 3-D high-dimensional tensor. This vector faithfully records the numerical changes, but it is not intuitive enough for the understanding of CBAM's modeling ability of spatial structure information. So, here we calculate the average of $\mathscr{F}_{weight}$ through channel dimension, map it to the two-dimensional space, and restore the size to the original input image size by interpolation. Then the two-dimensional weight change data is mapped to the range of 0 to 255, filtered by a simple threshold, and superimposed on the original image to show the attention effect brought by CBAM. Considering the information loss caused by the interpolation during recovering image size, the analysis of CBAM here only use the one after the first Res-Layer because its corresponding convolution feature has a relatively large spatial resolution.

Through the above visualization method and experiment planning, the final comparison results are shown in Fig. 8. It can be seen that the spatial weights brought by random initialization are randomly and evenly distributed over the entire image, which means that it has the same attention for different positions of the entire input image. With the progress of network training, the CBAM module gradually increases the weights for the regions of interest, and finally makes the attention of the module focus on the lane line and the road area. The upper and bottom parts of the input picture generally do not include the road area and lane line area, so the weight of these positions will gradually become smaller as the training progresses.

TABLE VII
INFLUENCE OF OCCLUSIONS (MIOU IN %).

Usage of SCNN	None	Upper	Middle	Bottom
None (Road mIoU)	81.04	57.31	43.65	47.35
None (Lane mIoU)	54.02	33.64	26.82	25.94
DULR (Road mIoU)	**82.71**	**66.72**	**54.57**	**58.32**
DULR (Lane mIoU)	**55.38**	**41.35**	**35.91**	**33.28**

2) Analysis of SCNN modules: In order to verify that SCNN brings sufficient internal spatial information flow to the network, the actual impact of SCNN is further explored through experiments. For the input traffic scene images, we use image processing tools to add different noise occlusions at different locations and then test the performance of the detection network in different occlusion situations with or without SCNNs separately. We use the occlusion scheme shown in Fig. 10, which is without occlusion and occluded in the upper, middle, and lower parts. The occlusion material is randomly generated low-density Gaussian noise.

Fig. 10. This figure shows the generation step of noise occlusion. The different parts of the input image are covered with a slide of noise block. Then the test results of processed images are compared to analysis the effect of SCNNs.

A few test images are separated from the dataset's validation set and covered with the different noise. The detailed test results are shown in Tab. VII. According to the test results, no matter where the occlusion occurs and which structure is used, the noise makes the detection accuracy suffer a severe drop. However, the detection network with SCNNs performs much better than the one without it. This result proves that SCNNs can set off internal information flow that helps improve the robustness of the existing network structure.

VI. CONCLUSION AND FUTURE WORK

This paper proposes an attention based spatial segmentation network for traffic scene understanding. It integrates convo-

lutional attention modules and spatial convolution operation with an encoder-decoder semantic segmentation network. Experimental results show that the proposed method effectively improves the traffic scene understanding ability, which means lane and road detection accuracy. It has good robustness to the influence of noise. Furthermore, a pixel-level road segmentation dataset named NWPU Road Dataset is released. It helps improve existing methods for road identification and detection.

The spatial and attention modeling modules are proved to be effective in the traffic scene understanding task, and it can be extended into generalized semantic segmentation and other pixel-level dense prediction tasks.

REFERENCES

[1] Y. Wang, E. K. Teoh, and D. Shen, "Lane detection and tracking using b-snake," *Image and Vision computing*, vol. 22, no. 4, pp. 269–280, 2004.

[2] A. H. Lai and N. H. Yung, "Lane detection by orientation and length discrimination," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 30, no. 4, pp. 539–548, 2000.

[3] Q. Wang, J. Gao, and Y. Yuan, "Embedding structured contour and location prior in siamesed fully convolutional networks for road detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 1, pp. 230–241, 2017.

[4] X. Zhang, Y. Yuan, and Q. Wang, "Roi-wise reverse reweighting network for road marking detection." in *BMVC*, 2018, p. 219.

[5] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer *et al.*, "Autonomous driving in urban environments: Boss and the urban challenge," *Journal of Field Robotics*, vol. 25, no. 8, pp. 425–466, 2008.

[6] Y. Xu, D. Xu, S. Lin, T. X. Han, X. Cao, and X. Li, "Detection of sudden pedestrian crossings for driving assistance systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 3, pp. 729–739, 2011.

[7] M. Aly, "Real time detection of lane markers in urban streets," in *2008 IEEE Intelligent Vehicles Symposium*. IEEE, 2008, pp. 7–12.

[8] J. Son, H. Yoo, S. Kim, and K. Sohn, "Real-time illumination invariant lane detection for lane departure warning system," *Expert Systems with Applications*, vol. 42, no. 4, pp. 1816–1824, 2015.

[9] B. Huval, T. Wang, S. Tandon, J. Kiske, W. Song, J. Pazhayampallil, M. Andriluka, P. Rajpurkar, T. Migimatsu, R. Cheng-Yue *et al.*, "An empirical evaluation of deep learning on highway driving," *arXiv preprint arXiv:1504.01716*, 2015.

[10] J. Li, X. Mei, D. Prokhorov, and D. Tao, "Deep neural network for structural prediction and lane detection in traffic scene," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 3, pp. 690–703, 2017.

[11] Y. He, H. Wang, and B. Zhang, "Color based road detection in urban traffic scenes," in *Proceedings of the 2003 IEEE International Conference on Intelligent Transportation Systems*, vol. 1. IEEE, 2003, pp. 730–735.

[12] H. Kong, J.-Y. Audibert, and J. Ponce, "General road detection from a single image," *IEEE Transactions on Image Processing*, vol. 19, no. 8, pp. 2211–2220, 2010.

[13] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.

[14] J. Fritsch, T. Kuehnl, and A. Geiger, "A new performance measure and evaluation benchmark for road detection algorithms," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, 2013, pp. 1693–1700.

[15] F. Yu, W. Xian, Y. Chen, F. Liu, M. Liao, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving video database with scalable annotation tooling," *arXiv preprint arXiv:1805.04687*, 2018.

[16] R. M. Haralick and L. G. Shapiro, "Image segmentation techniques," *Computer vision, graphics, and image processing*, vol. 29, no. 1, pp. 100–132, 1985.

[17] J. Shi and J. Malik, "Normalized cuts and image segmentation," *Departmental Papers (CIS)*, p. 107, 2000.

[18] F. Meyer and S. Beucher, "Morphological segmentation," *Journal of visual communication and image representation*, vol. 1, no. 1, pp. 21–46, 1990.

[19] L. Grady, "Random walks for image segmentation," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 11, pp. 1768–1783, 2006.

[20] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, 2015, pp. 3431–3440. [Online]. Available: https://doi.org/10.1109/CVPR.2015.7298965

[21] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1520–1528.

[22] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, 2017. [Online]. Available: https://doi.org/10.1109/TPAMI.2016.2644615

[23] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[24] J. Dai, K. He, and J. Sun, "Convolutional feature masking for joint object and stuff segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, 2015, pp. 3992–4000. [Online]. Available: https://doi.org/10.1109/CVPR.2015.7299025

[25] Q. Wang, J. Gao, and Y. Yuan, "A joint convolutional neural networks and context transfer for street scenes labeling," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 5, pp. 1457–1470, 2017.

[26] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. [Online]. Available: http://arxiv.org/abs/1412.7062

[27] 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 transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2018.

[28] L. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *CoRR*, vol. abs/1706.05587, 2017. [Online]. Available: http://arxiv.org/abs/1706.05587

[29] L. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VII*, 2018, pp. 833–851. [Online]. Available: https://doi.org/10.1007/978-3-030-01234-2_49

[30] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *European conference on computer vision*. Springer, 2016, pp. 102–118.

[31] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3234–3243.

[32] Q. Wang, J. Gao, and X. Li, "Weakly supervised adversarial domain adaptation for semantic segmentation in urban scenes," *IEEE Transactions on Image Processing*, 2019.

[33] Z. Li, J. Tang, L. Zhang, and J. Yang, "Weakly-supervised semantic guided hashing for social image retrieval," *International journal of computer vision*, vol. 128, no. 8, pp. 2265–2278, 2020. [Online]. Available: https://doi.org/10.1007/s11263-020-01331-0

[34] Z. Peng, Z. Li, J. Zhang, Y. Li, G. Qi, and J. Tang, "Few-shot image recognition with knowledge transfer," in *Proceedings of the IEEE conference on computer vision and pattern recognition*. IEEE, 2019, pp. 441–449. [Online]. Available: https://doi.org/10.1109/ICCV.2019.00053

[35] X. Li, M. Chen, F. Nie, and Q. Wang, "A multiview-based parameter free framework for group detection," in *Thirty-First AAAI Conference on Artificial Intelligence*, S. P. Singh and S. Markovitch, Eds. AAAI Press, 2017, pp. 4147–4153.

[36] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.
[38] Z. Li, J. Tang, and T. Mei, "Deep collaborative embedding for social image understanding," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 9, pp. 2070–2083, 2019. [Online]. Available: https://doi.org/10.1109/TPAMI.2018.2852750
[39] Q. Wang, M. Chen, and X. Li, "Quantifying and detecting collective motion by manifold learning," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, S. P. Singh and S. Markovitch, Eds. AAAI Press, 2017, pp. 4292–4298.
[40] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola, "Stacked attention networks for image question answering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 21–29.
[41] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in neural information processing systems*, 2015, pp. 577–585.
[42] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4651–4659.
[43] J. Park, S. Woo, J.-Y. Lee, and I. S. Kweon, "Bam: Bottleneck attention module," *arXiv preprint arXiv:1807.06514*, 2018.
[44] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 3–19.
[45] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," in *Advances in neural information processing systems*, 2011, pp. 109–117.
[46] X. Pan, J. Shi, P. Luo, X. Wang, and X. Tang, "Spatial as deep: Spatial cnn for traffic scene understanding," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
[47] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "Labelme: a database and web-based tool for image annotation," *International journal of computer vision*, vol. 77, no. 1-3, pp. 157–173, 2008.
[48] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
[49] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.
[50] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

Xuelong Li (M'02-SM'07-F'12) is a full professor with the School of Computer Science and the Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an 710072, Shaanxi, P. R. China.

Zhiyuan Zhao received the B.E. degree in electronics and information engineering from the Northwestern Polytechnical University, Xi'an 710072, Shaanxi, P. R. China, in 2018. He is currently pursuing the Ph.D. degree from Center for Optical Imagery Analysis and Learning, Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision and pattern recognition.

Qi Wang (M'15-SM'15) received the B.E. degree in automation and the Ph.D. degree in pattern recognition and intelligent systems from the University of Science and Technology of China, Hefei, China, in 2005 and 2010, respectively. He is currently a Professor with the School of Computer Science and with the Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision and pattern recognition.