RailNet: An Information Aggregation Network for Rail Track Segmentation

Haoran Li, Qichao Zhang, Dongbin Zhao, Yaran Chen
The State Key Laboratory of Management and Control for Complex Systems
Institute of Automation, Chinese Academy of Sciences
University of Chinese Academy of Sciences
Beijing, China
lihaoran2015@ia.ac.cn, zhangqichao2014@ia.ac.cn, dongbin.zhao@ia.ac.cn, chenyaran2013@ia.ac.cn

***Abstract*—As the basis of scenes understanding for the track inspection task, track segmentation is challenging due to the various illumination conditions, track crossing, and plant coverage. Since the rail has a strong shape prior, strict rail spacing and special distribution in the image, making full use of the spatial information of the rail features becomes an important factor to improve the accuracy of rail segmentation. In this paper, an information aggregation module is proposed to enhance the spatial relationship between pixels of the rail features. In other words, this module expands the receptive field. Furthermore, we build an information aggregation network based on this module, which is called as RailNet. Finally, the RailNet is evaluated in an open train track dataset. Experimental results show that RailNet can achieve the best performance so far in the dataset of trains.**

I. Introduction

Because of the low transportation cost and high safety performance, the railway plays a very important role in transportation activities for a long time. With the development of intelligent technology, the railway train has gradually changed from the human driving mode to the automatic and unmanned model. The autonomous train has become one of the development trends of train intelligence. It should be mentioned that the track inspection and maintenance are the basis tasks to ensure the security for the autonomous train. To reduce the manual labour and improve the maintenance quality in the traditional manner based on human driver, automatic track inspection technology becomes one of the urgent needs in the process of track intelligence.

Track segmentation, which is one of the core functions of an image-based railway inspection system(IRIS)[1], plays a crucial role in the autonomous train system. It can locate the position of the track in the image, to guide the following tasks such as track deformation analysis and defect detection. For the train automatic driving system, track segmentation provides a region of interest(ROI) for sensing systems. The system detects obstacles on or close to the track according to the sensor data belongs to the ROI. To achieve this, the track detector provides the exact position of the detected track in the image to the other sensors such as RADAR, LIDAR and so on.

Because of the texture and shape of the track, the edge detection method based on computer vision is often used in the track extraction[2][3][4]. Since a pair of rails are parallel in physical space, the image of the camera which installed on the top of the front of the train will usually be transformed by inverse perspective mapping. Then the track will be extracted by using Hough transform or some filtering methods[5]. However, in the real environment, the traditional vision based track segmentation often faces many severe challenges, such as complex track crossing scene, changeable lighting environment, plant growth, and coverage.

In recent years, the deep neural network, especially the deep convolution neural network, has made a great breakthrough in the field of the image. The image processing method based on deep neural network has been widely used in autonomous driving[6][7][8][9][10] and track inspection, such as rail surface defect detection[11][12], rail detection[13], railway track switches detection and so on. The performance of full convolution neural network-based methods in image segmentation task is much better than that based on traditional computer vision. Recently, the track detection based on image segmentation using the deep full convolution network has been investigated widely.

Note that the image segmentation method based on the deep full convolution network has strong robustness in dealing with complex traffic scenes and various illumination environments. However, these general segmentation methods do not consider the special shape and parallel distribution characteristics of the track, which are very useful for the track segmentation. On the other hand, if these characteristic features are considered in deep learning based computational vision methods, adaptability is relatively poor for the complex and changeable environment. Therefore, this paper focuses on the particularity of track shape and distribution, proposes the information aggregation module, improves the general semantic segmentation method based on the full convolution neural network, and realizes a special track segmentation method based on deep learning. The main contributions of this paper include:

- In this paper, considering the shape and distribution characteristics of rail, a new efficient rail segmentation

This work is supported by the Beijing Science and Technology Plan under Grants Z191100007419002, and the National Natural Science Foundation of China (NSFC) under Grants No. 61803371, No. 61533017 and No. 61603268.

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

method based on the deep convolutional neural network is proposed, which is named as RailNet.
- We design an information aggregation module to enhance the spatial relationship between neurons on the feature map and give two policies to calculate the weights of the aggregation module.
- Comparing to the current methods, the RailNet achieves state of the art for the track segmentation task on the train dataset.

II. Related Work

Track segmentation is an important module in an autonomous train system. Due to the lack of annotation datasets, most of the track segmentation methods are based on computer vision features and the geometric characteristics of the track. The most commonly pipeline[14][15] is to extract the candidate region of rail from the image by edge detection methods, and then filter the final rail region from the candidate region by Hough transform. In the practical environment, there may be a variety of backgrounds and different directions of light in the image, which will affect the extraction and determination of the track. In recent years, with the development of deep learning, the technology based on deep learning has been gradually introduced into the autonomous train system. For example, [13] uses Fast RCNN to extract the ROI including the track to be detected,

Track segmentation is similar with the lane segmentation, in which long and thin lane marks should be detected on the road. In the early methods of lane segmentation, the pipeline firstly extracts ROI region in the image, and uses Sobel operator or Canny operation to extract candidate region of the lane, then generates lane line through a series of post-processing. The track segmentation methods are mostly borrowed from the lane segmentation methods. However, the difference from the lane environment is that the spacing between tracks is more strictly constrained, and the lateral offset is generally constant. This makes the two kinds of methods have some differences in post-processing. With the open-source of large-scale lane datasets, many lane segmentation methods based on deep learning are as follows. VPGNet[16] uses a multi-task network to predict lane marker and vanishing points simultaneously. [17] uses the case segmentation method to the segment lane line, and designs H-net which learns lane equation parameters adaptively. [18] introduces the dense upsampling convolution(DUC) module[19] into the lane segmentation, which makes the segmentation result get better resolution. Although there are more strict constraints on the shape and distribution of the track, the track segmentation faces more challenging scenes such as weed coverage, ground embedding, track crossing, etc., and the realization of a robust track segmentation still faces great challenges.

The above track segmentation and lane segmentation methods almost do not consider the shape and distribution characteristics of the lane markers when the networks are designed. [20] considers the spatial information between features for the first time. Based on the conditional random field (CRF), the authors constructed spatial convolutional neural networks(SCNN) which enable the network to obtain stronger spatial connections and a greater receptive field. Unfortunately, this cascade recursive module limits the reasoning speed of the whole networks. Considering that the track is more strict than the lane, and motivated by the SCNN scheme, we propose a more efficient module to solve the information propagation during the feature maps.

III. Methodology

A. Brief review of SCNN

In the process of feature extraction of a semantic segmentation network, many convolutions and pooling operations are stacked to ensure that the receptive field of the feature map on the original image is large enough. However, the theoretical range of receptive field caused by this cascaded convolution will be degraded due to parameter sharing and pooling operation, so that the actual range is much smaller than the theoretical receptive field. In order to capture the strong structure prior, SCNN is proposed.

The goal of SCNN is that the network can explicitly construct the spatial relationship between different location features. Unlike Markov random field which establishes the spatial relationship between all pixels and the current pixel, SCNN takes rows and columns in the feature map as the layer of the network and performs cascading convolution. The specific process is as follows

$$X'_{i,j,k} = \begin{cases} X_{i,j,k}, & j = 1 \\ X_{i,j,k} + f(\sum_m \sum_n X'_{m,j-1,k+n-1} \\ \cdot \omega_{m,i,n}), & j = 2, 3, ..., H \end{cases} \tag{1}$$

Here X and X' are the feature maps before and after spatial convolution, correspondingly. f is the activation function while ω is the weights of the convolution kernel. By cascading four convolution operations in different directions, information can be propagated from top to bottom, from bottom to top, from left to right and from right to left.

In this way, SCNN enhances the spatial connection of adjacent features. This cascading convolution deepens the network structure in a disguised way and expands the range of the receptive field of the final feature map to a certain extent. But the problem with this structure is also obvious:

- In SCNN, the width of the convolution kernel controls the connection range between the neurons of the next layer and the previous layer. Although the way of cascaded connection between rows and columns implicitly expands the receptive field range of the latter layer, due to the parameter sharing mechanism, the actual receptive field range of this way is far from the theoretical value.
- The inference speed of this recursive convolution for information propagation is limited by the size of the feature map and the convolution kernel, which results in a certain limitation of the reasoning speed of the network, and it is difficult to speed up the inference.

- When the size of the feature map is large, it will cause the gradient disappearance or the gradient explosion for the gradient backpropagation processing, which makes the network difficult to converge.

B. Information Aggregation Module

SCNN establishes a stronger local connection relationship through the recursive convolution between rows and columns, which makes the receptive field of the feature map be larger. Considering the problem of SCNN, we design a more concise and effective information aggregation module(IAM). Different from SCNN's cascading recursive way of building the relationship between different rows and columns, IAM directly builds the relationship between each row or column and the others. The structure are shown in Fig. 1 The calculation formula is as follows

$$X'_{i,j,k} = X_{i,j,k} + f(\sum_{c} \omega_c \cdot X_{i,j,k}), \text{where } c = \{i, j\} \quad (2)$$

In this way, the recursive operation is avoided and the inference process can be accelerated. Adding the rows or columns directly strengthens the receptive field range of the current feature. This method also does not increase the depth of the network implicitly, to avoid the gradient disappearance of the network in the training process. In order to obtain the information in the vertical and horizontal directions, we parallel two IAMs in different directions to obtain the aggregate features in various directions, and finally get the final feature map by adding fusion.

The core of IAM is to calculate the corresponding weight of each row or column. The value of the weight determines the spatial connection strength between different positions, which will affect the quality of the final feature map. For the track segmentation, since the track usually appears in pairs under the image, this characteristic is reflected on the weight distribution. It means that the weight value of the upper half of the aggregate weight in the vertical direction may be small, while the weight value of the lower half is relatively large. Considering these distribution characteristics of tracks, we design two weight acquisition strategies.

1) Learnable weights: The intuitional way to obtain the weight is to set the weight as a learnable parameter, and learn directly through the gradient descent during the training. As the rail may appear from left to right in the image, parameter sharing is used in each column calculation when vertical information is aggregated. In order to facilitate the calculation, the same strategy is used for horizontal information aggregation.

During the vertical direction information aggregation, for one of the columns, we need to aggregate different information at different positions. For example, the position at the top of the image may probably collect background features, while the position at the bottom is more to aggregate the features of the complete rail. The weights used to aggregate other neurons

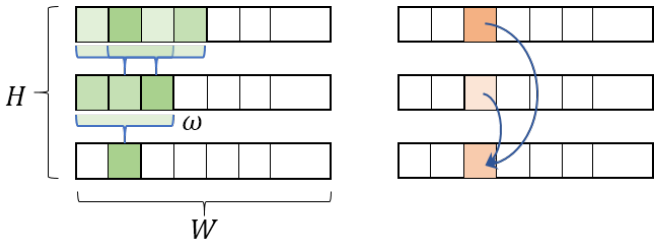

Fig. 1. Information propagation module. The left figure shows the information propagation process of SCNN from top to bottom. Where H and W are the height and the width of the feature map, correspondingly. ω is the convolution kernel. Starting from the second row, the information of each row is obtained by convolutional calculation based on the features of adjacent positions of the previous row and adding with the current features, and the third row can only be calculated after the second row is calculated, and so on. The right figure is the schematic diagram of the vertical information aggregation module. Different from the recursive calculation of SCNN, the module directly aggregates the features of the same column, thus simplifying the information propagation process.

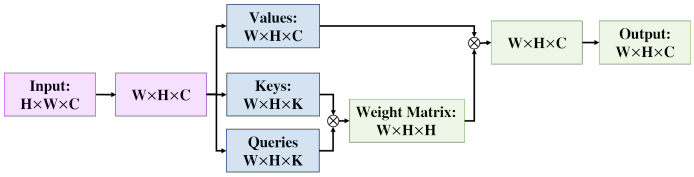

Fig. 2. The weight calculation method of vertical direction based on the attention mechanism. Firstly, we transpose the feature map, and then three tensors named values, keys and queries are obtained by three different 1×1 convolutions. Then keys and queries get the weight matrix by matrix multiplication. The feature map after aggregation information is obtained by matrix multiplication of weight matrix and values.

for different neurons in each row or column are different. The vertical direction is calculated as follows

$$X'_{i,j,k} = X_{i,j,k} + f(\sum_{m=1}^{H} \omega_{i,m} \cdot X_{m,j,k}) \quad (3)$$

Here ω is learnable weight. The horizontal direction is calculated in a similar way. This method can be implemented efficiently in the form of convolution.

2) Attention-based weights: Constructing the IAM with learnable weights is a very simple and convenient way. However, this method also has some potential problems. For example, for the weights in the vertical direction, the weight values of columns in the whole image are the same, and the weight distribution of each image is the same. Nevertheless, the actual situation is that the distribution of tracks in the different images is different. This method can only get the weight under a statistical distribution, and can not adjust it adaptively according to the changes of the content in the image.

We hope to establish the relationship between features and weights, so as to build a method that can change weight adaptively according to the content. In recent years, since Wang et al.[21] introduces attention mechanism into image field and proposed the non-local network, the image processing method based on attention mechanism has become brilliant[22][23].

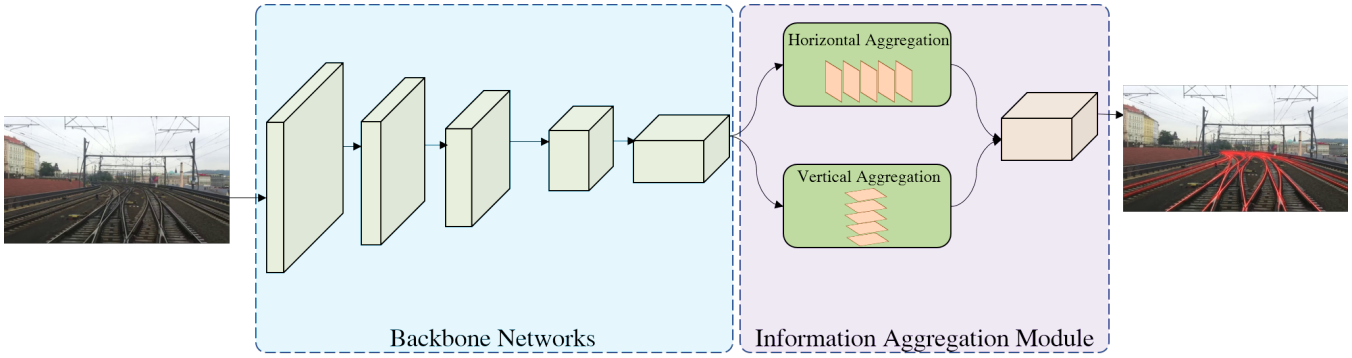

Fig. 3. The framework of RailNet. It consists of the backbone networks and the information aggregation module. The backbone networks extract context features while information aggregation module enhances the spatial connection between features.

It should be mentioned that the computational complexity of calculating the attention weight matrix on the whole image is very large for the above methods. Motivated by this, we design a method of orthogonal attention mechanism for calculating the information aggregation weight of horizontal and vertical methods for the rail segmentation scene. Similarly, taking the aggregation weight calculation in the vertical direction as an example, the calculation method is shown in Fig. 2.

The calculation of weight matrix in the Fig. 2 is as follows

$$\omega_{i,m}^{j} = \frac{\exp(X_{i,j,*}^{T} \cdot X_{m,j,*})}{\sum_{n} \exp(X_{i,j,*}^{T} \cdot X_{n,j,*})} \tag{4}$$

C. Network architecture

We use VGG16[24] as the backbone networks for feature extraction, and construct the RailNet with IAM. The framework is shown in Fig. 3. In order to maintain the resolution as much as possible to retain the spatial location information of the image, we remove the max-pooling between the fourth and fifth convolutional stages. At the same time, in order to make up for the loss of receptive field caused by the removal of the pooling layer, dilated convolution is used in the fifth stage, and the dilated rates of the dilated convolutions are set to 2-2-2-4-1.

D. Loss function

Different from the general task of image segmentation, the track takes up a very small proportion in the image, so track segmentation faces a very serious category imbalance problem. He et al.[25] proposes focal loss to solve the problem of unbalanced between foreground and background in object detection. We extend this method to the task of image segmentation. The loss function is defined as follows

$$L = \sum_{m,n} \sum_{c} \alpha_c (1 - y_{m,n,c})^{\gamma} \log(y_{m,n,c}) \tag{5}$$

Among them, α_c represents the imbalance proportion of the category c, $1-y$ represents the gradient value of the prediction probability of the sample, which is used to represent the difficulty of the pixel. γ controls the influence of the difficult pixels on the network training. In this paper, we set $\alpha_c = 0.2$ for the background and $\gamma = 0.25$.

IV. Experiments

In this section, using the open train track annotation dataset, several experiments are designed to verify the effectiveness of the proposed IAM. We use the stochastic gradient desent(SGD) method to train the network. The initial learning rate of the network is 0.01, the momentum parameter is 0.9, the weight penalty is 0.0001, and the batch size is 4. We train the networks on the computer with two Titan XP GPU, and the image size is resized to 320x640. In the process of training, we use random flip and random clipping to augment the dataset.

A. Dataset

RailSem19[26] is a segmented dataset specially used in the scene of a rail train. There are 8500 pictures in the dataset, covering 35 categories, such as buffer stop, crossing, guard rail, track car, platform and so on. It covers scenes in different seasons, various weather, lighting, and time periods, as well as complex rail crossing scenes, platform embedding scenes and scenes with wild plants. It is a very challenging dataset for the rail segmentation task. We randomly select 5000 pictures as the training set, 3500 images as the validation set.

B. Evaluation metric

In order to evaluate the efficiency of the methods, we use the recall and intersection of union(IoU) as the evaluation metric. For each image, the segmentation recall of the track is defined as the ratio of the number of pixels correctly classified into the track to the total number of pixels occupied by all the tracks. The calculation formula is as follows

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \tag{6}$$

For a dataset, we calculate the average of all image recall as the mRecall of the track segmentation on the dataset. The recall only reflects the false negative of the segmentation method, but not false positive. Therefore, we also use the ratio of the number of correctly classified pixels of the predicted track in the image to the union of the predicted track pixels

Fig. 4. Experimental results of several different algorithms on RailSem19 dataset. From left column to right column are the ground truth, VGG, SCNN, RailNet-LW, RailNet-AW results.

TABLE I
THE TRAIN/VAL RESULTS ON RAILSEM19 DATASET

	VGG	SCNN	RailNet-LW	RailNet-AW
mRecall	0.79/0.75	0.89/0.87	0.82/0.79	**0.92/0.89**
mIoU	0.45/0.40	0.56/0.52	0.50/0.47	**0.59/0.54**
Runtime(ms)	168	214	172	209

and the ground truth as the intersection ratio of the track segmentation. The calculation formula is as follows

$$\text{IoU} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive} + \text{False Negative}} \quad (7)$$

For a dataset, we also calculate the mean value of all image IOU as the mIoU of the method on the dataset.

C. Comparison results

On the basis of the VGG network, aiming at the problems of lane segmentation method SCNN, we propose RailNet to segment the track and give two calculation policies for the weight calculation of IAM. Here we verify the effectiveness of IAM on the RailSem19 dataset and compare the results with the method without IAM structure and SCNN structure. The experimental results are shown in Table I.

RailNet-LW and RailNet-AW in the table represent RailNets with the IAMs which are based on learnable weights and attention mechanism respectively. From the comparison between the results of the last three columns in the table and the baseline, it is obvious that the spatial connectivity of feature map can improve the performance of the network. Compared with SCNN, RailNet-LW simplifies the process of information propagation but loses some performance. RailNet-AW has the best performance among all methods, and its mRecall and mIoU are far higher than the baseline and SCNN. The last row in the table is the inference time of each method, and the size of the image is 320×640. It can be seen that although the accuracy of RailNet-LW is not very good, compared with the other two methods, the inference speed has a greater advantage.

The first row of the results in Fig. 4 shows that RailNet-AW can better capture the shape prior to the rail so that the rail can be segmented to a certain extent in the dark scene. SCNN performs better for rail bending, which is mainly due to the recursive information propagation mechanism. Compared with SCNN and RailNet-AW, RailNet-LW has more similar results with the baseline, and the performance is worse. The main reason is that RailNet-LW depends on stronger shape and distribution prior.

D. Ablation study

In order to analyze the impact of the IAM internal structure on network performance more clearly, we have carried out a more detailed experimental analysis. Next, we will analyze the role of information aggregation in each direction of IAM, and study different ways of combination aggregation in different directions.

1) The influence of information aggregation in different directions on performance: In order to verify the impact of information aggregation in different directions on network performance, we separately take out the horizontal and vertical information aggregation modules for comparative experimental analysis, and the analysis results are shown in Table II. RailNet-H and RailNet-V are the information aggregations in horizontal and vertical directions, correspondingly. To simplify, we use RailNet instead of RailNet-AW which has parallel IAMs with attention-based weights.

The results show that the information aggregations in horizontal and vertical directions have the gain compared with

TABLE II
THE INFLUENCE OF DIFFERENT IAM ON THE PERFORMANCE

	VGG	RailNet-H	RailNet-V	RailNet
mRecall	0.79/0.75	0.84/0.82	0.89/0.85	**0.92/0.89**
mIoU	0.45/0.40	0.53/0.52	0.57/0.55	**0.59/0.54**

TABLE III
THE INFLUENCE OF DIFFERENT PERMUTATIONS ON THE PERFORMANCE

	VGG	RailNet-HV	RailNet-VH	RailNet
mRecall	0.79/0.75	0.91/0.87	0.90/0.84	**0.92/0.89**
mIoU	0.45/0.40	0.56/0.55	0.54/0.53	**0.59/0.54**

the baseline. Information aggregation in the vertical direction has a more significant impact on network performance. The special shape and distribution of rail may be the potential cause of this phenomenon. Because rail is a kind of long and thin lane marker, the receptive field of the feature map extracted by VGG may not cover the whole rail, and the information aggregation in the vertical direction will compensate for the receptive field.

2) The influence of information aggregation with different permutations on performance: In addition, we also need to consider that the combination of information aggregation modules in different directions may have different effects on the network performance. Therefore, we have also studied all possible combinations and conducted comparative experiments. The experimental results are shown in Table III.

In the table, RailNet-HV and RailNet-VH represent information aggregation modules in series with different directions, HV represents a horizontal module in the front, vertical module in back, and VH represents the opposite. It can be seen from the results in the table that different arrangement modes have little influence on the network performance.

V. CONCLUSIONS

In this paper, we propose a new information aggregation network named RailNet for track segmentation. Considering the shape of the rail and the distribution in the image, we construct an information aggregation module, which can enhance the spatial relationship between the features from the horizontal and vertical directions, so as to improve the recognition performance of the network on the rail. We compare and analyze the results on the open dataset RailSem19. The experimental results show that the RailNet achieves very competitive results on the rail segmentation, and the mRecall and mIoU are increased by 14% compared with the baseline.

REFERENCES

[1] J. Jang, M. Shin, S. Lim, J. Park, J. Kim, and J. Paik, "Intelligent image-based railway inspection system using deep learning-based object detection and weber contrast-based image comparison," *Sensors*, vol. 19, no. 21, p. 4738, 2019.

[2] M. Gschwandtner, W. Pree, and A. Uhl, "Track detection for autonomous trains," in *Proceedings of the IEEE International Symposium on Visual Computing (ISVC)*. Springer, 2010, pp. 19–28.

[3] A. K. Singh, A. Swarup, A. Agarwal, and D. Singh, "Vision based rail track extraction and monitoring through drone imagery," *ICT Express*, vol. 5, no. 4, pp. 250–255, 2019.

[4] M. Karakose, O. Yaman, M. Baygin, K. Murat, and E. Akin, "A new computer vision based method for rail track detection and fault diagnosis in railways," *International Journal of Mechanical Engineering and Robotics Research*, vol. 6, no. 1, pp. 22–17, 2017.

[5] M. Maqsood, A. Javed, and N. Majeed, "A novel algorithm for railway tracks detection using satellite imagery," *International Journal of Computer Applications*, vol. 64, no. 14, 2013.

[6] Y. Chen, D. Zhao, H. Li, D. Li, and P. Guo, "A temporal-based deep learning method for multiple objects detection in autonomous driving," in *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2018, pp. 1–6.

[7] Y. Chen, D. Zhao, L. Lv, and Q. Zhang, "Multi-task learning for dangerous object detection in autonomous driving," *Information Sciences*, vol. 432, pp. 559–571, 2018.

[8] Y. Lu, Y. Chen, D. Zhao, and J. Chen, "Graph-FCN for image semantic segmentation," in *Proceedings of the IEEE International Symposium on Neural Networks (ISNN)*. Springer, 2019, pp. 97–105.

[9] D. Li, D. Zhao, Q. Zhang, and Y. Chen, "Reinforcement learning and deep learning based lateral control for autonomous driving [application notes]," *IEEE Computational Intelligence Magazine*, vol. 14, no. 2, pp. 83–98, 2019.

[10] D. Li, D. Zhao, Y. Chen, and Q. Zhang, "DeepSign: Deep learning based traffic sign recognition," in *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2018, pp. 1–6.

[11] Z. Liang, H. Zhang, L. Liu, Z. He, and K. Zheng, "Defect detection of rail surface with deep convolutional neural networks," in *Proceedings of World Congress on Intelligent Control and Automation (WCICA)*. IEEE, 2018, pp. 1317–1322.

[12] S. Faghih-Roohi, S. Hajizadeh, A. Núñez, R. Babuska, and B. De Schutter, "Deep convolutional neural networks for detection of rail surface defects," in *Proceedings of the IEEE International joint conference on neural networks (IJCNN)*. IEEE, 2016, pp. 2584–2589.

[13] S. Mittal and D. Rao, "Vision based railway track monitoring using deep learning," *arXiv preprint arXiv:1711.06423*, 2017.

[14] 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.

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

[16] S. Lee, J. Kim, J. Shin Yoon, S. Shin, O. Bailo, N. Kim, T.-H. Lee, H. Seok Hong, S.-H. Han, and I. So Kweon, "VPGNet: Vanishing point guided network for lane and road marking detection and recognition," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2017, pp. 1947–1955.

[17] Z. Wang, W. Ren, and Q. Qiu, "LaneNet: Real-time lane detection networks for autonomous driving," *arXiv preprint arXiv:1807.01726*, 2018.

[18] H. Li, D. Zhao, Y. Chen, and Q. Zhang, "An efficient network for lane segmentation," in *Proceedings of the IEEE Conference on Cognitive Systems and Signal Processing (ICCSIP)*. IEEE, 2018, pp. 177–185.

[19] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, and G. Cottrell, "Understanding convolution for semantic segmentation," in *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2018, pp. 1451–1460.

[20] 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.

[21] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018, pp. 7794–7803.

[22] H. Zhao, Y. Zhang, S. Liu, J. Shi, C. Change Loy, D. Lin, and J. Jia, "PSANet: Point-wise spatial attention network for scene parsing," in *Proceedings of the European Conference on Computer Vision (ECCV)*. IEEE, 2018, pp. 267–283.

[23] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018, pp. 7132–7141.

[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of the IEEE International Conference on Learning Representations (ICLR)*. IEEE, 2015.

[25] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2017, pp. 2980–2988.
[26] O. Zendel, M. Murschitz, M. Zeilinger, D. Steininger, S. Abbasi, and C. Beleznai, "RailSem19: A dataset for semantic rail scene understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2019, pp. 0–0.