Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 4405242, 11 pages
https://doi.org/10.1155/2022/4405242

Research Article

A Real-Time Complex Road AI Perception Based on 5G-V2X for Smart City Security

Cheng Xu,[1] **Hongjun Wu**,[1] **Yinong Zhang**,[2] **Songyin Dai**,[1] **Hongzhe Liu**,[1] **and Jin Tian**[3]

[1]*Beijing Key Laboratory of Information Service Engineering, College of Robotics, Beijing Union University, Beijing, China*
[2]*College of Urban Rail Transit and Logistics, Beijing Union University, Beijing, China*
[3]*Beijing University of Posts and Telecommunications, Beijing, China*

Correspondence should be addressed to Yinong Zhang; zdhtyinong@buu.edu.cn

Received 17 November 2021; Accepted 9 December 2021; Published 10 January 2022

Academic Editor: Narasimhan Venkateswaran

The Internet of Vehicles and information security are key components of a smart city. Real-time road perception is one of the most difficult tasks. Traditional detection methods require manual adjustment of parameters, which is difficult, and is susceptible to interference from object occlusion, light changes, and road wear. Designing a robust road perception algorithm is still challenging. On this basis, we combine artificial intelligence algorithms and the 5G-V2X framework to propose a real-time road perception method. First, an improved model based on Mask R-CNN is implemented to improve the accuracy of detecting lane line features. Then, the linear and polynomial fitting methods of feature points in different fields of view are combined. Finally, the optimal parameter equation of the lane line can be obtained. We tested our method in complex road scenes. Experimental results show that, combined with 5G-V2X, this method ultimately has a faster processing speed and can sense road conditions robustly under various complex actual conditions.

1. Introduction

The 5G network has the characteristics of high transmission speed and low transmission delay. It provides a more reliable communication environment for V2X. The 5G-based intelligent traffic management system has stronger management capabilities and robustness, which helps to improve traffic flow. With the help of 5G-V2X, real-time complex road perception becomes possible. Therefore, 5G-V2X is the key to a smart city [1, 2].

Lane detection is one of the most important tasks in understanding road scenes and is also the most complicated part. It can use extracted lane-marker information to locate the road and determine the relative position between the vehicle and the road. A lane-detection solution using visual algorithms is a relatively common solution. But there are a wide variety of lane markings. The markings may be blocked due to vehicle crowding, the line may be corroded and worn, and the weather and other factors can bring challenges to lane-detection [3–5].

Early cognitive algorithms for urban roads relied on manual design, which required a lot of work. These methods use Hough transform, random sample consistency, and other methods to segment the road area and detect lane lines [6]. The obvious disadvantage of this method is poor generalization. When the driving environment changes significantly, the accuracy may be significantly reduced [7].

Convolutional neural networks have achieved great success in computer vision. Methods based on deep learning can directly learn domain knowledge from large datasets, greatly improving the ability to understand complex urban road scenes [8–10]. On this basis, we propose a new method of marker detection based on a traditional detection algorithm and deep learning. First, we extract the overall road area by training the Mask R-CNN [11–15]. The identified road area is used as the constraint area, the lane mark is detected in the area, the obtained discrete lane-line feature-point information is clustered by the least-squares method, and the lane lines are fitted in a different field of view using straight-line and curve-fitting models [16, 17].

The main contributions of this paper are the following two aspects: (1) The model combines the convolutional neural network, 5G-V2X, with the traditional algorithm. The model is used to improve the detection speed of extracting feature points. (2) In different fields of view, the straight-line and curve-fitting models are used to fit the straight line to make the fitting result more accurate, and the optimal parameter equation of the lane line can be obtained.

This article is organized as follows. Section 2 introduces related work. Section 3 describes the improved deep learning method used and the clustering and fitting algorithms for lane lines. Section 4 reports the results of the experiment. Section 5 provides our conclusions.

2. Related Work

In recent years, with the availability of parallel computing, the training process of large-scale data has accelerated. The convolutional neural network (CNN) has become a research hotspot and is widely used in computer vision and pattern recognition [18, 19]. The convolutional neural network can automatically learn the hierarchical features of an image, avoiding the blindness of people's artificial design and selection features, and showing its excellent performance in the task of object detection and instance segmentation.

2.1. Semantic Segmentation. A convolutional neural network-based SegNet learns high-order features in a scene to perform road scene segmentation. It classifies other test images by applying a training algorithm to a common image dataset to generate training labels. A new texture descriptor based on color-layer fusion obtains the maximum consistency of the road area [20]. Offline and online information are combined to detect the road area. A combined hierarchical framework for road-scene segmentation can reliably estimate the topological structure of a scene and effectively identify traffic scenes of multilane roads [21]. The state-of-the-art instance segmentation methods Mask scoring R-CNN and Cascade Mask R-CNN are both improved based on Mask R-CNN, which prove the effectiveness of Mask R-CNN [22].

2.2. Lane Line Detection. Lane line detection is the most important part of the entire road surface inspection. In recent years, deep learning has enabled great success in computer vision. In the lane-line detection problem, a deep neural network is used to learn the lane-line feature, which improves the accuracy of lane-line feature extraction and is suitable for complex road environments [23]. The University of Sydney used CNN and RNN to detect lane lines, with the CNN providing geometric information on lane-line structures for use by the RNN to detect lane lines. Kyungpook National University combined CNN and RANSAC algorithms to stably detect lane-line information even in complex road scenes [24]. The Baidu map project team proposed a dual-view convolutional neutral network (DVCNN) for lane-line detection. The Korean Robotics and Computer Vision Laboratory proposed a method to extract multiple regions of interest, merging regions that may belong to the same class, and using a principal component analysis network (PCANet) and neural networks to classify candidate regions [25]. The laboratory proposed a vanishing point guided network (VPGNet) [26] to solve the problem of lane-line and pavement-marking recognition and classification under complex weather conditions. The Ford Research and Innovation Center used the DeepLanes Network to extract lane-line features acquired by cameras on both sides of a vehicle.

2.3. Clustering and Fitting. The lane-line features extracted by deep learning cannot be directly used, and it is still necessary to cluster and fit the lane-line feature points. The main purpose of fitting is to depict the lane markings on a picture and to display the location in the road image [27]. In lane-line clustering and fitting algorithms, the commonly used road models are linear, linear parabola, least-squares (LS) curve fitting, cubic curve, and Bezier curve fitting [28, 29].

2.4. 5G-V2X. The goal of the 5G-V2X communication system is to achieve accurate and efficient road scene perception and accident-free and efficient collaborative autonomous driving [30]. Literature [31–33] proposed a network protocol based on edge computing and a new vehicle network privacy protection protocol to enhance road safety, intelligent transportation, and smart city systems. Literature [34] introduced a real-time communication method based on 5G-V2X, which reduces the energy and time costs of the system and improves the management efficiency of vehicular networks. Literature [35] proposed an intelligent traffic Vehicle Detection Model based on 5G-V2X, which can dynamically coordinate computing and content caching and effectively allocate network resources.

3. Complex Road Lane Line Detection and Processing

We can divide the detection process of the lane line into three steps. The first is to use deep learning to extract the feature points. The second is to cluster the extracted lane-line feature points, and the third is to obtain cluster points.

3.1. Feature-Point Extraction Based on Improved Model. Building deep models is just a means of learning. Unlike traditional shallow learning, which relies on artificial features, deep learning has a deeper model depth, a greater emphasis on feature learning, and a greater amount of data for training. Therefore, it can better describe the internal correlation between the data.

As shown in Figure 1, Mask R-CNN consists of three parts. The first part is the backbone network, which is used for feature extraction. The second part is the head, which is used to obtain category scores and regression bounding boxes, and the third part is to generate a mask.

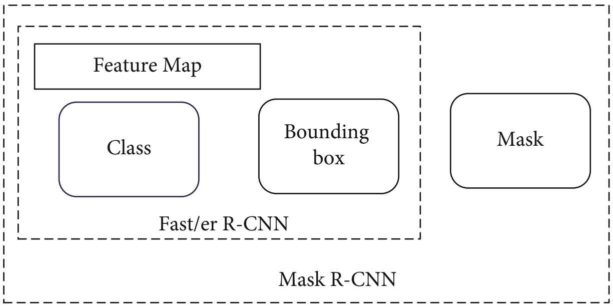


Figure 1: Difference between Fast/er R-CNN and Mask R-CNN on feature maps.

The RPNs in Mask RCNN and Faster R-CNN are the same, but after adding the mask layer, each ROI can be predicted in parallel for each category and "edge," and in parallel. ROI align of Mask R-CNN is shown in Figure 2.

The loss function of ROI is redefined as

$$L = \widehat{L}_{cls} + L_{\mathrm{box}} + L_{\mathrm{mask}}. \tag{1}$$

In the above formula, both work for the positive ROI. For the mask branch of each ROI, the output dimension is K_{m*m}. K means to encode two-class masks for $m * m$ images, each with categories. Therefore, one must apply a single-pixel sigmoid for two classifications and define L_{mask} as the average two-class cross-entropy loss. For an ROI of category K, L_{mask} is defined at the K_{th} mask.

The definition of L_{mask} allows the network to generate a mask for each category, and there is no competition between categories. A special classification branch is used to predict the category label, thus decoupling the category and mask predictions. The FCN uses a single-pixel softmax and multinomial cross-entropy loss. In this case, there is competition between the mask and the classes. Mask R-CNN uses per-pixel sigmoid and binary loss without this consequence. Experiments prove that this is the key point to improve instance segmentation.

We use ResNet101 as the backbone of Mask R-CNN for feature extraction. ResNet includes multiple computing blocks composed of convolution, bias, and Batch Norm (BN). After the training of the model, there are some redundant steps in the model which are only used for forwarding propagation, and the redundant parameters can be reduced by parameter combination.

$$\begin{aligned} X' &= \mathrm{cov}\,(X) + \beta_b, \\ Y &= \alpha \cdot \frac{X' - \mu}{\sqrt{\varepsilon + \sigma^2}} + \beta. \end{aligned} \tag{2}$$

Parameter combination is the 5 parameters of the bias layer and BN layer β_b, α, β, μ, and σ^2; combined into α' and β', the 2 parameters can then calculate it by $Y = \alpha' X + \beta$.

$$\begin{cases} \alpha' = \dfrac{\alpha}{\sqrt{\varepsilon + \sigma^2}}, \\ \beta' = \dfrac{\alpha(\beta_b - \mu)}{\sqrt{\varepsilon + \sigma^2}} + \beta. \end{cases} \tag{3}$$

3.2. Driving Area Division. In this paper, the samples are labeled according to the Mask R-CNN labeling rules. The labeled sample images are sent to Mask R-CNN for training.

Mask R-CNN is ideal for detecting and segmenting lane marking, and the architecture of Mask R-CNN is shown in Figure 3. However, the lane-marking feature obtained through deep learning cannot be directly used. Although the lane-line feature is learned through depth, the extracted lane-line feature only has the coordinate information of the lane line. In the lane lines formed by the dotted lines, we still must identify which lane lines each of these dotted lane-line segments belong to, and this information is not available for discrete coordinate points. Also, the real driving scene is multilane, and we must classify each lane line. Therefore, we propose a clustering method for feature points of lane lines, which can eliminate the interference between multiple lane lines and obtain their information. This provides more accurate and comprehensive input for subsequent lane-line fitting. First, it divides the pavement area, assuming that during data acquisition, the intelligent camera will have a collection period of 10 milliseconds and a top speed of 200 kilometers per hour. By calculation, we know that there are about 10 meters in one cycle. In principle, the camera can get a clear image of the 100-meter mark in front of the vehicle, and lane lines that are too far away are difficult to recognize.

The original image we acquired through the camera is shown in Figure 4. We will use the upper edge of the overall road information identified by the neural network as area C. Above this area is the sky image, which contains no lane-marking line information. Area A is a near-field area that is the main part of our field of vision, where the lanes can be approximated as straight lines. Area B is the midfield area, and there may be a curve with a small curvature. Area C is the far-field of view where the lane image reflects the curvature of the lane. After dividing the overall road information in this way, when designing the algorithm, we need only pay attention to the near and middle fields of view of the image of the front of the vehicle in the divided road area.

3.3. Clustering Feature Points. Let $F(x, y)$ represent the extracted feature-point coordinates. The feature segment B $(x_s, y_s, x_e, y_e, k, b, n)$ represents the lane-line marker segment information, composed of a series of lane-line coordinates, where (x_s, y_s) and (x_e, y_e) represent the starting and ending coordinates of a lane line, k and b are the line parameters of the cluster feature coordinate, and n represents the number of coordinates of the current lane-line segment.

The feature coordinates' information points already included in the lane-line segment extracted by deep learning may represent the current lane-line characteristic line

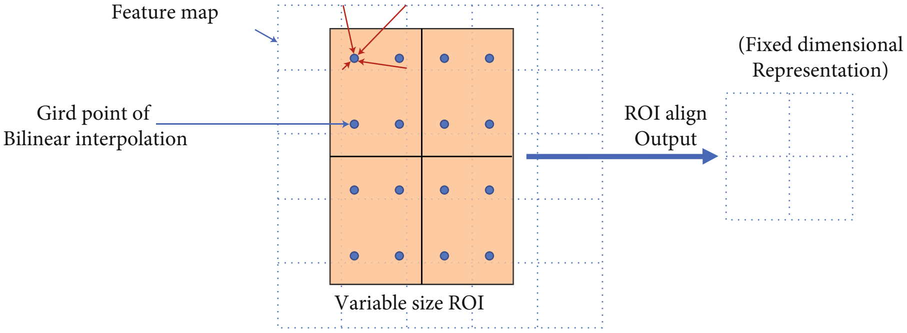


Figure 2: ROI align of Mask R-CNN.

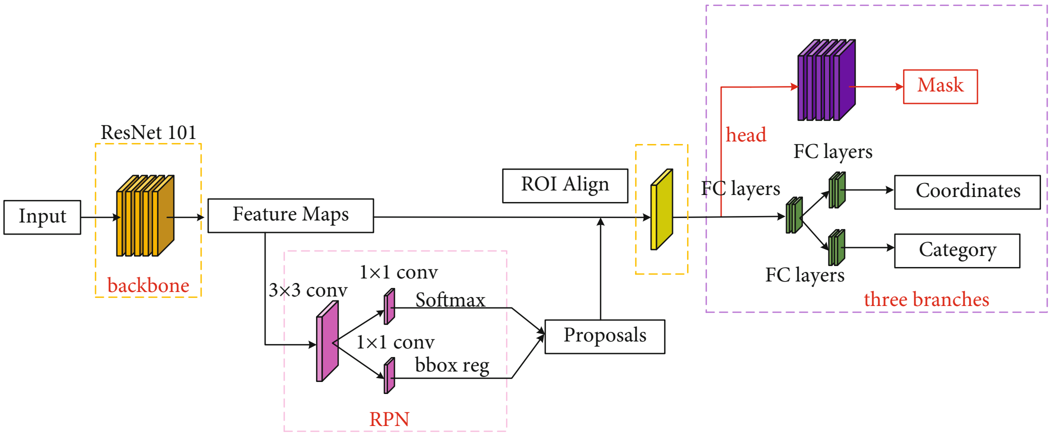


Figure 3: The architecture of Mask R-CNN. From left to right are the input, backbone, RPN, ROI align, and three branches of output.

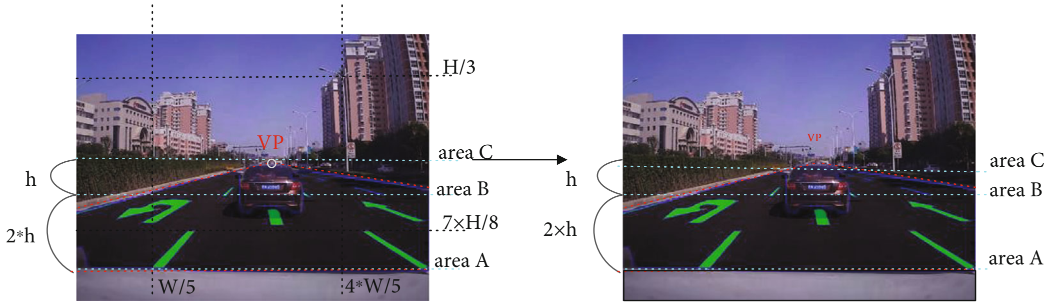


Figure 4: Driving area division process. After segmenting the overall road information, a large number of redundant areas are removed. The area where there may be lane lines is reduced to the near field of view of the image in the segmented road area.

segment. Then, we use the least-squares line combined with the feature points to cluster the feature lines and judge the continuity of the horizontal position of the line segments according to the feature points. The clustering equations are

$$x = B(k) \times y + B(b), \tag{4}$$

$$B(k) = \frac{\sum_n x \times \sum_n y - \sum_m (x \times y)}{\left(\sum_n y\right)^2 - \sum_n y^2}, \tag{5}$$

$$B(b) = \frac{\sum_n (x \times y) \times \sum_n y - \sum_n y^2 \times \sum_n x}{\left(\sum_n y\right)^2 - \sum_n y^2}. \tag{6}$$

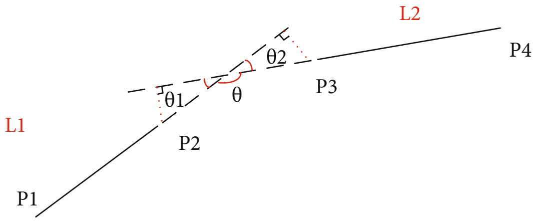


Figure 5: Distance similarity.

The horizontal constraint is using the following equation.

$$\left|F(x) - B(k) \times F(y) - B(b)\right| < d, \tag{7}$$

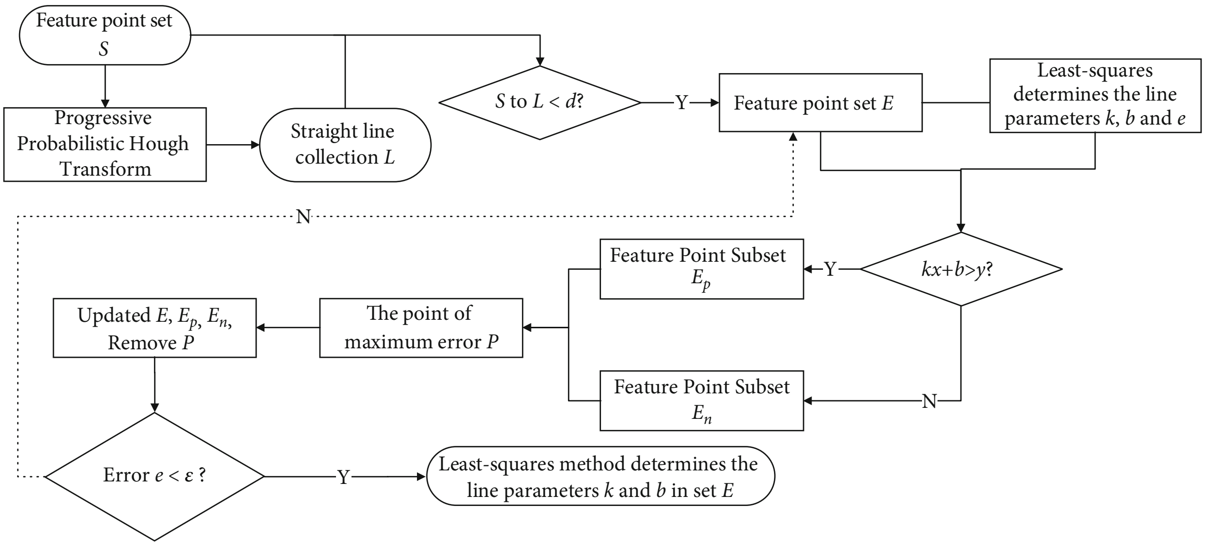


Figure 6: Straight line fitting process.

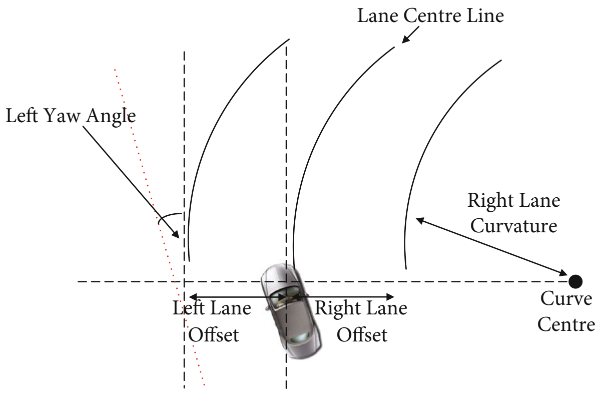


Figure 7: Curve fitting diagram.

where d is the maximum deviation distance of the feature points. Under the perspective effect of the image, d decreases with distance. The feature-point clustering effect under this constraint is shown in Figure 5.

3.4. Fitting Model Design. To get the optimality parameter equation of the lane line which can be obtained, we comb the linear and polynomial fitting methods of feature points in different fields of view.

3.4.1. Straight Line Fitting Model. To ensure that the regression line mainly includes normal data points of which the error is close to zero, the noise points with maximum error on both sides of the regression line are removed. Considering the advantages and disadvantages of HT and LS, we propose a line-detection method that combines the two algorithms. First, HT is used to determine the approximate region of the line, and then, the improved least-squares method is used to determine the line parameters based on the specific point of each region after clustering. The algorithm flow is shown in Figure 6.

(1) Under the parameters, a probability-based Hough transform operation is performed on the lane-line feature to obtain a straight line

(2) For each line obtained by HT detection, in all feature point sets S, feature points whose distance line is not greater than d are found, to form a set E

(3) The regression line parameters K and B of set E are determined and the standard deviation by the least-squares method

(4) Any feature point (x_i, y_i) in set E, all feature points satisfying $kx_i + b > y_i$ form a subset E_{pos}, all feature points satisfying $kx_i + b < y_i$ form a subset E_{neg}

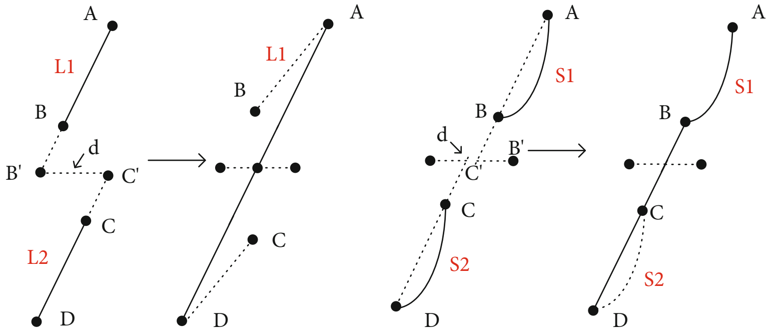
(5) In set E_{pos} and E_{neg}, find the point with the greatest error:

$$P_p = \arg\max{}_{P_p \in E_{\text{pos}}}^{d(P_p)} \quad P_n = \arg\max{}_{P_n \in E_{\text{neg}}}^{d(P_n)}, \tag{8}$$

where $d(p)$ represents the distance from point P to the regression line.

(6) Points P_p and P_n are removed; then, set E_{pos} and E are updated. Repeat step 3 until the error e less than ε

3.4.2. Complex Road Curve Fitting Model. For intelligent vehicles with simple functions and low-speed requirements, the lane-line position information obtained by the linear model can basically meet the requirements of lane-line detection, but it cannot adapt to the real-time road environment of different shapes. The commonly used parabolic curve-fitting model cannot adapt at the junction of straight lines and curves. We propose the following third-order



(a) (b)

Figure 8: Near- and medium-field merging.

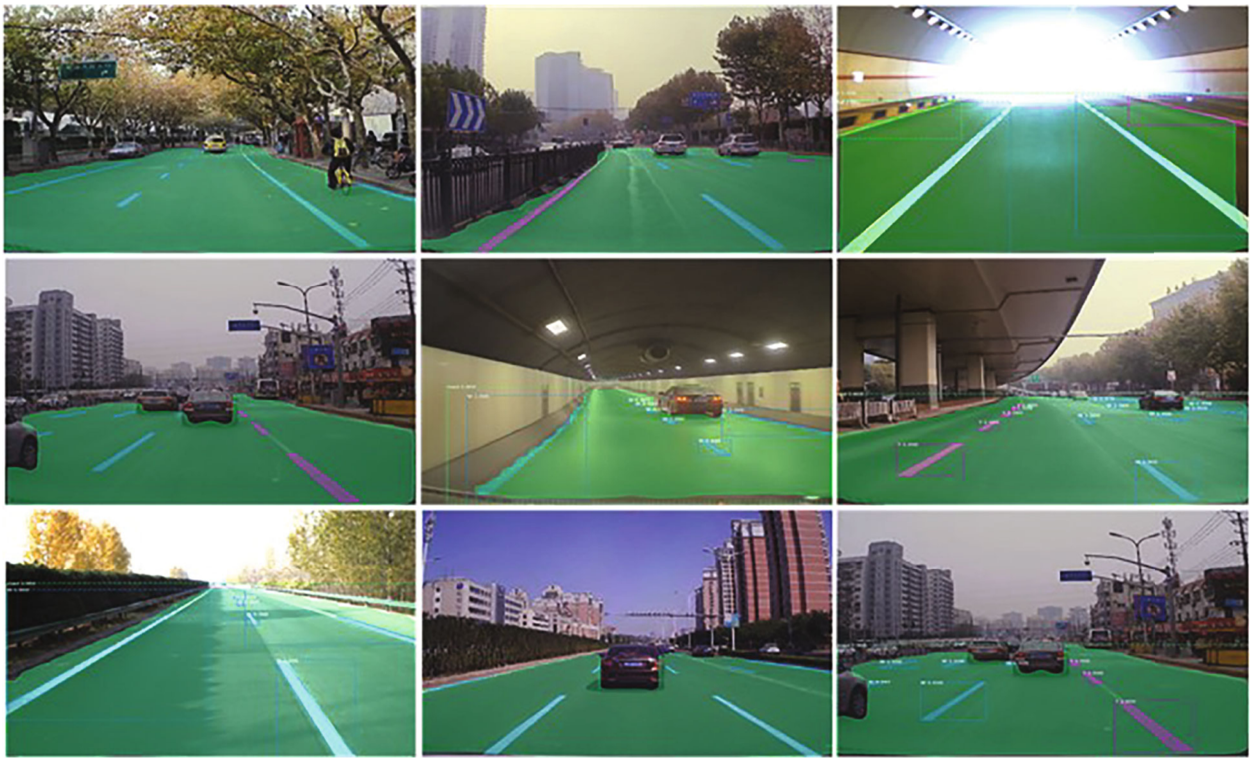


Figure 9: Mask R-CNN detection results in different scenarios.

polynomial fitting model as follows.

$$f(x) = l_0 + l_1 x + \frac{l_2}{2} x^2 + \frac{l_3}{6} x^3. \quad (9)$$

Figure 7 shows an example of a lane model with two lanes and one exit. The model consists of n segments, each with m points. We use the third-order polynomials $f(x)$ to fit each lane line with the points of the line segment. During the fitting process, each segment is an independent cubic spline model, where L_0, L_1, L_2, and L_3 constitute the parameter information of the function. L_0 represents the lane boundary offsets, L_1 is used to drive the heading or yaw angle, L_2 is used to derive the curvature, and L_3 is used to derive the rate of change of the curvature. The left and right lane models are modeled separately, which also makes it possible to model nonparallel lane boundaries.

3.5. Complex Road Multilane Line Constraint

3.5.1. Straight Line Merging Algorithm. A straight line is obtained after the Hough transform, and most of it is valuable lane line information. Some noise still must be eliminated. In principle, a lane line can only be matched by a straight line in the near or middle field. Therefore, we want to merge the straight lines in the near and middle fields of view. This paper introduces the similarity measure for evaluation. As shown in Figure 5, we cluster lines of similar distance and direction into one class and then use least squares

Table 1: Lane detection results on TuSimple dataset.

Method	IoU	Runtime (s)
FCN [36]	0.76	0.20
VPGNet [26]	0.89	0.41
LaneNet [37]	0.91	0.62
SCNN [38]	0.94	0.21
Ours	0.95	0.15

Table 2: Result of accuracy under different fitting methods.

Dataset	LS	Spline	Bezier	Ours
TSD-00000	93.9	93.4	94.8	98.6
TSD-00001	92.4	92.7	93.9	96.6
TSD-00002	92.5	92.6	93.4	97.1
TSD-00003	92.6	92.5	93.7	97.8
TSD-00004	92.3	93.1	94.6	96.8
TSD-00005	93.4	92.3	93.2	98.1
TSD-00006	92.9	93.7	94.2	97.4
TSD-00007	92.7	92.5	94.9	97.2
TSD-00008	92.7	93.6	94.0	98.5
TSD-00009	92.4	91.3	93.4	96.4
TSD-00010	94.1	93.8	95.2	99.2

to fit the lane-line feature points on all the lines.

$$\begin{cases} |k_1 - k_2| < T_k, \\ \left|xB' - xC'\right| < T_{d1}, \end{cases} \tag{10}$$

$$\begin{cases} dis = |(x_3 - x_2)\sin\theta_1 - (y_3 - y_2)\cos\theta_1| + \\ |(x_3 - x_2)\sin\theta_2 - (y_3 - y_2)\cos\theta_2|, \end{cases} \tag{11}$$

$$dir = |\theta_1 - \theta| + |\theta_2 - \theta|.$$

$P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ are the two end points of the line L_1, and the inclination angle is θ_1. $P_3(x_3, y_3)$ and $P_4(x_4, y_4)$ are the two end points of the line L_2, the inclination angle is θ_2. The inclination angle of the straight line made up by points P_2 and P_3 is θ.

3.5.2. Different Area Merging Algorithm. After getting the candidate marker lines in the different fields, we also need to merge the near midfield lines. For segmented lanes, each zone model needs to be connected first, and then, the lane model is built. The connection method is divided into a straight-line model and a curve model. The linear connection compares the slope of the line; the curve compares the curvature of the curve at the same point and determines the connection according to the distance between the segments to be merged. The specific method is as follows.

Table 3: Lane line detection accuracy under different scenes in TSD-MAX dataset.

Factor	S	MPR (%)	FPR (%)	TPR (%)
Shadow	9,925	0.5	1.7	97.8
Damaged	11,343	1.4	0.5	98.1
Underexposed	6,243	1.8	1.7	96.5
Overexposed	15,343	1.7	0.8	97.5
Fog	9,561	3.1	1.0	95.9
Rain	13,290	0.6	0.9	98.5
Snow stain	8,424	1.3	3.5	95.2
Night	7,898	1.2	0.6	98.2

As showed in Figure 8(a), in the straight-line connection mode, A and B are the two end points of the straight line L_1 and C and D are the two end points of the straight line L_2. B_0 and C_0 are the coordinates of the vertical axis to the intersection of the two straight lines on the dividing line. K_1 and K_2 indicate the slopes of straight lines L_1 and L_2, respectively. If the condition of equation (9) is satisfied, then A and D are connected to form a combined line segment. T_k is the slope difference threshold for merging. T_{d1} is the intercept difference threshold of merging. It can be set according to the actual situation.

As shown in Figure 8(b), in the connection mode at the curve, if B of curve S1 is above C of S2, we extend the two end points S1 and S2 to B0 and C0 of the divider. If formula (9) is satisfied, we use point B of S1 and points C and D of S2 to determine the new curved line segment after the merge. T_{d2} is the threshold at the time of the merge. However, when the connection is actually used, the selection of the threshold has a great impact; there are fluctuations in different sections of the test, and it is not easy to set manually.

$$\left|xB' - xC'\right| < T_{d2}. \tag{12}$$

Actually, from the lane-line model design, we have made an ideal assumption that the lane lines in the mid- and near-field regions are approximately linear. This greatly reduces the lack of connection at the corners.

4. Experiment and Analysis

4.1. Datasets and Validation. Our experiment was designed mainly to detect the real road information in real time, so the training and test datasets consisted of real road information. In this paper, we used the TuSimple lane dataset and TSD-MAX traffic scene dataset to verify the effectiveness of our method.

It labeled the lanes of the test dataset and checked the accuracy of the IPM picture. We defined the criteria for the lane line as follows: where the input discrete point is regarded as the positive checkpoint in the true value area, TP is the total length of the positive lane-mark; FN is the total length of the missed check, and it equals the total marked true value minus TP; FP is the total length of a false check, and its value is equal to all extracted lane-mark points minus TP. For Mask R-CNN detection accuracy intersection over union (IoU), the following formula (13) is also applicable.

$$S(\mathrm{IoU}) = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}. \tag{13}$$

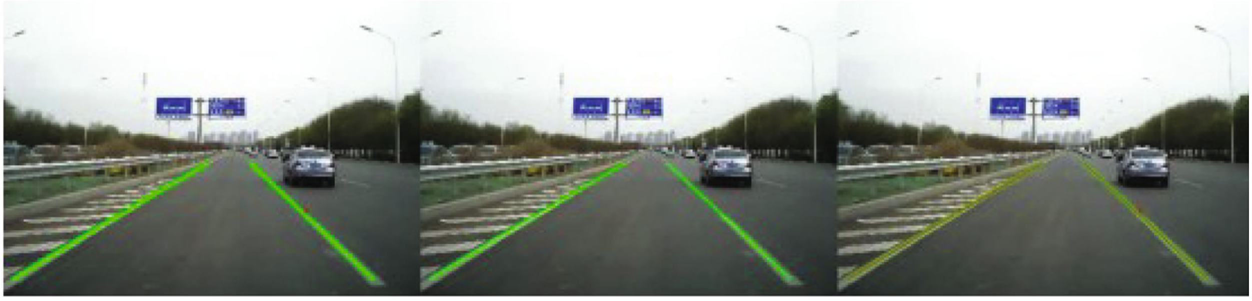


Figure 10: Straight line scenes.

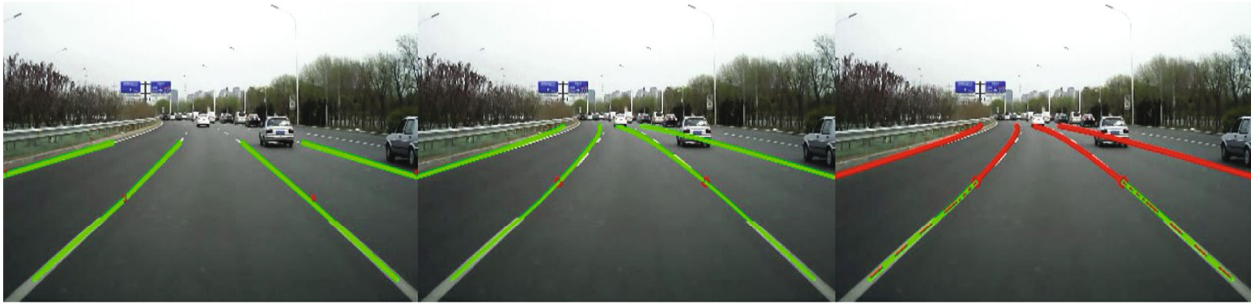


Figure 11: Multilane scenes.

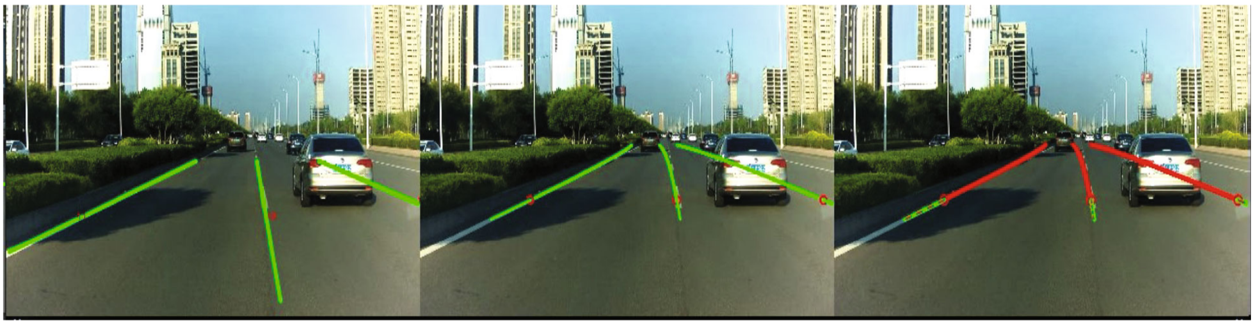


Figure 12: Block scenes.

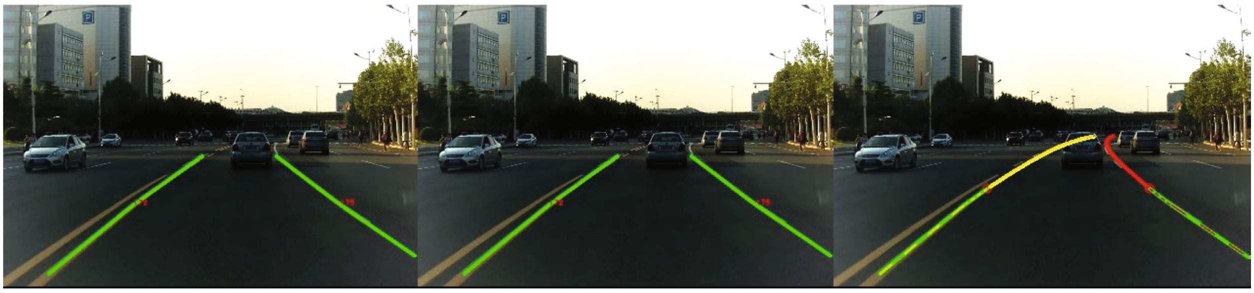


Figure 13: Dusk scenes.

4.2. Complex Road Lane Line Detection Results. In the driving process of an intelligent vehicle, due to the complexity and diversity of the driving environment, false and missed detections often occur. Our proposed deep learning-based lane-marker extraction algorithm can well shield the influence of these uncertain factors on the lane-marker extraction results, so as to adapt to the complex and varied real road environment. Figure 9 shows the results of training using our improved model. As is shown in Figure 9, our model clearly identifies all the information.

Table 1 shows the comparison of our improved method and some semantic segmentation methods on the TuSimple

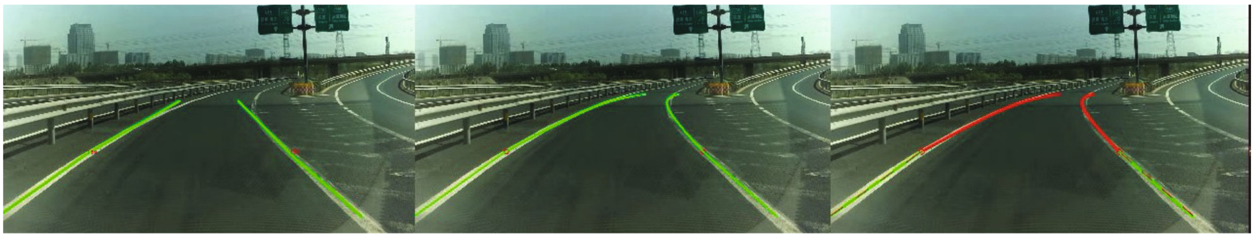


Figure 14: Curve scenes.

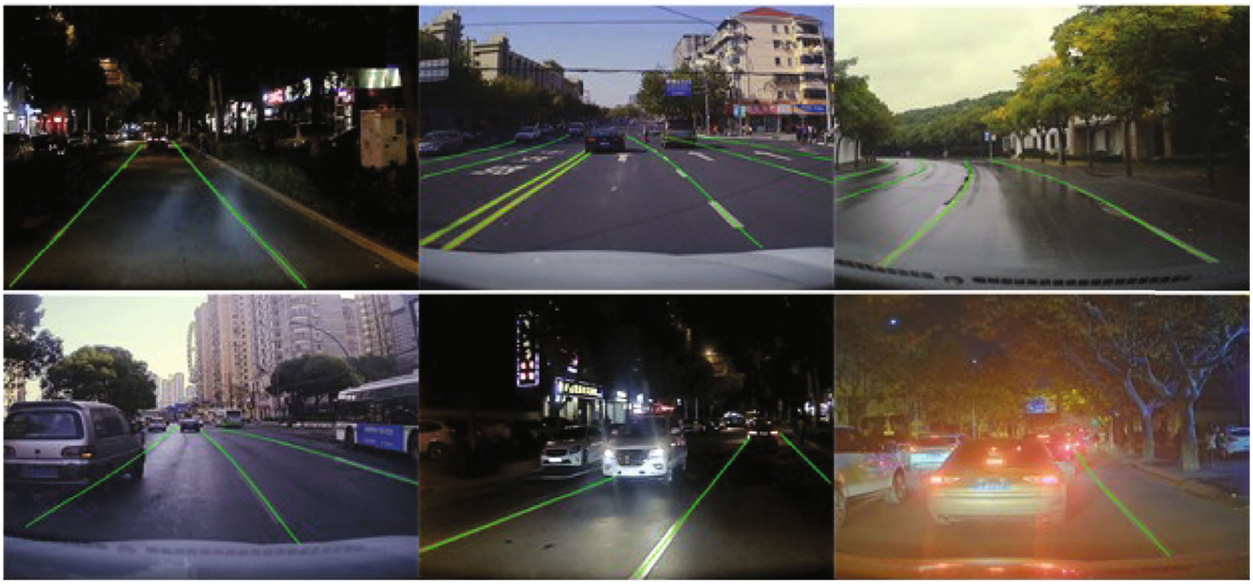


Figure 15: Other complicated scenes.

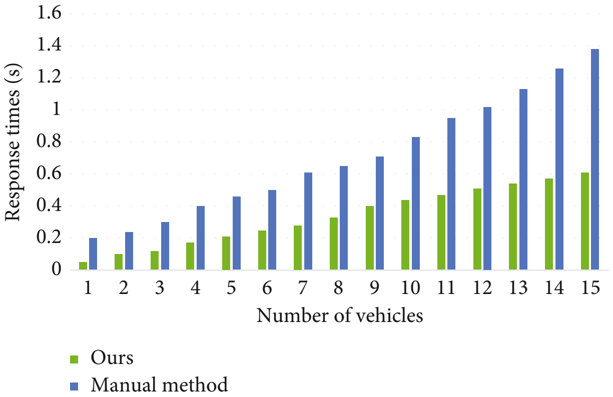


Figure 16: Comparison of response time.

lane dataset. As shown in Table 1, our method is more accurate than most of the state-of-the-art methods on the TuSimple lane test dataset. SCNN has certain requirements on the location distribution of lane lines, and it is suitable for the TuSimple lane dataset which have a fixed overall linear number and less scene changes. At the same time, Table 1 shows the processing speed comparison of the model. Our method takes only 150 ms to process a picture, which is better than other methods.

4.3. Complex Road Lane Line Fitting Results. After extracting the feature points, we also need to fit the lane-line feature points. To verify the robustness of the lane-line fitting algorithm, we performed current-lane detection and multilane-line detection on multiple datasets. As shown in Table 2, the average accuracy was 97.61%. Our method is more accurate than other fitting algorithms. In the meantime, combining linear and polynomial fitting methods of feature points in different fields of view has higher precision than other algorithms. Therefore, the robustness of the proposed algorithm was verified by the detection accuracy.

Table 3 shows the detection accuracy in different environments. S is the total number of frames, MPR is the missed positive rate, FPR is the false positive rate, and TPR is the true positive rate. The calculation formulas of FPR and TPR are (14) and (15), respectively.

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{FN}}, \tag{14}$$

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{TN}}. \tag{15}$$

Figures 10–14 show the results of fitting the lane lines in different scenes. The first column is the result of the line fitting, the second is the result of the curve fitting, and the third is the final fitting of the line and the curve result. Other complicated situations are shown in Figure 15. As can be seen from the figure, our fitting algorithm fits the real lane line better, and it is also very good at the corner.

4.4. Competition Results. Figure 16 shows the response time comparison between our method and the traditional method. The results show that our method has a faster response speed. This shows that 5G-V2X has greatly improved the perception of complex roads and has met the real-time requirements.

5. Conclusions

In this article, we propose a real-time road perception method based on deep learning and 5G-V2X. Compared with the traditional method, this method has higher road perception ability and faster response time. Using linear fitting and polynomial fitting methods of feature points in different fields of view, the lane markings can be extracted robustly under various complex practical conditions, and the optimal parameter equations of the lane lines can be obtained. The experimental results show that the method adopted in this paper can better adapt to various types of road scenes. The algorithm has good detection effect in different scenarios, fast processing speed, good fitting effect, wide application scenarios, and strong robustness. In future work, we will focus on the optimization of the fitting algorithm to further improve the real-time performance in a dense traffic scenario of the proposed method.

Data Availability

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Beijing Municipal Commission of Education Project (Nos. KM202111417001 and KM201911417001), the 2021 Vocational Education Project of China Association of Vocational and Technical Education (Grant No. SZ21C037), the National Natural Science Foundation of China (Grant Nos. 62102033, 61906017, 62006020, 61871039, and 62171042), the Collaborative Innovation Center for Visual Intelligence (Grant No. CYXC2011), and the Academic Research Projects of Beijing Union University (Nos. ZB10202003, ZK40202101, ZK120202104, and BPHR2020DZ02).

References

[1] C. Xu, H. Liu, Y. Zhang, and P. Wang, "Mutual authentication for vehicular network in complex and uncertain driving," *Neural Computing and Applications*, vol. 32, no. 1, pp. 61–72, 2020.

[2] C. Xu, H. Wu, and H. Liu, "An intelligent scheduling access privacy protection model of electric vehicle based on 5G-V2X," *Scientific Programming*, vol. 2021, 11 pages, 2021.

[3] H. Xuan, H. Liu, J. Yuan, and Q. Li, "Robust lane-mark extraction for autonomous driving under complex real conditions," *IEEE Access*, vol. 6, pp. 5749–5765, 2018.

[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *in 2016 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 770–778, Las Vegas, NV, 2016.

[6] S. Ma, P. Guo, H. You, P. He, G. Li, and H. Li, "An image matching optimization algorithm based on pixel shift clustering RANSAC," *Information Sciences*, vol. 562, pp. 452–474, 2021.

[7] 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.

[8] D. Peng, S. Xiong, W. Peng, and J. Lu, "LCP-Net: a local context-perception deep neural network for medical image segmentation," *Expert Systems with Applications*, vol. 168, p. 114234, 2021.

[9] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: multi-path refinement networks for high-resolution semantic segmentation," in *" in 2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 5168–5177, Honolulu, HI, 2017.

[10] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun, "Large kernel matters — improve semantic segmentation by global convolutional network," in *in 2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 1743–1751, Honolulu, HI, 2017.

[11] J. Tian, J. Yuan, and H. Liu, "Road marking detection based on mask R-CNN instance segmentation model," in *2020 international conference on computer vision, image and deep learning (CVIDL)*, pp. 246–249, IEEE, 2020.

[12] R. Bhandari, A. U. Nambi, V. N. Padmanabhan, and B. Raman, "Driving lane detection on smartphones using deep neural networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 16, no. 1, pp. 1–22, 2020.

[13] Z. Huang, L. Huang, Y. Gong, C. Huang, and X. Wang, "Mask scoring R-CNN," in *in 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6402–6411, Long Beach, CA, USA, 2019.

[14] K. Chen, "Hybrid task cascade for instance segmentation," in *in 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4969–4978, Long Beach, CA, USA, 2019.

[15] K. Wang and M. Z. Liu, "Object recognition at night scene based on DCGAN and faster R-CNN," *IEEE Access*, vol. 8, pp. 193168–193182, 2020.

[16] H. Liu, Q. Zhang, C. Xu, and Z. Ye, "A novel FastICA algorithm based on improved secant method for intelligent drive," *Journal of Intelligent & Fuzzy Systems*, vol. 40, no. 1, pp. 165–178, 2021.

[17] Z. Zhao, X. Li, H. Liu, and C. Xu, "Improved target detection algorithm based on libra R-CNN," *IEEE Access*, vol. 8, pp. 114044–114056, 2020.

[18] A. Rouari, A. Moussaoui, Y. Chahir, H. T. Rauf, and S. Kadry, "Deep CNN-based autonomous system for safety measures in

logistics transportation," *Soft Computing*, vol. 25, no. 18, pp. 12357–12370, 2021.

[19] A. Mumuni and F. Mumuni, "CNN architectures for geometric transformation-invariant feature representation in computer vision: a review," *SN Computer Science*, vol. 2, no. 5, pp. 1–23, 2021.

[20] L. Chen, S. Lin, X. Lu et al., "Deep neural network based vehicle and pedestrian detection for autonomous driving: a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3234–3246, 2021.

[21] H. Li, N. Li, R. Wu, H. Wang, Z. Gui, and D. Song, "GPR-RCNN: an algorithm of subsurface defect detection for airport runway based on GPR," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3001–3008, 2021.

[22] S. Mohanapriya, "Instance segmentation for autonomous vehicle," *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 9, pp. 565–570, 2021.

[23] W. Liang, P. Xu, L. Guo, H. Bai, Y. Zhou, and F. Chen, "A survey of 3D object detection," *Multimedia Tools and Applications*, vol. 80, no. 19, pp. 29617–29641, 2021.

[24] J. Li, X. Shi, J. Wang, and M. Yan, "Adaptive road detection method combining lane line and obstacle boundary," *IET Image Processing*, vol. 14, no. 10, pp. 2216–2226, 2020.

[25] S. König, M. Svetlosak, M. Grabowski et al., "Utilization and perception of same-day discharge in electrophysiological procedures and device implantations: an EHRA survey," *EP Europace*, vol. 23, no. 1, pp. 149–156, 2021.

[26] S. Lee, "VPGNet: vanishing point guided network for lane and road marking detection and recognition," in *in 2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 1965–1973, Venice, 2017.

[27] J. Yang, C. Wang, B. Jiang, H. Song, and Q. Meng, "Visual perception enabled industry intelligence: state of the art, challenges and prospects," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 3, pp. 2204–2219, 2021.

[28] Q. Luo, X. Fang, L. Liu, C. Yang, and Y. Sun, "Automated visual defect detection for flat steel surface: a survey," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 3, pp. 626–644, 2020.

[29] D. K. Dewangan and S. P. Sahu, "Driving behavior analysis of intelligent vehicle system for lane detection using vision-sensor," *IEEE Sensors Journal*, vol. 21, no. 5, pp. 6367–6375, 2020.

[30] C. Xu, X. Huang, M. Ma, and H. Bao, "An anonymous handover authentication scheme based on LTE-A for vehicular networks," *Wireless Communications and Mobile Computing*, vol. 2018, 15 pages, 2018.

[31] C. Xu, X. Huang, and M. Ma, "A secure and efficient message authentication scheme for vehicular networks based on LTE-V," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 12, no. 6, pp. 2841–2860, 2018.

[32] C. Xu, H. Liu, P. Li, and P. Wang, "A remote attestation security model based on privacy-preserving blockchain for V2X," *IEEE Access*, vol. 6, pp. 67809–67818, 2018.

[33] C. Xu, H. Luo, H. Bao, and P. Wang, "STEIM: a spatiotemporal event interaction model in V2X systems based on a time period and a raster map," *Mobile Information Systems*, vol. 2020, 20 pages, 2020.

[34] C. Xu, H. Liu, Z. Pan, W. Li, and Z. Ye, "A group authentication and privacy-preserving level for vehicular networks based on fuzzy system," *Journal of Intelligent & Fuzzy Systems*, vol. 39, no. 2, pp. 1547–1562, 2020.

[35] T. Liu, C. Xu, H. Liu, X. Li, and P. Wang, "A vehicle detection model based on 5G-V2X for smart city security perception," *Wireless Communications and Mobile Computing*, vol. 2021, 11 pages, 2021.

[36] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

[37] D. Neven, B. D. Brabandere, S. Georgoulis, M. Proesmans, and L. V. Gool, "Towards end-to-end lane detection: an instance segmentation approach," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, 2018.

[38] X. Pan, J. Shi, and P. Luo, "Spatial as deep: spatial CNN for traffic scene understanding," *in Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.