Hindawi
Journal of Sensors
Volume 2019, Article ID 7174602, 11 pages
https://doi.org/10.1155/2019/7174602

Research Article

Near-Infrared Road-Marking Detection Based on a Modified Faster Regional Convolutional Neural Network

Junping Hu,[1] Shitu Abubakar,[1] Shengjun Liu,[2] Xiaobiao Dai,[1] Gen Yang,[1] and Hao Sha[1]

[1]*Department of Vehicle Engineering, College of Mechanical and Electrical Engineering, Central South University, 932 Lushan South Road, Changsha, 410083 Hunan, China*
[2]*School of Mathematics and Statistics, Central South University, 932 Lushan South Road, Changsha, 410083 Hunan, China*

Correspondence should be addressed to Shitu Abubakar; abubakarshitu88@gmail.com

Received 30 July 2019; Revised 2 November 2019; Accepted 5 November 2019; Published 27 December 2019

Academic Editor: Antonio Martinez-Olmos

Pedestrians, motorist, and cyclist remain the victims of poor vision and negligence of human drivers, especially in the night. Millions of people die or sustain physical injury yearly as a result of traffic accidents. Detection and recognition of road markings play a vital role in many applications such as traffic surveillance and autonomous driving. In this study, we have trained a nighttime road-marking detection model using NIR camera images. We have modified the VGG-16 base network of the state-of-the-art faster R-CNN algorithm by using a multilayer feature fusion technique. We have demonstrated another promising feature fusion technique of concatenating all the convolutional layers within a stage to extract image features. The modification boosts the overall detection performance of the model by utilizing the advantages of the shallow layers and the deep layers of the VGG-16 network. The training samples were augmented using random rotation and translation to enhance the heterogeneity of the detection algorithm. We have achieved a mean average precision (mAP) of 89.48% and 92.83% for the baseline faster R-CNN and our modified method, respectively.

1. Introduction

With the current increasing rate of technological advancement, the numbers of automobile users are on the increase. Thus, it is paramount for drivers to interpret and understand the road markings to make the traffic safer [40]. Pedestrians, motorist, and cyclist are being killed at night as a result of poor vision and negligence of drivers to promptly identify a particular road marking [41, 44]. Intelligent driver-assistance systems need to be developed to mitigate the increasing rate of road accidents and assist the human driver in detecting and recognizing road markings [1, 5, 45, 46].

It was reported that 20–50 million people sustain physical injury yearly, and over a million die as a result of traffic accidents [33]. Traffic accidents are mostly caused by driver behavior at the wheel and inadequate road infrastructure. In this regard, the advanced driver-assistance system (ADAS) is being developed and incorporated into some modern automobile [45]. Developing a driver-assistance system that would complement the driver skills by prompting about impending danger or right direction to maneuver ahead would go a long way in reducing the rate of traffic accidents [24, 31, 42]. Identifying road markings in images are tasking; this is due to the occlusion of the road markings by vehicles, variation in view angle, changes in illumination condition, complex background, poor image quality, and shadow [4, 13, 31, 49]. Road-marking and traffic-sign detection systems are vital for driver-assistance systems and autonomous vehicles [6]. Conventionally, road markings can be of different types such as arrows, lane markings, pedestrian crossing, speed limits, and texts [23, 45, 46]. Lots of specific road-marking detection and recognition task were carried out focusing on lane markings [13], crosswalks, and arrows [24]. The current trend in nighttime object detection has far-infrared (FIR) and near-infrared cameras as reliable image acquisition systems [14]. Lots of researchers have reported the application of object detections using near-infrared (NIR) images. Govardha and Pati [14] use a NIR camera to detect pedestrians in nighttime vision. In their research, a combination of Haar-Cascade and Histogram of

Oriented Gradients-Support Vector Machine (HOG-SVM) was used as feature extractors. Han and Song [22] utilize the near-infrared camera for nighttime detection. They used aggregated channel features and AdaBoost methods to achieve the detection task. Dai et al. [37] integrated nighttime near-infrared images using a convolutional neural network to detect pedestrians. The images were acquired using a car-mounted NIR camera. The NIR images were evaluated using the convolutional neural networks, and satisfactory results were obtained. However, the current trend on deep learning detection focused on unsupervised detection task since deep learning framework requires large data for training to effectively generalize on the evaluation dataset [38]. NIR is one of the devices that can be used to detect road markings at night. It provides a good vision and sound image quality at night [21, 22].

Classical computer vision approach and deep learning-based approach are the two most commonly used techniques for traffic-sign detection. In classical methods, handcrafted algorithms such as Histogram of Oriented Gradient (HoG), Scale-Invariant Feature Transform (SIFT), Speeded Up Robust Features (SURF) are used. The classical approach is not too accurate since feature extractors are handcrafted [2, 3, 49].

In recent times, there was a rapid advancement of deep learning technology. Convolutional neural networks and alongside some detection algorithms have shown great prospect in image detection and classification [4, 16, 50]. The successes recorded by AlexNet [17] and Zeiler and Fergus [18] inspired researchers towards utilizing the convolutional neural network as a feature extractor. After the success of AlexNet, several modifications on the CNN classification models were presented. Road markings are detected using methods such as YOLO and SSD [12] or a region proposal-based method. The region-based method outperformed the sliding window search method by reducing the number of the proposals and searching time [8]. Many region proposal algorithms such as selective search [8] and EdgeBox [1] have yielded impressive results in object detection. Regional-based Convolutional Neural Network (R-CNN) has gained considerable popularity in road-marking detection in recent years [1]. The fast R-CNN and faster R-CNN evolved from the R-CNN. They are both region-based detection networks with the later outperforming the former in terms of detection accuracy [11].

The faster R-CNN consist of a base network usually ZF Net, VGG, and ResNet, from which feature maps are obtained, region proposal network (RPN), RoI pooling layers, and classification and regression networks. An RPN is a subnetwork that uses anchors at each pixel position of an image for a class-agnostic classification. The RPN and the fast R-CNN network can be trained separately or jointly. Some reasonable number of RoIs are fed into the RoI pooling layer after which the classification and detection task are performed [32]. The VGG16 has been widely used as a feature extractor in faster R-CNN. The network has thirteen convolutional layers, which are divided into stages. Within each of the stages, feature maps are preserved and they are downsampled by half after passing through a pooling layer. By passing through consecutive stages, final feature maps are obtained. These feature maps are rich in semantic information but lack position information. To overcome this challenge, many researchers have presented the application of feature fusion techniques in object detection to utilize both the position and the semantic information of a feature map [27, 28]. Road-marking detection is a vital aspect of the Intelligent Transportation System (ITS), and research is still ongoing in this area. Literature exists on road-marking detection using classical computer vision. Danescu and Nedevschi [23] developed a road-marking detection and recognition system using a two-step segmentation technique. The classification accuracy obtained ranges from 80 to 95% under different scenarios. Foucher et al. [24] proposed a robust algorithm for detecting and classifying crosswalks and arrows using city images. Their findings show that 90% of crosswalks and 78% of arrows were successfully detected. Qin et al. [20] studied four different types of road markings using a machine vision approach. The marking contours were extracted randomly from the image using an image processing technique. The extracted features were then sent to the classification and the detection modules. Chen et al. [42] utilized a binarized normed gradient (BING) algorithm to speed up the recognition of the detection target. The detected object is classified using a PCANet classifier. In this method, feature extractors are handcrafted. Thus, the method is not too accurate and not suitable for real-time road-marking detection.

Many researchers have reported their findings in the detection of road markings using the concept of deep learning [26]. Vokhidov et al. [30] presented a damaged arrow-road-marking recognition model using the convolutional neural network. Experiment using six types of arrow-road markings shows that the recognition model is promising. However, the author did not consider the detection of the road markings. A research on the detection of road markings using a convolutional neural network was also reported by Qian et al. [1]. In their research, a hybrid region proposal algorithm and a fast-region convolutional neural network were used for detecting road markings. However, the study recommends a further search for more accurate and faster detection algorithm.

Similarly, a study on road-marking recognition using a convolutional neural network was reported by Ahmad et al. [35]. The detection of the symbolic road markings was not within their scope of the study. Tian et al. [13] utilize a deep convolutional neural network based on faster R-CNN for lane marking detection only. The experimental results show that an average precision of 68% was obtained. Wen et al. [39] reported a deep learning model for road-marking extraction, recognition, and completion from three-dimensional (3D) mobile laser scanning point clouds. With the modified U-net model, the precision, recall, and F1-score obtained were 95.97%, 87.52%, and 91.55%, respectively.

Despite these beautiful findings from the literature, fewer studies were reported on road-marking detection. Much attention was paid to just recognition of road markings using a convolutional neural network (CNN). The cited works also focused on daytime images, which has better image quality

than the nighttime images. Moreover, the detection of diverse classes of road markings using a near-infrared camera sensor based on a convolutional neural network (CNN) was less studied.

There has been increasing concern on the road-marking detection in the adverse weather conditions such as in the night [19]. Many researchers have recommended the use of a NIR camera for road-marking detection and recognition [21]. Designing a model to detect and recognize all the road markings will be very tasking because some datasets contain text-based information and different countries use different characters to convey information. Thus, we have chosen some essential, frequently occurring road markings. In our research, we have used eleven (11) different kinds of road markings: (a) single markings: pedestrian crossing, forward arrow, left-arrow, right-arrow, turn-around arrow, and slow down sign; (b) dual markings: forward-and-right arrow, forward-and-left arrow, left-and-right arrow, forward-and-turn-around arrow, and left-and-turn-around arrow. Therefore, the objective of this research is to evaluate a modified faster region convolutional neural network algorithm for nighttime road-marking detection with potential application in the advanced driver-assistance system (ADAS).

Our contributions are summarized as follows:

(i) This research is a contribution towards developing nighttime advanced driver-assistance system (ADAS) for automotive application

(ii) We have created a new, real-life driving scenario nighttime road-marking dataset using a car-mounted near-infrared camera with a VIS filter and NIR supplement. We augmented the training samples using random rotation and translation to boost the heterogeneity of the detection algorithm

(iii) We have modified the state-of-the-art faster R-CNN algorithm with a new structure that takes multilayer feature fusion into cognizance. We have demonstrated another promising feature fusion technique of concatenating all the convolutional layers within a stage to extract image features. This modification boosts the overall detection performance of the model by utilizing the advantages of the shallow layers and the deep layers of the VGG-16 network

The remainder of this article is organized in the following manner: Section 2 provides the methodology followed. In Section 3, we presented the results of our experiments. In Section 4, we discussed our findings. Finally, in Section 5, we conclude based on our objectives and highlight the gap for future research. We then provided a list of reference materials.

2. Materials and Methods

The proposed road-marking detection system consists of three networks: (i) feature extractor (VGG-16 network) with feature fusion (ii) detector network—fast R-CNN detector which consists of a RoI pooling layer and the classification and the regression network and (iii) the region proposal network (RPN).

2.1. Feature Extractor (VGG-16). In our proposed method, we chose VGG-16 for the simplicity of its architecture as a feature extractor. A typical VGG-16 network consists of thirteen convolution layers (13) followed by thirteen (13) activation function layers—Rectified Linear Unit (ReLU), three fully connected layers, and five (5) pooling layers. The VGG-16 comprises of stages: the stage one (conv1) has two convolution layers and two ReLU layers and a pooling layer. The convolutional layers in conv1 are denoted as conv1_1 and conv1_2. The stage two (conv2) has two convolution layers, two ReLU layers, and one pooling layer. The convolutional layers in conv2 are denoted as conv2_1 and conv2_2. The stage three (conv3) has three convolution layers, three ReLU layers, and one pooling layer. The convolutional layers in conv3 are denoted as conv3_1, conv3_2, and conv3_3. The fourth stage (conv4) has three convolution layers, three ReLU layers, and one pooling layer. The convolutional layers in conv4 are denoted as conv4_1, conv4_2, and conv4_3. The last stage (conv5) has three convolution layers, three ReLU layers, and one pooling layer [41]. Within each stage, the feature map is preserved [6, 28]. After an image passes through each stage, its feature map shrinks by half. After several convolutions and pooling operations, coarse feature maps are obtained at the last convolution stage [13, 48].

2.2. Feature Fusion. The features from the deep layers have undergone several convolution operations and downsampling. Consequently, the information from these layers becomes deeply semantic yet more abstract [4, 29, 34, 50]. Conversely, those features from the shallow layers are rich with precise positioning information for the object in the image but score low in semantic feature representation ability [41]. Thus, to take the advantages from the shallow layers and the deep layers and to overcome the tradeoff between the spatial resolution of the lower layers and the distinctive semantic features of the deep layers, multilayer feature fusion is imperative [28, 41]. The feature maps obtained from the output of each convolution stage should have sufficient semantic and position information for an excellent performance of the model. The feature map size should be considered accordingly before fusion; a too small feature map supplies insufficient feature information, and too large increases the computation complexity [28]. In previous research, various techniques were used to fuse the shallow layers and the deep layers of convolutional neural networks [27]. In most existing literature [27, 41], features are extracted from the last convolutional layer within each stage. In our approach, we concatenate feature maps from the convolution layers within each stage. Since within each convolution stage, the feature map is preserved. Therefore, we concatenated the feature maps without resizing. Also, the channel dimensions were sized using a 1×1 convolutional layer. This convolutional layer can be used to increase or reduce the number of filters. By reducing the number of filters, the amount of training parameters in the network decreases. We then normalized the feature maps from conv1,

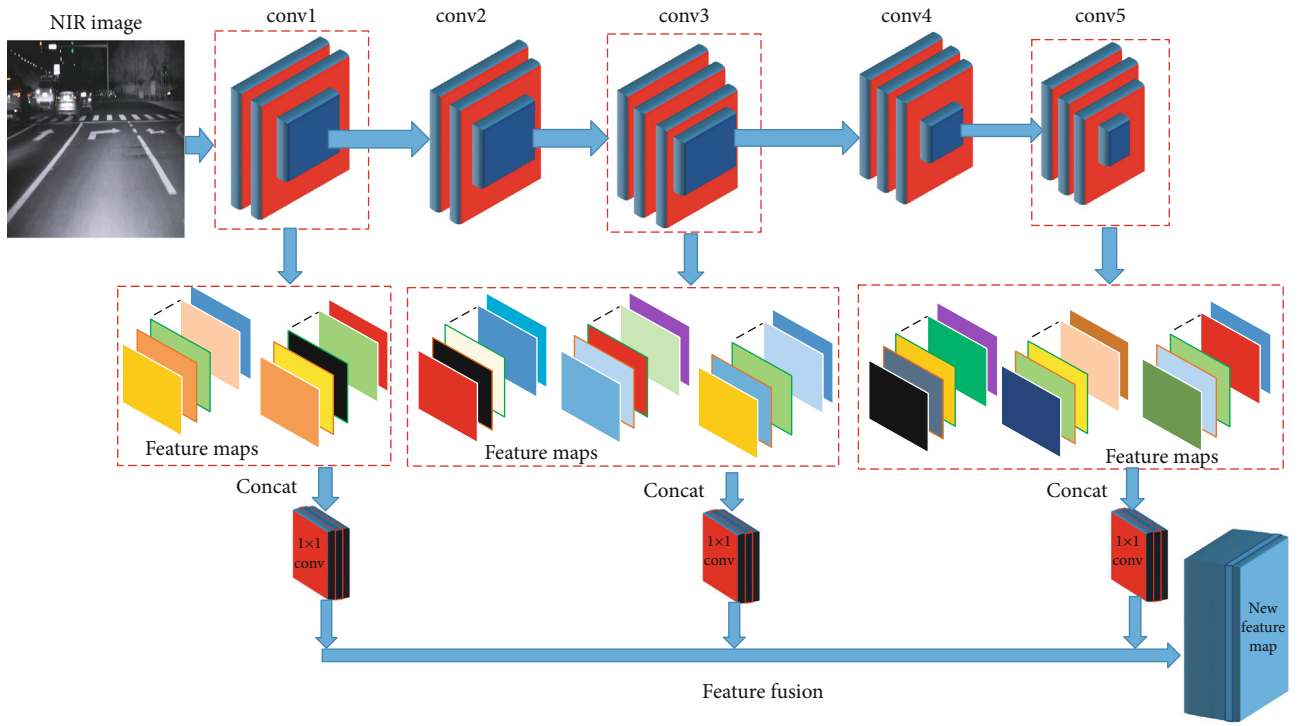

Figure 1: The proposed feature fusion module.

conv3, and conv5 using batch normalization. This concept of batch normalization was followed by Yan et al. [47]. After that, the convolutional layer feature maps from conv1, conv3, and conv5 of the VGG-16 were concatenated. The feature maps from the conv1 and the conv3 were downsampled to the size of the conv5 feature maps. This downsampling technique to the last convolution stage feature map size worked well for [15] in their research application. By setting the downsampling parameters accordingly, the feature maps were converted to the same size. The proposed feature fusion module is shown in Figure 1.

The resulting same-sized feature maps were merged using element-wise addition, and a 1×1 convolutional layer was appended to generate the required amount of the feature maps before being fed to the RPN network. In this way, the advantages of concatenation and summation techniques were utilized. These merits reflect on the mean average precision of the proposed method. The overall detection pipeline is depicted in Figure 2.

2.3. Detector Network. Initially, R-CNN relies on selective search algorithm to generate about 2000 regions of interest (RoIs) [4]. However, R-CNN is too slow for the real-time detection task. An improved version of R-CNN exists which utilizes RPN instead of selective search to generate regions of interest (RoIs). A region proposal network (RPN) takes feature maps as input and produces thousands of proposals. An RPN is a 3×3 kernel size fully convolutional layer which branch into two siblings 1×1 convolutional layers for class-agnostic classification and the regression of the RoIs. To generate proposals, we slide a 3×3 sliding window over the last shared convolutional feature map obtained from VGG-16. At each sliding window position, there are nine (9) anchors, which translate through a feature map from which varying sizes of proposals are obtained [9]. The RoIs are sent into two siblings 1×1 convolutional layers for the box-regression and box-classification task [48]. The maximum possible proposal for each location is denoted by k. The classification (cls) layer is assigned 2k scores (there is an object, or there is no object) for each proposal. The architecture of the detection system showing the detailed modifications on the VGG-16 is depicted in Figure 3. The regression layer is assigned 4k parameters (depicting the coordinate location of the proposal) [10]. The intersection over the union between the ground-truth bounding box area (X) and anchor box (Y) is used to determine the presence of a positive and negative proposal [11].

The expression for intersection over union is given by [1]:

$$\text{IoU} = \frac{X \cap Y}{X \cup Y}. \tag{1}$$

When training region proposal network, we consider two class labels (object or not object) to each anchor. We consider two cases for positive anchors: (i) the anchor that shows highest intersection-over-union (IoU) overlap with the ground-truth box or (ii) an anchor that has an intersection-over-union overlap more than 0.7 with any ground-truth box. Similarly, we assigned intersection-over-union of less than 0.3 for negative anchors. Any anchor with IoU greater than 0.3 but less than 0.7 is not considered in the training goal [11]. Anchors come in varying scale and varying aspect ratios; they can be sized accordingly to fit a particular application. The proposals generated by the RPN are fed into the

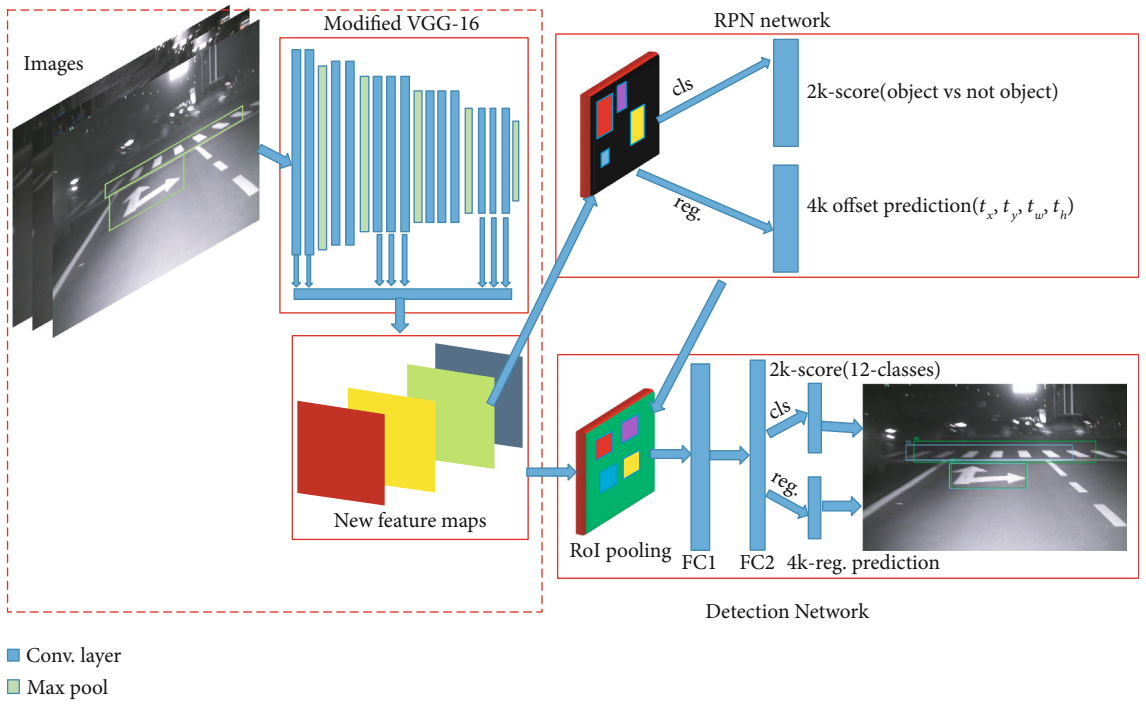

Figure 2: The pipeline of the proposed detection model.

region of interest (RoI) pooling layers. The pooling layer takes a fixed size of the RoIs. After that, the output is fed into the classification and the regression layers for classification and regression, respectively.

2.4. Dataset. The dataset used for training and testing the model was captured using a car-mounted near-infrared camera with NIR supplement from the street of Changsha City, Hunan Province, China. The specifications of the image acquisition system are given in Table 1. In order to increase the generalization ability of our model, data augmentation is necessary since CNN requires large data to train [36]; the default augmentation techniques in faster R-CNN are not suitable for our task. Therefore, we set the horizontal flip, vertical flip, and rotate_90 false. We introduced random translation and random rotation with bounding boxes to augment our dataset. We set the rotation range (0–2) in python which implies that each image from the randomly sampled images is selected to undergo either rotation between -10^0 and 10^0 with probability 0.5 or remain unchanged with the same probability. Similar procedure is followed for translation. The original rectangular-shaped image becomes a parallelogram, and the random rotation function rotates the four points of the ground-truth bounding boxes by an angle θ. Thus, the new image width and the height becomes

$$\begin{bmatrix} w_n \\ h_n \end{bmatrix} = \begin{bmatrix} \sin\theta & \cos\theta \\ \cos\theta & \sin\theta \end{bmatrix} \begin{bmatrix} h \\ w \end{bmatrix}, \quad (2)$$

where h, w and h_n, W_n are the original image height and width and the new image height and width, respectively.

The dataset captured amount to 8536 images, which contains 11981 road markings. We randomly selected 2,800 images for training and validation which is about three-quarter of the total images, while the remaining were set aside for testing and evaluation of the algorithm as per Ren et al. [11] guide. The ground-truth bounding boxes were subsequently labeled manually using MATLAB. Eleven (11) different road markings (12 classes, including background) were considered.

2.5. Training. Our road-marking detection system was implemented with python 3.6 in Pycharm2017 environment installed on a desktop computer with the following specifications: Corei7-8700; 2.6 GHz processor with 12-core CPU; NVIDIA GeForce GTX 1080Ti 8GBRAM. The nighttime detection system was implemented using the source code (with modifications) published in python [7].

We use the NIR images described in Section 2.4 to train the model with stochastic gradient descent algorithm [13]. We jointly trained the detection algorithm with the initial learning rate of 0.0001 for 550 epochs. We also set the momentum and weight decay as 0.9 and 0.0005, respectively. The VGG-16 was initialized using image Net pretrained weights and then fine tuning of the road-marking detection algorithm using the NIR images. We stopped the training process at 550 epochs for 2000 iterations when the training loss remain constant. We resized the images with the shorter side having 512 pixels. We used three scale anchors of 128^2, 256^2, and 512^2, and three aspect ratios of 1:1, 1:2, and 2:1. For RPN training, we consider all the anchors and randomly sampled all anchors to have a minibatch size of 256 with 1:1 foreground and background sample ratio based on our IoU threshold for positive and negative samples.

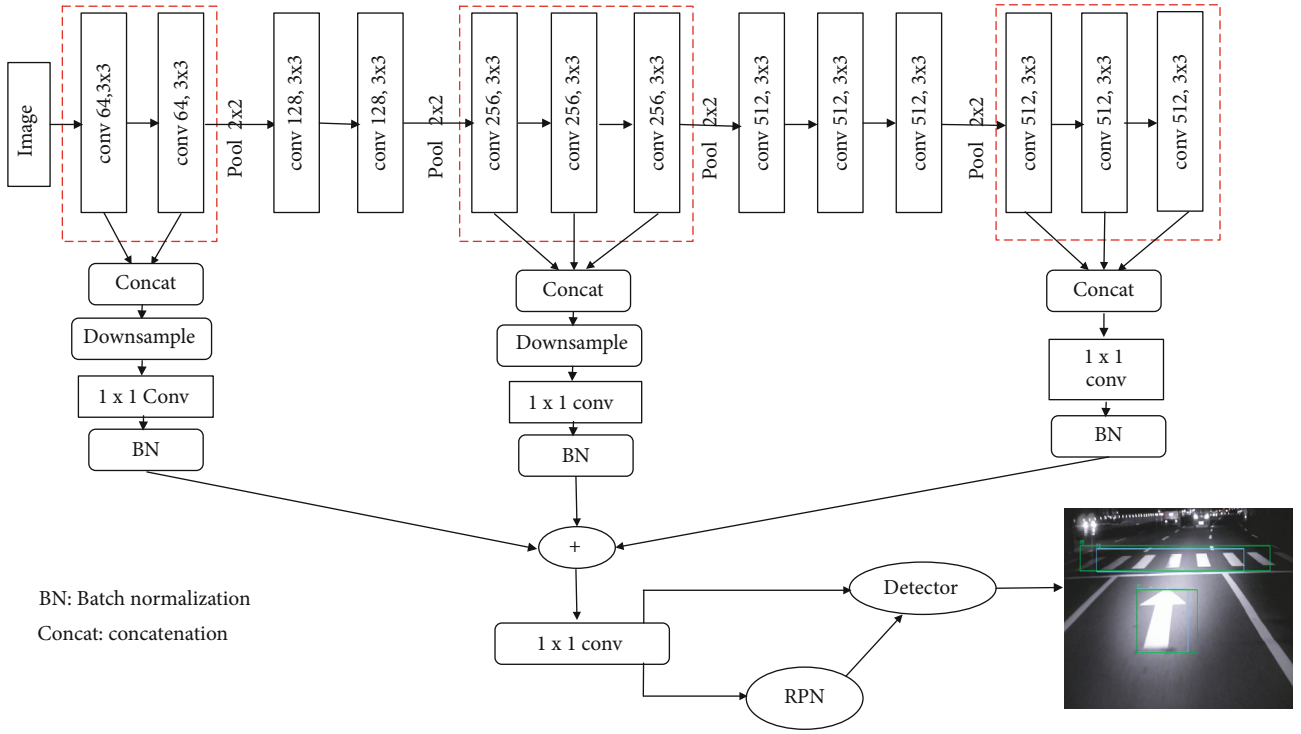

Figure 3: The architecture of the detection system.

Table 1: The specifications of the image acquisition system.

Items	Near-infrared camera
Make	Tongxinyuan
Wave length	850 nm (center band)
Resolution	1280 × 720 pixels
Frame frequency	5 fps

The RPN utilizes all the anchors to compute the classification loss using binary cross entropy. The minibatch identified as foreground is used to calculate the regression loss. The regression loss of the foreground is computed using smooth L1 loss. The nonmaximum suppression (NMS) algorithm was applied to reduce the number of the redundant anchors. The IoU threshold for the suppression and the proposal numbers per image were set to 0.7 and 2000, respectively, from which 300 best proposal per image are used to train our detector network in line with Ren et al. [27]. Similarly, for the detector training, the classification loss is a multiclass cross entropy loss over twelve road markings including background, and smooth L1 loss was used for the prediction offset losses.

We evaluated our detection algorithm using average precision and mean average precision (mAP) with a maximum intersection-over-union (IoU) threshold of 0.5 in line with VOC-2012 requirement. This is as per Ren et al. [11]. The main objective of the experiment was to jointly train the RPN and the detector network to minimize equation (2) given by [11]. In our experiment, the total loss represents the RPN loss and the detector network loss, since the model was jointly trained. The whole detection model was trained end-to-end by back-propagation using stochastic gradient descent (SGD) algorithm.

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{\text{cls}}}\sum_i L_{\text{cls}}\left(p_i, p_i^*\right) + \lambda\frac{1}{N_{\text{reg}}}\sum_i p_i^* L_{\text{reg}}(t_i, t_i^*). \tag{3}$$

The first term in equation (3) represents the classification loss over two classes (i.e., object vs. not object). The second term represents the regression loss of bounding boxes. This term has a value only when $p_i^* = 1$. Here, i is the index of anchor during training the faster R-CNN. p_i is the probability that an anchor i contains an object. p_i^* is the probability of the ground-truth label, $p_i^* = 1$ and $p_i^* = 0$ for positive and negative anchor, respectively. t_i is a vector which represents the four parameterized coordinates of the predicted bounding box, and t_i^* is also a vector which corresponds with the ground-truth box associated with the positive anchor [28]. N_{cls} and N_{reg} are the normalization factor. λ is a weighted balancing parameter. In our approach, $N_{\text{cls}} = 256$ and $N_{\text{reg}} = 2000$ and $\lambda = 10$, to set the N_{cls} and N_{reg} terms in equation (1) approximately equal weighted. L_{cls} is the classification loss which is a log loss over binary classes (i.e., there is object against there is no object). The regression loss L_{reg} is given mathematically as

$$L_{\text{reg}}(t_i, t_i^*) = R(t_i - t_i^*), \tag{4}$$

where R is the smooth L1 loss usually determined as

$$R(x) = \begin{cases} 0.5x^2, & \text{if } f|x| < 1, \\ |x| - 0.5, & \text{else.} \end{cases} \quad (5)$$

The t_i is mathematically expressed as

$$\begin{aligned} t_x &= \frac{(x - x_a)}{w_a}, \\ t_y &= \frac{(y - y_a)}{h_a}, \\ t_w &= \log\left(\frac{w}{w_a}\right), \\ t_h &= \log\left(\frac{h}{h_a}\right), \\ t_x{}^* &= \frac{(x^* - x_a)}{w_a}, \\ t_y{}^* &= \frac{(y^* - y_a)}{h_a}, \\ t_w{}^* &= \log\left(\frac{w^*}{w_a}\right), \\ t_h{}^* &= \log\left(\frac{h^*}{h_a}\right), \end{aligned} \quad (6)$$

where x, y, w, and h are the center coordinates of the predicted box, width, and height, respectively. x_a, y_a, w_a, and h_a represent the anchor centers, width, and height, and x^*, y^*, w^*, and h^* correspond to the ground-truth box; t_x, t_y, t_w, and t_h are the regression coefficients (RPN prediction).

In general, the training and detection procedure of faster R-CNN are summarized as follows:

(i) Pass the images through the VGG-16 net for feature extraction

(ii) Extract about 2000 region proposals for an image obtained from the region proposal network

(iii) Generate a fixed size feature vector from the RoI pooling layer

(iv) Pass the required RoIs to fully connected layers

(v) Train the RPN and detection network by back propagation using stochastic gradient descent algorithm

(vi) Perform classification and detection task to evaluate the trained model using the test images

2.6. Evaluation Metrics. We evaluated our method based on mean average precision (mAP) and false positive rate which is a metric for the false alarm probability as proposed by [13, 25, 28, 41]. These are vital evaluation parameters as used by [43]. Thus, the algorithm performs best when the mean average precision is high and vice versa. The predicted bounding box was compared with the ground truth bounding box to ascertain the validity of the prediction. The prediction is considered true if the IoU is greater than or equal to 50% as per VOC-2012 requirement; otherwise, false [29]. The IoU was computed using equation (1). The precision and the recall and false positive rate are given by [38, 43] as

$$\begin{aligned} \text{precision} &= \frac{\text{number of road markings correctly detected}}{\text{total number of detected road markings}} \times 100\%, \\ \text{recall} &= \frac{\text{number of road markings correctly detected}}{\text{total number of targetted road markings}} \times 100\%, \\ \text{false positive rate} &= \frac{\text{number of false detected road markings}}{\text{total number of detected road markings}} \times 100\%. \end{aligned} \quad (7)$$

3. Results

The results of our experiments are presented in Figure 4 and Table 2.

4. Discussion

First, our road-marking detection algorithm was implemented based on faster R-CNN with VGG-16 as the base network. Then, we run another experiment using our proposed method.

The results obtained with VGG-16 as the base network were used as a baseline for accessing the contribution of our proposed multilayer feature fusion. Figure 4(a) shows the average precisions of forward-and-right arrow (DU1), forward-and-left arrow (DU2), left-and-turn-around arrow (DU3), left-and-right arrow (DU4), left-and-right arrow (DU5), pedestrian crossing (PD), left-arrow (LT), right-arrow (RT), slow down (SD), forward arrow (ST), and turn-round (UT). The maximum and the minimum average precisions recorded were 99.00% and 69.5% corresponding to turn-round marking and pedestrian marking, respectively. Table 2 shows the mean average precision of the classes and the extent of variation between the classes average precision (standard deviation). The training losses recorded was 0.1042 for 550 iterating epochs. As shown in Figure 4(b), the training curve of the state-of-the-art faster R-CNN did not converge quickly. By setting the confidence score threshold between 0 and 1, different corresponding values of recall and false positive rate are obtained from which the detection performance curve is obtained. From Figure 4(c), the modified faster R-CNN algorithm performs better than the state-of-the-art faster R-CNN algorithm in terms of better detection rate and low false alarm rate. Thus, the proposed method has outperformed the state-of-the-art faster R-CNN algorithm.

The statistical t-test between the two mean average precisions of the algorithms resulted in a p value of 0.030 at a 95% confidence interval. Thus, the p value ≤ 0.05 which implies that the two detection algorithms do not perform equally, and there is a significant difference between the two mean average precisions of the modified faster R-CNN and the faster R-CNN algorithms as shown in Table 2. Essentially, the

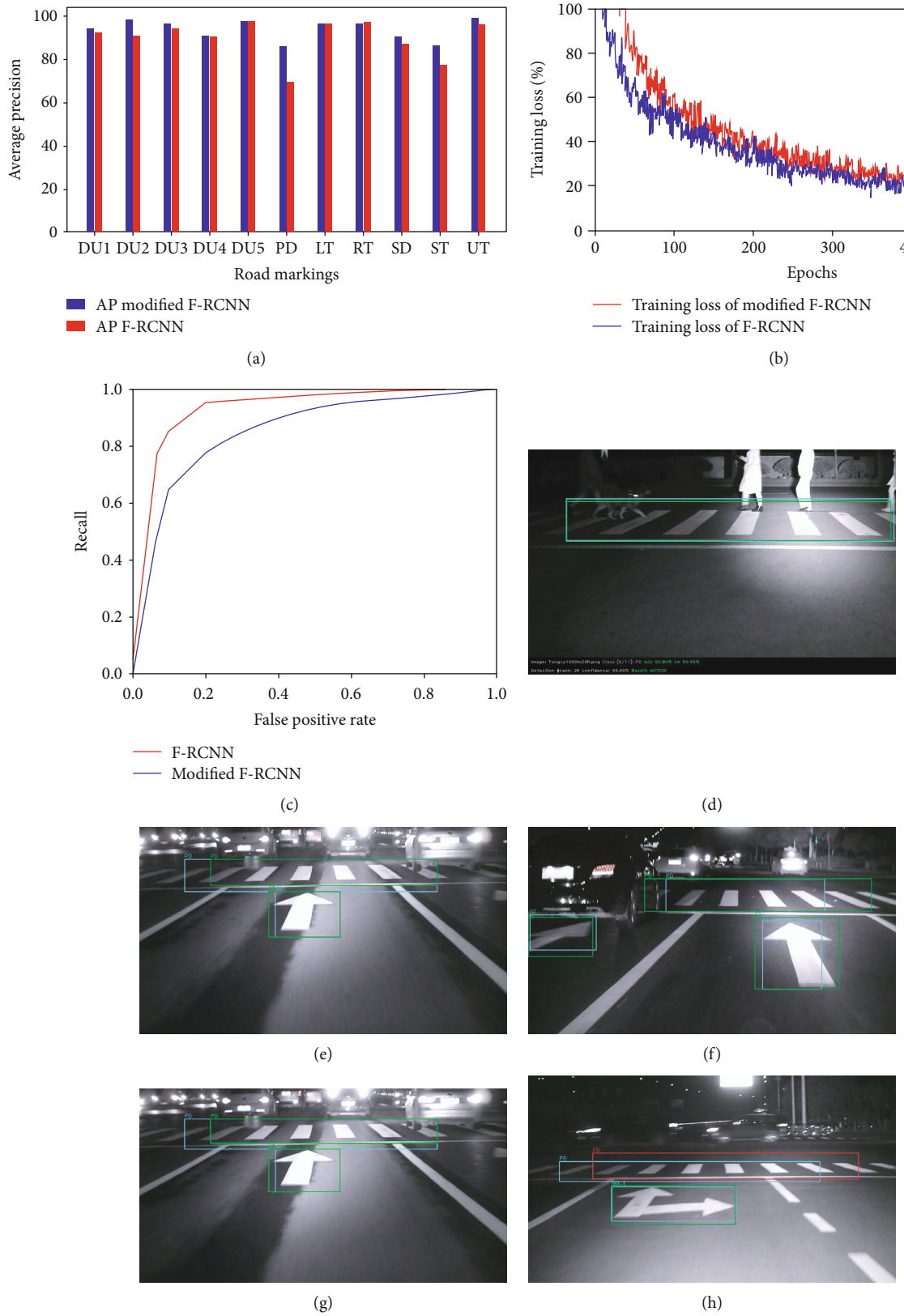

Figure 4: (a) Average precision of faster R-CNN and our modified F-RCNN algorithms. (b) Training loss of F-RCNN algorithm and modified F-RCNN (c) Detection performance curve of modified F-RCNN and F-RCNN algorithm. (d) Example of one-by-one detection result. (e) Example of multiple detection result. (f) Example of multiple detection result (f) Example of multiple detection results (g) Example of multiple detection result (h) Example of a missed detection result.

Table 2: Statistical analysis of the experimental result.

Variable	Modified faster R-CNN	Faster R-CNN
Number of classes	11	11
Maximum average precision	99.00	97.00
Mean average precision	93.48	89.61
Standard deviation	4.71	8.83

modified faster R-CNN also performed better in terms of the detection rate. Figure 4(d) shows an example of the detection result. The ground-truth bounding box is green, the detected mark is sky blue, the IoU for this example was 90.84%, and the detection accuracy was 99.99%. The examples of good detection results with multiple road markings are shown in Figures 4(e)–4(g). However, there were few cases of missed detection were the IoU was not satisfied as seen in Figure 4(h) for the case of pedestrian crossing sign. The mean average precision and the recall for the faster R-CNN and our modified model were 89.43% and 92.83%, respectively. The modification has contributed to an increase in mAP of ~4%. The mean average precision obtained in both cases has outperformed the results reported by [1, 13]. The detection time of a 1280×720 image (even with multiple road markings), using our desktop computer (Corei7-8700; 2.6 GHz processor with 12-core CPU; NVIDIA GeForce GTX 1060 6 GB and 12 GB-RAM) were ~0.34 (3.0 fps) and ~0.30 (3.33 fps), respectively, for faster R-CNN and modified faster R-CNN, respectively. The mAP, the false positive rate, and the detection speed of our model show that our modified algorithm is promising in real-life application.

5. Conclusions

This research demonstrated the application of a NIR camera as a reliable image acquisition system for nighttime driving assistance system. We built our dataset for the experiment from the images captured on the street of Changsha city, and we have also introduced random rotation and translation with bounding boxes to augment our training samples. The proposed method utilizes multilayer feature fusion to take the advantages of the shallow and deep layer features. The proposed algorithm was trained to minimize the loss using stochastic gradient descent algorithm. The trained model was evaluated based on mean average precision, false positive rate, and detection time. We have recorded an improvement of about 4% in mAP. We have accessed statistically the contribution of our proposed method in terms of mAP where we obtained a p value of 0.03% which shows that there is a significant difference between the mAP of our proposed method and the faster R-CNN. Also, our proposed method can detect road markings on 1280×720 pixel at ~0.30 s (3.33 fps). Thus, the model is suitable for application in the driver-assistance system. In the future, we would consider other algorithms such as YOLO and SSD for faster detection.

Data Availability

The dataset used has not been publicly made available yet.

Conflicts of Interest

The authors declare no conflict of interest.

Authors' Contributions

The following are the authors' contributions: Shitu Abubakar and Junping Hu performed the conceptualization; Shitu Abubakar contributed in the methodology; Shitu Abubakar and Junping Hu prepared the software; Xiaobiao Dai and Hao Sha helped in the validation; Shitu Abubakar performed the formal analysis; Shitu Abubakar wrote and prepared the original draft; Shengjun Liu wrote, reviewed, and edited the manuscript; Junping Hu and Shengjun Liu supervised the study; Junping Hu contributed in acquiring funds.

Acknowledgments

This research was funded by the National Natural Science Foundation of China, grant number 61505264, 2016.

Supplementary Materials

The details of the proposed feature fusion layers and the detection pipeline. *(Supplementary Materials)*

References

[1] R. Qian, Q. Liu, Y. Yue, F. Coenen, and B. Zhang, "Road surface traffic sign detection with hybrid region proposal and fast R-CNN," in *12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, pp. 555–559, Changsha, China, August 2016.

[2] A. Brunetti, D. Buongiorno, G. F. Trotta, and V. Bevilacqua, "Computer vision and deep learning techniques for pedestrian detection and tracking: a survey," *Neurocomputing*, vol. 300, pp. 17–33, 2018.

[3] S. Kumar, D. Datta, S. K. Singh, and A. K. Sangaiah, "An intelligent decision computing paradigm for crowd monitoring in the smart city," *Journal of Parallel and Distributed Computing*, vol. 118, pp. 344–358, 2018.

[4] T. Yang, X. Long, A. K. Sangaiah, Z. Zheng, and C. Tong, "Deep detection network for real-life traffic sign in vehicular networks," *Computer Networks*, vol. 136, pp. 95–104, 2018.

[5] R. Qian, B. Zhang, Y. Yue, Z. Wang, and F. Coenen, "Robust Chinese traffic sign detection and recognition with deep convolutional neural network," in *2015 11th International Conference on Natural Computation (ICNC)*, pp. 791–796, Zhangjiajie, China, August 2015.

[6] C. Han, G. Gao, and Y. Zhang, "Real-time small traffic sign detection with revised faster-RCNN," *Multimedia Tools and Applications*, vol. 78, no. 10, pp. 13263–13278, 2019.

[7] https://github.com/jinfagang/keras_frcnn.

[8] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104,

no. 2, pp. 154–171, 2013, https://ivi.fnwi.uva.nl/isis/publications/2013/UijlingsIJCV2013.

[9] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, Columbus, OH, USA, June 2014.

[10] R. Girshick, "Fast R-CNN," in *2015 IEEE International Conference on Computer Vision (ICCV),*, pp. 1440–1448, Santiago, Chile, December 2015.

[11] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[12] W. Liu, D. Anguelov, D. Erhan et al., "SSD: Single Shot MultiBox Detector," in *European Conference on Computer Vision*, pp. 21–37, Amsterdam, The Netherlands, 2016.

[13] Y. Tian, J. Gelernter, X. Wang et al., "Lane marking detection via deep convolutional neural network," *Neurocomputing*, vol. 280, pp. 46–55, 2018.

[14] P. Govardhan and U. C. Pati, "NIR image based pedestrian detection in night vision with cascade classification and validation," *2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies*, 2014, Ramanathapuram, India, May 2014, 2014.

[15] L. Sun, J. Chen, K. Xie, and T. Gu, "Deep and shallow features fusion based on deep convolutional neural network for speech emotion recognition," *International Journal of Speech Technology*, vol. 21, no. 4, pp. 931–940, 2018.

[16] A. K. Sangaiah, D. V. Medhane, T. Han, M. S. Hossain, and G. Muhammad, "Enforcing position-based confidentiality with machine learning paradigm through mobile edge computing in real-time industrial informatics," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 4189–4196, 2019.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *NIPS'12 Proceedings of the 25th International Conference on Neural Information Processing Systems*, pp. 1097–1105, Lake Tahoe, ND, USA, December 2012.

[18] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *13th European Conference on Computer Vision, ECCV 2014*, pp. 818–833, Springer, Zurich, Switzerland, 2014.

[19] M. Mathias, R. Timofte, R. Benenson, and L. Van Gool, "Traffic sign recognition—how far are we from the solution?," in *The 2013 International Joint Conference on Neural Networks (IJCNN),*, Dallas, TX, USA, August 2013.

[20] B. Qin, W. Liu, X. Shen et al., "A general framework for road marking detection and analysis," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013),*, pp. 619–625, The Hague, Netherlands, October 2013.

[21] N. Pinchon, O. Cassignol, A. Nicolas et al., "All-weather vision for automotive safety: which spectral band?," in *Advanced Microsystems for Automotive Applications 2018*, pp. 3–15, Springer, Berlin, Germany, 2018.

[22] T. Y. Han and B. C. Song, "Night vision pedestrian detection based on adaptive preprocessing using near infrared camera," in *2016 IEEE international conference on consumer electronics-Asia (ICCE-Asia)*, Seoul, South Korea, October 2016.

[23] R. Danescu and S. Nedevschi, "Detection and classification of painted road objects for intersection assistance applications," in *13th International IEEE conference on intelligent transportation systems*, pp. 433–438, Funchal, Portugal, September 2010.

[24] P. Foucher, Y. Sebsadji, J.-P. Tarel, P. Charbonnier, and P. Nicolle, "Detection and recognition of urban road markings using images," in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC),*, pp. 1747–1752, Washington, DC, USA, October 2011.

[25] X. Xiang, N. Lv, X. Guo, S. Wang, and A. El Saddik, "Engineering vehicles detection based on modified faster R-CNN for power grid surveillance," *Sensors*, vol. 18, no. 7, pp. 2258–2258, 2018.

[26] 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.

[27] Y. Ren, C. Zhu, and S. Xiao, "Small object detection in optical remote sensing images via modified faster R-CNN," *Applied Sciences*, vol. 8, no. 5, p. 813, 2018.

[28] Y. Xu, M. Zhu, P. Xin, S. Li, M. Qi, and S. Ma, "Rapid airplane detection in remote sensing images based on multilayer feature fusion in fully convolutional neural networks," *Sensors*, vol. 18, no. 7, p. 2335, 2018.

[29] Y. Lu, J. Lu, S. Zhang, and P. Hall, "Traffic signal detection and classification in street views using an attention model," *Computational Visual Media*, vol. 4, no. 3, pp. 253–266, 2018.

[30] H. Vokhidov, H. Hong, J. Kang, T. Hoang, and K. Park, "Recognition of damaged arrow-road markings by visible light camera sensor based on convolutional neural network," *Sensors*, vol. 16, no. 12, p. 2160, 2016.

[31] J. Khan, D. Yeo, and H. Shin, "New dark area sensitive tone mapping for deep learning based traffic sign recognition," *Sensors*, vol. 18, no. 11, p. 3776, 2018.

[32] C. Liu, S. Li, F. Chang, and W. Dong, "Supplemental boosting and cascaded ConvNet based transfer learning structure for fast traffic sign detection in unknown application scenes," *Sensors*, vol. 18, no. 7, p. 2386, 2018.

[33] G. Villalón-Sepúlveda, M. Torres-Torriti, and M. Flores-Calero, "Traffic sign detection system for locating road intersections and roundabouts: the Chilean case," *Sensors*, vol. 17, no. 6, p. 1207, 2017.

[34] E. Peng, F. Chen, and X. Song, "Traffic sign detection with convolutional neural networks," in *Communications in Computer and Information Science*, pp. 214–224, Springer nature, 2017.

[35] T. Ahmad, D. Ilstrup, E. Emami, and G. Bebis, "Symbolic road marking recognition using convolutional neural networks," in *2017 IEEE intelligent vehicles symposium (IV),*, Los Angeles, CA, USA, June 2017.

[36] N. Lv, C. Chen, T. Qiu, and A. K. Sangaiah, "Deep learning and superpixel feature extraction based on contractive autoencoder for change setection in SAR images," *IEEE transactions on industrial informatics*, vol. 14, no. 12, pp. 5530–5538, 2018.

[37] X. Dai, Y. Duan, J. Hu et al., "Near infrared nighttime road pedestrians recognition based on convolutional neural network," *Infrared Physics & Technology*, vol. 97, pp. 25–32, 2019.

[38] A. Ramchandran and A. K. Sangaiah, "Unsupervised deep learning system for local anomaly event detection in crowded scenes," *Multimedia Tools and Applications*, pp. 1–21, 2019.

[39] C. Wen, X. Sun, J. Li, C. Wang, Y. Guo, and A. Habib, "A deep learning framework for road marking extraction, classification

and completion from mobile laser scanning point clouds," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 147, pp. 178–192, 2019.

[40] N. Kryvinska, A. Poniszewska-Maranda, and M. Gregus, "An approach towards service system building for road traffic signs detection and recognition," *Procedia Computer Science*, vol. 141, pp. 64–71, 2018.

[41] Y. Zhu, C. Zhang, D. Zhou, X. Wang, X. Bai, and W. Liu, "Traffic sign detection and recognition using fully convolutional network guided proposals," *Neurocomputing*, vol. 214, pp. 758–766, 2016.

[42] T. Chen, Z. Chen, Q. Shi, X. Huang, and R. M. Detection, "Classification using machine learning algorithms," in *IEEE Intelligent Vehicles Symposium (IV)*, COEX, Seoul, Korea, June 2015.

[43] W. Ma, Q. Guo, Y. Wu, W. Zhao, X. Zhang, and L. Jiao, "A novel multi-model decision fusion network for object detection in remote sensing images," *Remote Sensing*, vol. 11, no. 7, p. 737, 2019.

[44] H. H. Aghdam, E. J. Heravi, and D. Puig, "A practical approach for detection and classification of traffic signs using convolutional neural networks," *Robotics and Autonomous Systems*, vol. 84, pp. 97–112, 2016.

[45] Y. Yu, J. Li, H. Guan, F. Jia, and C. Wang, "Learning hierarchical features for automated extraction of road markings from 3-D mobile LiDAR point clouds," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 2, pp. 709–726, 2015.

[46] H. Guan, J. Li, Y. Yu, Z. Ji, and C. Wang, "Using mobile LiDAR data for rapidly updating road markings," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2457–2466, 2015.

[47] Z. Yan, H. Zhang, Y. Jia, T. Breuel, and Y. Yu, "Combining the best of convolutional layers and recurrent layers: a hybrid network for semantic segmentation," 2016, http://arxiv.org/abs/1603.0487v1.

[48] Y. Xie, W. Dai, Z. Hu, Y. Liu, C. Li, and X. Pu, "A novel convolutional neural network architecture for SAR target recognition," *Journal of Sensors*, vol. 2019, Article ID 1246548, 9 pages, 2019.

[49] M. Cheng, H. Zhang, C. Wang, and J. Li, "Extraction and classification of road markings using mobile laser scanning point clouds," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 3, pp. 1182–1196, 2017.

[50] X. Yang, H. Sun, K. Fu et al., "Automatic ship detection in remote sensing images from Google Earth of complex scenes based on multiscale rotation dense feature pyramid networks," *Remote Sensing*, vol. 10, no. 1, p. 132, 2018.