IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received November 27, 2020, accepted December 15, 2020, date of publication December 22, 2020,
date of current version January 5, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3046498

Feature Enhancement Based on CycleGAN for Nighttime Vehicle Detection

XIAOTAO SHAO, (Member, IEEE), CAIKE WEI, YAN SHEN, (Member, IEEE), AND ZHONGLI WANG, (Member, IEEE)
School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China

Corresponding author: Yan Shen (sheny@bjtu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61702032, Grant 61573057, and Grant 61771042; in part by the Fund under Grant 61404130316 and Grant 61400010302; and in part by the Fundamental Research Funds for the Central Universities under Grant 2017JBZ002.

ABSTRACT Existing night vehicle detection methods mainly detect vehicles by detecting headlights or taillights. However, these features are adversely affected by the complex road lighting environment. In this paper, a cascade detection network framework FteGanOd is proposed with a feature translate-enhancement (FTE) module and the object detection (OD) module. First, the FTE module is built based on CycleGAN and multi-scale feature fusion is proposed to enhance the detection of vehicle features at night. The features of night and day are combined by fusing different convolutional layers to produce enhanced feature (EF) maps. Second, the OD module, based on the existing object detection network, is improved by cascading with the FTE module to detect vehicles on the EF maps. The proposed FteGanOd method recognizes vehicles at night with greater accuracy by improving the contrast between vehicles and the background and by suppressing interference from ambient light. The proposed FteGanOd is validated on the Berkeley Deep Drive (BDD) dataset and our private dataset. The experimental results show that our proposed method can effectively enhance vehicle features and improve the accuracy of vehicle detection at night.

INDEX TERMS Convolutional neural network (CNN), nighttime vehicle detection, feature enhancement, generative adversarial network (GAN), detection network.

I. INTRODUCTION

Vehicle detection is an important application in the field of target detection. More accurate vehicle detection systems for day and night conditions will facilitate the development of more reliable Automatic Driving System (ADS) and Driver Assistance System (DAS) in the future. In night (lowlight) condition, the probability of traffic accidents is increased [1] because less visual information about vehicles and the complex lighting environment is available. (a) *Less visual information about vehicles*. The contrast between the vehicle and the background is reduced at night, making vehicle features less obvious. (b) *Complex lighting environment*. Interference from various other lights is confused with vehicle headlights and taillights, which leads to a high rate of false vehicle detection and presents significant challenges for vision-based vehicle detection at night.

Most existing vehicle detection methods use the headlights and taillights as the primary nighttime vehicle detection features. Traditional detection methods that are not based on convolutional neural networks (CNNs) used the headlights and taillight to locate vehicles [4]–[11]. Taillights were localized by segmenting the image, and vehicle bounding boxes were predicted by assuming the typical width of the vehicle [4], [5]. Region proposals were firstly obtained by paired taillights of vehicles, next the vehicles whether in these region proposals were determined [6], [7]. In [10], a detection-by-tracking method was proposed to detect multiple vehicles by tracking headlights/taillights. These traditional non-CNN vehicle detection methods have two disadvantages. (1) Vehicle detection is susceptible errors due to the complex lighting conditions in urban areas, including vehicle lights, streetlights, building lights, and the reflected lights from vehicles, which increases the false positive rate. (2) Vehicle lights are sometimes obscured when the vehicle is occluded or only the side of the vehicle is photographed, which increases the missed detection rate.

Vehicle detection methods based on CNN have gradually become the research focus in recent years and some CNN-based nighttime vehicle detection methods have

The associate editor coordinating the review of this manuscript and approving it for publication was Jinming Wen.

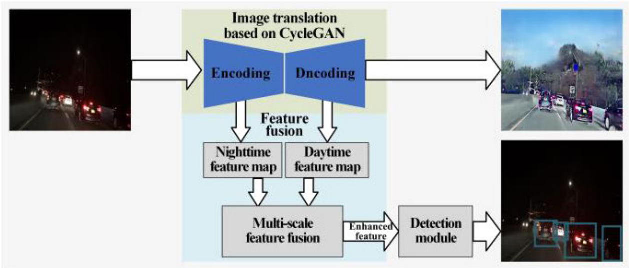


FIGURE 1. Vehicle detection and behavior analysis of proposed FteGanOd during nighttime.

been investigated. Lin *et al*. [23] proposed AugGAN to translate daylight images into night images for data augmentation, and the images were then used to train existing detection systems, improving the performance of the detector. However, only augments data processing systems in existing night vehicle detection methods. Kuang *et al*. [1] used a bioinspired enhancement approach to enhance night images for feature fusion and object classification in 2017. In 2019, they combined traditional features and CNN features to generate regions of interest (ROI) [2], [3]. The above methods, combining traditional machine learning methods and deep learning methods for object detecting, belong to multi-stage learning frameworks. However, they are not an end-to-end learning framework which makes the training process cumbersome.

Some object detection methods based on deep learning (Fast RCNN [29], SSD [33], etc.) can also be used for night vehicle detection. However, these methods are designed for daytime object detection and their use in nighttime conditions results in a low level of feature extraction accuracy from the network structure and a low rate of vehicle detection performance.

In a word, low-light environment, complex lighting, and the specialized structure of nighttime detection network are three challenges in nighttime vehicle detection. A low-light environment increases the rate of missed vehicles because of the faint features of the vehicles; complex lighting leads to a higher false detection rate when dealing with more complex traffic scenes; and the specialized nighttime detection network is still incomplete. However, generative adversarial network (GAN) is a style transfer network that can translate nighttime images into daytime images. GAN uses encode modules to extract features from nighttime images and uses decode modules to restore the daytime images.

Therefore, we propose a novel nighttime vehicle detection framework named FteGanOd (feature translate-enhancement generative adversarial network for object detection) to overcome the above challenges. FteGanOd includes a feature translate-enhancement (FTE) module and an object detection (OD) module, as shown in Fig. 1. (1) FTE firstly uses CycleGAN to translate images from night to day. The multi-scale features from CycleGAN are next used to fuse the encoded (nighttime) features and decoded (daytime) features to form the enhanced feature (EF) maps. Encoded features contain important information of night vehicle headlights and taillights; decoded features contain daytime features for enhancing the background brightness while suppressing most of the light sources. (2) The OD module (improved YOLO, RCNN or SSD) extracts abstract vehicle features and detects vehicles on the EF maps.

The remainder of this paper is given as follows: In Section II, the methods of vehicle detection at night, detection networks based on CNN and GAN are introduced. Section III describes our night detection network FteGanOd in detail. Section IV introduces the experimental processes and discusses the experimental results. Finally, conclusions and possibilities for future work are presented in Section V.

II. RELATED WORK

A. NIGHTTIME VEHICLE DETECTION

The headlights/taillights are used as key information in locating vehicles for almost all nighttime vehicle detection algorithms. Searching for red or highlight lights in night images is the main technique to obtain region proposals in previous methods, which has been proven effective in most literature.

For obtaining ROIs of vehicles, the following techniques can be adopted: threshold-based segmentation methods [5], [12], [18], paired vehicle lighting-based methods [6]–[8], [14]–[16], saliency map-based methods [17], [27], and artificially designed feature-based methods [13]. After the ROIs are obtained, we need to further determine whether these candidate regions contain vehicles. X. Dai [18] used Hough transform to detect the circles of the headlights and further segment the areas to locate vehicles. Pradeep *et al.* [14] used red thresholding to obtain ROIs and searched for paired taillights based on the shape similarity and region size to detect vehicles. Chen *et al.* [27] generated ROIs based on the saliency method and applied the deformable parts model (DPM) to detect vehicles. Kosaka *et al.* [13] used SVM to classify vehicles after using Laplacian of Gaussian operation to detect the blobs.

In recent years, CNN-based methods are increasingly developed in the research field of vehicle detection at night. [20]–[23] used GAN-based data augmentation methods to expand the training dataset for improving the performance of the detector. Cai *et al.* [19] combined visual saliency and prior information to generate ROIs and used CNN as a classifier. Kuang *et al*. [1] proposed a bioinspired method to enhance image contrast and brightness, and further extracted the fusion features of LBP, HOG and CNN. They also [3] proposed a feature extraction method based on tensor decomposition and feature ranking. In [2], Nakagami-image-based method and CNN feature were combined to generate ROIs. References [1]–[3] extracted different features to generate region proposals by combining traditional methods with CNN methods. Mo *et al*. [24] managed to solve the problem of confusion between other lights and vehicle lights by training a CNN-based highlight detector.

Existing nighttime vehicle detection methods with visual images mainly use vehicle lights for detection. Especially, the CNN-based methods have stronger adaptability and robustness than non-CNN methods.

B. OBJECT DETECTION BASED ON DEEP LEARNING

The object detection methods based on deep learning can be grouped into two detection methods: two-stage detection and one-stage detection.

1) TWO-STAGE DETECTION

RCNN is the first two-stage object detection network [28]. First, a selective search algorithm generates a series of region proposals; next, the proposals are input into CNN to extract features; finally, SVM is used to predict whether each region proposal contains an object. A series of improved networks based on RCNN were proposed, such as Fast RCNN [29], Faster RCNN [30], SPP-Net [31], etc. These networks use different methods to remove redundant parts in the detection network to improve detection speed and accuracy.

2) ONE-STAGE DETECTION

One-stage detectors, represented by SSD [33] and YOLO [34]–[36], have broken through the detection speed bottleneck of two-stage detectors. However, detection accuracy is reduced compared with two-stage methods, especially for small objects. YOLOv3 and SSD use multi-scale detection to improve small target detection performance.

These detection networks have better performance in the ideal (day) environment. However, the accuracy is lower when applied to night vehicle detection. We propose adding the FTE module before the CNN-based detection framework to detect vehicles by combining features of night and day.

C. GAN: GENERATIVE ADVERSARIAL NETWORKS

GAN, with a generator and a discriminator, has achieved sound results in some applications such as super-resolution [37] and de-raining [38]. Most of these applications involve with image generation by GAN to augment training data, which can be used to generate a ''false'' image similar to real image samples.

GAN-based image-to-image translation refers to the task of converting images from one scene to another different scene. Pix2pix [39], [40] uses paired images as supervision to achieve image-to-image translation. However, it is difficult to obtain pairs of both day and night image samples in fixed places because the vehicle move too much to capture traffic images. To solve the problem of inputting day and night paired images, CycleGAN [26] is firstly proposed with cycle consistency loss to realize image translation with unpaired images. Image translation realizes the conversion of the object features in an image. For example, after the image is translated from night to day, the background brightness of the image becomes brighter and the vehicles are easier to be recognized.

CycleGAN only needs images of different domains during training rather than expensive ground-truth paired image data, which is of great significance for practical applications. Therefore, we will use the structure of CycleGAN for feature translation at night.

III. PROPOSED METHOD

In this section, we elaborate on the proposed method for vehicle detection at night. To resolve the effect of weak environmental light or complex vehicle's light at night, we propose a feature cascade network structure FteGanOd to enhance vehicle features and improve vehicle detection accuracy. FteGanOd consists of two modules: feature translate-enhancement (FTE) module and object detection (OD) module. The FTE module adopts CycleGAN as the basic network to realize the translation of scenes from night to day by learning on unpaired input data. The OD module cascades with the FTE module to extract fused and enhanced daytime features and detect vehicles with higher detection accuracy.

A. PROPOSED FTEGANOD NETWORK

The ability to train with unpaired images as input is a typical advantage of CycleGAN for adapting different traffic scenes. The disadvantage of CycleGAN is that it cannot learn the category and position of the vehicle object when mapping the nighttime image to daytime image. Therefore, we conducted an in-depth study on this and realize the translation of local vehicle features and global scene features from night to day. We propose an FTE module based on CycleGAN and the feature cascaded OD module to overcome the disadvantage of CycleGAN and improve the detection accuracy. Unlike other training methods, here we train the FTE and OD modules together to guide FTE optimization in a direction that makes the vehicle features more prominent.

Fig. 2 shows the feature cascade structure of the proposed FteGanOd network, which is composed of four parts (A), (B), (C) and (D). (A) + (B) is the FTE module, where (A) is the generator based on CycleGAN that translate the image from night to day, (B) is the enhancement part that fuses the night and day multi-scale features map from (A) to enhance the vehicle features and finally generate the enhanced feature EF map as input for Part (C). Part (C) is the OD module based on the existing detector structure (such as YOLOv3, etc.). Part (D) is for generating daytime images *x_fake* in training process.

B. FTE MODULE

1) IMAGE TRANSLATION

The proposed FTE module is based on the CycleGAN network as shown in Fig. 2. Suppose the feature map f_i at layer i with dimension $h_i \times w_i \times c_i$, $i \in \mathbb{N}$. Let E^i be a generic function which acts as the basic generator block, which in our implementation consists of a convolution layer followed by an instance normalization and an activation function. E^i_{encode} is the operation of encoding the input image as multi-scale

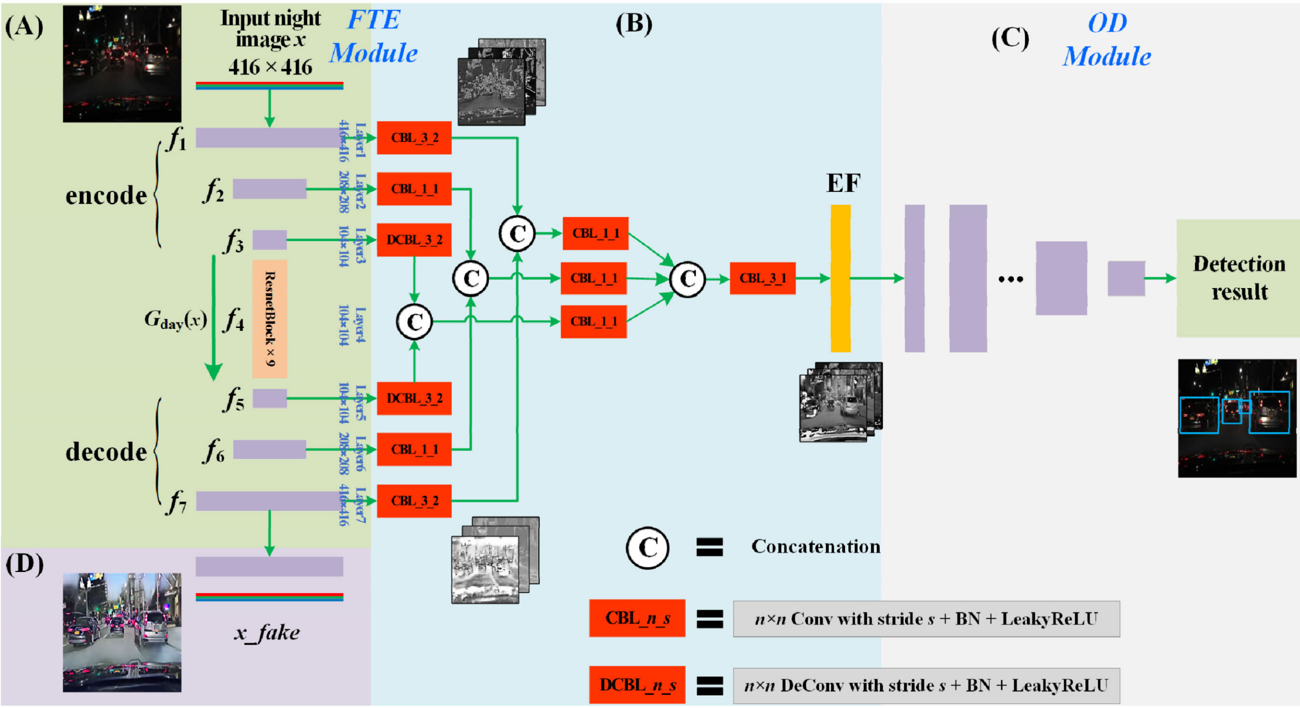


FIGURE 2. Architecture of the proposed FteGanOd: Input night image x, (A) + (B) is the FTE module to translate and enhance the features of vehicles in night images. In (A), G_{day} process maps images from night domain to day domain. Features f_1 - f_3 are the encoded feature maps to extracted night image features from x; features f_5 - f_7 are the decoded feature maps that translate the nighttime features into daytime features. In (B), multi-scale feature are fused to obtain enhanced features (EF). Finally, EF is fed into the OD module (C). In (D), the convolutional layers are used to generate the fake daytime image *x_fake*.

feature maps.

$$\begin{aligned} E^i_{encode} &= f_{i-1} \mapsto f_i \\ \text{where,} \quad f_i &= \mathbb{R}^{w_i \times h_i \times 2^{b+1+i}} \\ \text{and,} \quad w_i &= h_i = N/2^{i-1};\ i = 1, 2, 3 \end{aligned} \tag{1}$$

where f_0 is the input night image of the network with a size of $N \times N$, let $N = 416$. The dimension of the feature map $\mathbb{R}^{w \times h \times c}$ contains $[h \times w \times c]$ dimensional activations, c is the number of channels in the intermediate activations of the generator, $c = 2^{b+1+i}$, here we let $b = 4$. E^i_{decode} is the operation of decoding the translated daytime features to the final output restored image.

$$\begin{aligned} E^{b+i}_{decode} &= f_{b+i} \mapsto f_{b+i+1} \\ \text{where,} \quad f_{b+i} &= \mathbb{R}^{w_i \times h_i \times 2^{2b-i+1}} \\ \text{and,} \quad w_i &= h_i = N/2^{b-i-1};\ i = 1, 2, 3 \end{aligned} \tag{2}$$

The final output fake daytime image *x_fake* is f_8.

After the proposed FteGanOd network was trained, the encoded and decoded feature maps were obtained and some of them are shown in Fig. 3. The input nighttime image x is gradually encoded by several convolution layers to form the feature map f_1 to f_3. The vehicle features and contours in the feature map f_1 of the first convolution layer are not obvious due to the low-light background at night, as marked in the red boxes. Compared with f_1, the noise of feature map f_2 has been aggravated, but we can still see the basic shapes of the vehicles through the position of the road. After further

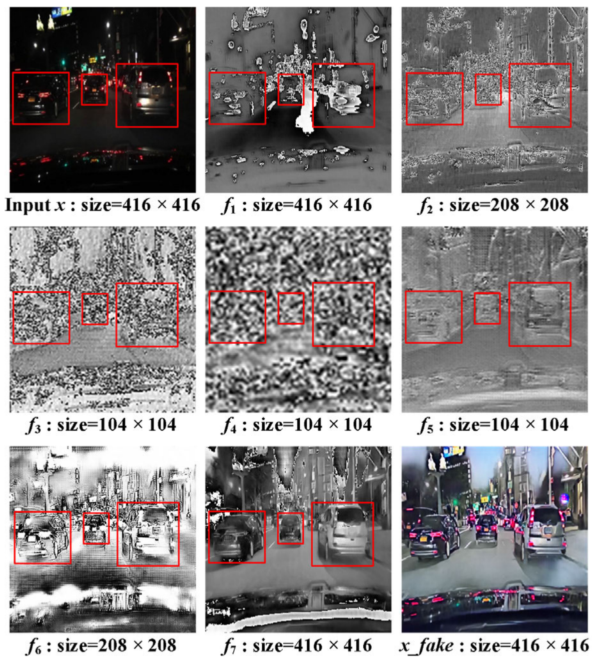


FIGURE 3. Features f_1 - f_7 are the feature maps from different convolutional layers of the G_{day}, respectively. The input night image x goes through the G_{day} to obtain the final translation daytime image *x_fake*. 'size' represents the actual size of the feature map. The red boxes mark the location of the vehicles.

encoding f_2 to form the high-level semantic feature map f_3, which represents the feature vector of the vehicles and the background in the night domain, we can still see the faint features of the vehicle in the little differences from image contrast. Subsequently, a converter composed of 9 Resnet blocks maps the night domain feature vector f_3 into the day domain feature vector f_5 through f_4. Resnet blocks complete the mapping of nighttime features to daytime features by keeping the similar features in both domains and by converting different features.

Next, the decoder translates the feature f_5 of the daytime domain into low-level feature f_6. It can be seen from f_5 that it already has a clearer vehicle shape and some details. In f_6, the contrast between the vehicle and the background is further improved and the daytime style is salient. Compared with f_1, the noises on vehicles and roads in feature map f_7 are significantly reduced and the vehicle details are greatly exhibited, which helps the detector to recognize the vehicle, even for distant vehicles. Finally, the daytime image *x_fake* is generated with clearer shapes and details than the input image x. Although some small areas have an unnatural appearance, the vehicles have been clearly distinguished from the background.

2) MULTI-SCALE FEATURE FUSION AND ENHANCEMENT

In Fig. 3, if the daytime images *x_fake* is used directly to detect vehicles, the relevant night information (vehicle lights and reflections) of the image x would be lost to some extent. Therefore, nighttime features should be adopted to enhance the vehicle features. As shown in part (B) of Fig. 2, we propose multi-scale feature fusion to fuse features of f_i (nighttime) and f_{2b-i} (daytime) to enhance vehicle features and further improve the detection accuracy of vehicles.

Fusion of multi-scale features may cause aliasing effects [32] and high feature dimension. To solve this problem, before we fuse the different scales of nighttime features and daytime features, every feature map is first passed through a convolutional kernel to reduce the aliasing effect, reduce the dimensions of the features, and resize the features of different scales into the same scale. Choosing an EF scale of half the size of the input image (208 × 208) can reduce the network complexity. If the EF scale size is 416 × 416, the calculations in part (B) of Fig. 2 will increase exponentially, and even will double the computational complexity of the detection network.

Denote F_{conv}, F_{deconv}, and F_{concat} as the convolution, deconvolution, and concatenate operation separately. The feature fusion unit follows the operations, as described in (3)–(7). F_{conv} means the feature map through a $k^{CBL\_n\_s}$ convolution kernel with size $n \times n$ and stride s, as shown in Fig. 2.

$$\begin{aligned} g_i &= F_{conv}(f_i, k_i^{CBL\_n\_s}), \\ g_{2b-i} &= F_{conv}(f_{2b-i}, k_{2b-i}^{CBL\_n\_s}) \end{aligned} \qquad i = 1, 2 \tag{3}$$

$$\begin{aligned} g_i &= F_{deconv}(f_i, k_i^{DCBL\_n\_s}), \\ g_{2b-i} &= F_{deconv}(f_{2b-i}, k_{2b-i}^{DCBL\_n\_s}) \end{aligned} \qquad i = 3 \tag{4}$$

The feature f_i is first adjusted through a convolution layer with learnable weights for scale matching and dimensionality

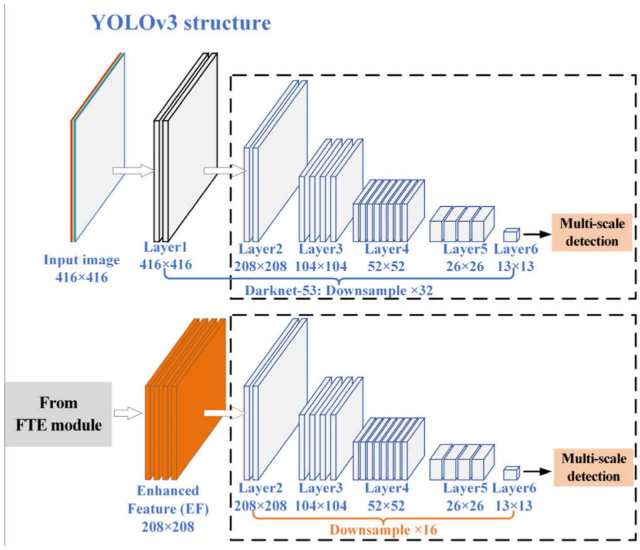


FIGURE 4. Detection module OD based on YOLOv3 structure.

reduction. This step is designed to adjust the size $h_i \times w_i$ to $N/2 \times N/2$ and reduce the number of channels by half to $c_i/2$.

Next these features are concatenated to form f_{concat_i}.

$$f_{concat\_i} = F_{concat}(g_i, g_{2b-i}), \quad i = 1, 2, 3 \tag{5}$$

The concatenated features f_{concat_i} of Eq. (5) are further fused and their dimensions are reduced through a 1 × 1 convolution kernel respectively and are concatenated again.

$$\begin{aligned} v_i &= F_{conv}(f_{concat\_i}, k^{CBL\_1\_1}) \\ f_{concat\_EF} &= F_{concat}(v_i) \end{aligned} \qquad i = 1, 2, 3 \tag{6}$$

Finally, the enhanced fusion features f_{EF} is obtained by Eq. (7).

$$f_{EF} = F_{conv}(f_{concat\_EF}, k^{CBL\_3\_1}) \tag{7}$$

The feature f_{EF} will be fed into the OD module for vehicle detection.

C. OD MODULE

After nighttime features and daytime features are fused in Section B, here the detection module OD is proposed to extract vehicle features and detect them on the enhanced fusion features EF.

The traditional detection methods use the daytime RGB image *x_fake* as the input for the detection module, which is a two-period network, and further introduces more parameters and computation complexity. In order to solve these problems, we propose a detection module OD, which is cascaded with the FTE module through EF to form a one-period network. The enhanced fusion features from FTE are directly used as the input of OD instead of RGB images to realize the end-to-end training and detection of the whole network and to reduce the network layers and parameters.

It can be seen from Fig. 2 that part (C) is the OD module, which can be selected as YOLOv3, SSD,

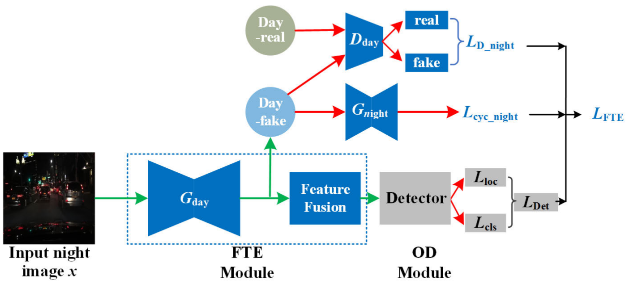


FIGURE 5. Overview of training framework: The green route in the figure corresponds to that in Fig. 2, and the rest is the structure used only for training.

Faster-RCNN, or other networks. Here we take the YOLOv3 framework as an example to illustrate our detection module OD. This example can also be applied to other detectors. As shown in Fig. 4, the upper part is the original structure of YOLOv3 using a 416 × 416 RGB image as the input. YOLOv3 adopts Darknet-53 for feature extraction and has the minimum feature scale of 13 × 13, which is 1/32 of the input size. Since the EF size is half (208 × 208) of the original input of YOLOv3, the scale of the feature map after ×32 down-sampling will be reduced to 7 × 7 if the traditional structure of YOLOv3 is used. Small-scale objects will not be detected when the feature map's resolution is reduced by half. In order to maintain the resolution of the feature scale, we take the network structure of Layer2 - Layer7 of YOLOv3 as our OD module to avoid the lower detection of small targets, as shown in the bottom part of Fig. 4, which removes the largest scale convolution layers in Darknet-53, reduces one down-sampling operation, and replaces the RGB image with EF module.

In order to verify the effectiveness of the proposed OD module, we use three detection networks (YOLOv3, SSD and Faster RCNN) including both one-stage and two-stage methods as detection modules.

D. LOSS AND TRAINING

In this section, we detail the loss of the FteGanOd network and the joint training of the FTE module and OD module.

Fig. 5 shows the training framework of our FteGanOd, the green route corresponds to the green route in Fig. 2, the red route parts of Fig. 5 are the structures used for training. Referring to the CycleGAN network [26], FTE contains two generators (G_{day} and G_{night}) and two discriminators (D_{day} and D_{night}). G_{day} is the generator that translates images from day to night. On the contrary, G_{night} is the generator that translates images from night to day. The Day-fake (shown as the blue circle in Fig. 5) output from G_{day} is the fake daytime image data, and Day-real is the practical real daytime image data. D_{night} and D_{day} are used to judge whether an image is real or fake in the nighttime and daytime image data, respectively. Here only the part of training of G_{day} is described.

The loss of the OD module (L_{Det}) includes bounding box regression loss (L_{loc}) and classification loss (L_{cls}). The loss of FTE module consists of three terms: the cycle consistent loss ($L_{\text{cyc\_night}}$), the discrimination loss ($L_{\text{D\_night}}$) and the detection loss (L_{Det}).

For image x of night domain, the image translation cycle should be able to bring x back to the original image, i.e., $x \rightarrow G_{\text{day}}(x) \rightarrow G_{\text{night}}(G_{\text{day}}(x)) \approx x$. This is called forward cycle consistency, the cycle consistency loss defined as:

$$\begin{aligned} L_{\text{cyc\_night}} &= E_{x\sim P_{\text{data\_night}}(x)}[\left\| G_{\text{night}}(G_{\text{day}}(x)) - x \right\|_1] \\ L_{\text{D\_night}} &= E_{x\sim P_{\text{data\_night}}(x)}[\log(1 - D_{\text{day}}(G_{\text{day}}(x)))] \\ &\quad + E_{y\sim P_{\text{data\_day}}(y)}[\log D_{\text{day}}(y)] \end{aligned} \tag{8}$$

Loss L_{Det} is added to the loss function of the FTE module, which is an important part of optimization allowing it to focus on enhancing the fused features of the target region and the vehicle. The loss L_{cls} in L_{Det} is for the recognition of target features. It has more effect on feature fusion enhancement than L_{loc}, so different weights are applied to L_{cls} and L_{loc}. The loss of the FTE module is:

$$L_{FTE} = L_{\text{cyc\_night}} + L_{\text{D\_night}} + \lambda_1 L_{\text{cls}} + \lambda_2 L_{\text{loc}} \tag{9}$$

Here, $\lambda_1 = 1.4$, $\lambda_2 = 0.7$ (empirically defined). The optimization of FTE and OD are carried out parallel.

IV. EXPERIMENTAL METHODOLOGY

The experiments on the FteGanOd network are described conducted in this section. We first introduce the datasets used for training and testing, following with the evaluation method, the experimental process and the training method of our proposed FteGanOd network. Finally, the results of our experiments are analyzed and discussed.

A. DATASET

Public dataset and private dataset are adopted to evaluate the proposed FteGanOd network. The public dataset, Berkeley Deep Drive (BDD) dataset[1] [25], has 100,000 real driving scene images, including training set (70,000), test set (20,000), and validation dataset (10,000). It is currently the large-scale, diverse and complex driving dataset with annotations. The labels of the images include 10 categories (bus, truck, car, person, train, etc.), different weather (sunny, rainy, etc.), multiple types of scenes (roads, city streets, etc.), time (dawn/dusk, day, night).

We selected all the night images from the BDD based on the time labels and further modified the label category. We defined vehicle category as dataset containing only bus, truck and car. Our training dataset (25,338 images) were selected from the BDD training set with more than 260,000 vehicle targets; our validation dataset (3526 night images) was selected from BDD validation dataset with more than 37,000 vehicle targets. Our training and validation dataset, referred as DATASET1 (hard), contain various

[1]https://bair.berkeley.edu/blog/2018/05/30/bdd/

TABLE 1. Explanation of Datasets.

Dataset	Frames	Remarks
DATASET1 (hard)	training dataset 25,338 images	Select all the night images from BDD and select three types of vehicles: bus, truck, and car for detection. It is a high-density traffic dataset in urban scenes.
	testing dataset 3526 images	
DATASET1 (easy)	training dataset 2000 images	Select 3000 images from DATASET1 (hard) and remove small objects and difficult-to-detect vehicles.
	testing dataset 1000 images	
DATASET2	training dataset 16,700 images	Picked from the video taken by our onboard optical cameras, the traffic density is moderate.
	testing dataset 4177 images	

weather, scenes, occluded and truncated targets, i.e., many difficult-to-recognize targets. DATASET1 (hard) is also a high-density dataset with an average of about 10 vehicles per image. Therefore, we carefully selected 3000 images (2000 for training and 1000 for testing) from the DATASET1 (hard). A vehicle with the bounding box less than 25 pixels and the vehicle that were difficult to distinguish were removed after the image was resized to 416 $\times$ 416 to form a low-density (average 5 vehicles per image) dataset, called DATASET1(easy).

Our private dataset was obtained by using the monocular camera on our intelligent driving car in urban areas from Changchun and Shenzhen cities in China. We selected 20,877 frames from the private dataset as DATASET2 dataset, 80% of which (16,700 frames) were used for training and 20% (4177 frames) for verification. The category of the vehicle was defined as the same way as DATASET1. The original image resolution of DATASET1 and DATASET2 is 1280 $\times$ 720 pixels, which was resized to 416 $\times$ 416 or 300 $\times$ 300 for our experiments. Fig. 6 shows some samples of the datasets of DATASET1 and DATASET2. It can be seen that the multi-lane, high-density traffic that increase the environmental complexity and detection difficulty. TABLE 1 is the detail explanation about the datasets.

B. PERFORMANCE EVALUATION

Based on the OD module, we used three networks frames YOLOv3, SSD and Faster RCNN as the detection model to compare with the state-of-the-art object detection methods. These networks were verified on DATASET1 and DATASET2.

1) EVALUATION METHOD

In order to evaluate the experimental results, we use the following commonly used indicators for vehicle detection. (1) Precision-Recall (PR) curve is used to describe the relationship between precision and recall. False detection rate (FDR) can be computed as FDR = 1 - precision. Recall is also called true positive rate (TPR), and the miss rate can be presented as Miss Rate = 1 - recall. (2) The established

FIGURE 6. Samples of our datasets with ground truth bounding boxes: (a) is from DATASET1 (hard). In a complex lighting environment, small targets and blocked vehicles in the distance are difficult to identify (red bounding boxes); (b) is from DATASET1 (easy); (c) is from DATASET2.

mean average precision (mAP) is widely used to evaluate the performance of object detection algorithms. Average precision (AP) is proportional to the area under the PR curve. Since there is only one category in this experiment, AP is equivalent to mAP. (3) We employ the frames per second (FPS) to quantitatively evaluate the detection speed of different networks. Moreover, the intersection over union (IOU) higher than 0.5 of the prediction bounding box and the ground truth box is considered to be assigned a positive label.

2) TRAIN CYCLEGAN

CycleGAN is the basic network of the FTE module, the source code of CycleGAN is publicly available.[2] The trained CycleGAN model was used as the pre-training model for the FTE, which helped to improve the stability and convergence speed during training FTE. The training parameters use the default settings. The training was finished when the average loss per 10 batches stabilized.

3) TRAIN DETECTORS

We adopt YOLOv3, SSD300 and Faster RCNN networks as the detectors.[3] The input image resolution of YOLOv3 and Faster RCNN is 416 $\times$ 416, while SSD300 is 300 $\times$ 300. The best model was saved and used as a pre-trained model for our detection module OD.

[2]https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix

[3]In order to realize the cascading of enhanced module and detection networks, and to compare the performance between different networks, the codes used are all based on the pytorch framework.

YOLOv3: https://github.com/eriklindernoren/PyTorch-YOLOv3
SSD: https://github.com/amdegroot/ssd.pytorch
Faster RCNN: https://github.com/jwyang/faster-rcnn.pytorch

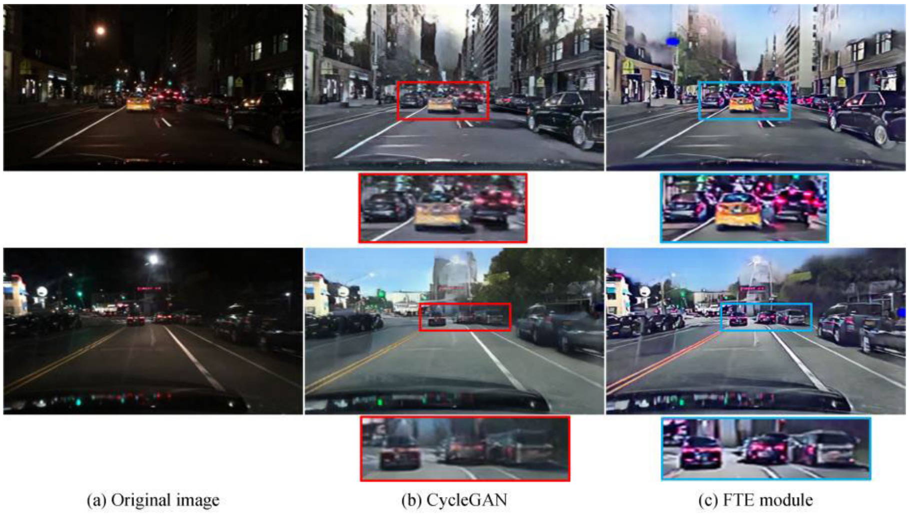


FIGURE 7. Image translate, fusion and enhancement results: (a) Original image. (b) Translate result by CycleGAN. (c) Translate-enhanced image generated by FTE.

4) TRAINING OUR NETWORK

We used three object detection networks as the structure of the OD module, and cascaded those with the FTE module to get three different nighttime detection networks. The detailed training steps are as follows.

The training process and loss are shown in Fig. 5. We used pre-trained models and fine-tuning network parameters to reduce training time. The generator of FTE was initialized using CycleGAN trained network weights, fine-tuning the generator parameters with a smaller learning rate (we set the learning rate to 0.00001 - 0.0001). Similarly, we used the weights trained by the detector in *3)* to initialize OD and fine-tuned with a smaller learning rate. We randomly initialized the weight of the feature fusion of FTE and set a large initial learning rate (0.001 - 0.01). The input image resolution was set to 416 × 416 (including the network with the SSD300-based structure as the detection module). Image preprocessing mainly included random brightness, color jittering and random mirror.

To ensure that each method got the best results, we trained each network for 50 epochs. The validation phase was performed during the training period, and the best model was saved. Due to the different parameters of each network, the training time were different. The training time for YOLOv3, SSD, Faster RCNN, and our method were about 52 hours, 20 hours, 46 hours and 63 hours, respectively.

5) EXPERIMENTAL PLATFORM AND COMPUTATIONAL PERFORMANCE

The proposed network was implemented with the pytorch framework running on a PC with Intel(R) Xeon(R) E5-2650V4 CPU 2.2GHz and NVIDA GTX1080Ti GPU. The machine was running Linux Ubuntu 16.04 with NVIDA CUDA 9.0 and cuDNN 7.0.

TABLE 2. Comparison of AP with different networks.

Networks	DATASET1 (hard)	DATASET1 (easy)	DATASET2 (private)
YOLOv3 [36]	0.519	0.852	0.794
FteGanOd + YOLOv3	0.583	0.904	0.856
SSD300 [33]	0.441	0.802	0.780
FteGanOd + SSD300	0.495	0.834	0.832
Faster RCNN [30]	0.496	0.875	0.819
FteGanOd + Faster RCNN	0.543	0.916	0.874

Note: FteGanOd + YOLOv3, FteGanOd + SSD300 and FteGanOd + Faster RCNN represent our proposed detection FteGanOd network based on FTE module and modified cascaded detectors.

C. RESULTS AND DISCUSSIONS

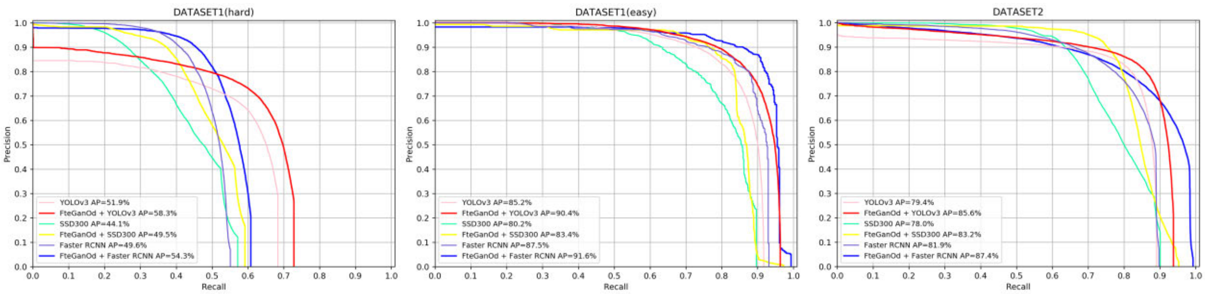
The experimental results are presented and discussed in this section. The proposed method was validated on different datasets as shown in TABLE 2.

On DATASET1 (hard), the AP of FteGanOd + YOLOv3 improve by 6.4% compared with YOLOv3, the AP of FteGanOd + SSD improve 5.4% compared with SSD, and FteGanOd + Faster RCNN improve 4.7% compared with Faster RCNN. On DATASET1 (easy) and DATASET2, the APs of the cascaded networks increase by about 6%.

Fig. 7 shows the comparison of the translation, fusion and enhancement results between traditional CycleGAN (Fig. 7 (b)) and our proposed FteGanOd (Fig. 7 (c)) testing on DATASET1 (hard). The results from traditional CycleGAN (Fig. 7 (b)) are quite different from the real daytime images because lots of detail textures are lost, such as the blurred contour of cars and weak taillights, which decreases the accuracy of the vehicle detection. Fig. 7 (c) shows the fused and enhanced results of our proposed FteGanOd with sharp vehicle contours and clear taillights compared with CycleGAN from the enlarged parts. This increased detail helps to distinguish the vehicles in the low-light environment and

(a) Original image (b) FTE method (c) FteGanOd + YOLOv3 (d) YOLOv3

FIGURE 8. Detection results from DATASET1(hard): (a) Original image. (b) FTE results. (c) Detection results with FteGanOd + YOLOv3. (d) Detection results with YOLOv3.



(a) DATASET1(hard) (b) DATASET1(easy) (c) DATASET2

FIGURE 9. PR curves of each network on different datasets.

reduces the interference from streetlights, leading to higher accurate vehicle detection.

Fig. 8 shows the comparison of vehicle detection results between YOLOv3 and our proposed FteGanOd method on dataset DATASET1 (hard). It can be seen that FteGanOd detects more vehicles than YOLOv3 in lowlight condition. Therefore, the proposed FteGanOd with FTE and OD modules significantly reduces the number of false positive detection.

Fig. 9 shows the PR curve of each network tested on different datasets. It can be seen that our proposed method achieved higher precision at the same recall rate. It can be seen from Fig. 9(a) that at recall = 0.5, the precision of FteGanOd + Faster RCNN is 0.824, Faster RCNN is 0.633, representing an improvement of about 19%. When the precision is fixed at 0.8, the recall of FteGanOd + Faster RCNN is 0.519, compared to 0.451 for Faster RCNN, an improvement of about 7% Therefore, our proposed FteGanOd method has higher precision and recall than other methods.

TABLE 3 shows the detection speed and parameters of each network with input batch size = 1 using the same testing dataset.

After adding our proposed FTE module, the detection speed decreases. FteGanOd + YOLOv3 achieves the detection speed of 16.2 FPS, and the average inference time for an image is about 0.061 s.

TABLE 3. Computational performance.

Networks	FPS	Average time (s)	Amount of parameters	AP
YOLOv3	37.4	0.027	61.53 M	0.852
FteGanOd + YOLOv3	16.2	0.061	74.48 M	0.904
SSD300	33.5	0.030	24.01 M	0.802
FteGanOd + SSD300	11.7	0.085	35.76 M	0.834
Faster RCNN	10.4	0.096	47.28 M	0.875
FteGanOd + Faster RCNN	2.7	0.370	59.98 M	0.916

D. COMPARISON WITH EXISTING METHODS

Most recent studies on nighttime vehicle detection validate their methods on their own private datasets, which are not available in public. It is difficult for us to compare our method with them as we have no benchmark dataset for night vehicle detection. Due to different scenes, there are great differences between these datasets. Fig. 10 shows some sample images from other datasets published in related papers. They used different density and complexity datasets. Therefore, we cannot directly compare our experimental results with theirs.

However, we can analyze the different results from other perspectives. TABLE 4 lists the detection accuracy from other literature. Fig. 10 shows the corresponding sample images published in their papers. Most methods were evaluated using low-density traffic conditions and low-complexity backgrounds. The datasets we used (DATASET1 (easy)

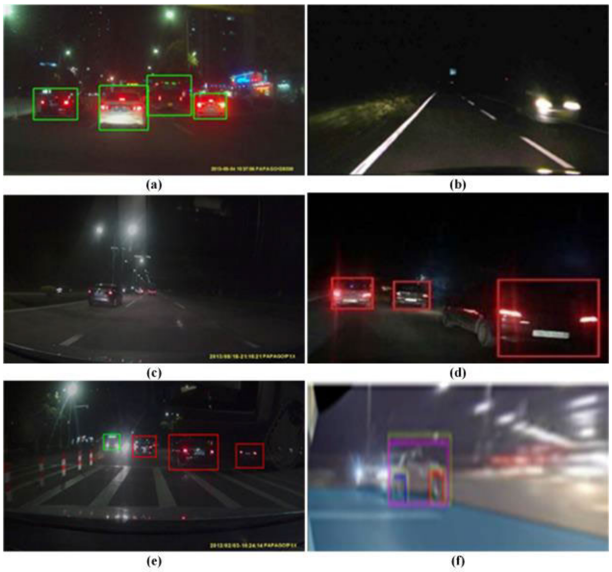


FIGURE 10. Sample images of other datasets published in related paper: (a) Images with complex scenes used in [1]; (b) Light blob detection dataset from [13]; (c) Night image from [14]; (d) Night image and detection result from [15]; (e) image from [24] with annotations. (f) Night image and detection result from [22].

TABLE 4. Comparison of accuracy and datasets.

Methods	Detection rate
[1]	95.95%
[13]	92.10%
[14]	95.00%
[15]	93.67%
[24]	87.66%
[22]	97.1%
Ours	97.8%

and DATASET2) contain an average of 5 target vehicles per image, and are more complex than those private datasets [1], [13]–[15], [24], [22]. Our method can adapt to different light conditions under natural driving conditions. It can be seen that our method (e.g. FteGanOd + Faster RCNN) either outperforms or achieves similar detection rates on more complex datasets compared with existing methods on lower complexity datasets. The method of [22] achieved a precision of 97.1% and a recall of 55.0%. Our method (FteGanOd + Faster RCNN) achieve a higher precision of 97.8% at the same recall.

V. CONCLUSION

In this paper, we propose an effective night detection method called FteGanOd that cascades the feature translate- enhancement FTE module and the object detection OD module. This method is designed to solve the problem of low detection of vehicles at night on city roads with weak/complex lighting environment and dense traffic flow. The proposed FTE module uses unpaired input image and CycleGAN to translate the nighttime images into daytime images, and further fuses multi-scale feature to enhance the vehicle features at night. The enhanced features retain the important information of vehicle lights at night and augment the vehicle features during the day. Our experimental results show that the proposed method effectively enhances the features of nighttime vehicles and suppresses the interference from other lights. It is helpful to improve the detection accuracy and reduce the false/missed detection rate. In addition, we found that there are some small targets in the remote distance in DATASET1 (hard) were missed after passing through the FTE module. This likely occurred because small targets were weakened and recognized as part of the background. To improve the detection rate for remote small targets at night is the important work we need to study in the future.

REFERENCES

[1] H. Kuang, X. Zhang, Y.-J. Li, L. L. H. Chan, and H. Yan, "Nighttime vehicle detection based on bio-inspired image enhancement and weighted score-level feature fusion," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 4, pp. 927–936, Apr. 2017.

[2] L. Chen, X. Hu, T. Xu, H. Kuang, and Q. Li, "Turn signal detection during nighttime by CNN detector and perceptual hashing tracking," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 12, pp. 3303–3314, Dec. 2017.

[3] H. Kuang, L. Chen, L. L. H. Chan, R. C. C. Cheung, and H. Yan, "Feature selection based on tensor decomposition and object proposal for nighttime multiclass vehicle detection," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 49, no. 1, pp. 71–80, Jan. 2019.

[4] Y.-L. Chen, C.-T. Lin, C.-J. Fan, C.-M. Hsieh, and B.-F. Wu, "Vision-based nighttime vehicle detection and range estimation for driver assistance," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, Singapore, Oct. 2008, pp. 2988–2993.

[5] Y. L. Chen, "Nighttime vehicle light detection on a moving vehicle using image segmentation and analysis techniques," *WSEAS Trans. Comput.*, vol. 8, no. 3, pp. 506–515, 2009.

[6] S. Gonner, D. Muller, S. Hold, M. Meuter, and A. Kummert, "Vehicle recognition and TTC estimation at night based on spotlight pairing," in *Proc. 12th Int. IEEE Conf. Intell. Transp. Syst.*, St. Louis, MO, USA, Oct. 2009, pp. 1–6.

[7] R. O'Malley, E. Jones, and M. Glavin, "Rear-lamp vehicle detection and tracking in low-exposure color video for night conditions," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 453–462, Jun. 2010.

[8] C.-L. Chien, H.-M. Hang, D.-C. Tseng, and Y.-S. Chen, "An image based overexposed taillight detection method for frontal vehicle detection in night vision," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA)*, Jeju, South Korea, Dec. 2016, pp. 1–9.

[9] J. C. Rubio, J. Serrat, A. M. Lopez, and D. Ponsa, "Multiple-target tracking for intelligent headlights control," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 2, pp. 594–605, Jun. 2012.

[10] S. Sivaraman and M. M. Trivedi, "Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1773–1795, Dec. 2013.

[11] D. Juric and S. Loncaric, "A method for on-road night-time vehicle headlight detection and tracking," in *Proc. Int. Conf. Connected Vehicles Expo (ICCVE)*, Vienna, Austria, Nov. 2014, pp. 655–660.

[12] J.-M. Guo, C.-H. Hsia, K. Wong, J.-Y. Wu, Y.-T. Wu, and N.-J. Wang, "Nighttime vehicle lamp detection and tracking with adaptive mask training," *IEEE Trans. Veh. Technol.*, vol. 65, no. 6, pp. 4023–4032, Jun. 2016.

[13] N. Kosaka and G. Ohashi, "Vision-based nighttime vehicle detection using CenSurE and SVM," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 5, pp. 2599–2608, Oct. 2015.

[14] C. S. Pradeep and R. Ramanathan, "An improved technique for night-time vehicle detection," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, Bengaluru, India, Sep. 2018, pp. 508–513.

[15] A. Zaarane, I. Slimani, W. Al Okaishi, I. Atouf, and A. Hamdoun, "An automated night-time vehicle detection system for driving assistance based on cross-correlation," in *Proc. Int. Conf. Syst. Collaboration Big Data, Internet Things Secur. (SysCoBIoTS)*, Casablanca, Morocco, Dec. 2019, pp. 1–5.

[16] T. Nakane, T. Takeshita, S. Tokai, and C. Zhang, "Vehicle rear-lamp detection at nighttime via probabilistic bitwise genetic algorithm," in *Proc. Int. Conf. Cyberworlds (CW)*, Kyoto, Japan, Oct. 2019, pp. 117–120.

[17] F. Gao, Y. Ge, S. Lu, and Y. Zhang, "On-line vehicle detection at nighttime-based tail-light pairing with saliency detection in the multi-lane intersection," *IET Intell. Transp. Syst.*, vol. 13, no. 3, pp. 515–522, Mar. 2019.

[18] X. Dai, D. Liu, L. Yang, and Y. Liu, "Research on headlight technology of night vehicle intelligent detection based on Hough transform," in *Proc. Int. Conf. Intell. Transp., Big Data Smart City (ICITBS)*, Changsha, China, Jan. 2019, pp. 49–52.

[19] Y. Cai, X. Sun, H. Wang, L. Chen, and H. Jiang, "Night-time vehicle detection algorithm based on visual saliency and deep learning," *J. Sensors*, vol. 2016, pp. 1–7, Nov. 2016, doi: 10.1155/2016/8046529.

[20] V. F. Arruda, T. M. Paixao, R. F. Berriel, A. F. De Souza, C. Badue, N. Sebe, and T. Oliveira-Santos, "Cross-domain car detection using unsupervised image-to-image translation: From day to night," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Budapest, Hungary, Jul. 2019, pp. 1–8.

[21] C.-T. Lin, "Cross domain adaptation for on-road object detection using multimodal structure-consistent image-to-image translation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Taipei, Taiwan, Sep. 2019, pp. 3029–3030.

[22] H. Lee, M. Ra, and W.-Y. Kim, "Nighttime data augmentation using GAN for improving blind-spot detection," *IEEE Access*, vol. 8, pp. 48049–48059, 2020.

[23] C.-T. Lin, S.-W. Huang, Y.-Y. Wu, and S.-H. Lai, "GAN-based day-to-night image style transfer for nighttime vehicle detection," *IEEE Trans. Intell. Transp. Syst.*, early access, Jan. 6, 2020, doi: 10.1109/TITS.2019.2961679.

[24] Y. Mo, G. Han, H. Zhang, X. Xu, and W. Qu, "Highlight-assisted nighttime vehicle detection using a multi-level fusion network and label hierarchy," *Neurocomputing*, vol. 355, pp. 13–23, Aug. 2019.

[25] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "BDD100K: A diverse driving dataset for heterogeneous multitask learning," 2018, *arXiv:1805.04687*. [Online]. Available: http://arxiv.org/abs/1805.04687

[26] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 2242–2251.

[27] J. Chen, J. Chen, and F. Gu, "Nighttime vehicle detection using deformable parts model," in *Proc. 7th Int. Conf. Intell. Hum.-Mach. Syst. Cybern.*, vol. 2, Aug. 2015, pp. 480–483, 2015.

[28] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, Jun. 2014, pp. 580–587.

[29] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 1440–1448.

[30] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

[32] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 936–944.

[33] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.

[34] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 779–788.

[35] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 6517–6525.

[36] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: http://arxiv.org/abs/1804.02767

[37] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 105–114, doi: 10.1109/CVPR.2017.19.

[38] H. Zhang, V. Sindagi, and V. M. Patel, "Image de-raining using a conditional generative adversarial network," 2017, *arXiv:1701.05957*. [Online]. Available: http://arxiv.org/abs/1701.05957

[39] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 5967–5976, doi: 10.1109/CVPR.2017.632.

[40] X. Wang, H. Yan, C. Huo, J. Yu, and C. Pant, "Enhancing Pix2Pix for remote sensing image classification," in *Proc. 24th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2018, pp. 2332–2336.

XIAOTAO SHAO (Member, IEEE) was born in Xi'an, Shaanxi, China, in 1965. She received the B.S. and M.S. degrees in wireless technology and electromagnetic field and microwave technology from Xi'an Jiaotong University, Xi'an, in 1986 and 1991, respectively.

From 1996 to 2001, she studied and worked as a Development Engineer and a Senior Engineer, Singapore. She is currently an Associated Professor with the School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China. She is the author of eight books and more than 20 articles. Her research interests include electromagnetic wave propagation, 5G technology, deep learning, object detection, and mesh networking.

CAIKE WEI was born in Nanning, Guangxi, China, in 1995. He received the B.S. degree in electronic and information engineering from Beijing Jiaotong University, Beijing, China, in 2018, where he is currently pursuing the M.S. degree with the School of Electronic and Information Engineering.

YAN SHEN (Member, IEEE) was born in Xi'an, Shaanxi, China, in 1979. She received the B.S. and M.S. degrees in telecommunication engineering from Xidian University, Xi'an, in 2001 and 2004, respectively, and the Ph.D. degree in communication and information system from the Institute of Electronics, Chinese Academy of Sciences, Beijing, China, in 2007.

She is currently an Associated Professor and the Tutor of Ph.D. student with the School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing. She is the author of three books, more than 40 articles, and more than ten inventions and software copyrights. Her research interests include deep learning, image restoration, sparse representation, and object detection.

ZHONGLI WANG (Member, IEEE) received the B.S. and M.S. degrees in welding automation from the Harbin Institute of Technology (HIT), Harbin, China, in 1992 and 1995 respectively, and the Ph.D. degree in mechatronics and automation from the Robot Institute, Harbin Institute of Technology (HIT), in 2000.

From 2001 to 2003, he held a postdoctoral position with the Department of Computer Science and Technology, Beijing Institute of Technology, Beijing, China. From 2006 to 2008, he was a Visiting Scholar with the Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Hong Kong. From 2012 to 2013, he was a Visiting Scholar with The Robotic Institute, Carnegie Mellon University, Pittsburgh, PA, USA. He has been an Associate Professor with Beijing Jiaotong University, since 2010. His research interests include perception and control of robot, computer vision, embedded computing, and system design.

Prof. Wang's awards and honors, including Second Class Award of Science and Technology Invention from the Ministry of Education and the Excellent Project Leader Award from the Institute of Computing Technology, Chinese Academic of Science, China.