AN INTEGRATED MODEL FOR PRECEDING VEHICLE TARGET DETECTION AND LOCATION BASED ON DEEP VISION

Guoqiang Chen[1], Zhongchao Bai[1], Mengchao Liu[1]
[1]School of Mechanical and Power Engineering, Henan Polytechnic University, Jiaozuo 454003, China
Email: chengq@hpu.edu.cn

Abstract - Vehicle location is a crucial task in autonomous driving. Traditional vehicle location methods use the GPS, the inertial navigation system, and the odometer. This study addresses a way to improve the preceding vehicle location accuracy based on deep vision. The preceding vehicle distance and azimuth detection model is trained based on the Regression Convolutional Neural Networks (R-CNN). The Faster Regression Convolutional Neural Networks (Faster R-CNN) network structure is optimized to make it more suitable for preceding vehicle location. The proposed method is verified through experiments. The proposed method can accurately detect the position, azimuth, and distance of the preceding vehicle. This study provides useful guidance for further research on preceding vehicle location.

Keywords: Deep vision; Preceding vehicle location; Regression Convolutional Neural Networks (R-CNN); Faster Regression Convolutional Neural Networks (Faster R-CNN).

1. Introduction

The environmental perception system and the positioning system are the fundamental parts in autonomous driving. The environmental information is matched with the autonomous vehicle system to obtain information including location, speed and direction of the surrounding vehicles. According to the positioning results of the preceding vehicle, the decision system can plan the system, control the driving and maintain a safe vehicle distance. If there is a large deviation in the positioning accuracy, it will lead to error planning and various errors in the decision system, such as failure in ensuring safe distance, illegal lane change, illegal overtaking and other unsafe behaviors. Preceding vehicle positioning includes two stages, target classification and target positioning. It is necessary to identify and locate the front target vehicle. The background is usually complex in the real road scene, which has a great challenge to the detection accuracy. Kamalesh et al. proposed a real-time pothole detection and warning system by combing Internet of Things technology [1]. Mekki et al. proposed the evolutionary game-based vehicular cloud access algorithm (EG-VCA) and the Q-learning-based vehicular cloud access algorithm [2]. These studies provide effective guidance for autonomous driving.

Target detection has one stage-target detection and two-stage target detection. One-stage target detection method with real-time performance includes YOLO v1, YOLO v2, YOLO v3, YOLO v4 (Real-Time Object Detection. You Only Look Once) [3] and Single Shot Multi Box Detector (SSD) [4]. For the target detection, Dow et al. combined deep learning classifier and zebra-crossing recognition techniques to reduce accidents at intersections [5]. Ren et al. designed a method with fast detection speed that feature extraction, target classification, and position regression are carried out in the whole Convolutional Neural Networks (CNN). The input image of the Faster Regression Convolutional Neural Networks (Faster R-CNN) [6] model uses CNN to extract image features, which has real-time performance [7]. Therefore, this study uses the Faster R-CNN to detect vehicles to improve real-time performance. Distance measurement methods are classified into several categories, which are ultrasonic ranging, millimeter-wave radar ranging, laser ranging, and visual ranging. Although ultrasonic ranging has a wide application range, it can't synchronize the real-time distance change with large measurement error. The millimeter wave radar has a relatively stable system and high cost. Its frequency bands are 60GHz and 120GHz.

Millimeter-wave radar is used to detect the speed on the highway.

Laser ranging has high precision, it needs to pay attention to human safety. Visual ranging is based on the camera to match feature points of two images captured by the camera. The monocular vision ranging system is widely used, but its own principle limits its accuracy. The pose estimation must be maintained independently of environmental factors. Above all, the traditional azimuth and range measurements are relying on sophisticated sensors, which increases the cost. The accuracy of

measurement is easily affected by complex environment. Deep vision is a method of processing and analyzing images based on deep learning. Deep vision technology can be regarded as deep learning technology in image research. Detecting the distance and azimuth of the preceding vehicle to accurately locate the vehicle is crucially important to autonomous vehicles based on deep vision.

In this paper, an integrated model for preceding vehicle target detection and location is proposed and tested. The objective of the paper is to construct the integrated model and to verify the detection accuracy and efficiency via real scene. It can rapidly and accurately identify the azimuth and distance of the preceding vehicle by using the deep vision method. This study provides a new reference for the positioning technology of the preceding vehicle.

2. An Integrated Model for Preceding Vehicle Location

The preceding vehicle positioning system based on the deep vision consists of three modules. Firstly, when the vehicle camera uses Faster R-CNN to realize the preceding vehicle target detection, this module can accurately detect the preceding target. To make the Regression CNN better adapt to the characteristics of the text data set, this study optimizes them, which are named Regression CNN1 and Regression CNN2.

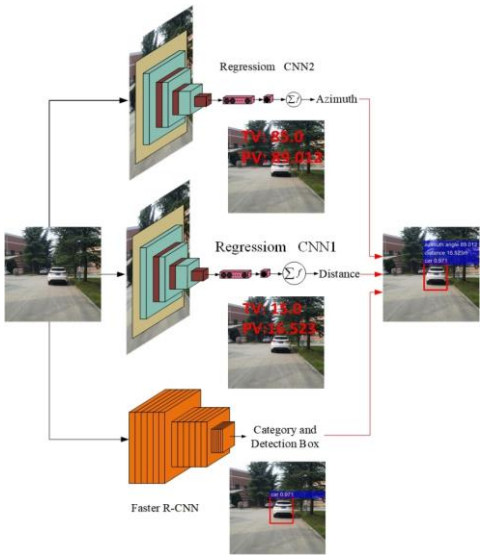

Figure 1. Integrated model for preceding vehicle target detection and location

When the preceding vehicle is accurately detected, the Regression CNN1 is used to detect the distance of the preceding vehicle, and the Regression CNN2 is used to detect the azimuth of the preceding vehicle. Faster-RCNN and Regression CNN are combined to synchronously detect the position, azimuth, distance of the preceding vehicle and realize the preceding vehicle positioning.

The detection system framework is proposed in this study shown in Figure 1. The preceding vehicle is located according to the images captured by the vehicle camera.

2.1 Preceding Vehicle Azimuth and Distance Detection Method

The preceding vehicle positioning method is proposed in this study, shown in Figure 2. When the vehicle runs on the smooth road, the preceding vehicle and its own vehicle are in the same horizontal position. The angle between target 1 and target 2 is the azimuth of the preceding vehicle. The preceding vehicle is located on immediately forward, and the preceding vehicle azimuth is 90°. The distance between the camera and the preceding vehicle is the detection distance. The location of the camera that captures the target image is shown in the image containing the frontal view of the vehicle.

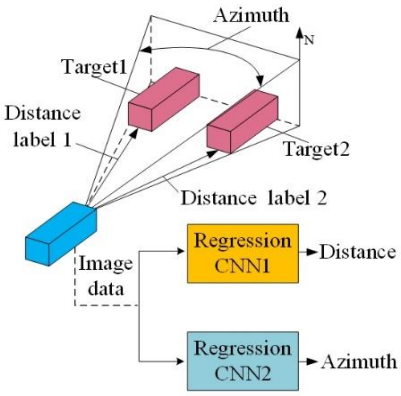

Figure 2. Preceding vehicle positioning method

2.2 Preceding Vehicle Azimuth and Distance Detection Network

The composition of the azimuth and distance detection network is as follows. Preceding vehicle's distance detection is based on Regression CNN1. Preceding vehicle's azimuth detection is based on Regression CNN2. The architecture of the Regression CNN can be divided into a data layer, three convolution layers, three pooling layers, two fully connected layers, and one M layer [8]. The data layer uses HDF5 data format that supports floating point tags [9,10]. To make this network more suitable for the data set, the structure parameters are optimized. Activation and loss functions are designed to adapt the application. The structure of Regression CNN is shown in Figure 3. The activation functions of the Regression CNN are Sigmoid and PReLu (linear parameter unit). The existence of activation function increases the nonlinear factor of Regression CNN and improves the expression ability of the model. The output range of Sigmoid is between 0 and 1. If there is an input signal in the network, the sigmoid responds to the output [11].

When the output value of neurons is close to the maximum, the function curve becomes very smooth. When the derivative of error feedback is close to 0, the activation process is blocked, and the lower error cannot be transmitted to the upper layer.

The network training will fail. Four PReLu activation functions behind the convolution layer and fully connected layer 1 to optimize the network structure [12].

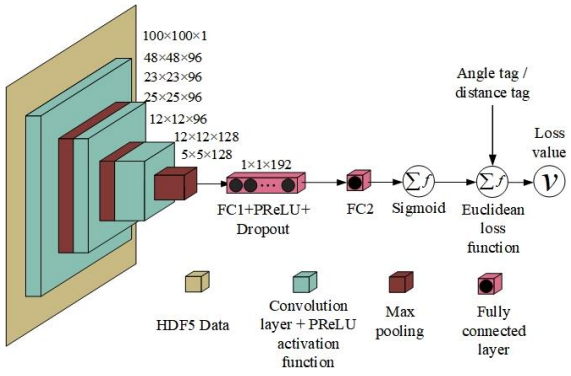

Figure 3. Optimized Regression CNN network structure

The loss value of Regression CNN using PReLu is less than that of Regression CNN using ReLu. The convergence speed is faster. The increased computation and overfitting risk for PReLu approach zero. PReLu [13] can be expressed as

$$f(x)=\begin{cases} x & x \geq 0 \\ ax & x \leq 0 \end{cases} \tag{1}$$

Suppose the number of training sample is N, the label value is expressed by y, the detection value is expressed by $\hat{y}$. The Euclidean loss function can be expressed as

$$E_{\text{EuclideanLoss}}=\frac{1}{N}\sum_{i=0}^{N}(y_i-\hat{y}_i)^2 \tag{2}$$

Faster R-CNN can be considered as a combination of Fast R-CNN and RPN. Its candidate region extraction network is RPN. It's equivalent to generating multiple candidate anchors on the scale of the input image.

It determines which foreground anchors contain detection targets and which background anchors do not contain detection targets based on CNN. The network will eliminate this part of the candidate regions, anchors with regional characteristics as the background need not participate in training. Boundary box regression foreground anchor only using RPN loss function. More accurate candidate regions can be obtained through the above process. RPN loss function is the sum of classification loss and boundary box regression loss. The application principle of Faster R-CNN in vehicle positioning showing in Figure 4.

RPN generates many anchors on the feature map, and the performs 3 × 3 convolution on the feature image. Then each pixel is mapped to the corresponding coordinate point of the input image. Finally, taking this point as the centre, three different sizes of ROI are generated by three different proportions and three different sizes of anchor boxes.

This part of the candidate regions, anchors participated in training. Boundary box regression for foreground anchor only using RPN loss function.

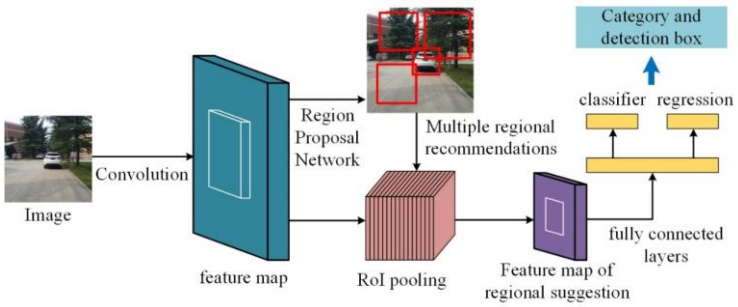

Figure. 4. The application principle of Faster R-CNN in vehicle positioning

Considering the convenience of comparing the difference between label value and output value, the Regression CNN adopts Sigmoid to receive the output of the full connection layer according to the characteristics of the vehicle data set. The Dropout layer is used to solve the network overfitting

problem in Regression CNN. The Euclidean loss function is applied to the loss function layer. Euclidean loss function is defined as the error between the calculated label value and the detected value. The anchor generation mechanism is shown in Figure 5.

The loss function of Faster R-CNN is expressed as:

$$L(\{p_i\},\{t_i\})=\frac{1}{N_{cls}}\sum_i L_{cls}(p_i,p_i^*)+\lambda\frac{1}{N_{reg}}\sum_i p_i^* L_{reg}(t_i,t_i^*) \quad (3)$$

Were, i is the index of an anchor in a mini-batch and p_i is the predicted probability of anchor i being an object. The ground-truth label p_i^* is 1 if the anchor is positive, and it is 0 if the anchor is negative. t_i^* Represents a vector composed of four parameterized coordinates and a real boundary frame coordinates associated with a positive label anchor. L_{cls} is classification loss, it is the logarithmic loss of two categories (target is not target). For the return losses, $L_{\text{reg}}(t_i,t_i^*)=R(t_i-t_i^*)$. R is robust loss function. The term $p_i^* L_{\text{reg}}$ means the regression loss is activated only for positive anchors ($p_i^*=1$) and is disabled otherwise ($p_i^*=0$).The outputs of the *cls* and *reg* layers consist of p_i and t_i .The two terms are normalized by N_{cls} and N_{reg} weighted by a balancing parameter λ .[6]

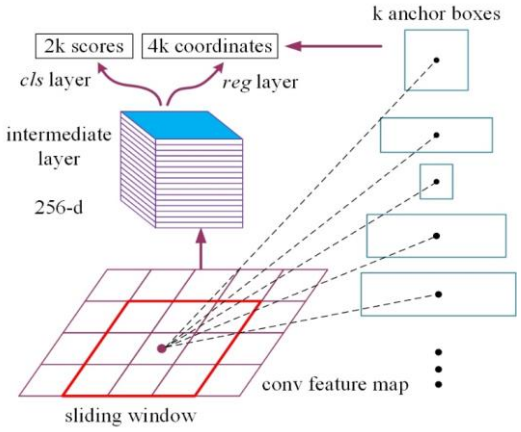

Figure. 5. Anchor generation mechanism in the Faster R-CNN

3. Experiment and Analysis

3.1 Preceding Vehicle Azimuth and Distance Detection Experiment

The data set training is obtained by actual measurement by Electronic Total Station. The collection environment is the smooth road in the campus, and the preceding vehicle center and the camera are in the same horizontal position.

It is calculated that all images in the dataset have 500 rectangles. To improve the detection accuracy, the anchor value of Faster R-CNN is fine-tuned.

The aspect ratio and scale of anchor are changed to (0.2,0.35,0.5,1,2,3,4) and (64,128,256,512).

In this study, the gray image is used to train and test. For the detection model, the gray image can be used to calculate faster.

The pixel normalization method in this study is as follows, the mean value of each image is reduced, and then each pixel value of the image is divided by the standard deviation of the mean value of all images. The training parameters for this network development platform are shown in Table 1.

Table 1. Composition of network development platform

Category	Name	Type
Hardware	Central Processing Unit	Inter(R) Core i5 CPU
	RAM	8G
	GPU	2080Ti
Software	Operating system	Ubuntu 18.04
	Deep learning framework	Caffe
	Deep learning network1	Faster R-CNN
	Deep learning network2	Regression CNN
	Programming language environment	Python 3.7
	Open source visual library	OpenCV 4.1.0

1) Data format processing: The format of the data set needs to be processed as HDF5 (Hierarchical Data Format Version 5) format, and the image label name format of the data set is image sequence number _ label value.jpg. The HDF5 dataset is saved as a data layer and entered Regression CNN for training Regression CNN model 1 and Regression CNN model 2.

2) Label value normalization: the label value of the training data set needs to be normalized to [0,1], which is conducive to accelerating the convergence rate of network training. It improves the detection accuracy of the Regression CNN vehicle detection model. In the test phase, the network internal test

algorithm enlarges the detection value to the same order of magnitude as the label value.

Another important reason for normalizing the label value is that the last activation function of the Regression CNN is Sigmoid. From the output value range of the Sigmoid activation function, it is convenient to compare the label with the detection value.

3) Image de-means: To standardize the image of the dataset, the pixel value of each image needs to subtract the average pixel value of the dataset image. The higher brightness value of the image can be removed by subtracting the average brightness value of the dataset image [14].

The purpose is to reduce the calculation amount and change the standard image matrix coordinate system of the original image matrix-vector into a new image matrix coordinate system with the coordinate origin as the mean of these vectors [15].

Table 2. Training parameters of the networks

Model / Training parameters	Regression CNN Model 1	Regression CNN Model 2	Faster R-CNN
Train sets	300	300	500
Validation sets	50	50	200
Test sets	100	100	100
Train loss	7.23821e-06	3.18294e-05	0.01253
Verification loss	9.73815e-05	9.36489e-06	0.07523
Iteration	50k	50k	120k
Global learning rate	0.001	0.001	0.0001
Average detecting time	0.043s	0.043s	0.043s

Figure. 6. Distance detection results of preceding vehicle

After subtracting the mean, the coordinate origin of the image matrix coordinate system is converted to the mean.

4) Image pixel normalization: From the perspective of CNN reverse propagation and weight optimization, if the gradient of the optimization function is large, a lower learning rate should be selected to ensure that the optimization function does not cross the optimal solution [16,17].

The size of the input data must be considered when selecting the learning rate. If pixel values are normalized, the selection of learning rates will become more convenient [18]. The normalization of image pixels is to ensure that all dimensions of the

image matrix are within the same variation range [19].

The total number of samples of the dataset is 1300, of which 700 are used for the training of the Faster R-CNN preceding vehicle positioning model, 300 are used for the training of the preceding vehicle ranging model 1 based on the Regression CNN, and 300 are used for the training of the preceding vehicle azimuth detection model 2 based on the Regression CNN. Based on the analysis of training loss and verification loss of the model, model 1 and model 2 have good convergence. The average detection accuracy of Faster R-CNN is 95.7 %. The training parameters of each network are shown in Table 2.

3.2 Detection Results of Preceding Vehicle Distance

The experimental results of detecting the preceding vehicle distance indicate that TV (True Value) in each test diagram represents the true value, by Electronic Total Station and PV (Predictive Value) represents the predicted value. The results are shown in Figure 6.

A total of 46 test charts with different distances were used in the test, and the distance ranged from 5 to 50 m. To the network to output the distance value directly during the test process, the Euclidean loss function layer is no longer required during the distance detection process. Instead, convert the magnitude of the detected value activated by the Sigmoid activation function directly to the range of the original distance value, and then output the distance value directly.

According to the results of Table 3, the predicted value and the real value are analysed. The prediction results are more accurate when the distance is large, and the prediction distance fluctuates when the distance is short. Above all, the predicted results have a good performance.

Table 3. Aberration table of ranging method based on Regression CNN

True distance/m	Detection distance/m	Absolute error/m	Relative error/%
10	9.58	0.42	4.20
15	15.36	0.36	2.40
20	19.83	0.17	0.85
25	25.03	0.03	0.12
30	30.10	0.10	0.33
35	35.13	0.13	0.37
40	39.82	0.08	0.20
50	49.83	0.17	0.34

The monocular vision ranging results in the Reference [20] are compared with the experimental results in this study. The maximum error in this study is 4.2 %, the minimum error is only 0.12 %, and the average error is 1.10125 %, while the maximum error of the monocular vision ranging algorithm is 15.68 %, the minimum error is 0.70 %, and the average error is 3.62375 %.

Figure 7 is the error comparison chart of the ranging algorithm and monocular ranging algorithm. The increase of distance and the ranging accuracy of the method is gradually stable, and the monocular vision ranging algorithm, with the increase of ranging distance, ranging error is gradually increased. Obviously, this method is better than monocular vision ranging.

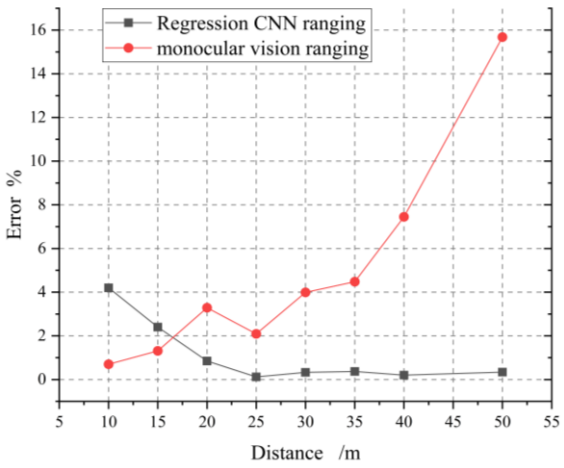

Figure. 7. Comparison between monocular vision ranging and Regression CNN ranging

3.3 Detection Results of Preceding Vehicle Azimuth

As shown in the Figure 8, it shows the partial results of the preceding vehicle's azimuth detection. The preceding vehicle azimuth range is between 60°and 120°, which is determined by the azimuth range of the preceding vehicle taken.

The principle of the detection process is the same as distance detection. According to the detection results, the preceding vehicle azimuth detection based on Regression CNN has high detection accuracy.

In the test results, PV represents the predicted value, TV represents the real value by Electronic Total Station, and the unit is degree.

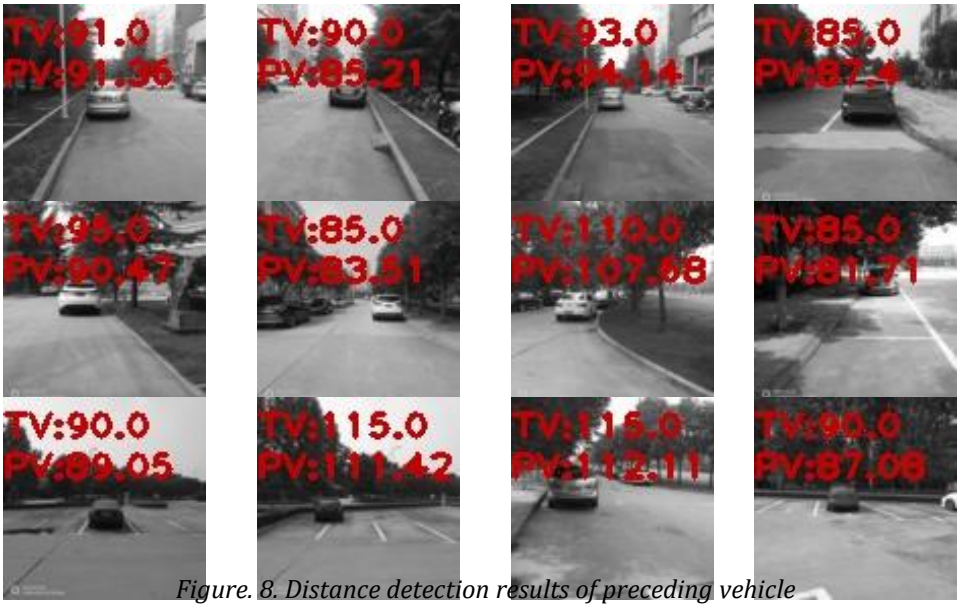

Figure. 8. Distance detection results of preceding vehicle

3.4 Preceding Vehicle Positioning Results

After the training of Regression CNN1, Regression CNN2 and Faster R-CNN model, the three training models are integrated into the preceding vehicle positioning. For verifying the accuracy of the preceding vehicle positioning, a single target vehicle image is randomly selected for testing. In the test results, the azimuth angle is the detection value of azimuth angle, and distance is the detection value of the preceding vehicle distance. The true data and detection data are shown in Table 4. The prediction results are shown in Figure 9. According to the test results of random images, the preceding vehicle location can first accurately identify the preceding vehicle, and it can also achieve good accuracy in azimuth and distance detection.

Table 4. True data and detection data

	True distance/ m	Detection distance/ m	True azimuth angle/ °	Detection azimuth angle/ °
1	4	4.53	90	91.642
2	12	12.04	90	87.549
3	9	8.97	85	85.139
4	8	7.81	90	91.527

Figure. 9. Vehicle Location Test Results

4. Conclusions

An integrated model for preceding vehicle target detection and location based on deep vision is proposed. The preceding vehicle positioning, azimuth and distance detection methods are constructed. The data set is images from taken campus roads.

Firstly, the distance and azimuth angle between the camera and the preceding vehicle is measured accurately. Then data as label values is used to match the corresponding image. The preceding vehicle distance and azimuth detection model is obtained by setting the learning rate and the number of iterations. In this study, Faster R-CNN is used to train the preceding vehicle positioning model to locate the preceding vehicle's specific location accurately. The result of experiment indicates that the algorithm has good performance in preceding vehicle positioning. In future work, the laser radar technology can be further combined to locate the preceding vehicle, detect the azimuth and distance based on the information fusion of point cloud data and deep vision technology.

Acknowledgement

This work is supported by the Key Technology R&D Program of Henan Province of China (No. 212102210045 and 212102210050).

References

[1] Kamalesh, M. S., Chokkalingam, B., Arumugam, J., Sengottaiyan, G., Subramani, S., & Shah, M. A. (2021). An Intelligent Real Time Pothole Detection and Warning System for Automobile Applications Based on IoT Technology. Journal of Applied Science and Engineering, 24(1), 77-81.

[2] Mekki, T., Jabri, I., Rachedi, A., & Jemaa, M. B. (2019). Vehicular cloud networking: evolutionary game with reinforcement learning-based access approach. International Journal of Bio-Inspired Computation, 13(1), 45-58.

[3] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, 779-788.

[4] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. In European conference on computer vision, 21-37.

[5] Dow, C. R., Ngo, H. H., Lee, L. H., Lai, P. Y., Wang, K. C., & Bui, V. T. (2020). A crosswalk pedestrian recognition system by using deep learning and zebra-crossing recognition techniques. Software: Practice and Experience, 50(5), 630-644.

[6] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems, 28, 91-99.

[7] Chen, H. D., Ding, X. Y., Liu, Y. X. (2021). Overview of target detection algorithms based on deep learning. Journal of Beijing United University, 35 (03), 39-46.

[8] Si, D., Moritz, S. A., Pfab, J., Hou, J., Cao, R., Wang, L., ... & Cheng, J. (2020). Deep learning to predict protein backbone structure from high-resolution cryo-EM density maps. Scientific reports, 10(1), 1-22.

[9] Di Natali, C., Beccani, M., & Valdastri, P. (2013). Real-time pose detection for magnetic medical devices. IEEE Transactions on Magnetics, 49(7), 3524-3527.

[10] QingJie, W., & WenBin, W. (2017). Research on image retrieval using deep convolutional neural network combining L1 regularization and PRelu activation function. In IOP Conference Series: Earth and Environmental Science, 69(1), 012156. IOP Publishing.

[11] Hou, R., Chen, C., & Shah, M. (2017). Tube convolutional neural network (t-cnn) for action detection in videos. In Proceedings of the IEEE international conference on computer vision, 5822-5831.

[12] Zhang, Y. D., Pan, C., Sun, J., & Tang, C. (2018). Multiple sclerosis identification by convolutional neural network with dropout and parametric ReLU. Journal of computational science, 28, 1-10.

[13] Zhong, Z., Sun, L., & Huo, Q. (2019). An anchor-free region proposal network for Faster R-CNN-based text detection approaches. International Journal on Document Analysis and Recognition (IJDAR), 22(3), 315-327.

[14] Dachasilaruk, S., Rangsanseir, Y., & Thitimashima, P. (1999). Application of multiscale edge detection to speckle reduction of SAR images. In Asian Conference on Remote Sensing (ACRS).

[15] Yaqiu, J., & Shiqing, W. (2001). An algorithm for ship wake detection from the SAR image using the Radon transform and morphological image processing. Journal of Systems Engineering and Electronics, 12(4), 7-12.

[16] Stuhr, B., & Brauer, J. (2019). Csnns: Unsupervised, backpropagation-free convolutional neural networks for representation learning. In 2019 18th IEEE International Conference on Machine Learning And Applications (ICMLA), 1613-1620.

[17] Ganin, Y., & Lempitsky, V. (2015). Unsupervised domain adaptation by backpropagation. In International conference on machine learning, 1180-1189.

[18] Wu, Y., Deng, L., Li, G., Zhu, J., & Shi, L. (2018). Spatio-temporal backpropagation for training high-performance spiking neural networks. Frontiers in neuroscience, 12, 331.

[19] Price, S. R., Price, S. R., Price, C. D., & Blount, C. B. (2018). Pre-screener for automatic detection of road damage in SAR imagery via advanced image processing techniques. In Pattern Recognition and Tracking XXIX, International Society for Optics and Photonics, 10649, 1064913.

[20] Chen Q. (2013). Research on real-time vehicle distance measurement method based on monocular vision, Degree Thesis of Wuhan University of Technology.