Research and Design of Computer Vision in Train Real-time Detection

Kai Sun[1,a], Shaobin Li[1,b], Yahan Yang[1,c], Xiaobin Di[1,d], Yu Song[1,e] and Ke Chen[1,f]

[1]School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing,China
a. 18125129@bjtu.edu.cn, b. shbli@bjtu.edu.cn, c. 18120280@bjtu.edu.cn
d. 18140007@bjtu.edu.cn, e. 17120268@bjtu.edu.cn, f. 17120211@bjtu.edu.cn

Keywords: Train detection, railway level crossings, YOLOv3DN, DenseNet, Darknet, real-time detection, video.

Abstract: In order to solve the current high cost of personnel and equipment, cumbersome process and difficult maintenance of traditional train detection methods at railway level crossings, this paper proposes the YOLOv3DN model, which adds the DenseNet network to the real-time target detection network yolov3 based on the Darknet architecture. It detects trains approaching and leaving the crossing in real time and provides early warning to ensure the safe passage of the crossing.The video samples of trains in various weather scenarios were collected at multiple sites near Beijing South Railway Station and Beijing Railway Station.And the network was trained and verified, which proved the algorithm's good real-time performance, accuracy and robustness. The accuracy reached 98.7%, the recall rate reached 96.86%, and the fps> 40, which can meet the needs of practical applications, and has great significance and broad prospects.

1. Introduction

In recent years, object detection has been widely used in the fields of face recognition and car recognition, but it has not been applied to train detection.At present, the train detection method is completed through a series of operations such as track circuit detection of the train, and more than three times of telephone reporting between the train driver and the railway crossing watchman.Therefore,Traditional train detection methods have many disadvantages: high cost of personnel and equipment,cumbersome process,difficult to maintain and popularize. In contrast, computer vision is a better way to detect trains approaching and leaving at railway level crossing. This article breaks the tradition and designs the YOLOv3DN algorithm on the basis of the yolov3 algorithm, which is known for its real-time nature. Through processing and analyzing the video information collected by the camera, then judge whether to send the control signal to the railroad crossing, audio and signal lights at the railway level crossing to give early warning, so as to ensure the efficient and safe passage at the railway level crossing.

In order to operate safely and reliably, many devices have "Hot Standby", such as: two sets of cameras are set up at each collection point; solar panels are used as the backup power for the cameras; 4G network cards are set up for network backup; Set the emergency control button of the control terminal.

The main work and innovation of this article: 1. Collect train samples under various scenarios to enhance the quality, representativeness and generalization of samples; 2. Improve the quality and quantity of samples by data enhancement methods such as photometric transformation, contrast transformation and image rotation; 3. Use DenseNet network to improve detection speed and accuracy;4. Breaking the tradition and adopting deep learning target detection technology to solve the many shortcomings of traditional detection of trains approaching crossings. The verification on a large number of multi scene videos proves the effectiveness and practicability of this method.

2. Key Technologies

2.1. YOLO Algorithm

YOLOv3 uses a regression idea, which is an end-to-end one-stage detection algorithm, known for its real-time performance.It uses darknet53 to extract features (including a series of 53 convolutional layers of (3x3, 1x1), introduces a residual network, and replaces softmax with a logistic regression, so it solves the problem of gradient disappearance and improves the classification accuracy.

First, the input image is resized to 416x416, and features are extracted through the darknet network. The FPN network is input, and the K-means algorithm is used for cluster analysis to extract feature maps at three different scales (13x13, 26x26, 52x52).Then the feature map is divided into corresponding grid regions, and each grid predicts 3 bounding boxes, so a total of (13x13 + 26x26 + 52x52) x3 = 10647 bounding boxes are generated. Each bounding box predicts 4 coordinates t_x, t_w, t_h, t_y ·As shown in Figure 1.

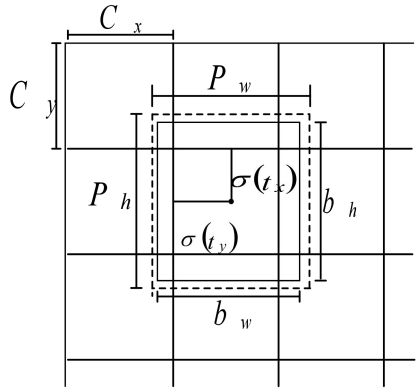

Figure 1: Regression target graph.

The distance from the target grid to the upper left corner is (c_x, c_y). The width and height of the corresponding bounding box are P_w P_h.The relationship is shown in(1).

$$b_x = \sigma(t_x) + c_x, b_y = \sigma(t_y) + c_y, b_w = p_w e^{tw}, b_h = p_h e^{th} \quad (1)$$

Each grid must predict the probability value of the object in the prediction frame, and score the prediction frame by (2)

$$\text{Conf}(\text{Object}) = \Pr(\text{Object}) \times \text{IOU}_{\text{pred}}^{\text{truth}} \quad (2)$$

among them, IOU_{pred}^{truth} is the intersection ratio of the prediction frame and the ground truth, and $Conf(Object)$ is the confidence degree. If there are targets in the grid, $Pr(Object) = 1$, otherwise it is 0.

$area(box_{pred} \cap box_{truth})$ represents the intersection area of the real target bounding box and the prediction bounding box, and $area(box_{pred} \cup box_{truth})$ represents the area of the union of the real target frame and the prediction frame. The calculation is shown in (3):

$$IOU_{pred}^{truth} = \frac{area\left(box_{pred} \cap box_{truth}\right)}{area\left(box_{pred} \cup box_{truth}\right)} \tag{3}$$

the confidence $Confidence(B)$ of a category B is calculated as (4).

$$Confidence(B) = Pr(class \mid object) \times IOU_{pred}^{truth} \times Pr(object) = Pr(class_B) \times IOU_{pred}^{truth} \tag{4}$$

the loss function is (5).

$$loss = l_{coordinate} + l_{confidence} + l_{class}$$

$$= \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} \left[\left(x_i - \hat{x}_i \right)^2 + \left(y_i - \hat{y}_i \right)^2 + \left(\sqrt{\omega_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] + \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} \left(C_i - \hat{C}_i \right)^2 + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{noobj} \left(C_i - \hat{C}_i \right)^2 +$$

$$\sum_{i=0}^{s^2} I_{ij}^{obj} \sum_{c \in classes} \left(p_i(c) - \hat{p}_i(c) \right)^2 \tag{5}$$

In the formula, $l_{coordinate}$ is the coordinate error, I_{ij}^{obj} is whether the j-th prediction target frame of the i-th grid is responsible for identifying the target, $l_{confidence}$ is a confidence error, and I_{ij}^{noobj} is the other prediction target frame in the i-th grid that is not responsible for identifying.

2.2. Improve YOLOv3

Because YOLOv3 directly trains the entire image, it does not use candidate regions or sliding windows, sacrificing accuracy for training speed. However, the core idea of the DenseNet network is skip connection. Some inputs directly enter the subsequent layer to achieve the integration of information flow, avoid information loss and gradient disappearance during layer transfer, strengthen feature connections between layers, and improve prediction accuracy.

It can be seen from Figure 2 that each layer has direct access to the gradients from the loss function and the original input signal, leading to an implicit deep supervision.This connection makes the transfer of features and gradients more efficient, and the network is easier to train.Therefore, it can reduce the disappearance or explosion of the gradient.

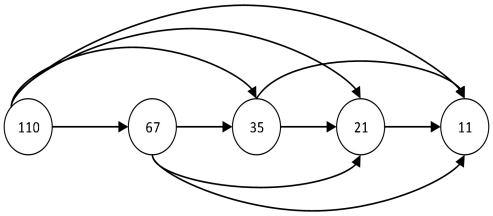

Figure 2: Improved YOLOv3DN's network structure.

$$x_\ell = H_\ell(x_{\ell-1}) + x_{\ell-1} \tag{6}$$

(6) is the ResNet formula of YOLOv3. Here ℓ represents the layer, x_ℓ represents the output of ℓ layers, and H_ℓ represents a non-linear transformation. So for ResNet, the output of ℓ layers is the output of ℓ-1 layers plus a non-linear transformation of the output of ℓ-1 layers.

$$x_\ell = H_\ell([x_0, x_1, \cdots, x_{\ell-1}]) \tag{7}$$

(7) is the formula of DenseNet. H_ℓ includes BN, ReLU and 3 * 3 convolution. $[x_0, x_1, \cdots, x_{\ell-1}]$ means concatenation of the output feature map of layers 0 to ℓ-1.Concatenation is the merging of channels. Like Inception, channel compression is performed by using a 1x1 convolution kernel. This can reduce the number of channels and thus reduce the amount of calculation,and each layer has direct access to the gradients from the loss function and the original input signal, reduce the disappearance or explosion of the gradient.However, in the previous Resnet, the pixel values were added, and the number of channels was constant. The overall detection process is shown in Figure 3.

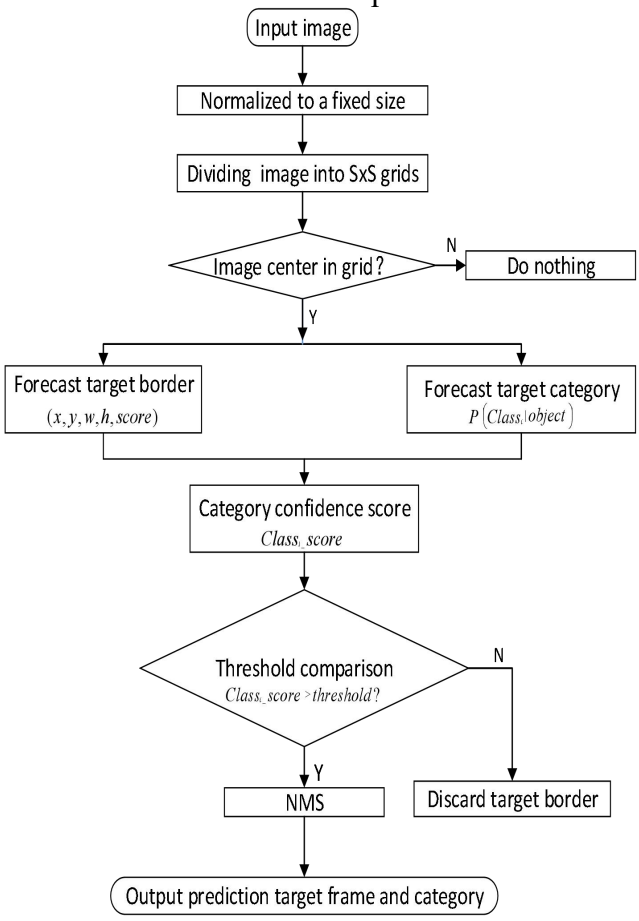

Figure 3: Detection flow chart of YOLOv3DN.

3. Solution

3.1. Information Collection Part

At the departure point and the approach point of 3-4km from the crossing, HIKVISION's DX-2CD3T25D-i3 camera is used to collect video train information in real time. Considering safety and reliability, two groups of cameras are used for Hot Standby.

3.2. Video Transmission Part

The collected information is transmitted to the host computer at the crossing through the wireless network for decoding and analysis. For the same safety and reliability, a network cable or a 4G network card is used to connect the camera for safe and reliable information transmission.

3.3. Host Computer Part

The main work of the host computer is to accept the video information transmitted from the camera, decode it, and process it through the YOLOv3DN algorithm to determine whether a train is approaching the crossing approach point or is leaving the crossing. If detected, a control command is sent to the 0control terminal at the crossing; otherwise, no control command is sent.

3.4. Terminal Part

The main equipment includes signal lamp, sound, crossing barrier, receiving the command from the host computer.If it is found that the train is approaching the crossing approach point, then the signal light will turn red, the sound will sound an alarm, the crossing barrier will automatically fall or slide out and lock the crossing to prevent pedestrian traffic.If the train leaves the crossing, after a delay of 2s, the signal light will turn green, the sound will turn off the alarm, the crossing barrier will automatically lift or return, and the crossing will be opened.Similarly, for the reliable and safe operation of railway level crossings, a set of buttons are set in the crossing room. In an emergency, people can operate the buttons to give priority to the control of the terminal.

3.5. Power Supply System Units

In order to operate safely and reliably, 2 shares of electricity are redundantly backed up.After the power failure occurs, the power can be switched to the solar panel battery immediately, the upper computer will be alarmed and the relevant personnel will be notified for maintenance.

3.6. System Framework

At both ends of the 3-4km distance from the crossing, approach points in both directions of the train going up and down are set, and train departure points are set up at the crossing. The camera collects video information in real time and transmits it back to the host computer for real-time detection via the network. Once the train is approaching or leaving the crossing, a control command is issued to the control terminal immediately. The overall system framework is shown in Figure 4:

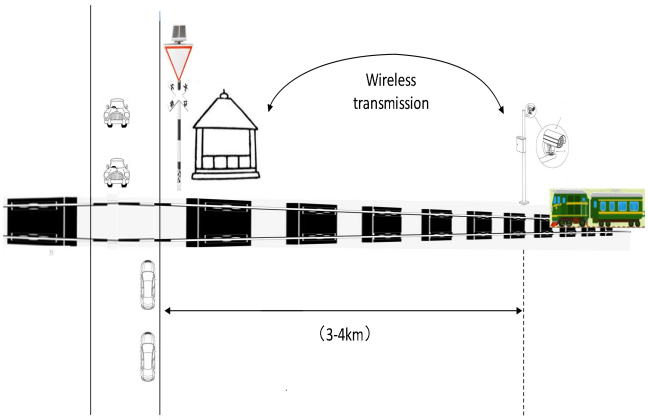

Figure 4: Overall system framework.

4. Experimental Environment and Data Set

4.1. Data Collection

The collection locations were near Beijing South Railway Station and Beijing Railway Station. Use HIKVISION's DS-2CD3T25D-i3 camera (50Hz, 25fps) and Nikon D7100 to collect train video samples from multiple weather scenarios.

The data set includes sunny, cloudy, rainy, snowy, night, day, evening and other scenes. There are three sources: 1) video data collected near Beijing south station and Beijing station; 2) train samples in PASAL VOC and coco data set; 3) visible pictures and video samples crawled by web crawlers. The pixels are between 720x1080 and 1080x3020.An example of a train data set for different weather and regions is shown in Figure 5.

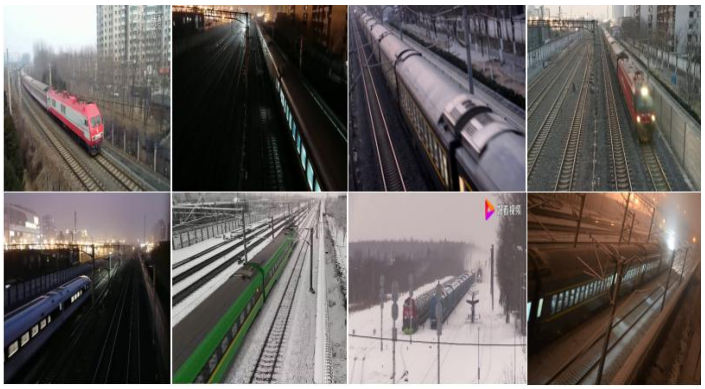

Figure 5: Train dataset example.

4.2. Data Preprocessing

There are mainly three parts: format conversion, normalization and data enhancement.

Format conversion: The video format is converted and framed by the VideoStudio Corel VideoStudio software.

Normalization: To unify the data set, a script is used to unify the naming format, and the resolution ratio is unified to 416x416.

Data enhancement: increase the brightness and contrast of the image, and flip image horizontally, so as to expand the data set.

4.3. Data Screening

The "redundant" pictures produced after video framing, background pictures without targets and some too fuzzy samples, which have no practical significance, even affect the training results, all of them should be deleted.

4.4. Data Annotation

Using LabelImage tool to label samples, there is only one class in the dataset: train. After labeling, it will generates an xml file marked with the target coordinates, name and sample path. Data set consists of training set and verification set.Put the dataset and xml file in the corresponding folder of YOLOv3DN, and then start training. The labimage working interface is shown in Figure 6.

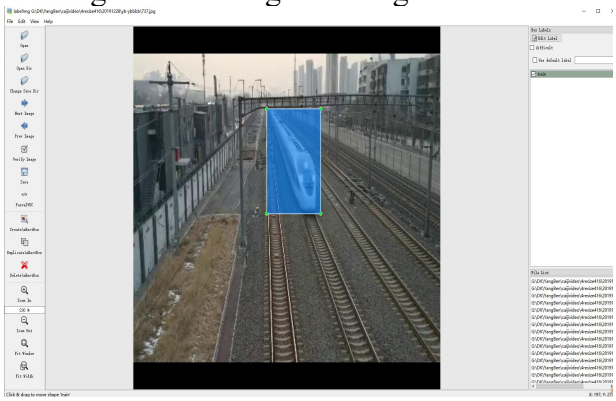

Figure 6: LabelImage working interface.

4.5. Experimental Environment

The experimental environment is shown in Table 1 :

Table 1: Experimental environment configuration.

Hardware environment	CPU	Intel i7 8700K
	Graphics card	GTX1080（8G/Nvidia）
	Memory	SamsungDDR5L 16GB 1600MHz
	SSD	Samsung 256GB
	HDD	WesternDitial 1TB
Software environment	Operating system	Ubuntu18.04
	Deep learning framework	Darknet
	Visualization tools	Opencv
	Programming IDE	Pycharm 19.1.3
	sample labeling software	LabelImage 1.8.1
	Other software environments	Anaconda4.5,CUDA 4.5,CUDNN 7.4.2,Opencv-python4.1.2, Python 3.7.5.

5. Analysis of Experimental Results

5.1. Analysis of Training Results

Use GPU to accelerate training, and dynamically adjust the learning rate according to the number of iterations. The initial learning rate is 0.001. When iterating to 3200 times, the learning rate is reduced by 10 times, that is 0.0001; when iterating to 3600 times, the learning rate is further reduced by 10 times, that is 0.00001. The main parameters are shown in Table 2.

Table 2: Main training parameters.

Parameter	Parameter value
batch	64
subdivisions	8
learning rate	0.001
max_batches	4000
momentum	0.9
decay	0.0005
lr_step	40
lr_factor	0.1
activation	leaky
angle, saturation, exposure, hue	0,1.5,1.5,0.1
steps	3200,3600

After continuous tuning and training, the training loss curve is shown in Figure 7, and the relationship between the average IOU and the number of training iterations is shown in Figure 8. It can be seen from the figure that the loss value tends to be stable after 2000 iterations, eventually reaching about 0.35, and the AVG IOU tends to be stable faster, close to 1.This means that the model has better training accuracy and faster convergence.

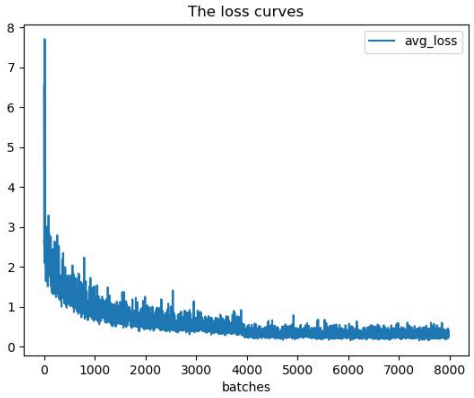

Figure 7: Loss curve.

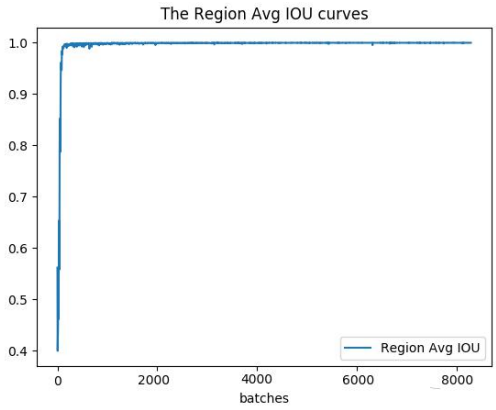

Figure 8: Avg IOU curve.

5.2. Accuracy Analysis

This paper uses the open source yolov3 to compare the precision, recall rate and time of different weather scenarios. The results are shown in Table 3 and Table 4.

It can be seen that the precision of YOLOv3DN is greatly improved, which can reach 98.7% in different scenarios. Recall can reach 96.86%, increased by 2.76%. And fps has also increased.When detecting the 1080x1920 resolution MP4 video, FPS > 40, which means that it can meet the requirements of real-time video detection.

Table 3: Comparison of network Precision.

	Day	Night	Evening	Early morning	Cloudy day	Snow day
YOLOv3	93.3	90.6	91.6	92.0	94.7	91.5
YOLOv3DN	98.7	96.8	97.5	97.3	98.5	96.4

Table 4: Test result.

	Total	Corrrect	Precision	Recall	Time/ms
YOLOv3	559	529	93.6	94.1	30
YOLOv3DN	559	541	98.7	96.86	24

5.3. Analysis of Video Test Results

In order to further verify the effectiveness of the model in real-time video detection, real-time detection and verification was performed on video samples of various scenes.The results show that each video can accurately detect the train. Although in some special test videos, trains with poor weather visibility and extremely small targets that have just entered the camera shooting range have not been identified. But as the train approaches the camera and the target becomes larger and clearer, it must be able to detect the train. In other words, the train must be detected throughout the video stream. This verifies the security, reliability and robustness of YOLOv3DN. The test results are shown in Figure 9.

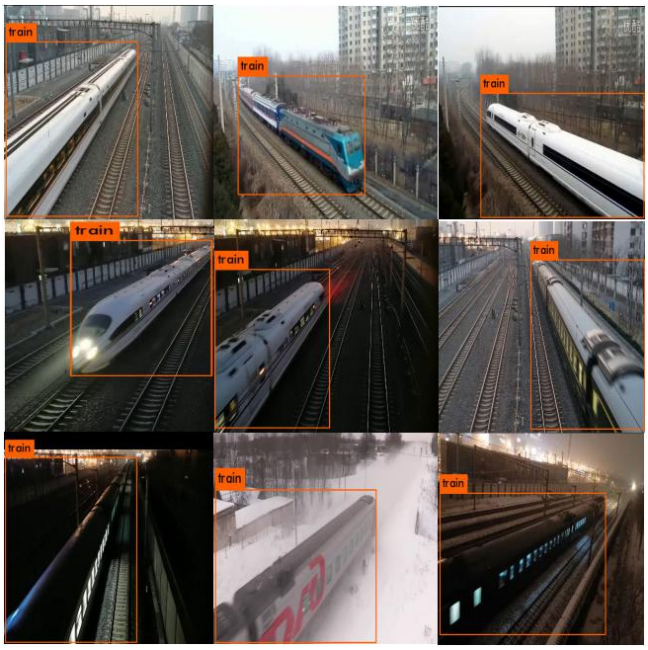

Figure 9: Video detection results.

6. Summary

In this paper, through the improvement of one-stge's algorithm of yolov3, a new algorithm of YOLOv3DN, which integrates DenseNet network, is proposed. Through the camera, the real-time and fast detection is carried out for the train approaching and leaving the crossing, and then the railroad crossing, audio, signal lights and other terminals are controlled to ensure the safe and reliable passage of the railway level crossing.

Collect the train samples of various scenes, and mark the train target training set and test set.The experimental results show that compared with the traditional detection method, YOLOv3DN proposed in this paper has the advantages of low cost of capital and personnel, simple and fast process, easy maintenance and upgrading, more security and reliability; compared with yolov3, the calculation amount is reduced, the detection speed and accuracy are improved, and the robustness is better. The precision and recall rate are 98.7% and 96.86%. However, in the scene where the visibility is very poor in special weather and the train target just enters the camera shooting range is very small, the detection ability is not very good. The following work will research and optimize the dataset, snow removal algorithm, rain removal algorithm and fog removal algorithm.

References

[1] N S Artamonov,N S Artamonov, P Y Yakimov.Towards Real-Time Traffic Sign Recognition via YOLO on a Mobile GPU[J].Journal of Physics: Conference series,2018, 1096(1): 012086 (8pp). DOI:10.1088/1742-6596/1096/1/012086.

[2] R. A. Hounsell.Variable Stellar Object Detection and Light Curves from the Solar Mass Ejection Imager (SMEI) [J]., 2011, 7(S285): 91-94. I:10.1017/S1743921312000312.

[3] Xing Liu,Xing Liu,Jian Chen, et al.Research on Simulation System of Small Unmanned Air-to-Ground Object Detection Platform[J].Journal of Physics: Conference Series,2018, 1087(6):062053 (7pp). DOI:10.1088/1742-6596/1087/6/062053.

[4] Hou, Saihui,Wu, Feng,Wang, Zilei.Object detection via deeply exploiting depth information[J]. Neurocomputing, 2018, 286 (Apr.19): 58-66.

[5] Abolfazl Saghafi,Sajad Jazayeri,Sanaz Esmaeili, et al.Real‐time object detection using power spectral density of ground‐penetrating radar data[J].Structural Control and Health Monitoring, 2019,26(6):n/a-n/a. DOI:10.1002/stc.2354.

[6] Xiaobo Zhang,Yan Yang,Tianrui Li, et al.Discovering Senile Dementia from Brain MRI Using Ra-DenseNet[C].The 23rd Pacific-Asia Conference on Knowledge Discovery and Data Mining.2019: 449-460.

[7] Juan Du,Juan Du.Understanding of Object Detection Based on CNN [J].Journal of Physics: Conference Series,2018,1004(1):012029 (8pp). DOI: 10.1088/1742-6596/ 1004/1/012029.

[8] Feng Yuxu, Li Yumei.Review of Deep Learning Optimizer Methods and Learning Rate Attenuation Methods [J] .Data Mining,2018,8(04): 186-200. DOI:10.12677/HJDM.2018.84020.

[9] Qicheng Lao,Thomas Fevens.Cell Phenotype Classification Using Deep Residual Network and Its Variants[J].International Journal of Pattern Recognition and Artificial Intelligence, 2019, 33(11). DOI: 10.1142/S0218001419400172.

[10] Prathamesh M Kulkarni,Prathamesh M Kulkarni,Zhengdong Xiao, et al.A deep learning approach for real-time detection of sleep spindles[J].Journal of Neural Engineering, 2019, 16(3): 036004 (19pp). DOI:10.1088/1741-2552/ab0933.