Application of New Generation Artificial Intelligence in Traffic Informatization

Jinling Hou*
Department of Transportation of Shanxi Province, Taiyuan 030001, China

*Corresponding author e-mail: houjingjing@bitzh.edu.cn

Abstract. With the rapid development of economy, continuous improvement of urbanization level, increasing number of private cars and increasingly prominent urban traffic problems, intelligent transportation has become a research hotspot at home and abroad. As a new generation of high and new technology, artificial intelligence technology plays a decisive role in the development of China's transportation. On the basis of studying the level of China's traffic informatization, this paper uses the YOLO algorithm model and selects the CCTSDB data set to study the traffic sign recognition in traffic informatization, and verifies the accuracy of the algorithm through experimental results, and changes the traditional driving mode, improve the level of informatization, and reduce the occurrence of traffic accidents.

Keywords: Artificial Intelligence, Transportation, YOLO, Recognition

1. Introduction
In recent years, people’s living standards have continued to improve, and the demand for private cars has increased. According to statistics, the number of cars in China has exceeded 200 million, and the number of motor vehicles has exceeded 300 million. The increase in the number of motor vehicles has led to an increase in traffic accidents in China. According to the statistics of China's Ministry of Transport, the number of traffic accidents in 2019 increased by 600,000 compared with 2018, and the total number exceeded 8.6 million, causing a large number of casualties [1-2]. Intelligent traffic assistance system can use artificial intelligence technology and big data technology to realize real-time identification of vehicles and traffic signs on the road, give early warning to potential traffic information, reduce unnecessary casualties, improve travel safety, and help to reduce traffic accidents [3].

2. Related Research Based on Traffic Information Detection Algorithm
2.1. Faster RCNN
Faster RCNN is composed of Faster RCNN and RPN. RPN includes two parts: reference frame extraction and reference frame processing. The extraction window has three sizes of 128, 256, 512, and the vector corresponding to each reference frame needs to be calculated. The classification layer judges whether the object exists, and calculates the corresponding confidence for all reference frames,
and arranges them in descending order\cite{4}. Faster RCNN first preprocesses the detection object in the recognition process, and then performs convolution and pooling\cite{5}. The structure is shown in Figure 1.

![Figure 1. Structure of Faster RCNN](image)

2.2. SSD
Both SSD and YOLO belong to single-stage detection algorithms. SSD belongs to multi-frame prediction, which can detect multiple categories. It uses VGG-16 basic network for feature extraction and uses forward propagation. The network structure is shown in Figure 2. Assuming that the number of bounding boxes is 1, the number of category scores is c, the position offset of each bounding box is 4, the number of feature maps that need to be filtered for each position of the feature map is \((c+4)\times l\), and the size of the feature map is \(m\times n\), and the output result is \((c+4)\times l\times m\times n\).

The default box expression is shown in formula (1), \(m\) represents the number of feature layers, \(S_{\text{max}}\) represents the default box of the highest layer to be detected, and \(S_{\text{min}}\) represents the default box size of the lowest layer to be detected

\[
S_i = S_{\text{min}} + \frac{S_{\text{max}} - S_{\text{min}}}{m-1} (i-1), i \in [1, m] 
\]

![Figure 2. Structure of SSD](image)

2.3. YOLOv3
The YOLO algorithm is a single-stage algorithm model. It uses the network model to classify and locate the input image, and converts traffic information detection into a regression problem. The YOLO algorithm model only uses one network model, so it is superior to other network models in detection efficiency, and the detection speed is increased by about 20 times. YOLO is divided into YOLOv1, YOLOv2, YOLOv3. YOLOv1 uses the classic CNN form, YOLOv2 regulates the detection speed and accuracy. YOLOv3 increases the number of layers of the neural network based on the YOLOv2 points. For deeper extraction of object features, Darknet19 is changed to Darknet53. Under the same circumstances, the speed is three times faster than SSD, and the YOLO algorithm can see the complete image during training and testing, and can learn the generalized representation of the target.
Therefore, YOLOv3 is used in this study to conduct real-time identification of traffic signs in traffic information. The YOLOv3 network structure is shown in Figure 3.

Figure 3. Structure of YOLOv3

3. Application of Artificial Intelligence in Traffic Informatization

3.1. Construction of YOLOv3 Model

On the basis of YOLOv2, YOLOv3 uses the Darknet53 network and uses the logistic regression classifier to extract the last three feature maps, and use the up-sampling method to enlarge each feature map to a large feature map size, perform individual predictions, and finally stitch together Small feature map and large feature map complete the prediction. YOLOv3 greatly increases the number of convolutional layers and pooling layers, the number of pooling layers is 5, and the number of convolutional layers is 53. Table 1 shows the performance comparison of feature extraction networks [6].

Table 1. Comparison of feature extraction network performance

| Backbone        | Top-1/% | Top-5/% | Bn Ops/10⁹ | BFL OP/s | Frame Rate/(fps) |
|-----------------|---------|---------|-------------|----------|-----------------|
| Darknet-19      | 74.1    | 91.8    | 7.29        | 1246     | 61              |
| Res Net-101     | 77.1    | 93.7    | 19.7        | 1039     | 51              |
| Res Net-152     | 77.6    | 93.8    | 29.4        | 1090     | 36              |
| Darknet-53      | 77.2    | 93.8    | 18.7        | 1457     | 77              |

The detection process of YOLO model is as follows: the input image is divided into S×S cells. When the center of the detection object falls into any cell, the cell is responsible for detecting the image, and each cell predicts the confidence value of the boundary box and the boundary box. If there is no target, the confidence value is 0. As shown in formula (2), \( P(O_{object}) = 1 \) means that the detected image exists in the prediction frame, otherwise it is 0. \( P(C_i \mid O_{object}) = 0 \) means that the cell predicts the confidence score of the i category.

\[
S_{conf} = P(C_i \mid O_{object}) \times P(O_{object}) \times I(truth,pred)
\]  

(2)

3.2. YOLOv3 Model Training

(1) Calculation of loss function

In the training of YOLOv3 model, the loss function of mean square error (including coordinate error, confidence error and classification error) is used to analyze the contribution rate and relative error of the loss and calculate the loss in the training process. The calculation formula is as shown in (3). The first two lines of the loss function represent the calculation of the coordinate error of the
bounding box, which is mainly the balance of the medium and small targets in the process of balance detection and prediction. The third line of formula (3) is to calculate the confidence error; The last line of formula (3) is the classification error, which is calculated only when the cell covers the image. \(I_{ij}^{obj}\) represents the center point of the image contained in the cell, \(\lambda_{noobj}\) represents the loss value of the image target not detected in the cell, so as to avoid instability of the model caused by large gradient crossing, and \(\lambda_{coord}\) represents the loss value of the predicted position of the boundary box.

\[
\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} [(x_i - x_i^*)^2 + (y_i - y_i^*)^2] + \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} [(\sqrt{h_i} - \sqrt{h_i^*})^2 + (\sqrt{w_i} - \sqrt{w_i^*})^2] \\
+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} (C_i - C_i^*)^2 + \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} I_{ij}^{obj} (C_i - C_i^*)^2 \\
+ \sum_{c=0}^{class} \sum_{i=0}^{S^2} I_{ij}^{obj} (p_i(c) - p_i^*(c))^2
\]

(3) Model Training

Setting of training parameters: The number of training samples is 7,500, the number of verification samples is 2,500, the ratio of training samples to verification samples is 3:1, and the model parameters are set as shown in Table 2.

Table 2. Setting of YOLOv3 model training parameters

| Parameters           | Values | Explanation                                      |
|----------------------|--------|--------------------------------------------------|
| activation           | leaky  | The activation function uses Leakyrelu          |
| Batch                | 64     | The number of training samples is updated every 64 times |
| decay                | 0.0005 | To prevent overfitting, the weight decays the regular term |
| Subdivision          | 16     | Number of samples sent to the trainer at one time |
| Learning_rate        | 0.001  | The learning rate controls the speed of weight updates |
| Batch_normalize      | 1      | To prevent over-fitting, add batch normalization layer at the input of each layer of convolution |
| Max batches          | 300000 | Stop learning after reaching max batches         |

3.3. Experimental Results and Analysis

(1) Loss function training results

The loss function curve of the YOLOv3 model is shown in Figure 4. At the beginning of training, the loss function drops rapidly. When it reaches 3 epochs (500 batches), the loss function value drops slowly. After 11 epochs (1750 batches), the loss value fluctuates slightly. The situation tends to be stable, and the training of the YOLOv3 model ends at this time.
Experiment and analysis of YOLOv3 model in traffic information recognition

After the training of the YOLOv3 model, save the trained model parameters. This study uses the CCTSDB data set to test three types of pictures: mandatory, warning, and prohibitory, and select the three environments of sunny, rainy, and driving to identify traffic signs. As shown in Figure 5(a) and 5(b), in sunny environment, the recognition accuracy reached 100% and the recognition time was 0.0327s, which completely met the expected target. As shown in Figure 6 (a) and 6 (b), traffic indicator signs can be identified in rainy environment. The identification rate on the left side is 95% and that on the right side is 85%, and the identification time is 0.0333s. Figures 7(a) and 7(b) are tests in a traffic driving environment. Although there are 5 types of traffic signs in the picture, the YOLOv3 model only trains one of them. Therefore, it can be judged only by accurately identifying the type. The accuracy of the model, from the experimental results, the model recognition rate is 100%, and the recognition time is 0.0333, which can verify the anti-interference ability of the YOLOv3 model.

**Figure 4.** Curve of YOLOv3 loss function

**Figure 5(a).** Recognition of traffic signs in sunny environment

**Figure 5(b).** Recognition of traffic signs in sunny environment
4. Conclusion

With the development of intelligent driving, traffic information recognition has become one of the important components of vehicle intelligence technology. The effect of traffic information recognition is greatly affected by the real environment, such as light, occlusion, rain, etc. This research uses the YOLOv3 model to identify traffic signs, complete the informatization research of traffic signs, and promote the application process of China's new generation of artificial intelligence technology in traffic informatization.

References

[1] Om K, Boukoros S, Nugaliyadde A, et al. Modelling email traffic workloads with RNN and LSTM models[J]. Human-centric Computing and Information Sciences, 2020, 10(1).

[2] Song Y, Lu J. RNN-based traffic flow prediction for dynamic reversible lane control decision[C]// Conference on Data Science and Knowledge Engineering for Sensing Decision Support (FLINS 2018). 2018.

[3] Zhene Z, Hao P, Lin L, et al. Deep Convolutional Mesh RNN for Urban Traffic Passenger Flows Prediction[C]// 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI). IEEE, 2018.

[4] Guo J, Liu Y, Yang Q K, et al. GPS-based citywide traffic congestion forecasting using CNN-RNN and C3D hybrid model[J]. Transportmetrica A Transport Science, 2020(4):1-24.

[5] B S D A, A K H. Developing a Twitter-based traffic event detection model using deep learning architectures[J]. Expert Systems with Applications, 2019, 118:425-439.

[6] Lv Z, Xu J, Zheng K, et al. LC-RNN: A Deep Learning Model for Traffic Speed Prediction[C]// Twenty-Seventh International Joint Conference on Artificial Intelligence IJCAI-18. 2018.