Research on Intelligent Security Video Image Analysis Based on Deep Learning

Ling Cheng*
Jiangxi police institute teaching & research section of safety precaution engineering, China, 330100

*Corresponding author e-mail: Lingcheng@163.com

Abstract. For video surveillance analysis, target detection algorithms are particularly important. Aiming at the problem of target detection in road traffic scenes, this paper proposes a non-motor vehicle target detection method that uses EdgeBoxes algorithm and deep learning fusion. It is inspired by deep learning target classification algorithm Fast R-CNN, and combines non-motor vehicle data samples in VOC format to integrate road traffic scenes. We use the EdgeBoxes algorithm to extract the target of the sample. It is recommended to construct an appropriate amount of region of interest, and inputs the network for iterative training with the sample. In the target detection of road traffic scenes, compared with the traditional method based on the fusion of the basic EdgeBoxes algorithm and the optimized Fast R-CNN method, the detection accuracy is slightly improved, the calculation is significantly reduced, and the algorithm has advantages in time complexity.

Keywords: Target Detection, Video Surveillance, Deep Learning, Fast R-CNN

1. Introduction
In recent years, the bicycle-sharing and take-away industries have developed rapidly, which has brought convenience to people's travel and diet. As a result, non-motorized vehicles such as bicycles on the road have proliferated, and road traffic safety issues have become more and more serious[3]. Due to their small targets and flexible mobility, traffic supervision faces huge challenges. Large motor vehicle detection, as a key technology for the construction of video surveillance of traffic conditions, has attracted widespread attention from researchers at home and abroad for a long time, and has also achieved good research results; Relatively little research. Object detection is an important branch in the field of image processing and computer vision. This paper proposes an object detection method based on the fusion of EdgeBoxes and deep learning. The EdgeBoxes algorithm for extracting target suggestions is combined with the deep learning algorithm Fast R-CNN, and the Fast R-CNN network structure is optimized and applied to non-motor vehicle target detection. OP is first extracted from the sample image, and then input into the Fast R-CNN network with the sample image for iterative training to obtain a non-motor vehicle target detection model. In terms of the OP extraction stage, the EdgeBoxes algorithm can guarantee a relatively high average recall (AR) [1] when extracting a small number, so the complexity of network training is reduced, and the accuracy of target detection is
guaranteed while improving classification speed. Experiments prove that the target detection method is effective and feasible[6].

2. Object detection model based on EdgeBoxes and Fast R-CNN fusion

2.1. Network structure
In order to reduce the complexity of the RoI pooling layer in the Fast R-CNN algorithm model, the EdgeBoxes algorithm is introduced. First, the algorithm is used to extract the sample images by OP[5], and then the related calculations are used to obtain RoIs. The target task studied in this paper is a small target such as a bicycle, so the small network structure Caffenet[2] in the Fast R-CNN algorithm model is selected. The main part of the network model is made up of five convolutional layers, 1 RoI pooling layer, and 2 fully connected layers. It is finally divided into two parallel fully connected layers for multi-task loss output calculation to obtain the target detection result.

Target labelling is a more complicated work in the early stage. First, the EdgeBoxes algorithm is used to perform OP extraction on the sampled image of the target label. Using the sliding window extraction strategy[4], the search results are controlled by the following parameters: $\alpha$ controls the search step, $\beta$ controls the IoU threshold, $\delta$ controls the search accuracy, minScore controls the minimum score and maxBoxes controls the maximum magnitude of the extracted OP. Then RoIs is obtained after a certain calculation. The sample image is input into the network to extract the feature map. The convolutional feature map and RoIs are input to the RoI pooling layer, which converts the features of any valid RoIs into a fixed size H×W (Eg 7×7) smaller normalized feature maps, where H and W are layer hyperparameters, independent of any particular RoI. Each RoI coordinate is defined by a quaternion (r, c, h, w), which represents the top left corner of the RoI (r, c) and its height and width (h, w). The RoI pooling layer divides the H×W RoI window into H×W sub-windows, and obtains about $h / H \times w / W$ sub-windows. The pooling layer is used to calculate each window to obtain the target feature matrix. Then the fully connected layer is input to calculate the feature vector. Finally, it enters two parallel fully connected layers and performs regression calculation to obtain the network output. The existence of the RoI pooling layer makes the size of the input image in the training process break through the limit of a single specification. The RoI pooling layer pools RoI feature vectors of different sizes into a unified specification.

![Network structure of the algorithm](image)

**Figure 1.** Network structure of the algorithm

2.2. Model training
The network model has two output layers that compute the target classification results and the target detection frame coordinate results, respectively. The first output layer uses the softmax algorithm regression calculation to obtain the probability P of each RoI in the 2 categories; the second output layer is responsible for calculating the coordinate values of the 2 types of target detection frames.
Among them, a regularization term $\frac{1}{n} \sum |\omega|$ is added to the first loss function to reduce the complexity of the network and prevent overfitting. Therefore, a multi-task loss function is used to perform regression calculation on the type of RoI and detection frame coordinates of each marker:

$$L(p,u,t^*,v) = L_{\text{cls}}(p,u) + \lambda_u [u >= 1)L_{\text{loc}}(t^*,v)$$  \hspace{1cm} (1)

$$L_{\text{cls}}(p,u) = \ln P_u + \frac{1}{n} \sum |\omega| = L_0 + \frac{1}{n} \sum |\omega|$$  \hspace{1cm} (2)

The working principle of regularization, first finds the partial derivative of equal (2), and get:

$$\frac{\partial L_{\text{cls}}}{\partial \omega} = \frac{\partial L_0}{\partial \omega} + \frac{1}{n} \text{sgn}(\omega)$$  \hspace{1cm} (3)

The above formula $\text{sgn}(\omega)$ represents the sign of $\omega$, then the update rule of weight $\omega$ is:

$$\omega \rightarrow \omega - \eta \frac{\partial L_0}{\partial \omega} - \frac{n}{\eta} \text{sgn}(\omega)$$  \hspace{1cm} (4)

Compared to the original update rule, $\frac{n}{\eta} \text{sgn}(\omega)$ is added. When $\omega$ is positive, the updated $\omega$ becomes smaller; when $\omega$ is negative, the updated $\omega$ becomes larger. Therefore, the effect of regularization is to make $\omega$ closer to 0, so that the weight in the network is 0 as much as possible, which is equivalent to reducing the network complexity and preventing over-coupling. Set the hyperparameter $\lambda_u = 1$ and the electric vehicle class $u = 2$, $L_{\text{cls}}$ is the log loss of the type probability (Pu), where Pu is calculated using softmax.

Another task loss, $L_{\text{loc}}$, is the loss of the detection frame. Through the real $u$-type detection frame coordinate value $v = (vx, vy, vw, vh)$, and the predicted coordinate value of the $u$-type $t^u = (t^u_x, t^u_y, t^u_w, t^u_h)$. Since the background class does not have the original markup box, the $L_{\text{loc}}$ of the background class can be ignored. For detection box regression loss, use the loss function:

$$L_{\text{loc}}(t^*,v) = \sum_{x,y,w=h} S_c(t^*_x - v_x)$$  \hspace{1cm} (5)

$$S_c(x) = \begin{cases} 0.5x^2, & x < 1 \\ 0.5|x| - 0.5x, & |x| > 1 \end{cases}$$  \hspace{1cm} (6)

3. Experimental results and analysis

Dataset includes Images: 4,096 pieces / jpg, Labels: 4,096 pieces / xml, BoundingBox: 10,708 pieces / rectangles. The sample data set includes bicycles: 2,262 and evbike: 1,834.

In the single-sheet test phase, first, select some sample images for single-image detection, and set the detection type probability $P$ (calculated by equal (5) and (6)) to a threshold of 0.2 to 0.7 for multiple detections. The list analysis part of the sample is compared with the detection results of Fast R-CNN, SPP-Net, R-CNN and other network models, and the detection data of two sample graphs of each type of target task under the four network models are recorded, as shown in Table 1. evbike performs well under the four network models. The extraction algorithm used in this paper has controlled the number of OPs and achieved good detection results in a small number of cases. Table 1 shows the detection accuracy of the bicycle. The evbike model in this paper is slightly higher than the other three model models, but the detection time is greatly shortened.
Table 1. Bicycle test result table

| Methods  | Number of OPs | Time/s  | Accuracy |
|----------|---------------|---------|----------|
| Fast R-CNN | 8024          | 36.332  | 0.832    |
| SPP-Net  | 3703          | 17.289  | 0.754    |
| R-CNN    | 3703          | 239.125 | 0.795    |
| Ours     | 3608          | 16.008  | 0.845    |

Table 2. Ebike test result table

| Methods  | Number of OPs | Time/s  | Accuracy |
|----------|---------------|---------|----------|
| Fast R-CNN | 7300          | 36.124  | 0.802    |
| SPP-Net  | 2793          | 77.451  | 0.851    |
| R-CNN    | 2793          | 259.154 | 0.813    |
| Ours     | 2542          | 14.026  | 0.795    |

4. Conclusions
This method effectively controls the number of Object Proposals by introducing the EdgeBoxes algorithm, and speeds up the training of the target model. It also uses deep neural networks to avoid the problem of manually designing feature storage. By introducing regularization ideas on the network structure, the model is reduced. Sex training complexity prevents overfitting. Through experiments, the detection model improves the detection speed and obtains better results of non-motor vehicle target detection.

References
[1] Uijlings J R, Sande K E, Gevers T, et al. Selective search for object recognition[J]. International Journal of Computer Vision, 2013, 104(2): 154-171.
[2] Morrison G S. Admissibility of forensic voice comparison testimony in England and Wales[J]. Criminal Law Rev, 2018, 1.
[3] Meuwly D, Ramos D, Haraksim R A. A guideline for the validation of likelihood ratio methods used for forensic evidence evaluation[J]. Forensic Science International, 2017, 276: 142-153.
[4] Leegwater A J, Meuwly D, Sjerps M, et al. Performance study of a score-based likelihood ratio system for forensic fingerprint comparison[J]. Journal of Forensic Sciences, 2017, 62(3): 626-640.
[5] Zhang H, Liu L, Long Y, et al. Unsupervised Deep Hashing With Pseudo Labels for Scalable Image Retrieval[J]. IEEE Transactions on Image Processing, 2018, 27(4): 1626-1638.
[6] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017, 39(4): 640.