Hindawi
Complexity
Volume 2021, Article ID 6620425, 11 pages
https://doi.org/10.1155/2021/6620425

Research Article

Application Research of Key Frames Extraction Technology Combined with Optimized Faster R-CNN Algorithm in Traffic Video Analysis

Zhi-guang Jiang[1] **and Xiao-tian Shi**[2]

[1]*Hebei University of Science and Technology, Shijiazhuang 050000, China*
[2]*Shi Jiazhuang University of Applied Technology, Shijiazhuang 050081, China*

Correspondence should be addressed to Xiao-tian Shi; shixt@sjzpt.edu.cn

Received 13 December 2020; Revised 7 January 2021; Accepted 13 January 2021; Published 2 February 2021

Academic Editor: Abd E.I.-Baset Hassanien

The intelligent transportation system under the big data environment is the development direction of the future transportation system. It effectively integrates advanced information technology, data communication transmission technology, electronic sensing technology, control technology, and computer technology and applies them to the entire ground transportation management system to establish a real-time, accurate, and efficient comprehensive transportation management system that works on a large scale and in all directions. Intelligent video analysis is an important part of smart transportation. In order to improve the accuracy and time efficiency of video retrieval schemes and recognition schemes, this article firstly proposes a segmentation and key frame extraction method for video behavior recognition, using a multi-time scale dual-stream network to extract video features, improving the efficiency and efficiency of video behavior detection. On this basis, an improved algorithm for vehicle detection based on Faster R-CNN is proposed, and the Faster R-CNN network feature extraction layer is improved by using the principle of residual network, and a hole convolution is added to the network to filter out the redundant features of high-resolution video images to improve the problem of vehicle missed detection in the original algorithm. The experimental results show that the key frame extraction technology combined with the optimized Faster R-CNN algorithm model greatly improves the accuracy of detection and reduces the leakage. The detection rate is satisfactory.

1. Introduction

Intelligent transportation is based on smart transportation. It makes full use of the Internet of things, cloud computing, Internet, artificial intelligence, automatic control, mobile Internet, and other technologies in the transportation field and collects traffic information through high tech, which is effective for traffic management, transportation, and the public. All aspects of transportation, such as travel, and the entire process of transportation construction management are controlled and supported, so that the transportation system has the ability to sense, interconnect, analyze, predict, and control in regions, cities, and even larger time and space, so as to fully guarantee traffic safety and provide the efficiency of transportation infrastructure, the improvement of the operation efficiency and management level of the transportation system, the service of smooth public travel, and sustainable economic development.

From the 1970s to the 1980s, intelligent transportation was proposed as a concept, but it was limited by computing power and communication means, and its development speed was slow. ITS research was in the preparatory stage, so the research at this stage mainly focused on the core of the ITS system, vehicle navigation system, and route planning guidance. Since the end of the last century, with the great development of data transmission speed, computing power, and positioning technology, the development speed of intelligent transportation has been greatly increased. Some developed countries such as the United States, Japan, and Europe have turned their research perspectives to verify intelligence through the establishment of some large-scale projects, the transportation concept, and, based on this, comprehensive research and development of supporting basic technologies. The monitoring system is an important part of road traffic management. The video is captured by camera equipment installed on both sides or above the road to meet people's needs for real-time monitoring of traffic scenes. A large amount of road monitoring video equipment is built, and massive video data is accumulated in the traffic video system, which puts forward higher requirements for the storage capacity, transmission bandwidth, data analysis, and abnormal situation identification of the system. The continuous improvement of video processing capabilities and smart video recognition capabilities is the key technology to realize smart video. These data have the characteristics of huge capacity and large amount of information. Traditional traffic data processing methods, processing architectures, and smart video recognition algorithms have been gradually out of time, and they cannot meet the processing needs of intelligent transportation big data, especially smart video recognition. Instead, big data-related technologies are required to conduct in-depth mining and development of relevant data and adopt more advanced recognition methods to realize data sharing and integration, to achieve the purpose of intelligent services.

2. Research Status

In order to improve the accuracy of traffic smart video recognition, this paper proposes the use of key frame technology set combined with Faster R-CNN vehicle detection algorithm to judge vehicles. Most existing works use CNN as the feature extraction method of video. They divide video segments, extract one or more video frames from the segments as input to CNN, and then fuse the features of the segments. Early research took a single frame as the input of CNN. This approach makes insufficient use of time information. Some studies try to use CNN to extract video features directly. Tran et al. [1] trained a large-scale 3DCNN network. This type of approach increases the dimension of CNN at different levels, and the video sequence is directly used as the input of CNN. Limited by the size of the convolutional network, 3DCNN cannot handle videos of variable length and essentially still cannot avoid video segmentation recognition. Donahue et al. [2] proposed the LRCN structure, based on the use of CNN to extract the features of independent video frames, and introduced LSTM (a more effective RNN structure) to fuse the features extracted from each frame. The training time and storage cost of LSTM and 3DCNN are relatively large. In order to extract time information, Simonyan and Zisserman [3] proposed a dual-stream method. This method uses dense optical flow as an auxiliary input and uses two independent convolutional neural networks to extract the features of a single frame of original image and multiple frames of optical flow image and merge them at the final scoring level. The dual-stream method usually uses the network used by the image recognition task, and the calculation scale is equivalent to the image recognition task. The introduction of optical flow has significantly improved the accuracy of behavior recognition, and dual-stream networks have become the mainstream. Most of the subsequent researches improved deep neural networks on the basis of dual-stream networks or used various methods to fuse features based on dual-stream network features. The TSN (Temporal Segment Networks) proposed by Wang et al. [4] is a powerful improvement of dual-stream networks, and most of the existing methods use this as a measurement standard. This framework introduces a pretraining model of optical flow; at the same time, a training method of predividing video segments and using pooling fusion between segments is proposed. However, the TSN structure uses uniform segmentation, ignoring the difference in the amount of information between segments. Some people try to use the idea of key frames to improve the effect of behavior recognition. Hu and Zheng [5] used the optical flow difference method to extract the key frames in the video and achieved certain results in the KTH data set. In terms of deep learning algorithms, Girshick et al. [6] introduced the convolutional neural network to the target detection task for the first time, using the (Selective Search, SS) method to select candidate regions, and then using CNN to extract features and attach the classifier to the volume Perform detection on the product feature map and finally return to adjust the final position of the detection frame. Compared with the traditional algorithm, the average accuracy of this algorithm on the PASCAL VOC2012 test set (Mean Average Precision, MAP) is improved by 30%. In 2015, He et al. [7] proposed SPP net, which uses a spatial pyramid pooling layer to reduce the size limit of convolutional neural networks. Girshick [8] also proposed Fast-RCNN based on the idea of pyramid pooling in SPP net. This network uses a kind of ROI pooling to solve the problem that candidate boxes of different sizes cannot be input to the detection network with the same length and combine the candidate regions. If it is marked on the Feature map, only one feature extraction is required for the image, which greatly speeds up the operation of the network. Ren et al. [9] proposed the Faster R-CNN algorithm, which uses the RPN network to select candidate regions, which further reduces the running speed of the network and improves the detection accuracy.

3. Processing Architecture Based on Big Data

The continuous expansion of the construction of intelligent transportation video processing has resulted in massive heterogeneous data of different types and structures, such as system data, video data, and detection data, forming big data and traditional traffic data processing. The method and technical architecture can no longer meet the processing requirements of intelligent big data. Therefore, it is necessary to use big data-related technologies to conduct necessary mining and development of videos to achieve data sharing, processing, and integration to achieve the purpose of system processing requirements. This article refers to the sharing management method of massive data proposed by Wang et al. [10–12] and proposes related solutions in combination with virtual technology and distributed storage technology in cloud computing [13, 14].

3.1. Parallel Computing Model Design. In the traditional sense, cloud computing is divided into service modes, which can be divided into private cloud, public cloud, and hybrid cloud. According to the needs of the smart transportation video system, the article proposes a parallel computing mode, which is divided into two components: distributed file system (DFS) and distributed computing system (DCS); this computing mode has the following characteristics: (1) the client has the characteristics of flexible joining or evacuating; (2) since the application of each node is consistent, it can be based on the division of labor and different command files are configured for different tasks; (3) the deployment is simple, and the computing scale and storage scale can be controlled arbitrarily; (4) the model hides details of parallel computing, data distribution, load balancing, etc., and users can realize flexible computing according to actual needs and flexible processing; (5) the model has strong storage and computing capabilities and is fully adapted to data processing and data storage of smart video; and (6) the model can easily implement smart video algorithms such as convolutional neural networks [15].

The parallel computing model is shown in Figure 1.

3.2. Design of Distributed File Processing Model. The file distribution system is a network server component that makes it easier for users to query and manage data on the network. Distributed file system is a way to combine files distributed on different computers into a single name space and make it more convenient to establish a single, hierarchical multiple file server and server sharing work on the network, the traditional processing method. It is a "root node-node" model. Although the system architecture is greatly simplified, the core "root node" is responsible for the task of managing and accessing all "child nodes." The network pressure and computational pressure are huge. If a failure occurs, the entire system will be paralyzed, as shown in Figure 2.

In order to solve the problems of high root node pressure and high system risk in traditional distributed file systems, a new model is proposed. This model is composed of data storage nodes and management servers. The data storage nodes are responsible for data storage and data management services. The management server is responsible for managing the service process, maintaining the currently registered data service process, and analyzing the data source address. The management server can be an arbitrary computer in the cloud, which is a dynamic distributed file system. The distributed computer system is suitable for a variety of data processing modes, including distributed computing workstations and computing clients, which execute task allocation processes and task execution processes, respectively, among which multiple task execution processes can run, and these task allocation and execution processes can all be deployed on any computer in the cloud, and the data service process is deployed on the machine that stores the data and is responsible for the distribution and reception of the machine's data. This model belongs to the Master-Worker two-tier structure. The Master is responsible for task decomposition, task assignment, and client-related work. Once the Worker is started, it first registers with the task assignment, and the process assigns tasks to perform corresponding functions. The data service process requests data or uploads data and reports to the task allocation process according to its current state, as shown in Figure 3.

3.3. Video Processing Architecture Design. The video processing architecture consists of three parts: (1) data service process processing (DataServer), (2) task allocation process processing (WorkStation), and (3) task execution process (WorkClient). Among them, the task execution process can run multiple times at the same time according to the actual situation. The executable programs of the three processing processes are the same, but the composition of the respective command execution files (CmdFile) is different. The data processing task allocation process uses dynamic scheduling to allocate tasks. First, assign tasks to them according to the currently registered task execution process. Once the task is completed, it will apply for a new task to the task allocation process. At the same time, the assignable task execution process can be changing at any time, and the task execution process can also be added and withdrawn at any time. Since the execution process communicates with the data server separately, the communication overhead can be greatly reduced, and the additional overhead caused by management allocation can also be reduced. The data service process is deployed in the video data storage server or storage server array, communicates with the task execution process, distributes corresponding data according to the needs of the task execution process, and receives and stores the data processed by the task execution process. The task of the task allocation process is to analyze the task of video data processing and the allocation task. After the task is analyzed, it can also be used as a task execution process. The task execution process is mainly to perform a certain video processing subtask assigned by the task allocation process, as shown in Figure 4.

In data storage, in order to maintain storage flexibility, video data can be stored separately or stored in a server, and any computer in the cloud can become the management terminal. The relevant calculation process is as follows: (1) run the management service process; (2) start the data service process of the site machine where the data involved in the calculation is located; (3) register the data service process with the management service process and submit the managed data source to the local machine; (4) the data client submits the required data network data to the management service process, and the management service process parses the relevant parameters to the data client; and (5) the data client requests data processing services from the site machine based on these parameters.

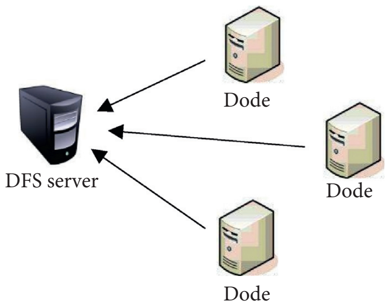


Figure 1: Parallel computing model based on cloud computing.

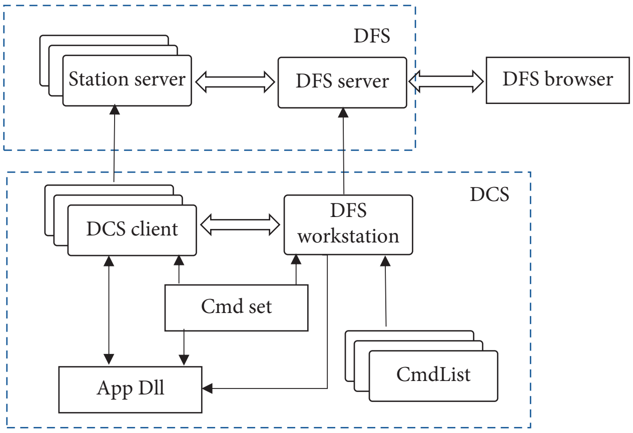


Figure 2: Structure of a distributed file system.

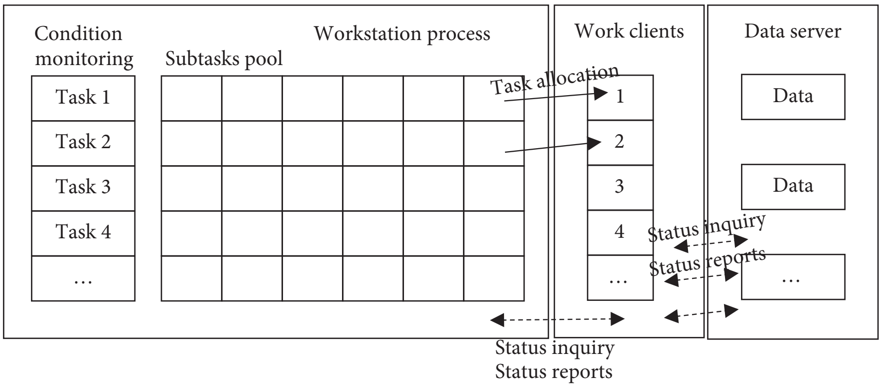


Figure 3: Distributed computing system architecture.

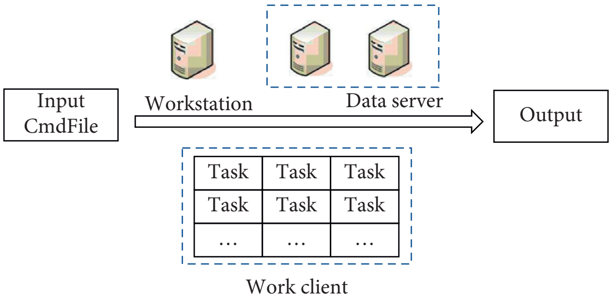


Figure 4: Video processing architecture.

4. Application of Video Key Frames Extraction Technology Based on Deep Learning

At present, in the main smart video surveillance field, pattern recognition technology is usually used. Common algorithms and detection methods include face recognition, anomaly recognition, and motion recognition, but most of these solutions are designed for specific application scenarios. The general scene recognition is not satisfied enough. Therefore, this research proposes a video key frame extraction and retrieval scheme based on deep learning. The key frame of the video represents the most significant feature of each shot of the video. Therefore, accurately extracting the key frame of each shot can effectively reduce the processing time of retrieval and improve the accuracy of retrieval. This scheme is divided into two steps. The first is to design an adaptive key frame recognition scheme and extraction scheme, and the second is to design a key frame retrieval algorithm based on convolutional neural network.

4.1. Video Layered Structure and Key Frames. The video can be divided into several scenes, each scene is divided into several shots, and each shot includes several key frames. Its structure is shown in Figure 5.

4.2. Shot Fragment Detection Algorithm. Video contains information such as spatial domain, time domain, plot, and features. Directly extracting and indexing features of the video are extremely complex work and consume a lot of storage space and computing time. If the accuracy of the extracted key frames is low, it will directly have an adverse effect on content-based video retrieval, video recognition, and scene analysis. As mentioned earlier, many researchers have proposed many key frame extraction schemes, but these schemes still have shortcomings. For example, there are many algorithmic solutions that select the first frame of each shot as the key frame, but this solution is easy to lose a large amount of visual information of the lens, and the randomness is strong, and the scale is not easy to grasp; there are some solutions to select key frames by enumerating and comparing each frame of the lens. The pressure on computing power and storage capacity is also great. According to the key frame extraction technology that needs to have the characteristics of high accuracy and fast calculation speed, combined with the scheme proposed by Liang and Wen [16], this paper proposes a new algorithm, as follows:

Input: video sequence, output: lens S_1, S_2, S_3, …, S_n.

4.3. The Key Frame Extraction Algorithm of the Lens. Each shot contains many repeated frames, so there is no need to process each frame in the shot. First, extract the summary information of each shot of the video, and the extracted key frames should contain the most salient features. The key frame extraction algorithm of the shot is as follows:

The input is video footage, and the output is key frame collection.

```
FOREACH Frame $f_n$
    Choose $f_1 f_2$   /*Choose $f_1 f_2$*/
    FOREACH Block$b_n$   /*$b_n$ is non-overlapping block of $16 \times 16$*/
        Choose $b_1 b_2$
        Wavelet exchange of $b_1 b_2 : F_{ij}(x, y) = X(x, y) f_{ij}(x, y) X^{-1}(x, y)$
        Calculate the distance between wavelet transform blocks:
        $\mathrm{L2}_{ij} = \sqrt{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (|F_{ij}(x, y) - F_{ij-1}(x, y)|)^2}$
          Save the distance to the vector
    Calculate the average distance of the distance vector
IF $\mathrm{L2}_{ij} \le \mathrm{LT}$,   /*LT is lens threshold*/
    Conditional frames belong to the same lens
```

```
FOREACH lens $st_n$
    FOREACH Frame $f_n$
    Calculate the average of each frame and save it in a vector
    Count the minimum, maximum, and mean values of the mean vector
    Select the frame closest to the average value as the key frame
```

Firstly, calculate the average value of all frames in the shot, and use the frame with the average value closest to the vector average as the key frame, thereby realizing adaptive key frame extraction.

5. Implementation of Vehicle Detection in Traffic Monitoring Video Based on Deep Learning

5.1. Related Theoretical Basis. The overall architecture of the Faster-RCNN network is shown in Figure 6. The main functions of each layer are as follows:

(1) Conv layers extract feature maps: as a CNN network target detection method, Faster R-CNN first uses a set of basic conv + relu + pooling layers to extract the feature maps of the input image, which will be used in the subsequent RPN layer and fully connected layer.

(2) RPN (Region Proposal Networks): the RPN network is mainly used to generate region proposals. First, a bunch of anchor boxes are generated. After cutting and filtering them, Softmax is used to determine whether the anchors belong to the foreground or the background, that is, object or not object, so this is a two-category, at the same time, another branch bounding box regression modifies the anchor box to form a more accurate proposal (note: the more accurate here is relative to the next box regression of the fully connected layer).

(3) ROI pooling: this layer uses the proposals generated by RPN and the feature map obtained from the last layer of VGG16 to obtain a fixed-size proposal feature map. After entering it, it can use the full

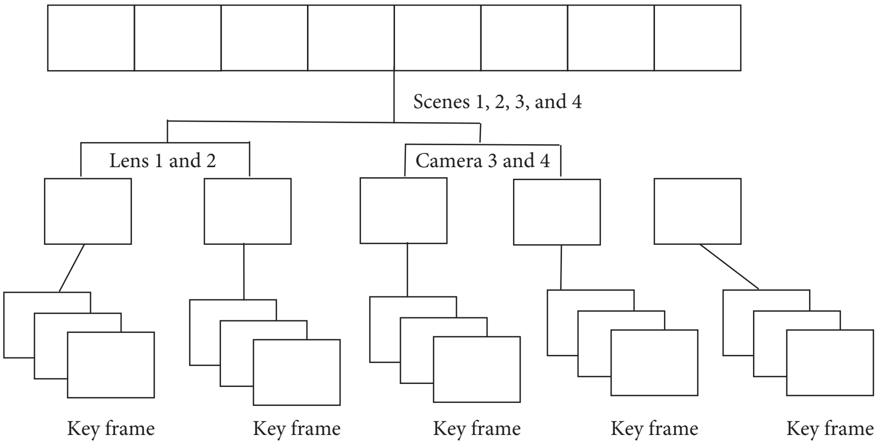


Figure 5: Video layered structure diagram.

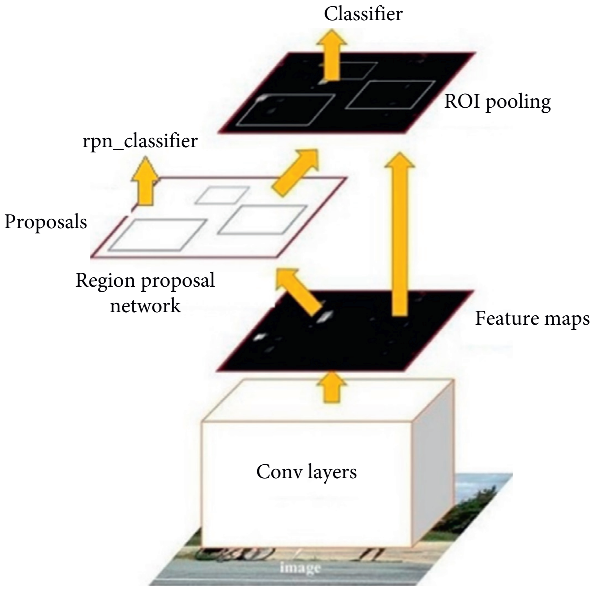


Figure 6: The overall architecture of the Faster R-CNN network.

connection operation to perform target recognition and positioning.

(4) Classifier: the ROI pooling layer will be formed into a fixed-size feature map for full connection operation, Softmax is used to classify specific categories, and at the same time, L1 Loss is used to complete the bounding box regression operation to obtain the precise position of the object.

When Faster R-CNN is applied to a traffic video surveillance system, high-resolution images may cause redundant feature information.

5.2. Vehicle Inspection Model Design. This research uses deep learning methods to solve the problem of vehicle detection under surveillance video [17]. The vehicle detection process is divided into two stages: training and detection. The execution steps of the training phase are as follows: (1) extract the training sample set, and make the sample training set according to the PASCAL VOC data set format and (2) input the sample training set into the neural network for training, and after multiple iterations, the trained vehicle detection network is obtained; the execution steps in the detection stage are as follows: (1) input the image directly into the trained neural network, and obtain the specific position of the outer frame of the vehicle and mark it on the original image and (2) output the vehicle detection result, as shown in Figure 7.

In traffic surveillance videos, high-resolution images are likely to cause feature information redundancy, and the use of nonmaximum values to suppress NMS when vehicles overlap most of the time can easily cause the detection frame to be lost. In order to solve this problem, we can consider using residual network technology to optimize the feature extraction layer, using hole convolution to filter redundant features, and using Soft-NMS to filter candidate frames. The structure is shown in Figure 8. Based on this model, it is divided into four components: feature extraction layer,

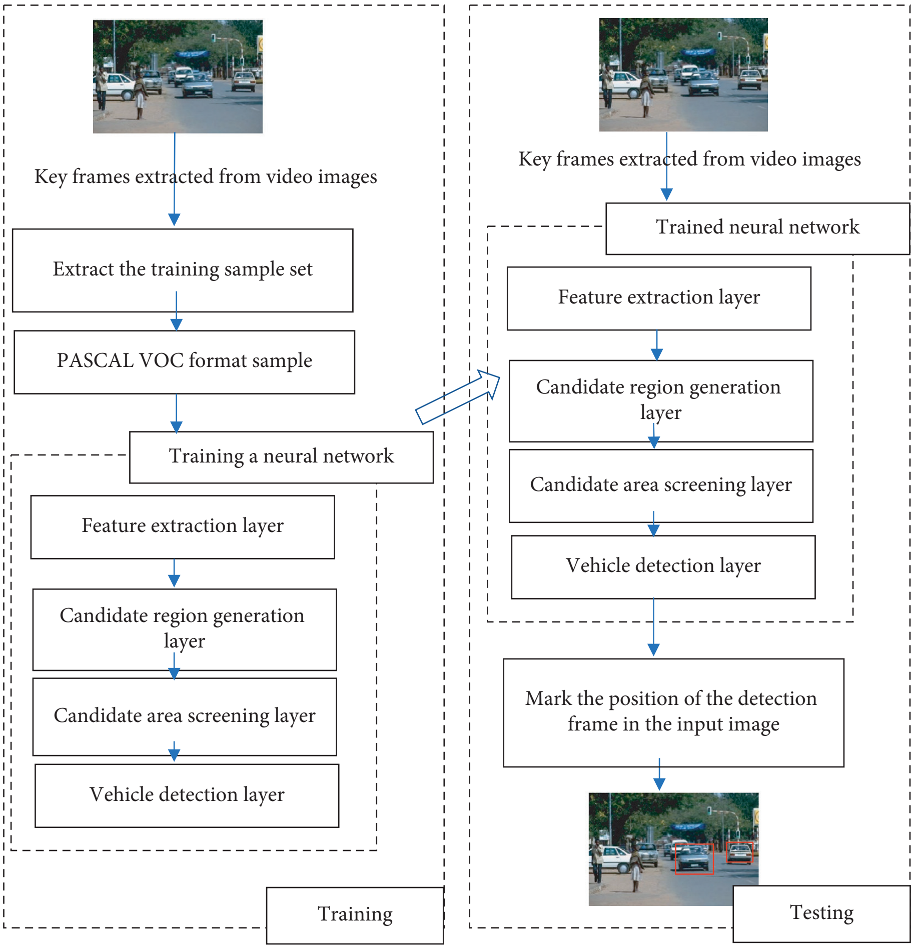

Key frames extracted from video images
Extract the training sample set
PASCAL VOC format sample
Training a neural network
Feature extraction layer
Candidate region generation layer
Candidate area screening layer
Vehicle detection layer
Training
Key frames extracted from video images
Trained neural network
Feature extraction layer
Candidate region generation layer
Candidate area screening layer
Vehicle detection layer
Mark the position of the detection frame in the input image
Testing

Figure 7: Vehicle detection algorithm flow based on faster R-CNN.

candidate region generation layer, candidate region screening layer, and vehicle detection layer.

(1) Feature extraction layer: perform feature extraction on the input image to generate a feature map

(2) Candidate region generation layer: use 3 × 3 hole convolution with expansion coefficient $r = 2$ to filter redundant features, and add anchor mechanism to generate initial candidate regions;

(3) Candidate region screening layer: compared with NMS, Soft-NMS is a softer screening criterion for candidate frames. Therefore, Soft Nonmaximum Suppression (Soft-NMS) is used for coarse screening of initial candidate regions.

(4) Vehicle detection layer: perform ROI pooling on the candidate area, realize a fixed length and output to the fully connected layer, and then connect to Soft-NMS again to screen the vehicle detection frame, and finally output the vehicle detection result.

5.3. Loss Function Design. The function of the loss function is to adjust the original model according to different environments to ensure the accuracy of data processing. The neural network is trained by defining a multitask loss function, such as the following:

$$L(\{p_i\}\{l_i\} = \frac{1}{N_{\mathrm{cls}}}\sum_i L_{\mathrm{cls}}(p_i p_i^*) + \lambda \frac{1}{N_{\mathrm{reg}}}\sum_i p_i^* L_{\mathrm{reg}}(l_i p l_i^*), \tag{1}$$

where $N_{\mathrm{cls}}, N_{\mathrm{reg}}$, and λ balance the normalized weights of classification loss and regression loss and I is the index of the i-th candidate frame in small batch processing.

The probability is that the i-th candidate box is the target. If the i-th candidate box is a candidate target, then $p_i^* = 1$; otherwise, $p_i^* = 0$. The classification loss function and regression loss function are defined as formulae (2) and (3):

$$L_{\mathrm{cls}}(p_i p_i^*) = -\log[p_i p_i^* + (1 - p_i^*)(1 - p_i)], \tag{2}$$

$$L_{\mathrm{reg}}(t_i t_i^*) = R(t_i - t_i^*), \tag{3}$$

where R is the smooth$_{\mathrm{L1}}$ function. $t_i = \{t_x, t_y, t_w, t_h\}$ is a vector-prediction parameterized candidate frame

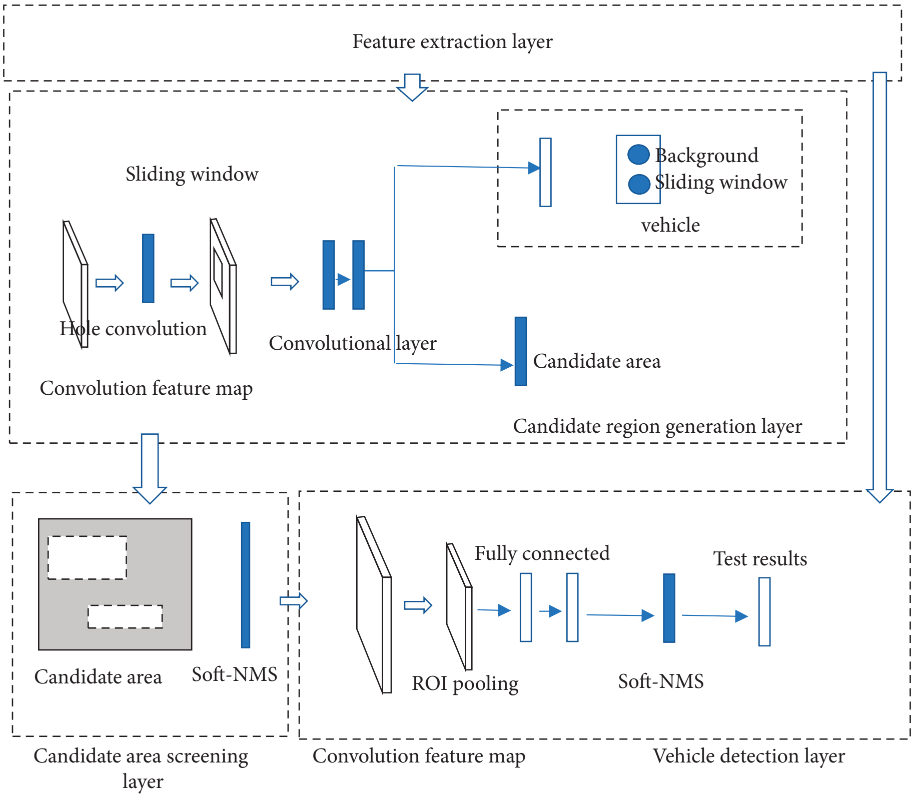


Figure 8: Structure diagram of neural network model.

coordinates and $t_i^* = \left\{t_x^*, t_y^*, t_w^*, t_h^*\right\}$ is the coordinate vector of real boundaries.

t_i and t_i^* are defined as follows:

$$
\begin{aligned}
t_x &= \frac{\left(x - x_j\right)}{w_j},\\
t_y &= \frac{\left(y - xy_j\right)}{h_j},\\
t_w &= \log\left(\frac{w}{w_j}\right),\\
t_h &= \log\left(\frac{h}{h_j}\right),\\
t_x^* &= \frac{\left(x^* - x_j\right)}{w_j},\\
t_y^* &= \frac{\left(y^* - y_j^*\right)}{h_j},\\
t_w^* &= \log\left(\frac{w^*}{w_j}\right),\\
t_h^* &= \log\left(\frac{h^*}{h_j}\right),
\end{aligned}
\tag{4}
$$

where (x, y), (x_j, y_j), and (x^*, y^*) are the forecasting area, candidate area, and official regional centre coordinates and (w, h), (w_j, h), and (w^*, h^*) are the width and height of predicted regions, candidate regions, and formal regions, respectively.

5.4. System Execution Process. The vehicle detection algorithm can be divided into two stages: training and detection. The main steps are as follows.

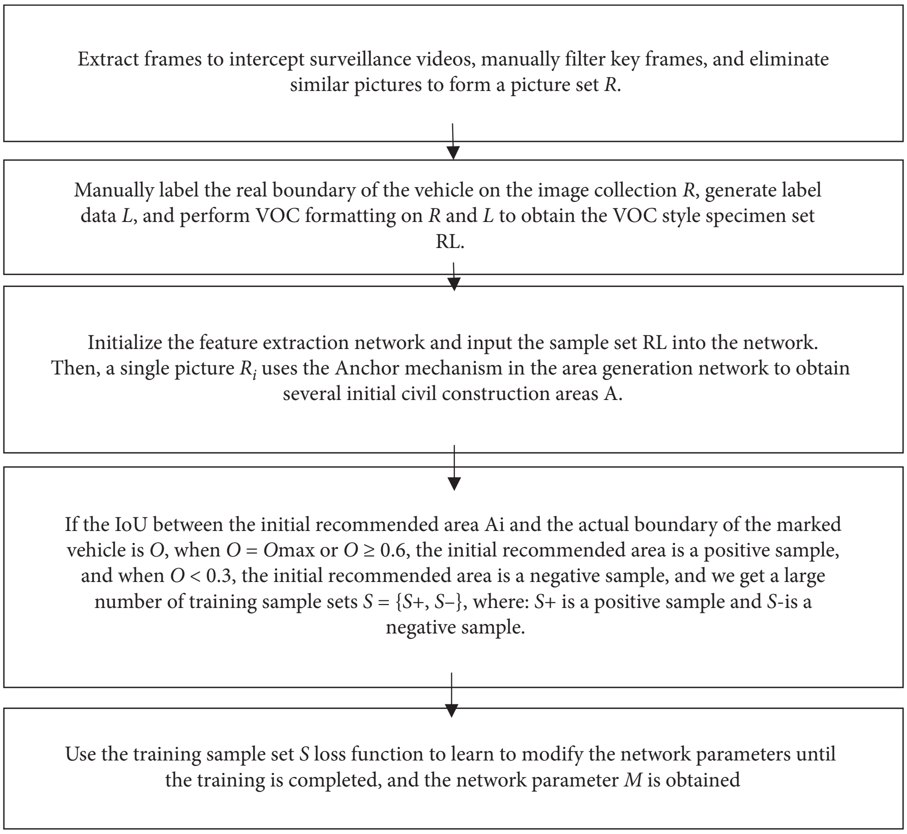
5.4.1. Training Part. The training process is shown in Figure 9.

5.4.2. Detection Section. The flow of the detection part is shown in Figure 10.

6. Implementation of Vehicle Detection in Traffic Surveillance Video Based on Deep Learning

Considering that there are different conditions in nature, such as daytime, night, rainy days, and traffic congestion, the research team conducted multiple sets of comparative experiments. The experimental results are shown in Table 1.

In the daytime environment, the accuracy of the system algorithm is more than 90%, and the effect is good. The accuracy of the system algorithm is more than 70% in traffic jams and rainy night environments, which is basically within the available range. From the comprehensive results, the algorithm designed in this paper is basically satisfactory.



Extract frames to intercept surveillance videos, manually filter key frames, and eliminate similar pictures to form a picture set *R*.

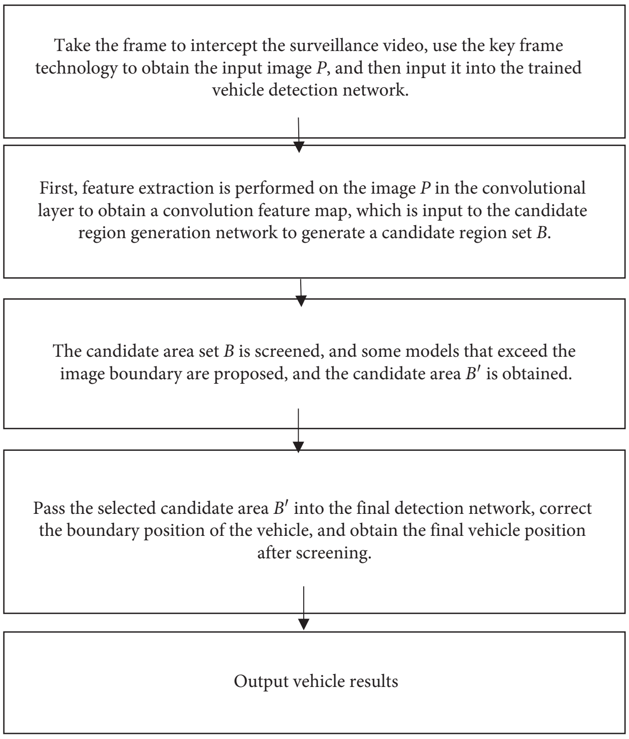
Manually label the real boundary of the vehicle on the image collection *R*, generate label data *L*, and perform VOC formatting on *R* and *L* to obtain the VOC style specimen set RL.

Initialize the feature extraction network and input the sample set RL into the network. Then, a single picture R_i uses the Anchor mechanism in the area generation network to obtain several initial civil construction areas A.

If the IoU between the initial recommended area Ai and the actual boundary of the marked vehicle is *O*, when $O = O\text{max}$ or $O \geq 0.6$, the initial recommended area is a positive sample, and when $O < 0.3$, the initial recommended area is a negative sample, and we get a large number of training sample sets $S = \{S+, S-\}$, where: *S*+ is a positive sample and *S*-is a negative sample.

Use the training sample set *S* loss function to learn to modify the network parameters until the training is completed, and the network parameter *M* is obtained

Figure 9: System implementation flow-training section.



Take the frame to intercept the surveillance video, use the key frame technology to obtain the input image *P*, and then input it into the trained vehicle detection network.

First, feature extraction is performed on the image *P* in the convolutional layer to obtain a convolution feature map, which is input to the candidate region generation network to generate a candidate region set *B*.

The candidate area set *B* is screened, and some models that exceed the image boundary are proposed, and the candidate area B' is obtained.

Pass the selected candidate area B' into the final detection network, correct the boundary position of the vehicle, and obtain the final vehicle position after screening.

Output vehicle results

Figure 10: System implementation flow-detection section.

Table 1: Statistics of algorithm accuracy in different environments.

	Daytime	Daytime with rain	Cloudy day	Night with light	Night with rain	Traffic congestion	Synthesis
Accuracy (%)	91.2	85.3	88.6	80.6	70.6	71.6	79.9

Table 2: Comparison of accuracy between this algorithm and other algorithms.

	Accuracy (%)						
	Daytime	Daytime with rain	Cloudy day	Night with rain	Night with rain	Traffic congestion	Synthesis
This algorithm	91.2	85.3	88.6	80.6	70.6	71.6	79.9
R-CNN	77.5	71.6	76.3	73.2	63.5	62.1	69.6
SA-FRCNN	73.6	68.9	70.5	69.8	60.3	59.8	66.5
DPM	54.4	50.3	52.6	48.4	42.3	43.2	47.7

It can be seen from Table 2 that the performance of the traditional vehicle detection algorithm DPM is poor, especially under complex conditions, and the accuracy is less than 50%. R-CNN and SA-FRCNN are given to deep learning algorithms. The accuracy is much higher than that of traditional DPM, and the average accuracy is close to 70%. However, the accuracy of the algorithm in this study is significantly higher than that of the traditional algorithm.

7. Conclusion

Aiming at the characteristics of complex traffic video surveillance scenes and high resolution of single-frame video images, in order to improve the accuracy and time efficiency of the video retrieval scheme, a deep learning-based video key frame extraction and video retrieval scheme is proposed, which is combined with Faster R-CNN. The improved algorithm of R-CNN vehicle detection is used in intelligent traffic video analysis. First, add a hole convolution to the network to filter out the redundant features in the high-resolution video image, and then replace the original NMS mechanism with Soft-NMS to adapt to the overlap of vehicles, making it more suitable for traffic monitoring video vehicle detection. The experimental results show that the improved model improves the accuracy of detection and reduces the missed detection rate, and the experimental results are satisfactory.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors are grateful for the financial support provided by the Research and Practice Project of Higher Education Teaching Reform in Hebei Province, China (2019GJJG604).

References

[1] D. Tran, L. Bourdev, R. Fergus et al., “Learning spatiotemporal features with 3d convolutional networks,” in *Proceedings of the Computer Vision (ICCV), 2015 IEEE International Conference on (S1550-5499)*, pp. 4489–4497, IEEE, Santiago, Chile, December 2015.

[2] J. Donahue, L. A. Hendricks, M. Rohrbach et al., “Long-term recurrent convolutional networks for visual recognition and description,” *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 39, no. 4, pp. 677–691, 2017.

[3] K. Simonyan and A. Zisserman, “Two-stream convolutional networks for action recognition in videos,” *Advances in Neural Information Processing Systems*, vol. 1, no. 4, pp. 568–576, 2014.

[4] L. Wang, Y. Xiong, Z. Wang et al., “Temporal segment networks: towards good practices for deep action recognition,” *Acm Transactions on Information Systems*, vol. 22, no. 1, pp. 20–36, 2016.

[5] Y. Hu and W. Zheng, “Human action recognition based on key frames,” in *Advances in Computer Science and Education Applications (S1865-0929)*, pp. 535–542, Springer Berlin Heidelberg, Berlin, Germany, 2011.

[6] R. Girshick, J. Donahue, T. Darrell et al., “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Columbus, OH, USA, June 2014.

[7] K. He, X. Zhang, S. Ren, and J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[8] R. Girshick, “Fast R-CNN,” in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, Las Condes, Chile, December 2015.

[9] S. Ren, K. He, R. Girshick et al., “Faster R-CNN: towards real-time object detection with region proposal networks,” *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2015.

[10] C. L. Wang and Z. S. Zhang, “Design of large-scale traffic video processing frameworks based on private cloud,” *Computer Engineering and Applications*, vol. 53, no. 21, pp. 254–257, 2017.

[11] M. Armbrust, A. Fox, R. Griffith et al., “A view of cloud computing,” *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.

[12] C. Chen, J. Lin, X. Wu, J. Wu, and H. Lian, “Massive geospatial data cloud storage and services based on NoSQL database technique,” *Journal of Geo-Information Science*, vol. 15, no. 2, pp. 166–174, 2013.

[13] L. L. Qin, N. W. Yu, and D. H. Zhao, “Applying the convolutional neural network deep learning technology to behavioural recognition in intelligent video,” *Tehnicki Vjesnik*, vol. 25, no. 2, pp. 528–535, 2018.

[14] L. L. Qin and L. H. Kan, “Application of video scene semantic recognition technology in smart video,” *Tehnicki Vjesnik*, vol. 25, no. 5, pp. 1429–1436, 2018.

[15] H. Qian and L. L. Qin, “The design of intelligent transportation video processing system in big data environment,” *Special Section on Big Data Technology and Applications in Intelligent Transportation*, vol. 8, pp. 13769–13780, 2020.

[16] J. S. Liang and H. P. Wen, “Key frame abstraction and retrieval of videos based on deep learning,” *Control Engineering of China*, vol. 26, no. 5, pp. 965–970, 2019.

[17] N. Bodla, B. Singh, R. Chellappa et al., “Soft-NMS—improving object detection with one line of code,” in *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, October 2017.