Chinese License Plate Recognition Based on YOLO v8 Convolutional Neural Network

Jiajie Li

School of Automation and Electronic Information

Xiangtan University, Xiangtan, China

lijiajie@xatu.edu.cn

Abstract. In recent years, with the acceleration of urbanization and the continuous increase in the number of vehicles, license plate recognition technology has been widely applied in fields such as intelligent traffic management and security monitoring. However, traditional license plate recognition systems have certain limitations in recognition accuracy and real-time performance, such as false detection and slow recognition speed. To address these issues, an efficient and accurate Chinese license plate recognition system is urgently needed. By combining YOLO v8 and CRNN, this study not only achieves end-to-end license plate detection and recognition but also overcomes the shortcomings of traditional methods, thereby improving recognition accuracy and real-time performance. The main work of this research is to utilize YOLO v8 and CRNN technologies to achieve high accuracy and efficiency of the license plate recognition system. This paper first introduces the basic knowledge of YOLO v8 and CRNN deep learning models and their application advantages in object detection and text recognition. Subsequently, the construction process of the Chinese license plate recognition system combining YOLO v8 and CRNN is elaborated, including basic principles and training steps. In the experimental phase, a comprehensive evaluation and testing of the system are conducted using public license plate datasets and self-made datasets containing license plate images under various complex conditions. Experimental data demonstrate that the system performs excellently in the accuracy and stability of Chinese license plate recognition, with an overall recognition accuracy of 94%. Finally, the application effects of the system and possible directions for future improvement are analyzed, emphasizing its application potential in fields such as intelligent traffic and security monitoring. This study provides a new perspective and solution for the innovation of Chinese license plate recognition technology and holds significant practical value.

Keywords: License Plate Detection; License Plate Recognition; Convolutional Recurrent Neural Network; Model Training; YOLO v8

1. Introduction

The research on license plate recognition technology is of great significance, mainly reflected in its promotion of traffic management and safety. Statistical data show that by 2023, the national motor vehicle ownership exceeded 435 million, including 336 million automobiles; the number of motor vehicle drivers reached 523 million, among which 486 million were automobile drivers. With the acceleration of urbanization and the rapid growth of vehicle ownership, the difficulty of traffic management has increased significantly. The application of license plate recognition technology helps manage such a large number of vehicles. By real-time recognizing vehicle license plate information, traffic departments can realize functions such as traffic violation monitoring, intersection traffic flow statistics, and parking management, thereby improving the efficiency and accuracy of traffic management[1]. In addition, license plate recognition technology also helps improve traffic safety by promptly detecting traffic violations and abnormal traffic conditions, reducing the possibility of traffic accidents.

Furthermore, the research and application of license plate recognition technology have a profound impact on the construction of smart cities. As an important part of the intelligent transportation system, license plate recognition technology can improve the efficiency and convenience of urban transportation, enhance citizens' travel experience, and promote the intelligent development of

https://doi.org/10.58244/joerai.263728

cities[2]. By applying license plate recognition technology to scenarios such as intelligent parking systems and traffic signal control systems, cities can better address urban traffic problems such as traffic congestion and difficulty in parking.

Traditional license plate recognition systems are mainly based on image processing and pattern recognition technologies, whose accuracy and robustness are affected, especially in the face of complex scenes and variable license plate forms[3]. In response to these situations, an efficient and accurate Chinese license plate recognition system is required. By improving the accuracy and real-time performance of the license plate recognition system, the efficiency and level of urban traffic management can be effectively enhanced, the capability of urban security monitoring can be further strengthened, and more reliable technical support can be provided for urban intelligent development and public safety.

In the international context, foreign countries started earlier in the research of license plate recognition, focusing on practical license plate recognition issues. They have gradually proposed many theories in traditional license plate recognition and deep learning-based license plate recognition methods, and applied them to real life to produce a series of reliable products, accumulating a lot of experience and technical achievements[4]. For example, Hsu et al. proposed an edge clustering algorithm, which effectively solves the license plate detection problem by detecting and clustering license plate edges, especially in complex backgrounds. Laxmi et al. utilized the frequency domain characteristics of images to improve the anti-interference ability of license plate recognition through frequency domain transformation and filtering methods[5]. In recent years, with the development of deep learning technology, more and more researchers have begun to apply it to license plate recognition. Abdullah et al. introduced the YOLO v3 algorithm into license plate detection, achieving high-accuracy license plate detection by optimizing the network structure and parameters[6]. Shivakumara et al. also adopted classification algorithms and applied a combination of convolutional neural networks and recurrent neural networks. In this scheme, CNN has high recognition ability and is therefore used for feature extraction, while BLSTM has the ability to extract contextual information based on past information[7]. For classification, the research team proposed voting based on dense clustering, which separates the foreground and background of the license plate, thereby successfully detecting the license plate.

Some countries have also successfully applied license plate recognition technology to practical traffic management systems. For example, the ARGUS system in the UK adopts advanced license plate recognition technology to realize automatic recording and punishment of illegal vehicles, with a recognition time of about 100 milliseconds, which effectively improves the efficiency and fairness of traffic management[8]. The See/Car system of Hi-Tech Company is also a successful example. This system not only has license plate recognition function but also can realize vehicle tracking, speed detection and other functions, providing comprehensive technical support for intelligent traffic systems.

Although the research on license plate recognition technology in China started late, significant progress has been made in recent years[9]. In the early stage, domestic researchers mainly studied and improved foreign algorithms to adapt to the actual situation in China. With the rise of deep learning technology, more and more domestic scholars have begun to apply deep learning models to license plate recognition. For example, some scholars have proposed license plate localization and character recognition algorithms based on deep learning for the characteristics of Chinese license plates, achieving good results[10]. For instance, the license plate detection algorithm based on CNN (Convolutional Neural Network) described by Li Xingwei et al. is a method combining modern deep learning technology. This method utilizes the powerful feature extraction ability of CNN to detect license plates from images faster and more accurately than traditional algorithms[11]. The one-stage algorithm they proposed, based on YOLO and MobileNetV2, combines the advantages of these two algorithms to further improve the accuracy and efficiency of license plate detection.

At the same time, many domestic enterprises and research institutions have also made remarkable progress in the field of license plate recognition[12]. Hanwang Company, affiliated with the Institute of Automation of the Chinese Academy of Sciences, has accumulated rich experience in the research

https://doi.org/10.58244/joerai.263728

and application of license plate recognition technology, and its launched license plate recognition products have been widely used in parking lots, road monitoring and other fields. Hong Kong Asia Vision Technology Co., Ltd. has also made important breakthroughs in the field of license plate recognition, and its developed license plate recognition system has played an important role in traffic management in various regions[13].

In summary, the research status of license plate recognition at home and abroad shows that this field has received extensive attention and active exploration[14]. However, existing systems still have a series of limitations and challenges. In complex scenes, such as bad weather, insufficient light or excessive vehicle speed, the recognition rate is generally low[15]. Therefore, the continuous exploration and application of methods and technologies to improve the performance of license plate recognition systems will provide important support for the realization of more efficient and accurate vehicle management and traffic monitoring, and promote the development and improvement of intelligent transportation systems.

2. Theoretical Basis of Deep Learning

2.1 Artificial Neural Network

The artificial neural network element model (also known as a single-layer perceptron) is a basic component unit in deep learning, simulating the basic functions and structure of biological neurons[16]. Its basic principle is to generate an output signal through the weighted sum transmission of input signals, thereby realizing information processing and transmission. A typical artificial neuron model includes the following elements:

Input layer: The artificial neuron receives input signals from other neurons or the external environment, and each input signal has a corresponding weight to adjust its influence within the neuron.

Weights: Weights are the basic learnable parameters that make up an artificial neural network. Their main function is to quantify the importance of input signals for network prediction. Through the training process, the weights are adjusted to optimize the response of the neural network to input data, thereby improving the prediction accuracy of the model. Each input signal has a corresponding weight, and a larger weight indicates a greater influence of the corresponding input signal.

Activation function: The activation function is a key component in the neural network for introducing nonlinearity. It performs a nonlinear transformation on the weighted input signal to generate an output signal. This nonlinear transformation is the basis for realizing complex data processing capabilities. Among various activation functions, the sigmoid function and ReLU (Rectified Linear Unit) function are the most widely used. These activation functions enhance the expressive ability of the model, enabling it to approximate nonlinear data distributions.

Output signal: The artificial neuron model calculates an output signal according to the input signal and the corresponding weight through the activation function, which is used to transmit to other neurons or as the output of the entire model.

The artificial neuron model simulates the basic principle of biological neurons, and realizes complex nonlinear function fitting and pattern recognition tasks through weight adjustment and activation function action[17]. In the field of deep learning, multi-layer neural networks built by numerous artificial neurons are trained using the backpropagation algorithm, which can effectively process and learn complex datasets. During training, the parameters of the network are gradually adjusted using optimization algorithms to minimize the difference between the network's predicted values and the target values. This process includes forward propagation to calculate the output, calculation of errors, backpropagation to calculate gradients, and parameter update according to the gradients, iterating repeatedly until the desired accuracy is achieved or the number of iterations is reached[18]. To improve training efficiency and stability, techniques and strategies such as batch training and learning rate scheduling are often adopted. These methods help the network converge faster during training and improve generalization ability, thereby better adapting to various complex data and tasks.

https://doi.org/10.58244/joerai.263728

2.2 Forward and Backward Propagation Algorithms

The forward propagation algorithm is the process by which information propagates from the input layer to the output layer in the neural network[19]. In this process, input data passes through various layers in the network and undergoes nonlinear transformation through activation functions to finally obtain the output result of the network.

The backpropagation algorithm is one of the core concepts in deep learning[20]. Its process includes forward propagation, loss calculation, backpropagation, and parameter update. First, through forward propagation, the neural network transmits the input data and calculates the output of each layer. Then, the loss function is used to measure the gap between the network output and the actual target. After backpropagation, the gradient of the loss function with respect to each parameter (weight and bias) is calculated using the chain rule, and propagated forward layer by layer until reaching the input layer. Finally, the parameters in the network can be updated according to optimization algorithms such as gradient descent to gradually reduce the loss function. During training, termination conditions are usually set, such as stopping when the preset number of iterations is reached or the loss function converges to a certain extent. This process enables the neural network to automatically learn and continuously optimize parameters, thereby improving performance on given tasks.

Assume that the existing weight value in the neural network is W, the value of the bias is b, and the calculated loss function is J. Then, on the basis of W and b, take the partial derivative of the loss function.

Where α in the above two equations is the learning rate, which is used to control the rate of gradient iteration.

The backpropagation algorithm mainly focuses on the loss value. Therefore, let δ_j^L be the loss value generated by the j-th neuron in the L-th layer of the network.

To further show the derivation process of the backpropagation algorithm, it is assumed that the loss function adopts the mean square error loss function, and the true target value is set as y. Then, according to the definition of the mean square error loss function, there is the following formula:

According to the loss of each layer, the bias gradient and weight gradient of this layer are derived, and then the weight and bias of each layer are updated by vectorization, so as to realize the parameter optimization of the neural network.

we deeply explored the theoretical basis of deep learning, focusing on artificial neural networks and their core algorithms, and then gave a basic explanation of the classic convolutional neural network[26]. First, we introduced the concept and basic structure of artificial neural networks, as well as their importance in deep learning. Subsequently, when delving into the training mechanism of neural networks, we focused on their core components—the forward propagation and backpropagation algorithms. These two algorithms not only form the basic framework of the neural network learning process but also play a crucial role in practical applications[27]. In the forward propagation process, input data is gradually transmitted to the output layer through network layers, and the final output result is obtained. The backpropagation algorithm calculates the gradient of the loss function to update network parameters, enabling the network to gradually optimize the fit to the target[28]. On this basis, we introduced the convolutional neural network, a neural network architecture that has achieved great success in fields such as image recognition[29]. We detailed each component of CNN, including the input layer, convolutional layer, pooling layer, and fully connected layer, and elaborated on their roles and principles in image processing, facilitating readers to have a basic understanding of the content of subsequent chapters.

3. License Plate Localization Algorithm

3.1 YOLO v8 Network Introduction

YOLO is an object detection model[30]. Traditional object detection methods are usually based on sliding windows and manually designed feature extractors. These methods often require complex processes and a lot of computing resources, resulting in slow detection speed and limited accuracy. To solve these problems, YOLO (You Only Look Once), as an innovative object detection algorithm,

https://doi.org/10.58244/joerai.263728

has emerged. Its unique feature is that it can recognize the category and position of objects in an image with only one scan, and its effect is fast and good.

The principle of YOLO is very ingenious, and its core idea is to only require the center point of the object to be located in a specific frame[31]. This means that we do not need to design a very large frame to cover the entire object, but only need the center point of the object to fall within the frame. In the implementation of YOLO, the image is divided into grid cells, and each cell predicts multiple bounding boxes and their category probabilities. Through the method of Non-Maximum Suppression (NMS), the system finally determines the detection result. This end-to-end detection method makes YOLO have significant advantages in speed and accuracy.

YOLO allows S^2 grids to each predict B bounding boxes. Each bounding box has five parameters, namely the center position (x,y) of the object, its height (h) and width (w), and the confidence of this prediction[32]. The function of the bounding box is to determine the spatial position of the object, which usually requires it to provide four key position parameters: x-coordinate, y-coordinate, height h, and width w. We want the size of the input image to be arbitrary, which is not too difficult for the convolutional neural network. However, if the output position coordinates are arbitrary positive real numbers, the model's generalization ability on objects of different sizes may vary greatly. Therefore, normalize the data to make the value of continuous data between 0 and 1. For x and y, x and y are the center positions of the object. The real x divided by the width of the grid and the real y divided by the height of the grid can achieve the operation. Considering that an object may be much larger than the size of the grid, the predicted height and width of the object may be larger than the height and width of the bounding box.

After understanding the above indicators, we can understand that because our bounding boxes are represented by midpoint coordinates + width and height, each bounding box predicted by the grid requires its center to be within this grid[34]. Therefore, if it is not the middle grid, the IoU of other grids will naturally be relatively low, so the corresponding confidence will decrease. Therefore, the confidence of the really middle grid is often relatively large. In this way, we only need the center of this object to fall within this frame, and we can identify the category and position of the object in the figure with only one browse.

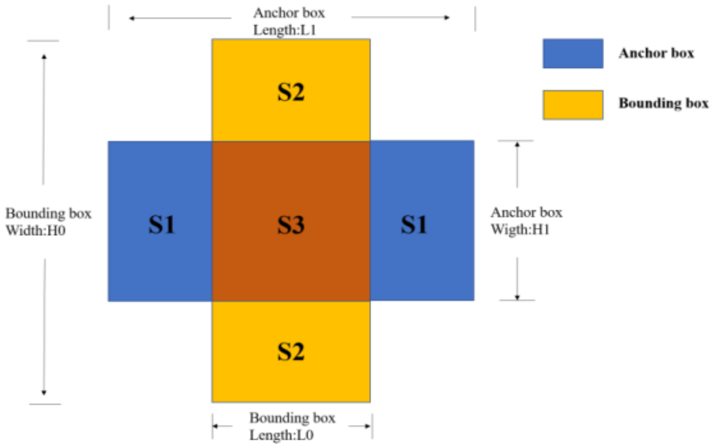

Figure 1. IOU

Another important technology of the YOLO network is introduced here, which is the technology of Non-Maximum Suppression (NMS)[35]. NMS is based on the intersection over union to solve the problem that a small object may be identified by multiple boxes during the recognition process. In plain language, what we do is first judge whether the categories of these grids are the same. Assuming that B1, B2, B3, and B4 (prediction boxes) in the above all identify dogs, then proceed to the next step. We keep B1, and then judge whether B2, B3, and B4 should be deleted[36]. We call B1 the maximum bounding box, calculate the IoU between the maximum bounding box and several other bounding boxes, and if it exceeds a threshold, such as 0.5, we consider that these two bounding boxes actually predict the same object, and delete the one with the smaller confidence[37].

The main network layer of YOLO v8 is divided into four parts[38]. First, the input layer responsible for inputting feature data; then the Backbone layer responsible for feature extraction, which is the core part of the entire network; then the Neck layer, responsible for pooling and feature fusion of the feature information extracted by the network, and its structure includes three types of networks: SSPNet, FPNet, and PANet; finally, the output layer Head, used to finally output the prediction result of the model.

Backbone plays the role of feature extraction in YOLOv8, usually adopting a deep convolutional neural network (CNN) architecture such as ResNet or Darknet[39]. The input image processed by Backbone is converted into feature maps, which contain rich semantic information and reveal the visual features of different levels in the image.

The role of FPN is to realize multi-scale fusion of features on the basis of the feature maps generated by Backbone to integrate feature information of different scales[40]. In the FPN stage, the feature layers obtained from Backbone are fused through upsampling and downsampling processes, enhancing the model's detection performance for multi-scale targets.

Yolo Head is responsible for the final object detection of the feature maps output by FPN, including classification and bounding box regression[41]. The feature map is regarded as a series of feature points at this stage, and each feature point is associated with a potential object position. Yolo Head analyzes these feature points to predict the existence and category of the object and its position.

Through the feature extraction of Backbone, the feature fusion and enhancement of FPN, and the accurate detection and localization of Yolo Head, YOLOv8 can effectively perform end-to-end object detection tasks.

3.2 Experimental Results and Analysis

Due to the complexity and large computational requirements of the YOLO v8 network model, the number of parameters of the model is also considerable, which leads to strict requirements on the training hardware[46]. We will train the model according to the existing resource equipment.

Dataset Introduction: The CCPD dataset is a large-scale open-source image collection designed specifically for Chinese urban license plate recognition[47]. It contains license plate images taken in various complex environments and has been carefully annotated. The dataset is divided into two subsets: CCPD2019 and CCPD2020, corresponding to images of traditional energy vehicles (blue license plates) and new energy vehicles (green license plates), respectively. The images in the CCPD2019 subset were mainly collected from parking lots in Hefei City, starting at 7:30 in the morning and continuing until 22:00 in the evening[48]. During the data collection process, the staff used Android POS devices for handheld shooting to obtain vehicle images. These images reflect a variety of complex environmental conditions, such as image blur, license plate tilt, and different weather conditions (including rainy and snowy days). The original intention of designing the CCPD dataset is to provide a resource containing a variety of license plate images to support the research and development of license plate recognition technology[49]. By including license plate images captured under various environmental conditions, the CCPD dataset aims to assist researchers and developers in more in-depth testing and optimization of their license plate recognition algorithms to improve the robustness and recognition accuracy of the algorithms.

The CCPD dataset does not have special annotation files. The annotations of the dataset are directly presented in the form of image file names, containing rich information such as the proportion of the license plate area, the angle of the license plate, the coordinates of the annotation frame, and the mapping relationship of the license plate number[50].

In addition to using the open-source dataset CCPD from the University of Science and Technology of China, this paper also collected images on the campus of Xiangtan University and obtained more than 500 license plate images from other open-source datasets on the Internet[51]. For these license plate images, we used the labelimg tool to label the files to be annotated, selected the yolo annotation format, marked the labels, and obtained real data. The pictures also include sample data under different shooting angles and different light conditions. The composition of the license plate localization dataset used in this paper is shown in Table 2:

First, it is necessary to select a pre-trained model. Referring to the official comparison of the performance of YOLO v8 models of different sizes on the COCO dataset and the conclusions obtained by others on the CCPD detection dataset through comparative experiments on the five different sizes of YOLOv8n/s/m/l/x: in comparison, the effect of the n series model is the lowest, the s series model is the next, the m series model is in the middle, and the effects of the l and x series models are close[52]. Considering the parameter level and inference speed, we finally selected the s series model as the inference model.

As two key adjustable parameters in the training of deep learning models, the reasonable selection of batch size and training epoch can improve the efficiency and performance of model training[53]. We need to conduct multiple experimental comparisons according to the number of our own datasets and the computer environment to confirm the parameters, so as to ensure that the model achieves the expected classification effect. The batch size determines the number of samples received by the model in each iteration, and the training epoch represents the number of times the entire dataset is traversed.

The selection of batch size directly affects the speed and stability of training. Usually, the number of times needs to be selected considering the performance of the own GPU and the ability of the trained model[54]. A smaller batch size may lead to more noise during the training process, making the model converge slower and may make the model fall into a local optimal solution. On the contrary, a larger batch size may increase memory requirements, leading to problems such as insufficient memory or insufficient computing resources during training. In model training, an appropriate batch size should be selected according to the actual situation so that the model can learn efficiently and maintain stability.

The number of training epochs should be sufficient to enable the model to fully learn the features of the dataset, but overfitting should also be avoided[55]. Determining the number of training epochs usually depends on monitoring the performance of the model on the validation set. Training should be terminated when the error rate on the validation set first rises to avoid overfitting. The following is the flow chart of the training process:

The following explains the specific indicator concepts involved in the YOLO visualization results:

Precision refers to the ratio of the number of samples correctly identified as positive classes by the classifier to the total number of samples classified as positive classes[56]. In academic writing, it is used to measure the accuracy of the classifier when determining positive classes. Specifically, precision indicates the reliability of the classifier in identifying positive classes, that is, the proportion of the positive classes identified by the classifier that are actually positive classes;

Recall represents the ratio of the number of samples correctly identified as positive classes by the classifier to the total number of samples actually positive classes[57]. In academic papers, recall is used to measure the ability of the classifier to correctly identify all positive samples. In other words, recall describes the ability of the classifier to find samples that are actually positive classes, that is, the recall rate of the classifier for all positive samples;

mAP: Abbreviation of Mean Average Precision, that is, mean average precision;

GIoU: It is the mean value of the GIoU loss function. As the number of training iterations increases, it is observed that the mean value gradually decreases, which indicates that the positioning accuracy of the detection box is continuously improving;

Objectness: It is the mean value of the object detection loss. The decrease of the mean value with training indicates that the object detection is more accurate;

Classification: It is the mean value of the classification loss. The decrease of the loss mean value means the improvement of classification performance;

After the dataset preparation is completed, download the selected pre-trained model, set the indicators BatchSize and epoch to 16 and 100 respectively. After obtaining the trained weights, we can test the model with the test set[58]. The detection result graph is as follows:

The upper left of Fig. 7 is a network license plate picture, taken from the front with bright light[59]. The recognition result of the license plate is shown in the figure, marked with a red box, the category is car, the confidence is 0.72, and the recognition is accurate;

https://doi.org/10.58244/joerai.263728

Figure 2. Vehicle Detection

The upper right of Fig. 7 is a network license plate picture, taken by downward shooting 30° from the front of the vehicle with a dark light environment[60]. The recognition result of the license plate is shown in the figure, marked with a red box, the category is car, the confidence is 0.81, and the recognition is accurate;

The lower left of Fig. 7 is a self-made license plate picture, taken by downward shooting 15° from the left front of the vehicle with very bright light[61]. The recognition result of the license plate is shown in the figure, marked with a red box, the category is car, the confidence is 0.92, and the recognition is accurate;

The lower right of Fig. 7 is a network license plate picture, taken by tilting 30° from the right side of the vehicle with a dark light environment[62]. The recognition result of the license plate is shown in the figure, marked with a red box, the category is car, the confidence is 0.84, and the recognition is accurate.

This chapter mainly introduces the license plate localization algorithm, focusing on the basic characteristics of domestic license plates and a commonly used localization algorithm—the YOLO v8 network[63]. First, we deeply explored the basic characteristics of domestic license plates, including size, shape, color, etc., laying a foundation for the subsequent localization algorithm. We then elaborated on the YOLO v8 network, getting a preliminary understanding from three aspects: basic idea, network structure, and loss function[64]. YOLO v8 is an object detection algorithm with high efficiency and accuracy, suitable for license plate localization tasks. In the experimental part, we first described the experimental environment and dataset preparation process, then showed the experimental process and result analysis[65]. Through the detailed analysis of the experimental results, we can evaluate the performance of the algorithm and further optimize the algorithm parameters to improve the accuracy and robustness of license plate localization[66]. The content of this chapter comprehensively introduces the theoretical basis and experimental verification of the license plate localization algorithm adopted, laying a foundation for subsequent character recognition.

4. License Plate Recognition Based on CRNN Network

After successfully obtaining and segmenting the license plate image through the license plate localization link, the subsequent character recognition process is one of the key links in license plate recognition, which is crucial to the reliability and accuracy of the entire system[67]. In this section, we focus on the topic of scene text recognition, which is the most common task in the field of image-based sequence recognition. It can be applied to people's daily lives and is a problem that machine vision has intended to solve since its birth[68]. Current license plate character recognition methods are mainly divided into two directions: traditional methods based on character segmentation and then recognition, and recognition technologies without segmentation.

Traditional license plate recognition usually consists of two steps: character segmentation and character recognition[69]. Character segmentation algorithms include methods based on template matching, vertical projection, connected domains, and cluster analysis. Through these methods, the license plate is segmented into individual characters before recognition. Traditional character recognition algorithms include methods based on template matching, feature statistics, and machine learning[70].

However, these traditional methods are usually complex and cumbersome[71]. The traditional character segmentation and recognition algorithm has the problem of strong correlation between front and back links. Any error in any link will directly affect the final recognition accuracy[72]. Especially when the license plate image is irregular, such as large-angle tilt, traditional character segmentation often leads to serious compression or deformation of the character image, which poses challenges to the subsequent recognition process[73].

Therefore, the traditional algorithm based on character segmentation and then recognition has been difficult to meet the actual needs[74]. This project adopts the CRNN neural network, which has the following characteristics: 1) Combining character segmentation and recognition into one step, effectively simplifying the recognition process[75]. 2) It can naturally handle license plates of any length, including blue plates and green plates[76]. 3) It can quickly and accurately output the license plate frame position information and license plate number[77]. Through this method, we can reduce the parking waiting time in practical applications, effectively improve the recognition accuracy, and avoid economic losses caused by recognition errors[78].

In the process of license plate recognition, large-angle license plates are often encountered; large-angle license plates generally refer to license plates with a horizontal tilt angle greater than 30 degrees and less than 60 degrees[79]. The correction method for large-angle license plates is a difficult problem that has plagued the industry for a long time. Traditional license plate tilt correction methods, such as straight line fitting, horizontal projection, and radon transformation, can usually only solve license plate tilt within 30 degrees, and are very dependent on accurate license plate localization and license plate binarization algorithms[80]. To solve the problem that the detected license plate has a certain inclination angle, which is not conducive

5. Acknowledgements

I would like to express my sincere gratitude to all those who have supported and assisted me throughout the research and writing of this thesis.

First and foremost, my deepest appreciation goes to my supervisors, Professor Xianyou Yang and Professor Peng Li. Their profound academic insights, rigorous research attitude, and patient guidance have been invaluable throughout the entire process of this thesis. From the initial topic selection, research framework design, to the optimization of experimental schemes and the revision of the thesis draft, their careful guidance and constructive suggestions have helped me overcome numerous difficulties and continuously improve the quality of this research.

I am also grateful to all the teachers and staff of the School of Automation and Electronic Information at Xiangtan University. Their diligent teaching and selfless dedication have laid a solid professional foundation for my academic research, enabling me to successfully carry out this thesis work. The comfortable learning environment and abundant academic resources provided by the university have also created favorable conditions for the smooth progress of this research.

My heartfelt thanks go to my family for their unwavering support and understanding. Their love and encouragement have given me the courage and motivation to face challenges during the research process. Whenever I encountered difficulties and setbacks, their care and support have always been my strongest backing, allowing me to focus wholeheartedly on my thesis research.

I would also like to thank my classmates and friends for their valuable discussions and exchanges during the research process. Their different perspectives and suggestions have inspired my thinking

and helped me broaden my research horizons. The mutual support and encouragement among us have made the arduous research journey full of warmth and joy.
Finally, I would like to acknowledge the authors of the referenced literatures, whose valuable research results have provided an important theoretical basis and technical reference for this thesis. I also appreciate the providers of the public datasets used in this research, which have ensured the validity and reliability of the experimental results.

References

[1] Li, X. W., Zhu, Z. Q., Wan, Q. B., et al. Real-time license plate detection based on deep learning[J]. Information & Communications, 2019(12): 1-2.

[2] Hsu, G. S., Chen, J. C., Chung, Y. Z. Application-Oriented License Plate Recognition[J]. IEEE Transactions on Vehicular Technology, 2013, 62(2): 552-561.

[3] Lu, J. Y., Xue, Z., Shao, K. Y., et al. License plate character segmentation based on improved connected component algorithm combined with vertical projection[J]. Techniques of Automation and Applications, 2015(12): 1-5.

[4] Matas, J., Zimmermann, K. Unconstrained licence plate and text localization and recognition[C]// IEEE Intelligent Transportation Systems. IEEE, 2005.

[5] Gao, C., Wang, F. L. License plate recognition algorithm based on template matching and local HOG features[J]. Computer Systems & Applications, 2017, 26(1): 1-7.

[6] Guan, J. Research on region iterative graph cut algorithm guided by K-nearest neighbor classification[D]. Soochow University, 2020.

[7] Ren, C., Song, J., Wang, Y., et al. A character segmentation method based on character structural features and projection[J]. Proc Spie, 2013, 8878: 2714-2739.

[8] Zhang, X. L. Optimization algorithm of license plate character recognition based on BP artificial neural network[J]. Computer Engineering and Applications, 2012, 48(35): 182-185.

[9] Gao, L. Research on the application of license plate recognition in complex scenes based on deep learning[D]. Xidian University, 2019.

[10] Chen, X. C. Research on deep learning algorithms and applications based on convolutional neural networks[D]. Zhejiang Gongshang University, 2014.

[11] Sun, P., Hu, X. D., Zhang, Y. J. Deep learning image target detection combined with attention mechanism[J]. Computer Engineering and Applications, 2019, 55(17): 1-5.

[12] Shi, B. G., et al. An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(11): 2298-2304.

[13] Redmon, J., Farhadi, A. YOLOv3: An Incremental Improvement[J]. arXiv preprint arXiv:1804.02767, 2018.

[14] Zhai, Y. H., Wang, J., Xu, L. Y., et al. Sheep behavior recognition method based on improved YOLO v5n[J]. Transactions of the Chinese Society for Agricultural Machinery, 2024, 55(4): 231-240.

[15] Zhao, P., Zhou, Y. X. DA-YOLO defect detection algorithm based on attention mechanism[J/OL]. Journal of China Jiliang University: 1-8.

[16] Liu, G. M., Li, Y. Target tracking and behavior recognition technology based on deep learning[J]. Technology Innovation and Application, 2024, 14(12): 164-167. DOI:10.19981/j.CN23-1581/G3.2024.12.038.

[17] Li, L., Lu, S. B., Ren, H., et al. Recognition and localization method of mulberry branches in complex environments based on improved YOLO v5[J]. Transactions of the Chinese Society for Agricultural Machinery, 2024, 55(2): 249-257.

[18] Xu, Q., Sun, S. K., Qian, J., et al. Research on cigarette carton upward code recognition method based on improved CRNN network[J/OL]. Acta Tabacaria Sinica: 1-9[2024-05-06].

[19] Guo, H., Ning, C. M., Han, S. S., et al. Natural environment text recognition system based on DBNET and CRNN-CTC[J]. Computer Applications and Software, 2023, 40(9): 132-136.

[20] Shu, S. S., Zheng, X. X., Wen, C. Y. Chinese handwritten text line recognition based on improved CRNN[J]. Journal of Chengdu University of Information Technology, 2023, 38(4): 422-428. DOI:10.16836/j.cnki.jcuit.2023.04.008.

[21] Feng, S. X. Research on OCR detection and recognition technology based on deep learning[D]. North China University of Technology, 2024. DOI:10.26926/d.cnki.gbfgu.2023.000830.

[22] Bochkovskiy, A., Wang, C. Y., Liao, H. Y. M. YOLOv4: Optimal Speed and Accuracy of Object Detection[J]. arXiv preprint arXiv:2004.10934, 2020.

[23] Ramchoun, H., Chikh, M. A., Idrissi, J., et al. Multilayer Perceptron: Architecture Optimization and Training[J]. International Journal of Interactive Multimedia and Artificial Intelligence, 2016, 4(1): 26-30.

[24] Lecun, Y., Bottou, L. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[25] Bochkovskiy, A., Wang, C. Y., Liao, H. YOLOv4: Optimal Speed and Accuracy of Object Detection[J]. 2020.

[26] Sifre, L., Mallat, S. Rigid-Motion Scattering for Texture Classification[J]. Computer Science, 2014, 3559: 501-515.

[27] Zhou, L. Chinese license plate recognition system based on YOLO v4 and CRNN+CTC algorithm[D]. Nanjing University of Posts and Telecommunications, 2023. DOI:10.27251/d.cnki.gnjdc.2022.000671.