2023 3rd International Conference on Frontiers of Electronics, Information and Computation Technologies (ICFEICT) | 979-8-3503-0235-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICFEICT59519.2023.00036

License Plate Recognition Method based on Convolutional Neural Network

1st Liping Zhao
Systems Engineering Institute
Academy of Military Sciences
Beijing, China
825889797@qq.com

2nd Fei Dai*
Systems Engineering Institute
Academy of Military Sciences
Beijing, China
df_iot@163.com

3rd Feng Li
Systems Engineering Institute
Academy of Military Sciences
Beijing, China
li_bopr@126.com

4th Zihan Zhao
Systems Engineering Institute
Academy of Military Sciences
Beijing, China
zzh_track@163.com

***Abstract*—Automatic license plate detection and recognition is very important for intelligent vehicle management. At present, the license plate recognition technology mainly obtains the better picture under the good illumination condition to achieve the high precision recognition effect. However, there are still great challenges to achieve better recognition effect by using the existing technology in long distance and large Angle license plate recognition. In order to solve the problem of license plate recognition under complex conditions, this paper constructs a suitable network structure and training parameters to improve the accuracy of license plate recognition. The license plate contains Chinese characters, whose structure is more complex than alphanumeric ones. Therefore, the paper uses the existing convolutional neural network LeNet-5 for transfer learning and improves it into a network suitable for license plate character recognition. Finally, 31 kinds of Chinese character data and 34 kinds of alphanumeric data are trained and tested to get the best parameter combination in the network. Meanwhile, the influence of parameter and function selection on recognition accuracy is analyzed and illustrated.**

Keywords-license plate recognition; deep learning; convolutional neural network; IMPLeNet-5

I. INTRODUCTION

As an important part of intelligent transportation system, license plate recognition becomes more and more difficult with the complexity of traffic scenes. As it is widely used for traffic data collection, parking lot entrance and exit management, highway toll management, road violation punishment, and other applications [1], [2]. There are two primary license plate recognition methods, namely indirect recognition and direct recognition. Indirect recognition involves recognizing the license plate through the information stored in the IC card or bar code installed on the car. Although this method has high accuracy and reliable operation, it also has high hardware equipment requirements and needs a national unified standard, making it difficult to promote. Direct recognition, on the other hand, is an image-based recognition method that not only improves recognition speed but also interactively solves recognition errors in the system. At the present stage, the license plate recognition system mainly adopts the way of image triggering to capture the vehicle image in real time, recognize the license plate, and carry out character segmentation and character recognition. Recognition accuracy and speed are directly related to the convenience and security of license plate recognition system, so this technology is of great significance for improving intelligent traffic management system.

Template matching, feature matching and neural network are commonly used in image recognition.The template matching method recognizes an object based on the positioning of a pattern in the image. However, it has limitations in that the matching target can only move in parallel, and the algorithm is invalid when the matching target rotates or changes in size. The feature matching method is essentially a sparse processing method of the feature space. This method may not effectively utilize implicit but potentially critical features, which limits its performance. Deep learning methods commonly used in various fields provide a new idea for image recognition. Among many deep learning algorithms, Convolutional Neural Networks (CNNs) can recognize the potential features of an image without complex manual feature extraction, avoiding the need for complex preprocessing of the image. CNNs have excellent image classification and recognition capabilities.By applying CNNs to license plate recognition, we can take full advantage of the benefits of this method. Additionally, further studies and optimizations of the network topology can enrich and expand the application fields of CNNs.

Uzun et al. [3] proposed a new feature extraction method using circular histogram technology to check characters. They create a feature vector based on the pixel density of characters in the sector and perform linear analysis on it. This method requires fewer parameters compared to optical character recognition (OCR) and existing feature extraction methods and has a high success rate.Zhao et al. [4] regards the license plate recognition is transformed into a sequence labeling problem, and the recursive neural network (RNNs) is used to train the license plate recognition model, and then the sequence characteristics of the license plate are summarized. This method does not need to segment the license plate, thus avoiding the errors caused by segmentation, but 38 of the given 650 license plates are mistaken because of preprocessing effect or noise influence. Asif et al. [5] adopted deep convolutional neural network technology to achieve multi-digit recognition and identify sequences with a length of up to 18 bits. The overall accuracy is 98%. However, this method only recognizes numbers and does not consider letters or Chinese characters.Pham et al. [6]

analyzed the mechanism of LeNet-5 algorithm, the model is optimized by reducing the number of convolution layers and neighborhood. Compared to LeNet-5, this method improved performance and real-time processing time but decreased accuracy by 2%.Dong et al. [7] proposed an improved CNN model to recognize letters and digits by simplifying the structure of LeNet-5. This method has high recognition rate and short recognition time, but did not recognize or verify Chinese character images.Liu et al. [8] applied deep convolutional neural networks to license plate recognition and trained the model using directly generated artificial images to judge the existence of license plates and recognize characters on the license plates.Ma et al. [9] used Convolutional Neural Networks (CNNs) to recognize visual types. The method maintains high accuracy under conditions of illumination variation and noise. However, the algorithm has a strong generalization ability, and the classification accuracy is not high.

Previous studies have shown that license plate letters and numbers have a simple structure and high recognition accuracy [10]-[13]. However, recognition accuracy for Chinese characters in license plates is significantly lower due to their complex structure. On the basis of in-depth analysis of the previous research, this paper improves the LeNet-5 model combined with the characteristics of Chinese characters, puts forward the IMPLeNet-5 model, carries out transfer learning, and verifies the effectiveness of the model, improve the accuracy of system identification, and improve the reliability of the system

II. METHOD

A. Introduction of Convolution Neural Network

The concept of convolutional neural networks (CNN) was initially introduced by Canadian neuroscientist Hubel et al. in 1959, who conducted research on cat visual information stimulated by multiple receptive fields layer by layer. Later, in 1980, Fukushima proposed the neurocognitive machine, which served as the predecessor of the convolutional neural network, by building on Hubel's model. In 1998, LeCun et al. proposed and trained the first CNN model, called LeNet-5 . This model integrated the advantages of sparse interaction of cognitive machines, reduced the number of parameters by using local receptive fields, weight sharing, and pooling, and achieved invariance of displacement, scaling, and distortion recognition. LeNet-5 is a typical example of the CNN model, primarily used to identify 32×32 digital images, and its network architecture is illustrated in Figure 1.

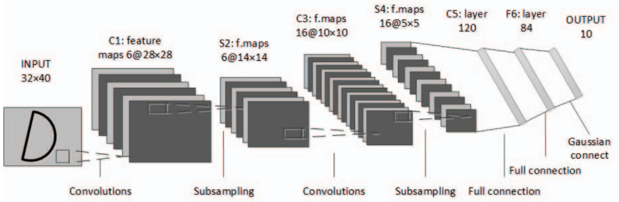

Figure 1. LeNet-5 network architecture.

LeNet-5 convolutional neural network consists of 7 layers (excluding input), and each layer includes trainable parameters (weights). The C-layer and S-layer are the network layers composed of neurons in the convolutional and subsampling layers, respectively. A convolution layer consisting of six characteristic graphs of C1 layer, and The 5×5 neighborhood of the input image is connected to each neuron, and the size of each feature map is 28×28. Layer S2 consists of 6 feature maps with the size of 14×14, and sub-sampling layer is obtained by sampling from layer C1, and the 2×2 neighborhood of layer C1 is connected with each neuron in the feature map.The C3 layer is the convolutional layer, which is composed of 16 10×10 feature maps, and several feature maps of the S2 layer are connected with each neuron. The S4 layer is the secondary sampling layer, and a 2×2 neighborhood of the C3 layer is connected to the neurons of the feature map. Layer C5 is the convolution layer, which is composed of 120 feature graphs. Layer F6 is a hidden layer of 84 neurons, fully connected to C5. Finally, the output layer consists of 10 neurons, consisting of Radial Basis Function (RBF) units, with each neuron in the output layer corresponding to a character class.

LeNet-5 is a typical example of a CNN model, which is widely used for image recognition tasks. Its success lies in its ability to reduce the number of parameters by using sparse interactions and weight sharing, and in achieving invariance to displacement, scaling, and distortion recognition.

B. Improvement of Convolution Neural Network

The first character of the license plate in China is generally Chinese characters, of which there are a total of 31 characters (mainly referring to the abbreviations of each province, excluding special vehicles). However, LeNet-5 is mainly designed to recognize hand-written digital images and can only classify into 10 categories, which are digits from 0-9. Therefore, there are certain limitations when using LeNet-5 for license plate recognition. According to the characteristics of license plate character image, the structure of LeNet-5 can be optimized, which can not only meet the requirements of recognition rate, but also reduce the recognition time. Therefore, this paper proposes an improved LeNet-5 model to recognize license plate characters. The improved CNN model structure includes: 2 convolution layers, 2 pooling layers and 2 fully connected layers. The convolution layer uses Relu activation function, and the pooling layer uses maximum pooling to reduce dimensions. The entire network structure is simply expressed in Figure 2.

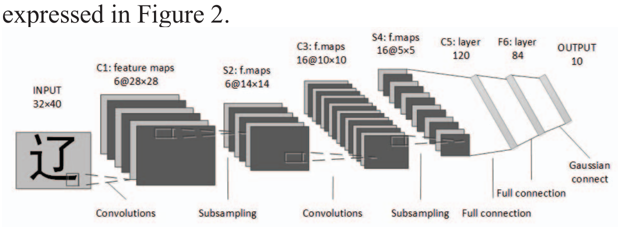

Figure 2. IMPLeNet-5 network architecture.

The layers of IMPLeNet-5 are shown in Figure 3:

1) Input: A license plate image with a size of 32×40 pixels.

2) C1: 32 convolution kernels with a size of 5×5 are applied to the input image. After the C1 convolution layer, 32 feature images with a size of (32-5+1) × (40-5+1) pixels, or 28×36, are generated.

3) S2: A pooling layer with a window size of 2×2 and a stride of 2 is applied to the output of C1. Similar to C1, 32 feature images are obtained, but with a reduced size of (28/2) × (36/2) pixels, or 14×18.

4) C3: 64 convolution kernels with a size of 5×5 are applied to the output of S2, resulting in 64 feature maps with a size of (14-5+1) × (18-5+1) pixels, or 10×14.

5) S4: A pooling layer with a window size of 2×2 and a stride of 2 is applied to the output of C3. Similar to S2, 64 feature images are obtained, but with a reduced size of 5×7 pixels.

6) F5: The fully connected layer connects the feature maps from S4 into a vector with 5×7×64=2240 neurons.

7) Output layer: This layer is a Softmax regression layer that contains neurons used to classify the input into 31 categories of Chinese characters and 34 categories of numeric letters (excluding I and O).

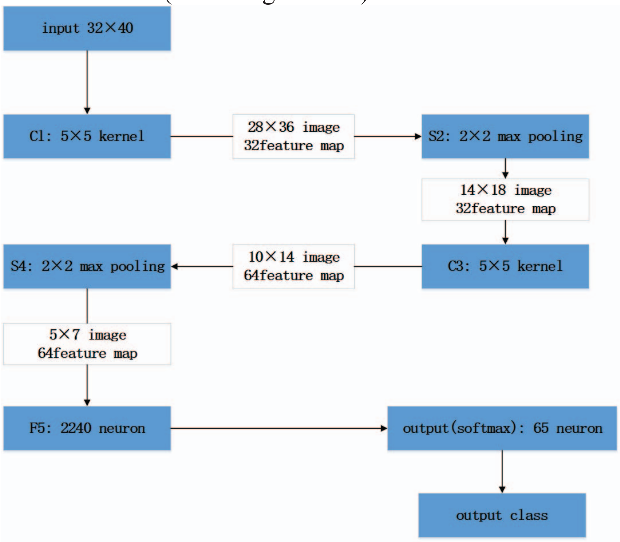

Figure 3. IMPLeNet-5 classification process.

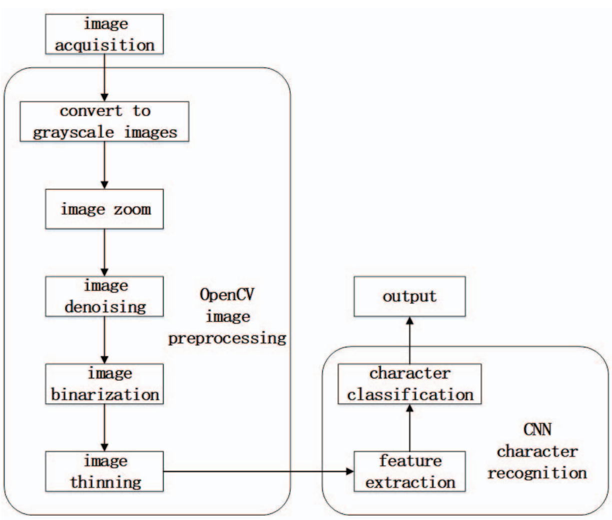

Figure 4. License plate character recognition method process based on IMPLeNet-5n.

C. *License Plate Recognition Based on Convolution Neural Network*

1) License plate character recognition process: The model proposed in this paper consists of two steps: the first part involves preprocessing the license plate character image. The second part involves training the IMPLeNet-5 classification model with the training set, and then feeding the character image of the license plate to be recognized into the trained model for classification and recognition. The overall recognition process is illustrated in Figure 4.

2) License plate character image preprocessing: The license plate character images used in this study were obtained from a high-speed intersection monitoring system that captures front images of passing vehicles. The license plate character images were then cropped from the car image. However, due to external environmental conditions such as weather and illumination, different shooting distances and angles, and varying degrees of license plate contamination, the cropped character images differed in size, color, and clarity, and had unremoved border nails. The original character images are depicted in Figure 5, and are unsuitable for training and recognition in IMPLeNet-5. To facilitate recognition, the original color images were preprocessed to extract essential shapes and structures of license plate characters, while eliminating unwanted background and noise.

To preprocess the original character image, we follow these steps:

- Convert the original color character image (shown in Figure 5) to a grayscale image (shown in Figure 6).
- Resize the grayscale image to a standardized size of 32×40. Since license plate characters may contain Chinese characters with complex structures, the image should not be too small (shown in Figure 7).
- Remove noise points in the grayscale image to improve clarity (shown in Figure 8).
- Convert the image to a binary graph and extract the characters from the background (shown in Figure 9).
- After binarization, some characters may have "black holes". To remove these holes, we perform a closed operation. We then erode and thin the image to enhance its clarity (shown in Figure 10)

.

Figure 5.　color original images.

Figure 6.　Grayscale original images.

Figure 7.　Grayscale image after uniform size.

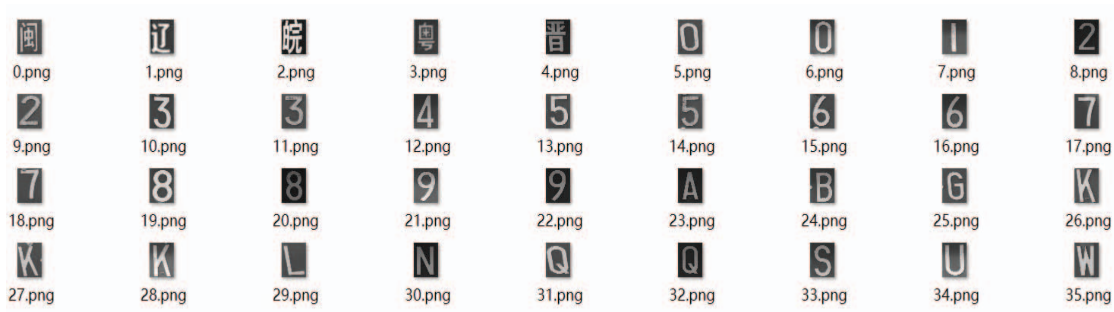

Figure 8.　Grayscale image denoising.

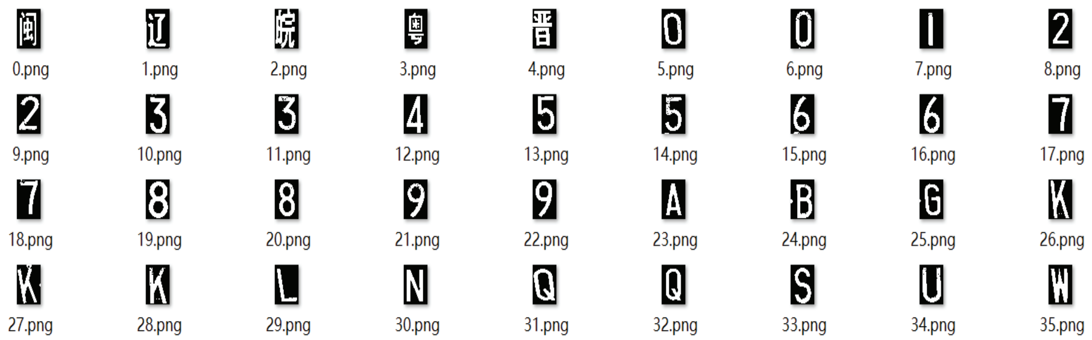

Figure 9. Binarization of grayscale image.

Figure 10. Character thinning.

III. EXPERIMENTAL STUDY

The experiments for this paper were conducted on a 64-bit operating system running on the Windows 10 platform, using an Intel (R) Core (TM) i3-8100 CPU with a main frequency of 3.6GHz and 4GB of memory. The DeepLearnToolbox-master deep learning algorithm software package was used, and the experimental code was programmed and implemented using Matlab R2020a.

The IMPLeNet-5 network was trained and tested on both Chinese characters and alphanumeric characters found on license plates. There were 31 types of Chinese characters, which have relatively complex structures, including 4000 training set images and 340 test set images, and 34 types of letters with relatively simple structures, including 4285 training set images and 340 test set images. Each type of data had about 125 training sets and 10 test sets, resulting in a balanced and orderly dataset that is conducive to classification training. Before training, all images were clipped and preprocessed using the methods described in Section 2.3 of this paper, resulting in character images with a size of 32×40.

A. Experimental Procedure

The experimental procedure is illustrated in Figure 11 and consists of the following steps:

- Classify the images in the training set into 34 types of Chinese character image data sets and 31 types of alphanumeric character data sets.
- Label each type of data and convert it into a unique binary code.
- Extract features from the license plate characters and train the CNN model using the training set.
- Check whether the training reaches the specified number of iterations or the error limit. If so, proceed to the next step; otherwise, return to Step 3.
- Stop training and save the model. The prediction accuracy and recognition time are obtained.

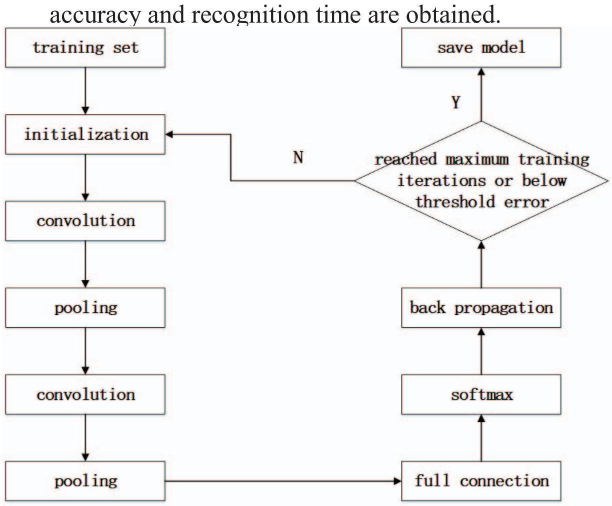

Figure 11. Model training process.

B. Experimental Design

In the simulation experiment, it is assumed that batch_size is 128 pieces, and the learning rate is 0.1.Training was stopped once the training accuracy exceeded 99%, and the misidentification of certain license plate characters was analyzed as shown in Figure 12, where the characters 'Yue' and '0' were mistakenly identified as 'Y' and 'U'. Upon analysis, it was found that the collected original images were not clear and the image obtained after preprocessing differed significantly from the original character, resulting in recognition errors.

(a) characters ‘Yue’ (b) characters ‘0’

Figure 12. Recognize the wrong license plate character.

1) Selection of Epochs: With the increase of epoch, the corresponding weight update times in the neural network model are also increasing. Generally, if epoch is too small, it is easy to cause underfitting, while if epoch is too large, it is easy to cause overfitting. Therefore, appropriate iteration times are crucial for experimental simulation.Based on the comprehensive consideration of training accuracy, prediction accuracy, and time consumption, the optimal number of iterations for this experiment is 40, as shown in Figure 13. The experimental data in Figure 14 shows that the accuracy increases with the increase of iterations, but when the epoch value increases to a certain extent, the prediction accuracy decreases, overfitting occurs, and the model takes a longer time to run. Thus, the best number of iterations needs to be selected by comprehensive consideration.

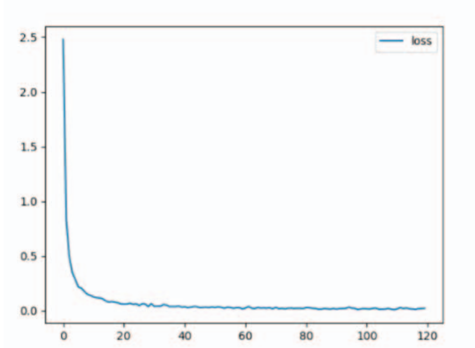

Figure 13. Change of loss function

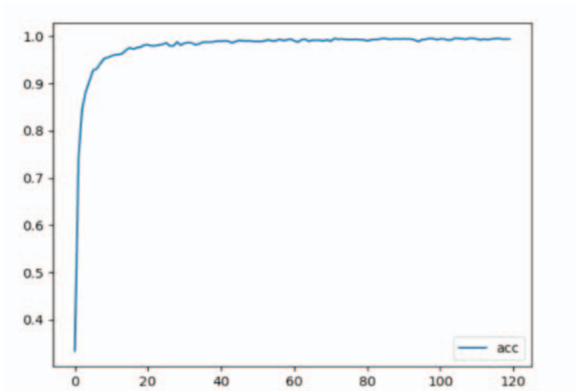

Figure 14. Change of recognition accuracy

2) Selection of Activation Functions and Optimizers: The activation function plays a crucial role in enabling CNN to model nonlinearity. There are three commonly used activation functions: sigmoid, tanh and ReLU. During the training process, CNN needs to update network parameters continuously through back-propagation to optimize the network. The optimization process requires the use of an optimizer to adjust and update the weights. In this paper, we compare and analyze two common optimization methods: the gradient descent method (GD) and the Adam optimizer. Simulation results are presented in the table below:

TABLE I. COMPARISON OF TWO OPTIMIZERS UNDER SIGMOID FUNCTION

Activation function	optimizer	epoch	Batch size	Training Accuracy	Recognition accuracy	Model Training Time
sigmoid	Adam	40	128	92.02%	86.14%	530.7s
	GD			7.83%	3.45%	522.3s

TABLE II. EXPERIMENTAL COMPARISON OF TWO OPTIMIZERS UNDER TANH FUNCTION

Activation function	optimizer	epoch	Batch size	Training Accuracy	Recognition accuracy	Model Training Time
tanh	Adam	40	128	98.14%	97.87%	523.4s
	GD			87.50%	78.60%	500.5s

TABLE III. EXPERIMENTAL COMPARISON OF TWO OPTIMIZERS UNDER RELU FUNCTION

Activation function	optimizer	epoch	Batch size	Training Accuracy	Recognition accuracy	Model Training Time
ReLU	Adam	40	128	99.57%	99.41%	526.9s
	GD			93.10%	89.71%	533.3s

After conducting comparative experiments as presented in the three tables above, it can be concluded that the Adam optimizer outperforms the gradient descent GD in terms of prediction accuracy, regardless of the activation function used. Furthermore, the Adam optimizer shows an advantage in terms of training rate under the ReLU function.

C. Comparison Experiment

To evaluate the performance of the IMPLeNet-5 license plate recognition system, we conducted a comparative experiment with the commonly used BP network under the same experimental environment. The IMPLeNet-5 network achieved a training accuracy of 99.57% with a training time of 526.9s. The recognition accuracy was 99.41%, and the average recognition time of a single character was 0.0005s, which is sufficient for practical applications. Simulation results demonstrate that the recognition accuracy of the IMPLeNet-5 network is approximately 5.5% higher than that of the BP network. In terms of time performance, the average recognition efficiency of a single character in the IMPLeNet-5 network is about 36.71% higher than that of the BP network.

TABLE IV. COMPARATIVE ANALYSIS OF EXPERIMENTAL RESULTS

Algorithm	Recognition Accuracy /%	Average Recognition Time of Single Character /s
IMPLeNet-5	99.41	0.0005
BP neural network	94.24	0.00079

Through comparative experiments, it is concluded that IMPLeNet-5 network model proposed in this paper has strong applicability, and the accuracy of IMPLeNet-5 license plate recognition can reach more than 99%. This is mainly due to the excellent feature extraction ability of the IMPLeNet-5 network model. Moreover, the font of the license plate characters themselves is neat and simple in shape and structure, despite great differences in the original images due to environmental factors. After preprocessing, the character images are uniform and most of them are clear and recognizable, without the diverse and complex shape and structure of handwritten characters. Therefore, the simpler structure of the IMPLeNet-5 network model is advantageous in achieving better recognition results.

IV. CONCLUSION

In this paper, we have implemented a IMPLeNet-5 license plate character classification and recognition algorithm by improving the LeNet-5 model. Through experiments, we have compared and analyzed the parameter variables of the model, such as the number of iterations, activation function and optimizer, and determined the optimal configuration for the model. The experimental results demonstrate that the IMPLeNet-5 model achieves a good recognition effect on the test set and outperforms the BP neural network in terms of recognition accuracy and speed. However, the recognition accuracy of color original images without preprocessing and images with large distortion still needs improvement, which requires further research.

REFERENCES

[1] Angara S, Robinson M. License plate character recognition using binarization and convolutional neural networks[C]//Advances in Computer Vision: Proceedings of the 2019 Computer Vision Conference (CVC), Volume 1 1. Springer International Publishing, 2020: 272-283.

[2] Yong-jie M A, Shi-sheng C, Yun-ting M A, et al. Review of convolutional neural network and its application in intelligent transportation system[J]. Journal of Traffic and Transportation Engineering, 2021, 21(4): 48-71.

[3] Arslan R U, İNCETAŞ M O, DİKİCİ S. Effects of character recognition with Shell histogram method using plate characters[J]. Politeknik Dergisi, 2019, 22(4): 1093-1099.

[4] Zhao Z H, Yang S P, Ma Z Q. License plate character recognition based on convolutional neural network LeNet-5[J]. Journal of System Simulation, 2010, 22(3): 638-641.

[5] Asif M, Bin Ahmad M, Mushtaq S, et al. Long multi-digit number recognition from images empowered by deep convolutional neural networks[J]. The Computer Journal, 2022, 65(10): 2815-2827.

[6] Pham V H, Dinh P Q, Nguyen V H. CNN-based character recognition for license plate recognition system[C]//Intelligent Information and Database Systems: 10th Asian Conference, ACIIDS 2018, Dong Hoi City, Vietnam, March 19-21, 2018, Proceedings, Part II 10. Springer International Publishing, 2018: 594-603.

[7] DONG J, ZHENG B, YANG Z. Character recognition of license plate based on convolution neural network[J]. Journal of Computer Applications, 2017, 37(7): 2014.

[8] Liu Y, Huang H, Cao J, et al. Convolutional neural networks-based intelligent recognition of Chinese license plates[J]. Soft Computing, 2018, 22(7): 2403-2419.

[9] Ma L, Zhang Y. Research on vehicle license plate recognition technology based on deep convolutional neural networks[J]. Microprocessors and Microsystems, 2021, 82: 103932.

[10] Huang Q , Cai Z , Lan T . A Single Neural Network for Mixed Style License Plate Detection and Recognition[J]. IEEE Access, 2021, PP(99):1-1..

[11] Alvar S R , Uyanik K , Baji I V . License Plate Privacy in Collaborative Visual Analysis of Traffic Scenes[C]// 2022.

[12] Taleb H , Li Z , Yuan C , et al. An Effective Method for Yemeni License Plate Recognition Based on Deep Neural Networks[J]. Springer, Cham, 2022.

[13] Wang Y , Bian Z P , Zhou Y , et al. Rethinking and Designing a High-Performing Automatic License Plate Recognition Approach[J]. IEEE transactions on intelligent transportation systems, 2022(7):23.