Proceedings of CONF-MLA 2025 Symposium: Intelligent Systems and Automation: AI Models, IoT, and Robotic Algorithms
DOI: 10.54254/2755-2721/2025.LD24514

Overview of Deep Learning Based License Plate Detection System

Yanhao Li

School of Microelectronics, Xidian University, Xi'an, China
24149100194@stu.xidian.edu.cn

Abstract. This paper conducts an in - depth exploration of the license plate detection system based on deep learning, providing a comprehensive analysis of its core principles, design architecture, implementation process, and performance evaluation. At the core principle level, it deeply analyzes how deep learning accurately recognizes license plates. By leveraging technologies such as convolutional neural networks, it can effectively extract license plate features from images. Regarding the design architecture, it elaborates on the functions of each component and their collaborative working mechanisms to achieve efficient detection. The implementation process encompasses various links, including data collection, annotation, model training, and optimization. The performance evaluation is carried out from multiple dimensions such as detection accuracy, speed, and adaptability to complex environments, comprehensively measuring the system's performance. Through comprehensive research, deep learning demonstrates significant advantages in license plate detection, offering innovative solutions for intelligent transportation and security fields. Meanwhile, this paper also looks ahead to the future development directions of this technology, such as improving efficiency and expanding application scenarios.

Keywords: : deep learning, license plate detection, intelligent transportation, computer vision

1. Introduction

1.1. Research background and significance

Intelligent transportation system is relying on 5G, Internet of things and other technologies to achieve multi-dimensional upgrading, and its market size is expected to exceed 200 billion yuan in 2025. At present, intelligent signal lights dynamically optimize timing to alleviate congestion. Meanwhile, the ETC penetration rate has exceeded 90%, significantly improving traffic efficiency, intelligent monitoring system realizes traffic monitoring and accurate identification of illegal behaviors, and vehicle networking technology promotes the evolution of vehicle infrastructure coordination to automatic driving. The integration of these technologies has built a real-time perception and intelligent decision-making traffic management system, which has become the core support for the digital transformation of urban governance.

Amid these technological advancements, license plate detection serves as the perception center of intelligent transportation, directly affecting the operation efficiency of key scenes such as ETC deduction, illegal capture, and parking lot management. As the perception center of intelligent transportation, license plate detection directly affects the operation efficiency of key scenes such as ETC deduction, illegal capture, and parking lot management. Its accurate identification ability not only guarantees the fairness of traffic law enforcement (the misjudgment rate is reduced by 40 %), but also supports the identity authentication requirements of hundreds of millions of vehicles per day. In the field of security, the license plate recognition system assists the case detection efficiency by 35 % through vehicle trajectory tracking, and builds an active security barrier in airports, parks adaptability of license plate format in multiple provinces, and weak robustness of occlusion / defacement scenes. However, breaking through these technical barriers can produce significant value, including improving the efficiency of traffic law enforcement by more than 30% and reducing the queuing time at toll stations by 15%, and enhancing the recognition ability of security systems for licensed vehicles and vehicles involved. These improvements will accelerate the evolution towards fully automated intelligent traffic management systems while establishing a more precise vehicle database, which is essential for smart city development.

2. Relevant technical basis

2.1. Overview of deep learning training algorithms

Training algorithms are key factors in improving the performance of deep learning-based license plate detection systems. Among various training methods, gradient descent-based algorithms play a particularly important role in deep learning training.

Stochastic gradient descent (SGD) randomly selects a small number of samples each time to calculate the gradient to update the model parameters. The computational complexity is greatly reduced. It is suitable for large-scale data sets and is widely used in a variety of machine learning tasks. However, it has the problems of slow convergence speed and difficult learning rate selection[1,2].

As an improvement over SGD, Mini-Batch stochastic gradient descent (Mini-Batch and other scenes.

Traditional methods, which rely on artificial feature extraction, face three major bottlenecks, with the primary one being that detection accuracy drops below 65% in complex illumination environments, poor SGD) calculates gradients using small batches of samples each time, which balances the computational efficiency and convergence stability, and can accelerate model training. However, the selection of hyperparameters depends on experience and has poor adaptability to special data sets[2,3].

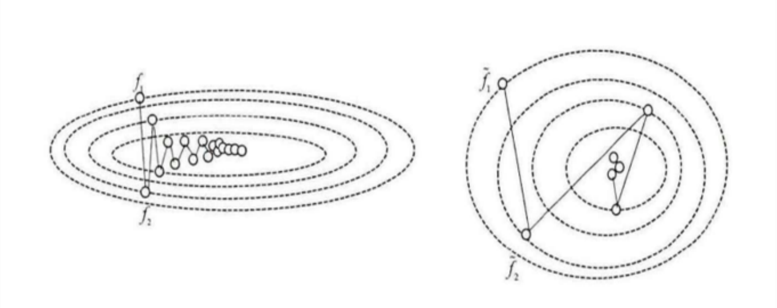

Figure 1: Number of iterations for gradient descent method[1]

The figure 1 shows that the horizontal axis in the graph represents the number of iterations, and the vertical axis is likely to be the loss value or other measurement indicators. Curves of different colors represent different gradient descent methods, such as stochastic gradient descent and batch gradient descent. From the trend of the curves, we can observe how the indicators of different methods change as the number of iterations increases, which allows for an intuitive comparison of their convergence speeds and effects.

By introducing the concept of momentum, the momentum algorithm imitates the inertia of object motion and accumulates the update direction of historical gradient optimization parameters, which can accelerate convergence, reduce oscillation and help the algorithm jump out of local optimum. Common momentum algorithms such as SGDM and Adam are widely used in deep learning model training. However, the momentum algorithm also has the problem of hyper-parameter dependence, and may be over-corrected, increasing the computational complexity[4,5].

The Nesterov accelerated gradient (NAG) algorithm features fast convergence and stable iteration, supported by a solid theoretical foundation. The convergence rate is better than the traditional gradient descent method when dealing with convex functions, but it is sensitive to parameters and has high computational complexity[6,7].

The adaptive learning rate algorithm can automatically adjust the step size according to the parameter gradient and improve the training effect. Adagrad introduces a diagonal matrix in the step denominator. This algorithm then accumulates past square gradient information using arithmetic averaging. It has significant advantages in dealing with sparse data. It can adaptively adjust the step size and alleviate the hyper-parameter dependence, but the step size is monotonically decreasing, which may lead to slow convergence in the later stage and high computational complexity[8]. RMSProp uses exponential weighted average to deal with historical gradients, adaptively adjusts the learning rate, and effectively alleviates the problem of premature attenuation of learning rate. The calculation process is relatively simple, but the theoretical basis is relatively weak, and the ability to deal with complex non-convex problems is limited[9,10]. Adam combines the advantages of both momentum and RMSProp algorithms, while automatically adjusting learning rates based on first-order and second-order moment estimations of the gradient. It has high computational efficiency and low storage requirements. It performs well in complex deep learning tasks, but it is sensitive to hyperparameters and has insufficient theoretical convergence[9].

2.2. Neural network architecture

Artificial neural networks, which simulate biological neural networks, consist of three main components: input layer, hidden layer and output layer. The input layer receives the data, the hidden layer performs nonlinear transformation to extract features, and the output layer produces the final result. The weight is adjusted by the back propagation algorithm to achieve data classification, prediction and other tasks.

Convolutional neural network (CNN) is an important architecture for processing data such as images in deep learning. The convolution layer uses the convolution kernel to slide convolution on the input data to extract local features such as edges and textures. Parameter sharing reduces the amount of calculation and retains spatial structure information. The pooling layer follows the convolution layer, and the dimension is reduced by maximum pooling or average pooling, which reduces the amount of calculation and enhances the robustness of features. At the end of the network, the fully connected layer integrates the features of the previous layer and maps high-dimensional features to low-dimensional space for classification or regression tasks.

In image processing, CNN's convolutional layers play a crucial role by progressively extracting features from shallow to deep levels during feature extraction and representation learning. In terms of text data processing, word vector models such as Word2Vec and BERT map words into vectors to capture semantic information. Representation learning allows the model to automatically learn effective feature representations by minimizing the loss function to improve task performance[11].

2.3. Target detection technology

In the region-based target detection algorithm, R-CNN is an important milestone in the application of deep learning to target detection. It uses selective search to generate candidate regions and employs CNN for feature extraction. The detection is then completed through SVM classification and bounding box regression, establishing a new paradigm, but the calculation is large, the training is complex, and the speed is slow. Fast R-CNN is improved on the basis of R-CNN. The RoI Pooling layer is introduced to share the feature map, and the Softmax classifier is used to simplify the training, which improves the detection speed and accuracy. Despite these improvements, the selective search method remains slow in generating candidate regions, and its performance in detecting small targets is suboptimal. Faster R-CNN proposes RPN network to generate candidate regions, which improves the detection speed and accuracy, and has good adaptability. However, the training process is cumbersome, the parameter adjustment is difficult, and there is a problem of missed detection[11].

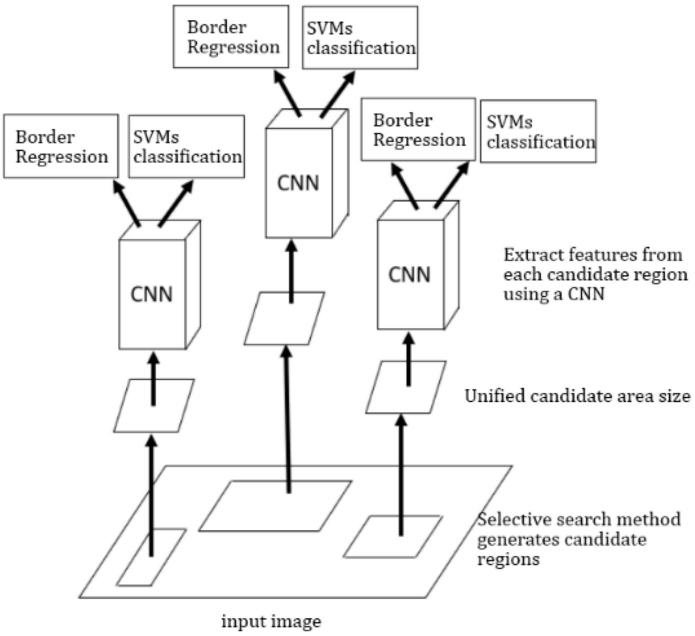
Border Regression
SVMs classification
Border Regression
SVMs classification
Border Regression
SVMs classification
CNN
CNN
CNN
Extract features from each candidate region using a CNN
Unified candidate area size
Selective search method generates candidate regions
input image

Figure 2: R-CNN basic flowchart[11]

As the figure shows, this diagram illustrates the basic flow of R - CNN (Region - based Convolutional Neural Network). Firstly, the selective search method is applied to the input image to generate multiple candidate regions, and the sizes of these candidate regions are unified. Subsequently, a Convolutional Neural Network (CNN) is used to extract features from each candidate region. Finally, the extracted features are utilized for Support Vector Machine (SVMs) classification and bounding box regression to determine the category and location of the target object.

In the single-stage target detection algorithm, the YOLO series regards target detection as a regression problem and directly processes the entire image. The detection speed is fast, the structure is simple, and the adaptability is strong, but the small target detection effect is poor, and the complexity of some versions of the model is high[12]. SSD directly detects targets on multi-scale feature maps. It has fast detection speed and can take into account different sizes of targets. The structure is relatively simple, but the training is complex and sensitive to hyperparameters. In complex scenes, the VGG16-based backbone network shows limited capability in feature extraction[12].

3. Comparison of related technologies

3.1. Comparison of deep learning training algorithms

In the comparison of deep learning training algorithms, various algorithms based on gradient descent have their own advantages and disadvantages. SGD has low computational complexity and strong versatility, but it converges slowly and the learning rate is difficult to adjust. Mini-Batch SGD has stable convergence and high computational efficiency, but the selection of hyperparameters depends on experience. The momentum algorithm accelerates convergence and reduces oscillation, but it is hyper-parameter dependent and may be over-corrected. NAG algorithm has fast convergence, stable iteration and perfect theory, but it is sensitive to parameters and complicated to calculate[1,3].

The structure could be: "Adagrad adaptively adjusts the step size to alleviate hyper-parameter dependence. However, it suffers from decreasing step size and high computational complexity." RMSProp adaptively adjusts the learning rate and alleviates the premature attenuation of the learning rate, but the theoretical basis is weak. Adam automatically adjusts the learning rate, has high computational efficiency, and integrates the advantages of multiple algorithms. However, it is sensitive to hyperparameters and lacks theoretical convergence[8,9].

3.2. Comparison of target detection techniques

In the region-based target detection algorithm, R-CNN is the first to start deep learning detection, but the calculation and memory requirements are large and the training is complex. Fast R-CNN reduces the amount of calculation, simplifies training, and improves accuracy, but generates candidate regions slowly and detects small targets poorly. Faster R-CNN improves speed and accuracy with good adaptability, but suffers from complicated training process and potential object misdetection[11].

In the single-stage target detection algorithm, the YOLO series has fast detection speed, simple structure and strong adaptability, but the small target detection is poor and the model is complex. SSD has fast detection speed, takes into account different sizes of targets, and has a simple structure, but it has complex training, is sensitive to hyperparameters, and has low detection accuracy in complex scenes[12].

These technologies show different performance characteristics, which helps in selecting suitable algorithms and architectures for deep learning-based license plate detection systems. In practical applications, it is necessary to comprehensively consider the requirements of the system on detection speed, accuracy, computing resources, etc., and weigh the advantages and disadvantages of each technology to achieve efficient and accurate license plate detection. At the same time, with the continuous development of technology, continuous attention to the improvement and innovation of algorithms will help to further improve the performance of license plate detection systems and meet the growing needs of intelligent transportation and security.

4. Conclusions and prospects

4.1. Summary of research work

When designing a deep learning-based license plate detection system, it is essential to select an appropriate detection algorithm. The YOLO series, known for its fast detection speed, is widely used for meeting real-time requirements. There is also Faster R-CNN algorithm, which performs well in detection accuracy. After the algorithm is selected, the matching network structure is further determined, such as the combination of Faster R-CNN candidate region generation network and Fast R-CNN. Then, the model is trained with the previously labeled data, and the parameters such as learning rate and optimizer are continuously adjusted during the training process. Through multiple iterative training, the detection accuracy of the model for license plates is gradually improved.

4.2. Suggestions for future research

At present, deep learning training algorithms have limitations such as large demand for computing resources, easy overfitting, and long training time. In this regard, I suggest using model compression techniques, such as pruning and quantization, to reduce model parameters and calculation, and reduce hardware requirements. To address overfitting, various data augmentation methods can be employed, including Mixup sample fusion. Additionally, regularization techniques such as L1 and L2 can be implemented to constrain parameter values. Aiming at the problem of long training time, the network structure is optimized and lightweight networks, such as MobileNet series, are introduced. It can also improve the training algorithm, such as using the adaptive learning rate algorithm, dynamically adjusting the learning rate according to the training situation, accelerating the convergence speed and improving the training efficiency.

References

[1] Zhang Zhuan, Research on gradient pretreatment stochastic gradient descent algorithm, Master thesis, Xidian university, 2021
[2] Tang Chun, Research on distributed stochastic gradient descent algorithm, Master thesis, university of electronic science and technology of China, 2018
[3] Pan Lei, The application of small batch stochastic gradient descent algorithm in seismic imaging, PhD thesis, university of science and technology of China, 2017
[4] Sun Weiguo, Study on the design and application of fractional gradient method based on momentum information, Master thesis, Lanzhou university, 2023
[5] An improved adaptive momentum gradient descent algorithm, Journal of Huazhong University of Science and Technology (Natural Science Edition), 2022, 51, 137-143
[6] Li Chunkai, Research on accelerated gradient algorithms, Master thesis, Beijing university of posts and telecommunications, 2021
[7] Lian Yongqiang, Research on three-step accelerated gradient algorithm in deep learning, Master thesis, east China normal university, 2019
[8] Adaptive NAG method based on AdaGrad and its optimal individual convergence, Journal of software, 2022, 33, 1231-1243
[9] Zhang Tianze, Li Yuanxiang, Xiang Zhenglong, Li Mengying, Particle swarm optimization algorithm based on RMSprop, Computer Engineering and Design, 2021, 42, 642-647

[10] Zhang Hui, Research and improvement of optimization algorithm in deep learning, Master thesis, Beijing university of posts and telecommunications, 2017

[11] Zhang Bicheng, Target detection and recognition algorithm based on regional convolutional neural network, PhD thesis, university of electronic science and technology of China, 2020

[12] Zhu Hao, Zhou Shunyong, Liu Xuezeng, Yalan Li Sicheng, A survey of single-stage object detection algorithms based on deep learning, Industrial Control Computer, 2023, 36, 101-103