PENINSULA PUBLISHING LLC

EDRAAK
Vol. (2025), 2025, pp. 111–118
ISSN: 3078-8412

Research Article

Detecting Similarity License Plate Vehicle License Via Using Deep CNNs in Complex Surroundings

Jinan J. Abdulkareem[1, *], Ahmed Sameer Abdulmohsin[2], Hanan J. Abdulkareem[2]

[1] Dept. Technical Engineering College, Al-Farahidi University, Baghdad, Iraq

[2] dept. Medical Instrumentation, Al-Esraa University College, Baghdad, Iraq

ARTICLE INFO

Article History
Received 22 Jun 2025
Accepted: 10 Aug 2025
Revised 12 Sep 2025
Published 1 Oct 2025

Keywords
ML, DCNN,
Deep Learning,
Plate image,
Convolutional Neural Networks,
Segmentation and Image.

ABSTRACT

As our society has developed, cars on the road have increased. Manual license plate recognition is challenging since it is significantly slower in real-time than when performed by a machine. The bulk of license plate recognition systems utilized morphological image processing until recently. The detection rate falls drastically under challenging situations (when the license plate is blurry, for example). We suggest a novel approach to license plate recognition as a solution to this problem. It was positioning a license plate. The study presented a deep learning-based vehicle classifier to localize license plates and numbers. Instead of bounding rectangles, the classifier outputs determining quadrilaterals for vehicle number estimate with license plate localization. The suggested DCNN model training began with an initial weight that had been trained without the classification head, resulting in a total training iteration of roughly 9000 for the DCNN model, including transfer learning. The DCNN model could start at an intelligent point and optimize all functional heads. The DCNN obtained 97.5% classification accuracy in "vehicle number estimate and license plate localization.

1. INTRODUCTION

License plate readers, one of the most fundamental components of today's advanced transportation networks, are ubiquitous. Various technologies, including digital image processing [1], pattern recognition, deep learning, and others, will be used to extract the license plate numbers from the vehicle photos or videos. Using post-processing methods, it can manage car park fees, measure road traffic control indices, determine the location of vehicles, provide protections for car theft, monitor top speed, assure compliance with stoplights, and collect tolls, among other things. Maintaining traffic safety and urban security, minimizing traffic bottlenecks, plus implementing automatic traffic control are all important from a practical standpoint. Character recognition and license plate reading are its two key subcomponents. The license plate data is retrieved to use a CNN [2] as well as a classifier for each character is constructed. Compared to those acquired using the usual recognition method, experimental results demonstrate a significant increase in precision. We conclude the challenges of vehicle classification and number estimation that will be solved with license plate localization:

1. Inability to estimate slanted and oblique vehicle images in real time.
2. dealing with more generic vehicle classification and location estimation, there is a loss of contextual information.
3. possible to produce training data from a vehicle image and average vehicle categorization ground truth
4. It possible to use a deep convolutional neural network (DCNN) to automate vehicle classification and location estimation measurements using real-time images
5. factors should be considered in the future if the research topic is the same

The answer algorithm the research would employ an anchor-free DCNN design for our head network; no default anchor boxes are required in the region proposal procedure. In contrast to DCNN-based vehicle detectors, no heuristic judgements are necessary during training; all that is required is for our model to acquire detection capacity spontaneously. Since our model individually regresses the four vertices of a quadrilateral, we can also calculate the exact vertices of a license plate following planar rectification by perspective transform. If the model does not converge, it is difficult to determine which head design is flawed. In our design strategy, we introduced each functional head one by one, ensuring that the notion of a

*Corresponding author email: jenan.jabar@alfarahidiuc.edu.iq
DOI: https://doi.org/10.70470/EDRAAK/2025/014

single head worked before adding the rest. Our research focuses on vehicle number estimation with license plate localization following vehicle categorization to precisely determine the automobile's position in single or multiple vehicles scenarios. License plate localization, automobile attitude, and license plate side (front or rear) are determined for improved vehicle categorization. With license plate localization, our proposed solution can resolve the problem statement's missing automobile numbers. Our principal contributions are:

- A DCNN-based network was built for vehicle classification and number estimating in real-world scenarios.
- Vehicle number information will be provided in addition to estimation and localization.
- Our model will recognize and classify the vehicle position's front and back ends.

In order to train our model, we gathered additional data from the literature, which included bounding boxes marked by hand from the front and back. A unique anchor-free method is described in the methodology and is used in the classification process; this method eliminates the need for a manually set anchor-box size and may be used as a template in future vehicle detection and identification applications.

2. BACKGROUND

CNN's work in recent years have led to substantial improvements in numerous domains, including speech recognition, facial recognition, general object detection, motion, natural language processing, and even brain wave analysis. Convolutional neural networks (CNNs) are distinguished from typical neural networks by its feature extractor, which comprises of a convolution and subsampling layer. Each neuron in the convolution of a CNN is only connected to a subset of the neurons in the following layer.

In a CNN convolution, there are typically several feature plane [3] (Feature Map) (Feature Map). Multiple neurons are organized in rows and columns on each feature plane. The cells on the same feature plane share weights, in addition to the weights that are being exchanged here. A value is the kernel of inversion. Generally, convolution kernels are initiated in the form of a randomized fractional matrix. During network training, the convolution kernel will be trained to achieve appropriate weights. The clear advantage of shared weights s(convolution cores) is the reduction of connections across the network's layers while lowering overfitting risk. Sub-sampling is frequently referred to as pooling [4] (pooling), of which there are normally two variants: smean sub-sampling (mean pooling) and maximum sub-sampling (max pooling) (max pooling). The maximum post is chosen as the maximum pooling in this paper. Subsampling can be thought of as a specific convolution method. Convolution and subsampling significantly reduce the model's complexity and decrease its parameters.

Wen-Min L [5] presented the image capture and placing of license plates in a vehicle number plate recognition system, achieving a placement accuracy of 96.4 percent. This same author segments the generated license plates and uses the SVM to classify the segmented characters, a technique similar to Wang L et al [6], who discovered the license plate by searching for a set of walk pixels in the grayscale image of the surrounding pixels in the image. The overall rate of recognition is approximately 95%.

The findings indicate that while the first layers are effective at seeing small cars, they are deficient in their capacity to recognize and classify vehicle kinds. However, deeper layers are more robust on varied translation, illumination, and transformation of a vehicle, but less capable of recognizing little vehicles. One strategy for dealing with the issue of insufficient spatial information is to intuitively train a single classifier on images of varying scales, or to train many classifiers on images of varying scales, and then combine the results, as shown in [7]. You can see an example of the Image Pyramid technique in the top left corner of the image. In order to train and test the model, we must repeatedly feed the image, which is computationally expensive as mentioned in [8]. Because the standard deep learning pipeline is broken up into four sub-tasks, each of which is related to and dependent on the others, optimizing it is a challenging procedure [9]. We need a comprehensive, end-to-end optimization strategy to take deep learning to the next level. With the advent of deep learning techniques, particularly Convolutional Neural Networks, deep learning has made considerable strides in recent years (CNN). Warped Planar Vehicle Detection Network was proposed by researchers in [10] to locate license plates inside of a single vehicle using DCNN, with the added benefit of obtaining parallelogram bounding boxes in addition to rectangle ones, thanks to the model's acquisition of the affine transform parameters and their application to rectangle ones. Their model can simply convert the license plate back into a rectangle via inverse affine transform after learning the affine transform parameters and applying them to rectangle bounding boxes.

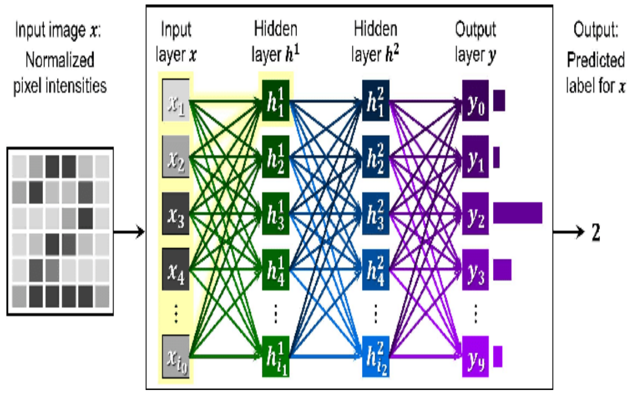


Fig. 1. The depiction of deep learning based DCNN network while learning the digits though hidden layers [10]

To estimate the number of vehicles, the majority of current systems only take into account the view from the front. As explained in [11], the open-source version of commercial software has trouble detecting vehicle number estimates and license plate changes such as a tilted license plate or a license plate belonging to an automobile in an oblique view. Failure in vehicle location estimate and detection is a significant performance barrier for machine learning-based systems because it means that a precise position cannot be understood.

3. METHODOLOGY

The model under consideration is a Deep Convolutional Neural Network (DCNN) (Deep Convolutional Neural Network). DCNN is an unanchored, single-stage deep learning classifier. The complete model is depicted in Figure 1. Prior to transmitting an RGB image through two stacks of the Hourglass Network for backbone feature extraction, its width and height will be scaled by a factor of four. The acquired features will then be utilized by three parallel heads to independently perform localization, region regression, and classification. In one example of our model's output, the license plate region is displayed on the right side of the vehicle, along with the front and rear of the owner's vehicle and its pose.

4. LICENSE PLATE IDENTIFICATION METHOD

Traditionally, license plate recognition required getting a candidate plate via image processing, then revisiting each candidate region to guarantee accurate plate alignment. Using a multi-tag classification strategy, the DCNN is employed for character recognition to produce several tags instantaneously.

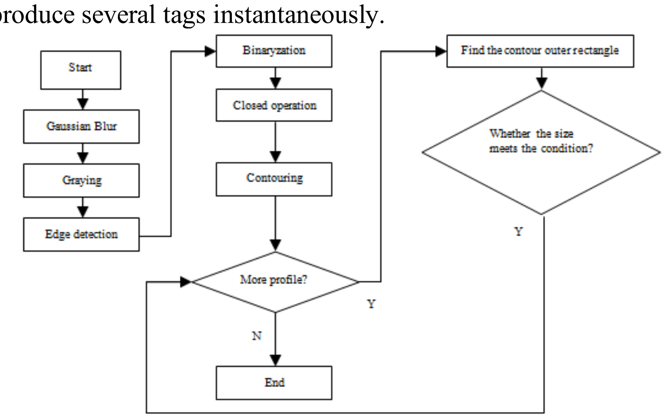


Fig. 2. The License plate position method

Figure 2 depicts the steps of the conventional positioning algorithm. First, a Gaussian blur filter is applied to the raw image. Before moving on to the next degree of sobriety, the goal of this procedure is to eliminate the interference [8]. To obtain the image's first-order lateral direction, we first grayscale the image and then do the sobel operation on it. A closed process is utilized to convert greyscale photos (with 256 potential values per pixel) into binary images in order to locate the outlines of all graphs (with only 1 and 0 values per pixel). This method aims to identify the contour of the entire graph by locating the lowest possible bounding area for the curves and ensuring that it does not meet the criteria.

5. CHARACTER IDENTIFICATION METHOD

As one of the most prevalent deep learning models in the field of picture identification, CNN can automatically learn racist and discriminatory pattern features from a massive dataset. This technique delivers near-accurate human identification with enough data. As a result, we altered the method for use in license plate character recognition, and the resulting rate of recognition was quite high.

6. BUILDING CONNECTIONS OF NETWORK

Figure 3 depicts the text's character recognition CNN architecture.

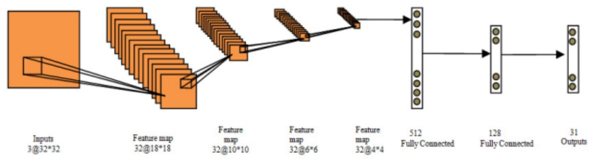


Fig.3. Building of deep convolutional neural network for license plate classification

Deep Convolutional Neural Networks (DCNNs) are multi-stage classifiers that sacrifice speed for performance, with little automobiles bearing the brunt of this trade-off. Because license plates are little vehicles (consider the difference in scale), the feature extraction capability of the backbone network topology is critical, and spatial information must be utilized to prevent missing small vehicle detections. After integrating the Hourglass Network in current one-stage detectors, researchers implemented the DCNN Network as our backbone architecture and discovered that it performed better than a standard DCNN. After validating our anchor-free approach for region regression, we added license number identification heads to the DCNN model. The study taught the newly additional automobile posture heads after loading the backbone architecture weights and license plate heads. After training in the detection of vehicle license plates, we added the ultimate challenge of license plate categorization.

Four convolution, four max pooling, two completely linked levels, and one output layer are depicted in Figure 3. As input, a 32x32x3 pixel image is used. After a convolution layer with a 5*5 convolution kernel and a Max-pooling layer, a feature map is generated. Both feature maps employ a convolution level and a Max-pooling level. The convolution operation is Leaky-Re-LU [9], the loss function is "categorical cross entropy [10]", and the output layer is preceded by a Drop - outs (rate=0.5) [11] layer. The optimizing algorithm (lr = 0.0001) [12] and the output function SoftMax [13].

7. Data groundwork and training

As can be seen in Fig.4 below, the experimental dataset consists of 1,000 images containing license plates generated in version 3.1.1. The first category on a Chinese license plate is Chinese characters, and there are 31 different ones. After that, the first character in the dataset is constructed by assigning a numeric value between 0 and all the other possible values for each Chinese character. The last six digits include both letters and numbers; 24 uppercase letters (except O and I) and 10 digits. In such case, there are 34 categories for a label. This is how individual information labels for license plates are generated. Nine times as many samples from the test set are used in training as from the training set, and 40 iterations are used in total. There is a total of 33 hours invested in this instruction. In the above-described network topology, the data is fed sequentially, and a model for predicting each digit of the license plate is built. In this lab, we use a computer with an Intel(R) Core(TM) i5-3337U CPU running at 1.80 GHz, together with 8 GB of RAM, a 64-bit operating system, and a python3 development environment.

Fig.4. Training data of character identification

8. RESULTS DETAILS OF TRAINING

Ubuntu 16.04 LTS was used for both training and testing, together with an Intel i5-6500 CPU running at 3.20 GHz, a GeForce GTX1080 GPU, and 8GB of RAM. To construct this system, we relied on deep learning libraries like TensorFlow and Keres. All the codes are freely available online. Windows and Python 3.7 users can get the codes as well. On the Kaggle dataset, the results of the classification are displayed as text above the bounding quadrilaterals of a car's front-rear, with Front, Rear, and Unknown for the backdrop class. The SoftMax activation function returns a number and a classification as its output. The license plate's likelihood is located at the base of the enclosing quadrilateral.
The purpose of this part is to discuss our full training approach, including our online data augmentation strategy, transfer learning technique for avoiding unstable learning states, and hyperparameter modification for varying training iteration durations. The Kaggle dataset was used for both training and evaluation by the researchers. Before feeding the training images into the model, a variety of augmentation processes were chosen at random. The research employs 70% of the data for training, 20% for testing, and 10% for validation using a ten-fold cross-validation technique.
The approaches were cycled through at random and used once in each photograph. Online data refreshment maintains the diversity of training data. Researchers introduced random parameters for each augmentation strategy to avoid the model from becoming overtrained too early and to expand the training data set. This implies that, following the application of all procedures, a single original image will be turned into an endless number of augmented images. In addition, the study provides some single-image samples of enriched data.

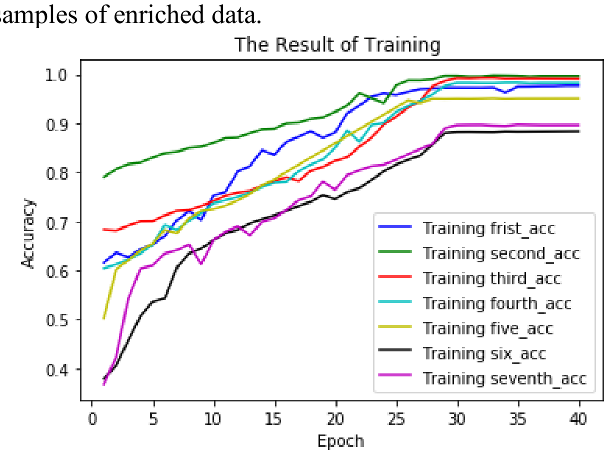
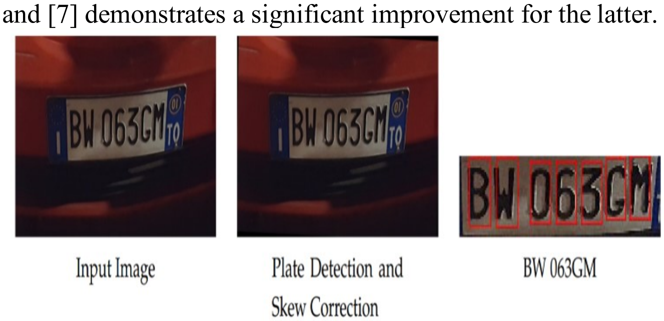


Fig.5. The training and testing accuracy with total 50 number of epochs

The ordinate indicates the precision of the training, whereas the abscissa indicates the total number of repetitions. Given that a Chinese license plate has seven characters, as seen in the diagram, "Training first acc" indicates the accuracy of the first character, "Training second acc" indicates the accuracy of the second character, etc. Each of the seven curves indicates the learning precision of a single bit character. Finally, each character's trained model is in hand. It is evident that the first five characters have a prediction accuracy of greater than 95%, while the last two numbers have an accuracy of greater than 90%. Comparing the effectiveness and precision of the proposed approach to those of license plate recognition algorithms disclosed in the literature [6] and [7] demonstrates a significant improvement for the latter.



(a) Input Image (b) Detection (c) Classification

Fig. 6. The input image, detection, and classification of sampled license plate on vehicle

(a) Input Image (b) Detection (c) Classification

Fig. 7. The input image, detection, and classification of sampled license plate on vehicle

(a) Input Image (b) Detection (c) Classification

Fig.8. The input image, detection, and classification of sampled license plate on vehicle

Some examples of the detection-missing problem associated with vehicle detection-based methods. Using the technique we described, we were able to avoid this issue and locate all license plates inside the image[15][16]. This implies that when the number of vehicles in an image increase, our method becomes more reliable because the overlap condition between vehicles also increases[17]. The research depicts the distribution of these high probability pixels to visualize vehicle number estimation with license plate localization probability for each pixel and the classification capabilities of the model. Our DCNN algorithm can localize the license plate inside a given area and identify all possible pixels for the front and rear of the vehicle.

9. DISCUSSION

This taught us that a well-designed labeling method can have a significant impact on the performance of a supervised learning vehicle detector. The classification task is rather simple for our Deep Convolutional Neural Network (DCNN) model to train; as noted in [14], the classification had already achieved high accuracy in the early iterations. Nonetheless, before numerous rounds, the accuracy was relatively low. This was owing to a bad label strategy in our early design, in which we labeled the front-rear region using the same method employed in region regression, resulting in a limited region for ground-true labels[18][19]. After observing the circumstance, we immediately changed the label encoding approach and resolved the problem. Fig. 8 show the comparison acc with other studies.

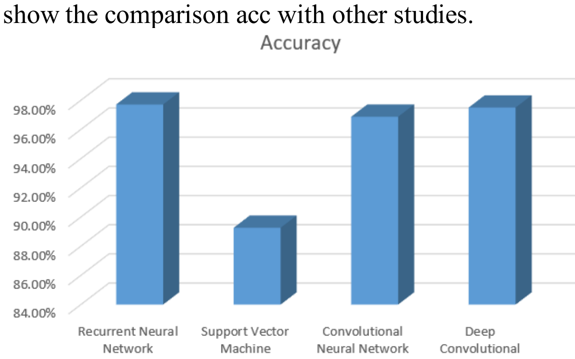


Fig. 9. The ACC comparison with other studies

TABLE. I. COMPARISON OF PROPOSED TECHNIQUE WITH EXISTING LITERATURE

Source	Algorithm	Accuracy
[20]	Recurrent Neural Network	97.71%
[21]	Support Vector Machine	89.26%
[22]	Convolutional Neural Network	96.87%
Proposed	Deep Convolutional Neural Network	97.5%

10. CONCLUTION

License plate detection capabilities of a Deep Convolutional Neural Network (DCNN) based vehicle number estimation with license plate localization model were exhibited. Our method creates a signal that is more accurate for both the assessment of the number of vehicles and the location of their license plates since it detects quadrilaterals rather than rectangles. In the process of calculating the number of vehicles and localizing their license plates, it is helpful to have vehicle data. We refer to this information as "contextual information" since it clarifies the relationships between variables such as the estimated number of vehicles and the location of license plates. Our attempt to categorize stances had an accuracy rate of 97.5%. The addition of contextual information to applications such as traffic scene analysis can improve the interpretation of vehicle number estimations and the localization of license plate locations. Now that we know where the automobile is parked, we can use this information to learn more about the owner, including the sort of vehicle they drive and specifics such as the car's make, model, and color. In addition, the application of our technology could assist with the management of parking lots by indicating to consumers the ideal parking direction.

Funding:

No external financial assistance or institutional funding was utilized for conducting this research. The authors assert that all research-related activities were self-financed.

Conflicts of Interest:

The authors declare that there are no competing interests associated with this work.

Acknowledgment:

The authors would like to thank their institutions for their steadfast encouragement and logistical support throughout this research journey.

References

[1] R. C. Gonzalez and P. Wintz, “Digital image processing,” *Prentice Hall Int.*, vol. 28, no. 4, pp. 484–486, 2008.
[2] M. A. Khan, M. Sharif, M. Y. Javed, *et al*., “License number plate recognition system using entropy-based features selection approach with SVM,” *IET Image Process.*, vol. 12, no. 2, pp. 200–209, 2018.
[3] Y. Yuan, W. Zou, Y. Zhao, X. Wang, X. Hu, and N. Komodakis, “A robust and efficient approach to license plate detection,” *IEEE Trans. Image Process.*, vol. 26, pp. 1102–1114, 2017.
[4] N. L. Yaacob, A. A. Alkahtani, F. M. Noman, A. M. Zuhdi, and D. Habeeb, “License plate recognition for campus auto-gate system,” *Indones. J. Electr. Eng. Comput. Sci.*, vol. 21, no. 1, pp. 128–136, 2021.
[5] W.-M. L. Wen-Min, “Research on image acquisition and license plate location in vehicle license plate recognition system,” *Comput. Modernization*, 2009.
[6] L. Wang, H. Wang, and H. Lianghua, “License plate recognition based on double-edge detection,” *Comput. Eng. Appl.*, 2013.
[7] A. A. Ahmed and S. Ahmed, “A real-time car towing management system using ML-powered automatic number plate recognition,” *Algorithms*, vol. 14, no. 11, p. 317, 2021.
[8] E. Neto, S. Gomes, P. Filho, and V. Albuquerque, “Brazilian vehicle identification using a new embedded plate recognition system,” *Measurement*, vol. 70, pp. 36–46, 2018.
[9] B. Xu, N. Wang, T. Chen, *et al*., “Empirical evaluation of rectified activations in convolutional network,” *Comput. Sci.*, 2015.
[10] H. Peng, F. Long, and C. Ding, *Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy*. IEEE Computer Society, 2005.
[11] N. Srivastava, G. Hinton, A. Krizhevsky, *et al*., “Dropout: A simple way to prevent neural networks from overfitting,” *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[12] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *Comput. Sci.*, 2014.
[13] S. Elfwing, E. Uchibe, and K. Doya, “Sigmoid-weighted linear units for neural network function approximation in reinforcement learning,” *Neural Netw.*, 2018.
[14] S. Rizvi, G. Cabodi, D. Patti, and G. Francini, “Accelerated deep object classification on a heterogeneous mobile platform,” *Electronics*, vol. 5, pp. 88–100, 2016.
[15] S. Rizvi, G. Cabodi, and G. Francini, “Optimized deep neural networks for real-time object classification on embedded GPUs,” *Appl. Sci.*, vol. 7, p. 826, 2017.
[16] P. Wang and J. Cheng, “Accelerating convolutional neural networks for mobile applications,” in *Proc. ACM Multimedia Conf.*, Amsterdam, The Netherlands, Oct. 15–19, 2016.
[17] B. Wang, H. Xiao, J. Zheng, D. Yu, and C. P. Chen, “Character segmentation and recognition of variable-length license plates using ROI detection and broad learning system,” *Remote Sens.*, vol. 14, no. 7, p. 1560, 2022.
[18] S. O. Shim, R. Imtiaz, A. Siddiq, and I. R. Khan, “License plates detection and recognition with multi-exposure images,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 4, pp. 112–120, 2022.

[19] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for vehicle detection,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017.
[20] P. Shivakumara, T. Palaiahnakote, D. Tang, M. P. U. L. Maryam, and H. Anisi, “CNN-RNN based method for license plate recognition,” *CAAI Trans. Intell. Technol.*, vol. 3, p. 113, 2018.
[21] H. Padmasiri, J. Shashirangana, D. Meedeniya, O. Rana, and C. Perera, “Automated license plate recognition for resource-constrained environments,” *Sensors*, vol. 22, no. 4, p. 1434, 2022.
[22] D. K. Francisco and M. Rodrigo, “Convolutional neural networks for license plate detection in images,” in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, 2017, pp. 3395–3399.