A NEURAL NETWORK FOR LICENSE PLATE DETECTION AND RECOGNTION

A Sravanthi Peddinti
Department of ECE
Aditya College of Engg & Tech
Surampalem, India
sravanthi.angara@gmail.com

K.R. Saikiran
Department of ECE
Aditya College of Engg & Tech
Surampalem, India
saikiran.kalapu@gmail.com

N. Bhavya Sri
Department of ECE
Aditya College of Engg & Tech
Surampalem, India
bhavyachowadry11u@gmail.com

R.V. V. Krishna
Department of ECE
Aditya College of Engg & Tech
Surampalem, India
rvvkrishnaece@gmail.com

B. G.M. Vara Prasad
Department of ECE
Aditya College of Engg & Tech
Surampalem, India
gowrimanikantavaraprasad@gmail.com

P. Balaji
Department of ECE
Aditya College of Engg & Tech
Surampalem, India
balajipatamsetti567@gmail.com

***Abstract*— This paper presents the information about the vehicle plate detection and recognition with the character segmentation. This system depends on YOLO for detection and Optical Character Recognition (OCR) for segmentation process. YOLO is a method that provides real-time object detection using neural networks. The popularity of this algorithm is due to its accuracy and quickness. This system evaluates the different models of license plate and aims to improve the accuracy or speed at each and every stage of process. The implementation of Convolutional Neutral Network (CNN) gives more accurate and effective results for detection and segmentation stages. CNN is trained for Vehicle License Plate Recognition (VLPR) to get more accurate results in different conditions (like poor-lightening conditions) CNN gives the detected and rectified vehicle license plate images. Optical Character recognition is capable for recognizes the numbers as well as characters in detected image. By using YOLO, CNN and OCR the experimental results are very accurate without any adaption, it outperforms the detection and recognition of license plate is very accurate and provides the enhancement images and converted text format data as a final output. When compared to state-of-the-arts, the suggested YOLO architecture performs well with a licence plate identification accuracy of 98.7%.**

Keywords: Optical Character Recognition (OCR), Convolutional Neural Network (CNN), You Only Look Once (YOLO).

I. INTRODUCTION

Vehicle license number plate recognition is a challenging task. This task is used in traffic management, security purposes, security surveillance, vehicle recognition, parking etc. This is a very huge task to detect images [1 2]. Due to so many factors the perfect image is not captured that are blurry images, poor lightening conditions, different types of license plates. Vehicle license plate detection and recognition includes three steps that are: license plate detection, license plate detection, license plate recognition.

a. Detected plate b. Segmented plate c. Recognized plate

Fig. 1. Process for Plate Recognition

Previously, the image is too difficult to segment from the detected image so that recognition process will become even more difficult. Later on the improvement and implementation of segmentation algorithm like YOLO, YOLO v2, YOLO v3[13]. Implementation of these algorithms increases the precision and accuracy. It is necessary to gather the license plate characters from the training samples in order to build a model with acceptable recognition performance. Our objective is to identify Chinese characters [4].

Before Recognition and segmentation processes, the image should be detected which is difficult. In previous method, the edge detection method is used so that it crops the detected image based on the edge border colors of the license plate [6,7]. This is a slow run process. Later on, there are too many different types of license plates are created. So many drawbacks are observed in Edge detected method. Now, Convolutional Neural Network [7] is being using for detection, segmented and recognition process. CNN is trained for Vehicle License Plate Recognition (VLPR) to get more accurate results in different conditions .CNN is considered as a special architecture in several artificial convolutional neural networks.

These are the following steps for detection and recognition of characters in vehicle license plate recognition method:

2023 International Conference on Advanced & Global Engineering Challenges (AGEC) | 979-8-3503-4096-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/AGEC57922.2023.00020

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DE GOIAS. Downloaded on September 25,2024 at 23:06:57 UTC from IEEE Xplore. Restrictions apply.

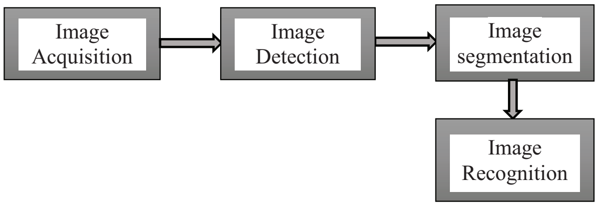

Fig. 2. Block Diagram for LPR system

Image Acquisition: As the input of image detection the camera must captures the red, blue green (RGB) colors. In camera, consists of microcontrollers and integrated components to get more image acquisition [8].

Image Detection: There are several methods used to detect the license plate detection like edge based, color based, region based. But in this Vehicle license plate recognize is developed using CNN. Here the vehicle plate detection is depending on CNN using You Only Look Once (YOLO). Here YOLO is used for two major operations that are detection and segmentation which are implements at a single stage simultaneously [8].

Image Recognition: Here the Optical Character Recognition (OCR) method is used which is a pre-trained dataset. It is training sheet is created by "precharfortraining". OCR is another arm to convolutional neural network (CNN) for character recognition. It consists the combination of real and artificial data generated by using the font type as it is similar to region target places. Although Implementation of Optical Character recognition (OCR) gives the result with high precision, high recall rates and high accuracy [8 10].

II. Literature Review

Laroca et.al [1] proposed the method which describes that ALPR process is divided into four parts that is vehicle plate, vehicle plate detection, vehicle plate segmentation and license plate recognition. Convolutional Neural Network (CNN) is used in each stage to improve the performance. YOLO, YOLO v2, CR-NET models are used. For character segmentation and recognition YOLO architecture is implemented. In order to use YOLO, the filters must be increased in the convolutional layers to match the number of classifiers.

Instead of implementing of powerful classification networks such as VGG, ResNET, GoogleNet. Sergey et.al [2] introduced the block of CNN is inspired by SqueezeNet fire block and Inception block. Also, implemented the networks of Batch Normalization and Rectified Linear Unit (ReLU). By implementing these algorithms the recognition accuracy will be greater than 94%.

Sergio et.al [3] proposed the method which consists of three primary steps: vehicle detection, LP detection, and OCR [3]. The first module recognizes license plate in the scene from an input image. The proposed method technique Warped Planar Object Detection Network (WPOD-NET) looks for LPs within each detection zone and regress one affine transformation per detection to enable rectification of the LP area into a rectangle shape a frontal view. For the purpose of character recognition, an OCR Network receives these positive and corrected detections. The study of a reliable feature extraction model and Back Propagation Neural Network (BPNN)-based character recognition and license plate detection algorithm was introduced by Xie.F et.al [4]. The experimental findings show that the suggested algorithm achieves a strong recognition performance and compatibility in distinguishing various license plates even in low-light conditions and against backgrounds with complicated patterns.

Younkwan Lee et.al [5] proposed an end-to-end pipeline for LPR is implemented. We first introduce the adversarial network, which will be used to reconstruct the output of the super-resolved input image. To discover the best model parameters, he finally outline the training method.

Chinmayi et al [6] introduced the Optical Character Recognition (OCR) method for license plate segmentation. Yao et.al [7] provided an end-to-end identification and license plate recognition system for all varieties of Chinese license plates in the outside setting, based on SSD technology. Two SSD-based networks are combined to form the LPR-SSD network. It primarily optimizes the classification layer for the license plate and eliminates the entire connection layer in order to improve placement and classification effectiveness.

Ibitoye et.al [8] gives a brief description of the methods used for picture acquisition, image preprocessing, license plate identification, license plate character segmentation, and license plate character recognition. The study also breaks down a few CNN-based approaches for a system that reads license plates.

Slimani et.al [9] proposed the usage of vertical edges to identify prospective candidates for license plates. The edges of vertical are first extracted using the 2D-Weight Decay (WD). The probable license plate possibilities are then found by calculating with high density of the vertical shaped edges. In order to eliminate the false positive, a plate/non-plate CNN classifier is trained on these candidates.

Laroca et.al [10] introduced a method for locating the cars in the input image first, and then identifying their specific LPs in the vehicle patches. Next, by sending the complete LP patch into the network, we simultaneously detect and recognize all characters. This eliminates the need for us to deal with the character segmentation task.

The overview of ANPR algorithms was proposed by Lubna [11] that have been suggested and tested in recent pertinent studies. We divided up these algorithms into groups based on the features that each stage of the recognition process requires. For a performance summary, each stage is described in full along with any relevant problems and difficulties. If the dataset is not widely used, it is challenging to conduct a standard assessment and comparison.

A multiple license plate recognition approach for high-resolution photos was introduced by Khan et.al [12]. This system operates in real-time scenarios in difficult lighting situations. Plate detection was divided into two parts in the suggested method. Faster-RCNN was utilized to detect all the automobiles in a picture in the first stage, producing scaled information to locate license plates. Then, a LUT

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DE GOIAS. Downloaded on September 25,2024 at 23:06:57 UTC from IEEE Xplore. Restrictions apply.

classifier employing adaptive boosting and MCT as a feature extractor performs character recognition.

Usama et.al [13] introduced a method having three distinct processes make up our suggested framework: i) vehicle identification and type recognition; (ii) license plate detection; and (iii) license plate reading. Each and every step is treated as an object detection and recognition issue, and a different model is trained for each of them. In order to identify the vehicle and its license plate in order to determine the toll tax, all three models are consequently applied in a cascading way during the run time.

Nguyen et.al [14] proposed a technique which is put into practice using the Pytorch framework and an NVIDIA Titan X GPU. For both the training and testing phases, all input photos are downsized to 320 320. This study uses a pre-trained ESPNetv2 model that was developed using the ImageNet dataset. The proposed model is end-to-end trained using synchronized SGD with a momentum of 0.9 and a weight decay of 0.0001. Since the suggested model trains on low-resolution images, substantial data augmentation approach is used to enhance training outcomes.

III. Methodology

In this paper, the proposed method Vehicle License Plate Recognition (VLPR) is described as in five steps they are data collection, license plate detection, number plate segmentation, license plate recognition and conversion into the text. We use distinct CNNs for these Vehicle License Plate detection and Recognition. Thus, CNN improves the performance and accuracy of the trained dataset system. The main important architecture used in this VLPR system is You Only Loo Once (YOLO) algorithm. CNN employs the YOLO for detection and segmentation. For Character Recognition, the Optical Character Recognition (OCR) was implemented.

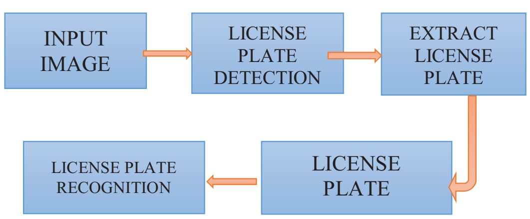

Fig. 3. Steps for proposed method approach

Now, the process of vehicle plate detection and recognition is explained in below:

A. *Input Image:*

License plate detection and recognition is a specific application of image recognition that involves capturing images of vehicles and extracting the license plate information from those images. In this case, the input image is combination of red, blue, green (RGB) colors which is captured by camera. Camera consists lots of electronic components like ultrasonic sensors and microcontrollers for better captures. For real-time implementation, the quality of camera is main priority for better results of recognition, other factors also influence for better results like camera positioning. Once the input image is captured, it is processed using various image recognition techniques to extract the license plate information. Finally, optical character recognition (OCR) techniques are used to recognize the characters on the license plate and convert them into a text string that can be used for further processing or storage.

B. *Preprocessing:*

The input image should be preprocessed to enhance its quality and remove any unwanted noise before applying the detection algorithm on it. Therefore, we are using picture resizing for this, which involves multiplying the image's height and width by a scaling factor. Rescaling alters the spatial extents of the pixels and the aspect ratio if the scaling factor is not the same in the vertical and horizontal axes.

C. *Image Detection:*

Although there are several methods used to detect the license plate detection like edge based, color based, region based. But in this Vehicle license plate recognize is developed using CNN. Through CNN, we can improve the performance and accuracy of each stage. Two object identification convolutional neural networks (CNN) are used in a deep learning-based system for recognizing license plates that was proposed [8]. Both the first and second networks were retrained to recognize characters based on the detected license plates, with the first network being trained for license plate detection and the second network being trained for segmentation.

Also in this system, the algorithm was implemented called CNN. This CNN is used for segmentation and detection. The license plate will be detected after the dataset will get trained through CNN. The object detection is carried out through the CNN using a model called You Only Look Once (YOLO). Here in this proposed method, we used the YOLO v2 technique. In this proposed method, YOLO architecture consists of 25 layers. In there are four main categories Batch normalization, Max Pooling, Convolution, Rectified Linear Unit (ReLU).

Convolutional method: The convolutional layers in YOLO use filters to extract number plate from the input image. Each filter slides over the image and performs a dot product between its weights and the values in the corresponding region of the input image. This process generates a feature map, which highlights the presence of certain features in the input image. The convolutional method used in YOLO allows the algorithm to extract License plate from the input image and use them to accurately detect and localize objects in real-time.

Max Pooling: In YOLO, max pooling is typically performed after every few convolutional layers to reduce the spatial dimensions of the feature maps while retaining the most salient features. Max pooling helps to achieve translation invariance, where the network can detect objects regardless of their position in the input image.

Rectified Linear Unit (ReLU): ReLU is an activation function used in the YOLO (You Only Look Once) for object detection algorithm. ReLU is a non-linear activation function that sets all negative values in the input to zero. It is an important component of the YOLO architecture that helps to introduce non-linearity and enable the network to learn more complex representations of the input image.

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DE GOIAS. Downloaded on September 25,2024 at 23:06:57 UTC from IEEE Xplore. Restrictions apply.

Batch Normalization: In YOLO, batch normalization is typically applied after each convolutional layer. The purpose of batch normalization is to normalize the activations of the previous layer in a batch-wise manner. In particular, it standardizes the activations by taking the mean and standard deviation and subtracting them, which lessens the internal covariate shift. Finally, by using this technique reduces the internal covariate shift and allowing for faster convergence and improved generalization.

TABLE I. LAYERS OF YOLO

1.	'input'	Input image	128*128*3
2.	'conv_1'	Convolution method	Convolution 16 3 * 3 *3convolutions with padding [1 1 1 1] and stride[1 1] 128*128*16
3.	'BN1'	Batch Normalisation	128*128*16
4.	'relu_1'	ReLU	128*128*16
5.	'maxpool1'	Max Pooling	Maximum pooling of 2*2 with padding [0 0 0 0] and stride [2 2] 64*64*16
6.	'conv_2'	Convolution	Convolution 32 3*3 with buffer [1 1 1 1] and stride [1 1] 64*64*32
7.	'BN2'	Batch standardization	64*64*32
8.	'relu_2'	ReLU	64*64*32
9.	'maxpool2'	Max Pooling	Maximum pooling of 2*2 with padding [0 0 0 0] and stride [2 2] 32*32*32
10.	'conv_3'	Convolution	Convolution 64 3*3 *3convolutions with padding [1 1 1 1] and stride [1 1]
11.	'BN3'	Batch Normalisation	32*32*64
12.	'relu_3'	ReLU	32*32*64
13.	'maxpool3'	Max Pooling	Greatest pooling of 2*2 with padding [0 0 0 0] and stride[2 2], 16*16*64
14.	'conv_4'	Convolution	Convolution 128 stride [1 1] and padding [1 1 1 1] in 3*3*64 convolutions
15.	'BN4'	Batch Normalisation	16*16*128
16.	'relu_4'	ReLU	16*16*128
17.	'yolov2Conv1'	Convolution	128 3*3*128 convolutions with strides [1 1] and padding 'same',16*16*128
18.	'yolov2Batch1'	Batch Normalization	16*16*128
19.	'yolov2Relu1'	ReLU	16*16*128
20.	'yolov2Conv2'	Convolution	Convolution 128 With steps [1 1] and padding "same," 3*3*128, 16*16*128
21.	'yolov2Batch2'	Batch Normalization	16*16*128
22.	'yolov2Relu2'	ReLU	16*16*128
23.	'yolov2ClassConv'	Convolution	Convolution 24 1*1 *128 convolutions with padding [0 0 0 0] and stride [1 1], 16*16*24
24.	'yolov2Transform'	Yolo v2 Transform Layer	16*16*24
25.	'yolov2Output Layer'	Yolo v2 Output	output

D. *Extracted License Plate*

Yolo uses a single CNN to simultaneously predict the boundaries and the class probabilities for each object in the image. This is done by dividing the input image into a grid of cells and applying object scores to each block, indicating the probability that the cell contains an object. Each block predicts a fixed number of bounding boxes, which are offsets from the block coordinates to the center and dimensions of the object.

YOLO also predicts the class probabilities for each bounding boxes. The final predicted classes for each object are determined by taking the class probability with the highest score. Non-maximum suppression is then used to eliminate redundant detections and retain only the most certain predictions once YOLO has forecasted the object bounding boxes and class probabilities for each cell.

In summary, YOLO architecture allows the extraction of detailed information about an image by predicting the bounding boxes and class probabilities for each object in the image. This information can be used for a variety of computer vision tasks, such as object detection, tracking, and recognition.

E. *License plate segmentation*

Once the license plate has been extracted from the image using YOLO, you can use a CNN for character segmentation to extract the individual characters on the license plate. The first step in character segmentation is typically to pre-process the license plate image to enhance its contrast, remove noise, and normalize its size and orientation. This can involve techniques such as histogram equalization, adaptive threshold, and rotation and scaling.

Next, you can use a CNN to classify each pixel in the license plate image as either foreground (belonging to a character) or background. This is typically done using a convolutional neural network (CNN), which takes the entire license plate image as input and outputs a segmentation mask that indicates which pixels belong to characters. By using this technique, possible to achieve high accuracy in vehicle plate recognition.To perform license plate segmentation using YOLOv2, you would typically follow these steps:

Collect and preprocess data: Collect a large dataset of license plate images and preprocess them by resizing, cropping, and normalizing the images to ensure consistency and reduce noise.

Train the model: Use the YOLOv2 architecture to train a CNN model on the preprocessed data. This involves feeding the model with the input images and their corresponding ground-truth labels (i.e., the bounding boxes around the license plates in the images). The model learns to identify and localize license plates in the images by minimizing a loss function that penalizes errors between the predicted and ground-truth bounding boxes.

F. *License Plate Recognition*

After segmentation of individual characters from the license plate image using CNN, Optical Character Recognition (OCR) techniques can be applied to recognize the segmented characters. There are several OCR techniques that can be used for license plate recognition, including template matching, feature extraction, and machine learning-based approaches.

Here we use the Feature Extraction. Feature extraction involves extracting a set of features from the segmented

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DE GOIAS. Downloaded on September 25,2024 at 23:06:57 UTC from IEEE Xplore. Restrictions apply.

character images, such as their shape, size, and texture, and using these features to recognize the characters. This approach is more robust to variations in character appearance but requires more complex algorithms and training data. OCR is a crucial part of the license plate recognition system and must be carefully chosen and optimized in order to achieve high accuracy in identifying the segmented characters.

A. Training loss for each iteration vs Number of iterations

For YOLO (You Only Look Once), which is a popular object detection algorithm that uses a CNN (Convolutional Neural Network), the training loss for each iteration typically decreases over time as the model learns to better detect and recognize license plates.

During training, the YOLO algorithm updates the weights of the neural network to minimize the loss function, which is a measure of how well the predicted bounding boxes match the ground truth bounding boxes. The loss function typically includes terms for the classification error, localization error, and confidence error.

As the model trains, the loss generally decreases, which means that the predicted bounding boxes become more accurate and the confidence scores become more reliable. The exact shape of the loss curve can vary depending on factors such as the size of the dataset, the complexity of the network, and the training parameters.

Typically, the loss curve is plotted against the number of iterations or epochs, where an iteration corresponds to one forward and backward pass through the network using a batch of training data. The loss curve may show an initial decrease followed by a plateau or oscillations as the model converges to a solution.

It's important to monitor the training loss and other performance metrics during training to ensure that the model is learning effectively and to diagnose any issues that may arise.

IV. Results and Discussion

A. Dataset

Data used to validate the proposed approach was obtained from the "Car Licence Plate Detection" dataset on Kaggle. This collection includes 433 pictures of car licence plates with bounding box comments.[14]

B. Implementation

In this paper, we proposed that the concept of license plate detection and recognition by Convolutional Neural Network (CNN) using You Only Look Once (YOLO). Here we are going to introduce CNN network for license plate detection and recognition which equally treats License plates and characters as objects to detect and classify and it conducts these two tasks simultaneously. The use of YOLO algorithm in this proposed method can improve the recognition accuracy and provides better detection.

The performance of license plate detection and recognition systems can vary depending on the specific dataset, model architecture, and training process used. However, with recent advancements in deep learning techniques, state-of-the-art models have achieved high accuracy and robustness in license plate detection and recognition. Challenges faced Despite the recent progress, license plate detection and recognition still pose some challenges.

Fig. 4. (a) Input image; (b) License plate detected; (c) license plate extracted; (d) license plate segmented.

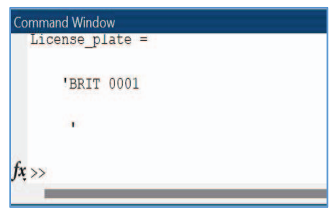

Fig. 5. License plate recognition

Some of the common challenges include:

1) Variability in license plate formats: License plates can have different formats and sizes, depending on the country or region. For example, in some countries, license plates may have different shapes, such as square or oval, while in others, they may have different numbers of characters.
2) Complex backgrounds: License plates may appear in complex backgrounds, such as in crowded streets, parking lots, or under different lighting conditions, which can make it difficult to detect and recognize them accurately.
3) Occlusions: License plates may be partially or fully occluded by other objects, such as cars, trees, or pedestrians, which can make it challenging to detect and recognize them accurately.
4) Motion blur: License plates in moving vehicles may suffer from motion blur, which can make it challenging to recognize them accurately.
5) Privacy concerns: License plate detection and recognition systems may raise privacy concerns, especially if they are used for surveillance or law enforcement purposes.

License plate detection and recognition is an important task in the field of computer vision, with many practical applications. While recent advancements in deep learning techniques have improved the accuracy and robustness of license plate detection and recognition systems, there are still some challenges that need to be addressed to improve the performance of these systems in real-world scenarios.

Plate detection: Once the image is given for input, the next step is to locate the license plate within the image. This can be done using techniques such as edge detection, template matching, or machine learning algorithms such as Haar cascades or convolutional neural networks (CNNs).

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DE GOIAS. Downloaded on September 25,2024 at 23:06:57 UTC from IEEE Xplore. Restrictions apply.

Character segmentation: Once the license plate is detected, the next step is to segment the individual characters within the plate. This can be challenging, as license plates can vary in size, shape, and font. Techniques such as connected component analysis, contour detection, or machine learning algorithms can be used for this step.

Character recognition: Once the characters are segmented, the next step is to recognize them. This can be done using optical character recognition (OCR) algorithms. These algorithms use pattern recognition techniques to match the segmented characters to a pre-defined set of characters.

Finally, the results of the recognition process are refined and verified. This can involve applying algorithms to correct errors, checking the validity of the license plate number, or using contextual information to improve the accuracy of the recognition.

The accuracy of the system depends on the quality of the input images, the sophistication of the algorithms, and the performance of the hardware used to process the data. Despite the challenges, license plate detection and recognition is a critical technology that is widely used in traffic management, surveillance, and security applications.

Our suggested approach is tested against the "Car Licence Plate Detection" dataset, and performance analysis is carried out utilising a few cutting-edge tools given by Rayson et al [10] and Muhammad et al [13].

TABLE II. COMPARATIVE ANALYSIS

METHOD	*Detection accuracy*	*Segmentation/ Recognition accuracy*
[10]	94.5%	95.7%
[13]	95.2%	96.3%
Proposed	98.7%	99.3%

V. CONCLUSION

Vehicle license plate detection and recognition systems have significantly advanced with the use of machine learning and computer vision techniques. These systems have become increasingly important in a range of applications, including traffic management, law enforcement, and parking management. The accuracy of license plate detection and recognition systems has improved significantly to 98.7% with the use of deep learning models, especially convolutional neural networks (CNNs). These systems can handle variations in lighting, weather, and other environmental factors.

REFERENCES

[1] Laroca, R., Severo, E., Zanlorensi, L. A., Oliveira, L. S., Goncalves, G. R., Schwartz, W. R., & Menotti, D," A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector",International Joint Conference on Neural Networks (IJCNN). doi:10.1109/ijcnn.2018.8489629, 2018

[2] Sergey Zherzdev, Alexey Gruzdev, "LPRNet: License Plate Recognition via Deep Neural Networks", . https://doi.org/10.48550/arXiv.1806.10447, 2018

[3] Sergio Montazzolli Silva, Claudio Rosito Jung, "License Plate Detection and Recognition",Proceedings of the European Conference on Computer Vision (ECCV), pp.580-596, 2018

[4] Xie.F, Zhang.M, Zhao. J, Yang. J, Liu. Y, & Yuan. X," A Robust License Plate Detection and Character Recognition Algorithm Based on a Combined Feature Extraction Model and BPNN", Journal of Advanced Transportation, pp.1–14. doi:10.1155/2018/6737314, 2018

[5] Younkwan Lee, Jiwon Jun, Yoojin Hong, Moongu Jeon, "Practical License Plate Recognition in Unconstrained Surveillance Systems with Adversarial Super-Resolution" https://doi.org/10.48550/arXiv.1910.04324, 2019

[6] Chinmayi Gurav , Vedika Kamble Rupali Gurav, "A Review Paper on Vehicle Number Plate Recognition",International Journal of Engineering Research & Technology (IJERT) http://www.ijert.org ISSN: 2278-0181 IJERTV8IS040246, 2019

[7] Yao, Li, et al. "Research and Application of License Plate Recognition Technology Based on Deep Learning." Journal of Physics: Conference Series. Vol. 1237. No. 2. IOP Publishing, 2019.

[8] Ibitoye, Oladapo, Temitayo Ejidokun, Olufemi Dada, and Olusegun Omitola. "Convolutional neural network-based license plate recognition techniques: A short overview." In 2020 International Conference on Computational Science and Computational Intelligence (CSCI), pp. 1529-1532. IEEE, 2020.

[9] Slimani, Ibtissam, et al. "An automated license plate detection and recognition system based on wavelet decomposition and CNN." Array 8: 100040,2020.

[10] Laroca, Rayson, et al. "An efficient and layout-independent automatic license plate recognition system based on the YOLO detector." IET Intelligent Transport Systems 15.4 : 483-503,2021.

[11] Lubna, Naveed Mufti, and Syed Afaq Ali Shah. "Automatic number plate Recognition: A detailed survey of relevant algorithms." Sensors 21.9: 3028,2021

[12] Khan, K., Imran, A., Rehman, H.Z.U., Fazil, A., Zakwan, M. and Mahmood, Z., 2021. Performance enhancement method for multiple license plate recognition in challenging environments. EURASIP Journal on Image and Video Processing, 2021(1), pp.1-23 https://doi.org/10.1186/s13640-021-00572-4,2021

[13] Usama, Muhammad, et al. "Vehicle and License Plate Recognition with Novel Dataset for Toll Collection." arXiv preprint arXiv:2202.05631 (2022).

[14] Nguyen, H., 2022. An efficient license plate detection approach using lightweight deep convolutional neural networks. Advances in Multimedia, 2022,pp.1-10, https://doi.org/10.1155/2022/8852142,2022

Authorized licensed use limited to: UNIVERSIDADE FEDERAL DE GOIAS. Downloaded on September 25,2024 at 23:06:57 UTC from IEEE Xplore. Restrictions apply.