



e-ISSN: 2319-8753 | p-ISSN: 2320-6710

IJIRSET

International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 10, Issue 7, July 2021

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 7.569



9940 572 462



6381 907 438



ijirset@gmail.com



www.ijirset.com

Automatic Licence Plate Recognition (ALPR) System

Harshvardhan Bhatia¹, Aayush Narula², Ms. Dhivya V³

B.E. Student, Department of CSE, Sir M Visvesvaraya Institute of Technology, Bengaluru, India¹

B.E. Student, Department of CSE, Sir M Visvesvaraya Institute of Technology, Bengaluru, India²

Assistant Professor, Department of CSE, Sir M Visvesvaraya Institute of Technology Bengaluru, India³

ABSTRACT: Automatic Licence Plate Recognition (ALPR) is the extraction of vehicle licence plate (LP) information from an image or a video. Because of the numerous real-world applications of this information, such as automatic toll collection, traffic law enforcement, access control of private facilities, and road traffic monitoring, ALPR has been a prominent research topic.

The main goal of this project work is to build an Automatic Licence Plate Recogniser to detect various vehicle plates along with their registration number. The system is based on You Only Look Once (YOLO), an object detection method, and Python-tesseract, a python-based optical character recognition (OCR) tool. The images are uploaded and provided as input for vehicle(s) detection, the output of which is given as input for LP detection. Once, the LP(s) are identified, the resultant image is passed for OCR. The paper addresses the findings of this study, and also ways for improving the model's performance.

KEYWORDS: Licence plate identification, Licence plate segmentation, Automatic Licence Plate Recognition (ALPR), Automatic Number Plate Recognition (ANPR), Python-tesseract, Optical Character Recognition (OCR), YOLO.

I. INTRODUCTION

ALPR, being a real-world application, must process licence plates swiftly and effectively in a variety of settings, including indoors, outdoors, day or night. It is also necessary to standardise the processing of licence plates from different countries, territories, or states.

ALPR systems are typically made up of three stages of processing: 1) identifying the LP region; 2) segmenting the plate characters; and 3) character recognition. The combination of the last two steps is referred to as OCR, for convenience. Furthermore, high accuracy is required for the initial step, as failure to detect the LP will probably result in failure in the subsequent steps. To decrease processing time and prevent false positives, many methods scan for the car first, then the LP.

Due to the availability of large scale annotated datasets and technology (GPUs) capable of handling massive amounts of data, many computer vision applications have recently experienced a considerable increase in efficiency. In this situation, deep learning (DL) approaches seem to be the most prominent. Many computer vision applications, such as Optical Character Recognition (OCR) and Object Recognition/Detection, have benefited from recent improvements in Deep Learning (DL) and Parallel Processing, which are clearly beneficial to ALPR systems. For detecting automobiles and licence plates (LPs), deep Convolutional Neural Networks (CNNs) have become the most commonly utilised machine learning technique. DL approaches have been examined by numerous commercial ALPR systems in addition to academic articles. They are frequently located in large data centres and run on web-services, which allows them to process thousands to millions of photos per day while continuously improving. The quantity and quality of training data are critical for the success of deep learning approaches. With more data, more resilient network topologies with more parameters and layers can be used, resulting in better results. The total LPR performance is also influenced by the quality of the testing cases. However, this component is frequently overlooked when evaluating or comparing performance, which is the appropriate criterion for algorithmic evaluation.

II. TASKS PERFORMED

Figure 1 depicts the steps that are taken in order to implement this project.

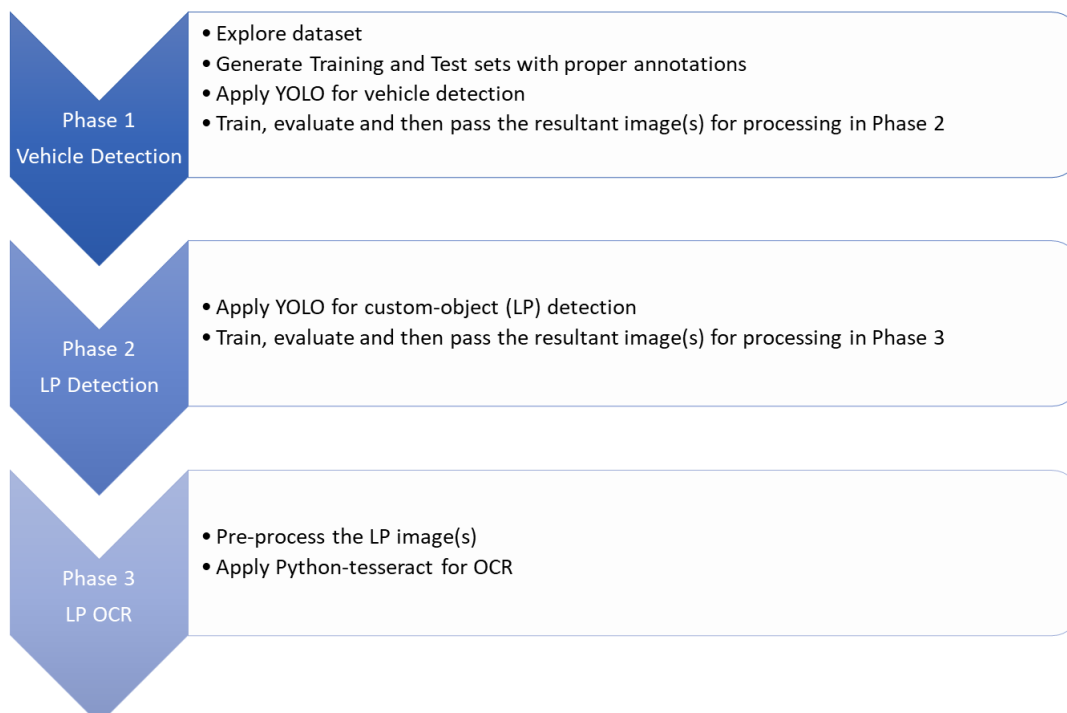


Fig. 1Tasks performed

III. DATASET AND ITS FEATURES

A total of 5 different datasets are used, out of which 1 is used for vehicle detection and the rest 4 are used for LP detection.

The dataset used to train the model for vehicle detection consists of 627 images of various vehicle classes derived from the Open Images Dataset V4 [1]. The median image ratio is 1024x751 and average image size is 0.72 mp. A sample of the dataset is shown in figure 2. After generating training and test sets of appropriate sizes and applying the YOLO model, it was observed that the model performed poorly on this task. Hence, we used a YOLO model pre-trained on Pascal VOC dataset. The Pascal VOC challenge is a widely used dataset for developing and testing image classification, object recognition, and segmentation algorithms.



Fig. 2Open Image Dataset of Vehicles

The dataset used to train the model for LP detection consists of 3331 images of various vehicle classes generated by merging LP images from 4 different datasets obtained from Kaggle [2, 3, 4, 5]. The dataset has been split into a training set of 2602 images and a test set of 729 images. A sample of this dataset is shown in figure 3.



Fig. 3 LP Dataset Sample

IV. METHODOLOGY

Phase 1: Vehicle Detection

Vehicle detection is essentially an object detection task, which itself falls in the realm of image processing and computer vision that deals with detecting instances of semantic objects of a certain class. In phase 1, images are acquired to create datasets for vehicles and licence plates. After appropriate annotations these are split into training and test sets with 80:20 ratio.

YOLO is then trained for custom-object detection. It first splits the input image into $M \times M$ grid cells. Each grid cell forecasts V bounding boxes and an "objectness" score that indicates whether or not the grid cell contains a vehicle. Each grid cell, additionally, predicts the conditional probability of the class the vehicle contained by the grid cell belongs to. Finally, appropriate bounding boxes are created in the image as shown in figure 4. YOLO generates a confidence score for each bounding box. If there is no vehicle in the bounding box, the confidence score is 0. The value of a confidence score can range from 0 to 1, and we opted 0.25 for our model because it delivered the best results out of all the other scores.

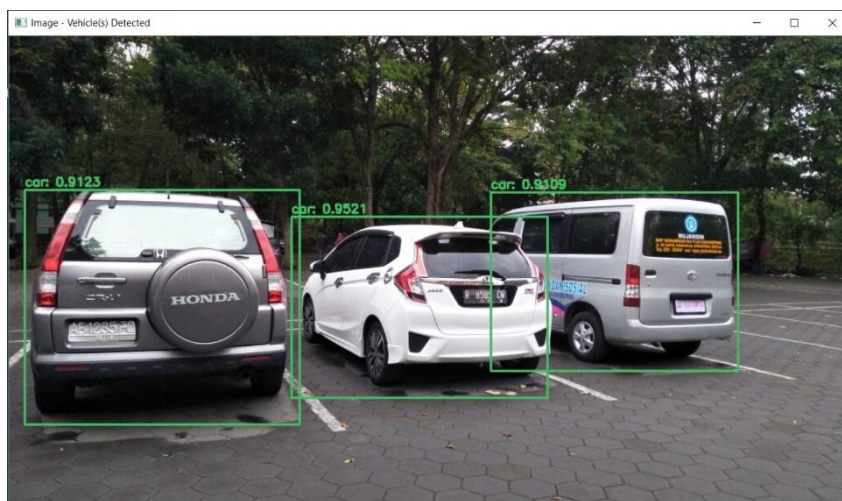


Fig. 4 Vehicle Detection

Phase 2: LP Detection

Once the vehicle is detected, the image inside the bounding box is passed for LP detection. The detection is done in the same way as vehicle detection using YOLO as shown in figure 5.

The only difference is that we have custom-trained the model on our LP images and the confidence score used. The confidence score in vehicle detection was set relatively low because if the vehicle is not detected, it will result in definite failure of LP detection. But even if a false positive is detected it can be eliminated by the LP detector in this step and hence the confidence score used here is 0.4.

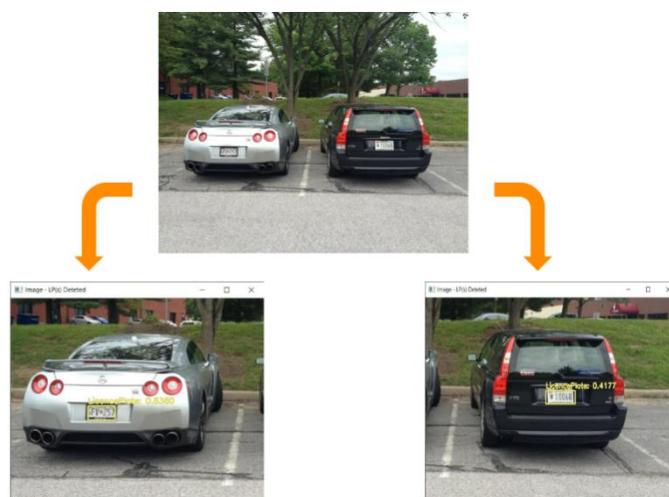


Fig. 5 LP Detection

Phase 3: Licence Plate OCR

The registration number from the LP is recognized using Python-tesseract, which is a wrapper for the Tesseract-OCR Engine developed by Google. Because colour information doesn't help us detect relevant edges or other features, we convert the LP to grayscale. Simultaneously, we also process the image such that the light areas are highlighted.

All the three images, including the original image, are fed for OCR and the output from all the three images is compared character by character. Frequency is counted for characters at each position in all the three images and the character with highest frequency or simply the character which has the majority is selected for each position respectively. Finally the recognized LP number is displayed. Figure 6 shows the OCR pipeline.

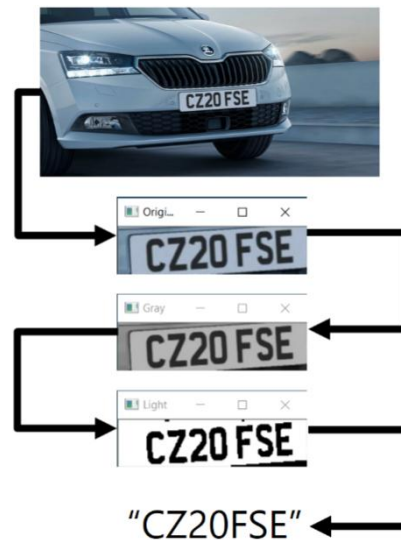


Fig. 6LP OCR

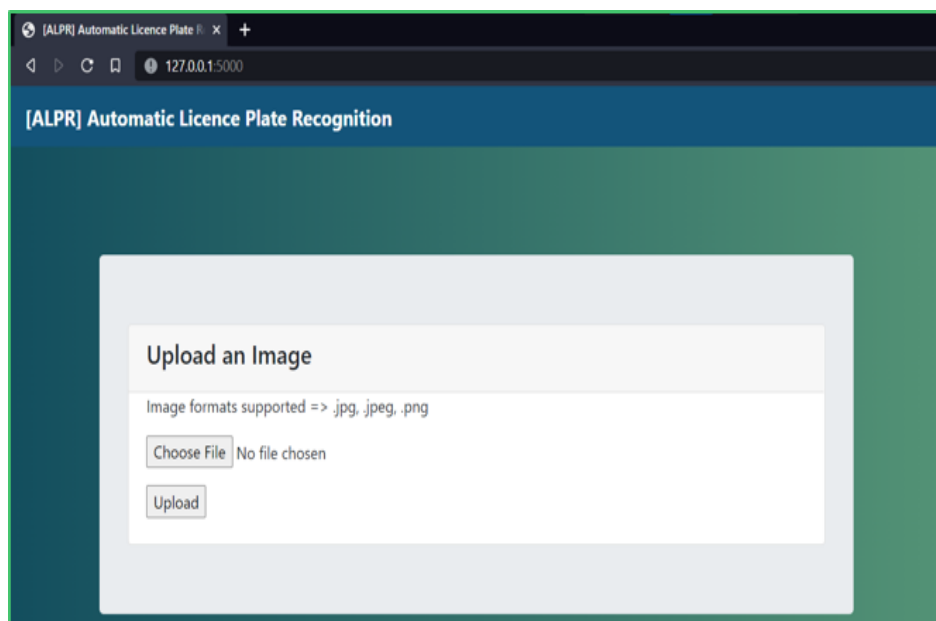
V. IMPLEMENTATION USING FLASK

Flask is a Python API that allows us to create web-based applications. We created a simple and easy-to-use webapp with a graphic user interface using the Flask framework. Flask's framework is more explicit than Django's, and it's also easier to learn because it requires less base code to create a simple web application.

The webpage has the feature of uploading images that can have one of the following 3 extensions: .png, .jpg, .jpeg. The webapp displays an appropriate output for the following 3 cases:

- no vehicle detected
- partial detection (only vehicle detected, no LP detection; vehicle detected, LP detected, but no LP OCR)
- complete detection

The entire webpage is hosted on the local host of the computer. Figure 7 shows the sample of the web page designed using the Flask App.



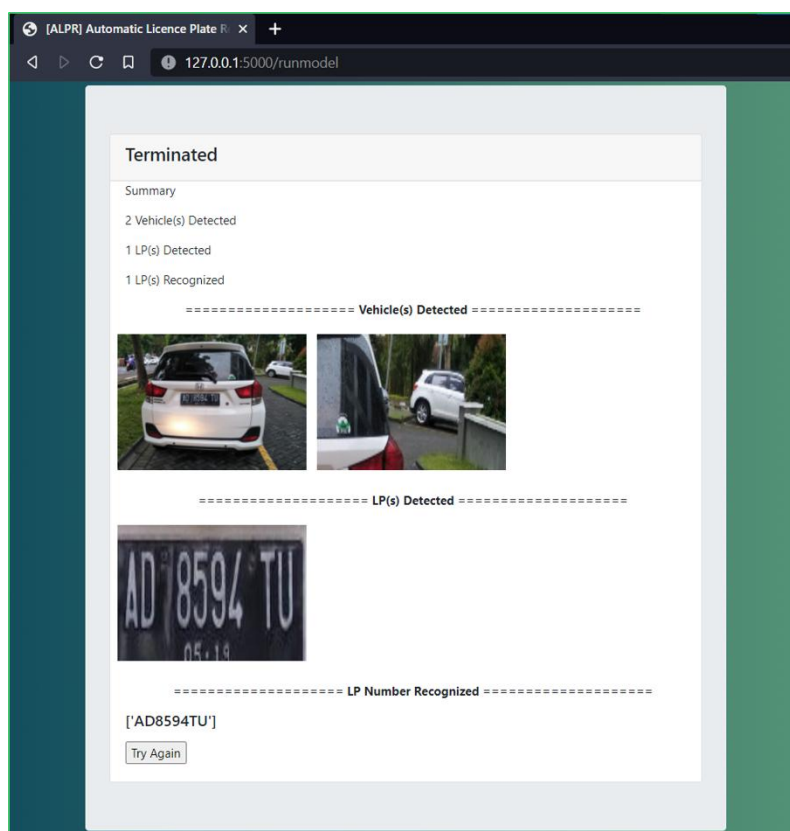


Fig. 7 Output Page for Complete Detection

VI. RESULTS

Once the training for each of the phase is completed, 150 random images are scrapped/ obtained from various sources for final testing. The model obtained an accuracy of 54% for complete detection, which includes successful vehicle detection, LP detection as well as LP OCR. Table 1 shows the result in detail for each phase and complete detection.

Table 1 Outcome of Each Phase

Detection Phase	Successful Detections	Accuracy (%)
Vehicle Detection	141	94
LP Detection	117	78
LP OCR	81	54
Complete Detection	81	54

VII. CONCLUSION

Although significant advances have been made over the past decade, there is still room for improvement, as a reliable LP detection system should function for a wide range of plate conditions/types, lighting conditions, and image acquisition settings. Because of growing mobility and globalisation, developing an ALPR system that can accommodate plates from various nations with different syntax and character sets has become difficult. As many ALPR systems were



country/region specific, this topic has not been adequately addressed in the literature thus far. In addition, it is apparent that the quality and quantity of testing samples have a direct effect on the overall ALPR performance. However, this factor is often ignored in performance comparison or evaluation, which is the appropriate criterion for an algorithmic assessment.

REFERENCES

- [1] Sergio Montazzolli Silva, Claudio Rosito Jung, "License Plate Detection and Recognition in Unconstrained Scenarios", Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 580-596
- [2] Li, Hui & Shen, Chunhua. (2016). Reading Car License Plates Using Deep Convolutional Neural Networks and LSTMs.
- [3] H. Li, P. Wang and C. Shen, "Toward End-to-End Car License Plate Detection and Recognition With Deep Neural Networks," in IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 3, pp. 1126-1136, March 2019, doi: 10.1109/TITS.2018.2847291.
- [4] Selmi, Zied & Ben Halima, Mohamed & Alimi, Adel. (2017). Deep Learning System for Automatic License Plate Detection and Recognition. 1132-1138. 10.1109/ICDAR.2017.187.
- [5] Lee, Younkwan & Jun, Jiwon & Hong, Yoojin & Jeon, Moongu. (2019). Practical License Plate Recognition in Unconstrained Surveillance Systems with Adversarial Super-Resolution.
- [6] Masood, Zain & Shu, Guang & Dehghan, Afshin & Ortiz, Enrique. (2017). License Plate Detection and Recognition Using Deeply Learned Convolutional Neural Networks.



**Impact Factor:
7.569**



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

 **9940 572 462**  **6381 907 438**  **ijirset@gmail.com**



www.ijirset.com

Scan to save the contact details