Detecting Multiple Indian Licence plates from Real time Videos using YOLOv4 and CNN

Abhishek K Wakchaure[1], Dr. Rajashree Shettar[2]

[1,2]*Department of Computer Science Engineering, RV College of Engineering, Bangalore, Karnataka*

***Abstract* - An Automatic Licence Plate Recognition (ALPR) system is able to read the characters on the number plate without human interference. It has been an area of continuous research due to a variety of practical applications but they are still dependent on many constraints. Indian licence plates are even tougher because of the density of cars on the roads. In this paper, we present a deep learning-based approach which treats ALPR as two separate problems, an object detection using YOLO v4 and a character recognition problem using a Convolutional Neural Network model (CNN). The YOLO v4 model is custom trained to detect multiple licence plates in complex environments in real time and the CNN model is fine-tuned so it is robust enough to recognize characters even in improper lighting conditions or low clarity. The approach yielded impressive results on two separate datasets, an UFPR-ALPR dataset which contains 150 videos and 4500 frames and an Indian Licence plate dataset with 640 images with different colour and lighting conditions.**

***Index Terms* - Automatic Licence Plate Recognition, complex environments, Real time, Optical Character Recognition, YOLOv4, CNN, UFPR-ALPR.**

I.INTRODUCTION

A licence plate is a unique identifier for a vehicle. Identifying a vehicle can be useful in a variety of applications. In a toll system which requires a lot of manual operators, detecting the number plate accurately can automate the entire process. Once the licence plate is detected, the system can link the vehicle to the owner and generate the toll bill without any human intervention. In traffic surveillance scenarios, if someone exceeds the speed limit or crosses a red light, the system can automatically detect the licence plate and raise a fine ticket for the vehicle. This can help in maintaining law and order and reduce the number of accidents on the roads. Licence plate recognition can also help the police department to search for stolen vehicles. If the vehicle is captured by a traffic surveillance camera, it can be easier to locate the stolen vehicle. The need for an accurate Automatic Number Plate Recognition (ANPR) [1] system in India is strong but there are many challenges.

A licence plate recognition problem is usually divided into two parts, the detection of the licence plate and then reading the characters on the detected plate. Older approaches to detecting licence plates included using Image processing techniques like edge detection. Many approaches eliminated false positives by detecting the vehicle before detecting the licence plate. Many Images processing and computer vision techniques have become more accurate due to better image quality and improved processing units. This is where deep learning comes in the picture. Availability of a large amount of data and fast GPU's have made it possible to train a model that processes images.

Due to frequent research in the field of ANPR, a variety of datasets are available for detection as well as recognition. The UCSD car dataset stores 10 hours of video recordings of cars at different times of the day. The UFPR-ALPR dataset, the Caltech cars dataset, the Chinese city parking dataset, all of them can be used to train a licence plate detection algorithm. However, as we only focus on detecting Indian licence plates, these datasets cannot be used for character recognition. Hence, we had to prepare a dataset of Indian cars to train the character recognition classifier. We have proposed an approach to detect licence plates using YOLOv4 [2]. The detection result is cropped and each character from the plate is segmented. We have then applied a CNN [3] classifier to detect the segmented characters and the predictions are combined from left to right to get our result.

The paper is organized as follows: In the second section, we explain the system architecture that an ANPR system generally follows. In section three, we review other related approaches that have been used in ANPR. Then in the next sections i.e 4,5 and 6, we describe the dataset, the methodology we used and the result that we have obtained. Finally, we present the conclusion in section 7.

II. SYSTEM ARCHITECTURE

The system architecture defines the structure and behaviour of a system. The figure above represents a usual pipeline of an ANPR system. The system receives the video or image input and after processing it will provide us with the recognition result.

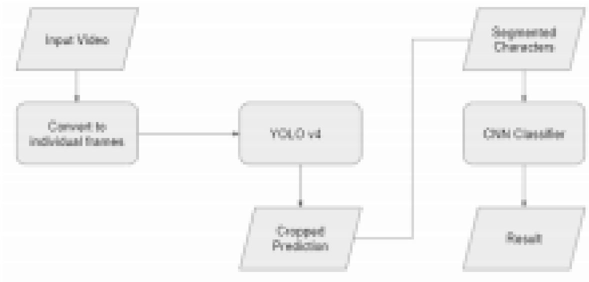

Fig. 1. ANPR pipeline

The input of the system is expected to be real time vehicle surveillance videos based on the application. This system however can work on any type of video or image input. The input is first given to the detection system which generates a bounding box around the licence plate and crops it. Each character of the licence plate is segmented using image processing techniques and each character is fed to the CNN classifier which predicts the characters on the licence plate.

III. RELATED WORK

ANPR has seen a lot of research. In this section, we discuss several approaches on licence plate detection and recognition that use deep learning. We review approaches specific to each stage in ANPR separately and we present a detailed walkthrough of the advancements in ANPR.

A. Licence Plate Detection

Traditional methods in licence plate detection mainly focus on feature extraction from the scene. Later, after the advancements in Deep learning, many authors have implemented object detectors based on Convolutional Neural Networks. In [4] - [5], the authors have discussed using two separate CNN's for detecting the licence plate. The first scene will detect the vehicle from the scene and the second CNN will then detect the licence plate on the vehicle. They use Fast-YOLO for vehicle detection and YOLO v2 for detecting the licence plates. This method reduces false positives. Both the vehicle as well as the frame coordinates are used for training the detectors. They also adjusted the filters from 30 to 35 in order to match the number of classes YOLO uses and achieved. They had to lower the threshold to 0.125 to get a precision of 99% and a recall of 100%.

In [6], Abhishek Kashyap and B. Suresh, implemented a combination of edge statistics and morphology detection image processing algorithms to obtain a binary of the input image and separate the licence plate from the scene. They applied contrast extension and median filtering on 9745 images to obtain a detection accuracy of 98% given that the licence plate edges are clearly defined and that there is no other text present in the input scene.

Detecting multiple licence plates [7] is important in complex situations. The licence plate region consists of a huge number of vertical edges which can be used as a major feature in the detection process. So techniques usually employ gray scale conversion to reduce the noise in the image. Aiswarya Menon and Bini Omman [8], used infrared cameras to capture the scene and then applied gray scale conversion, Sobel filter and morphological operations on the image to extract the licence plate.

Different from the traditional approaches that divide the licence plate detection and recognition tasks, Li, Wang and Shen [9], introduced a single deep neural network which is a combination of multiple deep convolutional layers for feature extraction, a pooling layer to get the region of interest and an RNN for plate detection. The complete network is trained jointly optimizing the detection loss and the recognition loss so that the entire image can be traversed by a single feed forward network.

The number of convolutional layers in the network heavily determine the detection accuracy of the model. A large number of convolutional layers will cause missing detections and a low number of convolutional layers will lead to false positives. Zhang, Li, Xun and Shan [10] demonstrate that using a model with five convolutional layers gives the best results. They used a stride of 16 pixels, 10 vertical anchors and varied the height of the anchors from 11 to 283. They found out that the confidence score of 0.7 was the most optimal to get accurate licence plate detections.

Naren Babu R, Sowmya V and Soman K P[11] focused their work on Indian licence plates. They have used a single YOLO model to detect and recognize the characters on the licence plate. The YOLO model treated the characters on the plates as objects and separately detected them from the scene. The detections were then combined together to obtain the result. They used a dataset of 36 characters ranging from 'a to z' and '1 to 9' and validated their results on 6500 Indian vehicle images taken both during the day as well as night. They achieved a recognition accuracy of 91%.

B. Character Segmentation

The character segmentation stage is treated as optional in various approaches of character recognition.

Once the plate is localized, the plate is either segmented to get individual characters that are separately fed to the recognizer or the entire plate is given as an input in which case this stage is skipped.

In [4], they use the detection stage CNN model to directly obtain character proposal regions. [6] uses the regionprops function of MATLAB to get the individual characters. The function gives a set of defined boxes around the individual characters as its result. In [8], they use a contour detection algorithm after applying the Sobel filtering and the threshold optimization to the localised licence plate. Image processing algorithms like grayscale filtering and contour detection have always been at the forefront when it comes to dividing the licence plates to obtain individual characters. In [9], they trained a CNN model using the licence plate images and the coordinate values as inputs along with a set margin. They obtained a recall of 98.33 when the margin was set to 10. In [11], the yolo model is trained to detect characters as objects, so the result of the first stage are segmented characters.

C. Character Recognition

Character recognition is the most important stage of the entire process. The accuracy of the entire approach depends on

the accuracy of the character recognizer model. The characters must be recognized in real time accurately for the applications based on licence plate recognizer work. Many supervised learning and deep learning algorithms have been employed to obtain accurate results.

In [4], the character localisation stage is followed by the character recognition stage. A padding of 2 pixels for alphabets and 1 pixel for numbers is applied to the segmented characters before they are fed to the YOLO model. Later a temporal redundancy check was applied to the input image and the accuracy of the system improved from 85.45% to 93.53%.

In [6] use template matching which finds out similarity between a given template and the input image. A database of characters is stored and the input image is compared with each image in the database pixel by pixel and the template with which it is the most similar is the result. This method produced an accuracy of 82.

[8] uses a multi layer perceptron after a histogram equalization and contour detection processes. The artificial neural network uses weights and activation functions on hidden layers to generate the result.

Li, Wang and Shen [9] after combining the three stages into a single feed forward deep neural network used a bidirectional RNN in their recognizer phase. They used a softmax layer at the end and achieved an accuracy of 94.12% on the Caltech Cars dataset.

In [10], Zou, Zhang and other authors use a Bi-LSTM with the input localised character such that the features associated with the character are enhanced and the features unrelated to the character are minimized. The forward computation of the Bi-LSTM takes the input image and the reverse computation takes the complement of the input image. The resultant vectors of both are combined to get the result. This strategy got them an accuracy of 95.3%.

The proposed YOLO based model in [11] which is trained to detect characters as objects returns the predictions. The predictions are sorted from left to right. They achieved a 91% accuracy in number plate recognition.

Licence plate recognition from video always depends on the quality of the frame. The plate quality and unconstrained and harsh environments impact the accuracy of the recognizer model. To remedy this problem, Zhang, Wang and Li [12] introduced an efficient approach called efficient quality aware LPR. They used a siamese architecture that gave a recognition accuracy of 92.91%.

Many papers address licence plate recognition on captured images and focus on detecting only a single image at a time. However in real life applications, it is important to detect multiple licence plates from a real time video feed coming from complex unconstrained scenarios. We apply a YOLO v4 algorithm for localization and a CNN based model for character recognition. YOLOv4 is based on a new CSPDarknet53 model with an improved detection accuracy and precision. That helps detect multiple licence plates from video streams more accurately.

IV. DATASET OVERVIEW

The major challenge in a deep learning problem is to obtain the dataset required to train the models. In our approach, we used specific datasets to train the detector model and the optical character recognizer model.

A. The UFPR-ALPR dataset

This dataset contains 150 videos recorded for one second each at a frame rate of 30fps. That makes it a total of 4500 frames. Fig 2 demonstrates some of the frames from the dataset. Each frame is accompanied with a text file that includes the coordinates of the licence plate on the frame. The videos are recorded using portable cameras and smartphones. The camera quality isn't the same but the frame rate is set to 30 for every recording. The dataset is divided into training, testing and validation sets of 40%, 40% and 20% respectively.

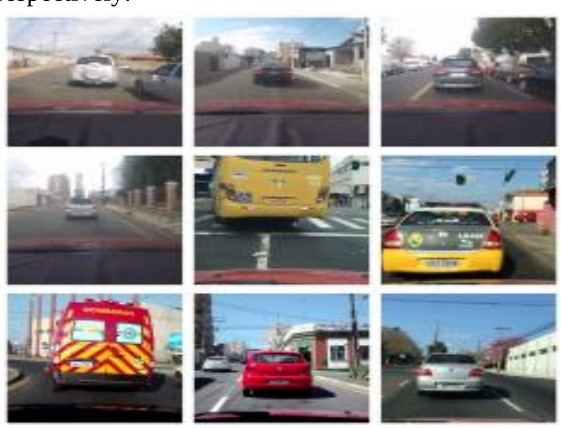

Fig. 2. The UFPR-ALPR dataset.

B. Characters for CNN

We put together a dataset of 36 classes, i.e 26 alphabets from 'A to Z' and 10 digits from '0 to 9'. Each class has over a 1000 images. And a separate validation set of 120 images was created to test the accuracy of the model.

V. METHODOLOGY

In this paper, we have proposed the use of YOLOv4 for localizing the licence plate and CNN classifier for optical character recognition. In this section, we discuss the detailed approach we used to develop the proposed ANPR system.

A. Licence Plate Detection using YOLO

Object detection is a task of localizing and classifying an object from an image which consists of multiple objects. YOLO is different from traditional object detectors because it treats object detection as a regression problem. It uses a single convolutional neural network to generate bounding boxes in the proposal region. It consists of 24 convolutional layers followed by 2 fully connected layers. YOLO is very generalised and needs to be trained on our problem specific dataset to get a specialised model.

We train the YOLO v4 model to detect a single class. i.e. the licence plate. For that we need a well labelled dataset. We use the UFPR-ALPR dataset which has the images along with the proper labels of the coordinates of the licence plate in the image. We use a GPU and train on darknet, an open source object detection framework that runs on YOLO. The number of batches and filters depend on the number of classes. Since we only have a single class, the number of batches is set to 2000 and the number of filters is set to 18.

Once the model is trained, we can provide it with new input images or videos and the model will detect the coordinates of the licence plate in the scene. The detections in the video are cropped and stored to be given as input to the next step.

B. Character Segmentation

Once we obtain the cropped licence plate, we resize it to 333 x 75 dimension so that all the characters on the plate are clear and distinct. Then we apply grayscale conversion and binarization with a threshold of 200. We then remove unwanted pixels from the image. This process reduces the noise in the image and makes it easier for character extraction. We have a clean binary image on which we apply contour detection. We consider only the contours which have a width ranging from 0 to (width of plate/number of characters) and a height of (height of plate/2) to 4(height of plate)/5. Now we have each character segmented correctly which can be fed to the recognizer model.

C. Optical Character Recognition

Indian licence plates have a specific structure, with the first two letters denoting the state. The next two are digits followed by one or two letters and four digits. This gives a very limited amount of characters that can appear at a specific position. For modelling the character recognizer we used a convolutional neural network with 3 layers. We start by creating a sequential object. We used a kernel size of 5x5 with 32 output filters in the first layer and a RELU activation function. Then we add a maxpooling layer of size 2x2 and a dropout rate of 0.4. Finally, we added 2 dense layers, One with a dimensionality of output space as 128 and the final layer with 36 outputs and the RELU activation function.

To train the model, we used a data split of 80:20 for the training and validation set. We used Adam as the optimization function and categorical cross entropy as the loss function. We achieved a training accuracy of 99.54% using this model.

D. Video Processing

Real time processing of frames is important for the model to be applicable in practical scenarios. To build the model to work on real time streams, we used 20 videos of 3-4 seconds each. We extracted each frame from the video and gave it as an input to the model. The same number plates would be extracted in every frame. To avoid redundancy, the result is compared each time with the previous output. If the result is similar, it can be stored as a single result. Our model was also able to detect multiple licence plates from video streams.

VI. EXPERIMENTAL RESULTS

In this section, we present the results we obtained while testing our system. All the tests were carried out on a Nvidia 940m GPU.We compiled a dataset of 250 Indian traffic scenes and 20 video streams to test the accuracy of the system.

A. YOLO Detection

The custom YOLOv4 licence plate detection model based on the Darknet framework which we used is very accurate in detecting licence plates. The detection process is depicted in fig 3. The model was accurately able to generate a bounding box around the licence plate in almost every test. We did generate some false positives but the model detected every licence plate. The detected licence plate is cropped and characters are segmented using binarization and contour detection. The false positives won't contain numbers hence will be eliminated in this step. The recall and precision of the model was an exceptional 100% during the tests. Even during the testing of video frames, the model generated cropped licence plates from the scene to be fed to the classifier model.

Fig. 3. YOLO Detection.

B. CNN Classifier

The CNN classifier model result is calculated by the number of characters it predicts correctly. The result is obtained as a command prompt output and then is superimposed on the image above its bounding box as shown in the figure below.

Adding a check for character and digit classification helped resolve the confusion the model can have between predictions for ’0’ and ’O’, ’1’ and ’I’, ’2’ and ’Z’ and many others to provide a more accurate result. We measured the reliability of the system by the number of predictions it got wrong out of all the characters in a number plate.

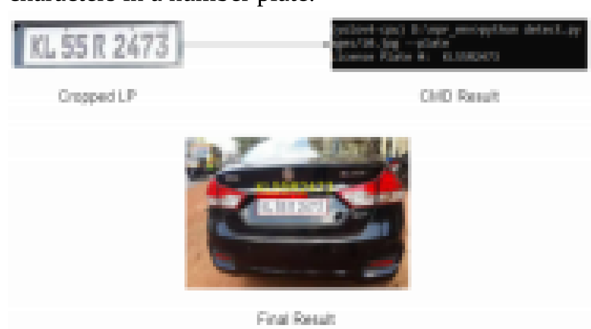

Fig. 4. CNN Recognition

Predictions wrong	Classifier Result	Average time
Zero	176	6.49s
One	14	5.38s
Two	4	5.76s
More than two	6	6.23s

In terms of computational speed, the method is a little slow but the results are pretty accurate. This method however does not consider licence plates which are not in the standard format.

VII. CONCLUSION

We have proposed an efficient two stage end-to-end approach for detecting and recognizing multiple licence plates using YOLOv4 and CNN in complex unconstrained scenarios for Indian vehicle surveillance applications. We implemented a custom YOLOv4 module based on darknet to detect licence plates and achieved a precision and recall of 100%. The model contained some false positives which would be eliminated in the next step. We then segmented each character from the plate. The segmented characters are fed to the CNN classifier model which has 36 output classes. We applied a character check according to the standard Indian number plate to determine if the output should be a letter or a digit. However, the speed of execution of the system still needs to be improved for it to be able to work accurately in real time.

We wish to explore more cnn based character recognizer architectures as a future work which would be able to detect characters even in blurred licence plates or detections which are taken from a distance which makes the segmented characters smaller in size and unreadable. This is an efficient and robust system which can be developed at a very low cost and has a wide range of real time applications.

REFERENCES

[1] P. Kulkarni, A. Khatri, P. Banga and K. Shah," Automatic Number Plate Recognition (ANPR) system for Indian conditions," 2009 19th International Conference Radioelektronika, 2009, pp. 111-114, doi: 10.1109/RADIOELEK. 2009. 5158763.

[2] Bochkovskiy, Alexey & Wang, Chien-Yao & Liao, Hong-yuan, "YOLOv4: Optimal Speed and Accuracy of Object Detection.," arXiv: 2004.10934v1 [cs.CV] 23 Apr 2020

[3] N. Sarika, N. Sirisala and M. S. Velpuru," CNN based Optical Character Recognition and Applications," 2021 6th International Conference on Inventive Computation Technologies (ICICT), 2021, pp. 666-672, doi: 10.1109/ICICT 50816. 2021. 9358735.

[4] Rayson Laroca, Evair Severo, Luiz A. Zanlorensi, Luiz S. Oliveira, Gabriel Resende Gonc¸alves, William Robson Schwartz, David Menotti, "A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector," 2018 International Joint Conference on Neural Networks (IJCNN), 2018, pp. 1-10, doi: 10.1109/IJCNN.2018.8489629.

[5] C. K. Sahu, S. B. Pattnayak, S. Behera and M. R. Mohanty,"A Comparative Analysis of Deep Learning Approach for Automatic Number Plate Recognition," 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2020, pp. 932-937, doi: 10.1109/I-SMAC49090.2020.9243424.

[6] A. Kashyap, B. Suresh, A. Patil, S. Sharma and A. Jaiswal," Automatic Number Plate Recognition," 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), 2018, pp. 838-843, doi: 10.1109/ICACCCN.2018.8748287.

[7] C. -H. Lin and C. -H. Wu,"A Lightweight, High-Performance Multi Angle License Plate Recognition Model," 2019 International Conference on Advanced Mechatronic Systems (ICAMechS), 2019, pp. 235-240, doi: 10.1109/ICAMechS.2019.8861688.

[8] A. Menon and B. Omman," Detection and Recognition of Multiple License Plate from Still Images," 2018 International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET), 2018, pp. 1-5, doi: 10.1109/ICCSDET.2018.8821138.

[9] 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.

[10] J. Zhang, Y. Li, T. Li, L. Xun and C. Shan," License Plate Localization in Unconstrained Scenes Using a Two-Stage CNN-RNN," in IEEE Sensors Journal, vol. 19, no. 13, pp. 5256-5265, 1 July1, 2019, doi: 10.1109/JSEN.2019.2900257.

[11] R. Naren Babu, V. Sowmya and K. P. Soman," Indian Car Number Plate Recognition using Deep Learning," 2019 2nd International Conference on

Intelligent Computing, Instrumentation and Control Technologies (ICICICT), 2019, pp. 1269-1272, doi:10.1109/ICI CICT46008.2019. 899 3238.