2019 International Conference on Computing, Communication, and Intelligent Systems(ICCCIS)

Real Time Indian License Plate Detection using Deep Neural Networks and Optical Character Recognition using LSTM Tesseract

Jaskirat Singh
Dept. of Computer Science and Engineering
HMR Institute of Technology and Management
New Delhi, India
jkrt.ngh69@gmail.com

Bharat Bhushan
Dept. of Computer Science and Engineering
HMR Institute of Technology and Management
New Delhi, India
bharat_bhushan1989@yahoo.com

Abstract—**Among the ranking of the largest road network in the world, India stood at third position. According to a survey held in 2016 the total number of vehicles in India were 260 million. Therefore, there is a necessity to develop Expert Automatic Number Plate Recognition (ANPR) systems in India because of the tremendous rise in the number of automobiles flying on the roads. It would help in proper tracking of the vehicles,expert traffic examining, tracing stolen vehicles, supervising parking toll and imposing strict actions against red light breaching. Implementing an ANPR expert system in real life seems to be a challenging task because of the variety of number plate (NP) formats,designs, shapes, color, scales, angles and non-uniform lightning situations during image accession. So, we implemented an ANPR system which acts more robustly in different challenging scenarios then the previous proposed ANPR systems.The goal of this paper,is to design a robust technique forLicense Plate Detection(LPD) in the images using deep neural networks, Pre-process the detected license platesand performLicense Plate Recognition (LPR) usingLSTMTesseract OCR Engine. According to our experimentalresults, we have successfully achieved robust results withLPD accuracy of 99% and LPR accuracy of 95%just like commercial ANPR systemsi.e., Open-ALPRand Plate Recognizer.**

Keywords—ANPR,ALPR,LSTM, LPR, OCR, Tesseract, Open-ALPR,NPRS,LPD,LPS,AI,ML,DL,RCNN,NP,IOT,LP,NPL,NPD,CS,CR,CNN, YOLO,SVM, COCO,SSD, RPN

I. Introduction

In recent years, Automatic License Plate Recognition (ALPR) hadcontributed a pivotal part in the growth of smart towns as a supervision system for automobile tracking, trafficregulation, enforcing strict traffic rules and policies [1, 2]. ALPR has been tremendously used in traffic toll system ,smart parking system and security systems across many countries. The recent advances in Artificial Intelligence(AI),Machine Learning(ML), Deep Learning (DL) especially neural nets and Parallel Processing have contributed to the development of numerous Computer Vision projects, such as Object Identification, Object Detection and Optical character recognition(OCR), which benefitted ANPR systems. Undoubtedly, Region-based Convolutional Neural Networks (RCNNs) have been the dominant ML technique applied for automobile and Number Plate (NP) detection [3, 4]. Besides Research papers, many commercialized ANPR systems have been immensely using DL Algorithms. These expert systems are situated in large cloud data centres and manage their work through webservices, processing thousands to millions of images every day and constantly upgrading. As famous examples of these systems,we can mention Plates-Smart Technologies like Open-ALPR,GenetecAuto-Vu ALPR,Samsung ANPR system, Plate Recognizer, Vax-ALPR.

ALPR is still an open problem becauseof the huge diversity in image Acquisition state (lightning condition, capturing angle, distance from camera, clarity and quality of image in terms of resolution of pixels) and Number plate(NP) format, which differs for different countries. Although ANPR has been integrated to smart parking and toll systems, it still faces a lot of problems in the supervision system such as crowded traffic with many NPs, unknown advertisements and signs, rotated and tilted NPs, as well as unclear images captured in night-time and bad weather. These variations result in false positives on plate detection and poor LPR accuracy. Despite the advancements in the commercial ALPR systems, they mostly captureautomobiles with frontal view and Number Plates(NP) which is common in applications such as smart parking system, Internet of Things (IOT) security systems, smart toll management systems. However, more informal image acquisition scenarios may lead to uneven views in which the NP might be highly oblique yet still able to read the NP and for which even the commercialized ALPR systems struggle.To solve these issues, we proposed a robust approach for Number Plate Recognition System (NPRS) that firstly detects the LP with DL techniques and then retrieves the detected LP[5,6,7]. Our major input is the integration of a Faster RCNN Networkwhich is able to successfully identify the NPs in numerous different scenarios and with an efficient detection accuracy. An additional contribution is, permitting the RCNN training from scratch using 1000 manually annotated images. The cropped License plate goes through a series of pre-processing methods that improves the quality for robust results

ISBN: 978-1-7281-4826-7/19/$31.00 ©2019 IEEE

in later stages. The pre-processing methods like grey scaling, blurring and binarizationare performed on the detected LP[8,9,10]. Then in the final stage, we used Tesseract for OCR from the pre-processed LP.

The remainder of the paper is organised as follows . Section II presents the literature review on various ANPR proposed systems in the past. Section III explains the challenges faced during completion of each phase of the ANPR expert system. Section IV describes the optimized solution approach used for the implementation of ANPR system in detail.Section V highlights the results and discussions of the proposed ANPR system. Section VIpresents the conclusion and open future research challenges in the this proposed ANPR AI System.

II. RELATED WORK

Conventional NPRS has three stages, including Number Plate localization(NPL), Character Segmentation(CS) and Character Recognition(CR). This technique is old and not so precise in terms of accuracy and robustness.Therefore,considering the advancement and development in the field of ALPR we have mentioned the most popular and latest proposed experiments that are reshaping ALPR expert systems.

Montazzolli et al. [11] proposed an expert ANPR system usingdeep neural networks for Brazilian NPs. In the research paper, Two YOLO(You only look once)CNN networks were integrated: the first was Frontal View/Number PlateDetection(NPD) Network which identifies vehicle Frontal Views, and the second was License Plate Segmentation(LPS)/Character Recognition (CR) Network which identifies and recognizes text in a cropped LP.Linet al. [12] have proposed a robust hierarchical NPRS. They concluded that the integration of YOLOv2 model and SVM can capture license plates with high accuracy. The Licence Plate Recognition (LPR) CNN model also has high accuracy in character recognition.Laroca et al. [13] proposed an efficient real-time ANPR expert system using the CNNYOLO API. They used anopen source dataset for ANPR that containsmore than 30,000 NP characters of 150 automobiles, 4500 annotated images.Yosinski et al. [14] explained the filters found by a huge neural network trained for months on a big dataset of thousands of images, can be transferred to a smaller neural network. Their research paper inspired us to deploy deep neural networkin ANPR.Jain et al. [15] worked on an algorithm that is used for feature extraction to identify the NP in the image.To perform detection of vehicle before LPD has been a popular strategy, resulting in reducing the false positive count and search region .This approach is implemented using Support Vector Machine(SVM)for classification and Histogram of Oriented Gradients (HOG) as feature descriptors described in[16,17,18].Local Binary Pattern(LBP),is another popular feature descriptor used by Open-ALPR [19] to detect USA and European LPs.Shu et al. [20] proposed LPD, LPS and CR using three deep neural networks. By a substantial margin, their result surpassed Open-ALPR.

III. PROBLEM STATEMENT

Our main priority in this research paper is to experiment deeply with the neural networks and find optimized solutions to LPS and CR problems within the LPR framework.Firstly, it is required to extract the vehicle images from the dataset and other images from Google using Web Scraping. After extracting the images our next task is to label the LP in the images that are extracted. We labelled the LPmanually with thespecific coordinates in each vehicle image. For this task we decided to use LabelImg software.With the help of LabelImg tool we can label the bounding boxes with the specified coordinates and mention the class of the labelled image which is NP. Whenever we label each and every NP using LabelImg tool the Coordinates of the NP(Xmin, Ymin, Xmax, Ymax), Width/Height of the image and the Class name of the labelled image which is NP is stored in XML format. After Annotating the images, we split the data into train and test samples. Out Of 1000 Images,200 Images (20%) were used as test Images and the rest were used as training images. Now our next taskwas to convert the xml files into distinctive csv file that is finally converted into TFRecord file. There would be two csv files one for test images and other for train images. Same is the case for TFRecord files. The TFRecord files contain the data about the annotated coordinates ,width, height and the labelled name for each LP. TFRecords serves as the input data to the TensorFlow training model.

Lastly, the object detection training config pipeline needs to be configured. It contains information about network architecture and parametersused while training. This is the last step before training the neural network. There are a number of config files out of which we decided to choose Faster R-CNN Inception V2 COCO config. The config files contain the learning rate, number of steps,path to fine tune checkpoint, number of classes to detect, number of examples, train and test record paths, label map path etc. If everything has been set up correctly, then it’s time to run the model for training.The time period for initialization during training varies from 20 to 40s. After that training is complete, the last step is to generate the frozen inference graph (.pb file).The inference graph file contains the LPD classifier.The final trained model can detect and extract LPs from the test images or videos.These extracted images of LPs are then passed to the next stage for extraction of individual characters.For this stage, we wrote our own pre-processing algorithm which involved passing the extracted LPs through a series of pre-processing methods like Grey Scaling, Gaussian Blurring, Binarizing filters with various thresholds followed by segmentation.What makes this Algorithm different is that it works very robustly for different lighting condition and even when some part of the LP is covered in shadows. This algorithm works very efficiently for motor bikes, trucks and car LPs which are either single or two lined. For the next stage we implemented OCR on the segmented LP using Python-Tesseract , or simply PyTesseract, a python library which is known as a wrapper for Google's Tesseract OCR Engine. Tesseract is widely used to recognize the characters from the detected NP. We considered this because it is completely open-source and being developed and maintained by the giant that is Google. The advantage of using Pytesseract module for OCR is that it can recognise alphanumeric characters as well recognise characters of different languages. We used the LSTM based recognition engine of Pytesseract for detection of characters from the pre-processed LP. LSTM is similar to RNN in terms of architecture.The paper works robustly with the standard Indian LPs but the parameters, techniques and algorithms which are used can be tuned easily for any similar LPs of different countries even with other alpha-numeric set.

IV. PROPOSED SOLUTION

The process of ANPR consists of three important steps explored in the subsections below.

A. License Plate Detection(LPD)

The latest ANPR system typically involves LP detection. It is essential in our approach to only detect LP rather than detecting both vehicle and LP. Our approach consumes less time and computation.We designed a strategy to identify theNPs based on an intelligent trained network of annotated LPs . The most important task was to implement a real-time ANPR system, withoutany costly software and hardware requirements. So, we proposed a deep neural network capable of performing detection in a very quick time and with high precision. To achieve our goals, we decided to use a Faster RCNN network that usesRegion Proposal Network(RPN) for object detection. It is much accurate and faster compared to its descendants RCNN and Fast RCNN. It is a slow search selective algorithm with fast neural network. According to the Faster RCNN architecture shown in Fig.1 the input is a vector with a multi-channelled RGB image that runs through a CNN to get a Feature Map(features for the input image).

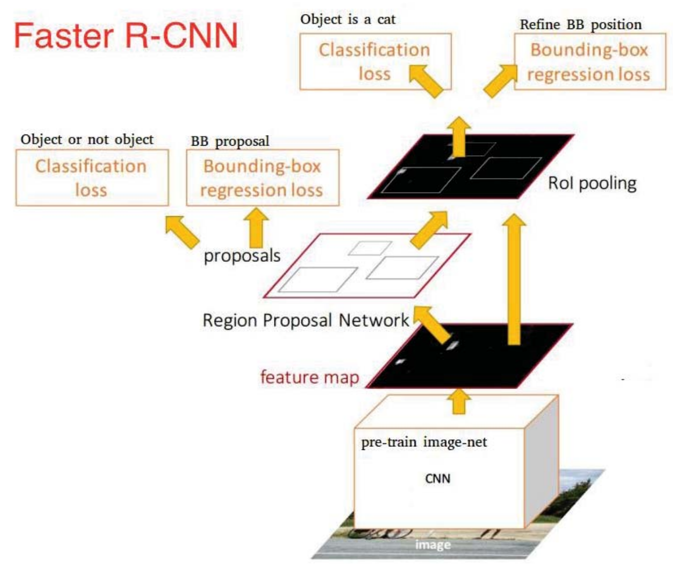

Fig.1. Faster RCNN Architecture

In the Faster RCNN architecture RPN is placed after last Convolutional layer of CNN.The Feature map runsthrough theRPN, that outputs interesting Region Proposals(Anchors/Bounding Box). The predicted Region Proposals from RPN are reshaped using Region of Interest (ROI)Pooling Layer followed by Classifier and Bounding Box Regressor which are used to classify the image within the proposed region and predict the offset coordinates with R-CNN for the Bounding Boxes. Faster RCNN based Inception V2 Coco Model was used for LPD which has a speed of 58ms and COCO map[^1] of 28(COCO map[^1] shows the Accuracy of the network). Inception V2 is a module that was used to reduce the complexity of CNN. It makes CNN wider than deeper. The module also reduced the CNN computation time.

The proposed network turned out to be an adaptable ANPR system that was capable of detecting and recognizing LPs successfully from dataset images using the same system parametrization[21,22,23]. The dataset which was used for implementing ALPR is available in Kaggle Inc and it's known as Vehicle Number Plate Detection provided by Data-Turks[24]. The Dataset is available in json format and contains 353 Indian vehicle images. The labels are divided in single category which is NP. The dataset contains web links of the vehicle images , points of the bounding boxes in each image, width and the height of each the image. Extra 647 Indian vehicle images were also extracted from Google Images and manually labelled.The python library used for LPD was tensor-flow. TensorFlow which is an open source library which is heavily used for Deep learning. It is used for ML applications such as neural networks and also used as a symbolic math library. By using the Tensor-flow GPU, the set of annotated images were sent into the CNN called as Faster RCNN where the metrics such as model learning rate, batch of images sent into the network and evaluation configurations were set. The training phase of the model took several days. At last the model came around with the positive result and detected the LP over the input images with an accuracy of 99%. The detected LP as shown in Fig.2 is extracted from the image and passed to the next stage where pre-processing methods are applied.

Fig. 2. License Plate Detection (LPD) with 99% Accuracy

B. Preprocessing

Image pre-processing is an essential task in an image analysing process. Without proper pre-processing, the recognition will be ineffectual or may give inaccurate results during NPR. The main reason of pre-processing is to enhance the quality and readability of the cropped LP characters that will be forwarded for segmentation and recognition. Various methods that we applied were converting RGB image to Grayscale, reduction of noise and image binarization.

1) RGB to Gray Scale Conversion: Greyscale conversion is used for converting an colourdpicture into black and white image. Coloured pictures don't help us to detect the important edges and other features. So, convertingan image into Greyscale format is of huge importance. Main reason of using this conversion is to reduce the number of colours

2) Gaussian Blurring: After converting picture to grey scale, the next step is to make the image Blur. Image noises are distortion in the image that arises due to fault in camera or result of poor visibility due to changing weather conditions. Noises are also the random variation in the intensity levels of the pixels.Noise can be of different types like Salt noice, Gaussian noise and pepper noise. In this proposed method, we used Gaussian Blur for noise removal. It provides the mechanismfor noise reduction while preserving edges. Mostly we used Gaussian Blur rather than Median Blur, because it seems to give better results

3) Binarization: Binarization is defined as the process of converting an image into an image with two pixels value only i.e. containing white and black pixels. Performing binarization process before detecting and extracting number plate from the image will make the task of detecting license plate easier as edges will be more clear in binary image.Binarization is performed by selecting a threshold value. After selecting the value, we analyse the pixel values in the image. If it's greater than threshold, then make that pixel fully white or black accordingly. This is simple thresholding method which may not yield proper result by selecting a global threshold value. Hence to overcome this, we use an adaptive thresholding or Otsu Thresholding method in which instead of selecting a global threshold value we calculate threshold of smaller region in the image which gives better result. These methods reduce the complexity of captured image (input). Afte Comparing the performances of the two threshold methods we concluded that Adaptive Threshold method was much robust than Otsu Threshold.

C. License Plate Segmentaion (LPS) and Optical Character Recognition (OCR) using LSTM Tesseract

The next step after doing pre-processing of the LP area is segmentation of the plate. If the segmentation is unsuccessful, then a text of the NP may be improperly divided into different items or different characters may be improperly unified along. We are using Pytesseract (Python wrapper for Google's Tesseract OCR) for segmentation and OCR. The module for Python-Tesseract was imported in Python and the results were stored in Mongo database for each image. By using the Google Tesseract-OCR (Package originally developed to scan hard copy documents to filter out the characters from it) the picture undergoes some conversions using computer vision package then the characters are filtered out.There are four different modes of operation chosen using the --OEM option (OEM stands for OCR Engine Mode). We used the Neural nets LSTM (Long Short-Term Memory) engine mode fordetectionofpre-processed cropped LPs.Tesseract LSTM OCR is the most accurate OCR engine on the planet.LSTM is an artificial recurrent neural network (RNN) architecture used in the field of DL. Page Segmentation Mode (PSM) can be very useful when the extra information about the orientation, structure of the text has to be added while segmentation . PSM varies from 1-13.TABLEIshows the description of each PSM mode.For different images we used different PSM modes that ranges from 8 to 13 because of different styles, positions, angles, intensity, width, height of characters in LP.

TABLE I. PSM EVALUATION TABLE

PSM	Description
0	Orientation and script detection (OSD)
1	Automatic page segmentation (APS) with OSD
2	APS excluding OSD and OCR
3	Fully APS, excluding OSD and OCR
4	Assuming a single column of text of different sizes
5	Assuming a single uniform block of vertically aligned text
6	Assuming a single uniform block of text
7	Picture is treated as a single text line
8	Picture is treated as a single word
9	Picture is treated as a single word in a circle
10	Picture is treated as a single character
11	Picture is treated as a Sparse Text. Text is found in a specific order as much as it can.
12	Picture treated as a Parse Text with OSD
13	Raw line. Picture is treated as a single text line, Tesseract-specific hacks are bypassed.

V. RESULTS AND DISCUSSIONS

We can view the improvement of the training by using Tensor-Board.Tensor-Board provides the visualization graphs. The one of the important graphs is the Losses graphs, which provides the overall losses of the trained model over time during training. Training the model took 2 days for GPU enabled OS . For our training on the FasterRCNNInceptionV2 Coco API, loss of the neural network(net)started with 5 and soon reached below 1. We recommended the neural net to train till the lossesreaches below 0.05 consistently, which took 39,431 steps. Fig.3 shows different losses while training of the classifier.

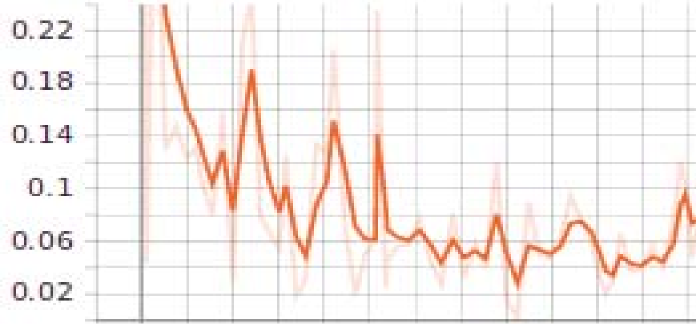

Fig. 3. (a) Box Classifier Classification Loss

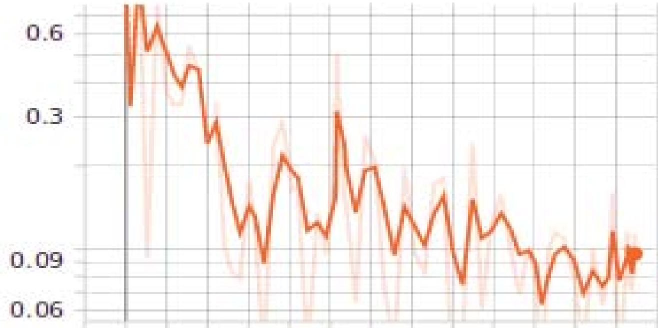

Fig. 3. (b) Box Classifier Localization Loss

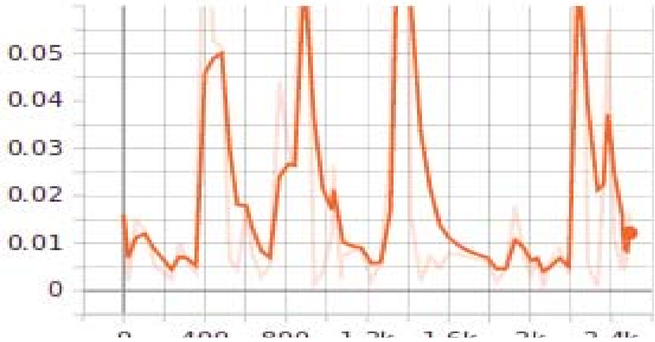

Fig. 3. (c) RPN Localization Loss

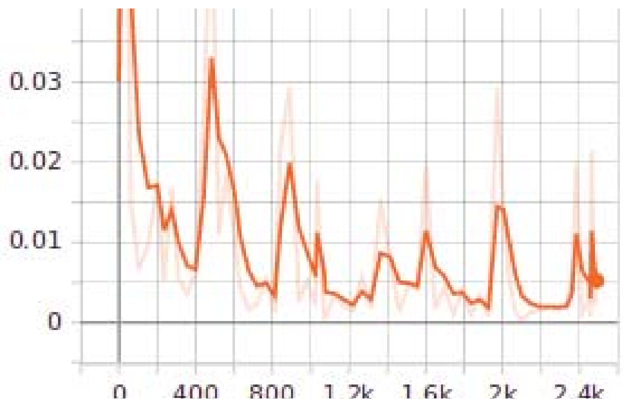

Fig. 3. (d) RPN Object-ness Loss

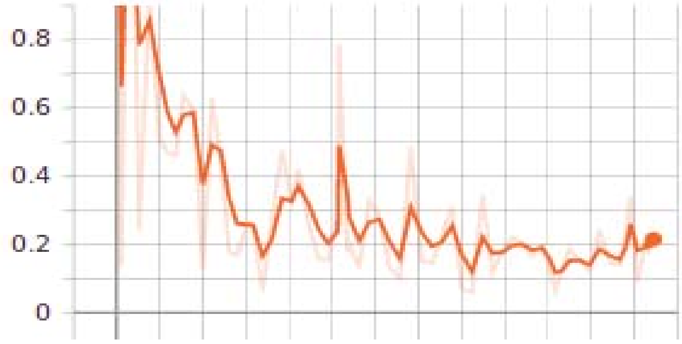

Fig. 3. (e) Total Loss

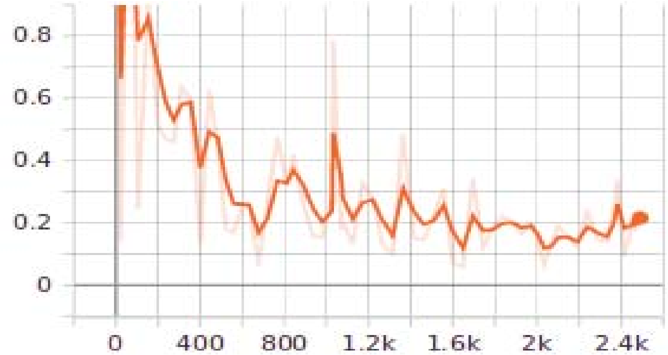

Fig. 3. (f) Clone Loss

Fig. 3. Different Loss Graphs during training of Faster RCNN Model using Tensor-board

The accuracy of the LPD model is 99%. Out Of 1000 Images 200 Images (20%) were used as test Images and the rest were used as training images. Our ALPR AI model successfully recognized 95% of the number plates of the testing dataset.

VI. CONCLUSION AND FUTURE SCOPE

In this research paper we proposed an expert ANPR AI systemfor Indian LPs using DL methodology, Image Pre-processing techniques and OCR . Faster RCNN Inception V2 Coco Model was used for detection of LPs and Tesseract LSTM RNN OCR Engine was used for segmentation and recognizing text within a cropped LP. The accuracy of the Faster RCNN model achieved 99% in the LPD task.Our expert ALPR system correctly recognized 95% of the NPs on the test dataset (all 9 characters).While considering the remaining 5% of the NPs, 98 % out of them recognized partial matches of at least eight characters correctly in a given NP .This methodology shows superiority in both accuracy and performance in comparison with traditional NPRS methods which usually requires a high-resolution camera to capture high quality images and an expensive computer to process the complex algorithms for recognition of NP.

In the first phase we have used Faster RCNN Inception V2 Coco for training the TFRecord files butin future we will be using SSD InceptionV2 COCO, SSD Mobile-net V2 COCO, SSD Mobile-net V1 0.75 Depth COCO ,Faster-RCNN Inception NAS, Mask-RCNN Resnet50Atrous Coco , Mask-RCNN V2Inception Coco, Mask-RCNN ResnetInception V2AtrousCoco, SSD-Mobile-net V1 Coco , Faster-RCNN Inception-Resnet V2Atrous Coco , Faster-RCNN Resnet 50 Coco, SSD-Resnet 50 FPN COCO models etc.In future, we intend to enhance the speed of our ANPR AI model by using an advance engine of OCR which is TopOCR. It is ahigh speed fixed-function Optical character RecognitionEngine which is our first priority for increasing the speed of the ANPR system. It analyses the shape of characters and classifies characters using ultra-high-speed decision tree. It is the fastest OCR engine currently present in the planet.TAO OCR - Tesseract Accelerated OCR is our second priority because it is the most trending recognition engine with high performanceresults,multi-threading and multi-lingual functionalitiesintegrated into the Tesseract OCR Engine. TAO OCR is faster compared to Tesseract's standardLong Short-Term Memory classifier and the accuracy level is almost same in both OCRs.Our next major goal would be to enhance the size of the image dataset. We have planned to use NP datasets of different countries like America , Australia, China , Brazil, India etc and implement a standard ALPR expert system that can detect and recognize License plate of any country or of any language.

REFERENCES

[1] Lee, D., Yoon, S., Lee, J., & Park, D. S. (2016). Real-Time License Plate Detection Based on Faster R-CNN. KIPS Transactions on Software and Data Engineering, 5(11), 511–520. doi: 10.3745/ktsde.2016.5.11.511

[2] Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(6), 1137–1149. doi: 10.1109/tpami.2016.2577031

[3] Silva, S. M., & Jung, C. R. (2018). License Plate Detection and Recognition in Unconstrained Scenarios. Computer Vision – ECCV 2018 Lecture Notes in Computer Science, 593–609. doi: 10.1007/978-3-030-01258-8_36

[4] Gou, C., Wang, K., Yao, Y., & Li, Z. (2016). Vehicle License Plate Recognition Based on Extremal Regions and Restricted Boltzmann Machines. IEEE Transactions on Intelligent Transportation Systems, 17(4), 1096–1107. doi: 10.1109/tits.2015.2496545

[5] Rahati, S., Moravejian, R., Kazemi, E. M., & Kazemi, F. M. (2008). Vehicle Recognition Using Contourlet Transform and SVM. Fifth International Conference on Information Technology: New Generations (Itng 2008). doi: 10.1109/itng.2008.136

[6] Rafique, S., Iqbal, M., & Habib, H. A. (2009). Space invariant vehicle recognition for toll plaza monitoring and auditing system. 2009 IEEE 13th International Multitopic Conference. doi: 10.1109/inmic.2009.5383107

[7] Goncalves, G. R., Diniz, M. A., Laroca, R., Menotti, D., & Schwartz, W. R. (2018). Real-Time Automatic License Plate Recognition through Deep Multi-Task Networks. 2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI). doi: 10.1109/sibgrapi.2018.00021

[8] Sung, M.-C., Jun, B., Cho, H., & Kim, D. (2015). Scene text detection with robust character candidate extraction method. 2015 13th International Conference on Document Analysis and Recognition (ICDAR). doi: 10.1109/icdar.2015.7333797

[9] Sung, M.-C., Jun, B., Cho, H., & Kim, D. (2015). Scene text detection with robust character candidate extraction method. 2015 13th International Conference on Document Analysis and Recognition (ICDAR). doi: 10.1109/icdar.2015.7333797

[10] Cho, H., Sung, M., & Jun, B. (2016). Canny Text Detector: Fast and Robust Scene Text Localization Algorithm. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). doi: 10.1109/cvpr.2016.388

[11] Matas, J., Chum, O., Urban, M., & Pajdla, T. (2002). Robust Wide Baseline Stereo from Maximally Stable Extremal Regions. Procedings of the British Machine Vision Conference 2002. doi: 10.5244/c.16.36

[12] Silva, S. M., & Jung, C. R. (2017). Real-Time Brazilian License Plate Detection and Recognition Using Deep Convolutional Neural Networks. 2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI). doi: 10.1109/sibgrapi.2017.14

[13] Lin, C.-H., Lin, Y.-S., & Liu, W.-C. (2018). An efficient license plate recognition system using convolution neural networks. 2018 IEEE International Conference on Applied System Invention (ICASI). doi: 10.1109/icasi.2018.8394573

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

[15] Yosinski, Jason, Clune, Jeff, Bengio, Yoshua, … Hod. (2014, November 6). How transferable are features in deep neural networks? Retrieved from https://arxiv.org/abs/1411.1792.

[16] A. Jain and J. Kundargi, "Automatic Number Plate Recognition Using Artificial Neural Network", *Semanticscholar.org*, 2019. [Online]. Available: https://www.semanticscholar.org/paper/Automatic-Number-Plate-Recognition-Using-Artificial-Jain-Kundargi/0102638e8a5343f56856a3eb97796a1459640d93.

[17] Hsieh, J.-W., Chen, L.-C., & Chen, D.-Y. (2014). Symmetrical SURF and Its Applications to Vehicle Detection and Vehicle Make and Model Recognition. IEEE Transactions on Intelligent Transportation Systems, 15(1), 6–20. doi: 10.1109/tits.2013.2294646

[18] R.f, P., G, C.-C., Schwartz, W. R., & D, M. (2013). Brazilian License Plate Detection Using Histogram of Oriented Gradients and Sliding Windows. International Journal of Computer Science and Information Technology, 5(6), 39–52. doi: 10.5121/ijcsit.2013.5603

[19] Goncalves, G. R., Menotti, D., & Schwartz, W. R. (2016). License plate recognition based on temporal redundancy. 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC). doi: 10.1109/itsc.2016.7795970

[20] M. Hill, “Open-ALPR,” https://github.com/openalpr/openalpr

[21] Masood, Zain, S., Shu, Guang, Afshin, Ortiz, & G., E. (2017, March 28). License Plate Detection and Recognition Using Deeply Learned Convolutional Neural Networks. Retrieved from https://arxiv.org/abs/1703.07330.

[22] Du, S., Ibrahim, M., Shehata, M., & Badawy, W. (2013). Automatic License Plate Recognition (ALPR): A State-of-the-Art Review. IEEE Transactions on Circuits and Systems for Video Technology, 23(2), 311–325. doi: 10.1109/tcsvt.2012.2203741

[23] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). doi: 10.1109/cvpr.2016.91

[24] DataTurks. “Vehicle Number Plate Detection.” Kaggle, 13 July 2018, https://www.kaggle.com/dataturks/vehicle-number-plate-detection.