**Intelligent Automatic Number Plate Recognition System**

Shivam S1; Aswani Kumar Cherukuri1\*, Member IEEE; Firuz Kamalov2; Ashraf Khalil3; Amril Nazir3; Annapurna Jonnalagadda4; and Chee Ling Thong5

1School of Information Technology and Engineering, Vellore Institute of Technology, Vellore 632014, India.

2Department of Electrical Engineering, Canadian University Dubai, UAE

3 Department of Information Systems and Technology Management, Zayed University, Abu Dhabi, UAE

4 School of Computer Science & Engineering, Vellore Institute of Technology, Vellore, India.

5 Institute of Computer Science and Digital Innovation, UCSI University, Malaysia

Corresponding author: Aswani Kumar Cherukuri (e-mail: cherukuri@acm.org).

**ABSTRACT.** Automatic Number Plate Recognition (ANPR) systems were created in order to address the challenges created due to rise in the volume, velocity and density of automobile vehicles. Most modern ANPR systems heavily depend on image processing techniques like contouring and gray-scaling to segment the license plate and use optical character recognition to extract the number from the manipulated image of the number plate. The major advantage of such systems is that they require less computational power and are quite cost-effective. But these systems process the complete images rather than just the region of interest. The more modern ANPR systems rely on object detection algorithms to overcome this challenge and process only the area of interest from the image. But these systems are not suitable for real-time applications due to high computational overhead and high processing time. To overcome the limitations of the existing ANPR systems, we propose an intelligent ANPR is a system for detecting and extracting number plate details from images or sequences of images of vehicles. The proposed deep learning model is based on an improved version of the You Only Look Once (YOLOv4) algorithm. The performance evaluation demonstrates that our model is able to recognize the number plates with an accuracy of above 95% under varied conditions such as high-speed moving vehicles, varying lighting, and vehicle dense area on Indian roads. The system is robust enough to detect the number plates of fast moving vehicles (speed≤ 80km/hr) as well as vehicles in highly traffic dense areas.

***Keywords:*** Automatic Number Plate Recognition, Deep-learning, You Only Look Once, Optical Character Recognition, Smart detection devices

1. **Introduction**

Automatic Number Plate Recognition (ANPR) is a high-precision technology that reads automobile license plates without the need of human involvement. The ANPR system can be used in multiple applications such as:

* *Traffic Management:* The rise in vehicle numbers has led to an urgent need to control the traffic to prevent accidents. Speed monitoring, Traffic rules enforcement, and setting up Limited Traffic Zones are just a few examples of operations in traffic management[1,2].
* *Police surveillance:* The capacity for police patrol cars to identify and extract number plates and their related information on the go in real-time with a dedicated black box [3,4].
* *Parking Management:* Identifying a vehicle and issuing a parking ticket automatically will be more economical and efficient than the manual process.
* Tolling: Contact-less toll tax collection and security surveillance are the challenges that our Intelligent ANPR can solve [6,7].
* Data Collection: Collecting vehicles data from a region will help in developing rules and regulations for that particular region and can be used in multiple surveys on serious topics like accidents [8,9].

Due to its huge real time applications, it attracted the attention researchers from various domains. Some of the renowned methodologies reported in the literature are categorized and presented in the following Table-1.

Table-1: Categorization of various methodologies reported in the literature

|  |  |  |  |
| --- | --- | --- | --- |
| S.No | Method | Methodology | Limitations |
| 1 | *Edge Detection Method* | * A set of mathematical techniques are used to detect the edges / curves based on the abrupt change in the brightness of the image or discontinuities. * Once the object characteristics are isolated/ extracted and after the boarders have been established, the image may be analysed for object recognition. | * It is fast but less accurate * Performance can vary with different conditions like angle at which the image is captured. |
| 2 | *ROI method* | * A region is set for images and only the region of interest is processed to detect and extract the Number Plate to reduce the complexity of computation. | * It is better than the edge detection method in terms of accuracy but performance varies with different conditions like exposure etc. |
| 3 | *Characteristic-based method* | * It depends on multiple characteristics like plate size, color, and so on to localize a vehicle in an image. * Based on Alphanumeric character features, the relative location of characters can be found in the image and extracted. | * It is a better approach than the previous two approaches. * The order of the detected number plate can be jumbled. |
| 4 | *Morphological-based method* | * Morphological operations are very important functional methods for surface texture assessment. * They are operators used to extract varied regions based on properties like the contrast from an Image. * These detected regions can be matched with templates to extract the number plate. | * It is only limited to restricted characters. * If a new character is present in the image it won't extract it. * Apart from this, computational overhead makes it unsuitable for real-time implementation. |
| 5 | *Machine learning-based* | * Multiple machine learning models have been introduced for object detection. * One such method is connected component labelling (alternatively Connected Component Analysis) is an algorithmic application of graph theory, where subsets of connected components are uniquely labelled. * These detected and segmented regions are then compared with original images using models like SVM | * A good dataset is essential for an accurate machine learning model. * ML models are less versatile when compared to DL models as DL models have a dense architecture which can handle multiple conditions with ease. |
| 6 | *Deep-Learning*  *Based* | * The Deep Neural Networks have led to a revolution in the field of Computer Vision due to faster and highly accurate models which are capable to detect multiple objects under varied conditions. * The models include SSD, YOLO family, Faster-RCNN, etc. | * Some models might have higher computation overhead which might not be good for real-time implementation. |

The evolution of technology over the years has resulted in multiple approaches for ANPR as discussed in Table-1. Each real-time ANPR system should fulfil the factors that are discussed below:

* ***Reliability:*** Even while building a trustworthy security gate and barrier system is a proven technique for preventing crime and safeguarding our property from invaders, it may be a time-consuming and tedious operation. However, if an ANPR system is in place, cars will automatically be allowed entry to locations since the system can match a vehicle's number plate to the main system. The system can be available 24/7 hours.
* ***Efficient Management:*** Vehicle number plate tracking may be a very effective approach to keep track of people's clock-in and clock-out timings. Because ANPR systems assist manage access control in conjunction with gates and barriers, our system can save time and streamline management procedures. Individuals and their cars can be tracked as they enter and exit the premises using ANPR technology. This information may then be saved and used for personal reference if an individual leaves or enters a place.
* ***Improved Security:*** In comparison to manual checks, which are known to have a larger risk of breaching security protocols if a human error is present, automated ANPR systems will provide higher levels of accuracy and security. By relying on modern technology and 99 percent accuracy to recognize illegal cars entering the premises, ANPR systems provide peace of mind to businesses.
* ***Real-Time Analysis:*** Historical data, as well as live data and reporting, are available to organizations. This information can then be saved on a device for future reference and proof in the event of an incident or accident.
* ***Reduction of Corruption:*** An Intelligent ANPR reduces human intervention to a high extent which in turn reduces the chances of human errors occurring. Corruption is a big social evil that can be eradicated by reducing human intervention in the traffic maintenance force. This system helps authorities to enforce laws more stringently.
* ***Easy Installation:*** ANPR systems are easy to set up with novice technical skills required to install as well as operate them. An ANPR basically consists of 3 components: high-definition camera to capture clear and detailed media, Computer with ANPR software installed, and a database to store and process information.

The organisation of this paper is followed as: Section 2 analyzes the literature review and details the motivation & objectives of the current research. We discuss proposed model in Section 3 followed by the description of data set in Section 4. Results are analysed in Section 5 follwed by conclusions and future prospects.

1. **Literature Review**

The rising prosperity of urban India has made automobile ownership a must. As a result, an unanticipated municipal issue has arisen: traffic regulation and vehicle identification. Over the last few decades, ANPR systems have been a hot topic of research. There have multiple ANPR systems in the market.

PK Suri **[10]** used Image processing and Edge detection method like Sobel to detect the license plates. It involves basic colour conversion, edge detection, and connection measuring to detect number plates from an image. The edge detection approach was further enhanced by R Ghosh et.al **[11]** throughregion-based filtering to improve the accuracy of the model. In the 'Region-Of-Interest (ROI)'-based filtering approach, number plate occurrence regions are discovered in the candidate regions in the Number Plate (NP) pictures by identifying vertical edges, then deleting long edges and stationary regions. Finally, the NP region is separated from the candidate regions before being sent to an Optical Character Recognition (OCR) system for recognition of the number plate's characters and digits. But the retrieved number, however, may not be as clear and precise as it should have been, due to incorrect or variable lighting effects and poor localization.

Chauhan and Govindan **[12]**, introduced a system for locating Indian number plates based on characteristics. Because number plate standards are not fully followed in India, there are many variances in factors such as number plate size, position of characters on number plate and so on. All of these discrepancies make the process of locating number plates much more complex. There are four modules involved in the proposed method of localization. The first three modules are based on alphanumeric character features, while the final step is based on the relative location of characters on the license plate. The system was able to put to the test on a series of photos with a broad range of lighting conditions, character sizes, and background and foreground colours and it performed well with localization simulation in MATLAB taking an average of 0.245 seconds. The most common sources of inaccuracy were an extremely slanted number plate, a non-English script, and significant variance in the proportions of the letters, such as very big or very small characters in a license number plate, all of which may be effectively eliminated by improving the technique.

Apart from image processing and characteristic scanning, morphological and template matching approaches are also introduced in ANPR systems. Kasaei et.al., **[13]**, proposed a system in which the key stage of is the separation of the license plate from a digital picture of the automobile. The picture was taken by a digital camera under various conditions such as lighting, slope, distance, and angle. Pre-processing and signal conditioning are the first steps of the method. Next, the license plate is located using morphological operators. The numerals and characters on the plate were then be recognized using a template matching methodology. However, it is only limited to restricted characters if a new character is present in the image it won't extract it and apart from this computational overhead makes it unsuitable for real-time implementation. Multiple issues including varied light conditions, skewed angles, the computational overhead, etc from edge detection and morphological approaches gave rise to supervised learning models to overcome such issues.

S Asthana et.al. **[14]**, proposed a three-hidden-layer multilayer feed-forward back-propagation algorithm. This approach divides an image into sub-images, each holding a single character. The sub-images are then converted to a binary format and this binary data is then sent into a neural network that has been trained to associate the character image data with a number value. The neural network's output is then converted to text. The performance of the algorithm was evaluated over real-world photos. However, the number of epochs is proportional to the number of hidden layers for an approach. This indicates that as the number of hidden layers grows, the network's training process slows down due to the increased number of epochs.

In order to enhance detection in low light and overexposure situations, a novel approach is presented by Babbar et.al. **[15]**, using Connected Component Analysis (CCA) and Threshold Modification. The picture of the car is collected and pre-processed utilizing grayscale and binarization algorithms. The resulting picture is sent to CCA for plate localization and number plate extraction. CCA and ratio analysis are used to segment the characters on the number plate. Finally, approaches such as SVC (linear), SVC (poly), SVC (RBF), KNN, Extra Tree Classifier, LR+RF, and SVC+KNN are used to compare the identified characters. This "Threshold Modification" approach was successful in recognizing number plates even in low light, extreme brightness, and other situations where prior machine learning models failed. SVC (Linear) had the best accuracy, recognizing 97.1 percent of segmented characters correctly. The method works well for detecting number plates at skewed angles as well but lacked in terms of real-time applications.

The boom in the deep learning field in 2010’s gave rise to multiple state-of-the-art object detection models. These deep learning models were faster and more accurate than the prior machine learning models. One such object detection model was YOLO introduced by Redmon et.al. **[16]**. It is a one-stage detection system that detects and classifies objects in a single step while moving across the network. YOLO pushes the boundaries of real-time object detection. It is also adaptable to new domains, making it excellent for applications that need quick and reliable object recognition. Although YOLO is one of the leading real-time state-of-the-art detector models, it still requires a good computing power to run efficiently. Therefore, it cannot be incorporated in devices with low computing power like smartphones.

Huang et.al.**[17]** developed YOLO-LITE, a real-time object detection model designed to operate on portable devices without a graphics processing unit (GPU. It was created to produce a smaller, quicker, and more efficient model based on the original object detection algorithm YOLOv2, expanding the accessibility of real-time object identification to a range of devices. YOLO-LITE demonstrates the enormous potential of shallow networks for lightweight real-time object identification networks. For a small machine, 21 frames per second on a non-GPU PC is really encouraging. But in comparison to the original YOLO design, lightweight architectures show a considerable loss in accuracy. In lightweight models, there is always a compromise between speed and accuracy, but in bigger models, there is always a tradeoff between speed and precision. Although the YOLO-LITE has a high mAP (mean Average Precision) in comparison to the state-of-the-art models, the model's accuracy hinders it from being used in real-world applications.

With rising in the new age of object detection models, many competing models were also introduced like Single Shot Detector (SSD) and Faster R-CNN. In W Liu **[18]**, Single Shot Detector (SSD), a multi-category, quick single-shot object detector is introduced. The usage of multi-scale convolutional bounding box outputs coupled to numerous feature maps at the top of the network is a crucial aspect of this model. SSD model can efficiently model the space of possible box shapes using this approach. SSD model when used with the right training procedures, a larger number of properly chosen default bounding boxes improves performance. SSD model was further improved by different architectures. One such efficient architecture was ResNet which was integrated with SSD by Lu et.al. **[19]**, which gave rise to increase the learning impact of features and improve accuracy. The most noticeable difference is that ResNet101 replaces SSD's VGG16 network architecture. The problem of network degradation is overcome by using the ResNet network's properties, which make it possible to establish a stacking layer network. SSD- ResNet is significantly superior to the original VGG16 network design and Yolov3 in terms of accuracy. When extracting scenes with little objects or more complicated surroundings, the evident difference is seen in the fact that the ResNet network can extract more representative characteristics. But the number of calculations will increase in the proposed model which can lead to overhead and makes it less suitable for real-time object detection.

In 2020, an improved lighter, and more accurate YOLOv4 was released which became a definitive object detection model and was significantly ahead of SSD models in terms of accuracy and speed. In A Bochkovskiy**[20]**, YOLOv4 is compared with other state-of-the-art detectors in different scenarios and at different GPU settings. ImageNet and MS COCO datasets were used to assess the mean average precision (mAP) of these detectors. Yolov4 architecture consists of three main elements which are the backbone (based on CSPDarknet53), Neck (based on SPP and PAN), and Yolo heads for detection and consecutively drawing the bounding box. Yolov4 is currently the best performing detector of the YOLO family in terms of both speed and accuracy. It outclassed and outperformed other detectors like Single Shot Detector (SSD), YOLOv3, RetinaNet, R-CNN, etc. In J Kim **[21]**, Faster-RCNN, YOLO, and SSD were being compared to checks which can be processed in real-time and have a high level of accuracy. A standard automobile data set was used to train YOLO, SSD, and Faster-RCNN. YOLOv4 model, SSD Mobilenet v1, and Faster-RCNN, the Inception v2 model are employed to compare the performance of each model. It was found that the best-performing model was YOLOv4. The Faster-RCNN model is the quickest among RCNN models, however, it does not have a suitable FPS since it utilizes CNN and the SSD is fast, but the model is light and employs mobilev1, resulting in bad accuracy. Some other work related to OCR/ automated analytics can be found in [22, 24 and 26].

* 1. **Limitations of Existing Approaches**

Early approaches to ANPR systems have their sets of advantages and disadvantages. The varying light conditions have drastic effect on the OCR/ image processing approach based ANPR systems[10,24]. The localization methods suffer with the misaligned images especially due to dynamic cameras [11,12,]. It’s already a tough task to extract number plates in the low-resolution images but this task becomes even tougher when the light conditions vary like at night[13,14,15]. Some models are able to perform really well and are able to extract images under varied conditions but have high computing costs and high processing time making them impractical for real-time usage [17,18].

Most of the highly accurate models are quite expensive [25] and moreover, some of them don’t work for Indian vehicles due to limitations of Data. *In this research*, we attempt to provide the ANPR technology with high precision for Indian vehicle number plates.

* 1. **Objectives of the proposed work:**

The objective of the proposed system is to accurately detect the number plates of high-speed vehicles (vehicles with speed<=80km/hr) as well as vehicles in high (vehicle) density regions.The proposed ANPR system is based on deep neural networks with images having varied light conditions to ensure accurate detection in most light conditions. The proposed system employs the YOLOV4 model, which is a single-stage detection model with high accuracy and less processing time. This makes it practical for real-time usage. YOLOV4 employed in the model is the super-fast and accurate deep neural network-based model which is able to process and detect number plates in a few seconds. The proposed ANPR system aligns the image properly before detection and extraction of license plates from a source of image or video frames. It's partially dependent on camera quality but the proposed system is also being trained on multiple low-resolution images as well to make the model accustomed to low-resolution images. The mAP(mean Average Precision) of the resultant license plate detection system should be at least greater than 90 percent under different lighting conditions.

The following section discusses the architecture of the proposed system.

1. **Proposed System Architecture**

The architecture of the proposed system is presented in Fig-1. It mainly comprises of three parts: Backbone, Neck and Detector.

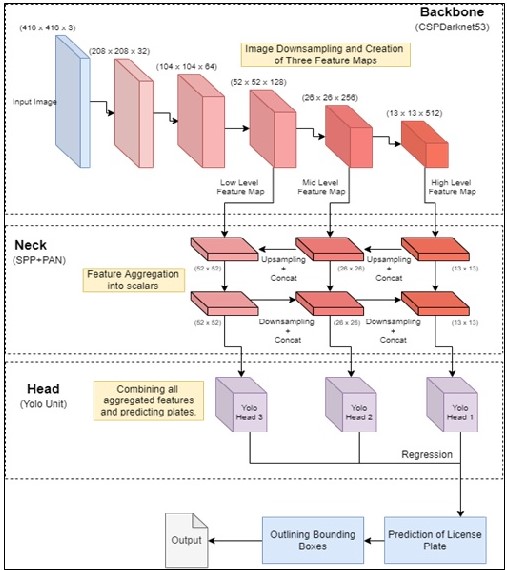


Fig.1 - Proposed System Architecture

* **Backbone:** An object detector's backbone network in fig 1 is usually pre-trained on ImageNet categorization. Here, yolov4 is using CSPDarknet53 as the backbone of our model. CSP stands for Cross Stage Partial DenseNet. This DenseNet is designed to connect convolution layers of our yolov4 model. It encourages our network to reuse features, and reduce the number of network parameters which in turn reduces the computational overhead and increases feature propagation throughout our model. Backbone generates 3 feature maps focusing on high, low, and mid features of the input image.
* **Neck**: This unit of our license detection model (yolov4) combines features extracted from the Cov-net Backbone. Yolov4 uses SPP (Spatial Pyramid Pooling) block after CSPDarknet53 backbone for the purpose of feature aggregation (i.e., convert the feature map response from the backbone into scalar values for faster processing). Whatever the size of our feature maps, the Spatial Pyramid Pooling Layer will allow us to generate fixed-size features. This helps in increasing the accuracy of our deep learning model. Yolov4 relies on PANet or PAN(Path Aggregation Network) to enhance the segmentation process by saving spatial information. A modified version of PAN is utilized in the YoloV4 implementation, where the new vector is formed by concatenating the input with the vector from a previous layer.
* **Head (Detector):** This unit is responsible for dense prediction. It is made up of a vector comprising the center, height, and breadth of the predicted bounding box, as well as the prediction's confidence score and label. Three YOLO layers detect the license plate based on the information from processed features from the neck.

**3.1. Modular Description**

The Module representation of Intelligent ANPR system can be depicted as:

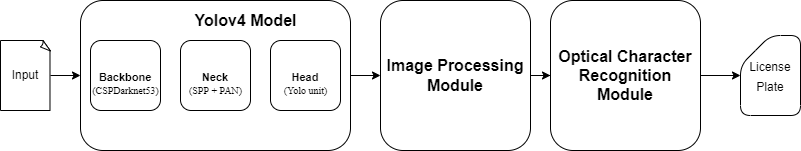
****

Fig.2 - Module representation of Intelligent ANPR system

The whole system is mainly broken into three modules as follows:

* + 1. **Yolov4 Detection Model**

This module in Fig 2 employs a state-of-the-art object detection deep learning-based model named “Yolov4” which was released in 2020 by Alexey Bochkovsky. It is the fourth and latest installment to Yolo.Yolov4 is a significant upgrade over Yolov3. Furthermore, training this neural network on a single GPU has become more straightforward. It is a real-time object identification system that detects multiple license plates in a single image. Yolov4 is a single-stage detection model which draws bounding boxes and identifies class labels for an image. Yolov4 uses a network of CNNs along with a detector Yolov3 heads to detect the license plate in the image. After extracting the detected regions of the License plate in the image, these regions are cropped and sent to the next module for processing.

* + 1. **Image Processing**

This module is responsible for processing cropped images of number plates obtained from the previous module. It makes the detected regions ready for character segmentation and optical character recognition in the next module. There mainly four techniques in the image processing module which are as follows:

* *Binarization:* It is a process of converting a pixelated image (multi-tone) into a binary image (two-tone). It contours and sharpens the outlines of distinct objects in an image. This helps in feature extraction and easy character segmentation in an image. Fig 3 shows the image after the binarization process.

|  |
| --- |
|  |
| Fig 3. Binarized Image |

* *Skewing*: It is a process of aligning and straightening a misaligned image. A misaligned image may affect the process of feature extraction. Fig 4 shows an image that has been skewed to align with the axes. For example, skewed image looks alike as depicted in figure 4:

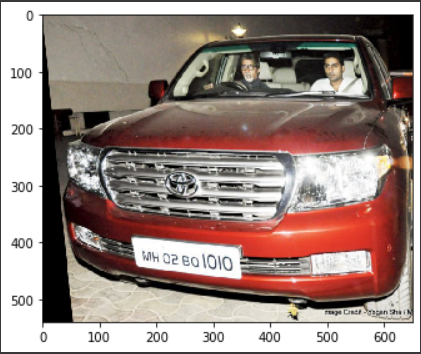


Fig 4. Skewed Image

* *Noise Removal:* It is a process of smoothening pixels in an image. Noise can lead to the loss of information from the image. Fig 5 illustrates the contrast between a noised image and a denoised image after image formatting.

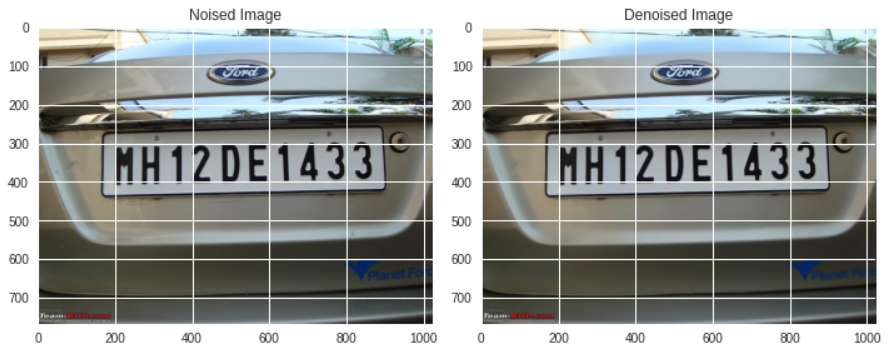


Fig 5. Denoised Image

* *Sharpening and Structuring:* Sharpening and structuring are techniques used to sharpen the images and to provide a definite structure to the components in an image to provide a contrast between different objects in an image. Fig 6. displays the difference between the input images and sharpened images using OpenCV libraries.



Fig 6. Sharpened Image

* 1. **Optical Character Recognition (OCR) and Character Segmentation**

This module is responsible for the extraction of License Numbers from the processed images of the number plates. Character Segmentation’s main motive is to segment different single characters in the image so that Optical Character Recognition (OCR) can occur more efficiently. If characters in the image have the same width, then this process becomes easier. OCR is used to extract the segmented characters in the image as text format and displays the extracted text to the user. The OCR engine used is EasyOCR. Fig 7 below shows the application of our OCR engine.

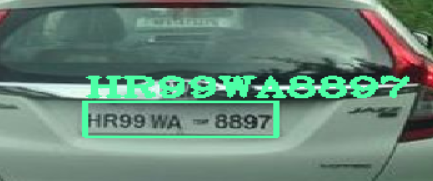


Fig.7- EasyOCR applied Image

1. **Dataset**

In this section, we discuss the data set used for simulation of the proposed ANPR system.

**4.1. Dataset Description**

The dataset is the collection of data required to train the model and generate accurate predictions. The quality, as well as quantity of the data, is necessary to build accurate object detection models. The Object model used in our proposed system is from the YOLO family. Therefore, the dataset should be in the YOLO format. The dataset used in training the proposed YOLOV4 model is a combination of manually labeled images of Indian Car plates from google and the Vietnamese License Plate dataset. The dataset contains 8000+ images of the number plates. The dataset is divided into two sets with 85% for training the model and the other 15% for testing the model. The reason why Vietnamese license plates were used was that they had alphanumeric number plates with many similarities with the Indian Number Plates system like region codes which are state codes in Indian License Plates. Apart from this, Vietnam is still a developing country with high traffic density just like India. Therefore, we used Vietnamese License Plates to increase the size of the dataset as there was no proper freely available dataset for Indian Vehicles on the Internet.

* 1. **Image Labelling**

The image labelling tool's primary aim is to allow users to highlight or capture a specific object in a photograph. The photos have been highlighted in order for the robots to read them. The accuracy of labelling is directly proportional to the performance of the model. If the dataset is poorly labelled then it may lead to incorrect predictions by the trained models. YOLO labelling works as follows: Each .txt file provides the object class, object coordinates, height, and width information for the matching picture file. A new line is formed in the .txt file for each item in the Image. This tool is used to label the images for the capstone paper is an online free labelling software called [*makesense.ai*](https://www.makesense.ai/). The platform is an open-source platform that is free to use under the GPLv3 License. This tool doesn’t require installation, which enhances its usability. It also makes no difference whose operating system we are using and makes this tool to be fully cross-platform. It also supports different formats like YOLO, VOC, JSON etc. It's ideal for modest computer vision deep learning papers, as it simplifies and speeds up the dataset preparation procedure. Labels that have been prepared can be downloaded in a variety of formats. The application is built on the React/Redux pair and was developed in TypeScript. Fig. 8 shows software’s Labeling Interface of vehicles.



Fig.8 - Make Sense Software’s Labeling Interface

* 1. **Image Formatting**

The Image Format specifies how the image's data will be processed during training. Data can be compressed, uncompressed, or vectorized. Each image format has its own set of benefits and drawbacks. The image format for the dataset is heavily dependent on the type of objection detection model used. The compatibility of format with the model plays a key role in the model’s performance. The YOLO format is compatible with the proposed YOLOV4 model. For each picture file in the same directory, a text file in Fig 9 with the same name is produced in the YOLO labeling format.

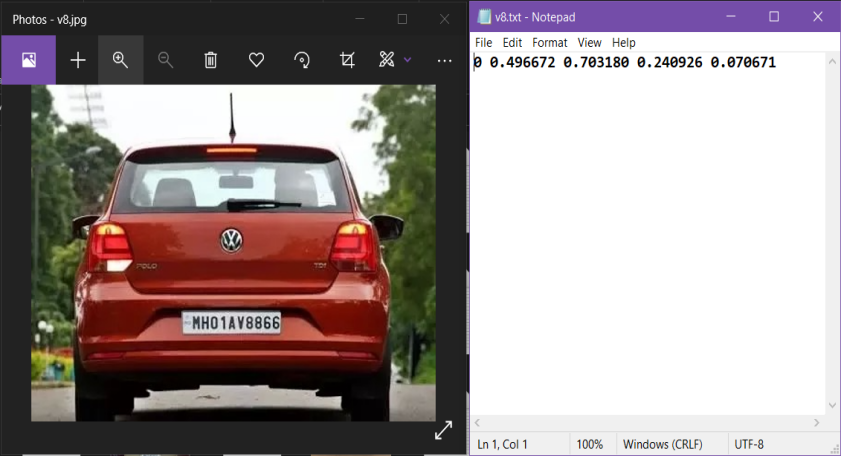


Fig.9 - Image with a YOLO labeled format text file

* 1. **Procedure followed by ANPR system**

**Step-1**:

1. Before feeding to YOLOv4 ANPR mode, convert the image into BLOB format of 416 x 416.
2. If the input is video then it gets converted into multiple frames of images and choose these frames to detect number plates.

**Step-2:**

The Backbone of the YOLOv4 model will extract features from the image and creates three feature maps with different level of features.

**Step-3:**

1. Forward the feature maps to the neck of the model.
2. The feature map is used to classify and localize the number plate in the input image. The features are aggregated from the feature map in the neck.

**Step-4:**

The aggregated features are then sent to YOLOv3 heads where regression is used to predict the region of License Plates.

**Step-5:**

The detected number plate portions from the image is transferred to an image processing module where the image is aligned, binarized, structured, and sharpened so that each character from the number plate is easily segmented.

**Step-6:**

The processed number plate portions are passed through an EasyOCR engine which extracts the characters from the number plate and returns them in text.

The following section discusses the performance of the proposed ANPR system on the described dataset.

1. **Results and Discussion**

This section evaluates the performance of the proposed model and analyses its operations under varying conditions.

**5.1. Image labelling using makesense.ai tool**

Image Labelling in YOLO format is done using an online labelling tool named [makesense.ai](https://www.makesense.ai/). The tool used is a web tool which provides a interactive UI and multiple format options to label the images. Dataset images were uploaded on this tool and then were labelled manually into YOLO format. Fig 10 illustrates the interface of the makesense.ai labelling tool.

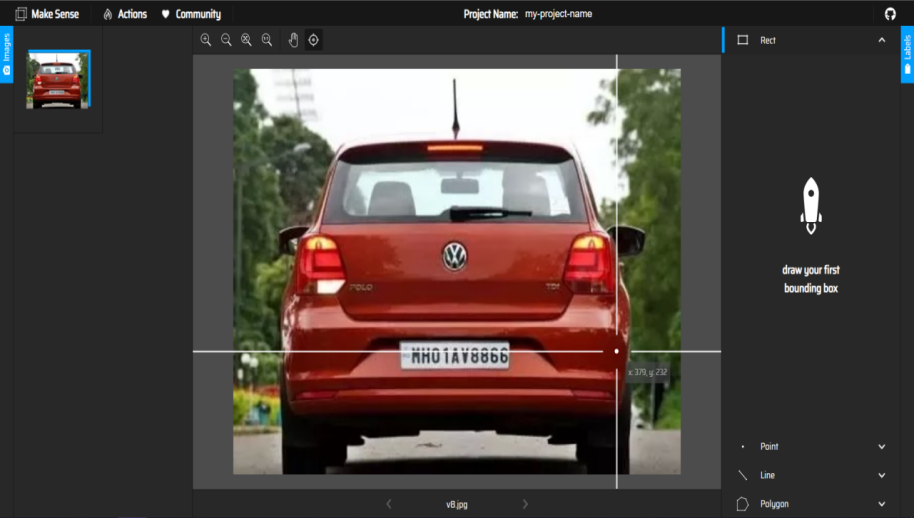


Fig.10 - Interface of MakeSense Image Labelling Tool

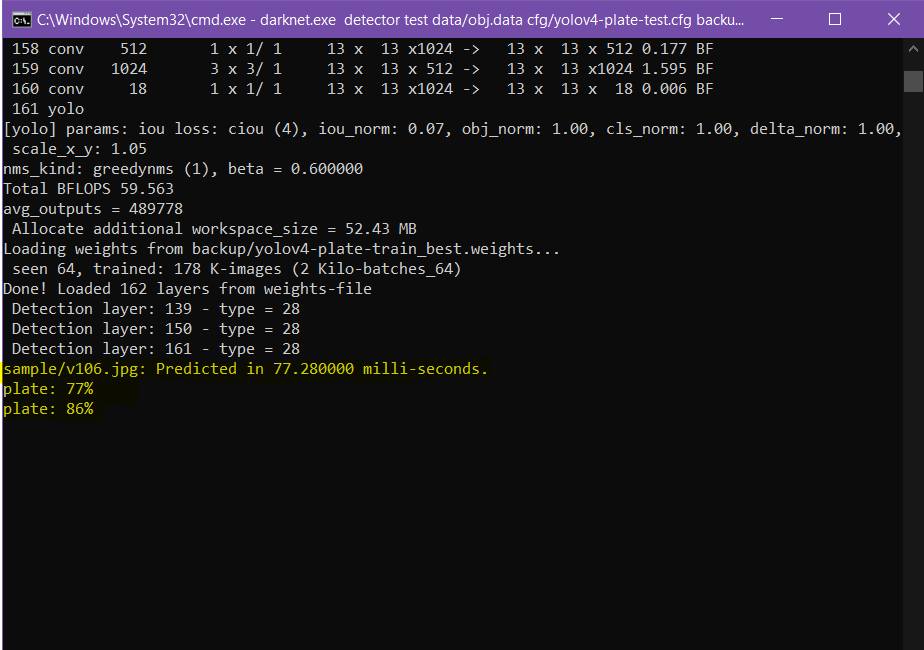


Fig. 11: Simulation results



Fig.12 - Number Plates Detection in Darknet Framework

The model in darknet framework is working in real-time frames. In Fig. 11, an image of two cars stuck in the traffic which is fed to proposed model in darknet framework resulting in detection of license plates of both the cars with a decent accuracy. The results obtained is visualized as shown in Fig.12.

Note: Since Darknet Framework has limited functionality and to add more functionality. We use dnn library from OpenCV to apply processing and OCR engine in Jupyter Notebook.



Fig.13 - Original Input Image

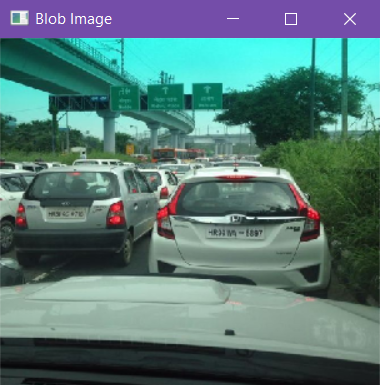


Fig.14- Blob Format of Input Image

The original image (shown in Fig.13) is converted into 416 x 416 blob image (shown in Fig.14) to make it compatible for processing by the ANPR model. The model is adjusted for minimum probability and threshold in order to discard weak predictions for the YOLOv4 detection model. The opencv library is used to visually represent the results over the image with detected bounding boxes for the license plates in the image (shown in Fig. 15). The detected regions (Bounding Boxes) from the image is sent to EasyOCR modules for the extraction of the License Plate number from the detected regions. The extracted number is visualized as shown in Fig. 16.



Fig.15 Bounding Boxes around Detected Number Plate



Fig.16 Number Plates after OCR application

The graph for plotting precision and recall for multiple threshold values (shown in Fig. 17) denotes the stability of the proposed model. The graph indicates that the proposed ANPR model performs consistently through out multiple threshold values. The area under P-R curve represents the accuracy of the model, and as shown in Fig. 18, the accuracy is quite high.

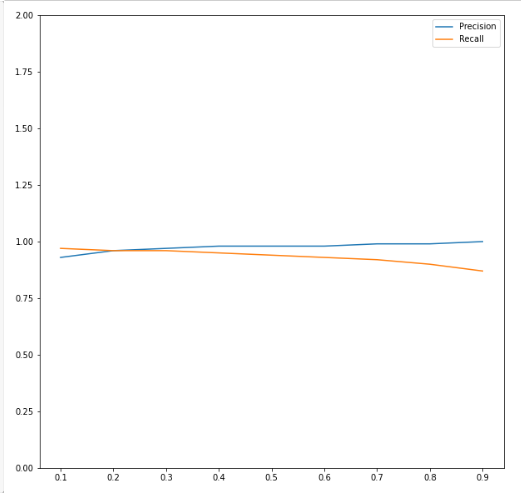


Fig. 17. Precision and Recall for multiple threshold values:

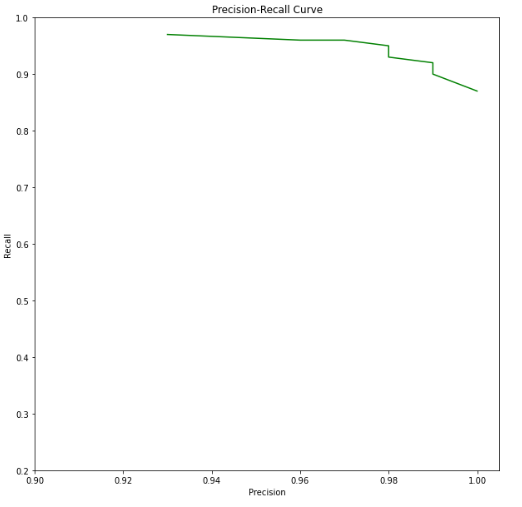


Fig. 18 Precision-Recall curve (P-R curve)

* 1. **Testing/Performance Metrics**

The object detection models are popularly evaluated using mean average precision (mAP). The mAP is dependent on other metrics such as AP, IoU, Precision, Recall, and F1score. These metrics are discussed as follows:

* **Average Precision** (**AP):** The average precision (AP) is a method of condensing the precision-recall curve into a single number that represents the average of all precisions. The following equation is used to determine the AP. To put it another way, the AP is the weighted sum of precisions at each threshold, with the weight representing the increase in recall.
* **Intersection over Union (IoU):** The IoU aids in determining whether or not a detected area contains a required item. The IoU is determined by dividing the area of intersection of the two boxes by the area of their union. Higher IoU is proportional to better precision. Object detection models are often assessed using different IoU thresholds, each of which may provide different predictions than the others.
* **Precision:** This metric is used to measure the number of positive class predictions that really belong to the positive class.
* **Recall:** This metric is used to measure the number of positive class predictions made out of all positive examples in the dataset.
* **F1-score:** This metric is used to measure the balance between Precision and Recall. It is obtained after taking the harmonic mean of Precision and Recall.
* **Mean Average Precision (mAP):** This metric is basically the average of all Average Precisions (APs) calculated over all the classes of objects.

Note: Since for ANPR system, there is only one class i.e license plate. Therefore, Average Precision (AP) is equal to the mean Average Precision (mAP).

The mean Average Precision (mAP) for the proposed intelligent ANPR system is **98.33**%. The metrics discussed above are compiled in the following table for three threshold values.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Threshold** | **IoU** | **Precision** | **Recall** | **F1 score** |
| **0.75** | 80.55 % | 0.99 | 0.91 | 0.95 |
| **0.5** | 79.30 % | 0.98 | 0.94 | 0.96 |
| **0.25** | 78.26 % | 0.97 | 0.96 | 0.97 |

Table 2. Performance metrics at different thresholds

**9.1 Graph / Visualization**

Visualization is an effective way to check the stability of the ANPR system. The two important graphs that are plotted for the model are as follows:

* **Precision and Recall with respect to the threshold:** The precision and recall for each threshold from the range [0.1,0.9] with a difference of 0.1 between each consecutive threshold values. Fig 19 shows a plot of precision and recall values at multiple thresholds.

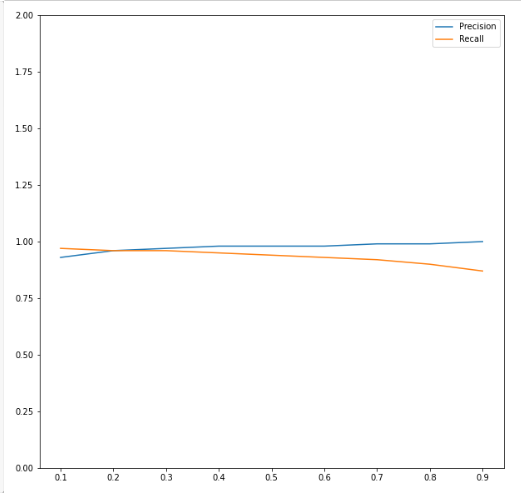


Fig.19 - Precision and Recall Visualization

* **P-R curve:** Precision-Recall Curve is an important curve whose area corresponds to the mean average precision of the trained model. The more area, the curve covers more will be its precision and performance. Fig 20 shows the P-R curve for our proposed model.

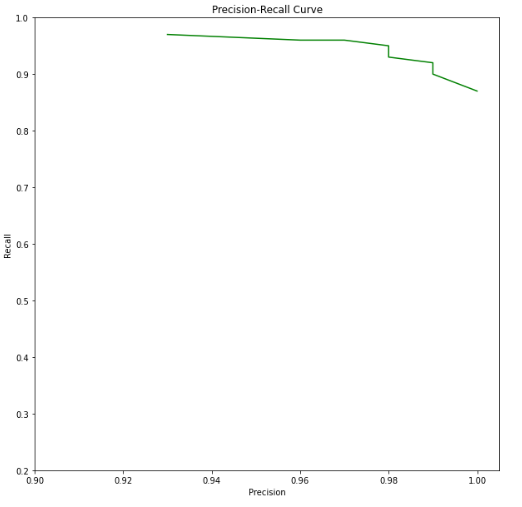


Fig.20 - P-R curve

**5.3. YOLOv3 vs YOLOv4**

Yolov3 and YOLOv4 both belong to the YOLO family. YOLOv3 is basically the predecessor to YOLOv4. YOLOv3 is based on Darknet53 as the backbone while the proposed YOLOv4 relies on CSPDarknet53 as the backbone. CSPDarknet is superior as it improves the feature extraction process and creates better feature maps. After training both the models in the darknet framework with the same dataset and in the same environment, the results obtained clearly state that there is a drastic difference between YOLOv4 and YOLOv3. The mean average precision is 98.33% for YOLOv4 model as compared to 46.43% of YOLOv3. YOLOv4 is ahead by a huge margin as shown in Table 3. This makes YOLOv4 more accurate to be used in real-time when compared with YOLOv3.

|  |  |  |
| --- | --- | --- |
|  | **YOLOv3 @ 0.5** | **YOLOv4 @ 0.5** |
| **Precision** | 0.60 | 0.98 |
| **Recall** | 0.54 | 0.94 |
| **F1-score** | 0.57 | 0.96 |
| **mAP (Mean Average Precision)** | 46.43% | 98.33% |

Table 3. YOLOv3 vs YOLOv4 at 0.5 threshold

Hence, figure 3 to figure 16 (excluding Figure 11) shows simulation set up and working of ANPR systems and Figure 11, Figure 17 to Figure 20 shows the experimental result of our proposed systems.

1. **Conclusions and Future Work**

In this research, we developed a novel automatic number plate recognition (ANPR) system that uses CSPDarknet53 based YOLOv4 and EasyOCR to address the huge variability of number plates across Indian states and under various situations. The modified network architecture/ Intelligent ANPR system improves detection accuracy and speed by including the DenseNet in the backbone to optimize feature transfer and reuse; two new residual blocks in the backbone and neck improve feature extraction and reduce computing costs; the Spatial Pyramid Pooling (SPP) improves receptive field, and a modified Path Aggregation Network (PANet) preserves fine-grain localized information and improves feature fusion. The localized information will be responsible for extracting the number from the image and convert into a string using an OCR engine named “EasyOCR”. This paper fulfills the objectives to detect number plates for fast-moving vehicles and vehicles in high vehicle density regions with decent accuracy. The YOLOv4 has a 98.33% mAP for detecting the number plates when tested on 1260 test images. However, the performance of the ANPR system is heavily dependent on the quality of images captured by the Camera. If the images are of low resolution will be able to detect the plate but the information may not be extracted properly by the EasyOCR engine. The proposed ANPR system performs better than the ANPR system based on YOLOv3 by a significant margin which makes it suitable for real-time implementation. The ANPR system is kept free to use as only open-source software and technologies were being to build it.

The major benefit of the proposed model is that it provides a free and robust ANPR system that can be easily installed. It can be concluded that the proposed low-complexity ANPR system performs robustly for Indian number plates that provide good accuracy under less duration for unconstrained plates in real-time. Future scope of work may include the model can further be experimented with by using different backbones like EfficientNet and can be trained on complete Indian Vehicles Dataset for more accurate results. Even Different OCR technologies can be used in order to further improve the performance.

**REFERENCES**

1. Arora, A., Saluja, D., Parmar, N., Goyal, D. and Senthil, V., 2021. Automatic number plate detection and un-manned challan generation for the odd/even rules in Delhi. Materials Today: Proceedings, 43, pp.231-236.
2. Mirzahossein, H., Gholampour, I., Sedghi, M. and Zhu, L., 2021. How realistic is static traffic assignment? Analyzing automatic number-plate recognition data and image processing of real-time traffic maps for investigation. Transportation Research Interdisciplinary Perspectives, 9, p.100320.
3. Haines, A., 2009. The role of automatic number plate recognition surveillance within policing and public reassurance (Doctoral dissertation, University of Huddersfield).
4. Merola, L.M., Lum, C. and Murphy, R.P., 2019. The impact of license plate recognition technology (LPR) on trust in law enforcement: A survey-experiment. Journal of experimental criminology, 15(1), pp.55-66.
5. Rashid, M.M., Musa, A., Rahman, M.A., Farahana, N. and Farhana, A., 2012. Automatic parking management system and parking fee collection based on number plate recognition. International Journal of Machine Learning and Computing, 2(2), p.94.
6. Ta, T.D., Le, D.A., Le, M.T., Tran, T.V., Do, T.T., Nguyen, V.D., Trinh, C.V. and Jeon, B., 2015, January. Automatic number plate recognition on electronic toll collection systems for Vietnamese conditions. In Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication (pp. 1-5).
7. Robinson, A. and Venter, C., 2019. Validating traffic models using large-scale Automatic Number Plate Recognition (ANPR) data. Journal of the South African Institution of Civil Engineering, 61(3), pp.45-57.
8. Hadavi, S., Rai, H.B., Verlinde, S., Huang, H., Macharis, C. and Guns, T., 2020. Analyzing passenger and freight vehicle movements from automatic-Number plate recognition camera data. European Transport Research Review, 12(1), pp.1-17.
9. Gunay, B., 2012. Using automatic number plate recognition technology to observe drivers' headway preferences. Journal of advanced transportation, 46(4), pp.305-317
10. Suri, P.K., 2010. Vehicle number plate detection using sobel edge detection technique.
11. Ghosh, R., Thakre, S. and Kumar, P., 2018, October. A vehicle number plate recognition system using region-of-interest based filtering method. In 2018 Conference on Information and Communication Technology (CICT) (pp. 1-6). IEEE.
12. Chouhan, P.S. and Govindan, V.K., 2014. Localization of license plate using characteristics of alphanumeric characters. International Journal of Computer Science and Information Technologies, 5(3), pp.3407-3409.
13. Kasaei, S.H.M. and Kasaei, S.M.M., 2011, September. Extraction and recognition of the vehicle license plate for passing under outside environment. In 2011 European Intelligence and Security Informatics Conference (pp. 234-237). IEEE.
14. Asthana, S., Sharma, N. and Singh, R., 2011. Vehicle number plate recognition using multiple layer back propagation neural networks. International Journal of Computer Technology and Electronics Engineering (IJCTEE), 1(1).
15. Babbar, S., Kesarwani, S., Dewan, N., Shangle, K. and Patel, S., 2018, August. A new approach for vehicle number plate detection. In 2018 Eleventh International Conference on Contemporary Computing (IC3) (pp. 1-6). IEEE.
16. Redmon, J., Divvala, S., Girshick, R. and Farhadi, A., 2016. You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).
17. Huang, R., Pedoeem, J. and Chen, C., 2018, December. YOLO-LITE: a real-time object detection algorithm optimized for non-GPU computers. In 2018 IEEE International Conference on Big Data (Big Data) (pp. 2503-2510). IEEE.
18. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y. and Berg, A.C., 2016, October. Ssd: Single shot multibox detector. In European conference on computer vision (pp. 21-37). Springer, Cham.
19. Lu, X., Kang, X., Nishide, S. and Ren, F., 2019, December. Object detection based on SSD-ResNet. In 2019 IEEE 6th International Conference on Cloud Computing and Intelligence Systems (CCIS) (pp. 89-92). IEEE.
20. Bochkovskiy, A., Wang, C.Y. and Liao, H.Y.M., 2020. Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934.
21. Kim, J.A., Sung, J.Y. and Park, S.H., 2020, November. Comparison of Faster-RCNN, YOLO, and SSD for real-time vehicle type recognition. In 2020 IEEE international conference on consumer electronics-Asia (ICCE-Asia) (pp. 1-4). IEEE.
22. Merola, L.M., Lum, C. and Murphy, R.P., 2019. The impact of license plate recognition technology (LPR) on trust in law enforcement: A survey-experiment. Journal of experimental criminology, 15(1), pp.55-66.
23. Salau, A.O., Yesufu, T.K. and Ogundare, B.S., 2021. Vehicle plate number localization using a modified GrabCut algorithm. Journal of King Saud University-Computer and Information Sciences, 33(4), pp.399-407.
24. Mokayed, H., Shivakumara, P., Woon, H.H., Kankanhalli, M., Lu, T. and Pal, U., 2021. A new DCT-PCM method for license plate number detection in drone images. Pattern Recognition Letters, 148, pp.45-53.
25. Tang, J., Wan, L., Schooling, J., Zhao, P., Chen, J. and Wei, S., 2022. Automatic number plate recognition (ANPR) in smart cities: A systematic review on technological advancements and application cases. Cities, 129, p.103833.
26. Li, J., Van Zuylen, H., Deng, Y. and Zhou, Y., 2020. Urban travel time data cleaning and analysis for Automatic Number Plate Recognition. Transportation Research Procedia, 47, pp.712-719.