AUTOMATED VEHICLE IDENTIFICATION BY LICENSE PLATE RECOGNITION

KUMARMANGAL ROY

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

UNIVERSITY OF MALAYA
KUALA LUMPUR

2020-21

ABSTRACT

Movement of vehicles in and out of predefined enclosure is an important security protocol that we are encountering on daily basis. Identification of vehicles is very important factor for the security surveillance. Manual identification and managing database of such vehicles is tedious. This kind of vehicle management for many vehicles is not only inconvenient but also time-consuming and requires physical intervention of security personnel, which under the present situation of Covid-19 impact is not that suitable. Not only that, for multiple entry-exit points, it will be more error prone. Rather a contactless option for smart vehicle management system is suitable. The best way to streamline such process is through automatic identification of unique license number plates. In this research report we propose a license number plate recognition approach using Haar Cascade Object detectors. Not only that we also compare performance of Haar Cascade for number plate detection with that of MobileNet-SSD (light deep neural network architecture integrated as the base network with single shot object detector architecture). Once the license plate is detected we use OCR for character identification. Most of the previous works in license recognition system have limitations like light exposure, having stationary backgrounds, indoor area, specific driveways etc etc. Our current approach is robust and works good on live object detection. The main objective of this project is to create an intelligent pipeline for automatic vehicle license number plate detection system that will provide smart authentication based on legitimacy of vehicles by showcasing the performance on real-time data for Malaysian number plates.

Keywords: ANPR, Haar-Cascade, OCR, Computer Vision, MobileNet-SSD

ABSTRACT

Pergerakan kenderaan masuk dan keluar dari kandang yang telah ditetapkan adalah protokol keselamatan penting yang kami hadapi setiap hari. Pengenalpastian kenderaan adalah faktor yang sangat penting untuk pengawasan keselamatan. Pengenalpastian manual dan pengurusan pangkalan data kenderaan sedemikian adalah membosankan. Pengurusan kenderaan seperti ini untuk kebanyakan kenderaan bukan sahaja menyusahkan tetapi juga memakan masa dan memerlukan campur tangan fizikal anggota keselamatan, yang dalam situasi semasa impak Covid-19 tidak begitu sesuai. Bukan itu sahaja, untuk berbilang titik masuk-keluar, ia akan lebih terdedah kepada ralat. Sebaliknya pilihan tanpa sentuh untuk sistem pengurusan kenderaan pintar adalah sesuai. Cara terbaik untuk menyelaraskan proses tersebut adalah melalui pengenalan automatik plat nombor lesen yang unik. Dalam laporan penyelidikan ini kami mencadangkan pendekatan pengecaman plat nombor lesen menggunakan pengesan Objek Haar Cascade. Bukan itu sahaja, kami juga membandingkan prestasi Haar Cascade untuk pengesanan plat nombor dengan MobileNet-SSD (seni bina rangkaian neural dalam cahaya disepadukan sebagai rangkaian asas dengan seni bina pengesan objek tembakan tunggal). Setelah plat lesen dikesan, kami menggunakan OCR untuk pengenalan aksara. Kebanyakan kerja terdahulu dalam sistem pengecaman lesen mempunyai had seperti pendedahan cahaya, mempunyai latar belakang pegun, kawasan dalaman, jalan masuk tertentu dsb. Pendekatan semasa kami adalah teguh dan berfungsi dengan baik pada pengesanan objek langsung. Objektif utama projek ini adalah untuk mewujudkan saluran paip pintar untuk sistem pengesanan nombor plat kenderaan automatik yang akan menyediakan pengesahan pintar berdasarkan kesahihan kenderaan dengan mempamerkan prestasi pada data masa nyata untuk nombor plat Malaysia.

Kata kunci: ANPR, Haar-Cascade, OCR, Computer Vision, MobileNet-SSD

ACKNOWLEDGEMENTS

Along this journey I came across many people who helped me to reach my goal. I would like to thank my supervisor Dr. Muneer Ahmad for the guidance and support. Time and again he kept me motivating. He is the person who had faith and faith in me. He has been constantly motivating me by sharing his thoughts and guidance. He is a very intuitive person. He knows very well how to explore the potential of students. Time and again he has shed light on relevant points. While some persons helped me with ideas and insight some spent their time to provide data for my work. In special, I would like to thank Mr. Chinmoy Dey for having some progressive discussion with me on relevant topic, as well as providing me with personal support. I would also like to thank Mr. Anant Chaturvedi, a good friend, ex-colleague, who helped me to collect data during initial phase of this work.

I want to acknowledge the financial support from UM Living Lab Grant Programme - Sustainability Science (LL055-2021) committee.

I would also like to show my gratitude and thank my beloved family members. Whatever I am today is just because of them. My parents Shivaji Roy and Seema Roy for the love and all effort they made to educate and raise me. I would like to thank my sisters Soomondrita Roy, Sananda Nandy and brother, Tarak Nandy for the constant love and faith they have been showing to me.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

For examples:

ANPR	:	Automatic Number Plate Recognition
OCR	:	Optical Character Recognition
CV2	:	Computer Vision
SSD	:	Single Shot Detection
VIN	:	Vehicle Identification Number
CNN	:	Convolution Neural Network
AI	:	Artificial Intelligence
KNN	:	K-Nearest Neighbor
FPS	:	Frame Per Second
UM	:	University Malaya

LIST OF APPENDICES

CHAPTER 1: INTRODUCTION

1.1 Background

With ever growing vehicle population and growth in auto sectors, managing and monitoring of vehicles manually is tiresome and building solutions which are costly is also not acceptable (Darapaneni et al., 2020). Thus, moving towards intelligent transport system is essence of time. Smart vehicle management system & intelligent transportation systems require automated vehicle identification architectures. This is a powerful tool for traffic management, electronic authentication at toll gates, vehicle monitoring for law enforcement, commercial transportation, access control etc. etc. The principal working architecture for all these applications require uniquely identifying each vehicle. To be uniquely identify each vehicle there are few components like number plate, vehicle identification number (VIN) and owner's detail. Every vehicle should be equipped with a device that can emit all these necessary information and feed it at different checkpoint receivers. But identifying and reading owner's detail raises concern on data privacy. Not only that, it will not be practically feasible for every vehicle to mount such information emitting device. Thus, the best possible way to uniquely identify vehicles is through identifying vehicle number plates. Manual inspection of vehicles and managing of database of such vehicles is tedious. This kind of vehicle management for many vehicles is not only inconvenient but also time-consuming and requires physical intervention of security personnel, which under the present situation of Covid-19 impact is not that suitable. Rather a contactless option for smart vehicle management system is suitable (Mangal et al., 2020). License plate recognition systems are important part of intelligent transportation system, and used in traffic management, electronic toll collection, access control, security purpose, vehicle identification, managing parking etc (W. Wang et al., 2019).

There has been quite a good advance in the object detection algorithms and processes due to enhancements in field of artificial intelligence and deep learning as discussed by (Sharma et al., 2019). The most common issue that exist in automated number plate detection domain is detection of the candidate region. Determining the region of interest at a fast pace to keep up with traffic/vehicle movement is always a concern. Most classical detection systems basically work by repurposing classifiers to detect desired object classes. What it means that the model is applied at multiple locations across the image and at varied scales. Most high scored values are returned as detection. This results in slower processing. Multi oriented, multi-directional detection and recognition fails in many cases (Li et al., 2019). Low light and poor resolution also lower detection accuracy rate (Xie et al., 2018). Not only that most of the existing ANPR system suffers similarity issue between certain set characters during optical conversion (K & Raghunadh, 2016). These characters are often not recognized clearly due to its similarity in shape and edge. Example: O and 0, O and D, 8 and B ,5 and S... etc. For number plate recognition it is very important that all the characters are read correctly without any discrepancy. Not only character recognition, generating candidate region remains to be the most important aspect for ANPR systems. Much analysis is required on algorithms based on the comparative performance study. Further for most of the scenarios, all the classical models are applied and tested over mostly still images and less study is documented on video/camera feed.

For real life object detection, speed is of the utmost importance. In real life scenario, number plate identification should be done really fast as it is impractical to have a system which may hinder normal traffic flow. Slower processing will lead to more power consumption and added cost. As mentioned by (Yulianto et al., 2021)in his paper, his model for intelligent parking system takes about 13.25 seconds to process at the entry of the parking lot and takes 19.75 seconds to process at the exit gate. This is may be suitable at parking lot gates, but for monitoring normal traffic cars this much of wait will be too much. Even for surveillance cases, the need is to process plates as quickly as possible. Not only that it, slow processing will cause traffic disruption and will lead to huge que and high carbon emission rate. Any detection method deployed should be robust and fast enough to segregate images/frames fast. Low light and poor resolution may lead to unsuccessful number plate detection (Shariff et al., 2021). But in recent times there has been good advancement in this area due to improvement in the cameras, which are powerful enough grab high resolution frames. But low light and poor resolution will affect vehicle number plate detection which will lead to security issue as there can be undetected vehicles inside bounded campuses/areas. Similarity issue between characters also raises security concern. There can be possibility that we want to locate/identify a particular vehicle, but we will fail to detect due to incorrect alpha-numeric representation. It is also possible that we detect a license plate a vehicle with incorrect details. This all leads to incorrect data.

To overcome the issue of applying the model at multiple locations of the image and at multiple scales across the image, we try to take a simple, fast approach to get the region of interest. Primarily we use Haar-Like features which is computationally cheap. It uses its line features, edge features and rectangle features the area of interest as discussed by (Cheon & Lee, 2019). The features are processed through convolution network, Gentle AdaBoost (GAB) is applied for our study. We compare performance with deep learning-based algorithm. In that we look at the full image at one particular time. This will enable us to get its predictions as per global context of the frame. Single Shot Detection is used for real time object detection on top of MobileNet base network. MobileNet is an architecture which uses depth-wise separable convolutions to construct light CNN which works good for embedded vision applications (Howard et al., 2017). The SSD algorithm, instead of using sliding window like in case of convolution network, divides each frame in a grid and every individual grid cell is made responsible for detecting class of objects in that region of the frame (Peker, 2019). Image processing, contour finding, edge detection and masking technique applied candidate retrieval (Aggarwal et al., 2019). These steps are going to enable us to process images/frames of poor resolution under low light. The percentage of error due to character similarity can be reduced with help of regex matching where we are putting exact alpha-numeric patters which are valid.

Mostly, the automatic recognition of license plate is employed for security risk mitigation to ban non-registered vehicles from entering the safer zones. Machine learning and deep learning methods have been found quite useful in identification and recognition of objects in certain scenarios. Automatic identification and recognition of number plates of vehicles will make the process hassle free, save time and fuel, help in prediction of other side measures e.g., traffic management, count of incoming and outgoing vehicles at periodic cycles. Predefined area like in case of our campus, where manual inspection of vehicles and managing of database of such vehicles is tedious. The UM campus security officials manually manage and monitor the vehicles at checkpoints and entry gates. We also have multiple entry/exit points which makes it inconvenient for manual vehicle database management. This research proposes a smart vehicle management system based on automatic scanning and recognition of licensed plates of vehicles entering the campus. A security camera is to be mounted at the security checkpoint of the UM campus. This should enable us to develop a system which will automatically scan the vehicle license number plate and identify the vehicle accessing the campus. The proposed system will help the security officers register new legitimate vehicles, delete and update the data of existing vehicles, assign warnings to illegitimate or suspected vehicles and calculate the statistics related to access history of vehicles. These statistics will help the authorities estimate the carbon emission to reduce carbon pollution for the pursuit of a sustainable campus. The system can also help us to monitor traffic management inside the campus during rush hours. The performance of the architecture proposed will be evaluated with the performance of prevailing similar systems to quantify the significance of the proposed system.

In this study we propose a Haar-Cascade classifier-based model for number plate detection. We also study performance of SSD-MobileNet for real time object detection and number plate recognition. Once we fetch our intended classified objects, we start image processing. During image processing we are converting frames to grayscale. We also proceed with filtering followed by canny process of detecting edges in an image. With the edge points we try to detect the contours for all the possible edge combination. Once we have that, we use approximation of polygon structures to fetch the boundary points of number plate region. Then we carry masking technique to fetch the number plate region. Once we have our region of interest, we apply optical character recognition for image to text conversion. We are using Reg-Ex conditioning for matching alpha numeric number plates with available formats. This helps us to reduce character similarity issue between few look-a-like characters.

1.2 Objectives

Please find objectives for our study mentioned below.

a. To study the performance of Haar Cascade filters compared to that of SSD MobileNet for number plate detection
b. Use OCR module in python for conversion from image to text
c. Detect vehicles with unique license number plate to provide authentication

1.3 Data Source

Multiple data sources are being used for different testing scenarios. For testing object detection model, we are using still images from Kaggle, which is an open forum. There is good variation in the images. Frontal view and rear-view images are most frequent. Apart from that there is different orientation of car images also present. The set of images also contain number plates for two-wheelers. There are also few images which are not having any number plate displayed or has poor lighting condition which reduces the visibility of the number plates.

For live detection of number plates, data is being collected directly from the ground. For initial testing phase, video data was downloaded from open sources. Later few video files were collected from the main entry gate of the campus. For final testing phase of the pipeline, a webcam was set up at the UM gate, which feeds live frame of all the vehicles entering the campus. Specification of systems used is as follows.

a. Lenovo L340, processor-Intel(R) Core(TM) i5-9300H CPU @ 2.40GHz, 8GB RAM
b. C933 Pro-HD webcam was used for performance analysis at 720p 60 FPS.

CHAPTER 2: LITERATURE REVIEW

Automatic License plate recognition uses two fundamental parts, license plate extraction and optical character recognition on frames to fetch license plate numbers. It has wide range of applications and is used in traffic management, electronic toll collection, access control, security purpose, vehicle identification, managing parking etc. (Zhang et al., 2019). With the upgradation of systems and use of AI in our natural surroundings, automatic toll collection, automated parking systems and security checks, researchers have dived in to find easy and possible techniques to smoothen automotive traffic at the same time prioritizing security instances. Useful knowledge extraction can support decision making and cost reduction of intelligent transportation systems.

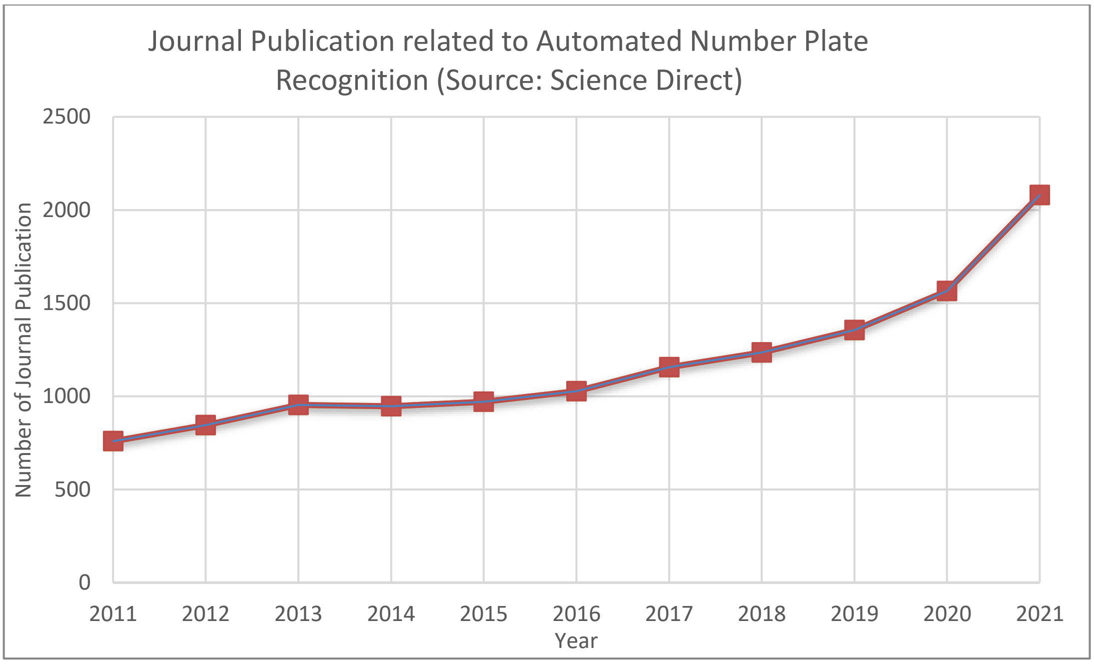


Figure 2.1: Journal Publication trend related to Automated Number Plate

ANPR systems are going to be widely applied across the world due to its usability. As shown in Figure 2.1 we can see the domain of license plate detection is trending and good amount of effort is put in research.

Mentioned by (Darapaneni et al., 2020), the number of vehicles in our daily world is increasing rapidly. With such increase, the complexity of traffic movement and management will be challenged. (Yulianto et al., 2021) mentions that from 2008 to 2018, the number of motorcycles increased by 20% in Indonesia itself. (Varkentin & Schukin, 2019) mentions that car population in Russia has increase to 0.4 per person, as of 2018. As of 2021, car population is 0.54 per person and interestingly the ratio was 0.44 per person as of 2019. With such growing number of cars, it is evident that a better system would be required to manage and monitor vehicles. (Sharma et al., 2019) specifies that the growing number of vehicles all over the world would require intelligent transport management system. (Mangal et al., 2020) mentions his growing concerns over such high traffic movement and checking for unauthorized vehicles. Intelligent traffic system will help us to curb wrong side driving, reckless driving, vehicle authorization (Goyal et al., 2021), (Srivastava et al., 2021). Author (Mandi et al., 2017) and (Babu et al., 2019) and (Montazzolli & Jung, 2017) also mentions the need to intelligent traffic management system in today's world.

With advancement of technology, both in hardware and software, it is now possible to make smart, automated, intelligent traffic systems. There are many parts to an intelligent traffic system. This includes ANPR system, Car-to-Car communication, GPS Systems, Smart Traffic lights etc etc. (Sasi et al., 2017) and (Wanwei Wang et al., 2019) both mentions the need of intelligent traffic management. The key part in any intelligent traffic system is license plate recognition. (Zhao et al., 2010) mentions importance of ANPR systems and how it will be an integral part of intelligent transport system. (Wanwei Wang et al., 2019) mentions in his paper that ANPR systems are not robust enough to function in real world scenarios. There are many factors which impacts the performance of ANPR systems. As mentioned by (Agarwal & Goswami, 2016), there are lot of challenges for reading number plates as it differs in shape and size. Moreover, there are also lighting issues which are dealt with powerful cameras. (Lib et al., 2018) mentions in his paper regarding the challenges of detecting number plates under natural scene images. (Silva &

Jung, 2018) mentions that thought there has been good advancement in the field, but it required to be studied more and deep.

Thanks to deep learning and artificial intelligence, there has been some serious growth. With good growth in computer vision, we can achieve a better intelligent transport system. (Zhang et al., 2020) and (Zhang et al., 2019) both in his papers clearly banks on the use of deep learning approaches such as SSD-mobileNet. Efficient models for mobile and embedded vision applications such as MobileNet is applicable for such intelligent traffic management (Howard et al., 2017). Deep learning, SSD-MobileNet had achieved high average precision of 99.76% for car class, for class person 97.76% & for class chair, 71.07% (Younis et al., 2020). Comparison of different object detection on car number plate, SSD-Mobile, Resnet, Faster Regional Convolution Neural Network(R-CNN), Region Based Fully convolution Network (RFCN) was studied by (Peker, 2019). ANPR with convolution networks on single pass is another method for ANPR (Li et al., 2019). Different approaches such YOLO, K-Means with Segmentation, CNN with graphic boards and Ant Colony Optimization were studied for car plate recognition, (Xie et al., 2018), (Stefanović et al., 2018), (Lee et al., 2017), (Sasi et al., 2017) respectively.

We also see the wide use of simple and fast algorithm like Haar Cascade classifier for number plate detection. Authors like (Yulianto et al., 2021), (Sharma et al., 2019), (Cheon & Lee, 2019) and (Zhao et al., 2010) has studied Haar Cascade classifier for number plate detection. Studies were based on performance of the classifier model. Thus, this paper focus on performance comparison of deep learning model as SSD-MobileNet and Haar-Cascade classifier along with OCR module for character recognition. Table 2.1 outlines a summary of selected 40 prior studies. For fair comparison, different technological adoption along with research outcomes and limitation of respective studies were noted.

Table 2-1: Summarization of selected prior studies

Study	Author	Weakness Identified	Proposed Solution	Technology Adoption	Research Outcomes	Limitation
1	(Darapaneni et al., 2020)	Object detection is one of the key aspects for number plate detection. While there are many algorithms which are at present being tried and used, deep learning models feature the best probabilistic outcome	A pipeline created with YOLO V3 algorithm, which to be applied for custom object detection and combined with OpenCV module find region of interest. On top of that PyTessearct OCR module used for character recognition	YOLO V3, OpenCV, PyTesseract	An 100% accuracy achieved for License Plate detection. For character recognition, accuracy of 90% achieved	Data was collected and trained specifically for the model which had only 300 images/videos. The model was tested on only 20 car images which may have lacked variation and thus such n high accuracy achieved.
2	(Mangal et al., 2020)	Covid-19 restrictions has taught the world many things and one such issue remains on how traffic movement, authorization of cars can be changed from manually control centric to automated formats	During movement restrictions, any car carrying valid e-pass, will require to go through barricades which will be able to read the number plate, and check for authorization in the database. Once authorized, the barricade will open automatically for the car to move	ANPR, Aurdino Uno, Raspberry Pi, IR sensors, QR Code reader	Social Distancing norms could be followed easily with such model. Character recognition module had low accuracy of 70%. Otherwise, the setup is cheap and implementable	Optical character recognition had less accuracy of 70% only. As this step is really important and pillar behind vehicle authentication, further study needs to be conducted on how to optimize for reaching a higher accuracy level.
3	(Yulianto et al., 2021)	Existing parking lot security systems in Indonesia is not able to counter bike thefts as they are more conventional and using outdated security systems	Intelligent parking system is proposed with Electronic Identity and image processing. Casscade filters applied at entrance for capturing region of interest and SIFT method used for matching images at the exit.	Haar Cascade Classifier, Scale Invariant Feature Transformation (SIFT), RFID-E-KTP	89% accuracy achieved with average processing time of 13.25 seconds at entrance. 100% accuracy achieved with average processing time of 19.75 seconds at exit	Region of interest detection is the major part in this scenario as that entry and exit to parking lot will depend on that. But the author did not mention anything regarding those scenarios where the detection was not successful. Moreover, the processing time at entry and exit seems a bit long
4	(Sharma et al., 2019)	Existing works are present for license plate localization but not much study has been made on identification of characters after region of interest has been selected	Haar cascade filters to provide region of interest and Py-Tesseract to be implemented for character recognition	Haar Cascade Classification for region detection. Py-Tesseract as the optical character module	90% accuracy achieved for LP detection. 94.4% accuracy achieved for character recognition	Certain characters read incorrectly during character recognition phase. This can be due to low light or poor image quality.

Table 2-2: Summarization of selected prior studies (Continued)

Study	Author	Weakness Identified	Proposed Solution	Technology Adoption	Research Outcomes	Limitation
5	(Cheon & Lee, 2019)	Most of the existing plate detection and recognition system works with fixed camera sources. With emergent of automated vehicles, it is also evident that license plate be detected by moving source too.	Haar Cascade filters used to get region of interest, DoG filters used for detecting edges, connected component labelling used to get candidate blocks, Histogram comparison and color quantization used to remove un-required color components	Haar-Cascade, DoG Filter, Connected Component labelling, Histogram, Color Quantization	Pipeline tested over Caltech dataset. Recall-86% Precision-96% F-Score-91%	Proposed pipeline was tested over Caltech Database. Real time performance study was not conducted
6	(Zhao et al., 2010)	Robust LPR system needs to be built which can handle 5 different styles of License (CY) Plates. The system built should be able to handle environmental and image variations	Expected input would be an enhanced image which will be smoothened (preprocessing). Two Haar-Cascade classifiers are joined together to create an ensemble. Input image will go through the cascade classifiers to give back the candidate region. Edge density to be applied to sort each candidate	Haar Cascade classifier for detecting license plate location, Adaboost classifier to train from number of features	Average True positive rate calculated from 5 different experiments having different mix of images was 85.54%. False positive rate of 11.44% was achieved.	Due to odd aspect ratio of the plates, when the camera is having a big angle with the license plate, detection may get failed. Also, when the headlight is on, detection may get failed due to contrast in candidate region. Moreover, the study was conducted on still images and only for Chinese license plates.
7	(Goyal et al., 2021)	Wrong side driving is one of the leading sources for major accidents. However, manual checking of such vehicles is not possible and thus such drivers evade traffic violations unnoticed	A smart process to detect and highlight such vehicle number plates, can notify incoming traffic with help of DMD display and also create a database of traffic violators	IR Proximity Sensors, Ardunio Uno Controller (ATmega328P), Raspberry Pi, IP camera, DMD Display, SSD-MobileNet, Optical Character Recognition	Overall mAP 74.3% achieved. mAP for 'car'class achieved was 76.1%. mAp for 'motorbike' class achieved was 82.3%.	IR proximity sensors has detection range of 85cm maximum. Any vehicle which moves beyond this range will not be detected.
8	(Srivastava et al., 2021)	Drivers are reckless and aware of how to exploit the loopholes of existing traffic manual system. Moreover, there is not enough man power to moderate all of the existing road round the clock. Existing systems use neural networks but not much relevant exploitation has been done in detecting bikers without helmets.	Object Detection and number plate recognition from a single camera feed. The frames collected from the feed is preprocessed with image standardization. Then the image is feed in three object detection models. These detection models detect cars, bikes and helmets	Deep learning architecture, Retinanet base model is used, ResNet is used for feature extraction, Feature Pyramid Network is used to generate multi-scale feature pyramid on single image resolution	92.6% accuracy achieved for red light violation 94.2% accuracy achieved for detecting for violation without helmets 91.7% accuracy achieved for license plate extraction	Test cases performance were better and there was a slight dip during live case scenario. For future scope, the data collected can be used for different analysis of traffic patterns. The author also mentions are dynamic traffic lighting based on the the video feed, which will reduce the unnecessary wait at traffic stoppages.

Table 2-3: Summarization of selected prior studies (Continued)

Study	Author	Weakness Identified	Proposed Solution	Technology Adoption	Research Outcomes	Limitation
9	(Shariff et al., 2021)	Human assistance for recording huge population of cars is tedious and time taking. With so many traffic violations on daily basis, it is required to handout fines in a quick process. Most current number plate recognition system is time taking and complicated	Easy application of computer vision. Conversion of image to greyscale, then bilateral filtering then canny edge detection for grabbing region of interest based on number of contours. Finally, Py-Tesseract for character recognition	Bilateral Filtering, Canny Edge Detection, Py-Tesseract	Accuracy of 88% achieved over 100 test images.	If there are multiple rectangle contours detected, then the system will not be able to detect number plate correctly. Moreover, the detection works only for number plates with white background. The proposed system has not been tested over video file and also does not include images under low light/rain/dark scenarios
10	(Younis et al., 2020)	Embedded devices cannot support huge number of calculations and configuring variables. Though deep neural nets can achieve huge accuracy it becomes computationally costly to operate on such device	Pre-trained model of MobileNet deployed through convolution networks which has separate hyper-parameters configurations for width and resolution	MobileNet, Single Shot Detection for object detection	Average Precision for specific classes achieved as shown below: Car-99.76% Person-97.76% Chair-71.07%	The dataset used had limited object classes. Performance of the pipeline will change once the dataset includes several other object classes
11	(Zhang et al., 2020)	Manual defect detection for vehicle body paint is not efficient and that accurate	Enhanced MobileNet-SSD algorithm is proposed for detecting paint defects by improving feature layer and bounding box strategy	MobileNet with Single Shot Detection, Data Enhancement for overfitting, image augmentation	Enhanced SSD MobileNet detects six traditional body paint defects with 95% accuracy.	Pool of defect image collected is less and thus may lead to low accuracy in real life. Low-level feature maps are usually not that good for predicting small targets of high-level features.

Table 2-4: Summarization of selected prior studies (Continued)

Study	Author	Weakness Identified	Proposed Solution	Technology Adoption	Research Outcomes	Limitation
13	(Peker, 2019)	Comparison of different object detection algorithms for license plate recognition	Object detection algorithms based on TensorFlow framework are trained and tested for license plate localization task.	SSD with MobileNet and Resnet50 features, Inception Layer features on Faster R-CNN, R-FCN with Resnet101 features	SSD architecture can detect larger objects compared to FR-CNN and R-FCN. SSD with Resnet50 achieved accuracy of 97.9%	Small size number plates are not detected with SSD architecture. This study is specific to Turkey
14	(Babu et al., 2019)	License plate recognition systems have limitations like light exposure, having stationary backgrounds, indoor area, specific driveways and distance between camera and object etc.etc.	Number plate detection using YOLO V3 which is applied for real time object detection and character recognition using YOLO algorithm	You only look once (YOLO algorithm) with combination of CNN	Recognition rate of 91.0% achieved	Not 100% accurate in character recognition. Similarity problem between characters like 0 and O.
15	(Varkentin & Schukin, 2019)	To control traffic flow, automatic collection of tolls, surveillance of particular vehicle	Apply neural network and pattern recognition for license plate recognition	YOLO, CNN	Recognition Rate of 0.86 achieved	Other pre-trained NN models like GoogLeNet needs to tried and the model was trained for only Russian number plates
16	(Li et al., 2019)	Car license plate recognition is considered under natural scene images. Existing approaches consider license candidate extraction and character recognition as two separate parts	Deep neural network to isolate license plates and recognize characters at the same time in single forward pass.	Convolution Network, Pooling	Detection rate of 99.73% achieved	Network faces hindrance for multi oriented license plates as multi oriented licenses get reflected
17	(Aggarwal et al., 2019)	The idea was to create an intelligent system which could recognize vehicles only by number plate	Computer vision approach inculcated to identify license plate of the vehicles. Various image processing steps were followed like, grey-scale conversion, dilation, edge processing	Smart detection system, Image segmentation, Edge processing, Region of interest	Achieved 93.34 % detection rate, FPR is 6.65%	Vehicle Position, Appearance of vehicle, damaged license plate, camouflage stickers around number plate.

Table 2-5: Summarization of selected prior studies (Continued)

Study	Author	Weakness Identified	Proposed Solution	Technology Adoption	Research Outcomes	Limitation
18	(Wanwei Wang et al., 2019)	Smart intelligent system for Chinese License Plate	Three networks proposed. P-Network, R-Network and O-Network. P-network used to obtain the candidate and bounding box. R-Network trains on the candidate. Non max suppression used to fine tune candidate. O-Network finds the four corner points of the license plate. NMS is used throughout to removed overlapping windows on the candidate	Cascaded CNN for object detection, Recurrent NN, Connectionist Temporal Classification (CTC) for character Recognition	The method achieves 98% recognition accuracy	Proposed solution was tested for Chinese Number plates and is yet to tested for other country scenarios.
19	(Zhang et al., 2019)	Object detection methods neglects the physical properties of vehicle number plates. This limits performance of detection	Two phase deep learning model to isolate license plates in unconstrained scenario. A vertical anchor mechanism is designed to jointly predict the position and confidence of each fix-width character	CNN, R-CNN, Bi-Directional LSTM	Average Precision of 97.11% achieved	The performance of the model gets degraded when the tilt angle of the characters is large
20	(Xie et al., 2018)	License number plate detection is effective under certain scenarios and assumptions but fails when the candidate region is tilted or having some part of rotation	YOLO framework based on CNN-based network for license plate identification. Using accurate rotation angle prediction and a fast intersection-over-union evaluation strategy is used.	Convolution Neural Network, YOLO Algorithm for detection.	Overall accuracy of 98.6% achieved	Multi-directional detection and recognition, Low light and poor resolution

Table 2-6: Summarization of selected prior studies (Continued)

Study	Author	Weakness Identified	Proposed Solution	Technology Adoption	Research Outcomes	Limitation
21	(Nguyen & Nguyen, 2018)	Vietnam does not have a recognition system which accommodates all the task of car number plate detection together. Full sequence of number to be recognized without pre-segmentation	Combine number plate detection, character segmentation, character recognition using CNN.	Convolution Neural Network	Precision - 98.5% Recall - 99.3% F-Measure - 98.9%	Only works on Vietnamese number plate fonts
22	(Lib et al., 2018)	Performance of deep neural network to detect car number plates and recognize the characters under natural scenarios	37 class (multiple) CNN classifiers used as cascade for detection. Multi scale sliding window used for character recognition	CNN, plate/non-plate CNN classifier, RNN with LSTM	97.56% Precision 95.24% Recall	Slow detection speed of multi scale sliding window compared to real life application.
23	(Silva & Jung, 2018)	Most license plate recognition system works on region specific plate systems.	Convolutional Neural Network for detection and to rectify distorted license plates. Once we have the candidate region, we process with OCR	License Plate, Deep learning, CNN, OCR	Average Accuracy of 89.33% achieved	Cannot detect motorcycle License Plates.
24	(Stefanović et al., 2018)	Image segmentation used to locate objects and boundaries, like lines and curves in digital image. Same to be used to isolate car number plate region from an image.	Segmenting car number plate digital image into regions with K-means algorithm. Some spatial information from a histogram-based windowing process is used	Data partitions, Digital image, K-means algorithm, Edge detector	With correct number of clusters defined by user, the model gives satisfactory results. There can be slight variation as the algorithm is tested in restricted conditions	The model performance is judged based on restricted parameters which are specific for Serbia. Region Specific

Table 2-7: Summarization of selected prior studies (Continued)

Study	Author	Weakness Identified	Proposed Solution	Technology Adoption	Research Outcomes	Limitation
25	(Lee et al., 2017)	Real-time image processing is computationally expensive which can cause load to normal computers.	Leveraging the use Graphics Processing Units (GPUs) to accelerate graphics operations along with deep learning algorithm such as CNN to handle complex operation	NVIDIA Jetson TX1 boards, CNN, Embedded System	Recognition Rate of 95.24% achieved	Fairly small data was used. Model was unable to detect broken and reflective number plates
26	(Omran & Jarallah, 2017)	Three different styles of number plate are being used in Iraq. All of them vary in size	Optical character recognition is used. Combination of template matching & correlation approach used for recognition by segmenting all spaces into number sub-image & character sub-image.	OCR, Template Matching, Character Segmentation	Extraction Rate - 87.5% Recognition Rate - 85.7%	Fairly small data was used and the model was region specific to Iraq
27	(Sasi et al., 2017)	Comparison of performance amongst algorithms such as Prewitt, Roberts, Canny, Mexican Hat and Sobel operators with proposed solution for edge detection	Ant colony optimization (ACO) for detecting edges. Hierarchical combined classification method based on inductive learning and SVM for individual character recognition.	Ant Colony Optimization for detecting edges. Kohonen Neural Network as extraction and segmentation algorithm. SVM for character classification	94% accuracy achieved by modified ant colony optimizer.	NN with other classification algorithm for character detection was not tested
28	(Mandi et al., 2017)	Absence of efficient, effective and cheap vehicle identification system is required for developing countries as there is spike in number of cars flowing. Along with that, there is also need of surveillance security	Object oriented analysis and design methodology implemented. Mobile based solution with potentiality of automatic license plate detection	Optical Character Recognition (OCR), ANPR, Character Segmentation, Character Recognition	Reduced car registration time from 30 secs to 6 secs	OCR may not convert characters very large or small font sizes.

Table 2-8: Summarization of selected prior studies (Continued)

Study	Author	Weakness Identified	Proposed Solution	Technology Adoption	Research Outcomes	Limitation
29	(Balaji & Rajesh, 2017)	Vehicle registration plate detection under different lighting condition is challenging for the current world scenario	Isolation and extraction of number plate candidate region, Character Segmentation, Character Recognition	License Plate Region Recognition, Segmentation, Noise removal, Bounding Box and filter process.	Detection failed for blur images. Similarity in shape and edge of certain characters such as O and 0, O and D, 8 and B ,5 and S etc. raised to failure in identification	Not able to detect blur images, broken number plate and character similarities
30	(Montazzolli & Jung, 2017)	Most license plate recognition system is country focused. Thus, the pipeline designed are purely for commercial purpose and utilize private datasets and lack detailed information. No such pipeline exists for Brazilian context	Convolution neural network architecture is built for Brazilian license plate detection	LPD Network, CNN, YOLO, Character Segmentation & Recognition	Letter Recognition - 63.43% Number Recognition - 93.03%	Unbalanced Dataset for character recognition which resulted in poor accuracy. Spatial transforming network and Part based network needs to be tested
31	(Khan et al., 2017)	Most car number plate recognition system faced hindrance to detect different design and font size of number plate characters	Multiple template matching process deployed for recognition of characters so it can detect variety of font and size. Once detected, noise is removed dynamically by adjusting pixel value.	Template Matching, Noise Reduction by pixel adjustment	Accuracy of 93% achieved	Intensity of light specially at night needs to be considered. Different illumination background needs to be tested on the same datasaet
32	(Khan & Shah, 2016)	Performance of Car Number Plate Recognition (CNPR) under low light	Multiple template matching process deployed to improve car number plate recognition algorithm via considering light intensity & distance between object and camera	Character Recognition, Digital Image Processing, ANN	Template Matching technique providing high accuracy. For hand written script accuracy achieved is 85.1% and for printed script accuracy is 90.93%	Impact of camera quality, light intensity (especially at night)

Table 2-9: Summarization of selected prior studies (Continued)

Study	Author	Weakness Identified	Proposed Solution	Technology Adoption	Research Outcomes	Limitation
33	(K & Raghunadh, 2016)	Surveillance security, vehicle identification of particular vehicle depending on vehicle number plate recognition	Image processing of capture frame, Isolation of license number plate region, Segmentation and Character Recognition of license plate.	LPD, Character Segmentation & Recognition	Extraction Accuracy 93.3% Segmentation Accuracy 86.67% Recognition Accuracy 93.33%	Not able to detect blur images, broken number plate and character similarities of certain characters such as O and 0, O and D, 8 and B ,5 and S etc. raised to failure in identification
34	(Tiwari et al., 2016)	Parking space management to restrict cars allotted to specific space be identified in case of wrong parking. Automatic number plate recognition system to retrieve car number plates of such scenarios	Vehicle detection at gate entrance of a parking area using computer recognition system with help of deep learning models	ANPR, Computer Recognition System, OCR, Vehicle Number Plate, Matlab	Less costly with good accuracy ANPR system developed	If the image thresholding is too low, class recognition will be ignored. The system is also designed for specific region
35	(Agarwal & Goswami, 2016)	Number plates vary from region to region and have different size, shape, font style. Pre-existing methods have computational expensive methods with low recognition results	Edge detection, Character segmentation with morphological operations.	Edge Detection, Plate Extraction, Character Segmentation	High Recognition Results with low computational time	Results achieved was only in context of Indian car number plates. Video stream was also not considered for this study
36	(Yogheedha et al., 2018)	Manual interference of human to collect data as license plate recognition in Malaysia manages to identify the license number but not store it automatically	Image processing for number plate segregation and template matching approach for character recognition	Image processing, image segmentation, optical character recognition	92.85% accuracy achieved	Dataset used was too small and for standard number plate template to region
37	(Soon et al., 2012)	As the number of vehicles is increasing in Malaysia, manual processing at electronic toll gates, surveillance system for law enforcement needs automated process for identifying vehicles	Number plate detection with combination of Adaboost and CCA (Connected Component Analysis) and then KNN for character recognition. Several preprocessing applied to enhance image quality	Adaboost & Connected Component Analysis for plate detection. KNN for character recognition	Overall accuracy of 96.84% achieved for detection	KNN character classifier is not accurate and robust enough

Table 2-10: Summarization of selected prior studies (Continued)

Study	Author	Weakness Identified	Proposed Solution	Technology Adoption	Research Outcomes	Limitation
38	(Zakaria & Suandi, 2010)	Hough transform is computationally expensive and requires huge memory. Edge-based approach is sensitive to complex background, but is simple and fast.	Combination of color information method along with template matching algorithm.	Template Matching, Top Hat Filtering, Contrast Correction Method, Color Information method	Accuracy of 97.1% achieved	Character Segmentation and Character recognition was not included. Detection process could have been enhanced by that
39	(Priya & Perumal, 2014)	Intelligent and smart transport system which can extract information from car number plate	Edge Detection and efficient morphological operations used to use to extract numbers from the candidate license plate.	Edge Detection, Character Segmentation, Filtering Techniques, Optical Character Recognition	75% Accuracy achieved. The Sample set was too small.	A very small dataset was tested. Only straight flat image was detected correctly.
40	(Du et al., 2013)	Most detection systems face challenge due to different environmental conditions. Light conditions and surrounding such as indoors or outdoors impact the output to great extent. Pipeline built should also be able to detect plates devoid of any region and state	Four Stage Solution provided: Stage1: Acquire car image using camera Stage2: Extract license plate based on boundary Stage3: Segment and extract characters Stage4: Recognize characters by template matching or NN	Image Acquisition & Processing, License Plate Segmentation with Edge Detection Algorithm, NN or template matching for character recognition	Image Acquisition, camera resolution & shutter speed matters. Extraction of edge (boundaries) is crucial. No explicit outcome on which is good method as per performance.	Limitation in detecting multi style plate design and video-based feed

Existing literature shows a variety of approaches which includes mostly CNN based approaches, template matching techniques, computer recognition systems, Adaboost & connected component analysis for license plate detection. For character recognition OCR is used in most cases, NN combined with template matching, K-nearest neighbor and connectionist temporal classification for character recognition. Most of the prior studies result in high prediction accuracy within 85%-100% range. Majority of prior literature faced challenge when came to processing time. (Mangal et al., 2020), (Yulianto et al., 2021), (Sharma et al., 2019), (Cheon & Lee, 2019) and (Zhao et al., 2010) studied Haar Cascade classifiers for detection. (Goyal et al., 2021), (Srivastava et al., 2021),(Younis et al., 2020), (Zhang et al., 2020) studied application of SSD-MobileNet. (Li et al., 2019) used deep neural network to isolate license plates and recognize characters at the same time in single forward pass., achieved the best accuracy of 99.73%. But network build was not suitable for multi-oriented license plates. (Darapaneni et al., 2020), (Babu et al., 2019), (Montazzolli & Jung, 2017), (Xie et al., 2018) and (Varkentin & Schukin, 2019) used YOLO algorithm, which sped up the processing power multiple times as the algorithm applied a single neural network to a frame divided into multiple grid system, were each grid calculates the probabilities of the bounding boxes formed. Earlier convolution models were applied to multiple location and scaled of image, which used to consume much of the processing time.

CHAPTER 3: METHODOLOGY

When it comes to segmenting any ANPR system we have three broad segments. As shown in Figure 3.1,these will be object detection, number plate region extraction and character recognition.

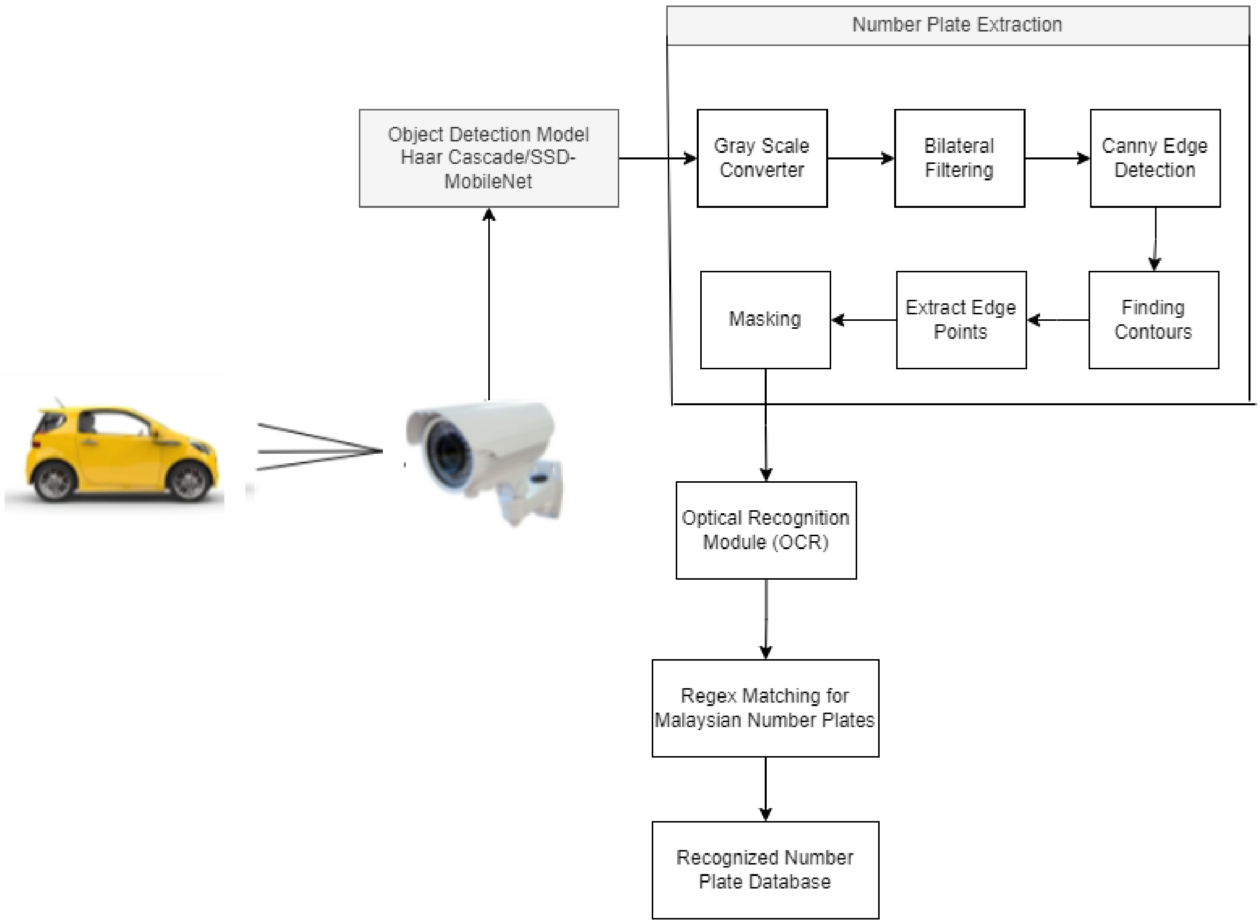


Figure 3.1: ANPR Architecture

3.1 Object Detection

The object detection module provides us with our region of interest. In this section we are going to discuss about functioning of both Haar-Cascade and MobileNet-SSD.

3.1.1 Haar-Cascade

The main object for ANPR system is to detect candidate region. And the utmost requirement to do so is with processing speed. As we are targeting to detect number plates from moving vehicles, the proposed system should be able to do it at a great speed. We also need to keep in mind the complexity involved in doing so as with moving vehicles the frame size will keep changing with lot of variations in the candidate region as well as change in direction.(Cheon & Lee, 2019).

(Viola & Jones, 2001) proposed a simple yet powerful object detection algorithm using boosted cascade of simple features. The authors propose an algorithm which can detect objects without getting impacted by the size of the frame or position of the object within the frame. Detection of region of interest was based on features that was considered for the paper. As shown in Figure 3.2, we can see that the Haar filters are distributed over basic three types, two-rectangle feature (A & B), three rectangle features (C) and four rectangle features (D).

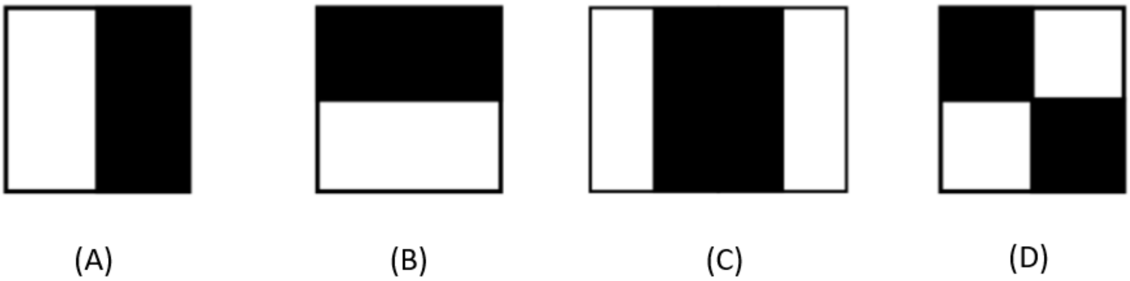
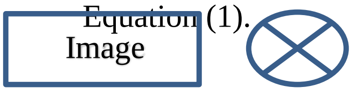


Figure 3.2: Haar Filters

The orientation may change as per need but the base features remain the same.Figure 3.2: Haar Each of these rectangles can be a pixel or even part of the frame, which thus becomes scalable. We have to remember that Haar features are basically computed using Haar filters. Haar filters are again based on Haar wavelets, which are basically the square function. As shown in Figure 3.2: Haar Filters, a very interesting thing that we can deduce is that all the filters are having only two values, either it is black or it its white. This makes it computationally cheap during operations. For each one of the Haar filters shown, we apply it as correlation on the image which provides us with a number at each pixel as shown in Equation (1).



Haar Filters [H]a = $v_a[i, j]$ **(1)**

Equation (1) provides us with all the Haar vectors. Every scale will have its own feature vectors. Every white is equal to +1 and every black is equal to -1. Equation (2) show the response to Filter [H]a at location, pixel value (i , j).

$$v_a[i,j] = \sum_m \sum_n I[m-i, n-j] H_a[m,n] \qquad \textbf{(2)}$$

Equation (2) can be simplified by writing it like equation (3) as shown below.

$$v_a[i,j] = \sum(\text{Pixel intensities in white area}) - \sum(\text{Pixel intensities in black area}) \qquad \textbf{(3)}$$

As shown in equation (3) we can now understand that all the operation is basically about addition and subtraction, which is computationally much more cheap and faster. Author (Viola & Jones, 2001) had used an degenerative binary tree (cascade) which would filter out fast the probable features which are not related to face. Small blocks or Haar filter values were looked up to understand if that is a feature value of any face. If the features matched then that small block of features was to be added to other features, thus making sure that all the feature blocks which is not containing any face part is being removed. Thus, objects were detected with Haar filters.

3.1.2 MobileNet Architecture

MobileNet object detection component provides us with class-index of the objects along with their respective prediction probabilities and bounding box edges. We filter out our desired class objects with more than 0.65 prediction probability. For object detection, we are implementing SSD-MobileNet. SSD-Single Shot Multi-Box Detection is a neural network architecture designed to detect object classes, which means extraction of bounding boxes and object classification at one go. The author in his novel literature has mentioned the use of VGG neural network as the base network, which is the feature extractor for the detection (Liu et al., 2016), on top of SSD architecture. So, combination of base network and detection network, two types of deep neural networks are being used here. The high-level features for classification object come from the base network. In this paper, we are using MobileNet as the base network. MobileNet is an efficient architecture introduced by Google which is using depthwise and pointwise convolutions as shown in Figure 3.3 (Howard et al., 2017). To reduce the model size and complexities depthwise separable convolution is used. It is being used for classification purposes, to extract features for detection. As shown in Figure 3.3, we can see that depthwise convolution is followed by pointwise convolution.

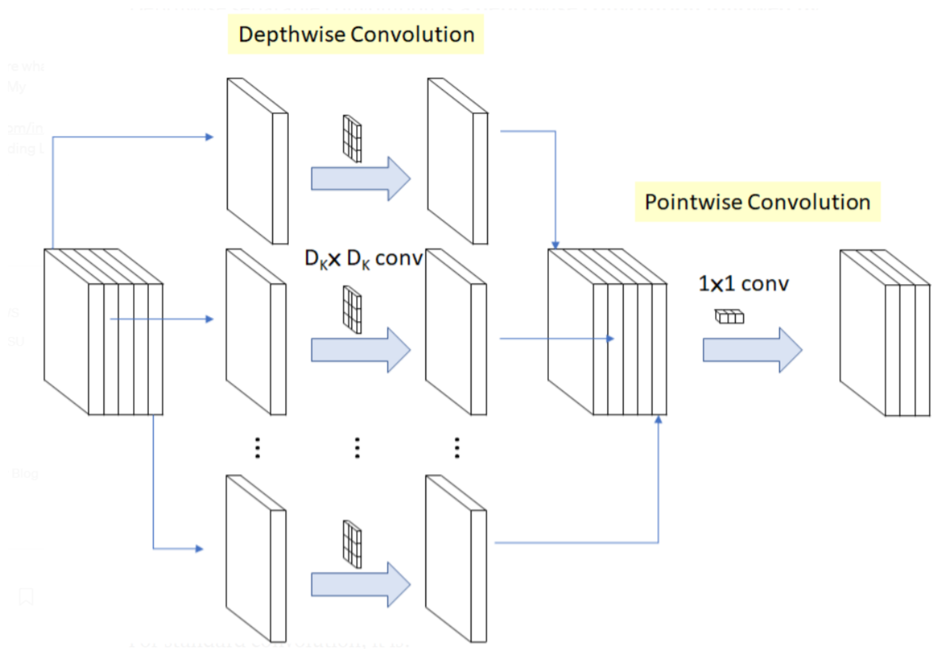


Figure 3.3: Depthwise Separable Convolution(Howard et al., 2017)

Depthwise convolution is the channel-wise $D_k\ X\ D_k$ spatial convolution. As shown in Figure 3.3 , for 5 channels, we have 5 X $D_k\ X\ D_k$ spatial convolution. Pointwise convolution is always 1×1 conv. This is to change the dimension of the channel.

Convolution is always a product of two functions which produces a third function which can be seen as an altered version of the first function. Let us suppose that we have two functions, j and k. The second function, k, is considered as the filter. A spatial convolution operation is when f function is defined on a variable(spatial) like x rather than a time(t). Convolution for functions j(x) and k(x) on a variable x is shown in equation (4).

In case of images, the functions are always of two variables. In image processing the image made by lens is always a continuous function. Thus the function happens to be j(x,y). In those cases a smoothening filter is applied k(x,y) and we get equation (5) as mentioned.

$$j(x) * \mathrm{k}(x) = \int_{-\infty}^{\infty} \mathrm{j}(\tau).\,\mathrm{k}(x-\tau)d\tau \tag{4}$$

$$j(x,y) * \mathrm{k}(x,y) = \int_{\tau_1=-\infty}^{\infty}\int_{\tau_2=-\infty}^{\infty} \mathrm{j}(\tau_1,\tau_2).\,\mathrm{k}(x-\tau_1, y-\tau_2)d\tau_1 d\tau_2 \tag{5}$$

where * means convolution operation and. means simple multiplication.

By referring to Figure 3.3 we can find the equation of the total operation that is being performed by depth-wise convolution and pointwise convolution. As shown in equation (6), the first part is referring to depth-wise operation and the second part is referring to pointwise operation. Here, M = No of in channels , N = No of out channels, D_k is the kernel size and D_f is the feature map size (Howard et al., 2017).

$$D_K.D_K.M.D_F.D_F + M.N.D_F.D_F \tag{6}$$

$$D_K.D_K.M.N.D_F.D_F \tag{7}$$

Equation (7) denotes standard convolution operation. When we divide equation (6) with equation (7), we get the total computational/operational reduction. When we apply architecture for 3x3 kernel size, we can achieve 8-9 times less computation by this method.

3.1.3 Single Shot Multibox Detection (SSD)

Single shot detection has two components. A base network, which in our case is MobileNet, for extracted feature mapping. Another component is to apply convolution filter to detect objects. Architectural representation of single shot detector is show in Figure 3.3: Depthwise Separable Convolution(Howard et al., 2017).

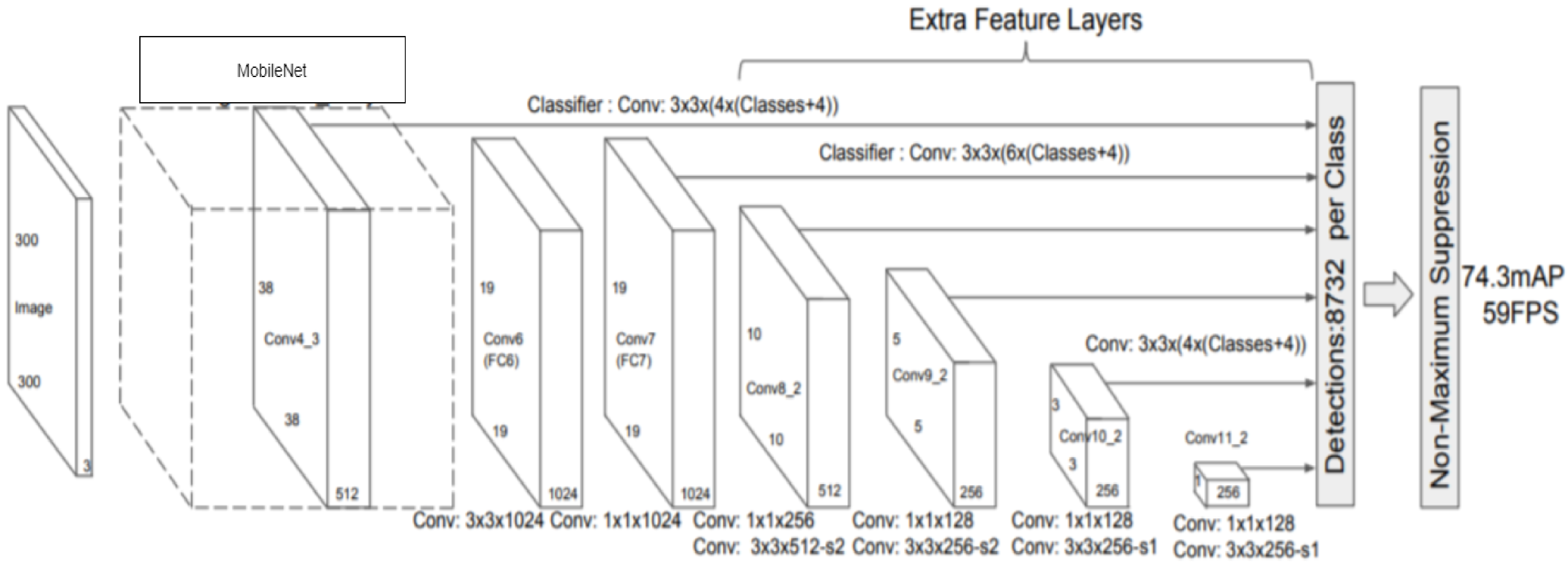


Figure 3.4 Single Shot Detection Architecture (Liu et al., 2016)

MobileNet provides us with all the required feature maps required. The six-layer convolution network will perform the classification detection of object class. The architecture is able to make 8732 predictions per object class. The architecture will check the confidence score each box of 8732 predictions per object class and will pick only top 200 predictions from the image. This is handled with help of Non-Max Suppression where it selects few entities out of overlapping entities based on probability. Calculation for reference is provided. At Conv4_3, it is of size 38×38×512.

3×3 Kernel convolution is applied. There are 4 bounding boxes and each bounding box will have (classes + 4) outputs. Classes cannot be zero. At least one class, background is always included. Thus, at Conv4_3, the output is 38×38×4×(class+4). Let us suppose that there are ten object classes. By default, one background class is always inclusive. Then the output will be 38×38×4×(10 + 1 + 4) = 86,640. So,for bounding box calculation we have:

- Conv4_3: 38×38×4 = 5776 bounding boxes (4 boxes for each location)
- Conv7: 19×19×6 = 2166 bounding boxes (6 boxes for each location)
- Conv8_2: 10×10×6 = 600 bounding boxes (6 boxes for each location)
- Conv9_2: 5×5×6 = 150 bounding boxes (6 boxes for each location)
- Conv10_2: 3×3×4 = 36 bounding boxes (4 boxes for each location)
- Conv11_2: 1×1×4 = 4 bounding boxes (4 boxes for each location)

The total number of bounding boxes comes upto 8732 which is our prediction probability per object class.

As object detection, the architecture not only predicts object classes but also locates their bounding boxes. The subtle difference in object classification and object detection is that in case of object classification the prediction is about if the class of object is present or not. But for object detection, not only it provides with prediction probability of the object class but also the boundary location of the object. Instead of using normal sliding window operation for convolution networks, single shot detector divides the image in multiple regions like a grid. Every individual grid is tasked with detecting objects in the that particular region of image. If there is no object detected in a particular grid, then we are considering it as null and that particular location is ignored. There are possibilities of certain scenario where there are many objects in a single grid or there are multiple objects of different size and shapes that need to be detected. To handle this, we have anchor. This is also known as ground truth boxes. While training, there is something called matching phase. During this phase anchor boxes are connected with the bounding boxes of every ground truth object within a frame. Anchor boxes are predefined, precalculated, fixed region of probable space and approximate box predictions. The highest region of overlap with the anchor box determines object's class, its probability and its location. This is the base principal of training the network and for predicting detected class of objects and their locations. In practice, each anchor box is specified by an aspect ratio. Aspect ratios of anchor boxes are pre-defined in single shot detection architecture. This allows to accommodate different size and shape of objects.

3.2 Number Plate Region Extraction

Once we have the desired frame with our intended class object, we have to find out our candidate region, number plate section. This is done with help of computer vision (CV2) library. The series of steps which are required to extract number plate region is shown in Figure 3.5 below.

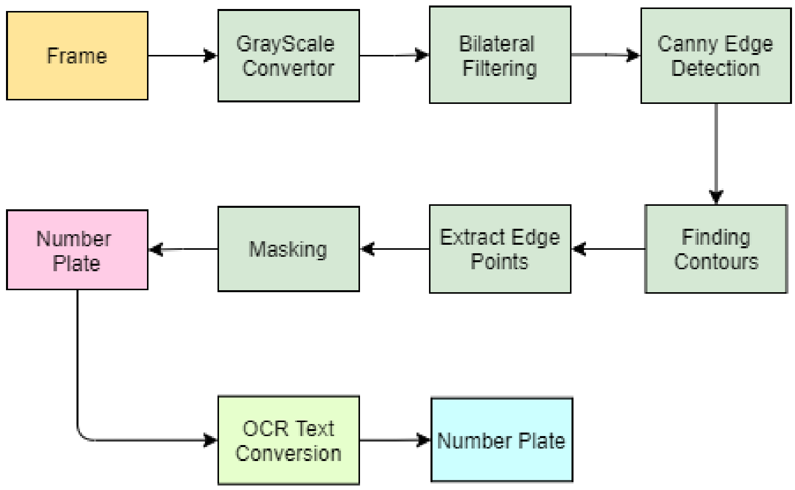


Figure 3.5 Number Plate Extraction

3.2.1 Gray Scale Conversion

In Figure 3.6 we can see our reference image. We need to convert the frame to Gray Scale as shown in Figure 3.7. This is very important as the consequent steps cannot be performed without this.

Figure 3.6 Original Car Image

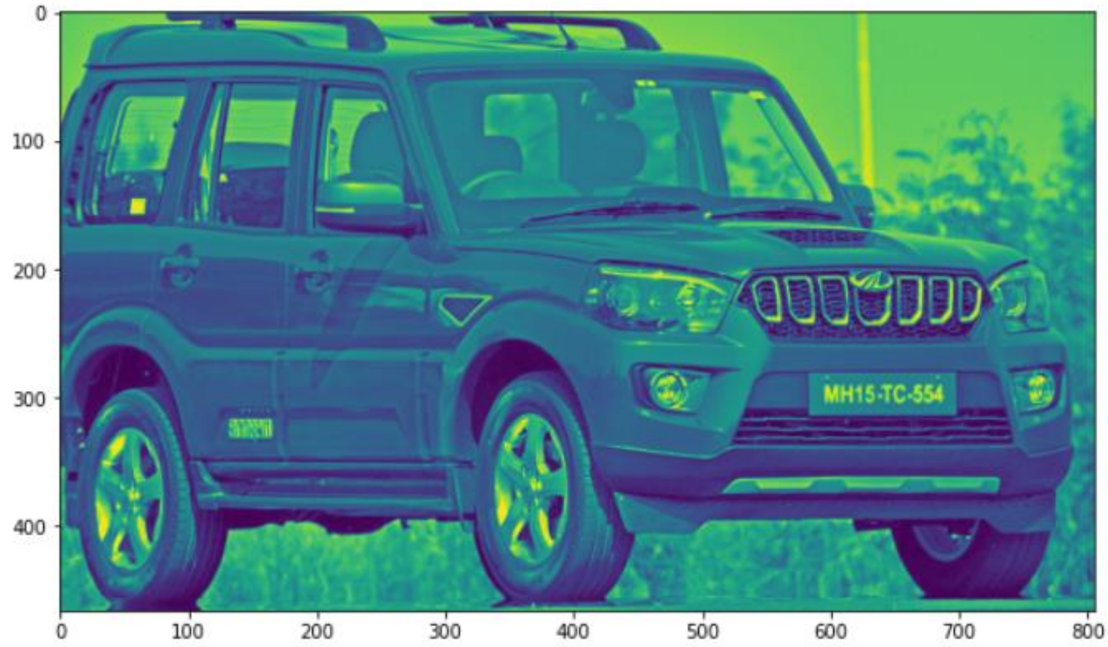


Figure 3.7 Gray Scale Converted Frame

3.2.2 Bilateral Filtering

Next, we are proceeding with bilateral filtering process as shown in Figure 3.8. A bilateral filter is used for reducing noise and smoothening images. There are other denoising filters which are present, like Average Filer, Median Filter & Gaussian filter. Bilateral filter is almost like a Gaussian filter with only difference is that Gaussian filter blurs uniformly the content, edges everything, but Bilateral filter retains the edge.

Figure 3.8 Bilateral Filtered Frame

3.2.3 Canny Edge Detection

Canny edge detector is used for detecting edges as shown in Figure 3.9. Pre-requisite for canny edge is to have noise reduced image, which we already have from bilateral filtering. The canny edge detector points out all the edges with help of non-maximum suppression as an edge is a sharp difference and change in frames pixel values.

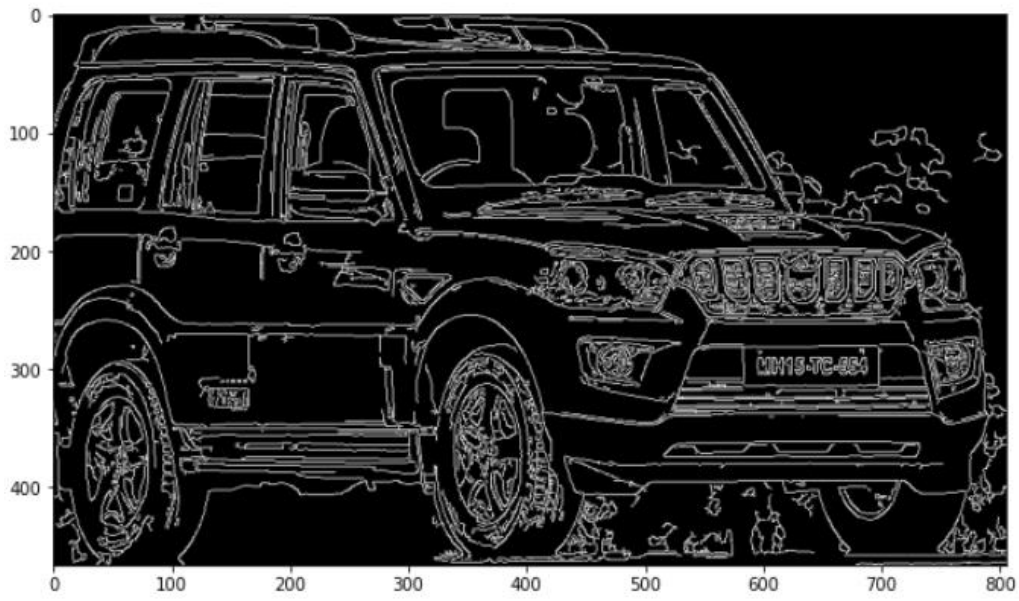


Figure 3.9 Canny Edge Detection

3.2.4 Find Contours

We find contours which is basically a curve which is joining all the points, continuous in nature, along the edges of same color intensity. When we are applying find-contour operation the original frame will get impacted. Copy of image should be processed during this phase. This function is to detect objects in an image. Sometimes objects are separated and located at different places without any overlapping. But in some cases, many of the objects overlap one another. The outer edge is parent and the inner edge is referred as the child. The relationship between contours in an image is established like these. Due to the relationship of parent and child we can call this a hierarchy. Retr_tree is passed as contour retrieval mode which provides the total hierarchy of all the edges in a tree hierarchy like structure. Contour approximation method needs to specified also in this step otherwise it will not be able to not be able to judge (x,y) coordinates of the boundary points of any shape. We use chain_approx_simple to return only the two end points of a line otherwise it will return all the point coordinates in a line.

3.2.5 Edge Point Extraction

Now we use approxPolyDP() to approximate polygons. The target here is to fetch the number plate region, which is always in a rectangular shape. This step helps us to fetch the polygonal shaped edges or contour points. Rectangle is a four-sided polygon. Thus, any approximation which returns four sides can be our possible edge point for the number plate. Edge points returned are basically (x, y) coordinates of four corners of a rectangle.

3.2.6 Masking

Masking is the process to applied before we draw contours of the new edge points that was approximated. Once the original image is masked and only the candidate region is passed, we get our desired number plate output as shown in Figure 3.10.

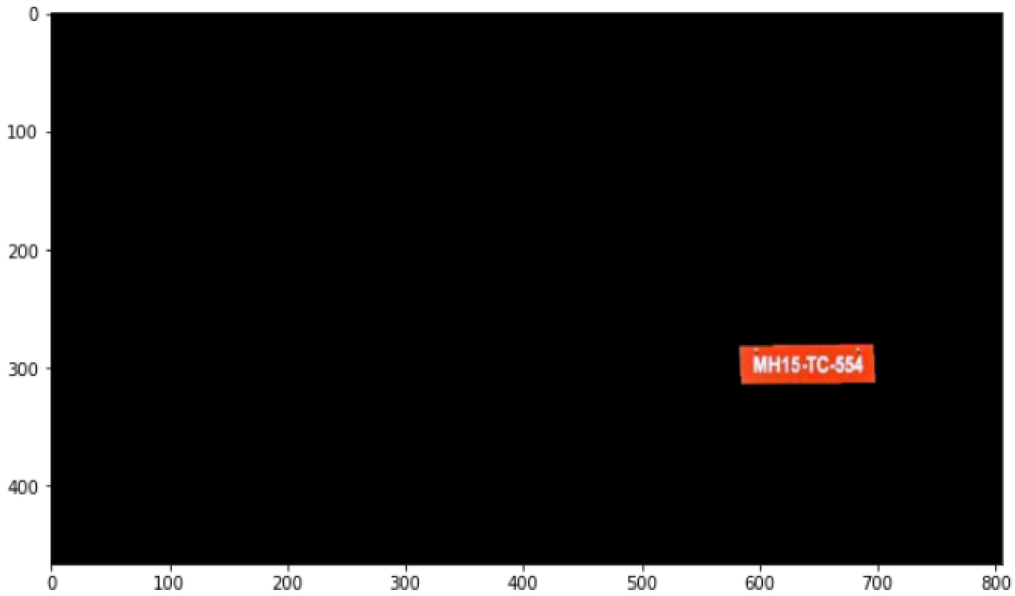


Figure 3.10 Masked Edge Points

Now we have our desired area. We segregate the number plate region from the image for further processing as shown in Figure 3.11.

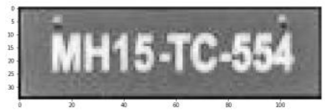


Figure 3.11 Number Plate

3.3 Character Recognition

For character recognition from the license plate segregated, we are using EasyOCR. This is a library for optical recognition. EasyOCR is based on deep learning models which involves text recognition and detection models. This optical character recognition process returns text along with its prediction probability. We have deployed a regex conditioning which helps us to format all the text as per Malaysian car number plate formats. Different car number plate formats were designed into regex formats to validate the text conversion. Most of Malaysian number plates follow a format of Sxx @@@@ were

- S - The state or territory prefix. (e.g.: W = Kuala Lumpur, A = Perak etc)
- x - The alphabetical sequences. (e.g.: A, B, C ... X, Y)
- @ - The number sequence (0,1, 2, 3 to, 9999)
- Please find below Table 3-1 for state references

Table 3-1: State Prefix for Malaysia

State	Prefix	State	Prefix
Kuala Lumpur	W	Kedah	K
Penang	P	Pahang	C
Terengganu	T	Johor	J
Malacca	M	Perak	A
Negeri Sembilan	N	Selangor	B
Perlis	R	Kelantan	D

Number plate format being used for Sarawak is as QDx @@@@ x, were

- Q - The constant prefix for all Sarawak number plates.
- D - The division prefix. (e.g. : A = Kuching, M = Miri)
- x - The alphabetical sequences. (e.g.: A, B, C ... X, Y, except Q & S are restricted)
- @ - The number sequence (0,1, 2, 3 to 9999)

Refer Table 3-2 for division prefix of different regions of Sarawak.

Table 3-2: Registration Plates for Region Sarawak

Division	Prefix	Division	Prefix	Division	Prefix
Bintulu	QT	Samarahan	QC	Kuching	QA/QK
Sarikei	QR	Limbang	QL	Sibu and Mukah	QS
Miri	QM	Kapit	QP	Sri Aman and Betong	QB

Number plate format being used for Sabah is as SDx @@@@ x, were

- S - The constant prefix for all Sabah number plates.
- D - The division prefix. (e.g.: A = West Coast, T = Tawau)
- x - The alphabetical sequences. (e.g.: A, B, C ... X, Y, except Q & S are restricted)
- @ - The number sequence (0,1, 2, 3 to 9999)

Refer Table 3-3Table 3-2 for division prefix of different regions of Sabah.

Table 3-3: Registration Plates for Sabah

Division	Prefix	Division	Prefix	Division	Prefix
Sandakan	SS	Kudat	SK	Beaufort	SB
Tawau	ST	Labuan (replaced)	SL	Lahad Datu	SD
Sabah Government	SG	West Coast	SA, SAA-SAB	Keningau	SU

Number plate format being used for Taxi is as HSx @@@@ were

- H - The constant prefix for all taxi number plates.
- S - The state or territory prefix. (e.g.: W = Kuala Lumpur, P = Penang)
- x - The alphabetical sequence. (e.g.: A, B, C ... X, Y)
- @ - The number sequence (0, 1, 2, 3 to 9999)

Refer Table 3-4for division prefix for different regions.

Table 3-4: Taxi License plates prefixes

State	Prefix	State	Prefix
Kuala Lumpur	HW	Selangor	HB
Johor	HJ	Penang	HP
Pahang	HC	Malacca	HM
Kelantan	HD	Sarawak	HQ
Sabah (replaced)	HE	Perlis	HR
Perak	HA	Sabah	HS
Kedah	HK	Terengganu	HT
Labuan	HL	Negeri Sembilan	HN

Certain taxis around Shah Alam use somewhat a different format HB #### SA. Certain limo service from airport also uses a different format of LIMO #### S. Military services use format ZB ####, were

- Z - Malaysian Armed Forces vehicles.
- B - Segment prefix. (e.g. : A = Prior use before division of different services, D = Malaysian Army, L = Royal Malaysian Navy, U = Royal Malaysian Air Force, Z = Ministry of Defense)
- # - The number sequence. (e.g.: 1, 2, 3 to 9999)

All the mentioned regex formats help to make sure that the optical recognized character is in format because there are times when the converted texts are incorrect with high confidence probability and at times texts are correct with low confidence probability.

CHAPTER 4: RESULTS

For this section, we would discuss the performance of the pipeline and the performance of individual components in the pipeline. First conducted our experiment with Haar-Cascade object detection on car images. Then we conducted experiment with SSD-MobileNet for comparison. Accuracy, Precision and Recall metrics were used for evaluation.

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{All Samples}} \quad \textbf{(8)}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad \textbf{(9)}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad \textbf{(10)}$$

True Positive (TP): Correct Prediction of positive class

True Negative (TN): Correct Prediction of negative class

False Positive (FP): Incorrect Prediction of positive class as negative class

False Negative (FN): Incorrect Prediction of negative class as positive class

4.1 Result for HAAR Cascade

As shown below in Table 4-1: Evaluation Metrics for Haar-Cascade, we can see that Haar-Cascade detection achieved an accuracy of 0.71. It has achieved high precision value with a moderate recall value as shown in the table.

Table 4-1: Evaluation Metrics for Haar-Cascade

Accuracy	Precision	Recall
0.72	0.98	0.67

A precision recall curve will show us the true summarization of trade-off between true positives and predicted values which are from positive class. As shown in Figure 4.1 Precision-

Recall Curve Haar Cascade, we see the precision recall trade off. We can see as the recall increases, precision decreases. This is so because as the number of positive samples (for our case, car) increase, classification accuracy for each sample decreases.

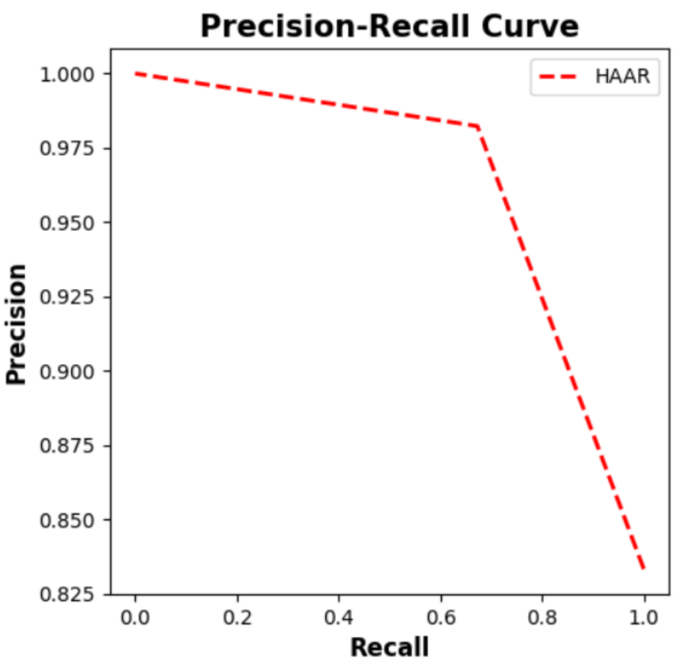


Figure 4.1 Precision-Recall Curve Haar Cascade

4.2 Result for SSD-MobileNet Detection

As shown below in Table 4-2: Evaluation Metrics for SSD-MobileNet, we can see that SSD-MobilNet has achieved a detection accuracy of 0.70. The model achieved near about similar precision and recall values. For this instance, we have low false positive rate and low false negative rate too.

Table 4-2: Evaluation Metrics for SSD-MobileNet

Accuracy	Precision	Recall
0.70	0.84	0.81

A precision recall curve will show us the true summarization of trade-off between true positives and predicted values which are from positive class. As shown in Figure 4.2 Precision Recall Curve for SSD-MobileNet Detection, we see the precision recall trade off. We can see as the recall increases, precision drastically decreases this is cause as the number of true positives increase, false positives also start increasing with change of threshold.

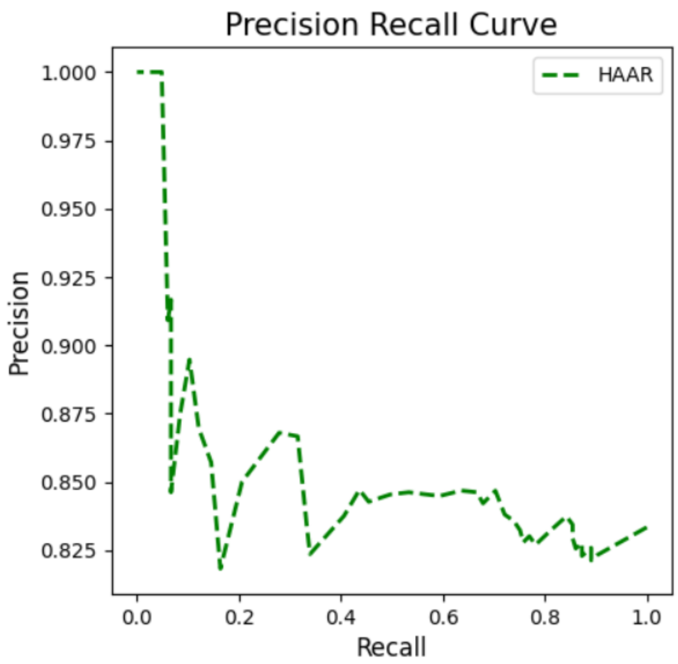


Figure 4.2 Precision Recall Curve for SSD-MobileNet Detection

Shown below in Figure 4.3 Precision Recall Curve between Haar-Cascade and SSD-MobileNet performanceFigure 4.3 Precision Recall Curve between Haar-Cascade and SSD-MobileNet performance, is for comparative study of the models.

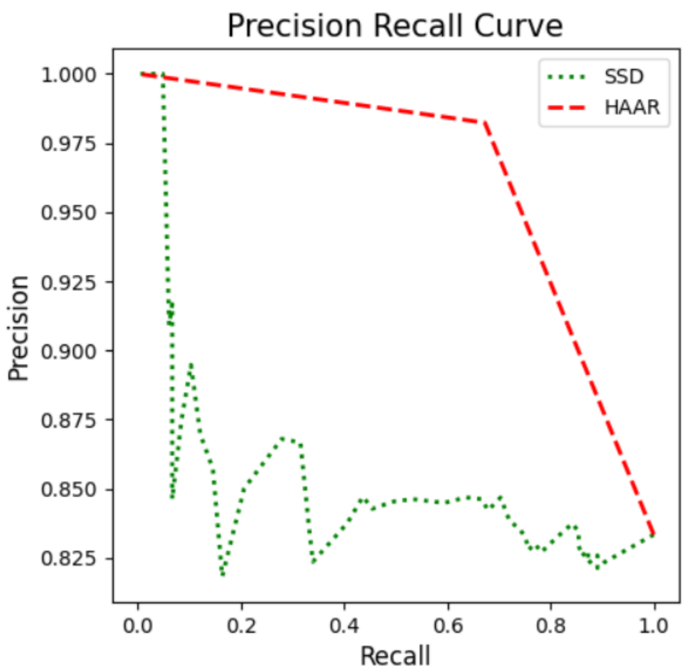


Figure 4.3 Precision Recall Curve between Haar-Cascade and SSD-MobileNet performance

Performance rate or success rate, to quantify the whole pipeline, which is basically the total number of successful number plate recognition to total number of number plates is used for comparison.

Table 4-3: Success Rate

Pipeline	Object Detection	Number Plate Extraction
Haar-Cascade	0.72	0.67
SSD-MobileNet	0.70	0.53

CHAPTER 5: CONCLUSION

With advancement in computer technology, both on the hardware parts and on the software part, and along with high end devices, we can achieve high automation systems like ANPR. This kind of system helps us to reduce the dependency from manual workforce and helps us with better reliable decision support system. During the experiments performed it was observed that Haar-Cascade model performed better with higher accuracy than SSD-MobileNet for number plate detection. It's not only much accurate, Haar-Cascade performed better for live video feeds. It was much faster than SSD-MobileNet. For Haar-Cascade, it was observed that the car number plate could only be detected once the car slows down to near about 0 km/hr or stops completely. Speed of the car will have a high impact on the number plate detection. Also, if the car is having a lot of stickers/decals fitted, the model does not perform good. The model performs best when the car number plate is in the front view or back view. For side views and all, the model performance goes down. Too much zoomed in on the car number plate will also cause performance issue as it fails to relate it to cars. Performance of SSD-MobileNet is also impacted if it has to find out small region of extraction. It is much suited for detecting large objects. SSD-MobilNet also misclassifies cars as bus or trucks at times. It was also noticed that OCR module had good accuracy for image to text conversions. Sometimes, it would generate multiple text for the same number plate, cause of the video feed. Detection of cars with unique number plates was achieved and updated in temporary database.

For future scope, camera live feed should be studied in depth as the quality and image speed will have a high impact on the model performance. Camera's installed should be robust to capture images at high pixel and FPS. Clock synchronization of all cameras at security gate should also be checked. Legitimacy of vehicles entering such bounded campus can also be checked with help of information from national vehicle registration service. Carbon emission of vehicles can also be estimated by mapping the entry-exit time. Moreover, traffic can be predicted and diverted for all vehicles with such a system in place.

REFERENCES

Agarwal, A., & Goswami, S. (2016). *An Efficient Algorithm for Automatic Car Plate Detection & Recognition* An Efficient Algorithm for Automatic Car Plate Detection & Recognition, sci-hub.se/10.1109/cict.2016.133

Aggarwal, A., Rani, A., & Kumar, M. (2019). A robust method to authenticate car license plates using segmentation and ROI based approach. *Emerald Insight*, *9*(4). https://doi.org/https://doi.org/10.1108/SASBE-07-2019-0083

Babu, R. N., Sowmya, V., & Soman, K. P. (2019). *Indian Car Number Plate Recognition using Deep Learning* 2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), https://sci-hub.se/10.1109/ICICICT46008.2019.8993238

Balaji, G. N., & Rajesh, D. (2017). Smart Vehicle Number Plate Detection System for Different Countries Using an Improved Segmentation Method. *Imperial Journal of Interdisciplinary Research (IJIR)*. https://www.researchgate.net/profile/Naveen-Balaji-Gowthaman/publication/317401424_Smart_Vehicle_Number_Plate_Detection_System_for_Different_Countries_Using_an_Improved_Segmentation_Method/links/593922674585153206fda4e8/Smart-Vehicle-Number-Plate-Detection-System-for-Different-Countries-Using-an-Improved-Segmentation-Method.pdf

Cheon, Y., & Lee, C. (2019). *License Plate Extraction for Moving Vehicles* 2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA),

Darapaneni, N., Mogeraya, K., Mandal, S., Narayanan, A., Siva, P., Paduri, A. R., Khan, F., & Agadi, P. M. (2020). *Computer Vision based License Plate Detection for Automated Vehicle Parking Management System* 2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON),

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 (Volume: 23, Issue: 2, Feb. 2013)*. https://doi.org/10.1109/TCSVT.2012.2203741

Goyal, A., Mangal, A., & Kulkarni, A. R. (2021). *Wrong Side Vehicle Detection* 2021 International Conference on Intelligent Technologies (CONIT),

Howard, A. G., Kalenichenko, M. Z. B. C. D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications*

K, M. B., & Raghunadh, M. V. (2016). *Vehicle Number Plate Detection and Recognition using Bounding Box Method* International Conference on Advanced Communication Control and Computing Technologies (ICACCCT), https://sci-hub.se/10.1109/ICACCCT.2016.7831610

Khan, J. A., & Shah, M. A. (2016). *Car Number Plate Recognition (CNPR) system using multiple template matching* [Journal]. 2016 22nd International Conference on Automation and Computing (ICAC),

Khan, J. A., Shah, M. A., Wahid, A., Khan, M. H., & Shahid, M. B. (2017). *Enhanced Car Number Plate Recognition (ECNPR) system by improving efficiency in preprocessing steps* 2017 International Conference on Communication Technologies (ComTech), sci-hub.se/10.1109/COMTECH.2017.8065766

Lee, S., Son, K., Kim, H., & Park, J. (2017). *Car plate recognition based on CNN using embedded system with GPU* 2017 10th International Conference on Human System Interactions (HSI), sci-hub.se/10.1109/HSI.2017.8005037

Li, H., Wang, P., & Shen, C. (2019). Toward End-to-End Car License Plate Detection and Recognition With Deep Neural Networks. *IEEE Transactions on Intelligent Transportation Systems*,. https://doi.org/10.1109/tits.2018.2847291

Lib, H., Wangb, P., Youa, M., & Shenb, C. (2018). Reading car license plates using deep neural networks. *Image and Vision Computing, Elsevier, Science Direct*, *72*. https://doi.org/10.1016/j.imavis.2018.02.002

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). *SSD: Single Shot MultiBox Detector* European Conference on Computer Vision,

Mandi, M. S., Shibwabo, B., & Raphael, K. M. (2017). An automatic number plate recognition system for car park management. *International Journal of Computer Applications (0975 – 8887)*. https://su-plus.strathmore.edu/bitstream/handle/11071/5488/An%20Automatic%20Number%20Plate%20Recognition%20System%20for%20Car.pdf?sequence=1&isAllowed=y

Mangal, A., Goyal, A., & Kulkarni, A. R. (2020). *Design and Development of IOT Based Prototype Police Barricade* 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS),

Montazzolli, S. e., & Jung, C. (2017). Real-Time Brazilian License Plate Detection and Recognition Using Deep Convolutional Neural Networks. *2017 30th SIBGRAPI Conference on Graphics, Patterns and Images*. https://doi.org/10.1109/SIBGRAPI.2017.14

Nguyen, T.-N., & Nguyen, D.-D. (2018). *A New Convolutional Architecture for Vietnamese Car Plate Recognition* 2018 10THINTERNATIONAL CONFERENCE ON KNOWLEDGE AND SYSTEMS ENGINEERING (KSE), sci-hub.se/10.1109/KSE.2018.8573375

Omran, S. S., & Jarallah, J. A. (2017). *Iraqi Car License Plate Recognition Using OCR* Annual Conference on New Trends in Information & Communications Technology Applications-(NTICT'2017), sci-hub.se/10.1109/NTICT.2017.7976127

Peker, M. (2019). Comparison of Tensorflow Object Detection Networks for Licence Plate Localization. 2019 1st Global Power, Energy and Communication Conference (GPECOM), Nevsehir, Turkey.

Priya, V. L., & Perumal, K. (2014). Detecting the Car Number Plate Using Segmentation. *International Journal of Engineering and Computer Science*. http://www.ijecs.in/index.php/ijecs/article/view/1970

Sasi, A., Sharma, S., & Cheeran, A. N. (2017). *Automatic car number plate recognition* 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), sci-hub.se/10.1109/ICIIECS.2017.8275893

Shariff, A. S. M., Bhatia, R., Kuma, R., & Jha, S. (2021). *Vehicle Number Plate Detection Using Python and Open CV* 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE),

Sharma, P. S., Roy, P. K., Ahmad, N., Ahuja, J., & Kumar, N. (2019). *Localisation of License Plate and Character Recognition Using Haar Cascade* 2019 6th International Conference on Computing for Sustainable Global Development (INDIACom), https://ieeexplore.ieee.org/abstract/document/8991267

Silva, S. e. M., & Jung, C. a. R. (2018). License Plate Detection and Recognition in Unconstrained Scenarios. *Computer Vision Foundation, ECCV*. https://openaccess.thecvf.com/content_ECCV_2018/html/Sergio_Silva_License_Plate_Detection_ECCV_2018_paper.html

Soon, C. K., Lin, K. C., Jeng, C. Y., & Suandi, S. A. (2012). Malaysian Car Number Plate Detection and Recognition System. *Australian Journal of Basic and Applied Sciences, 6(3): 49-59, 2012*. https://pdfs.semanticscholar.org/623b/450ac74911f73b9b1247ead3f9f43038d967.pdf

Srivastava, D., Shaikh, S., & Shah, P. (2021). *Automatic traffic surveillance system Utilizing object detection and image processing* 2021 International Conference on Computer Communication and Informatics (ICCCI),

Stefanović, H., Veselinović, R., Bjelobaba, G., & Savić, A. (2018). *AN ADAPTIVE CAR NUMBER PLATE IMAGE SEGMENTATION USING K-MEANS CLUSTERING* INTERNATIONAL SCIENTIFIC CONFERENCE ON INFORMATION TECHNOLOGY AND DATA RELATED RESEARCH, https://www.researchgate.net/publication/325173480_An_Adaptive_Car_Number_Plate_Image_Segmentation_Using_K-Means_Clustering

Tiwari, B., Sharma, A., Singh, M. G., & Rathi, B. (2016). Automatic Vehicle Number Plate Recognition System using Matlab. *IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)*. https://doi.org/10.9790/2834-1104021016

Varkentin, V., & Schukin, M. (2019). *Development of an Application for Car License Plates Recognition Using Neural Network Technologies.* 2019 International Conference "Quality Management, Transport and Information Security, Information Technologies" (IT&QM&IS). , sci-hub.se/10.1109/ITQMIS.2019.8928373

Viola, P., & Jones, M. (2001). *Rapid object detection using a boosted cascade of simple features* Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, Wang, W., Yang, J., Chen, M., & Wang, P. (2019). A Light CNN for End-to-End Car License Plates Detection and Recognition. https://doi.org/10.1109/access.2019.2956357

Wang, W., Yang, J., Chen, M., & Wang, P. (2019). A Light CNN for End-to-End Car License Plates Detection and Recognition. *IEEE Access*. https://doi.org/10.1109/ACCESS.2019.2956357

Xie, L., Ahmad, T., Jin, L., Liu, Y., & Zhang, S. (2018). A New CNN-Based Method for Multi-Directional Car License Plate Detection. *IEEE Transactions on Intelligent Transportation Systems*, *19*(2). https://doi.org/10.1109/TITS.2017.2784093

Yogheedha, K., Nasir, A. S. A., Jaafar, H., & Mamduh, S. M. (2018). *Automatic Vehicle License Plate Recognition System Based on Image Processing and Template Matching Approach* International Conference on Computational Approach in Smart Systems Design and Applications (ICASSDA). , sci-hub.se/10.1109/ICASSDA.2018.8477639

Younis, A., Shixin, L., Jn, S., & Hai, Z. (2020). Real-Time Object Detection Using Pre-Trained Deep Learning Models MobileNet-SSD. ICCDE 2020: Proceedings of 2020 the 6th International Conference on Computing and Data Engineering,

Yulianto, R. F., Irwansyah, A., & Tamami, N. a. (2021). *Smart Parking System Based on Haar Cascade Classifier and SIFT Method* 2021 International Electronics Symposium (IES),

Zakaria, M. F., & Suandi, S. A. (2010). Malaysian Car Number Plate Detection System Based on Template Matching and Colour Information *(IJCSE) International Journal on Computer Science and Engineering*. https://www.researchgate.net/profile/Shahrel-Azmin-Suandi/publication/49619244_Malaysian_Car_Number_Plate_Detection_System_Based_on_Template_Matching_and_Colour_Information/links/09e4150634ddea2b08000000/Malaysian-Car-Number-Plate-Detection-System-Based-on-Template-Matching-and-Colour-Information.pdf

Zhang, J., Li, Y., Li, T., Xun, L., & Shan, C. (2019). License Plate Localization in Unconstrained Scenes Using a Two-Stage CNN-RNN. *IEEE SENSORS JOURNAL*. https://doi.org/10.1109/JSEN.2019.2900257

Zhang, J., Xu, J., Zhu, L., Zhang, K., Liu, T., Wang, D., & Wang, X. (2020). An improved MobileNet-SSD algorithm for automatic defect detection on vehicle body paint. *Multimedia Tools and Applications* https://doi.org/https://doi.org/10.1007/s11042-020-09152-6

Zhao, Y., Gu, J., Liu, C., Han, S., Gao, Y., & Hu, Q. (2010). *License Plate Location Based on Haar-Like Cascade Classifiers and Edges* 2010 Second WRI Global Congress on Intelligent Systems,

APPENDIX A

For Still images, data was collected from open platform Kaggle.

Data Source: https://www.kaggle.com/tustunkok/license-plate-detection/data

For video analysis, data was collected from the security gate at UM Campus. Short video files were created to check the performance of the models. At later stage, live video feed was also studied from the security gate. This was dealt with help of a webcam mounted on a laptop.