OPTIMIZING AUTOMATION LICENSE PLATE RECOGNITION(ALPR) PERFORMANCE THROUGH ADAPTIVE IMAGE PROCESSING TECHNIQUES

By

YEOH ZHI YING

A Report Submitted in Partially Fulfilment of the Requirement for the Degree of BEng (Hons) Electrical and Electronic Engineering

Faculty of Engineering, Technology & Built Environment

UCSI University

May 2023

ACKNOWLEDGEMENT

I would like to express my grateful appreciation to my supervisor, Dr Mohammad Arif Bin Ilyas from UCSI University, Kuala Lumpur for his excellent guidance and mentorship. Thanks for his endless encouragement throughout these few months, and his wise advises and suggestions to improve the outcome of the study. This dissertation would not been possible to complete without his input and guidance.

DECLARATION OF ORIGINALITY AND EXCLUSIVENESS

I, Yeoh Zhi Ying, hereby declare that my dissertation is the result of my own work, except for quotations and citations, which have been duly acknowledged. I also claim that it has not been previously or concurrently submitted for any other degree at UCSI University or other institutions.

(Yeoh Zhi Ying 1001953809)

Date: 19/03/2024

Supervised by:

(Dr Mohammad Arif Bin Ilyas)

Date: 19/03/2024

ABSTRACT

Automated license Plate Recognition (ALPR) system plays a crucial role in various sectors such as traffic management, parking management, and law enforcement. However, these fields have been faces challenges in detecting and recognizing the license plates under diverse environmental conditions. There are few factors that can lead to difficulties on ALPR performance such as varying light conditions (day, night, sunny, and raining), sight impedance, and camera resolution. This project aims to enhance the ALPR accuracy by implementing adaptive image processing techniques flow. These technique flows are able addressing the issues such as noise reduction, and contrast enhancement to improve the performance on license plate detection and recognition. In this project, there are three image processing technique flow had been designed and developed for analysis. The outcomes from these image processing technique flows had been showed and further analyzed and discussed. The second of the image processing technique flow obtained the highest accuracy compared to the others. This technique had implemented with AI Image Enlargement, ESPCN model and Adaptive Thresholding Gaussian Mean (ATGM). Additionally, the project also integrated a few algorithms and methodologies which focusing on optimizing the ALPR performance. Real-world scenarios dataset provided by industrial partner are utilized to train and evaluate the proposed ALPR model. In this project, Faster R-CNN was utilized and implemented to the proposed ALPR model after having explored researched on the various model such as Convolutional Neural Networks (CNN), Faster Region-Based CNN, and You Only Look Once (YOLO). Moreover, Optical Character Recognition (OCR) algorithms are employed to recognize each character on the detected license plates. Finally, this research contributes to ALPR technology by proposing novel image processing technique flow for improving the ALPR recognition accuracy. By addressing the challenges interrelated with ALPR system, this project aims to enhance the reliability and applicability in real-world scenarios of the proposed ALPR.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
Adam	Adaptive Moment Estimation
ALPR	Automation License Plate Recognition
ATGM	Adaptive Thresholding Gaussian Mean
CCTV	Closed Circuit Television
cls	Classification layer
CNN	Convolutional Neural Network
DL	Deep Learning
FN	False Negative
FP	False Positive
FPS	Frame per Second
HPO	Hyperparameter Optimization
IoU	Intersection over Union
Lcls	Classification Loss
Lreg	Regression Loss
LPR	License Plate Recognition
mAP	mean Average Precision
ML	Machine Learning
OCR	Optical Character Recognition
R-CNN	Region-Based Convolutional Neutral Network
reg	Regression Layer
ReLU	Rectified Linear Unit
RGB	Red, Green, Blue
RPN	Region Proposal Networks
SDG	Sustainable Development Goals
SGD	Stochastic Gradient Descent
TN	True Negative
TP	True Positive
YOLO	You Only Look Once

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND STUDY / OVERVIEW

1.1.1 Automation License Plate Recognition (ALPR) System

In the ever-evolving landscape of security and operational efficiency, Automation License Plate Recognition (ALPR) system had sprung up to become a pivotal and advanced technology. The ALPR system utilise a camera to capture the license plate numbers that come into the system's view. The captured images undergo a transformation alphanumeric character seamlessly using Optical Character Recognition (OCR) then compares the license plates with a pre-existing database for the purpose of authorization and access control. The collected data is uploaded to a central server and can be accessed by authorised personnel or company with the appropriate login credentials.

Beyond its basic security and authentication applications, ALPR system proves invaluable in enhancing efficiency across diverse industry. The real-time ALPR monitoring, recognising the license plate characters, and analyse the license plate data facilitating to optimise traffic flow within manufacturing industry. On the contrary, ALPR system perform security measures in cargo terminal within logistic industry. The progress on real-time detection and recognising each character on license plate from the cargo vehicles, it assists in managing incoming and outcoming vehicles. Thereby, the ALPR system enables efficient data processing and integration in management system. The implementation of APLR system in the logistic industry brings out the logistic management efficiency, indicating as one of the primary advantages. From operational efficiency to bolstering security measures, the ALPR system exemplifies its adaptability, playing a pivotal role in the realms of industrial.

1.1.2 Detection and Recognition in ALPR System

The ALPR system integrates various technologies for the precise detection and recognition. Leveraging complicated image processing techniques and pattern recognition, the system adroitly integrating to identify and distinguish vehicles license plate.

Artificial Intelligence (AI) is a way allowing computers to learn and think critically like human thinking to become an intelligence computer. AI was created by analysing the cognitive process and studying human brain patterns. AI have been developed to several paradigms that including the concepts of Machine Learning (ML) and Deep Learning (DL). An overall view of AI family, as outlined in Figure 1. Machine Learning is a subset of AI that focuses on developing algorithms and models that learn, analyse, and predict and make decisions based on a dataset. On the other hand, Deep Learning, a narrow aspect of Machine Learning, stimulates the way or the pattern of human brain. Deep Learning is an artificial concept that can make prediction by employing the artificial neural network with multiples layers.

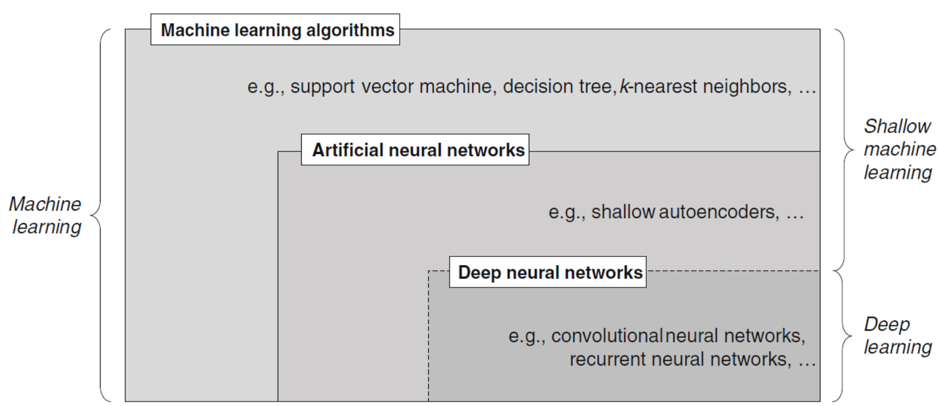


Figure 1.1 The Relationship Between AI, ML, and DL. (Janiesch et al., n.d.)

The optimization on vehicles license plate detection and recognition process is significantly influenced by the exists of AI (Lin et al., 2022), elucidating how AI transform

ALPR into robust and cutting-edge technologies. Specifically, deep learning approaches in the detection and recognition of license plates such as convolutional neural networks (CNNs),(W. Wang et al., 2019), like YOLOv5(Batra et al., 2022; El Ghmary et al., 2023a), Mask R-CNN(Onishchenko et al., 2023), and OCR engines.

Lately, the implementation of AI in ALPR system has demonstrated remarkable efficiency. One study by Kaur et al. performed the performance of ALPR system had achieving an impressive overall accuracy of 90% accompanied by high precision and recall rates for identifying vehicles license plates (Kaur et al., 2023). Another investigation focused on the development of a high-precision ALPR system with not only performed high accuracy in license plate detection but also able identification license plates in 85 milliseconds(Batra et al., 2022). Thereby, these studies have proven the significant role of AI in enhancing the accuracy and speed in identifying vehicles license plates.

The implementation of AI in image processing for ALPR system also can significantly improves the accuracy and speed of identification. Image processing involves the image enhancement to enhance quality and clarity of its image. Thereby, each character from the license plate can be recognised by the AI model with the high precision. The basic image preprocessing techniques includes as image rescaling, noise reduction and brightness adjustment for accurate recognition(Randive et al., 2016). Additionally, AI-based of image enhancement techniques such as edge detection, image sharpening also used to improve the image quality. Hence, the enhancement on vehicle license plates detection and recognition is indubitably to overcome poor illumination, noise and various environmental conditions.

In chapter 2 literature review had details showed some of researchers studied the architecture of deep learning and the performance of ALPR with using deep learning to perform detection and recognition task. There are various of deep learning algorithms such as Convolutional Neural Network (CNN), Deep Neural Network (DNN), You Only Look Once (YOLO) and more. CNN is the most existence used on ALPR system in three parts which are detection, segmentation, and character recognition for real-time processing (Kundrotas et al., 2023).

1.1.3 Challenges in ALPR Performance

In spite of the significant advancements in security and operational effectiveness, ALPR system also faces several challenges when it comes to real-world application that can influence the overall performance. One notably challenge is environmental conditions. Environmental surroundings such as lightning conditions (day, night, sunny and raining), unfavourable condition, sight impedance, providing to the complexity of license plate recognition. The consequences of this challenge can impact the quality of captured images and subsequently affect the recognition accuracy.

Image quality also considered as one of the challenges that impact ALPR performance on detection and recognition. Poor-quality image captured by the camera including license plate captured unclear and noise, or dirty. Consequently, each character on license plate becomes more harder to be identified and extracted by ALPR system which can lead to low precision performance. Additionally, ALPR performance influenced by the quality of cameras used during capturing vehicles license plate. High resolution cameras can capture more pixels of the image, hence improving identification and recognition, and vice versa.

1.1.4 Adaptive Image Processing, Enhancing ALPR Performance

Adaptive image processing is a technique utilize in digital image processing to modify and improve images based on their specific characteristics(Kim Hui et al., 2018). It involves analysing the image and fine-tuning image processing parameters to achieve the desired result. Adaptive image processing concerns examining images in a way considering the surrounding details. Throughout the evaluation assists developing new methods for improving and modifying images, including adjustments to processing parameters.

The term "adaptive" in this research denotes the proposed ALPR model able to adjusts adapt to diverse environmental conditions. This can be achieved through algorithms, image processing techniques and fine-tuned for effective handling of different situations, especially environmental and lighting challenges impacting image quality. This adaptability ensures clear visibility during the detection and recognition process on license plate. Hence, the ALPR system performance is enhanced with better accuracy and faster processing speeds. This versality results not only in improved ALPR system performance, but also suitable for real-time processing applications. Thereby, adaptive image processing is a critical key in field of ALPR system.

1.2 PROBLEM STATEMENT

In the era of digitalization, ALPR system is one of computer vision applications, significantly used in industrial and manufacturing sector like employed logistics facilities like warehouses, distribution facilities, and ports. It plays role on ensuring safety and security by identifying vehicle license plate accurately. With the implementation of AI, the system provides quick and accurate identification and recognition license plates of rear trucks entering and exiting the terminal. However, the efficiency of ALPR can be affected by those challenges such as adverse weather, vehicles' light effect, sight impedance.

One of the primary concerns regarding ALPR system is the absence of a standardized and robust baseline system for consistent and accurate license plate detection and recognition under normal environmental conditions. Despite the widespread implementation of ALPR system, the lack of such a baseline system contributes to variations in performance and reliability across different implementations.

Another significant concern arises from the variability in environmental conditions, particularly during nighttime and adverse weather, becoming a sophisticated challenge to ALPR system. Improving the image quality by applying image processing is significant for precise detection and recognition, particularly in those challenges conditions. Thereby, these issues can be addressed by requiring comparison and experimenting various image processing techniques to overcome challenges posed by adverse weather conditions.

Furthermore, ALPR system as a real-world application is contingent upon their performance in diverse scenarios, including those encountered in industrial settings. For more adaptability in real-world application, it is important using industrial images as it performs real-world scenarios. Moreover, the image processing techniques selected on enhancing image quality is crucial to evaluate its effectiveness in enhancing overall detection and recognition accuracy within ALPR system.

To evaluate better performance on ALPR system, each algorithm will be trained and tested using real-life dataset from industrial which detect on rear truck's license plates in various conditions.

1.3 OBJECTIVES

The objectives of this project are:

1. Develop a baseline ALPR system for standard license plate detection and recognition.

2. Compare image processing techniques to improve license plate image quality, emphasizing low light and adverse weather conditions.

3. Evaluate the selected image processing techniques within the ALPR system, using real-world scenarios and an industrial dataset, to enhance overall detection and recognition accuracy.

1.4 SCOPE OF STUDY

The research aims to develop an AI-based ALPR system that perform license plate detection and recognition seamlessly. The scope of the research includes several realms which are deep learning algorithms, training methods on the AI model, image processing techniques providing clearer image for recognition, and the dataset used during this project. All aspect of this scope is playing crucial role throughout this project to build a robust and advanced ALPR system.

In pursing on improving ALPR system, one of the aspects is focusing on exploring deep learning algorithms, particularly Convolutional Neutral Network (CNN), Faster R-CNN, and YOLO. By choosing suitable AI model on building robust ALPR system, this research aims to deepen understanding these algorithms and their implications within the ALPR system.

Training method also as one of the aspects within the scope of study for building a robust ALPR system. In training methods on AI model requires proficiency in Python programming and utilization of the TensorFlow framework. All the input are images obtained from real-world applications dataset, which annotates with labelling for training AI models. Every model performance is to be collected for evaluation and comparison to ensure the effectiveness of training process.

A clearer image is crucial in process of recognition in this project, as each character from the vehicles license plate can be identified and recognized by ALPR model. Therefore, this research involves comprehensive study on image processing techniques on images with various conditions. Undeniably, the characters on the license plate is hardly seen clearly during a raining day compared to a sunny day.

The dataset is sourced from a cargo terminal, featuring real images captured from the rear trucks provided by industrial collaborator. These images provide well presents the real-world scenarios and challenges that affected ALPR performance. Incorporating these images provides more practicable and valuable in this study.

1.5 GANTT CHART

A Gantt chart has been created for the duration of 28 weeks (8 months) for Final Year Projects A and Final Year Projects B. The overall tasks involved in the ALPR project can be divided and assigned based on quarterly planning. The student intends to complete 50% of the project tasks in Final Year Project A, with the remaining 50% to be completed in Final Year Project B. The Gantt chart in Figure 2 illustrates a detailed distribution of tasks to be completed over a period of 8 months during Final Year Project A and Final Year Project B. There are six phases of the FYP project. Each phase is expected to be completed within the scheduled period. Two presentations are scheduled: one at the end of FYPA for updating on the FYP progress, and the other at the end of the FYPB semester for presenting project completion.

Table 1: The Gannt Chart

	FYP A (MAY2023 - AUG2023)				FYP B (JAN2024 - APRIL2024)			
FYP Project	**MONTH 1**	**MONTH 2**	**MONTH 3**	**MONTH 4**	**MONTH 5**	**MONTH 6**	**MONTH 7**	**MONTH 8**
Phase 1: Research ALPR Systems								
Research and review ALPR systems projects								
Identify project requirement, goals and expected outcome								
Phase 2: Explore Adaptvie Image Processing Techniques								
Explore techniques for enhancing license plate visibility in diverse condition								
Assess adaptive image processing tehcniques								
Phase 3: Build the Baseline ALPR System								
Develop standard license plate detection and recognition model								
Phase 4: Implement Adaptvie Image Processing Techniques								
Implement adaptive techniques into ALPR system								
Conduct initial testing and debugging								
Phase 5: Evaluate Impact on Detection and Recognition								
Use Industrial dataset for evaluation								
Analyse and document the result								
Phase 6: Finalise Project and Documentation								
Documentation project coding and results								
Report writing								
Project Completion								
Deliver presentation								

1.6 SIGNIFICANT OF STUDY

This research project focuses on addressing issue within ALPR system. The primary concern is the absence of standardized baseline for consistent license plate detection and recognition. This results in performance variation among different implementations. Moreover, a pivotal challenge in ALPR performance is in different environments especially adverse weather conditions. The identification of the license plate becomes notably challenging in inclement weather, and the recognition of text characters on the license plate poses an even more difficulty for ALPR system. The research aims to address these issues by optimizing the ALPR performance through adaptive image processing techniques with using real-world scenarios dataset. This approach contributes novel insight to the realm, providing solutions to unexplored areas within ALPR systems.

CHAPTER 2

LITERATURE REVIEW

2.1 WORKING PRINCIPLES OF ALPR SYSTEM

The common working principle of ALPR system went through 4 main stages as proposed in Figure 2.1. The input images were captured from real time background by CCTV camera.

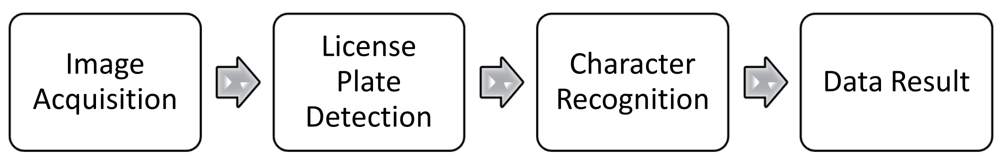


Figure 2.1 The Working Principle of ALPR System

Deep learning algorithm are employed to localize the vehicle license plate from the input images. Throughout process of localizing the license plate, there are features commonly used and obtained by extracting information from the format of the license plate and the alphanumeric characters which comprise the license number (Chang et al., 2004). The output is a sub-image containing only the license plate.

License plate segmentation aims to isolate each character on the license plate for subsequent recognition. Thus, image processing plays vital in this stage. There also various image processing techniques can be employed character segmentation such as siding windows approaches, bilateral filter (Gnanaprakash et al., 2021) as edge preserving, noise reduction filter. After applying those techniques, the extracted segmented characters are sent as an input for character recognition.

Character recognition is crucial stage in ALPR system as it assists in identifying and converting image text into editable text (Redmon et al., 2015). There are few techniques to detect the characters on license plate such as template machine, Optical Character Recognition (OCR). Deep learning techniques such as Convolutional Neural Networks (CNNs), Region-CNN model commonly utilized for character recognition. These models are trained on labelled datasets to learn the patterns and features associated with each character. The output of the ALPR system is the character license plate.

2.2 IMPORTANCE OF DATASET

In pursuing the development an effective ALPR system capable of adapting to diverse environments, it is crucial to collect a large and diverse dataset encompassing various conditions. This extensive dataset is employed in training the detection and recognition models for the ALPR system. Therefore, cameras are an essential component in the initial process of ALPR system.

Serving as the eyes of the ALPR system, these cameras are strategically placed on roads for traffic management purposes, capturing images of vehicles. These images are then collected into a dataset used for training purposes. The detection and recognition models learn the crucial features, such as license plate shape, fonts from this dataset. The goal of this project is to build an ALPR system capable adapts to adverse weather conditions. Thus, it is necessary to assemble the inclusion of vehicles captured in various conditions within the training dataset.

Nevertheless, challenges arise in real-life application, the captured images may not always perfectly clear. For instance, the captured license plates may be in hard to see each character, fonts, and shape of its license plate. Those blurry images lead difficult to detect and recognised correctly by the ALPR model.

Figure 2.2 A Car Captured in Day and Night

In Figure 2.2, a car is shown in both daylight conditions and in night conditions. A clear observation from comparing the two images of identifying license plate. The image captured during night condition is more challenging compared with the image captured during the daylight. In addition, license plate detection is further complicated by the presence of blurred captured images, which can result from factors such as focus problems and motion blur in the LPR camera. Thus, detailed consideration and image processing are essential to ensure the quality of the dataset and effectiveness recognition by ALPR system.

2.3 IMAGE PROCESSING TECHNIQUES

Image processing plays a pivotal role in the realm of ALPR systems as it significantly enhances speed of processing time(Heng Li & Yu Zhang, 2020; Saxena et al., 2016), versality and cost-effectiveness(Mohta et al., 2023).

Through researching on image processing techniques, it came across by Fakhrurroja et al. and Mohta et alemphasise that the image processing hinges upon positive impact on ALPR accuracy improvement. The study utilised YOLO-V4 and YOLO-V8 on license plate detection aid to accurately identify and localise the license plate in images(Fakhrurroja et al., 2023; Mohta et al., 2023).

The image preprocessing techniques included various methods such as grey scaling, binarization, thresholding, and histogram equalization, among others.

2.3.1 Gray Scaling

A greyscale image, shown in Figure 2.3 is characterised by the absence of colours other than shades of grey, ranging from black to white. These algorithms simplify the conversion by manipulating the colour information, resulting in an image that retains only the grey intensity values.Tripathi & Jain, 2021 proposed a pre-processing method by converting the colour image to grayscale then Gaussian filter used to remove the noise. Converting a colour or RGB image to a greyscale image serves several purposes, including reducing file size and improving loading speed(Ganchovska & Krasteva, 2022). Greyscale image, use only a single channel of colour information, whereas colour images use three channels (red, green and blue). Figure 2.4 shows the three channels of RGB image.

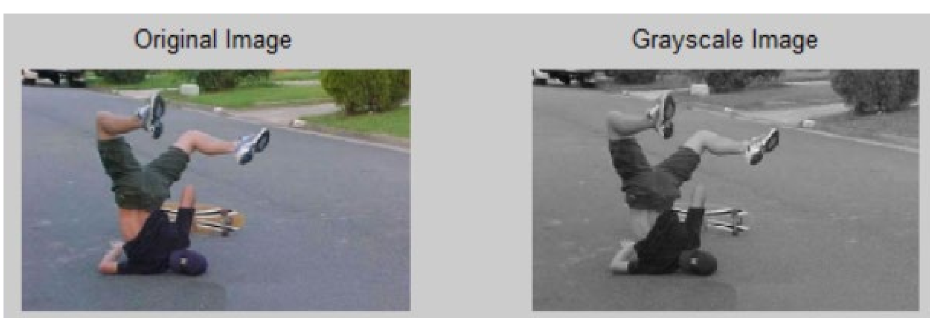


Figure 2.3 Original image and Grayscale Image

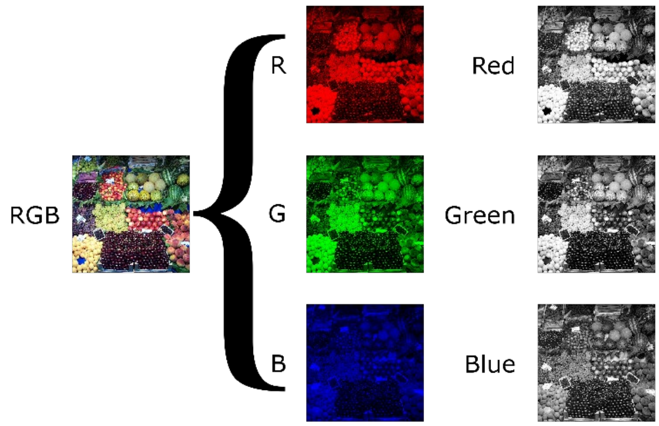
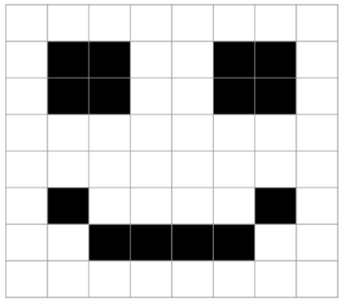


Figure 2.4 Three Channels of RGB Image

Thus, compared with RBG colour images, greyscale images have smaller file sizes, and is more efficient to store and transmit. Besides, the smaller file only requires shorter pre-processing time which the data can be pre-processed more quickly.

2.3.2 Binarization

Binarization is the process of converting a grayscale or colour image into a binary image, where each pixel is either black or white, and is a common technique used in image pre-processing. Figure 2.5 represent a binary image of a smile image. The white pixel is assigned to 0 while black pixel is assigned to 1. The most commonly used binarization method used by an ALPR system is the Otsu method, which is clustering based method. According to Peng et al., 2020 research and analysis on Otsu method, Otsu has a inadequate performance in preprocessing the image when it's in day and night condition. Moreover, Otsu's performance also highly effected by the ambient light.



0	0	0	0	0	0	0	0
0	1	1	0	0	1	1	0
0	1	1	0	0	1	1	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	0
0	0	1	1	1	1	0	0
0	0	0	0	0	0	0	0

Figure 2.5 Binary Image

Generally, image binarization uses grey-scale pixels to obtain a threshold value, and this threshold is used to determine the pixels of the image are considering to background or foreground (Budak et al., 2022). An effective image binarization algorithm improve the efficiency and the recognition performance of an ALPR system.

2.3.3 Adaptive Histogram Equalisation

Adaptive Histogram Equalisation (AHE) is a technique used to adjust the contrast of an image through redistributing pixel intensities. It works by transforming the image so that the output image histogram is approximately flat, meaning that pixel intensities are spread over a wider area. This can help to enhance the information in an image and make it easier to recognise certain features such as license plate. According Zhang Baohua et al., 2010 , the histogram equalization is a beneficial technique in image preprocessing for binarization threshold value selection. It effectively enhances the contrast gradient of the image by redistributing the various grey levels present in the image.

2.3.4 Bilateral Filter

Noise reduction in License Plate Recognition (LPR) image pre-processing improves accuracy by eliminating image noise and distortion, resulting in clearer and more reliable licence plate recognition (Palai & Jena, 2015). With the iterative bilateral filter, the filter can remove the noise and distortion from a grayscale image. Bilateral filter works by applying a weighted average to the pixel values of an image. These weights are determined based on the spatial distance and intensity difference from the central pixels. Equation 1 shows formula of bilateral filter with spatial distance depends on the image values. The bilateral filter starts with a Gaussian smoothing. The Gaussian function is used to calculate the weights, ensuring that pixels in close proximity with similar intensities are assigned higher weights than those with divergent intensity values.

Equation (1):

$$BF[I]_p = \frac{1}{W_p} \sum_{q \epsilon S} G_{\sigma_s}(||p - q||) G_{\sigma_r}(|I_p - I_q|) I_q$$

In Equation 1, a normalised weighted average is presented, where BF is bilateral filter and $[I]_p$ is the amount of filtering of the image. The G_{σ_s} represent a spatial Gaussian weighting that reduces the influence of pixel distance, G_{σ_r} represent the range Gaussian weighting that diminishes the influence of pixels q when their intensity values differ from I_p.

2.3.5 Adaptive Thresholding

Adaptive Thresholding is an image processing technique that can segmentize an image into several foreground and background regions. It involves setting a threshold value that separates the foreground and background regions. A general threshold or global threshold value usually is fixed for all pixels in the image. However, the threshold value in adaptive thresholding is adjusted locally based on the pixel intensity and its surrounding neighbourhood. The results between global thresholding and adaptive thresholding is shown in Figure 2.6. In case, the global thresholding with the uniform threshold value and leads the image is dim as all the pixel intensity is fixed in the image. In contrast, the image after applying adaptive thresholding method is more clearer and contrast. This is because the pixel intensity on the image that applied adaptive thresholding are hinges upon to the local properties of images. Thus, this technique is useful to ALPR due to the accuracy improvement of the license plate detection and character recognition (Fan et al.2019).

Figure 2.6 The Adaptive Thresholding

Gaussian Adaptive Thresholding is a local adaptive thresholding which able adjusts the thresholding to separate the foreground and background. Nevertheless, Gaussian Adaptive Thresholding is suitable for images with varying illumination levels. Illumination levels images refer to the amount of light distributed in an image. This technique suitable for use in ALPR system as it not only can handle varying illumination levels but also can improve image quality. Applying this technique reduces the noise in the image, thereby improving the recognition of the license plate(Golilarz et al., 2019).

2.3.6 Morphological Operators

Morphological operators are one of the image processing techniques developed to address the shape or morphology of image's features. Morphological operators are commonly employed to eliminate the imperfections during segmentation process. There are few examples of morphological operators as outlined in Figure 2.7 such as dilation, erosion, opening, closing and morphological filtering. This technique can be applied to ALPR system to improve accuracy and efficiency on license plate detection and character recognition.

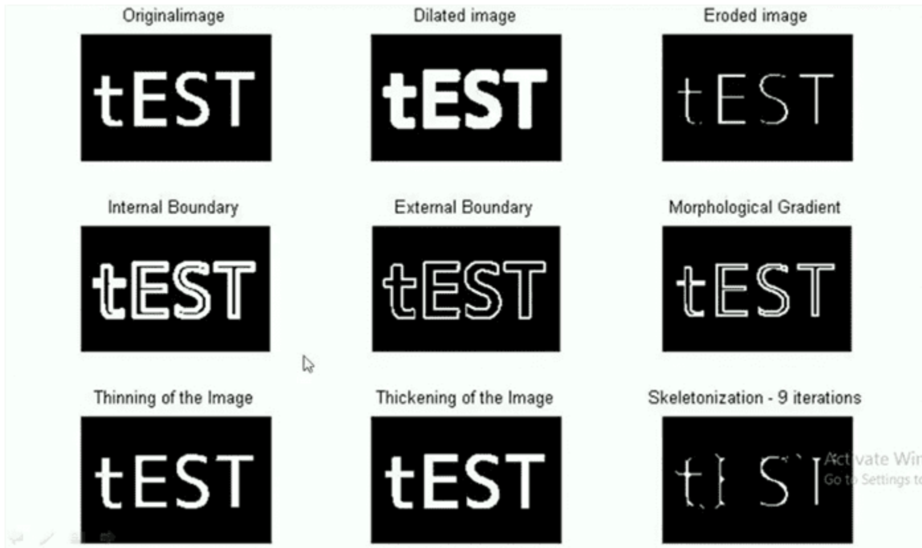


Figure 2.7 The Morphological Operators

A study by Hsrshitta et al. had applied morphological operator such as dilation on the image. The dilation technique can fill the gaps and holes in the license plate region, making it easier to segment and extract the characters from the background(Harshitta & Mr. Shrwan, 2018).

Another study from Parsun et al. had employed morphological operations, particularly erosion and dilation, to shape and resize image features. This process successfully removed connected components and tiny objects. The opening operation is performed on refined those boundaries, while dilation performed on filling up the holes, resulting in the smooth extraction of characters. overall, morphological operators proved its' role in enhancing image quality and extracting relevant components(Pasrun et al., 2020).

2.3.7 Fast Non-Local Mean Denoising

Fast Non-Local Means (Fast NLM) is an algorithm that used for image denoising. Its effective address challenges encountered by traditional denoising filters such as blurring. The Fast NLM algorithm leverages weights depend on Euclidean distance to perform a comparison of the geometric structure to improve the comprehensive analyse the image composition. Euclidean distance is a fundamental geometry and commonly used in image processing. The algorithm is defined at Equation (2), (3) and (4), where $\omega(m, n)$ represents the weights based on Euclidean distance.

Equation (2):

$$NL[\mathrm{l}](\mathrm{m}) = \sum_{N \epsilon l} \omega(m, n)\mathrm{I}(n)$$

Equation (3):

$$NL[\mathrm{l}](\mathrm{m}) = \frac{1}{Z(m)} \sum e^{-\frac{G\sigma(\tau)]]I(M+\tau)-I(n+\tau)]]_2^2}{d^2}}$$

Equation (4):

$$Z(\mathrm{m}) = \sum_n e^{-\frac{G\sigma(\tau)||I(M+\tau)-I(n+\tau)||_2^2}{d^2}}$$

The Fast NLM algorithm as shown in Equation (5) and Equation (6) outlined improvement achieved by modifying the weight calculation, $\omega(m, n)$ from one dimension to two dimensions. Fast NLM algorithm ideally improves time resolution approximately four times compared to the NLM algorithm, proving the Fast NLM algorithm reduce the processing denoising time(B.-G. Kim et al., 2020).

Equation (5):

$$\omega(m, n) = \frac{1}{Z(m)} H_i(I(m+s) - I(m-s))$$

Equation (6):

$$H_i = \sum_{q=0}^{s} e^{-\frac{||I-I(q+\tau)||_2^2}{d^2}}$$

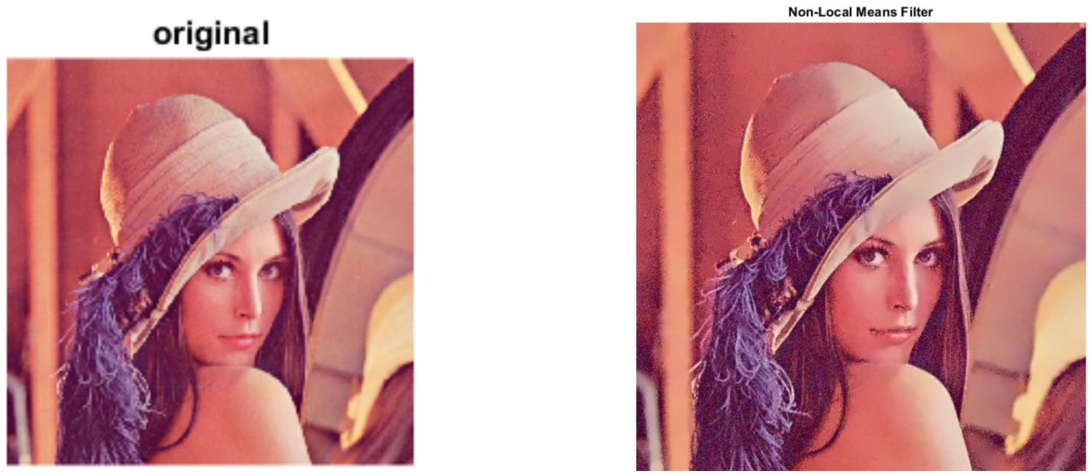


Figure 2.8 Fast NLM Denoising Technique Applied on An Image

With this approach, the Fast NLM algorithm allows to preserve image details while effectively reducing noise. The result detailed in Figure 2.8 proved the Fast NLM denoising performed well on noise reduction and preserving image details. However, there is no study on the application of Fast NLM denosing in ALPR system, the technique has been applied in various fields, such as medical imaging(Bhujle & Vadavadagi, 2018), remote sensing(Zhang et al., 2022), and those studies has been proved to outperform other denoising methods in terms of preserving image details and achieving effective noise reduction.

2.3.8 Image Enlargement

Image enlargement is one of the image processing techniques that focus on transforming the low-resolution image to high-resolution image. Image resolution represents the number of details or pixels of an image, typically measured in pixels per inch (PPI) and dots per inch

(DPI). This process is crucial due to it transforming the details or pixels in the certain region of the image. A high-resolution image consists of more pixels or high pixel intensity, resulting more sharper and refined visual content. Hence, it can be useful in various application such as in recognition characters part of ALPR system.

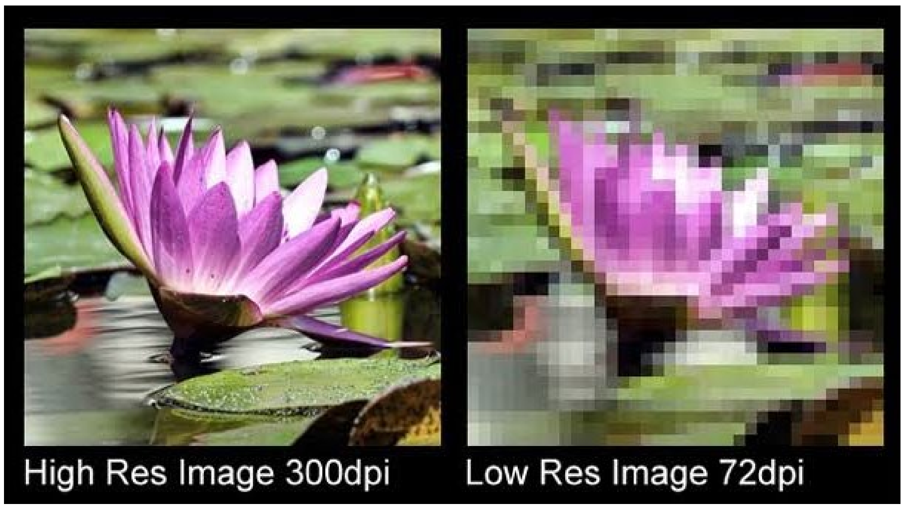


Figure 2.9 Two High- and Low-Resolution Image

Figure 2.9 outlined two same pictures with different resolution. As on the left-hand side presenting high resolution with 300 DPI while the low-resolution image with only 72 DPI. It inevitably that image with 300DPI has refined visual content and contrast compared to 72 DPI.

Furthermore, with the AI-based approach like generative adversarial networks (GANs), Faster R-CNN and CNN, image enlargement is more robust than traditional image enlargement, providing more fine details. Through the implementation of AI algorithms, neural networks nuance the details within the image and facilitating transformation from low-resolution image to higher resolution image.

2.4 OBJECT DETECTION ALGORITHMS

2.4.1 Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNNs) have emerged as a prominent tool in the field of image recognition and classification and are widely used by researchers. The power of CNNs has been harnessed in diverse fields, including human action classification, handwriting recognition, face classification, and a host of other cutting-edge areas. The widespread use of CNNs is due to their ability to effectively capture and extract complex patterns and features from images. The subsequent section focuses on outlining the practical implementation of Convolutional Neural Networks (CNNs) within a specialized system dedicated to license plate recognition. This system is purpose-built to leverage the capabilities of CNNs and applies them in the framework of license plate recognition tasks.

2.4.1.1 CNN Background Theory

The concept of CNNs can traced back to 1980s, Fukushima, 1980 had proposed a Neocognitron, a self-organising neutral network model which is a hierarchical artificial neural network inspired by visual cortex of human brain. It was a significant breakthrough in CNNs research during 1998s. It was the pioneering work of Lecun et al., 1998 that laid the foundation for modern CNNs with the introduction of the backpropagation algorithm and the LeNet-5 architecture for the purpose of handwritten recognition. In 2012, a CNN model called AlexNet, developed by Krizhevsky, made a major impact in the field of computer vision. It won the ImageNet image classification competition with a significant lead of 11% over the second-place entry (Krizhevsky et al., 2012). This achievement drew widespread attention to the potential of Convolutional Neural Networks and sparked increased research interest in the field.

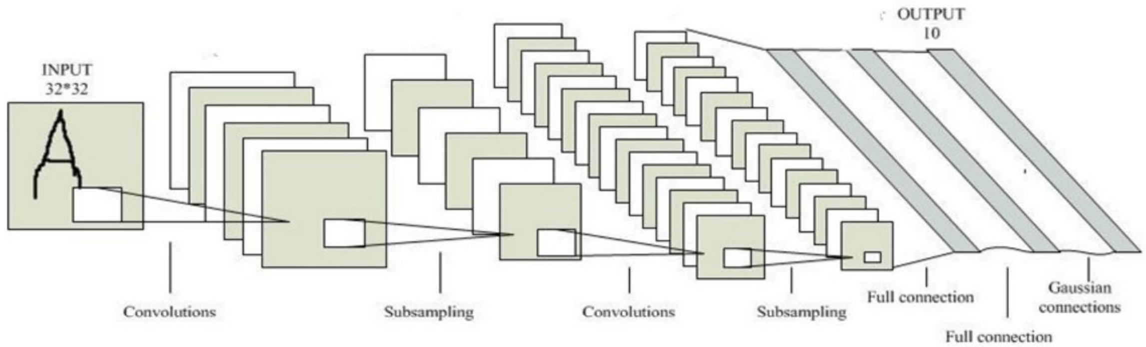


Figure 2.10 The Structure of CNN

A CNNs is primarily made up of input layer, convolutional layer, pooling layer and fully connected layer and output layer. The three main types of neural layers are the basic components of a CNNs shown in Figure 2.10. Each layer serves a specific purpose and contributes to the overall functionality of the network.

The convolution layer is foundation layer of CNNs which purpose to extract relevant features from raw input data and generate 'feature map' as an output. Convolution layer operates by employing a small square matrix, known as kernel or filter. The kernel is utilised to learn the raw input data by detecting the regional patterns and the spatial relationships between adjacent pixels. In Figure 2.11 visualised the process of convolution operation in two -dimensional image. The input image uses size of 5×5. A size of 3×3 kernel or filter is applied to perform the operation. During the convolution operation, the kernel or filter is sliding over the pixels of the raw image's height and width. By sliding through the pixels had perform a dot product between two matrices which are the kernel matrix and the limited region of the receptive field. The result of this operation is known as the convolved feature map or activation map. Then the feature map is generated as an output which is fed into other layers as an input.

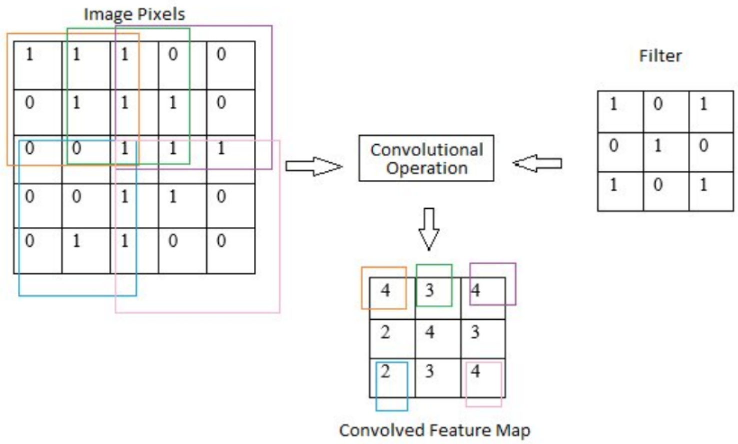


Figure 2 Convolution Operation (Vadlamani & Patel, 2021)

Stride determines the displacement of the convolution filter during operation. In default convolutional operation, the stride value is 1. Larger strides can be used to minimise overlap between receptive fields. However, this will reduce the size of the resulting feature map by skipping potential locations. Hence, padding operation is used to preserve the size of feature map(Vadlamani & Patel, 2021). Padding operation has two operations which are zero padding and valid padding applied to the input raw image during convolutional operation.

The feature map is fed into the Rectified Linear Unit (ReLU) layer. The ReLU which is an activation function. The ReLU defines the positive area of the function arguments, converting all the negative values to zero and remains all the positive values. Equation 7 shows the equation of the ReLU. (Ide & Kurita, 2017). The output is known as rectified feature map.

Equation (7):

$$f(x) = \begin{cases} x, & x > 0 \\ 0, & otherwise \end{cases}$$

Pooling layer, shown in Figure 2.12 is applied after a convolution operation to downsize the feature maps dimensionality. The main objective of pooling is to transform the combined features into a single, valuable representation that retains important information while reducing unnecessary detail.

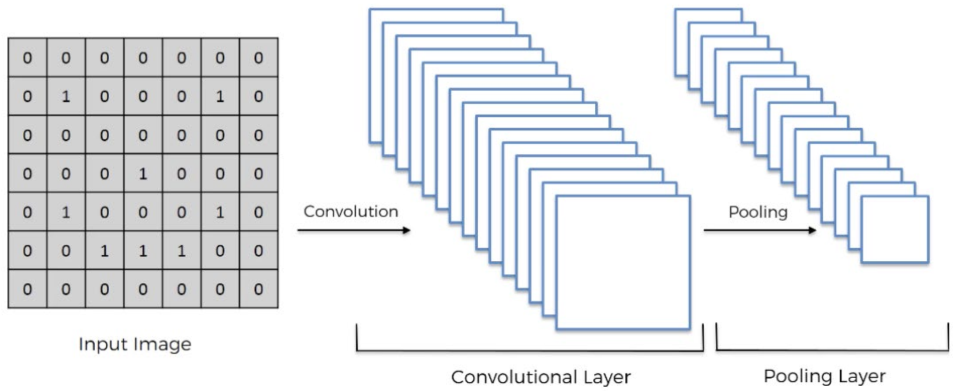


Figure 2.12 The Structure of Pooling Layer

During operation, the pooling layer reduces the resolution of the feature maps over the local neighbourhood of the previous layer by down sampling. There are different types of pooling which are max pooling, mean pooling, and sum pooling. Commonly, CNNs architecture used max pooling which selecting the highest value within a particular region of the feature map defined by the filter. As a result, the output of the max pooling layer consists of a feature map that highlights the most significant features from the previous feature map (Jie & Wanda, 2020). Figure 2.13 visualise the max pooling method.

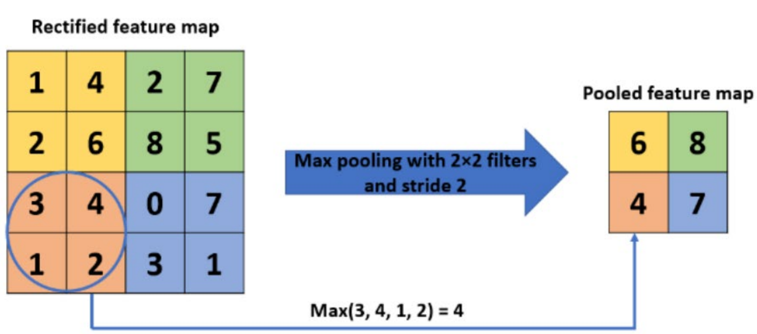


Figure 2.13 The Max Pooling Method (Gholamalinezhad & Khosravi, n.d.)

After pooling layer followed by full connected layer. The fully connected layer requires a 1-Dimensional vector of numeric values as input. Therefore, the rectified feature map, output of pooling layer being flattened into a vector to provide the appropriate input to the fully connected layer. This process involves rearranging the 3-Dimensional volume of numerical values into a 1-Dimensional vector. Flattening, shows in Figure 2.14 which does not involve any complex operations or transformations while it simply organises the data in a linear pattern.

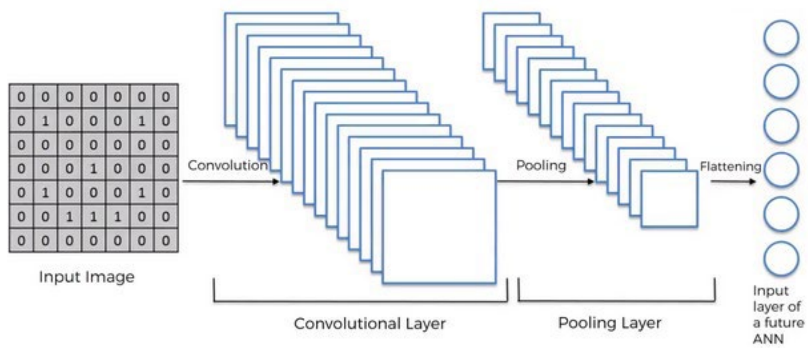


Figure 2.14 Flattening Process

The output of the flattened layer is received as the input for the fully connected layer. The input data is represented as a vector and the weights of the neural network are applied in this layer. The purpose of the fully connected layer is to use these weights to make predictions based on the input image. The SoftMax activation function is employed in this layer to estimate the probabilities of different classes for the output prediction. The SoftMax equation shown in Equation 8.

Equation (8):

$$o_i = \frac{e^{z_i}}{\sum_{i=1}^{M} e^{Z_i}}$$

The $\boldsymbol{o_i}$ is the output of the SoftMax activation function, while $\boldsymbol{z}$ is the output before the SoftMax function and $\boldsymbol{M}$ is the number of the output nodes(Albawi et al., 2018).

2.4.1.2 CNN Proposed Work

Selmi et al. proposed CNN model to classify license plate and non-license plate. The model performs 93.80% precision and 91.30% recall by using Caltech car dataset.

Bulan et al. proposed a CNN architecture model, known as Alexnet to detect and localize license plates. The method employs a strong classifier and a series of preprocessing steps to accurately identify license plate regions in images. The accuracy result using this method is 97% and recall is 86% (Bulan et al., 2017).

Wang et al. proposed a Multi-task CNN (MTCNN) model for license plate detection and license plate recognition. MTCNN model is trained to detect number plates and designed to handle multiple tasks simultaneously. While license plate recognition is introducing at end-to-end method. The proposed method shows that the MTLPR achieves a detection accuracy of 95.80% (W. Wang et al., 2019).

Delmar Kurpiel et al. proposed a license plate detection using CNN model. The input image is resized to 120 x 180 pixels and send to the sub-region of nine layer CNN model to get the confidence score which range 0 to 1. The licnese plate location is detected by combining the output values from sub-regions images. The proposed approach achieved a precision of 87.0% and recall of 83.0% with using images dataset captured by low-cost camera. This research paper had proposed processing time of the model is 0.23s which is 2300ms to detect the license plate (Delmar Kurpiel et al., 2018).

Table 1 shows the comparison of license plate detection precision and recall result of all CNNs related models on various dataset. From the table, the CNNs model produced the highest accuracy in license plate detection proposed by Bulan et al then followed by Wang et al.

Table 2 CNN Summarise Precision and Recall with Different Dataset

No.	Author & Year	Model	Dataset	Precision	Recall
1	Selmi at el.	CNN	Caltech Cars	93.80%	91.30%
2	Bulan at el.	Alexnet, CNN	NR Camera	97.00%	86.00%
3	Wang at el.	MTCNN	CCPD	95.80%	96.90%
4	Delmar Kurpeil et al.	CNN	Camera	87.00%	83.00%

The precision and recall of different CNN-based methods for license plate detection can vary. While CNN-based methods can generally be effective, their performance may be influenced by factors such as the dataset and the specific approach employed. Table 2 shows the comparison of license plate detection results using different methods on Caltech Cars dataset. Caltech Cars dataset is collected from American, and each image centered consist of one vehicle which easy to identify. From the table, the CNNs model produced the highest accuracy in license plate detection proposed by Zhou et al.

Table 3 Caltech Cars Dataset with Various Detection Methods

No.	Author & Year	Model	Dataset	Precision	Recall
1	Selmi at el.	CNN	Caltech Cars	97.00%	86.00%
2	Lim &Tay	MSER +SIFT	Caltech Cars	83.73%	90.47%
3	Le & Li	Hybrid License	Caltech Cars	71.40%	61.60%
4	Zhou et al.	PVM	Caltech Cars	95.50%	84.80%

2.4.2 Faster Region-Based Convolutional Neutral Network (Faster-R-CNN)

2.4.2.1 Background Theory

R-CNN model proposed by Girshick et al., has been crucial to the development of the deep learning-based object detection field. A R-CNN model consists of three main regions which are region proposal then followed by feature extractor and classifier. The R-CNN model had improved the object orientation compared with CNN model.

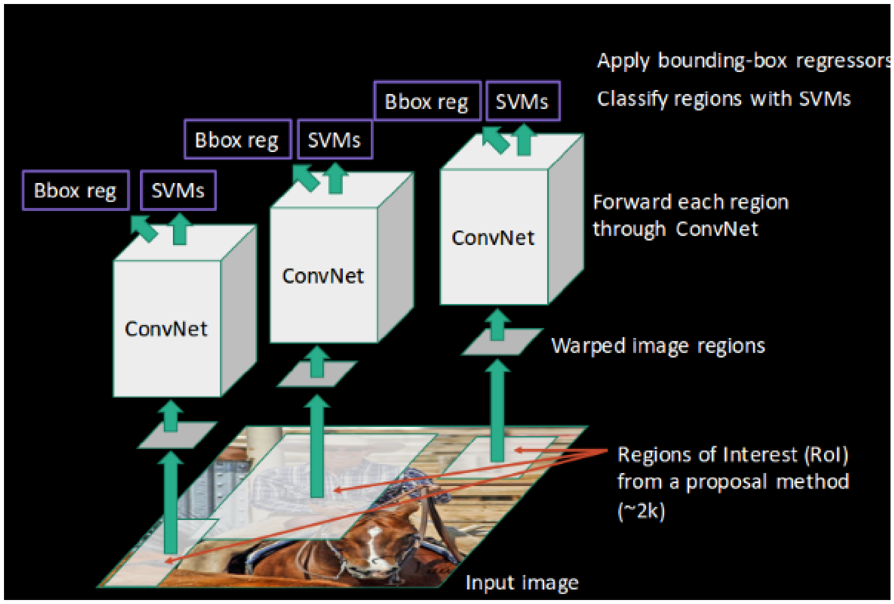


Figure2.15 Architecture of R-CNN Model

Figure 2.15 visualised the R-CNN model structure. In region proposal, the input image extracted around 2000 region proposals with different sized and aspect ratios. Each region proposals are then warped into square shape to ensure consistency. These region proposals are the input to the feature extractor region. In feature extraction region, the warped region proposals are passed through a pre-trained CNN model, playing role as feature extractor, processing the region proposal images, and producing a 4096-dimensional feature vector as an output. The output feature from CNN model is then fed into classifier to detect the presence of the object each region proposal. In classification region, SVM model is used, and the extracted features are used as input to train an SVM model. The SVM model predicts the class label of the object within the region proposal.

In 2015, Girshick et al. had proposed Fast R-CNN to address limitations of R-CNN model. R-CNN model suffered from slow computation speed due to the sequential processing of region proposal. The object detection process in Fast-R-CNN model shown in Figure 2.16. Firstly, an input image is processed through CNN model as refer to backbone network such as Alexnet, ResNet. The CNN model generates a convolutional features map that represent the image's visual features as an input of region proposal. Then, the region proposal identifies the potential object regions within the image. These regions are then cropped and resized by using Region of Interest (RoI) pooling. RoI pooling layer implement max pooling to transform the features within the ROI into a compact feature map. Then, the features vectors passed through the fully connected layers and generates the probabilities for each object class including background class and bounding box coordinates.

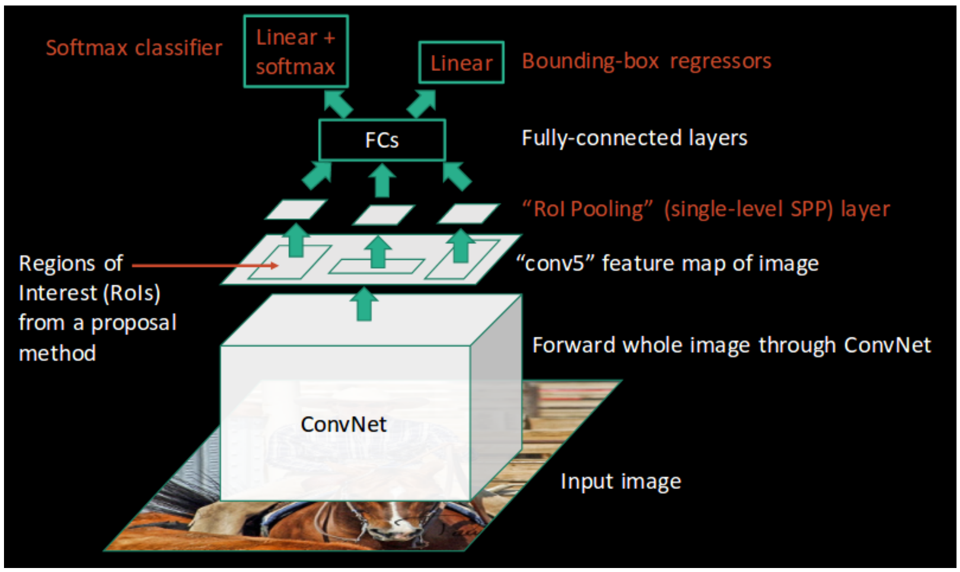


Figure 2.16 Architecture of Fast-R-CNN Model

Faster R-CNN, proposed by Shaoqing Ren et al. in 2015, is an enhanced iteration of the R-CNN object detection model(Ren et al., 2015). R-CNN and Fast R-CNN models in generating the region proposals are depending on the selective search algorithm. However, the Fast R-CNN model was developed to address this limitation by introducing an object detection algorithm that does not require the use of the selective search algorithm can reduce the running time.

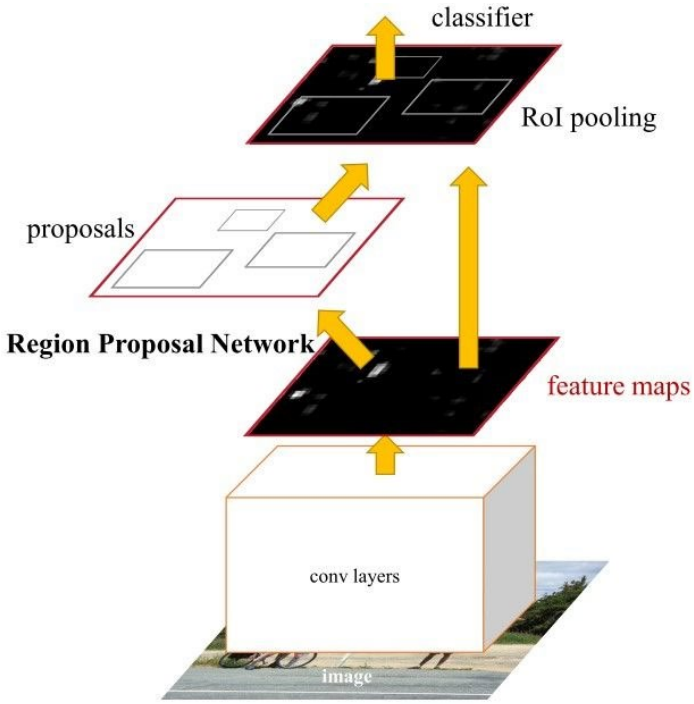


Figure 2.17 The Architecture of Faster R-CNN model

The working process of Faster-R-CNN and Fast-R-CNN almost similar. Figure 2.17 shows the architecture of Faster-R-CNN model, a unified, sole network for object detection consisting of two modules which are Region Proposal Network (RPN) and Fast R-CNN model. The RPN is created generating region proposals while Fast-R-CNN is for detecting objects in proposal regions. Firstly, the input image is passed through backbone CNN model and generates the convolution feature map as output. Then, the RPN predicts the region proposal by using sliding window method over the output convolution feature map and convert to a low-dimensional feature. It feeds this feature into two fully connected layers shown in Figure 2.18 a box regression layer (reg) and a box classification layer (cls). Reg, box regressoion layer returns a 4-Dimensional defining the bounding box of the region while cls, a box classification layer generates the objectness score for each region proposal. The reg layer outputs 4k coordinates for k boxes, while the cls layer outputs 2k estimates the probabilities of presence object. These proposals are parameterized relative to k reference boxes called anchors, which are associated with a particular scale and aspect ratio. The output of the RPN is a set of rectangular of object proposal with detection confidence.

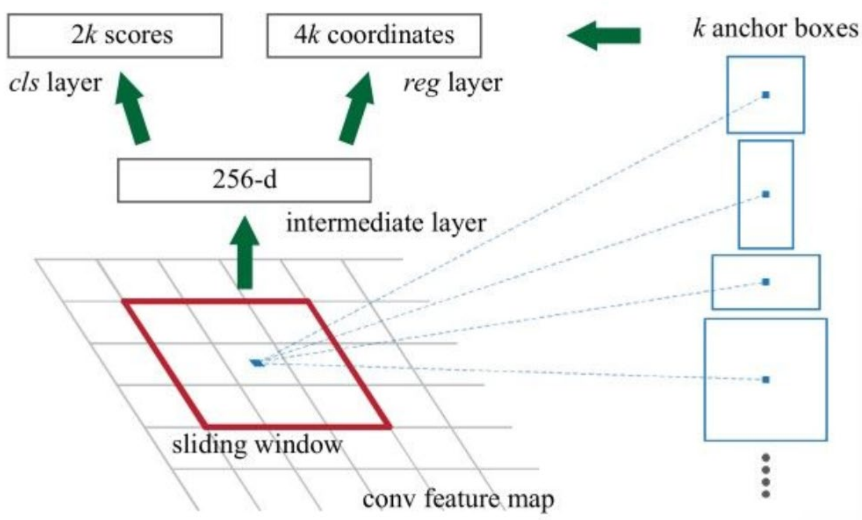


Figure 3 Region Proposal Network

The loss function in Faster R-CNN's Region Proposal Networks (RPN) assists to train the model by comparing the predicted outputs with the correct information. It has two parts which are the classification loss (Lcls) and the regression loss (Lreg). The Lcls part determines the presence of anchor, while the Lreg regression part regress the coordinates of the predicted bounding boxes. The loss function is normalised and weighted to minimize the difference between predictions and ground-truth data, making the model better at generating accurate region proposals. The loss equation for an image provided in Equation 9.

Equation (9):

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}(p_i, p_i^*) + \lambda\frac{1}{N_{reg}}\sum_i p_i^* L_{reg}(t_i, t_i^*)$$

$\boldsymbol{i}$ is the anchor index in a mini-batch and $\boldsymbol{p_i}$ is the predicted probability of the anchor being an anchor. $\boldsymbol{p_i^*}$ is the ground-truth label which 1 is the anchor in positive while 0 is the anchor in negative. $\boldsymbol{t_i}$, a vector representing 4 parameterized coordinates of the predicted bounding box, while the $\boldsymbol{t_i^*}$ represent the ground-truth box associated with positive anchor. The classification $\boldsymbol{L_{cls}}$ is log loss over two class which determine the presence of object. $\boldsymbol{N_{cls}}$ and $\boldsymbol{N_{reg}}$ are the number of anchors.

Table 4 Summarisation of R-CNN, Fast-R-CNN and Faster R-CNN Model

Model	R-CNN	Fast-R-CNN	Faster-R-CNN
Architecture	Region Proposal,	Backbone CNN	Backbone CNN
	feature Extractor	RoI Pooling	RPN
	Classifier	Classifier	Fast-R-CNN
Region Proposal	Selective search algorithm	Selective search algorithm	RPN
Feature Extraction	Pre-trained CNN model	Pre-trained CNN model	Pre-trained CNN model
Object Detection	SVM classifier	SVM classifier	Fast-R-CNN
Training speed	Slow	Faster than R-CNN	Faster than Fast-R-CNN

Table 4 provides a general summary of three R-CNN models used for object detection. The table clearly presents a corresponding study of the R-CNN models in terms of their architecture, region proposal mechanism, feature extraction approach, object detection methodology and training speed. Based on the analysis within three models, Faster R-CNN model will be selected on this project. Its features obviously shown such as training speed is the shortest time. The RPN is faster compared with the slower selective search algorithm. Faster R-CNN is expected to bring out the efficient detection results in this project.

2.4.2.2 Proposed Work

Ravirathinam & Patawari proposed a license plate recognition system using Faster Region Convolution Neural Network with their own dataset. The main aim of their work is to investigate Faster- R-CNN's capability in recognizing the license plate and segmenting the characters on the license plate. Before training, the license plates from the dataset need to be labelled as there has various shape and size in India. In their work, they trained Faster -R-CNN model with two different backbone CNN which are VGG16 and ResNet-50. Over several training examples, this method of labelling and training examples, and the system was able to identify the plates. They also found the optimal number of epochs for training to be training to be 100, using mAP as a measure of error.

In the character segmentation, the Faster R-CNN model was trained on a dataset of 711 images containing both single-line and multi-line number plates. Through experimentation, it was found that the optimal number of epochs for training the model was approximately 20, which was evaluated by measuring the mean Average Precision (mAP) on the test plates.

Their study shows that average accuracy increases with the number of epochs because the model continues to learn and refine its weights and parameters as it is exposed to more training data. As a result, the model produces more accurate information about the training parameters, which improves the overall average accuracy. The 88.4% accuracy achieved from 15 epochs improves significantly as the number of training images increases. With 100 epochs, the accuracy reaches 94.98%. Therefore, increasing the number of epochs will further improve the average accuracy. The result of this work is shown in Table 5 and Table 6.

Table 5 Faster R-CNN on LP Detection Precision

Number of Epochs	Overall mAP
15	88.40%
30	91.70%
50	92.50%
80	93.80%
100	94.98%
120	94.81%

Table 6 Faster R-CNN on Character Segmentation Precision

Number of Epochs	Overall mAP
5	93.80%
10	97.80%
15	98.92%
20	99.55%
25	99.32%

El Ghmary et al. proposed a Faster-R-CNN model in License plate detection. The model had performed well result which are 96.63% of precision and 94.40% of recall.

Amon et al. proposed multi-class detection system using Faster-R-CNN with Inception V2 model can recognize the alphanumeric characters in the license plate images. The model performs with the precision and recall have value of 90.00%.

Brillantes et al. proposed a Faster-R-CNN model and detect with various license plates. The model scored high result, 91.95% of precision and 83.97% of recall with 0.826 of mAP.

Kim et al. proposed Faster R-CNN with Inception V2 model. The proposed approach achieved a precision of 86.00% and recall of 94.00% with using real-time dataset captured by CCTV camera.

Benjdira et al had proposed a Faster R-CNN model with Inception ResNet V2 backbone network. They optimised the model using Stochastic Gradient Descent (SGD) during training process with a momentum value set to 0.89. The learning rate metric are optimised to 0.00019 and the batch size is set to 1. In preprocessing stage, the image is undergoing resizing operation with 600 x 1024 pixels. Overall performance of the proposed model results with 99.66% of precision, 79.40% of recall and 1.39s or 1390ms of processing time (Benjdira et al., 2019).

The precision, recall and mAP of Faster R-CNN models for object detection can vary depending on the selected basic CNN model. Table 7 shows a comparison of the detection results achieved by different Faster R-CNN models. The results show that the Faster R-CNN model proposed by El Ghmary et al. achieved the highest precision and recall rates. Among the four researchers considered, it was observed that the Inception V2 model was most frequently used as the basic CNN model.

Table 7 Faster-R-CNN Summarise with Various Base CNN Model.

No.	Author & Year	Model	Based CNN	Precision	Recall	mAP
1	El Ghmary et al.	Faster-RCNN	Inception V2	96.63%	94.40%	-
2	Amon et al.	Faster-RCNN	Inception V2	90.00%	90.00%	90.00%
3	Brillantes et al.	Faster-RCNN	ResNet-101	91.95%	83.97%	82.60%
4	Kim et al.	Faster-RCNN	Inception V2	86.00%	94.00%	93.40%
5	Benjdira et al.	Faster-RCNN	Inception V2	99.66%	79.40%	-

2.4.3 You Only Look Once (YOLO)

The YOLO (You Only Look Once) model is a deep learning-based object detection algorithm that has gained significant popularity due to its real-time performance and high accuracy. The subsequent section focuses on the application of the YOLO model for license plate detection, exploring its background theory, theoretical foundations, and previous research work conducted by other researchers in this domain.

2.4.3.1 Background Theory

The YOLO (You Only Look Once) model has been a groundbreaking advancement in the field of object detection since its introduction in 2016. The YOLO model revolutionised real-time object detection in single step which eliminated the needs of multistage of region proposal and classification. YOLOv1 was proposed by Redmon et al. in 2016 which introduced a significant approach to the object detection field.(Redmon et al., 2016). The model achieves detection by using a single convolutional neural network (CNN) that concurrently predicts both bounding box coordinates and object class probabilities. Over the years, the YOLO model has several enhanced versions including YOLOv2, YOLO9000, YOLOv3, and YOLOv4 until YOLOv8. Each iteration has introduced new techniques and architectural enhancements to further improve accuracy, speed of detection and the range of object classes that can be detected.

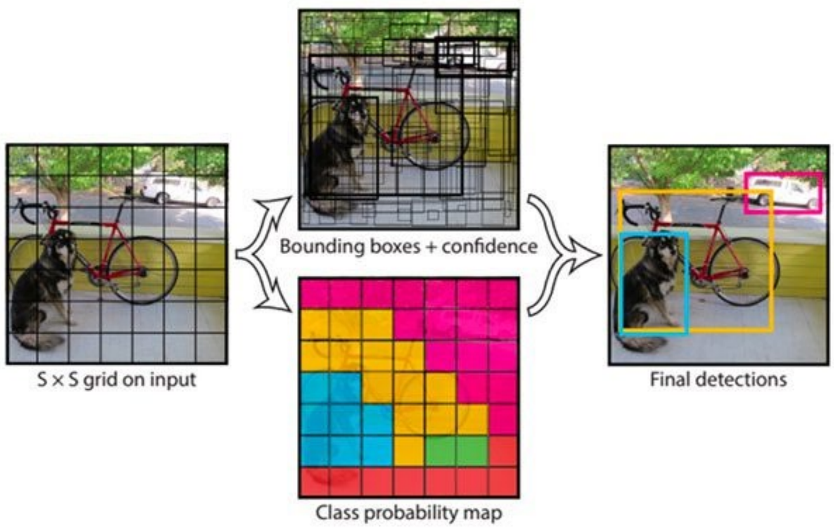


Figure 4 Overview YOLO Model Detection

In Figure 2.19 visualise the YOLO model detection basic operation. YOLO model combines different components of object detection into one single neutral network, which known as unified detection technique. This technique separates the input image into an S x S grid as detecting the centre of an object. Each grid predicts the bounding box of different components and the confidence threshold for those bounding boxes. The confidence score indicates the accuracy of the YOLO model on able to detect the objects in the bounding boxes. The confidence score is performed in Equation 10.

Equation (10):

$$Confidence\ Scores\ of\ box = Pr(Object) * IOU(truth\ pred)$$

IOU (Intersection over Union) is a measure which use to determine the overlapping or matching between two bounding boxes each other. IOU compares the shared area within two bounding boxes to their total area combined by two boxes. When there is no object detected in the grid, the confidence score is 0 and vice versa. Sometimes, the confidence score is needed to be equal range with IOU between predicted box and the ground truth. Ground truth is referring to the actual and correct labelling the components in bounding boxes. The grid cell also predicts the conditional class probabilities, C. The YOLO only predict a set of class probabilities per grid cell without regard to the number of bounding boxes. The Equation 11 shows the multiplication of class conditional probabilities and the individual class-specific confidence score of each box.

Equation (11):

$$Pr(Class_i|Object) * Pr(Object) * IOU\binom{truth}{pred} = Pr(Object) * IOU\binom{truth}{pred}$$

The convolutional neutral network is used in YOLO and evaluate it on detection dataset. The neural network has 24 convolutional layers and 2 full connected layers. The primary convolutional neural network play role as extracting features from the image while following by fully connected layers predicts the output coordinates and probabilities. According to the Redmon et al. research, the performance of fast speed detector on YOLOv1 model is 45 FPS (Frame per Second).

YOLOv2 proposed by same author in same year 2016 had improvements over YOLOv1. According to the research from Redmon et al., YOLOv2 utilised anchor boxes to predict the bounding boxes on an image. In this way, the fully connected layers are removed, allowing more accuracy predictions which are 88% recall and 69.2 mAP. Thought the mAP is little decreased and recall is increased by large margin. YOLOv2 also has high resolution classifier than YOLOv1 resulting 4% increase in mAP.

YOLOv3 an object detection model was proposed in 2018 consists of 106 convolutional neural network layers and performs well in detecting small objects compared to its YOLOv2. The backbone network used by YOLOv3 is Darknet-53 for deeper feature extraction. Hence, YOLOv3 can implement on multiple detection scale on different objects size. The performance of YOLOv3 was improved by using feature pyramid network (FPN), a feature extractor to detect the objects at different scales and combines the fine-grained and high-level features. Yolov3 capable to runs in 22ms providing real-time object detection performance. During the real time detection, the model achieved 28.2 mAP.

YOLOv4 had proposed in 2020 by Bochkovskiy et al. The model consists of CSPDarknet53 as a backbone network, SPP (Spatial Pyramid Pooling) and PAN (Path Aggregation Network) as a part of neck component to enhance the extracted features on various scale. Also, the model incorporates YOLOv3 as head architecture for predicting the bounding box coordinates and class probabilities. The proposed study from xxx shows the YOLOv4 model had achived 65 FPS and score highest average precision compared to YOLOv3 model and others (Bochkovskiy et al., 2020).

YOLOv5 also proposed by Jocher et al. in 2020 just few days after YOLOv4. YOLOv5 has 4 version which are YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. In Figure 2.20 shows the main 4 versions of YOLOv5 comparing with EfficientDet. The 4 versions of YOLOv5 able to detect the COCO dataset in shorter time and higher average precision comparing with EfficientDet. The architecture model of YOLOv5 is same to YOLOv4, however the speed is different. Based on the studied conducted by xxx on detection mold on food surface using YOLOv3, YOLOv4 and YOLOv5. Throughout the performance of YOLOs on recognizing mold on food surfaces, YOLOv5 recall, better precision and had score 100% of average precision compared with YOLOv4 and YOLOv3. (Jubayer et al., 2021)

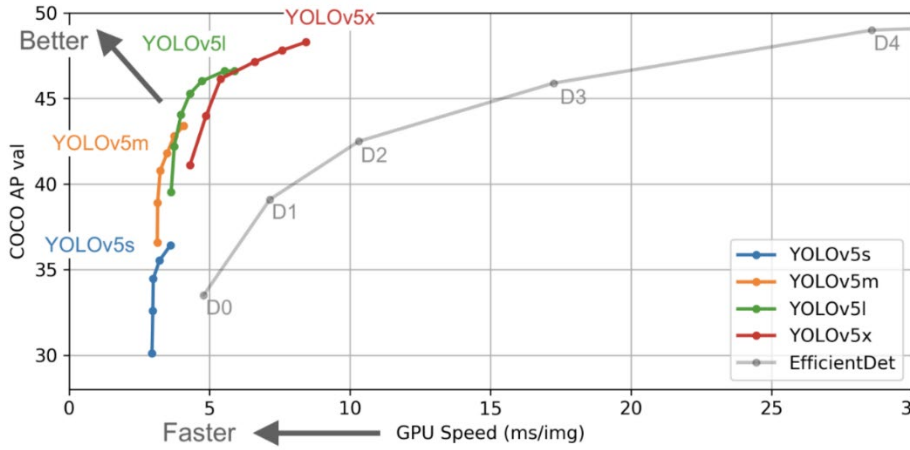


Figure 2.20 A Graph Results YOLOv5 and EfficientDet in Object Detection

YOLOv6 is proposed by Li et al. in 2022. YOLOv6 is designed to push the boundaries of object detection to the next level through the use of latest advances in network design, training strategies, testing techniques and optimization methods. The network design of YOLOv6 consists of RepVGG also known as EfficientRep as a neutral backbone, PAN topology as neck enhanced with CSPStackRep Blocks for the larger model. Besides the head of YOLOv6 simplify decompled head as known as Efficient Decoupled Head to make it more efficient. Decoupled head is splitting two heads and each handle classification and regression tasks to improving average precision. The researcher had evaluated the YOLOv6 model on COCO dataset and achived an AP of 57.2% around 29 FPS on an NVIDIA Tesla T4.

YOLOv7 also proposed in 2022 and same author of YOLOv4. YOLOv7 surpassed all detectors in the range of speed and accuracy at 5 FPS and 160 FPS. YOLOv7 architecture consists the Extended efficient layer aggregation network (ELAN) blocks which combining the features of various groups. YOLOv8 was proposed in January 2023 by Ultralytics, which the company had developed YOLOv5. The model is anchor-free model with decoupled head so the localisation, classification and regression task can be separate independently process. Besides, it also improves overall model accuracy. YOLOv8 had similar backbone of YOLOv5. The neural network backbone of YOLOv8 is CSPDarknet53, a feature extractor and C2f module as neck architecture with doubled head. The AP scored of YOLOv8 on MS

COCO dataset is 53.9% compared with AP scored 50.7% of YOLOv5 (Terven & Cordova-Esparza, 2023).

2.4.3.2 Proposed Work

Henry and Chen had proposed their own YOLO model known as Sliding-Window Single Class Detection (SWSCD) type of YOLO model. The model used 7 layers of convolutional layers for single class detection. The process of detection is sliding window as the method detect each digit of the license plate and each window is then detected by YOLO model architecture. After all the digits are detected with ensemble tiny YOLO model. The proposed system performed 98.22% accuracy, approximately 97.19% of recall and 97.18% of precision and the processing time only 800ms to 1000ms in license plate detection (Hendry & Chen, 2019).

Benjdira et al had proposed a YOLOv3 model for license plate detection. They optimised the YOLO model during training process using Stochastic Gradient Descent (SGD) with a momentum value set. Some parameters are set which are learning rate is set to 0.001 and the weight regularization is set to 0.005 and the batch size is set to 64. The proposed system performed 99.73% of precision, 99.07% of recall and 0.057s of processing time (Benjdira et al., 2019).

Kessentini et al. had proposed a YOLOv2 model for license plate detection. The model able to detect the license plate in various environment. The proposed system performed approximately 98.47% of precision and 100% of recall and 36.7ms of processing time.

Jain et al. had proposed a license plate detection by using YOLOv5 model. The YOLOv5 detect the license plate from various country such as United Kingdom, India, Tunisian and Arabian. The dataset consists of 800 images and separate for 3 parts, training, test and validation. The proposed system had results 95.3% of precision, 98.8% of recall and 98.4% of mAP.

Table 8 visualise a table within YOLOs on license plate detection. The model proposed by Henry&Chen using sliding window method on YOLO model was scored the best results as the processing time is 800ms to 1000ms. Besides, the model proposed by Benjdira et al. had performed well on precision, recall approximately 100% and 0.0057ms of processing time.

Table 8 YOLO Summarise Precision, Recall and Time with Different Dataset

No.	Auther &Year	Model	Precision	Recall	Time
1	Henry &Chen	SWSCDYOLO	97.18%	97.19%	800ms-1000ms
2	Benjdira et al.	YOLOv3	99.73%	99.07%	0.057ms
3	Kessentini et al.	YOLOv2	98.47%	100%	36.7ms
4	Jain et al.	YOLOv5	95.30%	98.80%	-

2.4.4 Comparison and Selection of Detection Models for ALPR System

Throughout the exploration on three detection models, Faster R-CNN model is selected as a detection model employed to ALPR system in this project. Numerous researchers, including Ravirathinam & Patawari, El Ghmary et al., Brillantes et al., Kim et al., and Benjdira et al., have consistently demonstrated the well performance of the Faster R-CNN model in the license plate detection and recognition. Despite the fact that YOLO model may able be the one of three models researched, achieving the highest accuracy and short speed, YOLO model also known for lower precision compared to Faster R-CNN(Reswara et al., 2023). Moreover, traditional CNN does not incorporate with RPNs, limiting their capability handling complex objects and achieve low precision compared to Faster R-CNN.

Faster R-CNN stends to achieve higher accuracy when compared to YOLO and traditional CNN. This fundamental aspect is crucial for ensuring precise and reliable object detection, a key requirement in various applications. A study by Juyal et al. had demonstrated that Faster R-CNN and YOLO V3 on dental caries detection. The results showed that Faster R-CNN had accuracy of 80% while YOLO had accuracy of 75%. The study explained Faster R-CNN implements RPNs to generate region proposals and these allowing it to handle objects with different shapes and sizes more effectively than YOLO(Juyal et al., 2023). This crucial feature ensures that the model can accurately detect or recognize objects regardless of their nuances.

Overall, these studies presenting Faster R-CNN's is consistent high performance, versality on various environment conditions, and effectiveness in license plate detection and recognition. These features position Faster R-CNN is suitable for this project to achieve the project's objectives.

2.5 OPTICAL CHARACTER RECOGNITION (OCR)

Deep learning-based OCR is a system that employed deep learning algorithms to recognise characters from images, letters, or scanned documents. The deep learning-based OCR works by analysing pixel data from source to identify individual characters and transcribe them accurately. Deep learning-based OCR system implements machine learning techniques and complex algorithms to process these images effectively. CNNs particularly commonly used in OCR because CNNs handles identifying patterns in visual data(Sarika et al., 2021). The main three components in CNNs, making it capable of recognizing features patterns and variations in text. The convolution layers for detecting input features, then pooling layers for down sampling data from the output of convolution layers and last fully connected layer perform prediction based on the output of the down sampling data.

Faster R-CNN builds on the CNN backbone, also widely use in image recognition task. A study by B. Wang et al., demonstrated Faster R-CNN as scene text recognition had achieve approximately 90% of accuracy. From those studies proved the Faster R-CNN suitable performs recognition task due to the RPN network able to distinguish all the possible candidates' boxes from the extracted source(B. Wang et al., 2017).

Lee et al. had proposed a real-time ALPR system that implemented Faster R-CNN to perform detection and recognition task under adverse weather environments. The result had achieved accuracy of 99.94% with the average operating speed of 80ms/image (Lee et al., 2016).

Some studies and work on license plate recognition had using various OCR such as PaddleOCR, Tesseract OCR, and EasyOCR to recognise the number and digits from license plate.

Kumar Prajapati et al. had proposed a license plate model with using PaddleOCR to recognise the character from the license plate. PaddleOCR is an open-source deep learning OCR framework developed by PaddlePaddle. PaddleOCR has providing multilingual practice OCR tools and able to analysis and understanding the pattern of the recognise text from an image. Besides, PaddleOCR is capable of support multiple languages and can recognise the text in various orientations, font styles. From the proposed work, the PaddleOCR had performed well with 95% of character recognition. According to Kumar Prajapati et al. stated the 95% using PaddleOCR is better than the traditional OCR as traditional OCR only achieve around 90% (Kumar Prajapati et al., 2023).

Awalgaonkar et al. had proposed ALPR system using OCR on recognition process. EasyOCR can support multiple languages and based on LSTM, CTC and ResNet models for the character recognition. EasyOCR has three main components which are feature extraction using trained model, Sequence labeling and final components is decoding. The unique feature of EasyOCR is reads the letters and digits from the images and returns the coordinates where the license plate located (Awalgaonkar et al., 2021). The EasyOCR recognition can achieve the accuracy around 95.3% based on the proposed project by Kulkarni et al.

On the other side, Tesseract OCR is systematic pipeline processing by cropping frame. At the beginning components, the outline part which nested together to create blobs are identified and stored using connect component analysis. Then, blobs are arranged into text and then splits it into words following with character and region spacing. The adaptive classifier is to accurate the Tesseract OCR recognition process at the end (Smith, 2007). Furthermore, Kulkarni et al. also compared between EasyOCR and Tesseract OCR corresponding on error rate on number and alphabets. The result show the EasyOCR obtained error rate than Tesseract OCR(Kulkarni et al., 2023). Another study by Sham et al. had proposed that Tesseract OCR performing recognition task in ALPR system. The ALPR system demonstrated combining YOLOv4, and Tesseract OCR had achieved 81% in recognition text characters(Sham et al., 2021). However, there is a drawback on Tesseract OCR that its' accuracy lower 14%, when Tesseract OCR works independently based on the study by Paglinawan et al.(Paglinawan et al., 2023)

Table 9 Comparison between Various OCRs

No.	Author & Year	Model	Accuracy
1	B. Wang et al.	Faster R-CNN	90.00%
2	Lee et al.	Faster R-CNN	99.94%
3	Kumar Prajapati et al	PaddleOCR	95.00%
4	Awalgaonkar et al.	EasyOCR	95.30%
5	Sham et al.	TesseractOCR	81.00%

Based on Table 9, Faster R-CNN model had achieved the highest accuracy compared to other OCR models. This performance makes it is the suitable choice to be selected in this project to perform recognition task in ALPR system.

2.6 HYPERPARAMETER OPTIMIZATION (HPO)

This section discusses about the optimization of the pre-training model with several methods such as hyperparameter tuning.

2.6.1 Pre-trained Neural Networks Model

2.6.2 Background Theory

Hyperparameter Tuning is a process involving trying out different combinations of model setting that control the model behaviour. The hyperparameter configuration can be set and use in training model to maximise model predictive accuracy. The ‘hyper’ from the word of hyperparameter stands for ‘top-level’. These ‘top-level’ model parameters influence the learning process and affect the final configuration of the model. Hyperparameter consists of some main components: number of hidden layers, ‘units’ per layer, loss function, optimizer, activation, learning rate, dropout rate, epochs, batch size and early stop patience. In the case of transfer learning is used during training process, the number of frozen layers also one of the components of hyperparameters.

2.6.3 Hyperparameters

The below main components are essential in hyperparameter optimization, as every component selects appropriate values can significantly affects the performance of an AI model.

1. Number of Hidden Layers

The number of hidden layers show the middle the layers in a neural network. It determines and analysis the depth of the model and its capacity to store the complex computational in the data. The complexity of the problem and the amount of data available define the optimal number of hidden layers of the model.

2. 'Units' per Layer

The term "units" represents the number of nodes (neurons) in each hidden layer of a neural network. The number of units in the hidden layer determine how well the nodes learn the pattern and representation with model’s capacity. It is important for choosing an optimal number of units for each layer as it is crucial to achieve a balance between underfitting and overfitting. Underfitting shows the overall model is simple and not capturing much more information while overfitting describe an overall model is too complicated and capturing too much of information.

3. Loss Function

The loss function use to measure the discrepancy between the predicted and actual values in a model. It is important to select an appropriate loss function as it depends on the specific aspect of the problems like sequence generation, regression, or classification. There are several loss function which are Binary cross entropy loss (BSC), Mean Square Error (MSE).

4. Optimizer

The optimizer determines the algorithm used to adjust the model's parameters during the training process, aiming to minimize the loss function. Common optimizers such as Stochastic Gradient Descent (SGD) (Xie et al., 2023), Adaptive Moment Estimation (Adam), Root Mean Square Propagation (RMSprop), and Adaptive Gradient Algorithm (AdaGrad). Each optimizer own characteristics and hyperparameters that can affect training speed and convergence. To achieve a stable model, optimal training speed and higher convergence are considering main factors.

5. Activation function

Activation functions introduce non-linearity to neural networks. It can be describing as open-close valve allowing the neural network to learn complex patterns and make nonlinear transformations. There are most common activation functions consists of sigmoid, ReLU, and SoftMax. It is important to select the correct activation function so the model can perform well.

6. Learning Rate:

An optimal learning rate is important as the learning rate can avoid slow convergence or overshooting. The selected learning rate controls the step size of the updates parameter during training. It determines the model training speed which the model learns the information and converges. Most common learning rate used are 0.1, 0.01 and 0.001. The higher the learning rate the faster the training model convergence.

7. Dropout Rate

Dropout rate is a regularization technique used to prevent overfitting in the neural networks. It randomly sets a fraction of the neuron outputs to zero during training. The dropout rate determines the proportion of neurons to be dropped out, and it assists to enhance the model's generalization capability.

8. Epochs

Epochs represent the number of times or turns the whole dataset is passed through the model during training. More epochs in training process allowing the model to learn more from the data, but excessive epochs could cause to overfitting situation.

9. Batch Size:

Batch size refers to the number of training samples processed before updating the model's parameters. Larger batch sizes can provide faster model convergence, but smaller batch sizes can offer better model generalization. The batch size depends on the available computational resources and the dataset's features.

10. Early Stop Patience:

Early stopping is a technique stops the training process when there no longer improves on model's performance on a validation set. Early Stop Patience shows the number of epochs to wait before stopping the training process. It can prevent overfitting situation of model and store the computational resources.

2.7 PERFORMANCE EVALUATION

Performance evaluation is crucial in license plate recognition systems as it provides insights into the system's capabilities and limitations, aiding in the development of more accurate and robust solutions. Evaluating the performance of a license plate recognition system typically involves assessing various metrics, including Precision, Recall, Confusion Matrix, and others. These metrics allow for deeper understanding of the system's accuracy and help drive improvements in the recognition process.

2.7.1 Confusion Matrix

A confusion matrix provides a tabular representation of the classification performance. The confusion matrix used in most image recognition system to evaluate the performance of the recognition system. Figure 2.21 displays the amount of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) predictions. Each column of the matrix prediction result while each row represents the ground truth of a category. It allows for a detailed analysis of the system's performance, including identifying common errors or misclassifications. The confusion matrix named as it obviously shown the model has confused various objects of different categories.

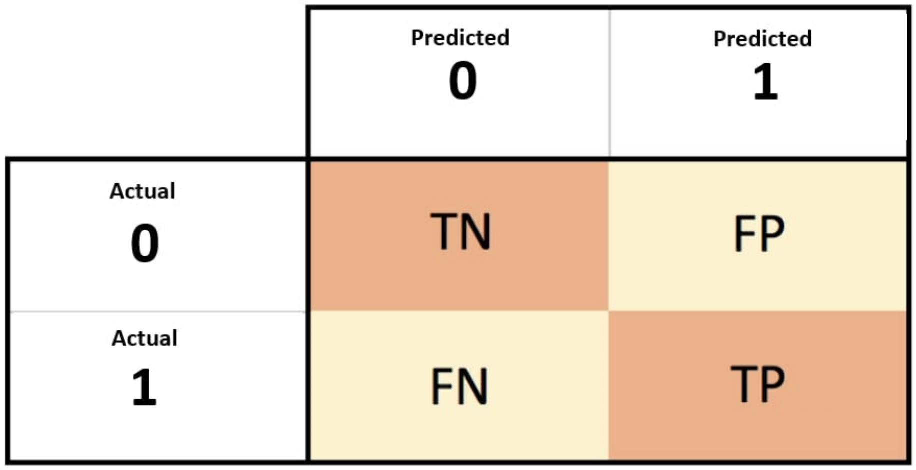


Figure 51 The Confusion Matrix

Besides, accuracy metric also can be calculated from confusion matrix. This metric determines overall proportion of a model performance. Accuracy can be calculated as summing up all the correct predicted on the matrix and divide it by the total of the predictions show in Equation 12.

Equation (12):

$$Accuracy = \frac{TP+TN}{TP+FN+TN+FP}$$

2.7.2 Precision, Recall, and F1-Score

Precision, Recall, FI-Score are derived from the concept of true positive (TP), false positive (FP), and false negative (FN) predictions. Precision determines the ratio of correctly recognized actual objects to the total number of predicted objects. Recall measures the proportion of correctly recognized object to the total number of actual objects. F1-score provides a harmonic mean of precision and recall, achieve a balanced on the model's performance. The precision, recall, and F1-Score of the model's performance results, as in Equation 13,14 and 15:

Equation (13):

$$Precision = \frac{TP}{TP+FP}$$

Equation (14):

$$Recall = \frac{TP}{TP+FN}$$

Equation (15):

$$F1Score = \frac{2 \times precision \times recall}{precision + recall}$$

2.7.3 Processing Time

Processing time mainly focuses on the efficiency and speed of a model. The metric describes the time taken for a process by an algorithm or an engine. Processing time is start to the end of a process and measure in second (s) or in millisecond (ms). It is important to have a shorter processing time which indicates the model has better efficiency can complete the process in a shorter time. Real-time applications utilise the metric to achieve a better system performance. In this project, the proposed neural networks model must be able to detect the license plate shorter 1000ms.

2.8 SUMMARY

In Chapter 2, the literature review provides a comprehensive understanding behind the process of ALPR system. The working principles of ALPR system performs detection and recognition on vehicle license plate, and each component in the process is integral to building a robust ALPR system. In this project, the Faster-R-CNN model has been selected for implementation in performing detection and recognition on input dataset. Also, a dataset with real-world scenarios utilized in this project to ensure the ALPR system is more applicable to diverse environments.

Furthermore, image processing emerged as the main core element of this project for enhancing license plate image quality and characters on license plate. A thorough review of multiple image processing techniques was conducted. This project aims to test and identify suitable techniques that are capable of solving issues related to image quality under challenging conditions. The ALPR system in this project aims to achieve well performance, thus HPO and fine-tuning model's parameters are important.

CHAPTER 3

METHODOLOGY

To ensure alignment with the project's objectives, the methodology will be distributed according to the intended goals. Each objective will be effectively addressed through the distribution of the methodology. This will help stay organise and focus on objectives.

3.1 ALPR SYSTEM PROCESS FLOW AND IMPLEMENTATION

In this project aim was to build an AI-based of identifying rear truck license plate for industrial purpose. The process was separate into several parts training data collection, pre-trained model with transfer learning and image processing and licence plate recognition by training model. A thorough process flow of is outlined in Figure 3.1.

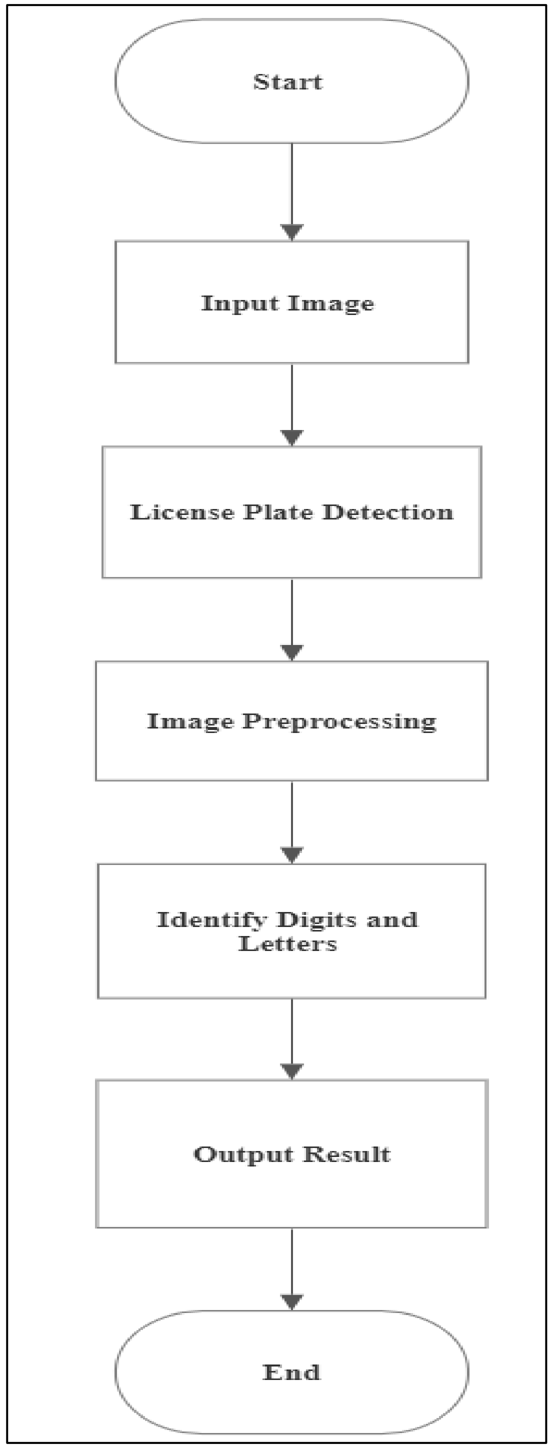


Figure 3.1 Flow Chart of The System

The input images were provided by industrial collaborator. Those images from industrial collaborator not only just as an input image for ALPR system but also as a training dataset for training and testing pre-trained model.

Figure 62 Proposed Model Detect the License Place with Bounding Box.

As shown in Figure 3.2, the process began with input image containing the rear truck with the license plate, which serves as the input to the system. The input image was then detected by the proposed detection neutral network model to localise the license plate of the image and the output of the model detection will be a sub-image of cropped image license plate from the input image as shown in Figure 3.3.

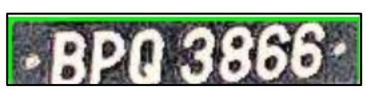


Figure 7 The Output Cropped Image of License Plate.

The OCR engine identified the digits and letters on the license plate. A proposed image processing technique was applied to the license plate sub-image to remove noise, making all the characters more contrasted and it easier to be recognised the characters on license plate.

3.1.1 Input Dataset

The massive input dataset had been provided from an industrial company and was specifically and personalised for used with ALPR system in the industrial sector. The company has generously provided a collection of images, all of which have been captured by cameras at cargo terminal and showing rear-ended trucks with license plates. To ensure effective training and evaluation on both detection and recognition, the dataset split into three parts: 80% for training, 15% for testing and 5% for validation. The dataset consisted of number plate images captured in various environmental conditions, as shown in Figure 3.4, Figure 3.5 and Figure 3.6.

Figure 3.4 The Image Captured During Day.

Figure 3.5 The Image Captured During Night.

Figure 3.6 The License Plate in the Image is Blurred.

Overall, these images with different scenarios are suitable for training purpose for license plate detection and recognition. Hence, the model learned the license plate patterns and able to distinguish the license plate from the background. To form an efficient dataset, it is crucial not only need to prepare the images under different circumstances of image but also need to annotate the images, providing material for model to learn.

3.1.2 Pre-Trained Neutral Networks Model in Detection

In this project, Faster R-CNN model is chosen after standing out from comparing of three models regarding to advantages and disadvantages and their performance on license plate detection and speed accuracy in Chapter 2. The training process involved utilising an image dataset to train the Faster R-CNN model, enabling it to localise the license plates within images effectively. A transfer learning was applied to re-train the pre-trained model with using fine tuning method. Several parameters will carefully be configured during training process, such as optimization, number of epochs, batch size, learning rate and loss function.

3.1.1.2 Training Pre-trained Model

The training pre-trained model process flow outlined in Figure 3.7 was started with selected Faster R-CNN model which has been trained on an image dataset. Then the following step was specific pre-trained model layers were selected for fine-tuning, focusing on license plate detection. Throughout the process, a singular class was designed to represent the license plate after had considering several numbers of classes to be detected. To facilitate the training process, the Faster R-CNN model was then prepared by loading input dataset into the detection model.

Besides, loss function, one of essential components applied during training to measure the difference within the predicted and the ground truth bounding boxes. The loss function selected to determine project are Binary Cross-entropy (BSC) loss which commonly implement in binary classification task. The equation can be performed as Equation 16.

Equation (16):

$$BSC = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot \log\big(p(y_i)\big) + (1 - y_i) \cdot \log\big(1 - p(y_i)\big)$$

The y_i represent the label of the i th of the label which is 0 or 1. 0 represent there is no license plate is detection while 1 is the license plate detected. The $p(y_i)$ represent the probability of 1 while the $1 - p(y_i)$ represent the probability of 0.

To optimize the training stage model, the Adaptive Moment Estimation (Adam) optimizer employ in this project to minimise the loss during training. Adam is the extension of SGD algorithm, uses adaptive learning rates for each parameter, provide it converge more faster than other methods such as AdaGrad, RMSProp, and SGD.

Then, the model continuing fine-tuning for stable convergence to obtain optimal detection model. The performance model evaluation is crucial to assess the accuracy and targeted speed of the license plate model.

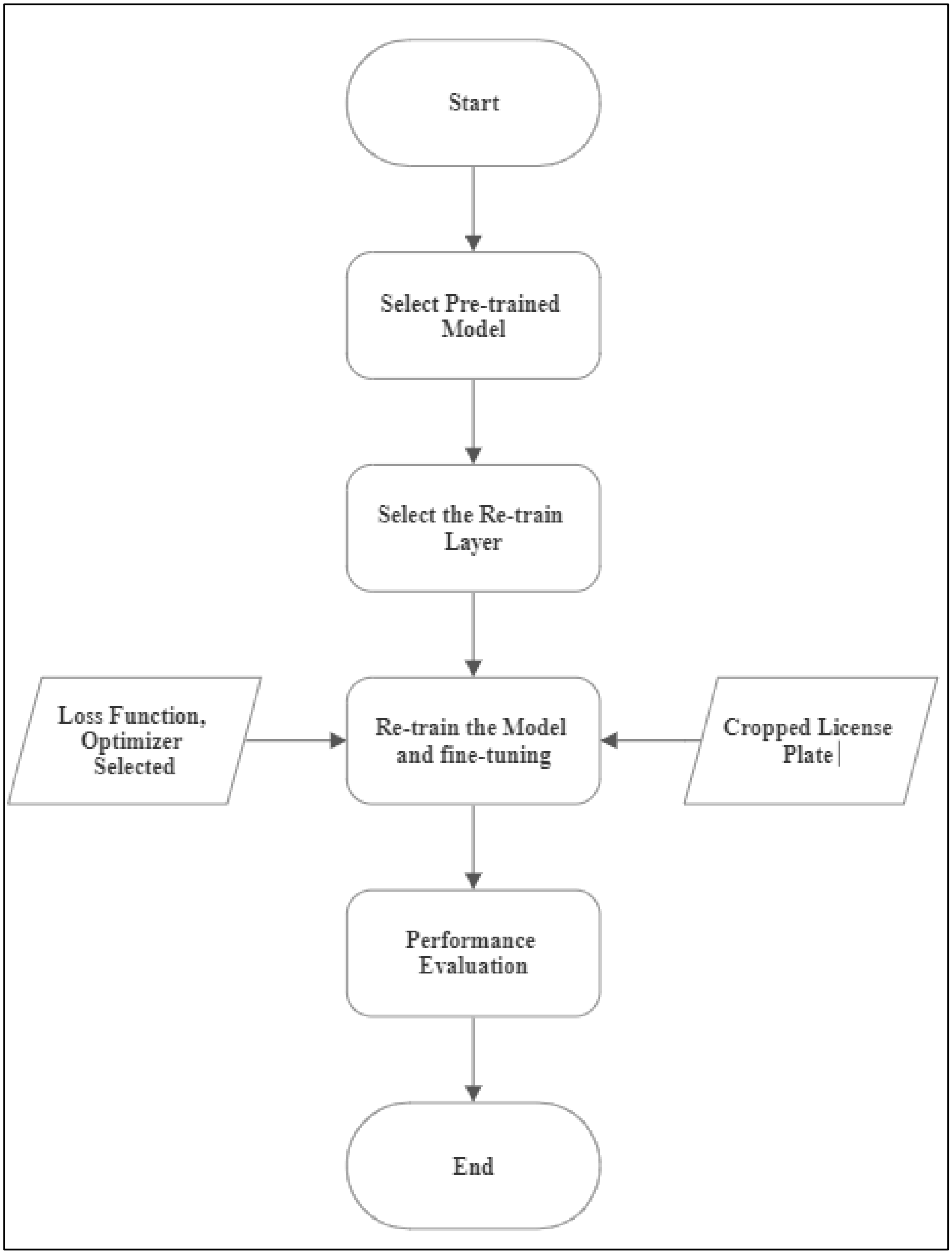


Figure 3.7 The Flow Chart of Training a Pre-trained Model.

3.1.3 Optical Character Recognition (OCR)

The OCR engine was applied on the licence plate recognition stage. As shown in Figure 3.8, the input was the preprocessing sub-image of the license plate. The sub-image was recognised by OCR engine and read into editable format. The recognised of the characters on license plate is the output result of the ALPR system. In this recognition steps, Faster R-CNN model had been selected as OCR engine to recognize characters accurately. The evaluations of Faster R-CNN demonstrated its highest accuracy when compared to other popular OCR engines, such as Tesseract OCR, and Easy OCR. It is easy to integrate into the proposed ALPR system as it built on the top of CNN model. After the license plate is being localised by neural networks model, each character was recognised by the Faster R-CNN model. The OCR engine processes the input and recognises the characters accordingly.

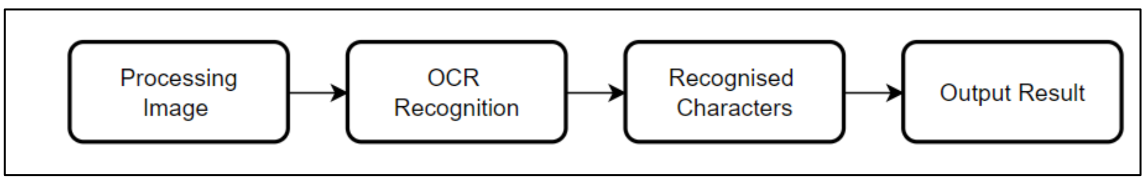


Figure 8 The OCR Engine Process.

3.1.2.1 Input Dataset

The input dataset for recognition is derived from the dataset used in the detection phase. In recognition, the task is focused on identifying the characters present on the license plate. Therefore, the training dataset for this purpose comprises sub-images containing the license plates of various vehicles. To ensure effective training and evaluation, the dataset split into three parts: 80% for training, 15% for testing and 5% for validation. The dataset consisted of number plate images after applying various image processing techniques, as illustrated in Figure 3.9.

(a) (b) (c)

Figure 9 The Sub-Image After Applying Image Processing Techniques

3.1.2.2 Training Pre-trained Model

Faster R-CNN model also chosen for playing role OCR in ALPR system. The training process involved utilising an image dataset to train the Faster R-CNN model, enabling it to localise the license plates within images effectively. A transfer learning was applied to re-train the pre-trained model with using fine tuning method. Several parameters will carefully be configured during training process, such as optimization, number of epochs, batch size, learning rate and loss function.

In this recognition steps, Faster R-CNN model had been selected as OCR engine to recognize characters accurately. The evaluation of Faster R-CNN demonstrated its highest accuracy when compared to other popular OCR engines, such as Tesseract OCR, and Easy OCR. It is easy to integrate into the proposed ALPR system as it built on the top of CNN model.

3.2 COMPARISON BETWEEN IMAGE PROCESSING TECHNIQUES FLOW

The optimization of the ALPR system had explored through a comparative analysis of various image processing techniques. The aim was to enhance the system's performance by designing an image processing process flow applying on the sub-image (consist only license plate). The process involved trial and error, where multiple image processing flows were experimented with to improve contrast and sharpness.

3.2.1 Fundamental Image Processing Steps for OCR Optimization

The image processing method flow involved several steps of image processing techniques, enabling the characters easier to be recognised by OCR model. There are some important image techniques need to apply on the images as a fundamental to improve the image quality. Denoised Converting RGB image to grayscale image, aiding to preserve tiny detailed. Grayscale image not only reducing the complexity on various channels, but also enhance contrast. Despite these improvements, the grayscale images still struggle to distinguish subtle details. To address this, sharpening techniques, adaptive thresholding, and noise reduction techniques are applied to the grayscale image.

3.2.2 Proposed Image Processing Techniques Flow

Designing the most effective and suitable image processing method flow for ALPR, there are three proposed image processing techniques flows were tested on images and subsequently subjected to OCR. The results of each flow were then compared and analysed. This systematic approach enables to identify and select the optimal image processing flow among those three flows to be integrated into the proposed ALPR system.

3.2.2.1 Method 1: Image Processing Technique with Basic Thresholding, Morphological Operations

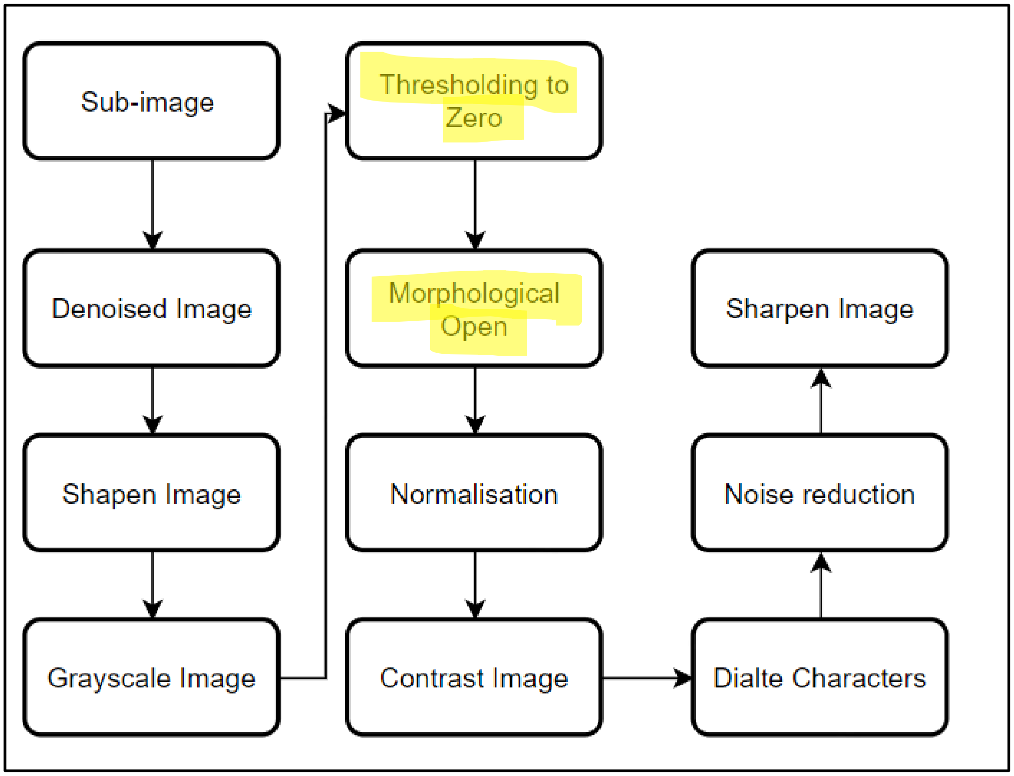


Figure 10 The First Proposed Image Processing Technique Flow

In Figure 3.10 outlined the first method of image processing technique flow for applying to ALPR system. Removing noise and sharpening the characters as fundamental steps before converting the sub-image into grayscale to increase contrast.

To enable characters stands out from the background, the grayscale image then converts into black background by applying Thresholding to Zero, as shown in Equation (17). Thresholding to zero is one of type of the thresholding in image processing as it set a threshold as a cutoff point. If the pixels are higher than the threshold value it remains while the pixels go to zero if they are lower than the threshold value.

Equation (17):

$$dst = \begin{cases} src(x,y) & if\ src(x,y) > threshold \\ 0 & otherwise \end{cases}$$

Through this process, the background of license plate ideally converts to black. The morphological open approached after the thresholding for removing tiny object from the foreground while remaining the large structural characters. The process involves operation which are erosion then dilation. Erosion operates by eroding the boundaries of the foreground object then dilation dilates the image by filling up the gaps. This operation had removed unwanted noise while preserving overall details on the image.

Normalisation technique applied to the image for ensuring consistent pixel intensity ranges. After the pixel consistent across the image, the contrast is applied. There is undeniable that the presence of tiny details still presents on the image, hence noise reduction takes place for removing noise. The bilateral filter for preserving the edge details and Fast Non-Local Mean Denoising were played role in noise reduction. By combining those techniques, the presence of noise can be removed well at the same time also preserve the overall characters are clear to see and recognised. The output of the image is visually improved on edge and shape of characters after applying sharpening techniques.

3.2.2.2 Method 2: Image Processing Technique with Image Enlargement and ATGM

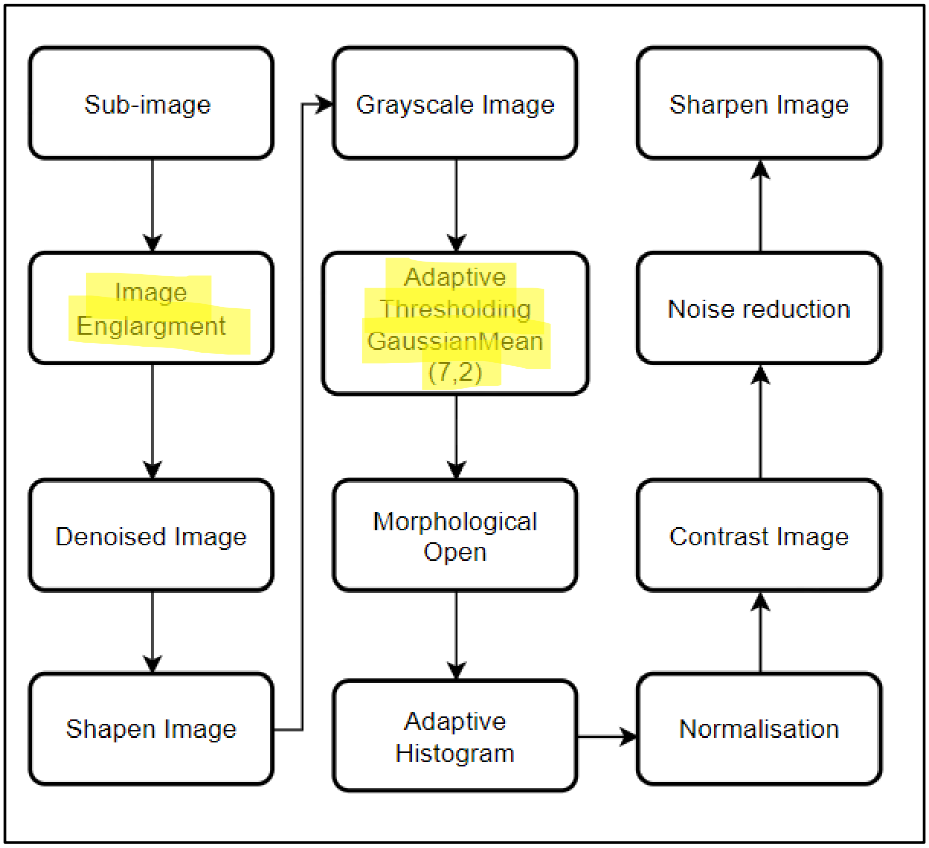


Figure 11 The Second Proposed Image Processing Technique Flow

In Figure 3.11 outlined the second method of image processing technique flow for applying to ALPR system. Image enlargement was applied on the input image by using ESPCN model. ESPCN also known as Enhanced Sub-Pixel Convolutional Network is an AI model, designing to enlarge the image from low resolution input image to high resolution image. The process involves passing the input image through convolutional neural networks and utilising the learn weighted to perform super resolution. Removing noise and sharpening the characters as fundamental steps before converting the sub-image into grayscale to increase contrast.

In second method, image processing flow, the Thresholding to Zero was replaced by Adaptive Thresholding Gaussian Mean. It can deal with uneven lighting in image by dynamically adjusting the threshold based on the local mean intensity values. The function call of the Adaptive Thresholding Gaussian Mean is shown in below. The thresholding value is a Gaussian weighted sum of the neighbourhood value, maxValue with minus the constant, C. With the block size, B, the size of the pixel neighbourhood to calculate the threshold value. The second method applied block size set to 7 and constant set to 2. In this condition, each pixel's local mean is calculated based on 7 x 7 neighbourhood, and this mean is subtracted by the constant value to determine the adaptive threshold.

$$adaptiveThresholding(src, dst, maxValue, adaptivemethod, B, C)$$

With this adaptive approach can effectively address issue related to uneven lighting, enhance image quality, and improve the visibility of license plate characters in images. Consequently, license plates captured under nighttime conditions can be recognized well.

The morphological open approached to eliminate noise and tiny unwanted details by utilising erosion and dilation operation. To make the characters more notably from the background. The subsequent Adaptive Histogram Equalisation (AHE) was applied to the image for enhancing the poor illumination image. The adaptive histogram works by dividing an image into smaller regions and applying histogram equalization independently to each region.

The subsequent steps are same those of the first method of image processing flow. Normalisation technique applied to the image for ensuring consistent pixel intensity ranges. The bilateral filter and Fast Non-Local Mean Denoising were played role in noise reduction. By combining those techniques, the presence of noise can be removed well at the same time also preserve the overall characters are clear to see and recognised.   The output of the image is visually improved on edge and shape of characters after applying sharpening techniques.

3.2.2.3 Third Method: Image Processing Technique with Larger Parameter on ATGM

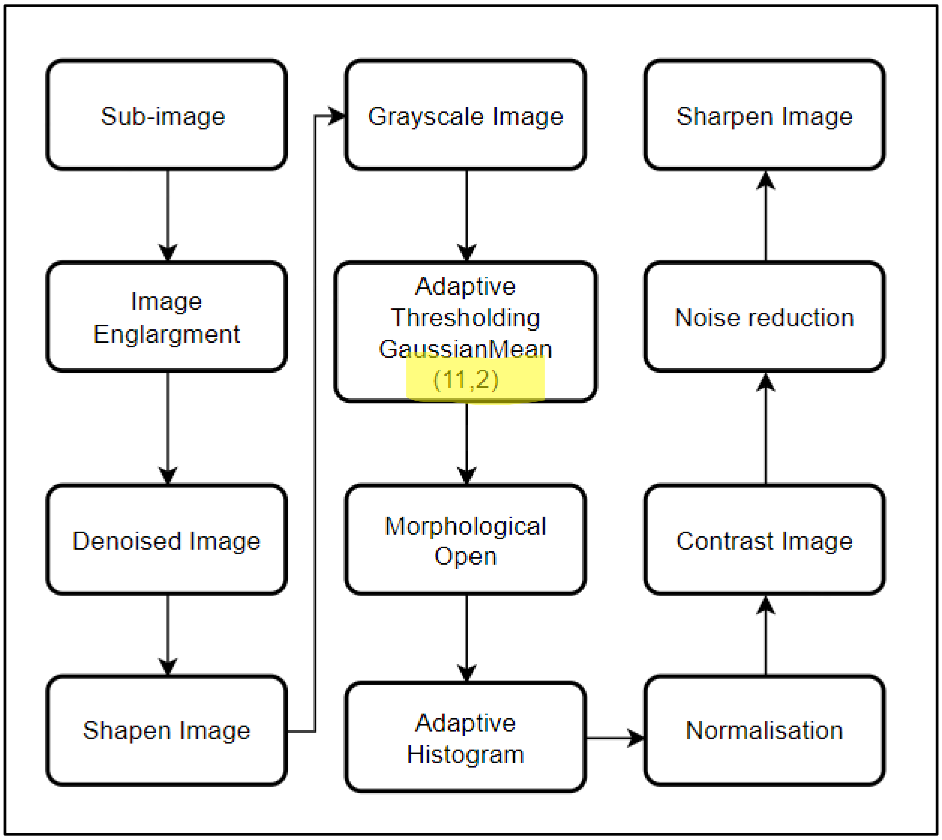


Figure 3.12 The Third Proposed Image Processing Technique Flow

In Figure 3.12 had outlined the third method of image processing technique flow for applying to ALPR system. The method closely resembled the second method, with the only different being the use of a Block Size, B is set to 11. Implementing the larger Block Size, each pixel's local mean is calculated based on 11 x 11 neighbourhood, and this mean is subtracted by the constant value to determine the adaptive threshold. As the license plate input image contains white characters on a black background, the larger block size covers a more extensive region, potentially providing a more contrasted image compared to the second method of the image processing flow.

3.3 PERFORMANCE EVALUATION

3.3.1 Performance Evaluation on Pre-trained Model

The training model performance can be analysed and concluded in license plate detection and recognition. In this project, Confusion Matrix, as shown in Table 10 is used to evaluate the performance of detection system. Each column of matrix represents the predicted class of license plate, and each row of matrix represents the actual class of license plate. In the confusion matrix, it split into four categories:

1. True Positive (TP): The pretrained model detect correctly in the presence of license plate image.
2. True Negative (TN): The pretrained model correctly identifies no presence of license plate on the image.
3. False Positive (FP): The pretrained model detect wrongly in the presence of license plate on the image.
4. False Negative (FN): The pretrained model wrongly identifies or fails to detect the presence of license plate.

From the confusion matrix of each four categories, the accuracy of the model can be calculated.

Table 10 Confusion Matrix for Model Performance

Predicted \ Actual	License Plate	No License Plate
License Plate	True Positive (TP)	False Positive (FP)
No License Plate	False Negative (FN)	True Negative (TN)

The overall ALPR system can be evaluated by calculating precision, recall, F1-Score based on the four categories in confusion matrix and referring the equations had been discussed in Chapter 2. The objective of this project to determine whether the proposed ALPR system can detect the license plate accurately in shorter time. Thus, the time taken for pre-trained model on license plate detection also chosen as model performance in this project. The time taken of model can determine the overall speed and responsiveness of the ALPR system.

3.3.2 Performance Evaluation on Image Processing Technique

The evaluation of image processing techniques, each image undergoes to the model recognition. The evaluation of each character is based on their recognition accuracy and OCR time processing. A proficient image processing technique is essential as it enhances the characters visibility, making them distinguish between from the background, easier recognition by the model. The importance of achieving a higher accuracy percentage is paramount in this evaluation, indicating the system's effectiveness in accurately detecting and recognizing characters across different scenarios.

3.4 SUMMARY

In Chapter 3, the methodology section had proposed the comprehensive ALPR system. This project had proposed the detection and recognition with applying same pre-trained model, Faster R-CNN model. The process of training the pre-trained model on detection and recognition is detailed, including the selection of the loss function and optimizer. The training process of detection and recognition was similar, but the difference lies on the input dataset.

Implementation of image processing technique on ALPR system has been remarkable aspect of this project. These techniques provide the characters stand out on the license plate, enhancing OCR recognition accuracy. Three image processing technique flows were proposed to solve the visuality characters on license plate in various environment conditions. The selection of the optimal image processing technique flow will be determined after a thorough comparison among the proposed flows.

CHAPTER 4

RESULT AND DISCUSSION

In this chapter presents the outcomes of ALPR system, detailing the model's training, validation, real-life accuracy, and a comparative analysis with image processing technique flows. The project assessment involved testing the ALPR system on a dataset comprising license plate images and evaluated its performance in terms of accuracy and speed.

4.1 OVERALL LICENSE PLATE IMAGE DETECTION ALGORITHM DESIGN

Figure 4.1 shows the overall of license plate detection flow. The first step would be the user clicking on the detection application and the detection application would be activated and ready to detect the image. Then the second step would be the user clicking folder with the name of the test_images. Inside this folder, there are few folders which are 'backup', 'input', 'lpd' and 'output'. The third step would be the user clicks the input and can upload the input image and wait for few seconds. Meanwhile, the 'input' folder passes the input image to the detection model. The fourth step would be the detection model receives the input image and performs license plate detection on the image and print the result to the 'output' folder, as illustrated in Figure 4.2.

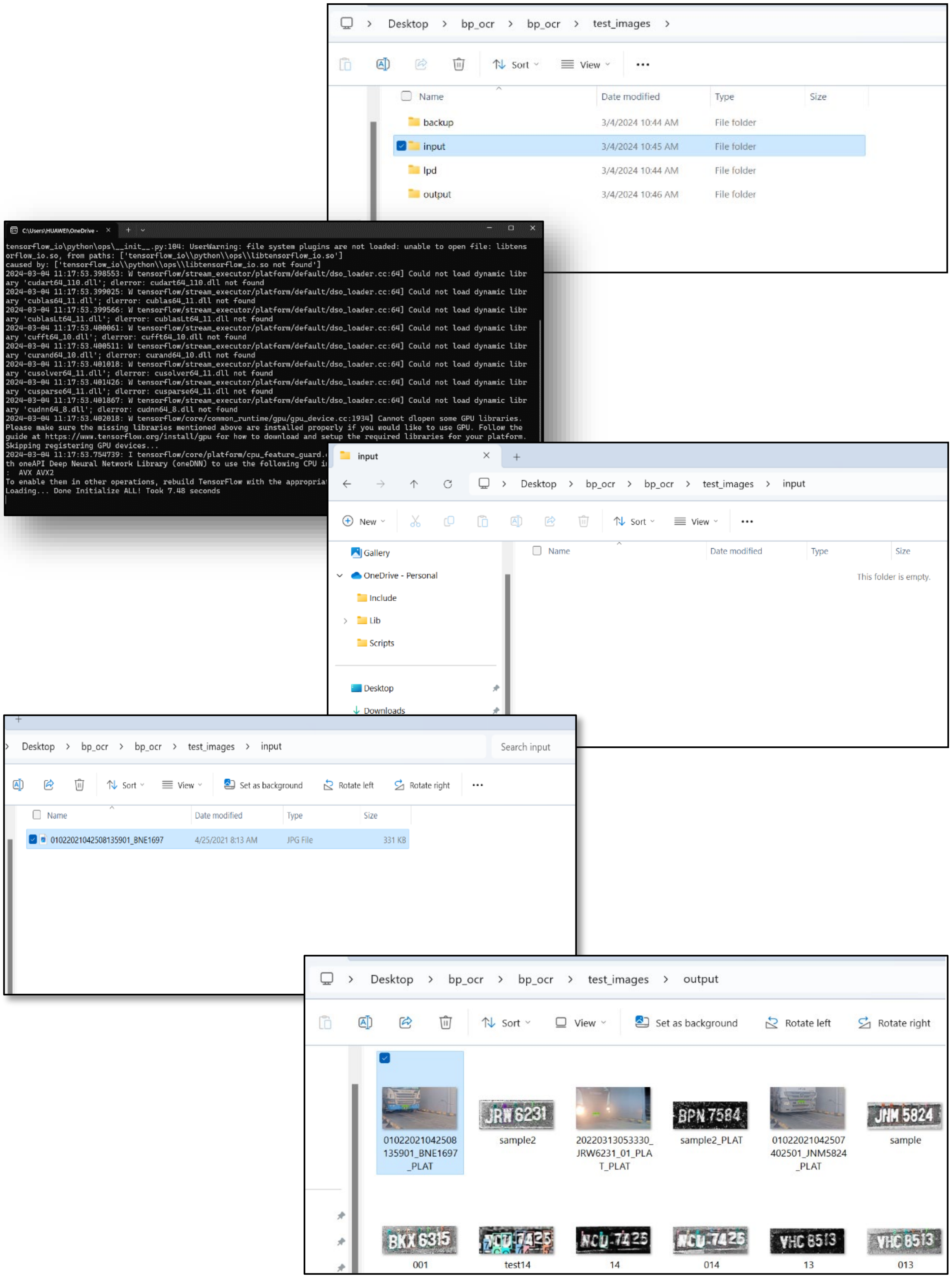


Figure 4.1 Flow of ALPR Detection System.

Figure 4.2 Result of License Plate Detection

4.1.1 Data Collection and Annotation

Figure 4.3 illustrates the steps for annotation images for training the dataset for the proposed model. Firstly, the input images are uploaded to the labelImg. Next, a rectangular bounding box is drawn around the license plate by clicking the bounding box on the left-hand side. Once the rectangular bounding box is drawn, there are two of the classes are presented for selection: ‘LICENSE PLATE’ and ‘NO PLATE’. When annotating the dataset, if the license plate on the image shown in visually clear categorised in ‘LICENSE PLATE’ class, while the license plate is blurry shown on the image consider then in NO PLATE’ class After completing the annotation process for the two classes, the images are saved into a separate folder along with their corresponding XML files as shown in Figure 4.4.

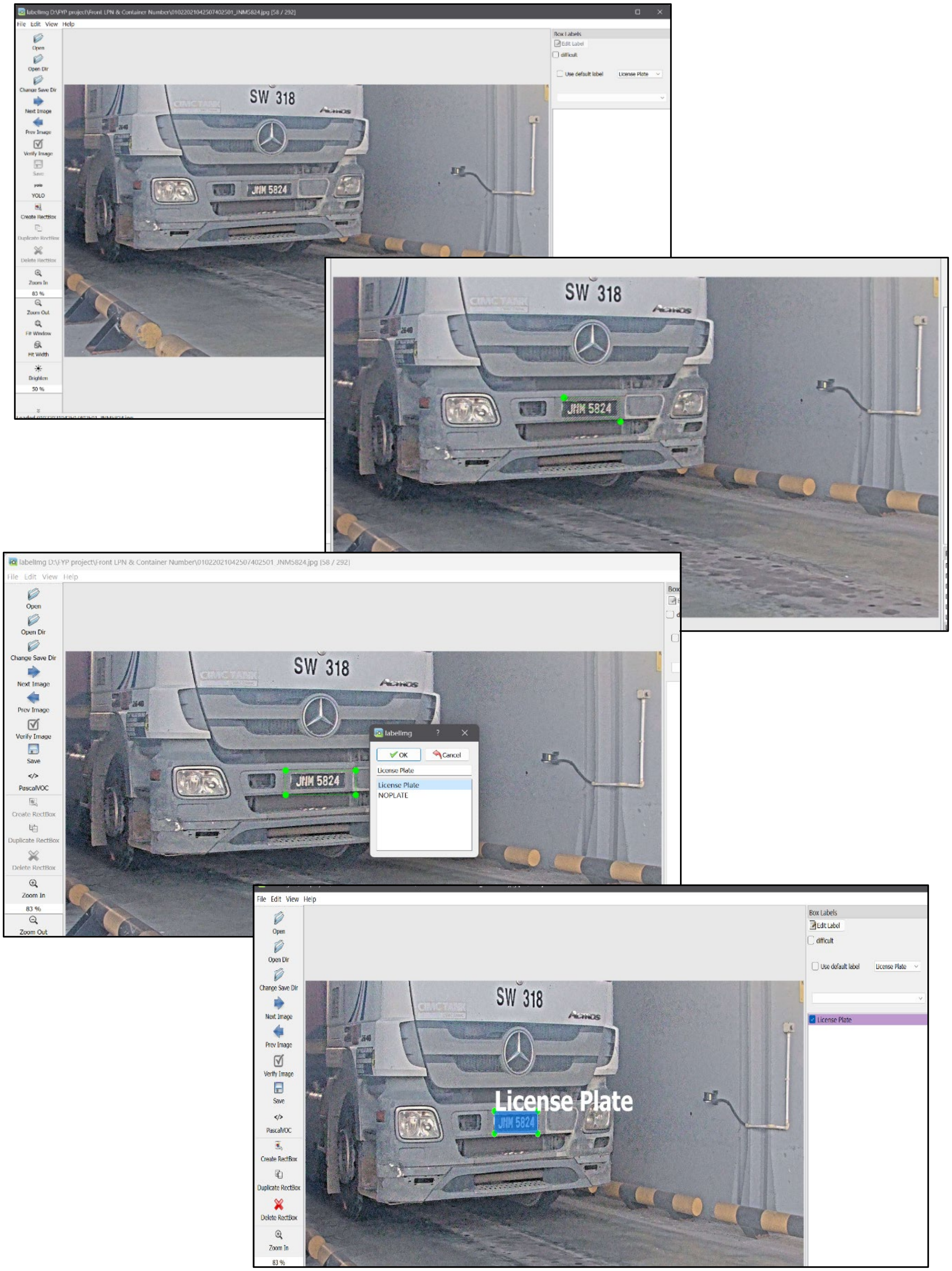


Figure 4.3 Annotation Dataset Steps

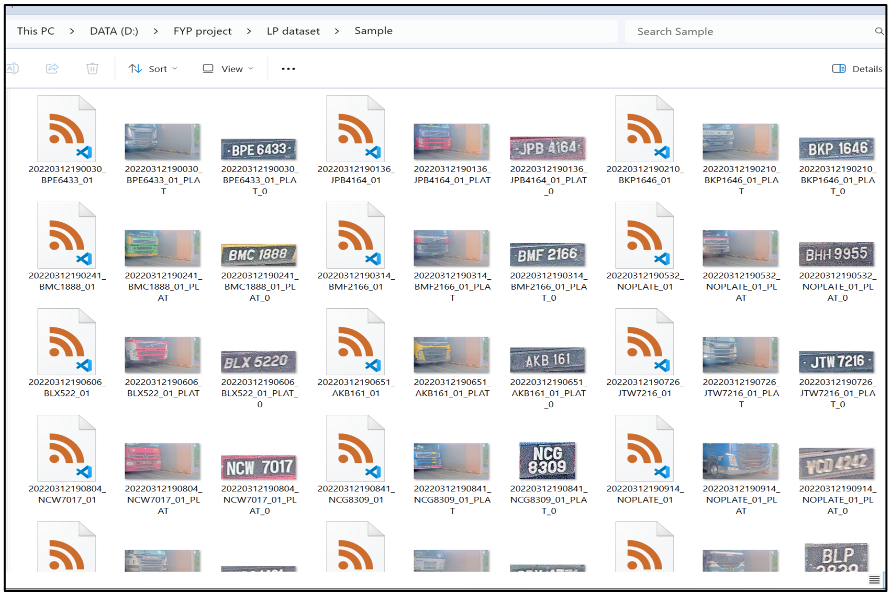


Figure 4.4 A Folder with Annotated Images and XML Files.

4.1.2 Faster R-CNN for Detection

The training of the Faster R-CNN model for license plate detection were conducted in collaboration with an industrial partner. The training process was conducted under strict privacy and confidentiality protocols. Hence, the training details process was not discussed in detail regarding the training remain proprietary. Nevertheless, the result of this collaborative effort on training model license plate detection yielded promising results. In this section, a detail discussion of the promising outcomes such as average confidence (CF), the average Intersection over Union (IoU), model accuracy, precision, and recall rate from proposed Faster R-CNN model for detecting 18 license plates.

In Table 11, some metrics were shown for evaluating the proposed ALPR performance. It includes several key components such as Ground Truth Label, Gt_label, indicating the actual license plate numbers as annotated by human. Ground Truth Bounding Box, Gt_bbox is a bounding box showing the details of precise location or coordinates of the license plate within each image. Prediction labelling, Pred_label consists of the license plate detected by the proposed ALPR system. While "Pred_cf", stands for Prediction Confidence, indicating that the ALPR system confidence level assigned to its predictions. Then, the "Pred_bbox" refers to Prediction Bounding Box specifying the detected location of the license plate which determined by proposed ALPR system. Intersection of Union, "IoU" quantifies the overlap between the ground truth bounding box and predicted bounding box for assessing the detection accuracy tasks. Then, Accuracy represents the overall accuracy of proposed ALPR system to check whether it can perform license plate detection successfully. Finally, the confusion matrix provides comprehensive summary of the correct and incorrect predictions for each class, including True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN) and perform analysis of classification errors and accuracy levels.

Table 11 The Outcomes on Proposed Model on License Plate Detection

Image	Gt_label	Gt_bbox	Pred_label	Pred_cf	Pred bbox	IoU	Accuracy	Confusion matrix
01022021042400140202.jpg	num plate	[992, 484, 1134, 566]	[992, 484, 1134, 566]	99.91747737	[1004, 491, 1131, 573]	0.762084	1	TP
01022021042400560602.jpg	num plate	[523, 516, 645, 623]	[523, 516, 645, 623]	99.88125563	[528, 519, 636, 621]	0.845152	1	TP
01022021042401581102.jpg	num plate	[549, 660, 727, 728]	[549, 660, 727, 728]	99.88791347	[545, 656, 733, 732]	0.848691	1	TP
01022021042410131302.jpg	num plate	[468, 673, 753, 780]	[468, 673, 753, 780]	99.8932898	[461, 673, 712, 768]	0.745247	1	FP
01022021042419192302.jpg	num plate	[762, 397, 891, 460]	[762, 397, 891, 460]	0	0	0	0	FN
01022021042419464002.jpg	num plate	[542, 545, 793, 632]	[542, 545, 793, 632]	78.50403786	[551, 546, 766, 631]	0.837662	1	TP

01022021042420241802.jpg	num plate	[651, 560, 823, 627]	[651, 560, 823, 627]	99.92254972	[644, 558, 819, 625]	0.886786	1	TP
20210429001814_Camera1_BNL120.jpg	num plate	[921, 250, 1049, 349]	[921, 250, 1049, 349]	99.82953668	[929, 253, 1037, 348]	0.811163	1	TP
20210429004421_Camera1_JMG7182.jpg	num plate	[989, 331, 1139, 398]	[989, 331, 1139, 398]	99.29246902	[984, 331, 1140, 394]	0.907247	1	TP
20210429082232_Camera1_JSG1328.jpg	num plate	[1211, 369, 1304, 434]	[1211, 369, 1304, 434]	99.92408752	[1214, 369, 1304, 435]	0.954091	1	TP
20210429094139_Camera1_BPY6451.jpg	num plate	[853, 300, 965, 385]	[853, 300, 965, 385]	99.88594055	[854, 301, 961, 387]	0.924099	1	TP
20210429190943_Camera1_JKT7195.jpg	num plate	[971, 355, 1122, 417]	[971, 355, 1122, 417]	99.93239641	[972, 352, 1126, 414]	0.881237	1	TP
20210429192349_Camera1_NOPLATE.jpg	num plate	[795, 387, 888, 469]	[795, 387, 888, 469]	97.26641774	[1182, 297, 1287, 361]	0	0	FP
20220310193530_BMJ783_01.jpg	num plate	[415, 551, 575, 597]	[415, 551, 575, 597]	99.86500144	[417, 545, 574, 602]	0.798066	1	TP

20220310201834_NOPLATE_01.jpg	num plate	[390, 572, 546, 634]	[390, 572, 546, 634]	99.74422455	[394, 573, 549, 631]	0.896603	1	TP
20220310223718_BJ4573_01.jpg	num plate	[416, 684, 593, 747]	[416, 684, 593, 747]	99.91502762	[418, 683, 588, 749]	0.919278	1	TP
20220311034253_NOPLATE_01.jpg	num plate	[435, 696, 626, 760]	[435, 696, 626, 760]	99.46086407	[447, 696, 611, 753]	0.766827	1	TP
20220311045744_BRA1377_01.jpg	num plate	[351, 701, 522, 752]	[351, 701, 522, 752]	99.84900951	[350, 701, 517, 757]	0.882879	1	TP

Table 14 had shown the outcomes obtained from the proposed Faster R-CNN model for detecting 18 license plates. These 18 images are the trucks captured in various environments. With these challenges images, the proposed model outcomes have been discussed.

The proposed model of the average confidence (CF) had scored of 92.94%, indicating high level of certainty in its predictions. Then, the average IoU, is a measurement of how accurately the proposed model localise the license plate within image. The IoU value, is the degree of overlap between predicted bounding boxes and ground truth license plate bounding boxes. Average IoU of this proposed model is 0.759. The higher the IoU value , the better the alignment on object localisation.

Next, the accuracy of the proposed model had demonstrated 88.89%. High accuracy showing the proposed model can making correct predictions correctly. Then, one of the outcomes had shown the precision of proposed model is 88.24%. A precision value of 88.24% obtained using Equation (4) had indicated the proposed model predict the license plate detection. This is a positive scenario because the higher precision value, the smaller chance of model predicts wrongly on localising license plate. By utilizing Equation (5), the recall rate of the model was found to be 93.75%, indicating the model capacity's to detect True Positive (TP). TP stands for the proposed model able to detect correctly in the presence of license plate image. With the outcomes of precision and recall of the proposed model, the F1 score of the proposed model can be obtained by performing the Equation (6), can found to be 90.91%. This outcome provided a balance assessment of the model's performance. A high F1 score indicates a good balance between precision and recall, so the model can localise the license plate effectively. Overall, these outcomes underscore the reliability and efficiency of the proposed Faster R-CNN model in performing license plate detection.

4.1.3 Faster R-CNN for OCR

In this section, the training process of proposed Faster R-CNN model in performing characters recognition tasks on license plate is not discussed as its classification under the privacy and confidentiality of industrial partner. Since, the proposed model we utilize it on testing phase of the image processing flows development to analysis the character recognition. A comprehensive discussion on those image processing flow will be presented in the next section.

4.2 COMPARISON BETWEEN THE IMAGE PROCESSING TECHNIQUE FLOWS

In this section, the performance of three image processing technique flows has been compared. These image processing technique flows were evaluated on their ability to recognise each character on license plate.

4.2.1 First Method: Image Processing Technique Flow

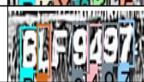
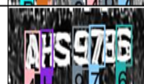
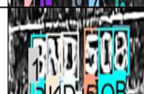
Table 11 displays the result analysis of the first image processing flow. The first image processing technique flow undergoes a ten-step process, including denoising, sharpening, and grayscale conversion, then setting threshold to zero for removing unwanted noise and details. Morphological operations further refine the image followed by normalisation, contrasting and additional denoising with applying Fast NI Mean Denoising.

There are 13 images are applied these techniques and recognised by the recognition model. Each of the characters accuracy and the time processing had been recorded in the Table 11. Throughout the analysis, there are some characters are lower than others in recognition. For instance, in image ‘test8.jpg’ and ‘test11.jpg’, the accuracy of the recognition model for the character ‘0’ is detect as character ‘Q’ and with the accuracy of 81.40% and 87.92%. The model faced difficulty distinguishing between '0' and 'Q'. In 'test2.jpg’ image, the recognition model detects two possible characters on the character '3': '3' with 99.98% accuracy and 'E' with 83.37%.

Nevertheless, the model would misinterpret the character ‘3’ and ‘E’ due to the denoising step might not be perfect, leaving residual noise, leading the model misinterpret the character ‘3’ to ‘E’ detection. Moreover, over sharpening also can leading to misinterpretation on characters. The over sharpening on characters’ edge or halos that expand certain details of ‘3’ making it to ‘E’ to the model.

The time taken to process the images which applied the first method image processing were varies slight from image to image. This is because the difference of complexities of the images. The average of time taken processing the images is 2.58 seconds.

Table 12 Result Analysis of the First Image Processing Flow

image processing without applying image enlargement													
original image->denoised -> sharpen->greyscale->threshtoZero->MorphologicalOpen->Normalisation->constrast->dilated->denoised ->FastMeanNl-> sharpen													
No.	Images	No.Plate		Detected	Each character accuracy							OCR Recognition	Total time processing
1	test11.jpg	BKX 6315		BKX6315	B_99.99	K_99.96	X_99.92	6_89.41	3_98.81	1_99.91	5_98.76	2.53 s	4.68 s
2	test10.jpg	BLF 9497		B4L9497	B_99.94	4_100.00	L_91.64	9_99.94	4_100.00	9_99.94	7_99.99	2.70 s	4.54 s
3	test9.jpg	AHS 9786		4H976	4_99.83	H_97.33		9_99.97	7_81.72		6_98.41	2.66 s	4.28 s
4	test8.jpg	BND 508		B1KND5Q8B	B_99.98,1_82.96	K_89.32, N_98.97	D_99.29	5_99.63	Q_81.40	8_98.53, B_97.06		2.48 s	4.04 s
5	test7.jpg	BQQ 243		BQQ243	B_99.96	Q_99.94	Q_99.91	2_99.95	4_99.99	3_99.64		2.46 s	4.05 s
6	test6.jpg	BMW 704		BMW704	B_99.94	M_99.99	W_99.88	7_99.98	0_99.98	4_99.99		2.40 S	4.17 s
7	test5.jpg	BLT6121		BLT6121	B_99.69	L_99.01	T_99.99	6_99.94	1_100.00	2_99.99	1_96.56	2.32 s	4.42 s
8	test4.jpg	PDY 4774		RQY4774	R_70.11	Q_95.82	Y_99.98	4_100.00	4_100.00	7_99.97	7_79.49	2.76 s	4.15 s
9	test3.jpg	BHH 9955		BHH9955	B_99.98	H_99.99	H_99.97	9_99.94	9_99.94	5_99.99	5_99.96	2.21 s	4.12 s
10	test2.jpg	BPE 6433		BPE643E3E	B_100.00	P_99.98	E_100.00	6_99.97	4_99.98	3_99.98,E_83.37	3_99.84,E_81.11	2.26 s	4.14 s
11	test1.jpg	NVC 9020		C9Q2			C_97.87	9_99.80	Q_87.92_	2_99.93		2.84 s	3.92 s
12	test13.jpg	JPB 4164		BJP64	J_99.53	P_97.86	B_69.69			6_98.26	4_69.31	2.61 s	4.22 s
12	test14.jpg	NCU 7425		7CGGQ745	7_74.80	C_84.22, G_97.92	G_96.32, Q_94.84	7_95.41	4_99.99		5_93.67	3.38 s	4.68 s

4.2.2 Second Method: Image Processing Technique Flow

Table 12 displays the result analysis of the second image processing flow. The second image processing technique flow undergoes multi-step process to enhance the character recognition. The process including image enlargement, denoising, sharpening, and grayscale conversion, then Adaptive Thresholding Gaussian Mean utilized block size of 7 and constant of 2 for converting grayscale into binary. It calculates the mean intensity of a local neighbourhood of 7 x 7 pixels, subtracts a constant of 2 from the mean, then the result was used as the threshold for converting the pixel to either black or white in the binary image. Morphological Opening operations further refine the image followed by normalisation, contrasting and additional denoising with applying Fast NI Mean Denoising.

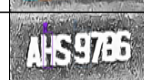
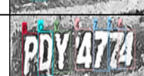
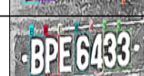
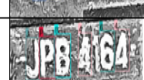
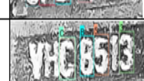
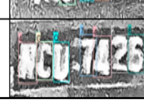
The enlargement operation on an image can enhance its resolution, transforming it to high-resolution image. In this project, ESPCN model performs enlargement with a factor of 3. With the factor of 3, the output image size was three times larger than the input image. Subsequently, the higher magnification is intended to provide even more enhanced details and clarity. The image enlargement as a crucial initial step in the image technique flow, as the subsequent application of image techniques can have distinct effects on the high-resolution image.

There are 14 images are applied these techniques and recognised by the recognition model. With applying second method of image processing flow, those characters in white distinctly stood out from the background. Each of the characters accuracy and the time processing had been recorded in the Table 12. Throughout the analysis, almost all of the images can well-recognised, with high accuracy on each character. This image enhancement had contributed improve visibility and facilitates more effective character recognition in the processed images.

There is only one image where the recognition of characters is comparatively lower than others. In the '003.jpg' image, which contains the license plate 'AHS9786,' the recognition result only captured 'H9.' This outcome can be attributed to the abnormal arrangement of characters on the license plate, where they are tightly grouped together. In typical scenarios, license plate characters are spaced at a normal distance. Hence, the model encountered challenge in segmentized the characters.

The time taken to process the images which applied the second method image processing were slower than first method image processing flow. This is because the complexities and more image processing technique steps applied in the image process flow. The average time taken to process the images is 3.65 seconds, which is more than 0.98 seconds compared to the time taken by the first method in the image processing flow.

Table 13 Result Analysis of the Second Image Processing Flow

Image Processing with applying ESPCNx3, Adaptive gaussian (7,2)													
original image->denoised -> sharpen->greyscale->AdaptiveThreshGaussianMean(7,2)->MorphologicalOpen->AdaptiveHistogram->Normalisation->contrast->denoised ->FastMeanNl-> sharpen													
No.	Images	No.Plate		Detected	Each character accuracy(%)							OCR recognition time	Total time processing
1	001.jpg	BKX 6315		BKXK6G315	B_100.00	K_99.86	X_98.50, K_99.86	6_90.22, G_94.10	3_99.99	1_99.68	5_99.96	3.79 s	5.18 s
2	002.jpg	BLF 9497		BLF9497	B_100.00	L_99.98	F_99.83	9_100.00	4_100.00	9_99.80	7_100.00	3.24 s	5.13 s
3	003.jpg	AHS 9786		H9		H_99.62		9_99.94				3.53 s	5.08 s
4	004.jpg	BND 508		BNQ 508	B_99.97	N_99.63	Q_99.87	5_99.97	0_81.68	8_99.81		3.50 s	5.00 s
5	005.jpg	BQQ 243		BQQ243B	B_100.00	Q_99.99	Q_99.83	2_100.00	4_100.00	3_92.59, B_76.12		3.97 s	5.05 s
6	006.jpg	BMW 704		BMW704	B_99.99	M_99.98	W_99.71	7_99.99	0_99.95	4_100.00		3.78 s	5.07 s
7	007.jpg	BLT6121		BLT6121	B_99.96	L_99.99	T_99.99	6_99.85	1_100.00	2_100.00	1_99.93	3.59 s	5.02 s
8	008.jpg	PDY 4774		PQY4774	P_99.99	Q_91.36	Y_99.97	4_100.00	7_99.99	7_99.97	4_100.00	3.77 s	5.00 s
9	009.jpg	BHH9955		BHH9955	B_100.00	H_99.99	H_99.99	9_99.99	9_99.94	5_99.96	5_99.96	3.86 s	5.05 s
10	010.jpg	BPE 6433		BPE 6433	B_100.00	P_99.99	E_99.99	6_99.99	4_100.00	3_99.99	3_99.99	3.70 s	5.17 s
11	011.jpg	NCV 9020		NG3 902Q	N_99.70	G_94.51, 3_81.43		9_99.98	0_100.00	2_99.99	Q_96.75	3.73 s	5.10 s
12	012.jpg	JPB 4164		JPB 4164	J_99.96	P_99.96	B_100.00	4_99.36	1_78.31	6_100.00	4_100.00	3.78 s	5.10 s
13	013.jpg	VHC 8513		GCB513			G_93.19, C_67.20	B_66.99	5_99.98	1_99.96	3_99.98	3.06 s	4.97 s
14	014.jpg	NCU 7425		4GCQ 7426	4_99.86	G_84.43, C_99.88	Q_99.77	7_99.97	4_74.33	2_99.98	6_96.37	3.74 s	4.98 s

4.2.3 Third Method: Image Processing Technique Flow

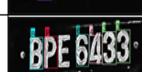
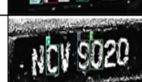
Table 13 displays the result analysis of the third image processing flow. The third image processing technique flow undergoes multi-step process to enhance the character recognition. Similarity to the second method of image processing flow, the third process also including image enlargement, denoising, sharpening, and grayscale conversion, then Adaptive Thresholding Gaussian Mean utilized block size of 11 and constant of 2 for converting grayscale into binary. It calculates the mean intensity of a local neighbourhood of 11 x 11 pixels, subtracts a constant of 2 from the mean, then the result was used as the threshold for converting the pixel to either black or white in the binary image.

Morphological Opening operations further refine the image followed by normalisation, contrasting and additional denoising with applying Fast NI Mean Denoising. The enlargement operation on an image can enhance its resolution, transforming it from low-resolution to high-resolution. The image enlargement as a crucial initial step in the image technique flow, as the subsequent application of image techniques can have distinct effects on the high-resolution image.

There are 14 images are applied these techniques and recognised by the recognition model. With applying second method of image processing flow, the background became black and only the characters were in white. Consequently, the characters can be distinctly seen and recognized. Accuracy and processing time for each character have been recorded in Table 13. Throughout the analysis, every image was successfully recognized. However, not all characters were identified accurately. This image enhancement had contributed improve visibility and facilitates more effective character recognition in the processed images.

The time taken to process the images which applied the third method image processing is the longest among three image processing image flows. This is because the complexities and more image processing technique steps applied in the image process flow. The average of time taken processing the images is 3.26 seconds.

Table 14 Result Analysis of the Third Image Processing Flow

Image Processing with applying ESPCNx3, Adaptive gaussian (11,2)														
original image->denoised -> sharpen->greyscale->AdaptiveThreshGaussianMean(11,2)->MorphologicalOpen->AdaptiveHistogram->Normalisation->constrast->denoised ->FastMeanNl-> sharpen														
No.	Images	No.Plate		Detected	Each character accuracy(%)							OCR recognition time	Total time processing
1	1.jpg	BKX 6315		B8635	B_93.38, 8_80.69			6_91.03	3_99.97		5_95.08	3.91 s	5.05 s
2	2.jpg	BLF 9497		BL949	B_99.45	L_76.99		9_99.91	4_97.28	_9_96.52		3.57 s	4.88 s
3	3.jpg	AHS 9786		S			S_85.08					3.71 s	4.83 s
4	4.jpg	BND 508		8559	8_70.19			5_99.86		5_72.06, 9_85.71		3.83 s	4.82 s
5	5.jpg	BQQ 243		BQQ24A3	B_93.03	Q_99.89	Q_99.95	2_99.99	4_95.87, A_70.44	3_99.92		3.63 s	4.98 s
6	6.jpg	BMW 704		BMWQ4	B_99.66	M_99.96	W_93.21		Q_98.88	4_100.00		3.70 s	4.68 s
7	7.jpg	BLT6121		525E	5_100.00	E_99.02		5_98.65		2_100.00		2.46 s	4.85 s
8	8.jpg	PDY 4774		PY4	P_99.98		Y_99.99	4_100.00				2.30 s	4.80 s
9	9.jpg	BHH9955		BHHH9955	B_96.29	H_95.67	H_90.02, H_77.16	9_99.98	9_99.98	5_99.98	5_99.94	2.46 s	4.83 s
10	10.jpg	BPE 6433		BP 6433	B_99.98	P_99.92		6_99.92	4_99.56	3_100.00	3_100.00	2.42 s	4.83 s
11	11.jpg	NCV 9020		C93		C_98.24		9_99.87	3_88.92			2.33 s	4.88 s
12	12.jpg	JPB 4164		JPB 6G4	J_97.34	P_99.78	B_83.92			6_98.57, G_70.31	4_99.72	3.78 s	4.87 s
13	13.jpg	VHC 8513		N/A								3.85 s	4.35 s
14	14.jpg	NCU 7425		4QU4	4_94.04		U_84.18, Q_70.16		4_74.93			3.65 s	4.87 s

4.2.4 Comparison Three Method of Image Processing Flow

The three method of image processing flow were discussed and analysed throughout the comparison. The first method of image processing flow did not involve image enlargement. This leads to recognition model might be recognised different characters. Figure 4.5 showed ‘test8.jpg’ image with the license number of ‘BND 508’, the recognition on character ‘8’ was detected out possible character ‘B’ with 97.06%. In addition, in the same image, the character ‘1’ was indeed misrecognised by the model. Lower resolution images resulted in pixelated and blurry characters, which further hindered the model's ability to recognise the intricate details that differentiate characters like '1' from others. Nevertheless, the OCR recognition time was the shortest compared with second and third method of the image processing flow.

Figure 4.5 The 'test8.jpg' Image

With the implementation of image enlargement, the pixels intensity higher and became high-resolution image. The characters were more easily recognised. By comparison the ‘test13.jpg’ that applied first method and ‘012.jpg’ image that applied second method, it is obviously all the characters were detected in ‘012.jpg’. Image enlargement involves the technique of upscaling, where a lower-resolution image is transformed into a higher-resolution version. Figure 4.6 illustrates the dimensions of 'test13.jpg,' whereas Figure 4.7 demonstrates that the size of '012.jpg' is three times larger than that of 'test13.jpg'. This process aims to enhance the details and overall quality of the visual content. The proficient recognition of characters can be attributed not only to the image enlargement but also to the implementation of various image techniques within the second method.

Figure 12 The Size of 'test13.jpg'

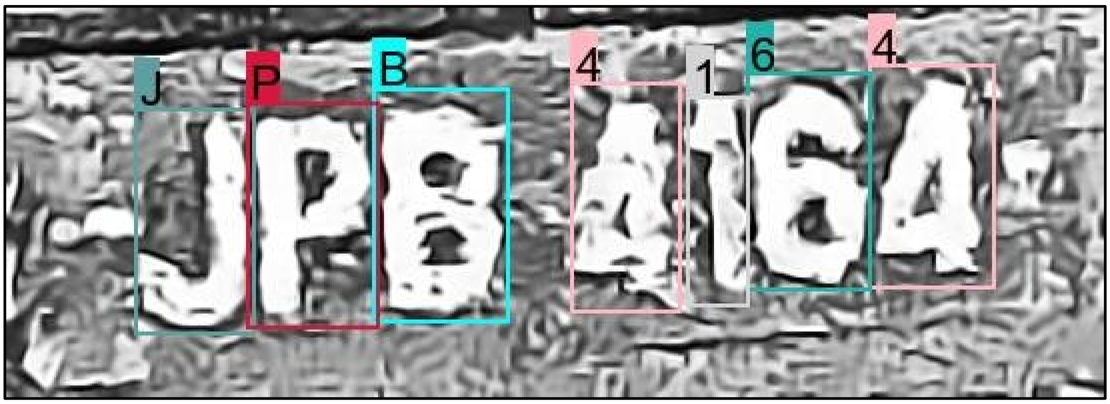


Figure 13 The Size of '012.jpg'

The Adaptive Thresholding Gaussian Mean was applied to the second and third methods of the image processing flow. The threshold value in the third method was larger than in the second method flow. This was because the mean intensity calculated from a larger neighbourhood of 11 x 11 pixels was likely influenced by a broader range of pixel values, leading to a higher threshold value compared to the smaller neighbourhood of 7 x 7 pixels under the same constant adjustment. Consequently, the background of the second method in Figure 4.8(a) became grey, while the background of the third method, as illustrated in Figure 4.8(b), turned black.

Nonetheless, the characters in (a) appeared whiter and clearer compared to the third method. In the third method, with a larger neighbourhood, the thresholding considered a bigger area. This might have included darker background pixels along with the character, resulting in a higher overall threshold. This higher threshold could have led to some character pixels being classified as black, making them appear less white and clear compared to the second method.

(a)

(b)

Figure 14 The background of Second Method(a) and Third Method(b) After Applying Adaptive Thresholding Gaussian Mean

The OCR recognition time utilized in the second and third methods was longer than the first method because both methods implemented image enlargement. However, with similar image processing steps, the second method proved to be faster than the third method. The potential reason for this was only the size of the neighbourhood used in both methods consumed time.

Table 15 had showed each characteristic of these image processing methods and the distinct differences of each method can be seen and compared. From the table, it concluded that the second method emerged as the most effective within the image processing flow among these considered methods. Each image subjected to the second method exhibited successful recognition, with characters being distinctly clear and easily identified following the application of image processing techniques. Although the processing time is relatively slower, the overarching objective of proposing a robust ALPR system is achieved, ensuring precise accuracy in character recognition.

Table 15 Summaries of Image Processing Flows

	Method 1	**Method 2**	**Method 3**
Image Enlargement Operation	No	Yes, ESPCN model with factor of 3	Yes, ESPCN model with factor of 3
Basic Thresholding Operation	Yes, Thresholding to zero	No	No
Adaptive Thresholding Operation	No	Yes, Adaptive Thresholding Gaussian Mean	Yes, Adaptive Thresholding Gaussian Mean
Morphological Operation	Yes, Morphological Opening and Dilation used	Yes, Morphological Open used	Yes, Morphological Open used
OCR Recognition	Fastest	Slower than Method 1 but faster than Method 3	Slowest
Characters Recognition Performance	Generally identifiable, but with occasional multiple predictions	Accurate recognition for each character without ambiguity	Some characters exhibit difficulty in recognition, leading to decreased accuracy
Total Processing Time	3.95 seconds per image	5.06 seconds per image	4.82 seconds per image

4.3 LICENSE PLATE RECOGNITION EVALUATION

In this section, the performance of the proposed model with applying second method of image processing was evaluated on four different scenarios of vehicle license plates. To ensure the proposed model's ability to handle these scenarios, 20 images were prepared for each, total 80 images. The results were then compared under various conditions, including daytime with clear visibility, daytime with visible characters, nighttime with normal light and nighttime with strong light.

4.3.1 Real Life Verification

In the scenario 1, Figure 4.9 showed that a rear truck's license plate able to detect by the system with 100% accuracy during daytime. This indicated that the proposed model had successfully learned the patterns and features of license plates and was able to recognize and interpret characters accurately.

Figure 15 The Rear License Plate was Detected by Proposed Model During Daytime

In the scenario 1, a rear license plate was captured during daytime condition, displayed clearly visible characters on the license plate. This visibility facilitated accurate segmentation and recognition, as highlighted in Figure 4.10. The proposed model demonstrated exceptional performance in this scenario, achieving a 100% accuracy rate in detecting and predicting the license plate characters in the correct sequence. The average accuracy of characters scored an impressive 99.96%, with the OCR recognition time recorded at 2.32 seconds.

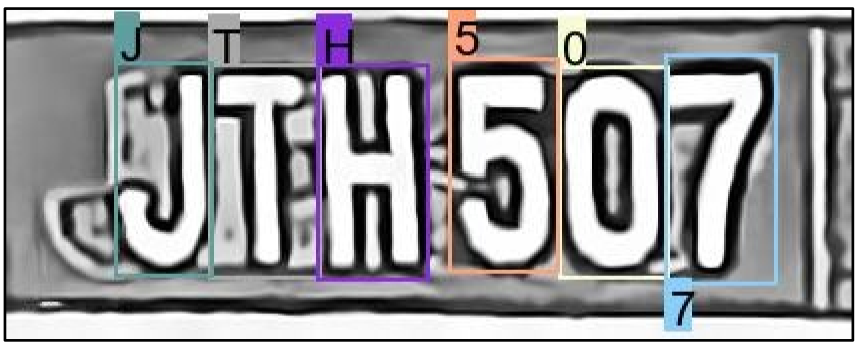


Figure 16 The Image was Recognised by Proposed Model During Daytime

The proposed ALPR system effectively performed performance in detecting and recognising 20 images in this scenario, achieving 100% accuracy in both detection and recognition as shown in Table 15. Furthermore, each character was well recognised and predicted by the proposed system. The optimal environment in this scenario contributed to the proposed system able to achieve well performance due to the surrounding environment has sufficient light and good quality camera.

One of the key factors contributing to the high accuracy of the system in the scenario 1 is the quality of the input image. The clear characters, distinct from the background, and the absence of noise in the image all contributed to an optimal input image. These factors ensured that the proposed model could detect, segment, and recognize the words accurately. Additionally, the proposed model's architecture, utilizing uses Faster R-CNN model with the

CNN backbone, is highly effective in detecting and recognizing license plate precisely. Overall, the result of the scenario 1 suggests that the proposed model is highly effective in detecting license plate during daytime where the characters in clear and easy to recognise.

In the scenario 2, a front license plate was captured in daytime conditions displayed visible characters on the license plate. However, the background of the license plate, as illustrated in Figure 4.11, could be visibly seen and was not completely black. This problem could have posed challenges for the proposed model in character recognition. The presence of a white scar on the license plate might have caused the proposed model to encounter difficulties in segmenting and recognizing characters, particularly considering that the characters were also in white. Despite this challenge, the proposed model was able to correctly predict each of the characters as shown in Figure 4.13.

Figure 4.11 The License Plate was Detected by Proposed Model During Daytime

Adaptive Threshold Gaussian Mean, an image processing technique designed to enhance the visibility and clarity of an image by optimizing different light levels. By applying a Gaussian mean in this scenario, the technique ensures a smooth transition between different light levels, contributing to an overall enhancement in image quality as shown in Figure 4.13.

Figure 17 The Image After Applying the Second Method Image Processing Flow

The characters can be seen clearly and stands out from the background. As the results in Figure 4.13, the individual characters segmentized and recognised correctly by the proposed model with the average accuracy of 99.89% in 2.23 seconds.

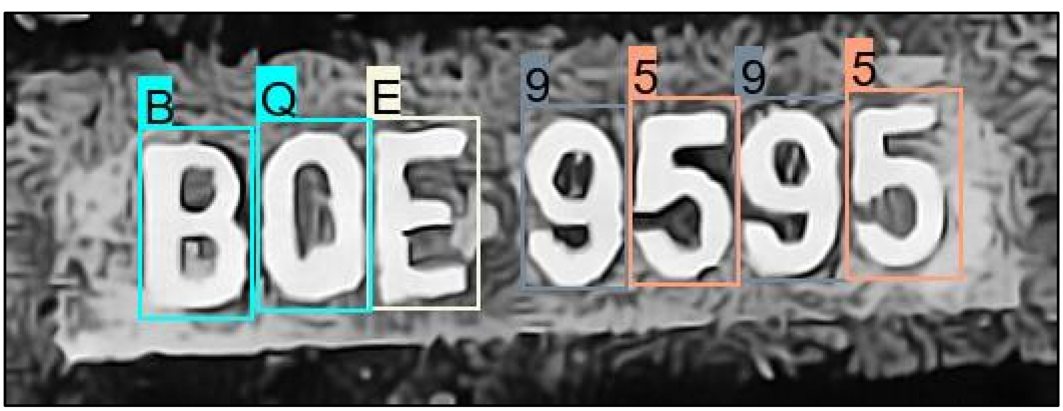


Figure 18 The License Plate was Recognised by Proposed Model During Daytime

In this scenario 2, the optimal environment contributed to the proposed system able to achieve well performance due to the surrounding environment has sufficient light. The proposed ALPR system demonstrated well performance in detecting and recognising 20 images in this scenario, achieving 100% accuracy in both detection and recognition as shown in Table 15. However, not each character in every image was well recognised and predicted by the proposed system.

Figure 19 The Presence of Dirt on the License Plate

There are potential reasons faced by proposed system that impact its performance. Firstly, the quality of the camera used can significantly impact the system's ability on detection and recognition. The camera used in scenario 2 is not good in the camera used in scenario 1. Secondly, the presence of dirt on the license plate as shown in Figure 4.14 cause hinder the system's ability to accurately recognised characters. The Figure 4.15 showed the result from proposed system recognition. Hence, the proposed system difficulties in license plate detection and recognition when encountered with these challenges.

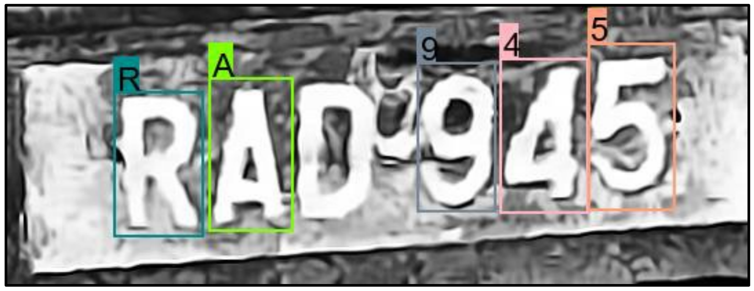


Figure 4.15 The Result of License Plate Detection

Overall, the proposed model was still able to correctly be recognised all the characters, indicating the proposed model capable to handle the image with varying illumination conditions.

In the scenario 3, a truck captured from the front in nighttime condition displayed visible characters on the license plate. In Figure 4.16, the surrounding conditions were dimmer compared to the scenario 1 and scenario 2. Additionally, the truck's light was considered one of the reasons for the difficulty in recognizing the characters clearly, as the light directly impacts the license plate. This could pose a challenge for the proposed model in license plate detection and recognition. However, the system managed to detect and recognise the license plate.

Figure 4.16 The License Plate was Detected by Proposed Model During Nighttime

The proposed model capable detect the license plate with 100% accuracy. Consequently, the characters recognition in Figure 4.17 predicted each character in correct sequence with average accuracy of 99.96% with 2.51 seconds. This is because the truck's light was not bringing huge impact on the license plate in Figure 4.16. With the implementation of Adaptive Threshold Gaussian Mean that capable handle the different light conditions, the proposed model predicted the characters well.

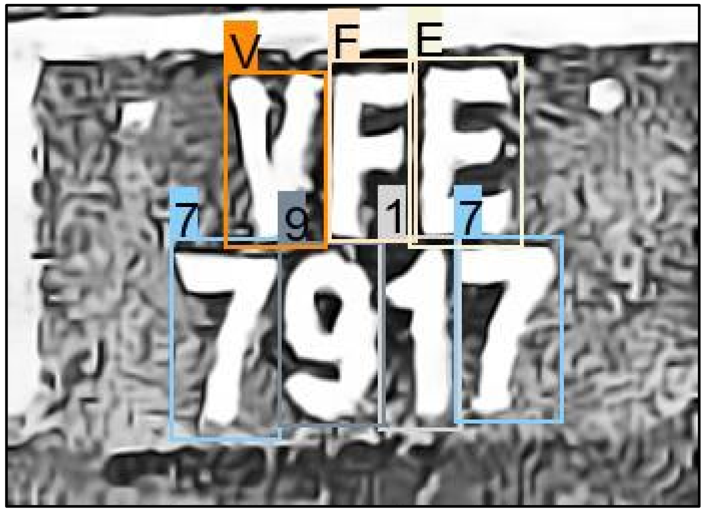


Figure 4.17 The License Plate was Recognised by Proposed Model During Nighttime

In this scenario 3, the dimmer environment contributed low risk to the proposed system. The proposed ALPR system maintaining well performance in detecting but a declined 20% on character recognition across these 20 images, shown in Table 16. In addition, there were also had 6 images showed that not every was well recognised and predicted by the proposed system. This indicates that the low light conditions can significantly affect the system's ability to read the license plate.

The overall accuracy of the proposed model was efficient in the scenario 3 during nighttime conditions with the vehicle's lights on. This scenario proved more challenging compared to the scenario 1 and scenario 2 due to the dark surrounding environment. The impact of the truck's lights on the license plate may have caused varying levels of light illumination in the images.

In the scenario 4, a truck was captured from the front in nighttime conditions. As shown in Figure 4.18, the truck's lights were strong and covered the license plate. Given that the license plate background is black, the proposed model was capable of detecting the license plate position with 100% accuracy.

Figure 4.18 The License Plate was Detected by Proposed Model During Nighttime

There were several challenges in this scenario, including a dark surrounding and the effect of the vehicle's lights on the license plate. These challenges made it difficult for the proposed model to distinguish the license plate, even though it could recognize the characters on the license plate. Nonetheless, the proposed model not only detected the license plate successfully but also segmented and recognized the characters effectively.

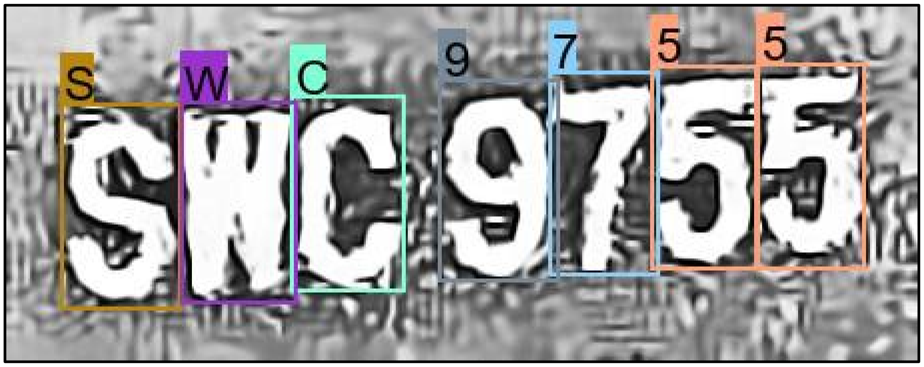


Figure 4.19 The License Plate was Recognised by Proposed Model During Nighttime

Figure 4.19 displays the results of character recognition by proposed model. The average accuracy for each character was 99.97%, achieved with a recognition time of 2.41 seconds. The implementation of image processing played a significant role in making the characters distinctly clear for recognition. While this scenario may represent an ideal situation, it serves as an important benchmark for evaluating the performance of the proposed model and can be utilized to establish a more robust standard for future comparisons in evaluations.

The dimmer environment and the strong light in scenario 4 had contributed low risk to the proposed system. The proposed ALPR system maintaining well performance on detection and character recognition across these 20 images, shown in Table 16. In addition, there were also had 10 images showed that not every was well recognised and predicted by the proposed system. This indicates that the vehicles' light can significantly affect the system's ability to read the license plate and even more challenging for ALPR system than the nighttime with normal light condition. Strong light can create glare, overexpose, and low contrast, making the character recognition faces difficulties.

Table 16 Comparison between Various Scenarios

	Scenario 1	**Scenario 2**	**Scenario 3**	**Scenario 4**
Environment	Daytime with clear visibility	Daytime with uneven colour background	Nighttime with Normal light	Nighttime with strong light
Challenge Level	Low	Low	Medium	High
Is License Plate Detected?	Yes	Yes	Yes	Yes
LP Detection Result	100%	100%	100%	100%
Character Recognition Result	100%	100%	80%	100%
Each Character Recognition Result	100%	90%	70%	50%
Can the proposed model handle this scenario?	Yes	Yes	Yes	Yes

All in all, the proposed model in this project achieved the expectation results. The proposed image processing flow, the Method 2, played a significant role in making the characters clear, standing out from the background. Table 16 showed the analysis of ALPR system performance highlights the impact of surrounding environments on license plate recognition accuracy. While the proposed system performed excellent in scenario 1. In nighttime scenario, particularly those with strong backlights, showed the biggest challenge, with recognition accuracy dropping to 50%. These findings emphasize the need for further development of ALPR system to improve their performance in dimmer surroundings and strong vehicle’s light.

CHAPTER 5

CONCLUSION

In this project, an ALPR system was developed for real-life applications in the industrial sector. After researching various models, Faster R-CNN model was chosen for training as both license plate detection and recognition model. The results demonstrated that the proposed Faster R-CNN model, as an ALPR model, was capable of detecting license plates and recognizing characters on the license plate.

There are various image processing techniques were analysed and compared to enhance license plate recognition. The study involved a comprehensive comparison of different image processing techniques flow for license plate recognition, evaluating the ALPR performance across four diverse types of scenarios. The results indicate that the second method of image processing technique flow surpasses first and third method of image processing flow in term of characters recognition and accuracy. Although the second and third methods of the image processing flow exhibit similarities, the second method stands out as the better choice for due to its incorporation of effective image techniques and optimized parameters, surpassing the performance of third method.

Furthermore, the evaluation of ALPR system revealed some factors affecting its performance such as spacing between characters, background noise, vehicles' light, surrounding lightning conditions can significantly impact its performance. These findings suggest that there is room for improvement in terms of making such ALPR system more robust to these factors in the context of license plate recognition.

Overall, this study provides valuable insights into license plate recognition and has important implications for future research in this field. Future research could explore other neural network architectures or investigate new image processing techniques method ways to improve the robustness of ALPR system.

5.1 ENGINEER AND SOCIETY

As an engineer, acknowledging the social implications of work and contemplating potential consequences is important. The designed and development of industrial license plate recognition system employing neural networks performing two parts of model processing such as license plate detection and character recognition. Nevertheless, the details consideration of potential implications associated with the utilization of such technology is deemed essential.

The proposed license plate detection system can have potential impact on the society by automating the crucial aspect of vehicle identification. This technology can aid law enforcement, aiding in the fast identification of vehicles linked to the criminal activities and contributing to enhanced community safety. Its applications also extend to urban planning and traffic management for optimizing the traffic flow by offering the valuable data.

Through the system applications, the ALPR system could streamline tasks on vehicle identification and documentation, saving both time and effort. This efficiency enhancement provides new opportunities for individuals to focus on more intricate aspects for their work.

However, ethical considerations play a crucial role in the development and deployment of the technology. It is essential to ensure that the system respects privacy and is used responsibly. Navigating the delicate equilibrium between innovation and privacy safeguards is imperative for the ethical implementation for the ALPR system. The careful balance knowledges the potential benefits of the ALPR technology while safeguarding individual rights and society values.

In conclusion, the ALPR system represents a contribution to society through addressing the real-world challenges and enhancing the efficiency in various domains. It is engineer's responsibility to implement a robust data protection measure, consider the ethical implications of using this system, and ensure that it is accessible to all individuals who may benefit from it.

5.2 ENVIRONMENT AND SUSTAINABILITY

The proposed ALPR system indicates positive prospect for contributing of Sustainable Development Goals (SDGs). The proposed system has potential environmental benefits. The ALPR system reduces the need for physical paper documents and associated transportation and storage costs. The system can help move towards a more sustainable and eco-friendly approach to documentation.

One potential benefit of applying neural networks on license plate recognition is the potential to reduce the energy consumption to traditional methods. Traditional license plate recognition methods typically encountered higher energy utilization due to their reliance on extensive hardware and computational resources. On the other hand, neural networks utilization is more efficient and streamlined contribute to more energy efficient and sustainable solutions for ALPR system. Companies and organizations can reduce their carbon footprint and save energy costs while at the same time still can achieving the high levels accuracy and efficiency. The implementation of neutral networks able fostering a positive impact on both environment and resource.

The ALPR system also plays a role in facilitating data-driven decision-making by offering the system has ability to provide accurate and timely data. This allows Businesses and governments to leverage the system's potential, obtaining the advantages in various aspect such as strategic planning, resource allocation, and operational efficiency. The ALPR system also performs decision-making by fostering a data-driven approach, aligning with the SDG 17. Overall, the proposed ALPR system not only reduces the paper documentation but also aligning with the sustainability goals in SDGs.

5.3 ENGINEERING ECONOMIC CONSIDERATIONS AND ECONOMIC POTENTIALS

The optimized ALPR system requires the comprehensive review on engineering economic requirements. The initial investment for the fundamental ALPR system includes the procurement costs for hardware components such as camera and servers, and software development such as the license fees of the image processing software. The ongoing costs involves the system maintenance, software updates, training programs, and operational expenses.

The benefits quantification on this optimized ALPR system is expected to yield costs savings in manual monitoring task, labour costs and operational ineffectiveness. Besides, the scalability of the optimized ALPR system also can across various industries brings further expanding its economic potential. For instance, the optimized ALPR system generates the revenue by enhanced the traffic management, toll collection, and more efficient parking management. Certainly, let's connect the paragraph about the ROI calculation with the rest of the report for better flow:

$$ROI = \left(\frac{Total\ Cost}{Net\ Benefit}\right) \times 100\% \qquad \text{Equation}(18)$$

To assess the economic feasibility and the potential impact on the optimized ALPR system, the Return of Interest (ROI) equation as shown in Equation (18) is utilized, the net benefit of this system can be determined by subtracting the total costs from the total benefits.

5.4 RECOMMENDATION

The proposed ALPR system in this project aims to enhance the efficiency and adaptability of ALPR systems. There are few recommendations focus on improving ALPR system which are continuous real-life data collection, optimization image processing flows, and the integration of a user interface.

One key responsibility for this system is keep collect real-life data and update the proposed Faster R-CNN model. This allows Faster R-CNN model trains on new data become a robust model that adapt in real-time. Hence, the improvement on license plate detection and characters recognition can be more advanced in refinement and optimization, improving the system's performance.

Optimizing of image processing flow also one of the recommendations for proposed ALPR system. It involves researching and exploring various image processing techniques to enhance the overall system efficiency. In addition, the parameter utilized in image processing techniques also crucial as different parameters can yield various results. By fine-tuning these parameters to obtain a suitable image processing flow and implement into the system to ensure have better performance on character recognition.

Integration a user-friendly interface is another important recommendation for the proposed ALPR system. Designing an intuitive, simple, convenient and user-oriented interface provides operators can interact seamlessly with the system. Features built in such as license plate number shown, accuracy of the model on detection and recognition performance and easy to control are basic components that contribute user experience.

REFERENCES

Albawi, S., Bayat, O., Al-Azawi, S., & Ucan, O. N. (2018). Social Touch Gesture Recognition Using Convolutional Neural Network. *Computational Intelligence and Neuroscience*, *2018*, 1–10. https://doi.org/10.1155/2018/6973103

Amon, M. C. E., Brillantes, A. K. M., Billones, C. D., Billones, R. K. C., Jose, J. A., Sybingco, E., Dadios, E., Fillone, A., Lim, L. G., & Bandala, A. (2019, November 1). Philippine License Plate Character Recognition using Faster R-CNN with InceptionV2. *2019 IEEE 11th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management, HNICEM 2019*. https://doi.org/10.1109/HNICEM48295.2019.9072753

Awalgaonkar, N., Bartakke, P., & Chaugule, R. (2021). Automatic License Plate Recognition System Using SSD. *2021 International Symposium of Asian Control Association on Intelligent Robotics and Industrial Automation, IRIA 2021*, 394–399. https://doi.org/10.1109/IRIA53009.2021.9588707

Batra, P., Hussain, I., Ahad, M. A., Casalino, G., Alam, M. A., Khalique, A., & Hassan, S. I. (2022). A Novel Memory and Time-Efficient ALPR System Based on YOLOv5. *Sensors*, *22*(14), 5283. https://doi.org/10.3390/s22145283

Benjdira, B., Khursheed, T., Koubaa, A., Ammar, A., & Ouni, K. (2019). Car Detection using Unmanned Aerial Vehicles: Comparison between Faster R-CNN and YOLOv3. *2019 1st International Conference on Unmanned Vehicle Systems-Oman (UVS)*, 1–6. https://doi.org/10.1109/UVS.2019.8658300

Bhujle, H., & Vadavadagi, B. (2018). Fast Non-local Means Denoising for MR Image Sequences. *2018 International Conference on Signal Processing and Communications (SPCOM)*, 177–181. https://doi.org/10.1109/SPCOM.2018.8724414

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). *YOLOv4: Optimal Speed and Accuracy of Object Detection*.

Brillantes, A. K., Billones, C. D., Amon, M. C., Cero, C., Jose, J. A. C., Billones, R. K. C., & Dadios, E. (2019, November 1). Philippine License Plate Detection and Classification using Faster R-CNN and Feature Pyramid Network. *2019 IEEE 11th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management, HNICEM 2019*. https://doi.org/10.1109/HNICEM48295.2019.9072754

Budak, L., Grbic, R., Cetic, N., & Kastelan, I. (2022). Color image segmentation based on thresholding for advanced driver assistance systems. *2022 IEEE Zooming Innovation in*

Consumer Technologies Conference (ZINC), 271–276. https://doi.org/10.1109/ZINC55034.2022.9840722

Bulan, O., Kozitsky, V., Ramesh, P., & Shreve, M. (2017). Segmentation- and Annotation-Free License Plate Recognition With Deep Localization and Failure Identification. *IEEE Transactions on Intelligent Transportation Systems*, *18*(9), 2351–2363. https://doi.org/10.1109/TITS.2016.2639020

Chang, S. L., Chen, L. S., Chung, Y. C., & Chen, S. W. (2004). Automatic License Plate Recognition. *IEEE Transactions on Intelligent Transportation Systems*, *5*(1), 42–53. https://doi.org/10.1109/TITS.2004.825086

Delmar Kurpiel, F., Minetto, R., & Nassu, B. T. (2018). Convolutional neural networks for license plate detection in images. *Proceedings - International Conference on Image Processing, ICIP*, *2017-September*, 3395–3399. https://doi.org/10.1109/ICIP.2017.8296912

El Ghmary, M., Ouassine, Y., & Ouacha, A. (2023a). *Automatic License Plate Recognition with YOLOv5 and Faster-RCNN* (pp. 351–361). https://doi.org/10.1007/978-3-031-28387-1_30

El Ghmary, M., Ouassine, Y., & Ouacha, A. (2023b). *Automatic License Plate Recognition with YOLOv5 and Faster-RCNN* (pp. 351–361). https://doi.org/10.1007/978-3-031-28387-1_30

Fakhrurroja, H., Pramesti, D., Hidayatullah, A. R., Fashihullisan, A. A., Bangkit, H., & Ismail, N. (2023). Automated License Plate Detection and Recognition using YOLOv8 and OCR With Tello Drone Camera. *2023 International Conference on Computer, Control, Informatics and Its Applications (IC3INA)*, 206–211. https://doi.org/10.1109/IC3INA60834.2023.10285750

Fan, R., Bocus, M. J., Zhu, Y., Jiao, J., Wang, L., Ma, F., Cheng, S., & Liu, M. (2019). *Road Crack Detection Using Deep Convolutional Neural Network and Adaptive Thresholding*.

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, *36*(4), 193–202. https://doi.org/10.1007/BF00344251

Ganchovska, V., & Krasteva, I. (2022). Converting color to grayscale image using LabVIEW. *2022 International Conference Automatics and Informatics (ICAI)*, 320–323. https://doi.org/10.1109/ICAI55857.2022.9960062

Gholamalinezhad, H., & Khosravi, H. (n.d.). *Pooling Methods in Deep Neural Networks, a Review*.

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 580–587. https://doi.org/10.1109/CVPR.2014.81

Gnanaprakash, V., Kanthimathi, N., & Saranya, N. (2021). Automatic number plate recognition using deep learning. *IOP Conference Series: Materials Science and Engineering*, *1084*(1), 012027. https://doi.org/10.1088/1757-899x/1084/1/012027

Golilarz, N. A., Demirel, H., & Gao, H. (2019). Adaptive Generalized Gaussian Distribution Oriented Thresholding Function for Image De-Noising. *International Journal of Advanced Computer Science and Applications*, *10*(2). https://doi.org/10.14569/IJACSA.2019.0100202

Harshitta, S., & Mr. Shrwan, R. (2018). A Literature Survey on Automatic License Plate Recognition System (ALPR)). *International Journal of Research In Electronics and Computing Engineering*, *6*(3), 2054–2058.

Hendry, & Chen, R.-C. (2019). Automatic License Plate Recognition via sliding-window darknet-YOLO deep learning. *Image and Vision Computing*, *87*, 47–56. https://doi.org/10.1016/j.imavis.2019.04.007

Heng Li, M., & Yu Zhang, M. (2020). Computational Benefits, Limitations and Techniques of Parallel Image Processing. *Journal of Medical and Image Computing*, 1–9. https://doi.org/10.46532/jmic.20200701

Ide, H., & Kurita, T. (2017). Improvement of learning for CNN with ReLU activation by sparse regularization. *Proceedings of the International Joint Conference on Neural Networks*, *2017-May*, 2684–2691. https://doi.org/10.1109/IJCNN.2017.7966185

Janiesch, C., Zschech, P., & Heinrich, K. (n.d.). *Machine learning and deep learning*. https://doi.org/10.1007/s12525-021-00475-2/Published

Jie, H. J., & Wanda, P. (2020). RunPool: A Dynamic Pooling Layer for Convolution Neural Network. *International Journal of Computational Intelligence Systems*, *13*(1), 66. https://doi.org/10.2991/ijcis.d.200120.002

Jubayer, F., Soeb, J. A., Mojumder, A. N., Paul, M. K., Barua, P., Kayshar, S., Akter, S. S., Rahman, M., & Islam, A. (2021). Detection of mold on the food surface using YOLOv5. *Current Research in Food Science*, *4*, 724–728. https://doi.org/10.1016/j.crfs.2021.10.003

Juyal, A., Tiwari, H., Singh, U. K., Kumar, N., & Kumar, S. (2023). Dental Caries Detection Using Faster R-CNN and YOLO V3. *ITM Web of Conferences*, *53*, 02005. https://doi.org/10.1051/itmconf/20235302005

Kaur, G., Jaiswal, A. K., Kumar, R., & Thakur, K. (2023). Automatic License Plate Recognition System. *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 1–6. https://doi.org/10.1109/ICCCNT56998.2023.10307008

Kim, B.-G., Kang, S.-H., Park, C. R., Jeong, H.-W., & Lee, Y. (2020). Noise Level and Similarity Analysis for Computed Tomographic Thoracic Image with Fast Non-Local Means Denoising Algorithm. *Applied Sciences*, *10*(21), 7455. https://doi.org/10.3390/app10217455

Kim Hui, Y., Ling, G., Stuart William, P., & Hau San, W. (2018). *Adaptive Image Processing: A Computational Intelligence Perspective, Second Edition Image Processing Series* (2nd ed.). CRC Press.

Kim, J., Sung, J.-Y., & Park, S. (2020). Comparison of Faster-RCNN, YOLO, and SSD for Real-Time Vehicle Type Recognition. *2020 IEEE International Conference on Consumer Electronics - Asia (ICCE-Asia)*, 1–4. https://doi.org/10.1109/ICCE-Asia49877.2020.9277040

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84–90. https://doi.org/10.1145/3065386

Kulkarni, U., Agasimani, S., Kulkarni, P. P., Kabadi, S., Aditya, P. S., & Ujawane, R. (2023). Vision based Roughness Average Value Detection using YOLOv5 and EasyOCR. *2023 IEEE 8th International Conference for Convergence in Technology (I2CT)*, 1–7. https://doi.org/10.1109/I2CT57861.2023.10126305

Kumar Prajapati, R., Nagar, T., Dangi, S., Bhardwaj, Y., Raj, P., Rao, S., Ritesh, &., & Jain, K. (2023). Automatic Number Plate Recognition using YoloV7 and PaddleOCR. *International Advanced Research Journal in Science*, *10*(2).

Kundrotas, M., Janutėnaitė-Bogdanienė, J., & Šešok, D. (2023). Two-Step Algorithm for License Plate Identification Using Deep Neural Networks. *Applied Sciences (Switzerland)*, *13*(8). https://doi.org/10.3390/app13084902

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278–2324. https://doi.org/10.1109/5.726791

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

Lin, C. J., Chuang, C. C., & Lin, H. Y. (2022). Edge-AI-Based Real-Time Automated License Plate Recognition System. *Applied Sciences (Switzerland)*, *12*(3). https://doi.org/10.3390/app12031445

Mohta, A., Swaroop, A., Fhanindra Reddy, K., & R, M. (2023). Automatic License Plate Recognition System Using YOLOv4. *International Research Journal on Advanced Science Hub*, *5*(Issue 05S), 280–286. https://doi.org/10.47392/irjash.2023.S038

Onishchenko, D., Liubchenko, N., & Podorozhniak, A. (2023). LICENSE PLATE RECOGNITION SYSTEM BASED ON MASK R-CNN. *Automation of Technological and Business Processes*, *15*(3), 37–43. https://doi.org/10.15673/atbp.v15i3.2623

Paglinawan, C. C., Hannah M. Caliolio, M., & Frias, J. B. (2023). Medicine Classification Using YOLOv4 and Tesseract OCR. *2023 15th International Conference on Computer and Automation Engineering (ICCAE)*, 260–263. https://doi.org/10.1109/ICCAE56788.2023.10111387

Palai, C., & Jena, P. K. (2015). *Automatic Vehicle Identification: LPR with Enhanced Noise Removal Technique* (pp. 143–153). https://doi.org/10.1007/978-81-322-2205-7_14

Pasrun, Y. P., Muchtar, M., Basyarah, A. N., & Noorhasanah. (2020). Indonesian License Plate Detection Using Morphological Operation. *IOP Conference Series: Materials Science and Engineering*, *797*(1), 012037. https://doi.org/10.1088/1757-899X/797/1/012037

Peng, C.-C., Tsai, C.-J., Chang, T.-Y., Yeh, J.-Y., Dai, H., & Tsai, M.-H. (2020). A Fast and Noise Tolerable Binarization Method for Automatic License Plate Recognition in the Open Environment in Taiwan. *Symmetry*, *12*(8), 1374. https://doi.org/10.3390/sym12081374

Randive, P. S., Bansod, S., Ahivale, S., Mohite, S., & Patil, S. (2016). Automatic License Plate Recognition [ALPR] System. *International Journal of Engineering Trends and Technology*, *35*(5), 224–227. https://doi.org/10.14445/22315381/IJETT-V35P248

Ravirathinam, P., & Patawari, A. (2019). Automatic license plate recognition for indian roads using faster-RCNN. *Proceedings of the 11th International Conference on Advanced Computing, ICoAC 2019*, 275–281. https://doi.org/10.1109/ICoAC48765.2019.246853

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2015). *You Only Look Once: Unified, Real-Time Object Detection*. http://arxiv.org/abs/1506.02640

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779–788. https://doi.org/10.1109/CVPR.2016.91

Ren, S., He, K., Girshick, R., & Sun, J. (2015). *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*. http://arxiv.org/abs/1506.01497

Reswara, E., Suakanto, S., & Putra, S. A. (2023). Comparison of Object Detection Algorithm using YOLO vs Faster R-CNN : A Systematic Literature Review. *Proceedings of the 2023 6th International Conference on Big Data Technologies*, 419–424. https://doi.org/10.1145/3627377.3627443

Sarika, N., Sirisala, N., & Velpuru, M. S. (2021). CNN based Optical Character Recognition and Applications. *2021 6th International Conference on Inventive Computation Technologies (ICICT)*, 666–672. https://doi.org/10.1109/ICICT50816.2021.9358735

Saxena, S., Sharma, S., & Sharma, N. (2016). Parallel Image Processing Techniques, Benefits and Limitations. *Research Journal of Applied Sciences, Engineering and Technology*, *12*(2), 223–238. https://doi.org/10.19026/rjaset.12.2324

Selmi, Z., Ben Halima, M., & Alimi, A. M. (2017). Deep Learning System for Automatic License Plate Detection and Recognition. *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, 1132–1138. https://doi.org/10.1109/ICDAR.2017.187

Sham, A. S. D., Pandey, P., Jain, S., & Kalaivani, S. (2021). AUTOMATIC LICENSE PLATE RECOGNITION USING YOLOV4 AND TESSERACT OCR. *INTERNATIONAL JOURNAL OF ELECTRICAL ENGINEERING AND TECHNOLOGY*, *12*(5). https://doi.org/10.34218/IJEET.12.5.2021.006

Smith, R. (2007). An Overview of the Tesseract OCR Engine. *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007) Vol 2*, 629–633. https://doi.org/10.1109/ICDAR.2007.4376991

Terven, J., & Cordova-Esparza, D. (2023). *A Comprehensive Review of YOLO: From YOLOv1 and Beyond*. http://arxiv.org/abs/2304.00501

Tripathi, S., & Jain, S. (2021). Automatic Number Plate Recognition System (ANPR): The Implementation Automatic Number Plate Recognition System (ANPR): The Implementation View project. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, *10*, 2278–3075. https://doi.org/10.35940/ijitee.H9257.0610821

Vadlamani, R., & Patel, A. V. (2021). *Deep Convolutional Net*. https://www.researchgate.net/publication/350487754

Wang, B., Xu, J., Li, J., Hu, C., & Pan, J.-S. (2017). Scene text recognition algorithm based on faster RCNN. *2017 First International Conference on Electronics Instrumentation & Information Systems (EIIS)*, 1–4. https://doi.org/10.1109/EIIS.2017.8298720

Wang, W., Yang, J., Chen, M., & Wang, P. (2019). A Light CNN for End-to-End Car License Plates Detection and Recognition. *IEEE Access*, *7*, 173875–173883. https://doi.org/10.1109/ACCESS.2019.2956357

Xie, R., Li, C., Zhou, X., & Dong, Z. (2023). Asynchronous Federated Learning for Real-Time Multiple Licence Plate Recognition Through Semantic Communication. *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5. https://doi.org/10.1109/ICASSP49357.2023.10097251

Zhang Baohua, Yu Dahua, Huang Hongmei, & Gao Lanying. (2010). License plate location algorithm based on histogram equalization. *2010 International Conference On Computer Design and Applications*, V1-517-V1-519. https://doi.org/10.1109/ICCDA.2010.5540710

Zhang, X., Li, M., Liu, X., Zhou, Z., Wang, X., & Xu, Z. (2022). An improved wavelet denoising algorithm for SAR interferogram using fast non-local means filtering. *Geocarto International*, *37*(27), 18600–18617. https://doi.org/10.1080/10106049.2022.2142967