DATA-CENTRIC DEEP LEARNING TECHNIQUE FOR ROBUST AUTOMATIC LICENSE PLATE RECOGNITION (ALPR)

BY

AHMED ABDULHAKIM MOHAMMED ASAAD

A dissertation submitted in fulfilment of the requirement for the degree of Master of Science (Mechatronics Engineering)

Kulliyyah of Engineering
International Islamic University Malaysia

JULY 2023

ABSTRACT

Automatic License Plate Recognition (ALPR) has become a common study area because of its many practical applications, such as automatic toll collection and traffic law enforcement. However, most existing methods of Malaysian ALPR are not robust enough to be used in everyday situations. The lack of high-quality benchmarked datasets that accurately represent real-world complexities in Malaysian license plates (LP) and the absence of a comprehensive dataset to demonstrate system robustness is a significant limitation. In addition, the reliance on shallow techniques in the studies on Malaysian ALPR causes inefficiency of the systems, particularly in handling complicated scenarios involving different image backgrounds and variations in LP size or shape and also the non-standard LPs. This dissertation presents a robust Malaysian ALPR system based on the single-shot detector You Only Look Once (YOLO) in two stages; license plate detection (LPD) and license plate recognition (LPR). The system is designed by evaluating and optimizing different models with different dataset optimization to achieve the best speed versus accuracy trade-off in ALPR system. The models are trained using a large-scale dataset containing images from several places around Malaysia, with the addition of data augmentation techniques to make them robust under various circumstances (e.g., with variations in lighting, camera position and settings, and license plate types). A dataset augmentation has also been accomplished by systematically generating a large, controlled synthetic dataset. The purpose is to achieve a balanced dataset and ensure the robustness of the dataset in terms of variations that exist in Malaysian license plates in the form of non-standardized license plates and special license plates. Thus, this work introduces a dataset for Malaysian ALPR with more than 176,000 images from real-world scenarios and synthetically produced images covering various aspects. This dataset will be public to the research community. The name of this dataset is Malaysian Number Plate and in short (MYNO). This dataset can be used for further training and evaluation of ALPR models. A separate challenging dataset is created for testing the models. Many experiments are carried out in detail with different models, data size, number of epochs, and real and synthetic datasets. When adding the synthetic dataset, the system performed better with 97.6% mAP compared to 85.5% mAP for the only real-world dataset at the same number of epochs. The proposed system achieved a recognition rate of 98.1% mAP on a real-world dataset collected from different toll plazas around Malaysia containing comprehensive environment distinctions with over 50 thousand labeled images. The system was tested on a challenging test dataset with low visibility and an unconstrained environment, resulting in 95.96% end-to-end accuracy. The results demonstrate the significance of incorporating synthetic datasets into the training process for improved performance in ALPR systems. The inclusion of a synthetic dataset led to a substantial increase in mean average precision (mAP), with a notable improvement of 12.1% when combined with the synthetic dataset. The system showcases its effectiveness in handling diverse environmental conditions by achieving 98.1% mAP on a real-world dataset collected from various toll plazas in Malaysia. In addition, achieving an impressive end-to-end accuracy of 95.96% despite low visibility further validates the system’s performance on challenging dataset.

خلاصة البحث

أصبح التعرف التلقائي على لوحات المركبات (ALPR) موضوعاً بحثياً شائعاً بسبب تطبيقاتها العملية الواسعة ، مثل إسهامها في نظام إنفاذ القوانين المرورية و نظام تحصيل الرسوم على الطرقات السريعة. ومع ذلك لا تزال معظم الأنظمة الحالية غير متينة بما فيه الكفاية لتتعامل مع الظروف المختلفة في الواقع العملي. يعد الافتقار إلى بيانات مرجعية عالية الجودة بحيث تمثل بدقة التعقيدات الموجودة في لوحات المركبات الماليزية (LP) قصوراً كبيرًا ، وكذلك عدم وجود بيانات شاملة تستخدم لإثبات كفاءة النظام. بالإضافة إلى ذلك ، فإن الاعتماد على التقنيات السطحية في الدراسات الخاصة بأنظمة ALPR في ماليزيا يسبب قصوراً في الأنظمة، لا سيما في التعامل مع السيناريوهات المعقدة التي تتضمن خلفيات صور مختلفة وتنوعات في حجم أو شكل LP وأيضًا LPs غير القياسية. هذه الدراسة تقدم نظام تعرف تلقائي على لوحات المركبات الماليزية باستخدام نظام الكشف ذو اللقطة الأحادية (YOLO) ، وذلك على مرحلتين هما الكشف عن لوحة المركبة (LPD) والتعرف على هوية اللوحة (LPR). تمت عملية تصميم هذا النظام من خلال عملية التقييم للنماذج وتحسينها باستمرار عبر التحسين والتطوير للبيانات والصور المستخدمة للوصول إلى أفضل نموذج يأخذ بعين الاعتبار أفضل مقاربة بين السرعة والدقة كونهما متضادين عند التطبيق. أُجريت عملية تدريب النماذج باستخدام مجموعة بيانات واسعة النطاق تحتوي على صور مركبات من عدة أماكن حول ماليزيا ، مع إضافة تقنيات زيادة البيانات لجعلها قادرة على التعامل مع الظروف المختلفة (على سبيل المثال ، الاختلافات في الإضاءة ، وضعية الكاميرا أو إعداداتها ، والأنواع المختلفة من اللوحات). كما تم أيضًا مضاعفة البيانات من خلال إنشاء مجموعة بيانات كبيرة مبنية بطريقة منهجية ، الغاية منها هي أن تكون البيانات المستخدمة متوازنة ومتضمنة للمتغيرات الموجودة في لوحات المركبات الماليزية مثل اللوحات الغير ممتثلة للقواعد أو اللوحات ذات التعريفات الخاصة والتي تعتبر نادرة. وبالتالي ، فإن البيانات المنشأة للنظام المقترح تحوي أكثر من 176000 صورة من صور حقيقة وصور مُصطنعة تحوي تغيرات كثيرة و مختلفة. ستكون هذه البيانات متوفرة للباحثين المهتمين بصورة عامة وأطلقنا عليها اسم أرقام اللوحات الماليزية (MYNO). يمكن الاستفادة من هذه البيانات

لتدريب نماذج ALPR ، كما يمكن استخدام بيانات الاختبار الصعبة والتي هي جزء من هذه البيانات وجعلها مرجع في تقييم واختبار النماذج المصممة. لقد تم إجراء تجارب مختلفة ومفصلة لنماذج مختلفة ببيانات ذات حجم ونوع مختلف و عدد دورات تدريب مختلفة كذلك، بالإضافة لاختبار البيانات الحقيقية والاصطناعية. وجد أنه عند إضافة البيانات المركبة من الحقيقية والصناعية ، كان أداء النظام أفضل بنسبة 97.6٪ بمعيار (mAP) ، و بـ 85.5٪ للبيانات الحقيقية فقط على نفس العدد من دورات التدريب. حقق النظام المقترح معدل تعرف بنسبة 98.1٪ mAP على مجموعة البيانات الحقيقية تم جمعها من مواقع تحصيل رسوم مركبات في جميع أنحاء ماليزيا تحتوي على تغيرات بيئية واسعة مع أكثر من 50 ألف صورة. وقد تم اختبار النظام على بيانات اختبار معقدة عند رؤية منخفضة وبيئات غير مقيدة ، أسفرت عن نتيجة تمثلت بنسبة دقة كلية بمعدل 95.96٪. توضح النتائج أهمية دمج البيانات المصطنعة في عملية التدريب لتحسين الأداء في أنظمة ALPR. أدى إدراج مجموعة البيانات الاصطناعية إلى زيادة كبيرة في متوسط الدقة (mAP) ، مع تحسن ملحوظ بنسبة 12.1٪ عند دمجها مع مجموعة البيانات المصطنعة. يُظهر النظام فعاليته في التعامل مع الظروف البيئية المتنوعة من خلال تحقيق معدل mAP بنسبة 98.1٪ على مجموعة بيانات حقيقية تم جمعها من أماكن تحصيل رسوم مركبات مختلفة في ماليزيا. بالإضافة إلى ذلك ، فإن تحقيق دقة عالية لنظام كلي (من البداية للنهاية) بنسبة 95.96٪ على الرغم من ضعف الرؤية ، يثبت مزيداً من فعالية أداء النظام على بيانات معقدة.

APPROVAL PAGE

I certify that I have supervised and read this study and that in my opinion, it conforms to accepTable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Master of Science (Mechatronics Engineering).

…………………………………..
Hasan Firdaus Bin Mohd Zaki
Supervisor

…………………………………..
Ahmed Jazlan Bin Haja
Co-Supervisor

I certify that I have read this study and that in my opinion it conforms to accepTable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Master of Science (Mechatronics Engineering).

…………………………………..

Examiner

This dissertation was submitted to the Department of Mechatronic Engineering and is accepted as a fulfilment of the requirement for the degree of Master of Science (Mechatronics Engineering).

…………………………………..
Ali Sophian
Head, Department of Mechatronic Engineering

This dissertation was submitted to the Kulliyyah of Engineering and is accepted as a fulfilment of the requirement for the degree of Master of Science (Mechatronics Engineering).

……………………………………..
Sany Izan Ihsan
Dean, Kulliyyah of Engineering

DECLARATION

I hereby declare that this dissertation is the result of my own investigations, except where otherwise stated. I also declare that it has not been previously or concurrently submitted as a whole for any other degrees at IIUM or other institutions.

Ahmed Abdulhakim Mohammed Asaad

Signature .... Date 14 July 2023 ....

INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA

DECLARATION OF COPYRIGHT AND AFFIRMATION OF FAIR USE OF UNPUBLISHED RESEARCH

ROBUST AUTOMATIC LICENSE PLATE RECOGNITION (ALPR) SYSTEM AT LOW VISIBILITY USING DEEP NEURAL NETWORKS

I declare that the copyright holders of this dissertation are jointly owned by the student and IIUM.

No part of this unpublished research may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise without prior written permission of the copyright holder except as provided below

1. Any material contained in or derived from this unpublished research may be used by others in their writing with due acknowledgement.

2. IIUM or its library will have the right to make and transmit copies (print or electronic) for institutional and academic purposes.

3. The IIUM library will have the right to make, store in a retrieved system and supply copies of this unpublished research if requested by other universities and research libraries.

By signing this form, I acknowledged that I have read and understand the IIUM Intellectual Property Right and Commercialization policy.

Affirmed by Ahmed Asaad

……..…………………………..   14 July 2023 ………………………..

Signature   Date

This thesis is dedicated to my parents for laying the foundation of what I turned out to be in life.

ACKNOWLEDGEMENT

In the beginning foremost, I praise Allah Almighty, The most Beneficent, as he alone has made this journey easy for me from the beginning until the end. Although it has been tasking, His Mercies and Blessings on me eased the herculean task of completing this thesis. Alhamdullilah.

I would like to express my sincere gratitude to my supervisor Assoc. Prof. Dr. Hasan Firdaus Bin Mohd Zaki for the continuous support of my Master's study, and for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. Despite his commitments, he took time to listen and attend to me whenever requested. The moral support he extended to me is in no doubt a boost that helped in building and writing the draft of this research work. I could not have imagined having a better supervisor and mentor for my Master's study. My thanks and gratitude also go out to my second supervisor my co-supervisors Assoc. Prof. Dr. Zulkifli Zainal Abidin and Ahmed Jazlan Bin Haja Mohideen, whom allowed me to continue my thesis until completion.

From the bottom of my heart, it was my sincere intention to dedicate this work to my loving Dad and Mom as without their undying support and tolerance of my being away from home, I wouldn't have been able to achieve my goals. With their continuous support and prayers for me every day, I am achieving this big achievement. My gratitude goes to my beloved wife for her great sacrifice and the patience to be away from her for years to achieve this success, and she was the best wife and supporter.

A special tribute goes out to my fellow lab mate in CUTe lab: Ahmed Rimaz for everything he helped me from the first day I started my research until the end of it. I am grateful to you, brother Rimaz, for your patience in guiding me in the right direction.

Once again, we glorify Allah for His endless mercy on us one of which is enabling us to successfully round off the efforts of writing this thesis. Alhamdulillah

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
ALPRL	Ength of a Square Plate (m, in.)
AOLP	Application-oriented License Plate
BA	Bee Algorithm
BB	Bounding Box
BoW	Bag of Words
BRNN	Bidirectional Recurrent Neural Networks
CC	Connected Component
CCPD	Chinese City Parking Dataset
CNN	Convolutional Neural Network
COCO	Common Objects in Context
CPU	Central Processing Unit
CTC	Connectionist Temporal Classification
CUDA	Compute Unified Device Architecture
DL	Deep Learning
ETC	Electronic Toll Collection
FN	False Negative
FNN	Feedforward *Neural Network*
FP	False Positive
GPU	Graphics Processing Unit
GWHI	Gaussian Weighted Histogram Intersection
HOG	Histogram of Oriented Gradients
HSV	Hue Saturation Value
ID	Identification
IoU	Intersection Over Union
ITS	Young's Modulus in 1-direction (GPa,psi)
JPJ	Jabatan Pengangkutan Jalan
KNN	K-nearest Neighbor
LBP	Local Binary Pattern
LP	License Plate
LPD	License Plate Detection
LPR	License Plate Recognition
LSTM	*Long Short-term Memory*
mAP	mean Average Precision
ML	Machine Learning
MLFF	Multi-Lane Free Flow
MTCNN	Multi-Task Convolutional Neural Network
NMS	Num Maximum Suppresion
NN	Neural Network
OCR	Optical Character Recognition
OD	Object Detection
PANet	Path Aggregation Network
PASCAL VOC	PASCAL Visual Object Classes Challenge
PSNR	Peak Signal-to-Noise Ratio
RAM	Random Access Memory
R-CNN	Region-based Convolutional Neural Network

RFID	Radio-frequency identification
ROI	Region of Interest
RPN	Region Proposal Network
SIFT	Scale-Invariant *Feature* Transform
SPP	Spatial pyramid pooling
SSD	Single Shot Detector
SURF	Speeded-Up Robust Features
SVM	Support Vector Machine
TP	True Positive
TRDG	Text Recognition Data Generator
VEDA	Vertical Edge Detection Algorithm
VGG	Visual Geometry Group
VQ	Vector Quantization
XML	eXtensible Markup Language
YOLO	Yolu Only Look Once

CHAPTER ONE

INTRODUCTION

1.1 BACKGROUND OF THE STUDY

Automatic License Plate Recognition (ALPR) systems play a crucial role in many real-life applications. They are an essential part of intelligent transportation systems (ITS), where they are implemented to enhance traffic operations, traffic monitoring and law enforcement. ALPR is also used in vehicle parking systems and road toll payment (Janowski et al., 2014). Additionally, ALPR systems are used for safety purposes such as tracking stolen vehicles and vehicles used in criminal activity (Saini et al., 2019). Indeed, these implementations make ALPR systems to be significantly essential.

ALPR replaces human efforts to read License Plates (LP) for situations when manual LP recognition by humans is not the best option for situations where accuracy and efficacy are essential considerations. For example, highway traffic monitoring demands a system that can accurately monitor high-speed automobiles and high-volume traffic in real-time whereby human skills cannot provide. Furthermore, the enormous number of applications that use license plate recognition (LPR) and the requirement for 24-hour operations make ALPR the ideal choice for reducing high labor costs. In Addition, ALPR is integrated with other systems to perform tasks automatically. As a result, ALPR systems outperform manual alternatives in every way, including cost, efficiency, and time consumption.

ALPR systems ultimately aim to extract the alphanumeric characters on the LP of the vehicle and provide them as a text entry. The output from ALPR will be exploited in a further task that ALPR is integrated with. For instance, in the Electronic Toll Collection (ETC) system, parking lots management, identifying vehicles violating rules, and tracking vehicles used in crimes. Many computer vision techniques and approaches were used and experimented to develop the ALPR system. Generally, all these techniques and methods have the same pipeline, starting with using a camera to capture the scene, then applying a chosen technique and bringing the alphanumeric characters on the plate as an output. Figure 1.1 depicts the pipeline of the ALPR system. Approaches of ALPR systems vary from using fundamental image processing techniques and statistical classifiers to utilizing deep learning (DL) methods. Image processing techniques are based on features of the LP, such as edge, color and texture. However, in statistical classifiers, machine learning (ML) algorithms based on feature extraction, such as Haar-like features, Adaboost classifier and SVM classifier, are used. Deep learning methods are also widely used due to their high performance compared to image processing and statistical classifiers especially with the convolutional neural networks (CNN) which performed remarkably in deep learning.

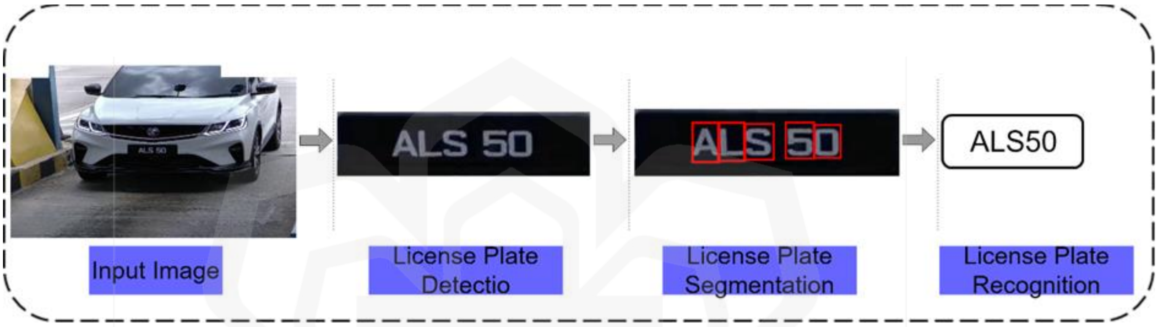


Figure 1.1 General ALPR System Pipeline

The diversity of applications that utilize ALPR systems and the different operation conditions related to those applications lead to difficulty in developing a generic solution for ALPR (Shashirangana et al., 2021). For that, we can see many ALPR proposed methods focusing on solving a specific problem in a constrained environment. Such issues are lightning changes, poor visibility due to light or motion blur or bad weather conditions, occlusion, various plate formatting, image shot angle and distance from the camera, camera resolution and complex backgrounds.

Mostly, methods used in designing ALPR consist of three main stages, namely, license plate detection and localization, character segmentation and character recognition. However, employed techniques that eliminate the need for segmentation and come up with segmentation-free ALPR systems.

In the LP detection stage, the car's front or rear license plate is extracted from the image. Many approaches are used for this step. For example, feature-based techniques utilize the traditional image processing approaches, ML classifiers and object detection (OD) methods with deep learning.

The second stage is the segmentation stage. This stage is popular in the optical character recognition (OCR) where number plate recognition is considered a case of OCR. In the segmentation process, the characters on the LP are separated from the whole plate and presented to the next stage: recognition. This process facilitates the characters recognition since the focus will be only on the separated regions on the LP. However, some ALPR systems do not include the segmentation stage especially that implement the deep learning OD methods as the LP recognition is considered a case of the OD. In the OD methods, the process of proposing regions of the potential objects will localize the object in the image and then a CNN used to extract features of the proposed regions in order to classify them.

The last stage is LP recognition. With LP recognition, the system reads and identifies all alphanumeric characters on the LP and presents them as an output text form. In case the segmentation stage is available, all segments are classified one by one using image processing methods, feature extractors classifiers, or deep learning. However, OD methods can be used without segmenting characters where the whole LP image is fed to the model, and the model brings out the objects (the characters in this case).

1.2 MALAYSIAN ITS PLAN AND MLFF SYSTEM

Malaysia has been working on a strategic plan to implement and develop an Intelligent Transport System (ITS) throughout the country by 2023. One of the primary objectives of the Malaysian ITS Blueprint is to achieve a congestion-free network. A significant reason causing the congestion on road transport is vehicles' long queues at the toll booths due to the increase in traffic. Even though the Electronic Payment Systems were 100% implemented in Malaysia in 1994, the focus is now to eliminate the delays on toll roads at toll booths to reduce congestion and hence have a faster journey. The plan is to have the infrastructure that helps reduce congestion, like implementing Multi-Lane Free Flow (MLFF). With MLFF, each car must pass through a certain toll gantry which picks up a personalized signal from a transponder embedded in the vehicle. It's a combination of Electronic Toll Collection (ETC), RFID, and ALPR to collect the charges of cars based on the detection, identification, and classification of vehicles. ALPR is incorporated with RFID to fulfill the system's smoothness and ensure the ETC's robustness.

Migration to Multi-Lane Free-Flow (MLFF) is in its seventh implementation phase and needs to integrate RFID with ALPR to minimize errors as a mandatory requirement.

Tampering with the RFID system will result in considerable losses in fee collection. Hence, ALPR with RFID ensures the robustness of the ETC. It will provide a solution for such a problem by verifying the vehicle’s identity and notifying relevant authorities if there is any evasion of paying the correct fees. Moreover, the ALPR would help law enforcement by exposing those who do not comply with the standard plate numbers or making the number plate invisible to the camera (Kerja Raya et al., 2019).

1.3 MALAYSIAN LICENSE PLATE STANDARD FORMAT

Malaysian License plates are issued and administered by Malaysian Road Transport Department (Malay: JPJ). All vehicle number plates in Malaysia have the same design with white characters and black backgrounds except for taxis, diplomats, and vehicle dealers (JPJ Malaysia, n.d.).

JPJ adopts two types of orientation for LP (Road Transport Department Malaysia JPJ, n.d.). The first orientation comes with one line containing the alphabets and then digits after. The second type of orientation has two lines: the top has alphabets, and the bottom includes numbers. Figure 1.2 shows the two types of License plates in Malaysia with other standardized specifications like dimensions of the plates and characters and spaces between characters and spaces from edges of the plate.

Figure 1.2 The Two Types of Orientations of LP in Malaysia with Standardized Specifications

Although the JPJ standards for Malaysian LP are clearly stated to the public, the public compliance to the stipulated rules is discouraging. To some extent, authorities force the drivers to follow the specifications of the standard format for LP, like color, layout, and character size, to have accurate identification and the best visibility. However, there is less strict on the LP dimensions (Wikipedia, n.d.). There is no specific font typeface that should be followed,, but the recommended one from JPJ is a compact version of Arial Bold. In fact, any kind of readable font is accepTable. Indeed, many drivers tend to choose fancy typefaces that are difficult to recognize and do not follow the specifications.

JPJ also organized the formatting of Malaysian LP throughout the country. All the Malaysian LPs except for diplomatic vehicles have the same formatting (Wikipedia, n.d.). The plates take the xxx #### format. The first three places from the left are alphabetical prefixes and two alphabetic sequences, and the next four are filled with number sequences. The most left alphabet depicts the unique character of the state or territory followed by the number sequence with the range 1-9999. For each state or territory, its prefix represents a series. Once the last number sequence is achieved (ex: W 9999), an alphabet sequence is added to the right of the state or territory prefix until it reaches the end of the sequence (WYY 9999). An extended series is added in case a specific series goes to its most number plate sequence to have more LPs for that series. The extended series takes the xxx #### x format, having the same sequence as the original one but adding a new alphabetical sequence to the right of the number sequence. When a series has the last original sequence (ex: WYY 9999), it rests the sequence by adding the new alphabetical sequence, i.e., W 1 A. The new series is updated once the original one ends. For Example, if the current LP is WYY 9999 A, next will be W 1 B, and so on.

Letters that are not used in the systematic format of Malaysian LP are I and O (Wikipedia, n.d.). This is because of the similarities to the numbers 1 and 0, which may lead to misidentification of these numbers and letters. Letter Z is only reserved for military vehicles in Malaysia. The format of Military vehicles' LPs has the same concept of sequence as other vehicle types LP with a slight difference in the formatting, which contains an alphabetical sequence with only two characters instead of three Zx ####.

The only difference in Malaysian LP format is in LP associated with the diplomatic, consular, and United Nations (UN) vehicles (Wikipedia, n.d.). Formatting of these LPs follows the form ##-##-xx. The first and second codes denote information related to the category, and the last constant suffix DC, CC, and UN refer to diplomatic corps, consular, and United Nations, respectively. Dashes split the three parts of the format.

A category also represents special number plates authorized by JPJ (Wikipedia, n.d.). Vehicle owners can get them at higher prices. Primarily, the special plates denote the manufacturer or special event. They can also refer to famous places like the name of the country (Malaysia), the capital (Putrajaya), or the initialism of some well-known universities in addition to unique alphabetical sequences such as UUU or YY.

Special number plates do not have a specific format, which is rare compared to the general LP. Therefore, getting enough datasets for these LPs to be used in training is a problem. Moreover, some special number plates use the unused letters I and O, and small letters can be found in some of them, resulting in a problem in representing these datasets well in the training process. Hence, accuracy is severely low for these classes in the testing stage. We have come to some strategies to tackle this issue which will be discussed in chapter three.

Table 1.1 A List of Special Number Plates in Malaysia
Source: (Wikipedia, n.d.)

Commemorative Plates of Malaysia	
Prefix	Notes
Malaysia	Malaysia number plate series
PROTON	Issued for certain Proton cars.
PERODUA	Issued for certain Perodua cars.
WAJA	Issued for Proton Waja cars.
Chancellor	Issued for Proton Chancellor cars.
Putra	Issued for Proton Putra cars.
Persona	Issued for Proton Persona cars.
Satria	Issued for Proton Satria cars.
Tiara	Issued for Proton Tiara cars.
Perdana	Issued for Proton Perdana cars.
LOTUS	Issued for Lotus cars.
KRISS	Issued for Modenas Kriss motorcycle.
Jaguh	Issued for Modenas Jaguh motorcycle.
NAZA	Issued for certain Naza vehicles.
Putrajaya	Issued in 1995 during the establishment of Federal Territory of Putrajaya.
SUKOM	Issued only during the 1998 Commonwealth Games which was held in Kuala Lumpur that year.
BAMbee	Issued only during the 2000 Thomas and Uber Cup which was held in Kuala Lumpur that year.
XIIINAM	Issued only during the 2003 13th NAM Summit which was held in Kuala Lumpur that year.
XOIC	Issued only during the 2003 10th OIC Summit which was held in Kuala Lumpur that year.
XXVIASEAN	Issued only during the 2005 11th ASEAN Summit which was held in Kuala Lumpur that year.
XXXIDB	Issued to members and participants of the 2005 30th Islamic Development Bank Annual Meeting.

1M4U	Issued in March 2013 to symbolise the 1Malaysia for Youth (IM4U)concept.
PATRIOT	Issued on 24 June 2015
PERFECT	Fund-raiser for the Table Tennis Association of Malaysia, only run from 'PERFECT 1' to 'PERFECT 100'
TTB	Transformasi Terengganu Baharu (New Terengganu Transformation) Issued in January 2016
NAAM	New Affirmative Action Movement
VIP	Visit Pahang Year 2017
RIMAU	Referred to Malayan Tiger Mascot in 2017 SEA Games for the ASEAN region hosting by Malaysia
IQ	IQ number plate series, IQ 1 to IQ 999
G	G series number plates by Lee Chong Wei Foundation
GG	Great Generation
G####G	'G1G' to 'G999G' number plate series
GT	Proposed by Kelab Eksplorasi 7 Benua Malaysia (KE7B)
GTR	Issued by Pertubuhan Suara Wanita Malaysia, range from 'GTR 1' to 'GTR 1000'
G1M	Issued to symbolise the Gagasan 1Malaysia concept.
GP	Issued for the cars used by officials during GrandPrix at Sepang International Circuit.
A1M	Yayasan Artis 1Malaysia, a foundation set up in support of local performing artists including actors, singers and musicians who have aged, deteriorated in health or deceased. Only run from 'A1M 1' to 'A1M 1000'
T1M	Transformasi 1 Malaysia, only run from 'T1M 1' to 'T1M 1000'
K1M	Kembara 1 Malaysia, only run from 'K1M 1' to 'K1M 100'
NBOS	National Blue Ocean Strartergy
Q	Introduced by the Football Association of Sarawak (FAS)
SAS	Issued on 5 September 2014 to symbolise KDYMM Sultan Ahmad Shah of Pahang (SAS) in conjunction with Sultan Ahmad Shah's 40th anniversary.

SAM	Saya Anak Malaysia
SMS	The SMS plate is in celebration of the Sultan of Kedah's two-term reign as the Yang di-Pertuan Agong
E	E number plate series
X	X number plate series
XX	XX number plate series
Y	Yayasan Nur Jauhar, the foundation aspires to provide aid and assistance to the underprivileged community in Sabah regardless of their background, religion or ethnic group.
YY	YY number plate series
YA	YA number plate series
U	The ‘U’ series – U 1 to U 9999 – allows for a short number plate with only one alphabet.
UU	UU number plate series
UUU	UUU number plate series
US	"Untuk Seniman" is brought about by the Malaysian Artistes Association (Seniman), only run from 'US 1' to 'US 1000'
UP	UP number plate series stands for Unique Plates, runs from ‘UP 1’ to ‘UP 999’
UA	UA number plate series
UM	Universiti Malaya number plate series
UTM	Universiti Teknologi Malaysia number plate series
UiTM	Universiti Teknologi Mara number plate series
UKM	Universiti Kebangsaan Malaysia number plate series
UUM	Universiti Utara Malaysia number plate series

1.4 STATEMENT OF THE PROBLEM

The Malaysian ITS Blueprint aims to implement the MLFF by 2023 to achieve congestion-free infrastructure. The MLFF system consists of two key components: RFID and ALPR systems. The deployment of ALPR is mandatory alongside RFID to address challenges and enhance the RFID experience. To achieve this successfully, the

ALPR system needs to exhibit strong resilience to effectively perform in real-time. However, there is no study could address the problem of Malaysian ALPR robustly.

Many challenges attribute to the limitations of these studies in developing robust ALPR systems in Malaysia. Firstly, the lack of high-quality benchmarked datasets that accurately represent the complexities of real-world license plates. This includes small and low-quality datasets, inadequate performance on challenging scenarios, and the absence of a comprehensive dataset to demonstrate system robustness. The Malaysian license plate format poses reliability issues, as the characters on the plates often exhibit variations in typefaces, spacing, and positions, deviating from the official guidelines. Moreover, many existing works on vehicle ALPR in Malaysia are proposed in constrained environments, such as an indoor area with good lighting conditions. However, the accuracy of such systems becomes significantly suboptimal when tested in unconstrained environments comprising of low visibility, abrupt illumination, and harsh weather condition.

Imbalanced datasets can significantly deteriorate network performance, leading to incorrect detection of less-represented classes. This imbalance is prevalent in LP datasets due to the way characters are allocated based on authorities' guidelines. Attempts to address this issue by adding or removing images become ineffective since both over-represented and under-represented classes can coexist within the same image.

Efficiency is a crucial requirement for implementing ALPR methods on embedded platforms. However, the state of art object detection-based methods which can bring better results are not exploited in ALPR systems in Malaysia. Most of the utilized techniques are shallow methods, which suffer from limitations when complicated scenarios are introduced to them such as the variation in LP size, shape or deformation. They also struggle with complex background images and non-standard LPs. In addition, the reliance on segmentation techniques like connected components (CC) or projection profiles as a prerequisite for recognition adds complexity to the system and makes it slower.

1.5 RESEARCH OBJECTIVES

This research aims:

1- To develop a comprehensive, customized Malaysian ALPR dataset using publicly available resources and a newly proposed synthetic custom dataset generation technique.

2- To implement a segmentation-free and two-stage hierarchical ALPR system.

3- To evaluate the proposed ALPR system on a realistic, challenging, and large-scale testing dataset for Malaysian LPs.

1.6 METHOD OF THE STUDY

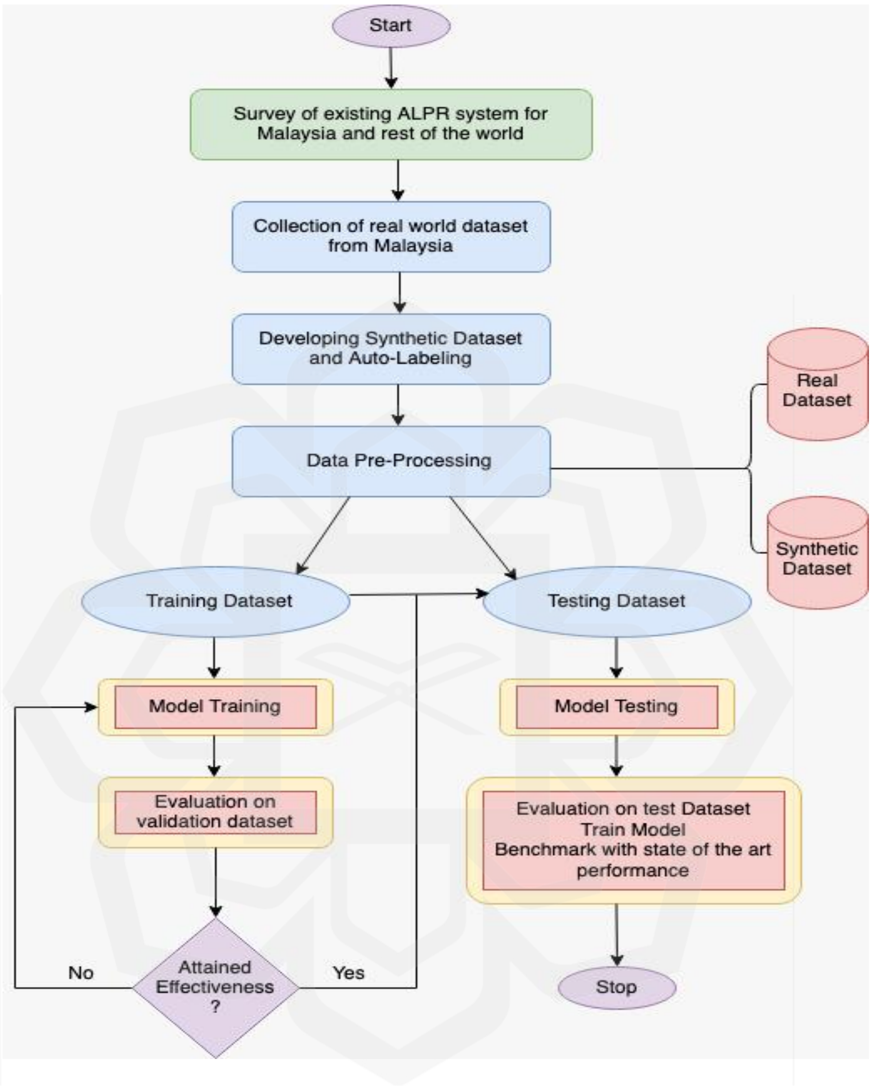
In order to achieve the objectives of the project, the methodology shown in Figure 1.3 is implemented. We started our work by surveying the literature review of ALPR systems in global as well as in Malaysia. After that, we collected a Malaysian real-world dataset that will be used to develop the ALPR system. We then worked on creating a synthetic dataset for Malaysian LPs to improve the system's accuracy. A pre-processing of the dataset is also performed for both real and synthetic datasets. Then, we used the training dataset to develop a deep learning-based ALPR system utilizing single-shot object detection (YOLO) and validated it on the validation dataset. We will test the model on the test dataset if the results are satisfactory. Otherwise, the process is repeated.

1.7 SIGNIFICANCE OF THE STUDY

For Electronic Toll Collection (ETC), robustness is essential. Therefore, ALPR - which is part of the system – must ensure that it is also robust. This study provides an ALPR system with high accuracy in an unconstrained environment. It also harnesses the powerful YOLO object detector, which works in real-time and provides high accuracy at the same time. Thus, the system is reliable to be implemented in the ETC system as it can prevent the tampering of toll payments and therefore save a lot of possible lost of charges.

The system can also solve the problem of detecting non-standard license plates, a widespread problem in Malaysia, by providing a well-representative dataset and conducting many training experiments to make the system robust against the non-standard license plates. This will contribute to the authorities' efforts to apprehend those who do not comply with the rules and then enforce the law.

Figure 1.3 Flow Chart of Research Methodology



1.8 SCOPE OF THE STUDY

The research pertains to a data-centric approach to solving the Malaysian ALPR problem. Particularly, as the data collection and annotation are expensive for real-world conditions due to a high degree of variations, a technique to efficiently augment the dataset is proposed. All possible fonts and typesetting that are used in the Malaysia license plate system are considered in this research. In situations where the fonts are not available in public resources or hard to find, the fonts are synthetically generated using the proposed technique. Several data augmentation techniques are also used to include more variations to the license plate to mimic the license plates available in the real world. Furthermore, this research does not consider the license plate dataset for other countries.

As the proposed method is data-centric, the main aim of this dissertation is not to improve the architecture or hyperparameters of the model. However, the newly proposed data-centric techniques inject robustness into the model.

1.9 THESIS ORGANIZATION

The conducted study is presented in this thesis in five chapters, of which the following is its organization.

Chapter one presents the introduction, which introduces an overview of the whole thesis. The background of the study, problem statement, research objective, scope and methodology are highlighted. In addition, an overview of the MLFF system and Malaysian LPs system is introduced.

Chapter two explores the existing literature on ALPR systems. In the beginning, methods used in ALPR systems are discussed for both stages, LPD and LPR, including the traditional and the deep learning-based approaches. Then, an exploration of the literature on the Malaysian ALPR systems is provided, highlighting the gap in these studies. After that, the datasets in the field of ALPR and the utilized algorithms are discussed.

In the third chapter, a discussion of the two stages of the proposed ALPR system is given in detail. Additionally, there is an elaboration on the dataset utilized in the proposed method. This includes the collection of the real-world dataset and the process of generating a synthetic dataset.

In chapter four, different experiments are presented to show how different data-centric strategies affect each stage in the proposed ALPR system. The results are presented, and a discussion of the conducted experiments is introduced.

Chapter five provides a discussion of how the study achieved all its objectives. Moreover, it gives a summary of the results and findings of the study with the enhancements intended in the future.

CHAPTER TWO

LITERATURE REVIEW

2.1 INTRODUCTION

The crucial need to replace the human effort and implement the ALPR system in the transport system, law enforcement and access control applications has led to a great attempt to design accurate and efficient ALPR systems. While different methods have been developed over time, varying from traditional image processing methods, statistical classifiers, and deep learning methods, they ultimately provide LP's unique ID number to use in further application after the camera captures the vehicle image.

This chapter covers the work that has been done on ALPR systems and their related topics in the literature. Firstly, the techniques employed in the field are explored with a high focus on deep learning-based studies of the OD. Then, the studies on Malaysian ALPR systems and datasets used are presented with a comparison. Finally, the theory of the used method (OD using deep learning) and an overview of the algorithm utilized (YOLOv5) are explained.

2.2 ALPR METHODS

This section discusses the different methods utilized to develop the automatic license plate recognition systems. It also lays the shade on the weaknesses and strengths of those methods. The section will be divided into subsections, where each one talks about one stage of the ALPR system pipeline.

2.2.1 License Plate Detection LPD

LPD is the process of detecting the license plate in the input vehicle image by localizing and classifying it. Localization of the LP is accomplished by finding the position of any LP in the image, whereas the classification of the LP means assuring that the localized LP is an LP by assigning it the proper class name. This stage is critical to efficiently identify and recognize the LP as it makes the system focus on a specific area in the whole image called the region of interest (ROI). Failing in this step will lead to complete process failure. The LP is detected by searching for its features in the image rather than scanning the whole image pixel by pixel (Du et al., 2013). These features can be edges, color, and texture. Based on the hand-crafted feature extraction, many studies with different techniques and approaches have been conducted to locate the LP, which are explored in the following subsections.

2.2.1.1 Traditional Methods

Conventional image processing techniques and algorithms have been used to detect the LPs. Some of the studies applied edge-based methods to find the edges boundaries of the LPs since they have distinct shapes appearing on the vehicle and fixed aspect ratio of its horizontal and vertical sides. Some studies such as Busch et al. (1998), Luo et al. (2009), Sanyuan et al. (2004a) and Sarfraz et al. (2003a) used this method by using the Sobel filter to find the edges. Hough transformation and contouring algorithms have been used (Duan et al., 2005). Some researchers have used other edge detection approaches like the Gabor filter (Pechiammal & Renjith, 2017), Vertical Edge Detection Algorithm (Dev, n.d.) and Canny Edge-detection (Mousa, 2012). Edge-based approaches have the advantage of simplicity and high speed. However, they are sensitive to other edges that may appear in the image. They are ineffective with low-quality images unless they use morphology techniques to improve image quality, which may result in high computational costs.

Another feature of the license plate is exploited, which is the color combination of characters and background of the license plate. Also, the image's color contrast between the license plate and the vehicle is mostly clear. In this regard, Gaussian Weighted Histogram Intersection algorithm is presented by W. Jia et al. (2006). This method matches two colors by their color histogram. A Gaussian function is added to the histogram intersection to take over the sensitivity due to illumination. F. Wang et al. (2008) proposed a fuzzy logic-based technique to detect the color of LP despite the variation in illumination. The HSV color space is used to extract color features, and the components of HSV (hue, saturation and value) are mapped in the fuzzy sets in terms of different membership functions. The color-based method is very good at detecting deformed license plates or those with an inclined position. However, it is often combined with another method because it has some drawbacks when used alone. Such drawbacks are the sensitivity to illumination and the different specifications of used cameras.

The texture feature of the license plate is used to detect the LP. LP image with grayscale has a significant difference between the character color of LP and the background of the license plate, which leads to noTable transition color and high edge density in the LP. The study conducted by Sferle & Moisi (2019) used Local Binary Pattern (LBP) and Histogram of Oriented Gradients (HOG) for number plate detection. The histogram is calculated, and the texture is classified using LBP and HOG, taking into account the rectangular shape of the LP. Vector Quantization (VQ) based method is proposed by Zunino and Rovetta (2000). This method does not rely on the features like edges and contrast. Instead, the actual content of the LP is considered. Anagnostopoulos et al. (2006) proposed the sliding concentric window, which uses the texture irregularities in the image. Candidate regions are selected in case there is a sudden change in the image. This method reported 96.5% accuracy in LP detection. Wavelet transform is used by Y. R. Wang et al. (2011) to locate the license plate. Although the texture-based approach is robust in detecting deformed LP, they are computationally expensive when there is illumination variation or complex background.

2.2.1.2 Machine Learning Based Methods

Machine learning-based classifiers are utilized in the field of LP detection. These classifiers are trained on the extracted features in the image using hand-crafted feature extraction methods like HOG and SIFT. Q. Wu et al. (2006) used Haar-like features and statistical features to train Adaptive Boosting (AdaBoost) classifier for license plate detection. This approach achieved 94.5% detection accuracy for LP. Support Vector Machine (SVM) classifier is also used in many studies to classify a region as a license plate or non-plate based on its color and texture features. A two stages classifier method is proposed by Ho et al. (2009). In the first stage, the possible character regions are proposed by thirty-six AdaBoost classifiers and then passed to another classifier which utilizes an SVM classifier trained on SIFT descriptors.

2.2.1.3 Deep Learning Based Methods

A four layers CNN network is utilized by Selmi et al. (2018) to classify input images to LP or non-LP after applying some pre-processing operations on the input image to help the network to perform well. Then, the bounding boxes of the possible LP are cropped from the image classified as LP with a prediction threshold value of 0.7. The drawback of this study is that it requires many pre-processing steps to prepare input images for the CNN network, which consumes much time in the execution. In addition, it fails when two LPs appear on the same image.

Anson and Mathew (2020) used a similar idea to the previous study with additional CNN to classify characters. First, the LP is detected by plate and non-plate CNN classifiers. The cropped LP is fed to a network that scans the image to classify characters and non-character regions. Non-character regions are eliminated, and characters are recognized.

Wang et al. improved the cascaded Multi-Task CNN (MTCNN) algorithm to detect the LP (W. Wang et al., 2019). The cascaded network comprises three layers: P-Net, R-Net and O-Net. The first layer is to get the LP candidates and vector of the bounding box (BB). The second layer fine-tunes the acquired candidates and removes overlapping windows. While removing overlapping windows and selecting the high-scored candidate by non-maximum suppression (NMS) approach, the last layer outputs the LP and presents the coordinates of the LP. This approach balanced the detection accuracy and real-time requirements with the end-to-end network.

Object detection algorithms like R-CNN, SSD and YOLO are also highly utilized in the LPD and LPR. The enhanced Fast R-CNN called Faster R-CNN, which adopts the Region Proposal Network (RPN) and Fast R-CNN is leveraged by Huang Z.-K. and Hou L.-Y. (2018) and Singh J. and Bhushan B. (2019). This approach achieves high accuracy but fails to get results in real-time as Faster R-CNN, comprised of two CNN networks, works separately.

YOLO algorithm is another object detection algorithm which can tackle this problem. It provides a very fast actual time performance with satisfying accuracy. Many researchers took advantage of YOLO and its versions in the field of ALPR. Hsu G.-S. et al. (2017) customized YOLO and YOLO-9000 as they were not initially designed to detect small objects. They modified the grid size from 7X7 to 11X11 and made every grid predict only one object. The modified version reaches 54 FPS compared with 45 in the original version and with better accuracy. A solution to manage the rotation problem during the detection of LP is presented by Xie et al. (2018). A modified YOLO (MD-YOLO) could predict LP rotation angle by adding angle information to the object parameters in YOLO. A new approach for computing the intersection over union area (IoU) for two rotational rectangles was also proposed. This study used a prepositive YOLO network before the MD-YOLO to work as attention to the LP region. This could help increase accuracy, but this will result in expensive time as two networks must be trained and performed separately.

Several studies used YOLO to detect LP without refinement (S. Abdullah et al., 2018; Min et al., 2019). The performance of these studies depends on the training dataset's quality, which should contain enough variations to tackle issues related to LP detection.

2.2.2 License Plate Recognition

This stage is responsible for "reading" the license plate after the detection stage succeeds in locating it. This is a specific case of optical character recognition that considers specific features on the license plate. For example, many countries have strict regulations regarding the font and color of the license plate, which are usually chosen to be easy to read. However, there are some peculiar issues with license plates (Saha, 2019). For example, because the image is captured outdoors, the system designers must account for variables such as variable ambient light, uneven brightness, and weather effects. Even with the standard license plate as they could be damaged or rotated.

Many optical character recognition approaches perform segmentation on the characters before recognizing them as a part of the multi-stage ALPR systems. Techniques for segmenting license plate characters take advantage of contrasting colors for the background and characters. To facilitate this separation, binarization of the image is performed since it results in opposite "colors" to the foreground (character) and background pixels. Among the segmentation method, component connectivity, projection profiles and deep learning are the most used methods.

Pixel connectivity is a simple character segmentation method (Omran & Jarallah, 2017; B. F. Wu et al., 2007). Based on their connectivity, the character pixels in the binarized image are identified, and their aspect ratio and size are compared to those of the number plate characters. Methods based on pixel connection are easy to deploy and robust against license plate rotation. However, this method appears to be useless for linked or broken characters. Segmentation using the projection profile method utilizes the fact that the character and background pixels in the license plate after image binarization have contrasting colors (Sanyuan et al., 2004b). Project-based techniques are sensitive to image quality and noise. Character segmentation using neural networks has been used recently. Characters are detected and highlighted by bounding boxes by CNNs from the LP input image (Silva & Jung, 2017). However, the segmentation of the characters using CNN can be combined with the recognition stage to reduce the model parameters (Tourani et al., 2020).

2.2.2.1 LPR Using Traditional Methods

Once the characters are segmented, the classification of these characters is performed. Usually, the learning model receives inputs of a fixed-sized. Thus, since they have variable sizes, the segmented input characters are rescaled before classification. Because the characters are limited and known, they are individually classified by one of the classification methods. Using template matching approaches to classify characters on license plates is popular due to the fact that the font and character size are typically known (Sarfraz et al., 2003b). The utilization of template matching was introduced by Yogheedha et al. (2018) and Y. Jia et al. (2016). Template matching technique is used with binarized images. It uses predefined samples of the characters on the license plate. Each segmented character on the detected image is compared to the samples to find the similarities. Template matching is simple but inefficient if the segment has a different size or shape or is deformed due to rotation. Thus, this method is not robust and cannot be generalized.

2.2.2.2 Machine Learning Based Methods

Recent studies have also explored the use of statistical machine learning classifiers for the recognition stage in ALPR system. These methods are based on the idea of using statistical models to learn the features in the data and extract them, which can then be used to make predictions about new, unseen data.

Several algorithms have been used to classify the extracted features SVM and HNM (Li et al., 2018; Llorens et al., 2005). Primarily, the feature vector of the character is generated using feature extraction methods like the Gabor filter and Kirsh edge detection. The Gabor filter is used by Hu et al. (2002) to achieve character extraction. The highest filter response will be found on character edges with the same orientation angle as the filter. It can be used to create character-specific characteristic vectors. While Kirsch edge detection is used by S. Abdullah et al. (n.d.) to extract features from the character images with various directions.

2.2.2.3 Deep Learning Based Methods

CNNs have been used in many studies to recognize the LP, such as by Selmi et al. (2018). A customised CNN with four convolutional layers and two fully connected layers is used to predict the segmented characters from LP images. With some modification in kernels and applying dropout of 0.5 in the fully connected layers, they could reach test results up to 94.8% in the Caltech dataset and an average accuracy of 95.5% in the AOLP dataset. However, this approach fails in some conditions, such as bad weather conditions. Bidirectional recurrent neural networks (BRNN) and long short-term memory (LSTM) are used to recognize the detected LP Anson & Mathew (2020); W. Wang et al., 2019). First, features are extracted from the preprocessed cropped LP and fed to the BRNN for sequence recognition. In the sequence labeling, BRNN predicts the next character while carrying the current and previous ones as inputs. The network keeps processing the whole sequence in the same manner until the end, and predictions are given as probability values. As RNN cannot hold information in its memory for a long time, LSTM is used, which can remember information for a longer time. The last step is obtaining the characters using connectionist temporal classification (CTC), which converts probability values to character strings.

2.3 YOLO ALGORITHM

YOLO (You Only Look Once) is a family of deep learning-based object detection models known for their fast and accurate performance. The YOLO models have evolved since the first version was released by (Redmon et al., 2016).

The YOLO models are based on a deep learning architecture known as a Convolutional Neural Network (CNN), which is trained on a large dataset of images to learn the features and patterns that are important for object detection. The CNN consists of multiple layers that are designed to extract and process information from the input image, as well as a final layer that is used to predict the locations and classes of objects in the image.

One of the critical features of the YOLO models is their ability to perform object detection in real-time, making them suiTable for applications that require fast and accurate predictions. This is achieved through the use of a single-stage object detection approach, which allows the model to make predictions about the locations and classes of objects in an input image in a single pass without the need for multiple stages or complex algorithms.

In addition to their fast performance, the YOLO models also offer high accuracy, with the ability to detect and classify a wide range of objects in various scenarios. This is achieved through a large, diverse training dataset and a carefully designed CNN architecture that can learn the complex patterns and relationships in the data.

Overall, the YOLO family of models is a powerful and efficient tool for real-time object detection. Their fast and accurate performance has made them a popular choice for a variety of applications that require fast and accurate object detection.

2.3.1 YOLO Versions

YOLOv1 divides the image into equal-sized S*S grid cells. If the object's center falls inside a grid cell, the cell is responsible for object detection. With a confidence score, each cell can predict a fixed B number of bounding boxes. Each bounding box is made up of five values: x, y, w, h, and a confidence score. x, y, w, and h are the bounding box's centre, width, and height, respectively. Following the prediction of a bounding box, YOLO employs IoU to select the appropriate bounding box of an object for the grid cell. YOLO uses NMS to remove excess bounding boxes. NMX removes the extra bounding boxes with a low confidence score if IoU is more than 0.5. YOLO employs the sum of squared errors to determine the loss. To increase accuracy and lessen the overfitting issue, batch normalisation was applied to YOLOv2 along with convolutional layers (Redmon & Farhadi, 2017).

In YOLOv3 (Redmon & Farhadi, n.d.), Darknet19, which has problems identifying small objects, was replaced by Darknet53 to solve this issue. The algorithm's accuracy was significantly increased in that work by introducing Residual Block, Skip Connections, and Up-Sampling. The backbone of the feature extractors was once more modified in YOLOv4 to CSPDarknet53, which greatly accelerated and increased the algorithm's accuracy (Bochkovskiy et al., 2020). The most recent and least resource-intensive iteration of earlier YOLO algorithms, YOLOv5, uses the Pytorch framework rather than the Darknet framework (Jocher et al., 2021).

2.3.2 YOLOv5

YOLOv5 is distinct from earlier releases. Instead of Darknet, PyTorch is used. SPDarknet53 serves as the system's backbone. This backbone eliminates the redundant gradient information seen in large backbones and incorporates gradient change into feature maps, which speeds up inference, improves accuracy, and shrinks the model's size by reducing the number of parameters. It boosts the information flow using the path aggregation network (PANet) as a neck. Using a novel feature pyramid network (FPN) with numerous bottom-up and top-down layers, PANet has been adopted. Low-level features in the model are propagated better as a result. PANet improves the localization accuracy of the item by boosting localization in lower layers. Additionally, the head in YOLOv5 is identical to those in YOLOv4 and YOLOv3, which produces three distinct outputs of feature maps to provide a multi-scale prediction. Additionally, it aids in the model's improved ability to predict small to large objects. The image is passed through CSPDarknet53 for feature extraction before being fed to PANet for feature extraction. Finally, the results are generated by the YOLO layer. Figure 2.1 depicts the architecture of the YOLOv5l algorithm. The Focus layer is an evolution of the YOLOv3 structure. It takes the first three layers of YOLOv3 and replaces them with a single layer in

YOLOv5. C3 is made up of three convolution layers and a module that is cascaded by several bottlenecks. Spatial pyramid pooling (SPP) is a pooling layer used to overcome the network's fixed size constraint.

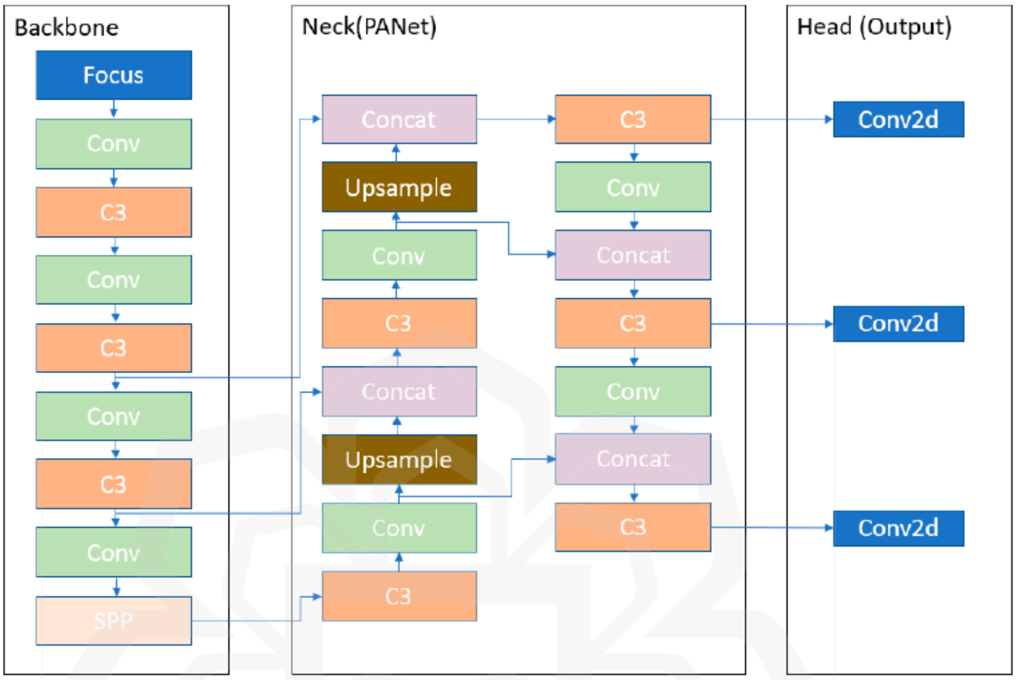


Figure 2.1 YOLOv5 Architecture

Source: (Nepal & Eslamiat, 2022)

The primary distinction between the YOLOv3, YOLOv4, and YOLOv5 architectures is the usage of Darknet53 as the backbone in YOLOv3. The YOLOv4 and YOLOv5 architectures use Focus structures with CSPdarknet53 as their backbone. In YOLOv5, the Focus layer is added for the first time, and it replaces the first three layers in YOLOv3, resulting in reducing the required CUDA memory, reduced layer and parameters, and increased forward propagation and backpropagation speed without a significant impact on the mAP.

2.4 ALPR FOR MALAYSIA

Many studies have been conducted to develop the ALPR systems on Malaysian LP. Similar to the other countries' ALPR systems, different approaches are applied to achieve better performance. Here we explore the related studies to the Malaysian ALPR systems on both license plate detection and license plate recognition. We will also be covering the limitations of the existing approaches.

Some researchers exploit image processing techniques to conduct the LPD stage on Malaysian LP. In this regard, the edge detection-based approach has been widely used to detect Asian LPs. The Sobel operator is used by Choong et al. (2020), Yaacob et al. (2021) and Chai et al. (2015). Weng Keong & Iranmanesh (2016) used a method to localize and crop the LP utilizing the Canny edge. Before applying edge detection, much preprocessing must be done to enhance the image and remove unwanted information. Then the region of interest is cropped utilizing horizontal and vertical projections. The projection profile was also used by Jalil et al. (2015).

Another study by Al-Ghaili et al. (2013) proposed a vertical edge-based algorithm which proved to be faster than the Sobel operator by five to nine times. This study has a rate successful rate of 91.4%.

Shatnawi (2018a) proposed an approach that utilizes the multilevel thresholding method using bees algorithms (BA) with Otsu thresholding and peak signal-to-noise ratio (PSNR) for preprocessing and segmentation. The BA and PSNR optimise the multilevel image thresholding on the grayscale image to obtain better-quality images with reduced processing time. Then Sobel operator detects the edges of the LP as a preliminary step to classifying the characters using SVM. Although the study could reach good accuracy when including special plates (95.43%), the dataset quality shown in the paper is ideal and straightforward without complicated real-world instances.

A hand-crafted feature extraction method is used by Khaleel et al. (2013). Featured extracted by Speeded Up Robust Features (SURF) and Bag of Words (BoW) and combined. K-means clustering is used to cluster the descriptors. The results showed high accuracy, with 98% in the webcam. However, test images which were shown appear that there is no complexity in the dataset.

The license plate is detected using a system with the AdaBoost method and connected components (CC) (Suandi et al., 2012). The CC is applied to reduce the significant false positives resulting from the AdaBoost classifier. M. Abdullah et al. (2017) proposed a vertical and horizontal scanning (smearing algorithm) on the image to locate the license plate after binarization of the image applied. The smearing algorithm also segments the LP before the recognition phase.

Malaysian LPD also has been conducted using deep learning for its powerful and effective performance. Two CNNs proposed to detect the LP on Malaysian LP. The first one is used to classify 37 classes in a sliding window to detect the characters in the image. After that, another CNN classifies the detected regions from the first network as plate/non-plate to eliminate the false positives. An accuracy of 97.58% was recorded. Liang et al. (2020) proposed an object detection approach to localize the Malaysian LP by applying region-based CNN ((R-CNN) using the AlexNet CNN. Eliminating false positives was considered by defining a high threshold value (0.92). Thy system recorded precision and record rate of 95.19% and 97.84%, respectively.

A study from Lee et al. (2022) focused on optimising YOLOv3 training parameters to adapt the model to work perfectly for the LPD task. The model's backbone CNN has been kept without any change. Parameters were involved in the study, including image aspect ratio, anchor number, learning rate, epochs number, penalty threshold and others. The effect of the parameters was examined through two different levels of experiments. In each experiment, the correlation and interaction of model parameters showed different impacts on the system's performance. This method improved the recognition rate of the license plate detection and achieved Average Precision (AP) of 99%.

In the license plate recognition stage, the approach of template matching was exploited by many researchers such as Jalil et al. (2015), M. Abdullah et al. (2017), Choong et al. (2020), Yaacob et al. (2021), Abu et al. (2019) and Yogheedha et al. (2018). Template matching technique is used with binarized images. It uses a predefined sample of the characters on the license plate. Each segmented character on the detected image is compared to the samples to find the similarities. Template matching is simple but inefficient if the segment has a different size or shape or is deformed due to rotation. Thus, this method is not robust and cannot be generalized. A study by Ng et al. (2015) used template matching with SIFT feature extractor. The advantage of using SIFT is that it is invariant to the changes that may occur by scaling and rotation by finding the relevant point (key points) in the image. The best matching of the key points in the input image with the template is found by calculating the Euclidean distance from the given descriptor vector. The recognition rate was 81% after testing with 500 images of 150 unique LPs. The high-blurring and low-resolution images caused false negative results as SIFT is not good at extracting features from such images.

Other studies implemented machine learning statistical classifiers in the LP recognition stage. SVM classifier is utilized by S. N. H. S. Abdullah et al. (2009) and Shatnawi (2018) to exploit its ability to classify problems with easy training and the trade-off between error and complexity of the classifier (Hao et al., n.d.). Although Shatnawi (2018a) could reach good accuracy when including special plates, the dataset quality shown in the paper is ideal and straightforward without complicated real-world instances. In the study done by Suandi et al. (2012), the training feature and class matrices are fed to the K-nearest neighbour (KNN) to classify the segmented characters from the LP. The recognition rate on 100 images reached 88% based on no error per image. The system could not deal with the complex background images and non-standard LP.

Deep learning-based methods have been used in some studies to perform the Malaysian LP recognition process. S. N. H. S. Abdullah et al. (2006) used a multi-perceptron NN consisting of the input layer, one hidden layer with 200 nodes and the output layer to classify the 36 characters. Results show an average classification error of 63.12%. The same approach was used by (Islam, 2007). The study recorded an accuracy of 93.2% on 150 images. Although they describe the dataset as complex, the paper shows high-resolution images.

How & Sahari (2017) trained a convolutional neural network (CNN) with 74K segmented characters from the CIFAR10 dataset. The study compared different multi-layer NNs and CNNs with different parameters and hidden layers. With the CNN, a similar architecture to the VGG convolutional neural network. Results show that CNN with deeper layers has achieved the best recognition rate of 95.89%. The drawback of this study is that the training dataset lacks variation and complexity as the rotation, skewness and distortions are not included.

The approaches above required a segmentation technique such as component connects CC or projection profile preceding the recognition step. However, Koon Cheang et al. (n.d.) and Shivakumara et al. (2018) proposed a segmentation-free approach, taking advantage of the CNN and RNN. The CNN is utilized to perform features extraction from the image and RNN to obtain the sequence order of the recognized characters from the LP. Koon Cheang et al. (n.d.) applied this approach on the Malaysian LPs and achieved a recognition rate of 76% on a license plate basis and 95.1% based on character recognition. Moreover, Shivakumara et al. (2018) added another classification step for private and public LP before the CNN-RNN does the LP's recognition. Experiments showed that the recognition after the classification resulted in better accuracy than without classification, with 71.23% before and an average of 83.77% after classification on the UCSD dataset. The system was unable to deal with the skewed and rotated LP images.

From the forementioned literature, it is obvious that existing studies on Malaysian ALPR discussed various approaches for license plate detection and recognition, including traditional image processing methods, statistical classifiers, and deep learning-based methods. However, these approaches typically required a segmentation step, such as connected components or projection profile, prior to recognition. Although few studies utilized segmentation-free approaches with ALPR systems in Malaysia, there are still limitations in terms of accuracy and robustness when dealing with complex real-world instances.

To sum up, ALPR systems in Malaysia could benefit from the integration of segmentation-free techniques, which have the potential to simplify the recognition process, eliminate the need for explicit segmentation steps, and potentially enhance accuracy and robustness when dealing with diverse license plate variations, orientations, and backgrounds.

2.5 ALPR DATASETS

Collecting a sufficient amount of LP dataset for the ALPR systems is essential to utilize the state-of-the-art deep learning algorithms. However, managing the adequate size of images for this mission is costly and challenging. Moreover, obtaining a dataset for all types of license plates in all countries is difficult since LPs' specifications differ from region to region. Thus, collecting the ALPR dataset tends to be a country-wise approach to acquiring a robust dataset. Therefore, a dataset might not be helpful enough for all studies.

To benchmark a dataset for public use, some requirements must be fulfilled. Otherwise, the dataset will not be enough to measure the system's performance in real-case scenarios. Among these specifications are environmental challenges such as weather conditions and lighting background. Further, the variation in the license plate should be considered, like the style, size, color, typeface, and language. Other variations coming from the camera effects, such as rotation, skewness, scale, and resolution, must be fulfilled. In this direction, some publicly available datasets have become popular, and many studies evaluate and compare the performance of their systems according to them. Table 2.1 lists the most popular public datasets and some information about them.

Table 2.1 Most Popular Public ALPR Datasets

Dataset	**Country**	**Volume**	**Description**
ChineseLP (Zhou et al., 2012)	China	410 images	250 images captured, 150 images from internet, multiple plates in an image, rotations, various scales, various backgrounds, different illumination,
AOLP (Kessentini et al., 2019a)	Taiwanese	2049 images	Different categories for different vehicles location, different illumination, different weather conditions.
OpenALPR – EU (OpenALPR, 2016)	European, Brazilian and US	108 images	Daytime only, properly centered, no various illuminations
SSIG-SegPlate (Gonçalves et al., 2016)	Brazilian	2000 images	High-resolution, static, daytime, different fonts
PKU Dataset (Kang, 2009)	Chinese	3828 images	Weather condition variations, illumination variations, multi-plate and single-plate

UFPR-ALPR (Laroca et al., n.d.)	Chinese	4500 images	illumination variations, different view angle, moving vehicles and camera
CCPD (Xu et al., 2018)	Chinese	250K images	Blured images, rotation, illumination variations, Weather condition variations
GAP-LP (Kessentini et al., 2019b)	Tunisian	9175 images	illumination variations, different view angle
RodoSol-ALPR (Laroca et al., 2022)	Brazilian	20K images	Day and night, from distinct lanes, on clear and rainy days, captured at pay tolls.
KarPlate (Henry et al., 2020)	Korean	30K images	Static, augmented
MediaLab LPR(*Medialab LPR Database*, n.d.)	Greece	716	Blur, lighting variations, shadow
CALTECH CARS (Weber & Perona, 2022.)	US	126	Static, small volume, daytime, no variations, taking at parking lot

2.5.1 Existing Malaysian License Plates Dataset

The implementation of ALPR on Malaysian LP presents some challenges. First, there is no known accessible benchmarked LP dataset for Malaysian vehicles. The Malaysian LP format's unreliability is the second issue. Many on-road Malaysian LP characters may use varied typefaces, spacing, and positions, breaching that official LP guideline. As a result, it is found that the majority of Malaysian ALPR studies lack a high-quality dataset that can contribute to developing and testing robust ALPR systems due to the insufficient dataset and the fact that they do not take into account all the complexities present in the LP in the real world. Table 2.2 illustrates these shortcomings in the Malaysian ALPR studies.

Each of the studies presented in Table 2.2 has limitations in terms of the size and quality of the datasets. In addition, most approaches could not ensure high performance on large challenging datasets. The ALPR model must be robust for real-time applications such as MLFF systems. From the Table, it can be noted that the systems in the studies conducted by Angeline et al. (2012), Suandi et al. (2012), Weng Keong & Iranmanesh (2016), Yogheedha et al. (2018) and Marzuki et al. (2019) reported good recognition accuracy. However, the systems lack a comprehensive and representative dataset showing system robustness. The drop in accuracy can be seen in the study of Tay et al. (2015) when the special plates are included.

The factors mentioned in Table 2 are: 1. Illumination variation 2. Day and night 3. Noise 4. Blur 5. Different view angle 6. Rotation 7. Skewness 8. weather complexity: sunny, foggy, and rainy, and Issues are a) Shallow Method, b) Small train and test dataset, c) Variation in dataset) The numbers assigned to these factors will be used in Table 2.2 to indicate which study considered these factors. CCA: Connected Component Analysis, FNN: Feedforward Neural Network, TM: Template Matching, KNN: k-Nearest Neighbors, SA: Smearing Algorithm, BA: Bees Algorithm, SVM: Support Vector Machine, RNN: Recurrent Neural Network.

Table 2.2 Key parameters and performance of Malaysian LPR proposed systems.

Ref.	**Method**	**Size of Train Dataset**	**Size of the Test Dataset**	**Used Special Plates**	**Double-row LP**	**Factors Included**	**ALPR Accuracy %**	**Issues**
(Angeline et al., 2012)	CCA-FNN	136 char.	50	No	No	3	90	a,b,c
(Tay et al., 2015)	Edge detection-TM	-	350	Yes	No	4	81.33	a,b,c
(Suandi et al., 2012)	CCA-KNN	330 char.	100	No	No	None	95	a,b,c
(Weng Keong & Iranmanesh, 2016)	CCA-Pearson Correlation	-	270	No	No	1,7	91.5	a,b,c
(M. Abdullah et al., 2017)	SA	-	150	Yes	Yes	1,3,4,5	76	a,b,c
(Shatnawi, 2018b)	BA-SVM	-	1216	Yes	No	None	95.43	a,b,c
(Marzuki et al., 2019)	CCA-CNN	700	528 char.	No	No	None	94.6	b,c
(Koon Cheang et al., n.d.)	CNN-RNN	2,304	409	No	No	1,3,5,7	95.1	b,c

(Yogheedha et al., 2018)	TM	-	14	No	No	None	92.8	a,b,c
(Shivakumara et al., 2018)	CNN-RNN	14,073	6,032	No	No	1,2,4,6,7	83.9	c
(Lee et al., 2022)	YOLOv3	7,000	3,000	No	No	6,7	99%*	c

* Accuracy for LP localization only.

Similarly, M. Abdullah et al. (2017) added special plates and double-row LP and made many variations in their dataset. Consequently, the system could not achieve good accuracy due to the complexity of the dataset. Although the study by Shatnawi (2018b) could reach good accuracy when including special plates, the dataset quality shown in the paper is ideal and straightforward without complicated real-world instances. Double-row LPs and special plates were not considered in the study of Cheang et al. (n.d.), which makes this study not robust to this parameter. A large dataset utilizing the standard UCSD dataset and the dataset MIMOS funded by the Malaysian government (Shivakumara et al., 2018). UCSD is a dataset consisting of 18,270 public (taxi LP) images, es and the MIMOS dataset covers 1,835 images from public and private vehicles LP. These datasets are not publicly available. Although the volume of the dataset is large, all variations and factors, such as double-line and unique LPs, are not covered. Moreover, most of the used datasets contain public vehicle LPs with white backgrounds and black foregrounds, which exist significantly less in reality than private LPs. Furthermore, the system could not achieve high accuracy and failed to deal with rotated LP. In order to achieve a robust ALPR system, all mentioned factors must be considered in the dataset. In addition, the size of the dataset plays a crucial role since it will probably cover all variations in LP instances. Deep learning-based methods can extract features from an image and classify them. However, there is no significant deep learning-based attempt at the Malaysian LPR system.

2.6 SUMMARY

In this chapter, we presented a literature review in the field of ALPR. We explored the approaches used to build ALPR systems in the two stages. For each stage, subsections of the traditional, conventional ML and DL approaches are discussed. Then, we highlighted the existing studies conducted on the Malaysian ALPR system and touched upon the drawbacks of these studies and defined the gaps. Moreover, we explored the most popular publicly available dataset in the field of ALPR in the global world. We also looked at the datasets utilized in the Malaysian ALPR systems and mentioned how ideal these datasets are. A quick look at the theory of the utilized networks in this study is presented. The different versions of YOLO are explained, and the architecture of YOLOv5 is introduced.

CHAPTER THREE

METHODOLOGY

3.1 INTRODUCTION

The proposed ALPR system must be robust to environmental challenges and variations in Malaysian LPs. Moreover, it must execute in real-time since it will be implemented in the toll payment collection system. This chapter describes the end-to-end methodology of developing a robust ALPR system.

Many textual blocks in traffic images, such as road signs, can be misinterpreted with LPs. Additionally, characters on the LPs might fill up a very small part of an image. Therefore, we suggest detecting the LPs in the interested areas first. The complete patch of the LP is then fed into another stage, allowing us to localize and recognize all LP characters simultaneously. Therefore, an approach utilizing the deep learning single-shot object detector (YOLO), which has demonstrated excellent performance and capabilities in numerous applications, is used in two stages in the proposed system. The Block diagram in Figure 3.1 depicts the proposed system's methodology. As the Figure shows, the two stages are License Plate Detection (LPD) stage and License Plate Recognition (LPR) stage. The two-stage approach has the advantage that the objects (characters) in the image are relatively small, making it difficult to detect them from the entire image in a single stage.

Moreover, the difficult character segmentation task is avoided, as it has already been done in related applications (Koon Cheang et al., n.d.; Tourani et al., 2020). Furthermore, by initially extracting the LP, the network concentrates exclusively on the ROI and disregards any potential text in the image. The two stages are sequential and use the same pipeline. The first step in each stage is to gather the datasets used to train and validate the model. Then, the model is used for the inference code applied to the real-world dataset. A preprocessing step is added between the two stages to enhance the image in terms of image positioning and image quality.

YOLO network is applied for both stages since it is suiTable for real-world applications. Utilizing the YOLO algorithm was motivated by the network's real-time processing time and high accuracy. Additionally, the network's capability as a single-shot detector is advantageous since it allows for the simultaneous localization and classification of objects. So, the output is presented as soon as the network receives the input.

Utilizing the data-driven technique in computer vision models has demonstrated undeniable development and acceleration in computer vision (Tsotsos et al., 2019). Moreover, the utility of large datasets in previous studies was impactful. Therefore, the data-centric approach is used in this research. This strategy involves systematically modifying/improving datasets to raise the precision of the DL applications. Therefore, the proposed method significantly considered an extensive collection and a systematic generation of Malaysian LP datasets.

The performance of the model was evaluated progressively as the dataset was collected. Many batches have been acquired, and for each batch, training was conducted and results recorded. Initially, the system did not perform well despite the relatively large dataset acquired. However, adding more datasets to the new batches improved accuracy. Chapter four reports all experiments and results, and analysis is discussed.

The following sections will describe the proposed approach, divided into four subsections; Section 3.2 explores the two stages approach with YOLO algorithms implementation. In Section 3.3, the inference of the real-world system is illustrated. Section 3.4 describes the dataset collection process. Finally, the proposed synthetic dataset generation method is explained in Section 3.5.

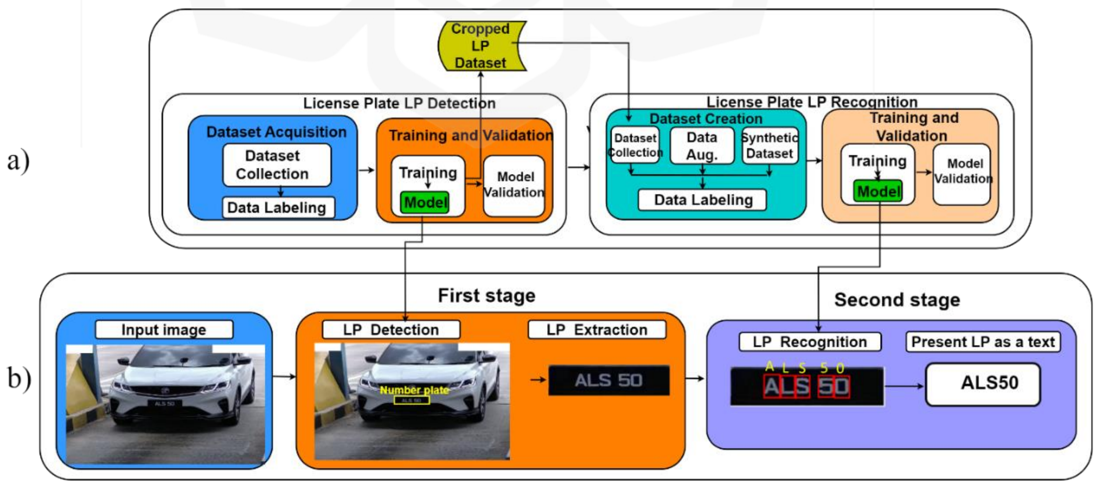


Figure 3.1 ALPR Proposed Method: a) Training and Validation Process. b) The Pipeline of The Two Stages ALPR System

3.2 TWO STAGES ALPR APPROACH

The system comprises two deep object detection models built using YOLO architectures. YOLO was selected for both stages since we require a real-time platform with high accuracy. Figure 3.1 depicts the two stages of execution. This diagram shows that the input is an image, a series of images, or video frames. The output is a sequence of characters corresponding to the context inside the vehicles' license plates.

A vehicle image is fed into the system and resized to fit the model input. The LPD stage then activates and searches for the object that matches the features of the vehicle license plates. The system uses a pre-trained YOLOv3 deep architecture in this stage to find all existing license plates. A second YOLO network (YOLOv5) is activated for LPR to recognize the letters and digits on the cropped license plates. As a result, a character classification process is achieved using character segmentation and detecting the characters at once.

3.2.1 License Plate Detection (LPD)

In order to develop a two stages system, as explained previously, the first stage is aimed at detecting the license plate from the scene. In the LPD, the only class is the number plate on the vehicle. It is essential to extract the LP independently in a standalone network to prevent any detection errors for characters in the image other than the number plate. In this regard, we performed experiments with the YOLOv3 and YOLOv5 models concerned with a speed/accuracy trade-off. When testing the models on the test dataset, YOLOv3 outperformed YOLOv5.

We utilized the open-source models presented by Ultralytics (Jocher & et al, 2021). Both YOLOv3 and YOLOv5 were pretrained on the COCO dataset (COCO - Common Objects in Context, n.d.), and several model versions are available to select based on the speed/accuracy trade-off concerns. Based on the default setups, network parameters, such as the confidence threshold and anchors, remained the same. YOLO accepts an image size of 1:1 aspect ratio in multiples of 32 as default. This depends on the model version and size of objects in the image, i.e., if images contain small object classes, it is recommended to have a larger image size. The image size utilized in our case is 640X640 since this is the default setting of the selected version, and LP is a small object in the image.

Training of the models was performed in many stages since several dataset batches were introduced, and performance was observed. All experiments are discussed in detail in Chapter 4. In general, the training dataset is fed into the model for the training process, and then the model is validated on the same training dataset.

The proposed system will get the input image; then, it is fed to the trained LPD detector model, which extracts and crop the LP and becomes the input for the next stage. Before presenting the cropped LP to the next stage, it is subjected to a resizing step to enhance the LPR's job. There is no need to preprocess the image for the LPD process. This confirms the robustness of our trained model to the different environmental conditions, as the result shows.

YOLO's deep architecture executes grid division on the image, with grid sizes that are all S×S. Two parameters are required for the prediction process: several bounding boxes B in each grid and a box confidence score P_c for each bounding box, known as the "abjectness," or how likely the box contains an object. When no object is in the box, the confidence score of P_c is 0. But P_c in the presence of an object is given by

$$P_c = P_r(object) \times IoU$$

Equation 3.1

It represents the likelihood that the box contains an object (objectness) as well as the accuracy of the bounding box. The IoU between the predicted (A) and the actual bounding box (B). Equation 3.2 calculates the IoU:

$$IoU\ (A, B) = \frac{Intersection\ of\ (A,B)}{Union\ of\ (A,B)}$$

Equation 3.2

A higher value of IoU indicates a higher dimension similarity between the two boxes. Each bounding box has five predictions: x, y, w, h, and confidence P_c. The (x, y) coordinates represent the centroid of the box with respect to the grid cell's dimensions. The width and height are anticipated in relation to the entire image. Then, if an object exists in the grid cell, the prediction of the class probabilities of that object is obtained. The conditional class probability is $P_r\ (C_i|Object)$. P_r is only calculated when the object is present in the grid cell. Regardless of the number of bounding boxes in the cell, only one class probability set is calculated per grid cell. The conditional class probability denotes the likelihood that the identified object belongs to a specific class C_i. So, one probability for each class is given per cell. The conditional class probability set has the shape of (S, S, $B\times 5$ +C).

Finally, the class confidence score is computed by multiplying the conditional class probabilities and the individual box confidence score. It measures the confidence in the classification and the localization of the object. Equation 3.3 is used to determine the class-specific confidence score for each box.

class confidence score = box confidence score × conditional class probability

$$class\ confidence\ score\ = \Pr(Class(i)|Object) \times \Pr(Object) * IOU$$

$$class\ confidence\ score\ =\ \ Pr(Class(i)) \times IOU$$

Equation 3.3

In our training, we utilized a pre-trained YOLOv3 model, which was trained on the COCO dataset containing 80 classes. The default parameters of the model from Ultralytics are used in this training as listed in Table 3.1, which have been chosen based on the experiments conducted by them.

Table 3.1 Model Parameters in The Training Process

Anchor boxes	4
Threshold value	0.20
Initial Learning rate	0.01
Final Learning rate	0.1
Batch size	18

3.2.2 License Plate Recognition

The cropped LP image is fed to the second stage LPR. In this step, the classification process of the characters is treated as an object recognition problem. This means that the characters' segmentation and classification will be executed simultaneously in a single step. Here, the task is to locate and classify 37 classes with ten digits, 24 Latin letters (dropping out I and O), and three special plate names (MALAYSIA, PUTRAJAYA, and PATRIOT). The trained LPR model will take the input image and present the unique LP in the output.

The YOLOv5 version is now utilized to perform the recognition stage. The network was trained on 640×640 image size.

3.3 INFERENCE

In the execution in the real world, the system acquires an input (video frames in our case) that contains the vehicle object and delivers the unique characters of the LP as a final output passing it through the two models: license plate detection LPD and license plate recognition LPR. If LP is detected, the system will proceed to the next stage (LPR), which takes the cropped LP as input and produces a text output for the LP name.

Before LP is given to the second stage, some preprocessing is performed on the cropped image. The preprocessed LP image is used in a second-stage trained model of LPR. The image is resized to 640×640 before the recognition step. Here, the task is to locate characters on LP with 37 possible classes comprising ten digits, 24 Latin letters, and special plates. We propose a majority vote system for improved recognition. Since the system receives a series of video frames for the same vehicle in the region of interest (ROI) set for each toll booth, this ROI becomes an advantage for the LPR stage. In this regard, all captured frames are recognized and compared for the same vehicle. Since all frames here have the same characters, every character is compared in all frames, and the majority prediction is taken as the final prediction. This strategy is applied to all characters in the LP. This technique, called temporal redundancy (Henry et al., 2020), ensures higher system accuracy.

3.4 DATASET

This section explores the Malaysian ALPR system's collection and dataset augmentation process. Since we intend to implement the data-centric deep learning method to the proposed system, much of our work focused on the dataset to improve the ALPR system. This was driven by data being incredibly vital in AI research. Therefore, it is crucial to create a strategy that puts getting high-quality data first. Various steps were achieved towards building a comprehensive dataset through obtaining a good quality and quantity of Malaysian LP dataset.

It is worth mentioning here the Malaysian License plate standardization assigned by authorities and the problem of non-compliance to the standard layout of Malaysian LP. The Road Transport Department Malaysia (JPJ) clearly states the standards of Malaysian LP to the public to ensure accurate identification and visibility. However, most people do not abide by these standards, which makes it challenging to recognize the LP in many cases. All specifications like size, layout, fonts, distances between characters and others are discussed in Section 1.3. Figure 3.2 shows some samples of the non-standard Malaysian LP. This problem is a big challenge to our ALPR system, as demonstrated in the experiments in chapter four. The non-predicted variations in the Malaysian LP make the task of the LPR hard, and accuracy is degradable every time the system faces new variations in the LP image. In this regard, the required solution must bring robustness to the system regarding the dataset. Thus, the proposed solution is to acquire as much dataset as possible so that it is most likely to cover all variations in the Malaysian LP. A large dataset was collected progressively in several batches as experiments were conducted, and the system performance was observed at every dataset batch.

Figure 3.2 Samples of Non-standard Malaysian LP

During the dataset collection, the target was to get a dataset with all the real-world possible circumstance variations such as the change in illumination, day and night-time, different camera mounting, and different weather conditions. Additionally, we worked on the dataset augmentation technique to cover all situations that might not be properly represented in the collected dataset. Moreover, a systematic approach is proposed for creating a synthetic dataset of Malaysian LPs. Dataset acquisition and description are presented in the coming subsection with more details.

3.4.1 Dataset Acquisition

As with any deep learning task, the dataset plays a significant role in the model's performance. Looking at the studies on Malaysian LPs, as discussed in section (2.3.1), it is seen that there is no publicly available benchmarked dataset for Malaysian LPs. Therefore, the work started by collecting real-world datasets for the training process by extracting video frames from mounted cameras where the system will be installed. As a result, thousands of video frames were extracted from cameras distributed in four toll plazas across Malaysia: Gombak Plaza Toll, Bentong Plaza and Karak Plaza Toll. Cameras are set up to capture the vehicle from view angles 45° with a distance from the ground of one meter, and the distance range away changes as the vehicle moves as depicted in Figure 3.3. Videos are taken in almost all environmental conditions, daytime, night-time, sunny weather, cloudy weather, rainy weather, noise and illumination variations. Table 3.2 lists the details of the dataset collection process and specifications related to the camera and environment.

Figure 3.3 Camera set-up depiction at the toll Plaza

Table 3.2 Specifications of the Dataset Collection Process

Toll Name	**Camera installation**			**Variations in Tolls**		
	Angle	**Height**	**Camera Spec**	**Lighting**	**Background**	**Weather condition**
Gombak Plaza Toll	45°	1.5m	3MP, 45fps 120 WDR 1/2.8″ Progressive Scan CMOS	Normal, Day, Night	Normal	Sunny/Cloudy
Bentong Plaza Toll	45°	1.45m	3MP, 45fps 120 WDR 1/2.8″ Progressive Scan CMOS	Normal, Day,	Normal	Sunny/Rain
Karak Plaza Toll	50	1.48m	3MP, 45fps 120 WDR 1/2.8″ Progressive Scan CMOS	Normal	Normal, Cluttred	Sunny/Rain

3.4.2 Dataset Description

This section provides a detailed description of the acquired dataset and explains how the dataset covers a large volume under different challenging circumstances. In addition, the augmentation process on the collected dataset is introduced.

3.4.2.1 License Plate Detection Dataset

Although the LP location is fixed on the vehicle and standard features for LPs, which makes it easy for the network to detect them, the variation in the images makes it challenging to perform the LP detection task to extract LP features in the image. So, all variations exist in the dataset to train and test the model on a realistic and representative dataset. Accordingly, as mentioned earlier, the dataset for this stage (LP detection) has been collected across the three toll plaza booths during different days, times, and weather conditions. By doing so, we can ensure the dataset is diverse and contains images with variations of blur, noise, rotation, skew, and variants of illumination and lighting, as well as regular images. Figure 3.4 shows the wide variations in the collected dataset regarding weather conditions and object positions in the frame. As observed in the Figure, the dataset presents challenges for a good training process.

Additionally, since the dataset is collected from different toll booths with many images, it was possible to cover most Malaysian LPs types and common nonstandard LPs with all layouts discussed in section 1.3. We collected 16,000 images for vehicles extracted from video frames to be included in the training and validation experiments. All images were manually labeled using the Labelimg tool.

Figure 3.4 Samples Instances of the LPD Dataset in Harsh, Low Light

3.4.2.2 License Plate Recognition Dataset

The LPR dataset is obtained from the LPD dataset. ROI in first-stage images is the license plate detected by the LPD network, and then, bounding boxes containing the LP part were cropped. The cropped regions are going to be passed to the second stage. Since this stage is critical and more challenging than the LPD stage, the size of the LPD dataset was not adequate to get good performance measured by the mean average precision (mAP), and we needed to add more vehicle images to get more LPR dataset datasets with better quality in them. Therefore, we accumulated much more images from the natural environment over time, and experiments have been done accordingly. The final number of LP collected images from the d was 30,000 images.

Since the source images in the dataset are vast and diverse, we could get a large LPR dataset with massive diversity in image characteristics and Malaysian LP categories variation. Figure 3.5 shows samples from our dataset and indicates the diversity explained previously. The classes in the LPR dataset are 37 classes: 24 Latin letters (I is labeled as one and O is labeled as 0), ten digits (0-9), and three special plates, Malaysia, Putrajaya and Patriot. Other special number plates were not dedicated as classes since they do not contain small letters, which is not the case with the plates containing Malaysia, Putrajaya, and Patriot. Thus, they can be treated as regular LPs where letters are recognized separately.

An augmentation strategy was also introduced to enhance the system performance by adding more challenges to our collected dataset, which may not be available enough. The following section discusses the augmentation strategy we applied to our dataset.

Figure 3.5 Samples Instances of the LPR Dataset

3.4.2.3 Augmentation for Dataset

In computer vision applications, data augmentation includes data production by manipulating images to have as many variants as possible. It can be used to create images with varying degrees of realism or to add more training data. In this way, adding data that the model has not seen during the training helps to increase the number of relevant data. Furthermore, it can assist in creating a balanced dataset to improve the model's performance.

In our research, a data augmentation strategy has been conducted on the collected dataset to include more challenges which may occur in the real world. Consequently, better accuracy of the system is achieved. The augmentation actions include rotation, shear, brightness, exposure, blur, and noise. Every action contributes to getting a solution for each real-world problem. With rotation, the problem related to the camera roll is solved. So that model can detect characters even when the camera or plate is not perfectly aligned. The shear action will help the model to be robust against the pitch and yaw of the camera or plate. Additionally, we added brightness and exposure options to the images to make the model resilient to lighting and camera setting options. Furthermore, camera focus may change during operation and cause images to look blurry, or weather conditions may cause the camera lens to have. That makes applying some blur to the dataset worthwhile and therefore obtaining a robust model against blur vision. Finally, we added noise effects to make the model resilient to camera artifacts and possible environmental dirt.

In Chapter 4, the augmentation process is explained in detail, where it was applied at many stages as experiments have been conducted over time. It demonstrated the effectiveness of the applied augmenting steps.

3.5 SYNTHETIC DATASET

In deep learning, unbalancing in the dataset causes an issue in the network's performance since learning some patterns might be biased (Laroca et al., 2021). It can lead to wrong predictions for the less-represented classes.

Generally, the character classes in the LP dataset are unbalanced due to the allocation policies, which leads to more appearance of some character classes than others. Having said that, although the augmentation strategy of creating more images with different challenges impacted the performance, as discussed in Chapter 4, the unbalancing issue in the dataset remained without enhancement. This can be seen clearly in the distribution of instances in our LP dataset in Figure 3.6. The number of a dataset is extensive and covers all required classes. However, the classes are not represented properly. For instance, the highest represented class is "eight" among the numbers and "W" among the letter, with 29,472 and 24,262 occurrences, respectively. The less representation of the classes occurs with the letter "Z" among the letters and the number "zero" among the numbers characters with 187 and 15,820 labels, respectively. Moreover, special LPs are significantly less represented, with 74, 63, and 33 labels for PATRIOT, PUTRAJAYA, and MALAYSIA, respectively.

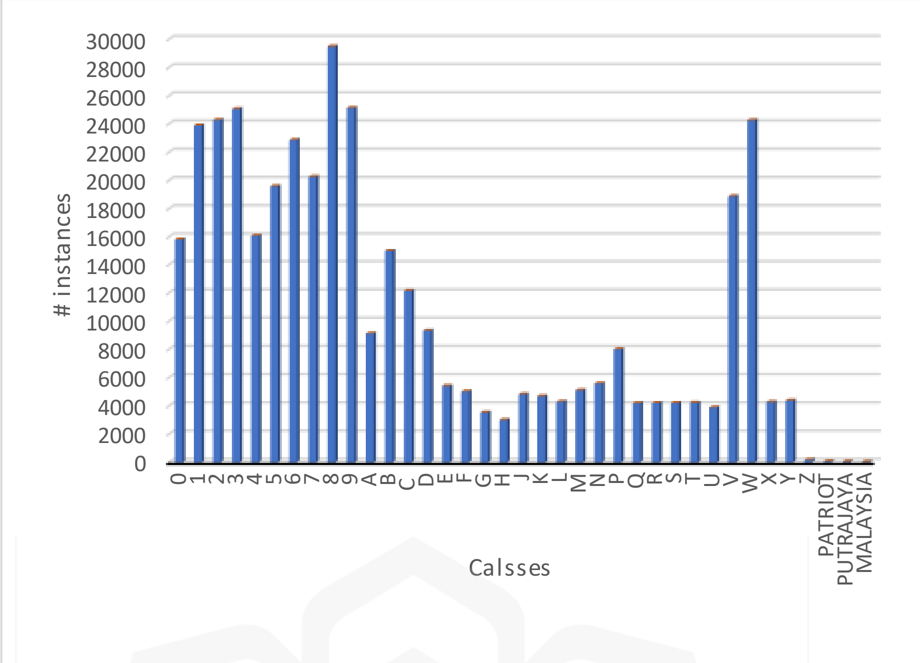


Figure 3.6 Classes Distribution in Our Dataset

Dropping out plate images to decrease the overrepresented classes to reach a balanced dataset cannot solve the issue. The imbalances in a dataset are due to overrepresented and underrepresented classes may exist on the same plate, which makes it an uncontrollable way to manage the classes on the same plate. Thus, we had to find an approach to tackle the unbalancing problem successfully and efficiently. As a result, we came up with the solution to add a controllable synthetic dataset to build on the underrepresented classes while maintaining the highly represented classes as they are. Therefore, we introduced a novel systematic approach to generate a systematic synthetic Malaysian LP dataset. The approach compromises two main stages: synthetic image generation and fully automatic annotation for the characters. Besides, a systematic way is introduced to allocate characters in the images to have a balanced dataset, control the variations in terms of challenges on Malaysian LP, and train and test portions of the dataset. Figure 3.7 depicts the steps of the proposed approach, and the following subsections discuss the steps in detail.

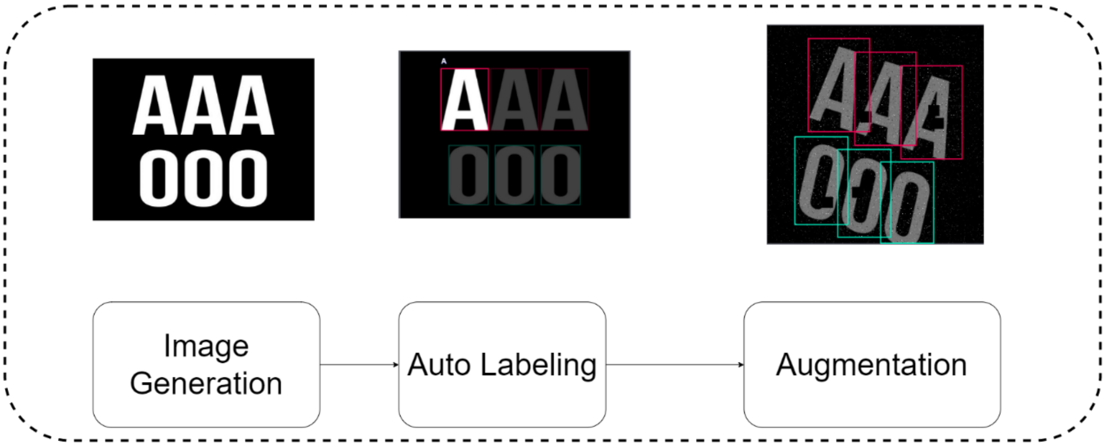


Figure 3.7 The Pipeline of the Synthetic Dataset Generation

3.5.1 Automatic Generation of Malaysian LP Images

First, we planned to generate the synthetic Malaysian LP automatically to overcome the issue of characters appearing on the LP randomly. In this regard, we exploited the repository Text Recognition Data Generation (TRDG) (Belval et al., 2020), which is dedicated to the synthetic generation of the dataset for OCR problems. It has the advantage of text generation in a flexible and controlled way. It generates texts randomly from different languages from selected text scripts, dictionaries, or Wikipedia websites. It is also possible to specify text specifications like font type, font size, margins, background, and other details, adding many distortion techniques to generated images to mimic the possibilities of real-world text images. Additionally, the generated image file is named with the exact text in the image. This feature will be helpful in the automatic annotation process, as discussed later.

However, we had to manipulate the repository to make it worthwhile for our case. We exploited the functionality of this repository to produce a synthetic Malaysian LP dataset which mimics the real Malaysian LPs. In this regard, we used the dictionaries-based method to let the code generate specific words (a combination of letters and numbers in our case). We built dictionaries that contain all the combinations of letters or numbers we plan to generate. After that, all specifications like font type, background, margins, and the distance between letters and image size are included in the command line. Figure 3.8 shows the pipeline of the approach we followed to generate the Synthetic Malaysian LP.

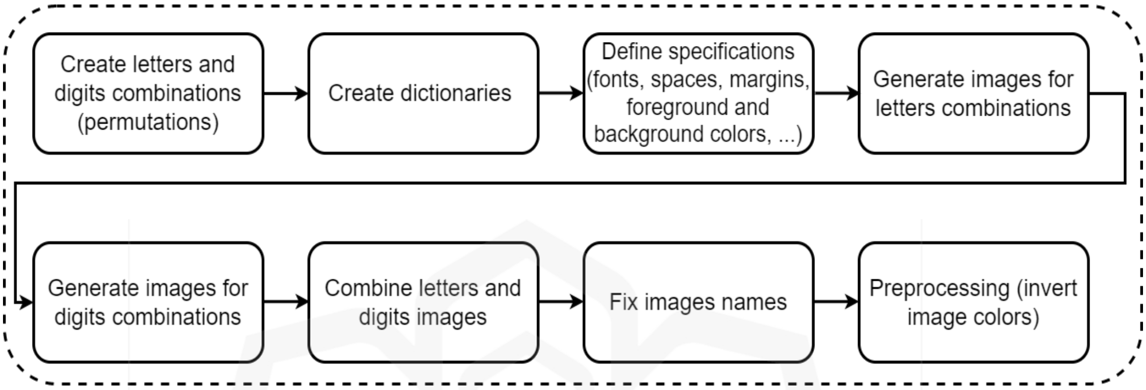


Figure 3.8 Steps of Image Generation in the Synthetic Dataset

Firstly, the generation of the alphabet and the digits is done separately because we need to customize and control what letters and digits should appear in the image in order to have the compensation of less-represented classes in the real-world collected dataset. Then, once we generate the customized alphabet and the digits, we combine them to form alphanumeric images representing the LPs. In this step, we select the font, the desired letters, the digits, and the number of images we want to generate. Some samples of the customized generated images are in Figure 3.9.

a) b)

Figure 3.9 Samples of Synthetic Images: a) Double-line Layout. b) Single-line Layout

Finally, we reached a balanced dataset after working on the synthetic Malaysian LP dataset. Figure 3.10 declares the size of the dataset before and after the process of synthetic dataset generation. The process of organizing and categorizing the letters and numbers to adequately compensate for the less-represented classes is explained in detail in Section 3.3.3.

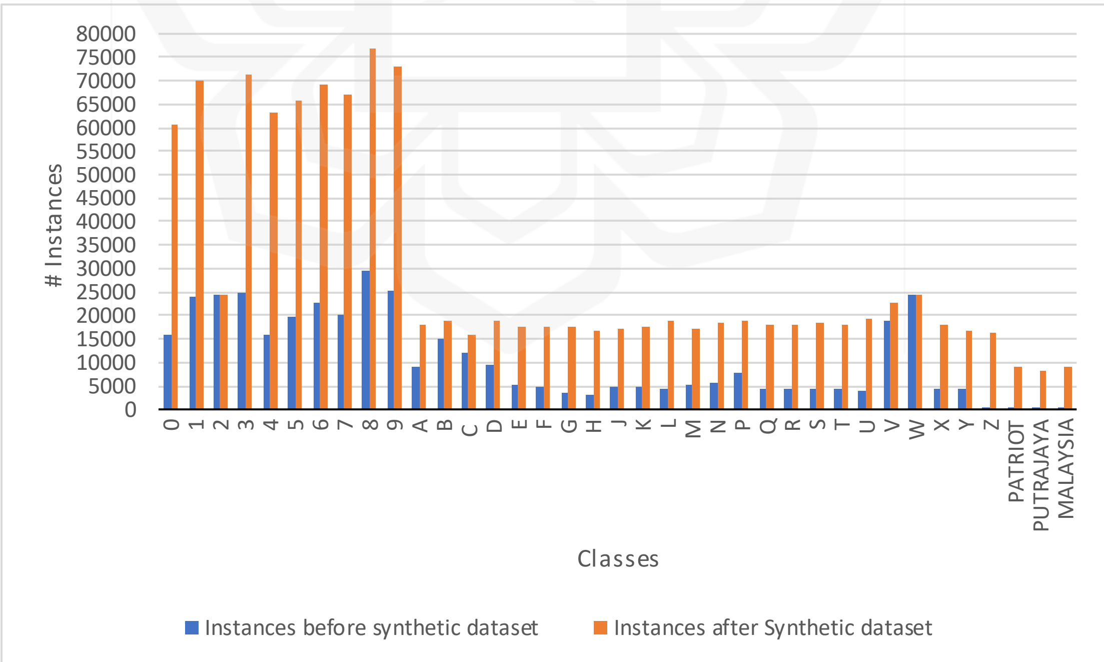
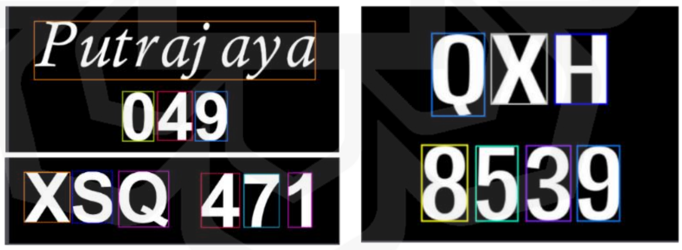


Figure 3.10 Classes Distribution Before and After Synthetic Dataset Generation

3.5.1.1 Automatic Annotation for the synthetic dataset

Typically, image annotation is performed manually, which is tedious and time-consuming. Moreover, human work might not be accurate since bounding boxes should be tight and exactly around objects for better feature extraction. Therefore, we proposed a method to annotate the Synthetic Malaysian LP dataset automatically and accurately after the synthetic dataset generation. The results from the automatic labeling look ideal in labeling and drawing the bounding boxes, as seen in the samples shown in Figure 3.11.

a)



b)

Figure 3.11 Labeling the Synthetic Dataset ideally, a) Labels with Classes Visualized in Roboflow, b) Samples of Different Layout LPs Labeling.

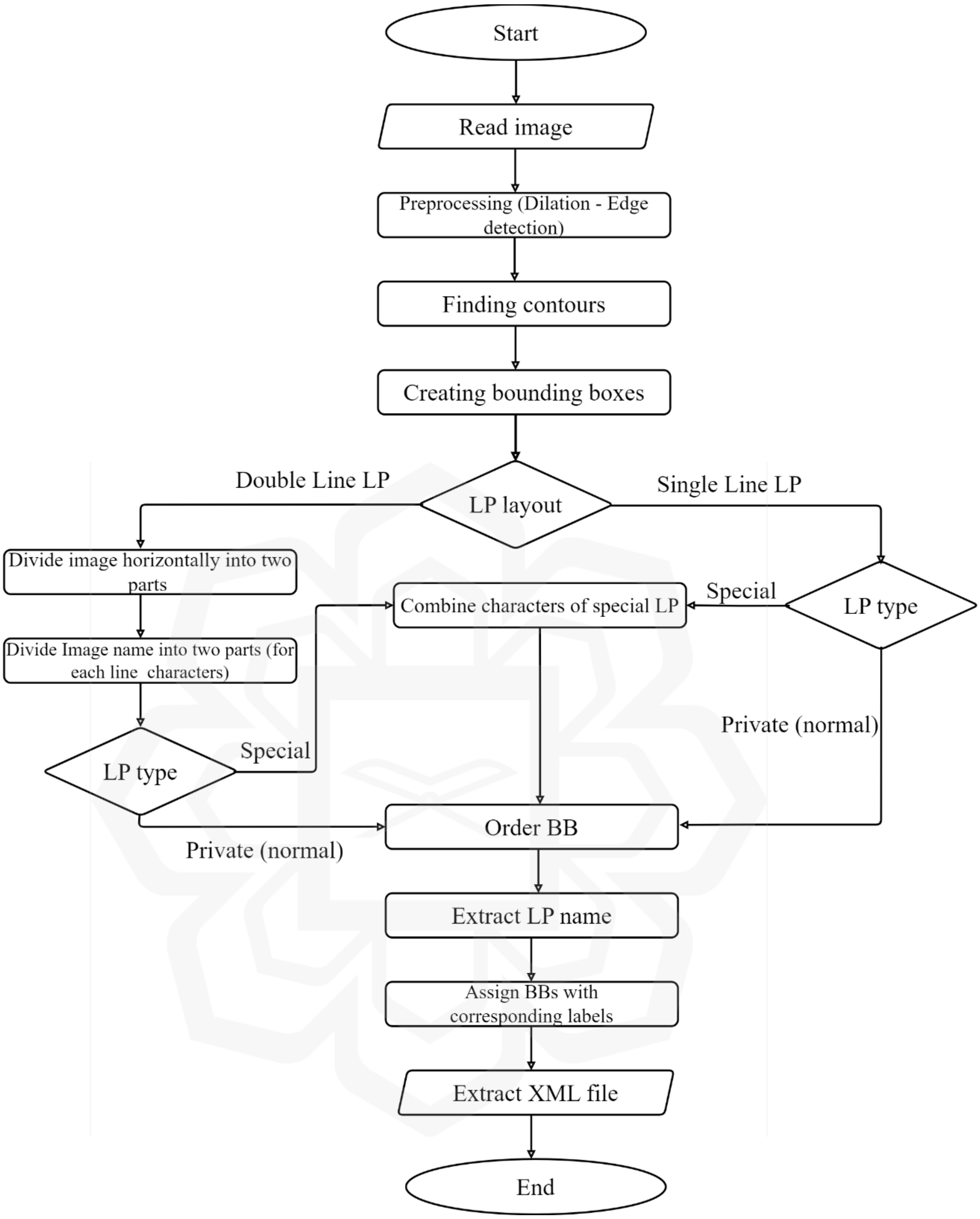


Figure 3.12 Auto-labeling Process for the Synthetic Dataset

The process of automatic annotation is depicted in Figure 3.12. At first, the code reads the image and performs some preprocessing steps, such as edge extraction and dilation. Edge extraction is applied on the object to get the correct contours. By performing dilation, the object - (letter or digit) – is filled with more pixels around and becomes thicker with a kernel size of 2X2. The bounding box coordinates are taken from the dilated image while the box is drawn on the source image. Doing that will let the bounding box surround the object with 2 pixels padding to ensure better feature extraction for the whole object. Then, the points of X_{max}, X_{min}, Y_{max}, and Y_{min} of the bounding boxes are determined by finding contours around the objects in the image and extracting the four points. Once the bounding boxes are extracted for the whole letters and digits in the image, An XML file is generated containing all the related data, such as the classes and box coordinates. All generated dataset images are fed to the code; the corresponding XML files are generated automatically for all images.

We faced some problems during the annotation process. Since the idea is to take the sequence of the plate ID and match each character in the ordered sequence with the bounding boxes in the image, the bounding boxes must also be presented in the same sequence order as the ID name. Ordering the characters on the image is simple for the Single line LPs by considering the Xmin points and order boxes ascendingly. However, it becomes hard for the double line to do so since ordering characters based on Xmin points results in a wrong sequence. This is because, in the lower line, some characters have Xmin points smaller than some characters in the upper line, letting them lead in the sequence wrongly. Therefore, the layout of the LP is determined before deciding how to perform the BB order. In the case of double line LP, the image is treated as it is two sections (upper and lower), and the ID name of the LP is divided during the LP generation step. Therefore, the upper and lower sections can be treated separately, and BB is labeled correctly. After that, the two sections are combined to form the LP image's correct annotation.

Another problem related to the special plate classes where the whole text - (PATRIOT, PUTRAJAYA, MALAYSIA) - must be labeled as a whole class instead of individually labeling the letters of the text. However, by modifying the code, we could solve this problem. Instead of taking the X_{max}, X_{min}, Y_{max}, and Y_{min} around each contour, we calculated the X_{max}, X_{min}, Y_{max}, and Y_{min} of all contours within the whole text of "PUTRAJAYA" for example and then the bounding box is taking for the whole text as it is shown in Figure 3.11(b). The code must also be slightly edited and customized for the double-line LP images since the first code was designed to auto-label the single-line LP images.

3.5.2 Controlled Systematic Synthetic Dataset

A systematic way is followed to generate the synthetic dataset to have a controllable dataset. At first, we analyzed the current collected dataset by observing the drawbacks. The most practical issue was the unbalancing in the dataset, where some classes were highly represented, and some were less represented. Table 3.3 indicates this issue. The highest represented class was “eight” among the numbers and “W” among the letters, with 29,472 and 24,262 occurrences, respectively. The less representation of the classes occurs with the letter “Z” among the letters and the number “zero” among the numbers characters with 187 and 15,820 labels, respectively. Special plates are significantly less represented, with 74, 63 and 33 labels for PATRIOT, PUTRAJAYA, and MALAYSIA, respectively.

Therefore, we decided to generate an adequate number of instances for the least represented classes with less than twenty thousand labels and increase them to approach 20,000 instances for each class. So, the idea is to have a complement of the current real-world dataset in order to get a balanced dataset.

After determining the number of instances that should be created for each class as stated in Table 3.3, 10% of these instances are decided to be the raw dataset. The raw dataset comprises basic images that will be used to create the remaining dataset, 90%. This portion of the raw dataset is chosen to avoid many raw images since the synthetic generated images are sharp and high-resolution images, which are not suiTable for learning a model for real-world scene complexity. The rest of the images (90%) will be generated through the augmentation process conducted on the 10% of images. The raw and total number of instances for each class is indicated in Table 3.3.

Table 3.3 Existing Number of Labels of the Classes in the Collected Dataset and the Number of Labels for Each Class to be Generated in the Synthetic Dataset

Class	Exist Labels	Total Samples to be Generated	Raw Samples to be Generated	Class	Exist Labels	Total Samples to be Generated	Raw Samples to be Generated
All	390,149			J	4,812	15,188	1,519
0	15,820	44,180	4,418	K	4,680	15,320	1,532
1	23,889	36,111	3,611	L	4,282	15,718	1,572
2	24,281	35,719	3,572	M	5,095	14,905	1,491
3	25,054	34,946	3,495	N	5,582	14,418	1,442
4	16,069	43,931	4,393	P	8,002	11,998	1,200
5	19,578	40,422	4,042	Q	4,188	15,812	1,581
6	22,859	37,141	3,714	R	4,199	15,801	1,580
7	20,249	39,751	3,975	S	4,204	15,796	1,580
8	29,472	30,528	3,053	T	4,224	15,776	1,578
9	25,119	34,881	3,488	U	3,881	16,119	1,612
A	9,113	10,887	1,089	V	18,875	1,125	113
B	14,971	5,029	503	W	24,262	-4,262	-426
C	12,143	7,857	786	X	4,273	15,727	1,573
D	9,324	10,676	1,068	Y	4,372	15,628	1,563
E	5,400	14,600	1,460	Z	187	19,813	1,981
F	5,011	14,989	1,499	PATRIOT	74	19,926	1,993
G	3,519	16,481	1,648	PUTRAJAYA	63	19,937	1,994
H	2,990	17,010	1,701	MALAYSIA	33	19,967	1,997

3.5.2.1 Characters Allocation on the Images

Since the license plate comprises a combination of alphanumeric characters, random generation of the images makes it difficult to maintain the desired number of instances for each class. Thus, after determining the number of instances for each class that needed to be generated, we looked at the classes with the same number of instances and combined them in one group. Then, we take the group-level average of the occurrence counts to be the final tally for those characters. In this way, we ensure that the generation of the classes with the desired number remains in control. Table 3.4 shows the grouping based on the classes with the closest number of desired samples. The class combinations are made up of three characters in each group and are obtained by finding all possible "permutations with repetition" for the group's elements. For example, the group (B-C-V-B) has combinations like (BCV, CVB, VBC, VVB and so on). Then from the permutation result, the occurrences of each element in all permutations are calculated – using a simple code - to determine how many times the permutations need to be repeated. The repetition of the permutations in nothing but the number of images will be generated for the group. The whole process is illustrated in Table 3.4.

Table 3.4 Combinations of Letters for Systematic Generation of Synthetic Dataset in Private LPs with Related Statistics

Group/ Combination	**The Average Number of Sample Generations for Each Character**	**All Possible Permutations**	**Occurrences of Each Character in all Permutations**	**Repetition Needed for Each Character (Average Samples/Occurrences in Permutation)**	**Number of Images Will be Generated (Permutations * Repetition)**
B-C-V-P	450	64	48	9	576
A-D-P	805	27	27	30	810
E-F-M-N-U-L	1430	216	108	13	2808
J-K-Y	1535	27	27	57	1539
T-X-Q-R-S-G-H	1625	343	147	11	3773

It is worth mentioning that class “Z” does not appear in any of the previous groups because at it has a different use in the Malaysian license plate system. It is only used for military vehicles LPs and comes only with the following combinations (ZA – ZD -ZL -ZU - ZZ). Therefore, the generation of these combinations takes different groups of combinations. Table 3.5 shows the combinations of “Z” and the number of images that will be generated. Since there are other letters in the combinations of “Z”, it should be taken into account that these letters are generated in the combinations shown in Table 3.5 because this could result in them occurring more frequently than desired.

The special classes (MALAYSIA, PUTRAJAYA, PATRIOT) are handled like that with the letter “Z” but without any combinations, as special classes appear on the license plate alone. Table 3.6 shows the number of images generated with the special classes.

Table 3.5 Combinations of “Z” with Letters for Systematic Generation of Synthetic Dataset with Related Statistics

Combination	**Number of Generations for each Character**	**Repetitions for each Character**	**Number of Images will be Generated**
ZZ	330	660	330
ZU	330	330	330
ZA	330	330	330
ZD	330	330	330
ZL	230	230	230

Table 3.6 Systematic Generation of Synthetic Dataset with Related Statistics for the Special Plates

Special LP	**Number of Generations for each Special Name**	**Number of Images to be Generated**
MALAYSIA	1997	1997
PUTRAJAYA	1994	1994
PATRIOT	1993	1993

3.5.2.2 Factors Considered in the Systematic Generation Process

Throughout the systematic generation process of the synthetic dataset, we considered many factors to incorporate the perfect dataset criteria and cover all variations in the real-world dataset. The following factors are included: balancing in the dataset, single line and double line LP, train and test dataset, different fonts, image effects to simulate weather conditions (blur, noise, illumination, exposure, brightness), and camera position effect (rotation, shear). These factors are applied carefully and systematically, as explained in the coming sections.

First, after identifying the total number and the raw number of each class, the strategy is to generate the raw images (10% from the whole generation) by considering the factors (single line and double line LP, train and test dataset, different fonts) only. The rest factors, including image effects to simulate weather conditions and camera position variations, will be applied later when generating the rest images (90%) from the augmentation process. In this regard, the raw images are organized hierarchically, where the selected fonts are used for both single-line images (70% of the images) and double-line images (30% of the images). Each font has a train and test dataset with a 90% and 10% distribution, respectively. As a result, for the single-line LP dataset, both the training and test dataset include all selected fonts, similar to the double-line LP dataset but with a percentage of 30% of the whole generated images. Table 3.7 illustrates the hierarchal approach of the systematic generation.

Table 3.7 Distribution Policy of Generated Synthetic Dataset

Group	Raw Dataset							
	Single Line LP (70%)				**Double Line LP (30%)**			
A-B-C	Font 1		Font 2		Font 1		Font 2	
	Train (90%)	Test (10%)	Train (90%)	Test (10%)	Train (90%)	Test (10%)	Train (90%)	Test (10%)

To decide which fonts to be used in the synthetic dataset generation, we observed the regularly utilized fonts in the real-world Malaysian LP. From the dataset we have gathered, we noticed lots of fonts used, which is illegal according to regulations. Afterwards, the private LPs are designed using ten typefaces frequently used in reality. Meanwhile, for special plates, the fonts have been selected differently according to the most frequently used fonts for every special plate, which sometimes differ from some special plates to others. We end up with five fonts for each special LP. The fonts used are listed in Table 3.8 for both private and special LP and are also shown in Figure 3.9.

Table 3.8 Selected Fonts in the Synthetic Dataset

Private LPs		**Special LPs**					
		MALAYSIA		**PUTRAJAYA**		**PATRIOT**	
Font	Portion	Font	Portion	Font	Portion	Font	Portion
Arial Bold	14%	Calisto MT Italic	15%	Calisto MT Italic	20%	Calisto MT Italic	20%
bebas_neue	14%	Palo Compressed Medium	15%	Helvetica Now Micro	20%	Helvetica Now Micro	20%
Grand Junction	14%	Helvetica Now Micro	15%	bebas_neue	20%	Aguda Black Unicase W01 Rg	20%
Helvetica Now Micro	14%	SSI	15%	Grand Junction	20%	Handel Gothic Regular	20%
PaloCompressed Medium	14%	Longhaul	40%	Charles Wright	20%	Charles Wright	20%
Aguda Black Unicase W01 Rg	6%						
Charles Wright	6%						
Digital_Sans_EF_ Bold_Italic	6%						
FE-FONT	6%						
Handel Gothic Regular	6%						

Due to the limited number of digit classes (from 0 to 9) and the fact that every license plate has numbers, generating the numbers for the synthetic LP is slightly different. Additionally, the digit "two" is not included in the generation because the real dataset has more than 20,000 instances, and it can be confused with the letter "Z" if there is a significant difference between them in the instances. Therefore, we intended to keep it near the 20,000 instances as the planned range for the "Z" letter. Considering all that, the number combinations are formed from three or four digits (0 to 9 with two excluded). Thus, the "permutations with repetitions" of the two combinations are generated based on the details in Table 3.9. The permutations need to be repeated to have the same number of digit combinations as the letter combinations to form the license plate images. Therefore, there must be 17,040 combinations of digits as the letter combinations. This will make it possible to create LP images by combining letter and number combinations (images).

Table 3.9 Combinations of Digits with Number of Images to be Generated for the Synthetic Dataset

Digits	**Combinations**	**Number of Possible Permutations**	**Number of Generations for each Digit**	**No. of Images to be Generated**
0,1,3,4,5,6,7,8,9	Three Digits	729	243	729 Permutations * 14 times
0,1,3,4,5,6,7,8,9	Four Digits	6561	2916	6561 Permutations

3.5.2.3 Augmentation of the Raw Synthetic Dataset

The generated synthetic dataset looks ideal but cannot be used to simulate the real-world LP dataset in the training and testing. Thus, an augmentation operation is carried out on the Synthetic dataset to enable the network train on the realistic-looking dataset. We employed a similar augmentation method to that utilized with the real-world dataset. Table 3.10 indicates the augmentation operations performed on the synthetic dataset as well as the size of the dataset we ended up with after the augmentation process.

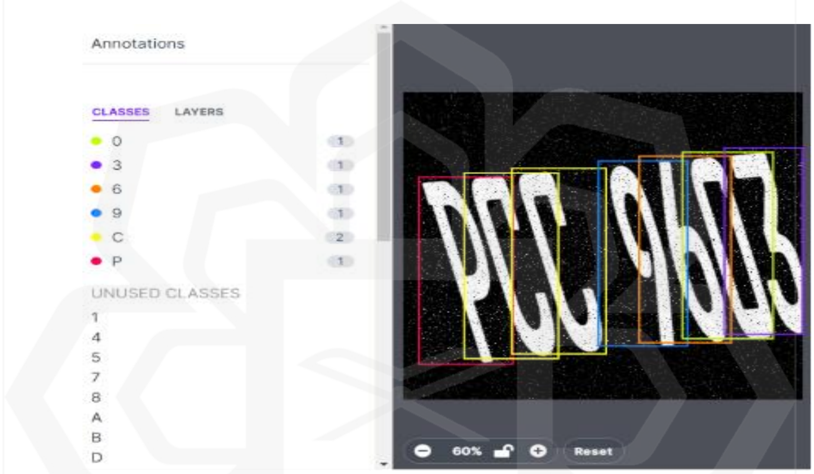
With 10% of the total dataset, the method explained in section 3.3.2 produces the raw images. However, most instances (90%) are produced by the augmentation process on the raw dataset. The raw dataset consists of very high-resolution images, which will not positively impact system accuracy due to the absence of variances. For this reason, the percentages are divided at a high level (90%) in the augmentation process to reflect all changes that may occur in the real world.

A systematic way to generate the augmented dataset is proposed to obtain a high degree of variation and complexity. Table 3.10 illustrate the approach we followed to augment the raw synthetic dataset. The Table shows four categories suggested with various degrees of complexity (Easy – Medium – Difficult – Complicated). Under these four categories, nine groups are made up of different image effects combinations. The complexity of the augmented dataset is getting higher from the “easy” category to the “complicated” category. It is worth noting that each group contributes 10% of the whole generated dataset (raw and augmented). In other words, the augmented dataset in each group will have the same number as the raw dataset since every image in the raw dataset is augmented only once in the group.

Table 3.10 Augmentation Strategy with Effects Applied on Each Group

Category	Easy	Medium		Difficult			Complicated		
Group	—	Medium 1	Medium 2	Difficult 1	Difficult 2	Difficult 3	Complicated 1	Complicated 2	Complicated 3
Rotation	Between - 8 and +8°		Between - 10° and +10°	Between - 12° and +12°	—	Between - 14° and +14°	Between - 14° and +14°	Between - 14° and +14°	Between -14° and +14°
Shear	—	±10° Horizontal, ±10° Vertical	—	—	±15° Horizontal, ±15° Vertical	±15° Horizontal, ±15° Vertical	±15° Horizontal, ±15° Vertical	±15° Horizontal, ±15° Vertical	±15° Horizontal, ±15° Vertical
Blur	Up to 4px	—	Up to 3.5px	Up to 4px	Up to 4px	Up to 4px	—	Up to 4px	Up to 4.25px
Noise	Up to 7% of pixels	Up to 10% of pixels	Up to 12% of pixels	Up to 15% of pixels	Up to 17% of pixels	Up to 18% of pixels	Up to 15% of pixels	Up to 11% of pixels	Up to 11% of pixels
Brightness	—	—	—	Between - 55% and +0%	Between - 55% and +0%	Between - 60% and +0%	Between - 65% and +0%	Between - 45% and +0%	Between - 45% and +0%
Exposure	—	Between - 70% and +70%	Between - 70% and +70%	—	Between - 75% and +75%	—	Between - 80% and +80%	Between - 80% and +80%	Between - 99% and +99%
Cutout	—	—	—	—	—	—	—	—	15 boxes with 5% size each

The online data annotation tool Roboflow is where the augmentation operations are carried out. Roboflow preserves the annotations correctly even after image effects are applied to the dataset, as shown in Figure 3.13. Roboflow applies the effect randomly over whole images. For example, if you apply a rotation between -10° and +10°, the rotation will be applied to all images with varying degrees of rotation within the applied range. Table 3.11 details the effects applied to the raw images and explains each effect.



a)

b)

Figure 3.13 Augmentation for the Labeled Dataset, a) Labels and Classes Preserved After Augmentation, b) Augmentation for Different Layouts LP.

Table 3.11 Impact of Applied Effects in the Augmentation Process

Effect	**How will it Benefit**	**Why is it Used**
Rotation	Add variability to rotations to help the model be more resilient to camera roll.	It helps the model detect objects even when the camera or subject are not perfectly aligned on the roll axis.
Shear	Add variability to perspective to help the model be more resilient to camera and subject pitch and yaw.	It helps the model detect objects even when the camera or subject are not perfectly aligned on the pitch and yaw axes.
Blur	Add random Gaussian blur to help the model be more resilient to camera focus.	If the subjects in-the-wild might not be in focus or the model is overfitting on hard edges.
Noise	Add noise to help the model be more resilient to camera artifacts.	Noise can help defend against adversarial attacks and prevent overfitting.
Brightness	Add variability to image brightness to help the model be more resilient to environmental lighting.	Brightness can bring simulation of the luminance effect created by a light source.
Exposure	Add variability to image brightness to help the model be more resilient to camera setting changes.	To simulate the factors that affect the sensor when it is exposed to light.
Cutout	Add a cutout to help the model be more resilient to object occlusion.	Cutout removes sections of images to simulate occlusion.

The size of the synthetic dataset was planned to be 170K images. However, we end up with 130K images. This reduction in the planned dataset size resulted from two reasons. First, some of the generated images come with characters connected together by a few pixels when using specific fonts which causes wrong auto-labeling for these characters. As a results, we discarded all the images with this issue from the whole dataset. Second reason is that during the augmentation step, some images were generated with no characters on them. This issue raised up when the brightness effect is applied with different degrees. Despite the overview of the effect on the images did not show any problem before implementing the effect, around 6000 images produced with no objects appear on them and were discarded from the dataset.

3.6 SUMMARY

In this chapter, we explained the end-to-end methodology we followed to develop the proposed ALPR system for Malaysian LPs. We first elaborated on the suitability of the object detector network YOLO for implementing the ALPR. After that, we discussed in detail the implementation of the YOLO in both stages, LPD and LPR. Furthermore, we explored the methodology we used for dataset creation. First, we discussed the steps of dataset collection from the real-world environment with detailed information on the quality of the dataset collected. In addition, we proposed a systematic synthetic dataset generation in which we had the data-driven model for the proposed ALPR system.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 INTRODUCTION

This chapter introduces the results of the experiments performed to develop a robust ALPR system. The results of a two stages ALPR system are explained separately, and the end-to-end system performance is presented. The discussion of the training of networks utilized in the two stages and the dataset analysis are also highlighted. It also includes an elaborate discussion. The evaluation protocol used in this research is also described. In addition, the results analysis of the conducted experiments is presented.

4.2 EVALUATION PROTOCOL

To see how the proposed Method of LPR performs in the two stages (i) LP detection (ii) LP recognition, precision, and recall measure metrics are reported. Precision refers to the number of correct detections (true positives) to the number of all detections (either true or false), while recall represents the number of the correct detections (true positives) to the whole number of ground truths (objects labeled initially in training dataset) or:

$$Precision\ (P) = TP/(TP + FP)$$

Equation 4.1

$$Recall(R) = TP/(TP + FN)$$

Equation 4.2

, where *TP* is the true positive, *FP* is the false positive, and *FN* is a false negative. However, no single metrics picture the system's performance. Hence, the solution by combining both metrics in one evaluation metric called mean Average Precision (mAP). The mAP plots precision values against all recall values in one graph. The bigger the area under the curve, the better the system performance. The evaluation metric F1 is also introduced, where it is given by Equation 3.4. F1 score is used to select the model that give the best precision and recall rates if both matters. If one of them matters more in the application, then F1 is ignored.

$$F1 = \frac{1}{\frac{1}{Recall} + \frac{1}{Precision}}$$

Equation 4.3

We report the mAP rates of the two stages, LPD and LPR, with an IoU threshold value of 0.5 based on the PASCAL VOC Challenge (Everingham et al., n.d.), which is a standard evaluation metric of object detectors.

In the aforementioned evaluation method, correct predictions are considered by the correct localization and classification of the object compared with ground truth. The IoU threshold value determines the localization, which is the overlapping between the ground truth and the predicted bounding box. At the same time, the predicted class must be the same as the ground truth with confidence.

Another system performance evaluation is introduced, which is an end-to-end performance evaluation. For the end-to-end performance evaluation, we did not consider the mAP metric. Instead, we looked at the final accuracy by dividing the correctly recognized LP by the whole test dataset. In this regard, if the final LP recognition result is the same as the real LP, then it is considered correct detection. Otherwise, any missed plates or misclassification in the characters will lead to false detection for the whole plate, resulting in false identification of the vehicle. This is the ultimate goal of the result evaluation in the ALPR problem, which must provide the LP exactly as in reality.

As mentioned in section 3.2.2, the final number of images collected manually is reported as 16 thousand images for the LP detection and 58 thousand images for the LP recognition stages. Nevertheless, various experiments have been conducted with different images introduced before reaching the mentioned dataset size as discussed in the next section. We specify 1040 images that were not included in any training process as a test dataset. These images, all variations, are considered to have a comprehensive and challenging test dataset. As for the synthetic dataset, we split the synthetic dataset into train and test sets with 90% and 10%, respectively, as discussed in section 3.3.3.2.

The results of the different experiments are then reported, and comparisons are made. The computer used for all experiments had i7 10th gen CPU, 24 GB of RAM, and an NVIDIA GeForce RTX 3060 GPU (3584 CUDA cores and 12 GB of RAM). We have utilized the Pytorch framework for training and testing the networks YOLOv3 (Jocher & et al., 2021) and YOLOv5 (Jocher et al., 2021) in LPD and LPR stages, respectively.

4.3 LICENSE PLATE DETECTION

To develop LPD, two deep-learning networks explored are YOLOv3 and YOLOv5. The following sub-section will cover the training results.

4.3.1 YOLOv3 LPD Results

For this purpose, the number of classes used is only one. Fourteen hundred images with 1,400 labels in the context of toll camera view are collected from Gombak Plaza Toll. The average size of the license plate was about 15% of the image size. The challenge is similar to small object detection in this case. The details of the dataset and its label size are depicted in Figure 4.1.

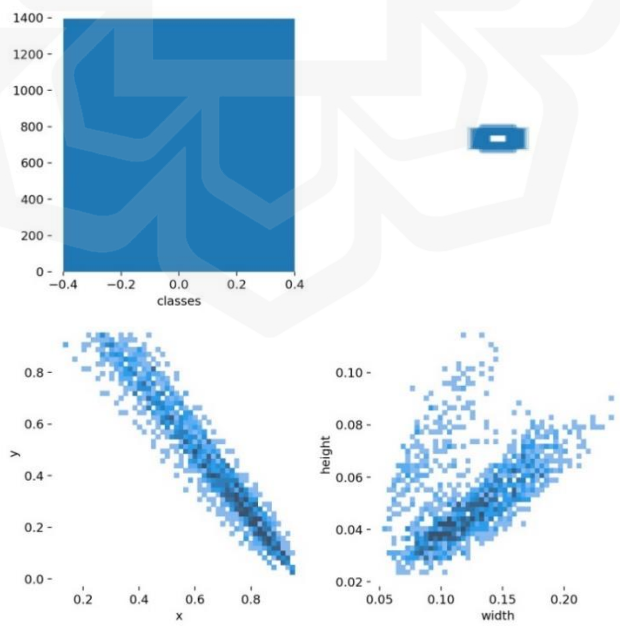


Figure 4.1 Label Distribution in the LPD training on YOLOv3

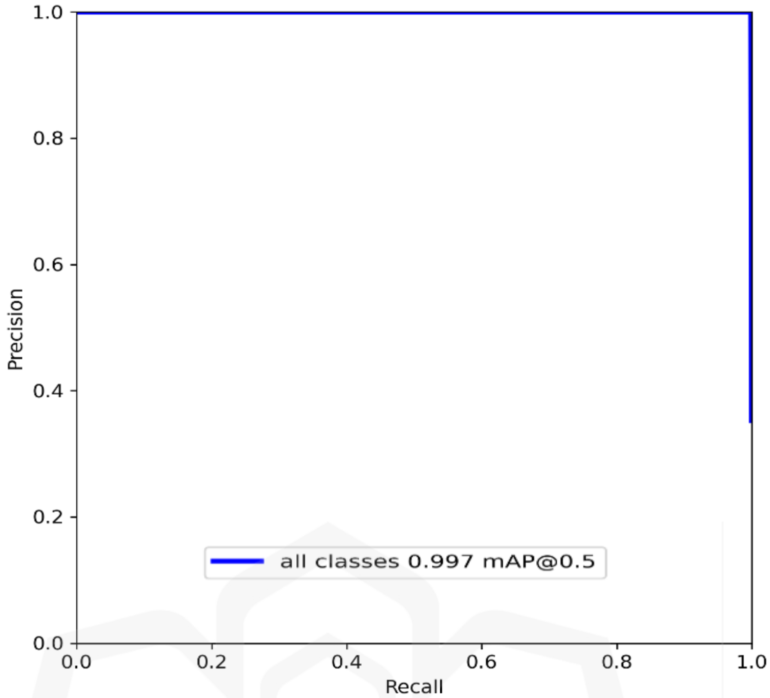


Figure 4.2 Precision/Recall Curve of LPD Validation on YOLOv3

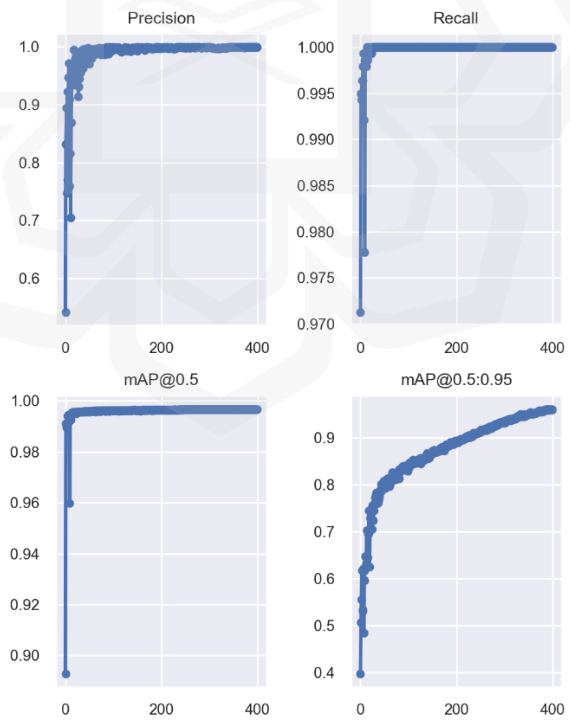


Figure 4.3 Training Results of LPD Stage on YOLOv3

Figure 4.4 Samples of LP Detections on YOLOv3

The training of over 400 epochs on YOLOv3 resulted in an overall mAP of 99.7% for the license plate detection task. The precision-recall curve can be seen in Figure 4.2, showing its robust performance. Moreover, precision and recall improvement per epoch is illustrated in Figure 4.3. The performance of LPD on field tests and test datasets shown in Figure 4.4 depicts very challenging conditions of low light properly illuminated and various distances in the toll camera region of interest. It signifies the robustness of LPD achieved. On-field testing was also reported with over 98.8% accuracy.

4.3.2 YOLOv5 LPD Results

Sixty thousand images from different locations in Malaysia, like KL and the Penang toll cameras, are collected. The average size of the license plate was about 10% of the image size. In this instance, the mission is to detect small objects in the image. Figure 4.5 shows the dataset's details, including the label size and the heat map of the BB' positions and scales. Heat maps provide a visual representation of where the object detection model is focusing its attention within an image. This can give an insight into the model behavior. It also gives an intuition on how dataset are variable in terms of the presence on the image. The more distribution of BB positions, the better variation in the dataset.

The license plates at this time were collected over all the year's seasons in different weather conditions and day and night situations from coastal, urban, and rural settings.

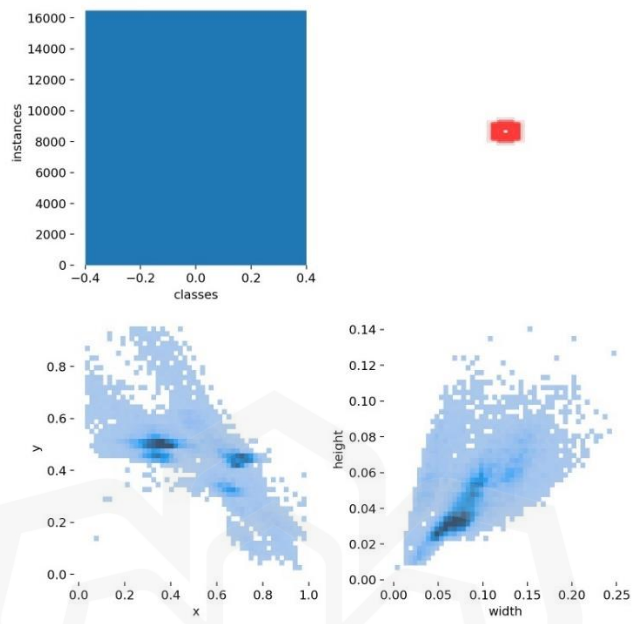


Figure 4.5 Label Distribution in the LPD training on YOLOv5

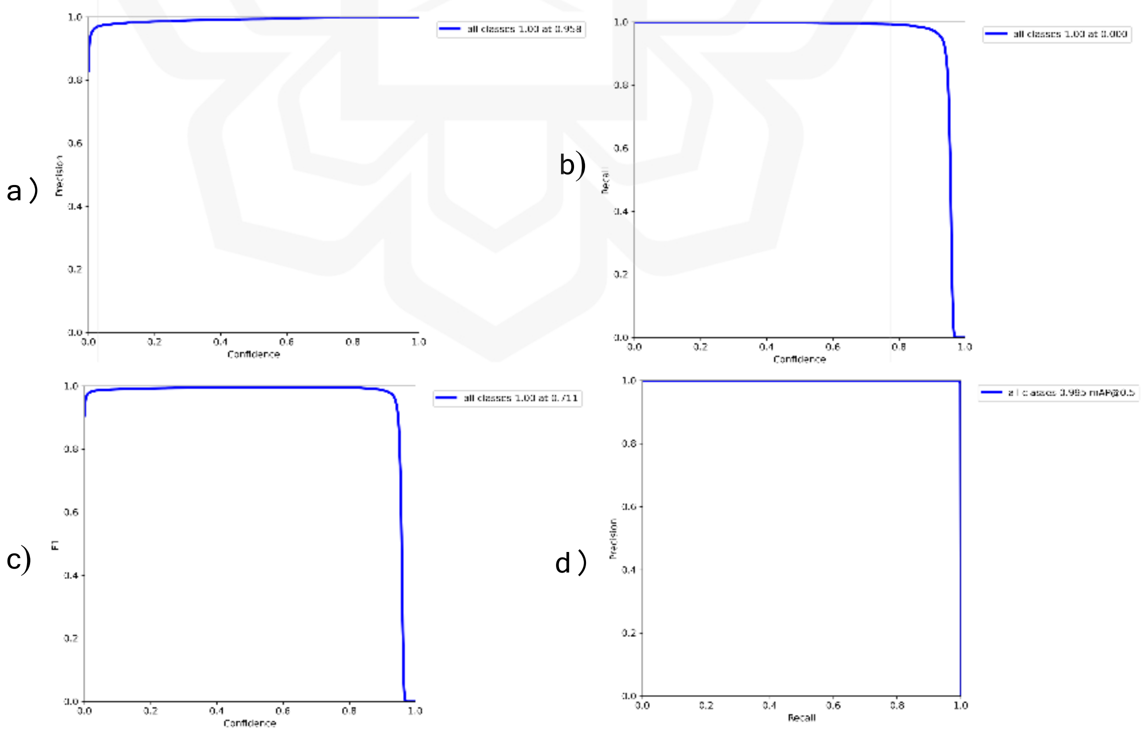


Figure 4.6 Different Training Metrics for LPD on a Massive Dataset Using YOLOv5 a) Precision Curve, b) Recall Curve, c) F1 Curve, and d) PR Curve

The training of over 1000 epochs on YOLOv5 resulted in an overall mAP of 99.48% for the license plate detection task. The Precision, Recall, and F1 score performance and precision-recall curve can be seen in Figure 4.6, showing its robust performance on massive datasets. Moreover, mAP and mAP95 improvement per epoch is illustrated in Figure 4.7.

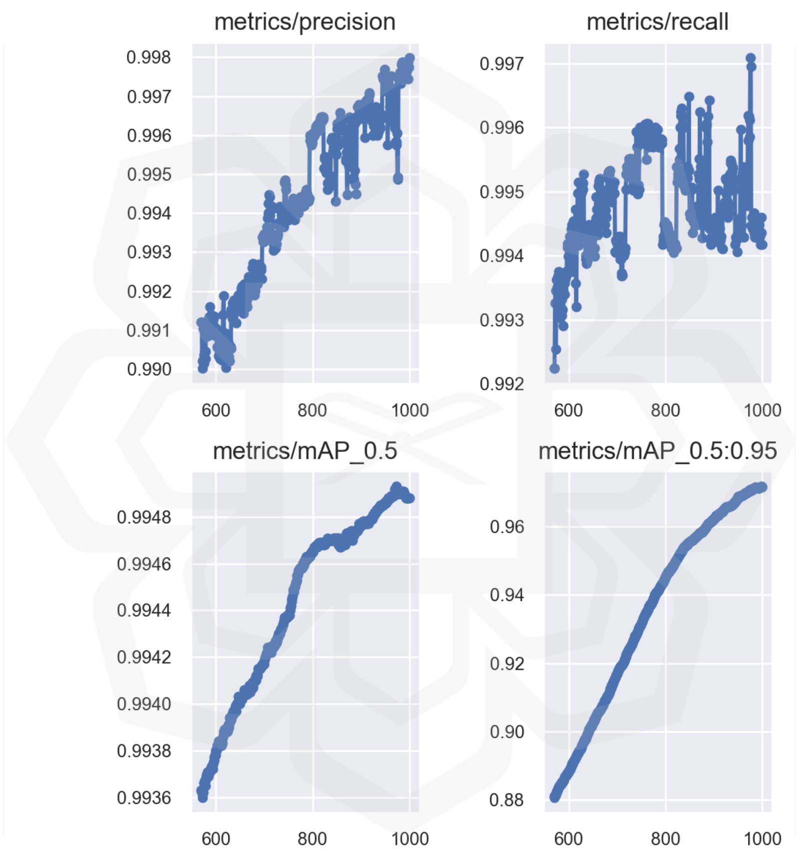


Figure 4.7 Training Results of LPD Stage on YOLOv5

Figure 4.8 Samples of LP Detections on YOLOv5

The performance of LPD on field tests and test datasets shown in Figure 4.8 depicts very challenging conditions of low light properly illuminated and various distances in the toll camera region of interest. It signifies the robustness of LPD achieved. On-field testing was also reported with over 99.95% accuracy.

4.3.3 Benchmarking Result

This section explains the result of two versions of YOLO included in the training process and compared. For comparison of YOLOv3 and YOLOv5, the models have been trained on the same size dataset. Both models are trained for 400 epochs for benchmarking purposes, as mentioned in Table 4.1. The source dataset (vehicle images) with 16,000 images and their corresponding labels with YOLO format is fed to the LPD network to train the model to localize and classify the license plate. All datasets are collected from Malaysian toll plazas and labeled manually using the labelimg software tool.

Table 4.1 Comparison of LPD Training results with YOLOv3 and YOLOv5

Model Version	**Dataset**	**Number of Epochs**	**Results**			**Accuracy**
			P	**R**	**mAP 0.5**	
YOLOv5	15,953	400	99.79%	99.41%	99.48%	98.8%
YOLOv3	15,953	400	99.94%	100%	99.68%	99.95%

The same test data is used to validate the model and calculate the recall and rates. The results obtained by the networks trained to train the dataset containing vehicle images are reported in Table 4.1. From the Table, YOLOv3 recorded a higher mAP rate than YOLOv5. The result of 99.86% mAP YOLOv3 was obtained after 400 epochs, while YOLOv5 reached the rate of mAP with 99,48%. This unexpected, better result of YOLOv3 can be interpreted by the fact that the LP region is small in the images, and the YOLOv3 network was originally designed to detect small objects. In addition, in deep learning, the more complicated models show a degradable performance for simple recognition tasks. However, field accuracy on YOLOv5 is found to be higher and used in a final pipeline. Finally, the deployed model of YOLOv5 is trained for 1000 epochs.

4.4 LICENSE PLATE RECOGNITION (LPR)

The LPR stage is more complicated and challenging due to the more complex features in the objects (characters) to be detected and the wide variations in Malaysian LP designs. Therefore, it is required to acquire much more images for this stage. As we want to ensure the LPD model's performance is high, we formed a bigger dataset for the LPR stage. The collection of the LPR dataset has been done in many batches with gradual increments on every batch. For each batch, training is carried out, and when the training shows very high mAP, the model is tested on an unseen dataset. Figure 4.9 shows the results of experiments conducted on the YOLOv3 model with many batches and a different number of epochs. Therefore, the increment of the dataset went on up to the point that we reached 40,000 images with 29,700 real-world images and 10,500 images resulting from augmentation.

The results show a big difference between the training results and accuracy results. Additionally, as the size of the training dataset becomes larger, the accuracy improves. This indicates the impact of the dataset size, which contains more variations on the test accuracy.

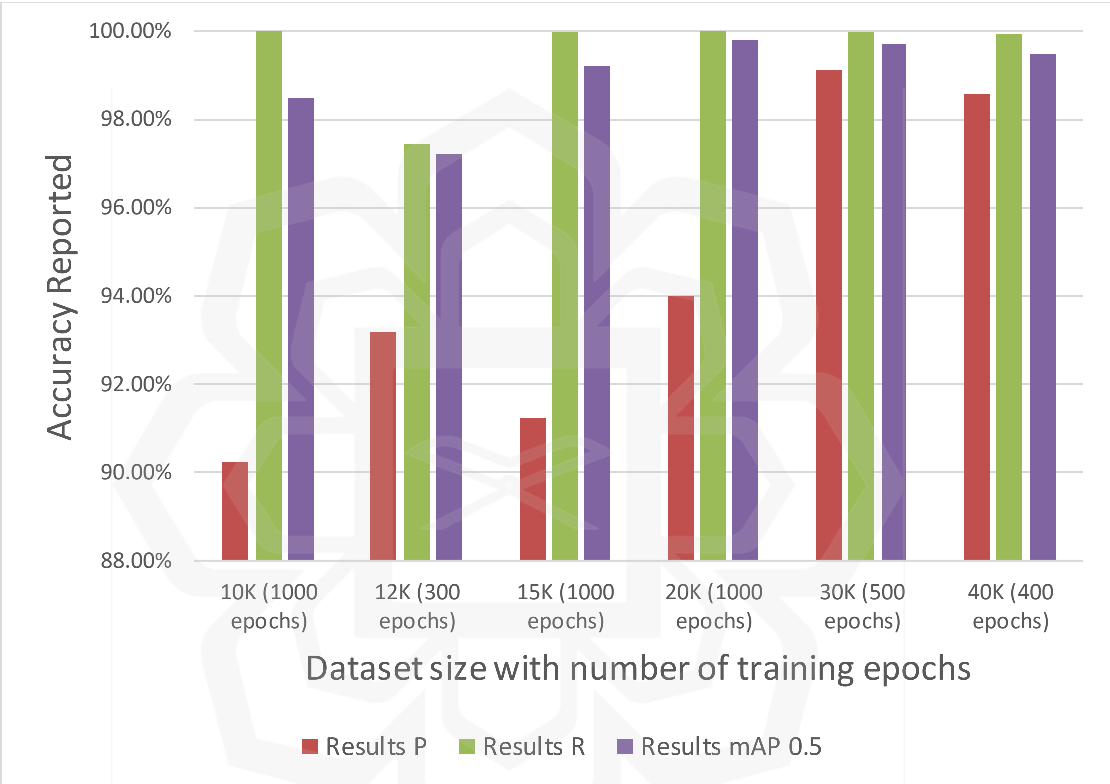


Figure 4.9 LPR Training Results with YOLOv3

The performance in the LP recognition stage is compared with other object detection (OD) methods. The study performed by (Tourani et al., 2020) compared object detection networks and showed the performance of YOLOv3 over other OD methods like Fast R-CNN, Faster R-CNN, and YOLOv2. To our knowledge, no ALPR studies have compared YOLOv3 and YOLOv5. We performed a study on both models to decide which model would be selected. To make a fair comparison, the networks were trained on the same dataset (40K images) and a number of epochs (400 epochs) and tested on the same test dataset, as shown in Table 4.2.

Table 4.2 Comparison between YOLOv3 and YOLOv5 in the LPR Stage

Method	**Performance Evaluation**		
	P	**R**	**mAP 0.5**
YOLOv3	0.979	0.989	0.991
YOLOv5 (Proposed)	0.978	0.972	0.993

As shown in Table 4.2, YOLOv5 outperformed YOLOv3 for this stage. This improvement is due to replacing the first three layers of YOLOv3 with a single layer (Focus layer) and the Auto-learning bounding box anchors in YOLOv5 (Souidene Mseddi et al., 2021). As a result, we decided to select YOLOv5 as opposed to YOLOv3 in the LPR stage due to the high accuracy YOLOv5. From this point, we continued conducting more experiments on the YOLOv5 version with a different dataset. The following subsection explores these experiments.

4.4.1 Experiments with Dataset Variations in the Real-world Dataset

Table 4.3 summarizes the experiments on the YOLOv5 network utilizing different dataset sizes. It is worth mentioning that the YOLOv5s6 version is utilized in training due to the speed/accuracy trade-off this version offers. A comparison between YOLOv5 versions based on the COCOval2017 dataset is depicted in Figure 4.10. As the figure shows, YOLOv5s6 records 63.7 mAP at 0.50 IoU, which is in the middle compared to most other models. Although YOLOv5s also recorded a medium speed with an inference time of 8.2 ms per image on GPU, a real-time system with a little sacrificing accuracy would be the best choice, as accuracy can be compensated for with a comprehensive dataset and enough epochs number. Therefore, YOLOv5s6 has been selected for the system as it provides a comparatively high mAP with less execution time and small parameters size that makes it suitable for real-time systems and also on GPUs with less power.

Table 4.3 LPR Experiments Conducted on YOLOv5

Experiment	Dataset	Labels	Number of Epochs	Results			Accuracy
				P	R	mAP 0.5	
Experiment1	40K+	267,656	1000	99.84%	99.88%	99.63%	94%
Experiment2	18,724	125,662	800	93.49%	99.76%	96.35%	53%
Experiment3	58,240	390,149	1200	96.11%	99.6%	98.13%	95%

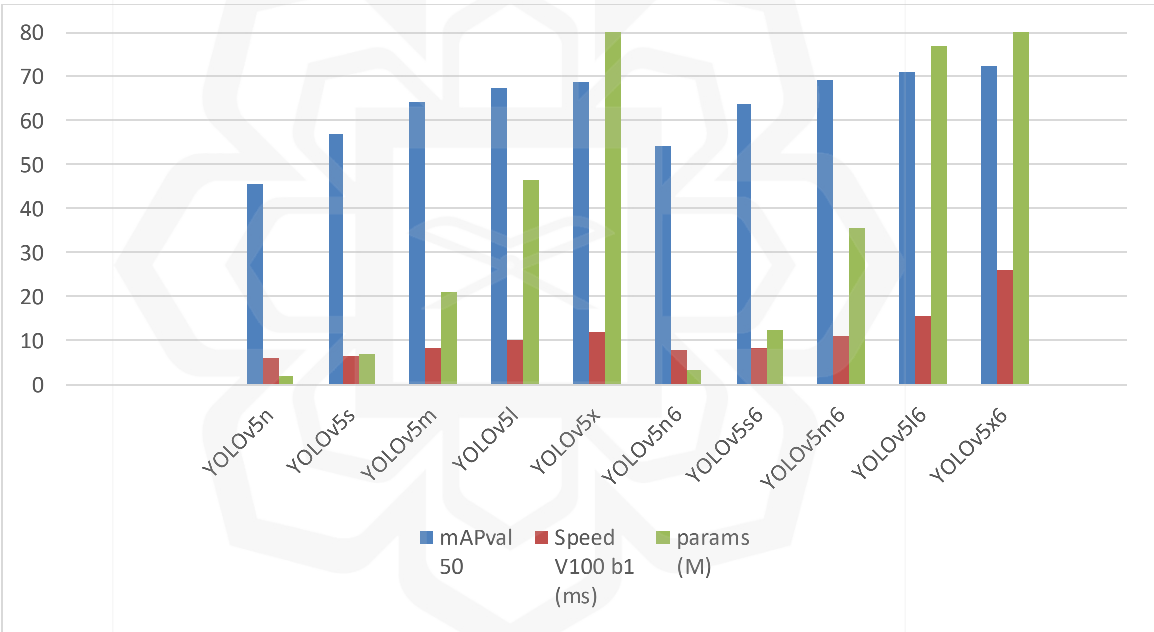


Figure 4.10 Comparison between YOLOv5 Checkpoint Versions

4.4.2 Real Dataset with Augmentation Results

The various experiments on real LP datasets are carried out, and experiment 1 details are discussed here. The network YOLOv5 has been utilized for training a model with 40K images containing 29,700 real images; the rest were obtained from the augmentation process. The real images are wide range but not clean, i.e., broken and occluded LP images. In addition, very bad-looking images are included, such as images with very low resolution. Moreover, there was no systematic and controlled augmentation process for this dataset but randomly generated.

Over 40 thousand images used in this training are collected from different toll booths over different parts of the year to capture variation in different weather conditions and are processed and labeled for training purposes for LPR purposes. The data details used for training purposes are visualized in Figure 4.11 with nearly 37 classes. The training performance of over 500 epochs for precision, recall, and mAP is illustrated in Figure 4.12. The gray line in various metric curves of Figure 4.13 shows the performance of various classes, and the blue line shows the performance of a system.

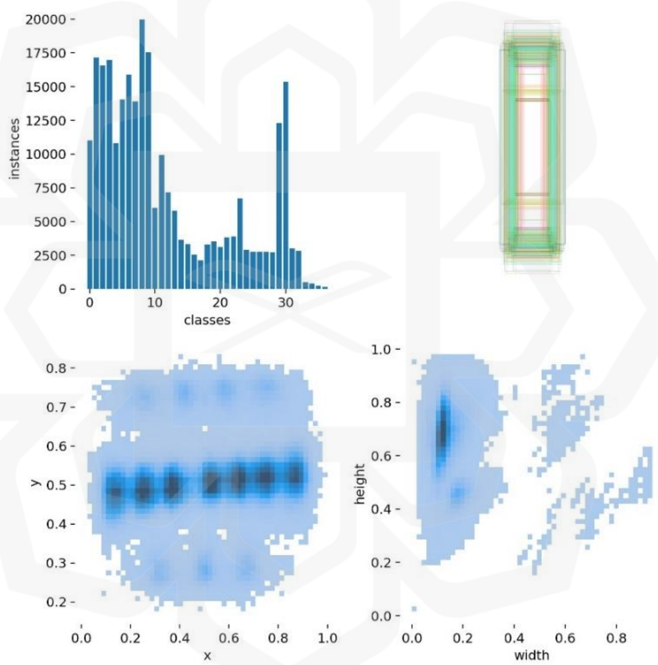


Figure 4.11 Label Distribution in the LPR Training on YOLOv5 with Experiment 1

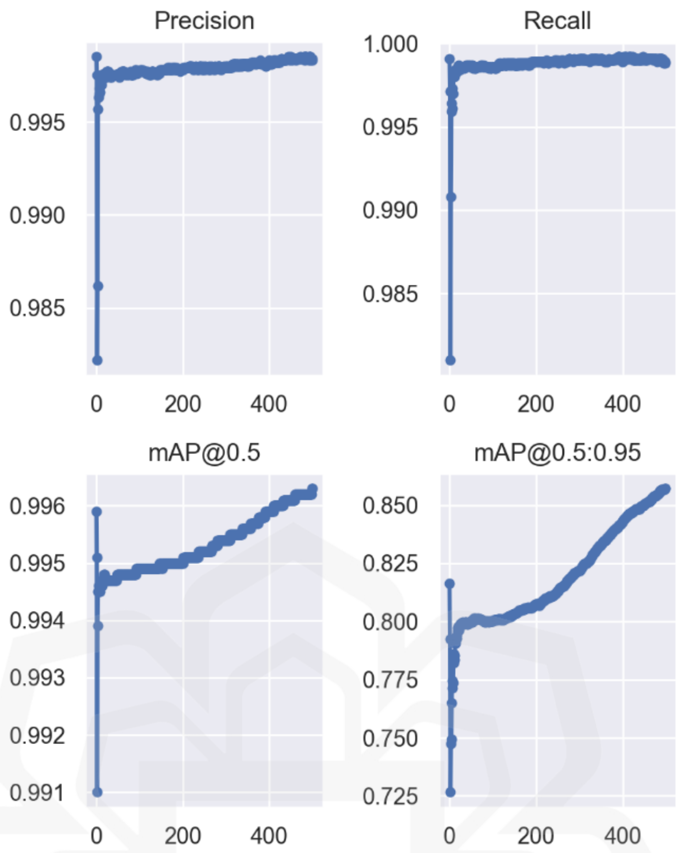


Figure 4.12 Training Results of LPR Stage on YOLOv5 (Experiment 1)

The performance of this model reported an overall mAP of 99.6% and a Precision of 93.3%. Figure 4.14 shows the confusion matrix for this trained model. The alphanumeric recognition has nearly no confusion. However, Special LP has lower mAP. To note that the validation dataset was the same as the training dataset, the model reported 99.61% mAP after 500 epochs.

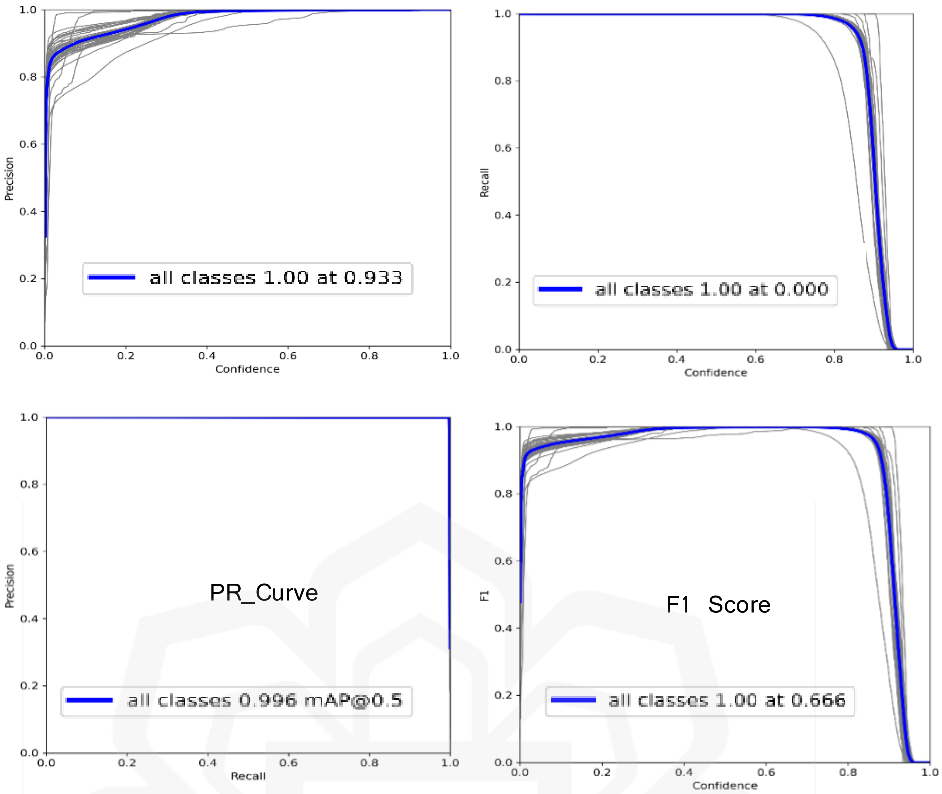


Figure 4.13 Validation Results of LPR on YOLOv5 Model Experiment 1

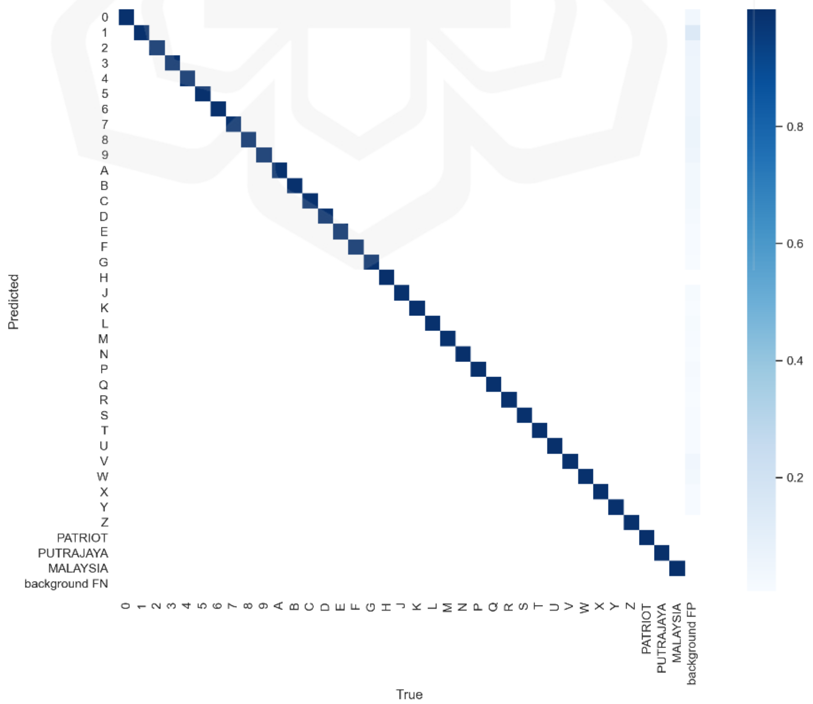


Figure 4.14 Confusion Matrix of LPR Results on YOLOv5 (Experiment 1)

As results indicate, the system achieved impressive overall performance metrics, including an mAP of 99.6% and a precision of 93.3%. However, it is important to consider the impact of bad images on the accuracy and reliability of the model in real-world testing scenarios. The system could not surpass the accuracy of 94% despite the increment in the number of training epochs. Moreover, the lack of systematic and controlled augmentation process for this dataset may contribute to the inclusion of inconsistent dataset that may lead to this result.

While the trained model demonstrated high performance metrics on the given dataset, it is important to note that the accuracy and reliability of the model can be compromised when encountering similar bad images or real-world scenarios that differ significantly from the training data. The presence of broken, occluded, or low-resolution license plate images in the training data might lead the model to learn patterns and features that are specific to those instances, making it less robust when faced with similar variations in real testing scenarios.

Therefore, it is crucial to ensure that the training dataset encompasses a diverse range of real-world scenarios, including variations in license plate appearance, environmental conditions, and image quality. By excluding the images that do not represent the normal LP, the model's accuracy and reliability in real testing can be improved in general. Additionally, a controlled and systematic augmentation process can be implemented to generate augmented images that simulate real-world variations more effectively.

4.4.3 Clean Real Dataset without Augmentation Results

The collected data had much noise in the form of broken plates, missing characters, and low-resolution images. This dataset was cleaned manually through human-level verification. The result of the model trained on this data is discussed in the coming paragraph.

The 18,000 LP image data are obtained after cleaning from over 29,000 images. The class distribution details used for training are visualized in Figure 4.15 with 37 classes. The training performance of over 800 epochs for precision, recall, and mAP is illustrated in Figure 4.17. The gray line in various metric curves of Figure 4.16 shows the performance of various classes, and the blue line shows the performance of a system. The three gray lines in the lower half of each metric in this image show the poor performance on special LP.

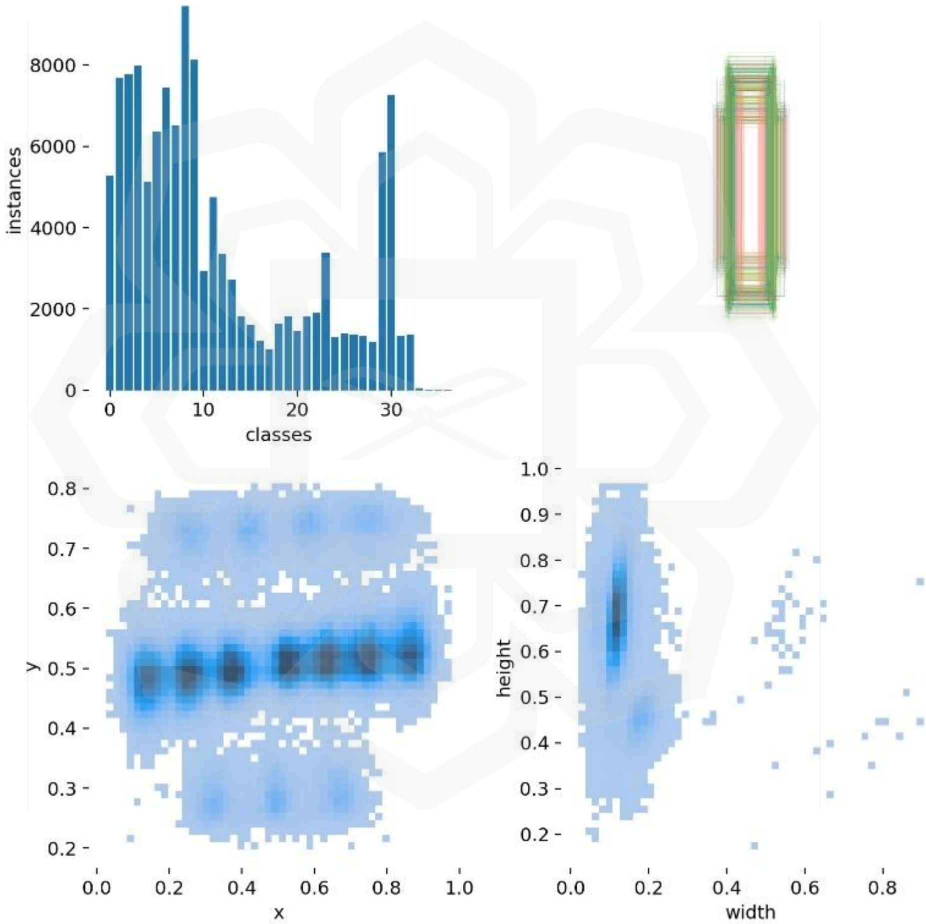


Figure 4.15 Label Distribution in the LPR training on YOLOv5 with Experiment 2

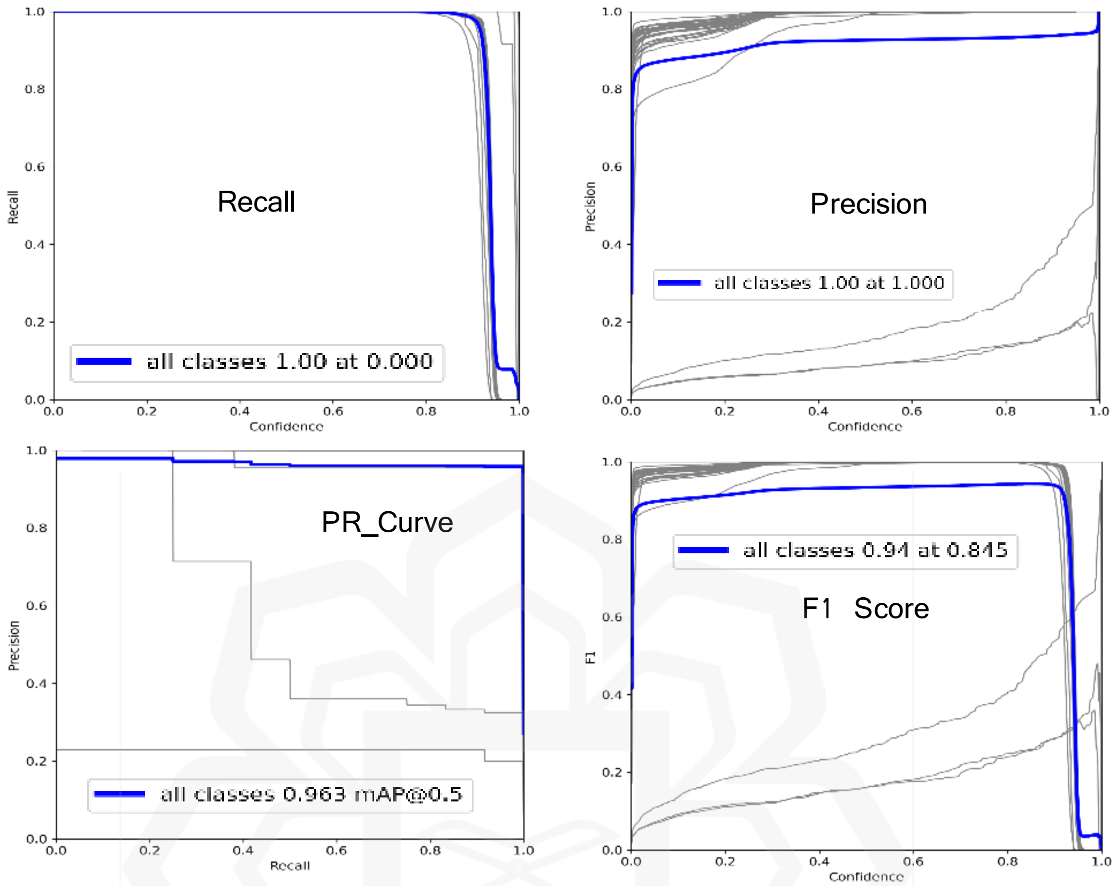


Figure 4.16 Validation Results of LPR on YOLOv5 Model Experiment 2

The performance of this model reported an overall mAP of 96.35%. Figure 4.18 shows the confusion matrix for this trained model. The alphanumeric recognition has nearly no confusion. The special LP model has very high confusion, which degrades the overall LP performance and results in lower mAP. The reported accuracy of the model on the test dataset is just 53%. So, we conclude that clean small datasets seems to not be used for a robust ALPR model.

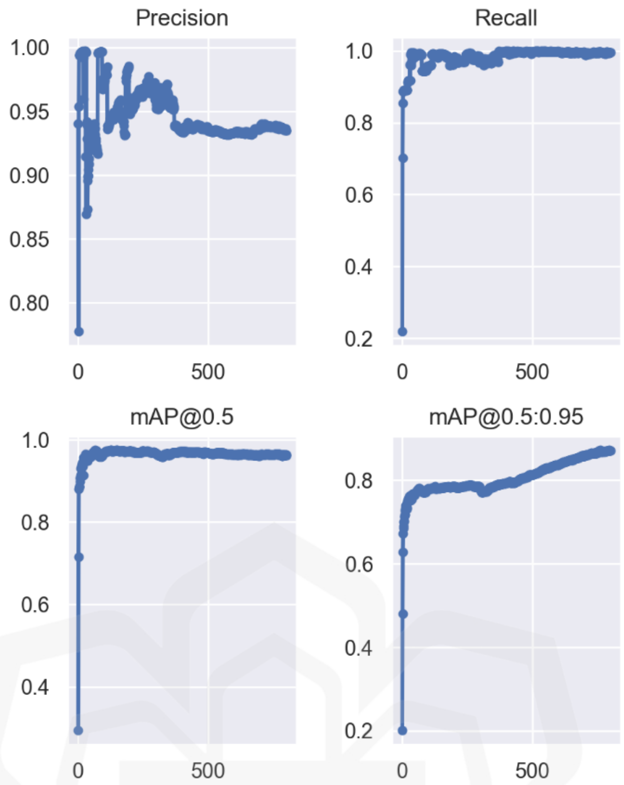


Figure 4.17 Training Results of LPR Stage on YOLOv5 (Experiment 2)

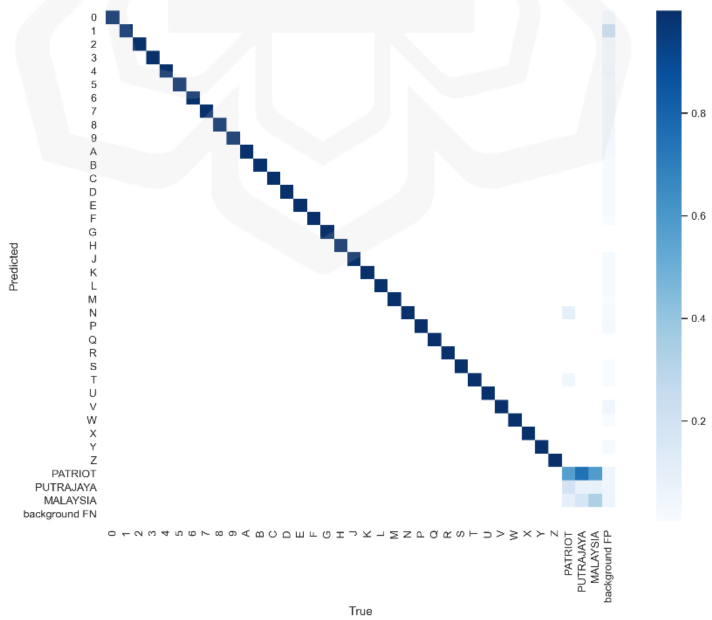


Figure 4.18 Confusion Matrix of LPR Results on YOLOv5 (Experiment 2)

To discuss the results from this experiment, the dataset underwent manual cleaning to address issues such as broken plates, missing characters, and low-resolution images. However, despite the cleaning process, the dataset lacked variations and real-world complexities, which had a significant impact on the accuracy of the system.

The model's reported overall mAP of 96.35% seems relatively high, but it is essential to consider the context. Figure 4.18, displaying the confusion matrix, reveals that the model performs well in recognizing alphanumeric characters with minimal confusion. However, when it comes to special LPs, there is a high level of confusion, leading to a degradation in overall LP performance.

Furthermore, the reported accuracy of the model on the test dataset being just 53% highlights the limitations of using a clean and straightforward dataset for training a robust ALPR model. To achieve higher accuracy and real-world applicability, it is crucial to introduce variations that represent the complexities and challenges encountered in actual scenarios. One way to address this limitation is by incorporating a controlled and systematic augmentation process that encompasses real-world variations.

4.4.4 Clean and Augmented Real Dataset Results

A clean dataset of an additional 10,000 images is collected for another three months and is further augmented with the following strategy to make a robust model. After removing the broken and blurred images from a dataset, we did some augmentation on the clean dataset. We took 10,000 images, divided them into eight groups, and applied different augmentation strategies since having diversity and variability in images is important. Applying augmentation on different image groups is to avoid over-represented classes. Table 4.4 shows the augmentation work done on the chosen dataset.

Table 4.4 Augmentation Operations Carried out on the Clean Dataset

#Source images	Augmented Images	Action Applied
1000	3000	Rotation -5,+5 ; Exposure -10,+10; Noise 5%
1000	3000	Rotation Between -7° and +7° Brightness Between -20% and +20% Blur Up to 0.5px
1000	3000	Rotation Between -10° and +10° Exposure Between -20% and +20% Blur Up to 0.75px
1000	3000	Shear: ±5° Horizontal, ±0° Vertical Exposure: Between -15% and +15% Noise: Up to 7% of pixels
1000	3000	Shear: ±10° Horizontal, ±0° Brightness Between -25% and +25% Blur Up to 0.75px
1000	3000	Exposure Between -20% and +20% Blur: Up to 1px Noise: Up to 8% of pixels
2000	6000	Shear 8° Brightness Between -30% and +30% Noise 5%
2000	6000	Rotation: Between -7° and +7° Blur: Up to 0.5px Noise: Up to 10% of pixels

Table 4.4 shows that from the 10K clean dataset, the 30K image dataset is generated with wide variation in scenes. This diversity in the dataset will help get better performance for the model. Accuracy is expected to be higher due to our augmented dataset's higher possibility of covering real-world instances. Additionally, a model will

not overfit when tested on a new dataset. Finally, we developed more than 58K images for Malaysian LP with 28K original images and 30K augmented dataset. This relatively huge dataset is necessary to have all classes with reasonable instances and to cover all standard and nonstandard designs in LP, which is popular in Malaysian LP.

The experiment 3 model is trained on the clean and augmented dataset of 58K images. Figure 4.19 shows the details of the dataset here and its label size. The labels' histogram shows that a few classes have nearly 30,000 instances, whereas special LPs have fewer than 300 instances. Moreover, the heat map of many labels is near the center of the y-axis. However, with augmentation, labels are moved along the y-axis for robust recognition.

The training of 1200 epochs on YOLOv5 resulted in an overall mAP of 98.1% for the license plate detection task. The Precision, Recall, and F1 score performance and precision-recall curve can be seen in Figure 4.20, showing its robust performance on massive datasets. Moreover, mAP at 0.5 and mAP at 0.5-0.95 improvement per epoch is illustrated in Figure 4.20.

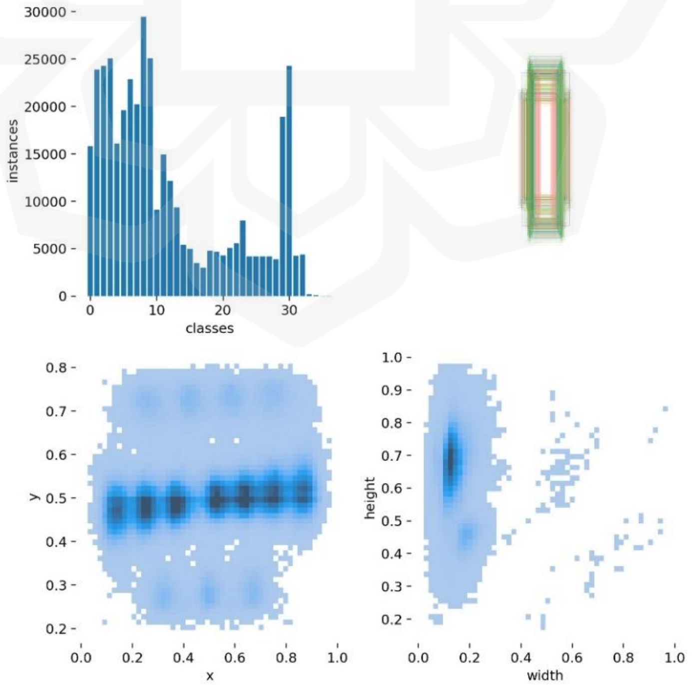


Figure 4.19 Label Distribution in the LPR Training on YOLOv5 with Experiment 3

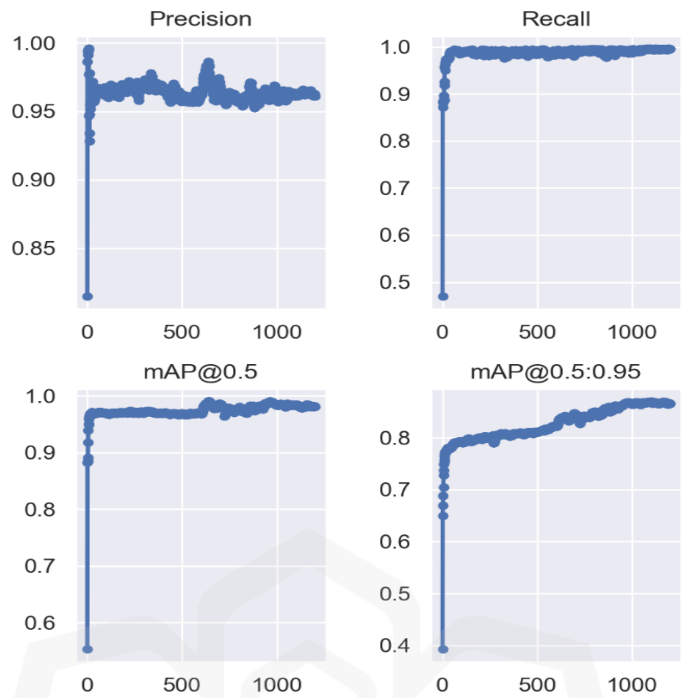


Figure 4.20 Training Results of LPR Stage on YOLOv5 (Experiment 3)

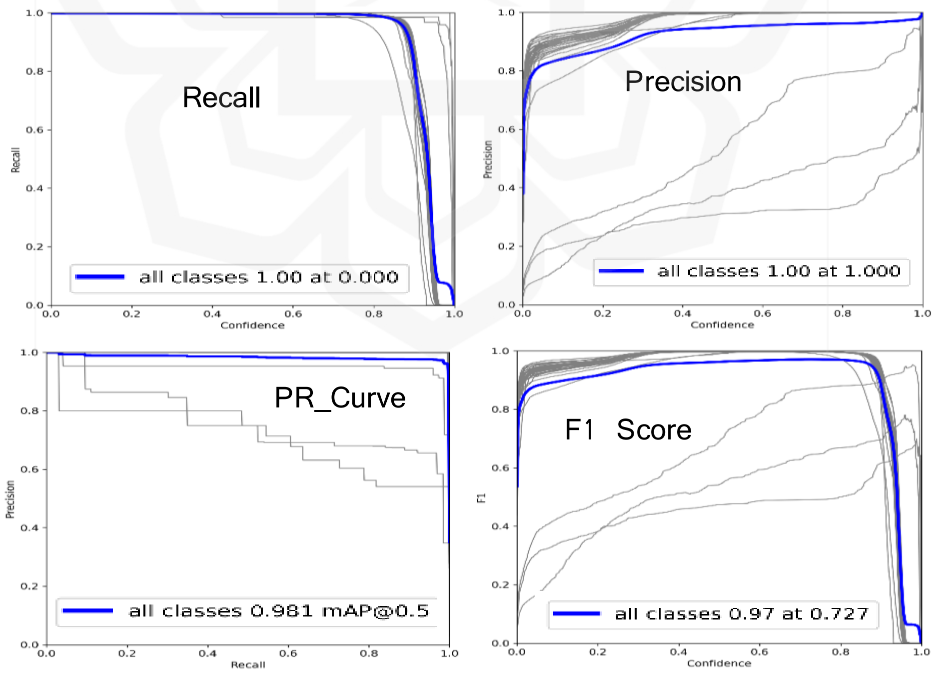


Figure 4.21 Validation Results of LPR on YOLOv5 Model (Experiment 3)

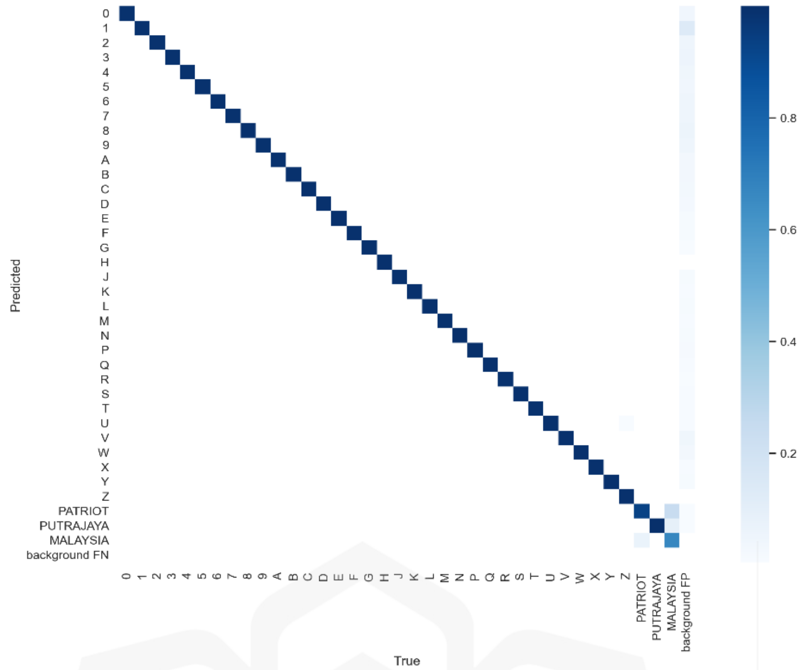


Figure 4.22 Confusion Matrix of LPR Results on YOLOv5 (Experiment 3)

The performance of LPR on field tests and test datasets shown in Figure 4.21 depicts that the performance of special LP is still not accurate with lower confusion among a few alphabets such as Z. It signifies that the robustness of LPR cannot be achieved by making large datasets. As shown in Figure 4.22, there is a lot more confusion about special plates because there is an imbalance in the dataset. Table 4.5 shows the mAP of each class. There is a large disparity between the well-represented and the under-represented classes, which comprise only 0.1% of the population. However, it ends up being the best model of all its predecessors. On-field testing was also reported with over 95% accuracy.

Table 4.5 Detailed Results of Experiment 3 on YOLOv5

Class	**Labels**	**P**	**R**	**mAP**	**Class**	**Labels**	**P**	**R**	**mAP**
All	390149	0.961	0.996	0.981	J	4812	0.999	0.998	0.997
0	15820	0.999	0.996	0.997	K	4680	1	0.998	0.997
1	23889	0.998	0.986	0.997	L	4282	1	0.998	0.997
2	24281	0.999	0.996	0.997	M	5095	1	0.998	0.997
3	25054	0.999	0.995	0.997	N	5582	0.999	0.999	0.997
4	16069	1	0.996	0.997	P	8002	0.999	0.997	0.996
5	19578	0.998	0.995	0.997	Q	4188	0.999	0.998	0.996
6	22859	0.998	0.996	0.997	R	4199	0.999	0.997	0.996
7	20249	0.999	0.995	0.997	S	4204	0.999	0.997	0.996
8	29472	0.999	0.995	0.997	T	4224	1	0.997	0.997
9	25119	0.999	0.996	0.997	U	3881	1	0.997	0.996
A	9113	1	0.996	0.997	V	18875	0.999	0.998	0.998
B	14971	1	0.997	0.997	W	24262	0.999	0.996	0.997
C	12143	0.999	0.997	0.997	X	4273	1	0.995	0.996
D	9324	1	0.998	0.997	Y	4372	0.999	0.996	0.996
E	5400	0.999	0.998	0.996	Z	187	0.999	1	0.996
F	5011	0.999	0.998	0.996	PATRIOT	74	0.781	.986	.974
G	3519	0.999	0.998	0.996	PUTRAJAYA	63	0.481	0.984	0761
H	2990	0.999	0.996	0.996	MALAYSIA	33	0.326	1	0.711

Table 4.6 Wrongly Detected Classes during Inference Testing on Model Resulted from Experiment 3 on YOLOv5

True Class	**Detected Class**
0	8
9	8
D	B
Y	V
E	F
Z	2

From the on-field test, the commonly noticed confusion errors are listed in Table 4.6. The 'Z' is confused since it is a less-represented class (Z) with only 187 samples among the alphabet. The letter 'B' has more training than 'D'; hence, 'D' is confused with 'B.' In comparison, some classes have several instances with the range 2990 and 5582 (E, F, G, H, J, K, L, M, N, Q, R, S, T, U, X, Y) and classes within the range 8002 and 9324 (A, B, P). At the same time, the rest of the classes are with high samples.

Even though the required operational accuracy is achieved with this model, the accuracy of special LP recognition was a persistent problem, along with the confusion mentioned in Table 4.6. The said problems cannot be solved by merely collecting more data. Augmentation also preserves these imbalances. Therefore, a new method to address this issue was developed. Its finding will be discussed in more detail in the next section.

4.4.5 Experiments with Synthetic Dataset

The targeted operational accuracy resulting from the experiment 3 model is above 95% and technically Acceptable, but we kept the benchmark of 99% accuracy for complete LP. For the benchmarking, we did not consider the character recognition level accuracy. For this purpose, we have to take another step of building a synthetic dataset in order to enhance the performance. It is observed that the augmentation strategy could no longer increase the accuracy. There is still some confusion in some class recognition, such as in Table 4.6.

By observing the class instances in Figure 4.19, it can be seen that all the true detected classes are under-represented. In contrast, the false detection classes are over-represented. Thus, having a balanced dataset is expected to solve this problem. To establish this observation, further experiments were carried out.

As per the methodology for generating synthetic LP in chapter three, Section 3.4, the synthetic dataset was generated, and the experiment with combined real and synthetic data model was compared with the only real data model. The resulting combined dataset class distribution is shown in Figure 4.23. The histogram has more balancing among alphabets and special LPs than the real and augmented dataset. The resulting dataset details and comparison is Tabled in Table 4.7.

Table 4.7 Experiments Conducted for Highlighting Synthetic Dataset Impact on the Results

Experiment	**Train Dataset**	**Test Dataset**	**Results**		
			P	**R**	**mAP 0.5**
Only Real Dataset	58K images	13.7K images	0.815	0.849	0.855
Combined (Real and Synthetic) Dataset	176K images	13.7K images	0.943	0.951	0.976

Two experiments have been conducted to demonstrate the ability of real data combined with synthetic datasets to improvise the results. Two models are trained, the first on the real-world collected dataset (58K images) and the second model trained on a combined real and synthetic dataset (176K images). Both models were validated on the same validation dataset (13.7K images) containing real-world and synthetic datasets.

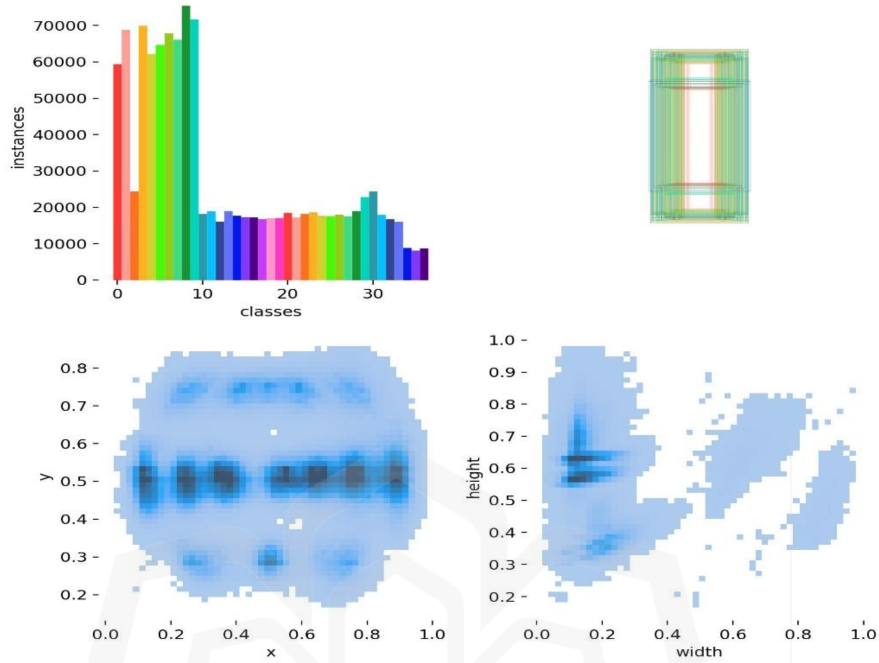


Figure 4.23 Label Distribution in the LPR training on YOLOv5 with Combined Dataset (Real and Synthetic Dataset)

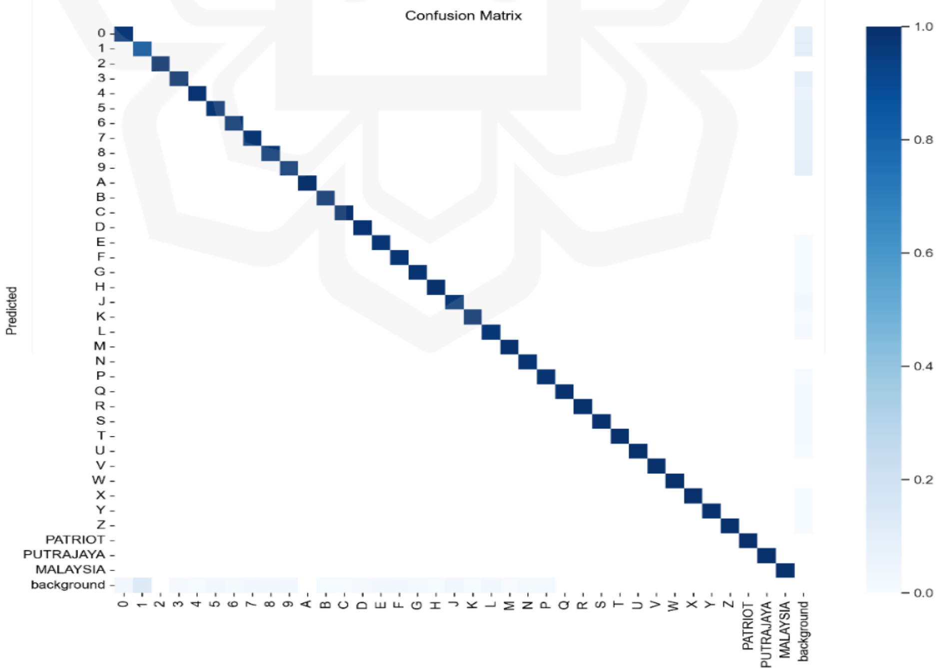


Figure 4.24 Confusion Matrix of LPR Results (Combined Test Dataset) on YOLOv5 Model with Combined Training Dataset (Real and Synthetic Dataset)

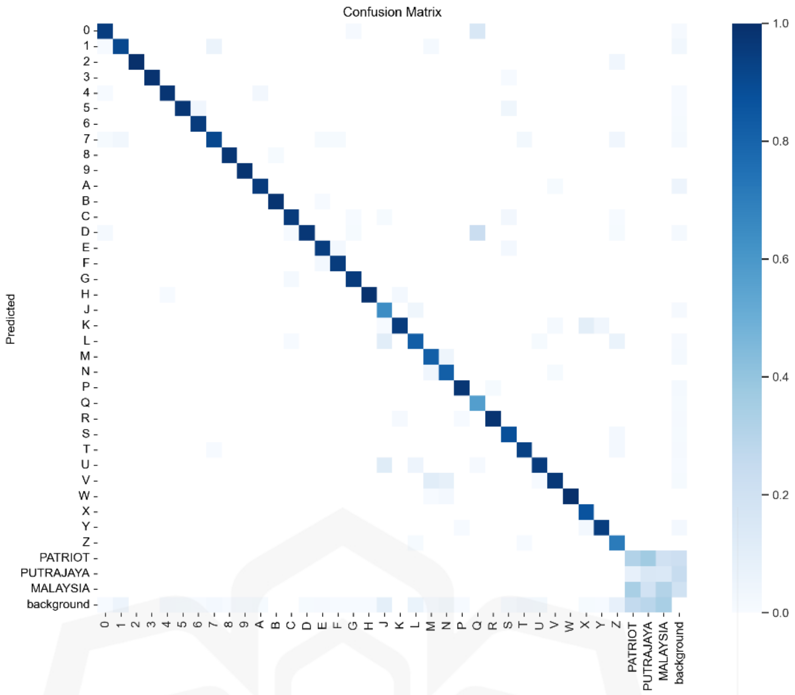


Figure 4.25 Confusion Matrix of LPR Results (Combined Test Dataset) on YOLOv5 Model with Only Real-world Training Dataset (Real Dataset)

The combined model diminishes the confusion among all the classes, as evident from Figure 4.24. The model trained on a combined dataset resulted in over 12% better performance with 97.6% overall mAP. The confusion of the only real dataset can be seen in Figure 4.25. Figure 4.26 a) shows the actual LP and b) shows the inference from the combined training model. At the same time, 4.26 c) shows that only the real dataset LPR model has failed in all these cases. LPR (Combined Test Dataset) validation results on the YOLOv5 model with Combined Training Dataset (Real and Synthetic Dataset) are quite superior to real LP datasets and training with the same number of epochs.

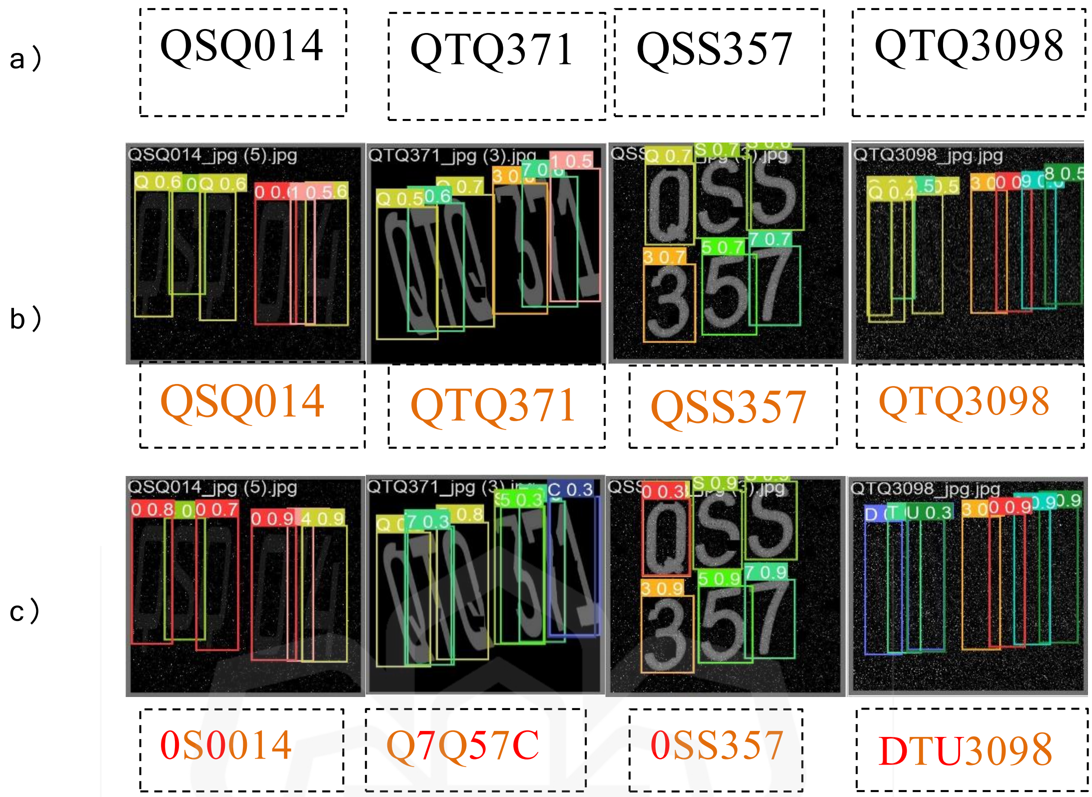


Figure 4.26 Samples of the Combined Testing LPR Dataset on YOLOv5 Models Trained with Combined Dataset (Real and Synthetic Dataset) and Only Real-world Dataset. a) The Actual LP ID. b) Results from the Model Trained on Combined Dataset. c) Results From Model Train

The results indicate that combining real-world data with a synthetic dataset led to improved performance in license plate recognition. This improvement can be attributed to several factors. Firstly, the more balanced class distribution that introduced in Chapter three, Section 3.4, helped achieve a more balanced class distribution among alphabets and special license plates (LPs) compared to the real and augmented datasets. Figure 4.23 demonstrates that the combined dataset had a more evenly distributed representation of classes. This balancing of class instances ensures that the model receives sufficient exposure to all classes, mitigating the issues caused by under-represented classes. Additionally, the synthetic dataset introduced a wider range of variations and scenes, increasing the diversity and variability within the combined dataset. This diversity is crucial for training a robust model capable of handling real-world instances. By incorporating synthetic data, the model becomes more exposed to different scenarios, leading to better performance when faced with unseen data during testing.

Moreover, the introduction of synthetic data, which covers wide variations, helps the model learn to distinguish between similar classes more effectively. As a result, the combined model exhibits improved accuracy and reduces instances of misclassification as shown in Figure 4.26. Furthermore, The combined dataset leverages the complementary strengths of both real-world and synthetic data. Real-world data captures the nuances and characteristics of actual license plates, while synthetic data enables controlled variations and diversification. The synergy between these two types of data enhances the model's ability to handle various real-world scenarios effectively.

The combined model achieved an overall mAP of 97.6%, outperforming the model trained solely on real data. This further supports the effectiveness and robustness of the proposed method.

4.5 SUMMARY

In this chapter, we first evaluated each stage of ALPR to provide a realistic evaluation of the overall ALPR system, in which well-performed LP detection and recognition are critical for achieving overall ALPR system accuracy. Following that, the end-to-end ALPR system is evaluated using model-centric and data-centric strategy. We found that YOLOv5 is a better model for ALPR system based on the higher accuracy was achieved. Several experiments have been conducted to explore the role of the dataset in improving the ALPR system. First, with randomly augmented and unclean dataset, the model performance reported 99.63% mAP and accuracy of 94%. After that, the dataset have been cleaned from broken and occluded images. The system achieved a relatively high mAP with 96.35%. However, the reported accuracy was only 53% on the test dataset. That led to addressing the systematic augmentation to the clean dataset to increase diversity and variation. The model was able to surpass 98% of mAP and moreover, the accuracy improved on the test dataset achieving 95% which was the best among the previous experiments. Finally, the real and augmented dataset has been combined with the systematically generated synthetic dataset. The combined dataset strategy with synthetic, real and augmented data with balanced classes representation is more robust and effective. This is demonstrated by the achieved results from the two models trained with the only real, augmented dataset and the combined dataset which showed a better mAP with increment of 12% when validated on the test dataset.

CHAPTER FIVE

CONCLUSION

In this final chapter of the thesis, we present the overall conclusion, followed by suggestions for future research that could enhance this work.

5.1 CONCLUSION

In this work, we presented an efficient end-to-end Malaysian ALPR system using the state-of-the-art YOLO object detection network. The first model was used for LP detection. Then, we detected and recognized all LP characters simultaneously using YOLOv5. We performed the data-centric method to train the networks to achieve the system's best speed/accuracy trade-off.

Firstly, we aimed to develop of a comprehensive, customized Malaysian ALPR dataset using a real-time dataset and a newly proposed synthetic custom dataset generation technique. In this regard, a dataset for Malaysian ALPR includes more than 176,000 fully annotated images, from which 58,000 images were collected from real-world scenarios, and the rest were synthetically produced. The dataset is collected for over 18 months in different locations of Malaysia, from coastal Penang to KL. All images in the real-world dataset were labeled manually for the two stages. However, we proposed a method that performs labeling automatically for the synthetic dataset. It is worth noting that there is no such dataset publicly available for Malaysian ALPR. Thus, we will make this dataset publicly available to the research community helping with the development and improving Malaysian ALPR systems and benchmarking the dataset for fair comparison in the conducted studies.

Secondly, we utilized the state-of-art object detection networks YOLOv3 and YOLOv5 for the training process in both LPD and LPR to implement a segmentation-free and two-stage hierarchical ALPR system. Several experiments have been conducted in the two stages to reach the best model. Moreover, since the LP recognition stage is considered an ALPR systems' bottleneck, most of the work is focused on this stage, and many ablation studies based on the data-centric approach are carried out.

Lastly, we evaluated the system on a new, realistic, and large-scale Malaysian ALPR testing dataset covering a lot of real-world challenges. A separate dataset with 1,040 images that embraces real-world images with wide variations is introduced. Additionally, a synthetically generated test dataset with augmenting is proposed to increase the challenge.

On the real dataset, which was gathered from several toll plazas throughout Malaysia and contained wide environmental distinctions with more than 50,000 labeled images, our system had a recognition rate of 98.1% mAP. Furthermore, the combined dataset from real and synthetic images showed a remarkable improvement in the system performance over the only real-world dataset when the model was trained with the same number of epochs with mAP of 97.6% and 85,5%, respectively. The system's end-to-end accuracy was given using a complicated test dataset in an open environment with low visibility, and the result was 95.96%.

The results demonstrated that the proposed dataset that compromised both real and synthetic datasets is the most robust since it could contribute to a higher system performance on a challenging environment and test dataset.

5.2 FUTURE WORK

In future work, we intend to improve the current system by providing more data-centric techniques. The enhancement aims to perform an automatic refinement on the real dataset labels as they were done by human work, which is not perfectly labeled. In addition, we intend to include other types of Malaysian LPs not included in this work, such as public LPs -"Taxi LPs"- in both real and synthetic datasets. Moreover, the synthetic dataset is intended to have different characters' allocations to simulate the real-world dataset.

In addition, we plan to use other object detection systems, such as YOLOv8, for ALPR to design a system that achieves higher recognition rates or processes images at a lower computational cost. We also plan to perform a model-centric approach for the ALPR system by playing with the models' hyperparameters to tune the network based on the best performance.

REFERENCES

Abdullah, M., Ali, M., Bakhtan, H., & Mokhtar, S. A. (2017). NUMBER PLATE RECOGNITION OF MALAYSIA VEHICLES USING SMEARING ALGORITHM. *Sci.Int.(Lahore)*, *29*(4), 823–827.

Abdullah, S., Khalid, M., … R. Y.-2006 2nd I., & 2006, undefined. (n.d.). License plate recognition using multi-cluster and multilayer neural networks. *Ieeexplore.Ieee.Org*. Retrieved December 12, 2022, from https://ieeexplore.ieee.org/abstract/document/1684663/

Abdullah, S., Mahedi Hasan, M., & Muhammad Saiful Islam, S. (2018). YOLO-Based Three-Stage Network for Bangla License Plate Recognition in Dhaka Metropolitan City. *2018 International Conference on Bangla Speech and Language Processing, ICBSLP 2018*.

Abdullah, S. N. H. S., Khalid, M., Yusof, R., & Omar, K. (2006). *License Plate Recognition using Multi-cluster and Multilayer Neural Networks*. 1818–1823.

Abdullah, S. N. H. S., Omar, K., Sahran, S., & Khalid, M. (2009). License plate recognition based on support vector machine. *Proceedings of the 2009 International Conference on Electrical Engineering and Informatics, ICEEI 2009*, *1*, 78–82.

Abu, A., Jamaluddin, A. Z., Ghani, M. F., & Hanaffi, M. S. (2019). Development of car plate number recognition using image processing and database system for domestic car park application. *International Journal of Innovative Technology and Exploring Engineering*, *8*(12), 5630–5635.

Al-Ghaili, A. M., Mashohor, S., Ramli, A. R., & Ismail, A. (2013). Vertical-edge-based car-license-plate detection method. *IEEE Transactions on Vehicular Technology*, *62*(1), 26–38.

Anagnostopoulos, C. N. E., Anagnostopoulos, I. E., Loumos, V., & Kayafas, E. (2006). A license plate-recognition algorithm for intelligent transportation system applications. *IEEE Transactions on Intelligent Transportation Systems*, *7*(3), 377–391.

Angeline, L., Lau, H. K., Ghosh, B. K., Goh, H. H., & Kin Teo, K. T. (2012). Development of a License Plate Recognition System for a Non-ideal Environment. *International Journal of Simulation: Systems, Science and Technology*, *13*(3 C), 26–33.

Anson, A., & Mathew, T. (2020). Detection and Recognition of License Plate Using CNN and LSTM. In Jayakumari J., Karagiannidis G., & H. S. Ma M. (Eds.), *Advances in Communication Systems and Networks. Lecture Notes in Electrical Engineering, vol 656* (pp. 701–721). Springer.

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

Busch, C., Doerner, R., Freytag, C., & Ziegler, H. (1998). Feature based recognition of traffic video streams for online route tracing. *IEEE Vehicular Technology Conference*, *3*, 1790–1794.

Chai, H. Y., Woon, H. H., Meng, L. K., & Li, Y. S. (2015). Non-standard Malaysian car license plate recognition. *ISCAIE 2014 - 2014 IEEE Symposium on Computer Applications and Industrial Electronics*, 152–157.

Choong, Y. J., Keong, L. K., & Cheah, T. C. (2020). License plate number detection and recognition using simplified linear-model. In *Journal of Critical Reviews* (Vol. 7, Issue 3, pp. 55–60). Innovare Academics Sciences Pvt. Ltd.

COCO - Common Objects in Context. (n.d.). Retrieved July 4, 2023, from https://cocodataset.org/#home

Dev, A. (n.d.). *A Novel Approach for Car License Plate Detection Based on Vertical Edges* . Retrieved May 13, 2022, from https://ieeexplore-ieee-org.ezlib.iium.edu.my/stamp/stamp.jsp?tp=&arnumber=7433888

Du, S., Ibrahim, M., Shehata, M., & Badawy, W. (2013). Automatic license plate recognition (ALPR): A state-of-the-art review. *IEEE Transactions on Circuits and Systems for Video Technology*, *23*(2), 311–325.

Duan, T., Tran, D., Tran, P., & Hoang, N. (2005). Building an Automatic Vehicle License-Plate Recognition System. In *Proc. Int. Conf. Comput. Sci. RIVF*.

Everingham, M., Luc, ·, Gool, V., Christopher, ·, Williams, K. I., Winn, J., Zisserman, A., Everingham, M., van Gool, L., Leuven, K. U., Williams, B. C. K. I., Winn, J., & Zisserman, A. (n.d.). *The PASCAL Visual Object Classes (VOC) Challenge*. http://www.flickr.com/

Gonçalves, G. R., Silva, S. P. G. da, Menotti, D., & Schwartz, W. R. (2016). Benchmark for license plate character segmentation.

Hao, D., Multimedia, F. X.-J. of, & 2014, undefined. (n.d.). Research on license plate recognition algorithm based on support vector machine. *Search.Ebscohost.Com*. Retrieved December 4, 2022, from https://search.ebscohost.com/login.aspx?direct=true&profile=ehost&scope=site&authtype=crawler&jrnl=17962048&AN=94785198&h=0RwkpP0IC8fZmymEpqXtnKi25JcdBEd7GyrgKq6hCF9BdUUKql6MDA%2BeHxIUG06K4VWXrXIv79TAyLzWzB0Y0g%3D%3D&crl=c

Henry, C., Ahn, S. Y., & Lee, S. W. (2020). Multinational License Plate Recognition Using Generalized Character Sequence Detection. *IEEE Access*, *8*, 35185–35199.

Ho, W. T., Lim, H. W., & Tay, Y. H. (2009). Two-stage license plate detection using gentle adaboost and SIFT-SVM. *Proceedings - 2009 1st Asian Conference on Intelligent Information and Database Systems, ACIIDS 2009*, 109–114.

How, D. N. T., & Sahari, K. S. M. (2017). Character recognition of Malaysian vehicle license plate with deep convolutional neural networks. *IRIS 2016 - 2016 IEEE 4th International Symposium on Robotics and Intelligent Sensors: Empowering Robots with Smart Sensors*, 1–5. https://doi.org/10.1109/IRIS.2016.8066057

Hsu G.-S., Ambikapathi A., Chung S.-L., & Su C.-P. (2017). Robust License Plate Detection In TheWild. *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS).*, 1–6.

Hu, P., Zhao, Y., Yang, Z., & Wang, J. (2002). Recognition of gray character using gabor filters. *Proceedings of the 5th International Conference on Information Fusion, FUSION 2002*, *1*, 419–424.

Huang Z.-K., & Hou L.-Y. (2018). Chinese License Plate Detection Based on Deep Neural Network. *2018 International Conference on Control and Robots, ICCR 2018*, 84–88.

Islam, Md. R. (2007). Malaysian Vehicle License Plate Recognition. *The International Arab Journal of Information Technology*. https://www.academia.edu/36614761/Malaysian_Vehicle_License_Plate_Recognition

Jalil, N. A., Basari, A. S. H., Salam, S., Ibrahim, N. K., & Norasikin, M. A. (2015). The utilization of template matching method for license plate recognition: A case study in malaysia. *Lecture Notes in Electrical Engineering*, *315*, 1081–1090.

Janowski, L., Kozłowski, P., Baran, R., Romaniak, P., Glowacz, A., & Rusc, T. (2014). Quality assessment for a visual and automatic license plate recognition. *Multimedia Tools and Applications*, *68*(1), 23–40.

Jia, W., Zhang, H., He, X., & Wu, Q. (2006). Gaussian weighted histogram intersection for license plate classification. *Proceedings - International Conference on Pattern Recognition*, *3*, 574–577.

Jia, Y., Gonnot, T., & Saniie, J. (2016). Design flow of vehicle License Plate reader based on RGB color extractor. *IEEE International Conference on Electro Information Technology*, *2016-August*, 494–498.

Jocher, G., & et al. (2021). *ultralytics / yolov3*.

Jocher, G., Stoken, A., Chaurasia, A., Borovec, J., NanoCode012, TaoXie, Kwon, Y., Michael, K., Changyu, L., Fang, J., V, A., Laughing, tkianai, yxNONG, Skalski, P., Hogan, A., Nadar, J., imyhxy, Mammana, L., … wanghaoyang0106. (2021). *ultralytics/yolov5: v6.0 - YOLOv5n "Nano" models, Roboflow integration, TensorFlow export, OpenCV DNN support*.

JPJ Malaysia. (n.d.). *Vehicle Number Plate Specifications*. Retrieved October 22, 2021, from https://web.archive.org/web/20130516074518/http://www.jpj.gov.my/web/eng/specifications-of-vehicle-number-plates

Kang, D. J. (2009). Dynamic programming-based method for extraction of license plate numbers of speeding vehicles on the highway. *International Journal of Automotive Technology 2009 10:2*, *10*(2), 205–210.

Kerja Raya, K., Sultan Salahuddin, J., & Lumpur, K. (2019). *MINISTRY OF WORKS MALAYSIA, HIGHWAY PLANNING DIVISION 2 nd Floor*.

Kessentini, Y., Besbes, M. D., Ammar, S., & Chabbouh, A. (2019a). A two-stage deep neural network for multi-norm license plate detection and recognition. *Expert Systems with Applications*, *136*, 159–170.

Kessentini, Y., Besbes, M. D., Ammar, S., & Chabbouh, A. (2019b). A two-stage deep neural network for multi-norm license plate detection and recognition. *Expert Systems with Applications*, *136*, 159–170.

Khaleel, F. M., Sheikh Abdullah, S. N. H., & Ismail, M. K. bin. (2013). License plate detection based on Speeded Up Robust Features and Bag of Words model. *2013 IEEE International Conference on Smart Instrumentation, Measurement and Applications, ICSIMA 2013*.

Koon Cheang, T., Shean Chong, Y., & Haur Tay, Y. (n.d.). *Segmentation-free Vehicle License Plate Recognition using ConvNet-RNN*.

Laroca, R., Cardoso, E. v., Lucio, D. R., Estevam, V., & Menotti, D. (2022). *On the Cross-dataset Generalization in License Plate Recognition*.

Laroca, R., Severo, E., … L. Z.-… joint conference on, & 2018, undefined. (n.d.). A robust real-time automatic license plate recognition based on the YOLO detector. *Ieeexplore.Ieee.Org*. Retrieved December 14, 2022, from https://ieeexplore.ieee.org/abstract/document/8489629/?casa_token=dRLr7_1Uh2cAAAAA:fyaIcZ5RT8ZvHqcCWPLLvobKpUdHSv7AlrixZKc3NVDqGTLqN_tOr_1d7Fx337WKsCowhBRhTutkSxU

Lee, Y. Y., Abdul Halim, Z., & Ab Wahab, M. N. (2022). License Plate Detection Using Convolutional Neural Network-Back to the Basic With Design of Experiments. *IEEE Access*, *10*, 22577–22585.

Li, M., Sun, T., & Liu, H. (2018). Image recognition of steel plate based on an improved support vector machine. *2018 IEEE International Conference on Information and Automation, ICIA 2018*, 1411–1415.

Liang, T. E., Sheikh, U. U., & Mohd, M. N. H. (2020). Malaysian car plate localization using region-based convolutional neural network. *Bulletin of Electrical Engineering and Informatics*, *9*(1), 411–419.

Llorens, D., Marzal, A., Palazón, V., & Vilar, J. M. (2005). Car license plates extraction and recognition based on connected components analysis and HMM decoding. *Lecture Notes in Computer Science*, *3522*(I), 571–578.

Luo, L., Sun, H., Zhou, W., & Luo, L. (2009). An efficient method of license plate location. *2009 1st International Conference on Information Science and Engineering, ICISE 2009*, 770–773.

Marzuki, P., Syafeeza, A. R., Wong, Y. C., Hamid, N. A., Nur Alisa, A., & Ibrahim, M. M. (2019). A design of license plate recognition system using convolutional neural network. *International Journal of Electrical and Computer Engineering*, *9*(3), 2196–2204.

Medialab LPR Database. (n.d.). Retrieved December 14, 2022, from http://www.medialab.ntua.gr/research/LPRdatabase.html

Min, W., Li, X., Wang, Q., Zeng, Q., & Liao, Y. (2019). New approach to vehicle license plate location based on new model YOLO-L and plate pre-identification. *IET Image Processing*, *13*(7), 1041–1049.

Mousa, A. (2012). Canny Edge-Detection Based Vehicle Plate Recognition. In *International Journal of Signal Processing* (Vol. 5, Issue 3).

Ng, H. S., Tay, Y. H., Liang, K. M., Mokayed, H., & Hon, H. W. (2015). Detection and Recognition of Malaysian Special License Plate Based On SIFT Features. *Undefined*.

Omran, S. S., & Jarallah, J. A. (2017). Iraqi License Plate Localization and Recognition System Using Neural Network. *2017 2nd Al-Sadiq International Conference on Multidisciplinary in IT and Communication Science and Applications, AIC-MITCSA 2017*, 73–78.

OpenALPR. (2016). *Openalpr-Eu Dataset.* https://github.com/openalpr/openalpr

Pechiammal, B., & Renjith, J. A. (2017). An efficient approach for automatic license plate recognition system. *ICONSTEM 2017 - Proceedings: 3rd IEEE International Conference on Science Technology, Engineering and Management*, *2018-January*, 121–129.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2016-December*, 779–788.

Redmon, J., & Farhadi, A. (n.d.). *YOLOv3: An Incremental Improvement*. https://pjreddie.com/yolo/.

Redmon, J., & Farhadi, A. (2017). YOLO9000: Better, faster, stronger. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, *2017-January*, 6517–6525.

Road Transport Department Malaysia JPJ. (n.d.). *Plate Number Specification*. Retrieved October 27, 2021, from https://www.jpj.gov.my/en/web/main-site/kenderaan1-en/-/knowledge_base/vehicle/plate-number-registration

Saha, S. (2019). A Review on Automatic License Plate Recognition System. *ArXiv*.

Saini, P., Bidhan, K., & Malhotra, S. (2019). A Detection System for Stolen Vehicles Using Vehicle Attributes with Deep Learning. *Proceedings of IEEE International Conference on Signal Processing,Computing and Control*, *2019-October*, 251–254.

Sanyuan, Z., Mingli, Z., & Xiuzi, Y. (2004a). Car plate character extraction under complicated environment. *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, *5*, 4722–4726.

Sanyuan, Z., Mingli, Z., & Xiuzi, Y. (2004b). Car plate character extraction under complicated environment. *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, *5*, 4722–4726.

Sarfraz, M., Ahmed, M. J., & Ghazi, S. A. (2003a). Saudi Arabian license plate recognition system. *Proceedings - 2003 International Conference on Geometric Modeling and Graphics, GMAG 2003*, 36–41.

Sarfraz, M., Ahmed, M. J., & Ghazi, S. A. (2003b). Saudi Arabian license plate recognition system. *Proceedings - 2003 International Conference on Geometric Modeling and Graphics, GMAG 2003*, 36–41.

Selmi, Z., ben Halima, M., & Alimi, A. M. (2018). Deep Learning System for Automatic License Plate Detection and Recognition. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, *1*, 1132–1138.

Sferle, R. M., & Moisi, E. V. (2019). Automatic number plate recognition for a smart service auto. *2019 15th International Conference on Engineering of Modern Electric Systems, EMES 2019*, 57–60.

Shatnawi, N. M. (2018a). Bees algorithm and support vector machine for Malaysian license plate recognition. In *Int. J. Business Information Systems* (Vol. 28, Issue 3).

Shatnawi, N. M. (2018b). Bees algorithm and support vector machine for Malaysian license plate recognition. In *Int. J. Business Information Systems* (Vol. 28, Issue 3).

Shivakumara, P., Tang, D., Asadzadehkaljahi, M., Lu, T., Pal, U., & Anisi, M. H. (2018). CNN-RNN based method for license plate recognition. *CAAI Transactions on Intelligence Technology*, *3*(3), 169–175.

Silva, S. M., & Jung, C. R. (2017). Real-Time Brazilian License Plate Detection and Recognition Using Deep Convolutional Neural Networks. *Proceedings - 30th Conference on Graphics, Patterns and Images, SIBGRAPI 2017*, 55–62.

Singh J., & Bhushan B. (2019). Real Time Indian License Plate Detection using Deep Neural Networks and Optical Character Recognition using LSTM Tesseract. *Proceedings - 2019 International Conference on Computing, Communication, and Intelligent Systems, ICCCIS 2019*, 347–352.

Souidene Mseddi, W., Sedrine, M. A., & Attia, R. (2021). YOLOv5 Based Visual Localization For Autonomous Vehicles. *European Signal Processing Conference*, *2021-August*, 746–750.

Suandi, S. A., Soon, C. K., Lin, K. C., & Jeng, C. Y. (2012). Malaysian Car Number Plate Detection and Recognition System. *Australian Journal of Basic and Applied Sciences*, *6*(3), 49–59. http://ee.eng.usm.my/eeacad/shahrel/index.html

Tay, Y. H., Tunku, U., Rahman, A., Liang, K. M., & Mokayed, H. (2015). *Detection and Recognition of Malaysian Special License Plate Based On SIFT Features Deep Learning for Forestry View project Intelligent Surveillance Platform (ISP) View project Detection and Recognition of Malaysian Special License Plate Based on SIFT Features*. https://www.researchgate.net/publication/275588094

Tourani, A., Shahbahrami, A., Soroori, S., Khazaee, S., & Suen, C. Y. (2020). A robust deep learning approach for automatic Iranian vehicle license plate detection and recognition for surveillance systems. *IEEE Access*, *8*, 201317–201330.

Tsotsos, J. K., Kotseruba, I., Andreopoulos, A., & Wu, Y. (2019). *Why Does Data-Driven Beat Theory-Driven Computer Vision?* (pp. 0–0).

Wang, F., Man, L., Wang, B., Xiao, Y., Pan, W., & Lu, X. (2008). Fuzzy-based algorithm for color recognition of license plates. *Pattern Recognition Letters*, *29*(7), 1007–1020.

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.

Wang, Y. R., Lin, W. H., & Horng, S. J. (2011). A sliding window technique for efficient license plate localization based on discrete wavelet transform. *Expert Systems with Applications*, *38*(4), 3142–3146.

Weber, M., & Perona, P. (n.d.). *Caltech Cars 1999 (1.0) [Data set]. CaltechDATA*. Retrieved December 14, 2022, from https://data.caltech.edu/records/fmbpr-ezq86

Weng Keong, W., & Iranmanesh, V. (2016). *Malaysian Automatic Number Plate Recognition System using Pearson Correlation*. https://www.researchgate.net/publication/307953265

Wikipedia. (n.d.). *Vehicle registration plates of Malaysia*. Retrieved October 25, 2021, from https://en.wikipedia.org/wiki/Vehicle_registration_plates_of_Malaysia#cite_ref-LP4_2-0

Wu, B. F., Lin, S. P., & Chiu, C. C. (2007). Extracting characters from real vehicle licence plates out-of-doors. *IET Computer Vision*, *1*(1), 2–10.

Wu, Q., Zhang, H., Jia, W., He, X., Yang, J., & Hintz, T. (2006). Car plate detection using cascaded tree-style learner based on hybrid object features. *Proceedings - IEEE International Conference on Video and Signal Based Surveillance 2006, AVSS 2006*, 15–20.

Xie, L., Ahmad, T., Jin, L., Liu, Y., & Zhang, S. (2018). A New CNN-Based Method for Multi-Directional Car License Plate Detection. *IEEE Transactions on Intelligent Transportation Systems*, *19*(2), 507–517.

Xu, Z., Meng, A., Lu, N., Huang, H., Ying, C., & Huang, L. (2018). Towards End-to-End License Plate Detection and Recognition: A Large Dataset and Baseline. *Proceedings of the European Conference on Computer Vision (ECCV)*, *11217*(1), 261–277. https://github.com/detectRecog/CCPD.

Yaacob, N. L., Alkahtani, A. A., Noman, F. M., Zuhdi, A. W. M., & Habeeb, D. (2021). License plate recognition for campus auto-gate system. *Indonesian Journal of Electrical Engineering and Computer Science*, *21*(1), 128–136.

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

Zhou, W., Li, H., Lu, Y., & Tian, Q. (2012). Principal visual word discovery for automatic license plate detection. *IEEE Transactions on Image Processing*, *21*(9), 4269–4279.

Zunino, R., & Rovetta, S. (2000). Vector quantization for license-plate location and image coding. *IEEE Transactions on Industrial Electronics*, *47*(1), 159–167.

APPENDIX

LIST OF PUBLICATIONS

Here is listed all publications and participations out of this thesis:

1) Asaad, A., Firdaus, H., Zaki, M., & Faizabadi, A. R. (2022). MALAYSIAN AUTOMATIC LICENSE PLATE RECOGNITION USING SINGLE-SHOT OBJECT DETECTION MODEL AT LOW VISIBILITY AND UNCONSTRAINED ENVIRONMENT. *PERINTIS EJournal*, *12*(2), 81–94. https://perintis.org.my/ejournalperintis/index.php/PeJ/article/view/130
2) Participated in the 8th International Conference on Mechatronics Engineering (ICOM'22) and was awarded the Best Presentation Award.
3) Participated in Kulliyyah of Engineering Research, Innovation and Commercialization competition (KERICE- 2021) and won the Golden Award.