B.M.S. COLLEGE OF ENGINEERING

(Autonomous Institute, Affiliated to VTU)

Bull Temple Road, Basavanagudi, Bengaluru - 560019

A Capstone Project Report on

"LICENSE PLATE RECOGNITION"

Submitted in partial fulfillment of the requirements for the award of degree

BACHELOR OF ENGINEERING

IN

INFORMATION SCIENCE AND ENGINEERING

By

ADITYA SRIVASTAVA (1BM19IS011)

SHIVEN SWAROOP (1BM19IS148)

SIDDHARTH MANOJ (1BM19IS157)

Under the guidance of

Prof. MAHALAKSHMI B S
Assistant Professor

Department of Information Science and Engineering

2022-2023

B.M.S. COLLEGE OF ENGINEERING

(Autonomous Institute, Affiliated to VTU)
Bull Temple Road, Basavanagudi,
Bengaluru – 560019

Department of Information Science and Engineering

C E R T I F I C A T E

This is to certify that the project entitled "**LICENSE PLATE RECOGNITION**" is a bona-fide work carried out by **Aditya Srivastava (1BM19IS011), Shiven Swaroop (1BM19IS148) Siddharth Manoj (1BM19IS157)** in partial fulfillment for the award of degree of Bachelor of Engineering in **Information Science and Engineering** from **Visvesvaraya Technological University, Belgaum** during the year **2022-2023**. It is certified that all corrections/suggestions indicated for Internal Assessments have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the Bachelor of Engineering Degree.

Guide	HOD	Principal
MAHALAKSHMI B S	**Dr. P Jayarekha**	**Dr. S.Muralidhara**
Assistant Professor	Professor and HOD	Principal

Examiners

Name of the Examiner	**Signature of the Examiner**
1.	
2.	

DECLARATION

ADITYA SRIVASTAVA (1BM19IS011), SHIVEN SWAROOP (1BM19IS148), SIDDHARTH MANOJ (1BM19IS157) students of B.E. Information Science and Engineering, B.M.S. College of Engineering, Bangalore - 19, hereby declare that the capstone project entitled **"LICENSE PLATE RECOGNITION"** is an authentic work carried out under the supervision and guidance of **Prof. MAHALAKSHMI B S,** Assistant Professor, Department of Information Science and Engineering , B.M.S. College of Engineering, Bangalore. We have not submitted the matter embodied to any other university or institution for the award of any other degree.

Place:

Date:

Name	**USN**	**Signature**
Aditya Srivastava	**1BM19IS011**	……………...
Shiven Swaroop	**1BM19IS148**	……………...
Siddharth Manoj	**1BM19IS157**	……………...

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of this Capstone Project Phase-1 would be incomplete without the mention of the people who made it possible through constant guidance and encouragement.

We would take this opportunity to express our heart-felt gratitude to **Dr. B. S. Ragini Narayan**, Chairperson, Donor Trustee, Member Secretary & Chairperson, BMSCE. **Dr. P. Dayananda Pai,** Member Life Trustee, BMSCE and **Dr. S. Muralidhara**, Principal, B.M.S. College of Engineering for providing the necessary infrastructure to complete this Capstone Project Phase-1.

We wish to express our deepest gratitude and thanks to **Dr. Jayarekha P,** Head of the Department, Information Science and Engineering and the Project Coordinator's **Dr. Nalini M K** and **Prof. Harini S** for their constant support.

We wish to express sincere thanks to our guide **Prof. MAHALAKSHMI B S,** Assistant Professor, Department of Information Science and Engineering for helping us throughout and guiding us from time to time.

A warm thanks to all the faculty of the Department of Information Science and Engineering, who have helped us with their views and encouraging ideas.

Aditya Srivastava (1BM19IS011)

Shiven Swaroop (1BM19IS412)

Siddharth Manoj (1BM19IS157)

ABSTRACT

In today's world, the necessity for accurate and reliable vehicle identification and monitoring has grown. The need for a dependable license plate recognition system has increased due to the rise in vehicle populations and the requirement for better traffic management and security measures. As a result, deep learning techniques have been included into the creation of systems that recognise license plates, offering an exact and reliable solution to this issue.

In this project, we create a license plate identification system using deep learning techniques. Tensorflow object detection module is used to recognise license plates in real-time and we use EasyOCR to determine the characters present in the plate with great accuracy and robustness, even in difficult situations.

There are many advantages to using license plate recognition technology. It permits automatic enforcement of traffic laws and regulations in the area of traffic control, eliminating the need for manual supervision. It enables accurate vehicle recognition and fee computation in parking management. The system can also be utilized in security and surveillance applications, aiding in real-time vehicle identification and tracking.

As a result, the license plate recognition system is an essential instrument in today's technologically advanced society, offering advantages in a number of areas like traffic management, parking management, security, and surveillance.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In light of the increasing need for reliable vehicle identification and tracking systems as well as the technology's many practical uses, license plate recognition has received a great deal of attention recently. The necessity for better traffic management, parking management, security, and surveillance as a result of the rise in vehicle populations has made license plate recognition a vital component of contemporary technology.

In this project we use deep learning techniques to develop an automated license plate recognition system using tools such as TensorFlow and EsayOCR. The system uses binary segmentation for the enhancement of the performance of image recognition. In order to obtain high accuracy and robustness in detecting license plate numbers even in difficult circumstances, the model is trained on a large dataset of license plates.

This project offers a thorough examination of the conception and execution of a deep learning-based automated license plate recognition system. The system's effectiveness and the advantages it offers in a variety of disciplines, including traffic control, parking management, security, and surveillance, will be tested using real-world data.

Overall, the license plate recognition system makes a significant contribution to the field of computer vision by offering a remedy for the issue of accurate and efficient vehicle tracking and identification. This system advances the development of license plate recognition systems significantly through the application of cutting-edge deep learning tools and methodologies.

1.1 OVERVIEW

A technology that uses a combination of tools like OpenCV, deep learning frameworks like TensorFlow, skimage, optical character recognition for the number plate recognition of vehicles commuting through an area.

The system can automate the scanning of license plates and tracking of license plate numbers in a variety of settings, including traffic management, parking control, security, and surveillance. For instance, the system can be used to monitor and control traffic in a parking lot, enabling effective and efficient parking management. The device can be utilized in security and surveillance to locate and track questionable vehicles, giving law enforcement and security officers a useful tool.

The system's effectiveness and the advantages it offers in a variety of disciplines, including traffic control, parking management, security, and surveillance, will be tested using real-world data.

Overall, the license plate recognition system makes a significant contribution to the field of computer vision by offering a remedy for the issue of accurate and efficient vehicle tracking and identification. This system advances the development of license plate recognition systems significantly through the application of cutting-edge deep learning tools and methodologies.

1.2 MOTIVATION

The motivation of license plate recognition is mainly driven by the increasing demand for tracking systems and vehicle identification.The ability to quickly and precisely identify license plate numbers has emerged as a critical component of contemporary technology as a result of the development in vehicle populations and the corresponding rise in the demand for better traffic control, parking management, security, and surveillance.

Additionally, license plate recognition offers an answer to many of the difficulties communities experience in controlling and tracking automobiles, including clogged roads, a lack of parking spots, and elevated security concerns. The technology has the ability to increase traffic flow, boost security and surveillance, and provide more effective and efficient parking management by automating the procedure of reading and tracking license plate numbers.

The license plate recognition system makes use of cutting-edge deep learning methods and tools like Convolutional Neural Networks (CNNs) and TensorFlow, which have the potential to transform the field of computer vision and provide fresh approaches to real-world issues. In conclusion, the need for a practical and effective method for vehicle identification and tracking systems has been the driving factor behind the creation of license plate recognition. The technology has the ability to improve security, improve traffic flow, and improve parking management, all of which might have a positive impact on day-to-day life.

1.3 OBJECTIVE

- Automating the reading and tracking of license plate numbers hence increasing accuracy and reducing the manual labor.
- Providing a security and surveillance tool for law enforcement and security personnel to track stolen automobiles and spot fake license plates using license plate recognition technology.
- Optimization of traffic control and providing quick and accurate identification of vehicles.
- Effective and efficient parking enforcement hence involving less manual supervision and allowing space for more vehicles.
- Enhanced information collection and processing that gives important insights into traffic patterns, vehicle demographics, and other important variables.

1.4 SCOPE

The scope of license plate recognition system may include the following:

- Parking management: This system can be utilized to improve parking management systems, reduce the need for manual oversight, and provide room for more vehicles.

- Traffic control: This system can be used to enhance traffic flow, lessen congestion, and increase road safety.
- Security and surveillance: This system can be used by law enforcement and security personnel as a tool to improve security and surveillance procedures.
- Law enforcement: The system makes it possible to track stolen cars and spot fake license plates, enhancing vehicle security and safety.
- Data collection and analysis: License plate recognition can enhance data collecting and analysis efforts by offering insightful information about vehicle populations, traffic patterns, and other important variables.

1.5 EXISTING SYSTEM

The existing system for vehicle identification, license plate recognition and tracking mostly relies on manual methods, like manual entering of data into computer systems or license plate inspections by security officers. These procedures don't allow for real-time tracking, are frequently time-consuming, are prone to human mistake, and are labor-intensive.

Disadvantages of the Manual System:

- Time consuming: The manual methods are often slow and it requires a significant amount of time and resources
- Prone to error: Manual data entry and inspections are prone to human mistake, which can provide inaccurate data and unreliable findings.
- Limited tracking capabilities: The manual system does not offer real-time tracking as it is challenging to keep track of vehicles in real time.
- Inefficient: The manual processes now in use are ineffective as they take a significant amount of resources and time to execute.

These drawbacks of the current system highlight the requirement for a more effective, efficient, and dependable vehicle identification and tracking solution. This technology has advanced significantly, and the planned license plate recognition system has the potential to completely change how vehicles are identified and tracked.

1.6 PROPOSED SYSTEM

The system has the ability to take pictures of license plates, improve the images using image processing techniques, and then extract the license plate numbers from the photos using binary segmentation techniques. After comparison with a database of authorized cars, the retrieved data offers real-time vehicle identification and tracking capabilities.

The suggested system is preferable to the manual methods employed in the current system for vehicle identification and tracking since it is quicker, more effective, and more accurate, and it offers real-time tracking capabilities.

The suggested system has the potential to change the process of vehicle identification and tracking, making it more efficient, effective, and accessible. It marks a substantial improvement in the field of license plate recognition overall.

CHAPTER - 2

PROBLEM STATEMENT

2.1 PROBLEM STATEMENT

The quick and precise identification and tracking of cars is becoming more and more crucial in today's society for a variety of purposes, including security, traffic management, and law enforcement. However, the current approaches to vehicle identification and tracking, such as manual data entry into computer systems or manual examinations of license plates by security staff, are frequently time-consuming, prone to human error, and do not support real-time tracking.

This poses a problem for the development of a more effective, efficient, and trustworthy vehicle identification and tracking system. The demand for such a solution has given rise to license plate recognition technology, which automates the identification and tracking of vehicles using cutting-edge deep learning tools and methodologies.

In order to provide real-time vehicle identification and tracking capabilities, the goal of this project is to develop and implement a license plate recognition system that can capture images of license plates, process them using image refinement and binary segmentation techniques, and compare the extracted information against a database of licensed vehicles. The system should have real-time tracking capabilities and be quicker, more effective, and more accurate than the manual processes utilized in the current system.

2.2 MOTIVATION

Modern society is rapidly expanding and becoming more complicated, which has increased the requirement for reliable and precise vehicle identification and tracking techniques. This need results from the requirement for efficient traffic management, stronger law enforcement initiatives, and adequate security measures.

The current methods for tracking and identifying vehicles, such as manual data entry into computer systems or manual verification of license plates by security staff, are frequently time-consuming, prone to human error, and do not offer real-time tracking capabilities. As a result, there is an increasing need for a vehicle identification and tracking system that is more efficient, effective, and trustworthy.

The advancement of license plate recognition technology is a significant achievement in this area since it automates the identification and tracking of vehicles using state-of-the-art deep learning tools and methodologies. This technology has the potential to dramatically increase the efficiency, speed, and accuracy of vehicle tracking and identification, making it a useful tool for many different applications.

Therefore, the goal of this study is to create and put into use a license plate recognition system that overcomes the shortcomings of the current approaches and offers a more time- and cost-effective, dependable, and efficient technique for vehicle identification and tracking. The system should have real-time tracking capabilities and be quicker, more effective, and more accurate than the manual processes utilized in the current system

2.3 OBJECTIVES

- To develop a license plate recognition system that automates the identification and tracking of vehicles using deep learning tools and techniques.
- Processing photos of license plates and extract pertinent data using image refining and binary segmentation algorithms.
- Develop and implement a database of registered automobiles so that the information taken from the license plates may be compared to locate the vehicles.
- Provide real time tracking capabilities.
- Scalable to accommodate future expansions.
- Ensure the system is secure and reliable.

CHAPTER - 3

DETAILED SURVEY

1. **[2017] License number plate recognition system using entropy-based features selection approach with SVM, Muhammad Attique Khan et al.**

   A vital part of security applications like monitoring traffic, observing street activities, spotting potential threats, etc. is the license plate recognition (LPR) system. Although many techniques have been used for LPR, a cohesive strategy that can deal with difficulties including shifting lighting, occlusion, and many viewpoints is still required. The suggested method addresses these problems by fusing and extracting numerous features.Four steps make up the process: choosing the luminance channel from the CIE Lab color space, binary segmenting the chosen channel, fine-tuning the image, combining geometric and Histogram of Oriented Gradients (HOG) features, choosing the most important features using a novel entropy-based method, and finally classifying the features using a Support Vector Machine (SVM). The accuracy of the suggested approach, which could achieve a maximum of 99.5%, the false positive rate (FPR), and the false negative rate (FNR) were used to assess its performance. The outcomes demonstrate that the suggested methodology outperforms current methods.

2. **[2020] Analyzing passenger and freight vehicle movements from automatic-Number plate recognition camera data, Sheida Hadavi et al.**

   In order to influence policy making, urban transportation planning needs evidence-based insights into transport flows. In study and practice, it is necessary to distinguish between passenger and freight vehicles since they behave differently. Digital data sources make it possible to comprehend urban transportation better and identify areas that call for policy solutions. The literature on digital counting methods like loops, ANPR cameras, and floating automobile data is reviewed in this article. It focuses on the possibilities of ANPR cameras, which are often utilized and can be enhanced with information on different types of vehicles. The process for turning unprocessed ANPR camera data into useful information for city planners is described, with an emphasis on comprehending the movements and stops of passenger and freight vehicles and highlighting distinctions and similarities among different vehicle types. The methodology is illustrated with a case study of the Mechelen-Willebroek neighborhood in Belgium, which made use of enhanced data from 122 ANPR cameras over a two-week period. GPS data from heavy goods vehicles at the same time period is used to confirm the findings. Through the case study, the potential of augmented ANPR camera data and its promising themes and applications are demonstrated.

3. **[2018]Automatic Number Plate Recognition for Motorcyclists Riding Without Helmet, Yogiraj Kulkarni et al.**

The technique described in this work uses CCTV footage to identify motorcycle riders who are not wearing helmets and to retrieve their license plates. The method first extracts moving items from the video's background in order to classify them as either motorcyclists or non-motorcyclists. The head is located and classified for the categorized motorcyclists as either wearing a helmet or not. The system recognises the motorcycle's license plate and extracts the characters from it if it is judged that the rider is not wearing a helmet. Convolutional neural networks are used by the system to classify data, and they are taught via transfer learning on a pre-trained model. This technique yields a high accuracy of 98.72% when identifying motorcycle riders without helmets in traffic recordings.

4. **[2020] Real-time Jordanian license plate recognition using deep learning, Salah Alghyaline**

Because each country has its own requirements, creating an automatic license plate recognition (ALPR) system that works for all sorts of license plates is a difficult undertaking. This study aims to develop a precise ALPR tailored to Jordanian license plates. Convolutional neural networks (CNNs) with two stages that are based on the YOLO3 architecture are used in the suggested method. The YOLO3 network has to be changed to a shallower design in order to improve the detection of small objects due to the license plate characters' small size in comparison to the frame size. The method makes use of temporal information from many frames to get rid of inaccurate predictions, and an array-based data structure to track the licence plates and get rid of fake ones. This is the first complete Jordanian ALPR that can instantly process video streams. This study introduces a new dataset called JALPR, which includes genuine videos of moving automobiles in Jordan and is available online, since there isn't one already for Jordanian license plates. The proposed method outperformed the commercial systems, which had recognition rates of less than 81%, by 87% when it was compared to two well-known commercial software packages.

5. **[2017] Automatic Vehicle License Plate Recognition using Optimal K-Means with Convolutional Neural Network for Intelligent Transportation Systems, Irina Valeryevna Pustokhina et al.**

   The creation of efficient and precise Intelligent Transportation Systems is receiving more attention as a result of rising vehicle sales and improvements in roadway technology (ITS). Computer vision and digital image processing both depend on accurate object detection in images. Because there are differences in viewpoint, shape, color, numerous image formats, and non-uniform illumination when taking pictures, vehicle license plate recognition (VLPR) is a difficult operation. The OKM-CNN model, a deep learning-based VLPR model, is put out in this research. Convolutional Neural Networks (CNN) and optimal K-means clustering-based segmentation are both used in this model. The model runs through three stages: detecting license plates, segmenting using the OKM clustering method, and recognising license plate numbers using the CNN model. For the purpose of locating and detecting license plates, the Connected Component Analysis (CCA) and Improved Bernsen Algorithm (IBA) models are employed. Three datasets—the Stanford Cars dataset, the FZU Cars dataset, and the HumAIn 2019 Challenge dataset—were used to thoroughly assess the proposed approach. The outcomes of these simulations show how the OKM-CNN model performs better than alternative approaches.

6. **[2019] Automatic License Plate Recognition Via Sliding-Window Darknet-Yolo Deep Learning, Hendry et al.**

   In this paper, the authors suggest employing deep learning and the You Only Look Once (YOLO) framework to solve the issue of detecting Taiwan car license plates. The detection method uses a sliding window approach and 7 YOLO convolutional layers to recognise the six digits of a license plate. On the AOLP dataset, where the system was evaluated, it had detection accuracy of 98.22% and recognition accuracy of 78%. The system was also tested in a variety of settings, including dim lighting and varying visual saturations and colors. On a single image, the detection and recognition phase takes between 800 milliseconds and 1 second to complete.

7. **[2017] Automatic License Plate Recognition Technique using Convolutional Neural Network, Surajit Das et al.**

This study's objective is to demonstrate a convolutional neural network-based automated license plate recognition system (CNN). The majority of license plate formats and fonts in India are used on automobiles, with a yellow or white backdrop and black writing. The suggested system is composed of four steps: character segmentation, padding and resizing, character recognition, and image digitization. CNN is utilized for character recognition, and linked component analysis is used for character segmentation. Character recognition is carried out using PYTHON, while image segmentation and scaling are done in MATLAB.The suggested method has been tested on photos of actual cars, and its effectiveness has been assessed using a dataset of 45 images taken in a variety of environments, focusing mostly on cars in West Bengal.

8. **[2021] An efficient robust method for accurate and real-time vehicle plate recognition, Jamshid Pirgazi et al.**

The vehicle license plate identification system encounters a number of difficulties, such as shifting lighting, moving objects, unfavorable camera angles, and adverse weather. The recognition of worn-out or faded plates can also be impacted. Using four steps—vehicle identification, plate detection, character segmentation, and character recognition—this work proposes an effective and reliable approach for reading license plates. Utilizing background emission, the first part of the process involves vehicle detection. In the second phase, plate localisation is carried out utilizing pattern matching and character recognition. Utilizing statistical characteristics, filtering, and morphological operators, the third stage involves segmenting and extracting characters. The final stage involves categorizing each segment using Random Forest into one of 37 classes and ranking characteristics using F-Score. With datasets in English and Persian, the suggested method was tested, and in the Persian dataset, it had accuracy rates of 99.2% for plate recognition, 100% for plate segmentation, and 98.41% for character recognition. For plate detection and character recognition in the English dataset, the accuracy was 100% and 97.5%, respectively. The suggested method is independent of language and performs better than more recent methods in terms of speed and accuracy. The suggested approach can process at least 8 frames per second, making real-time applications possible.

9. **[2021] License Plate Recognition System, Farheen Ali et al.**

The most effective and economical approach for identifying vehicles is to detect a vehicle's license plate. Depending on the image's quality, the location of the car, the illumination, and if it's a single frame, several strategies and methods can be applied. The approach must also be able to handle changes in license plate designs among nations and states, as well as variances in plate widths and character counts that may be visible in photographs that are being collected.
This project focuses on building and creating a software system for reading license plates that can be used for e-challans, parking services, and even vehicle identification. The main goal is to detect and identify several license plates in a single frame. The system is composed of two stages: the identification of the characters and digits and the identification of the license plate number. In the first phase, the license plate number is recognised from the image that was acquired, and in the second phase, the segmented plate is processed to identify the characters and numbers.

10. **[2019] Urban travel time data cleaning and analysis for Automatic Number Plate Recognition, Jie Li et al.**

Automated number plate recognition (ANPR) cameras can offer useful information on traffic patterns, including origin-destination (OD) matrices, journey time statistics, and real-time transit time. In this study, ANPR data gathered in Changsha, China, were assessed. Outliers, which result from drivers diverting from the intended path or making intermediate stops, are frequently present in the derived journey time data. The accuracy of journey time and dependability calculations may be adversely affected by these outliers. The paper presents two techniques for locating and eliminating these outliers. Outliers are found using the Rapid-Moving Window in the first method, while wavelet analysis in the second method offers a more precise means of finding and deleting outliers.The quality of the data for traffic network monitoring and management can be improved by removing outliers from the journey times and then analyzing the link between average travel time and standard deviation.

11. **[2017] Segmentation- and Annotation-Free License Plate Recognition With Deep Localization and Failure Identification, Orhan Bulan et al.**

In many road imaging applications, Automated License Plate Recognition (ALPR) is essential. However, it is challenging for ALPR systems to achieve high recognition accuracy and scalability in the US due to variances in character width, spacing, and the existence of noise sources such as shadows and uneven lighting. Additionally, the amount of manual annotation needed to train classifiers increases due to variations in font and plate layout among countries, adding operational and financial burden. The improved ALPR procedure shown in this paper enhances plate localisation and failure identification while doing away with the requirement for manual annotation. The procedure begins with a two-stage approach for plate localization, followed by a probabilistic inference method based on Hidden Markov Model for segmentation and optical character recognition. The usage of artificial LP pictures or character samples from other running ALPR systems is suggested in order to reduce the amount of manual annotation required to train OCR classifiers. Performance is assessed using realistically obtained LP photos from around the US, and an unsupervised domain adaptation method is employed to reduce the performance gap between the training and target domains.

12. **[2020] A Flexible Approach for Automatic License Plate Recognition in Unconstrained Scenarios, Sergio M. Silva et al.**

Access control and traffic monitoring are two applications of Intelligent Transportation Systems where Automatic License Plate Recognition is essential. Existing methods, however, frequently have restrictions on their applicability, either concentrating on a particular configuration, such as toll control, or an area of license plates, such as European or US plates. In this study, we provide a thorough ALPR system that covers unconstrained capture cases where license plates could be noticeably deformed because of oblique views. The four corners of a license plate can be detected using the Improved Warped Planar Object Detection Network (IWPOD-NET), which can then be warped to a fronto-parallel image to lessen perspective-related distortions.The repaired license plates are then put through two separate object-based optical character recognition techniques. Trials' findings show that even with little training data, the proposed detector can compete with cutting-edge techniques. Our method yields top-scoring outcomes for overall ALPR findings on numerous datasets that cover a variety of capture scenarios and vehicle types, including motorcyclists.

13. **[2018] A Robust Real-Time Automatic License Plate Recognition Based on the YOLO Detector, Rayson Laroca et al.**

Due to its many applications, automatic license plate recognition (ALPR) is a subject of great relevance. Despite this, a lot of current solutions still aren't reliable enough for usage in the real world because they rely on a lot of limitations. Based on the most recent YOLO object detection, the ALPR system shown in this study is reliable and effective. Convolutional neural networks (CNNs) are trained and adjusted for each step of the ALPR process, making them robust to changes in backdrop, lighting, and camera settings. For character segmentation and recognition in particular, a two-stage strategy is used, involving the use of basic data augmentation methods like flipped and inverted license plates. In two datasets, the resulting ALPR system produced remarkable results.2000 frames from 101 vehicle films made up the SSIG dataset, which outperformed prior results of 81.80% and commercial systems like Sighthound and OpenALPR with an identification rate of 93.53% and a frame rate of 47 FPS. Commercial systems' identification rates on the bigger UFPR-ALPR dataset, which was intended to imitate more realistic settings, were below 70%, whereas our system had a recognition rate of 78.33% with a frame rate of 35 FPS.

14. **[2021] An Efficient and Layout-Independent Automatic License Plate Recognition System Based on the YOLO detector, Rayson Laroca et al.**

The goal of this study is to develop an effective Automatic License Plate Recognition (ALPR) system using the YOLO object detection. Utilizing post-processing techniques, the system uses a hybrid approach for license plate detection and layout classification. To achieve the best possible balance between speed and accuracy, the models were optimized. The networks were trained using photos from diverse datasets, along with a variety of data augmentation techniques to increase robustness under different circumstances. Across eight public datasets, the system showed an average end-to-end recognition rate of 96.9%, exceeding prior works and commercial systems in four of the datasets. The outcomes in the remaining datasets were competitive.The system also produced high FPS on a top-tier GPU, making it appropriate for real-time application even when there are numerous vehicles in the area. The authors have provided a service to the scientific community by manually annotating 38,351 bounding boxes on 6,239 photos from open-access datasets.

15. **[2019] Segmentation-Free Vehicle License Plate Recognition Using CNN, Pan Gao et al.**

The deep convolutional neural network (CNN)-based method for locating and identifying car license plates in intricate natural settings is proposed in this research. The suggested method makes use of the 30-class CNN trained as an optimized version of the YOLO target detection method. RDNet, a very effective network model that combines the benefits of DenseNet and ResNet, is used to recognise license plates. Combining these two highly developed networks, the final system can accurately detect and identify license plates without the need for human involvement. The testing findings demonstrate the effectiveness and reliability of the suggested technique, which has a 99.34% recognition accuracy.

CHAPTER - 4

SURVEY SUMMARY TABLE

Sl no.	Title of the paper	Problem Addressed	Authors Approach / Method	Results
1.	License number plate recognition system using entropy-based features selection approach with SVM	LPR gets affected by factors such as light variations, occlusion, and multi-views.	CIE-Lab color space, binary segmentation, Image refinement, Support Vector Machine	Accuracy achieved is maximum up to 99.5%
2.	Analyzing passenger and freight vehicle movements from automatic-Number plate recognition camera data	Provide a better understanding of passenger and freight vehicle movements and stops, identifying similarities and differences between vehicle categories	Inductive loops, Bluetooth detectors, Floating Car Data (FCD), and ANPR cameras	Time restrictions affect the different vehicle categories' actions
3.	Automatic Number Plate Recognition for Motorcyclists Riding Without Helmet	Traffic police manually monitoring motorcyclists at road junctions or through CCTV footage and penalizing those without helmet	Helmet detection, number plate recognition, computer vision, machine learning; convolutional neural networks; transfer learning	Experimental results on traffic videos show an accuracy of 98.72% on detection of motorcyclists without helmets.
4.	Real-time Jordanian license plate	Develop an accurate ALPR for Jordanian Licence Plate	Two-stage CNNs, YOLO3 framework, ALPR	proposed approach achieved 87% recognition

	recognition using deep learning			accuracy,
5.	Automatic Vehicle License Plate Recognition using Optimal K-Means with Convolutional Neural Network for Intelligent Transportation Systems	Automatic Vehicle License Plate Recognition	VLPR model,OKM-CNN model,Improved Bernsen Algorithm, Connected component analysis	ensured the effective performance of the OKM-CNN model over the compared methods in a considerable way.
6.	Automatic License Plate Recognition Via Sliding-Window Darknet-Yolo Deep Learning	To recognize Taiwan’s Licence plate	YOLO darknet deep learning framework	Achieves approximately 98.22% accuracy on license plate detection and 78% accuracy on license plate recognition
7.	Automatic License Plate Recognition Technique using Convolutional Neural Network	Purposes an automated system for recognizing license plate technique using Convolutional Neural Network	Image processing, character segmentation, convolutional neural network, character recognition.	LPR using CNN achieved accuracy of 85.83% tested against 555 images
8.	An efficient robust method for accurate and real-time vehicle plate recognition	Addresses problems like Accuracy and real-timeliness in vehicle plate recognition	AVPR, Plate Detection, CCA, Character recognition, Random Forest	Overall accuracy achieved by the proposed method was 98.27%
9.	License Plate Recognition System	Development of a LPR software system that would be useful in	OpenCV, Machine Learning	Ability to recognize the characters

		vehicle identification as well as in vehicle parking services and even e challan.		correctly on license plate by neural network has probability of 95% even in presence of noise with 50% density
10.	Urban travel time data cleaning and analysis for Automatic Number Plate Recognition	Eliminating outliers in travel times implemented in real time to enhance the data quality for traffic network monitoring and management.	Automated Number Plate Recognition data, Data cleaning,Travel time reliability, Wavelet analysis	Improved quality of data used by AVPR
11.	Segmentation- and Annotation-Free License Plate Recognition With Deep Localization and Failure Identification	For ALPR systems deployed in the United States, variation between jurisdictions on character width, spacing, and the existence of noise sources present in LP images makes it challenging for the recognition accuracy	Plate localization, deep learning, convolutional neural networks, segmentation, annotation, domain adaptation, image quality assessment, character recognition.	Better performance of our proposed methods on LP images captured in several US jurisdictions under realistic conditions.
12.	A Flexible Approach for Automatic License Plate Recognition in Unconstrained Scenarios	Detect the four corners of an LP in a variety of conditions, so that it can be warped to a fronto-parallel view and alleviate perspective-related distortions.	IWPOD-NET, Optical Character Recognition	For some datasets, they achieve 100% accuracy, which is also an indirect clue that the LPD module can correctly locate the LP.
13.	A Robust Real-Time	Training and fine tuning each ALPR stage so that	ALPR system based on the	In the SSIG dataset,

	Automatic License Plate Recognition Based on the YOLO Detector	they are robust under different conditions	YOLO object detection Convolutional Neural Networks (CNNs)	composed of 2,000 frames from 101 vehicle videos, the system achieved a recognition rate of 93.53%
14.	An Efficient and Layout-Independent Automatic License Plate Recognition System Based on the YOLO detector	Aims at achieving the best speed/accuracy trade-off at each stage.	Vehicle Detection, Layout Classification, YOLO	The proposed system achieved an average end-to-end recognition rate of 96.9% across eight public datasets
15.	Segmentation-Free Vehicle License Plate Recognition Using CNN	Using deep convolutional neural networks (CNN) to detect and recognize vehicle license plates in complex natural scenes.	Vehicle plate detection and recognition · Feature extraction Convolutional neural network	recognition accuracy achieves 99.34%.

CHAPTER - 5

SYSTEM REQUIREMENT SPECIFICATION

5.1 FUNCTIONAL REQUIREMENTS

A functional requirement defines the function of a system or its components. a function is described as a specification of behavior between inputs and outputs.

- System should be connected to a CCTV camera to provide traffic footage.
- System should be able to recognise and retrieve the license plate number from the CCTV footage
- System should update all the data in the cloud
- System should have proper error message in case if it not able to identify the number plate

5.2 NON - FUNCTIONAL REQUIREMENTS

The non-functional requirements are divided into availability, usability, reliability, flexibility, and safety.

1. **Scalability**: The system should be able to handle a large number of vehicles and be able to expand as the number of vehicles increases.

2. **Reliability**: The system should have a high uptime and be able to function in presence of noise .

3. **Data Security**: The system should be able to protect the data it collects from

unauthorized access and ensure data privacy.

4. **Latency**: The system should have low latency, and be able to process and transmit data quickly.

5. **Accuracy**: The system should have high accuracy identifying different license plate numbers of the vehicles.

6. **Flexibility**: The system should be flexible and easy to install in various environments and adapt to different license plate designs.

7. **Power Efficiency**: The system should be designed to minimize power Consumption.

8. **Remote monitoring and management**: The system should be able to be monitored and managed remotely, so that it can be easily maintained.

9. **Interoperability**: The system should be able to integrate and work with other systems, such as traffic management systems.

10. **Cost**: The system should be cost-effective and affordable for the end-users

5.3 HARDWARE REQUIREMENTS

PROCESSOR i3: An Intel Corei3 is a proprietary processor based on the multiprocessor architecture framework. It is a kind of dual-core processor that has a GPU integrated into it. It is an Intel processor that replaces the Core 2 series.

RAM 2GB: The short-term memory (RAM) in your computer stores data until the processor needs it. This is not the same as long-term data that is stored on your hard drive and remains there even after your computer is turned off.

HDD 50GB: One type of technology that stores your computer's operating system, applications, and data files like documents, images, and music is known as a hard drive (HDD). Your computer's remaining components collaborate to display the applications and files on your hard drive.

CCTV: a light-proof device with an aperture for a lens and a shutter that projects an object's image onto a surface for recording (like on a photosensitive film or electronic sensor) or for converting into electrical impulses

5.4 SOFTWARE REQUIREMENTS

opencv-python 3.4.2: OpenCV-Python is a library of Python bindings designed to solve computer vision problems

numpy 1.17.2: NumPy is a Python library used for working with arrays.

skimage 0.16.2: Scikit-image, or skimage, is an open source Python package designed for image preprocessing.

tensorflow 1.15.0: TensorFlow is a Python-friendly open source library for numerical computation that makes machine learning and developing neural networks faster and easier

imutils 0.5.3: A series of convenience functions to make basic image processing functions such as translation, rotation, resizing, skeletonization, displaying Matplotlib images, sorting contours, detecting edges, and much more easier

CHAPTER - 6

SYSTEM DESIGN

6.1 SYSTEM DESIGN

An automatic number plate recognition system can be designed using the following components:

● CCTVs that are placed at strategic locations on the road to detect vehicle's number plate.

● A gateway device to collect data from the sensors and send it to the cloud.

● A cloud-based server for storing and analyzing the data.

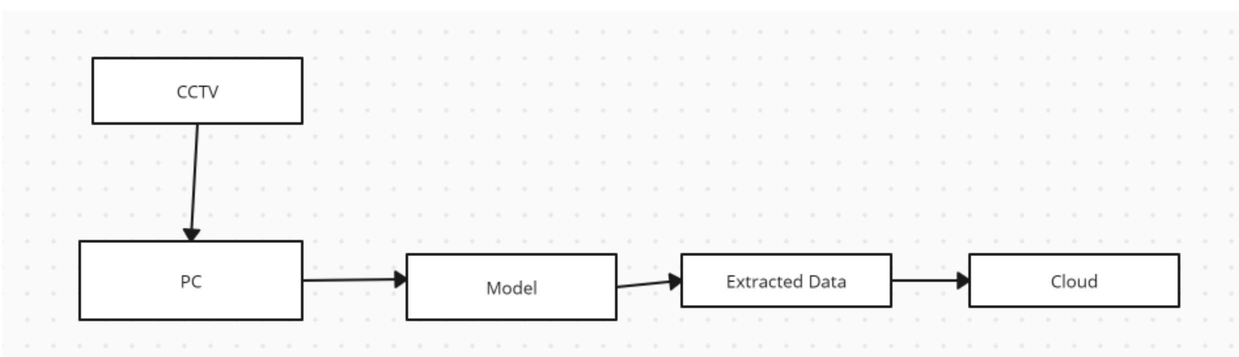
● A model to process the data and accurately determine the number plate.

● A user interface (such as a web or mobile app) to display the number plate data in real-time.

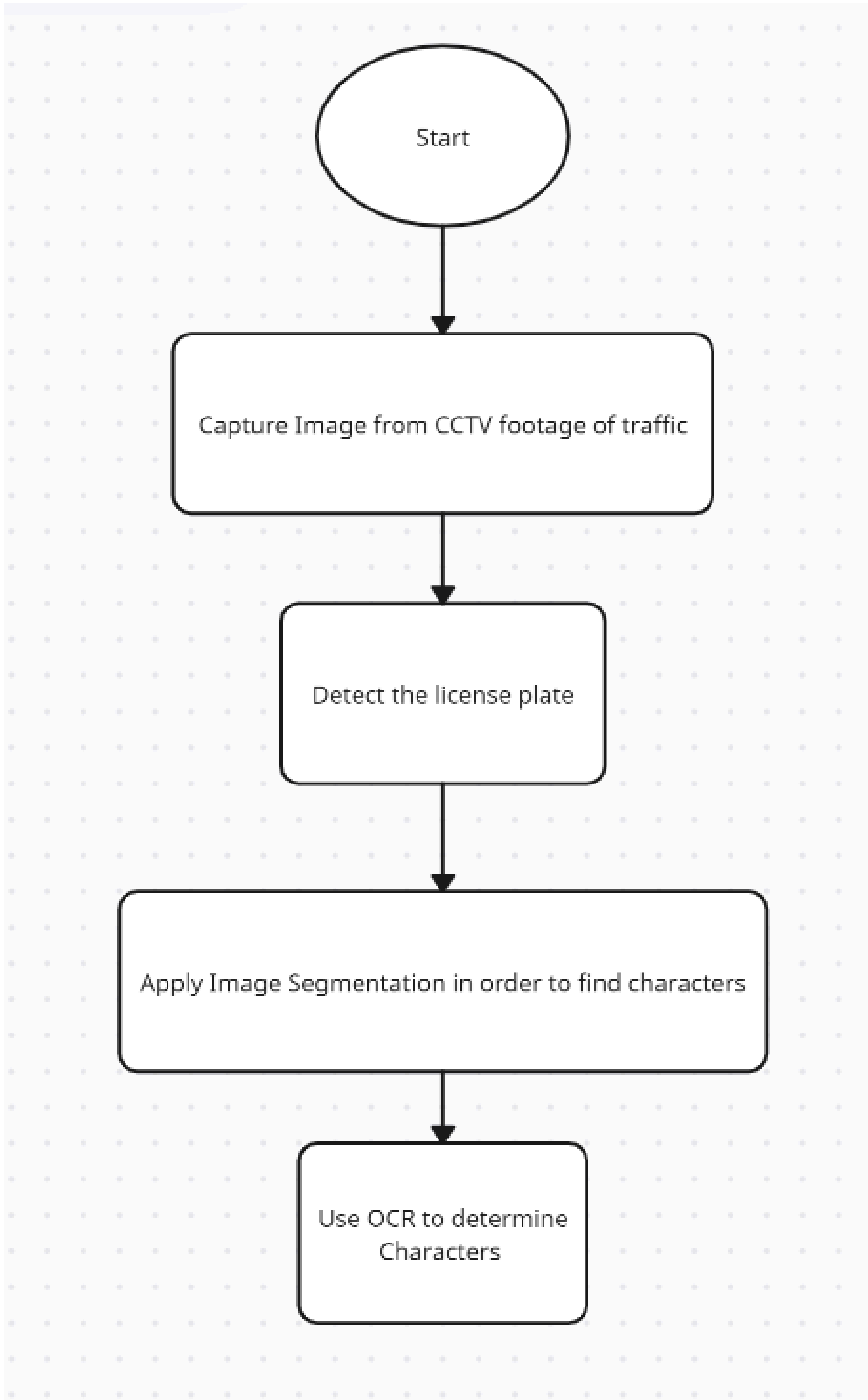
● The system should be designed to handle high volume of data and should be scalable for future growth.

● The security of the system should be designed to protect the data from unauthorized access.

6.1.1 SYSTEM ARCHITECTURE

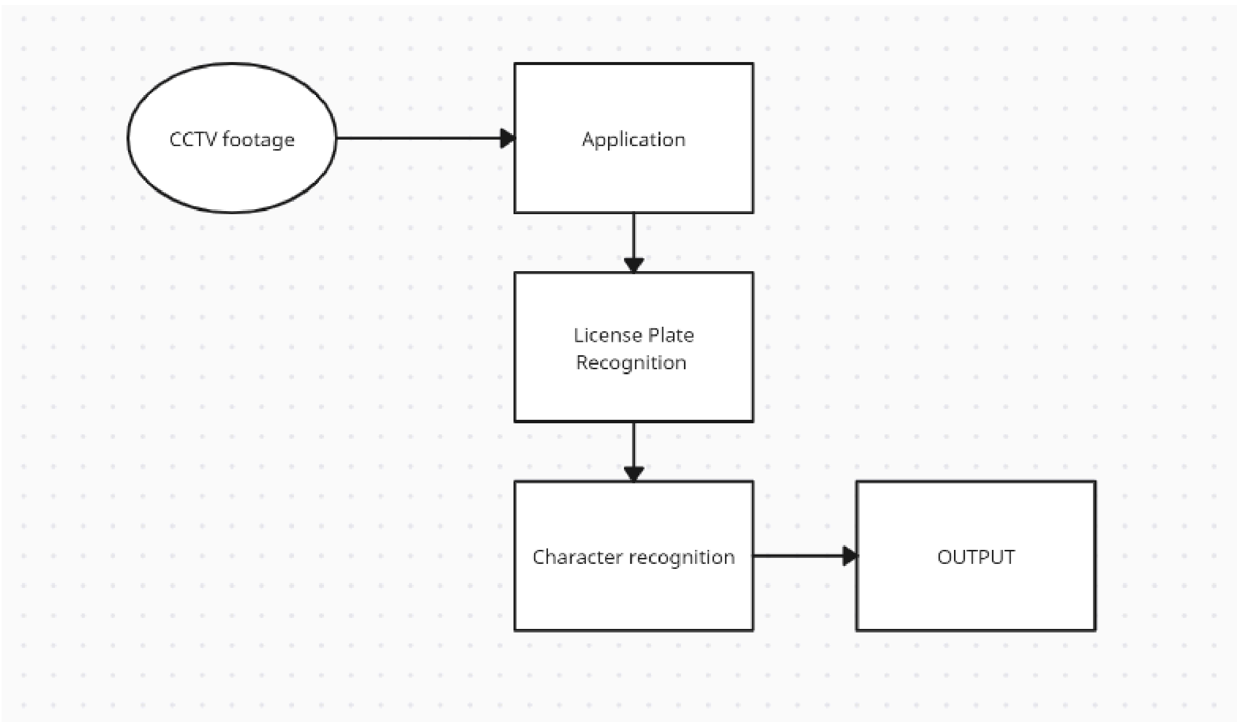


6.1.2 MODULE DESIGN

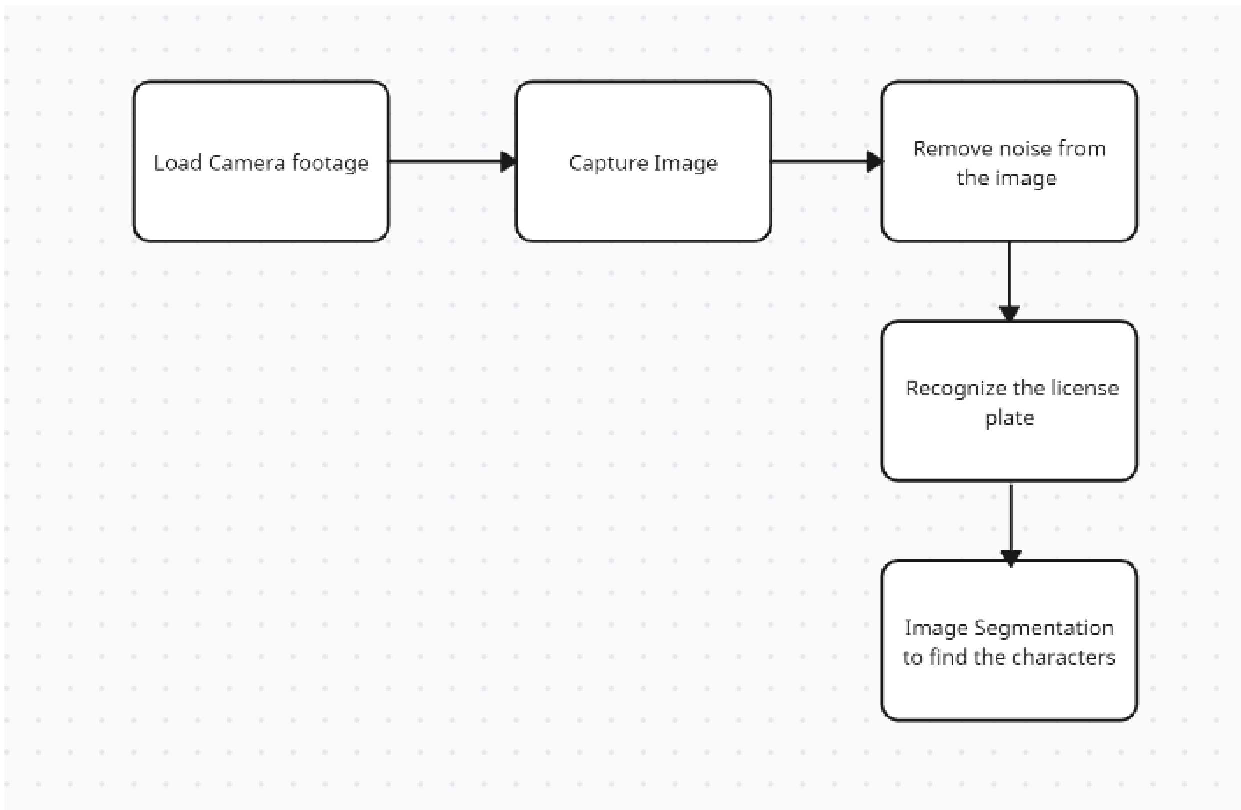


6.2 DETAILED DESIGN

6.2.1 CLASS DIAGRAM

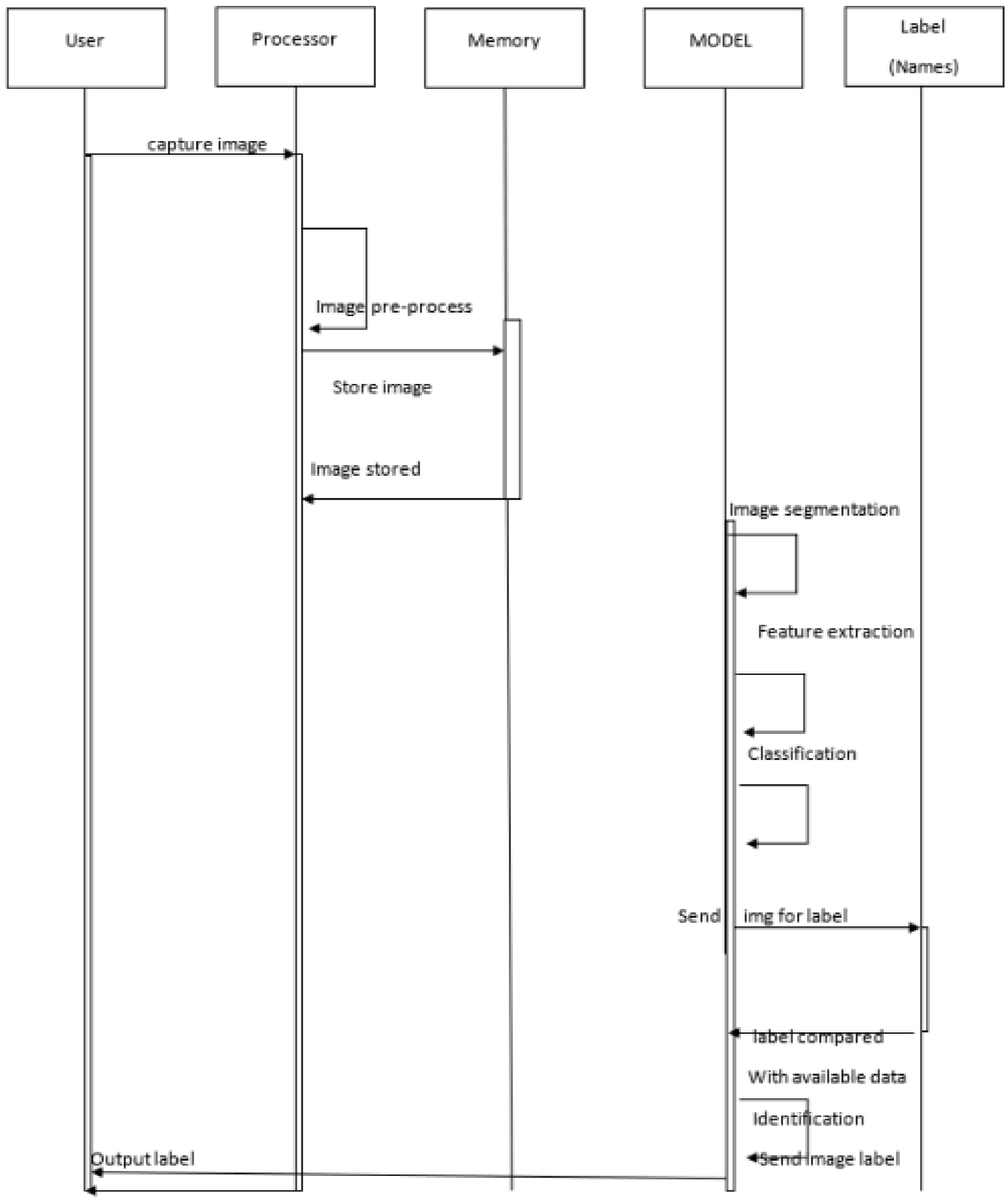


6.2.2 ACTIVITY DIAGRAM



6.2.3 USE CASE DIAGRAM

6.2.4 SCENARIOS



CHAPTER - 7

APPLICATION

There are many uses for automatic license plate recognition (ALPR), including:

- Traffic management: The monitoring and control of traffic flow, the enforcement of parking restrictions, and the detection of stolen vehicles all depend on ALPR systems
- Police use ALPR to find stolen cars and criminal suspects in the course of their work.
- Management of toll roads: ALPR systems are used for electronic toll collection to automatically deduct toll fees from the registered owner's account.
- Security: To track and identify vehicles entering and leaving secure locations like airports, military sites, and gated communities, ALPR is employed.
- Parking management: By identifying parked vehicles and enforcing parking regulations, ALPR systems automate parking enforcement and management.
- Border control: To automatically identify and track vehicles entering and leaving a country, ALPR systems are used at border crossings and customs checkpoints.

CHAPTER - 8

IMPLEMENTATION

Tensorflow Object Detection

We can carry out object detection tasks using pre-trained models from the TensorFlow Model Zoo thanks to the robust framework of TensorFlow Object Detection. Object detection entails locating and identifying items inside a frame of an image or video. TensorFlow offers a selection of pre-trained models that have been applied to a variety of object identification tasks and have been trained on sizable datasets.

A variety of pre-trained models for various computer vision applications, such as object detection, are available in the TensorFlow Model Zoo repository. These models are developed using sizable datasets as Pascal VOC (Visual Object Classes) and COCO (Common Objects in Context). They are made to find and categorize various objects, including people, vehicles, animals, and commonplace items.

We can use these pre-trained models for our own object identification tasks by utilizing the TensorFlow Object identification API. The API offers a practical method for developing, testing, and deploying object detection models. It offers a flexible architecture that enables us to alter the backbone network, feature extractors, and post-processing procedures, among other elements of the detection pipeline.

We can significantly reduce the amount of time and computational resources needed for training by utilizing the pre-trained models from the TensorFlow Model Zoo. These models have learned to recognise a wide range of items accurately after being trained on extensive datasets. We can quickly and efficiently deploy object detection solutions for a variety of applications, including object tracking, surveillance, autonomous cars, and more with TensorFlow Object Detection and the Model Zoo.

EasyOCR

EasyOCR is a well-known optical character recognition (OCR) open-source library. With the aid of OCR technology, text may be recognised and extracted from photographs or scanned documents. Due to its simplicity and user-friendliness, EasyOCR has grown in popularity as a tool for performing OCR jobs.

Here are some important characteristics and facets of EasyOCR:

- EasyOCR supports many different languages, including well-known ones like English, Spanish, French, German, Chinese, Japanese, and Korean. This qualifies it for use in multilingual and international applications.

- Deep Learning-based Approach: EasyOCR uses convolutional neural networks (CNNs), a deep learning approach, to produce reliable and accurate text recognition. The underlying models can manage different font sizes, styles, and orientations because they were trained on vast datasets.

- Simple API: EasyOCR offers a clear API that makes incorporating OCR functionality into your own applications or workflows a breeze. The API accepts images as input, and it returns the recognised text and accompanying confidence scores along with it.

- EasyOCR is implemented in Python and supports a variety of operating systems, including Windows, macOS, and Linux. The library is simple to set up and use and can be quickly installed using pip. It also has few dependencies.

- To increase the accuracy of OCR, EasyOCR provides a number of options for preprocessing and postprocessing photos. This covers methods like text correction, denoising, binarization, and image resizing.

- EasyOCR lets you process numerous photos at once, which is useful if you need to run OCR on a lot of documents or images.

- EasyOCR includes pre-trained models that function effectively right out of the box, but it also gives you the option to make any necessary adjustments or train new models on your unique datasets. When handling specialized or domain-specific OCR tasks, this can be useful.

Due to its ease of use, accuracy, and compatibility for several languages, EasyOCR has become increasingly popular among programmers and researchers. It has many different uses, such as document digitization, text extraction from photos, automatic data entry, text recognition in images for computer vision tasks, and more.

CODE

 # Creating Label Map

```
labels = [{'name': 'licence', 'id': 1}]
label_map_path = files['LABELMAP']

with open(label_map_path, 'w') as f:
    for label in labels:
        f.write('item {\n')
        f.write('\tname: \'{}\'\n'.format(label['name']))
        f.write('\tid: {}\n'.format(label['id']))
        f.write('}\n')
```

 # Updating Configuration for Transfer Learning

```
config = config_util.get_configs_from_pipeline_file(files['PIPELINE_CONFIG'])

pipeline_config = pipeline_pb2.TrainEvalPipelineConfig()
with tf.io.gfile.GFile(files['PIPELINE_CONFIG'], "r") as f:
    proto_str = f.read()
    text_format.Merge(proto_str, pipeline_config)

pipeline_config.model.ssd.num_classes = len(labels)
pipeline_config.train_config.batch_size = 4
pipeline_config.train_config.fine_tune_checkpoint = os.path.join(paths['PRETRAINED_MODEL_PATH'], PRETRAINED_MODEL_NAME, 'checkpoint', 'ckpt-0')
pipeline_config.train_config.fine_tune_checkpoint_type = "detection"
pipeline_config.train_input_reader.label_map_path = label_map_path
```

```
pipeline_config.train_input_reader.tf_record_input_reader.input_path[:] =
[os.path.join(paths['ANNOTATION_PATH'], 'train.record')]
pipeline_config.eval_input_reader[0].label_map_path = label_map_path
pipeline_config.eval_input_reader[0].tf_record_input_reader.input_path[:] =
[os.path.join(paths['ANNOTATION_PATH'], 'test.record')]

config_text = text_format.MessageToString(pipeline_config)
with tf.io.gfile.GFile(files['PIPELINE_CONFIG'], "wb") as f:
    f.write(config_text)
```

 # Detection from Image

```
img = cv2.imread(IMAGE_PATH)
image_np = np.array(img)

input_tensor = tf.convert_to_tensor(np.expand_dims(image_np, 0), dtype=tf.float32)
detect = detect_fn(input_tensor)

num_detect = int(detect.pop('num_detect'))
detect = {key: value[0, :num_detect].numpy() for key, value in detect.items()}
detect['num_detect'] = num_detect

# detection_classes should be ints.
detect['detection_classes'] = detect['detection_classes'].astype(np.int64)

label_id_offset = 1
image_np_with_detect = image_np.copy()

viz_utils.visualize_boxes_and_labels_on_image_array(
```

```
    image_np_with_detect,
    detect['detection_boxes'],
    detect['detection_classes'] + label_id_offset,
    detect['detection_scores'],
    category_index,
    use_normalized_coordinates=True,
    max_boxes_to_draw=5,
    min_score_thresh=.8,
    agnostic_mode=False
)

plt.imshow(cv2.cvtColor(image_np_with_detect, cv2.COLOR_BGR2RGB))
plt.show()
```

 # Applying OCR for Detection

```
image = image_np_with_detect
detection_threshold = 0.5  # Set your desired threshold
scores = list(filter(lambda x: x > detection_threshold, detect['detection_scores']))
boxes = detect['detection_boxes'][:len(scores)]
classes = detect['detection_classes'][:len(scores)]

width = image.shape[1]
height = image.shape[0]

# Apply ROI filtering and OCR
for idx, box in enumerate(boxes):
    print(box)
    roi = box * [height, width, height, width]
    print(roi)
```

```
    region = image[int(roi[0]):int(roi[2]), int(roi[1]):int(roi[3])]
    reader = easyocr.Reader(['en'])
    ocr_result = reader.readtext(region)
    print(ocr_result)
    plt.imshow(cv2.cvtColor(region, cv2.COLOR_BGR2RGB))
```

 # Real-time Detection

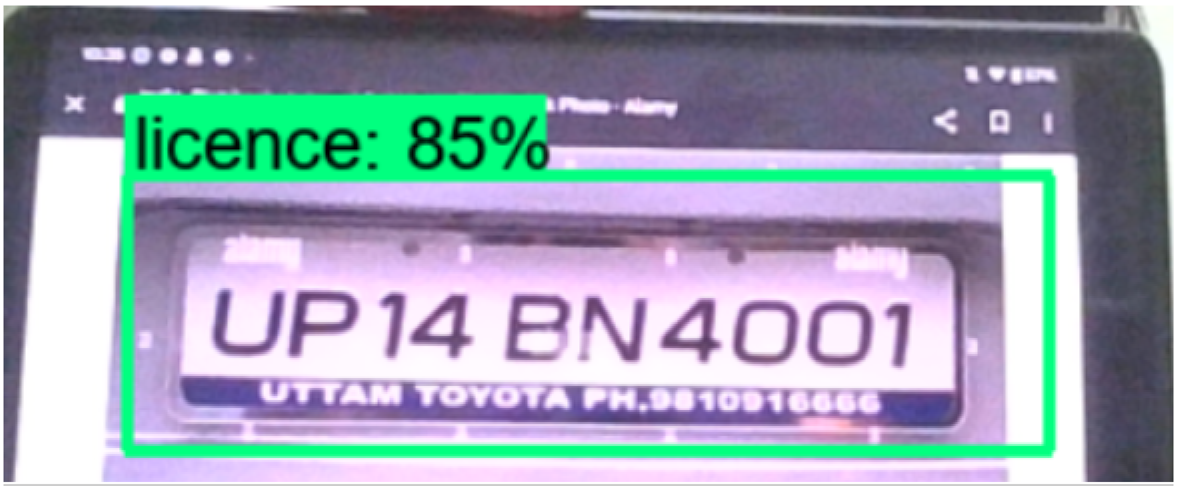
```
cap = cv2.VideoCapture(0)
width = int(cap.get(cv2.CAP_PROP_FRAME_WIDTH))
height = int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))

while cap.isOpened():
    ret, frame = cap.read()
    image_np = np.array(frame)

    input_tensor = tf.convert_to_tensor(np.expand_dims(image_np, 0), dtype=tf.float32)
    detect = detect_fn(input_tensor)

    num_detect = int(detect.pop('num_detect'))
    detect = {key: value[0, :num_detect].numpy() for key, value in detect.items()}
    detect['num_detect'] = num_detect

    # detection_classes should be ints.
    detect['detection_classes'] = detect['detection_classes'].astype(np.int64)

    label_id_offset = 1
    image_np_with_detect = image_np.copy()
```

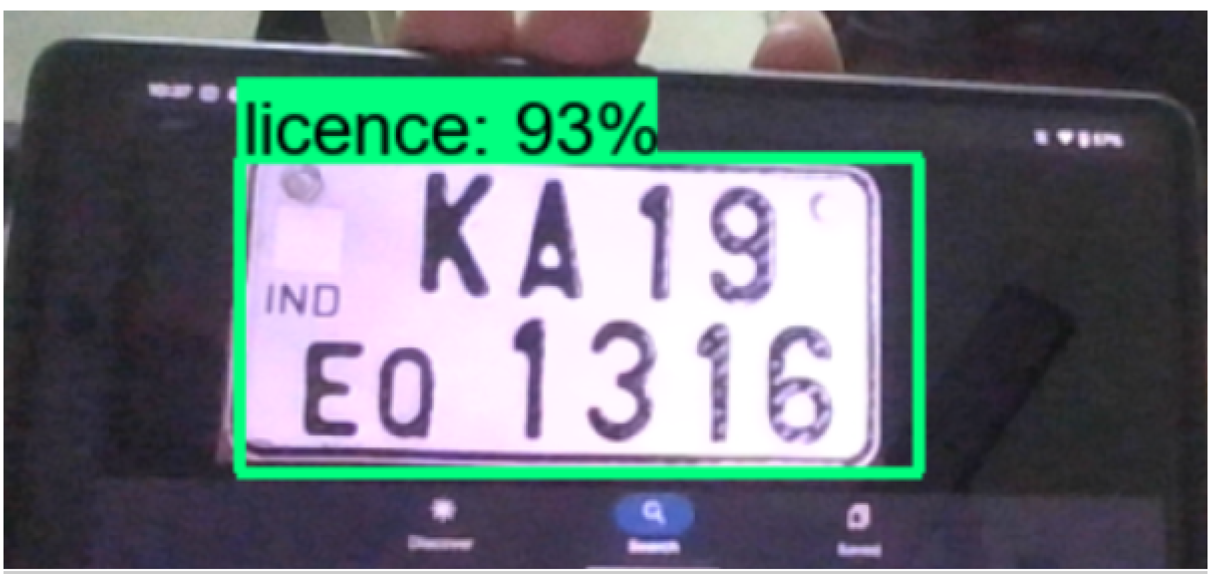
```
viz_utils.visualize_boxes_and_labels_on_image_array(
    image_np_with_detect,
    detect['detection_boxes'],
    detect['detection_classes'] + label_id_offset,
    detect['detection_scores'],
    category_index,
    use_normalized_coordinates=True,
    max_boxes_to_draw=5,
    min_score_thresh=.8,
    agnostic_mode=False
)



cv2.imshow('object detection', cv2.resize(image_np_with_detect, (800, 600)))

if cv2.waitKey(10) & 0xFF == ord('q'):
    cap.release()
    cv2.destroyAllWindows()
    break
```

CHAPTER - 9

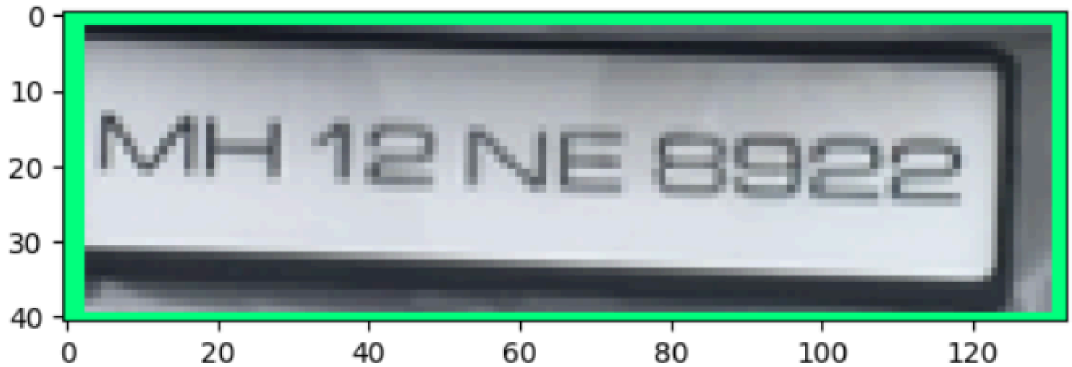
RESULT(SNAPSHOT)

Real-time Plate Detection





Detection with results




```
['MH12NE 8922']
```

CHAPTER - 10

TESTING

Software testing is the process of analyzing a software application or system to gauge its performance, functionality, and quality. To find flaws, faults, or variations from the planned behavior, it entails running the software with multiple inputs, analyzing the output, and comparing it against the expected outcomes. Software testing's main objective is to make sure that the programme operates as intended, complies with all specifications, and produces the desired results.

Software testing includes a variety of tasks, such as:

1. Verification: Testing is done to make sure the software complies with the requirements and follows the development and design standards. It guarantees that the software has been constructed properly.

2. Validation: Testing confirms that the programme satisfies the user's requirements and expectations. It guarantees that the software has been properly coded and accomplishes its intended function.

3. Defect Detection: Testing aids in finding software flaws, errors, or defects. It entails running test cases, comparing the outcomes to what was anticipated, and looking for any differences.

4. Quality Assurance: A key component of quality assurance initiatives is testing. It aids in guaranteeing the overall effectiveness, dependability, and quality of the software.

5. Functional testing: This kind of testing is concerned with making sure the software is functional and that it carries out the required tasks as intended.

6. Non-functional Testing: Non-functional testing looks at things like scalability, compatibility, performance, security, usability, and reliability. It makes sure that the programme complies with the desired non-functional requirements.

7. Test Planning: Test planning entails defining the test's goals, parameters, and plan. It involves establishing test scenarios, creating test cases, and figuring out the testing deadlines and resources that will be required.

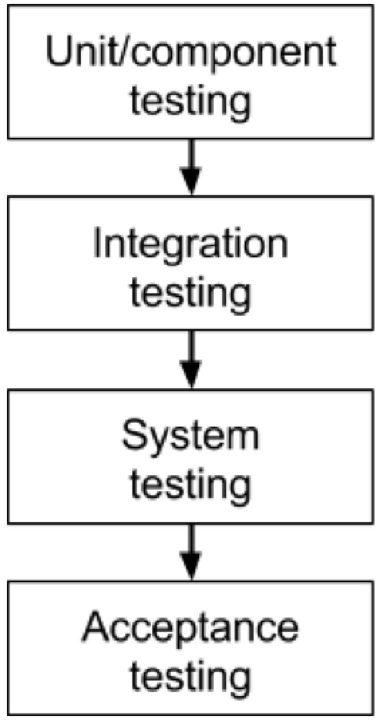
8. Test Execution: Testing is the process of running test cases against software and recording the outcomes. It also entails documenting any flaws or problems discovered during testing.

9. Test Reporting: Test reports contain information on how tests were carried out, such as the test cases that were run, the results, any faults that were found, and how serious they were. These reports aid in decision-making and offer perceptions into the caliber of the software.

10. Regression Testing: In order to make sure that updates or patches don't create new flaws or affect current functionality, regression testing involves retesting previously tested functionalities.

11. Test Automation: Test automation is the process of automating the execution of test cases using specialized tools or scripts. It enhances the effectiveness, consistency, and coverage of testing efforts.

The software development life cycle (SDLC) often includes the iterative process of software testing. Early fault detection and correction lowers the cost and effort needed to address problems at a later stage of development or during production. The reliability, success, and general quality of the programme are all influenced by effective testing.



Components of testing

Each testing type in software development has a particular function and focuses on a different area of the software. Four popular forms of testing are listed below:

1. Unit testing: Isolated testing of the software's constituent modules or components is known as unit testing. It emphasizes on ensuring that the tiniest testable components, such functions, methods, or classes, operate as intended. To make sure that each unit functions as expected, developers routinely write and run unit tests. They aid in early fault detection and correction, enhancing the quality of the code and making integration simpler.

2. Integration Testing: Integration testing is carried out to confirm how well various software modules or components interact with one another. It guarantees proper integration of the merged units. Depending on the complexity of the software, integration testing can be done at several levels, such as module, subsystem, or system levels. The goal is to locate any problems resulting

from component interactions, such as data transfer faults, interface flaws, or improper behavior brought on by integration.

3. System testing: System testing entails checking the functionality of the complete software system. It confirms that the integrated system complies with the requirements and operates well in the environment for which it was designed. In system testing, the functionality, performance, dependability, security, and overall behavior of the system are evaluated. To make sure the system performs as planned, it may comprise both functional and non-functional testing that covers a range of use cases and situations.

4. Acceptance Testing: Acceptance testing is done to see if the programme satisfies the needs and acceptance standards of the user. End users, clients, or other stakeholders frequently carry it out to confirm that the programme meets their requirements and expectations. User acceptability testing (UAT), in which users act out real-world scenarios, and business acceptance testing (BAT), in which the software is assessed in relation to corporate objectives, are two examples of different types of acceptance testing. Making sure the programme is suitable for deployment and serves the intended function is the main concern.

These are only a few of the several testing methods that are available in software development. Regression testing, performance testing, security testing, usability testing, compatibility testing, and other sorts are also available. The requirements, objectives, and particular areas of concern of the project will determine the best testing methods to use. Combining several testing strategies improves the software's overall quality and dependability and helps to assure thorough test coverage.

TEST CASES

Test Case	1
Name of Test	Detection of License Plate
Input	Image
Expected Output	License Plate Detection With Accuracy Percentage
Actual Output	Box detecting License Plate
Result	Successful

Test Case	2
Name of Test	Detection of License Plate in Real-time
Input	Video
Expected Output	License Plate Detection With Accuracy Percentage
Actual Output	Box detecting License Plate
Result	Successful

Test Case	3
Name of Test	Character Determination from Image
Input	Image
Expected Output	Characters Present in License Plate
Actual Output	Character set
Result	Successful

Test Case	4
Name of Test	Character Determination from Real-time Video
Input	Video
Expected Output	Characters Present in License Plate
Actual Output	Character set
Result	Successful

CONCLUSION

The way we manage and monitor vehicles has been completely transformed by the highly developed technology known as automatic license plate recognition (ALPR). It is a useful tool for a variety of applications, including traffic management, law enforcement, toll road management, security, parking management, and border control. It offers a quick, accurate, and efficient way to identify and track vehicles. Despite some privacy concerns, ALPR is still widely used and its adoption is anticipated to increase in the future as technology develops and its advantages become more and more clear.

BIBLIOGRAPHY

[1] Khan, M.A., Sharif, M., Javed, M.Y., Akram, T., Yasmin, M. and Saba, T., 2018. License number plate recognition system using entropy‐based features selection approach with SVM. *IET Image Processing*, *12*(2), pp.200-209.

[2] Hadavi, S., Rai, H.B., Verlinde, S., Huang, H., Macharis, C. and Guns, T., 2020. Analyzing passenger and freight vehicle movements from automatic-Number plate recognition camera data. *European Transport Research Review*, *12*(1), pp.1-17.

[3] Kulkarni, Y., Bodkhe, S., Kamthe, A. and Patil, A., 2018, March. Automatic number plate recognition for motorcyclists riding without helmet. In *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)* (pp. 1-6). IEEE.

[4] Alghyaline, S., 2022. Real-time Jordanian license plate recognition using deep learning. *Journal of King Saud University-Computer and Information Sciences*, *34*(6), pp.2601-2609.

[5] Pustokhina, I.V., Pustokhin, D.A., Rodrigues, J.J., Gupta, D., Khanna, A., Shankar, K., Seo, C. and Joshi, G.P., 2020. Automatic vehicle license plate recognition using optimal K-means with convolutional neural network for intelligent transportation systems. *Ieee Access*, *8*, pp.92907-92917.

[6] Chen, R.C., 2019. Automatic License Plate Recognition via sliding-window darknet-YOLO deep learning. *Image and Vision Computing*, *87*, pp.47-56.

[7] Pustokhina, I.V., Pustokhin, D.A., Rodrigues, J.J., Gupta, D., Khanna, A., Shankar, K., Seo, C. and Joshi, G.P., 2020. Automatic vehicle license plate recognition using optimal K-means with convolutional neural network for intelligent transportation systems. *Ieee Access*, *8*, pp.92907-92917.

[8] Laroca, R., Zanlorensi, L.A., Gonçalves, G.R., Todt, E., Schwartz, W.R. and Menotti, D., 2021. An efficient and layout‐independent automatic license plate recognition system based on the YOLO detector. *IET Intelligent Transport Systems*, *15*(4), pp.483-503.

[9] Ali, F., Rathor, H. and Akram, W., 2021, March. License plate recognition system. In *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)* (pp. 1053-1055). IEEE.

[10] Li, J., Van Zuylen, H., Deng, Y. and Zhou, Y., 2020. Urban travel time data cleaning and analysis for Automatic Number Plate Recognition. *Transportation Research Procedia*, *47*, pp.712-719.

[11] Bulan, O., Kozitsky, V., Ramesh, P. and Shreve, M., 2017. Segmentation-and annotation-free license plate recognition with deep localization and failure identification. *IEEE Transactions on Intelligent Transportation Systems*, *18*(9), pp.2351-2363.

[12]Silva, S.M. and Jung, C.R., 2021. A flexible approach for automatic license plate recognition in unconstrained scenarios. *IEEE Transactions on Intelligent Transportation Systems*, *23*(6), pp.5693-5703.

[13] Laroca, R., Severo, E., Zanlorensi, L.A., Oliveira, L.S., Gonçalves, G.R., Schwartz, W.R. and Menotti, D., 2018, July. A robust real-time automatic license plate recognition based on the YOLO detector. In *2018 international joint conference on neural networks (ijcnn)* (pp. 1-10). IEEE.

[14] Laroca, R., Zanlorensi, L.A., Gonçalves, G.R., Todt, E., Schwartz, W.R. and Menotti, D., 2021. An efficient and layout‐independent automatic license plate recognition system based on the YOLO detector. *IET Intelligent Transport Systems*, *15*(4), pp.483-503.

[15] Gao, P., Zeng, Z. and Sun, S., 2019. Segmentation-free vehicle license plate recognition using CNN. In *Signal and Information Processing, Networking and Computers: Proceedings of the 4th International Conference on Signal and Information Processing, Networking and Computers (ICSINC) 4th* (pp. 50-57). Springer Singapore.

DrillBit

DrillBit Similarity Report

24	90	B	A-Satisfactory (0-10%) B-Upgrade (11-40%) C-Poor (41-60%) D-Unacceptable (61-100%)
SIMILARITY %	MATCHED SOURCES	GRADE	

LOCATION	MATCHED DOMAIN	%	SOURCE TYPE
1	arxiv.org	2	Publication
2	Automated License Plate Recognition A Survey on Methods and Techniques by Shashirangana-2020	1	Publication
3	www.researchgate.net	1	Internet Data
4	Automated License Plate Recognition A Survey on Methods and Techniques by Shashirangana-2020	1	Publication
5	classyboutiquebymizzha.com	1	Internet Data
6	worldwidescience.org	1	Internet Data
7	ijircce.com	1	Publication
8	Real-time Jordanian license plate recognition using deep learning by Alghyaline-2020	1	Publication
9	www.academia.edu	1	Internet Data
10	mdpi.com	1	Internet Data
11	moam.info	1	Internet Data
12	www.ijeat.org	<1	Publication
13	docplayer.net	<1	Internet Data

14	solidrockglasgow.com	<1	Internet Data
15	An Algorithm Based on Text Position Correction and Encoder-Decoder Network for T by Huang-2020	<1	Publication
16	abnews-wire.blogspot.com	<1	Internet Data
17	goggleboy.nl	<1	Internet Data
18	High-speed rail pole number recognition through deep representation and temporal by Yang-2020	<1	Publication
19	www.dx.doi.org	<1	Publication
20	www.atlantis-press.com	<1	Publication
21	Submitted to Visvesvaraya Technological University, Belagavi	<1	Student Paper
22	www.dx.doi.org	<1	Publication
23	www.meninavaidosacosmeticos.com.br	<1	Internet Data
24	asbmr.onlinelibrary.wiley.com	<1	Internet Data
25	www.dx.doi.org	<1	Publication
26	www.mdpi.com	<1	Internet Data
27	mdpi.comjournalacoustics	<1	Internet Data
28	coek.info	<1	Internet Data
29	journals.ku.edu	<1	Internet Data
30	moam.info	<1	Internet Data
31	Real-time Jordanian license plate recognition using deep learning by Alghyaline-2020	<1	Publication

32	Sparks fly over origin of altruism by Marshall-2010	<1	Publication
33	www.readbag.com	<1	Internet Data
34	www.researchgate.net	<1	Internet Data
35	cock.info	<1	Internet Data
36	IEEE 2018 22nd International Computer Science and Engineering Confer	<1	Publication
37	IEEE 2020 International Conference on Smart Electronics and Communication (ICO	<1	Publication
38	www.mdpi.com	<1	Internet Data
39	moam.info	<1	Internet Data
40	Performance Evaluation of UHF RFID Technologies for Real-Time Bus Recognition in by Own-2013	<1	Publication
41	Thesis Submitted to Shodhganga Repository	<1	Publication
42	IEEE 2012 20th Iranian Conference on Electrical Engineering (ICEE) - by	<1	Publication
43	A Cascade Framework for a Real-Time Statistical Plate Recognition System by Wang-2007	<1	Publication
44	Direct Single Frequency Synthesis from a Prescribed Scattering Matrix	<1	Student Paper
45	dochero.tips	<1	Internet Data
46	emcw.org	<1	Internet Data
47	Thesis submitted to shodhganga - shodhganga.inflibnet.ac.in	<1	Publication
48	IEEE 2017 IEEE International Conference on Systems, Man and Cyberne, by Lee, Sung Min Youn- 2017	<1	Publication

49	An 8times8 Cell Analog Order-Statistic-Filter Array With Asynchronous Grayscа by Poikonen-2009	<1	Publication
50	itercriminis.com	<1	Internet Data
51	qdoc.tips	<1	Internet Data
52	sample.belltreeoils.com	<1	Internet Data
53	cameonetwork.org	<1	Internet Data
54	Creep of Dam Concrete Evaluated from Laboratory and In Situ Tests by C-2012	<1	Publication
55	Distribution of messenger ribonucleic acid in polysomes and nonpolysomal particl by Infante-1981	<1	Publication
56	Science Di Rect	<1	Publication
57	wiredspace.wits.ac.za	<1	Publication
58	www.freepatentsonline.com	<1	Internet Data
59	IEEE 2015 37th Annual International Conference of the IEEE Engineeri by	<1	Publication
60	12 Information Management by -2016	<1	Publication
61	An Iterative Leave-One-Out Approach to Outlier Detection in RNA-Seq Data by George-2015	<1	Publication
62	Article Published by International Research Journal of Engineering and Technology (IRJET) - www.irjet.net	<1	Publication
63	arxiv.org	<1	Publication
64	A theorem on the difference of the generalized inverses of two nonnegative matri by Milliken-1977	<1	Publication

65	bmcinfectdis.biomedcentral.com	<1	Publication
66	Boosting Feature Matching Accuracy With Pairwise Affine Estimation by Dai-2020	<1	Publication
67	China in Astleys New General Collection of Voyages as It Appeared in a Spanish by Russell-2016	<1	Publication
68	citeseerx.ist.psu.edu	<1	Internet Data
69	docplayer.net	<1	Internet Data
70	home.ijasca.com	<1	Publication
71	IEEE 2018 International Conference on Computer, Communication, Chem, by Dhar, Prashengit G- 2018	<1	Publication
72	ijrte.org	<1	Publication
73	mdpi.com	<1	Internet Data
74	mdpi.com	<1	Internet Data
75	mdpi.com	<1	Internet Data
76	moam.info	<1	Internet Data
77	Modified Enrichment Strategies Coupled with Molecular and Conventional Methods t by Rohonczy-2013	<1	Publication
78	Phases of health promotion implementation into the Scottish school system by Gugglberger-2014	<1	Publication
79	scholar.sun.ac.za	<1	Internet Data
80	symphonyindustrial.ai	<1	Internet Data
81	Thesis Submitted to Shodhganga, shodhganga.inflibnet.ac.in	<1	Publication

82	www.acquisition.gov	<1	Internet Data
83	www.freepatentsonline.com	<1	Internet Data
84	www.freepatentsonline.com	<1	Internet Data
85	www.ijert.org	<1	Internet Data
86	www.iraj.in	<1	Publication
87	www.science.gov	<1	Internet Data
88	www.scribd.com	<1	Internet Data
89	www.slideshare.net	<1	Internet Data
90	IEEE 2018 IEEE High Performance Extreme Computing Conference (HPEC), by Krastev, Plamen Re- 2018	<1	Publication