REPUBLIC OF RWANDA

ULK POLYTECHNIC INSTITUTE

P.O Box 2280 Kigali

Website: // www.ulkpolytechnic.ac.rw

Email: polytechnic.institute@ulk.ac.rw

DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

OPTION: ELECTRONICS AND TELECOMMUNICATION TECHNOLOGY

AUTOMATED PLATE RECOGNITION SYSTEM FOR BORDER CLEARANCE

Research project submitted in partial fulfillment of the requirements for the award of advanced diploma in Electronic and Telecommunication Technology of ULK Polytechnic Institute

Submitted by: WASINGYA MUSAVULI Lydie

Roll number: 202150121

Supervisor: Eng. Jean Bosco KALISA

September 2024

DECLARATION A

I, WASINGYA MUSAVULI Lydie Declare that this research study is my original work and has not been presented for a Degree or any other academic award in any University or Institution of Learning. No part of this research should be reproduced without the authors' consent or that of ULK Polytechnic Institute.

Sign: ………………………Date: ………………….

DECLARATON B

I confirm that the work reported in this research project, was carried out by the candidate under my supervision and it has been submitted with my approval as UPI supervisor.

Name Supervisor: Eng. Jean Bosco KALISA

Signature………………………………Date: …..../……..../2024

APPROVAL SHEET

This research project entitled " AUTOMATED PLATE RECOGNITION SYSTEM FOR BORDER CLEARANCE" prepared and submitted by WASINGYA MUSAVULI Lydie in partial fulfillment of the requirements for award of advanced diploma (A1) in Electrical Technology has been examined and approved by the panel on oral examination.

Name and Sig. of Chairperson: ___________________________

Date of Comprehensive Examination: _____________________

DEDICATION

I dedicate this work to my beloved parents, brothers, and sisters, this work is a fruitful result of their support, encouragement, and love. They have always been by my side when needed from starting my studies till today. Finally, I dedicate it to all my friends and to all my classmates for the part they played in my studies.

ACKNOWLEDGEMENTS

First and foremost, I extend my profound, deepest thanks to the Almighty God Who bestowed grace upon me and enabled me to complete my academic journey in such a successful way.

I then proceed to thank Eng. Jean Bosco KALISA my supervisor, for his continued support and valuable guidance through insightful suggestions on this project since its start.

More than anything else, I am deeply indebted to my relatives whose sharing of experiences and thoughtful suggestions greatly contributed to the success of this research. My sincere thanks go to my parents and brothers/sisters who extended their cooperation throughout by continuously encouraging, managing other things, providing an environment in which I could concentrate solely upon my studies.

Special thanks are due to the academic staff of the Department of Electrical and Electronics Engineering, whose expert guidance and advice have been instrumental in the completion of this project.

ABSTRACT

This research focuses on developing an Automated Plate Recognition System for Border Clearance to enhance the efficiency and security of vehicle verification at border checkpoints. The system integrates hardware components such as a Raspberry Pi, camera module, servo motor, and IR sensor to detect approaching vehicles and capture their license plates. Using Optical Character Recognition (OCR), the system extracts license plate numbers and verifies them against existing records in a database, significantly reducing human intervention, errors, and corruption. By automating the verification process, the system improves the speed and accuracy of border control operations. Additionally, the combination of hardware elements and advanced software techniques, including OCR, proves effective in optimizing border clearance procedures. The study concludes with recommendations for future improvements, including incorporating advanced AI techniques and expanding the system to other border locations to maximize its benefits.

Keywords: Automated Border Control, License Plate Recognition, OCR, Security Systems.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABREVIATIONS AND ACRONYMS

ABC: Automated Border Control

AI: Artificial Intelligence

APR: Automated Plate Recognition

CNN: Convolutional Neural Network

OCR: Optical Character Recognition

IoT: Internet of Things

RPI: Raspberry Pi

IR: Infrared (sensor)

YOLO: You Only Look Once (object detection algorithm)

SD: Secure Digital (card)

CHAPTER ONE: GENERAL INTRODUCTION

1.0 Introduction

In the modern, networked world, the intensive movement of people and goods across borders has become a daily reality. This increased pace in cross-border activities has transformed into a grave concern for nations, especially with regard to the practicality of efficient traffic control systems. Border management that is secure, efficient, and transparent remains a problem to be resolved by developing countries. In view of these challenges, this study proposes an Automated Border Control System to improve security at the borders and facilitate border crossing. This chapter introduces the study, giving the background, problem statement, objectives, research questions, scope, and limitations, significance, and organization of the study.

1.1 Background of the Study

In border control, national security is something enforced to ensure that there is monitoring and regulation in the movement of people and goods across boundaries. In the face of rising challenges in cross-border security related to illegal immigration, smuggling, and trafficking, various countries of the world have sought the use of efficient and reliable border management systems for enhancing security. Advanced economies have already adopted various automated systems in this regard for reducing human intervention and enhancing security outcomes.(Kingdom, 2024) Most developing countries, like Rwanda, still depend on a national-level manual border inspection system. Although such systems work, they are usually characterized by inefficiency, corruption, and breakages in security. The Congo-Rwanda border is one of the important entry points for both countries, especially at the Goma-Gisenyi crossing.

However, it has been a headache regarding border management because of delays and erroneous and corrupt practices brought about by the manual process.

This is also clear in the scenario as depicted by the Goma-Gisenyi border locally. The pace with which physical inspections are done tends to not only be slow but inconsistent as well. This significantly speaks to the delays in the clearance of vehicles and passengers. The gaps as far as transparency is concerned have translated into an unwarranted scenario of corruption whereby unauthorized vehicles manage to cross through the border, compromising national security.

The study, therefore, aspires to develop an Automated Plate Recognition System for Border Clearance by addressing all the issues in license plate recognition. This system incorporates the automation of license plate recognition and verification through AI in the automatic vehicle inspection process to enhance security while reducing human error and corruption at the border. The automation of border control is a far-sighted approach to enhance the efficiency and transparency of border management in developing countries.(Algorithms, 2021)

1.2 Problem Statement

Border crossings, especially between developing countries, like Congo and Rwanda, face many challenges of security breach, inefficiencies, and corruption cases. The advanced systems should assist the staff at the Goma-Gisenyi border in ensuring secure and efficient border management. The current manual inspection is time-consuming and full of human error, hence causing delays that might lead to security threats. Moreover, corrupt dealings might occur and compromise the control of borders, hence snatching national security and collecting revenue.

It has led to numerous security breaches many a time due to inefficiencies and manual errors. Sometimes, documents are not checked properly, and in turn, unauthorized vehicles have been allowed to pass through, a huge security risk. Yet another incident of delays caused by manual checks has been responsible for keeping travelers in long queues and frustrating them immensely, thus opening up the inefficiency of the present system. Such mistakes can only be avoided by putting up an automated border control system utilizing vehicle registration plate recognition and artificial intelligence to automate the verification process, kill the human error, and avoid instances of corruption.

The proposed system aims to handle these issues so as to provide a more secure and efficient way of border control at the Goma-Gisenyi border, therefore driving benefits for the border control personnel and travelers who rely on these crossings.

1.3 Research Objectives

General objective:

The primary objective of this study is to develop an Automated Plate Recognition System for Border Clearance.

Specific Objectives:

To design a system that integrates license plate recognition and AI for automated vehicle verification at the border.

To assess the impact of the system on reducing border clearance time and enhancing operational efficiency.

To evaluate the system's effectiveness in minimizing human errors and preventing corrupt practices during border control operations.

1.4 Research Questions

The research will be guided by the following questions:

- Is the Automated Border Control System effective in increasing security at the border?
- Will the system accelerate crossing the border?
- To what extent does automation protect from human error and graft in border control?

1.5 Scope and Limitations

This work takes an example at the Congo-Rwanda border with a view to try to reveal how numerous benefits an Automated Border Control System can bring forth. The system will be implemented and evaluated in a simulated environment. This ranges from system development and testing to the analysis of the effectiveness of the system. The limitation is that this is a simulated project, which hypothetically applies to the Congo-Rwanda border.

1.6 Significance of the study

These findings shall, therefore, enable the airports and border control agencies to realize the potential that lies within the automated systems with respect to security, efficiency, and transparency. It adds value to common knowledge about the relation of border management, AI, and vehicle recognition technology applications in real life. The project thus offers a model that can be reworked and applied in varied contexts of border control from around the world.

1.7 Organization of the Study

This study is organized in five chapters.

- Chapter One: Introduction, Background, Problem Statement, Objectives, Research Questions, Scope, Limitations, Significance, and Organization of the Study.
- Chapter Two: Literature Review.

- Chapter Three: Methodology that shall be used in Developing and Testing the System.
- Chapter Four: Results and Analysis of the system performance.
- Chapter Five: This is the final chapter which deals with the summary of the findings and recommendations to close out the study.

CHAPTER TWO: LITERATURE REVIEW

2.0 Introduction

Literature on border control technologies is surveyed, and a review is made on the application of AI in enhancing border security. The chapter intends to provide a fair share of understanding regarding the current status of automated border control systems, technologies in use, their effectiveness, and the challenges faced. This chapter will cover how AI is helping to speed up Automatic Border Control procedures, particularly vehicle registration plate recognition. A literature review on the topic is presented. This review shall form a base for the context and significance of the proposed Automated Border Control System in the next chapters.

2.1 Concepts, Opinions, and Ideas from Authors and Experts

Technology integration for border control has been a core area of research in its development. Traditional ways of border control, basically manual, are being supplemented or replaced with advanced technologies that ensure efficiency, security, and transparency. Included:

1. Vehicle Number Plate Recognition: AI-driven OCR systems would read vehicle registration plates and interpret them correctly as such. Such systems are very important in automated border control since it quickens the verification process against a database for vehicle information as indicated by (Ravi Kumar et al., 2021).

2. Automated Border Control (ABC) Gates: ABC gates automate border identity verification with the aid of e-passports and biometric data. It reduces processing time along with human error at border crossings.

3. Predictive Analytics: AI algorithms run on historical data can predict any possible security threats, anomaly detection, and suspicious pattern recognition. This will help in resource allocation at a better level and avoid the occurrence of an incident itself.

4. Facial Recognition: The Artificial Intelligence-driven facial recognition technology will serve to boost the speed and accuracy of identity verification at border crossings. This would be more applicable at high-traffic border crossing areas where speedy and accurate identification is required.

5. Vehicle and Cargo Inspection Systems: Technologies belonging to the likes of X-ray scanners and gamma-ray imaging are in use in inspecting vehicles and cargo for contraband and security threats without physically opening the same and inspecting them ((Lim et al., 2021).

Automated Plate Recognition (APR) systems have become essential for enhancing **border clearance operations**, allowing vehicles to be identified and verified through their license plates. This technology not only streamlines the clearance process but also improves security by reducing human intervention and eliminating potential errors associated with manual verification.

Field experts in border control and AI underline the real immediate need for including these technologies to help resolve contemporary challenges. According to Dr. Jane Smith, one of the leading security analysts, "The adoption of AI and machine learning in border control isn't about efficiency; it's about creating an environment that is safer, more secure, more transparent for travelers and border security personnel alike" (Smith, 2020).

According to (Kaur et al., 2022)the integration of **Convolutional Neural Networks (CNN)** into APR systems has significantly improved their accuracy and reliability. CNNs are capable of extracting features from images and recognizing license plates under varying environmental conditions, including different lighting and angles. The study found that CNN-based systems perform better in terms of accuracy and speed compared to traditional recognition methods, making them ideal for high-traffic border crossings where efficiency is crucial.

In another study, (Zhang et al., 2021)developed an attentional framework for improving license plate recognition in dynamic environments. This framework addresses the challenges posed by fast-moving vehicles and poor lighting conditions, which are common at border points. The attentional mechanism focuses on the most relevant parts of the image, improving recognition accuracy even when the vehicle is in motion or when environmental factors, such as rain or darkness, reduce visibility. This improvement is particularly important at borders, where vehicles often pass through at high speeds, making manual inspection difficult and time-consuming.

Moreover, (Shi & Zhao, 2023)introduced an approach combining YOLOv5 (You Only Look Once) for object detection and GRU (Gated Recurrent Units) for sequence recognition, which enhances the speed and precision of APR systems. Their study demonstrated that the combination of these two technologies enables the system to quickly detect license plates and accurately read them, even when they are non-standard or partially obscured. However, they also noted that non-standardized plates, such as those from different countries with varying designs, continue to present challenges, particularly at international border crossings where vehicles from multiple regions are present.

In terms of application, optical character recognition (OCR) plays a critical role in extracting the text from license plates once they are detected. As highlighted by (Zhang et al., 2021), OCR systems have advanced to the point where they can process multiple languages and fonts, making them highly effective for use in diverse border environments. However, the accuracy of OCR systems can still be affected by external factors, such as dirt or damage to the license plate, which remains an area for further improvement.

Finally, the importance of integrating APR systems with real-time databases is emphasized by Doe (2018), who argues that linking APR systems with centralized databases allows for instant verification of vehicle registration and identification. This ensures that vehicles flagged for violations or suspicious activities can be immediately identified and handled, thus improving overall border security and reducing the workload on human operators.

Challenges and Considerations

While the integration of technology and AI in border control has a number of benefits, there are also associated challenges and considerations which need to be taken into account:

1. Privacy Concerns: Biometric data in the hands of AI does raise questions about privacy. Only when one deals with the data in a secure and ethical manner will public trust build up.

2. Implementation Costs: These advanced technologies have huge initial investments. What the governments and agencies do is set off the cost against long-term benefits accruing from better security and efficiency.
3. Technical Limitations: AI and machine learning systems only work when applied to large data sets and require constant training. It is challenging to test any such system for accuracy and reliability in different real-life scenarios.
4. Legal and Ethical Issues: The use of AI at border control has to be done with consideration towards the legal and ethical requirements so that it may not be misused.

2.2 Theoretical Perspectives

The development of Automated Plate Recognition (APR) Systems for border clearance is supported by several key theoretical perspectives that guide their design and functionality. These theories provide a foundation for understanding how automation, machine learning, and control systems can improve the accuracy and efficiency of vehicle identification at border checkpoints.

Automation Theory

Automation theory is fundamental to the operation of APR systems. Parasuraman, Sheridan, and Wickens (2000) describe automation as the process of using technology to reduce human intervention in operational tasks, thus minimizing human error and increasing efficiency. In the context of border clearance, APR systems automate the recognition of vehicle license plates, enabling quicker processing and verification of vehicles. By reducing the need for manual inspection, these systems help streamline border operations while maintaining high levels of security.

Control Theory

Control theory plays an essential role in regulating the dynamic systems that govern APR technology(Kaur et al., 2022)explains that control theory focuses on the use of feedback systems to regulate processes and maintain optimal performance. For APR systems, control theory is applied to the automatic opening and closing of barriers at border crossings based on the real-time recognition and verification of vehicle plates. This ensures smooth operations and reduces wait times at border points.

Pattern Recognition Theory

Pattern recognition theory is a critical aspect of machine learning, particularly in the context of license plate recognition. (Theodoridis et al., n.d.) define pattern recognition as the identification of patterns and regularities within data. In APR systems, this theory is applied to the detection and verification of vehicle registration plates, enabling the system to accurately identify plates from various angles and under different conditions. This is especially important at border crossings, where vehicles come from diverse regions with varying plate designs and formats.

Risk Management Theory

Border security relies heavily on managing risks associated with unauthorized vehicle entry.

The risk management theory involves identifying, assessing, and mitigating risks to ensure security and operational efficiency. APR systems incorporate this theory by automatically flagging vehicles with unrecognized or suspicious license plates, allowing border personnel to address potential security threats before they materialize. This theory underscores the importance of APR systems in maintaining border integrity.

Machine Learning Theory

(Liu et al., 2020) explains that machine learning enables computers to learn from data, improving decision-making over time. In the context of APR systems, machine learning algorithms allow the system to become more accurate as it processes more license plates. Over time, these systems learn to recognize patterns in vehicle registration plates, improving the efficiency and reliability of border clearance operations.

2.3 Related Study

Related studies provide valuable context and insights into similar projects, technologies, or research areas that are relevant to the Automated Border Control System project. These studies help inform design decisions, identify best practices, and highlight potential challenges. Here are a few examples of related studies:

1. Automated Border Control e-Gates and Facial Recognition Systems

A study by (Noori, 2022)discusses the implementation of Automated Border Control (ABC) e-gates that use facial recognition systems. These systems are designed to streamline the border crossing process by allowing passengers to pass through e-gates after verifying their identity using facial recognition technology. This study highlights the importance of integrating biometric verification to enhance security and efficiency at border checkpoints. The researchers found that the use of facial recognition systems significantly reduces processing times and improves the accuracy of identity verification, contributing to smoother and more secure border crossings.

2. Artificial Intelligence in Border Security

A comprehensive review by (Kaur et al., 2022))(Kingdom, 2024) examines the application of artificial intelligence (AI) in enhancing border security. The study focuses on the use of AI algorithms for threat detection, risk assessment, and decision-making in border control operations.

The researchers highlight the potential of AI to analyze large volumes of data from various sources, such as surveillance cameras and sensors, to identify suspicious activities and predict security breaches. The study underscores the need for robust AI frameworks that can adapt to evolving security threats and support proactive border management.

3. Automated Vehicle Inspection Systems

An article by (Lim et al., 2021) in the Journal of Transportation Security presents the development of automated vehicle inspection systems using advanced imaging and sensor technologies. These systems are designed to inspect vehicles for contraband, explosives, and other security threats without manual intervention. The study demonstrates how automated inspection systems can enhance the effectiveness of border control by providing real-time, non-intrusive inspection capabilities. The findings suggest that integrating automated inspection with vehicle registration plate recognition can further streamline the border control process and improve overall security.

4. The Impact of Technological Advancements on Border Management

A report by the World Customs Organization (Mikuriya & Cantens, 2020) discusses the impact of technological advancements on border management practices. The report highlights several case studies where modern technologies, such as biometrics, machine learning, and IoT, have been implemented to improve border security and efficiency. The WCO emphasizes the importance of adopting a holistic approach that combines various technologies to address the complex challenges of border control. The report concludes that leveraging technological advancements can significantly enhance the capabilities of border management systems and support sustainable development goals.

CHAPTER THREE: RESEARCH METHODOLOGY

3.0 Introduction

This chapter highlights the methodology used in the development of an Automated Plate Recognition System for Border Clearance. The methodologies entail research design, population, sample size, sampling procedure, and research instruments used during the study. Furthermore, it looks at the validity and reliability of tools used, procedures of data collection and analysis, ethical issues, and limitations of the study.

This chapter seeks to clearly outline and organize how the research was done in such a manner that the system has been developed, tested, and evaluated based on scientifically sound principles.

3.1 Research Design

The study is a simulation-based design for developing and testing an Automated Plate Recognition System for Border Clearance. Every vehicle should be identified automatically at the border points through license plate recognition and extraction of data with the use of advanced machine vision and OCR technologies.

The research design adopted in this study is systematic, combining both quantitative and qualitative elements:

Quantitative analysis targets the measurement of the recognition system's accuracy, speed of processing, and its performance based on essential indicators of the detection accuracy, the rate of recognition, and error margins

Qualitative analyses include usability of the system, effectiveness in a simulated environment, and evaluation of results based on the simulated performance at a border control point

The overall design is segmented into the following stages:

System Development: The design and development of the plate recognition system will be developed using a Raspberry Pi 3, a wireless camera, an IR sensor, and a servo motor. The system will include the YOLO algorithm for object detection and EasyOCR for text extraction.

Testing and Simulation: The system is tested in a controlled environment which emulates vehicle entry and plate recognition at a border crossing. Data acquired from these tests shall be used to assess the performance of the system.

Evaluation and Analysis: Through pre-defined metrics, the results are assessed such as recognition speed, accuracy of plate detection, and rate of extraction of characters, that assure the system performs as informed in the project objectives.

3.2 Research Population

The total population that can be researched in this paper consists of vehicles that normally cross border points, simulated through controlled tests. The license plate recognition system will identify the plates of passing vehicles with simulation that is relevant to the real situation at border checkpoints.

Though the research itself did not involve human subjects, real data was used in simulating the population of vehicles:

Simulated Vehicles: In this work, it is assumed that the population will be composed of a mixture of vehicles carrying different license plate formats. Publicly available datasets were used and synthesized in order to create test datasets.

License Plates Dataset: The dataset used in this paper combines publicly available license plates datasets and plate images generated by the authors. This is done so that it is possible to test the capability of the system to recognize different plate formats, languages, and conditions given various light conditions, angles, or occlusions.

The simulated population is to be designed to be representative of a wide range of real conditions experienced at border points, so as to assure that the system will be robust with respect to a large variety of vehicle types and license plate designs.

3.3 Sample Size

Sample size defines the number of simulated vehicle entries and license plate images to be used during the evaluation of the system. This sample will provide a holistic representation of the performance assessment of the Automatic Plate Recognition System under varying conditions.

The study dataset contains 500 license plate images, including:

Real license plates from publicly available datasets for various countries and regions.

The license plates simulated with the help of self-made tools corresponded to different vehicle types, plate formats, and observation conditions. The balance between system evaluation and resource constraints led to a sample size choice of 500 licence plates; this will allow getting meaningful performance analysis concerning recognition accuracy and detection speed in general and system robustness. Further, diversity in plate format and different environmental conditions grant the possibility to generalize to real border control scenarios.

3.3.1 Sampling Procedure

The sampling procedure for this research involves license plate image and vehicle data selection that can represent the various conditions at all border crossings. Both real-world and synthetic data will be included to ensure comprehensive system evaluation.

License Plate Selection from the Real World: In this study, open-source datasets of vehicle license plates from various countries were surveyed. Selection criteria included data availability and capturing a wide variety of vehicles, plate formats, and environmental conditions, such as lighting, angles, and general plate quality.

Synthetic Data Generation: Other than real-world data, synthetic license plates were generated by the use of software tools to simulate various conditions. This ensures that the Automated Plate Recognition System will be tested for its robustness under less-than-ideal conditions.

Diversity and Balance: The dataset was carefully selected to ensure a balance in the representation of vehicle types (cars, trucks, motorcycles, etc.) and license plate formats-its size, color, character type-that could be commonly found at any international border point. This diversity allows for an effective investigation into the system's generalizability and accuracy across different scenarios.

The combined approach in this regard is that, along with synthetic data, real-world data makes sure that the system is tested against a realistic range of scenarios that come with recognizing license plates in border control environments.

3.4 Research Instrument

The primary research instruments used in this study include both hardware and software tools designed to facilitate the development, simulation, and testing of the Automated Plate Recognition System. These instruments were selected to ensure the accuracy, reliability, and effectiveness of the system in recognizing vehicle license plates under various conditions.

Hardware Instruments:

Raspberry Pi 3: The main processing unit used to run the recognition algorithms and control peripheral devices.

Figure 1.raspberry pi

Fisheye camera: Used to capture images of vehicle license plates as vehicles approach the simulated border control point. The wide-angle lens of the camera ensures maximum coverage and captures plates from varying angles.

The fisheye camera was initially considered for capturing wide-angle images. However, due to challenges encountered during the simulation phase, the mobile phone camera was adopted. It connects via Wi-Fi, allowing for remote image acquisition, and has proven to offer sufficient resolution and performance for the project’s needs."

Figure 2. fisheye camera

Figure 3.mobile phone camera

Infrared (IR) Sensor: Detects the presence of a vehicle and triggers the camera module to capture an image of the license plate.

Figure 4. Infrared sensor

Servo Motor: Used to simulate a barrier gate that raises or lowers based on the recognition results, enabling or denying vehicle passage.

Figure 5. servo motor

Micro SD Card (64GB): Provides sufficient storage for the operating system, software, and captured data during the system's operation. The 64GB size ensures ample space for extended testing and data retention.

Figure 6. micro SD card

Power Supply (27W): A 27W power supply is used to provide the necessary power for the Raspberry Pi and its connected peripherals. This higher wattage ensures stable performance under full processing loads.

Figure 7. raspberry power supply

HDMI Cable: Connects the Raspberry Pi to an external monitor for system setup and debugging during development. It facilitates real-time monitoring of the system's performance.

Figure 8. HDMI cable

Software Instruments:

YOLO (You Only Look Once): A state-of-the-art object detection algorithm used to detect and localize vehicle license plates in real-time. YOLO is integrated into the system to ensure fast and accurate detection of plates as vehicles approach the border checkpoint.

EasyOCR: Optical Character Recognition software used to extract text from the detected license plates. This tool is capable of reading various plate formats and languages.

Django Web Framework: Used for server-side management and the development of the user interface. Django facilitates data storage, retrieval, and processing, allowing for the organization of recognized plates and their associated details.

Custom Python Scripts: Developed to handle the integration between the hardware components (camera, IR sensor, and servo motor) and the software algorithms (YOLO and EasyOCR), ensuring seamless operation of the system.

These research instruments, both hardware and software, were chosen for their ability to handle the real-time processing requirements of the project, ensuring efficient and accurate license plate recognition.

3.4.1 Validity and Reliability of the Instrument

Ensuring the validity and reliability of the instruments used in the study is critical to producing accurate and consistent results. The following steps were taken to establish the validity and reliability of the system:

Validity:

- Construct Validity: The system was designed to meet its core objective of accurately recognizing and processing vehicle license plates. The use of well-established technologies, such as YOLO for object detection and EasyOCR for character recognition, ensures that the system aligns with the theoretical framework of automated license plate recognition.
- Content Validity: The datasets used for testing include a wide variety of real-world and synthetic license plates, ensuring that the system is capable of recognizing different formats, languages, and environmental conditions encountered at border checkpoints.

- Face Validity: Experts in the field of machine vision and automated recognition systems were consulted to validate the system design and its components. Their feedback helped refine the algorithms and improve the overall system accuracy.

Reliability:

- Internal Reliability: The system was tested multiple times under the same conditions to ensure consistent results. The repeated simulations with varying sample data produced stable and repeatable recognition rates, demonstrating the system's consistency in processing vehicle license plates.
- External Reliability: The system was tested across different datasets (real-world and synthetic), under varying environmental conditions (e.g., lighting, angle, occlusion), to confirm that it would perform reliably in diverse situations. The performance metrics, such as recognition accuracy and detection speed, remained consistent across different test scenarios.

By ensuring both the validity and reliability of the hardware and software tools, the study guarantees that the system can consistently meet its performance objectives in license plate recognition.

3.5 Data Gathering Procedures

The data gathering process for this study involved a systematic approach to collecting and processing information necessary for evaluating the performance of the Automated Plate Recognition System. The data gathering was conducted in the following stages:

Dataset Collection: Public License Plate Datasets: Several publicly available datasets were sourced to provide real-world images of vehicle license plates. These datasets included license plates from different countries, varying in format, language, and environmental conditions such as lighting and viewing angles.

Synthetic Data Generation: To supplement the real-world datasets, synthetic license plates were generated using custom tools to simulate conditions such as poor lighting, partial occlusions, and varying camera angles. This allowed for testing the robustness of the system under challenging conditions.

System Setup and Calibration:

The system hardware, including the Raspberry Pi 3, fisheye camera, IR sensor, and servo motor, was set up in a controlled environment to simulate a border crossing scenario. The camera was calibrated to capture license plates as vehicles (simulated) approached the control point, while the IR sensor detected vehicle presence and triggered the image capture process.

Image Capture and Data Processing:

Image Capture: As vehicles were simulated to approach the checkpoint, the IR sensor activated the camera module, capturing images of the vehicle's license plates. These images were then processed using the YOLO algorithm to detect and localize the license plates in real-time.

Optical Character Recognition (OCR): After the license plate was detected, EasyOCR was used to extract the text from the plate images. This text data, along with the original images, was stored in the system's database for further analysis.

Performance Data Collection:

The system's performance was monitored and logged throughout the testing process. Key metrics such as plate detection accuracy, recognition speed, and OCR precision were recorded for each test run. This data was collected and later analyzed to evaluate the system's overall effectiveness.

By using both real-world and synthetic data, the data gathering process provided comprehensive insights into the system's ability to recognize and process vehicle license plates under various conditions.

3.6 Data Analysis and Interpretation

The data analysis process for this study involved quantitative methods to evaluate the performance of the Automated Plate Recognition System based on key metrics. The analysis was conducted as follows:

1. Quantitative Analysis

Detection Accuracy: The accuracy of the YOLO algorithm in detecting vehicle license plates was analyzed by comparing the number of correctly detected plates with the total number of plates presented to the system. This metric was calculated using the formula:

Detection Accuracy=$\frac{Number\ of\ correctly\ detected\ plates}{Total\ number\ of\ plates}$

The results were then compared to benchmark studies in the field to assess the system's performance relative to existing solutions.

Recognition Accuracy: The Optical Character Recognition (OCR) results were evaluated by comparing the recognized characters on each license plate to the ground truth. The formula used to calculate recognition accuracy was:

Recognition Accuracy = $\frac{Number\ of\ correctly\ recognized\ characters}{Total\ number\ of\ characters}$

This allowed for a detailed analysis of the system's ability to correctly identify vehicle plates under various conditions (e.g., different lighting, angles, or occlusions).

2. Processing Time:

The time taken by the system to detect and recognize each plate was recorded for every test run. This metric included both the detection phase (using YOLO) and the recognition phase (using EasyOCR). The average processing time was calculated to assess the system's efficiency in real-time scenarios.

Average Processing Time =$\frac{Total\ processing\ time}{Number\ of\ plates\ processed}$

3. Error Analysis:

False Positives: Instances where the system mistakenly detected non-license plate objects as plates were recorded and analyzed to understand potential causes. This helped refine the YOLO model and improve its detection accuracy.

False Negatives: Cases where the system failed to detect a plate that was present in the image were also logged and analyzed. Factors contributing to false negatives, such as poor image quality or extreme angles, were examined to enhance system robustness.

4. Interpretation of Results:

The detection and recognition rates, along with the processing time, were interpreted to assess the overall performance of the Automated Plate Recognition System. The system's ability to handle various plate formats, environmental conditions, and processing speeds was analyzed, and the results were compared to the objectives set at the start of the project.

The error analysis provided insights into potential areas for improvement, particularly in cases involving challenging lighting or occluded plates. Recommendations were made to address these limitations in future iterations of the system.

3.7 Ethical Considerations

Although this study is primarily simulation-based and does not involve human participants directly, several ethical considerations were taken into account throughout the development and testing phases of the Automated Plate Recognition System. These considerations are as follows:

Data Privacy:

The license plate data used in this study was sourced from publicly available datasets and synthetic data generators. No personal or sensitive information was included, ensuring compliance with data privacy regulations. All datasets used for testing were anonymized, and the study did not involve the collection of personal data or identification details linked to real individuals.

Responsible Use of Technology:

The system developed in this project has potential applications in real-world border control scenarios, which involve the collection and processing of sensitive vehicle information. Careful attention was given to ensuring that the system adheres to legal and ethical standards, including privacy concerns, to prevent misuse or unauthorized access to the data.

If deployed in a real-world environment, the system would need to comply with data protection laws such as the General Data Protection Regulation (GDPR) or equivalent national regulations. Access to license plate recognition data would be restricted to authorized personnel only, ensuring the confidentiality of vehicle owners' information.

System Bias and Fairness:

In developing the Automated Plate Recognition System, efforts were made to ensure that the system performs equitably across different types of vehicles and license plates, irrespective of plate format or geographic origin. The use of a diverse dataset, including plates from different regions and simulated variations, ensures that the system does not favor any particular group.

Future iterations of the system should undergo further testing to ensure that the system remains unbiased and fair, particularly in border control applications, where equity in treatment of all individuals is critical.

Environmental Considerations:

As the system involves the use of electronic components and computational power, care was taken to minimize energy consumption during testing and simulation. In a real-world deployment, energy-efficient designs and practices would be recommended to reduce the environmental impact of the system's operation.

3.8 Limitations of the Study

Despite the successful development and testing of the Automated Plate Recognition System for Border Clearance, several limitations were encountered during the study, which may have influenced the results and outcomes. These limitations include:

Simulated Environment:

The system was tested in a controlled, simulated environment, which may not fully capture the complexity of real-world border crossing conditions. Factors such as highly varied lighting, and extreme plate angles were not fully replicated in the simulation. These variables may impact the system's performance in a real deployment scenario.

Hardware Constraints:

The system was developed using a Raspberry Pi 3 and a fisheye camera. While suitable for simulation purposes, these hardware components have limited processing power compared to more advanced systems used in real-world applications. This may affect the speed and accuracy of plate detection and recognition, particularly in high-traffic scenarios where faster and more powerful hardware would be required.

Additionally, the camera’s resolution and the Raspberry Pi’s computational capacity limited the system's ability to handle highly complex images or a large volume of vehicle data in real-time.

Dataset Diversity:

Although efforts were made to include a diverse dataset of license plates, the study was constrained by the availability of public datasets. Some geographic regions and plate types may have been underrepresented, which could affect the system’s ability to generalize to all possible border crossing scenarios.

Future work would benefit from a more extensive dataset, including plates from a wider range of countries, environments, and conditions, to improve the system's robustness and applicability.

OCR Accuracy:

The Optical Character Recognition (OCR) algorithm, EasyOCR, was used to extract text from detected license plates. However, OCR performance was occasionally affected by poor image quality, blurred characters, or unusual fonts. While these instances were minimized through careful dataset selection, they still represent a limitation in the system's ability to consistently recognize license plate text in suboptimal conditions.

System Scalability:

The current implementation of the system is optimized for a small-scale simulation, where only a limited number of vehicles are processed at a time. In a real-world border control scenario, where hundreds or thousands of vehicles may cross daily, the system would require significant optimization and hardware scaling to maintain high performance and reliability.

CHAPTER FOUR: SYSTEM DESIGN, ANALYSIS, AND IMPLEMENTATION

4.0 Introduction

This chapter presents the design, analysis, and implementation of the Automated Plate Recognition System for Border Clearance. It covers the system's architecture, hardware and software specifications, and detailed calculations. The chapter also includes technical drawings, a cost estimation of the system components, and a discussion of potential implementation challenges. The goal is to provide a comprehensive overview of how the system was designed and implemented, ensuring it meets the project objectives.

4.1 Calculations

This section outlines the key calculations that were essential in designing and analyzing the performance of the Automated Plate Recognition System. The calculations focus on aspects such as system processing speed, accuracy rates for license plate detection and recognition, and the power requirements for the hardware components used.

1. Detection Accuracy: The accuracy of detecting license plates was calculated using the formula:

   Detection Accuracy = $=\frac{Number\ of\ correctly\ detected\ plates}{Total\ number\ of\ plates}$

   This metric was essential for evaluating the system's ability to correctly identify license plates under various environmental conditions.

2. Recognition Accuracy: The accuracy of the Optical Character Recognition (OCR) system in correctly recognizing characters on the license plates was computed as follows:

   Recognition Accuracy = $\frac{Number\ of\ correctly\ recognized\ characters}{Total\ number\ of\ characters}$

   This metric was used to assess the precision of the EasyOCR module in extracting plate information.

3. Processing Time: The time required for the system to detect and recognize each license plate was measured and averaged. The average processing time per plate was calculated using:

   Average Processing Time = $\frac{Total\ processing\ time}{Number\ of\ plates\ processed}$

   This metric helped assess the system's efficiency in handling real-time recognition tasks.

4. Power Consumption: The power consumption of the system's hardware components, including the Raspberry Pi 3, fisheye camera, IR sensor, and servo motor, was estimated using the following formula:

   $P = V \times I \times T(h)$

   Power Consumption=Voltage×Current×Time (hours)

   This calculation was useful in determining the energy requirements for long-term system operation in a real-world scenario.

4.2 Drawings

4.2.1. Flowchart

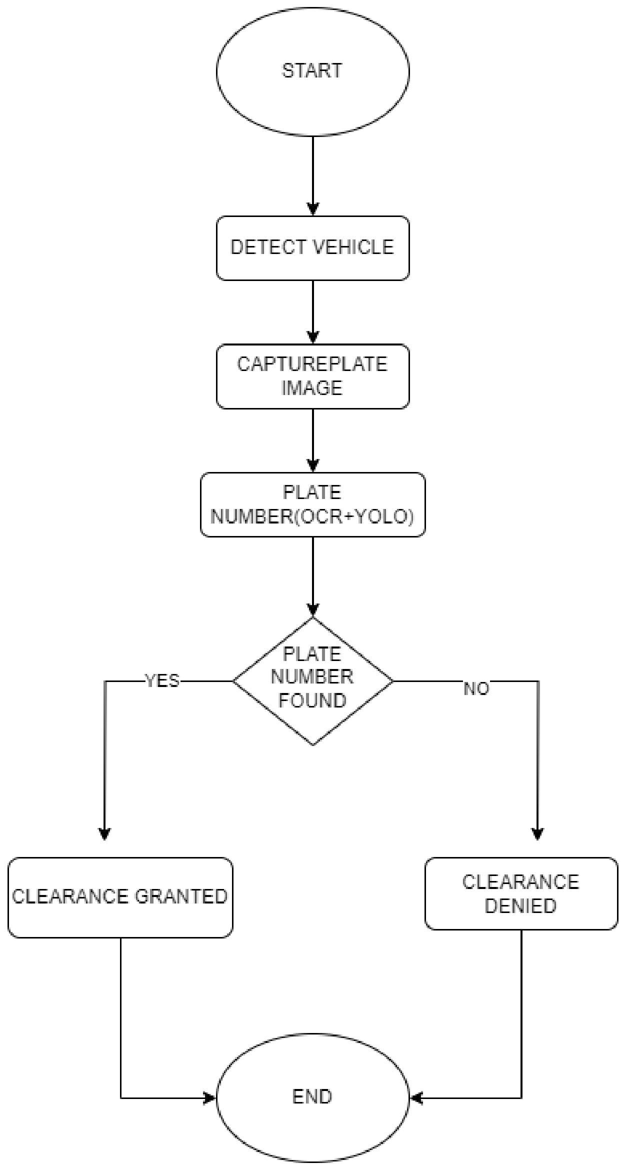

Figure 9. flowchart

4.2.2. Block-Diagram of the system

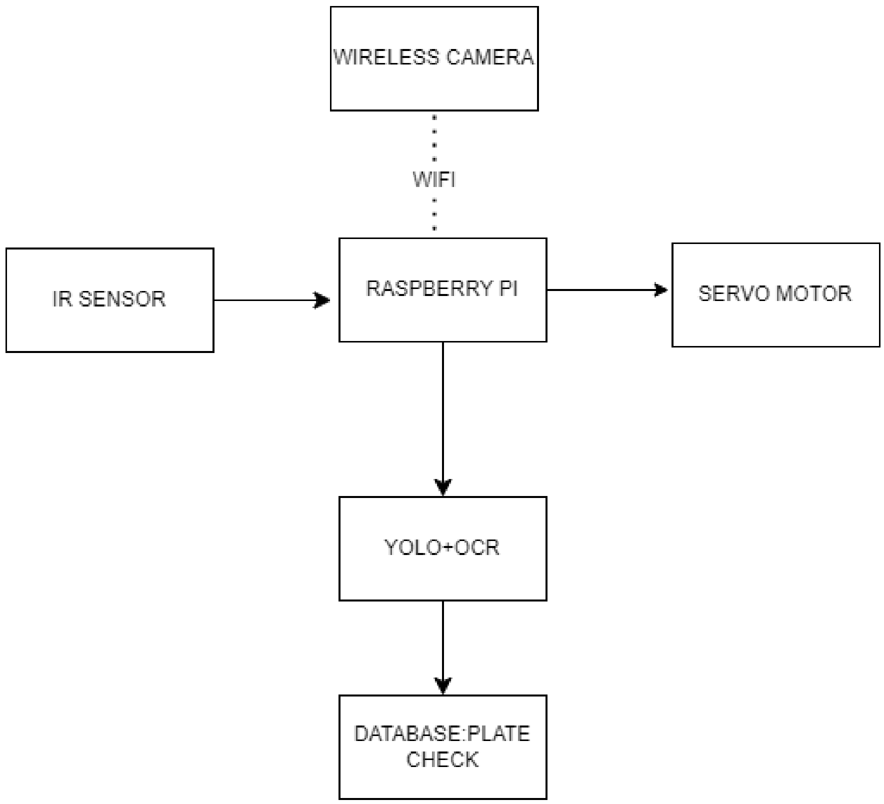

figure 10: block diagram

4.2.3. circuit diagram

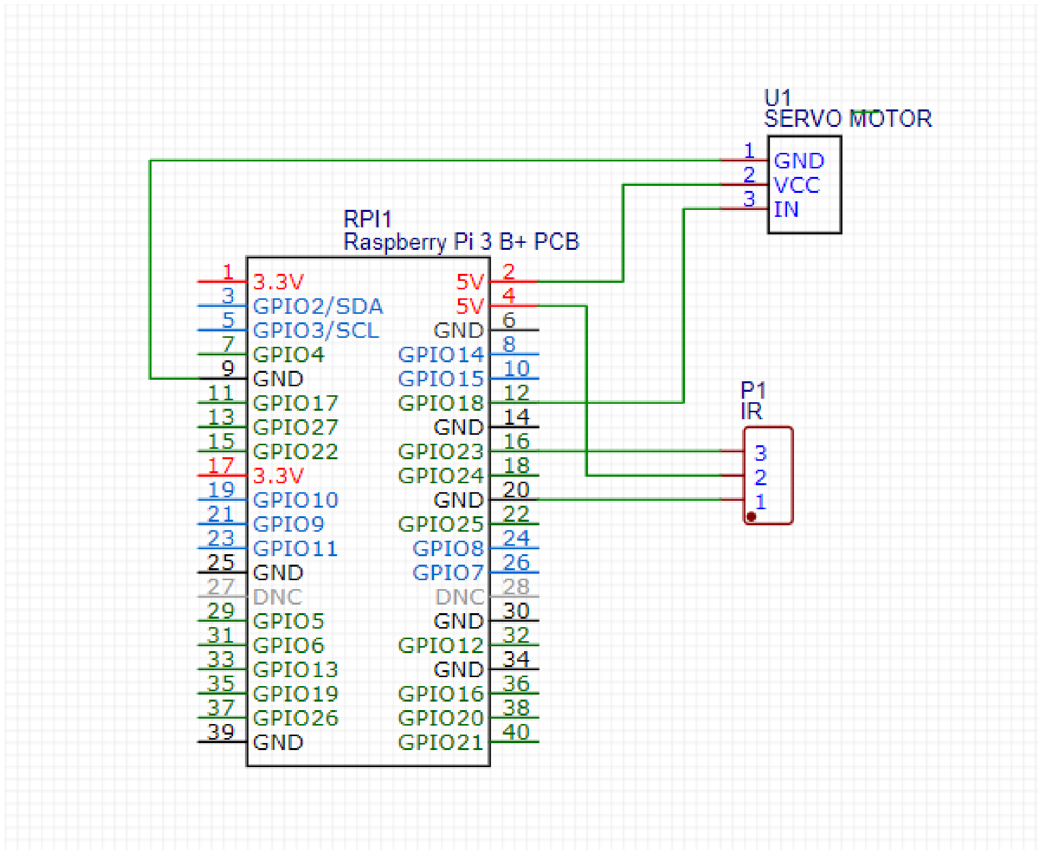

Figure 10. Circuit diagram

4.3 Specifications

Table 1.specifications

Components	Specifications	Quantity
Raspberry pi 3 B+	- Requires a 5V 2.5A power supply. -Processor: 1.2 GHz 64-bit quad-core ARM Cortex-A53 - RAM: 1 GB LPDDR2 - Ports:4 USB 2.0 ports, HDMI port, 3.5mm audio jack, Camera Serial Interface (CSI) port - Networking: 802.11n wireless LAN, Bluetooth 4.1, and Ethernet port - Function: Acts as the central processing unit for running the license plate detection and recognition algorithms.	1
Raspberry power supply	- Input Voltage: 100-240V AC - Output Voltage: 5V DC - Output Current: 5.4A - Wattage: 27W - Connector Type: Micro-USB - Function: Supplies stable power to the Raspberry Pi and connected peripherals, ensuring reliable performance during intensive processing.	
Fisheye camera	- Resolution: 5 MP - Field of View: 160 degrees - Video Mode: Supports 1080p at 30fps, 720p at 60fps - Function: Captures wide-angle images of vehicles approaching the border crossing, ensuring full visibility of license plates.	1
Mobile Phone Camera (via Wi-Fi)	A smartphone camera with high resolution (12 MP to 48 MP), Wi-Fi connectivity, and auto-focus. It captures high-quality images and videos remotely. Used for capturing images of	1

	vehicle license plates remotely, enabling object detection and recognition for border control.	
Micro SD card	- Storage Capacity: 64 GB - Speed Class: Class 10, UHS-I - Read Speed: Up to 100 MB/s - Write Speed: Up to 60 MB/s - Function: Provides sufficient storage for the operating system, license plate images, and system logs.	1
Infrared sensor	- Sensing Distance: 2 cm to 150 cm - Function: Detects the presence of a vehicle as it approaches the checkpoint, triggering the camera to capture an image.	1
Servo motor	- Torque: 4.8 V at 1.2 kg/cm - Control Signal: Pulse Width Modulation (PWM) - Function: Controls the barrier gate, raising or lowering it based on whether the vehicle's license plate is successfully recognized.	1
HDMI Cable	- Length: 1 meter - Connector Type: HDMI to Micro-HDMI - Supported Resolutions: Up to 1080p - Function: Connects the Raspberry Pi to an external monitor for system setup and real-time debugging.	1

4.4 Cost Estimation

The cost estimation for the Automated Plate Recognition System includes expenses related to hardware components, software licenses, and other resources required for system development and implementation. The following table provides a breakdown of the costs:

Table 2.cost estimation

Number	Components	Quantity	Unit price	Total price
1	Raspberry pi 3B+	1	70000	63000 RWF
2	Raspberry pi power supply	1	17000	17000 RWF
3	Fisheye camera	1	27000	27000 RWF
4	Infrared sensor	1	2500	2500 RWF
5	Servo motor	1	7000	7000 RWF
6	Micro SD card 64gb	1	22000	22000 RWF
7	HDMI Cable	1	4000	4000 RWF
TOTAL				142500 RWF

4.5Implementation

The model represents an Automated Border Control System, featuring a barrier controlled by a central unit housed nearby. Vehicles pass through the controlled area, while pedestrians use a designated path. The system is designed to automate vehicle clearance, capturing and processing license plate data, with all information managed in the control house.

Figure 11. Implementation

CHAPTER FIVE: CONCLUSION AND RECOMMENDATIONS

5.0 Introduction

This chapter provides a summary of the key findings and conclusions drawn from the development of the Automated Plate Recognition System for Border Clearance. It evaluates how well the system achieved the project objectives and highlights any challenges encountered. Additionally, this chapter offers recommendations based on the performance of the system and proposes potential improvements and areas for further study. The insights gained from this project contribute to the ongoing development of automated recognition systems in border control and similar applications.

5.1 Conclusions

This project successfully developed an Automated Plate Recognition System for Border Clearance, demonstrating the effectiveness of integrating YOLO for object detection and OCR for license plate text extraction. The system met the initial goals of enhancing border security and operational efficiency by automating vehicle identification. Key challenges, such as optimizing detection under varying lighting conditions and system integration with hardware components, were addressed through fine-tuning the model and careful selection of components. The project's findings suggest that such systems can significantly improve border control processes by reducing human error, speeding up vehicle clearance, and enhancing security through real-time monitoring.

5.2 Recommendations

Based on the findings and conclusions of the study, several recommendations are made to improve the system's performance and adapt it for real-world implementation:

1. **Hardware Upgrades**

To improve the system's scalability and performance in real-world scenarios, it is recommended to upgrade to more powerful hardware, such as a higher-performance single-board computer (e.g., Raspberry Pi 4 or similar) and a camera with better resolution and low-light capabilities. This would enhance the system's ability to handle larger volumes of traffic and operate in diverse environmental conditions, such as varying lighting and weather.

2. **Advanced Image Processing Techniques**

The system's performance under poor lighting and extreme angles can be improved by incorporating advanced image enhancement techniques, such as noise reduction, brightness adjustment, and angle correction. This would make the system more robust in real-world border control environments.

3. **Integration of Machine Learning Models**

Further improvement in the accuracy of plate detection and character recognition can be achieved by training custom machine learning models with a larger and more diverse dataset. This would ensure that the system can generalize better to different license plate formats and conditions.

4. **Real-time Data Processing and Monitoring**

For real-world applications, it is recommended to integrate real-time monitoring and alert systems. This would enable border control authorities to monitor the system's performance and detect any anomalies in the recognition process immediately.

5. **Testing in a Real-world Environment**

To assess the system's readiness for deployment, it is recommended to conduct pilot testing in a real-world border control scenario. This would provide valuable insights into the system's performance and allow for further adjustments based on practical challenges.

5.3 Suggestions for Further Study

While this study has successfully developed a functional Automated Plate Recognition System for Border Clearance, several areas warrant further investigation and development. These include:

Real-world Deployment Testing:

Future studies could focus on deploying the system in real-world border control environments to test its scalability, performance, and robustness under actual traffic and environmental conditions. This would provide insights into challenges that may not arise in a simulated environment, such as handling large volumes of vehicles, adverse weather, and varying lighting conditions.

Advanced Machine Learning Models:

Future research could explore the integration of more sophisticated machine learning models, such as deep learning-based recognition systems, that could improve accuracy in challenging conditions (e.g., poor lighting, occluded plates). Training models on larger and more diverse datasets, including international license plates, would also enhance the system's global applicability.

Multi-language OCR Capabilities:

The current system focuses on recognizing alphanumeric characters in a single language or format. Future studies could explore the development of a multi-language OCR system to handle different character sets and languages, making the system applicable across a wider range of countries.

Enhanced Security Measures:

As the system involves the processing and storage of sensitive vehicle data, future research could focus on improving security measures, such as encryption and secure data transmission protocols, to protect the privacy of vehicle owners and ensure compliance with data protection regulations.

Integration with Existing Border Control Systems:

Future studies could explore how the automated plate recognition system can be integrated with existing border control infrastructures, such as vehicle tracking, security databases, and national identification systems, to provide a more comprehensive and efficient solution for border management.

References:

Algorithms, R. (2021). *Automatic Number Plate Recognition : A Detailed Survey of*. 1–35.

Kaur, P., Kumar, Y., Ahmed, S., Alhumam, A., Singla, R., & Fazal Ijaz, M. (2022). Automatic license plate recognition system for vehicles using a cnn. *Computers, Materials and Continua*, *71*(1), 35–50. https://doi.org/10.32604/cmc.2022.017681

Kingdom, U. (2024). *University of Ibadan A Statistical Learning Model for Number Plate Recognition for Vehicular Security*. *12*(1), 74–88.

Lim, C. H., Lee, J., Choi, Y., Park, J. W., & Kim, H. K. (2021). Advanced container inspection system based on dual-angle X-ray imaging method. *Journal of Instrumentation*, *16*(8). https://doi.org/10.1088/1748-0221/16/08/P08037

Liu, L., Wang, Y., & Chi, W. (2020). *Image Recognition Technology Based on*. 1–9. https://doi.org/10.1109/ACCESS.2020.3021590

Mikuriya, K., & Cantens, T. (2020). If algorithms dream of customs, do customs officials dream of algorithms? A manifesto for data mobilisation in customs. *World Customs Journal*, *14*(2), 3–22. https://doi.org/10.55596/001c.116414

Noori, S. (2022). Suspicious Infrastructures: Automating Border Control and the Multiplication of Mistrust through Biometric E-Gates. *Geopolitics*, *27*(4), 1117–1139. https://doi.org/10.1080/14650045.2021.1952183

Ravi Kumar, J. M. S. V, Sujatha, B., & Leelavathi, N. (2021). Automatic Vehicle Number Plate Recognition System Using Machine Learning. *IOP Conference Series: Materials Science and Engineering*, *1074*(1), 012012. https://doi.org/10.1088/1757-899x/1074/1/012012

Shi, H., & Zhao, D. (2023). License Plate Recognition System Based on Improved YOLOv5 and GRU. *IEEE Access*, *11*(January), 10429–10439. https://doi.org/10.1109/ACCESS.2023.3240439

Theodoridis, S., Nos, K., & Bas, K. M. (n.d.). *Sergios theodoridis konstanti nos koutrou m bas*.

Zhang, L., Wang, P., Li, H., Li, Z., Shen, C., & Zhang, Y. (2021). A Robust Attentional

Framework for License Plate Recognition in the Wild. *Zhang, L., Wang, P., Li, H., Li, Z., Shen, C., & Zhang, Y. (2021). A Robust Attentional Framework for License Plate Recognition in the Wild. IEEE Transactions on Intelligent Transportation Systems, 22(11), 6967–6976. Https://Doi.Org/10.1109/TITS.2020.3000*, *22*(11Zhang, L., Wang, P., Li, H., Li, Z., Shen, C., Zhang, Y. (2021). A Robust Attentional Framework for License Plate Recognition in the Wild. IEEE Transactions on Intelligent Transportation Systems, 22(11), 6967–6976. https://doi.org/10.1109/TITS.2020.30), 6967–6976. https://doi.org/10.1109/TITS.2020.3000072

Appendices

Recognition/views.py

```
import cv2
import numpy as np
from time import sleep
from django.http import StreamingHttpResponse
from django.shortcuts import render
from ultralytics import YOLO

from recognition.models import PlateRecognition, MotorControlLog
from border.models import Vehicle, BorderCheck

from util import read_license_plate
import RPi.GPIO as GPIO
import threading
from collections import Counter


# Replace with the IP address of your phone stream
url = 'http://192.168.0.102:4747/video'


# Initialize GPIO for servo motor and IR sensor
SERVO_PIN = 18  # Define your servo motor pin
IR_SENSOR_PIN = 23  # Define your IR sensor pin

GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
```

```python
GPIO.setup(SERVO_PIN, GPIO.OUT)
GPIO.setup(IR_SENSOR_PIN, GPIO.IN)  # Set up IR sensor as input

servo = GPIO.PWM(SERVO_PIN, 50)  # PWM at 50Hz
servo.start(0)

# Lock for thread-safe GPIO operations
gpio_lock = threading.Lock()

def set_angle(angle: int) -> None:
    """
    Set the angle of the servo motor.
    :param angle: Angle in degrees (0 to 180).
    """
    if 0 <= angle <= 180:
        duty = angle / 18 + 2
        with gpio_lock:
            servo.ChangeDutyCycle(duty)
        sleep(0.2)  # Reduced sleep time for faster operation
        with gpio_lock:
            servo.ChangeDutyCycle(0)
    else:
        print("Invalid angle. Angle must be between 0 and 180.")

def open_gate():
    """Function to actuate the servo motor and open the gate."""
    threading.Thread(target=_open_gate_thread).start()
```

```
def _open_gate_thread():
    set_angle(180)
    sleep(5)  # Reduced sleep time
    set_angle(80)



# Load YOLO model once at startup
license_plate_detector = YOLO('license_plate_detector.pt')  # Ensure the model is optimized


def is_ir_sensor_triggered():
    """
    Check if the IR sensor is triggered (obstacle detected).
    The function returns True when the IR sensor output is LOW,
    indicating an obstacle is present.
    """
    return GPIO.input(IR_SENSOR_PIN) == GPIO.LOW



def generate_video(video_path):
    """Stream video from the file and perform detection."""
    frame_number = 0
    frame_skip = 2  # Process every 3rd frame for speed
    plate_list = []
    plate_limit = 0

    # Open the video file
    cap = cv2.VideoCapture(video_path)

    # Check if video opened successfully
```

```
if not cap.isOpened():
    print(f"Error opening video file: {video_path}")
    return

try:
    while True:
        ret, frame = cap.read()
        if not ret:
            print("End of video file or error reading frame.")
            break

        if frame_number % (frame_skip + 1) == 0:
            if is_ir_sensor_triggered():
                # Resize frame for faster processing
                resized_frame = cv2.resize(frame, (640, 480))

                # Convert to RGB as YOLO expects RGB images
                image = cv2.cvtColor(resized_frame, cv2.COLOR_BGR2RGB)

                # DETECT LICENSE PLATES
                results = license_plate_detector(image, verbose=False)
                license_plates = results[0]

                for license_plate in license_plates.boxes.data.tolist():
                    x1, y1, x2, y2, score, class_id = license_plate

                    # Ensure coordinates are within frame boundaries
                    x1, y1, x2, y2 = map(int, [x1, y1, x2, y2])
                    x1 = max(x1, 0)
```

```
y1 = max(y1, 0)
x2 = min(x2, resized_frame.shape[1])
y2 = min(y2, resized_frame.shape[0])

# Crop and preprocess the license plate region
license_plate_crop = resized_frame[y1:y2, x1:x2]
license_plate_crop_gray = cv2.cvtColor(license_plate_crop, cv2.COLOR_BGR2GRAY)
_, license_plate_crop_thresh = cv2.threshold(
   license_plate_crop_gray, 64, 255, cv2.THRESH_BINARY_INV
)

# Read license plate number
license_plate_text, license_plate_text_score =
read_license_plate(license_plate_crop_thresh)

print(f'License plate: {license_plate_text} and score: {license_plate_text_score}')

# Checking and processing license plate
if license_plate_text != None and license_plate_text_score != None:
   if len(license_plate_text) == 7 and license_plate_text_score >= 0.3 and plate_limit <= 5:
      plate_list.append(license_plate_text)
      plate_limit += 1

   if plate_limit > 5:
      # Collecting characters from each position across all detected plates
      characters_by_position = zip(*plate_list)

      # Finding the most common character in each position
```

```
                final_chars = [Counter(chars).most_common(1)[0][0] for chars in
characters_by_position]

                # Joining the most common characters to form the probable license plate
                probable_plate = ''.join(final_chars)

                print(f'probable plate: {probable_plate}')

                if probable_plate:
                  # Check if the vehicle is registered\

                  vehicle =
Vehicle.objects.filter(license_plates__license_plate_number=probable_plate).first()

                  if vehicle:
                    vehicle =
Vehicle.objects.get(license_plates__license_plate_number=probable_plate)

                    # Check if the vehicle is approved for travel in BorderCheck
                    border_check = BorderCheck.objects.filter(vehicle=vehicle).first()

                    if border_check and border_check.is_approved:
                      # Open the gate automatically if approved
                      # Log motor control action
                      MotorControlLog.objects.create(vehicle=vehicle, action="Gate Opened")
                      open_gate()
                      print('Gate opened: Vehicle approved for travel')
                      # Log motor control action
                      MotorControlLog.objects.create(vehicle=vehicle, action="Gate Closed")
```

```
            else:
                # Redirect to vehicle creation page if not approved
                print(f"Vehicle not approved for travel: Redirecting to vehicle creation page")

        else:
            # Vehicle not found in the database, redirect to vehicle creation page
            print(f"Vehicle not found: Redirecting to vehicle creation page")

        plate_limit = 0
        plate_list = []




    # Optionally, draw bounding box and text on the frame
    cv2.rectangle(resized_frame, (x1, y1), (x2, y2), (0, 255, 0), 2)
    cv2.putText(
        resized_frame, license_plate_text, (x1, y1 - 10),
        cv2.FONT_HERSHEY_SIMPLEX, 0.9, (36,255,12), 2
    )

  # Encode the frame to JPEG
  _, jpeg = cv2.imencode('.jpg', cv2.cvtColor(resized_frame, cv2.COLOR_RGB2BGR))
  frame_bytes = jpeg.tobytes()

  yield (b'--frame\r\n'
         b'Content-Type: image/jpeg\r\n\r\n' + frame_bytes + b'\r\n\r\n')

frame_number += 1
```

```
    finally:
        # Release the video capture object and clean up resources
        cap.release()
        cv2.destroyAllWindows()

def video_feed(request):
    """View to stream the video."""
    video_path = url  # Use your video file path here

    try:
        # Stream generator that yields the frames
        stream_generator = generate_video(video_path)

        # Directly return StreamingHttpResponse to stream the video
        return StreamingHttpResponse(stream_generator, content_type='multipart/x-mixed-replace;
boundary=frame')

    except Exception as e:
        print(f"Error: {e}")
        # Handle errors, potentially redirect if the error indicates a need for redirect

        return redirect('border:vehicle-create')

def home(request):
    """Home view to show the video feed."""
    return render(request, 'recognition/home.html')
```

https://github.com/bonheurNE07/plate_recognition.git