

# LICENSE PLATE DETECTION APPLICATION

**Mohammed Sraj GSR/5161/16**

Addis Ababa Institute of Technology

## ABSTRACT

The License Plate Detection Application is a web-based system developed using the Flask framework, designed to address misconduct in public transportation services. By empowering passengers to report illegal activities such as overcharging, the application aims to enhance the integrity and accountability of public transportation. Utilizing advanced digital image processing techniques and optical character recognition (OCR) through OpenCV and PyTesseract, the application processes images of vehicle plates submitted by users. It registers and tallies the number of complaints associated with each vehicle, enabling authorities to identify and take action against repeat offenders. Developed on a laptop with an Intel Core i5 processor and 16GB RAM, the system ensures efficient performance and scalability. The project incorporates a robust architecture featuring a user-friendly frontend, a scalable backend, and a secure database, providing reliable reporting and actionable insights for authorities. This application represents a significant step towards improving public transportation services by leveraging modern technologies to ensure real-time, accurate, and efficient reporting of misconduct..

## 1 INTRODUCTION

The License Plate Detection Application is a sophisticated web-based system designed to empower users to report any misuse of public transportation services. Public transportation is a critical infrastructure in urban areas, and maintaining its integrity is essential for ensuring public trust and safety. This application aims to address the issue of misconduct by service providers, which can range from overcharging passengers to more severe illegal activities. By providing an easy-to-use platform, passengers can quickly report such incidents, ensuring that they are documented and addressed by the appropriate authorities.

## 2 PROBLEM STATEMENT

Public transportation users often face issues such as overcharging, unprofessional behavior, and other forms of misconduct from service providers. These issues can lead to a lack of trust in the public transportation system and can discourage its use. Traditional methods of reporting these incidents are often cumbersome and ineffective. There is a clear need for a streamlined, efficient system that allows passengers to report such incidents in real-time, providing authorities with the necessary information to take corrective action.

## 3 METHODOLOGY

The License Plate Detection Application employs a combination of advanced digital image processing techniques and optical character recognition (OCR) to accurately detect and read vehicle plates from user-submitted images. The Flask web framework is utilized to develop the application, ensuring a robust and scalable backend. Version control is managed with Git to facilitate collaboration and maintain a history of changes.

## 4 PROPOSED SOLUTION

The proposed solution involves a user-friendly web-based application where passengers can upload images of vehicles involved in misconduct. The system processes these images to detect the license plate and extract its characters using PyTesseract for OCR. Once the plate is detected and the characters are extracted, the application registers the report in a database. The system then counts the number of complaints associated with each vehicle, allowing authorities to identify repeat offenders and take appropriate action.

## 5 SYSTEM ARCHITECTURE

**Frontend:** The frontend is designed to be intuitive and user-friendly, allowing users to easily upload images and view the status of their reports. It is built using modern web technologies to ensure responsiveness and accessibility across different devices.

**Backend:** The backend processes the uploaded images using OpenCV for image processing and PyTesseract for OCR. The Flask framework handles the routing, user authentication, and database interactions.

### Implementation Details

**Framework:** Flask is used for its simplicity and scalability, allowing for rapid development and easy deployment.

**Version Control:** Git is employed to manage the codebase, enabling collaboration among developers and ensuring a history of changes.

**OCR:** PyTesseract is used for its accuracy and ease of integration with OpenCV.

**Image Processing:** OpenCV is utilized for its robust image processing capabilities, including features for detecting and extracting license plates from images.

**Development Machine:** The development was carried out on a laptop with an Intel Core i5 processor and 16GB RAM, ensuring adequate performance for testing and development purposes.

### Results

The application successfully processes images to detect license plates and extract the characters with high accuracy. The system effectively registers and counts the number of reports, providing a reliable method for passengers to report misconduct. Initial testing shows that the application can handle a large volume of reports without significant performance degradation.

## 6 DEPLOYMENT

The system can be deployed on a web server accessible to the public, allowing users to report incidents in real-time. Deployment involves setting up the Flask application on a server, configuring the database, and ensuring that all necessary dependencies are installed. Authorities can access the data through a secure portal to review and act on the reports, with features for filtering and sorting reports based on various criteria.

## 7 CONCLUSION

The License Plate Detection Application provides a robust solution for reporting and addressing misconduct in public transportation services. By leveraging digital image processing and OCR technologies, the system ensures accurate and efficient reporting. This application not only improves the accountability of service providers but also enhances the overall trust in the public transportation system. Future work could involve integrating additional features such as real-time notifications for authorities and expanding the system to cover other forms of public transportation.

## 8 SOURCE CODE AND WEBAPP LINKS

[https://github.com/berito/cv\\_project\\_1.git](https://github.com/berito/cv_project_1.git)  
<http://188.166.153.41/>

## 9 REFERENCES

- pyteseract Documentation.
- Flask Documentation.
- Digital Ocean Deployment Guides.