AZCON

ENSORCELL DOCUMENTATION

Vehicle Security Registration Number Recognition Application

Submitted by: Udyan Sharma
EB07 - E20CSE006

Abstract

With the rising number of vehicles, there is an urgent need for an application that can easily recognize and authenticate a vehicle with its number plate. This project aims to create an unparalleled vehicle registration number recognition Java application that will scan a vehicle’s number plate, convert the number plate's image into text, query an API and then update the database at security gates, barriers, and checkpoints. These features will enrich the application's functionality and aid users in their day-to-day work.

The application uses multiple technologies to implement an efficient way for vehicle registration number recognition and verification. The application also allows its users to keep a record of the vehicle information in a MySQL database. The database can be manipulated using the sophisticated GUI provided. Also, database operations such as creation and deletion of records are also supported.

Introduction

The project is a vehicle number plate recognition Java application that uses the Tesseract OCR Engine working on a neural network system based on LSTM(Long Short-term Memory) for image to text conversion, MySQL and JDBC for the database operations and Java Swing for the graphical user interface. The project aims to tackle the vulnerability of security threats encountered at security checkpoints by providing information associated with a vehicle from its registration number plate without reaching out to the authorities for the same and using an API. The Java application has various uses at security checkpoints of facilities such as corporate buildings, DMRC parking lots, mall parkings and hotel entrances. The application allows the user to edit, update and delete the MySQL database using its Graphical User Interface. The application further allows the user to print the database table or save it as a PDF file.

Problem Statement

In the recent years, purchase volumes pertaining to the automobile industry have risen significantly.

The roads, parking lots of huge corporates and business facilities are facing the challenge of monitoring unknown vehicles at entry checkpoints more than ever now.

The unavailability of information of a vehicle's owner on demand and whether a vehicle has a genuine and authentic registration number plate has become a major vulnerability for security breaches as well as violation of traffic laws.

We aim to solve this problem by developing an application that allows the user to read the registration number plate of a vehicle and provide the user with various attributes of the vehicle. Hence, we are endeavouring to tackle this challenge by providing a solution to security agencies that will allow faster and easier recognition of vehicles at security checkpoints.

Methodology

We plan to use Java as the main programming language for the development of our application. Our methodology:

- Create a GUI based Java application using Java Swing.
- Add an image-to-text functionality to the application using the Tesseract OCR Engine.
- Create a database in MySQL and connect it to the Java application using the mysql connector and the JDBC API (Java Database Connector).
- Add create and delete entries feature to the database table.
- Add print feature to the database table.
- Fetch details like the vehicle owner's name, location, model, engine number, chassis number, insurance and vehicle type using a Pan India API updated regularly with data received from different RTOs of India.
- Store the user details in the database and display the entries.

The flow of the Java application

STEP 1

Image input

The user needs to input an image of a vehicle's number plate.

STEP 2

Image-to-text OCR Conversion

The application processes and converts the image to text using the Tesseract OCR Engine.

STEP 3

API Connection

The application connects to an RTO vehicle information API and fetches the vehicle information.

STEP 4

Database Management

The attributes of the vehicle received from the API connection are stored in the database. Add, Print and Delete options are supported.

STEP 5

Displaying Information

After the image of the vehicle's registration number plate is successfully processed, converted into text and after receiving the information from the API connection and storing it in the database, the necessary information about the vehicle is displayed on the user interface.

Features of the application

The ease of use and the availability of features makes the application ideal for daily use at security checkpoints.

IMAGE TO TEXT CONVERSION

Our application uses the latest Tesseract OCR Engine to convert the vehicle registration number plate image into text, which is stored as a string value.
The Tesseract OCR engine uses a neural network system based on LSTM.

GRAPHICAL USER INTERFACE

The application will allow the user to utilize its functionality through a desktop GUI application developed using Java Swing. Various Swing elements such as jFrames, jButtons, jPanels, jTable and jLabel etc have been used.

DATABASE ACCESS

The user will be able to access the database and either delete or print the previous entries made and logged after retrieving the data from the vehicle registration RTO API.
The user can view the database information using the graphical user interface of the application.

Solution Approach

Our approach to solving this problem is simple and efficient. We have created a sophisticated Java GUI application for the ease of the user and for faster and accurate vehicle recognition using an API updated with datasets of different RTOs across India. We have used different technologies like Java Swing, Tesseract Deep Learning OCR Engine, JDBC API and MySQL for the same.

Technologies Used

1. Java: Java has been used as the core programming language for this project. It is a high level object-oriented programming language.
2. Java Swing: The graphical user interface for the application has been developed using Java Swing. Java Swing is included in the Java Foundation Classes and is used to create Windows applications. It is written entirely in Java on the top of AWT API.
3. MySQL Database: MySQL has been used as the database backend for storing the data and information of the vehicles, scanned and queried with the RTO API.
4. Web API: A web API has been used to fetch the vehicle and its owner's information. The application calls the API to retrieve the required details.
5. WordPress: WordPress has been used to create a website for the project. The application can be downloaded from the website.

Outcomes

The application was successfully developed with a number of good and impactful outcomes.

- An intuitive and easy to use GUI was developed using Java Swing and its elements such as JButton, JTextField, JFileChooser, JDialog, JTable and JFrame.
- The application took an image file as input, with different extensions chosen by the user.
- The application processed the image using the Tesseract OCR engine working on a neural network system based on LSTM(Long Short-term Memory) and converted it into a text string with an average time of 3.8 seconds.
- The vehicle registration number was successfully queried with an all India RTO registration number API and the details of the vehicle and its owner were fetched and displayed.
- The database was accessed and updated using different buttons and the user was able to print and save the database.
- The application solved the problem of vehicle verification at security checkpoints by providing the details of the vehicle and its owner and allowing the user to store them in a database which can be manipulated and printed.
- The users can benefit from the easy interface of the application.

Conclusion

Ensorcell has been developed to take a step forward in the image processing and vehicle management domain. By releasing this project in the market, we aim to simplify and sophisticate the process of vehicle security verification and management using Java Swing. We also aim to make it simpler for security agencies to be able to punch in vehicle numbers into their security logs without much effort by using our application. The project has several applications in the security domain.

Future Work

In future, the application can be integrated with live IP security surveillance cameras to provide an enriching experience which can be used to recognize the vehicles entering or leaving a facility like a car parking, an office building or a commercial establishment. This will reduce the costs by eliminating the need to hire security personnel to keep track of the vehicles entering and exiting a facility. The plans for the coming years would be to evolve the project into an android application and add features to it constantly. The Android application may be released on the Google Play Store in future. Also, the Java Swing and the Android applicatoin will be integrated to a web application with a professional dashboard for linking multiple security checkpoints and managing their data from a single dashboard, thus creating a unified interface.

References

1. https://docs.oracle.com/en/java/
2. https://docs.oracle.com/javase/8/docs/technotes/guides/jdbc/
3. https://tesseract-ocr.github.io/
4. http://www.carregistrationapi.in/data/doc.aspx
5. https://docs.oracle.com/javase/7/docs/api/javax/swing/package-summary.html