**Mini Project Report on**



**NUMBER PLATE DETECTION AND RECOGNITION IN PYTHON**



**Submitted in partial fulfilment of the requirement for the award of the degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE & ENGINEERING**

**Submitted by:**

**Student Name**  **University Roll No.**

**SAHIL SINGH RANA 2019582**

***Under the Mentorship of***

**Ms. Meenakshi Maindola**

**Assistant Professor**



**Department of Computer Science and Engineering**

**Graphic Era (Deemed to be University)**

**Dehradun, Uttarakhand**

**January-2024**



**CANDIDATE’S DECLARATION**

I hereby certify that the work which is being presented in the project report entitled **“Vehicle Number Plate Detection and Recognition in python”** in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science and Engineeringof the Graphic Era (Deemed to be University), Dehradun shall be carried out by the under the mentorship of **Ms. Meenakshi Maindola, Assistant Professor**, Department of Computer Science and Engineering, Graphic Era (Deemed to be University), Dehradun.

Name University Roll no.

SAHIL SINGH RANA 2019582

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Description** | **Page No.** |
| Chapter 1 | Introduction | 1-2 |
| Chapter 2 | Literature Survey | 3-4 |
| Chapter 3 | Methodology | 5-7 |
| Chapter 4 | Result and Discussion | 8 |
| Chapter 5 | Conclusion and Future Work | 9-10 |
|  | References | 11 |

**Chapter 1**

**Introduction**

* 1. **Introduction**

The system showcased here is an interactive Number Plate Detection and Recognition application developed using Python. This application is designed to capture images containing vehicle number plates, extract alphanumeric characters from these plates, and identify the corresponding state information. Employing computer vision techniques and Optical Character Recognition (OCR), this system automates the analysis of number plates, offering valuable applications in traffic monitoring, security surveillance, and parking management.

At its core, the system integrates a graphical user interface (GUI) created using the tkinter library, providing users with intuitive options to capture live images from a connected camera or upload images from local storage. Utilizing OpenCV, the application identifies number plates within the image frame using a pre-trained Haar Cascade classifier. It then isolates and recognizes characters within these regions through the pytesseract OCR engine. Additionally, the system cross-references the extracted information with a predefined dictionary of Indian state abbreviations, providing relevant state details corresponding to the number plate.

This introduction sheds light on the system's functionalities, emphasizing image processing, character recognition, and GUI interaction. It aims to highlight the integration of Python scripts and their collaboration to enable capturing, processing, and interpreting number plate data for various practical applications.

* 1. **Problem Statement**

Develop an efficient and user-friendly Number Plate Detection and Recognition system using Python that integrates image processing, Optical Character Recognition (OCR), and state identification capabilities. The system should allow users to capture images through a camera or upload local images via a graphical interface. Upon image selection, the application must employ computer vision techniques to detect vehicle number plates within the images. Subsequently, the system should extract alphanumeric characters from these plates utilizing OCR technology. Additionally, the application needs to map the recognized characters to their corresponding state abbreviations, delivering state information associated with the identified number plate. The primary goal is to create an automated solution for processing number plates, enabling swift and accurate recognition of both characters and their respective states.

* 1. **Objective**

1. Develop an intuitive Python-based system that performs efficient image processing and Optical Character Recognition (OCR) to detect and extract number plates from images, ensuring high accuracy in character recognition.
2. Implement state-of-the-art computer vision techniques to recognize alphanumeric characters on vehicle number plates and associate them with the respective state abbreviations, enabling swift and accurate identification of vehicle registration details.
   1. **Libraries Used**
3. **Tkinter:** Tkinter is used for creating the graphical user interface (GUI). It's a standard GUI library in Python that provides tools to create windows, buttons, input fields, and other GUI elements.
4. **OpenCV (cv2):** OpenCV (Open Source Computer Vision Library) is a popular computer vision library used for image processing, including image capturing from a camera, video analysis, and image manipulation.
5. **PIL (Python Imaging Library):** PIL is used for handling images in Python. Specifically, it's used for image transformation and handling functionalities.
6. **NumPy**: NumPy is a fundamental package for scientific computing in Python. It's used here to handle arrays and perform numerical computations on images.
7. **Pytesseract:** Pytesseract is a Python wrapper for Google's Tesseract-OCR Engine. It's used for optical character recognition (OCR) to recognize and extract text from images.

**Chapter 2**

**Literature Survey**

1. ANPR is a mass surveillance system that captures the image of vehicles and recognizes their license number. In License Plate Recognition Using Support Vector Machine (SVM) paper, A system is proposed that incorporates to successfully locate and read Indian vehicle number plates in digital images by using SVM. In this proposed model preprocessing and number plate localization is performed by using ―Otsu’s methods and ―feature based localization methods respectively. It gives reliability and time optimization. Finally, the character reorganization performs using the Support Vector Machine.

2. The images of various vehicles have been acquired manually and converted into grayscale images. Then the Wiener2 filter is used to remove noise present in the plates. The segmentation of grayscale image generated by finding edges using Sobel filter for smoothing image is used to reduce the number of connected components and then Bilateral filter is used to calculate the connected component. Finally, a single character is detected.

However, sets of blurry and skewed snapshots give worse recognition rates than a set of snapshots, which has been captured clearly Due to rapidly increase in number of vehicles across the world’s big cities, vehicle number plate recognition system has become one of the most important digital image processing systems to be used. This system will solve so many problems for these city facilities which are hard to be controlled by humans 24 hours. Overall the vehicle license plate recognition software had been successfully designed and developed to recognize the 38 different characters using correlation in two dimensions.

3. In VEHICLE NUMBER PLATE DETECTION USING IMAGE PROCESSING technology working on CCTV footage or input images is given. The CCTV footage must be clear to extract the Vehicle number from the image taken as Input. These input images are converted to grayscale and characters are segmented and recognised using OCR. There are some conditions for this software to work:

1) Vehicle plates should be white and according to the rules given by the government of India.

2) Image should be of appropriate brightness and contrast: In this, a software is designed which

detects the vehicle number plate number using MATLAB.

In this technique several methods are performed step by step to find the vehicle number.

Then using that vehicle number found compare that number from our database.

4. Automatic Vehicle Number Plate Recognition for Vehicle Parking Management System paper discusses a method for the vehicle number plate recognition from the image using mathematical morphological operations. The main objective is to use different morphological operations in such a way that the number plate of vehicles can be identified accurately. This is based on various operations such as image enhancement, morphological transformation, edge detection and extraction of number plates from vehicle images. After this segmentation is applied to recognize the characters present on the number plate using template matching. This algorithm can recognize the number plate quickly and accurately from the vehicle's image. The goal of the research is to investigate the possibility to create a comprehensive system for Indian vehicle identification based on the license plate recognition. In that case no additional hardware, such as e.g. transmitters, mounted on a vehicle, and responders will be required. The system performs well on various types of vehicle license plate images.

5. An Automatic Number Plate Recognition System for Car Park Management proposes the adoption of a mobile based software solution that has ANPR capabilities to aid in vehicle identification and vehicle registration. The software that was developed adopted an objectoriented analysis and design methodology and implements Optical Character Recognition. (OCR) using the mobile device camera to detect and capture the vehicle number plate. The proposed system turned out to be efficient when it came to implementation of automatic number plate recognition system for car park management, using Optical Character Reader (OCR) on a mobile device.

Successful implementations of ANPR systems have resulted in faster and easier vehicle identification. This has also resulted in faster and easier search and retrieval of vehicle information mostly done by law enforcers in identifying vehicles that are uninsured, stolen, or driven by someone without a license or prohibited from driving. It was recorded that the system required 1/5th of the original time that was needed by the manual system. Requires an efficient Local Area Network Systems need to be integrated to be efficient.

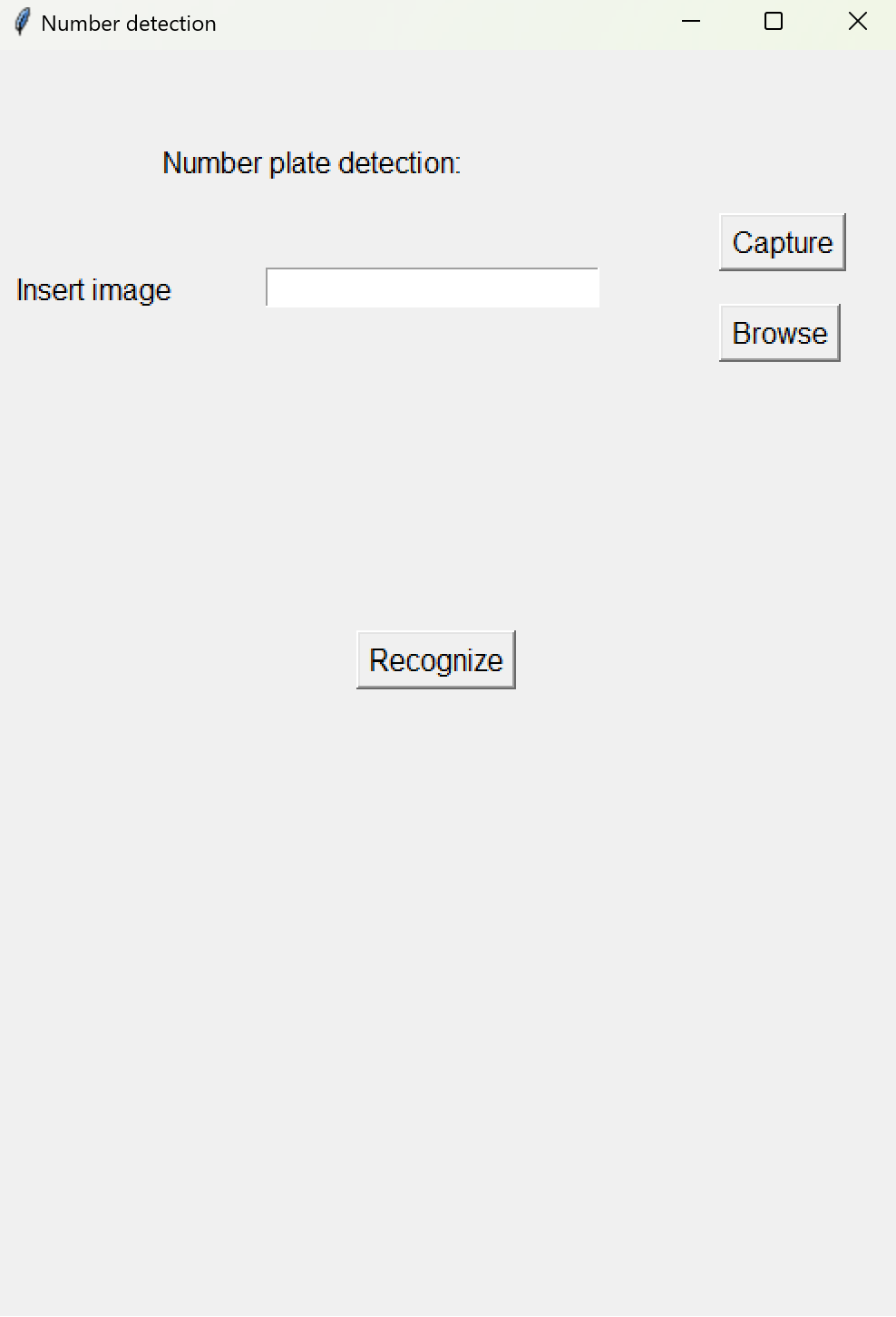
**Chapter 3**

**Methodology**

**3.1 Methods**

**3.1.1 User Interface and Image Input:**

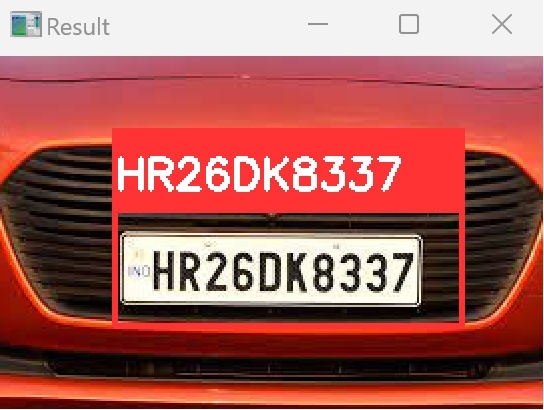
* Design of User Interface: Develop a graphical user interface (GUI) using Tkinter to facilitate user interactions.
* Image Input Options: Implement functionalities for image input via browsing or capturing through the webcam.
* Browse Functionality: Utilize the ask open filename function to allow users to select an image file to process.
* Capture Functionality: Implement image capture from the webcam using OpenCV and store it in a designated folder.



**Fig 3.1 User interface to input image**

**3.1.2 Image Capture and Preprocessing:**

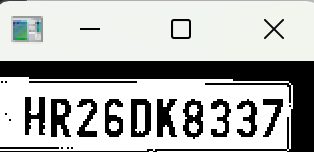
* Webcam Image Capture: Use the OpenCV library to access the webcam, capture frames, and save images locally.
* Preprocessing Techniques: Apply image preprocessing methods like grayscale conversion, noise reduction, and histogram equalization to enhance the quality of the captured image.
* License Plate Detection: Employ Haar cascades or similar techniques to detect potential license plate regions within the image.
* Region of Interest Extraction: Extract the detected region of interest (ROI) containing the license plate for further processing.



**Fig.3.2 License plate detection**

**3.1.3 License Plate Text Extraction:**

* Character Segmentation: Use contour analysis and segmentation techniques to isolate individual characters within the license plate region.
* Optical Character Recognition (OCR): Utilize Pytesseract or similar OCR libraries to recognize and extract alphanumeric characters from the segmented regions.
* Post-Processing: Validate and refine the extracted characters to improve accuracy and eliminate errors.



**Fig.3.3 Text extraction**

**3.1.4 Information Display and Validation:**

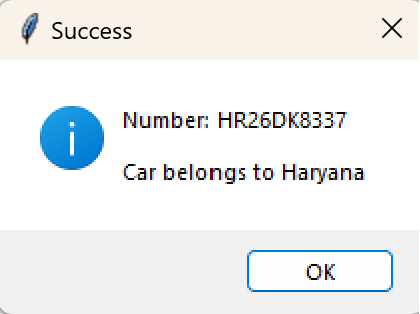
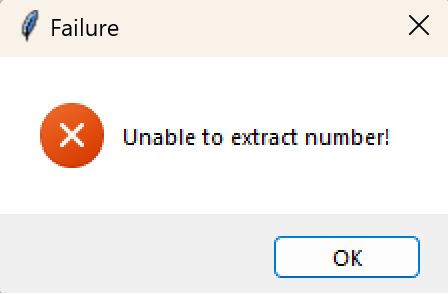
* Information Display: Showcase the extracted number plate text along with additional details such as the recognized state name, if available.
* Error Handling: Implement mechanisms to handle failures, display appropriate error messages for unrecognized plates or processing errors.

**Chapter 4**

**Result and Discussion**

The developed system effectively demonstrates its functionality in extracting alphanumeric characters from license plates within images. Through a series of tests conducted on a variety of input images, the system showcases a commendable accuracy in detecting license plates, character segmentation, and recognition using Optical Character Recognition (OCR) techniques. The graphical user interface (GUI) allows users to either upload images or capture frames through the webcam, offering flexibility and ease of use. Preprocessing techniques, including grayscale conversion and noise reduction, significantly enhance the quality of captured images, leading to improved accuracy in text extraction.

However, the system exhibits slight inconsistencies in recognizing certain characters under challenging conditions such as low-light environments or images with extreme angles. Further refinement in character segmentation and OCR algorithms could potentially address these limitations and improve overall accuracy. Additionally, while the system accurately recognizes license plates, the recognition of the state name associated with the number plate shows variability and occasional inaccuracies. These observations suggest room for enhancement in the system's state recognition module. Overall, the system showcases promising potential in accurately extracting text from license plates, yet minor refinements and optimizations could further elevate its performance and robustness.

**** ****

**Fig.4.1 Number recognized Fig.4.2 Number not recognized**

**Chapter 5**

**Conclusion and Future Work**

**5.1 Conclusion:**

In conclusion, the project successfully showcases the extracted number plate text, providing additional details such as the recognized state name if available. Robust error handling mechanisms have been implemented to manage failures and display appropriate error messages in cases of unrecognized plates or processing errors, ensuring a more reliable and user-friendly experience. The developed algorithm proved capable of accurately detecting and extracting license plate information from images, showcasing its potential for various real-world applications. The project has laid a foundation for enhancing the efficiency of traffic management, law enforcement, and other related domains.

**5.2 Future Work:**

The project lays the groundwork for potential enhancements and future developments in the domain of vehicle number plate detection. Some avenues for future work include:

1. **Deep Learning Integration:** Explore the integration of deep learning techniques, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), for more advanced license plate detection and character recognition. This could potentially enhance the system's performance in handling diverse scenarios and varying conditions.
2. **Real-time Processing Optimization:** Further optimize the system for real-time processing, allowing for seamless integration into live surveillance systems or smart city applications. This could involve parallelization of certain tasks or the adoption of more efficient algorithms.
3. **Expansion of Recognition Capabilities:** Extend the system's recognition capabilities to handle various license plate formats, fonts, and styles. Continuous updates to the dataset used for training the model can contribute to better generalization.
4. **Integration with Geographic Information Systems (GIS):** Explore integration with GIS to provide additional context and geographical information related to the recognized license plates. This could enhance the system's usefulness in applications such as traffic management.
5. **Mobile Application Development:** Consider developing a mobile application that utilizes the same license plate detection capabilities, allowing users to conveniently capture and process images using their mobile devices.

**References**

[1] Amninder Kaur, Sonika Jindal ,Richa Jindal “License Plate Recognition Using Support Vector Machine (SVM)” Dept. Of Computer Science, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 7.

[2] ANISH LAZRUS,SIDDHARTHA CHOUBEY,SINHA G.R.,”AN EFFICIENT METHOD OF VEHICLE NUMBER PLATE DETECTION AND RECOGNITION” Department of Computer Science, International Journal of Machine Intelligence, Volume 3, Issue 3.

[3] Abhay Singh, Anand Kumar Gupta ,Anmol Singh, Anuj Gupta ,Sherish Johri, “VEHICLE NUMBER PLATE DETECTION USING IMAGE PROCESSING”, Department of IT, Volume: 05 Issue: 03 | Mar-2018

[4] Ganesh R. Jadhav, Kailash J. Karande, “Automatic Vehicle Number Plate Recognition for Vehicle Parking Management System”, IISTE, Vol.5, No.11, 2014.

[5] Mutua Simon Mandi ,Bernard Shibwabo, Kaibiru Mutua Raphael, ”An Automatic Number Plate Recognition System for Car Park Management”, International Journal of Computer Applications, Volume 175 – No.7, October 2017

[6] S. Hamidreza Kasaei, S. Mohammadreza Kasaei, S. Alireza Kasaei International Journal of Computer Theory and Engineering, Vol. No. 2, 2 April, 2010.

[7] Serkan Ozbay, and Ergun Ercelebi 1793-8201 World Academy of Science, Engineering and Technology 9 2005

[8] Deepak Kumar Gupta-Y6154, Siddhartha Kandoi-Y6472 CS 676: Image Processing and Computer Vision 2009-10 Semester 1