



Automatic License Number Plate Recognition System

G. Bharath Kumar, Ch. Jaya Rama Krishna, K. Jitha Priya, Ch. Naga Vamsi Chandh

Department of CSE, Chalapathi Institute of Engineering & Technology, Guntur, India.

To Cite this Article

G. Bharath Kumar, Ch. Jaya Rama Krishna, K. Jitha Priya and Ch. Naga Vamsi Chandh. Automatic License Number Plate Recognition System. International Journal for Modern Trends in Science and Technology 2022, 8(S08), pp. 63-67. <https://doi.org/10.46501/IJMTST08S0812>

Article Info

Received: 26 May 2022; Accepted: 24 June 2022; Published: 28 June 2022.

ABSTRACT

Automatic License Number Plate Recognition System is a real time embedded system which automatically recognizes the license plate of vehicles. There are many applications ranging from complex security systems to common areas and from parking admission to urban traffic control. Automatic license plate recognition (ALPR) has complex characteristics due to diverse effects such as of light and speed. Most of the ALPR systems are built using proprietary tools like MATLAB. Here, we are using an alternative method of implementing ALPR systems using Free Software including Python and the Open Computer Vision Library.

1. INTRODUCTION

1.1 Introduction

The scientific world is deploying research in intelligent transportation systems which have a significant impact on peoples' lives. Automatic License Plate Recognition (ALPR) is a computer vision technology to extract the license number of vehicles from images. It is an embedded system which has numerous applications and challenges. Typical ALPR systems are implemented using proprietary technologies and hence are costly. This closed approach also prevents further research and development of the system. With the rise of free and opensource technologies the computing world is lifted to new heights. People from different communities interact in a multi-cultural environment to develop solutions for mans never ending problems. One of the notable contributions of the open-source community to the scientific world is Python. Intel's researches in Computer Vision bore the fruit called Open Computer Vision (OpenCV) library, which can support computer vision development.

1.2 Background

Automatic license plate recognition (ALPR) plays an important role in numerous applications such as unattended parking lots security control of restricted areas traffic law enforcement congestion pricing and automatic toll collection. Due to different working environments, LPR techniques vary from application to application. Most previous works have in some way restricted their working conditions, such as limiting them to indoor scenes, stationary backgrounds fixed illumination, prescribed driveways limited vehicle speeds or designated ranges of the distance between camera and vehicle.

1.3 OpenCV

OpenCV (Open-Source Computer Vision Library) is an open-source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code.

1.4 Motivation

The main purpose of this project is to detect a license plate from an image captured by a camera. An efficient algorithm is developed to detect a license plate in various luminance conditions. This algorithm extracts the license plate data from an image and provides it as an input to the stage of Car License Plate Recognition. Extracted image of the number plate can be seen on monitor. The scope of this project is to detect the license plate from the given image and observe the output on monitor. This project can work as a base for future improvements in the field of image processing, especially in license plate extraction and plate number recognition.

1.5 Objective

The main objective of this project is to recognize license number plates. In order to detect license number plates, we will use OpenCV to identify number plates and EasyOcr or python pytesseract to extract characters and digits from the number plates.

1.6 Description

License Plate Recognition is an image-processing technology used to identify vehicles by their license plates. This technology is used in various security and traffic applications. Below are the steps involved in License Plate Recognition are

- a) **License Plate Detection:** The first step is to detect the License plate from the car. We will use the contour option in OpenCV to detect for rectangular objects to find the number plate. The accuracy can be improved if we know the exact size, color and approximate location of the number plate. Normally the detection algorithm is trained based on the position of camera and type of number plate used in that particular country. This gets trickier if the image does not even have a car, in this case we will an additional step to detect the car and then the license plate.
- b) **Character Segmentation:** Once we have detected the License Plate, we have to crop it out and save it as a new image. Again, this can be done easily using OpenCV.
- c) **Character Recognition:** Now, the new image that we obtained in the previous step is sure to have some characters (Numbers/Alphabets) written on it. So, we can perform OCR (Optical Character Recognition) on it to detect the number.

2. LITERATURE SURVEY

2.1 LITERATURE SURVEY

Paridhi Swaroop, Neelam Sharma, "An Overview of Various Template Matching Methodologies in Image Processing", International Journal of Computer Applications (0975 – 8887) Volume 153 – No 10, November 2016.

The following paper gives a comparison about applications and methods where template matching is used. Template is primarily a sub-part of an object that is to be matched amongst entirely different objects. The techniques of template matching are flexible

and generally easy to make use of, that makes it one amongst the most famous strategies of object localization. Template matching is carried out in versatile fields like image processing, signal processing, video compression and pattern recognition. The following template matching techniques are used Naive Template Matching, Image Correlation Matching, Sum of Absolute Difference, sum of square difference.

Lucky Kodwani&Sukadev Meher "Automatic License Plate Recognition in Real Time Videos using Visual Surveillance Techniques "ISSN (PRINT): 2320 – 8945, Volume -1, Issue -6, 2013.

This paper presents full-featured vehicle detection, tracking and license plate recognition system. It consists of vehicle detection, license plate extraction and a character recognition module. Here, first foreground estimation is done by Gaussian mixture model, then a real time and robust method of license plate extraction based on block variance technique is proposed. License plate extraction is an important stage in license plate recognition for automated transport system. The extracted license plates are segmented into individual characters by using a region-based approach. The recognition scheme combines adaptive iterative thresholding with a template matching algorithm.

Riazul Islam, Kazi Fatima Sharif and Satyen Biswas, "Automatic Vehicle Number Plate Recognition Using Structured Elements", IEEE Conference on Systems, Process and Control December 2015, pp 44-48.

This research presents a prosperous method to identify vehicle number plates. The proposed technique is built on morphological operations based on different structuring elements in order to maximally exclude non-interested region and improve object area. This

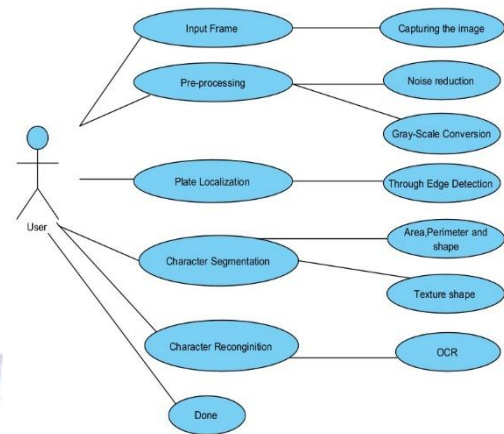
system has been experienced using a database of number plates and simulated results demonstrate major improvements as compared to other conventional systems.

Muhammad Tahir Qadri, Muhammad Asif "Automatic Number Plate Recognition System for Vehicle Identification using Optical Character Recognition" IEEE2009.

The objective is to design an efficient automatic authorized vehicle identification system by using the vehicle number plate. The system is implemented on the entrance for security control of a highly restricted area like military zones or area around top government offices e.g. Parliament, Supreme Court etc. The developed system first detects the vehicle and then captures the vehicle image. Vehicle number plate region is extracted using the image segmentation in an image. Optical character recognition technique is used for the character recognition. The resulting data is then used to compare with the records on a database so as to come up with the specific information like the vehicle's owner, place of registration, address, etc.

Aniruddh Puranic, Deepak K. T, Umadevi V "Vehicle Number Plate Recognition System: A Literature Review and Implementation using Template Matching" International Journal of Computer Applications (0975 – 8887) Volume 134 – No.1, January 2016.

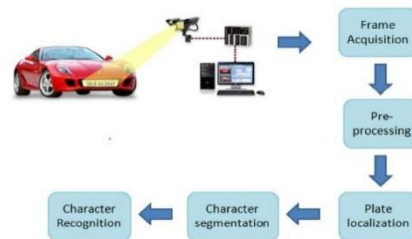
The growing affluence of urban India has made the ownership of vehicles a necessity. This has resulted in an unexpected civic problem - that of traffic control and vehicle identification. The Automatic Number Plate Recognition System (ANPR) plays an important role in addressing these issues as its application ranges from parking admission to monitoring urban traffic and to tracking automobile thefts. There are numerous ANPR systems available today which are based on different methodologies. In this paper, we attempt to review the various techniques and their usage. The ANPR system has been implemented using template Matching and its accuracy was found to be 80.8% for Indian number plates.



3. DESIGN AND ANALYSIS SYSTEM ARCHITECTURE

System architecture is the conceptual model that defines the structure, behaviour and views of a system. The below figure is an architectural design for the Automatic License Plate Recognition (ALPR) system. ALPR system is a system that reads and process the image that consists of vehicle

number plate as input and recognizes the number plate as output automatically.



SYSTEM ARCHITECTURE ALGORITHM

The Working of ALPR system consists of the following steps:

- Step 1: Capture the car images and store them in repository(i.e., dataset)
- Step 2: Apply Pre-processing techniques
- Step 3: Perform plate localization on the pre-processed from based on aspect ratio
- Step 4: Extract the individual character of the license plate
- Step 5: Perform character recognition on the segmented characters and display the result on the terminal

USE CASE DIAGRAM

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. Hence, when a system is analysed to gather its

functionalities, use cases are prepared and actors are identified.

4. IMPLEMENTATION

PLATFORM SELECTION

Anaconda-Jupyter Notebook

The Jupyter Notebook is an open-source web application that you can use to create and share documents that contain live code, equations, visualizations, and text. Jupyter Notebook is maintained by the people at Project Jupyter. Jupyter Notebooks are a spin-off project from the IPython project, which used to have an IPython Notebook project itself. The name, Jupyter, comes from the core supported programming languages that it supports: Julia, Python, and R. Jupyter ships with the IPython kernel, which allows you to write your programs in Python, but there are currently over 100 other kernels that you can also use.

Python

IDLE (short for integrated development environment or integrated development and learning environment) is an integrated development environment for Python, which has been bundled with the default implementation of the language since 1.5.2b1. According to the included README, its main features are:

- Multi-window text editor with syntax highlighting, autocompletion, smart indent and other.
- Python shell with syntax highlighting.
- Integrated debugger with stepping, persistent breakpoints, and call stack visibility.

PROGRAMMING LANGUAGE GIST

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse.

CODING STANDARDS

1. Naming Conventions

Module Names: – Short, lowercase names, without underscores. Example: myfile.py

Class Names: – CapWords convention. Example: MyClass

2. Organising imports

They should be always put at the top of the file, just after any module comments and document strings, and before module global and constants. Imports should be on separate lines.

3. Indentation and Line lengths

Indentations: – 2 spaces (no tabs!) – Avoid using more than five levels of indentation. Line length: – Maximum of 72 characters (never exceed 79 characters) – You can break a long line using “\”.

4. Break Lines

Leave one line between functions in a class. Extra blank lines may be used to separate groups of related functions. Blank lines may be omitted between a bunch of related one-liners. Use blank lines in functions, sparingly, to indicate logical sections.

5. White Space

Multiple statements on the same line are discouraged.

WRONG: if foo == 'blah': doBlahThing()

CORRECT: if foo == 'blah': doBlahThing()

6. Comments

Block Comments: They are indented to the same level as the code they apply to. Each line of a block comment starts with a # and a single space. Paragraphs inside a block comment are separated by a line containing a single #.

7. Document Strings

Write document strings for all public modules, functions, classes, and methods. Document strings are not necessary for non-public methods, but you should have a comment that describes what the method does. This comment should appear after the "def" line. Insert a blank line before and after all document strings that document a class.

Class: The document string for a class should summarize its behaviour and list the public methods and instance variables. If the class is intended to be subclassed, and has an additional interface for subclasses, this interface should be listed separately. If a class subclasses another class and its behaviour is mostly inherited from that class, its document string should mention this and summarize

the differences. The class constructor should be documented in the document string for its `__init__` method.

Function or method: The document string should summarize its behaviour and document its arguments, return value, side effects, exceptions raised, and restrictions on when it can be called. Optional arguments should be indicated. Use the verb "override" to indicate that a subclass method replaces a super class method and does not call the super class method; use the verb "extend" to indicate that a subclass method calls the super class method. The document string should contain a summary line, followed by a blank line, followed by a more elaborate description.

5. CONCLUSION

ALPR applications are becoming increasingly complex in Indian context with the phenomenal exponential growth in car, two-wheeler and auto Industries. ALPR applications like automatic toll collection, automatic charging system in parking spaces, management vehicles in parking spaces, and traffic monitoring, etc., have posed new research tasks in ALPR with newer dimensions. We have developed a software for automatic license plate recognition by taking inputs from car images. Character segmentation has been implemented on extracted number plates. Finally, segmented characters are recognized by using optical character recognition (OCR) method.

LIMITATIONS

1. Camera should be of good quality. Otherwise, correct text from image would not be extracted properly.
2. There should be proper lighting.
3. This system does not respond properly under different illumination conditions.
4. Although accuracy is high, other environmental factors leads to low computational results.

FUTURE ENHANCEMENTS

The implementation of the proposed system can be extended for the recognition of number plates of multiple vehicles in a single image frame. User friendly android applications can be developed for traffic surveillance management systems. Also, character recognition can be done using various deep learning algorithms as they yield more accuracy. GPUs can be used to achieve more performance in terms of computational time.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

- [1] Paridhi Swaroop, Neelam Sharma, "An Overview of Various Template Matching Methodologies in Image Processing", *International Journal of Computer Applications* (0975 – 8887) Volume 153 – No 10, November 2016.
- [2] Lucky Kodwani&Sukadev Meher "Automatic License Plate Recognition in Real Time.
- [3] Videos using Visual Surveillance Techniques ",ISSN (PRINT): 2320 – 8945, Volume -1, Issue -6, 2013.
- [4] Riazul Islam, Kazi Fatima Sharif and Satyen Biswas, "Automatic Vehicle Number Plate Recognition Using Structured Elements", *IEEE Conference on Systems, Process and Control* December 2015, pp 44-48.
- [5] Muhammad Tahir Qadri, Muhammad Asif "Automatic Number Plate Recognition System for Vehicle Identification using Optical Character Recognition" *IEEE* 2009.
- [6] Aniruddh Puranic, Deepak K. T, Umadevi V "Vehicle Number Plate Recognition System: A Literature Review and Implementation using Template Matching" *International Journal of Computer Applications* (0975 – 8887) Volume 134 – No.1, January 2016 .
- [7] Byung-Gil Han, Jong Taek Lee, Kil-Taek Lim, and Yunsu Chung "Real-Time License.
- [8] Plate Detection in High Resolution Videos Using Fastest Available Cascade Classifier and Core Patterns" *ETRI Journal*, Volume 37, Number 2, April 2015.
- [9] Nighat Naaz Ansari, Ajay Kumar Singh "License Number Plate Recognition using
- [10] Template Matching" *International Journal of Computer Trends and Technology (IJCTT) – Volume 35*
- [11] Number 4- May 2016, M. M. Shidore, S. P. Narote," Number Plate.
- [12] Recognition for Indian Vehicles" *IJCSNS International Journal of Computer Science and Network Security*, VOL.11 No.2, Feb. 2011