



**Image Processing and**

**Computer Vision Project**

**Problem Statement : Vehicle Number Plate Detection**

**Submitted to :**

Dr. Vikas Tripathi

**Submitted By :**

Rahul Kumar Singh, 75

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Description** | **Page No.** |
| Chapter 1 | Introduction | 3 |
| Chapter 2 | Literature Survey | 5 |
| Chapter 3 | Methodology | 7 |
| Chapter 4 | Result and Discussion | 12 |
| Chapter 5 | Conclusion and Future Work | 13 |
|  | References |  |

**Chapter 1**

**Introduction**

Number plate detection and recognition is a crucial task in the field of traffic management and vehicle security. It involves identifying and extracting the number plate of a vehicle from an image or video, and then recognizing the characters present on the number plate. Accurate and efficient detection and recognition of number plates can help in various applications such as automatic toll collection, traffic monitoring, and vehicle tracking.

There are several approaches to solve this problem, including traditional image processing techniques and modern deep learning approaches. In this report, we will focus on the latter approach, which has proven to be more accurate and efficient. The process of vehicle number plate detection and recognition using Python can be divided into two main steps: Number plate detection: In this step, we need to detect the number plate in the image.

This can be done using an object detection algorithm, such as YOLO (You Only Look Once) or SSD (Single Shot Detector). These algorithms are trained to identify objects in an image and draw a bounding box around them. We can use a pre-trained model on a dataset of number plates, or train our own model using a dataset of images containing number plates.

Number plate recognition: Once the number plate is detected and a bounding box is drawn around it, the next step is to recognize the characters present on the number plate. This can be done using a character recognition algorithm, such as Tesseract. Tesseract is an open-source OCR (Optical Character Recognition) engine that can recognize characters from images and convert them into machine-readable text.

**PROBLEM STATEMENT**

One of the main challenges in vehicle plate detection is the wide variety of license plates that exist in the world. License plates come in different shapes, sizes, and colors, and they can be made from different materials such as metal, plastic, or paper. This means that the appearance of license plates can vary significantly, making it difficult to identify them using a single method or algorithm. Another challenge is the fact that license plates can appear at different orientations and scales in an image or video. They may be partially obscured by other objects in the scene, or they may be distorted due to perspective effects. This can make it difficult to accurately locate and extract information from the license plate. To address these challenges, vehicle plate detection systems typically use a combination of image processing techniques and machine learning algorithms. Image processing techniques can be used to pre-process the image or video, such as by enhancing the contrast or removing noise. Machine learning algorithms can then be used to analyze the processed image or video and identify features that are characteristic of license plates. These algorithms can be trained on large datasets of labeled images to learn how to accurately detect and recognize license plates. Once the license plate has been detected, it can then be processed to extract the text or numbers on the plate

**Chapter 2**

**Literature Survey**

**1.Siddharth U. Mishra(Author) B. E. Computer, Department of Computer Engg.,pune 2. Sushant N. Asija(Author) B.E. Computer, Department of Computer Engg.,pune 3.Amit A. Badgujar(Author) B. E. Computer, Department of Computer Engg.,pune 4. Bhavana Julme(Guide) Professor, Department of computer Engg.,pune**

Automatic license plate recognition system aims at extracting the license plate from a vehicle and using it for various purposes. In this paper we do a systematic study of the existing ALPR systems, the basic algorithm used, the variations in the existing algorithm to improve the overall system. We also present the list of applications where this system could be used, we elaborate one such application which is the criminal surveillance. The system which we are developing recognizes a six digit license plate which could be also modified to detect various other types of license plates as well. The algorithm mostly concentrates on localization of license plates and then go on to extract the characters by using morphological operations such as dilation, eroding the image, dilating, filtering etc. All these morphological operations leads to the efficiency of overall system. Index Terms – Automatic license plate recognition(ALPR), Vehicle license plate recognition(VLPR) , character recognition.

1. INTRODUCTION Automatic number plate recognition systems (ANPR) provide a means to overcome the drawbacks and deficiency of successful surveillance of the cctv cameras. The ANPR system is well developed in certain countries such as USA and Dubai, and existed from a long time, but only in the late 90s it became an important application because of the large increase in the number of vehicles. The information extracted from the license plates is mainly used for traffic monitoring, access control, parking, motorway road tolling, and border control, making car logs for parking systems, journey time measurement for toll booth etc. by the law enforcement agencies. The recognition problem is generally sub-divided into 5 parts: (1) image acquisition i.e. capturing the image of the license plate (2) pre-processing the image i.e. normalization, adjusting the brightness, skewness and contrast of the image (3) localizing the license plate (4)Detecting web character segmentation i.e. locating and identifying the individual symbol images on the plate, (5) optical character recognition. There may be further refinements over these (like matching the vehicle license number with a particular database to track suspected vehicles etc.) but the basic structure remains the same.

A guiding parameter in this regard is country-specific traffic norms and standards. This helps to fine tune the system i.e. number of characters in the license plate, text luminance level (relative index i.e. dark text on light background or light text on dark background) etc. So the problem can then be narrowed down for application in a particular country. For example, in India the norm is printing the license plate numbers in black color on a white background for private vehicles and on a yellow background for commercial vehicles. The general format for the license plate is two letters (for state code) followed by district code, then a four digit code specific to a particular vehicle. In U.S.A no strict guidelines have been set regarding the fonts that can be used for this purpose

**Chapter 3**

**Methodology**

**3.1) Proposed System**

Character segmentation and recognition relies on the accuracy of number plate extraction. Hence this processing step is critical one and emphasizes on developing numerous techniques so that number plate detection is faithful. Methods to detect the number plate are based on: texture, edge detection, histogram, morphological processing, and transformation.

**MODEL DIAGRAM**

****

Texture based Candidate Extraction Texture and edge information are considered for number plate extraction in [1]. The method uses projection properties to detect the candidate number plate. Since the maximum projection does not correspond to the plate area all the time, several peak lines in vertical projection are detected and maximum peak values are considered as reference lines. After binarization, all connected component areas across each reference line, are sorted by acreage, which is used to calculate the size of sliding window.

All candidates are extracted along with Rank filter and Robert’s operator. The result of searching reference line is as shown in Figure. 1. To reject the detected fake candidates, autocorrelation followed by projection algorithm has been implemented. The method is used to detect different types of Chinese name plates and different illumination conditions in real scene. The performance was reported for 1704 images with the recognition accuracy of 97.5%

**3.2) DATA PREPROCESSING**

Preprocessing is done to transform raw data into a format that can be used by machine learning. A data scientist can use an applied machine learning model to obtain more accurate findings by using structured and clean data. Data formatting, cleansing, and sampling are all part of the method.

**3.3) DATASET SPLITTING**

A dataset used for machine learning should be partitioned into three subsets — training, test, and validation sets.

Training set: A data scientist uses a training set to train a model and define its optimal parameters it has to learn from data.

Test set: A test set is needed for an evaluation of the trained model and its capability for generalization. The latter means a model’s ability to identify patterns in new unseen data after having been trained over a training data. It’s crucial to use different subsets for training and testing to avoid model overfitting, which is the incapacity for generalization we mentioned above.

**MODEL TRAINING**

A data scientist can start building a model after preprocessing the acquired data and separating it into train and test sets. This procedure comprises "feeding" training data to the algorithm. Predictive analysis uses an algorithm to evaluate data and produce a model that can locate a target value (attribute) in fresh data. To create a model is the goal of model training.

For model training, two machine algorithms have been used to compare and evaluate the performance. We did the train test split to see the results of the model.

**Methods for Character Segmentation**

I. Number plate candidates considered in extraction stage are examined in the phase of character segmentation. A wide variety of techniques to segment each character after plate localization has been developed. Segmentation techniques based on projection, Hough transform, region growing, and plate models are considered for addressing. A. Segmentation by Projection In [10], horizontal histogram is used to find out upper and lower bounds of a number plate on the vertical edging image first. After the area bounded by upper and lower bounds is found, the areas above the upper bound and the areas below the lower bound of a number plate are removed. Area without the upper and lower bounds is considered for character segmentation. Character segmentation phase is executed in two steps. First the selected area is binarized and secondly vertical projection is performed to find the gaps between characters for a number plate. Based on the results of vertical projection, each number plate is separated into blocks horizontally starting from the zero points in the projection histogram. Figure.4 shows the block segmented image of number plate. Because of the influence of noise, rivet, space mark, rotation of plate, it may happen that one character region has been separated or two characters are grouped into one single character. Hence a refining algorithm is developed by the authors. This refining is based on width of original number plate characters. The result of refining is demonstrated . This method has correct character segmentation rate of 98.82%

**Region Growing based Segmentation Six step algorithms :**

Image preprocessing, enhancement of character regions, edge detection of characters, location of the candidate regions of the characters, determination of character segmentation regions, characters segmentation and binarization. This algorithm is processed on extracted car number plates. Character regions are enhanced by contrast stretching transformation followed by edge detection using Laplacian Transformation. Location of the candidate regions are found out with the help of region growing algorithm. Region Growing is a procedure that groups pixels or sub regions into larger regions based on predefined criteria. The basic approach is to start with a set of ‘seed points’ and from these seed points growing the regions by appending to each seed, to those neighboring pixels that have properties similar to seed point. In this paper, starting pixels are selected having the grey value of 255 and region is grown from seeds to with the pixels that are 8-connected to at least one pixel in that region. To eliminate the fake regions, standard value of height and width of characters are used for comparisons. Refining process is also proposed for region growing algorithm, which then extracts correct segmented characters. After getting the exact position of characters, characters are extracted from original image and binarized to get higher contrast. Figure.4 illustrates the execution of this region growing algorithm. In this method 320 images were tested for character segmentation and the success rate obtained is 97.2%.

**The pattern matching technique**

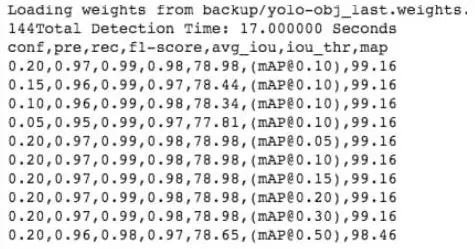
It is suitable for the recognition of shape invariant, non-rotated characters. P. Comelli et al. used in [19], cross co-relation to find out the matching between normalized characters and templates. This algorithm is implemented on Italian cars passing through tollgate. Templates are designed for two-letter province, single prototypes for character string. Algorithm tested more than three thousand real images acquired under different weather and illumination condition obtaining a recognition rate near to 91%. Authors in [20] developed an OCR technique, for recognition of Thai car license plates, which are generally shape complicated, using essentialelements-based method. This algorithm recognizes the character patterns relying on the essential elements of characters. Using this method, majority of characters are reduced to a set of essential strokes. In [21], matching of characters is based on edge Hausdorff Distance (HD), also called max-min distance. It is a measure of similarity between two arbitrary point sets. The method possesses all the mathematical properties of a metric, and its recognition rate is very similar that obtained with neural network classifiers but slightly slower.

LPR algorithms addressed were country specific. To predict the number plate, algorithms presented in the literature are restricted to the working conditions such as, distance, background, illumination, vehicle position. LPR can be a measure of vehicle identification, it may be further exploited for vehicle model identification and speed estimation. The measure of section travel time could be useful in traffic management, planning and control. Increased mobility and globalization set the challenges of developing effective LPR system that could handle plate from various countries with different character sets and syntax. Most LPR system focuses on processing of one nameplate in the image. But, as, input image may contain more than one name plates; the future challenging task is to detect these plates.

**You Only Look Once (YOLO) YOLO**

This architecture proposes the use of end-to-end neural networks to predict object class probabilities and detect the bounding boxes all at once. This approach potentially beats other real-time object detection techniques by a large margin. State of the art for YOLO has seen many versions and improvements with time, listing here different versions of YOLO proposed till date: YOLO YOLO v2 YOLO v3 YOLO v4 YOLOR (You Only Learn One Representation) As an outcome of the object detection (here object being the number plate), using any of the techniques as mentioned above, we would obtain the bounding box images of the license plates detected for each supplied vehicle image. Note that we can use pre-trained license plate recognition models or make a training model from real-time data using the above-mentioned algorithmic approaches. Some of the available datasets for Automatic License Plate Detection are: Indian License Plates Dataset License Plate images Dataset of 500 vehicles captured from rear view Dataset of South America Mercosur license plates with images and labels Chinese City Parking Dataset



****

**Chapter 4**

**Result and Discussion**

Market Dynamics of Vehicle Number Plate Detection Massive growth in the automobile sector, advancements in the processing technologies, development of integrated unmanned aerial vehicles (UAVs), infrastructure growth in emerging economies, increasing allocation of government funds for intelligent transportation systems, etc., are expected to propel the vehicle number plate recognition market growth globally. The adoption rate of the license number plate recognition mechanism has also increased to locate vehicles – in the defaulter’s list for toll collection purposes, have violated traffic laws, have expired insurance policies, or been stolen. The USA and other European countries are actively taking advantage of the computer vision-powered mechanism for security and traffic enforcement applications. The market size was USD 2.3 billion in 2020, which is expected to reach USD 3.8 billion by 2025, at a CAGR of 10.0% during the forecast period. Apart from this, the license plate recognition system allows direct connectivity, identifies the full string, automates tasks, and inspects international plate syntax to define the country of origin. It also offers cost-effective, flexible, and scalable storage solutions. With automated systems like parking payment systems, booking systems, electronic toll collection, etc., the AI-enabled tool enhances the experience of end-users. This ultimately drives the growth of the license plate recognition system market.

The solution was finally implemented using YOLOv3 with the neural network framework Darknet. ● For text extraction, Microsoft Vision API is considered. ● For certain image processing steps, a combination of PIL and OpenCV has been used

1. Detection model: YOLOv1, YOLOv2, YOLOv3.For implementing any of the Yolo version, we need a neural network framework i.e either Darknet or Dark flow B) Text Extraction from Number Plate: A number of pre-existing libraries in python were tried out: Pytesseract, Pyocr, textract. Google Vision API and Microsoft Vision API were also tried up.

**Chapter 5**

**CONCLUSION**

Vehicle number plate recognition has become a mature technology and is broadly used in various applications serving vehicle detection, localization, and recognition. This computer vision technology captures photographic surveillance and owes the capacity to transform the optical data from the images to identifiable digital information in real-time scenarios. Indeed, this technology provides an easy-to-understand, cost-effective, better, faster, touchless, and frictionless vehicular identification and parking service.

**FUTURE SCOPE**

From automating tasks to managing space, improving the mobility of people and goods, reducing traffic congestion, and managing incidents effectively, the vehicle number plate identification system offers numerous benefits. Get in touch with our experts to implement the vehicle number plate recognition system and enhance the urban mobility experience.

**REFERENCES**

* [**www.researchreader.com**](http://www.researchreader.com/)
* [**https://mobisoftinfotech.com/resources/blog/vehicle-number-plate-detection-mechanism-using-computer-vision/**](https://mobisoftinfotech.com/resources/blog/vehicle-number-plate-detection-mechanism-using-computer-vision/)
* [**https://medium.com/data-science-in-your-pocket/vehicle-number-plate-detection-and-ocr-tcs-humain-2019-a253019e52a1**](https://medium.com/data-science-in-your-pocket/vehicle-number-plate-detection-and-ocr-tcs-humain-2019-a253019e52a1)