iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** IV **Month of publication:** April 2023

DOI: https://doi.org/10.22214/ijraset.2023.50169

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue IV Apr 2023- Available at www.ijraset.com

Vehicle Allowance System

Avinash Mudda[1], P. Sashi Kiran[2], Ashish Kumar[3], Venkata Sreenivas[4]
Gitam University, India

Abstract: ***License plate detection is an image processing technology that uses a license (number) plate for vehicle identification. The objective is to design and implement an efficient vehicle identification system that identifies the vehicle using the vehicle's license plate. The system can be implemented at the entrance of parking lots, toll booths, or any private premises like colleges, etc. to keep records of ongoing and outgoing vehicles. It can be used to allow access to only permitted vehicles inside the premises. The developed system first captures the image of the vehicle's front, then detects the license plate, and then reads the license plate. The vehicle license plate is extracted using image processing of the image. Optical character recognition (OCR) is used for character recognition. The system is implemented using OpenCV and its performance is tested on various images. It is observed that the developed system successfully detects and recognizes the vehicle license plate.***

To recognize License number plates using the Python programming language. We will utilize OpenCV for this project to identify the license number plates and the python py-tesseract for the characters and digits extraction from the plate. We will build a Python program that automatically recognizes the License Number Plate.

The development of an intelligent vehicle identification system based on the optical character recognition (OCR) method is to be used in intelligent transportation systems. The proposed system makes use of an intelligent parking system named Smart Parking Service (SPANS), which is used to manage public or private spaces. Using computer vision techniques, the SPANS system is used to detect if the parking slots are available or not. The proposed system makes use of the SPANS framework to capture images of the parking spaces and identifies the license plate number of the vehicles that are moving around the parking as well as parking in the parking slots.

I. INTRODUCTION

Computer vision is used in many different areas. One of them is image and video processing for purposes of object detection in these images or video files. Nowadays, a very common application of object detection in images is the detection of car license plates.

The solution proposed in this paper was implemented using the OpenCV library. OpenCV (Open Source Computer Vision Library) is an open-source library used for creating applications in the field of computer vision and machine learning. The library supports many programming languages, including Python which was used for the implementation of the solution. The library is compatible with operating systems Windows, Mac OS, iOS, Linux, or Android. During the implementation of the designed solution, we used the OpenCV library for the detection and recognition of license plate objects in a way of applying filters and operators for image and video processing.

In the Image Processing Module experimentation, four parts are going to be examined. Each part will influence the algorithm and the detection threshold of the license plate identification. The first part which is going to be examined is the ratio of the image. This process is relevant to compare the detected blob with its actual size as the requirement for the license plate candidate. The second part is a test on the rotation or alignment of the image. This particular process is important in determining the threshold of the skew that the license plate does have for it to be identified by the program. It is also conducted to anticipate the probability in which the image will slant as a result of the inappropriate position of the camera or a tilted license plate. The third part is a distance measurement. This particular measurement is needed to evaluate the 2 megapixels camera in terms of determining the required distance for it to identify and detect a license plate accurately. The fourth part is determining the binary threshold value with the Global Threshold and Adaptive Threshold.

Automatic Vehicle number plate recognition system used for identifying numbers and obtaining owner information from a large database of registration details. The Recognition process includes submitting a query and extracting characters of the image that best matches with the template if matched, obtaining the owner details. In visual contents, normally called features are used to recognize alphabets and numeric characters to obtain registration details from large databases. Vehicle number plate recognition systems are used as core modules for intelligent infrastructure systems like electronic payment systems (toll payment and parking fee payment) and freeway and arterial management systems for traffic surveillance

©IJRASET: All Rights are Reserved

To introduce a solution for car license plate detection used for access restriction and monitoring cars accessing the parking area through the entrance ramp. The cars are being monitored by a camera on the entrance ramp. After video processing, access can be granted. The purpose of the solution is the detection of the license plate area in this video file and recognition of the license plate number.

II. LITERATURE REVIEW

1) *Dalarmelina, N. do V., Teixeira, M. A., & Meneguette, R. I:* Automatic License Plate Recognition has been a recurrent research topic due to the increasing number of cameras available in cities, where most of them if not all, are connected to the Internet. The video traffic generated by the cameras can be analyzed to provide useful insights for the transportation segment. This paper presents the development of an intelligent vehicle identification system based on the optical character recognition (OCR) method to be used on intelligent transportation systems. The proposed system makes use of an intelligent parking system named Smart Parking Service (SPANS), which is used to manage public or private spaces. Using computer vision techniques, the SPANS system is used to detect if the parking slots are available or not. The proposed system makes use of the SPANS framework to capture images of the parking spaces and identifies the license plate number of the vehicles that are moving around the parking as well as parking in the parking slots. The recognition of the license plate is made in real time, and the performance of the proposed system is evaluated in real-time.

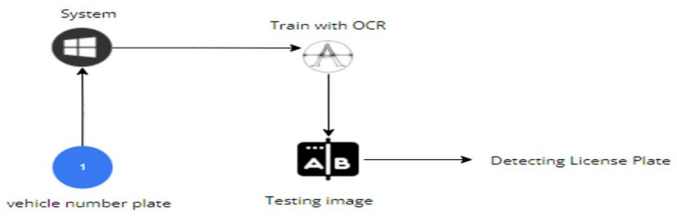
- *Summary:* The proposed system makes use of the intelligent parking service (SPANS) which is a framework used to detect available parking slots using computer vision functions. The proposed system uses the camera of the SPANS to get images and information about the parking slots. Once a vehicle is detected, the proposed system takes a picture of the vehicle and uses this image to identify the license plate number of the vehicle. So, the identified number is stored in the system, and this information can be made available to public agencies such as traffic departments.

2) *Cheng, G., Zhou, P., & Han, J:* Object detection in very high-resolution optical remote sensing images is a fundamental problem faced for remote sensing image analysis. Due to the advances in powerful feature representations, machine-learning-based object detection is receiving increasing attention. Although numerous feature representations exist, most of them are handcrafted or shallow learning-based features. As the object detection task becomes more challenging, their description capability becomes limited or even impoverished. More recently, deep learning algorithms, especially convolutional neural networks (CNNs), have shown their much stronger feature representation power in computer vision. Despite the progress made in natural scene images, it is problematic to directly use the CNN feature for object detection in optical remote sensing images because it is difficult to effectively deal with the problem of object rotation variations. To address this problem, this paper proposes a novel and effective approach to learning a rotation-invariant CNN (RICNN) model for advancing the performance of object detection, which is achieved by introducing and learning a new rotation-invariant layer based on the existing CNN architectures. However, different from the training of traditional CNN models that only optimizes the multinomial logistic regression objective, our RICNN model is trained by optimizing a new objective function via imposing a regularization constraint, which explicitly enforces the feature representations of the training samples before and after rotating to be mapped close to each other, hence achieving rotation invariance.

- *Disadvantages*
  - Less feature compatibility
  - Low accuracy

III. SYSTEM DESIGN

A. *Proposed System Architecture*



System Requirements Specification

B. Proposed System Block Diagram

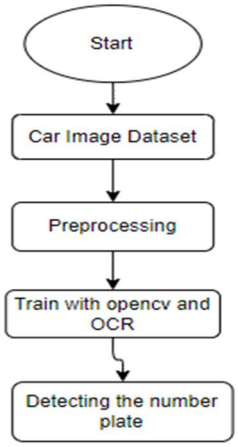


Fig 1. Block diagram of the proposed method

C. System Specifications

1) H/W Specifications

- Processor : I5/Intel Processor
- RAM : 8GB (min)
- Hard Disk : 128 GB

2) S/W Specifications

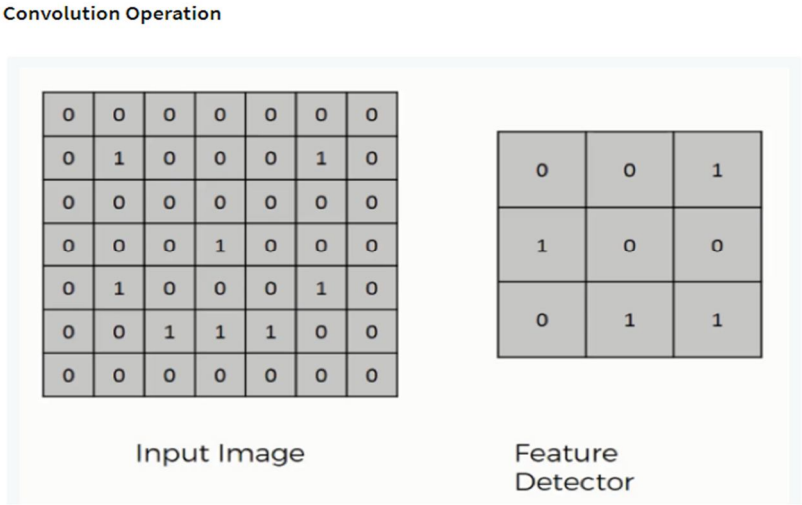
- Operating System : Windows 10
- Server-side Script : Python 3.6
- IDE : Pycharm, VS code
- Libraries Used : Numpy, IO, OS, Keras, pandas, OpenCV, Pytesseract OCR

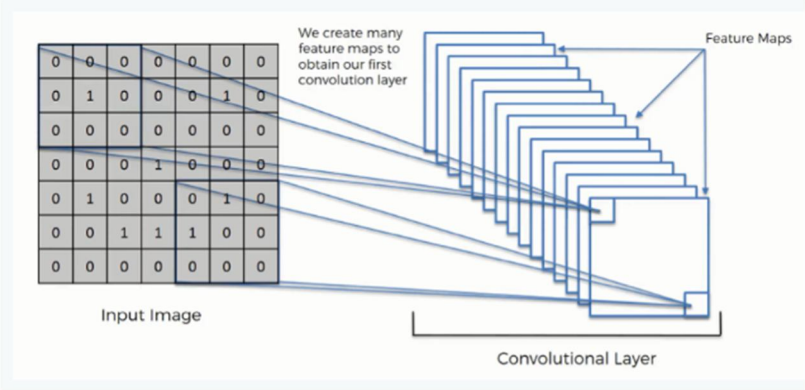
IV. ALGORITHM

A. Convolutional Neural Network

1) Step1: Convolutional Operation

The first building block in our plan of attack is the convolution operation. In this step, we will touch on feature detectors, which serve as the neural network's filters. We will also discuss feature maps, learning the parameters of such maps, how patterns are detected, the layers of detection, and how the findings are mapped out.

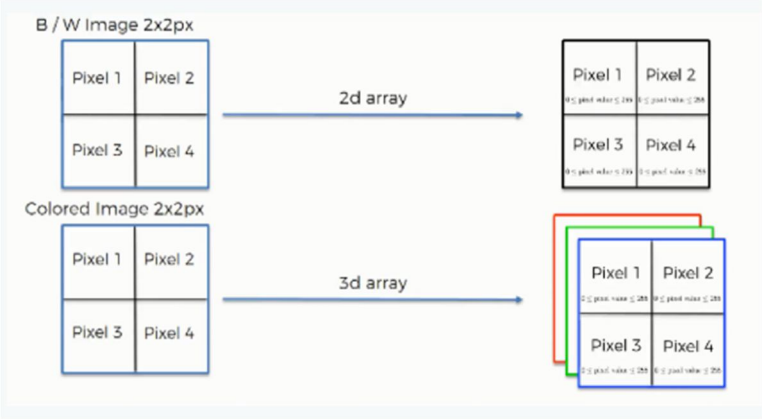




- *Step (1b): Relu Layer*

The second part of this step will involve the Rectified Linear Unit or Relook. We will cover Relook layers and explore how linearity functions in the context of Convolutional Neural Networks.
Not necessary for understanding CNN, but there's no harm in a quick lesson to improve your skills.

Convolutional Neural Networks Scan Images



2) Step 2: Pooling Layer

In this part, we'll cover pooling and will get to understand exactly how it generally works. Our nexus here, however, will be a specific type of pooling; max pooling. We'll cover various approaches, though, including mean (or sum) pooling. This part will end with a demonstration made using a visual interactive tool that will sort the whole concept out for you.

3) Step 3: Flattening

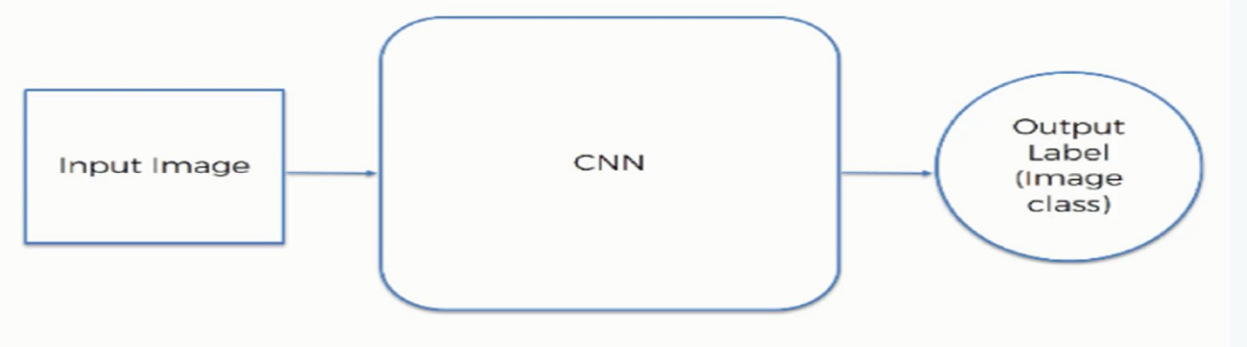
This will be a brief breakdown of the flattening process and how we move from pooled to flattened layers when working with Convolutional Neural Networks.

4) Step 4: Full Connection

In this part, everything that we covered throughout the section will be merged. By learning this, you'll get to envision a fuller picture of how Convolutional Neural Networks operate and how the "neurons" that are finally produced learn the classification of images.

V. SUMMARY

In the end, we'll wrap everything up and give a quick recap of the concept covered in the section. If you feel like it will do you any benefit (and it probably will), you should check out the extra tutorial in which Soft ax and Cross-Entropy are covered. It's not mandatory for the course, but you will likely come across these concepts when working with Convolutional Neural Networks and it will do you a lot of good to be familiar with them.

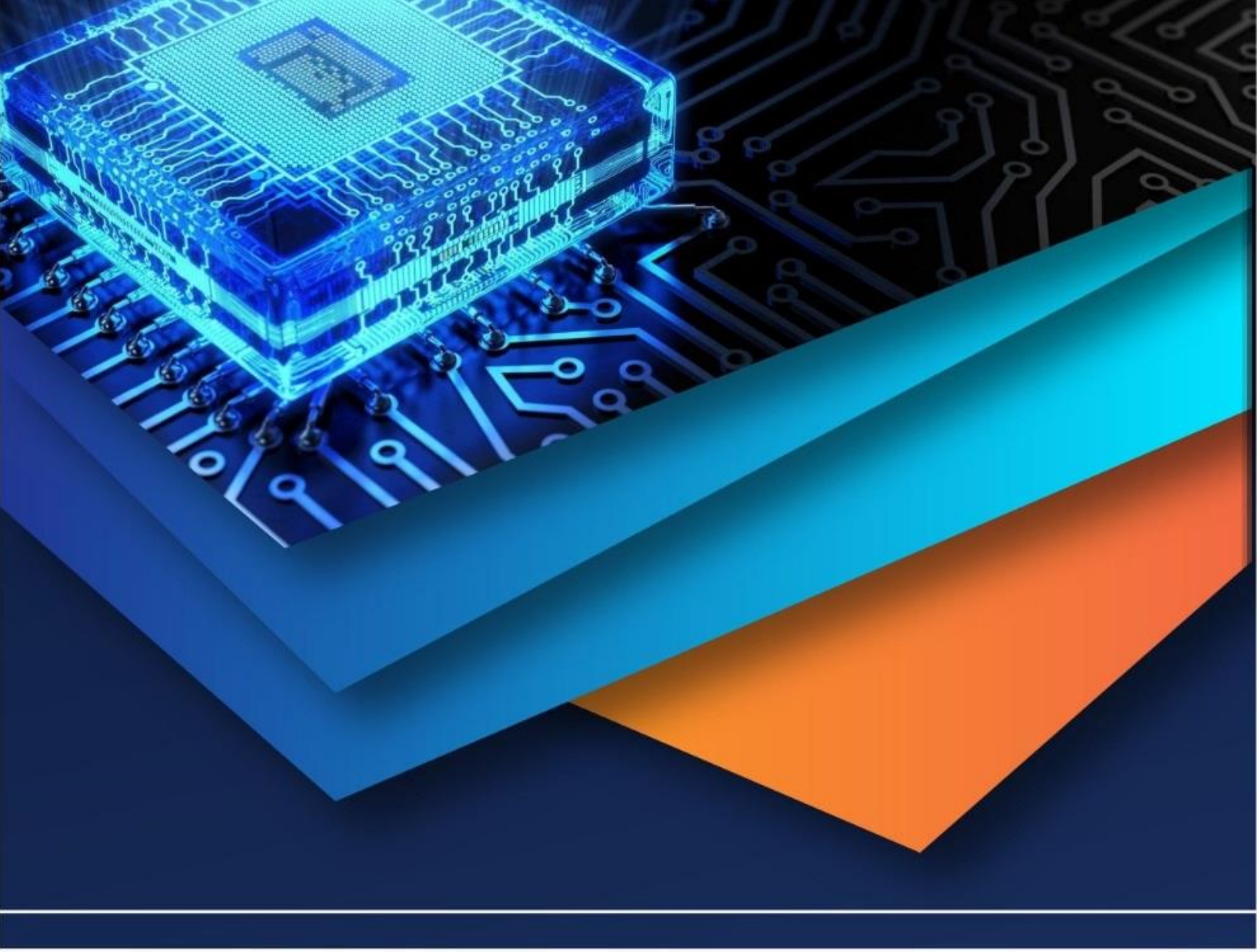


A. Test Cases

Input	Output	Result
Input text	Tested for the Detecting License plate in images	Success

B. Test Cases MODEL Building

S.NO	Test cases	I/O	Expected O/T	Actual O/T	P/F
1	Read the dataset.	Dataset path.	Dataset needs to be read successfully.	Dataset was fetched successfully.	P
2	Performing pre-processing on the dataset	The Pre-processing part takes place	Pre-processing should be performed on the dataset	Pre-processing completed.	P
3	Model Building	Model Building for the clean data	Need to create a model using required algorithms	Model Created Successfully.	P
4	Detecting	Input image provided.	The Output should be Detecting License plates in images	Model classified successfully	P



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)