International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 07 | July 2020 www.irjet.net p-ISSN: 2395-0072

Smart Parking using Image Processing and Morphological Operation's

Aditya Santhosh[1], Akarsh S[2], Ananda N[3]

[1-3]*Department of Information Science Engineering Nitte Meenakshi Institute of Technology, Bengaluru 560 0641*

Abstract - *This paper aims to develop a robust, flexible and simple to use application with its main functionality to ease the way people find parking spots by helping them navigate to vacant spots. The overall focus of this application is to make the process of parking much simpler. We are using image processing techniques to keep a record of the vehicles in the parking slots and also to help find vacant parking spots. License plates are detected at the entry to keep a record of the vehicles, this is done using opencv and pytesseract library in python. The system can be implemented on the entrance of parking lots, toll booths, or any private premises like college, etc. to keep the records of ongoing and outgoing vehicles. The next part is to find vacant spots, this is done using image processing capabilities of OpenCV. It removes the need for independent sensors to detect a car and instead, uses real-time images derived from various sources and servers to consider a group of slots together. This greatly decreases the expenses required to design an efficient parking system and increases the flexibility of the operation.*

Key Words: edge detection, bilateral filters, OpenCV, PyTesseract.

1. INTRODUCTION

The main focus for us developing this application is to make the validation of a vehicle and the problem of finding a vacant parking slot in the current situation of traffic much easier. Traffic arising from automobiles searching for vacant parking spaces is prominent in populated urban areas. Real-time parking occupation data provides critical input for a parking management system, which are usually acquired by on-site sensors. However, insertion and sustaining of these sensors can be cost inducing. To solve the parking organization issue, different techniques have been developed and research was conducted to develop efficient parking systems. These technologies include Digital Image processing systems, embedded systems and Internet of Things.

In this paper, a very unique algorithm is designed taking into consideration the joined capabilities of Gaussian blurring, Truncate thresholding, Canny edge detection, contour Detection combined with the capabilities of IOT to give user efficient results and access to data. But before all this the first thing that we have done is to include automatic number plate detection so that we can keep a record of the vehicles entering and leaving the parking space, this can even be used to calculate parking fee. The basic model of automatic number plate recognition system consists of 4 main phases: 1) Image Acquisition and Preprocessing phase 2) Number Plate Extraction3) CharacterSegmentation4) Character Recognition phase. Now we will get back to the algorithm to find vacant spots. By use of Gaussian Blur the figure is stripped of high frequency and sharp contrast for smooth analysis. This augments the texture detection of the set image while removing noise added during acquisition the image. Textures from images will give a substantial deal of data about the images. The edge of the image is the extreme at which the gray value of the neighboring pixel at the sudden point of the signal changes drastically.

Presently, many applications depend on accurate tracking of targeted contours obtained through many approaches. It is one of the fundamental and still open image processing tasks due to the complexity of analysis of images of different types with a large number of objects considered. The added slot booking, and navigation capabilities give user the flexibility and option to integrate the algorithm with other autonomous devices. The system proposed can save time and energy needed to search for vacant slots while providing with data to be integrated with autonomous vehicles.

1.1 PROPOSED APPROACH FOR NUMBER PLATE EXTRACTION

License plate of the vehicle is detected using various features of image processing library OpenCV and recognizing the text on the number plate using python tool named as tesseract. To recognize the license plate, we are using the fact that a number is rectangular. So, after processing an image, we will find the contour having for points inside the list and consider as the license plate of the vehicle.

This consists of the following main steps:

(1) Import libraries and image

(2) Preprocessing

(3) Detecting Plate

(4) Text Recognition.

Import libraries and image

To implement the project first various python tools and libraries are imported. I had imported four libraries OpenCV for image processing, Numpy for mathematics, Matplotlib for

plotting an image and Pytesseract for optical character recognition (OCR).

Preprocessing

The basic aim of pre-processing is to improve the contrast of the given image, to reduce the noise in the image, which in turn enhances the processing speed. In pre-processing RGB image is converted into gray scale image and then into binary image. The contrast enhancement is done by histogram equalization, contrast stretching etc. Various filters are used to remove noise from the image. In the proposed approach for number plate extraction, the input image is enhanced by applying adaptive histogram equalization technique and noise is removed by iterative bilateral filtering.

Fig-1:Gray Scale Conversion

Detecting Plate

After we sort the contours, we will now take a variable plate and store a value none in the variable recognizing that we did not find number plate till now. Now we iterate through all the contours we get after sorting from the largest to the smallest having our number plate in there so we should be able to segment it out. Now to that, we will look through all the contours and going to calculate the perimeter for each contour. I had used contour approximation and it approximate a contour shape to another shape with a smaller number of that is dependent on the position I specify so the 0.02 is the precision that worked. After that we will compare if edges count is equal to 4 so we found our number plate.

After that we will find the coordinates of the rectangle formed using cv2.boundingRect(c) and store the one coordinate in x, y and store width and height of the contour in another. After that we put the image of detected rectangle in the plate variable.

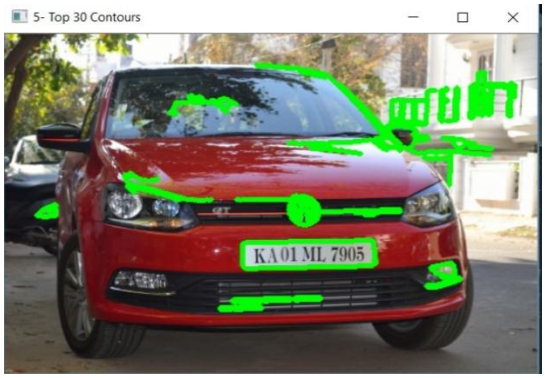


Fig-2:Top 30 Contours

Fig-3:Approximate Number Plate Contour

Text Recognition

After detecting the license plate of the vehicle, we will recognize the characters on the license plate using tesseract. Python-tesseract is an (OCR) optical character recognition tool in python. That is, it will recognize and read the text embedded in images. It is a wrapper for Google's Tesseract-OCR Engine. It is also useful as a stand-alone invocation script to tesseract, as it can also read all the image types supported by the Leptonica imaging and Pillow libraries, including png, jpeg, gif, BMP and others. Optical character recognition (OCR) is a conversion of printed text images or handwritten text scanned copy, into editable text for further processing. This technology gives an ability to the system to recognize the text automatically. It is like a combination of the mind and eyes of the human body. An eye can only view the text from an image but the brain actually processes as well as interprets that extracted text read by eye.

1.2 PROPOSED APPROACH FOR PARKING SLOT DETECTION

Our proposed approach to find vacant parking slots is by using the Open Source Computer Vison library in Python. OpenCV is an extensive open source library (available in python, Java, and C++) that's used for image analysis and is pretty neat. To detect the parking spots, we knew that we could take advantage of the lines demarking the boundaries.

So, from the live video the image taken will be converted to a gray scale image to reduce information and to smoothen the image. We are also using Gaussian blur to remove unnecessary noise and canny edge detection function in python to draw out the contours of the rectangular parking slots. The following is the algorithm that we will be using:

Initially: Mark the coordinates of slots in yml

Step 1: Record the video and process each frame

Step 2: Draw approximate rectangle around rectangle and highlight the region of interest

Step 3: Draw the contours

Step 4: Image smoothening (blurring)

Step 5: Gray scale conversion

Step 6: For each index of slot call procedure FIND change according to status value

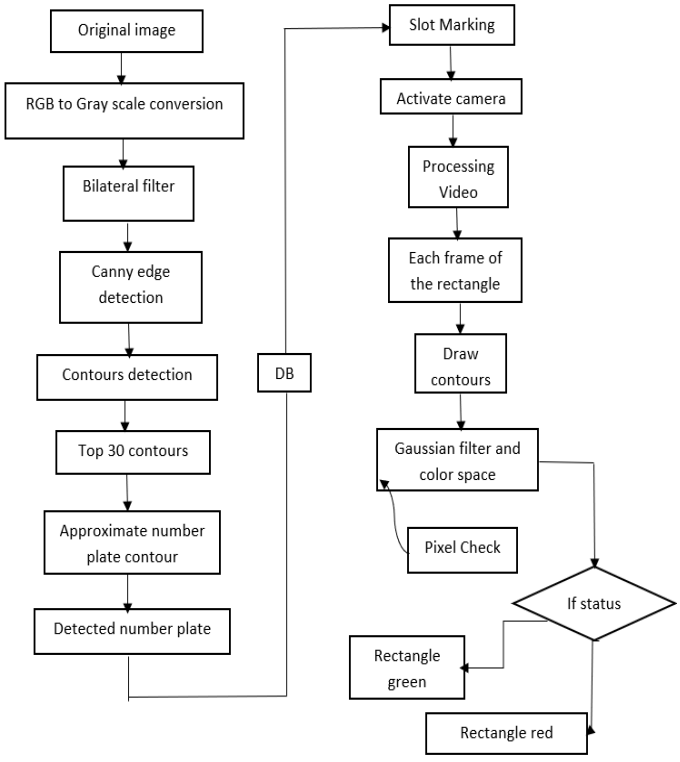
Procedure FIND

Step 1: Find edges and get pixel value (Laplacian)

Step 2: Find average of pixel change

Step 3: Return status

2. METHODOLOGY



Video Acquisition of the Parking Area

The first step consists of installing multiple high-resolution cameras at various places at appropriate angles in the parking area. Determine the initial layout of the parking lot In order to determine the number of available slots in the parking lot, it is required to have knowledge about the layout of the parking lot being used for the implementation. So, the next step is to detect the lines of parking space directly from the images of the empty parking lot. Camera calibration is the process of estimating intrinsic extrinsic parameters. Intrinsic parameters deal with the camera's internal characteristics like its focal length, skew, distortion, and image center. Extrinsic parameters describe its position and orientation in the world. Knowing intrinsic parameters is an essential step for 3D computer vision, as it allows to estimate the scenes structure in the Euclidean space and removes the lens distortion. Videos will be acquired and recorded from multiple cameras and captured videos will be stored in the video repository.

Video Pre-processing

Video sequences from video repository will be converted into video frames/images and sent to image processing framework. Image processing framework consists of 4 major stages: pre-processing, segmentation, feature extraction and classification which include many existing techniques such as feature extractors (based on color, texture, shape, logo, etc.). and machine learning algorithms (Deep learning, SVM, Bayesian Network, Fuzzy classifier, etc.). Images will be converted to suitable format i.e. Gray scale, HSV or any other suitable format depending upon the properties extracted.

Drawing contours of the parking spots

The next step is to draw contours of the rectangular parking slots. The coordinates of the moving object are found and a blue rectangle is drawn around it. If the contour area is less than the minimum area that has been set, then the contour will be discarded.

Finding out if the slot is available

There is method known as background difference method, the difference between the image frame sequence and the background image is calculated, and then the target object is segmented and extracted. When using the background difference method to detect the vehicle in parking lot, the changes in other areas in the image need to be separated, it increases the calculation and reduces the detection accuracy. Therefore, this paper makes improvement on background difference method by segmenting the image of all the parking area, only need to carry on the background difference calculation to the interested parking space. Since the vehicle covers a larger area after parking, it is possible to eliminate the interference of external factors by setting the threshold of

pixel variation. Assuming the background is B(x,y,t), the target vehicle is M (x, y, t), the image is I (x, y, t), the noise is N(x, y, t), the target vehicle can be expressed as:

$$M(x, y, t) = I(x, y, t) - B(x, y, t) - N(x, y, t) \quad (2)$$

Where x and y represent the horizontal and vertical coordinates of the area, t represents the current moment. To determine whether the object currently extracted is a vehicle or not, it needs to be compared with the preset threshold value T. If the accumulation amount of the pixel change in the area caused by the vehicle entering the parking space exceeds the preset threshold value, i.e. M (x, y, t) >T, and tends to be stable after reaching a certain value, you can determine the current parking space has vehicle parking. The rectangle will turn to green if its vacant and red if it is already occupied.

3. CONCLUSIONS

Fig-4:Vacant slot

Fig-5:Occupied slot

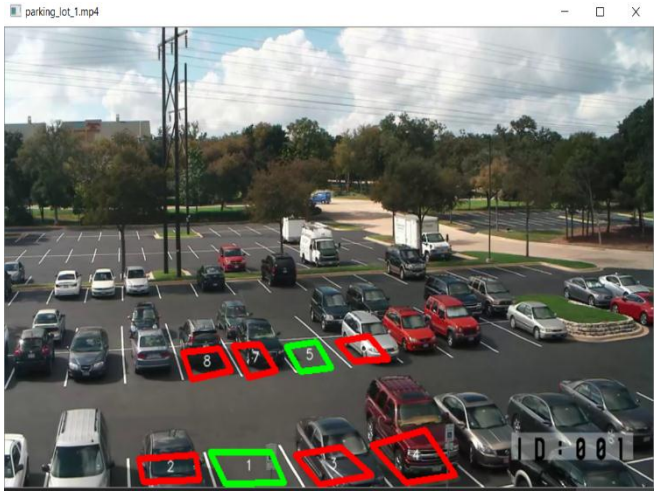


Fig-6:Implementation of proposed system on a parking lot

From the above images it can be seen that this proposed approach can be used in commercial parking lots.

Parking will always be an issue in the current scenario in all metropolitan cities etc. What we have done is tried to make an application which makes it easier to keep record of all vehicles entering a parking lot and also assisting them in navigating to a vacant parking spot. The algorithm proposed was designed to combat these situations. The algorithm combines the image processing capabilities of OpenCV and image processing. The algorithm proposed has an efficient false contour detection and elimination technique to remove false predictions. The results are displayed on a website where the user has the option to book slots or to set navigation path to the desired vacant spot. With the proposed method the user can select the desired spot for parking and give the data to any autonomous mobile device via website or the app. An efficient less time-consuming vehicle number plate detection method is proposed which performs on multiple image. By using, Canny edge detection method along with the tesseract library in python detects edges and fills the holes less than 8 pixels . The number plate detection is mainly to keep a record of the vehicles entering the parking lot.

ACKNOWLEDGEMENT

- We extend our sincere thanks to our HOD Dr. Sanjay H.A
- We would also like to thank our project guide Assistant prof Chandrashekar B.N.

REFERENCES

[1] Indian Vehicle Number Plate Detection Using Image Processing by Prof.Amit Kukreja , Swati Bhandari , Sayali Bhatkar , Jyoti Chavda , Smita Lad International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056 Volume: 04 Issue: 04 | Apr -2017J.

[2] An Efficient Method of Number Plate Extraction from Indian Vehicles Image by Sarbjit Kaur , Sukhvir Kaur International Journal of Computer Applications Volume 88 – No.4, February 2014

[3] Image Handling and Processing for Efficient Parking Space Detection and Navigation Aid NIT Warangal