Indian Journal of Science and Technology, Vol 9(44), DOI: 10.17485/ijst/2016/v9i44/105279, November 2016
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Image Processing Based Vehicle Parking Management Using Mobile Application

Suraj Dhalwar*, Yashwanth Chinta and B. Karthikeyan

Embedded Systems, VIT University, Vellore - 632014, Tamil Nadu, India;
surajdhalwar@gmail.com, yashwanthchinta9@gmial.com, bkarthikeyan@vit.ac.in

Abstract

In modern day scenario due to heavy traffic, it is always significant to find a free parking slot to park a vehicle in crowded area. The main motto of this project work is to find the available parking slot to park the vehicle and the current status of free parking slot should available on user mobile application even before visiting the parking area. In this paper the image processing technics is adapted to find the free parking slot. Camera captures the image of parking lot and the image is analyzed to find the possible number of free parking slot. Contour method is implemented to find coloured circle on parking lot which indicate the availability of slot. The various modifications like converting RGB images to HSV are performed on Open CV platform which is installed on Raspberry pi. As Raspberry pi is used it is easy to bridge image processing task and internet connectivity over server. It is efficient to send data over internet in small duration of time. This system provides the proficient way for the user to park vehicle without any guide. This system can be implemented in any public parking place, shopping complex, corporate offices etc. This method can be easily implemented in a short period of time and does not require regular maintenance.

Keywords: Image Processing, Mobile Application, Parking Slot Status, Vehicle Parking Management

1. Introduction

Now a day, in the high traffic world the vehicle parking is most difficult for the user if he not properly known about the parking lot status. In indoor parking lot driver take more time to decide about the parking lot. There are various ways to know about the real time status of parking lot. In this parking lot there may use the sensing element like RF sensor, ultrasonic sensor. But in that parking system each lot needed the separate sensing element[1]. This technique may take the more time to update information about the status of lot, also it has many limitations.

The main objective of this system is to get the details of free parking slot in a parking area. This information will be available to user on his mobile application[2]. The main part in this parking management system it gives the exact location of the vehicle in parking lot. The block diagram of the working system is given in Figure 1. In this system the camera is used as the sensing element it is possible to capture the image of many parking slots at a time. The camera can be mounted at the corner to take snap of the lot and send image to processing unit, here the processing and controlled unit is raspberry pi2. With the help of USB web camera, it captures the image of parking lot. As open CV is installed on raspberry pi using proper program it is possible to image processing on captured image. On the parking slot floor there is one colored painted circle in each slot by using image processing we have to find which circle is visible in image if painted circle is present in image it means there is no vehicle present in slot and hence slot is free. If painted circle is not visible in image it means vehicle is present on slot and slot is not available for parking[3].This status of empty slot is available to user on it mobile application, for this raspberry pi is connected to internet using Wi-Fi and it data over server this data is received by mobile application[2,4]. In this scenario

*Author for correspondence

Telegram is used is used as mobile application. User can easily check the status of available parking slot from anywhere to park his vehicle.

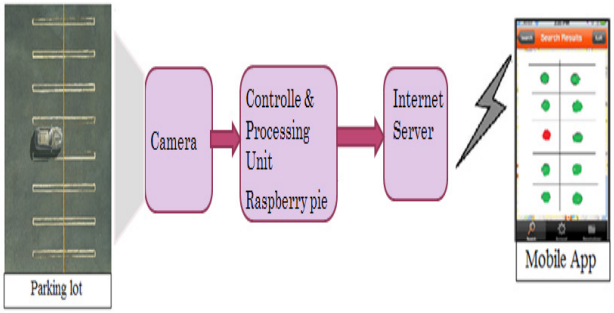

Figure 1. Block diagram.

2. Methodology

The system is based on both hardware and software. The system prototype contains the one camera that is used for the image capturing. This camera captures the image continuously after particular time interval. As mentioned in the Figure 2, this image is providing to the embedded board Raspberry pi2. The raspberry pi module used to do the image processing and send the information according to the lot status get from processing. The raspberry pi board is more compatible for the image processing as it is work as minicomputer in the system. We can easily install the software in it and use it. In raspberry pi module C, C++, Python programming is possible. In this system C++ programming is used to process image. The main software component in the system is the OpenCV libraries which contain the all package and header file that support image processing. By including OpenCV libraries in the C++ coding the function can used which needed for the image processing.

3. Result and Discussion

In this vital part the conversion of image from one plane to other takes place to extract the information from the parking lot image[5]. The operation flow of the image processing is shown Figure 3.

3.1.1 Image Identification

In this system the first task is to take the image by the camera that install at the corner of the parking lot. The image is 640x480 resolution. This image taken by the code function "cv Capture Camera ()". This function has parameter of the id of camera. If there are only one camera the parameter is -1 and if it is more than one, then the parameter may vary between 1 to 255.

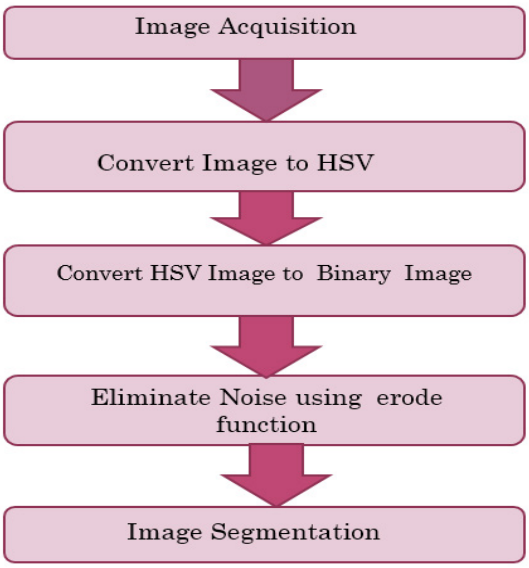

Figure 2. Algorithm.

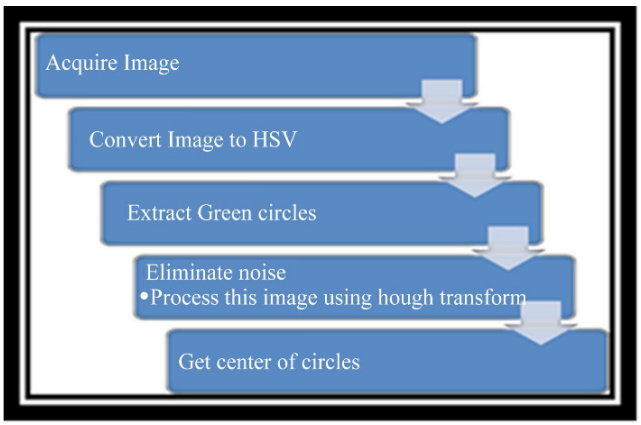

Figure 3. Image processing flow.

This image is the RGB image this image need to process hence we convert it to the HSV image by using function "RGB2HSV" the image from 3 components of RGB will get converted into HSV image. This HSV image is have the component of hue, saturation and value. This conversion gives the high intensity at light color[5].

After converting this image in HSV which not provide the much information. The image then converted into gray image by using function "HSV2GRAY". This gray image

gave the prior image near to binary image. The gray image is obtained to get the binary image. The formula for the RGB to GRAY image is shown.

Gray = (0.299*r + 0.587*g + 0.114*b)

In gray image the all pixel value is either near to zero or near to 1 hence it is possible to obtain the binary image from the gray image using threshold function. In threshold function the threshold value is fixed from 0 to 255. Generally, the thresholds value is nearly selected as 120 to 160. After thresholding the effect is we get the desired area as one and rest all area as dark i.e. zero intensity. The pixel below the threshold value will become totally dark and the pixel above the threshold value become totally bright.

The example of the test data is shown in the Figure 4. This is all we do in the programming to convert the image into the information. The binary image has more information related to this project while make decision on presence of the free lot. The all process of the image processing is display[5].

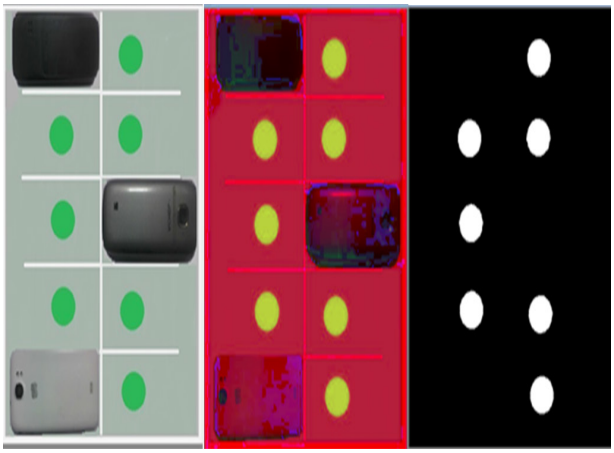

Figure 4. Processed image.

3.1.2 Image Acquisition

In this system the camera is mounted exactly at top of the parking lot. From the camera all the lot or maximum lot that we want to process should see. Any vehicle can be park in that region should not disturbed the camera. In the installation of camera one more thing is that the any unwanted light at that place should not allow which may effect on the image RGB pattern and processing of that image. The image should be get with same format every time from the camera. The camera position is fixed at one place should not vary again and again. This variation will effect on the image separation and may show the wrong analysis.

3.1.3 Image Separation

After converting the image into the binary image which gives the binary view of all lot. In this project we marked the spot of green color on the parking lot this spot is viewed as white in binary image and rest all in dark. There may be the possibility of having green color vehicle or other obstacle. To differentiate the circular spot from this all unwanted region we are going for image separation.

In image separation the image is separated in such a way that in each separate section the individual lot will get. This individual section now processes easily. On this individual section the counters are found and this contour is check where it is circular and the area of this counter is same as that of spot or not.

The counter is the one method to determine the presence of desired pattern in the binary image. The counter is find using the "findcounor ()" function in which the parameter of source image According to the pattern, the decision is made and the information is generated. The separation of the image is done by calculating the width and height of the image. The function "cvsubImage ()" it can give you the portioning of the image in rectangular format. In this partitioned image the contour will find and the lot is free or not will decide using the presence of the circular spot.

3.1.4 Image Development

In the image processing the proper capturing of the image is important. The image with less noise and less blur need in the processing. The image captured by the camera may be contain unwanted noise. In ordered to remove the noise in the image the eroding is used. In eroding process the small dots in binary image are easily remove which not affect the further processing[3]. Image development did at each stage after image separation. Due to the removal of unvented white pixel the number of counter in the image will get decrease and the process become faster and more accurate.

3.1.5 Image Determination

Image at last stage is gives the information date. The information data is the match count or not, if the lot is free the image shows the circular spot. This makes sure that the

slot is free. This information which is shown in the Figure 5, is then send from the system to user.

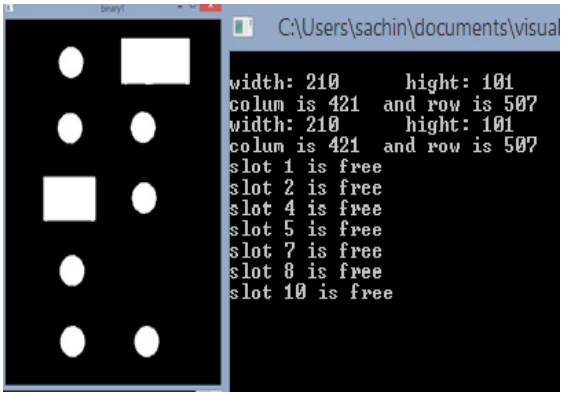

Figure 5. Information from processing.

3.2 Hardware

3.2.1 Raspberry Pi 2

Raspberry Pi 2 is the minicomputer that have the operating system based on the Linux. Raspberry pi is very high speed computing device which is more compatible with the image processing. Raspberry pi is product by the Broadcom. This computing device is based on the ARM cortex A7 based dual core processor. The speed of the raspberry pi 2 is up to 1 GHz. This computing device also compatible to windows 10.

In this work raspberry pi2 module is used. The OpenCV libraries are install in the raspberry pi2 computer. As raspberry pi operating system is based on Linux. The command line execution is possible in the raspberry pi. The C, C++ and python code can run on the raspberry. In this project the C++ coding is used with including Open CV library files. On raspberry pi 2 there are Ethernet connection, separate slot for the raspberry camera and four USB port. Hence it makes easy to interface the raspberry camera or normal USB web camera. Also the raspberry has Ethernet cable which allow the user to operate the raspberry on user computer connecting it with Ethernet. Raspberry have its own server, or it can right the raspberry pi is itself a server. Hence the data from raspberry can send to the other user over net using raspberry pi server.

3.2.2 Telegram

Telegram is the android software which is use to send the data between user over internet. Telegram is able to send the text as well as photo. In this project the information regarding the free lot or lot status. This information is send from the raspberry pi to user mobile[2,6], as shown in the Figure 6.

Telegram should install on both user and raspberry pi module to send the information. In the software coding the shell script is used for sending message automatically[7].

Figure 6. Status of free slot on mobile app.

4. Conclusion

In this modern scenario of increasing numbers of vehicles, the trouble is to find the free parking slot to park the vehicle in crowded area. Vehicle parking management using Raspberry pi is one such project which shows the user available parking slot and the exact position of the slot in the parking lot. This work has been completed using minimum available resources and minimum cost. The vital advantage of this vehicle parking management system is that it does not require any sensors at all. If any obstacle other than car is present it does not detect it as a vehicle and shows it as a free slot. Now a day everyone having mobile and the output of this project is available on mobile application, so anyone can assess and get information about free parking slot without any cost. If user knows the free parking slot the driver of the vehicle directly approaches to relative parking slot and it save the time of user. This method can be easily implemented in a short period of time and does not require regular maintenance.

5. References

1. Cervantes. A hybrid middleware for RFID-based parking management system using group communication in overlay networks. Intelligent Pervasive Computing, IPC, The International Conference on lloilo; 2007. p. 521–26.
2. Backhouse J, Cohen J. What is a smart city for information systems research in Africa? Review protocol and initial results. African Cyber Citizenship Conference on Port Elizabeth, South Africa; 2014 Nov 5–6; 2014.
3. Kharusi HA, Bahadly IA. Intelligent parking management system based on image processing. World Journal of Engineering and Technology. 2014; 2(2):55–67.
4. Yu H. The design of the embedded wireless vehicles monitoring management system based on GPRS: Evidence from China. International Symposium Intelligent Information Technology Application Workshops, USA; 2008. p. 677–80.
5. Banerjee S. Real time car parking system using image processing. 3rd International Conference on Electronics Computer Technology, Kanyakumari. 2011; 2:99–103.
6. Longfei W. Integrating mobile agent with multi-agent system for intelligent parking negotiation and guidance. Industrial Electronics and Applications. 4th IEEE Conference on, Xi'an, China; 2009. p. 1704–7.
7. Jin J. An information framework for creating a smart city through internet of things. Internet of Things Journal IEEE. 2014; 1(2):112–21 .