Improved Parking Solution Based on Image Recognition Algorithms

Jovana Sofronioska, Boban Temelkovski and Jugoslav Ackoski

April 6, 2022

Improved Parking Solution Based on Image Recognition Algorithms

1st Jovana Sofronioska
Military Academy "General Mihailo Apostolski"
Skopje, Macedonia
jovana.va1351@student.ugd.edu.mk

2nd Boban Temelkovski
Military Academy "General Mihailo Apostolski"
Skopje, Macedonia
boban.temelkovski@ugd.edu.mk

3rd Jugoslav Achkoski
Military Academy "General Mihailo Apostolski"
Skopje, Macedonia
jugoslav.ackovski@ugd.edu.mk

Abstract— The rapid increase in vehicle population in recent years have given rise to a number of global problems such as air pollution, blockage of roads, waste of fuel and time. Today car drivers struggle to find a free parking space for their cars.

Based on the above, people expect technology to address this problem and hence find a suitable solution to end the suffering of motorists. On the other hand, IoT is a promising platform for providing unusual applications for helping people and provide them a better way to manage their day's lives. Control systems are replacing manual operators and fully automated machines are replacing human labor.Even if many of the existing parking systems provide passive information to the drivers regarding the availability of parking slots. This paper proposes a solution for making this problem easier by using build-in tools and library in the matlabsimulink opera and will address the issue of the size of free parking spaces taking into account the dimensions of the vehicle.

Keywords—*vehicle, algorithm, camera, dimensions, parking space.*

I. Introduction

One of the main concerns in most cities are parking problems who play a major role in any transport systems. This problem leads to an increased usage of fuel and that affects the environment and air pollution.

The main reasons for the problem of difficult parking are as follows the current urban parking spaces are in short supply; more importantly, in the process of searching for parking spaces, people can get less valuable parking space information; and there is a lack in parking space information management platform to guide vehicle drivers to park reasonably.(1)

Automated parking identification methods can play a major in achieving this goal, which can be useful for drivers, parking lot owners, and the environment. Parking detection methods can either determine the occupancy of an entire parking lot or identify the occupancy of every parking space. The objective of this research is to show how the dimensions of the vehicles affect the finding of the most appropriate free parking space.(2)

In this example we aim to make easy development effort by using the matlabsimulink opera. We start this parking example by including image processing techniques for finding empty parking spaces as well as calculating the dimensions of the vehicle. For simplicity we use pre-decoded data to substitute for localization information.

Counting Available Parking Space using image Processing helps to solve the problem that the driver faces at low cost. This system uses image processing to detect the existence of the car and also provides information such as number of available parking space. The system captures image using CCTV cameras and processes the image to count the available parking space. The development of this system will use techniques of image processing that will be implemented in each phase of the methodology. This system gives information about the number of available parking space. It will provide benefit to all the drivers when they enter the parking lot. The system uses image processing, since the whole area in the parking lot can be observed with relatively few cameras. Other than that, the system is compact and the cost is not high. The image of a parking lot is taken by a surveillance camera set at some height in the parking lot. MATLAB is used as software platform in this project.(3)

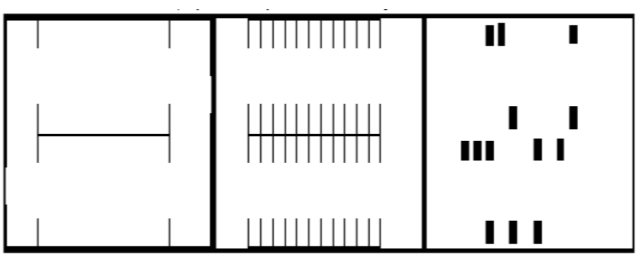
The logistics industry uses boxes for protecting, loading, and packaging a plethora of items. To improve the efficiency of these actions, technology for box dimension measurement can be of high value. When developing measurement technology for logistics use cases, three important requirements have to be taken into account. First, the technology needs to be deployed in an environment that is continuously operating, with boxes passing one after another over conveyors at a constant speed. If it is necessary to stop a conveyor for measuring the dimensions of a box, then a lot of time and energy may be wasted. Second, the technology needs to be deployed in a low-speed environment. Indeed, people typically need to intervene at some point in time, resulting in a limitation on the conveyor speed that can be utilized. Third, boxes have been designed to protect their content, allowing for deformations in order to absorb the impact of shocks. In practice, this implies that a certain error tolerance is allowed when performing dimension measurement.Therefore, in that research and development effort, they have decided to make use of a widely available RGB-D camera for box dimension measurement, and where this type of RGB-D camera is also frequently used in the area of computer vision. Specifically, in this paper, they present Box-Scan, a hierarchical algorithm that can perform box dimension measurement on a conveyor in real time, making use of a single RGB-D camera, a common desktop computer, and some fixing equipment.(4)

II. PARKING VALLEY

A. Environment model

The environment model represents a map of the environment. Occupancy maps are a common representation for this form of environment model. The parking lot example is composed of three occupancy grid layers.

- Stationary obstacles: This layer contains stationary obstacles like walls, barriers, and bounds of the parking lot.
- Road markings: This layer contains occupancy information pertaining to road markings, including road markings for parking spaces.
- Parked cars: This layer contains information about which parking spots are already occupied.(5)



Map Layers – Stationary Obstacles, Road markings and Parked Cars

When we combine into one layers mentioned above, they give us a complete picture of the parking. This includes identifying occupied and empty parking spaces.

III. MATHEMATICAL BACKGROUND OF ALGORITHM

These are the main steps on which the algorithm of this system is based:

A. The camera identifies the vehicle entering the parking lot.
B. Classifies it in the table according to which type it belongs to.
C. Captures the images of the car when it enters or leaves the parking lot.
D. Parameters of the vehicle are calculated according to which are the most appropriate free space for parking.

A. Identification

At the moment when the vehicle is approaching the parking lot, there is a camera at the entrance which is placed in order to take the necessary information about its dimensions. The vehicle is identified by image recognition.

There is a process called ROI which takes the form of a line of points, both in the horizontal and the vertical direction, and where this line-shaped ROI can be used to perform dimension measurement. A point p (u, v) ∈ ROI, located on the conveyor surface, is stable at first. This means that the distance between the ROI point $p_t(u,v)$ in the current depth frame t and the same ROI point $p_{t+1}(u,v)$ in the depth frame t+1 is smaller than$\in_d$. The threshold $\in_d$is used to determine whether an object is present or absent on the conveyor surface below the RGB-D camera. If an object passes through the ROI, some of the points in the ROI will have a distance difference that is greater than $\in_d$. We say that these points are in an *excited state*. We subsequently define s_t as the state of a point at time t:

$$s_t(u,v) = \begin{cases} 1(exited), & if\ d(p_{t-1}(u,v), p_t(u,v)) > \in_d \\ 0\ (stable), & otherwise \end{cases}$$

In the above equation, d denotes the Euclidean distance between two points. Furthermore, we represent the number of excites-state points in the ROI as $N_{excited}$:

$$N_{excited}(ROI) = \sum_{(u,v) \in ROI} s(u,v)$$

Assuming that an object passes through the ROI, we can then leverage Equation (2) to define the set E , whichcontains the excited-state points, and Equation (3) to define the set S, which contains the stable points, at time t:

$$E_t = \{p_t(u,v) : s_t(u,v) = 1, p_t(u,v) \in ROI_t\}$$

$$S_t = \{p_t(u,v) : s_t(u,v) = 0, p_t(u,v) \in ROI_t\}$$

Furthermore, we introduce the following notation: $N(ROI_t) = |E_t| + |S_t|$.

In addition, we represent the initial set of points by means of S_0 , with this set having the following property: $N(ROI_0) = |S_0|$ (this is, all points in the ROI are initially stable).

Algorithm 1 ROI-measurement

Procedure ROI-Measurement()

height ← 0

width(length) ←0

D← Ø

//height by averaging distances

for all $p_{E_t}(u,v), p_{S_0}(u,v)$ *where* $p(u,v) \in E_t$ **do**

D ← Distance $\left(p_{E_t}(u,v), p_{S_0}(u,v)\right) \cup D$

end for

height ← $(\sum_{d \in D} d) \ / \ |D|$

//width (length) by finding maximum distance within E_t

E_r ←RANSAC – LineFitting (E_t)

for all $p_i \in E_t$ **do**

for all $p_j \in E_t$ **do**

width (length) ← Max (width (length), Distance (p_i, p_j))

end for

end for

return width(length), height

end procedure

Procedure Distance (p,q)

Convert 2-D points p and q into 3-D points p' and q'

return Euclidean distance between p' and q'

end procedure

Using the equations mentioned above, we present the ROI-driven measurement of the width and height of a vehicle in Algorithm 1. As will be further clarified in the next section, we use the same approach for determining the width and length of a vehicle.The procedure *ROI-Measurement* takes as input the set E_t of excited-state points and the initial set S_0 of stable points, calculation the width (length) and height of a vehicle. The procedure *Distance* takes as input a point p from the set and a point from the set E_t and a point q from the set S_0, returning the 3-D Euclidean distance between the two given points, after having converted the 2-D points p and q into the 3-D points p and q.For calculating the height of a vehicle, the *ROI-Measurement* procedure makes use of an average. For calculating the width (length) of a box, the *ROI-Measurement* procedure makes use of an iterative approach to find the distance between the farthest two points, deploying RANSAC-LineFitting to eliminate outliers when creating straight lines.(4)

B. Classification

The next step on which the algorithm should work is based according to a specially designed scheme that classifies vehicles in tables. These tables take information about the dimensions of the vehicles from the camera and the system stores them in a group according to the type. These tables should consist of information about what type of vehicle is entering the parking lot and based on previously generated data should find out its dimensions for width, height and length. In addition there is an example of this table:

TABLE I. EXAMPLE TABLE

Number	**Table**			
	Type of vehicle	***Width***	***Height***	***Length***
1	Audi R8	1904mm	1230 mm	4426 mm
2	Van Hool CX35	2590 mm	3505 mm	10845 mm
3	Yamaha YZF-R3	720 mm	1135 mm	2090 mm

C. Monitoring

The whole parking lot is constantly monitored in order to know which parking space is empty and where a vehicle has already been parked and what size it occupies, according to its dimensions.With the help of Image segmentation module which converts the RGB image from the camera, into binary image,this search for empty space can be made possible.

The RGB (Red, Green and Blue or Colored Image) image acquired from the camera is then converted to gray-scale image and then binary image is created in the Image segmentation module. The equation used for the conversion to gray-scale image is:

$$Gray=0.229R+0.587G+0.11B$$

The gray scale image of the parking space with cars from the resulting gray-scale image, binary image is obtained using threshold technique. The binary image contains all the information about the position and shape of interest. The threshold level is set in such a way that objects of interest are made into white and the rest of the image black.(3)

The converted binary image will contain some amount of noise and trace the boundary of the object. In orderto remove the noise which the image has got from the variety of sources will then be removed using morphological operation namely dilation, erosion, etc. The operation is basically used in the most of the image processing operations. In a binary image, if any pixel value is 0 then output is 0. Here the picture is enhanced by adding pixels to cover the holes and the pixels are removed to remove unwanted objects.(6)

In order to detect the cars, blob analysis is done using predefined functions in MATLAB and the number of cars is counted. After the execution of the code there are two output figures. One is observation of the whole parking area, second one is observation of the each lane in parking area and the vacant lane is shown.

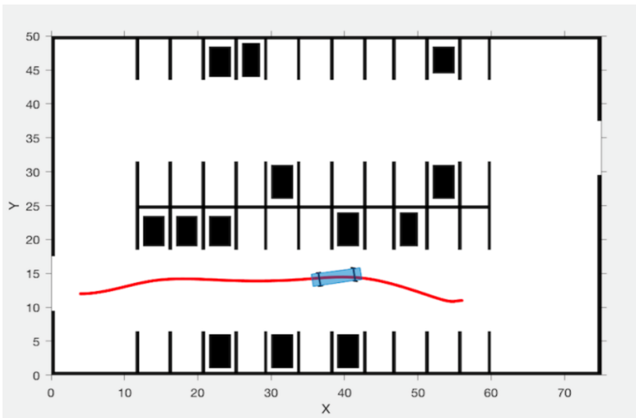
D. Comparing sizes and Executing Complete Model

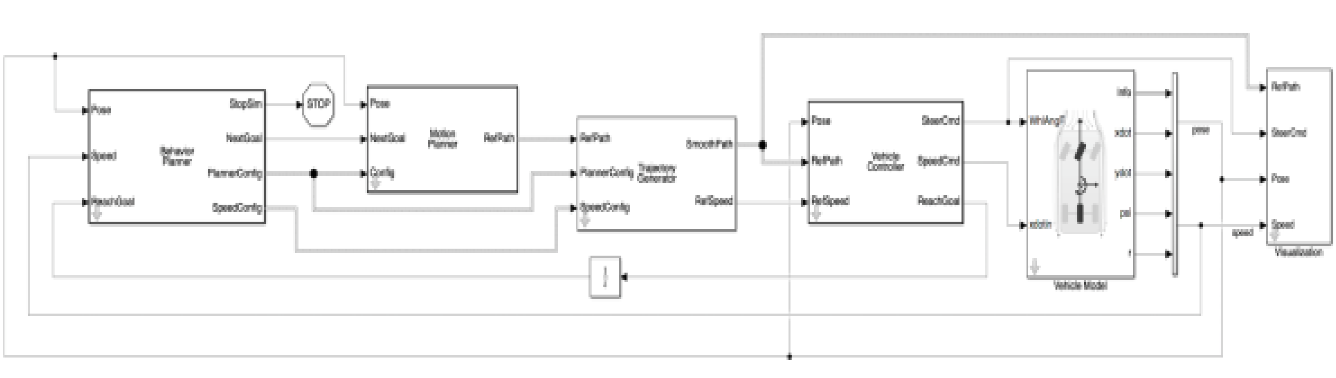
Once the empty parking space is found, it is calculated whether the size of that parking space corresponds to the size of the vehicle.It is taken into considerations that the found empty parking space should be larger than the vehicle itself. When this is done, the model should provide proper navigation to the vehicle.

As described until now automated parking valley application is designed to be implemented as a single model using the design principle based on a model for image recognition. The partition in this application into different sub-components is moral. Since the model already contains functionalities divided into different subsystem it is easy to partition this existing model for the purpose of the deployment. The single parking valet model now can be broken out into two separate models to facilitate deployment as a distributed system. The first part covers recognizing the type of vehicle, and the second part covers finding an empty space.

Before deployment to the target, this should simulate a whole entire system to verify each performance one again. In here we use modern reference for each functional module in one single simulink model. In another word, image recognition algorithm and ROI algorithm all have their individual model references information exchange between models. We can than deploy each model as a standalone node learning on actual vehicle and replacing simulation by real time actual vehicle.

At the end of the whole simulation, this is the outcome:





In this photo we are shown a vehicle that has already entered the parking lot. With the help of the camera placed at the entrance and according to the image recognition algorithm with help of ROI it is placed in the table where we know all its dimensions. Next an empty parking space was found and dimensions of the vehicle were calculated compared to those in the environment model. Now the scheme is marking the direction of the path with the red line to the empty parking space and leads the vehicle there.

ACKNOWLEDGMENT

I would like to send a special greeting to the Military Academy “General Mihailo Apostolski”, which supported my system with which I want to contribute to solving traffic problems, in order for drivers to rationally use the parking space.

REFERENCES

1 Y. Yang, Design and Implementation of Intelligent Parking Guidance System, Jilin University, 2018.

2 Sh. Khaled, H. Tounsi, Journal of Traffic and Transportation Engineering, in press

3 Park H., Messem A., Neve W., Box-Scan: An efficient and effective algorithm for box dimension measurement in conveyor systems using a single RGB-D camera, 7th IIAE International Conference on Industrial Application Engineering, 2019.

4 Matlab, (2019). Available on: https://www.mathworks.com/help/driving/ug/automated-parking-valet.html

5 Smart Parking System (2020). Available on: https://github.com/abhishekapk/SMART-PARKING-SYSTEM

6 Rajani Z., Viegas A. M., Smart Parking Space Detection Using MATLAB an Internet of Things, 2017.