SpotPark

Embedded System to Detect Free Space on Streets

Lucas Santos
Universidade Federal de Pernambuco
lhcs@cin.ufpe.br

Renato Sousa
Universidade Federal de Pernambuco
rsb5@cin.ufpe.br

Roberto Fernandes
Universidade Federal de Pernambuco
rcf6@cin.ufpe.br

Edna Barros
Universidade Federal de Pernambuco
ensb@cin.ufpe.br

ABSTRACT

Traffic on streets starts to be a normal situation around the world. People get used to sit and wait for hours inside the car, attempting to reach their destination. However the difficulty of find a free spot to park increase this problem. Researches made by TomTom - Traffic Index (2016) shows that, out of the 10 worse cities to drive around the world, 3 of them are Brazilian. According to Donald Shoup (2006), professor from the California University, 74% of the traffic on streets is caused by people searching for a place to park. SpotPark comes to decrease the volume of traffic and all the problems related to it. It simply makes it possible to know where the nearest parking spot to your destination is, even before leaving the house. The module installed on the poles turns the city into a smart city. Once you gather the information collected around the poles, it is possible to discover where the available parking spots are and its size. Discovering these places, the system users will know where to park right away, saving time and helping the city.

KEYWORDS

park, spotpark, pollution, traffic, carbon dioxide, smart city

1 INTRODUCTION

According to a research made by the maps application TomTom - Traffic Index 2016 [1], on the top ten worse city traffic, Brazil has 3 cities. This leads the population to a high level of stress, air pollution and a huge waste of time. The time wasted in traffic is not only related to car congestion, red lights and long paths to overcome inside the city. Many researches [2] estimate that drivers waste, on average, 100 hours every year looking for a place to park their cars. This means around 10 minutes per day, driving at only 9 kilometers per hour (km/h), enough time to bother and slow the traffic down.

So parking causes a mobility problem for the cities around the world, and Donald Shoup, a professor at the California University, made a study called "Cruising for Parking" in 2006, that according to it, these drivers are responsible for up to 74% of congestion in big cities [2]. In the same research, Shoup suggested the private parking is one solution, as it enables to guarantee spots to park.

Detect cars on streets is not simple as it seems. Around the world, there are a few tries to map the free spots to park in public zones. Companies like the Brazilian Serttel have already done a system to identify spaces to park using computer vision. However, obstacles such as lighting and the use of markers on the street to define the space where a car can be parked can turn the project impractical. Another project to detect parking spot was announced by Siemens with Intel partnership. Anyhow, nothing more than a video showing its usability and its benefits for the environment was announced by the companies. Consequently, no technologies were delivered or tested on streets yet.

In order to find a solution, it was necessary to unite the laser distance sensing technology, once it is not influenced by luminosity and has a long range, with precise engines was necessary. Allowing to scan the ground around the detection module. Despite of that, this solution requires precise control of engines and fast sensor response. The use of a pivotable support causes problems with the dependent engine's movements, so if only one moves, both coordinates on the ground changes. This engines set creates difficulties to parameterize the movement of the point on ground. The sensor ends up doing a reading in a semi-elliptical line, leading to a far more distant than the desired points on a straight line.

To accomplish SpotPark goal, the system needs to find a parking spot closer to user's destination using sensor modules strategically placed on the streets, generates enough data in order to avoid wasting time searching for a spot. So, it's possible to increase the average speed of cars on the road, making traffic run smoother. Although people thinks that increasing cars average speed is just a comfort, it helps to prevent health and economic issues, since they are related with greenhouse gases emissions [3] [5].

2 SPOTPARK SYSTEM DESCRIPTION

2.1 Project Description

In order to achieve the project's goal, was proposed to use the spot detection modules distributed strategically throughout the city, so it monitors public parking spaces. Each of these systems includes a sensor module, a process unit for the algorithm that calculates the amount of free spaces and each size, and a communication module to combine the data in a server. The user enters his destination and car type in a mobile application and then the server returns the amount and the location of free spots near to its destination, as shown in figure 1.

An important point of parking spots in public areas, such as streets, is the not well-defined spot, which could hamper the space detection. Nevertheless, the proposed system does not supervise specific positions, it supervises the free spaces, as shown in figure 2. With this free space and the size of a car provided by the user, it

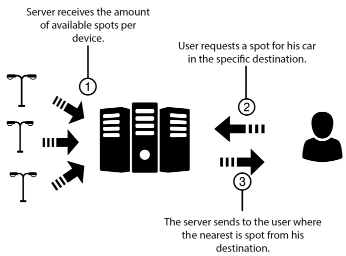

Figure 1: System overview

is possible to determinate the amount of parking spots in a given location.

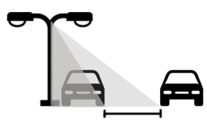

Figure 2: Detecting free spot

2.2 Modules Description

The system was divided into two main parts: the detector module, which is spread through the city on poles, and the central module, which is connected to a computer.

2.2.1 Spot Detector. Having the sensor data module and the communication module, the parking spot detection module generates a map with all the available heights around the sensor. Despite all readings, two factors that may contribute to error: the amount of noise and the inaccuracy in the linearization. The noise is produced by the loss of some readings, and also the sensor inaccuracy. With the relief map, it is required to do a discretization in it, that means, turn it in a map that displays if a certain point is free or not. The first method used was a threshold, with average height of all cars, to set if some point was occupied or not. This first approach didn't show good results because the height difference of a spot with car and an empty one were subtle, when measuring for a distant point. The second method used was a supervised machine learning algorithm.

2.2.2 Sensor. This module has a laser distance sensor, with 40 meters range, coupled to a pivotable support, see figure 3, which allows the sensor to move in two axes, to be possible to cover the two sides of a road. The sensor in constant motion to map the distances of the objects occupying the roads and sending the calculations of all heights to the detector module.

2.2.3 Communication. This module is responsible for sending the number and size of each space provided by post to the server, and it's based in mobile technology (GSM). A protocol for this communication has been developed so that it could take place safely and reliably. Headers were implemented in order to differentiate users from the detector modules and also to discriminate the desired service.

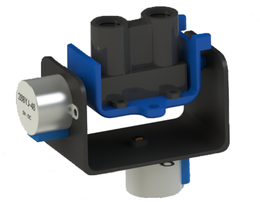

Figure 3: Pivotable support with the laser sensor

2.2.4 Server. In order to manage the information generated by the system installed on multiple posts, and sent by GSM technology, a server is needed to interconnects the multiple devices to the user in internet. So, every spot detector has its information concentrated in the server. When a user makes a spot request with his destination, the server answer where is the available spot closest to the user destination.

2.3 Implementation

To implement each module some technologies were need to be chosen, due to their advantages. First the pan tilt of the sensor module was built using two stepper motors, with their drivers, and 3D printing parts. For the detection of parking spots it was used algorithms adopting the sensor angle and a machine learning technique. The communication between the parking spots detection module and the server is made of GSM technology and uses text messaging (SMS). In order to connect all modules and to keep the compression module, a circuit was designed and made on a milling machine. An overview of module can be seen in figure 4.

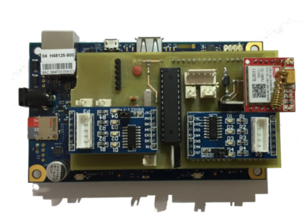

Figure 4: Overview of spot detect circuit

2.3.1 Engine Control. The engines used in the pivotal support were stepper motors, after all, they are widely used for systems that require precision and they are fundamental for pivoting module. Stepper motors are built with several magnetic poles, that defines the amount of steps that an engine can perform in a rotation. The engines used were 28BYJ-48, which allows a control of approximately 0,088 °. In order to control them an engine driver has been developed, with an ATmega328 controller coupled with seven Darlington transistors, as it can be seen in figure 4.

The driver controller, communicates with a Galileo board via I2C protocol. To reach the desired angles, the driver consider the current position of the engines and precisely moves the engines to the desired position. In the engine controller there are two end cursor sensors, so the drivers resets the engine position periodically, ensuring security angle control. The data sent for the driver are always obeyed, otherwise, the driver will inform Galileo an error in the engine, this it will request maintenance.

2.3.2 Spot Detection. In order to detect the spots, the module running in Galileo synchronously sends the objective angles to the driver. After receive the readings of the distances and their respective opening angles, the detection module makes a linearization process, obtaining the actual sizes of all obstacles. With the actual heights, it is possible to see the car projection in the blue line in figure 5. However, in order to design an algorithm that precisely identifies the existence of a car, was necessary to use a logistic regression, the algorithms learn to identify more accurately the cars. Logistic regression is a statistical technique that aims to produce, from a set of observations, a model that allows the prediction of values taken by a categorical variable (binary) from a series of explanatory variables continuous and/or binary. The explanatory variables used were the height of a point, the engines angles and the height points near to its. Thereby giving a better prediction. Even more, it begins to ignore momentary obstacles such as pedestrians passing by the time the scan is being performed.

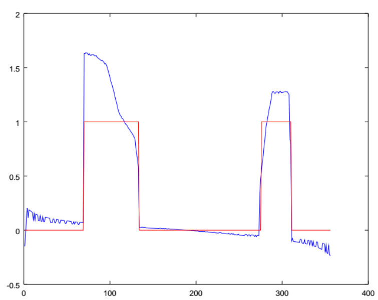

Figure 5: Linearized sensor readings and threshold

2.3.3 Spot Size. In order to find the actual distance between obstacles in a lane, a series of trigonometric relationships involving the first and last point of a free spot is made, as shown in figure 6. Once you have the opening angle from the pole sensor, it is possible to find the projection of the triangle on the ground. Having two projections, is possible to find the size of the space between them with the subtraction of projections, resulting in the real space of a parking space as $R_1 \times \sin\theta - R_2 \times \sin\alpha$.

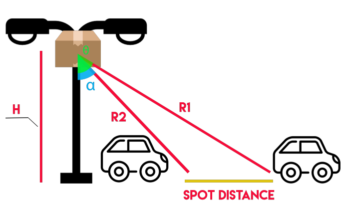

Figure 6: Spot size calculation

The vertical axes are in real scale, but not horizontally, since reading distances are made at predefined angles. The greater is the difference between a reading angle and the next one, the more separated will be the readings. It is possible to see in figure 5 the relief map of two vehicles, and the car farther away looks bigger, because the reading were made with larger angles.

A simple diagram of the algorithm detection module is found in figure 7. First Galileo receives the distances from the sensor, then it made the linearization of the scanned distance. With a machine learning technique obstacles are discretized, to facilitate the next step, the spot size calculation.

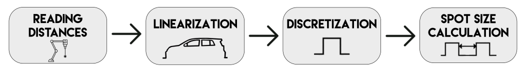

Figure 7: Detector module diagram

2.3.4 Communication Module. For the communication module, located with the detector module on the pole, it was used a SIM800L, a GSM module, attached to the Galileo shield. In order to perform all communication, an AT protocol was chosen and used, by its simplicity and ruggedness. Communication is done by sending text messages (SMS), it was chosen for its wide coverage area and because some poles does not have good internet connection. To ensure system reliability, it was created a communication protocol where the detection module only sent an update command of the amount of free spots, their sizes and a periodic command to confirm if that is still active.

2.3.5 Server Module. For the server module, it was adopted one shield GPRS / GSM, which uses SIM900L module that communicates with a Arduino Mega through a serial port, and send data to computer by another serial port. With this architecture, the amount of available parking spots, and its sizes, arrives on server by the Arduino, and the users requests by internet. In order to ensure the safety of the system, the data on server is only accepted if the phone number that sent was already registered on the server. The message of a registered device on the server, is saved on a local database, at the device's localization, keeping the number of free spots, and its sizes. So the server will receive from the user his destination and type of his car, via a mobile application. With these information the server checks the nearest spot available that fits the user's car, and answers where and how many spots are there.

2.3.6 Mobile Application. In order to facilitate user interaction with the system, a mobile application on the Android system was developed. The user registers in the application choosing the type of his car and, after the registration, picks on the map its destination like Google's map. After receiving and processing this request, the server sends to the application the location and number of spots available within tree blocks. Then the user can authorize the SpotPark application starts a navigation in a map application. When the user confirms it, the system will inform any changes in the availability of parking spots in the desired location. If there aren't any parking spots on this location anymore, the application will suggest the nearest place of destination with available parking spots. The update rate is due to the devices updates, that takes almost 2 minute. So, if spots changes, the device will send the new

Table 1: Results comparison

Item	Real Distance (meter)	Simulation Distance (meter)	Prototype (centimeter)
First Free Spot	3.0	3.92	36.8
Unavailable Spot	2.5	2.29	23.8
Second Free Spot	3.0	3.71	38.9

configuration to the server, and there, the new scenario will be sent to the users application section. If the actual destination is filled, a new one will be necessary, on a future version the application will suggest to change Google Maps destination automatically.

2.4 Validation

In order to previously test and validate the spot detection algorithm, a simulator V-REP was utilized, widely used in robotics projects. This allowed to simulate the detection module even before having some hardware. With the help of the simulator, it was possible to create multiple scenarios and situations, see figure 8. In place of validating all the hardware and software integrated, a model was built always obeying the 1:10 scale.

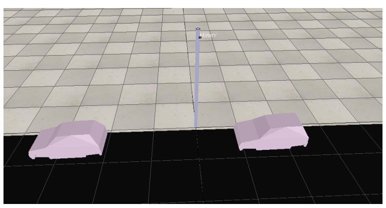

Figure 8: Simulation scene on V-REP

3 PERFORMANCE AND ANALYSIS

It was necessary to analyze multiple aspects. However, two of them are essential to the well functioning of the project. The first was if the distances calculated by the system approximates the actual distances. The second point, the machine learning and efficiency to detect whether a point on the scan is occupied or not. In order to analyze the accuracy of the prototype, two tests scenarios were made before analyzing in a real situation. One virtual environment with the V-REP simulator and the other using a prototype in the 1:10 scale. To validate the detection of available spots provided by the machine learning, it was held a prior training with four situations so that it could finally make the tests with other situations. This allowed to assess the real situation with the probability distribution generated by the algorithm. The chosen scenario was a vehicle center aligned to the pole and the desired result is to find two large openings, separated by an occupied space (a car), this result needs to be equal on simulation and scaled scenario. After collecting the results in both environments, the table 1 was made. And as you can see, in both, simulation and prototype the situation was recognized, the difference on the distances exists but is barely can be considered as a free spot.

Several bar graphs were generated, illustrating the probability distribution with all the read points. The dashed line (threshold) indicates when is regarded as an unavailable spot (50%). The situation used to illustrate is the same as in figure 9. The result was very satisfactory, since the occupied points are really above the threshold and the available spots are at a safe distance limit. To raise the accuracy is necessary to use more training cases.

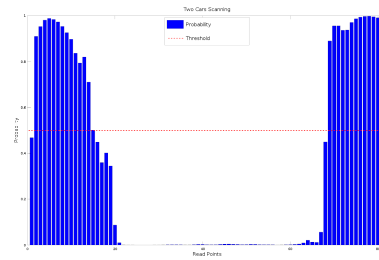

Figure 9: Spot occupation probability distribution

4 CONCLUSIONS

The issues caused by traffic goes further than loss of time, air pollution and money. It affects and degrades the welfare of the population. With this project it's noticeable the possibility of detecting parking spots on public roads, in a cheap and efficient way.

The SpotPark being used in pole makes possible to map the position of the obstacles around it, so it's allow to map free spots. Due to the difficulty in the installation of the sensor on poles, a scale model of 1:10 was created to hold most of the tests. The results have shown 11cm of precision between two free spots, it was possible because of the use of methods to predict as machine learning. It is a project that allows the system run in a whole city, or neighborhood. Besides the performance test, crucial ones are performed in real scale, confirming the effectiveness of the system. According to the statistical references and data collected through the project, investment on the system can be converted. After all, the traffic reduction provided by the project would save billions dollars annually.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Edna Barros to help with all the project's development. The authors also thanks the Centro de InformÃątica (CIn), that provided structure and materials, and to Dr. Hansenclever that helped with the simulation process.

REFERENCES

[1] TomTom Traffic Index 2016. Available at: http://corporate.tomtom.com/releasedetail.cfm?ReleaseID=961546. Last access in October 29, 2016
[2] Cruising for Parking. Available at: http://shoup.bol.ucla.edu/Cruising.pdf. Last access in October 29, 2016
[3] https://trid.trb.org/view.aspx?id=404838
[4] Traffic Congestion, Driver Stress, and Driver Aggression. Available at: http://onlinelibrary.wiley.com/doi/10.1002/(SICI)1098-2337(1999)25:6<409::AID-AB2>3.0.CO;2-0/abstract. Last access in October 29, 2016
[5] The Cost of Air Pollution: Strengthening the Economic Case for Action. Available at: https://openknowledge.worldbank.org/handle/10986/25013. Last access in October 29, 2016
[6] HavardMedicalSchool; Too much sitting linked to heart disease, diabetes, premature death. Available at: http://www.health.harvard.edu/blog/much-sitting-linked-heart-disease-diabetes-premature-death-201501227618. Last access in October 29, 2016