Automated Parking System

Omkar Hatwalne
Computer Science
State University of New York
at Binghamton
ohatwal1@binghamton.edu

Aakash Chavan
Computer Science
State University of New York
at Binghamton
achavan4@binghamton.edu

ABSTRACT

Nowadays in many multiplex systems, there is a severe problem for car parking systems. There are many lanes for car parking, so to park a car one has to look for all the lanes. Moreover, there is a lot of men labor involved in this process for which there is a lot of investment.

Conventionally, car parking systems do not have any intelligent monitoring system. Parking lots are monitored by human beings. All vehicles enter into the parking and waste time searching for a parking slot. Sometimes it creates a blockage. The condition becomes worse when there are multiple parking lanes and each lane has multiple parking slots. The use of the automated system for car parking monitoring will reduce human efforts. So the need is to develop a system which indicates directly which parking slot is vacant in any lane.

In this paper, we propose an idea of an automated car parking system. We propose a system that involves an infrared transmitter and receiver in every lane, a centralized server that collects the data from all the IR sensors and a mobile application that displays users the information about the available slots. So the person entering the parking area can book a slot through a mobile app and can decide which lane to enter to park the car

CCS Concepts

IR Sensor → Centralized server which processes the Information → forward it to all connected Mobile App

Keywords

Processing Unit, Co-Coordinator, Android Application, TCP Protocol.

1. INTRODUCTION

When Over the decade's many countries have developed drastically, now we are in this state that we have a lot of well-contacted roads, commercial buildings and an increasing number of automobiles. While parking these automobiles in parking space we use the manual procedure of parking.

When looking for a parking spot in a large parking lot it could take a lot of time to find an opening parking space. The traditional way when looking for a parking spot is searching each row and column until the driver finds a spot. This approach is linear because the driver starts at the beginning of the parking lot and searches until he finds a spot or until he reaches the end of the parking lot. In the case that there isn't a spot in the parking lot the driver would have wasted his time searching through the whole parking lot. We thought that there had to be a way to solve this problem using sensors and integrating it with a graphical user interface.

Our solution will be reducing the amount of time searching for a parking spot because we implemented a series of sensors and display the information on the Mobile Application. The availability of parking slots can be checked through our Mobile application and do the reservation for a particular slot. It will greatly reduce the amount of time for users to find a parking spot. Our approach is different compared to the traditional parking lot sensor system. In a traditional parking lot sensor system they have a display before the user enters the parking lot, so the user can see an open spot on the display and drive to the open parking space. There's a major problem with this method because if a driver is currently going to the open parking space and another driver pulls up to the display, they will think that the spot is open. When in reality the open spot is about to be occupied. This is a timing issue because we have one driver heading to the open spot, while we have another driver thinking that the spot is open. We thought of this problem and decided to develop an application so that if this problem would occur the second driver would just have to look at the application for another free parking space.

2. DESIGN

The approach we took in designing our system was to divide it into three modules and those modules are parking IR sensors, Processing Unit and Mobile application. We decided to use this modular design because it made it easier for each person to work on each part without affecting the other module. Then we just need to connect the modules together with standard protocols like TCP for communication.

3. IMPLEMENTATION

The implementation of the “Automated Parking System” is three-fold. The IR Sensors, Processing Unit (RaspberryPi), and a Front-end (Android mobile application). Following sections explain the implementation of each part.

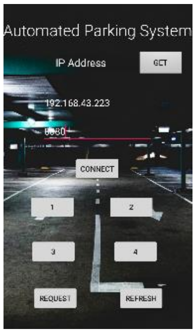
3.1 RaspberryPi

The RaspberryPi is a very important component of the system. Multiple IR Sensors are connected to RaspberryPi over GPIO ports. Each parking slot has its dedicated IR Sensor. RaspberryPi is also connected to Mobile Application over TCP socket. We have used C++ to program RaspberryPi. We have used wiringpi library which lets you access the RaspberryPi GPIO through C/C++ code.

3.2 IR Sensors

IR Sensors detects an obstacle present in front of it (in our case it's going to be a parked car) IR sensors are connected to RaspberryPi on GPIO pins. Each parking space will have its own IR Sensor.

3.3 Mobile Application



Automated Parking System

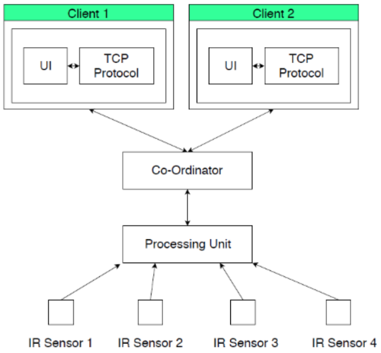
IP Address

192.168.43.223

8080

Mobile Application will connect to RaspberryPi using TCP protocol. On the welcome screen, there is a provision of setting information like IP & Port No. Mobile Application displays availability of parking slot in for of buttons. If a button is enabled it signifies that the corresponding booking slot is free & if it's disabled it signifies that the chosen slot is already occupied.

4. System Architecture



As shown in the figure above, our system is a single server-multi client system. It can processes requests from multiple clients at the same time. Each mobile application (a client) will try to connect to RaspberryPi (server) with IP & port number using TCP protocol. Once the connection is established client can request for status information from the server. Later the mobile application (client) can send a booking request for a slot which is free. Server code running on RaspberryPi will process the booking request and it will send an acknowledge back to the client.

Server code running on RaspberryPi has two units.

1. Processing Unit: It fetched the status of IR sensors which are connected over GPIO ports. Processing Unit receives two types of requests, a Status request and a Booking Request. On receiving the Status request Processing Unit reads all IR sensor values & forms a message packet and sends it to the client as a response. On receiving the Status request Processing Unit checks the availability of requested parking slot, if it's available it books the chosen slot for the user and sends the “Booking: Success” message in response. If the slot is already occupied then it sends the “Booking: Failure” message in response to the client.
2. Co-ordinator: Co-ordinator checks the loads on each Processing unit instance periodically. It selects the least occupied instance of processing unit and forwards the request to that node.

5. EVALUATION

We have successfully evaluated our project by using 4 IR sensors, Raspberry pi, and the Android application. Each sensor is planted at the parking lot to indicate the availability of it. We tested it with three scenarios:

A. Successful Booking parking slot: with the help of IP address and a PORT number we get the status of the parking slot, which slots are available and which slots are blocked. Based on available slots the parking number slot button is enabled/disabled
B. Time out: If the user book the parking slot and does not arrive in a certain time (will be checked using IR sensors) the booked slot will be available on an android application to all users. Right now, time out is 15 seconds, it is flexible and can be changed easily as per requirements.
C. Realtime status of slots: If the user fails to refresh the available slot and accidentally tried to book the same or different slot which is already occupied, then the booking will be unsuccessful. A suitable message will be displayed on the android device

YouTube Link - https://www.youtube.com/watch?v=jnc7pJVHnps&feature=youtu.be

6. CONCLUSION

In this paper, we proved that Automated Car Parking is a very useful utility for booking the parking slot with real-time status checking and time out feature. User-friendly Android UI makes it very easy for a user to make the reservation. The process that was involved to create our solution started by formulating the problem of car parking and hurdles in finding the available parking slot in the day to day life of people and coming up with a better solution Our project has exceeded our expectations by using multi-client architecture, generic code and real-time availability of parking slots. We created a model parking lot with toy cars and out of cardboard to prove that this implementation would be possible in a real-world setting. We have created an application for booking parking slots that have never be created before with the right approach and suitable implementation. The major advantage is appealing android user Interface, it is easy to understand to all users.

7. ACKNOWLEDGMENT

We would like to thank Professor Mo Sha for all the help getting to start with guidance UI functionality and android features. The suggestion on using a real device for android development was very useful in starting and finishing our project, and that we’re grateful. We also thank our classmates for listening to the presentation and giving us feedback about our project.

8. LIMITATIONS

The limitations of the system would be not knowing how many IR sensors the Raspberry pi could handle because we have tested the system with 4 sensors. Another concern is how many user requests can be handled at a time, Android implementation is multithreaded adding more requests to a server would increase the network load, which could break our system.

9. FUTURE ENHANCEMENTS

In the future, we would like to deploy a webserver implementation for sending/receiving requests and grow it larger to handle more clients. Another feature we would like to develop is to stop unauthorized access to car parking space by taking number-plate details from user and use camera to verify if the same number-plate car is arrived or not. Additionally, payment mechanism for the parking reservation should be in place with reasonable amount.