CHAPTER 1
INTRODUCTION

1.1 GENERAL INTRODUCTION

Car-Parking in any Universal environment presents a unique challenge. To provide access control, user authentication, allocation of spaces to specific users, and data which can be used for predictive analysis requires a complex IoT solution. Such complexity must be balanced against the power requirements of a mobile platform deployment. To adequately scope a solution requires a deeper understanding of the issues associated with car parking. The design of a smart parking system consists of several essential parts—the detection stage of licence plate numbers, Information processing central system, network and mobile application.

Drivers will not only receive information through an application on their cell phones about parking places, how many places, how to get there fast and the parking fee after they leave in a certain parking lot, but are also told about the location of nearby parking lots and the suggested route to them. The whole system is implemented using Node-RED and OpenALPR. There are also several hypotheses of some functions that we attempt to realise in our design.

As mentioned before, the use of cameras to capture the images of licence plates and recognize the numbers, so that the system can tell whether there is a car parked in a certain place or not. To receive the accurate results, Node-RED was utilised and OpenALPR along with various images of plate numbers were experimented with as a machine learning trial to ensure the cameras were getting the correct information.

When the driver enters the parking lot, the cameras begin to work, they focus on the license plate number of the car. We use ultrasonic wave to confirm parking. After parking is completed, the image becomes static from dynamic. Then the cameras recognize the numbers. Each license plate number is bond with the driver’s account of the app. After the cameras get the number and transport it to the Information processing central system, the system starts to keep time and calculate the price. We use each parking lot as a node of the network and each app on cell phones as a feedback terminal. By connecting every device, drivers will be informed about every single detail they need to know.

1.2 OBJECTIVES

In today's world parking lots have become redundant and need a lot of manpower to handle and maintain it. These parking lots are not user friendly and do not provide data regarding availability of free spaces. Many researchers have contributed to this issue and formalised various methods to better optimise the parking lot to serve the needs. The author proposed a smart parking reservation system using short message services (SMS), for that he uses Global System for Mobile(GSM) with microcontroller to enhance security. The ZigBee technique is used along with the GSM module for parking management and reservation. The author uses the Global Positioning System(GPS) and Android platform to show available parking spaces. However, reservation for the same is not available.

The use of Artificial Intelligence (AI) techniques to process images, which recognizes the parking occupied only by vehicles. The system provides guidance images towards the assigned slots, thus making it intelligent. Inter integrated circuit (I2C) protocol is used along with car parking framework(CPF) to assign radio frequency identification (RFID) to each car which will be used to identify cars parked over a slot. Variable message screen(VMS) shows cars parked over a given floor. The system assigns and reserves an optimal parking space based on the driver's cost function that combines proximity to destination and parking cost. Driver request processing centre (DRPC) provides infrastructure to vehicle (I2V) communication for assigning and reserving parking spaces using smart parking allocation centre (SPARC).

The author uses a wide angle camera as a sensor which detects only free parking spaces and records them. These records are then used to assign parking space to the incoming user. Intelligent Transport System (ITS) and Electronic toll collection (ETC) using optical character recognition (OCR) creates a record for all entering vehicles. This creates tagless entry for all vehicles in the parking lot, but it does not assign a slot to the user. A universal OCR algorithm is not available, making it difficult to create said records.

The system automatically checks the unique registration number stored in the Bluetooth chip to check if the new vehicle needs to be parked. This system is a vertical parking arrangement for the vehicles with sensors that confirm placement of the car. Various other sensors are used to confirm that there are no passengers left in the vehicles and then the system moves the vehicle to a storage area employing rack and pinion (RaP) mechanism.

1.3 SCOPE

The can be upgraded to be able to be deployed with radio frequency identification (RFID) to authenticate at the gate management service (GMS) to assign a definitive slot. The system provides an additional feature to monitor parking lot over the internet.

Analysis shows that in almost every parking lot there will be some drivers disobeying the rules. For instance, some drivers may occupy more than one place and some may park on the road instead of the correct places. Our system will also set up some punishment for those drivers who break the rules. The system will warn those drivers at the first instance, which will be recorded in the system. If these drivers do not change their behaviours, they will be fined. The fine will be added to their parking fee. However, if they keep doing the same things, they will be forbidden to park in this parking lot.

The system also utilises a dashboard for the management of each parking lot, which is aimed at helping the manager monitor the usage of the parking lot. Each dashboard reflects information of its parking lot to the manager: the current empty spaces, cars parked in the lot per day, user credentials of those cars whose drivers break the rules and so on.

There can be no doubt this proposed solution could add huge value to any organisation, yet it has its limitations. Multiple technologies are needed to provide a full solution. There are also simpler solutions, yet these require the user to possess an object, thus denying them flexibility. This paper proves that there is a viable solution to this challenge, yet debunks the myth that IoT can simply solve all challenges.

CHAPTER 2
LITERATURE SURVEY

2.1 A Survey on "Smart Parking" System

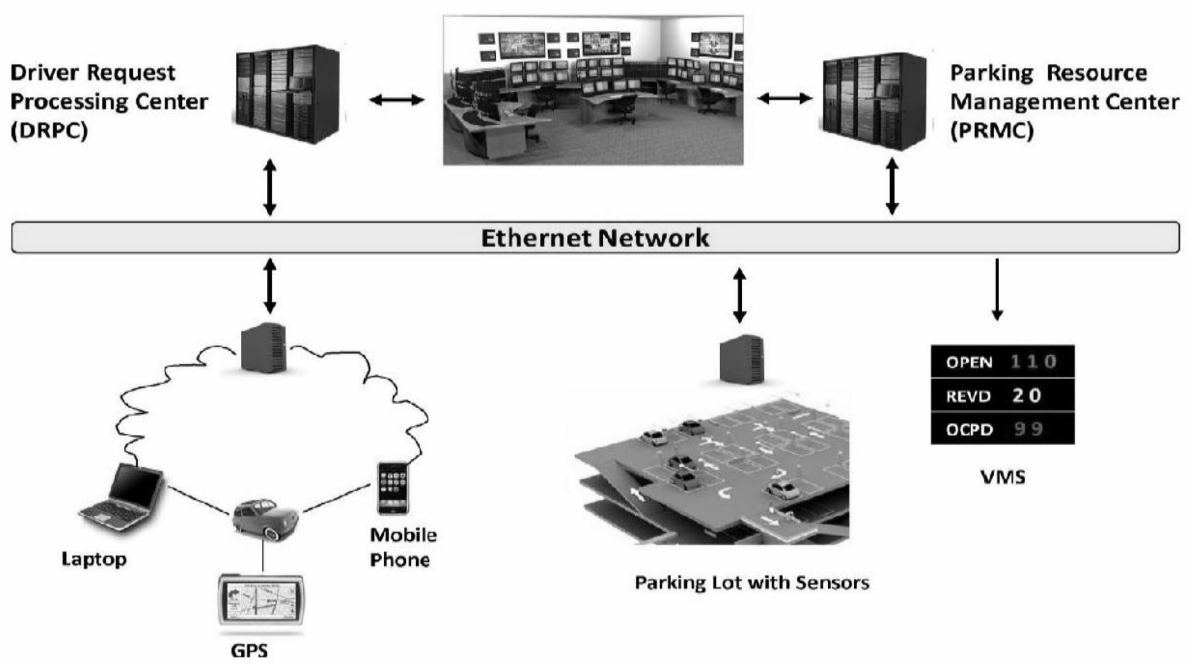
In today's world parking lots have become redundant and need a lot of manpower to handle and maintain it. These parking lots are not user friendly and do not provide data regarding availability of free spaces. Many researchers have contributed to this issue and formalised various methods to better optimise the parking lot to serve the needs. The author proposed a smart parking reservation system using short message services (SMS), for which he uses Global System for Mobile(GSM) with microcontroller to enhance security. The ZigBee technique is used along with the GSM module for parking management and reservation. The author uses the Global Positioning System(GPS) and Android platform to show available parking spaces. However, reservation for the same is not available.

The author uses Artificial Intelligence (AI) techniques to process images, which recognizes the parking occupied only by vehicles. The system provides guidance images towards the assigned slots, thus making it intelligent. Inter integrated circuit (I2C) protocol is used along with car parking framework(CPF) to assign radio frequency identification (RFID) to each car which will be used to identify cars parked over a slot. Variable message screen(VMS) shows cars parked over a given floor. The system assigns and reserves an optimal parking space based on the driver's cost function that combines proximity to destination and parking cost. Driver request processing centre (DRPC) provides infrastructure to vehicle (I2V) communication for assigning and reserving parking spaces using smart parking allocation centre (SPARC).

The author uses a wide angle camera as a sensor which detects only free parking spaces and records them. These records are then used to assign parking space to the incoming user. Intelligent Transport System (ITS) and Electronic toll collection (ETC) using optical character recognition (OCR) creates a record for all entering vehicles. This creates tagless entry for all vehicles in the parking lot, but it does not assign a slot to the user. A universal OCR algorithm is not available, making it difficult to create said records.

The author proposes robotic garage (RG) using Bluetooth which would be used to fully automate the placement of a car in the slot without the aid of the driver. The system automatically checks the unique registration number stored in the Bluetooth chip to check if the new vehicle needs to be parked. This system is a vertical parking arrangement for the vehicles with sensors that confirm placement of the car. Various other sensors are used to confirm that there are no passengers left in the vehicles and then the system moves the vehicle to a storage area employing rack and pinion (RaP)mechanism. The author proposes an upgraded system to the above, which is deployed with radio frequency identification (RFID) to authenticate at the gate management service (GMS) to assign a definitive slot. The system provides an additional feature to monitor parking lots over the internet.

Fig 1 Smart Parking Allocation Centre (SPAC)



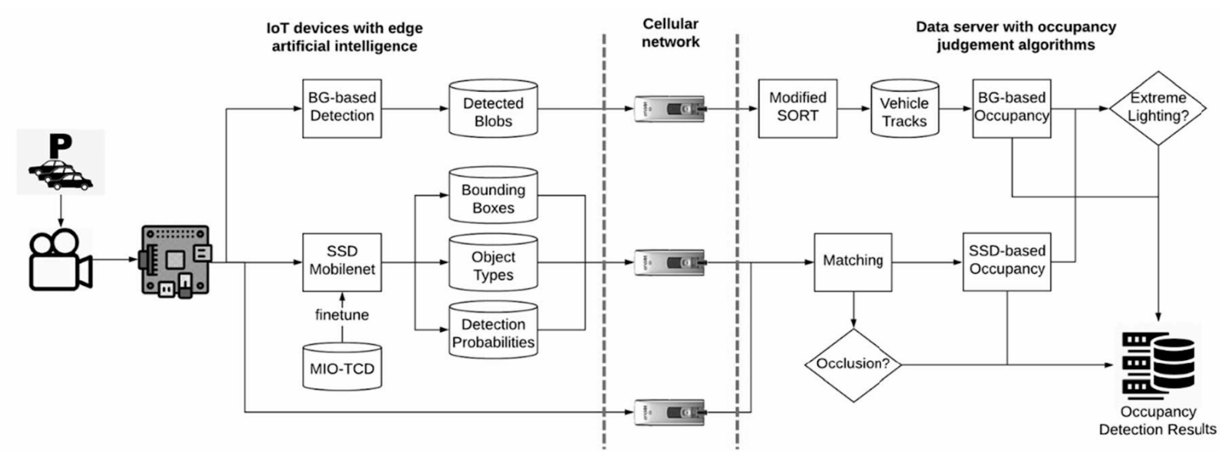
2.2 A Smart, Efficient, and Reliable Parking Surveillance System With Edge Artificial Intelligence on IoT Devices

The overview of the system design is shown as a flow diagram in Figure 2. The system is composed of camera nodes, IoT devices, cellular data transmission modules, and a centralised server. In this study, the IoT devices are Raspberry Pi 3B, yet other IoT devices like Arduino and Jetson Nano could be the alternatives. The overall design considers the balance between computational load and data transmission volume, as well as the reliability and scalability of the system.

Two efficient computer-vision-based object detection algorithms are implemented at the edge as two threads. They utilise the limited computation power of the IoT device to convert the raw video frames to detections in an online manner, thus largely reduce the data transmission volume and ensures efficient updates. Also, one video frame is transmitted to the data server every a few minutes for parking space labelling, results verification, and demonstration purposes.

On the server side, we propose a real-time object tracking algorithm based on SORT, as well as occupancy judgement algorithms considering occlusion and extreme lighting conditions. The modified SORT algorithm is implemented on the server side rather than the edge side because this design reduces the computation load at the edge while this implementation does not increase the transmission volume. Background-based occupancy detection results and SSD-based occupancy detection results are combined based on the occupancy judgement algorithms for improved robustness and accuracy.

Fig 2 Overview of the system design and technology



Vehicle Detection at the Edge

There are two detection methods implemented at the edge of our system: single shot multibox detector (SSD) and background (BG) modeling detector. They work in separate threads at the edge and then their detection results are combined in occlusion or extreme lighting conditions on the server for enhanced performance.

1) Enhanced SSD With MIO-TCD for Edge Artificial Intelligence: SSD with a Mobilenet backbone network is the primary detector. There are different backbones for SSD, while Mobilenet has the lightest structure which makes the detection faster than other backbones. This is appropriate for an IoT device with limited computational power. We recommend using TensorFlow Lite for the SSD implementation since it is designed for deep learning on mobile and IoT devices. A normal state-of-the-art object detector like YOLO-V3 [48]with the TensorFlow platform still runs slowly with a speed lower than 0.05 frames-per-second (FPS) on Raspberry Pi 3B, and has a slightly lower detection accuracy as well. However, SSD-Mobilenet with TensorFlow Lite runs over 1 FPS on the same device according to our test. The detection results including bounding boxes, object type, and detection probabilities (how likely the result is true) are transmitted back to the server. Compared to sending videos, it reduces the data volume by thousands of times (the exact number depends on the number of detections in the video).

TensorFlow models can be converted to TensorFlow Lite models. We recommend training a TensorFlow model and then convert it to the TensorFlow Lite model. In order to improve the detection performance to make it more appropriate for practical applications, we enhance a pre-trained SSD on the Pascal VOC dataset with a new traffic surveillance dataset called MIO-TCD , which contains 110,000 surveillance camera frames for traffic object detection training. This dataset includes a variety of challenging scenarios for traffic detection. such as nighttime, truncated vehicle, low resolution, shadow, etc. To our knowledge, this is the first time MIO-TCD been adopted for parking detection, and we find it works well.

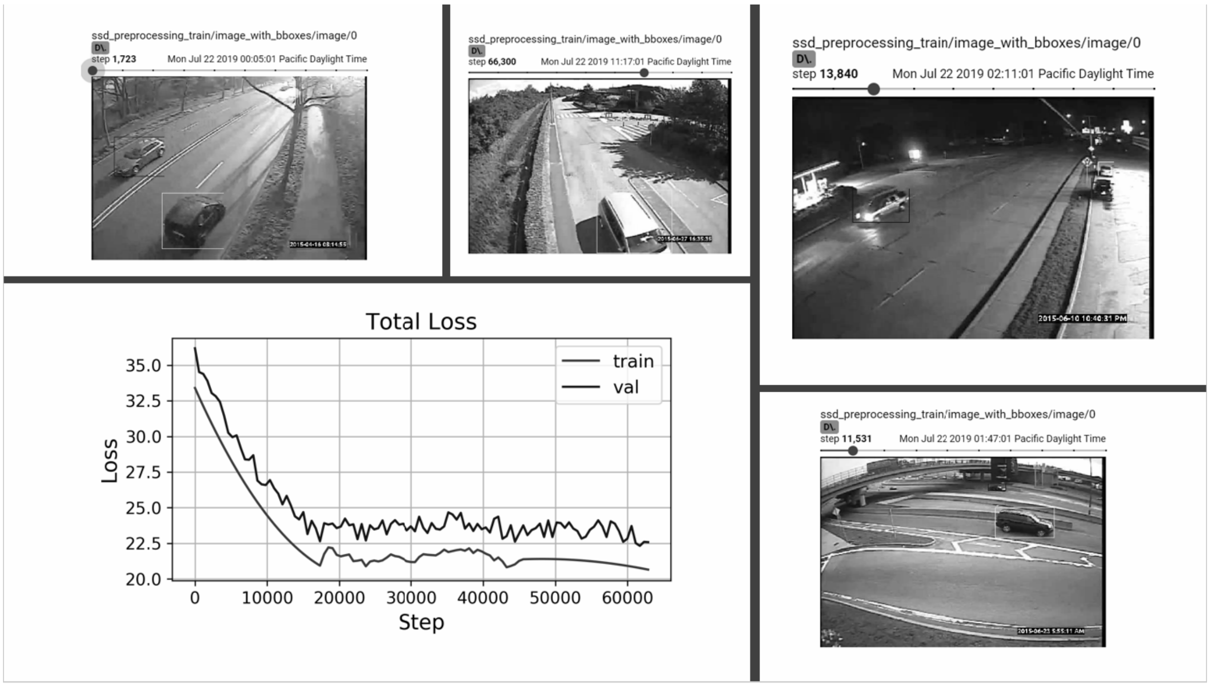
Some key parameters for the training are listed as follows: the learning rate is 0.00001, the weight decay is 0.0005, the optimizer is Adam, the batch size is 32, and the training-validation split ratio is 10:1. All layers are trainable. The training and validation loss curves, as well as some sample images at certain training steps, are displayed in Figure 3.

The enhanced SSD-Mobilenet model demonstrates great performances on traffic detection, especially in challenging surveillance image data. Figure 3 shows three examples comparing detection results between SSD trained on Pascal VOC and Pascal VOC + MIO-TCD. In the first column, the pretrained SSD detects all big targets but misses two small targets in the back; in the second column, the pre-trained SSD misses two vehicles partially blocked by a tree; in the third column where there is snow in the nighttime, the pretrained SSD misses most of the vehicles. Overall, the enhanced SSD produces much better detection results with few missed detections and no false detections.

2) Background-Based Detection at the Edge: Despite the enhanced performance of the SSD, the detection results are still not universally satisfying if your objective is to apply it to various real-world scenarios due to two reasons: though much improved in speed, the SSD running 1 FPS still does not meet real-time detection at the edge, which limits the use of video temporal information; deep learning model’s performance depends much on the training data, but the training data can never cover all real-world scenarios, so the detector itself could still perform poorly in extreme cases. Standalone SSD-based detection may be a good option for lab demonstration, but not for field practice universally.

With this observation and consideration, we propose to add BG-based detection to the edge. BG-based detection is a widely used traditional method for traffic video surveillance that is sensitive to video noises and has no classification ability. But it has two advantages that can help compensate SSD:it is very efficient and operates in real-time locally at the edge; it has a relatively more stable detection performance in extreme scenarios where SSD does not work, though not as good in normal cases. The BG-based detection is followed with a regular blob detection step, then the bounding boxes of the detected blobs are transmitted back to the server.

Fig. 3. The training and validation loss curves and sample images from MIO-TCD at certain training steps.



Data Transmission

The data transmission module in the system is composed of a 4G LTE Huawei USB Modem E397u-53, a T-Mobile data-only SIM card with 6GB monthly, and the software part. The T-Mobile data card is plugged into the 4G modem, and the modem connects with the Raspberry Pi via the USB interface. The connection of the device to the cellular network is activated via the Network Manager API in the software. The Network Manager allows automatic network connection upon start-up and automatic reconnection to the Internet whenever the connection fails. It is a reliable and helpful network connection tool that we recommend for IoT applications.

The reason we use cellular network connection for the data transmission is that the place where we do the field test does not have available wifi or ethernet. This will also be the case for many real-world IoT applications since the cellular network covers most urban areas and quite some rural areas. Other communications like Zigbee and LoRa are getting popular in IoT applications; however, they are good for short-distance communication rather than remote communication to the server. Cellular network communication is expensive with limited data amount, which, from another perspective, encourages data processing and reduction on edge. With the edge computing modules in the proposed parking system, it transmits BG-based detection results and SSD-based detection results to the server as strings. Also, the system transfers a video frame every ten minutes to the server for demonstration, validation, and space labelling. For an average camera, assuming one frame is 100Kb and the frame rate is 10 FPS (which is usually higher), and the detection results are 40Kb per minute, our system reduces the data transmission amount from around 86Gb per day per device to around 70 Mb per day per device.

Occupancy Judgement Pipeline and Algorithms

With the detection results from the edge, we develop a parking occupancy judgement method on the server. This method first calculates the SSD-based occupancy based on a proposed matching algorithm and BG-based occupancy based on multiple object tracking, then combines them together considering extreme lighting conditions and occlusion conditions.

2.3 Smart Parking System Using Image Processing and Artificial Intelligence

The detection of the licence plate numbers is of great importance. To make sure that the cameras make no mistakes, Node-RED was used for the programming and OpenALPR as the recognition tool. In this process, an input colour image is exposed to a sequence of processes to extract the relevant 2-D objects that may represent the symbols constituting the LP. These processes that are carried out in different stages will be presented in the following subsections.

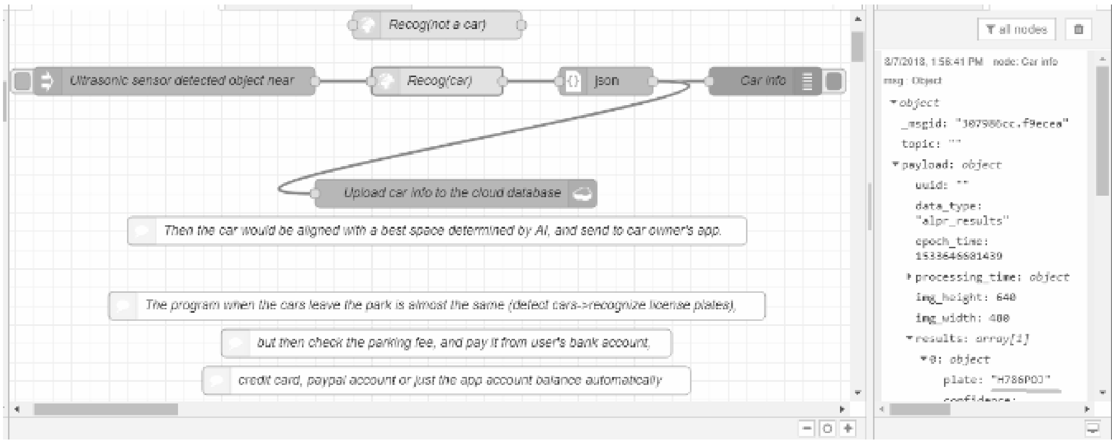
i. Colour to Grayscale Conversion

First an image of the car park should be taken when there are no cars in the car park. This would be used for the system to record the location of all the car parks. The RGB value can be used to find spaces that represent empty car spaces. With these spaces identified the system will know where to look for cars. For the empty car park, the image is converted into an (hue, saturation, value) HSV image. This is achieved through using the rgb2hsv command in Matlab. The HSV image is simplified to a black and white binary image so that it is easier to deal with by making the pixels white where the threshold is more than 40%. In order to do this, the HSV image needs to be converted to a grayscale format so that each pixel can be easily compared with the threshold. The equation to do this is shown in Equation. The rgb2gray command in MATLAB is used to do this.

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

ii. Image Acquisition

Fig 4 Ultrasonic Detection of car arriving in the space



After a car enters the parking lot, cameras placed at several fixed positions above the cars are used to acquire the image, then the system will select and arrange an available parking space for the car. When the car arrives at the space, ultrasonic wave detection is used to tell whether the car is parked or not, as shown in figure 4.

There will be some other cameras in the place that will confirm the information of the licence plate number and start counting time and calculate the parking fee. The calculation will be stopped after the car leaves the parking lot, which will also be confirmed by ultrasonic waves. Figure 2 below shows the Node Red code for the car exiting a space, a similar process for the space to be re-entered into the database as an Node-RED and OpenALPR as our programming platform and recognizing tool. The system took advantage of the depth recognition algorithm to train our cameras, allowing them to recognize the number of licence plates.

Figure 5 Node Red code for car leaving the space

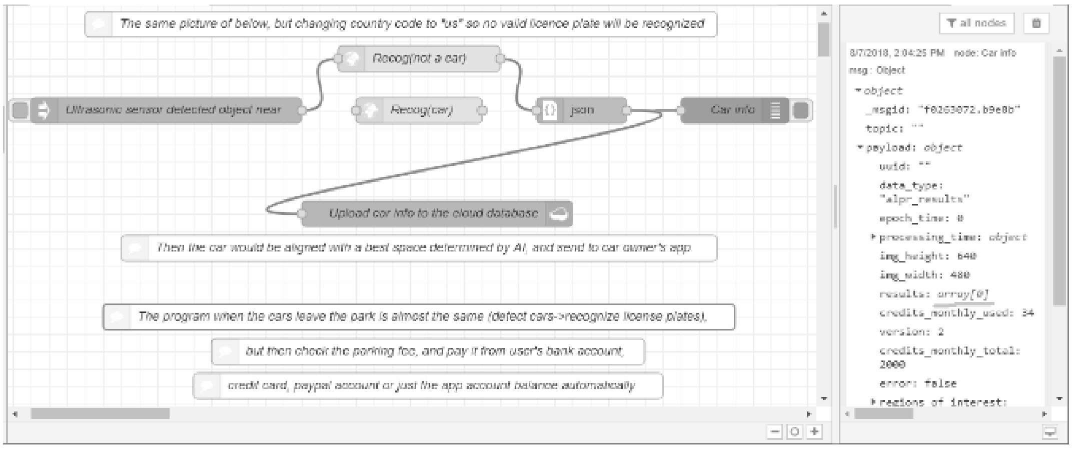
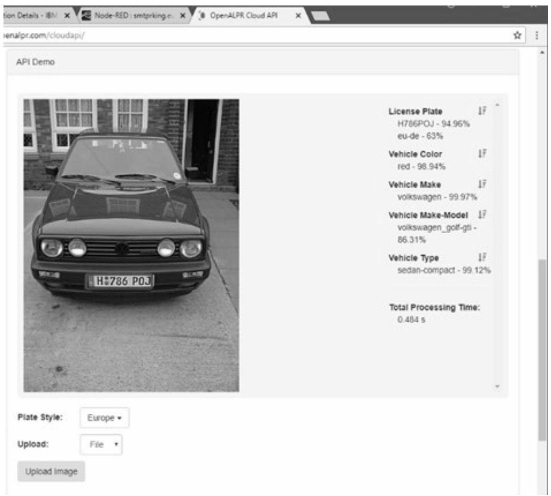


Figure 6 Licence plate recognition



iii Arrangement of the Parking Places

Arrangement of the parking places depends on the number of the cars that enter the parking lot. If there is only one car entering the lot, the system will arrange the nearest place for the car and show the shortest path to it on the driver's mobile phone. If there are several cars captured by cameras, the system will arrange one place for each car according to the number of the cars and average time and distance each car might take to park in their place. This method will raise the efficiency of the arrangement work. Both image and voice will be used to remind and guide the drivers. The ALPR camera is the central device in the proposed solution responsible for access control to each car park. The low approach speeds of the incoming traffic require a frame rate of 10 frames per second to accurately read the number plate of the upcoming car. The registration information is then routed via Wi-Fi to the Central Processing Hub which will contact the Cloud database to ascertain if the registration belongs to the white list or black list.

iv Time and Fee

As mentioned earlier the system starts calculating the time after the car is parked and ends after the car leaves the space, of which the result will be sent to the driver's mobile phone. This is a benefit to the driver as the driver is then only paying for the time the car is parked and not the time to find a space in the parking lot. Ultrasonic sensors were used to confirm the parking or leaving of the car by detecting whether the distance is changed or not which was then used to calculate an usage for the space over a set length of time. Parking fees can also be applied according to usage at peak times. For example, if one car enters the parking lot in the peak period, we can increase the fee. On the contrary, if the car parks at a time when there are few cars, the fee will decrease correspondingly. In the future, it is hoped the parking fee will be calculated on the basis of more elements, such as nearby events e.g. Concerts. Drivers pay the price through their bank card or other preferred method of payment that is entered by the user into the app on their smartphone.

v Parking Recommendation

Parking Recommendation refers to the fact that our system will predict the situation of the nearby parking lot when the driver arrives on the basis of the current traffic flow, the time that the driver may take to the lot, previous records and so on. Based on the calculation of prediction and parking fee of each parking lot, the system will recommend a suggested one to the driver. Drivers only need to click on the related button, then our App will estimate the number of available places when drivers arrive and navigate them to it by connecting to other navigation apps like Google Map. It will also show a time period-number of cars graph of previous record to the driver, providing a visualised information.

vi Warning and Punishment

Analysis shows that in almost every parking lot there will be some drivers disobeying the rules. For instance, some drivers may occupy more than one place and some may park on the road instead of the correct places. Our system will also set up some punishment for those drivers who break the rules. The system will warn those drivers at the first instance, which will be recorded in the system. If these drivers do not change their behaviours, they will be fined. The fine will be added to their parking fee. However, if they keep doing the same things, they will be forbidden to park in this parking lot.

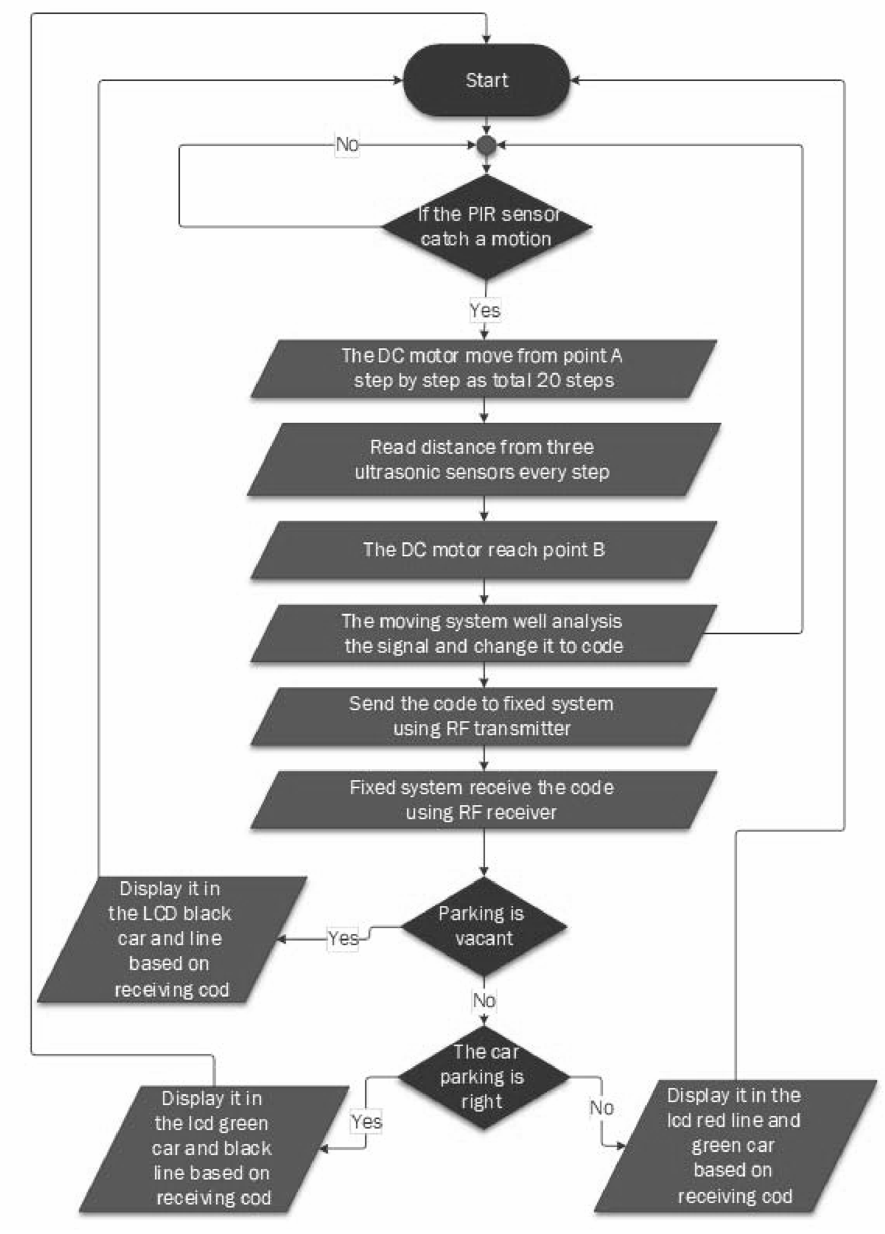
vii Dashboard for parking manager

The system also utilises a dashboard for the management of each parking lot, which is aimed at helping the manager monitor the usage of the parking lot. Each dashboard reflects information of its parking lot to the manager: the current empty spaces, cars parked in the lot per day, user credentials of those cars whose drivers break the rules and so on.

2.4 Smart Parking System for Monitoring Cars and Wrong Parking

When the car enters the multi-story car park or underground parking garage there is a motion sensor that feel 360 degrees will feel the car when its enter then will send a command to the DC motor, the DC motor will move to the first park, then the two Ultrasonic will read every step for each opposite park and line, the middle line will use the third Ultrasonic sensors, the Ultrasonic sensor will save the signal in the moving Arduino. The moving Arduino will analyse the signal and change it to code after that it will send the code to the fixed Arduino by using Radio Frequency wireless, the fixed Arduino will analyse the code. If the car was parking correctly then the fixed Arduino will display it in the Nextion display with a green car, if the car was parking wrong then the fixed Arduino will display it in the Nextion display with green car and red line, if the parking is vacant then the fixed Arduino will display it in the Nextion display with a black car. Figure 8 shows the Moving Arduino and Fixed Arduino; Figure 9 shows the design model for the proposed system and the rustle Nextion display.

Figure 7 Working principles of the proposed system flow chart



The cars come in the parking, and the PIR motion sensor detecting a motion as shown in Figure 9. The Moving Arduino start moving from point A step by step taking accurate readings for all area as shown in Figure 9. The Moving Arduino stops at point B than analysis the reads giving them codes after that send the codes to Fixed Arduino using RF transmitter and RF receiver. Fixed Arduino displays the codes in the Nextion display as shown in Figure 9.

Figure 8 Diagram of Moving Arduino and Fixed Arduino

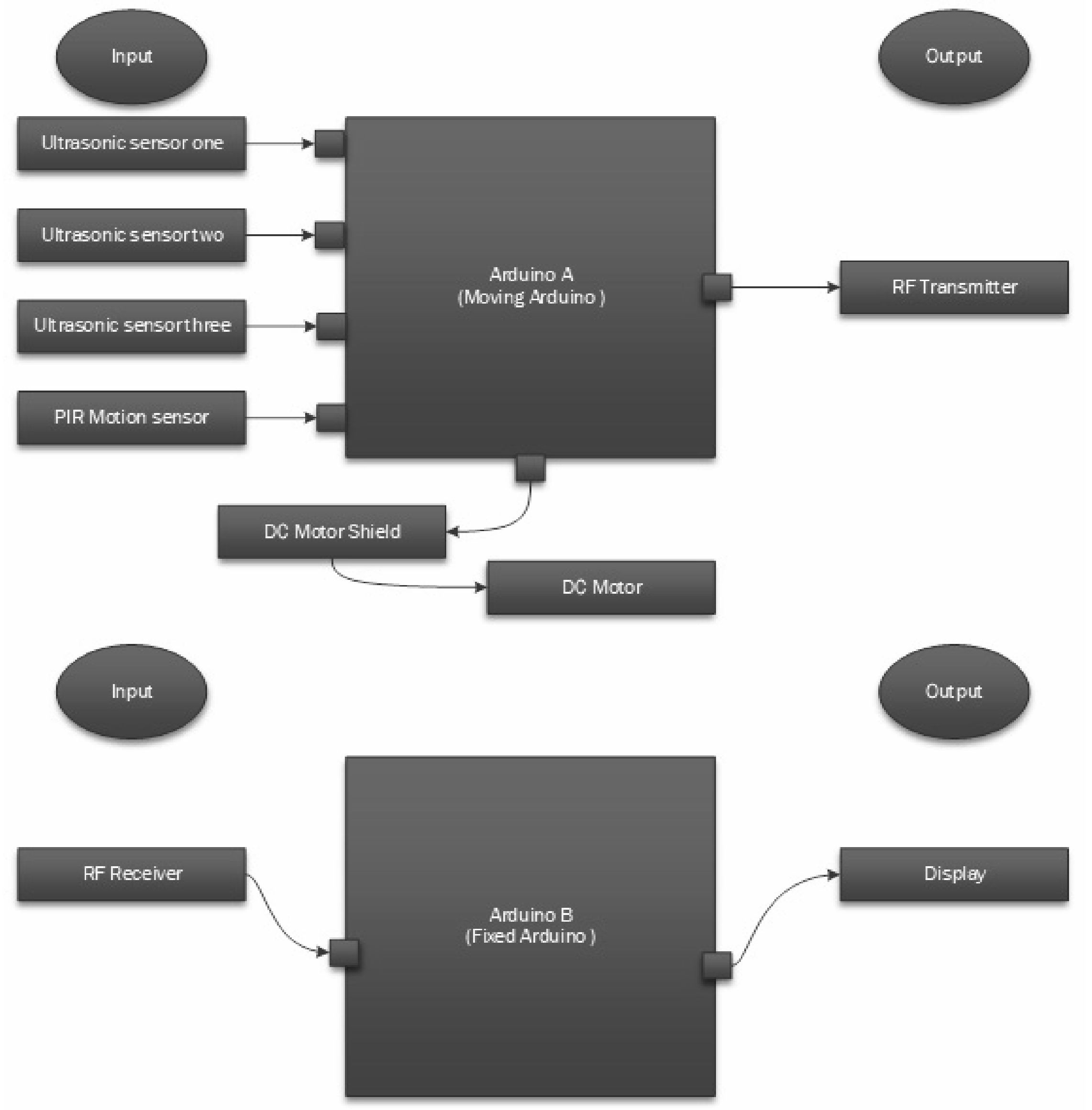
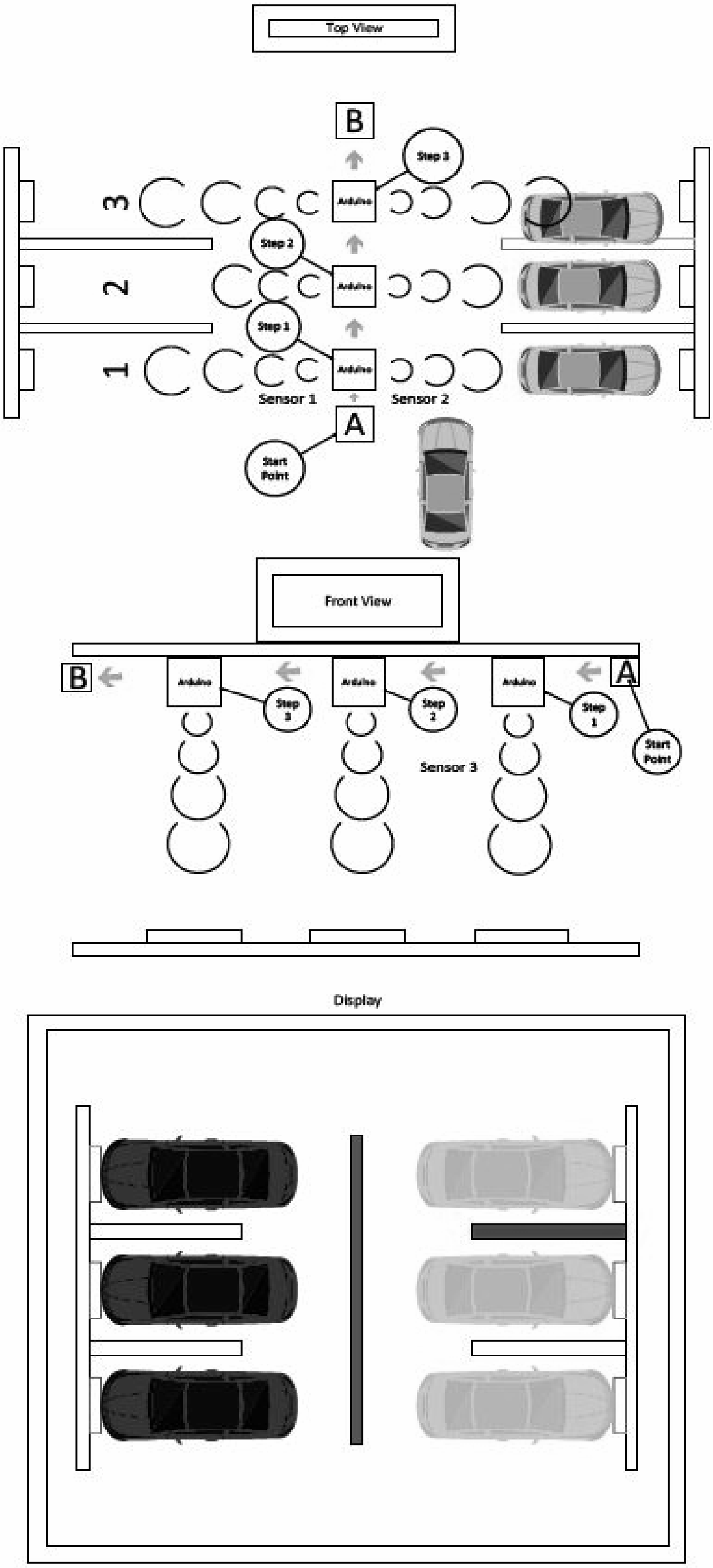


Figure 9 Design model for proposed system and the rustle Nextion display



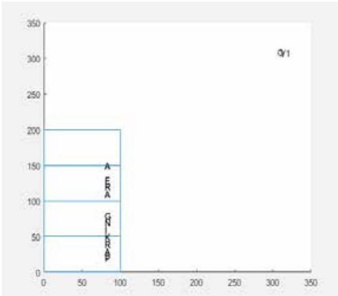
2.5 An Enhanced Vehicle Parking Management using Artificial Intelligence

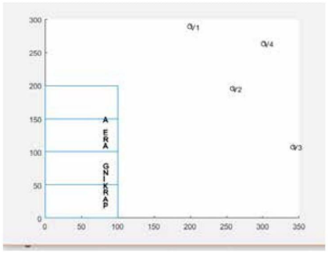
Parking vehicles as relay nodes and has introduced an enhanced technique to enable the parked vehicle for providing the services in an energy effective way. The core technique is categorised in two steps, the initial one is the clustering for moving the vehicles on the basis of communication between the driving vehicles and another one is the dynamic usage of external environment factors for achieving the energy conservation. The results have shown that the energy saving proposed method has achieved enhancement as compared to other parked vehicle methods.

Finding a vacant space for parking in saturated vicinity is regularly an irritating challenge and outcomes in extra delays of time for drivers. Author designs a web parking real time steerage gadget that examines availability of parking lots to the drivers. The author proposed an optimised parking architecture to reduce patron's riding and strolling fee with the usage of dynamic offers and requirement inside the community lot, extensively lowering parking contest in a proper time. Author implemented and evaluated the solution with a dataset of parking records gathered from various places, in which ground sensors are dispensed all over. The assessment outcomes displays that this approach achieves as many as sixty three percent put off discount in comparison with present solutions.

The proposed work aims to reduce the vehicle parking overhead by utilising Machine Learning architecture. The work done till now is construction of vehicle parking zones and other zones of the area. A total of one kilometre area is constructed out of which a 250 metre area is considered as a parking area. Vehicles come and go to the parking lot. What happens if the vehicle to be parked gets prior information of the parking lot and the best way to reach there in the shortest span of time. The proposed work aims to utilise Artificial Neural Network for the same. Similar kinds of architecture are seen in countries like Japan, Korea and China. The difference is, they lack in the utilisation of Artificial Intelligence for the same. The work done till now is shown in figure 10 and figure 11.

Fig10 Vehicle and Parking Area 		Fig11 Number of Vehicles Approaching Area





2.6 A New “Smart Parking” System Based on Optimal Resource Allocation and
Reservations

2.6.1 SYSTEM FRAMEWORK

Our proposed “smart parking” system adopts the basic structure of PGI systems. In addition, such a system includes a Driver Request Processing Center (DRPC) and a Smart Parking Allocation Center (SPAC). Fig. 2.6.1.1 depicts this framework. The Parking Resource Management Center (PRMC) collects and updates all real-time parking information, and disseminates it via VMS or Internet. The DRPC gathers driver parking requests and real-time information (i.e., car location), keeps track of driver allocation status, and sends back the assignment results to drivers. Based on the driver requests and parking resource states, the Smart Parking Allocation Center makes assignment decisions and allocates and reserves parking spots for drivers.

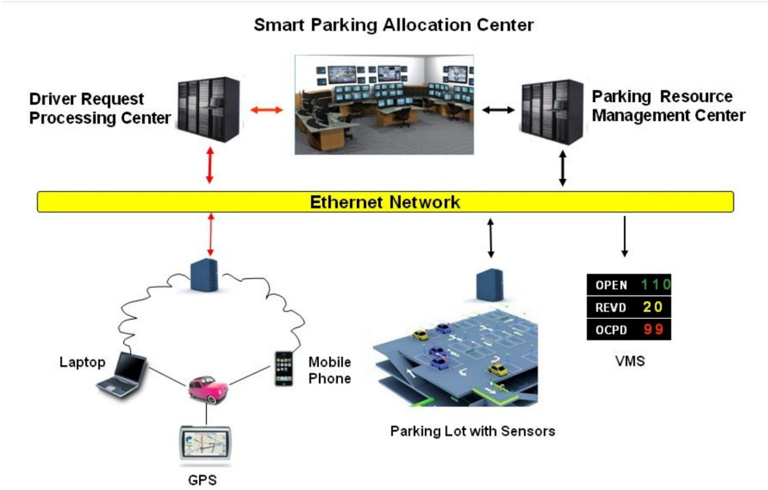


Fig. 2.6.1.1 Smart Parking Framework Overview

The basic allocation process is described as follows. Drivers who are looking for parking spots send requests to the DRPC. A request is accompanied by two requirements: a constraint (upper bound) on parking cost and a constraint (upper bound) on the walking distance between a parking spot and the driver's actual destination. It also contains the driver's basic information such as license number, current location, car size, etc. The SPAC collects all driver requests in the DPRC over a certain time window and makes an overall allocation at decision points in time seeking to optimize a combination of driver-specific and system-wide objectives. An assigned parking space is sent back to each driver via the DRPC. If a driver is satisfied with the assignment, he has the choice to reserve that spot. Once a reservation is made, the driver still has opportunities to obtain a better parking spot (with a guarantee that it can never be worse than the current one) before the current assigned spot is reached. The PRMC then updates the corresponding parking spot from vacant to reserved, and provides the guarantee that other drivers have no permission to take that spot. If a driver is not satisfied with the assignment (either because of limited resources or his own overly restrictive parking requirements) or if he fails to accept it for any other reason, he has to wait until the next decision point.

2.6.2 DYNAMIC RESOURCE ALLOCATION MODEL

For the sake of generality, we will employ the term “user” when referring to drivers or vehicles and the term “resource” when referring to parking spots. We adopt a queueing model for the problem as shown in Fig. 2.6.2.1, where there are N resources and every user arrives randomly and independently to join an infinite-capacity queue (labeled WAIT) and waits to be assigned. At the kth decision point, the system makes allocations for all users in both the waiting queue and the queue (labeled RESERVE) of users who have already been assigned and have reserved a resource from a prior decision point. If a user in WAIT is successfully assigned a resource, he joins the RESERVE queue, otherwise he remains in WAIT. A user in RESERVE may be assigned a different resource after a decision point and returns to the same queue until he can physically reach the resource and occupy it. A user leaves the system after occupying a resource for some amount of time at which point the resource becomes free again.

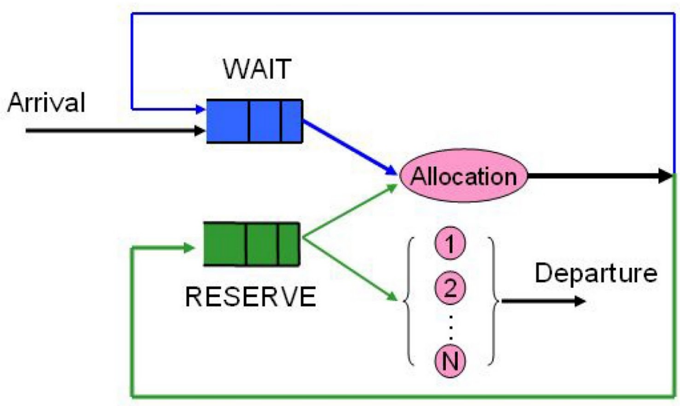


Fig. 2.6.2.1 Queueing Model for Dynamic Resource Allocation

CHAPTER 3
METHODOLOGY

3.1 HARDWARE USED

To achieve the car parking solution demands the use of multiple hardware components. For the most part, these components would operate in a standalone fashion, yet to complete the proposed solution required these platforms to integrate seamlessly. The central interface for all the technologies deployed in this solution proposal was the Central Processing Hub. This device is the centre point of all communications in the system. It is recommended that one hub is deployed at the entrance/exit to each car park. It is also proposed that this hub will be a Raspberry PI 3 board, due to its low cost, small form factor, the availability of compatible modules, and the on-board capabilities to interface with the technologies needed to enable the completion of the proposed solution .

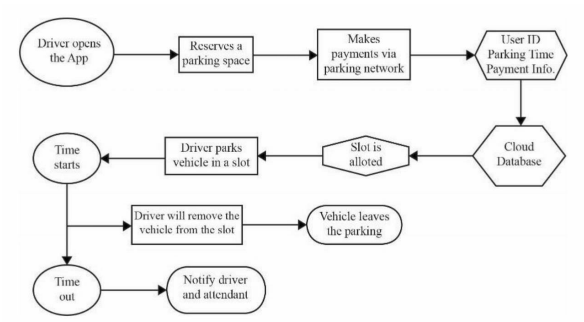


Fig 3.1.1 Functional diagram of Smart Parking System

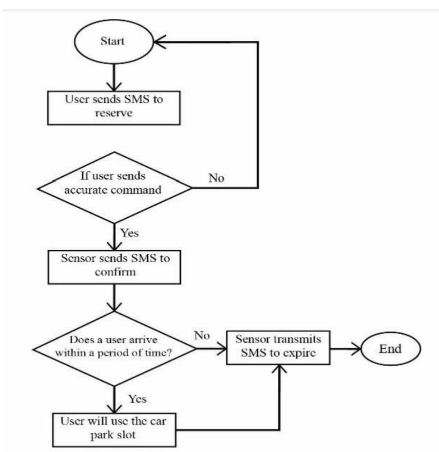


Fig 3.1.2 Flow chart for slot reservation

As mentioned above the Raspberry Pi can be used as a Wi-Fi router. This offers the system the opportunity to interface with the peripherals in the system, which need higher data rates and don’t have a power concern, securely and easily. It is then proposed that Wi-Fi be used for full-duplex communications with the ANPR camera, and between the barrier/signage to the Central Processing Hub.The modules, which will be added to the Raspberry Pi, are as follows:

1. Raspberry Pi 4G + GPS Shield. This device will be mounted on the Raspberry Pi board, a Subscriber identity module (sim card) will be placed inside this device and it will be used to communicate with the Cloud over the 4G network if available, and will move on to 3G/Edge/GPRS depending on the network
available

2. Edimax wireless nano USB Adapter. This device will be attached to the Raspberry Pi through the USB port. It enables the Raspberry Pi to become a Wi-Fi router with theoretical speeds of up to 150 Mbps

CHAPTER 4
CONCLUSION

4.1 Advantages and Disadvantages

Advantages:

Efficiency - Manually handling parking is rarely as precise and efficient as some might think. The work requires a lot of focus, as it deals with managing a large volume of people simultaneously in the same place. And if you miss or ignore one of them during that time, it could result in fines or even an accident for them! Then you'll have to deal with issues which would be much worse. You can avoid this by working with software instead; an automated ticketing solution is much more convenient for drivers and easy to use because they just need to get on their smartphone, enter their vehicle registration information, and drive into the car park. If a user wants to pay digitally, they can do so using their pre-registered credit card details; above all else, there are no mistakes allowed while parking.

Faster processing - Sometimes we get very bored and want to go home as quickly as possible. We are in a rush, and standing at the gates is a waste of our precious time. Are there any parking spaces available? Are my car keys still working? How many more minutes will I still have to wait? A lot of our thoughts race through our heads. Employees can now take advantage of the fantastic parking management systems to avoid the long line and wait for entry. These parking solutions can completely automate the parking process and make car parking time- efficient.

Report Tracking - Present-day parking management systems don't require paper and pencil, as everything is automated and can be outsourced to app-based reports. These parking systems allow managers to focus on more substantial tasks than counting vehicles or even one-on-one instances of people using a particular parking space. The app- based programs also make it easier for users to find and navigate free spaces by determining where they are.

All in all, the RFID technology effectively gives both users and managers a fair share of advantages when it comes to monitoring and managing parking spaces vs keeping an accurate record of all vehicles that come in contact with the system.

Improved Security - One of the perks of a parking management system is that it provides security. A barrier and reservation feature controls the vehicles allowed to enter and exit a space, for instance, at a business or event. This way, one doesn't have to worry about guests leaving valuables in their car, vandalism, theft, or illegal dumping. It can make your car park utterly secure. A CCTV (security camera) will monitor cars as well as licence plate numbers, so owners can keep track of their vehicles' whereabouts even if they've left them for someone else to watch over. Hence, having parking management systems in a car park is indispensable.

Disadvantages:

Expensive Construction and Installation - A parking management system can cost a lot of money. For example, the statistical feature, ticketing technology, and reporting tools are just some things that increase the price. In addition, the other things you might need to pay for include high usage or peak access fees, software maintenance fees, and fee waivers, to name a few. Your budget may not allow you to purchase everything at once, so make sure you prioritise your needs based on your organisations requirements.

System Breakdown - Utilising technology to manage a car park is unquestionably an excellent decision. Still, we cannot ignore that machines can start malfunctioning anytime, no matter how meticulously they are manufactured or what software they are integrated with. In these cases, chaos may occur. Imagine if cars couldn't access buildings and parked inside vehicles couldn't move. If the system malfunctions, this could lead cars to park in the wrong places. This is another considerable downside of using a parking management system.

4.2 Implementation

To check the implementability of the “smart parking” approach, we have built an urban-like testbed. This contains the basic elements of an urban traffic environment, such as roads, vehicles, traffic lights, urban blocks, etc. To meet the three main requirements of "smart parking," we include the following features. First, vehicle locations and parking spot status are detected by overhead cameras, which serve as a GPS. Second, vehicles communicate with the central allocation system (a computer) by Wifi. Third, we use wireless controllable parking barriers to guarantee a reservation. The computer periodically executes the optimal allocation algorithm and makes assignments, sends the assignment results to vehicles, and controls the opening and closing of each parking barrier. Due to space limitations, we only have 5 parking spots and 4 vehicles. However, the "smart parking" concept is clearly illustrated in the implementation, including allocations dynamically updated when random events occur, e.g., a vehicle joining the system or delays due to traffic lights. One can also see how vehicles access reserved parking spots that meet their requirements without blindly competing.

4.3 Case Study and Simulation

In this section, we describe a simulation environment based on parking at part of the Boston University (BU) main campus within the city of Boston. There are totally 679 on-street parking spots and 1932 off-street parking spots in this part of the campus. We assume that all these spots are monitored and can be used by any drivers (student, faculty or visitor) without any time limit.

Preprocess: Even though we are facing an allocation problem in a small district, the problem scale is still very large. Thus, as a first step, we reduce the decision variables and constraints of problem (P) by "grouping" parking spots. Intuitively, all spots in the same garage or parking lot can be treated as one resource. We also group the on-street parking spots in the same street block. With this grouping method, we aggregate 679 on-street parking spots to 27 groups and 1932 off-street parking spaces to 14 groups. If a driver reserves a parking spot in one group, the system simply selects any available spot in that group for him when he arrives. Following the same strategy, we also aggregate driver destinations by their locations. Buildings in the same block are treated as one destination, and we have totally 12 destinations. Fig. 3 shows the parking configuration after grouping, where red triangles represent destinations, blue squares represent parking lots and darkblue bars are on-street parking spaces.

Settings: In all simulations, we assume that user arrivals to each destination i are Poisson distributed with rate λi. User travel times to reach their destination are exponentially distributed with rate γ. The resource occupancy time is also exponentially distributed with rate μ. The user cost parameter Mi is uniformly distributed in the interval [0, Mmax], and the walking-distance parameter Di is also uniformly distributed in [0, Dmax].

Performance Metrics. In order to assess the overall system performance over some time interval [0, T], we define several appropriate metrics evaluated over a total number of users NT served over this interval (simulation run length). From the system's point of view, we consider resource utilization as a performance metric and break it down into two parts: ur(T) is the utilization of resources by reservation (i.e., the fraction of resources that are reserved) and up(T) is the utilization by occupancy (i.e., the fraction of resources that are physically occupied by a user).

4.3 Result and Discussion

It is apparent that the request for the smart car parking system will continue to increase in the forthcoming years. Though the smart parking system already exists, this paper adds a new idea by detected the wrong parking and is aimed at making the system more cost effective and user-friendly thus increasing its adoption in the market. The paper was successful and cost-effective, user-friendly and had 95% accuracy, we did a 20 case, and there was one case form the 20 is wrong cases. We found that if the model and the objectives were bigger, then the reads will be more accurate.

The proposed solution integrates multiple technologies to provide the most useful long-term solution. In the future the use of the parking space allocation algorithm, could be used to data model user parking requirements in any parking lot, potentially maximising occupancy by allocating the correct ratio of parking spaces per user group.

REFERENCES

Nihal Deochand Chaudhari and Ayushi Singh, “Automatic car parking system using Arduino” International research journal of engineering and technology May 2020.

E. Saranya, V Gokula Prasath, G. Gokulraj and Kowshik’ “IoT based smart car parking system" International journal of Research in Engineering, science and management, Feb 2019.

Suvarna S. Nandyal and Sabiya Sultana, “Smart car parking system using Arduino UNO”, International journal of Computer application July2017.

Omkar Walvekar, Amar Kulkarni and Rohith mane, “Automatic car parking “, International Research journal of Engineering and Technology March 2017.