Parkease: A Smart Parking Solution based on Image Processing

Ann Mariya Johnson[1]
AmalJyothi College of Engineering
annmariyajohnson2023@cs.ajce.in

Aswin Sasikumar[2]
AmalJyothi College of Engineering
aswinsasikumar2023@cs.ajce.in

Athul Saji[3]
AmalJyothi College of Engineering
athulsaji2023@cs.ajce.in

Christy Maria Eappen[4]
AmalJyothi College of Engineering
christymariaeappen2023@cs.ajce.in

Dr.Juby Mathew[5]
Head Of Department
AmalJyothi College of Engineering
jubymathew@amaljyothi.ac.in

***Abstract*— Parking management systems are frequently specialised hardware-intensive systems installed in buildings and shopping centres. However, such expensive systems cannot be installed in many locations due to a number of factors, including cost and urgent/temporary setup needs. Our project's objective is to develop a parking management system based on image processing to find empty parking spaces in places without automated systems. An image processing method is applied to the camera photos of the parking lot to identify virtual spaces there and extract occupancy data to advise approaching cars of the availability and location of open spaces. The technology uses the number plate image captured by the camera to identify the car. Additionally, the system focuses on guiding drivers to the parking spot using a map-creation algorithm that gives the motorist the shortest route to the slot with the least amount of cruising in the parking area.**

I. INTRODUCTION

Finding parking lots for private automobiles in public locations is a very known issue that has grown in importance in recent years as the urban population has grown rapidly. According to studies, drivers spend 30% of their time cruising for parking, and the average time to find a proper space ranges between 3.5 and 14 minutes. The increased traffic congestion caused by drivers hunting for a free parking lot in the city has a significant influence on the quality of life of residents, providing compelling reasons to research and adopt revolutionary Smart Mobility solutions.

In our project we propose a system for smart parking where it uses image processing. The major two algorithms to build the application of smart parking system are OpenCV and Tesseract [1]. OpenCV is used for image processing. The project's purpose is to be able to automatically detect whether a parking space is open or occupied from any static photograph or video of a parking lot. The still image file taken is a frame from the video file, and once the code is run, the still picture appears, and the mouse is used to form a quadrilateral from the input points. To start the display of the video feed, press the coordinates of the input points. The quadrilaterals used to zone each parking space are either red or green in colour. The colour red indicates that the parking space within the quadrilateral is occupied by a car, while green indicates that it is accessible for parking. Tesseract algorithm is used for character recognition. Tesseract package contains an OCR engine - libtesseract and a command line program - tesseract. When the car enters parking lot, we

capture the number plate image. We perform image processing methods on the captured image. We then find the contours of the plate and crop the image along those contours. The processed image is then fed to Tesseract algorithm for character recognition.

Map generation for parking slots using Python is the process of creating a visual representation of a parking lot that shows the location and dimensions of individual parking spaces.This map helps the drivers to find their appropriate parking spot in mean time. This is done using various libraries and tools in Python, such as OpenCV and matplotlib,to generate a map that can be used for parking management, transportation planning, and navigation.

II. Literature Review

Shruthi B et al. present a solution for smart parking using image processing in [1]. Where the system is intended for use in open parking lots, multi-story parking garages, and numerous other applications. The suggested system architecture uses the combined edge detection and coordinate bound pixel sections to assess whether a parking space in the recorded video is occupied. It is implemented in Python and the OpenCV library. Additionally, it demonstrates how to convert images to text. Text can be extracted from a processed image using Tesseract. Variable image processing guarantees that different images are handled differently to produce text that is optimised. The system's fundamental workings are as follows: The parking lot is then inspected for available places after a car pulls in. If there are any, the first camera reads the licence plate of the automobile, stores it in the cloud, and then shows the car the places that are open for parking. The same will be displayed if there are no open slots. When the car exits, the second camera scans the number plate once more and compares the time of the arrival. The amount of time the car is parked determines the cost of the bill.

Jiang Ruili et al. present a solution for Smart Parking System Using Image Processing and Artificial Intelligence in [2].This process involves using image processing to detect the license plate numbers of vehicles in a parking lot. To ensure accuracy, the system uses NodeRED for programming and OpenALPR as a recognition tool. The system begins by processing an input colour image and extracting 2-D objects that may represent the symbols on the license plate. This process is carried out in multiple stages, including colour to grayscale conversion, image acquisition, and depth recognition. This text describes a smart parking system that uses image processing and artificial intelligence (AI) techniques to identify and track vehicles in a parking lot or garage. The system includes cameras that capture images of the parking environment, which are then processed using computer vision algorithms and AI techniques to extract relevant information about the parking environment. The system uses machine learning algorithms to recognize the shape and size of typical parking spaces, as well as other features such as entrances, exits, and obstructions. It also uses ultrasonic wave detection to confirm when a vehicle is parked or has left a parking space, and calculates parking fees based on the duration of the vehicle's stay. The system is programmed using Node-RED and uses OpenALPR as the recognition tool. It also uses depth recognition algorithms to train the cameras and improve their ability to recognize license plate numbers.

Luca Ciampi et al. shows number of cars in the entire parking lot can be automatically estimated by a multi-camera system in [3]. Using images from smart cameras, this research provides a novel method for autonomously counting cars in a parking lot.. Uses CNN, which can identify and count the number of vehicles present in pictures taken by specific cameras. Data aggregation from all devices is accomplished via a decentralised geometry-based methodology. Messages and photos from a single camera are used to spread the total count. Parking lot surveillance with multiple cameras. Many suggested systems process the captured images aboard Raspberry Pi devices. The extracted data is sent to a central server, which uses a model based on the HOG feature descriptor and SVM classifier to assess parking spaces in overlapping areas. Smart cameras, on the other hand, automatically localise and count the vehicles that are in their range of view, disseminating individual results through messages to the other edge devices. Compared to the more established suggested systems, this approach is quicker and

more versatile.

Ravi Kiran Varma Pa et al. proposed a system which has [4] three main phases: detection, recognition, and pre- processing. Images or videos are also acceptable inputs. Before beginning to detect number plates on video, the image source must be prepared for additional processing because video is thought of as a series of images or frames. In this study, morphological transformation, Gaussian smoothing, and Gaussian thresholding are just a few of the image processing techniques that are used in the pre-processing step. Following that, borders are followed to apply contours, and then those contours are filtered based on character dimensions and spatial localization for number plate segmentation. Character recognition is done using the K-nearest neighbour technique after region of interest filtering and de-skewing. This work effectively detects and recognises Indian licence plates. The primary contribution of this research is the development of an image processing system for the detection and recognition of Indian vehicle number plates. handling various lit images taking care of both light and dark items. handling noisy image situations. dealing with uncommon licence plates. dealing with number plates that are slanted or crossangled. managing our licence plates that are partially damaged.

Alessandro Floris et al. present in [5] an innovative IoT-based smart parking solution meant to provide information on the state of parking spots available in on-street parking zones.For the SP system for on-street parking, they create virtual replicas of physical things using the Social IoT (SIoT) Lysis ecosystem. The usage of social virtual entities makes it possible to address interoperability issues across diverse IoT device types used by independent solutions placed in close proximity. Each parking space is equipped with magnetometer sensors that use low-energy Wi-Fi to identify the presence of a vehicle. Concentrators that cover the entire parking lot with low-energy Wi-Fi then collect the sensor data. Additionally, an Android App can be used to access a control dashboard that has been created to manage the monitored parking areas and provide responsive data analytics regarding the occupancy of parking spaces in the city. Finally, a smart payment service uses Bluetooth beacons to automatically charge users for services they use. The developed testbed was used in experiments to show the system's capacity to recognise the presence of a vehicle and determine the owner ID in order to start the payment process. An LSTM RNN and a tree decision model are trained on historical data to predict the condition of parking lots in a specific location in order to demonstrate the capabilities of the proposed Data Analysis ME. The LSTM attained an average accuracy of 96.69%, exceeding the decision tree's performance (91.61%).

M.R.M. Veeramanickam et al [6] proposed a method that utilizes an IoT platform and data processing from sensors to manage parking lots. Data sets, including item id, parking lot id, creation date, location, latitude, and longitude, are collected and processed to understand parking demand and allocation. The prototype system was tested in a college campus during a festival event. The system uses sensor-based IoT to collect data and process it using smart devices. The system is broken down into three layers: the smart parking model layer, the transmission layer for peer-to-peer communication, and the IoT sensor layer to gather signal data. The FCFS algorithm with a priority queuing model is used to manage parking slots based on booked parking tasks. The goal is to collect and apply current traffic data to improve parking management during peak demand.

Thomas Westfechtel et al.[7] proposes a method for automatically generating an environment map that is enhanced with information regarding parking place locations. Using a graph-based technique, this system estimates the parking lot topology and infers free parking spots from parked automobiles in the surrounding region. To collect data, we used an autonomous electrical vehicle (AEV) as a stand-in for future self-driving automobiles. The AEV was outfitted with numerous sensors, which includes Velodyne HDL-32E LIDAR. The data of LIDAR was utilised to construct an environment map using a simultaneous localization and mapping (SLAM) approach while also detecting and localising cars which uses a convolutional neural network (CNN). In a graph-based technique, the findings are fused, and the topology of parking lots are approximated, as well as vacant parking

places inferred. The main features of the suggested method in this study include a graph-based method that estimates the topology of the parking lots and infers unoccupied parking spots, as well as a system which builds the map of the environment while enriching it with parking lot locations.

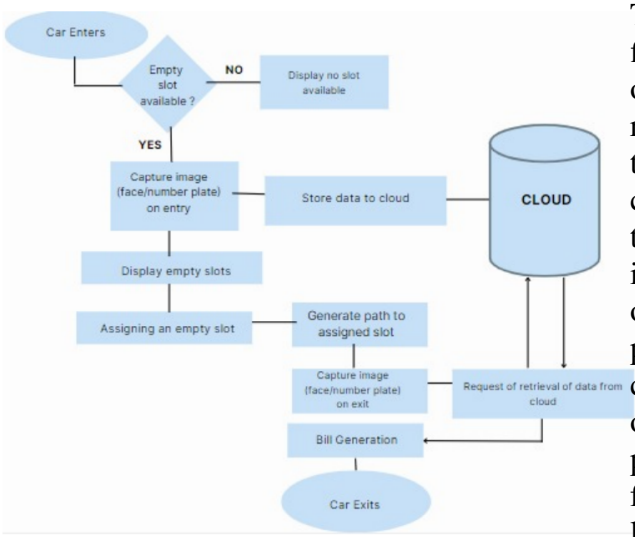
Gongjun Yan et al.[8] presents a secure and smart parking system using NOTICE framework. Diverse techniques are used to find credible information. Widespread use of initially wired sensors Inductive loops, pneumatic road tubes, magnetic sensors, piezoelectric sensors, weigh-in-motion systems, and inductive loops are only a few examples. The control computers are physically wired to these components or sensors. The need for lengthy and intricate wiring from parking lots to the central control unit is a drawback of wired sensor systems. The need for numerous sensor units also drives up the price of building this system. As a result, some wireless sensors can be used to detect parking spaces. This system employs GPS to track the location of vehicles and roadside units to communicate parking messages. The security certificates are kept up by roadside units and information on parking. The security concern discussed in this work is how to guarantee fair play among drivers through encryption and frequent spot-sensing. By lying, drivers are stopped from taking advantage of the system further. A roadside assailant, for instance, could reserve as many spots as he wanted by posing as a car. All of the communications in this system are activated by short-range signals and physical pressure on belts, so that a roadside attack cannot be carried out.

E. Cassin Thangam et al.[9] explains idea of Internet of Things (IoT) based Smart Parking Reservation System using Raspberry-pi. In the proposed system, the user must first register his details in the mobile application. He can go into his account and reserve a spot for parking his automobile after the credentials have been registered. If he reserved a spot, he gets 15 minutes to enter the parking space before the period runs out. As a result, when his car pulls into the parking lot, its licence plate is compared to the one entered while reserving a space. Using a Raspberry Pi 3 camera, this is accomplished. Additionally, a photo taken using a Raspberry Pi 3 camera is used to verify the driver's identification. In order to prevent vehicle theft and other security difficulties, this is done. Once everything is done, the car is allowed to drive into the parking lot and the closest open slot number is issued. His parking time therefore begins as soon as the car is successfully parked in the space. Therefore, the parking time can be halted and the receipt delivered to the driver's mobile device as he gets back in and removes his car. Then, he can make an online payment using his e- wallet. The driver's face identity is then once more confirmed at the exit barricade using facial recognition technology to match with the earlier photographs, and he is then permitted to leave the parking lot.

D. Di Mauro et al[10]They examine three computer vision technologies—picture classification, object detection, and semantic segmentation—with a focus on parking lot management. We take into account two scenarios: stall-based occupancy estimation and stall-free occupancy estimation. The assumption behind stall-based occupancy estimation is that each parking space's location is known. A camera is then utilised to detect whether each space is occupied or free. On the other hand, stall-free occupancy calculation does not depend on the availability of stalls. Since image classification cannot be employed in this situation, two methods based on object recognition and image segmentation are instead studied. A Region of Interest (RoI) is given to specify the area in each photograph where automobiles should be tallied, allowing for the elimination of areas where cars may be moving, such as roads.

III. METHODOLOGY

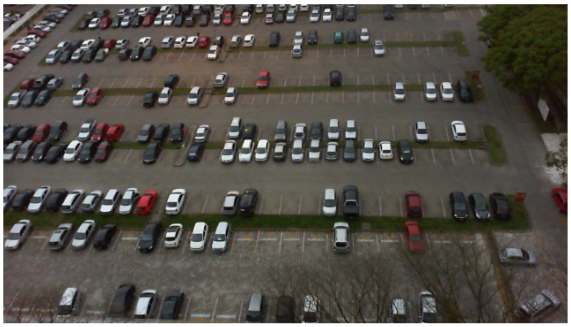
As the car enters the parking lot, the parking lot is checked for empty slots. If there exists any then the number plate of the entered car is scanned using the first camera and stored in cloud and let into the parking lot by displaying the slots available. If there are no empty slots available, then the same will be displayed. When the car exits then the number plate is scanned again using the second camera and compared with the time of its arrival. A bill is generated based on time of stay of the vehicle.



The basic steps are: 1.The system may use sensors embedded in parking spaces to detect the presence of vehicles and transmit this information to a central server or directly to drivers' smartphones or in-car navigation systems. This can help drivers find available parking spaces more easily. 2.The system may also use cameras and other sensors to monitor traffic flow in and out of the parking lot, and adjust the parking fees based on demand. For example, if the parking lot is full, the system may charge a higher fee to discourage drivers from entering. 3.The system may also allow drivers to pay for their parking fees using their smartphones, credit cards, or other electronic payment methods. This can help streamline the payment process and reduce the need for cash transactions. 4.In order to protect privacy, the system may use encrypted data transmission and storage, and may also allow drivers to opt out of certain features such as facial recognition. 5.In addition to helping drivers find parking and pay for it, the system may also provide valuable data to cities and businesses about parking usage patterns, which can be used to optimize the allocation of parking spaces and improve traffic flow. Main modules in this system are: 1. Empty Slot Detection 2. Number Plate Detection 3. Facial Recognition 4. Vehicle Counting 5. Map Generation

A. *Empty Slot Detection*

Here is a high-level outline of a methodology for empty slot detection in a parking lot using image processing: Firstly, Acquire a series of images or a video of the parking lot. This can be done using a camera mounted at a fixed location, such as on a traffic signal pole or overpass. Pre-process the images or video to remove noise and improve the contrast between the parking spaces and the surrounding area. This can be done using techniques such as smoothing, thresholding, and morphological operations. Use image segmentation techniques to divide the image or video into regions corresponding to individual parking spaces. Techniques used can be such as clustering, edge detection, and region growing. Use object detection techniques to identify any vehicles present in the parking spaces. Various techniques for this are Haar cascades, HOG features, and deep learning-based object detectors.Finally determine which parking spaces are empty by identifying those that do not contain any vehicles.



B. *Number Plate Recognition*

We photograph the licence plate as soon as the car pulls into the parking space. We use image processing techniques to the image that was taken. Next, we locate the plate's contours and crop the image along those lines. The Tesseract method is then used to recognise characters from the processed image. The extracted image is next examined to see if it is present in

the cloud or not. If the plate is already present in the cloud, we retrieve the data from it and generate a bill based on the length of the stay. If the plate

is not already in the cloud, we upload it together with the entrance time.

C. Facial Recognition

The face of the driver is recognised as the car pulls into the parking lot for security reasons. OpenCV is utilised by the Raspberry Pi camera to create a dataset of the driver's face, which is then used to train the camera to recognise the driver. To provide complete security, the driver's face is recognised once more at the barricade as he pulls out of the parking space. He is only permitted to leave the lot if his face matches the learnt information. The Support Vector Machine technique and the Eigenfaces approach based on Principal Component Analysis were used to generate this model (SVM). Python is used to implement the facial recognition algorithms, while OpenCV is utilised for image processing and feature extraction.

D. Vehicle Counting

Image processing for vehicle counting can entail a number of processes, including: Firstly, Install cameras in key areas along the highway or road where you wish to count the number of vehicles. These cameras ought to be able to record pictures of the passing cars. Pre-process the photographs to enhance their quality and get rid of any distractions or noise that can affect the vehicle counts. Techniques like edge enhancement, colour correction, and picture denoising could be used for this. In order to distinguish the automobiles from the background in the photos, employ image segmentation algorithms. Techniques like thresholding, blob identification, or morphological processes may be used in this.Then, identify and categorise the automobiles in the photographs using object recognition techniques. Techniques like feature extraction, template matching, or machine learning algorithms might be used in this. Follow the motion of the cars as they pass through the camera's field of view by using image tracking algorithms. Techniques like optical flow, Kalman filters, or particle filters may be used in this. Use the car count data to create reports and analytics about things like occupancy levels, congestion hotspots, and traffic flow patterns. The utilisation of this data can enhance traffic control and optimise transportation networks. To ensure that the image processing system keeps functioning precisely and dependably throughout time, update and maintain it frequently.

E. Map Generation

The methodology for map generation for parking lots using Python typically involves the following steps: Data collection: The first step in map generation is to collect data about the parking lot. This may include the location and dimensions of individual parking spaces, the layout of the parking lot, and any obstacles or landmarks within the area. This data can be collected using a variety of methods, such as manual measurement, laser scanning, or the use of sensors or cameras. Data processing: Once the data has been collected, it needs to be processed and formatted in a way that can be used to generate the map. This may involve tasks such as cleaning and formatting the data, interpolating missing values, or projecting the data onto a particular coordinate system. Python provides a range of libraries and tools for processing and manipulating data, such as NumPy and pandas. Map generation: With the data processed and formatted, the map can be generated using a variety of techniques, such as creating a raster or vector image, or using a 3D modeling tool. Python provides libraries such as OpenCV and matplotlib for generating images, and libraries such as PyOpenGL and Pygame for creating 3D models. Map customization: The map can then be customized to include additional details or features, such as labels, legends, or annotations. Python provides libraries such as Matplotlib and Seaborn for customizing maps and visualizations. Map storage and distribution: Finally, the map can be stored in a suitable format and distributed to users as needed. This may involve creating a static image file, a web-based interactive map, or a mobile app. Python provides a range of libraries and frameworks for storing and distributing data and maps, such as NumPy, pandas, and Flask. Overall, map generation for parking lots using Python involves the use of data collection and processing techniques, as well as various libraries and tools for generating and customizing maps. Python's versatility and wide range of libraries make it a suitable choice for implementing map generation systems.

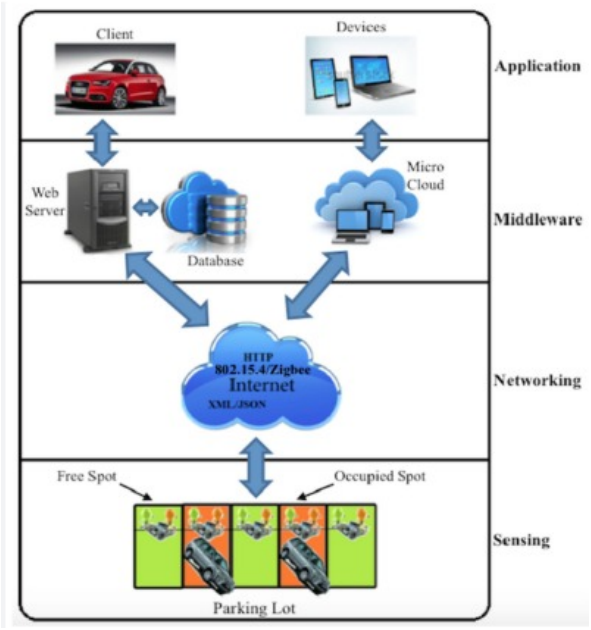
IV. Result

In this project, a proposed image-processing-based scheme for a smart parking system has been successfully tested and ran with a few movies acquired from underground parking lots. The technology accurately determines whether parking spaces are occupied or not by displaying a red outline if a vehicle is inside or occupying a spot, and then turning green when it is vacant.

For number plate detection, the number plate's image is taken for number plate detection, and OCR is then used to extract the characters. Together with the time, the identified number plate character is saved in the Firestore cloud as text. After the vehicle exits the parking lot and the bill is created, this will be used once more.

The system has effectively recognised more than 96.2% of the characters from plates and has successfully detected 98% of the number plates from the dataset of 101 plates, including Indian and international plates. A system that detects parking lot status and reservations gives information in real-time.

Facial recognition in the entry and exit of parking lots have identified and recognized the faces of individuals entering and exiting the parking lot accurately. The counting of vehicles is done using certian evaluation metrics, counted on various scenarios (sunny, overcast, rainy). The solutions obtained are best when compared to baseline conditions.



In the output of the multi-camera algorithm for a pair of images from two different cameras with a shared area in their fields of vision, the masks projected from one camera to the other using the previously computed homographic transformations are indicated in red and blue

V. Conclusion

An image processing-based smart parking system is a technologically advanced solution that uses cameras and image processing techniques to identify the presence of cars in parking spaces. The most effective utilisation of available parking spaces and increased effectiveness of parking operations can both be achieved using this information. Some of the key benefits of a smart parking system using image processing includes the Improved accuracy: Image processing algorithms can provide more accurate information about the availability of parking spaces, compared to other technologies such as sensors;Reduced cost: Image processing can be more cost-effective than other technologies, as it does not require the installation of expensive sensors or other hardware; Enhanced functionality: In addition to only detecting the presence of a vehicle, image processing methods can be used to identify a number of details about it, including its make, model, and licence plate. Overall, a smart parking system using image processing can provide significant benefits in terms of accuracy, cost, and functionality, making it an effective solution for managing parking in urban areas.

References

[1] 1. Shruthi B, Raghottam J Kulkarni, Sanath Kumar S, Dilip Kumar S. "Smart Parking System Using Image Processing" International Research Journal of Engineering and Technology (IRJET) Volume: 07 Issue: 06 — June 2020.

[2] 2. Jiang Ruili, Dr Eoin O'Connell, Dr Sean McGrath, "Smart Parking System Using Image Processing and Artificial Intelligence" 2018 Twelfth International Conference on Sensing Technology (ICST) J. Vis. Commun. Image R. 62 (2019) 234–244.

[3] 3. Luca Ciampi,Claudio Gennaro, Fabio Carrara, Fabrizio Falchi, Claudio Vairo, Giuseppe Amato, "Multi-camera vehicle counting using edge-AI" 2022 Institute of Information Science and Technologies of the National Research Council of Italy (ISTI-CNR), via G. Moruzzi 1 - 56124, Pisa, Italy

[4] 4. Ravi Kiran Varma Pa, Srikanth Gantaa, Hari Krishna Bb, Praveen SVSRK, "A Novel Method for Indian Vehicle Registration Number Plate Detection and Recognition using Image Processing Techniques" International Conference on Computational Intelligence and Data Science (ICCIDS 2019)

[5] 5. Alessandro Floris , Simone Porcu , Luigi Atzori , Roberto Girau , ”A Social IoT-based platform for the deployment of a smart parking solution” Computer Networks Volume 205, 14 March 2022, 108756

[6] 6. M.R.M. Veeramanickam , B. Venkatesh , Laxmi A. Bewoor , Yogesh W.Bhowtd, Kavita Moholkar , Jyoti L. Bangare, ”IoT based smart parking model using Arduino UNO with FCFS priority scheduling” Sensors 24 (2022) 100524

[7] 7. Thomas Westfechtel , Kazunori Ohno , Naoki Mizuno, Ryunosuke Hamada , Shotaro Kojima , and Satoshi Tadokoro, ”Parking Spot Estimation and Mapping Method for Mobile Robots” IEEE Robotics and Automation Letters, Volume 3, No. 4, October 2018

[8] 8.Gongjun Yan, Stephan Olariu, Michele C. Weigle, Mahmoud Abuelela ”SmartParking: A Secure and Intelligent Parking System Using NOTICE” 2018 Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems Beijing, China, October

[9] 9.E. Cassin Thangam, M. Mohan, J. Ganesh, C.V. Sukesh, ”Internet of Things (IoT) based Smart Parking Reservation System using Raspberrypi” International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 8 (2018) pp. 5759-5765

[10] 10.D. Di Mauro, A. Furnari , G. Patane, S. Battiato , G.M. Farinella ”Estimating the occupancy status of parking areas by counting cars and non-empty stalls” J. Vis. Commun. Image R. 62 (2019) 234–244