# Vehicle Number Plate Recognition and the Slot Allocation in a Cloud-Based Automatic Parking Management System under varying Weather Conditions.

MSc Research Project
MSc in Cloud Computing


Deepak Chandrashekar
Student ID: 23221178

School of Computing
National College of Ireland


Supervisor: Abubakr Siddig

**Student Name:** Deepak Chandrashekar

**Student ID:** 23221178

**Programme:** MSc in Cloud Computing **Year:2024**

**Module:** MSc Research Project

**Supervisor:** Abubakr Siddig
**Submission Due Date:** 12/12/2024

**Project Title:** Vehicle Number Plate Recognition and the Slot Allocation in a Cloud-Based Automatic Parking Management System under varying Weather Conditions

**Word Count:** **12192**

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

**Signature:** Deepak Chandrashekar

**Date:** 12/12/2024

**PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST**

| | |
|---|---|
| Attach a completed copy of this sheet to each project (including multiple copies) | □ |
| **Attach a Moodle submission receipt of the online project submission,** to each project (including multiple copies). | □ |
| **You must ensure that you retain a HARD COPY of the project**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. | □ |

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

| **Office Use Only** | |
|---|---|
| Signature: | |
| Date: | |
| Penalty Applied (if applicable): | |

# Vehicle Number Plate Recognition and the Slot Allocation in a Cloud-Based Automatic Parking Management System under varying Weather Conditions.

Deepak Chandrashekar
Student ID : 23221178
Research Project, MSc in Cloud Computing
National College of Ireland
Email: x23221178@student.ncirl.ie

**Abstract**

Rise in the growth parking issues has been caused by the increase in urbanization and the use of cars, most of which clog roads, causing traffic jams, pollution, and time exhaustion. As a solution to the issues mentioned above, we have designed an Intelligent Parking Solution based on IoT, ANPR, and Computer Vision. This system applies real-time data, artificial intelligence, and navigated cloud services to optimize parking organization, minimize parking time, and advance environmentally friendly urban mobility. The key features which includes the real-time parking slot availability, management of entry and exit through number plate recognition, automatic parking slot assignment, automated fee collection, and a centralized database for analytics and business intelligence data. The solution raises user satisfaction, increases income, optimizes flow, and supports urban growth. The YOLO-based models used for parking detection and car number identification allow the system to work under different environmental conditions. Besides these advantages, the system provides an easy-to-use interface for users, promotes the environmental conservation and helps in the development of towns and cities. By enhancing parking management, it supports smart city strategies for improved sustainable urban mobility.

***Keywords***— Cloud Computing,Deep Learning, EC2, Load Balancing, Autoscaling Group, Vehicle Parking Solution, Convolutional Neural Network, AWS CloudWatch

## 1 Introduction

The congestion, pollution, and wasted time are all the outcome of parking problems made worse by the proliferation of both cities and cars. In response, smart city projects are reshaping parking management with cloud-enhanced frameworks. Optimizing space usage (Xie et al. 2023)reducing search times, and promoting sustainable urban transportation are all possible outcomes of the cloud-based parking systems that would integrate IoT, machine learning, and real-time analytics (MDPI, 2023; Cities Today, 2024). Installed in towns like Linköping, Sweden, these systems allow drivers to easily locate or find, book, and can be paid for parking through smartphone apps, making the process more streamlined and effective.
By making use of the cloud frameworks (Yang, He and Wang 2023), these smart parking systems facilitate the interchange of data among st linked devices, allowing for the smooth and quick communication of information regarding traffic patterns and available places. By directing cars to available spots, these frameworks would also enhance traffic management, which in turn decreases driving time and emissions. Moreover, these frameworks' data-driven features enable proactive urban planning, which improves ability and economic growth (Cities Today, 2024; MDPI, 2023) by enabling cities to adjust infrastructure based on real-time data insights. Advanced technologies like AI (Mnyakin 2023) and transfer learning play a significant role in smart parking systems, making them more scalable in varied urban situations. Together, these advancements and cloud computing have great potential for creating sustainable cityscapes by making solutions to urban congestion more flexible and resilient (MDPI, 2023).

## 1.1 Motivation

Especially, in the recent year urbanization has developed and increased the use of cars, which also in turn increased parking problems, due to struggling to find the parking spaces, traffic jams, fuel consumption and emissions. For these issues, we present an Intelligent Parking Solution that works like a search engine to solve parking experiences. The smart parking spot utilizes current data and information technology, IoT, and ANPR for detecting available parking space, slot allocation of the car without the driver's effort and detail evaluation of the vehicle for efficient recovery of fee through exit gate. As the solution would also be linked with a centralized database, it enhances parking operations and significantly lowers the time needed for searches, as well as diminishing driving to find an empty standing area, which in turn would contribute to environmental conservation. Further, fee collection automation and usage analysis as to the optimization of business operations and revenues originate from parking management. Flexible and modular, it encourages smart initiatives such as city building and enhances the fluidity of traffic in urban regions and provides a convenient and eco-friendly solution to the current parking systems of today.

## 1.2 Research Objective

I would like to suggest that the next step is to create an Intelligent Parking Solution, which is in fact a search engine for parking spots. Computer vision, the ANPR technology, IoT, and real-time analytics shall be incorporated to turn parking into an efficient, manageable system as explained next. Key Features of the Solution.

- **Real-Time Parking Availability Detection** : The car park itself will use sensors with camera feeds confirming space availability in real-time based on data feeds. Parking can be searched on an app or via a website, similar to a browser where users will be guaranteed of getting the available slots with ease.
- **Automatic Slot Assignment**: When the user gets to the parking facility, a space will have been reserved for him/her depending on availability of space, size of the car and the preferences of the user in terms of nearness to other structures etc.
- **ANPR-Based Data Integration**: ANPR technology will be utilized for capturing number plates of the vehicles upon arrival and also upon departure from parking lot premises. This makes sure that people minimize their contact with others during check in and checking out of the app.
- **Automated Fee Deduction**: Tolls for parking will be computed by the time taken in parking and this will be charged directly from the user's associated payment method, digital wallet or the associated credit card.
- **Comprehensive Database and Analytics**: Existing parking will have central database to organize data about parking and utilization, revenue generation, and peak parking hours. This data can help facility operators make informed decisions on usage of parking lot and help them to effectively plan for expansions in the future.

**Advantages associated with the solution to the problem**:

- Convenience and Time Savings: Excessive time spent in search or a looking of a parking space is eliminated thus decreasing the stress level of drivers and increasing the user satisfaction.
- Environmental Impact: By eliminating unneeded trips in search of a parking space, the system saves fuel and consequently decreases emissions of some pollutants that harm the environment.
- Enhanced Revenue and Operational Efficiency: Automated fee collection and data analytics remove doubts and errors which adversely affect revenue for parking operators.
- Improved Traffic Flow: Reduced circulation of cars around parking facilities and other central business core areas is made possible with efficient parking management.
- Scalability and Integration: This makes the system adaptable to the smart city concepts, public transportation system, and tool for urban development hence making it scalable in the future.

This system has the capability to change the face of mobility in urban areas and improve citizens' standard of living with technologically integrated solution while effectively solving operational and management issues of parking operators and future development plans of city planners.

## 1.3 Research Question

- How does a cloud server deployed parking solution help in a connected system?
- How will this system help in a smart and better parking solution in different smart cities?
- How will artificial intelligence-based computer vision system help in achieving these solutions?

## 1.4 Research Area

The main focus and the goal of my research project is to help in making one server for all so that all the search results are stored in one system. This will not only prevent multiple allocations of the parking slot but also track the vehicle presence at one particular time. The main specific issue which has to be addressed such as Parking Space and the Number plate detection would be detected almost same as the human being detecting the objects using the Deep Learning and the Computer vision. I would be comparing a lot of deep learning based algorithms for the parking place detection. The idea is to find a best parking place.This will help in assigning that particular parking place (Google Search like Parking place detection). The ANPR analysis will be using a detailed YOLO models from v8-v11. These YOLO models will not only extract the number plates but are agnostic in different counties number plates as well. Also here I will be extensively using the deep learning models to be trained on two different transfer learning-based parking detection datasets, parking space and PKLot. In case of ANPR we will make an agnostic solution to extract the number plates. This method of Unified Solution would also consist of a real time deployed weight and that would also auto update in a real time using the scheduler.

# 2 Literature Review

## 2.1 Research on Smart Parking Optimization

Result information provided on the Xi'an urban road network (Xie et al. 2023) has been demonstrated that the compared established allocation parking model would improves the parking space utilization, revenue, and also it would decreases the traveling cost within the shared parking zones with the first book first server, secondly the integration of the GPR-A can also minimize the travel time by 14.9% and 5% in comparsion with the RPR and the SPR methods along with this the allocation of the parking model.
In the paper (Rajyalakshmi and Lakshmanna 2024) authors have used a technique called the Hybrid Deep DenseNet Optimization which would combine a Machine learning and the Deep learning algorithm for availability rate forecast of parking spots. The ML model is based on the HDDNO and the secondary data collected from CNR Park located in Pisa, Italy. The prediction applies various regression algorithms to predict the parking space availability in the given time. While implementing DenseNet technique has produced better results, but the accuracy acquired by implementing HDDNO model was better. In more detailed way, the five optimizers such as RMSProp, SGD, ADAM,AdaGrad and AdaDelta which is also abbreviated as 1.Root Mean Squared Propagation, 2. Stochastic Gradient Descent , 3.Adaptive Moment Estimation[ADAM], 4.Adaptive Gradient[AdaGrad], 5. AdaDelta which have been contributed to a very great extent in decreasing the model's loss. The section of Adam the predictions produced by HDDNO model were accurate in the extent of 99.19% and have loss 0f 0.0306% thereby proving that the model is good for prediction. This proposed methodology would go a long way in enhancing the safety of the environment as well as be an effort in evaluation of smart cities. The architecture (Yu 2023) utilizes local area network-level deployment Platform as a Service the dew service technique recognizing the number plate and dew technique for creating a machine learning model, these designs would gives a architecture a number of benefits including the examining the outcomes, they are as follows: (1) Decreasing of the maintenance and deployment problem and improving the reliability and sustainability of an Dew system, (2) Releasing the network constraints of the cloud computing and increasing the system scalability both vertically and horizontally, which would improves the dew computing to addressing the issue of the heavy computing process and suggesting the dew machine learning mechanism which can eliminate the need for restraining.

Here in the paper (Rafique et al. 2023) the authors has discussed related to the Hardware Based solutions utilizing the different kinds of sensors which have been installed at relevant sites to monitor or identifying the status of parking slots which can makes them the less scalable, costly to install complex, also it would be more expensive to maintain. In difference, the data-driven solutions would always utilize the existing available infrastructure of the surveillance cameras which has been installed at parking space area, thus overcoming the restrictions of sensor-based solutions, However, the current and only classification-based approaches which are prevalent and also adopted to monitor on the or keep an eye on the parking space area conditions also which would limits the systems capacity and also it would also slows down their performance. In this paper author proposes an intelligent parking management system which would utilize the deep learning method to reduce or ease the limitations in the data driven solutions by providing the high performance and also it provides fast inference capacity or the capability of YOLO v5 for vehicles detection instead of parking space classification. Models was also gauged or evaluated by making use of the PKLot dataset which was a benchmark for identifying or detecting the status of the parking space by achieved having an accuracy of 99.5%.

In the research paper (Türker 2023) the authors used an IoT based parking smart management systems which are being described and also used for building a model which are provided for the system design architecture where the parking lots are also being controlled at the same time. In the same way the smart parking architecture model, networks and the cloud infrastructure using a IoT-Based systems which were both assessed, bounded by the requirements. Criteria translated from the criteria of an application or implementation based on the design central management which have been described. This criteria of an implementation on the application was based on the design center management are described. By means of utilizing a smart parking system to be established based on these all criteria and the closest parking space available will be selected.

## 2.2 Research on Parking Management using various techniques

(Xing and Long 2023) study shows that the The parking space lock is placed on parking lot area, parking space lock will be equipped with a camera 6, a parking gear lever 1, a wireless data transmission module 2, a single chip computer 3, a photoelectric switch sensor 7 is connected to the above parking lock and scanning the vehicle information in front of the parking lock is converted into an image and stored. The wireless data transmission module 2 is adapted for wireless transmitting the vehicle information image to the cloud based vehicle information matching system by image recognition. The single chip 3 can control the lifting of the parking gear lever 1, Through the image recognition on the vehicle information image in the cloud and recognizing the license plate information after algorithm processing and database comparison to send the control signal to control the opening and closing of the parking space lock. This project has provided for the convenient control of parking lots and addressed such issue of low intelligence associated with regular car parking space lock.

In the research paper (Sudhakar et al. 2023) the Smart-Parking Systems makes use of Image processing method for capturing the vehicle registration place and also an automated door opening and the closing attribute or the feature, when a system recognize the vehicle at entrance of the parking space area. Security feature or an attribute would be covered in the mobile application. The available parking slots explanation and security bits of the parking space area such as gas leakages or fire. But Raspberry Pi are the one that is in charge of commanding and processing all the works of a system. The LCD display would be placed at the entrance of a parking space area so it gives an idea or shows the current situation of the parking slots or place available for the parking. Infrared Radiation proximity sensors which can be used to detect the presence of the vehicle at entry point of the gate. The pictures or Images which have captured or the character from the vehicle number plate, the plate can be detected and will be Raspberry Pi would sends the signals to servo motor to open the gate for a specific interval of the time.

(Alam et al. 2023) In Their study since most of the spaces of these productions are non de-script with natural sources of light missing, there is the call to get a large source of power to light up the large numbers of lighting equipment frequently. This would leads to wastage of the time, loss of equipments and the hitches the great financial blow to property management. The management of parking area resources using an IoT for parking purposes can go along way in optimizing the utilization of the available resources. Facing sensor based smart parking spot as depicted by the identified location information via wireless technology would be of good benefit for the vehicle owners and the parking area authorities and the owner of the spot. They can meet a lightening demand of parking space zone where the minimum energy would be utilized. The main noticeable characteristics such as IoT based systems which includes user authentication,online reservation to enhance parking direction, security, online charging, and also the efficient company of energy. IoT enabled Smart Parking Management Systems is discussed in this paper.

There are severe shortage of the parking space due to the increase in number of vehicle populated densely in the smart cities (Yang, He and Wang 2023), There is a high importance on the dedicated sensor which are the costliest sensors devices that can make a necessary installation expense, when it could come into the systems that allow vehicles to see which parking spaces are available. Now that a low-cost IoT system is available, embedded cameras can monitor how often parking spots are used. It is important to manually specify the placements of parking spaces before drivers can use these devices, even for images captured by cameras. This case study would also proposed that uses a cloud platform to improve the accuracy of car location predictions and decrease the amount of time vehicles have to wait to park. The suggested approach employs a machine learning technique to categorize parking spot topologies according to their stationary locations.

In the paper (Magdum et al. 2017) project describes first impression on algorithm to optimize smart parking system by of all the technological advancements we have looked at, RFID and cloud technologies. In the present work, here we have used Google map to know the geographical location of a user. This is a project enables the users do a search on an available parking slot and at a low cost and this depends on the latitude and longitude of the user. This project estimates users' parking cost on the basis of total available places in each of the car parking. Here the user will look for an empty and a free parking space. in response to an incoming request from the user and if the current car parking slot is occupied in turn, this system identifies a new car parking slot into the designated parking area. In this case, it possible to launch the proposed project in the real scenario.

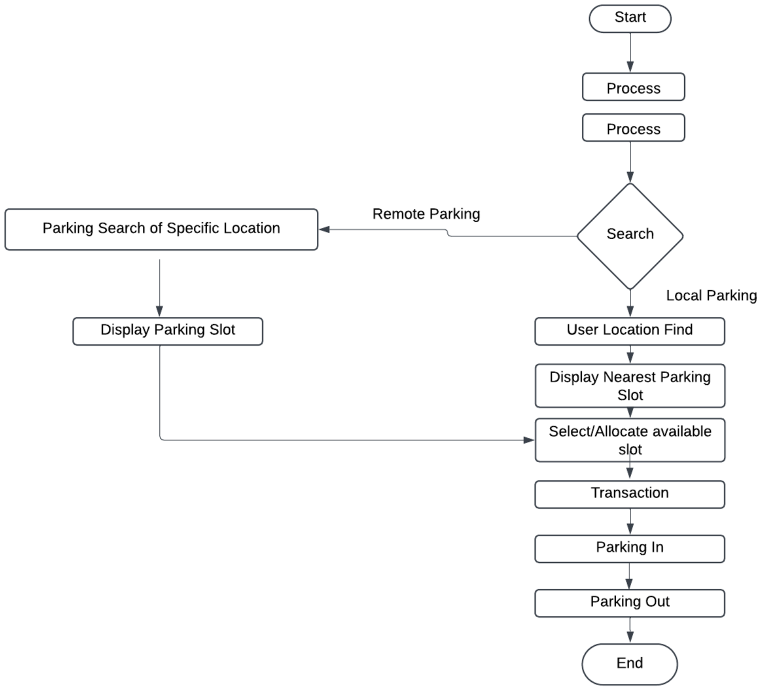



Figure 1: System Operation Architecture used in (Magdum et al. 2017)

## 2.3 Review on Auto scaling[ASG] and Elastic Load Balancing [ELB] using AWS Cloud

Here in this research (Ashalatha and Agarkhed 2015)It is possible to employ an auto scaling method to distribute resources concerning types of client requirements on SLA in the hybrid cloud platform. Service Architecture is also referred to as the Service Motif in the Auto Scaling system indicates that sub-modules are required to perform the autoscaling. The SLA Driven VM Auto-Scaling Algorithm includes the Run-time Scaling and the SLA monitoring, Scheduling for performance optimization. A new auto-scaling model to apply the discussion of virtual resources on the cloud for real time scenario tasks has also been discussed in the paper by the authors and the functional concepts in auto-scaling which will also includes the Executer, Monitor, Planner and Analyzer. Auto scaling method provides the means by which an application is run to meet the given time or the deadline. A Auto-Scaling Algorithm are composed of Runtime-Scaling and the Performance based oriented Scheduling system or the Algorithm. Some of the logtraces of the auto-scaling of the data centers cluster such as Google data centers which are Auto-scaling Demand in the Index metric, which are also used in the auto-scaling system strategy. The Adaptive strategy for an appropriate mode selection of scaling steps in the auto-scaling system operation are also included with the dual step size configuration technique: constant(for ordinary utilized parameters) and Adaptive (for irregular and peaky system load). The very Different approaches, such as can be used to trigger Auto-scaling, are classified as Predictive, Conservative, Reactive and Approaches.The Load balancing in Amazon EC2 is available through Amazon ELB which is abbreviated as Elastic Load Balancer services, the technique which also provides high the resources of EC2 instances and optimizes EC2 applications available on the web, which can be made available to the application users by directing the incoming application traffic to multiple instances on different availability zones. EC2 consists of OS like Open Solaris, Red Hat, Open Suse, Ubuntu, Linux, on Windows, Suse Linux, Fedora, etc. Any users can be given the web interface to manipulate or communicate with the EC2 and the interfaces are a set of SOAP messages. Elastic load balancer offer high availability of EC2 instances and also which improves the on Going cost of elasticity of EC2 application availability via depending all incoming applications traffic across 'instances' It seems that other researchers have also noted that self-correcting features can not only help to traffic across multiple instances. The Elastic load balancing would also recognises subordinate, and it diverts the traffic as per the requirement and necessary required. When an application arrives, Amazon EC2 would intelligently distributes the load it to generate traffic between multiple instances through the feature of Elastic Load simple monitoring or surveillance method using the Cloud Watch methods with the high scaling policies.

## 2.4 Smart Parking Review

In the research paper (Mičko and Papcun 2023) focuses on classic computer vision approaches adapted for CUDA cores of Jetson Nano without the usage of deep learning techniques. Python and MongoDB are both applied to the backend development since Python is highly elastic and MongoDB has document-based storage. The employed open dataset revealing the performance of our system yields high overall accuracy, precision, recall and F1 measures, providing proof of concept towards realistic urban parking situations. Besides offering a perfect solution for managing the parking spaces, this work also provides the impetus for applying similar approaches to traffic volume assessment and urban design.

The application of Cloud Computing (Mnyakin 2023) can enable vehicle-to-cloud connection, elasticity, computational and storage capacity, transportation as a service, and prognostics and health management. The combining of these technologies can lead to the Cloud Computing can be applied for vehicle to cloud interface, scalability, computation and storage of data, transport as per demand and, diagnosis and prognosis of vehicles. Combined these technologies can lead to the smart transportation system which can enhance the transport system further.

The researcher demonstrates the Quick Response with the purpose (Buian, I. M. Siddique and A. A. Siddique 2024) of encoding information, the type of code is known as a QR code. Because of its specificity, QR codes are used for the allocation of the areas in real-time. The system identifies it and permitted while examining the code of the car which wants to enter the premises. After the scanning the slot is assigned to the user provided there is an available slot left and after the vehicle leaves the slot. By so doing, we will be in effect using less elaborate tools and applications to reduce human labor. This system shall be responsible for the authorization as well as the selection of spots. access parking system that would makes utilize of the ESP8266. The parking systems are developed, this kind of website which can enable the user to get the access the level of occupancy or spots. It also helps the device to store the information of the user in a database and park it quickly. availability, and consequently traffic jams have reduced. Hardware are developed using an ESP8266,which are an infrared sensors.

The research (Channamallu et al. 2023) involved with extensive analysis of 124 published journals, research papers and the articles revealed an interesting trend: The growth and implementation of IoT-based SPS in context of urban environment are affecting decision-making process and improving the worth of user experience. Wireless sensor networks are now one of the most common SPSs among a wide variety of options because, in addition to not requiring a large number of wires, the costs of acquiring and installing them are also reasonable. The detection of vehicles requires fewer sensors but heavier computer vision or image processing system including huge installation charges due to camera system.

Here in the research(Aditya et al. 2023) As one of the major challenges of smart city, this paper aims at employing IoT based approach to detect the availability of parking space. The proposed Idea of Intelligent Parking System (IPS) incorporates Internet of Things (IOT) which gets real time data, transmits it to cloud and then suggests the user best available place in a particular location where he or she can park his or her vehicle at a particular location nearby. As the one of elements of the proposed framework, there is a mobile application developed which can inform the user about the number of parking places available in near vicinity and then allows to sign up for parking place area. The paper describes several scenarios with an individual for searching a parking space area and the parking it at the right parking area. Here proposed system have been carried out by utilizing the NodeMCU,Raspberry Pi and IR sensor and the RFID, as described in some of the later sections, the results which have been presented here support the viability of this IPS.

Table 1: Research Summary of different papers studied

| **Name** | **Author** | **Results** |
|---|---|---|
| Optimized real-time parking management framework using deep learning | (Rafique et al. 2023) | Here based on the proposed datasets performance of the suggested system was verified which resulted in an overall accuracy of 96.8% and nearly it was a real time. The capability of 45 frames per second which can make more portable flexible and also makes computationally optimized. |

| Name | Author | Results |
| --- | --- | --- |
| A shared parking optimization framework based on dynamic resource allocation and path planning | (Xie et al. 2023) | When it is compared to the Rolling parking Routing[RPR] and the Static Parking Routing[SPR], the built GPR-A* algorithm which works with the parking allocation can save the travel time by an average of 5.5% and 14.9% |
| Detection of car parking space by using Hybrid Deep DenseNet Optimization algorithm | (Rajyalakshmi and Lakshmanna 2024) | The part of Adam has enabled the HDDNO model to generate predictions with high accuracy 99.19% and low loss 0.0306%. This proposed methodology would significantly improve environmental safety and act as an initiative toward developing smart cities. |
| Internet of things and cloud based smart parking system design criteria | (Türker 2023) | The Adam's effort has allowed the HDDNO model to generate the predictions with a low loss of 0.0306% and with a high accuracy of 99.19%. This described approach would serve as a first step towards the developing smart cities and significantly improve the environmental safety. |
| A review of smart parking systems | (Channamallu et al. 2023) | The results of study would assist the researchers in pointing areas that need more and more investigations and the city planners in incorporating smart parking systems into bigger smart city projects. |
| Parking Management System Based on Image Processing Technology | (Xing and Long 2023) | This project has also addressed that issue of traditional parking space lock intelligence have made parking management more convenient |

## 2.5 Research Gap

Modern parking systems have the following research limitations which make it hard to optimize them for smart city use: Other existing systems such as SPR and RPR do not incorporate real time slot detection with high degree of accuracy as incorporated in this system. More specialized versions of YOLO exist, including YOLOv8 and YOLOv9, whose abilities to increase real-time detection of available parking spaces have been left unutilized. Scalability and latency are the other technical factors; established solutions are highly dependent on cloud services; this leads to latency difficulties within high population density areas. The integration of edge computing, possibly in a combination with cloud and dew computing could potentially decrease latency, enhance scalability and optimization of resources, however, little is known about this issue. There are social and environmental effects, although it is not clear that economic effects occur quantitatively. It is another research direction for further work to incorporate the parking data for assessing the traffic conditions, pollution levels and fuel efficiency. Likewise, access to parking through IoT has the same problem of security and privacy which can be solved by encrypted YOLO model result and a distributed approach. Dynamic parking management using adjustable slot allocation incorporating machine learning and YOLO models is feasible, though most existing systems lack incorporation of these. Furthermore, a significant problem is that the development of the user interface is under emphasized, and obvious shortcomings in usability exist. The former could prompt the development of highly sustainable, secure, and friendly smart parking services to fill these voids.

## 2.6 Research Outcome

The outcomes of the research can contribute to designing state-of-art, optimized, and deployable smart parking systems that integrate well with smart city conditions. These systems will bring real-time, high-accuracy YOLO models for parking slot detection to edge and hybrid computing with the lowest latency, optimal computing resource utilization. Qualitative has widely defined measurable benefits of smart parking by proposing measurable surpluses of environmental and economic effects, including traffic proliferations, emissions, and fuel. Improved security solutions such as encrypted prediction and distributed data handling will promote the use of IoT based solutions. Flexible dynamic use of machine learning to allocate the slots will enhance parking through better observation of demand rates at specific times. It will also improve the application adoption due to convenient slot discoverability and navigation resulting from user-centric application designs. Altogether these milestones will contribute to the developments of smart sustainable urban transport and foster smart city solutions. Also, the number plate detection we have to make a generic solution across countries and different weather conditions.

# 3 Methodology

## 3.1 Dataset Description

### 3.1.1 Parking Space Dataset

The Dataset contains pictures of parking spaces, along with bounding box annotations for each of the image. Bounding boxes is being utilized to establish the precise limits of each parking space in the images, aiding in accurate localization and identification of individual parking spots. The masks of bounding boxes allow for the accurate determination of available, taken, or partially taken parking spaces by highlighting the shape and position of the spots. Every parking spot is given a classification indicating if it is available or taken, which allows for the testing and training of the suggested model for identifying and categorizing parking spots automatically.

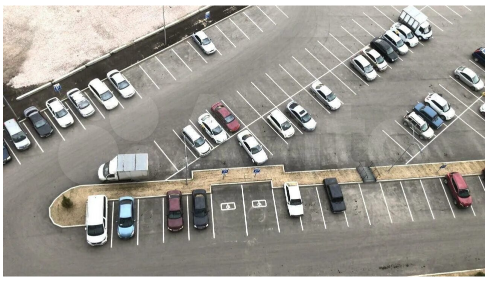
### 3.1.2 RoboFlow Parking Detection Dataset (PkLot)

The RoboFlow Parking Detection Dataset comprises a total of 5,565 images with bounding boxes for detecting parking spaces. The dataset is split into three partitions: 70% the images (3,903) for the training, 15% (832 ) of images for the validation, and 15% (830) of the images for testing.
In terms of preprocessing, auto-orientation is applied, and all images are resized to 640×640 pixels. Various augmentations are employed to enhance the model's generalization capabilities, including rotating images clockwise and counterclockwise, applying upside-down transformations, adjusting brightness by plus or minus 25%, changing exposure by plus or minus 10%, and adding noise to up to 2% of the image pixels.This comprehensive setup is designed to create a robust model for detecting parking spaces. The images taken are snapped from specific cam positions stabilizing in position and postures which provides for cases to be solved by machine learning algorithms. The audience primarily employs PKLot for verifying parking slot identification, occupancy status, prediction and actual parking lot. This is largely accessibly for public use for especially for academic use and is useful for development of intelligent transportation and smart parking systems.

### 3.1.3 ANPR Dataset

The design of ANPR systems in India is challenging because license plates are of different sizes, shapes, fonts, and scripts. In India if person owning a vehicle in Delhi his number plate would start from DL and If a person owning a vehicle in Bengaluru his number plate would start from KA naming convention so it is a biggest challenging task. Developing effective ANPR solutions requires a diverse dataset to capture these variations. However, without a comprehensive dataset specific to Indian license plates, progress in the creation of accessible and reproducible solutions has been slow-moving. While countries like the United States have developed extensive datasets like The Chinese City Parking, The Connected Car Payload Detector (CCPD) and the Application-oriented Number Plate (AOLP)., India didn't have an equivalent. To bridge this gap, a new dataset is currently being designed specifically for the Indian market, with 1,500 images presently being used to support the development of ANPR systems.

Below are the sample images



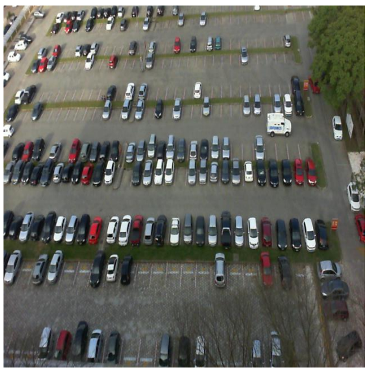


Figure 2: Dataset Images

## 3.2 Transfer Learning Models

### 3.2.1 DenseNet

DenseNet consist of the dense blocks, in the transition layers size of the feature maps are decreased so that the model have control over the size and the computational complexity. This is the architectural requires fewer parameters, provides rigid stability during training and is suggested for use especially in areas related to feature extraction such as in application like image and even object detection.

### 3.2.2 ResNet

Typically, ResNet has four types of blocks; however, with an extension in the network's depth, more residual blocks are placed. First, at the initial stage of the network, the number of layers are one of the convolutional layers, which would analyses the input data and one of max pooling s layers. Pooling layer which reduces the size of input data in spatial dimension and that makes the network to be efficient. The residual blocks in this part will contain 64 filters and will allow to receive simple features from the initial data. As for the third part of the network there will be 128 filters which the net allows to learn finer and more complicated patterns. The last one is the fourth part of the network with 256 filters, so the network can learn less significant details.

### 3.2.3 MobileNet

The architecture which includes an initial convolution layer, followed by the series of these efficient layer, with ReLU activation and the batch normalization for an non-linearity and the stability. At the end, global average pooling condenses the features, and a fully connected layers would produces the final output. This Adjustable settings like the width and resolution multipliers allow further customization for different devices and tasks.

### 3.2.4 VGGNet

VGG is abbreviated as Visual Geometry Group, it is also an typical or the classic deep convolutional neural network (CNN) design comprising numerous layers. The network's depth is attributed to its numerous convolutional layers, with examples such as Visual Geometry Group VGG-16 is having a 16 layers and Visual Geometry Group VGG-19 consist of 19 layer each. The use of compact 3x3 convolutional filters is efficient in capturing intricate details, and the quantity of filters increases two-fold following each max-pooling layer to gradually understand more intricate features.

### 3.2.5 EfficientNet

EfficientNet is a deep learning model designed specifically for image recognition, with high accuracy but fewer resources. Its architecture is based on a concept known as compound scaling, which gives a balance between the width, resolution and the depth of the developed network, that would also be the number of layer or the number of channels and input image size, respectively. EfficientNet is based on mobile-friendly blocks, known as mobile inverted bottleneck convolutions (MBConv), to achieve efficiency.

## 3.3 YOLO Models

### 3.3.1 YOLOv3

YOLOv3 by Joseph Redmon has many more advanced features than its predecessors. It supports the detection of objects at three scales in one network pass. It supports multiscale predictions using detection kernels of different sizes. Being a fully convolutional network, it uses the Darknet-53 as the backbone for the feature extraction. It comprises the the Leaky ReLU activations and 53 convolutional layer with the batch normalization. Rather than traditional pooling down sampling, YOLOv3 relies on stride-2 convolutional layers for down sampling and retaining the low-level information lost by traditional pooling, thereby increasing the detection accuracy. YOLOv3 provides mAP and IoU scores similar to those offered by competitors while being relatively faster to run, and therefore well-suited to real-time applications.

### 3.3.2 YOLOv5

YOLOv5 are the highly popular object detection model. It is efficient and accurate. The network efficiency of this model is improved by using a modified version of ResNet called Cross-Stage Partial (CSP) Net. The CSP connections split the feature maps, thus enhancing the gradient flow and minimizing redundancy.To this end, the model incorporates Spatial Pyramid Pooling (SPP) blocks in order to capture features at different resolutions. It has a Path Aggregation Network (PAN) at its neck and up-sampling layers for enlarging feature map resolution to enhance its small object detection capacity. The head is a convolutional structure which estimates the class probabilities, along with the bounding box. Another kind of prediction adopted by YOLOv5 is the anchor based approach where the boxes belonging to different shapes and sizes are bound to corresponding anchor shapes and sizes for accuracy.

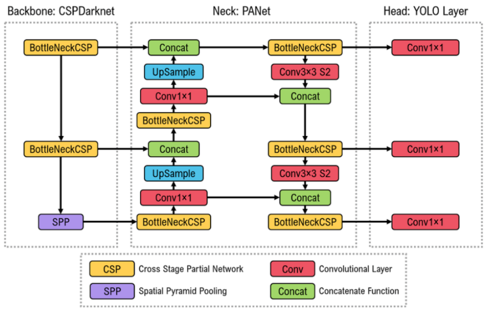



Figure 3: Architecture of Yolov5 (Katsamenis et al. 2022)

### 3.3.3 YOLOv8

YOLOv8 are the highly efficient model and also a accurate model, especially suited for the resource-limited environments like edge devices or real-time applications. Its architecture is divided into 3 important components Head,Neck and Backbone. The Backbone network applied in the system employ a Modified CSPDarkNet53 which comprises of 53 convolutional layers and CSP connections to improve on solid feature extraction and optimization. It incorporates C2f modules for combining high-level and contextual information and an SPPF module for capturing multi-level features with minimal computational cost. The Neck features a Path Aggregation Feature Pyramid Network (PA-FPN), which improves multiscale feature fusion. By combining upsampling and C2f modules, it processes spatial features from various levels (P3, P4, P5), enables the better detection of the objects at different sizes. The Head converts the features into bounding box coordinates and class probabilities. It uses specialized branches for scoring, classification, and regression, ensuring precise and fast predictions.

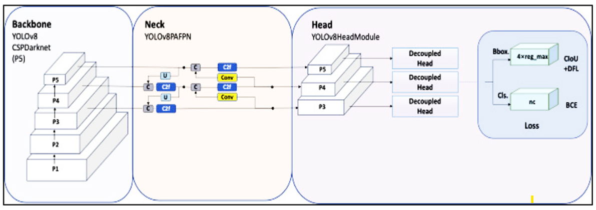



Figure 4: Architecture of Yolov8 (Sohan et al. 2024)

This optimized structure enables it to perform the object detection with unparalleled speed and with very little amount of resources which makes it perfect for the real-time scenario applications like traffic control, identifying vehicles on the road or even ease the detection of pedestrians and it is also a pedestrians safety.

### 3.3.4 YOLOv9

YOLOv9 is a another best model where further enhancement of the strengths of an earlier YOLO versions with the new methods to makes the efficiency and accuracy of an object prediction process better. Issue is of information loss in the network layers of the data is addressed by integrating two main features are innovative features: which is based on the Generalized Efficient Layer Aggregation Network and theProgrammable Gradient Information, both of which improve gradient stability as well as increase prediction accuracy.

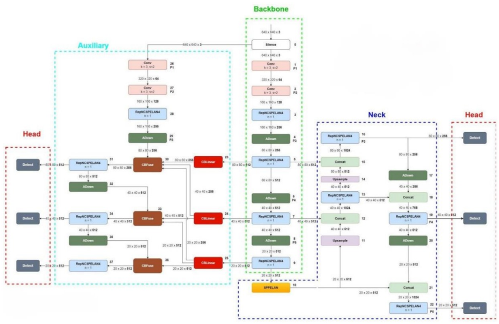



Figure 5: Architecture of Yolov9 (Imran, Hulikal and Gardi 2024)

YOLOv9 architecture applies vision transformers especially designed for recognizing long-range dependencies in the data of visual streams. Thus, this modification improves the presentation of features which may elevate the accuracy in object detection. However, the applicability of ViTs involves additional costs in terms of computation. Hence, it may worsen the efficiency in certain scenarios. Nevertheless, YOLOv9 represents a landmark step toward attaining an excellent balance between performance and precision for complex tasks of detection.

## 3.4 Cloud Solution Framework

### 3.4.1 Implementation

Technological Framework The Structural integration of the IoT with an ANPR technology with cloud computing produces an Automated Parking Parking solution. Using the deep-learning methods models like YOLOv8 and YOLOv9 would analyze data in real time which can enable the or predict the precise number plate recognition and also detecting parking slots. The AWS services including the EC2 instance and Load Balancer and Auto Scaling group can enable the both fault tolerant and scaling the deployments. The main trackable parking space availability detection can be made using the Cameras. The ANPR Enables the automated entry and the exit through an System Framework. The system can also link the user payment methods to computerized fee collection without involing the human beings. The Use of transfer learning models such as DenseNet ResNet and the MobileNet which can enhances the detection of accuracy for resource constrained environments.
The Real time parking availability detection utilizes Camera Feeds as the system component. Automatic Slot Assignement selects vehicle locations according to vehicle size and park slot availability. The system implements ANPR technology to automate station entries and exits. Centralized database and the analytics for the optimizing operations and the urban planning.

### 3.4.2 Advantages

- A system unites automated space management with fee payment features During implementation the system technology minimizes human interaction while providing better traffic flow and reducing parking search congestion. The automation system enables better traffic handling by decreasing spot-search congestion in parking zones.
- A Real time slot data and Automatic reserving capabilities can overcome the parking difficulties and the access application based parking searching user friendly functionality.
- The Environmental goals which can support through this system because it helps minimizing the driving waste while decreasing the emission which can lead to improved urban mobility.
- The system achieves scalability through its adoption of cloud infrastructure which can maintain the efficient control of changing traffic patterns and also expands the user needs.

### 3.4.3 Challenges

- The Main challenge here is when the operators perform within the adverse weather condition as fog or rain. Tests must be performed in the various environmental conditions for the system to gain the reliable outcomes. The cloud can deplete financial resources because of continuous usage periods during the peak usage.
- Initial deployment costs for the Cameras, Sensors and the Cloud Infrastructure could be more expensive.
- Handling Sensitive data would requires the robust encryption and the compliance with the data protection regulations.

## 3.5 Evaluation Matrices

### 3.5.1 Confusion Matrix

A confusion matrix represents the excellent tools for ascertaining accuracy of a model's predictions against the actual labels of ground truth. Such a matrix defines the false positive, false negative, true positiveand misclassification, which occur as off-diagonal elements. Rows of matrix which can corresponds to the exact classes, whereas the column will represents the predicted class. Accurate predictions are thus represented along the diagonal. This matrix enables accuracy of individual classes to be estimated, patterns associated with errors to be identified, and insight into how the model may misclassify similar classes.

### 3.5.2 Accuracy

Accuracy is the measure which is being used for the evaluating of the performance of classification models. It would determines calculate the percentage of the instances that has been accurately classified of the entire data instance. Accuracy = (Number of Correct Predictions)/(Total Number of Predictions)

### 3.5.3 Precision

The Precision quantifies how correctly a model is able to predict positive instances to a specific class when targeting the minimization of false positives such as instances which are wrongly classified by the model-as related to that class but is actually from some other class, or is negative.
Precision = (True Positives)/(True Positives + False Positives)

### 3.5.4 Recall

A Recall is an evaluation metric which measures how well a classification model finds all true examples of a specific class. It checks how good the model is at recognizing the real positive cases by reducing false negatives, which happen when the model does not correctly classify a true example of a class.
Recall = (True Positives )/(True Positives + False Negatives)

### 3.5.5 F1 Score

F1 score is the performance metric for classification models which can also gives a fair assessment of a model's accuracy by combining or mixing of the precision and recall into a one value. Also It guarantees that precision and recall are given equal weight by taking the harmonic mean of the both. A higher F1 score indicates a better classifier; its value ranges from 0 to 100%.
f1 Score = 2×(Precision ×Recall)/(Precision+Recall)

# 4 Implementation

### 4.0.1 Step 1: Automatic Parking Slot Detection

- Data Acquisition: Images are collected from cameras or sensors installed in parking lots. Cameras are designed with wide field of views so that they can be able to capture multiple parking spaces in one frame, while the limits of each spot are clearly evident. Datasets such as the Roboflow Parking Detection dataset and the PKLot dataset are publicly available. PKLot contains images of more than 12,000 parking lots and nearly 700,000 segmented images of parking spaces. This dataset offers fine-grained data for analysis. The Roboflow dataset provides annotated images for parking space images with a bounding box. A combination of these datasets ensures a rich and diversified collection to effectively train and test parking space detection models.
- Data Preprocessing: All such images received need to be pre-processed towards higher quality enhancement and noise reduction. Typical procedures of pre-processing include all these images to resize their dimensions into a uniform one, converting them into grayscale data, and enhancing their contrast followed by enhancing the visibility of features. These are required steps to make appropriate arrangements in the images during training detection models. Besides this, the dataset Roboflow Parking Detection, was already preprocessed and annotated, therefore providing ready-to-use images that require no further tremendous cleaning or labeling.
- Object Detection models for Parking Slot: First, various transfer learning models, namely DenseNet, ResNet, MobileNet, VGGNet, and EfficientNet, were evaluated in this project is improvize the object detection, efficiency of parking and accuracy spots. All models have been pre-trained features on bigger datasets, meaning faster convergence and for the better performance even with the limited data, as per the learned features which have been used and contributed to the task at the other hand. The Object recognition of the algorithms are applied using the transfer learning methodologies, such as the YOLO (You

Only Look Once), for the real-time classification & the detection of things. Especially, here the models, such as YOLOv3, YOLOv5, YOLOv8, and YOLOv9, are the most efficient to find open parking spots instantly. Then, these algorithms, such as thresholding, contour detection, and morphological operations, were also applied in order to separate each parking spaces from the surrounding region with clear accuracy and the vision related to the location. These methodlogy work together to make the strongest system for finding parking spaces and which can be applied in time-based situations aswell.

- Predictions: All the models which have tested on the new images, after being trained, to evaluate their performance and detection of thei capabilities. Transfer learning models are used for density prediction: DenseNet, ResNet, MobileNet, VGGNet, and EfficientNet; they utilized features learned during training which can determine parking spaces and which can make same as the almost accurate predictions with the given limited data. The YOLO models - YOLOv3, YOLOv5, YOLOv8, and YOLOv9 - are evaluated in terms of the speed and accuracy in real-time world detection by generating bounding boxes around the parking spots along with the confidence scores. Below are of Following predictions, contour detection for segmentation fine-tune the detected spots and NMS techniques remove overlapping boxes while thresholding.
- Model Evaluation: A detailed performance evaluation using a confusion matrix and the several key performance metrics which have been applied to all the models for thorough analysis. Here The confusion matrix gives a very clear idea or the overview of how the model have actually behaved, showing the false positives-the incorrectly identified occupied spaces-and and true positives-the correctly detected parking spots-and true negatives-the correctly identified empty spaces false negatives-the missed parking spots, which have been based on this key performances metrics which are computed such as the accuracy, recall, precision and the F1 score.

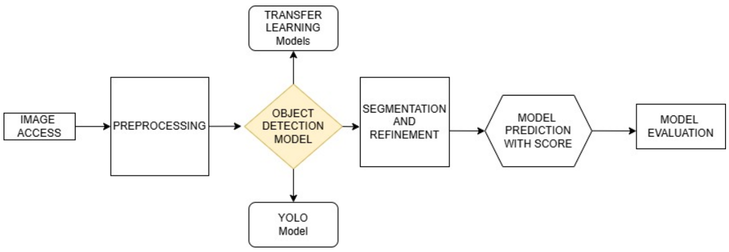



Figure 6: Flow Diagram of My Project

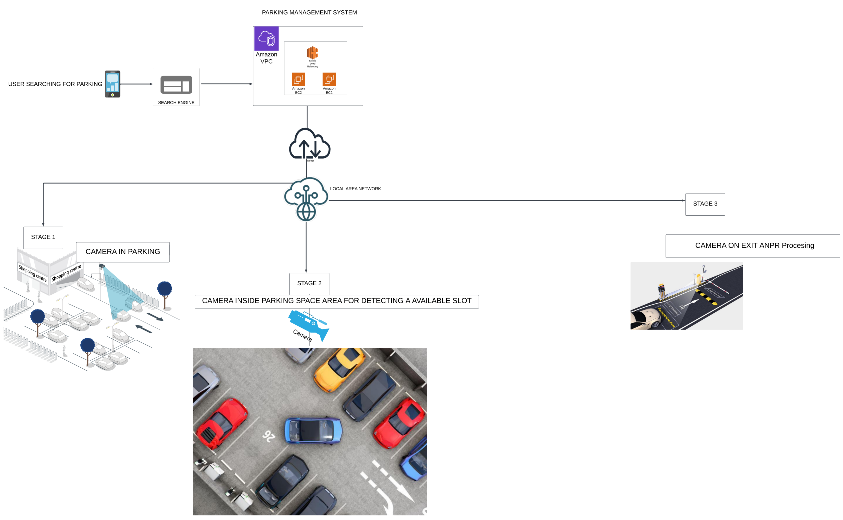



Figure 7: Implementation Flow for the Parking Solution of my Project

## 4.1 Step 2: Automated Number Plate Recognition

- Dataset Preparation: The Roboflow Python API will join the project of "Vehicle Registration Plates," which exists in the "Augmented Startups" workspace. It will be identifying its version as version 2 to get the proper version of the dataset. The ‘version.download(”yolov8”)‘ will then download the dataset in the format of Yolov8. This makes the dataset pre-structured, with all the namely annotations, necessary components, configuration files and the images and like ‘data.yaml‘, so that it is now ready to be trained on it. The datasets are probably images of vehicles with number plates annotated on them, thus it makes a strong foundation for training the ANPR detection model.

- Model Selection: This project utilizes the YOLOv8 framework as it is the latest highly efficient algorithm used for object detection. It makes use of a pre-trained model for this detection; specifically, a small variant known as yolov8.
- Model Training: The training process is very much essential for the YOLOv8 model to detect number plates accurately and correctly using or with the help of the given datasets. The training process is usually done by the data.yaml file, which contains the training data and classes that the model has to recognize the object. This model is trained for 50 cycles, known as epochs, on images that are resized to 640x640 pixels. Here in the Model training the learning rate is usually set to the 0.01 in order to control how the model would adapts its learning during training. Data augmentation would also used by included in the variations in the training data in order to improve the effectiveness of learning by model and also generalize well to the new images. Here It also generates very much important performance metrics such as the loss which would indicates the recall, precision,error, mean average precision which would also reflects how well it can capable of learning.
- Model Validation and Testing: After the training of the model, it is usually validated and tested on new, unseen data to measure its performance. Validation evaluates the model based on the test set set in data.yaml, which also in turn computes key metrics such as mean Precision, inference speed,Average and the recall. The validation is also performed with a test image size of 640, a batch size of 16, a threshold to classify the confidence for the identification of objects at the setting of 0.25, and an IoU or the Intersection over Union set up to a threshold value of 0.6 with the optimized detection results in comparison. The same configurations of testing ensure the consistency when creating, identifying the outputs to let the users display the extent to which such models identify or detect the number plates out in real-life scenarios.
- Benchmarking: Benchmarking the model for how well it would be performed with respect to its deployment readiness will be performed. The Benchmark functions are usally used in measuring the parameters such as speed of precision, utilization of memory and the computation. With regard to the improved performance, one tests of the model on half precision floating-point operation (half=True), which will also increase its inference speed while utilizing less memory without the necessarily sacrificing the accuracy levels. This would also ensures that the model is well-optimized and ready for the real-time application deployment.

## 4.2 Step 3: Consolidated Approach using Streamlit and Flask

The system is developed around two key objectives, that is, parking space identification through images of parking lots and extraction of text from the vehicle registration plates. Both the objectives are very much relevant to the automation of the parking system and the efficiency of urban infrastructure. It applies advanced deep learning techniques specifically object detection and Optical Character Recognition (OCR), to provide accurate and reliable results.

- Object Detection with YOLO: For both tasks, the we use the YOLO (You Only Look Once) framework, which is known for the real-time object detection capabilities and high precision. YOLO’s architecture allows it to predict the class probabilities and bounding boxes simultaneously, making it as efficient for identifying multiple objects in an image. For parking space detection, a custom YOLOv9 model was trained to classify parking spaces as free or occupied. A separate YOLOv8 model was optimized for detecting number plates in vehicle images. These models were fine-tuned on the customized datasets to improve their accuracy in these specific tasks.
- Preprocessing and User Interface: The platform utilizes Streamlit to construct its interface, which is well-suited for developing interactive web apps. Users have the option to upload an image and select from two tasks: identifying parking spots or identifying and reading license plates. Once the image is uploaded, it is prepared and transformed into a compatible format for YOLO model processing. The models create bounding boxes for identified objects, which are superimposed on the original image for visualization purposes.
- **TASK 1: Parking Lot Detection** : The YOLOv9 model creates bounding boxes around parking spaces during the parking lot detection task. The bounding boxes are screened using a confidence threshold defined by the user to guarantee result accuracy. The identified parking spots are then sorted into two groups: vacant or taken. The results are shown in a table using pandas, illustrating the number of open and filled spaces in the image that was uploaded. This gives users a fast summary of parking availability.
- **TASK 2: Number Plate Detection and OCR** : In order to detect number plates, the YOLOv8 model initially locates the regions of interest that contain number plates. Next, the image is cropped to isolate these regions before being analyzed using the Optical Character Recognition (OCR) pipeline for extracting

text. Default system utilizes EasyOCR, an OCR library based on deep learning that can recognize text in various languages and fonts. Moreover, Tesseract, a freely available OCR engine, can be used as another option. Prior to being inputted into the OCR system, the cropped images go through preprocessing steps like being converted to grayscale and undergoing binary thresholding in order to enhance text readability by boosting contrast.

- **TASK 3: Code Deployment** : For deployment of the project we started by clonning the repository through git, then creating and also activating a Python virtual environment then the requirements for the project were installed. I then followed that up by installing Node.js and pm2 which I use for process management. The project was initiated with pm2 so as to make it run continuously. I checked it with the below command.

  pm2 list and set up auto-restart.

# 5 Results And Analysis

## 5.1 CASE 1. Parking Lot Classification Analysis

### 5.1.1 Using YOLO Models

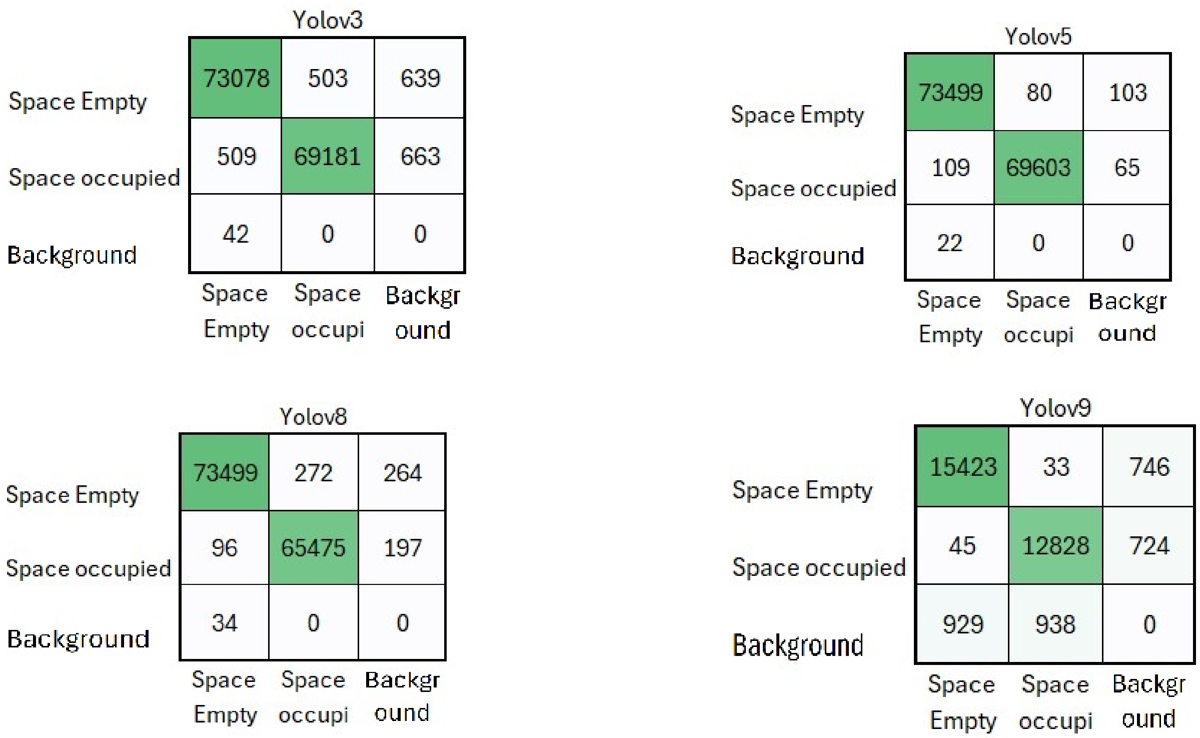



Figure 8: Confusion Matrix of YOlO models-Yolo3, Yolo5, Yolo8, Yolo9

The performance of YOLO models—YOLOv3, YOLOv5, YOLOv8, and YOLOv9—was assessed using confusion matrices for three classes: "space-empty," "space-occupied," and "background" (or similar, for example "car" and "free" in the case of YOLOv9.) Several of these models were shown to have their own set of advantages and disadvantages based mostly on how effectively they could differentiate between these classes and the issues that can arise from the "backgrounds" class. YOLOv3 performed well in classifying the instances of space-empty (73,078) and space-occupied (69,184) but many of the misclassifications were observed. For example, 503 instances of "space-occupied" were labeled as "space-empty," and 509 cases of "space-empty" were labeled as "space-occupied." Likewise, pre-estimation of background class To conclude, table1 shows that 'background' consist large number of errors: 639 samples were classified as 'space–empty'; 663samples were classified 'space–occupied.' These mistakes shed the light on a problem of class separation, especially the "background" class. With YOLOv5 there were improvements in the true positive and a decrease in false positives. It learned space-empty: 73,499 and space-occupied: 69,607 . There were few mistakes, only 80 samples of space-empty were classified as space-occupied, and 109 samples of space-occupied were classified as space-empty. The "background"

class showed very few misclassifications: Only three as “space-empty” and a few as other classes only twenty one. In general, YOLOv5 provided longer and more accurate detection, as well as improved recognition of what the model defines as “space-empty” and “space-occupied.”
“Space-empty” achieved good detection with 73,499 while “space-occupied” was 69,415. Misclassification errors to some extent were identified as 96 of ‘space-empty’ was recorded as ‘space-occupied,’ 272 of ‘space-occupied’ was recorded as ‘space-empty’ while more concerning was 34 instances of ‘background’ that were misclassified and 197 instances of ‘space-occupied’ misclassified as ‘space-empty.’ Nevertheless, overall all metrics for the YOLOv8 were good, particularly for the contrast between “space-empty” and “space-occupied.” YOLOv9 was also applied to “car,” “free,” and “background” classesondere predominantly on the first two categories, but encountered a lot of issues with “background.” It was able to well distinguish 15,423 “car” and 12,829 “free” and there were very fewer. Accuracy of number counts ranged from 0.89-0.94 with misclassifications occurring between the two primary classes of “car” and “free”. Nevertheless, the “background” class was diagnosed with 919 misclassified data samples as “car” and 774 misclassified as “free” due to the fact that this class was hardly distinguishable from the others, potentially due to imbalance of the classes or insufficiency of the differences in features.
There were improvements in all the models with YOLOv5 and YOLOv9 having higher accuracy, all models however struggled with what they classified as “background”. A detailed analysis of the improvement of the ‘background’ differentiation in the yolo models by performing class balancing or even feature enhancement will be helpful for improving the overall performances of all types of yolo models.

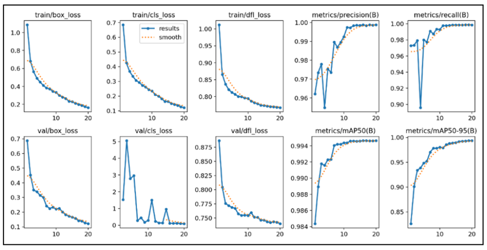


Figure 9: Model Graphs while Training the YOLOv3

The validation and Training metrics over the 10 epochs shows the clear trend of improvement in the YOLOv3 performance. Validation loss and the training, including classification loss, distribution focal loss and box loss was observed to decrease steadily. This proved that the model was gradually improving its prediction capability. Meanwhile, relevant mean Average Precision SaNQ (mAP), recall, and performance measures-precision increased. This suggests that the model would becomes much accurate in identifying and classifying object. Adding to this, with improved mAP across different threshold settings in Intersection over Union (IoU), the advancement indicates that the model has matured to detecting objects accurately in diverse conditions. Overall, these metrics depict that the model is learning quite well to generalize well over the new and unseen data.

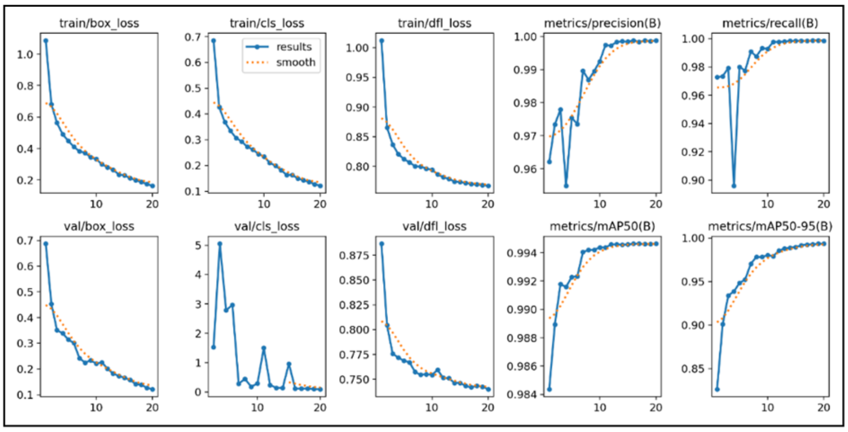



Figure 10: Model Graphs while Training the YOLOv5

The model makes progress, with improvements in both its validation and training metrics over the 20 training epochs. Training losses, ‘box_loss‘, ‘cls_loss‘, and ‘dfl_loss‘, are monitored and show an ongoing trend of lowering, indicating proper learning. In particular, reducing ‘train/box_loss‘ implies that model would be improving in its capability or the ability to locate objects; that is, it is better able to find where objects actually are. The drop in ‘train/cls_loss‘indicates or gives an idea that the model is learning to correctly classify objects, while a drop in ‘train/dfl_loss‘ gives a clear indication or that the model is improving in its predictions over the bounding boxes.

The validation losses, including ‘val/box_loss‘, ‘val/cls_loss‘, and ‘val/dfl_loss‘, also show a drop, which means that the model is generalizing very well to new, unseen data. Although there is some fluctuation in ‘val/cls_loss‘, it anyway remains positive in the trend, meaning the model was not overfitting its training data and was doing a decent job in the validation data as well.

Finally, the evaluation metrics that include precision, recall, ‘mAP50‘, and ‘mAP50-95‘ all increase through iterations.

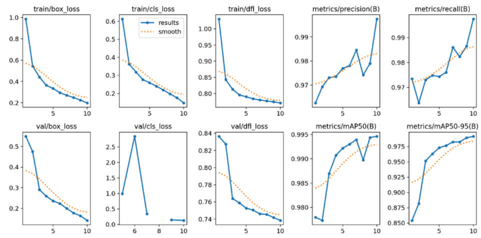



Figure 11: Model Graphs while Training the YOLOv8

Over 10 epochs, the training metrics show consistent improvement. The train losses, ‘box_loss‘, ‘cls_loss‘, and ‘dfl_loss‘, decrease gradually because that is when the models would be learning well. A downward trend in ‘train/box_loss‘ reflects better localization of objects, while the decrease in ‘train/cls_loss‘ suggests better classification. ‘train/dfl_loss‘ decreases too, meaning the model is now capable of predicting better boxes. As is consistent with the training metrics, ‘val/box_loss‘, ‘val/cls_loss‘, and ‘val/dfl_loss‘ decrease too, indicating the model will be generalizing well to new data, although ‘val/cls_loss‘ fluctuates a little. The evaluation metrics—precision, recall, ‘mAP50‘, and ‘mAP50-95‘—increase, showing better object detection. Overall, these trends suggest the model is training well with higher accuracy and good generalization.

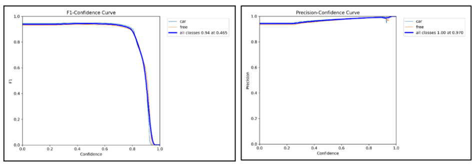


Figure 12: Confidence Curve for a YOLOv9 model, Precision-Confidence Curve for a YOLOv9

The amount of time the YOLOv9 classification model takes to make the classification across the different confidence levels is also as shown below; The maximum area of 0.94 is achieved with negligible to moderate confidence of 0.465, and both precision and recall metrics are balanced for the “car” and “free” classes. F1 scores remain high across a wide range, but its higher rates are lower than those that decrease when recall is reduced by higher confidence. There is a gradual enhancement of precision as thresholds rise, signifying the reduction of false positive circumstances in that, perfect precision of 1.000 is achieved at 0.970 likelihood. The metrics trends are smooth for both ‘car’ and ‘free’ as shown in the following model, which performs equivalent for both terms. The patterns pointed out above are useful to particularize the most appropriate confidence levels to meet a proper trade-off between accuracy and selectivity.

## 5.2 Using Transfer Learning Models

| Model | F1 Score | Precision | Recall | Accuracy |
|---|---|---|---|---|
| DenseNet | 99.59% | 99.41% | 99.49% | 99.47% |
| ResNet | 99.32% | 99.08% | 99.20% | 99.15% |
| MobileNet | 98.81% | 98.72% | 98.77% | 98.76% |
| VGGNet | 98.63% | 98.51% | 98.57% | 98.55% |
| EfficientNet | 98.78% | 98.92% | 98.85% | 98.89% |

Table 2: Performance Metrics of Different Models

- **Table 5.1: Performance comparison of different models**

DenseNet excels in spotting a parking spot as it achieved an accuracy rating of 99.47%. Additionally, both its precision and recall stood at 99.59% and 99.41%, respectively, meaning the F1 score was at 99.49%. All these findings suggest that DenseNet highly relied on consistency in error-free spot detection. Their dense connections help recycle features back into layers to learn patterns and avoid errors related to training. ResNet shows good efficiency, with an accuracy of 99.15%. The model's precision is at 99.32%, and the recall is slightly low, at 99.08%, which sums up to an F1 score of 99.20%. The introduction of residual connections in ResNet helps in training deep neural networks by avoiding the errors and allowing smooth learning. These benefits make ResNet a good choice for accurate and consistent parking spot detection. MobileNet shows good performance in the parking spot detection task with an accuracy of 98.76%. The precision of the model is 98.81%, recall is 98.72%, and F1 score is 98.77%. Its lightweight architecture with efficient depthwise separable convolutions makes MobileNet very useful for computationally constrained scenarios without sacrificing performance. VGGNet yields robust results with a score of 98.55%. Precision and recall are closely the same as they have scores of 98.63% and 98.51% respectively. The F1 score becomes 98.57%. The structure of the network is so simple and includes the application of very small and homogeneous 3x3 filters that increases the performance of parking spot detection without a compromise over efficiency. EfficientNet shows an excellent balance between effectiveness and resource usage. It achieves 98.89% accuracy with precision at 98.78% and recall at 98.92%, which in turn gives an F1 score of 98.85%. The use of its compound scaling method increases the depth, width, and resolution of the network, making it both robust and computationally efficient for the task of parking spot identification.

### 5.2.1 CASE 2. Automatic Number Plate Detection

The confusion matrix of automatic number plate detection with YOLOv8 shows the details of the model's performance in license plate detection. It indicates that this model correctly classified 1807 license plates, making this class highly accurate. There were 78 false positives: the model identified the background as license plates. Overprediction is some evidence here, and there are 33 false negative cases where the model classified actual license plates as the background. These errors suggest that the license plates and the background images are not well separated to significantly enhance overall detection accuracy.

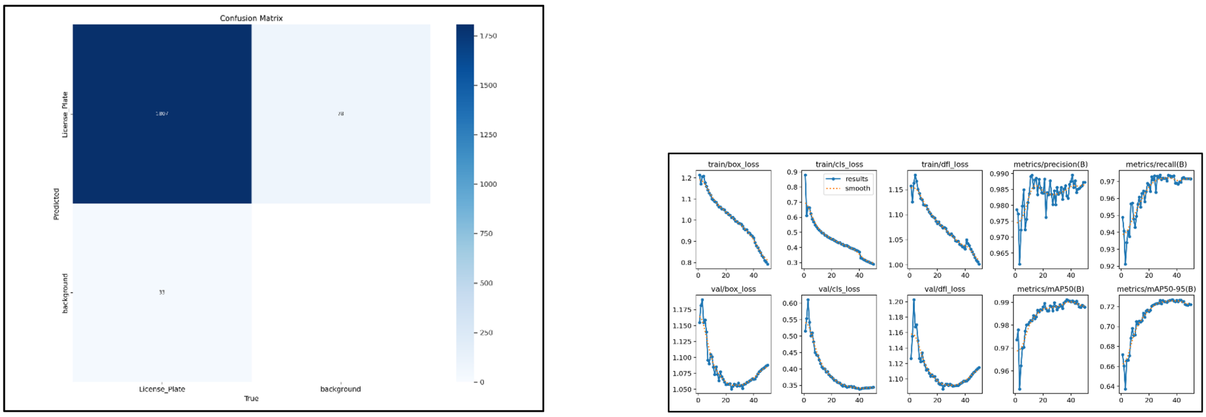



Figure 13: Confusion Matrix and Model graphs for YOLOv8 while training

The following are the graphs of training and validation performance for a YOLOv8 model working on automatic number plate detection. The training losses (`box_loss`, `cls_loss`, and `dfl_loss`) generally decrease, meaning that the model improves the number plates' detection and classification accuracy with bounding box refining. Their validation losses are also generally exhibiting a downward trend - this means the model is generalizing well without significant overfitting. The curve for precision and recall also improves as time progresses, which implies that the model is capable of detecting plates more accurately and consistently. Other important metrics like mAP50 - mean Average Precision at 50% IoU and mAP50-95-average precision across a range of IoU thresholds- increase steadily as well, indicating strength in localization and detection tasks. Overall, the model is learning well: from the figures, both the training and validation metrics are improving in a positively correlated manner.

### 5.2.2 CASE 3. AWS Cloud Deployment using Streamlit

As I have already discussed earlier in this report related to Streamlit where using the Streamlit on AWS cloud to deploy the best model which provide the accuracy and results, as I have deployed the YOLO model where Streamlit involved creating a web application for the tasks where that web application involves in a object detection in a real time. As Streamlit acting as a user interface on the AWS cloud where the Yolo model has been deployed and its dependencies have in installed which I have already explained above, Steamlit application is deployed on to the EC2 instance to allow the user to upload the photos or the videos for detection of Parking Space availability and the number plate recognition for the parking management team, also I have used the Elastic IP for my EC2 instance have a Static IP address.

Here is a hyperlink to .

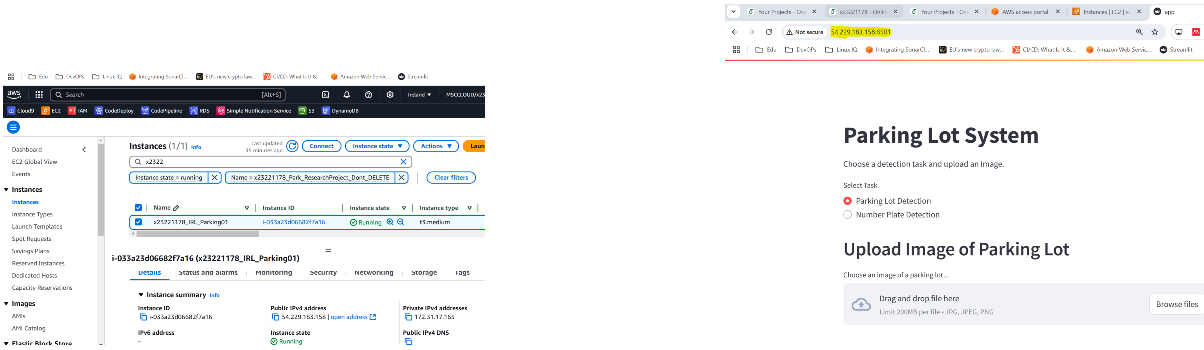



Figure 14: EC2 Instance and Streamlit Real Time Detection Deployment

### 5.2.3 CASE 4. Under a Tough Weather Condition like Foggy Weather

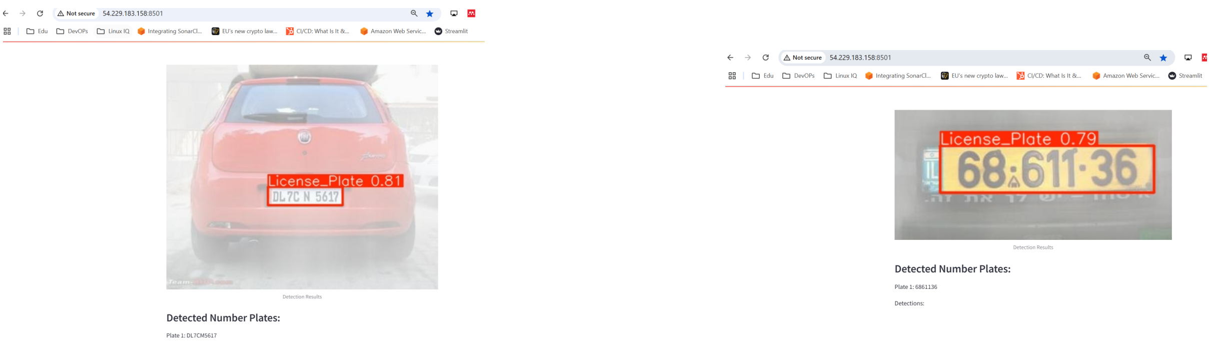



Figure 15: Real Time Detection using AWS Cloud under Foggy Weather

Figure 16: Results of the Vehicle Detection

#### 5.2.4 Configuring the Elastic Load Balancer and the Auto Scaling Group

As we all know the importance of High Availability and the Scalable. if the applications is deployed on the server or on the real time environment users or the customers expect the application to be Highly available for accessing the application, when the load or the traffic increases to the server so it should scale itself and it should handle the varying traffic. So Elastic Balancer would always distributes the load to different AZ's so it can be highly available during the Disaster. Also this would reduce the risk of overloading on a single instance if there is a high network traffic. The Autoscaling group would always dynamically alters the number of ec2 instances base on the incoming traffic, when the traffic decreases the number of instances would be reduced so it would also ensures the cost efficiency. So for testing the configured environment I have performed the stress test on the ec2 instance to increase traffic by installing stress package and I have shown in the below figures, this test showed that both load balancer and the auto scaling mechanism can handle high traffic without any services interruption and also I confirm that the setup would be more high available and cost-efficient.

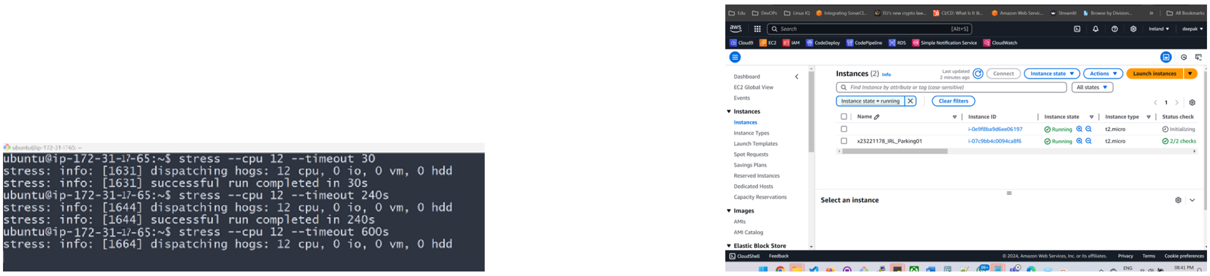



Figure 17: Stress Test on the EC2 instance, resulting the Scaling

# 6 Conclusion and Future Work

The Intelligent Parking Solution fits the list of concerns that are current in the urban society today including traffic congestion, pollution, and bad management of the parking. The system shows how to enhance the parking operations by combining the up-to-date technologies, including IoT, ANPR, computer vision, and real-time analytics. This disturbs user experience prominently via shortening search time, automatically assigning slots and introducing non-intrusive digital means in which to collect fee. Also, the centralized DBMS and corresponding analytics reveal the essential information for parking operators and their further actions. Environmental impacts of less fuel use and fewer emissions bear additional evidence on the role of the system in supporting sustainable development of cities. This solution is consistent with strategies for the development of smart cities and the improvement of urban transport systems underpinning smart cities.

## 6.1 Future Work

While the current system demonstrates strong potential, several enhancements can be explored to make it more robust and scalable:

- Integration of Predictive Analytics: Subsequent to that, it is relatively possible to integrate machine learning models to predict parking usage and demand, weather conditions and event calendar to make better allocation of resources.

- Dynamic Pricing Models: The adaptive tariffs regarding occupancy and time of use may bring maximal income and be justified for each user.
- Interconnectivity with Public Transport: The integration of the system overlaid with public transport applications and framework can help foster transit mobility and shift people from parking their cars for long distance haulage.
- Energy Efficiency and Sustainability: Reducing the energy used by parking facilities can be enhanced by using renewable energy sources to power sensors and cameras.
- Enhanced Security Features: The use of surveillance and allied technologies that the AI utilizes could increase the protection of users' identity and their financial data.
- Global Deployment and Localization: Extending the system to different areas of the world will be possible by adapting it to local languages, rules, and attitudes toward the usage of such environment.
  This Intelligent Parking Solution is a manifestation of changing the traditional method of parking management and establishing itself as the foundation of driving intelligent and sustainable cities across the globe.

# References


Aditya, Amara et al. (2023). "An IoT assisted intelligent parking system (IPS) for smart cities". In: *Procedia Computer Science* 218, pp. 1045–1054.

Alam, Md Rokibul et al. (2023). "A Survey on IoT Driven Smart Parking Management System: Approaches, Limitations and Future Research Agenda". In: *IEEE Access* 11, pp. 119523–119543.

Ashalatha, R and Jayashree Agarkhed (2015). "Evaluation of auto scaling and load balancing features in cloud". In: *International Journal of Computer Applications* 117.6.

Buian, Mohammad Fokhrul Islam, Iqtiar Md Siddique and Anamika Ahmed Siddique (2024). "Efficient Parking Management through QR Technology". In: *Journal of Scientific and Engineering Research* 11.2, pp. 1–9.

Channamallu, Sai Sneha et al. (2023). "A review of smart parking systems". In: *Transportation Research Procedia* 73, pp. 289–296.

Imran, Amna, Meghana Shashishekhara Hulikal and Hamza AA Gardi (2024). "Real Time American Sign Language Detection Using Yolo-v9". In: *arXiv preprint arXiv:2407.17950*.

Katsamenis, Iason et al. (Sept. 2022). "TraCon: A Novel Dataset for Real-Time Traffic Cones Detection Using Deep Learning". In: pp. 382–391. DOI: 10.1007/978-3-031-17601-2_37.

Magdum, Ms Shubhangi S et al. (2017). "CLOUD BASED PARKING SYSTEM USING RFID". In.

Mičko, Kristián and Peter Papcun (2023). "Parking Management System Based on Key Points Detection". In: *Acta Electrotechnica et Informatica* 23.3, pp. 33–39.

Mnyakin, Maxim (2023). "Applications of ai, iot, and cloud computing in smart transportation: A review". In: *Artificial Intelligence in Society* 3.1, pp. 9–27.

Rafique, Sarmad et al. (2023). "Optimized real-time parking management framework using deep learning". In: *Expert Systems with Applications* 220, p. 119686.

Rajyalakshmi, Vankadhara and Kuruva Lakshmanna (2024). "Detection of car parking space by using Hybrid Deep DenseNet Optimization algorithm". In: *International Journal of Network Management* 34.1, e2228.

Sohan, Mupparaju et al. (2024). "A review on yolov8 and its advancements". In: *International Conference on Data Intelligence and Cognitive Informatics*. Springer, pp. 529–545.

Sudhakar, M Venkata et al. (2023). "Development of smart parking management system". In: *Materials Today: Proceedings* 80, pp. 2794–2798.

Türker, Gül Fatma (2023). "Internet of things and cloud based smart parking system design criteria". In: *Journal of Innovative Transportation* 4.2, pp. 33–41.

Xie, Minghui et al. (2023). "A shared parking optimization framework based on dynamic resource allocation and path planning". In: *Physica A: Statistical Mechanics and Its Applications* 616, p. 128649.

Xing, Yaxin and Yanbin Long (2023). "Parking Management System Based on Image Processing Technology". In: *Journal of Computing and Electronic Information Management* 10.2, pp. 29–32.

Yang, Jie, Jinbao He and Xiongwei Wang (2023). "Design of Intelligent Parking System Based on Internet of Things and Cloud Platform". In: *International Journal of Grid and High Performance Computing* 15.2, pp. 1–18.

Yu, Yuan-Chih (2023). "Smart parking system based on edge-cloud-dew computing architecture". In: *Electronics* 12.13, p. 2801.