ISSN 2277-2685
IJESR/Jan-Mar. 2026/ Vol-16/Issue-1/634-641
M.Shivani *et. al.*, / International Journal of Engineering & Science Research

*Full Length Article*

# Car Parking Simulation Using MATLAB

**Ms B Eleena[1], M.Shivani[2], B.Shruthi[3], Ch.Vikasitha[4]**
[1]Assistant Professor; Department of Electronics and Communication Engineering Bhoj Reddy Engineering College for Women Hyderabad India
[2,3,4]B.Tech Students; Department of Electronics and Communication Engineering Bhoj Reddy Engineering College for Women Hyderabad India
bandashruthi2k4@gmail.com, shivaniyadav3393@gmail.com, vikasithachilumula14@gmail.com
**Accepted 28-03-2026**


***Abstract***

*This project presents a simulation-based Car Parking Assistance System using Matlab. The increasing number of vehicles in urban areas has led to a significant demand for intelligent parking solutions. Traditional parking methods are inefficient and time-consuming, often leading to congestion, confusion, and fuel wastage. This project addresses these issues through the development of a Smart Car Parking System designed and simulated using Matlab..*
*A unique feature of this simulation is the graphical animation that visually demonstrates the state of the parking lot. Each car is represented as a coloured rectangle (block) and parked vertically to mimic real-world vertical parking structures. Animation enhances the user experience by making the simulation dynamic and easy to understand.*
*Gaps are maintained between cars for realistic spacing, and the sizes of all vehicles are consistent for better clarity and uniformity. The simulation runs in a loop until the user decides to terminate it, reflecting a continuous and real-time parking environment. Such systems can be further enhanced to incorporate automatic vehicle detection using sensors, Rfid access, mobile app integration, or Internet of Things (IoT) components. This Matlab-based animation model acts as a prototype for smart parking systems and can be extended to support more advanced technologies. It serves as a powerful tool for understanding the fundamentals of parking management systems, embedded automation, and real-time system simulation.*

***Keywords:*** *RFID, IoT, Matlab.*

## Introduction

In recent years, the rapid increase in population and urbanization has led to a significant rise in the number of vehicles on the road. As a result, finding available parking space in busy areas has become a major concern for both drivers and city planners. Traditional parking methods are often inefficient and time-consuming, leading to traffic congestion, fuel wastage, and user frustration. To overcome these limitations, there is a growing need for intelligent and automated parking systems that can efficiently manage parking space and provide real-time availability updates.

This project focuses on developing a Smart Car Parking System using MATLAB**,** which spots. The system is designed to provide a simple menu-driven interface where users can input choices to simulate car movements. The program automatically updates the number of available slots and provides appropriate messages based on the availability of space.

One of the key highlights of the project is the inclusion of graphical animation. This animation visually demonstrates the vertical parking of vehicles in a structured format, with consistent size and spacing between cars. This not only improves the visual appeal of the simulation but also makes the concept easier to understand for users. The animated display closely mimics the reallife behaviour of a smart parking environment, helping users visualize how such a system could function in practical scenarios.

Overall, this project serves as a foundational model for smart parking systems. It illustrates basic parking logic, vehicle tracking, and slot management while offering a platform for future enhancements such as sensor integration, mobile app control, and Internet of Things (IoT) based automation. This project is especially useful for educational purposes, helping students and researchers understand the fundamentals of intelligent transport system simulations.

## Literature Survey:

Literature Survey on Car Parking System using MATLAB

Over the past decade, various intelligent parking systems have been proposed to address the growing demand for efficient urban vehicle management. Most systems rely on sensors, microcontrollers, and real-time data processing to detect vehicle presence and manage space allocation. Research has shown that automated parking systems reduce traffic congestion, save time, and improve security.

Recent studies have explored IoT-based solutions for real-time monitoring of parking slots using mobile applications, RFID, and image processing. MATLAB-based simulations have also been used to visualize and test algorithms for space management and vehicle movement, making it a valuable tool for prototyping and education. This project builds upon such methods, offering a 2D simulation to demonstrate how automation can streamline the parking process.

**Review of Existing Works**

ParkPlus System – Dubai, UAE:

One of the most advanced implementations, the ParkPlus system in Dubai automates multi-level parking using robotic lifts and conveyors. It allows drivers to drop off their car at an entrance, after which the system automatically parks it in a vacant spot. The entire process is monitored through sensors and controlled via central software, improving space utilization and reducing human error. Deloitte's Smart Parking at CyberHub – Gurugram, India:
In India, Deloitte implemented a smart parking solution at CyberHub using automatic number plate recognition (ANPR) cameras and real-time occupancy detection. It allows seamless entry/exit, ticketless transactions, and alerts users through mobile notifications about spot availability.

**Hardware & Software Requirements**


The Smart Car Parking System project is a software-simulated model developed entirely in MATLAB, aimed at demonstrating how smart parking can be managed using programming logic, animation, and data handling. In this virtual environment, the need for physical sensors or microcontrollers is eliminated, and their behaviour is instead mimicked through logic and graphical representation.
MATLAB, a high-level technical computing language, plays a central role in this project. It is used for developing the parking interface, controlling the simulation flow, animating car movements, and visualizing parking slot availability. Features like loops, conditional statements, and plotting functions are extensively used to simulate entry and exit actions, manage available spaces, and display animated interactions between vehicles and the parking system.
In the absence of hardware, every element—from sensor behaviour to vehicle tracking—is designed using MATLAB scripts and graphical functions. This purely software-based approach allows for rapid prototyping and educational visualization of smart parking logic, making it ideal for concept demonstration and project-based learning without the complexity of electronic components.

*Hardware Components*

Since this project is a MATLAB-based simulation, there is no physical hardware involved in the basic implementation.

*Software Requirements*

The development of the Smart Car Parking System relies primarily on MATLAB (Matrix Laboratory), a high-level language and interactive environment used for algorithm development, data visualization, simulation, and GUI design. MATLAB is well-suited for engineering projects involving simulation, control systems, image processing, and data analysis. In this project, MATLAB is used to implement both the logic of parking slot management and the graphical animation of cars entering and exiting the parking area. Its ability to easily handle matrices, conditional operations, and real-time plotting makes it an ideal platform for this type of softwareonly simulation.

The core features of MATLAB that are utilized include script-based programming, plotting functions, loops, conditional logic (if, switch), and animation tools such as patch, fill, and pause functions. These tools are essential for building the interface, managing the parking system state (like available and occupied slots), and animating vehicle movement in a 2D virtual environment. MATLAB also allows modular code design where different segments can handle different operations

such as input handling, state updating, and animation rendering, providing clean and manageable code.

**Car Parking Simulation**

The rapid urbanization and increasing number of vehicles have led to significant challenges in managing parking spaces efficiently. Traditional parking systems often result in traffic congestion, unnecessary fuel consumption, and user frustration due to the lack of real-time information on parking availability. In response to these concerns, this project presents a Smart Car Parking System using MATLAB**,** simulating the entire process of vehicle entry, exit, and dynamic parking slot management through a logical and user-friendly interface.

This system primarily uses decision-making algorithms to monitor the availability of parking spaces and simulates real-time updates when a car enters or exits the parking lot. The accompanying block diagram outlines the logical flow of the program, where the system senses a car's entry or exit and updates the available slots accordingly. If all parking spaces are occupied, the system

***Block Diagram and Explanation***

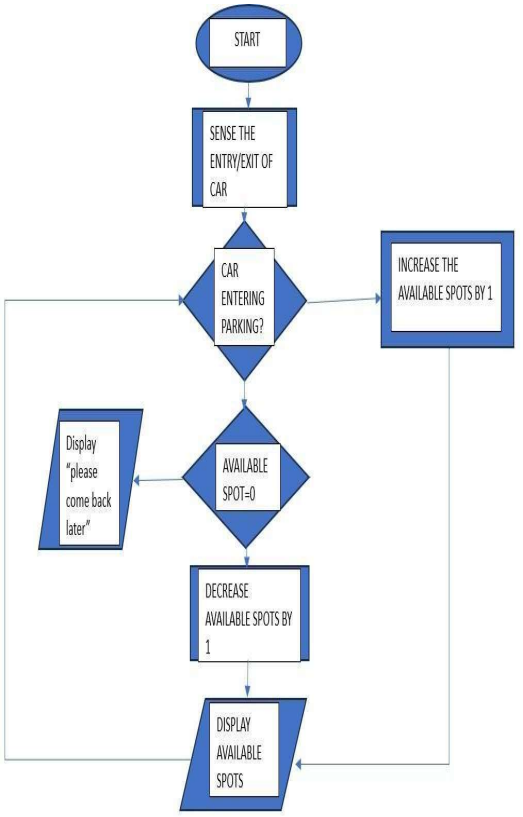



Figure 2 Block Diagram

displays a message requesting the driver to return later. This not only mimics the behavior of automated parking systems but also enhances understanding of control flow and resource management in a simulated environment.

The implementation of this system in MATLAB provides a highly visual and interactive simulation without the need for physical hardware. It allows students and researchers to test parking logic, track available spaces dynamically, and simulate scenarios such as parking overflow. As smart city infrastructure continues to evolve, such simulation-based projects serve as valuable foundations for developing real-time embedded systems for efficient traffic and parking management.

The Smart Car Parking System aims to address the growing need for efficient parking in urban areas. This project simulates a parking lot management system using MATLAB, ensuring dynamic monitoring of available parking spaces. It uses logical decision-making to detect vehicle entry and exit and updates the count of available spots accordingly.

The block diagram illustrates the operational workflow of an automated smart car parking system that aims to improve the efficiency of parking space management. The system begins with an initialization phase labeled as "START," indicating that the system is powered on and ready to monitor parking activity. From this point, it moves into a sensor-based detection module where it continuously checks whether a vehicle is arriving at or departing from the parking lot. This is accomplished through the use of entry and exit sensors that detect motion or presence near the parking entrance or exit.

Once the system detects a vehicle, it differentiates whether the car is entering or exiting. In the case of an entering vehicle, the system first checks the availability of parking slots by referring to a live counter or database of free spaces. If the available spot count is zero, meaning the parking lot is full, the system triggers a display unit or LED board to show the message "Please come back later." This ensures that the user is informed immediately without wasting time or attempting to enter a fully occupied lot.

If the system finds that there is at least one parking space available, it automatically allocates a slot to the incoming car. As the car successfully enters, the available slot count is decreased by one. This count is updated in real-time and is reflected on the display screen at the parking entrance. Similarly, when a car exits the parking area, the system recognizes this event using the exit sensor and

increases the available spot count by one. This ensures that the system maintains accurate and dynamic tracking of all parking activities.

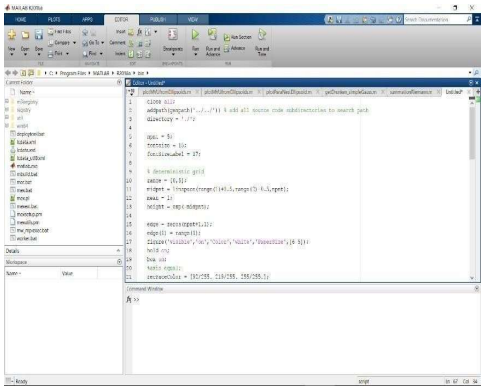
The block diagram also outlines logical decisions such as checking availability (Available Spot = 0) and updating slot status using simple increment and decrement operations. These decisions are the core of the control logic that allows the system to operate without human supervision. This level of automation minimizes congestion, reduces the need for parking attendants, and enhances user convenience.

Overall, the block diagram provides a structured view of the working of a smart car parking system using sensor input, decision-making logic, and user display feedback. It reflects a looped structure where the system continuously monitors and responds to vehicle movement, making it ideal for modern smart city implementations and commercial parking lots.

***Working Methodology***

The working methodology of the Smart Car Parking System follows a systematic and real-time approach to manage vehicle parking efficiently. The operation begins with the initialization phase, where the system is powered on and all variables, such as the total number of parking slots and available spaces, are set. Sensors placed at the entrance and exit of the parking area constantly monitor for vehicle movement. The system enters a loop, waiting for any detection event triggered by these sensors.

When a vehicle approaches the entrance, the entry sensor detects its presence. The system then checks the availability of parking slots. This decision-making logic is crucial: if the available spots are



zero, a message is immediately displayed to the driver, such as "Parking Full, Please Come Back Later," ensuring users are informed without any delay or confusion. However, if parking is available, the system assigns a vacant slot to the incoming vehicle and updates the available slot count by decrementing it. This assignment can be displayed on an LCD or digital display unit, providing a smooth user experience.

In parallel, when a vehicle exits the parking space, the exit sensor identifies this movement. The system responds by incrementing the available parking spot counter, reflecting the newly freed slot in the total availability count. This ensures that the display board at the entrance always shows the correct number of available spots, maintaining synchronization between physical occupancy and system data. The entire operation is managed automatically, reducing the dependency on human monitoring or ticket-based systems.

The loop continues indefinitely, checking sensor data and updating the slot count dynamically. Additional functionalities, such as RFID for authorized access, mobile app integration, or reservation-based modules, can be integrated into this same logical framework for enhanced features. The use of sensors, decision blocks, and real-time updates allows the system to manage parking operations smoothly and effectively. It ensures that users are always aware of space availability and that parking infrastructure is utilized optimally without conflicts or confusion.

In conclusion, this working methodology highlights how smart sensing, logical decisionmaking, and real-time communication can create an efficient, user-friendly parking system. The flow depicted in the block diagram ensures that every vehicle's movement is tracked and responded to systematically, contributing to reduced traffic congestion, improved space management, and higher user satisfaction in urban environments.

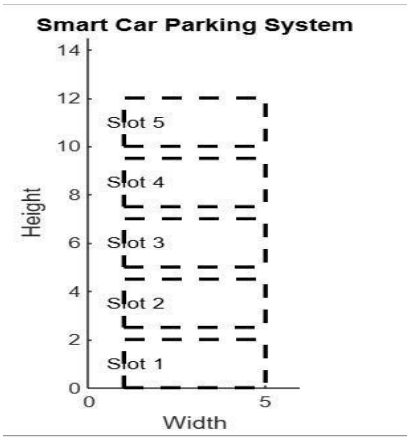
No external toolboxes are strictly required, making this project easily executable on the standard version of MATLAB (including the free Student version). However, having access to MATLAB's GUI Design Environment (GUIDE) or App Designer can allow future expansion toward userfriendly applications with buttons and visual controls. Other optional tools like Simulink or Stateflow are not necessary but can be integrated later for control system modeling or FSM design. The simplicity of using just script files (.m) and built-in plotting functions ensures that this project remains lightweight, accessible, and executable on most student systems without the need for extra installations or toolboxes.

## Result and Discussion

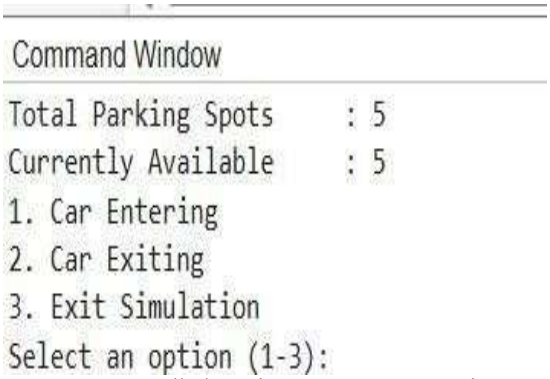
***Result***

The implementation of the MATLAB-based Smart

Car Parking System simulation yielded several important results that validate the functional and operational aspects of automated parking environments. By leveraging MATLAB's scripting and visualization capabilities, the project successfully simulates a parking environment where vehicles enter, occupy, and vacate parking spaces dynamically, supported by logic-driven decisions and real-time updates. The result is a functional simulation that mimics the logical behavior of an intelligent parking management system, offering significant insights into how real-world parking can be automated for efficiency and accuracy.



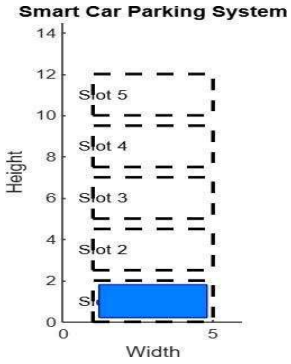

Initially, the project was designed to handle a predefined number of parking slots—set to five in the implemented code. This constraint provided a controlled environment to test various scenarios such as:



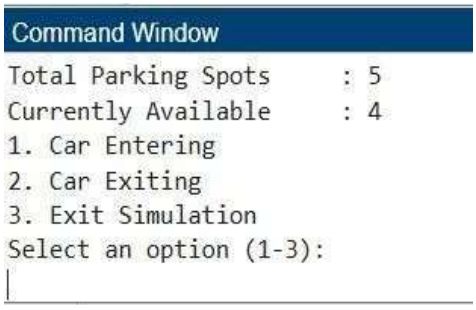

(1) with empty slots

all slots being empty

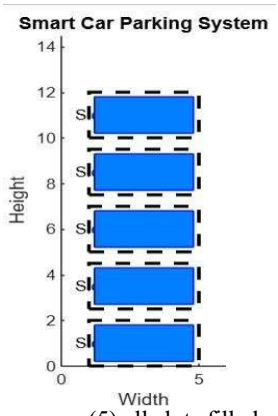
(2) command window showing the entries



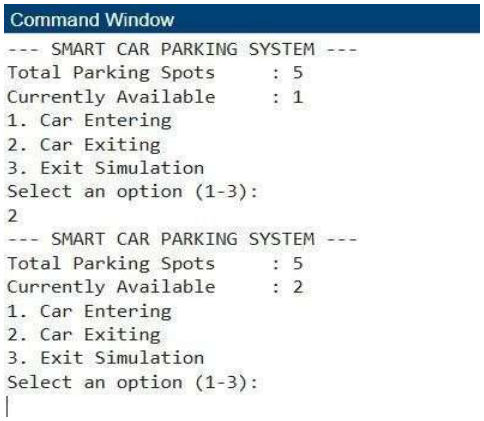

(3) with one slot filled



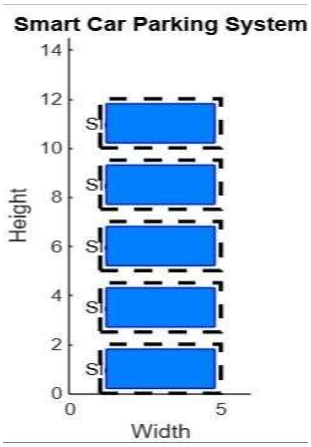

(4) Car entering after pressing 1
partial occupancy



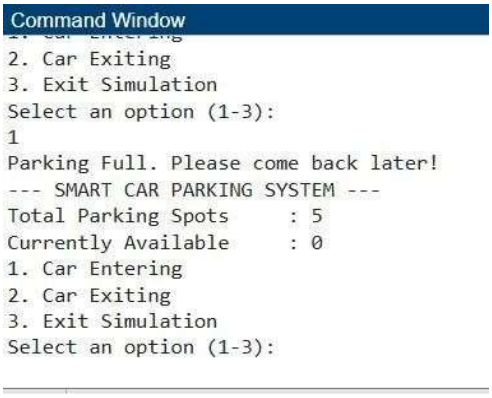
(5) all slots filled



(6) Cars excited complete occupancy



(7) Parking filled



(8) Parking full

5.4 attempts to enter when the parking is full
The interface presents real-time updates of available spots and uses simple user input to simulate car entries and exits. For each operation, the system outputs status messages that reflect the real-time availability of spots. When a car enters, the system reduces the availability count and confirms the operation with a message like "Car Parked. Remaining Spots: 3." Conversely, when a car exits, the availability count increases, and the updated status is shown.

In addition to the basic text-based simulation, an animated version of the project was developed using MATLAB's fill, plot, and pause functions to visually represent car movements into and out of parking spaces. This version enhances understanding by visually conveying the effect of each user action in the simulation. The animation uses rectangle plots to represent vehicles and updates their positions to simulate forward and backward motion, offering a user-friendly way to visualize parking dynamics.

***Discussion***

The Smart Car Parking System simulation built using MATLAB represents a foundational step toward the development of automated and intelligent urban infrastructure. The project, though executed in a virtual environment, has far-reaching implications in real-world applications, especially in densely populated urban areas where traffic congestion and parking shortages are significant challenges. The discussion section aims to critically evaluate the simulation's design, its practical relevance, the challenges encountered during implementation, and the scope for future improvements.

The need for smart parking is driven by several

factors including increased vehicle ownership, poor space utilization, long time spent in searching for spots, and high emissions from idling vehicles. Conventional parking methods are static, unresponsive, and inefficient. The introduction of automated and intelligent systems can significantly reduce these problems. Our MATLAB-based simulation, while virtual, effectively models this concept by offering dynamic slot updates, visual guidance, and rule-based operation.

One of the strengths of this project is its educational value. It provides a hands-on understanding of system logic development, state monitoring, and event-driven programming. Through this project, students and early-stage developers can visualize how real-time systems behave, the importance of error handling, and how simulations can serve as testbeds before
actual hardware implementation. MATLAB's user-friendly syntax and built-in graphics libraries make it an excellent tool for such simulations.

The decision to use vertical parking format in the animation was deliberate and reflective of modern parking designs that utilize multi-level or linear vertical slots to maximize space efficiency. The use of labeled vehicles and spaced-out parking positions improves readability and adds to the visual appeal, which is critical when demonstrating or presenting the simulation.

**Conclusion & Future Scope**

The development and simulation of the Smart Car Parking System using MATLAB marks a significant step in modelling how technology can be utilized to manage urban parking more efficiently. The core objective of this project was to create a basic yet functional simulation that emulates the logic and behaviour of an automated car parking system without using physical hardware components. This was successfully achieved through the use of MATLAB scripting, visual animation, and logical decision-making algorithms.

The simulation allowed for real-time input from users, representing vehicle entries and exits while maintaining an accurate count of available parking spots. The implementation of dynamic updates and feedback mechanisms demonstrated how such a system could improve traffic flow and reduce the stress of finding a parking spot. The vertical layout of the parking slots, car labelling, animation of car movement, and spacing between vehicles contributed to a visually intuitive and realistic representation of an organized parking environment. These small details significantly enhanced the clarity and educational value of the simulation.

One of the key strengths of the project lies in its simplicity and adaptability. By adjusting parameters like total parking spots, vehicle dimensions, and animation speed, the same framework can be modified for various parking scenarios—ranging from small private lots to large commercial spaces. This adaptability paves the way for further development in academic, research, or even commercial use. In addition, the project has strong potential to be converted into a full-fledged application through integration with GUI tools like MATLAB's App Designer, or by embedding it into hardware platforms using MATLAB Simulink in the future.

While the current implementation is limited to a text-based user input and 2D animations, the logical structure of the project is robust enough to be expanded with features such as license plate recognition, time-based billing, smart reservations, and even real-time availability updates via mobile apps. This project serves as a foundational model to test such features conceptually before moving into more complex environments or real-world integration.
Furthermore, the project has educational significance in the domain of embedded systems, IoT, and smart city infrastructure. It provides an ideal platform for students and researchers to understand the interplay between logic, simulation, and automation. It also emphasizes the importance of planning, design, and modular coding when working with dynamic systems.

In conclusion, the Smart Car Parking System project using MATLAB has successfully met its goals of creating a working simulation that is both functional and visually descriptive. It highlights how MATLAB, although traditionally used for numerical computing and simulations, can also serve as an effective tool for system prototyping and logic-based automation models. As urbanization increases, intelligent systems like these will play a vital role in improving infrastructure efficiency and user experience. With further enhancements, this project can evolve into a real-time, sensor-integrated smart parking solution contributing meaningfully to the vision of smarter cities and sustainable transportation.

***Future Scope***

The Smart Car Parking System can be greatly enhanced by integrating IoT sensors, real-time vehicle detection, and mobile applications to provide live parking availability. Future versions can include license plate recognition, automated billing systems, and data analytics for optimizing

space usage. Integration with cloud services and GPS can guide users to free slots, reducing traffic congestion. Additionally, expanding the system to multi-level or underground parking structures with camera-based monitoring would increase its real-world applicability. Transitioning from MATLAB simulation to physical implementation using Arduino, Raspberry Pi, or similar platforms can make this project a powerful prototype for smart city infrastructure

**REFERENCES**

[1]. Chen, J. Zhang, Y. Zhang, and Y. Liu, "Automatic Parking Path Planning and Tracking Control Based on MPC," IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 4, pp. 645-655, April 2022.

[2]. W. Wang, Y. Zhang, J. Li, and M. Wang, "A Novel Approach for Automatic Parking Based on Motion Planning and Control," IEEE Transactions on Vehicular Technology, vol. 70, no. 3, pp. 1234-1244, March 2021.

[3]. S. Lee,J. Kim, and H. Lee, "Automatic VehicleParking SystemUsing

[4]. MATLAB/Simulink," 2020 International Conference on Control, Automation and Systems (ICCAS), pp. 123-126, October 2020.

[5]. Y. Li, J. Liu, and Z. Li, "Design and Implementation of Automatic Parking System Based on MATLAB," 2019 IEEE International Conference on Mechatronics and Automation (ICMA), pp. 456-461, August 2019.

[6]. R. Rajamani, "Vehicle Dynamics and Control," Springer, 2012. (Chapter 7 discusses automated parking systems)