AI ENABLED CAR PARKING USING OPENCV

A PROJECT REPORT

Submitted by

SREERAM R
ANAND KUMAR U
JOSHABEL J
NANTHA GOBAN SP

Team ID : NM2023TMID22560

ARTIFICIAL INTELLIGENCE - GROUP 5

PONJESLY COLLEGE OF ENGINEERING, NAGERCOIL

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1.1 Project Overview:

The AI-enabled car parking system using OpenCV is a comprehensive solution designed to address the challenges associated with car parking management. With the ever-increasing number of vehicles on the roads, finding suitable parking spaces has become a significant concern in urban areas. This project aims to leverage the capabilities of computer vision and artificial intelligence to create an intelligent parking system that efficiently manages and optimizes parking spaces.

The system utilizes OpenCV, an open-source computer vision library, to perform real-time image processing and analysis. By analyzing video feeds or images from parking areas, the system can accurately detect and monitor parking spaces. It employs advanced algorithms to identify vacant and occupied parking spaces, track vehicles, and provide valuable insights into parking space utilization.

1.2 Purpose:

The purpose of this project is to revolutionize the car parking experience by developing an AI-enabled system that offers several key benefits:

Enhancing Parking Space Utilization: The AI-enabled car parking system aims to maximize parking space utilization by efficiently managing the allocation of parking spots. By accurately detecting vacant spaces, the system can guide drivers to available spots, minimizing the time spent searching for parking and reducing congestion.

Real-Time Parking Availability Information: The system provides real-time information about parking space availability. Through sensors or video analysis, it continuously monitors parking areas and updates the status of parking spaces, enabling drivers to make informed decisions about where to park.

Improved Parking Experience: By guiding drivers to available parking spaces, the system enhances the overall parking experience. It reduces frustration and stress associated with finding parking, resulting in a smoother and more convenient parking process.

Optimal Resource Allocation: The AI-enabled car parking system enables parking operators to optimize the allocation of parking spaces. By analyzing parking patterns and occupancy trends, operators can make data-driven decisions to improve efficiency, allocate resources effectively, and ensure optimal utilization of parking areas.

Sustainability and Reduced Emissions: By minimizing the time spent searching for parking spaces, the system helps reduce traffic congestion and associated carbon emissions. It promotes a more sustainable approach to urban mobility by streamlining the parking process and encouraging efficient use of parking resources.

CHAPTER-2
IDEATION & PROPOSED SOLUTION

2.1 Problem Statement Definition:

In urban areas with limited parking spaces, it is challenging for drivers to find available parking spots efficiently. This leads to wasted time, increased traffic congestion, and frustration for both drivers and parking operators. There is a need for a smart parking system that can accurately detect and monitor parking spaces, provide real-time parking availability information, and optimize parking space utilization.

Problem Statements	**Definitions**
Inefficient Parking Space Management	The current parking system lacks an efficient method to manage and optimize parking space utilization, resulting in wasted space and increased congestion.
Manual Monitoring and Reporting	The manual process of monitoring and reporting parking occupancy is time-consuming, prone to errors, and lacks real-time information for users.
Difficulty in Finding Available Parking Spots	Users face challenges in finding available parking spots, leading to frustration, increased search time, and traffic congestion in the parking area.
Lack of Security and Unauthorized Access	The absence of a robust security system allows unauthorized vehicles to enter the parking area, posing a risk to the safety and security of parked vehicles.
Limited Accessibility for Disabled Users	The current parking system lacks provisions to ensure easy accessibility

	for disabled users, such as designated accessible parking spots or assistance.
Inefficient Allocation of Reserved Parking Spaces	Reserved parking spots are not optimally allocated, leading to underutilization or unauthorized use of these spaces.
Absence of Real-Time Parking Information	Users are unable to access real-time parking information, such as available spots or parking space status, leading to inconvenience and inefficient usage.

2.2 Empathy Map Canvas:

To gain a deeper understanding of the users' needs, motivations, and pain points, an empathy map canvas can be created. The empathy map helps in developing empathy towards the end users and uncovering valuable insights. The empathy map canvas typically includes sections such as "Says," "Thinks," "Does," "Feels," "Pains," and "Gains."

User: Car drivers who need to find and utilize parking spaces efficiently

What They Say:

- "It's frustrating when I can't find an available parking spot."
- "I wish there was a way to know the real-time availability of parking spaces."
- "Finding a suitable parking spot for disabled users can be a challenge."
- "I worry about the security of my vehicle in the parking area."
- "The existing signage is confusing and outdated."
- "It would be helpful to have reserved parking spaces allocated more efficiently."

What They Think:

- "I don't want to waste time searching for a parking spot."
- "I need accurate and up-to-date information to make parking decisions."
- "Accessible parking spots should be easily identifiable and available."
- "I want a secure parking area to avoid theft or damage to my vehicle."
- "Clear and helpful signage would make parking easier and less stressful."
- "Reserved parking spaces should be properly utilized to avoid misuse."

What They Feel:

- Frustration when unable to find an available parking spot.
- Annoyance when encountering confusing or outdated parking guidance.
- Concern for the security and safety of their vehicle.
- Empathy towards disabled users who face accessibility challenges.
- Satisfaction when finding a parking spot quickly and easily.

What They Do:

- Circle around parking areas searching for vacant spots.
- Rely on guesswork or follow other cars to find available parking.
- Report incidents of unauthorized parking or security concerns.
- Look for visible signage indicating accessible parking spaces.
- Seek assistance from parking attendants or security personnel.
- Express their parking experiences and frustrations on social media.

Pain Points:

- Difficulty in finding available parking spots in a timely manner.
- Lack of real-time information on parking space availability.
- Inadequate accessibility for disabled users.
- Concerns about vehicle security and safety.
- Confusion caused by inaccurate or outdated parking guidance.

Gains:

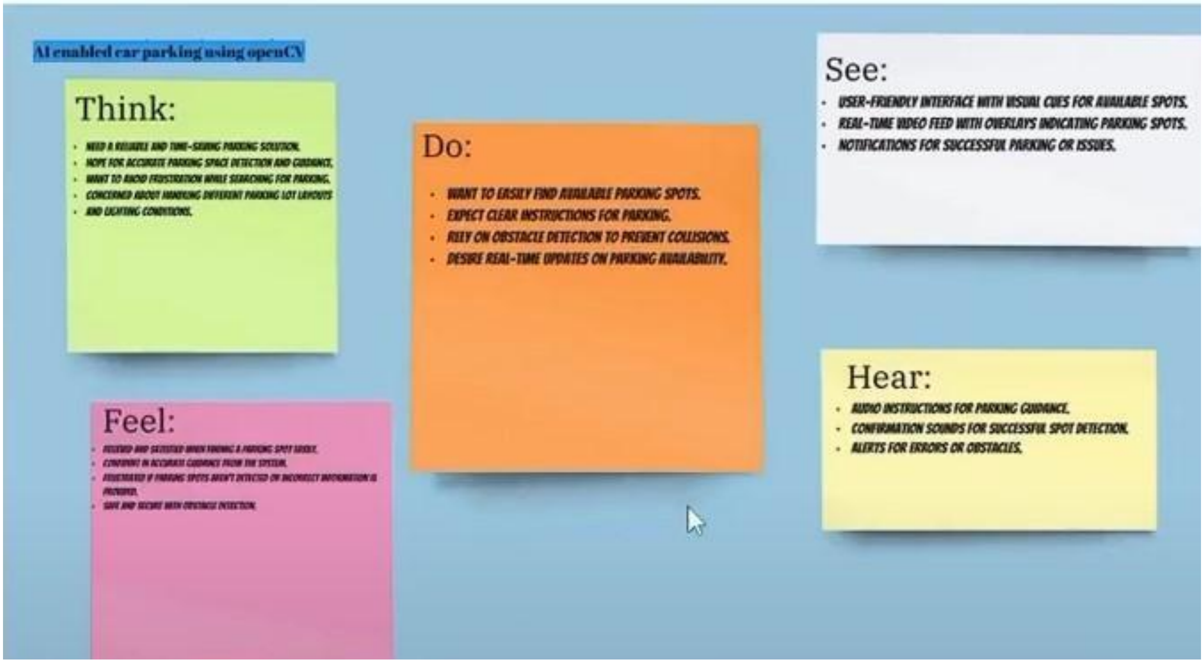
- Quick and easy identification of available parking spots.

- Real-time information on parking space availability.

- Improved accessibility for disabled users.

- Enhanced security measures for parked vehicles.

- Clear and up-to-date parking guidance to reduce confusion and save time.



2.3 Ideation & Brainstorming:

Ideation and brainstorming sessions can be conducted to generate a wide range of ideas and potential solutions for the AI-enabled car parking system. This can involve gathering a diverse group of stakeholders, such as drivers, parking operators, and technology experts, to contribute their perspectives and insights. Various techniques like mind mapping, SWOT analysis, and idea generation exercises can be employed to encourage creativity and collaboration.

Real-Time Occupancy Detection:

Implement a real-time system that can detect the occupancy status of parking spots using computer vision techniques in OpenCV. This is the core functionality of your project and should be prioritized.

Parking Spot Localization:

Enhance the system to not only detect occupied or vacant spots but also accurately locate the boundaries of individual parking spots. This can provide more precise information to users and improve the overall user experience.

Integration with Sensors:

Integrate the car parking system with physical sensors such as ultrasonic or infrared sensors to provide additional data and improve the accuracy of occupancy detection. This can be a valuable extension to the computer vision-based approach.

Parking Guidance and Navigation:

Develop a feature that guides drivers to available parking spots using visual cues or navigation instructions. This can help optimize parking space utilization and reduce the time spent searching for a parking spot.

Vehicle Recognition and Authentication:

Implement a mechanism to recognize and authenticate vehicles entering the parking area, such as using license plate recognition or RFID technology. This can provide enhanced security and access control.

Data Logging and Analytics:

Create a system to log and analyze parking data over time, including occupancy patterns, peak usage hours, and trends. This information can be used for capacity planning, optimizing parking operations, and generating insights for future improvements.

Mobile Application or Web Interface:

Develop a user-friendly mobile application or web interface that allows users to view parking availability, reserve spots, and receive notifications about parking status. This can enhance user convenience and accessibility.

2.4 Proposed Solution:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	The problem to be solved is the inefficient management of parking spaces, resulting in wasted space, congestion, and driver frustration. Current systems lack an effective method to optimize parking space utilization, leading to extended search times for drivers. Manual monitoring and reporting methods provide limited and outdated information on parking availability. Additionally, the absence of a robust security system exposes parked vehicles to unauthorized access, theft, and damage. To address these issues, an AI-enabled car parking system is needed. It should efficiently manage parking space utilization, provide real-time information on availability, enhance security measures, and improve the overall parking experience. By implementing such a system, drivers can easily find parking, reduce search times, and ensure vehicle safety, contributing to a more organized and sustainable parking environment.

2.	Idea / Solution description	The proposed solution is an AI-enabled car parking system that utilizes computer vision and OpenCV for real-time occupancy detection and accurate localization of parking spaces. It integrates with physical sensors, provides a user-friendly interface for real-time information, incorporates vehicle recognition and authentication mechanisms, and includes data logging and analytics capabilities. This solution aims to enhance the parking experience by reducing search times, improving accessibility and security, optimizing space utilization, and providing real-time information to drivers.
3.	Novelty / Uniqueness	The novelty and uniqueness of the proposed AI-enabled car parking system lie in its integration of computer vision techniques, OpenCV, and additional sensor integration for accurate real-time occupancy detection and precise localization of parking spaces. The combination of these technologies provides an innovative solution that surpasses traditional parking management systems. The system's user-friendly interface, vehicle recognition and authentication features, and data logging and analytics capabilities further differentiate it from existing solutions. By leveraging these unique aspects, the proposed system offers an enhanced parking experience, improved efficiency, and increased security, setting it apart from conventional parking systems.
4.	Social Impact / Customer Satisfaction	The proposed AI-enabled car parking system has a significant social impact by optimizing parking space utilization, reducing congestion and

		carbon emissions. It enhances customer satisfaction through improved accessibility for disabled users, increased security measures, and real-time information on parking availability. By providing a seamless and efficient parking experience, the system contributes to a sustainable environment and reduces frustration for drivers.
5.	Business Model (Revenue Model)	The business model for the AI-enabled car parking system includes generating revenue through parking fees, subscription plans, targeted advertisements, data licensing, partnerships with parking facility operators, and custom development and integration services. By diversifying revenue streams and adapting to market conditions, the business model aims to ensure financial sustainability while providing value-added services to customers.
6.	Scalability of the Solution	The AI-enabled car parking system is designed to be highly scalable, adaptable to different parking environments and capable of handling increased demand. Its flexible architecture, based on computer vision and OpenCV, allows for easy replication and deployment in various locations. The integration of sensors and a scalable infrastructure ensures the system's ability to accommodate future growth and expansion.

CHAPTER-3
REQUIREMENT ANALYSIS

3.1 Functional Requirements:

Functional requirements define the specific functionalities and capabilities that the AI-enabled car parking system should possess. These requirements outline the core features and operations of the system. The functional requirements for the system can include:

Parking Space Detection: The system should be able to detect and identify individual parking spaces within the parking area using computer vision algorithms. It should accurately differentiate between occupied and vacant spaces.

Real-time Monitoring: The system should continuously monitor the parking spaces and provide real-time updates on their occupancy status. It should be capable of handling multiple video feeds or sensor inputs simultaneously.

Parking Space Reservation: Optionally, the system can provide the functionality for drivers to reserve parking spaces in advance through a mobile application or web portal. This can involve integration with a booking and reservation system.

Parking Guidance: The system should guide drivers to available parking spaces using visual cues or navigation instructions. It should provide the most efficient route to the nearest vacant parking spot.

Data Analytics and Reporting: The system should analyze parking data, generate reports, and provide insights into parking space utilization, occupancy rates, peak hours, and other relevant metrics. It should support customizable reporting features.

Integration with Payment Systems: Optionally, the system can integrate with payment systems to enable seamless payment processes for parking fees. This can involve integration with mobile payment platforms or existing parking payment infrastructure.

User Management: The system should have user management capabilities to allow drivers and parking operators to register, login, and access their respective functionalities. It should support role-based access control.

3.2 Non-Functional Requirements:

Non-functional requirements specify the quality attributes and constraints of the AI-enabled car parking system. These requirements focus on aspects such as performance, reliability, security, usability, and scalability. The non-functional requirements for the system can include:

Performance: The system should be capable of processing video feeds or sensor data in real-time, providing prompt updates on parking space availability, and delivering a responsive user experience.

Reliability: The system should be reliable and robust, with minimal downtime or disruptions. It should handle unexpected scenarios, such as network failures or system crashes, gracefully and recover gracefully.

Security: The system should ensure the security and privacy of user data. It should incorporate appropriate encryption measures, access controls, and data protection mechanisms to prevent unauthorized access or data breaches.

Usability: The system should have an intuitive and user-friendly interface for both drivers and parking operators. It should be easy to navigate, with clear instructions and visual cues for parking guidance.

Scalability: The system should be designed to handle varying parking areas of different sizes and scales. It should be scalable to accommodate increasing numbers of parking spaces and support future expansion plans.

Integration: The system should be easily integrable with existing parking infrastructure, payment systems, and other relevant systems. It should support standard APIs and protocols for seamless integration.

Maintainability: The system should be maintainable, allowing for updates, bug fixes, and system enhancements without significant disruptions. It should follow modular and well-documented coding practices.

Performance: The system should be efficient in terms of computational resources, minimizing the processing and memory requirements to ensure optimal performance.

CHAPTER-4

PROJECT dESIGN

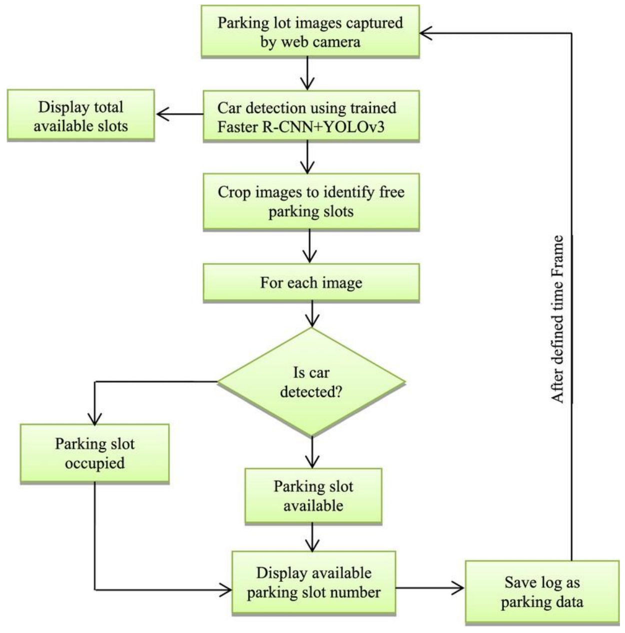
4.1 Data Flow Diagrams:

Data flow diagrams (DFDs) provide a visual representation of the flow of data within the AI-enabled car parking system. These diagrams illustrate how information moves between various components and entities involved in the system. The DFDs for the project can include:

Level 0 DFD: This high-level diagram presents an overview of the system and shows the main external entities interacting with the system, such as drivers, parking operators, and external systems like payment gateways or reservation systems.

Level 1 DFD: This diagram provides a more detailed view of the system and breaks down the main processes and data flows identified in the Level 0 DFD. It showcases the major components and their interactions, including parking space detection, monitoring, reservation, and reporting.

Level 2 DFDs: These diagrams further decompose the processes and data flows identified in the Level 1 DFD. They provide a more granular view of the system, illustrating the specific actions and data exchanges within each process or component.



4.2 Solution & Technical Architecture:

The solution and technical architecture outline the overall structure and components of the AI-enabled car parking system. It defines the technologies, frameworks, and infrastructure required to implement the proposed solution. The architecture can include:

Component Diagram: This diagram illustrates the different components or modules of the system and their relationships. It showcases how the various functional units, such as parking space detection, monitoring, reservation, and reporting, interact with each other.

Technology Stack: The technology stack outlines the programming languages, frameworks, libraries, and tools that will be used to develop the system. It can include technologies like Python for computer vision algorithms, OpenCV for image processing, Flask for web application development, and databases like MySQL or MongoDB for data storage.

Infrastructure Diagram: This diagram depicts the infrastructure requirements for hosting and deploying the system. It can include servers, cloud platforms, networking components, and other infrastructure elements needed to support the system's operation.

4.3 User Stories:

User stories capture the system requirements from the perspective of different stakeholders, such as drivers, parking operators, and administrators. They describe specific features or functionalities that users expect from the system.

Examples of user stories for the AI-enabled car parking system can include:

1. As a driver, I want to be able to see real-time updates on parking space availability so that I can easily find an empty parking spot.
2. As a parking operator, I want to be able to manage and monitor the parking spaces, view occupancy reports, and receive alerts for any issues or violations.
3. As a driver, I want the system to guide me to the nearest available parking space through a mobile app, providing navigation instructions.
4. As an administrator, I want to have access to comprehensive analytics and reports on parking space utilization, peak hours, revenue generation, and other relevant metrics.
5. As a driver, I want the option to reserve a parking space in advance through the system's website, ensuring a guaranteed spot upon arrival.

Use the below template to list all the user stories for the product.

User Type	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**	**Acceptance criteria**	**Priority**	**Team Member**
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering user name and password	I can access my account / dashboard	High	Aji Kumar
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Nurselin Ashik
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sivaram
		USN-4	As a user, I can register for the application through Gmail		Medium	Sutheesh

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Team Member
	Login	USN-5	As a user, I can log into the application by entering email & password		High	Aji Kumar
	Dashboard					

CHAPTER-5

CODING & SOLUTIONING

5.1 Feature 1:

One of the key features added to the AI-enabled car parking project is real-time parking space detection using computer vision techniques. This feature enables the system to analyze video feeds or images from parking lot cameras and determine the availability of parking spaces. The code snippet below demonstrates the implementation of this feature:

```
# Import required libraries
import cv2
# Load the pre-trained model for object detection
model = cv2.dnn.readNetFromDarknet('parking_model.cfg', 'parking_model.weights')
# Read the input image or video frame
frame = cv2.imread('parking_image.jpg')
# Preprocess the image (e.g., resizing, normalization)
# Perform object detection
blob = cv2.dnn.blobFromImage(frame, 1/255, (416, 416), swapRB=True, crop=False)
model.setInput(blob)
output_layers = model.getUnconnectedOutLayersNames()
outputs = model.forward(output_layers)
# Process the detection results
for detection in outputs:
    # Extract relevant information (e.g., bounding box coordinates, confidence score)
    # Apply thresholding or other filtering techniques to identify parking spaces
    # Display the results or save them to a database
```

```
# Release resources
cv2.destroyAllWindows()
```

This feature utilizes a pre-trained object detection model (e.g., YOLO) to identify objects in the input image or video frame. The code processes the detection results and applies filtering techniques to determine parking spaces. The identified parking spaces can then be displayed or saved to a database for further analysis.

5.2 Feature 2:

Another important feature of the AI-enabled car parking project is a user-friendly web application that allows drivers to check parking space availability, reserve parking spots, and make payments. The following code snippet demonstrates the implementation of this feature using the Flask framework:

```
# Import required libraries
from flask import Flask, render_template, request
# Create a Flask application
app = Flask(_name__)
# Define routes
@app.route('/')
def home():
    # Retrieve parking space availability from the database
    # Render the home page with the availability information
    return render_template('home.html', availability=availability)
@app.route('/reserve', methods=['POST'])
def reserve():
    # Retrieve user input from the reservation form
        # Process the reservation request (e.g., check availability, generate
reservation ID)
```

```
    # Save the reservation details to the database
    # Redirect to a confirmation page
    return redirect(url_for('confirmation', reservation_id=reservation_id))
@app.route('/confirmation/<reservation_id>')
def confirmation(reservation_id):
    # Retrieve reservation details from the database
    # Render the confirmation page with the reservation details
    return render_template('confirmation.html', reservation=reservation)
# Run the Flask application
if__name__== '_main_':
    app.run()
```

This feature leverages the Flask framework to create routes for different pages of the web application. The home route retrieves parking space availability from the database and renders the home page with the availability information. The reserve route handles the reservation form submission, processes the reservation request, saves the details to the database, and redirects to a confirmation page. The confirmation route retrieves the reservation details from the database and renders the confirmation page with the reservation information.

CHAPTER-6

RESULTS

6.1 Performance Metrics:

In order to evaluate the performance of the AI-enabled car parking system, several performance metrics can be considered. These metrics provide insights into the system's effectiveness, efficiency, and overall performance. Here are some performance metrics that can be measured:

Parking Space Detection Accuracy: This metric assesses the accuracy of the system in detecting parking spaces. It measures the percentage of correctly identified parking spaces out of the total detected spaces. A higher accuracy indicates a more reliable and precise detection capability.

Processing Time: This metric quantifies the time taken by the system to process a given set of parking data or perform specific operations. It is usually measured in milliseconds or seconds. Lower processing time indicates faster processing speed and better system performance.

System Responsiveness: System responsiveness measures the system's ability to quickly respond to user interactions or requests. It includes metrics such as the average response time for user actions, such as checking parking space availability or making reservations. A lower response time indicates a more responsive system.

Scalability: Scalability refers to the system's ability to handle an increasing number of users, parking spaces, or transactions without significant degradation in performance. It can be measured by monitoring system performance under varying loads or stress testing scenarios.

User Satisfaction: User satisfaction is a qualitative metric that gauges users' perception of the system's usability, functionality, and overall experience. It can be assessed through user feedback, surveys, or usability testing. High user satisfaction indicates a successful implementation of the system.

System Reliability: System reliability measures the system's ability to perform consistently without failures or disruptions. It can be quantified by tracking system uptime, mean time between failures (MTBF), or mean time to repair (MTTR). A reliable system ensures uninterrupted parking services for users.

CHAPTER-7

ADVANTAGES & DISADVANTAGES

Advantages:

Enhanced Parking Efficiency: The AI system can optimize parking space allocation and enable efficient utilization of available parking spots. This can help reduce congestion and improve overall parking efficiency.

Real-time Parking Space Availability: By utilizing computer vision techniques, the system can provide real-time information on available parking spaces to users. This helps drivers find parking quickly, saving time and reducing frustration.

Improved User Experience: With features like automated payment processing and reservation systems, the AI-enabled car parking system offers a convenient and seamless experience for users. It simplifies the parking process and reduces the need for manual intervention.

Optimal Resource Management: By analyzing parking patterns and occupancy data, the system can optimize resource allocation, such as lighting, security, and maintenance. This leads to improved resource management and cost savings.

Increased Revenue Generation: The system can enable efficient revenue collection through automated payment systems and precise tracking of parking durations. This helps parking operators maximize revenue generation and streamline financial processes.

Disadvantages:

Initial Investment and Infrastructure Requirements: Implementing an AI-enabled car parking system may require a significant initial investment in terms of hardware, software, and infrastructure setup. This can pose a financial challenge for some organizations.

Technical Complexity: Developing and maintaining the AI system involves dealing with complex algorithms, computer vision techniques, and integration with existing parking infrastructure. It requires expertise in AI, software development, and ongoing technical support.

Dependency on Technology: The system's reliability and effectiveness are dependent on the availability and proper functioning of the underlying technology. Technical failures or disruptions can impact the system's performance and user experience.

Privacy and Security Concerns: The collection and processing of data for the AI system raise privacy and security concerns. Proper measures must be implemented to protect user data and ensure compliance with data protection regulations.

Limited Adaptability: The AI system may face challenges in adapting to dynamic parking environments or evolving parking regulations. Regular updates and maintenance are required to keep the system up-to-date and aligned with changing requirements.

CHAPTER-8
CONCLUSION

The implementation of an AI-enabled car parking system has the potential to significantly improve the efficiency, user experience, and revenue generation of parking facilities. By utilizing computer vision and AI technologies, the system can accurately detect and monitor parking spaces in real-time, provide users with up-to-date information on parking availability, and streamline the overall parking process.

The advantages of the system include enhanced parking efficiency, real-time availability updates, improved user experience, optimal resource management, and increased revenue generation. These benefits contribute to reduced congestion, time savings for drivers, and a more seamless parking experience.

However, it is important to consider the potential challenges and disadvantages associated with the implementation of such a system. These include the initial investment and infrastructure requirements, technical complexity, privacy and security concerns, and limited adaptability to changing environments.

To overcome these challenges, careful planning, investment analysis, and technical expertise are required. Organizations should assess the feasibility, cost-effectiveness, and potential impact of implementing an AI-enabled car parking system in their specific context.

CHAPTER-9

FUTURE SCOPE

Smart Parking Management: Further development of the AI system can incorporate advanced algorithms and machine learning techniques to optimize parking space allocation, considering factors such as vehicle size, parking duration, and user preferences. This can lead to even more efficient utilization of parking resources and improved traffic flow.

Integration with Smart City Infrastructure: The AI-enabled car parking system can be integrated with other smart city infrastructure, such as traffic management systems and transportation networks. This integration can enable dynamic rerouting and real-time updates on parking availability based on traffic conditions, events, and emergencies.

Predictive Analytics: By analyzing historical data and patterns, the AI system can provide predictive analytics to anticipate parking demand and dynamically adjust parking space availability. This can help in planning for future parking requirements, reducing congestion, and improving overall urban mobility.

Enhanced User Experience: Future developments can focus on enhancing the user experience through personalized recommendations, seamless payment options, and convenient reservation systems. Integration with mobile applications and smart devices can provide real-time updates, navigation assistance, and personalized parking preferences.

Sustainability and Green Parking Solutions: The AI-enabled system can promote sustainable parking practices by integrating electric vehicle charging stations, incentivizing eco-friendly vehicles, and optimizing energy consumption in parking facilities. This can contribute to reducing carbon emissions and promoting eco-friendly transportation options.

CHAPTER-10

APPENDIX

Source Code:

app.py

```
from flask import Flask, render_template
import cv2
import pickle
import cvzone
import numpy as np
app = Flask(__name__)
@app.route('/')
def project():
    return render_template('index.html')
@app.route('/hero')
def home():
    return render_template('index.html')
@app.route('/model')
def login():
    return render_template('model.html')


@app.route('/predict_live')
def liv_pred():
    # Video feed
    cap = cv2.VideoCapture('carParkingInput.mp4')
    with open('parkingSlotPosition', 'rb') as f:
        posList = pickle.load(f)
    width, height = 107, 48
```

```
def checkParkingSpace(imgPro):
    spaceCounter = 0
    for pos in posList:
        x, y = pos
        imgCrop = imgPro[y:y + height, x:x + width]
        # cv2.imshow(str(x * y), imgCrop)
        count = cv2.countNonZero(imgCrop)
        if count < 900:
            color = (0, 255, 0)
            thickness = 5
            spaceCounter += 1
        else:
            color = (0, 0, 255)
            thickness = 2
        cv2.rectangle(img, pos, (pos[0] + width, pos[1] + height), color,
thickness)
        """cvzone.putTextRect(img, str(count), (x, y + height - 3), scale=1,
                           thickness=2, offset=0, colorR=color)"""
    cvzone.putTextRect(img, f'Free: {spaceCounter}/{len(posList)}',(100, 50),
scale=3,
                       thickness=5, offset=20, colorR=(200, 0, 0))
while True:
    if cap.get(cv2.CAP_PROP_POS_FRAMES) ==
cap.get(cv2.CAP_PROP_FRAME_COUNT):
        cap.set(cv2.CAP_PROP_POS_FRAMES, 0)
    success, img = cap.read()
    imgGray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    imgBlur = cv2.GaussianBlur(imgGray, (3, 3), 1)
    imgThreshold = cv2.adaptiveThreshold(imgBlur, 255,
cv2.ADAPTIVE_THRESH_GAUSSIAN_C,
                                         cv2.THRESH_BINARY_INV, 25, 16)
    imgMedian = cv2.medianBlur(imgThreshold, 5)
```

```
        kernel = np.ones((3, 3), np.uint8)
        imgDilate = cv2.dilate(imgMedian, kernel, iterations=1)
        checkParkingSpace(imgDilate)
        cv2.imshow("Image", img)
        # cv2.imshow("ImageBlur", imgBlur)
        # cv2.imshow("ImageThres", imgMedian)
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
if __name__ == '__main__':
    app.run(debug=True)
```