A

Project On

PARKING SPACE TRACKER

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

A. MADHAVI	**22BH1A0502**
A. RIKSHITH	**22BH1A0507**
B. RAKESH	**22BH1A0523**
B. ANSHU	**22BH1A0537**
B. CHARITHA	**22BH1A0546**
D.MEGHANA	**22BH1A0551**
D.SAITEJA	**22BH1A0561**
B. PURNA SAI	**22BH1A0519**

SUBMITTED TO

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ST. MARY'S ENGINEERING COLLEGE

(Approved by AICTE, NEW DELHI. & Affiliated to JNTU-HYDERABAD)

Deshmukhi (V), Pochampally(M), Yadadri- Bhuvanagiri (D), Telangana -508248

2024-2025

CERTIFICATE

This is to certify that, the Presentation entitled **"PARKING SPACE TRACKER"** is submitted by **A.Madhavi (22BH1A0502)**, **A.Rikshith (22BH1A0507)**, **B.Rakesh (22BH1A0523)**,**B.Anshu (22BH1A0537)**, **B.Charitha (22BH1A0546)**, **D.Meghana (22BH1A0551)**, **D.Saiteja(22BH1A0561), B.Purna Sai(22BH1A0519)** B.Tech Computer Science and Engineering, In Partial fulfilment**.** For the requirement of the presentation and for the award of the Degree of Bachelor of Technology during the academic year 2024-25.

ABSTRACT

In modern era, the trouble of parking is also growing because of the growth within side the quantity of vehicles. With increasing the number of vehicles over the years, parking has become an important issue particularly in commercial environments such as shopping malls, schools, government offices, and airports. Finding a vacant parking space in town areas is time-consuming and therefore not satisfying for potential visitors or customers. In this article, a low-cost video-based system is proposed for parking space detection. A CCTV feed is used in combination with a desktop computer for computation. Different feature extractors and computer vision algorithms were evaluated in order to retrieve accurate state information for each of the observed parking space. There is an auto mechanism that can park vehicle automatically but it is required to detect which parking slot is available and which one is busy. In this project parking space detection using image processing. In this project parking-space occupancy detection, Visualization of free parking spaces, Parking statistics, Wireless communication, easily available components, System will get Live-stream video of the parking lot from camera. Images are captured when a car enters or leaves the parking lot. System will also work in Mobile phone (Browser).

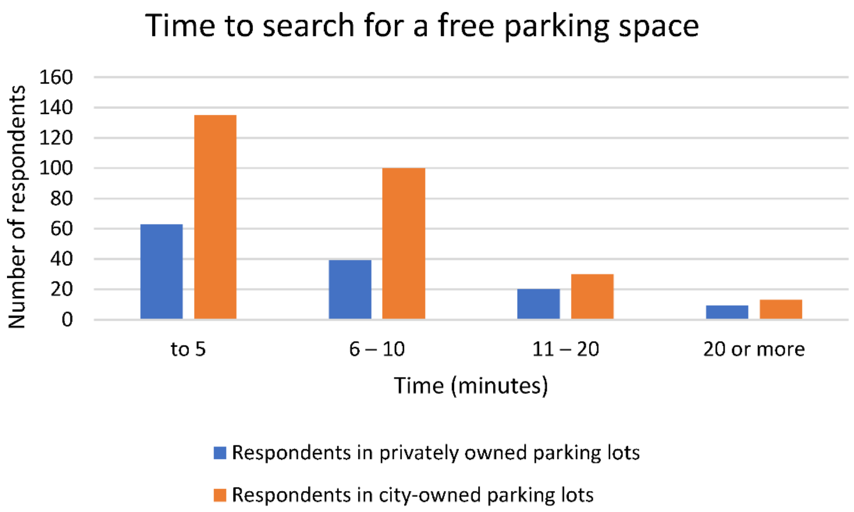
TABLE OF CONTENTS

INTRODUCTION

Nobody enjoys circling in car parks looking for non-existent empty parking spaces and nobody enjoys driving while looking at both sides for an available space as they are really busy. A lot of time and effort could be saved if parking information on parking lot availability could be accessed by the driver's phone or whatever online method that anybody can be reached (Ichihashi et al., 2010).

Most of the current vacant parking space detection systems engage simple parking lot entrance and exit sensors which have no way of determining where the free parking slots are available. Moreover, the current vehicle parking space detection systems cannot determine when more than one parking space has been obstructed either by an oversized vehicle or by a poorly parked car. Existing parking space occupancy detection systems use elementary trip sensors at the entry and exit points of parking lots. Miserably, this type of system fails when a vehicle takes up more than one slot or when a parking lot has different types of parking slots or unequal slots. Recently, some researches have been done on improving parking lots detection systems. Systematized car-park routing systems could support drivers to get an optimal parking space instantly and with no trouble. But existing approaches which detecting vacant parking spaces are either high priced due to hardware requirement for each parking lot or do not provide a full-scale occupancy guidance.

Some approaches use visual surveillance that requires real-time interpretation of image sequences in order to automate the detection of parking spaces. Some other approaches keep tracking and recording the movement of vehicles in order to find the empty parking space. Nevertheless, these real-time detection methods require high computation and large storage (Jain et al. 1995). Empty parking space detection system that can automatically identify empty parking spaces and guide users to move the vehicle to it will save a lot of time, money, effort and are highly desirable (True, 2007). Therefore, a camera-based system is proposed that would use computer vision algorithms for noticing vacant parking spaces. This algorithm is used as a combination of car feature point detection and colour histogram classification to detect vacant parking spaces in overhead images.



Many shopping centres do not have specific information about parking availability. This causes the driver to find the parking lot himself. This is not effective, because many cars enter the shopping area and each of them is looking for a parking space. So, many cars were milling about causing local congestion in the parking lot. In general, parking attendants help to find a parking space to show an empty parking location. This can be helpful but ineffective as parking attendants have to monitor all locations manually. Some research has made breakthroughs to solve parking problems effectively, the installation of sensors at every point of the parking area can be used to detect and provide information about the location of available parking areas.

1.1 PURPOSE

- Enhanced Parking **-** This enables drivers to quickly find the best spot available which will save time, resource and effort. The parking space would be utilized efficiently. Various factors such as size of vehicle, weight, etc. are also considered while allotting the spot.
- Reduced Congestion - Traffic flow around the parking lot will increase as overcrowding of vehicles will decrease.
- Reduced Pollution - It goes along with reduced congestion. Smart parking opens the option of quicker parking which decreases the emissions by car standing idle in parking lot looking for space.
- Enhanced User Experience - Parking management solution will expand and enhance the experience for a user by giving them a unified procedure. Driver's payment, spot identification, space search and time notification all becomes convenient.
- Driver's Experience - Parking management solution helps driver find a space quickly and easily with zero frustration level. It enables them to have a seamless experience, from locating spot to live notifications to paying, all at once.
- Improved Safety - Safety and security can be achieved easily by smart parking. Data can be provided to parking lot employees for better management and lookout for violations and suspicious activities.
- Reduced Operational and Management Costs - More mechanization and less manual action save money on work cost. A smart parking management solution is surely a great investment for any building, office, mall, etc. As the global population rises, we will witness more growth and urbanization which will result into a greater number of vehicles. Before selecting a parking management solution ensure it meets your requirement, is convenient to install and easy to use.

2. LITERATURE SURVEY

2.1 Smart Parking Systems

Generally, smart parking systems obtain information about available parking spaces in a particular geographic area and mechanism is real-time to place vehicles at vacant slots. The use of accurate sensors, real-time data collection and mobile-phone-enabled booking systems that allow people to reserve parking in advance or very accurately predict where they will likely find a place to park their vehicles are some critical factors need to be considered in this system development (Ahad et al., 2016; Leogane et al., 2015). When deployed as a system, smart parking thus scales down car emissions in urban centers by reducing the need for people to needlessly circle city blocks seeking for parking. It also permits capitals to carefully manage their parking supply. Smart parking helps one of the huge problems with driving in urban areas such as finding empty parking spaces and controlling illegal parking.

Recently, a number of solutions that employ different sensors have been suggested, but they are either too expensive to implement or have failed to be effective. We can conveniently categorize these car parking management or guidance systems based on their technologies into:

1. Counter-based
2. Wired Sensor-based
3. Wireless Sensor-based
4. Image-based.

The counter-based system that counts the number of cars that enter and exit a car park area considered to be the simplest of all. This system is capable of providing information on the total number empty parking spaces in an area but cannot guide the driver to the location of empty parking spaces. The wired sensor system relies on installing ultrasonic systems in each parking lot to detect the occupancy of parking spaces. These sensors are managed by a control unit to which they are wired and are capable of directing maintenance costs that result from the long and complicated wiring that is required to get them functional.

A simple extension of wireless sensor systems to parking guidance has also been available in recent years. The sensors in most wireless systems are micro-controlled and typically include multi-sensor controls as well. Some common examples of these systems include that uses the extended crossbow network architecture, that adopts an anisotropic magneto-resistive magnetic field sensor together with a microprocessor transceiver.

The image-based techniques based on video captures are the next trend. These are most suitable where parking areas are already monitored by CCTV surveillance systems (Bhaskar et al.). Image-based parking guidance systems are simple and inexpensive.

2.2 AUTOPARK: A Sensor Based, Automated, Secure and Efficient Parking Guidance System

Each slot is equipped with an Ultrasonic Distance Sensor which is used to detect the occupancy of slots i.e., whether a particular slot is vacant or occupied. The connection between the sensor and the microcontrollers at the entry gate is wired, as the system follows an infrastructure-based architecture and we are aware of the entire region of interest. Thus, a wired connection would be reliable, appropriate, and cost-effective. The ultrasonic sensor deployed at the slots provides precise measurements of distance up to 3 m. The sensor works by transmitting an ultrasonic burst and providing an output pulse that corresponds to the time required for the burst echo to return to the sensor (Magog & Aswani, 2013). The distance to the target can easily be calculated by measuring the pulse width.

In this scenario, the target is the car being parked at each slot, and the distance is measured and occupancy is decided on the basis of a predefined threshold value. The sensor values are aggregated at the entry gate where the microcontrollers are placed. The microcontrollers then transmit the data to the base station using serial communication (Magog & Aswani, 2013).

The system was evaluated against the conventional parking systems for time and human effort it saves (Hastie et al., 2001). In a conventional parking system, the incoming vehicle has no information about any free parking slot available in the parking area. It searches for free parking slot moving in different directions in order to reach a free slot, this results in a longer searching time in cases where the parking area is highly occupied or the parking area itself is very large (Magog & Aswani, 2013). This AUTOPARK system helps in reducing this searching time and effort by providing information as well as direction guidance towards the free parking slot.

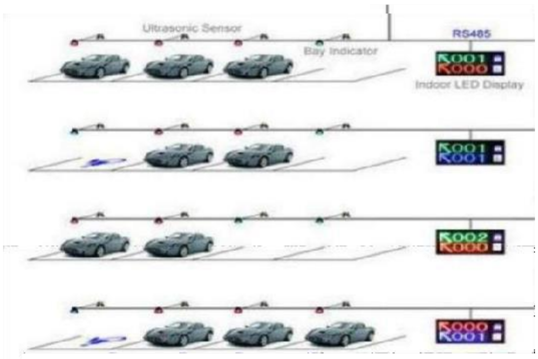


Figure 2.1: Ultrasonic Sensor Configuration of AUTOPARK

2.3 Smart Parking System using Wireless Sensor Networks

The idea behind this system is when a user enters the parking facility, at the entrance, there will be a keypad and a display. The driver types his mobile phone number using the keypad. On successful entry of the phone number, the ID of the nearest empty parking slot, time of entry and route direction information will be displayed on the monitor and it will also be sent to the user's mobile phones via SMS. In order to incorporate the SMS feature, a GSM modem is connected to the CSS. A java-based SMS gateway at the CSS provides the essential functionality using at commands to the send the SMS (Jeffrey et al., 2012).

When the driver parks the car in the designated slot, a timer is started in the mote present in that slot. The move will inform the CSS that the slot has been currently occupied. The CSS will update the database of the motes with the occupancy information along with the corresponding user mobile phone number. This is done to uniquely identify the parking slot with the vehicle.

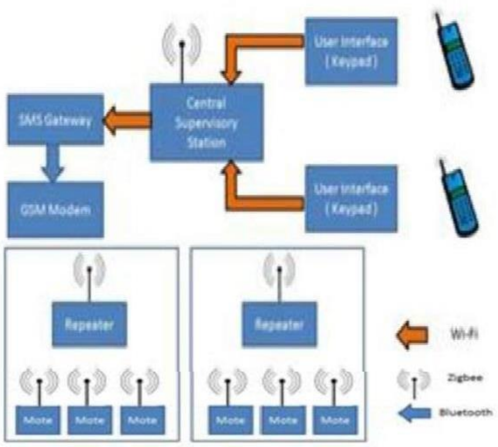


figure 2.2: Wireless Sensor Network Overview

When the driver leaves the parking slot area, immediately the timer in that slot’s mote stops. The timer data is communicated to the CSS. The CSS consults its database, extracts the mobile phone number field corresponding to the mote ID it received, and sends an SMS to the user. The SMS will contain information on how long the vehicle was parked and the billing amount. By the time the driver reaches the exit of the parking facility, he/she would have received the billing information via an SMS.

3. SOFTWARE REQUIREMENTS

- Back – End:
  - Python
  - OpenCV
  - Pickle Module
  - NumPy
- MS Word or later
- Web Browser: Microsoft Internet Explorer, Mozilla, Google Chrome or later
- Operating System: Windows XP / Windows7/ Windows 8/ Windows 10/ Windows 11

figure 3: Software Requirements

4. DESIGN AND IMPLIMENTATION

4.1 DESIGN SYSTEM

To detect and find available parking locations using vision-based systems, the procedures that must be performed are manually segmenting parking models, car and non-car data, training data to determine cars and non-cars, reducing adaptive backgrounds, extracting features, and making decisions. Figure is a procedure for detecting and providing information about the location of available parking areas.

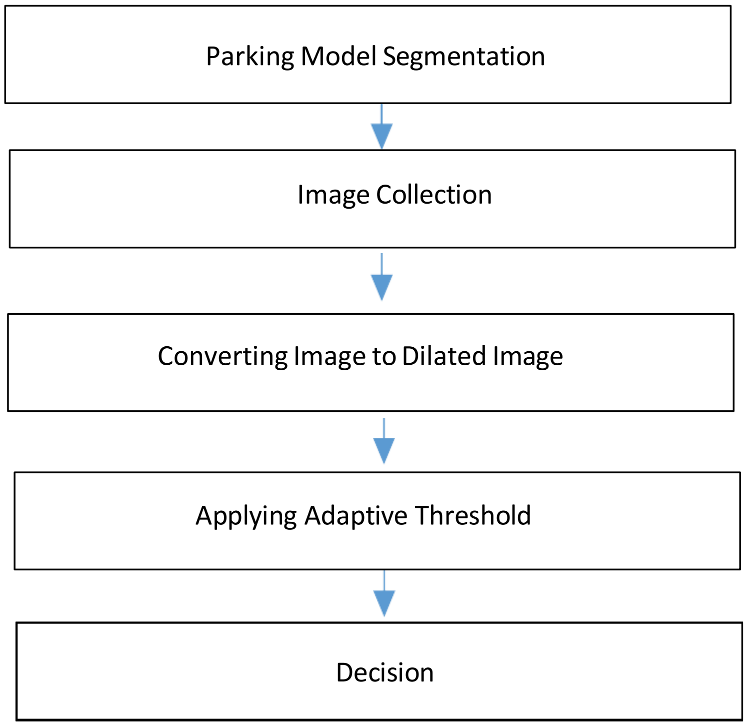


Figure 4.1: Design System

4.2 IMPLEMENTATION

➢ Overview of the steps:

There are two main steps in building this parking detection model:

- Detecting the position of all available parking spots
- Identifying if a parking spot is vacant or occupied

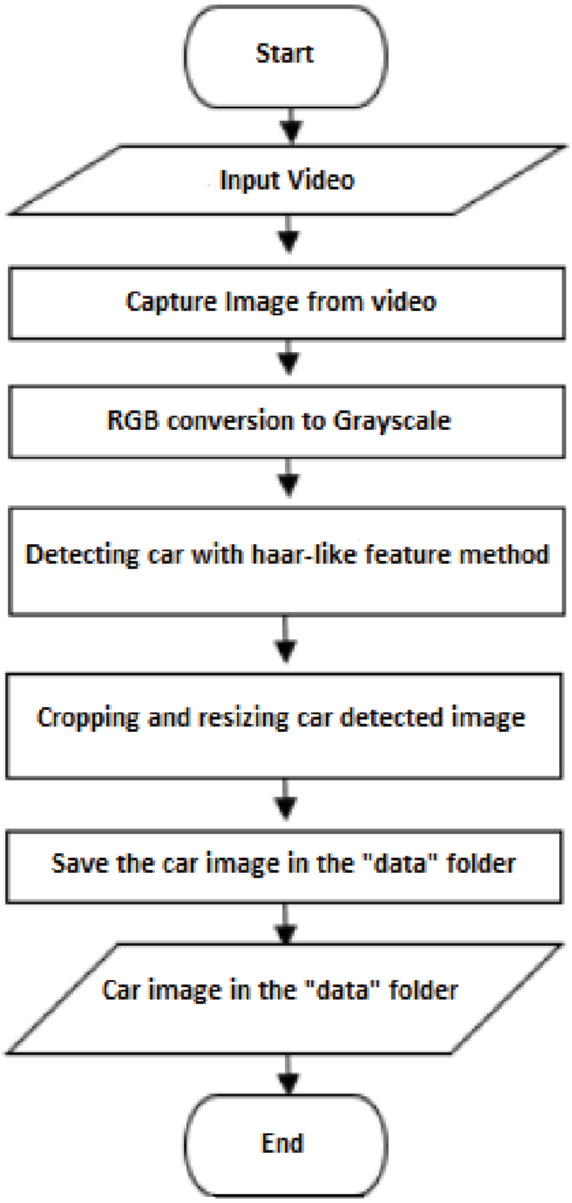


Figure 4.2: Implementation

4.2.1 Video File Processing

Firstly, we need to collect the top-view (i.e., bird's eye view) of the parking lot which can be collected using the drones but the main problem we face with drones is that the real-time video from the drones is not study for the marking of the keyframe so it is recommended to collect the video using the study source like CCTV cameras such that the video is easy to keep track of and easy to mark the keyframes from the image (i.e., extracted from the frame). The video consists of 30 frames or segments. Then the video is reduced into single frame or segment. Then from each segment, a keyframe is extracted and further processing is applied to this keyframe to reduce computational complexity.

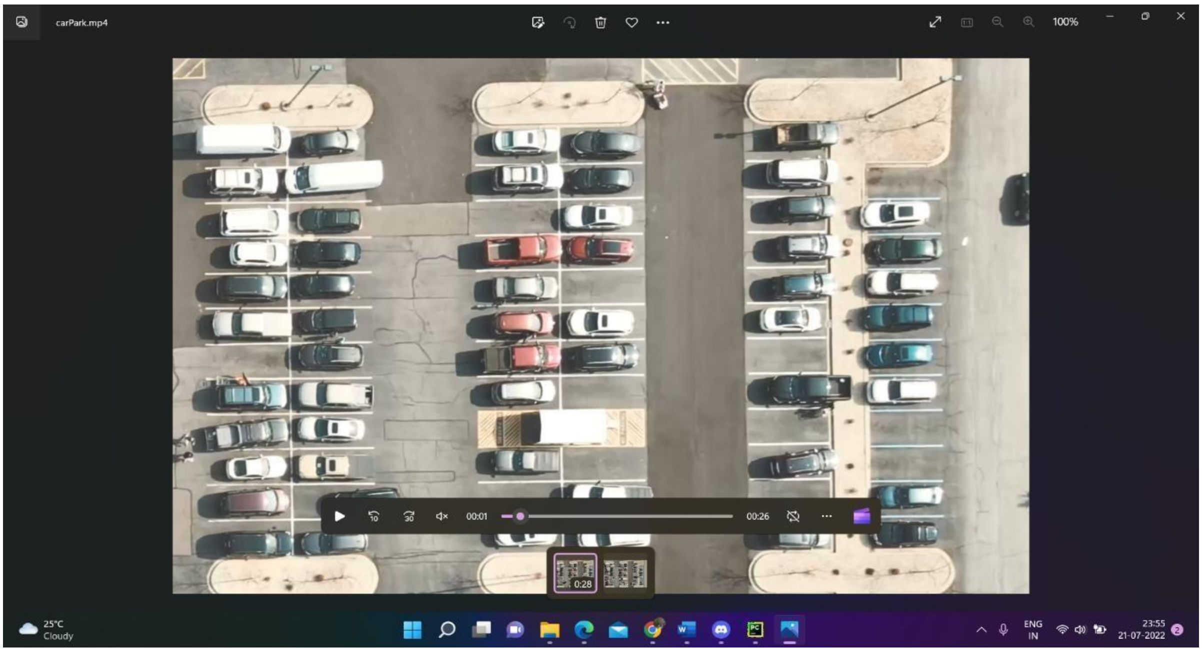


Figure 4.2.1: Video file Processing

Figure 4.2.2: Image or Frame extracted from video

Figure 4.2.3: Example of CCTV view of parking lot

4.2.2 Boundary Drawing

Initially, the parking arena has no parking lines. User will manually input the coordinates of the parking area and parking slots. Many online tools are available to markout the rectangles on a static image. That coordination's are then fed to the YAML file which is based on data-oriented language structure. The system automatically generates virtual parking lines keeping in view the size of the vehicle.

Boundaries can be simply marked using the "cv2.rectangle" and the mouse-clicks i.e., using the left click for the marking and the right click for removing the existing marks from the image. We can mark the one column and duplicate it for remaing columns using the for loops. However, we can see that the columns are not identical, for example in the figure 11 second column consists of a trailer in between and third column consists of a foot path. We need to accommodate for the differences in the columns and tracking the parking lot manually is the difficult yet a better option for tracking the boundaries.

Figure 4.2.4: Boundary Drawing

4.2.3 Processing

A unique numeric label is assigned to each parking plot. Then the system analyzes each frame and checks the virtual static bounding for occupied parking spaces and then the system increases the number of occupied slots by 1. Likewise, free slots also calculated and keep a track of the count.

The image with the boundaries is converted into the binary grayscale image which is the black and white image and further the grayscale image is applied with gaussian blur and median blur. Then the result image is converted to dilated image to emphasis on the corners and vertices in the keyframe. Finally adaptive threshold is applied to the each keyframes which in this case is 900px.If the pixel count is more than 900px that is if the corners and edges are more than the parking space is occupied and if the pixel count is less than 900px in that particular key frame that is number of corner and vertices are less then parking space is free. A tracker keeps of the free parking spaces in the parking lot.

Figure 4.2.5: Example of Processed image

4.2.4 Presenting

The number of detected objects is based on the number of car objects detected in the processing area of each parking lot. The processing area is the masking area obtained from the area segmentation process. The results of the number of objects are used as a benchmark for calculating the remaining empty parking slots which will later be used as parking space information. For the classification of the availability status of the parking lot consists of two variables, namely the presence and absence of objects in the parking lot slot.

CODE

```
import cv2
import pickle
import cvzone
import numpy as np
# Video feed
cap = cv2.VideoCapture('carPark.mp4')
with open('carParkPos', '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]
 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: {str(spaceCounter)}/{len(posList)}', (100, 50),
scale=3, thickness=5, offset=20, colorR=(0, 200, 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()
    if not success:
        break
    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.waitKey(10)
# Separate code for setting positions
width, height = 107, 48
try:
    with open('carParkPos', 'rb') as f:
        posList = pickle.load(f)
except:
    posList = []
def mouseClick(events, x, y, flags, params):
    if events == cv2.EVENT_LBUTTONDOWN:
        posList.append((x, y))
    if events == cv2.EVENT_RBUTTONDOWN:
        for i, pos in enumerate(posList):
```

```
            x1, y1 = pos
            if x1 < x < x1 + width and y1 < y < y1 + height:
                posList.pop(i)
                break
    with open('carParkPos', 'wb') as f:
        pickle.dump(posList, f)
while True:
    img = cv2.imread('carParkImg.png')
    for pos in posList:
        cv2.rectangle(img, pos, (pos[0] + width, pos[1] + height), (255, 0, 255), 2)
    cv2.imshow("Image", img)
    cv2.setMouseCallback("Image", mouseClick)
    cv2.waitKey(1)
```

https://drive.google.com/file/d/1NKBqVlVz7NrpYEbltXZIqz8TX92mds5_/view?usp=drivesdk

5. SIMULATION RESULTS

As the conceptualization of this project is to discover the parking system by using image processing instead of using sensor base this intelligent parking system is developed using an integrated image processing approach to reduce the cost of sensor and wiring hassle. The following results were obtained when the experiments are carried out.

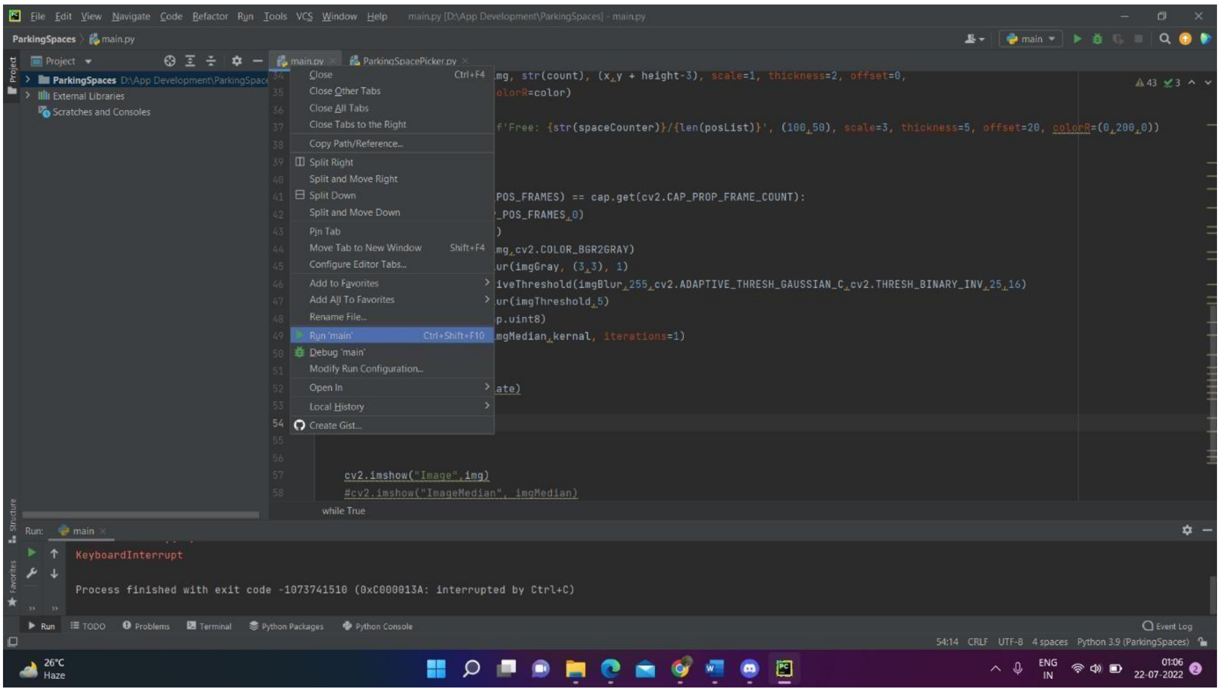


Figure 5.1: Running the code

At first, the parking lot is vacant with a maximum capacity of 5 slots. Only two slots were occupied. Therefore, the available parking slot numbers were shown and the location guidance for the available slots were listed for the user through the android application. Furthermore, status was showing as "AVAILABLE" for the convenience of the user as shown in Figure 4. The recorded video was on a sunny day. Detection was 99.9% accurate.

At first, the parking lot is vacant with a maximum capacity of 13 slots. Only 56 slots were occupied. Therefore, the available parking slot numbers were shown and the as "FREE:" on the top.

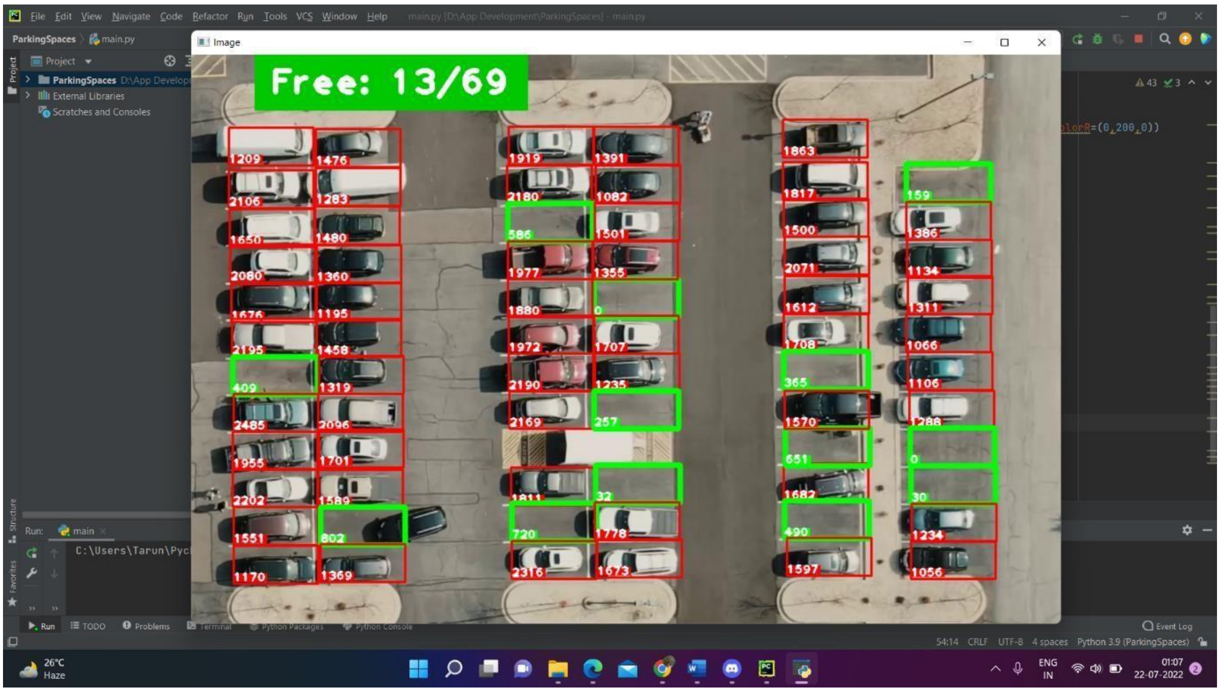


Figure 5.2: Initial Result

We can notice that number of free slots increasing as the cars are moving out of the keyframes (i.e., Parking Slots). In the next result, the parking lot is vacant with a maximum capacity of 15 cars in the arena. All slots were occupied.

Figure 5.3: Final Result

CONCLUSION

In this paper, we developed a promising video-based system for vacant parking space detection which can be adopted by a car-park routing system to navigate drivers to a comfortable parking space. We evaluated different combinations of image features and machine learning algorithms. This system reached an accuracy of 94.95% on classifying parking situations. Furthermore, we achieved real-time speed for all six modules and the system as a whole.

Even though our system is tested only on outside parking lots, it is also imaginable to use it in parking garages. In this case, on the one hand, more cameras are needed since the cameras must be positioned closer to the parking spaces due to the lower suitable space and on the other hand, the lightning conditions have to be adjusted.

Improvements can be achieved by minimizing the influences of adjacent cars overlaying the labeled area due to the camera perspective. Further experiments should be done on other car-parks featuring different visual properties.

REFERENCES

- https://www.computervision.zone/courses/parking-space-counter/
- https://github.com/topics/parking-space
- https://www.youtube.com/watch?v=caKnQlCMIYI
- Ahad, A., Khan, Z.R., & Ahmad, S.A. (2016). Intelligent Parking System, In Scientific Research-Academic Publisher, WJET Vol.4 (2).
- Bhaskar, H., Wergh, N., Mansoori, S.A. Rectangular Empty Parking Space Detection Using sifts Based Classification. https://mafiadoc.com/rectangular-empty-parking-space-detection-using-semantic-scholar_59b44b151723ddd5c6186bbe.html.
- Maggo, S., Aswani, R. (2013). AUTOPARK: A Sensor Based, Automated, Secure and Efficient Parking Guidance System.