

Project Planning Phase-4

Technology Stack

Date	8th November 2023
Team ID	592031
Project Name	Project - AI Enable car parking using OpenCV
Maximum Marks	4 Marks

a) Technical Architecture

Component	Description
Sensors and Input Devices	Cameras, additional sensors (e.g., ultrasonic sensors)
Hardware Platform	Computer, embedded system (e.g., Raspberry Pi), specialized AI hardware
Image/Video Processing Layer	OpenCV for video capture, analysis, object detection, and transformations
Object Detection	Pre-trained/custom models for car, pedestrian, obstacle detection
Control and Path Planning	Control algorithms (e.g., PID controllers), path planning
User Interface	Optional UI for monitoring and control
Integration Layer	Manages data flow and communication between components
Cloud Connectivity (Optional)	Cloud services for remote features (e.g., monitoring, updates)

b) Open Source Frameworks

Framework/Library	Description
OpenCV	Image and video processing, object detection
TensorFlow	Machine learning framework (object detection training)
PyTorch	Deep learning framework (object detection training)
ROS (Robot Operating System)	Robotic control and communication
NumPy	Numerical operations and data manipulation
Matplotlib	Data visualization (plotting)
OpenCV-Python	Image display and manipulation
Pandas	Data manipulation and analysis

c) Third-party APIs

API	Use Case
Maps and Geolocation APIs	Finding nearest parking space, geolocation
Speech Recognition and NLP APIs	Voice commands, natural language interactions
IoT and Sensor APIs	Sensor data collection and management
Cloud Services APIs (for cloud deployment)	Data storage, remote monitoring, updates

d) Cloud Deployment

Cloud Deployment Element	Description
Data Storage	Cloud storage for video footage, logs, data
Scalability	Scalable infrastructure for more cameras, parking spaces
Remote Monitoring	Dashboard or remote monitoring application
Over-the-Air Updates	Simplified process for software updates
Security	Cloud security features to protect data and system
Cost Management	Cost-effective resource usage

