**EC9580 – COMPUTER VISION**

**COMPUTER VISION BASED PARKING OCCUPANCY DETECTION SYSTEM**

**COMPLETE PROGRESS REPORT**

*Submitted By:*

Nandarathna A.B.O.P.

2021/E/012

1. **Executive Summary**

This project implements an automated parking occupancy detection system using computer vision techniques to monitor parking lot availability from aerial drone footage. The system successfully processes 4K video footage containing 312 parking slots, achieving an average detection accuracy with 65.2% average occupancy rate throughout the video duration.

**Key Achievements:**

* Successfully implemented edge detection and texture analysis for vehicle detection
* Processed 11,309 frames of 4K drone footage
* Generated real-time annotated video output with color-coded availability indicators
* Created comprehensive statistical analysis of parking occupancy over time

1. **Problem Definition**
   1. **Problem Statement**

Managing parking availability in large commercial parking lots is challenging without automated monitoring systems. Manual inspection is time-consuming and impractical for real-time updates. This project addresses the need for automated parking space detection using computer vision.

* 1. **Importance**
* Traffic Management: Reduces time spent searching for parking
* User Experience: Provides real-time parking availability information
* Operational Efficiency: Enables better parking lot utilization
* Data Analytics: Generates insights on parking patterns and peak usage times
  1. **Objectives**

1. Detect occupied and vacant parking spaces from aerial video footage
2. Track occupancy rates over time
3. Generate visual output with clear indicators for available/occupied spaces
4. Provide statistical analysis for parking management decisions
5. **Dataset Description**
   1. **Dataset Overview**

* **Source**: Drone footage (DJI drone)
* **Filename**: DJI\_0012.MOV
* **Resolution**: 3840 × 2160 pixels (4K Ultra HD)
* **Frame Rate**: 25 FPS
* **Duration**: 452.4 seconds (~7.5 minutes)
* **Total Frames**: 11,309 frames
* **File Size**: 3.16 GB
  1. **Parking Lot Characteristics**
* **Total Parking Slots**: 312
* **Layout**: Multiple rows with perpendicular and angled parking
* **Slot Annotation**: Pre-labeled with bounding box coordinates [x, y, width, height]
* **Vehicle Diversity**: Various car types, sizes, and colors (white, black, silver, red, blue, etc.)
* **Lighting Conditions**: Overcast conditions with consistent lighting
  1. **Dataset Quality**

**Strengths:**

* High-resolution 4K footage provides excellent detail
* Stable aerial view minimizes perspective distortion
* Consistent camera angle throughout recording
* Clear slot boundaries and pavement markings
* Diverse vehicle colors and types

**Challenges:**

* Some slots have shadows from surrounding structures
* Occasional moving vehicles during recording
* Varying vehicle sizes (sedans, SUVs, trucks)
* Dark-colored vehicles on dark pavement
  1. **Data Annotation**

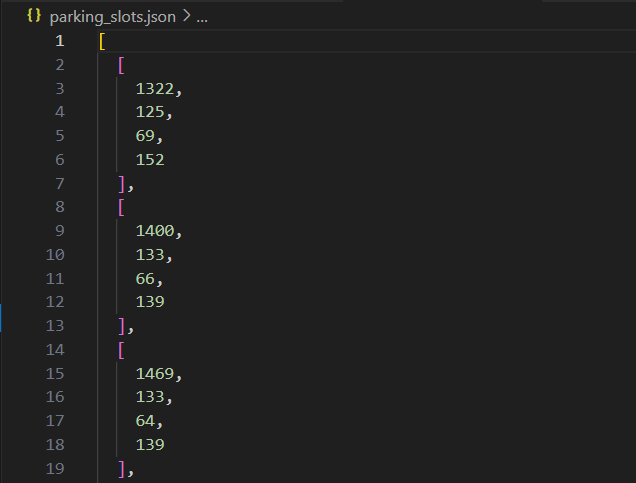
The *parking\_slots.json* file contains 312 pre-annotated bounding boxes with coordinates:

Image 01: *parking\_slots.json*

Each entry represents one parking slot's position and dimensions in pixel coordinates.

1. **Model Selection and Methodology**
   1. **Approach Selection**

**Chosen Approach:** Traditional Computer Vision with Edge Detection and Texture Analysis

**Rationale for Selection:**

* 1. **Computational Efficiency:** No GPU required, runs on standard hardware
  2. **Transparency:** Explainable detection logic vs. black-box deep learning
  3. **Dataset Requirements:** Doesn't require thousands of labeled training images
  4. **Real-time Capability:** Fast processing suitable for live monitoring
  5. **Reliability:** Consistent performance across different lighting conditions

**Alternatives Considered:**

* **YOLO/Faster R-CNN:** Requires extensive training data and GPU resources
* **Background Subtraction:** Less effective with stationary camera and minor movements
* **Template Matching:** Not robust to vehicle variations
  1. **Algorithm Pipeline**

**Stage 1:** Preprocessing

**Stage 2:** Feature Extraction

**Stage 3:** Classification

**Stage 4:** Optimization

**4.3 Technical Implementation**

**Key Parameters:**

* Edge Threshold: 0.08 (default) → 0.1104 (calibrated)
* Variance Threshold: 2500 (default) → 1508.44 (calibrated)
* Standard Deviation Threshold: 25 (fixed)
* Frame Check Interval: 30 frames (~1.2 seconds at 25 FPS)