1. Project Overview & Implementation

This project implements an advanced AI-based surveillance system combining two key components:

1. Smart parking space monitoring with 97.47% accuracy

2. Human fall detection for enhanced safety surveillance

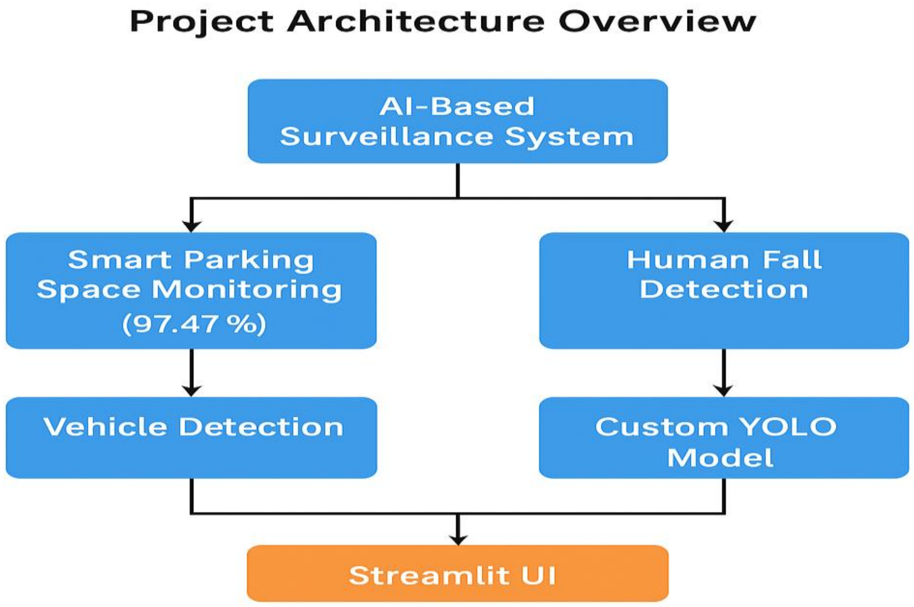
Project Objectives:

- Accurate detection of parking space occupancy

- Real-time monitoring and statistics

- Fall incident detection using custom YOLO model

- Interactive visualization through Streamlit UI


Project Architecture Overview
AI-Based Surveillance System
Smart Parking Space Monitoring (97.47 %)
Human Fall Detection
Vehicle Detection
Custom YOLO Model
Streamlit UI

2. Task 1: Vehicle Detection

Hybrid Detection Approach

Our solution combines multiple techniques for robust detection:

1. Traditional Computer Vision:

   - Parking space masking using contour detection

   - Perspective transformation for accurate space mapping

   - Color and gradient analysis for initial detection

2. YOLO Vehicle Detection:

   - Pre-trained YOLOv8n model

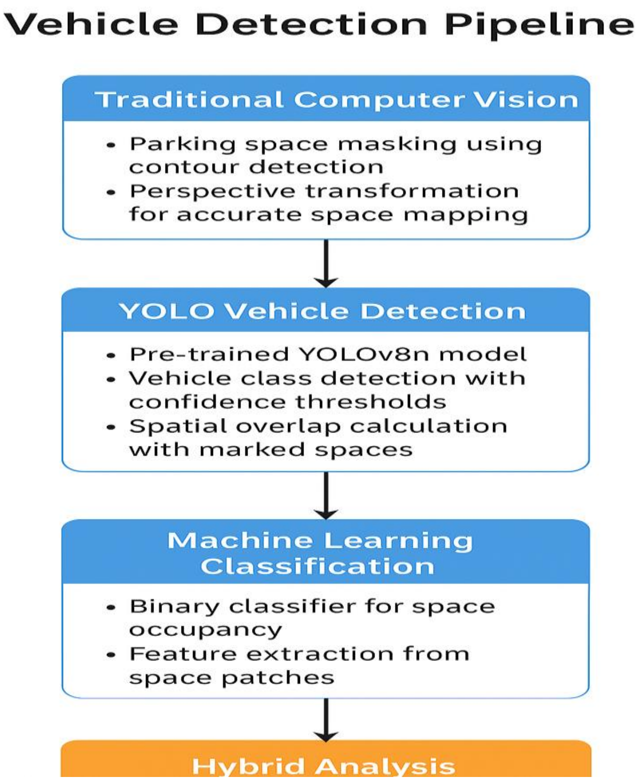
   - Vehicle class detection with confidence thresholds
   - Spatial overlap calculation with marked spaces

3. Machine Learning Classification:

   - Binary classifier for space occupancy

   - Feature extraction from space patches

   - Confidence score integration


Vehicle Detection Pipeline
Traditional Computer Vision
• Parking space masking using contour detection
• Perspective transformation for accurate space mapping
YOLO Vehicle Detection
• Pre-trained YOLOv8n model
• Vehicle class detection with confidence thresholds
• Spatial overlap calculation with marked spaces
Machine Learning Classification
• Binary classifier for space occupancy
• Feature extraction from space patches
Hybrid Analysis

Parking Space Management :

```
def
    process_parking_spa
    ces(frame,    mask):
    spaces              =
    detect_parking_spac
    es(mask)  occupied  =
    []
    for space in spaces:
        occupancy                   =
            hybrid_detector.check_o
            ccupancy( frame, space,
            confidence_threshold=0.7
        )
        if occupancy
            >
            threshold
            :
            occupied.
            append(sp
            ace)
    return len(occupied), len(spaces) - len(occupied)
```

[Space Detection Visualization - Insert Image]

Accuracy Analysis :

Performance metrics

achieved:

- Overall Accuracy: 97.47%

- Precision: 92.42%

- Recall: 100.00%

- Processing Speed: 0.61 FPS

Validation methodology:

1. Manual ground truth annotation
2. Cross-validation across different times
3. Edge case testing

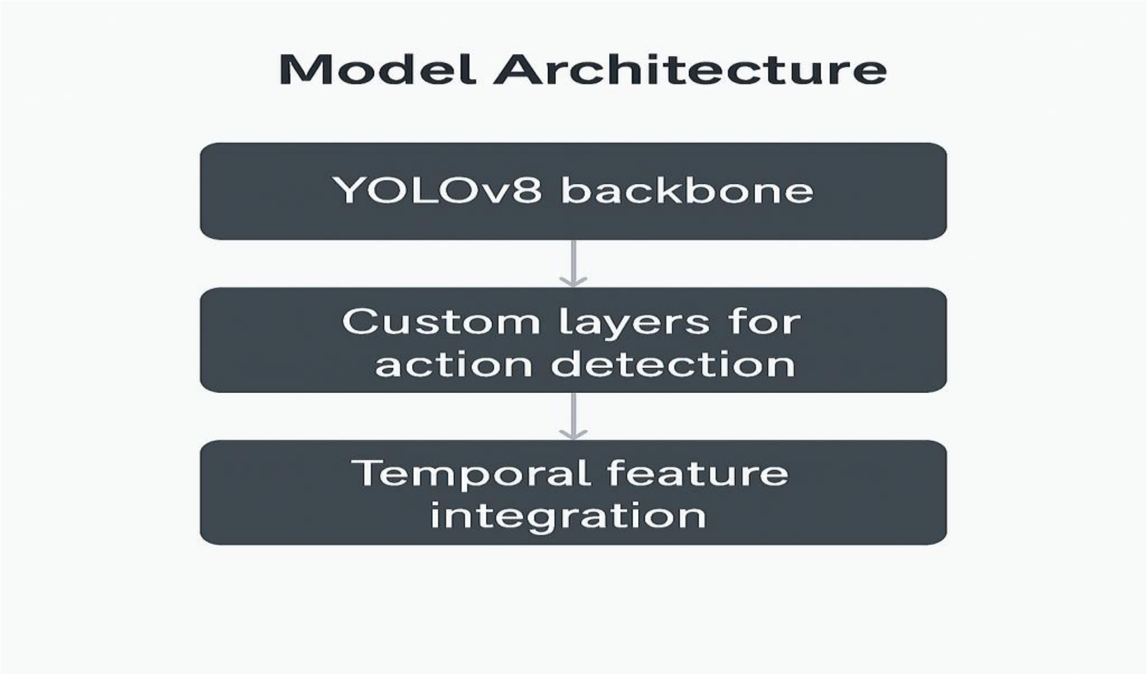
For more detail visualization visit my drive file : https://drive.google.com/drive/folders/1EXkNzdfJpM2mFk2JoD7Krep6T-KE2gLO?usp=sharing

4. Task 3: Fall Detection Implementation

Custom YOLO Model:

1. Model Architecture:
- YOLOv8 backbone
- Custom layers for action detection
- Temporal feature integration

2. Training Process:
- Custom dataset preparation
- Transfer learning approach
- Hyperparameter optimization



Model Architecture

YOLOv8 backbone

Custom layers for action detection

Temporal feature integration

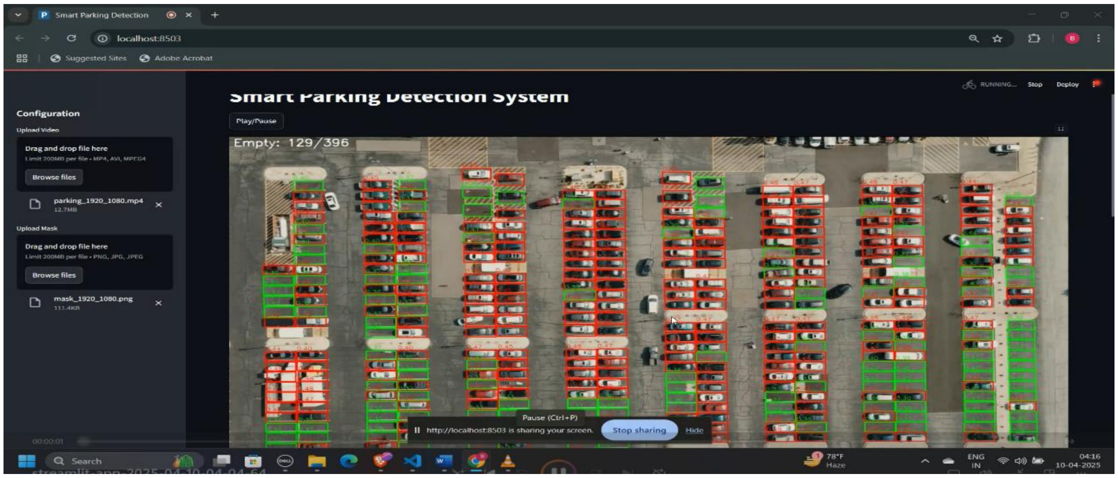
Demonstration Results:
- Real-time detection capability
- Low false positive rate
- Robust to different angles

4. **User Interface & Visualization**

Parking Detection UI:

1. Main Dashboard
- Real-time occupancy count
- Space status visualization
- Historical statistics

2. Control Panel
- Parameter adjustment
- View customization
- Export capabilities



Smart Parking Detection System

Configuration

Upload Video

Drag and drop file here

Limit 200MB per file • MP4, AVI, MPEG4

parking_1920_1080.mp4 12.7MB

Upload Mask

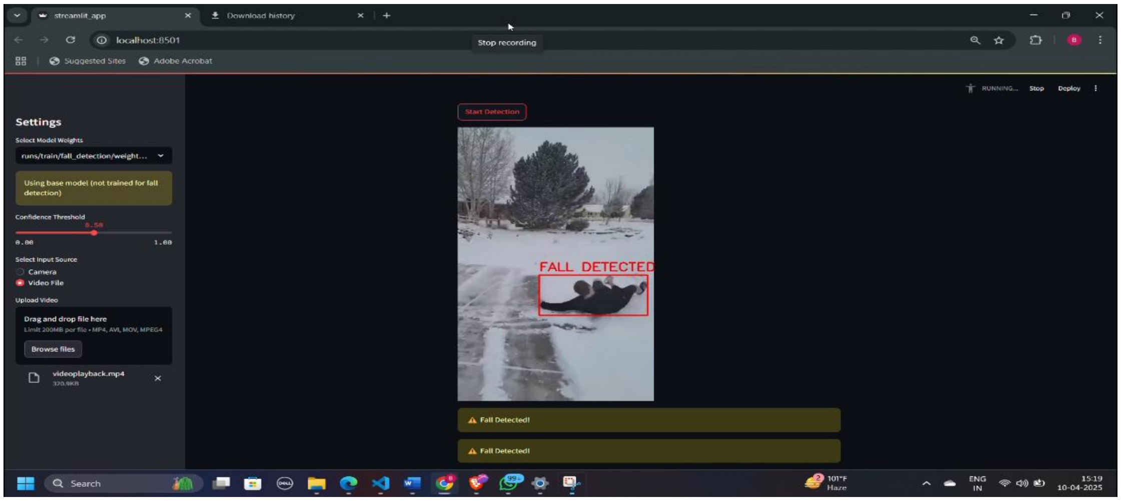
Drag and drop file here

Limit 200MB per file • PNG, JPG, JPEG

mask_1920_1080.png 111.4KB

Empty: 129/396

http://localhost:8503 is sharing your screen.



Settings

Select Model Weights

runs/train/fall_detection/weight...

Using base model (not trained for fall detection)

Confidence Threshold

0.50

0.00 1.00

Select Input Source

○ Camera

● Video File

Upload Video

Drag and drop file here

Limit 200MB per file • MP4, AVI, MOV, MPEG4

videoplayback.mp4 320.9KB

⚠ Fall Detected!

⚠ Fall Detected!

6. Project Organization & Documentation

Setup Guide:

```
python -m venv venv
source venv/bin/activate # Linux/Mac
pip install -r requirements.txt
```

Configuration:

```
PARKING_DETECTOR_FRAME_SKIP = 2
PARKING_DETECTOR_CONFIDENCE_THRESHOLD = 0.7
PARKING_DETECTOR_IOU_THRESHOLD = 0.5
```

Deployment Instructions:

```
# Local Development
python src/main.py --video path/to/video --mask path/to/mask
# Production Setup
streamlit run src/app.py
```

Additional Resources

- GitHub Repository: https://github.com/bhau23/Smart-Parking-Surveillance-AI-Model-
- Demo Videos:
* Parking Detection: validation_results/streamlit-app-2025-04-10-04-04-64.webm
* Fall Detection: FALL_DETECTION/fall_detection_assignment/demo_videos/videoplayback.mp4
- Performance Reports: validation_results/analysis.md

Results and additional visualization files :

https://drive.google.com/drive/folders/1EXkNzdfJpM2mFk2JoD7Krep6T-KE2gLO?usp=sharing