Unlocking Efficiency in Number Plate Recognition via Parallel Computing

Presented by Pavan Kumar Etta

Student -ID: 02179303

Professor: Alfa Heryudono

BACKGROUND

1. Market Growth: The ANPR market is projected to reach $4.8 billion by 2027, with a CAGR of 9.4%, driven by demand in smart cities, tolling, and traffic management.
2. Data Challenges: With 1.4 billion vehicles worldwide, urban traffic cameras generate massive data loads, requiring real-time processing of thousands of images per second.
3. Parallel Processing Benefits: Using GPUs and tools like YOLO, systems can process up to 500+ frames per second, improving speed and accuracy by 30-50%.

Project Overview

This project implements a real-time Number Plate Recognition (NPR) system using OpenCV and Haar Cascade for plate detection and OCR for character extraction. It applies image preprocessing techniques to improve accuracy under varying conditions like motion blur and poor lighting.

01. Objective: Build a real-time Number Plate Recognition (NPR) system using OpenCV and Haar Cascade for vehicle plate detection and recognition.

02. Approach: Leverage Haar Cascade Classifiers for detecting number plates, followed by OCR (Optical Character Recognition) for accurate plate character extraction. Image preprocessing techniques are applied to handle issues like motion blur and poor lighting.

03. Outcome: Achieve real-time processing with high detection accuracy and low latency, suitable for smart city applications, law enforcement, and tolling systems.

Applications and Problems

Applications	Problems
Allows seamless entry for prepaid members in parking.	Broken or damaged number plates.
Automates fee calculation and cross-verifies toll tickets.	Blurry images reducing recognition accuracy.
Monitors vehicle crossings internationally and domestically (border security).	Plates not adhering to legal specifications.
Directs vehicles to lanes based on entry permits (traffic control).	Low-resolution images causing difficulty in character recognition.
Reduces ticket fraud by capturing car images and number plates (airport parking).	Similar-looking characters (e.g., O vs. D) and poorly maintained plates.

Data Preprocessing & Exploration

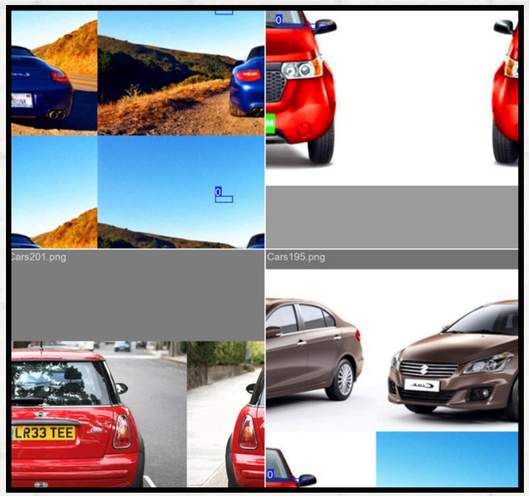
- Image Loading: Images are loaded from the images folder into a list for processing.
- Plate Detection: Haar Cascade is used to detect license plates in each image, converting them to grayscale for processing.
- Performance Comparison: The project compares serial vs. parallel detection using multiprocessing.Pool and measures execution time, speedup, and efficiency.
- Preprocessing & Visualization: Detected plates are highlighted with red rectangles, cropped, and saved. Visualizations show the performance comparison of detection methods.

Data visualization

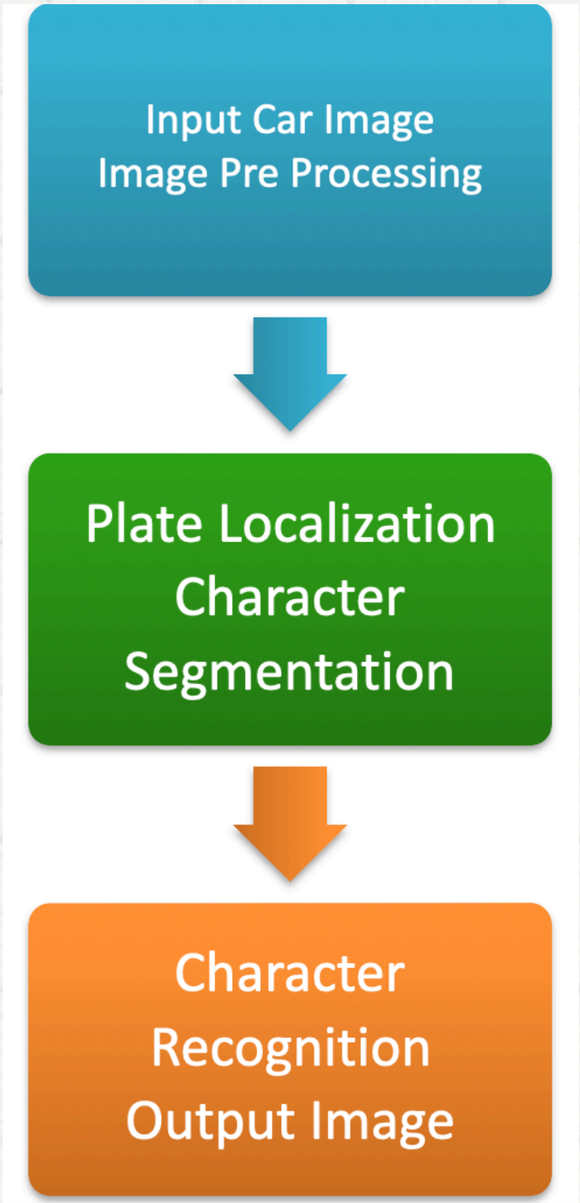
Train

Test

Serial Execution



Parallel execution

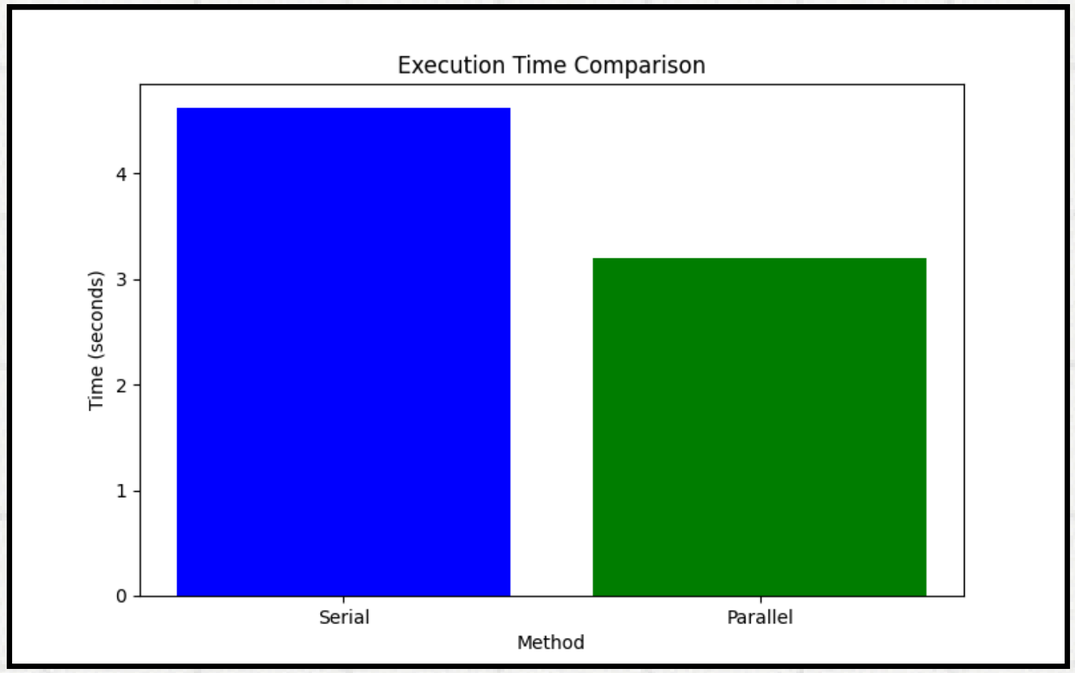


Analysis

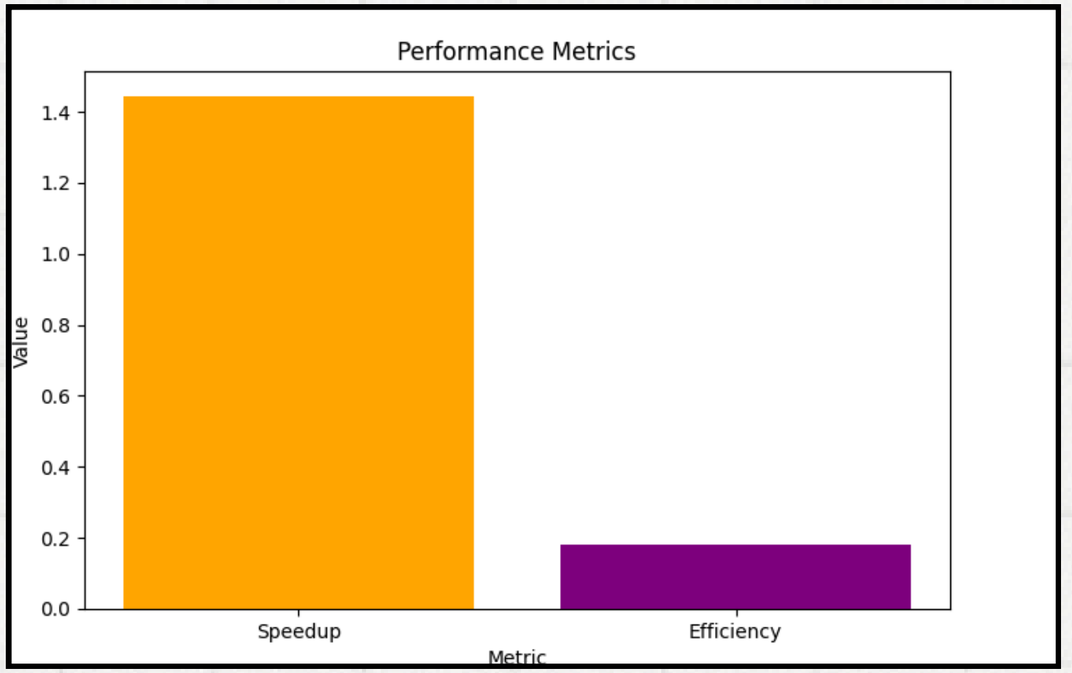
- Parallel Processing Benefits:
- Execution time is significantly reduced with parallel processing, showing clear speedup. Efficiency metrics reveal optimal CPU utilization, but there’s room for further improvement.
- Accuracy:
- Haar Cascade effectively detects plates, though performance can be impacted by motion blur and lighting.
- Optimization:
- Potential for further optimization through classifier fine-tuning or using deep learning methods for better speed and accuracy.
- Application:
- The system is well-suited for real-time traffic monitoring, law enforcement, and tolling systems.

Performance Analysis Graphs

Serial vs Parallel computation

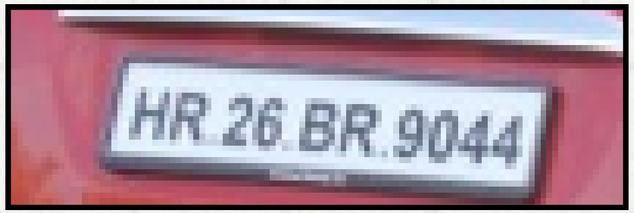


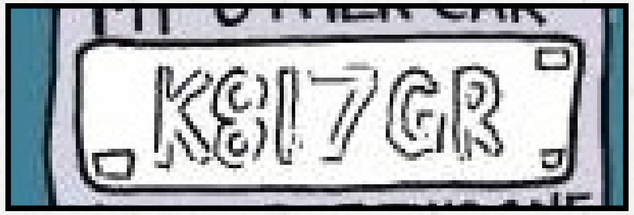
Performance metrics



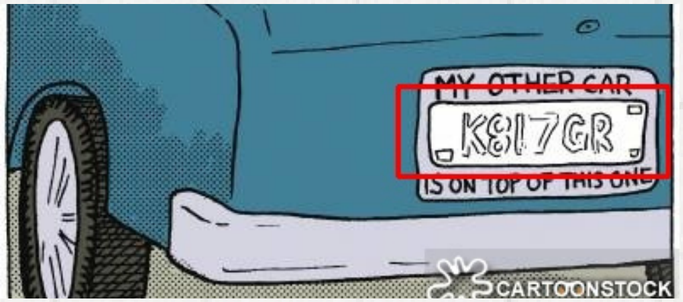
Number plates detection

Cropped



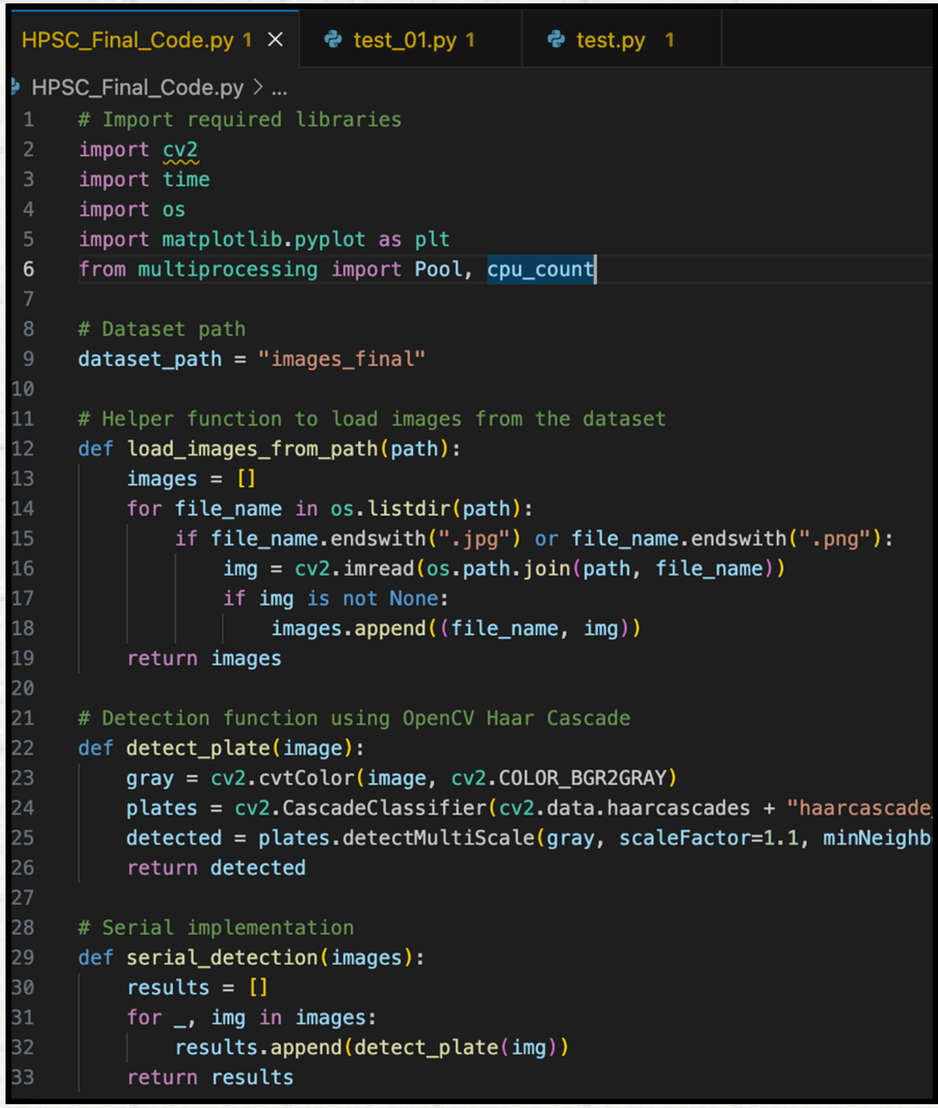


Rectangle rounded



Code part and explanation

- Detects license plates in images using OpenCV's Haar Cascade classifier.
- Compares serial and parallel processing performance.
- Utilizes Python's multiprocessing to leverage all available CPU cores.
- Outputs key performance metrics, including execution time, speedup, and efficiency.
- Provides visualizations to compare execution times and performance metrics.
- Saves annotated images with detected plates and cropped license plate regions.
- Highlights the efficiency of parallel processing in computer vision tasks.



Conclusion and Insights

The project demonstrates efficient real-time Number Plate Recognition using Haar Cascade and parallel processing, achieving significant speedup and reliable detection. It is well-suited for applications like traffic monitoring and tolling systems. Future improvements could focus on integrating deep learning for enhanced accuracy and performance.

References:
https://www.researchgate.net/publication/3427867_Automatic_License_Plate_Recognition
https://www.mdpi.com/1424-8220/24/9/2791

Thank you very much!