# Car License Plate Recognition using YOLOv8 + Streamlit

**Real-time License Plate Detection and OCR Integration**


## Abstract

This report presents a complete system for real-time car license plate detection and recognition using the YOLOv8 model integrated with an interactive Streamlit web interface. The system enables users to upload images or videos, automatically detect license plates, and extract text using Optical Character Recognition (OCR). It supports both EasyOCR and Tesseract for OCR processing and displays real-time results with bounding box visualization and confidence scoring.


## 1. Features

- YOLOv8-powered license plate detection with high accuracy.
- OCR integration using EasyOCR or Tesseract for text extraction.
- Image and video upload support.
- Streamlit web interface for interactive use.
- Real-time processing and bounding box visualization.
- Confidence scoring and text overlay.

## 2. YOLOv8 Overview

YOLOv8 (You Only Look Once version 8) is the latest version of the YOLO object detection family, developed by Ultralytics. It provides a balance between speed and accuracy, ideal for detecting small rectangular objects like vehicle license plates. The model architecture comprises an enhanced backbone (CSPDarknet), an improved neck (PANet), and an anchor-free detection head.

### Key Advantages

- Enhanced backbone for superior feature extraction.
- Anchor-free detection head for faster inference.
- Better small object detection and lighting robustness.
- Multiple variants (n, s, m, l, x) for flexibility in speed vs accuracy.

# 3. Project Structure

The system is modularly designed, comprising model weights, datasets, OCR utilities, detection scripts, Streamlit app, and notebooks for training experiments.

```
models/kbest.pt              { Trained YOLOv8 weights
src/yoloapplication.py       { Streamlit main app
src/detection.py             { Detection pipeline logic
src/ocr_utils.py             { OCR text extraction utilities
notebooks/                   { Training and testing experiments
```

# 4. Model Training & Performance

The YOLOv8n (nano) model was trained for 100 epochs on 345 images (train), validated on 44, and tested on 44. Image resolution was set to 320×320 with batch size 16 using AdamW optimizer and learning rate 0.01.

## Performance Metrics

- mAP@0.5: 91.7%
- mAP@0.5:0.95: 53.9%
- Precision: 86.3%
- Recall: 86.2%
- Inference time: 50 ms (˜23 FPS)

# 5. Proposed Solution

The proposed system integrates YOLOv8 with an OCR engine inside a Streamlit interface. Upon user input (image/video), the YOLOv8 model detects license plates, crops the regions, and forwards them to the OCR module for text extraction. The combined output (bounding boxes + recognized text) is visualized and stored for analysis.

## Pipeline Overview

- Image preprocessing (resize, normalize).
- YOLOv8 inference for bounding box detection.
- Post-processing using Non-Maximum Suppression (NMS).
- OCR extraction using Tesseract or EasyOCR.
- Result overlay with confidence scores and bounding boxes.

# 6. Future Enhancements

- Support for multilingual license plates.
- Real-time webcam and live stream integration.
- Database for storing recognition results.
- Vehicle make/model recognition integration.
- Edge device deployment on Raspberry Pi or Jetson Nano.

# 7. Applications

- Traffic monitoring and law enforcement.
- Automated toll and parking management.
- Smart city surveillance and access control.
- Vehicle tracking in logistics and transport.

# 8. Conclusion

This system effectively demonstrates the potential of YOLOv8 for real-time license plate detection and recognition. The integration of OCR and Streamlit makes it both user-friendly and practical for a wide range of applications such as traffic management, surveillance, and automated toll systems.

# Acknowledgements


- **Ultralytics** – YOLOv8 framework
- **Streamlit** – Interactive web application
- **Tesseract OCR** – Text extraction library
- **OpenCV** – Computer vision utilities