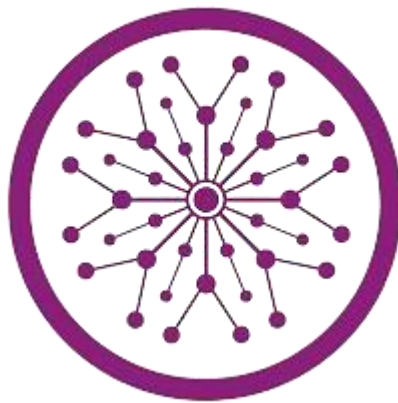


Project Title: License Plate Recognition using YOLOv8

Semester Project Report

Artificial Intelligence



Full Name: Qaswa Chaudhary

Roll No: SU92-BSDSM-F23-002

Section: 3-A

Institution Name: The Superior University

Supervisor's Name: Prof. Rasikh Ali

Submission Date: 02-11-2024

Abstract:

The License Plate Recognition (LPR) System aims to automatically detect and recognize vehicle license plates in real-time using computer vision and machine learning techniques. This system focuses on the detection of vehicles, tracking their movement, recognizing license plates, and extracting the license plate number for further processing. It uses two primary models: a YOLO-based vehicle detection model and an OCR-based license plate recognition system. The YOLO model detects vehicles, and the EasyOCR library is employed to read the license plate numbers. Additionally, a tracking algorithm (SORT) is used to associate license plates with their respective vehicles across multiple frames. The detected license plate text is then validated, formatted, and stored in a CSV file for further analysis. Interpolation techniques are applied to handle missing data when frames are skipped, and the final output is visualized with bounding boxes drawn around the vehicles and their corresponding license plates.

Introduction:

This project involves the development of a License Plate Recognition (LPR) system using machine learning and computer vision techniques. The system detects vehicles in video frames, tracks their movements, and extracts license plate information. The detected license plates are then processed using Optical Character Recognition (OCR) to identify the characters on them. The project's main components include:

- Vehicle Detection & Tracking
- License Plate Detection & Recognition
- Data Interpolation for Missing Frames
- Visualization of Results

Tools and Models used:

Programming Language: python

Version: 3.12

Libraries and Framework:

OpenCV: Used for video processing, image manipulation, and drawing bounding boxes around detected objects.

YOLO (You Only Look Once): A deep learning model used for real-time object detection, specifically for detecting vehicles in the video frames.

- The model used is **YOLOv8n** for vehicle detection.
- The **License Plate Recognition System (LPR)** model based on YOLO is used for license plate detection.

NumPy: Used for numerical operations, especially for array handling and bounding box operations.

Pandas: Used for storing and managing the results in CSV files.

Scipy: Used to interpolate missing bounding box data between frames when tracking data is incomplete.

Sort (Simple Online and Realtime Tracking): A tracking algorithm that associates detected vehicles and their license plates across multiple frames.

ast: Used for safely evaluating strings containing Python literals (e.g., lists, tuples).

csv: Used for reading and writing CSV files.

string: Used for accessing predefined sets like uppercase alphabets (string.ascii_uppercase).

easyocr: A library for Optical Character Recognition (OCR) that extracts text from images.

Models:

YOLOv8n Model: This is used for vehicle detection, where it identifies different classes of vehicles (car, bike, bus, truck) and generates bounding boxes around them.

License Plate Recognition Model: Another YOLO-based model fine-tuned for recognizing license plates in vehicle images. This model detects the bounding box of the license plate in each vehicle frame.

EasyOCR: For extracting text from license plates using OCR after detecting the bounding box.

Project Files Overview:

- **main.py:** The main script that coordinates the detection of vehicles, license plates, and tracking using the YOLO model.
- **util.py:** Utility functions for handling license plate recognition and processing.
- **add_missing_data.py:** A script for interpolating missing bounding box data for vehicles and license plates in the video frames.
- **visualize.py:** A script for visualizing the results, drawing bounding boxes on the video frames, and displaying license plate information.

Data Processing:

- The system processes video frames in real-time, detecting and tracking vehicles across frames. It uses the SORT algorithm to track vehicles and associate detected license plates with their corresponding vehicles.
- Interpolation is used to fill in missing data points when some frames lack detection or when vehicles are temporarily occluded.

Output:

- The system writes the results to a CSV file, storing information about the frame number, vehicle IDs, bounding boxes for vehicles and license plates, license plate numbers, and associated scores.
- The processed video with overlaid bounding boxes and license plate information is saved as a new video file.



Here is an example image of model detecting car and number plate at a time and also displaying the car number plate right above it.

Conclusion:

The License Plate Recognition System effectively detects vehicles and reads license plates in a video stream. By leveraging YOLO for object detection and OCR for text recognition, the system provides a comprehensive solution for automatic vehicle and license plate identification. The project also addresses missing data by interpolating bounding box positions for frames with incomplete tracking information, ensuring a smooth and continuous tracking process. Finally, the results are visualized and saved in an output video for easy review.

Google drive link for sample and output videos

[Sample and output video links](#)