

BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



8th Report

Foundation Internship

Project Title: Traffic License Recognition

Instructor: Kim Ngoc Bach

Student Name : Bui Xuan Hai

Student ID : B22DCAT105

Lớp : E22CQCN05-B

Hà Nội - 2025

INTERNSHIP BASE REPORT

I. Project Introduction

This project aims to develop a web application that supports automatic recognition of vehicle license plates from uploaded images or videos. The system uses a deep learning model (trained by myself) to detect and recognize license plate numbers. This can be applied in real-life scenarios such as entry/exit management in parking lots or garages.

Users can interact with the system via a web interface, upload photos or videos, and the system will display the detected license plates. The backend is implemented using Flask, while the frontend is designed using HTML, CSS, and JavaScript (with optional integration of Bootstrap or Tailwind CSS).

II. Key Features

- Allow users to upload vehicle images or videos.
- Automatically detect and recognize license plate numbers.
- Display recognition results in a user-friendly format.
- Possibility to store recognition history or export results.
- Designed for application in garage or parking management systems.

III. Technologies Used

- **Frontend:** HTML, CSS, JavaScript (Bootstrap/Tailwind CSS).
- **Backend:** FastAPI (Python).
- **AI Model:** YOLO-based license plate detection + OCR model (e.g., PaddleOCR or custom model).
- **Storage:** Local server or database (SQLite/MySQL)

IV. Week 8

1. Weekly goals

- Optimize image processing performance and system response speed
- Test the YOLOv5 model with lighter variants
- Test and improve the overall accuracy of the license plate recognition pipeline
- Start preparing for the online deployment phase of the system

2. Work done

- Optimize input images
- On the frontend, resize images to a fixed size before uploading
- Test the YOLOv5n (nano) model
- Compare performance between YOLOv5s (small) and YOLOv5n (nano)
- With clear images, YOLOv5n still gives stable recognition results → suitable for resource-limited environments (such as free hosting)
- Improve the combined recognition logic
- If there are many license plate regions detected → prioritize the region with the highest confidence
- Add error checking to the return strings from OCR: remove invalid characters, check the license plate format valid numbers (eg: 29A-xxxxxx)
- Evaluate accuracy and performance

3. Results achieved

- The processing system is faster and more stable than previous weeks
- The pipeline is capable of real-time recognition (pseudo-real-time) with good quality images
- It is possible to flexibly use lighter models while maintaining acceptable performance

4. Difficulties encountered

- Some strings returned from PaddleOCR are incorrect due to blurry images, too small or distorted license plates
- Not all license plates follow the common standard (eg: foreign vehicles, old motorbike license plates)
- Need to write more logic to process and filter the results after OCR to avoid returning invalid data