

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



6th 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 6

1. Weekly goals

- Allow users to upload photos from the web interface
- Send photos to the server via API and display license plate recognition results directly on the browser

2. Work done

- Design the user interface (Frontend)
- A simple interface with main parts:
- "Recognize license plate" button to call the API
- The area displaying the result image and the recognized license plate string
- Process uploads and call the API using JavaScript
- After receiving a response from the server, the image will be displayed again with the license plate caption (if any)
- If the server returns an error (not recognized), a clear error message will be displayed

3. Results achieved

- The user interface works well, is easy to use and intuitive
- Successfully connected the frontend and backend via the FastAPI API

4. Difficulties encountered

- Not using a framework, so uploading, fetching, DOM manipulation must be handled entirely with pure JS → time consuming

- The interface is still simple, not handling many advanced cases such as previewing images, drag & drop