BỘ THÔNG TIN VÀ TRUYỀN THÔNG

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

4th 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**: Flask (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 4

1. **Weekly goals**

- Integrate PaddleOCR to read characters from detected license plate areas
- Write a pipeline to process cropped images from YOLOv5 and input them into OCR
- Display character recognition results and evaluate accuracy

2. **Work done**

- Install and configure PaddleOCR
- Use paddleocr==2.6.1 with lightweight paddlepaddle GPU
- Download available recognition models that support Vietnamese and English
- Write an OCR wrapper to process input as Numpy images or cropped images from OpenCV
- Create a pipeline to crop images from detected areas and input them into OCR
- Use bounding box coordinates from YOLOv5 to crop areas containing license plates
- Convert images to RGB format if the input image is in the wrong mode
- Pass images through OCR and get character strings with confidence
- Display recognition results on input images
- Draw bounding boxes and overwrite character strings on output images

- Display confidence at the same time for convenience quality assessment
- Save the results as new images and log them to the results.csv file
- Test the system with real images

3. **Results achieved**

- PaddleOCR successfully integrated into the system
- Recognize clear characters in the license plate area
- Pipeline YOLOv5 → crop → OCR works stably
- Record the results as images and statistical tables

4. **Difficulties encountered**

- PaddleOCR's accuracy is reduced with tilted, overexposed or too blurry images
- Some images after cropping are too small → the image is distorted, difficult to read
- OCR sometimes returns incorrect characters such as "B" as "8", "O" as "0"