



TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO
PBL5 - ĐỒ ÁN KỸ THUẬT MÁY TÍNH

QUẢN LÍ BÃI GIỮ XE

Giảng viên đồng hướng dẫn: TS. Trần Thế Vũ

STT NHÓM: 38 HỌ VÀ TÊN SINH VIÊN	LỚP HỌC PHẦN ĐỒ ÁN
Lê Việt Hưng	20N15A
Giáp Ngọc Hiệu	20N15A
Phạm Huỳnh Nhật Tân	20N15A
Lê Tự Minh Tuấn	20N15A

ĐÀ NẴNG, 06/2021

TÓM TẮT ĐOÁN

Trong quá trình tham khảo và nghiên cứu, nhóm nhận thấy rằng có việc quản lý bãi giữ xe một số nơi chưa được tối ưu và hiện đại. Do đó, nhóm chúng em đề xuất đề tài “Quản lý bãi giữ xe” với mong muốn có thể tối ưu nguồn nhân lực và công sức của nhân viên và quản lý. Nhóm đã sử dụng công nghệ AI trong việc nhận diện và đọc biển số xe, sử dụng máy tính và mạch Arduino cho việc xử lý các tác vụ chụp ảnh, nhận diện, đóng mở cửa bãi, quản lý với ứng dụng desktop. Sau khi thử nghiệm và kiểm thử thì ứng dụng và hệ thống hoạt động tốt. Tuy nhiên, vẫn còn có một số nhược điểm cần được cải thiện và một số hạn chế cần được mở rộng. Nhóm sẽ cố gắng phát triển và đạt được kết quả và hiệu suất tốt hơn

BẢNG PHÂN CÔNG NHIỆM VỤ

Thành viên	Nhiệm vụ	Tự đánh giá
Giáp Ngọc Hiệu	Huấn luyện mô hình YOLOv8 phát hiện biển số	Đã hoàn thành
	Huấn luyện mô hình YOLOv8 đọc biển số	Đã hoàn thành
	Viết báo cáo	Đã hoàn thành
	Làm slide	
Lê Việt Hưng	Mô hình Cascade và CNN để phát hiện và đọc biển số	Đã hoàn thành
	Trực quan hóa dữ liệu bãi đỗ xe	Đã hoàn thành
	Viết báo cáo	Đã hoàn thành
	Mô hình dự đoán doanh thu	Đã hoàn thành
Phạm Huỳnh Nhật Tân	Làm ứng dụng desktop và Firebase	Đã hoàn thành
	Trực quan hóa dữ liệu bãi đỗ xe	Đã hoàn thành
	Viết báo cáo	Đã hoàn thành
	Làm slide	Đã hoàn thành
Lê Tự Minh Tuấn	Làm phần cứng, giao tiếp phần mềm và phần cứng	Đã hoàn thành
	Làm các hàm chức năng trong hệ thống	Đã hoàn thành
	Viết báo cáo	Đã hoàn thành
	Làm slide	Đã hoàn thành

MỤC LỤC

1. Giới thiệu.....	7
2. Giải pháp.....	8
2.1. Giải pháp về phần cứng và phần mềm.....	8
2.1.1. Tổng quan cách hoạt động của hệ thống:.....	8
2.1.2. Sơ đồ hoạt động của các chức năng.....	8
2.2. Giải pháp về dự đoán doanh thu.....	12
2.3. Giải pháp về nhận dạng.....	13
3. Kết quả.....	14
3.1. Kết quả mô hình nhận dạng.....	14
3.1.1. Kết quả nhận dạng khi sử dụng Cascade và CNN.....	14
3.1.2. Kết quả nhận dạng khi sử dụng YOLOv8.....	18
3.1.3. So sánh 2 phương pháp.....	22
3.2. Kết quả mô hình dự đoán doanh thu.....	22
3.3. Kết quả chương trình.....	23
4. Kết luận.....	28
4.1. Kết luận.....	28
4.2. Hướng phát triển.....	29
5. Danh mục tài liệu tham khảo.....	29

DANH MỤC HÌNH VẼ

Hình 2.1. Sơ đồ tổng quan cách hoạt động của chương trình.....	8
Hình 2.2. Sơ đồ hoạt động trong quá trình xe vào.....	8
Hình 2.3. Sơ đồ hoạt động lúc có xe ra.....	9
Hình 2.4. Cơ sở dữ liệu của ứng dụng.....	10
Hình 2.5. Mô hình RNN.....	12
Hình 2.6. Minh họa kiến trúc của mô hình RNN.....	13
Hình 2.7. Kiến trúc mô hình RNN Many to one.....	13
Hình 2.8. Quá trình nhận dạng và đưa ra kết quả của biến số.....	14
Hình 2.9. Kết quả phát hiện vị trí kí tự.....	14
Hình 3.1. Tập dữ liệu kiểm thử tự chụp.....	14
Hình 3.2. Phương pháp sử dụng Cascade trong nhận dạng.....	15
Hình 3.3. Kết quả phát hiện biến số trên mô hình Cascade.....	15
Hình 3.4. Các lớp CNN trong mô hình phát hiện kí tự.....	16
Hình 3.5. Mô hình phân loại kí tự.....	16
Hình 3.6. Kết quả của mô hình Cascade và CNN.....	16
Hình 3.7. Các kết quả nhận dạng sai và lỗi trong mô hình.....	17
Hình 3.8. Một số kết quả nhận dạng đúng.....	17
Hình 3.8. Số lượng mẫu của các class trong tập train và tập validation.....	18
Hình 3.9. Kết quả mAP50 với mô hình phát hiện biến số.....	19
Hình 3.10. Kết quả mAP - 95 với mô hình phát hiện biến số.....	20
Hình 3.11. Kết quả mAP – 50 với mô hình nhận diện kí tự.....	20
Hình 3.12. Kết quả mAP – 95 với mô hình nhận dạng kí tự.....	21
Hình 3.13. Kết quả sau khi kết hợp 2 mô hình YOLOv8.....	21
Hình 3.14. Đồ thị hàm loss của mô hình RNN.....	22
Hình 3.15. Kết quả dự đoán doanh thu của tháng tiếp theo.....	23

Hình 3.16. Giao diện đăng nhập.....	23
Hình 3.17. Giao diện chính của quản lí vào ra.....	23
Hình 3.18. Giao diện đăng ký thông tin xe khi chưa đăng ký.....	24
Hình 3.19. Giao diện của khi hoàn thành thủ tục đăng ký.....	24
Hình 3.20. Giao diện quản lí bãi đỗ xe.....	24
Hình 3.21. Giao diện quản lí tài khoản đăng nhập.....	25
Hình 3.22. Giao diện quản lí giá cả của các khoảng thời gian đỗ xe.....	25
Hình 3.23. Giao diện quản lí xe và thời gian ra vào bãi.....	26
Hình 3.24. Giao diện quản lí doanh thu.....	26
Hình 3.25. Giao diện biểu thị doanh thu của từng tháng trong các năm.....	26
Hình 3.26. Giao diện doanh thu của 6 tháng gần nhất.....	27
Hình 3.27. Giao diện dự đoán doanh thu.....	27
Hình 3.28. Giao diện top 5 khách hàng.....	27
Hình 3.29. Mô hình mô phỏng bãi đỗ xe.....	28
Hình 3.30. Mô hình mô phỏng bãi đỗ xe (tt).....	28

1. Giới thiệu

Hiện nay ở thời đại 4.0, con người dần tiếp cận và sử dụng các công nghệ của nghành công nghệ thông tin ngày càng phổ biến rộng rãi hơn, đặc biệt là trong chuyên nghành Trí tuệ nhân tạo. Trong đó, việc quản lý các bãi giữ xe đã có một số sự kết hợp giữa giữ xe truyền thống và áp dụng các công nghệ, phần mềm để quản lý. Nhóm chúng em thực hiện đề tài này nhằm áp dụng AI trong việc nhận diện và đọc biển số xe một cách tự động. Trong đề tài này, nhóm tập trung nhiều vào việc quản lý các bãi giữ xe ô tô trong các bãi đỗ xe theo thời gian.

Các vấn đề mà nhóm cần giải quyết là:

- Nhận diện và đọc biển số xe lúc vào và ra bãi đỗ xe bằng AI.
- Quản lý thông tin của xe, chủ xe và thời gian ra vào bãi giữ xe.
- Quản lý doanh thu của bãi giữ xe theo ngày, tuần, tháng, quý hoặc năm.
- Dự đoán doanh thu của bãi giữ xe trong thời gian tiếp theo.

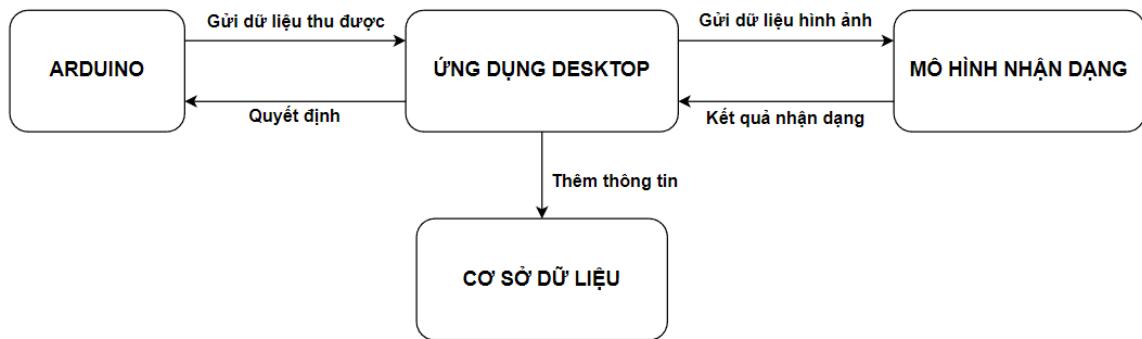
Giải pháp cho các vấn đề trên:

- Sử dụng 2 mô hình YOLOv8 để nhận diện biển số trong ảnh và đọc biển số từ ảnh thành chữ.
- Xây dựng ứng dụng desktop bằng tkinter để quản lý doanh thu, thông tin của xe và chủ xe, thời gian đỗ xe của các xe.
- Dự đoán doanh thu cho các tháng, quý, năm tiếp theo bằng mô hình.
- Làm mô hình mô phỏng quá trình quản lý xe, quản lý xe ra vào bãi bằng Arduino và các thiết bị.

2. Giải pháp

2.1. Giải pháp về phần cứng và phần mềm

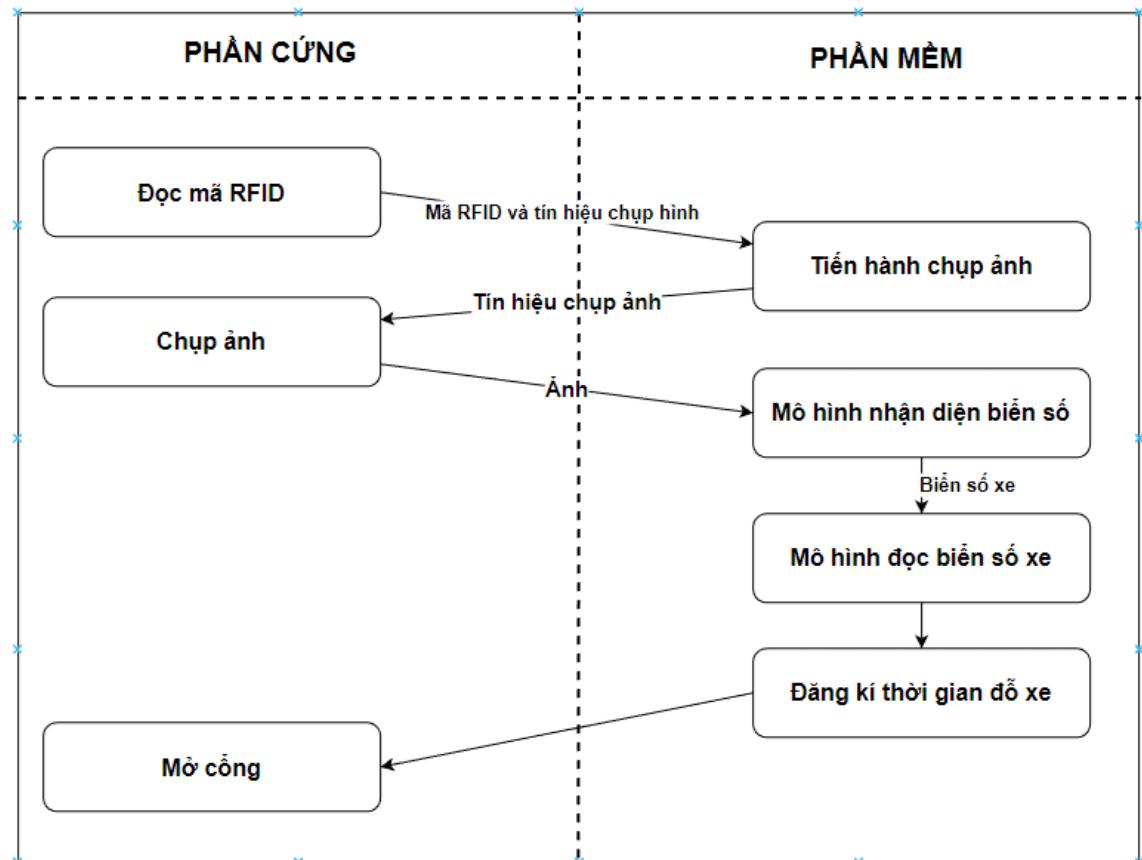
2.1.1. Tổng quan cách hoạt động của hệ thống:



Hình 2.1. Sơ đồ tổng quan cách hoạt động của chương trình

Hệ thống bao gồm mạch Arduino mega để thu và gửi tín hiệu qua ứng dụng bên desktop để xử lý. Ứng dụng desktop xử lí sẽ nhận dữ liệu từ Arduino, chụp ảnh và đưa qua mô hình để nhận diện và từ kết quả sẽ đưa ra quyết định tiếp theo và gửi lại tín hiệu

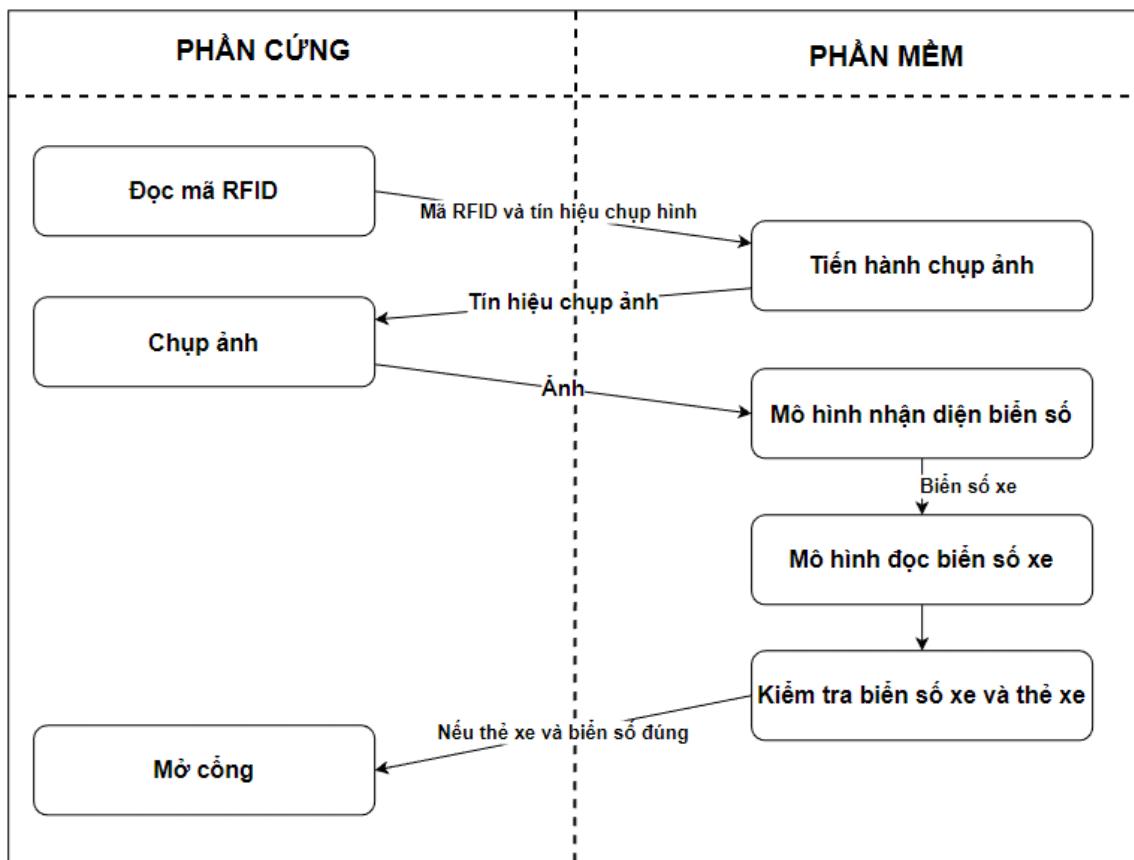
2.1.2. Sơ đồ hoạt động của các chức năng



Hình 2.2. Sơ đồ hoạt động trong quá trình xe vào

Quá trình hoạt động khi xe vào: sau khi có xe vào, nhân viên sẽ quét mã của thẻ RFID để tiến hành đăng kí theo các bước sau:

- Bước 1: khi đọc được thẻ thành công, Arduino sẽ gửi tín hiệu cho ứng dụng biết có xe vào
- Bước 2: đọc hình ảnh của camera, đưa qua mô hình nhận dạng và đọc biển số
- Bước 3: Tiến hành đăng kí thông tin cá nhân, khoảng thời gian gửi xe, tính tổng tiền. Đối với các xe đã từng sử dụng bãi giữ xe thì không cần đăng kí thông tin mà chỉ cần đăng kí khoảng thời gian gửi và thanh toán.
- Bước 4: Sau khi nhập thông tin đầy đủ thì thêm vào cơ sở dữ liệu bao gồm các thông tin
 - o Thông tin của khách hàng
 - o Thông tin thời gian gửi, tổng tiền
 - o Thông tin về biển số xe và mã của thẻ RFID tương ứng



Hình 2.3. Sơ đồ hoạt động lúc có xe ra

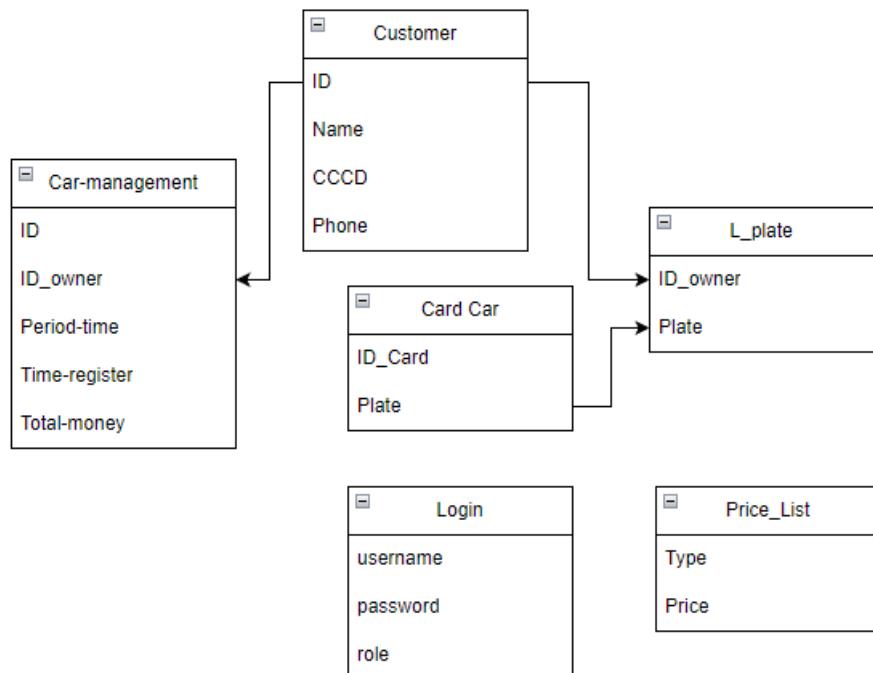
Quá trình hoạt động khi xe ra khỏi bãi giữ xe: sau khi quét mã thẻ RFID, hệ thống sẽ hoạt động theo các bước sau:

- Bước 1: khi đọc được thẻ thành công, Arduino sẽ gửi tín hiệu cho ứng dụng biết có xe ra khỏi bãi
- Bước 2: đọc hình ảnh từ camera và đưa qua mô hình nhận dạng và đọc biển số
- Bước 3: kiểm tra mã thẻ và biển số có trùng khớp với nhau hay không, nếu không trùng khớp thì cần kiểm tra lại thông tin thẻ hoặc biển số

Khi ra khỏi bãi giữ xe, có 3 trường hợp sau:

- Xe lấy sớm hơn thời gian đăng ký: có thẻ vào lại bãi gửi xe
- Xe lấy quá giờ đăng ký: ra bãi đỗ xe và trả thêm phí phạt
- Xe lấy đúng giờ: ra bãi đỗ xe

Ngoài 2 quá trình hoạt động trên, còn có chức năng quản lý bãi đỗ xe, doanh thu của từng ngày, tháng, ... và dự đoán doanh thu của tháng tiếp theo trên ứng dụng desktop. Nhóm xây dựng ứng dụng desktop có giao diện với tkinter trong Python giúp người dùng có thể dễ dàng tương tác với các thông tin, dữ liệu. Về phần dữ liệu, nhóm lưu trữ trên Firebase để có thể truy xuất và cập nhập giữa các máy một cách dễ dàng



Hình 2.4. Cơ sở dữ liệu của ứng dụng

Việc quản lý doanh thu cần phải đăng nhập dưới quyền admin và bao gồm các việc sau:

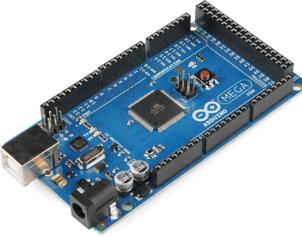
- Quản lý doanh thu của cửa hàng theo từng ngày, tháng, năm, ...
- Quản lý thông tin các nhân viên của khách hàng

- Quan sát, thống kê được tình trạng ra vào bãi đỗ xe và dự đoán được doanh thu của các tháng sau.
- Quản lý thông tin về tài khoản đăng nhập, các thông tin về giá cả của các khoảng thời gian giữ xe

Về phần cứng của đề tài, nhóm xây dựng ứng dụng desktop có giao diện với thư viện tkinter trong Python và sử dụng Firebase để lưu trữ dữ liệu và truy xuất dữ liệu lúc cần sử dụng.

Về phần cứng của đề tài này, nhóm sử dụng chủ yếu 2 webcam để lấy dữ liệu hình ảnh từ webcam và đưa qua ứng dụng và đưa vào mô hình nhận dạng và đưa ra kết quả. Ngoài ra, nhóm có làm mô hình minh họa quá trình thực tế của mô hình quản lý bãi giữ xe với các linh kiện như module RFID RC522, RFID card, servo, ...

Bảng 2.1. Thông số của các linh kiện

Linh kiện	Hình ảnh	Thông số kỹ thuật
Arduino Mega 2560	 A photograph of an Arduino Mega 2560 microcontroller board. It is a blue PCB with a central ATmega2560 chip, various pins, and a USB port.	Ví điều khiển: AVR ATmega2560 8bit Nguồn: 7 – 12V Chân I/O số: 54 chân Chân I/O tương tự: 16 chân Xung clock: 16 MHz Bộ nhớ flash: 128 KB SRAM: 8KB PWM: 12 chân (2 – 16 bit) USART: 4 Giao tiếp: USB (Lập trình với ATmega8), ICSP, SPI, I2C và USART
Webcam Rapoo C200	 A photograph of a Rapoo C200 USB Webcam. It is a black device with a built-in microphone and a flexible gooseneck mount.	Độ phân giải: HD 720p Tốc độ khung hình: 30fps Cổng giao tiếp: USB 2.0 Có hỗ trợ mic thu âm

Webcam Dahua z2		Độ phân giải: HD 720p Tốc độ khung hình: 30fps Cổng giao tiếp: USB 2.0 Có hỗ trợ mic thu âm
--------------------	---	--

Bảng 2.2. Thông số của các linh kiện

Linh kiện	Chi phí	Ghi chú
Arduino Mega 2560	315.000	Mua
Webcam Rapoo C200	390.000	Mua
Webcam Dahua Z2	290.000	Mua
	995.000	

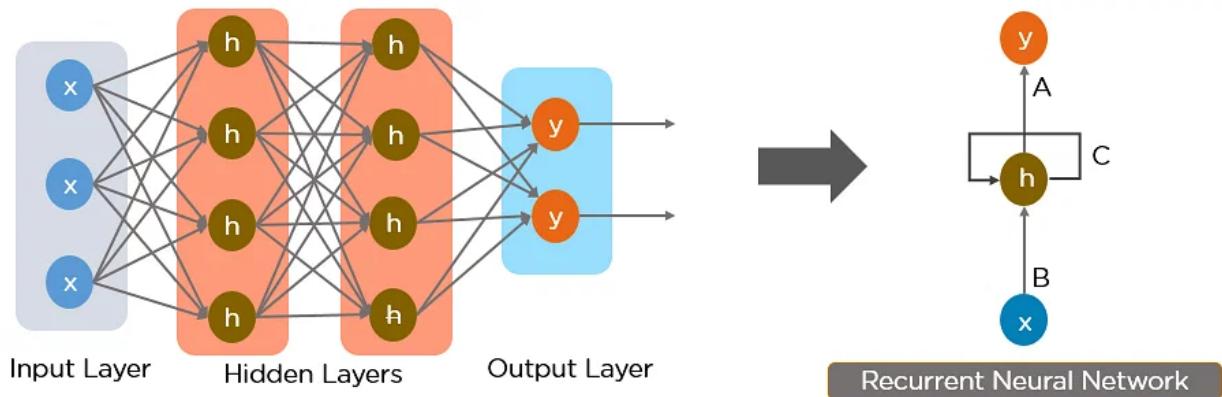
2.2. Giải pháp về dự đoán doanh thu

Trong đồ án lần này, nhóm chúng em mong muốn áp dụng kiến thức bộ môn Khoa Học Dữ Liệu vào nên đưa ra thêm một tính năng mới cho chương trình là dự đoán về doanh thu của cửa hàng. Trong đó có các vấn đề sau:

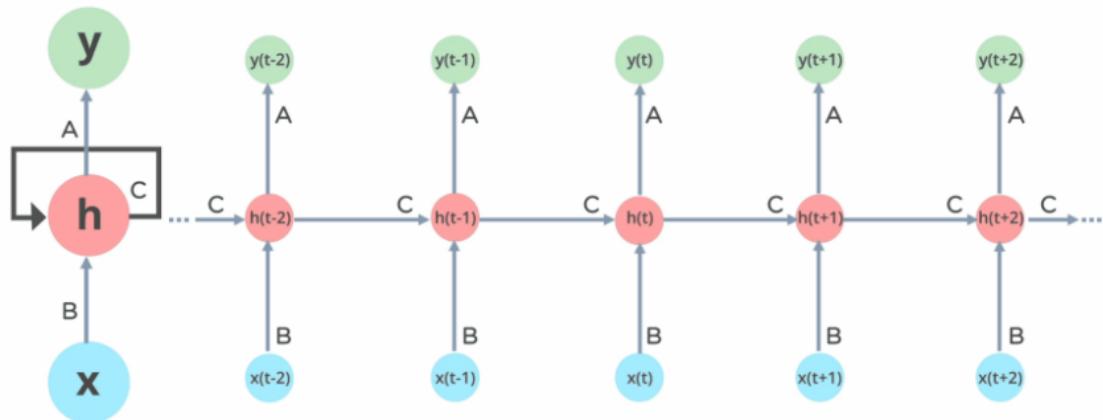
- Trực quan hóa dữ liệu doanh thu của bãi đỗ xe theo các khoảng, mốc thời gian. Từ đó giúp người quản lý có thể đưa ra các nhận xét về lưu lượng xe, mật độ các xe xuất hiện theo các khoảng thời gian và đưa ra các giải pháp hợp lý
- Dự đoán doanh thu của các khoảng thời gian tiếp theo giúp đưa ra các định hướng và phương pháp xử lí và cải thiện phù hợp với bãi đỗ xe cũng như đưa ra chỉ tiêu cho các nhân sự thực hiện

Các giải pháp nhóm đề ra là:

- Trực quan hóa dữ liệu về doanh thu trên các biểu đồ đã được học
- Sử dụng mô hình RNN trong việc dự đoán doanh thu của các tháng tiếp theo

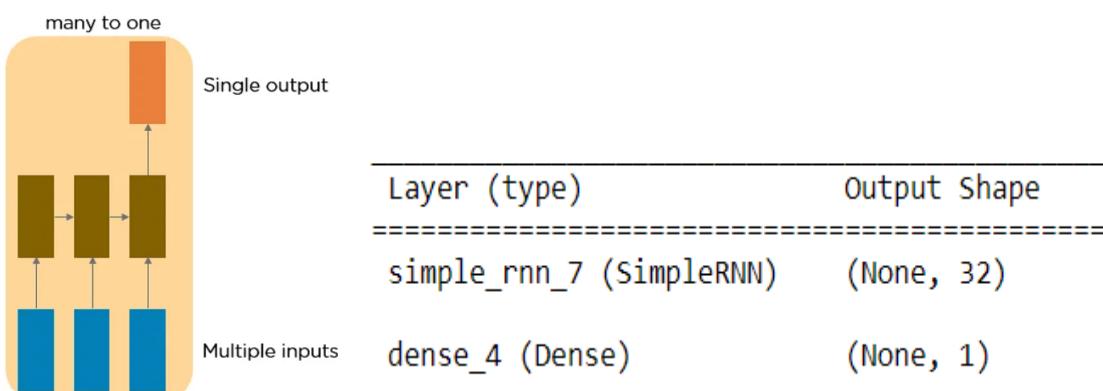


Hình 2.5. Mô hình RNN



Hình 2.6. Minh họa kiến trúc của mô hình RNN

Sử dụng mô hình RNN Many to One: nhận vào một chuỗi đầu vào dữ liệu và tạo ra một đầu ra duy nhất với đầu vào trong đề tài lần này là doanh thu của 9 tháng trước đó và đầu ra sẽ là doanh thu của tháng dự đoán



Hình 2.7. Kiến trúc mô hình RNN Many to one

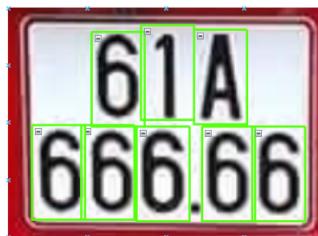
2.3. Giải pháp về nhận dạng

Quá trình nhận dạng tổng quát được nhóm đưa ra và thực hiện theo các bước sau:

- Bước 1: lấy dữ liệu ảnh được chụp từ Camera đưa qua mô hình thứ nhất để xác gán nhãn và lấy được vị trí của biển số trong ảnh đó. Từ đó thu được ảnh sau khi được cắt của biển số xe
- Bước 2: đưa ảnh biển số xe vừa được cắt qua mô hình thứ hai để xác định vị trí của các kí tự và phân loại các kí tự trên biển số xe. Cuối cùng thu được kết quả là chuỗi kí tự trên biển số xe



Hình 2.8. Quá trình nhận dạng và đưa ra kết quả của biển số



Hình 2.9. Kết quả phát hiện vị trí kí tự

Về công cụ thực hiện quá trình nhận dạng, nhóm sử dụng 2 mô hình sau để nhận dạng:

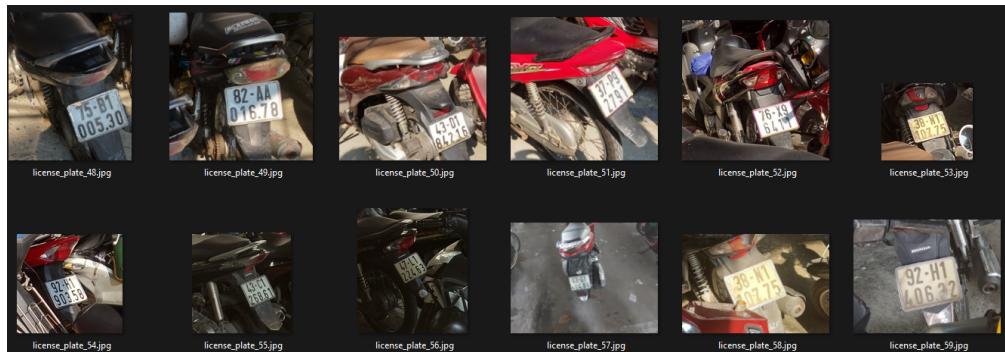
- Xác định vị trí của biển số xe: trong đề tài lần này, nhóm sử dụng YOLOv8 để xác định vị trí của biển số xe trong khung ảnh. Ngoài ra, nhóm còn thử nghiệm việc xác định vị trí của biển số xe thông bằng mô hình Cascade để xác định vị trí.
- Xác định các kí tự trên biển số xe: để có được kết quả trên biển số xe, nhóm sử dụng YOLOv8 để xác định vị trí và gán nhãn của các kí tự trên biển số xe trong ảnh đã được cắt ra từ quá trình xác định vị trí của biển số xe. Ngoài ra, nhóm có thực hiện thử nghiệm với mô hình CNN để đọc các kí tự trên biển số.

3. Kết quả

3.1. Kết quả mô hình nhận dạng

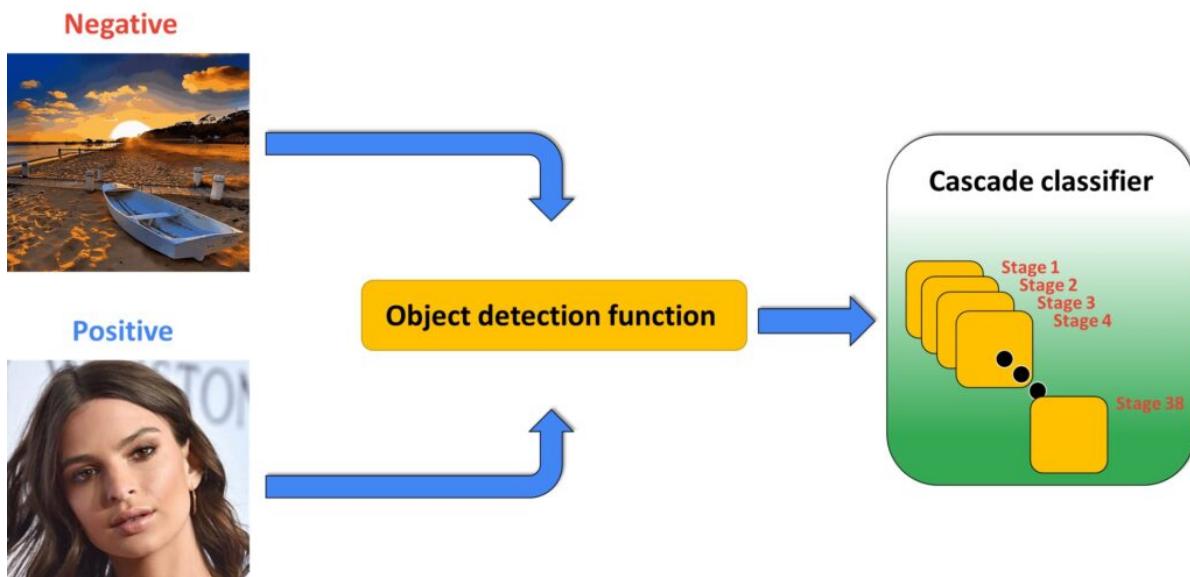
3.1.1. Kết quả nhận dạng khi sử dụng Cascade và CNN

- Tập dữ liệu được thu thập và tổng hợp từ việc chụp hình trong bãi đậu xe của Trường đại học Bách khoa do nhóm tự chụp



Hình 3.1. Tập dữ liệu kiểm thử tự chụp

- Mô hình phát hiện: sử dụng Cascade để phát hiện biển số trong bức ảnh:
 - Phương pháp sử dụng Cascade Classifier là một phương pháp dựa trên học máy, hàm cascade này được train dựa trên các ảnh positive (ảnh có biển số xe) và ảnh negative (ảnh không bao gồm biển số xe)



Hình 3.2. Phương pháp sử dụng Cascade trong nhận dạng

- Phương pháp này dựa trên các đặc trưng cạnh và cấu trúc, kích thước của 1 biển số xe để phát hiện 1 khung biển số xe trong 1 bức ảnh. Sử dụng file cấu trúc (.xml) biển số xe Việt Nam đã được train sẵn để detect ảnh



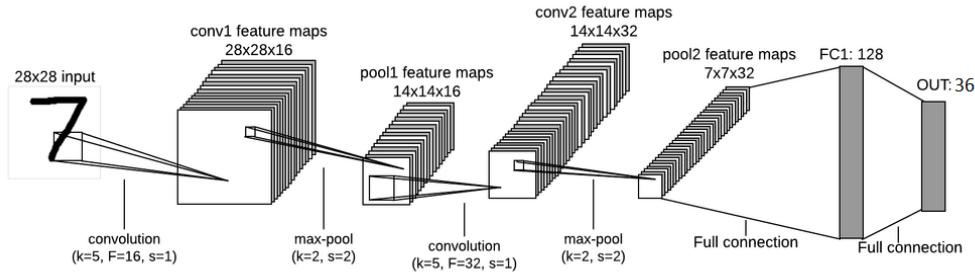
Hình 3.3. Kết quả phát hiện biển số trên mô hình Cascade

- Mô hình phát hiện các kí tự trên biển số xe bằng CNN với các lớp như hình 3.4.

Model: "sequential"		
Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 28, 28, 16)	23248
conv2d_1 (Conv2D)	(None, 28, 28, 32)	131104
conv2d_2 (Conv2D)	(None, 28, 28, 64)	131136
conv2d_3 (Conv2D)	(None, 28, 28, 64)	65600
max_pooling2d (MaxPooling2D)	(None, 7, 7, 64)	0
dropout (Dropout)	(None, 7, 7, 64)	0
flatten (Flatten)	(None, 3136)	0
dense (Dense)	(None, 128)	401536
dense_1 (Dense)	(None, 36)	4644

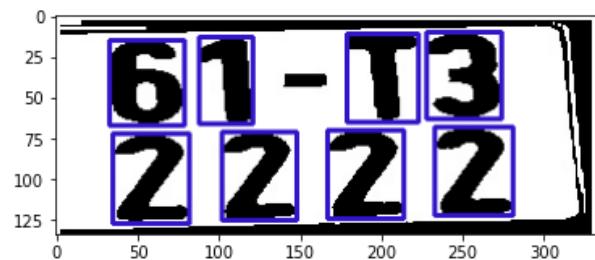
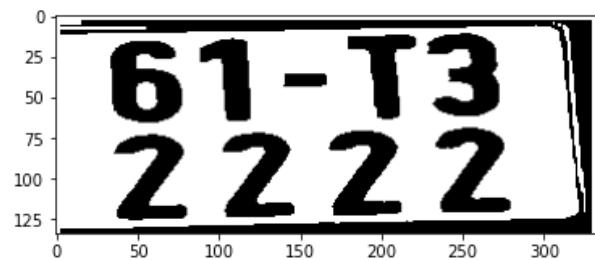
Total params: 757,268
Trainable params: 757,268
Non-trainable params: 0

Hình 3.4. Các lớp CNN trong mô hình phát hiện kí tự



Hình 3.5. Mô hình phân loại kí tự

- Kết quả sau khi trải qua 2 quá trình là phát hiện biển số bằng Cascade và phát hiện kí tự trên biển số bằng CNN thì ta được kết quả sau



61T3222

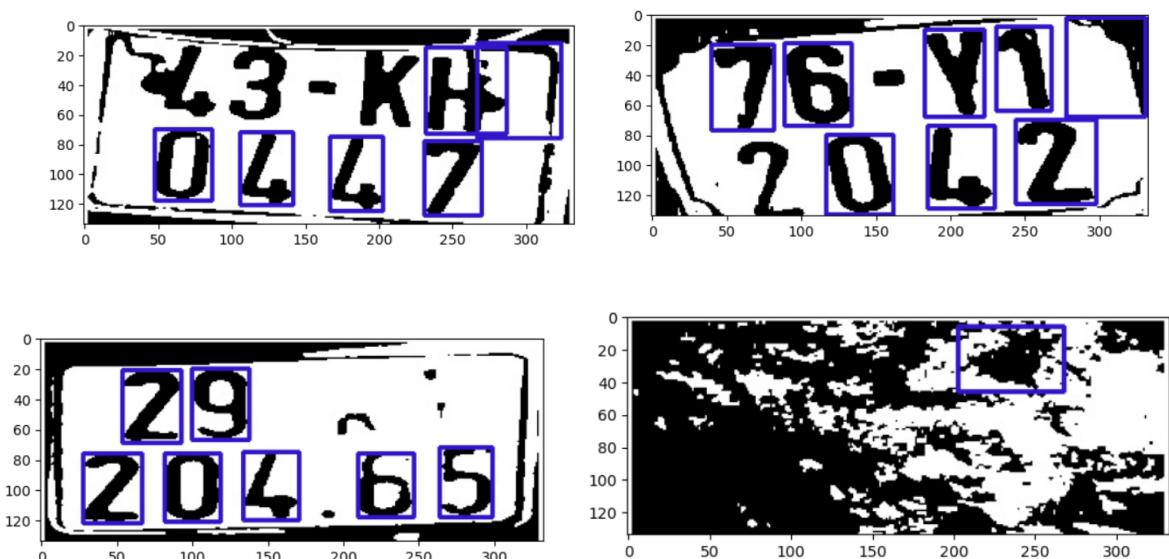
predicted: 6	predicted: 1	predicted: T	predicted: 3
predicted: 2	predicted: 2	predicted: 2	predicted: 2

Hình 3.6. Kết quả của mô hình Cascade và CNN

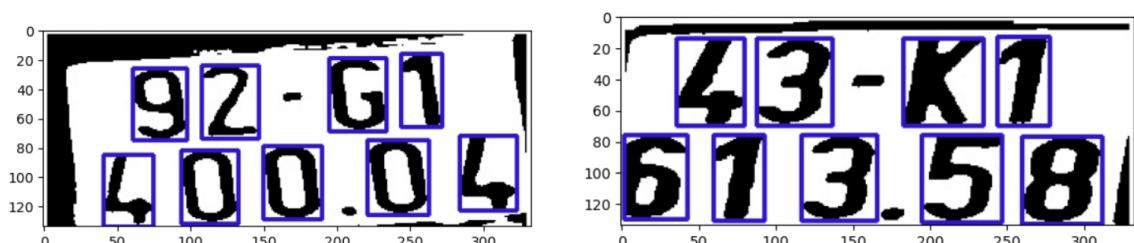
Nhận xét:

- Với mô hình phát hiện biển số xe trong khung hình bằng Cascade:
 - o Thời gian phát hiện khá nhanh
 - o Trong quá trình phát hiện, thường xuyên gặp lỗi không phát hiện được vì một số bức ảnh có chứa biển số quá lớn hoặc quá nhỏ so với khung ảnh. Hoặc do độ sáng, chói của bức ảnh khá lớn dẫn đến không thể phát hiện.
 - o Các biển số bị nghiêng sẽ không phát hiện được biển số trong khung ảnh. Nếu có thể phát hiện được thì cũng ảnh hưởng đến độ chính xác của việc đọc các ký tự trên biển số
- Với mô hình CNN dùng để đọc các ký tự:
 - o Thời gian đưa ra kết quả dự đoán khá nhanh
 - o Đôi khi các ký tự bị nhận dạng sai giữa “2” và “Z”, “1” và “I”, “0” và “O”
- Khi sử dụng vào tập dữ liệu kiểm thử thì ta được độ chính xác thấp: 1.6%

Một số kết quả nhận dạng:



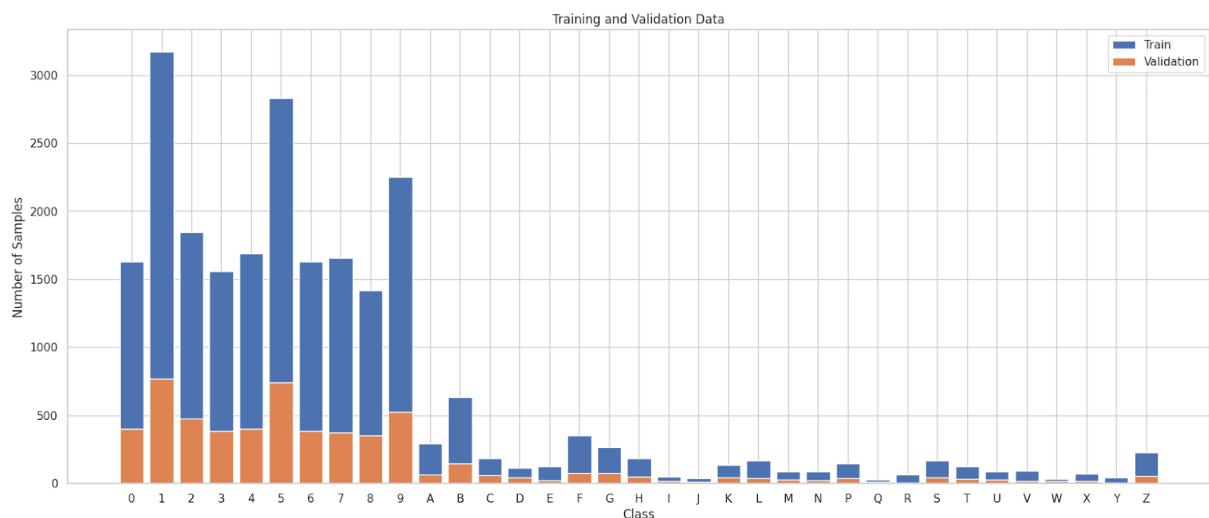
Hình 3.7. Các kết quả nhận dạng sai và lỗi trong mô hình



Hình 3.8. Một số kết quả nhận dạng đúng

3.1.2. Kết quả nhận dạng khi sử dụng YOLOv8

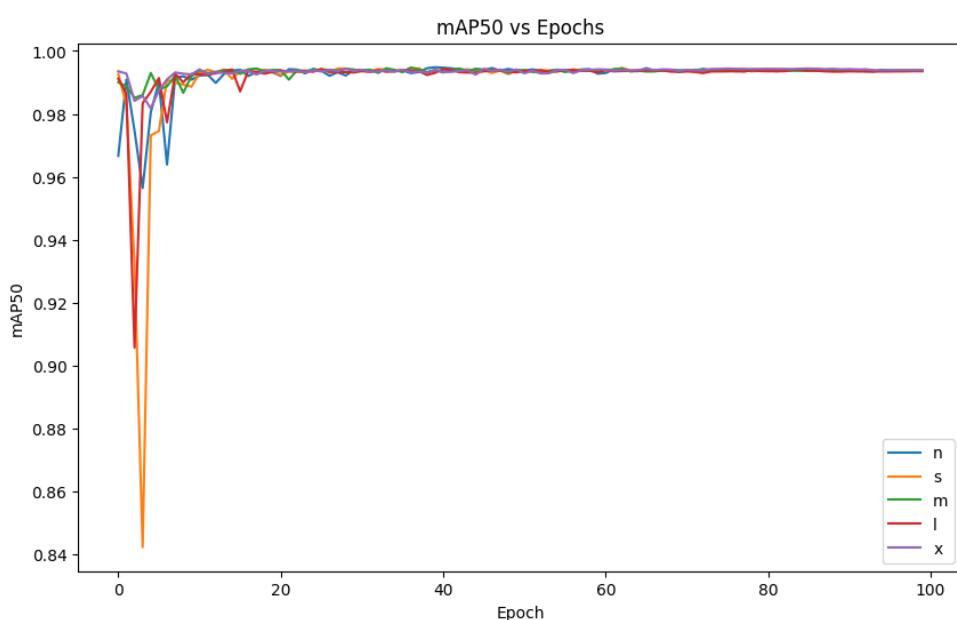
- Tập dữ liệu: Được thu thập và tổng hợp từ:
 - o <https://github.com/winter2897/Real-time-Auto-License-Plate-Recognition-with-Jetson-Nano/blob/main/doc/dataset.md>
 - o Dữ liệu nhóm chụp thực tế từ bãi đỗ xe trong trường
- Sau khi xử lí và làm sạch dữ liệu thì ta được 2 tập dữ liệu gồm:
 - o Dữ liệu cho việc nhận dạng biển số gồm:
 - Có 6847 ảnh train và 1712 ảnh validate với tỷ lệ 80/20
 - Có 1 nhãn dữ liệu là “plate”
 - o Dữ liệu cho việc nhận dạng các kí tự trên biển số gồm:
 - Có 2888 ảnh train và 715 ảnh validate với tỷ lệ 80/20
 - Gồm các nhãn: '0', '1', '2', '3', '4', '5', '6', '7', '8', '9', 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z'.
 - Theo Wikipedia thì hiện tại, đối với biển số xe máy dân sự ở Việt Nam thì không có chữ 'I', 'J', 'O', 'Q', 'W' nên ta sẽ không quan tâm đến các nhãn này



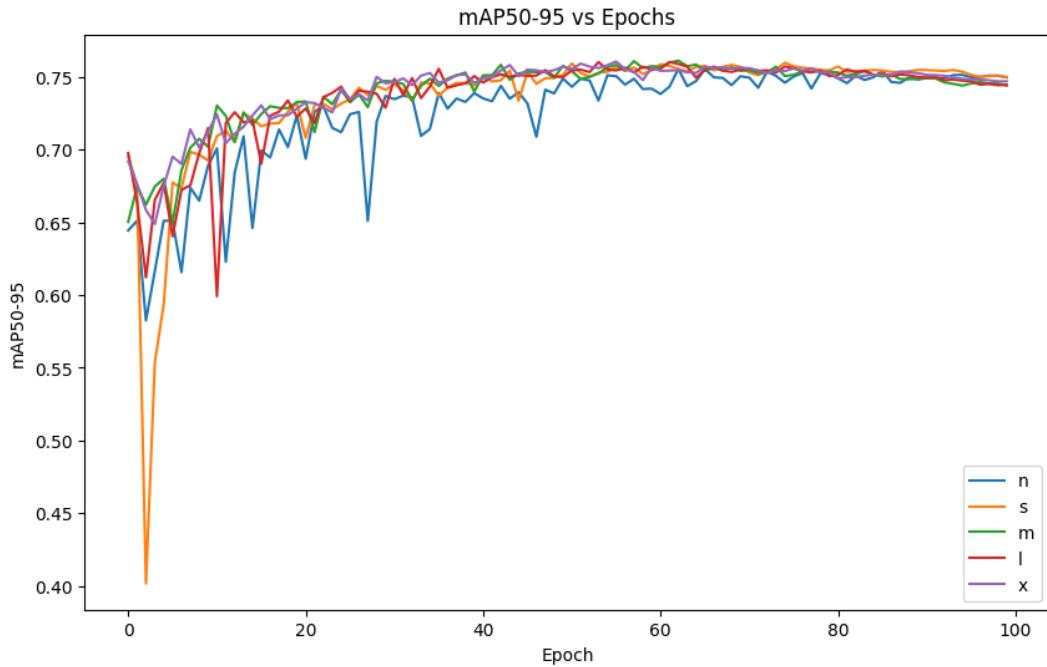
Hình 3.8. Số lượng mẫu của các class trong tập train và tập validation

- Sử dụng mô hình YOLOv8
 - o Mô hình đầu tiên: phát hiện biển số
 - o Mô hình thứ hai: nhận dạng các kí tự trên biển số

- Huấn luyện mô hình:
 - o Mô hình phát hiện biến số
 - Thực hiện huấn luyện với 5 mô hình lần lượt là yolov8n, yolov8s, yolov8m, yolov8l, yolov8x
 - Huấn luyện với các tham số: imgsz = 640, epochs = 100, batch = 112/64/36/24/16 lần lượt tương ứng với 5 mô hình trên
 - o Mô hình nhận dạng kí tự
 - Thực hiện huấn luyện với 5 mô hình lần lượt là yolov8n, yolov8s, yolov8m, yolov8l, yolov8x
 - Huấn luyện với các tham số: imgsz = 640, epochs = 100, batch = 102/56/32/24/16 lần lượt tương ứng với 5 mô hình trên
- Kết quả:
 - o Sử dụng mAP để đánh giá hiệu suất của các mô hình: mAP (mean Average Precision) là một thước đo được sử dụng để đánh giá hiệu suất của các mô hình phát hiện đối tượng trong lĩnh vực học sâu và thị giác máy tính. Nó thường được sử dụng trong các bài toán như phát hiện đối tượng (object detection) và phân đoạn (segmentation).
 - o Thực hiện đánh giá mô hình dựa trên mAP50 (ngưỡng IoU = 0.5) và mAP50-95 (ngưỡng IoU từ 0.5 đến 0.95 với bước nhảy 0.05)
 - o IoU (intersection over union) là tỉ số của diện tích trùng lặp của 2 bounding box trên diện tích hợp của 2 bounding box.

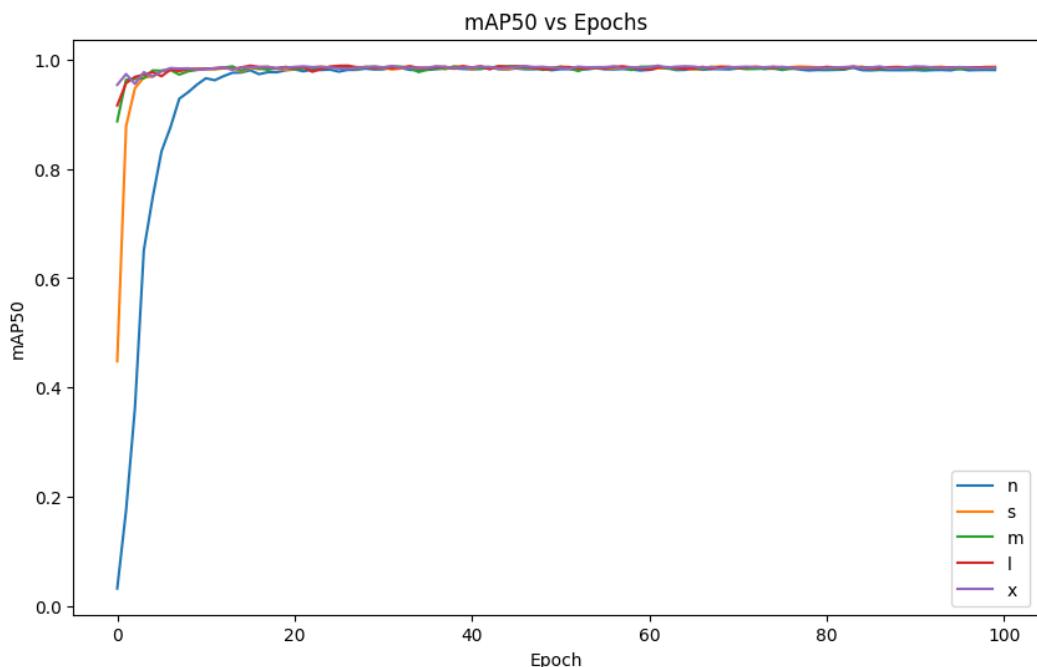


Hình 3.9. Kết quả mAP50 với mô hình phát hiện biển số

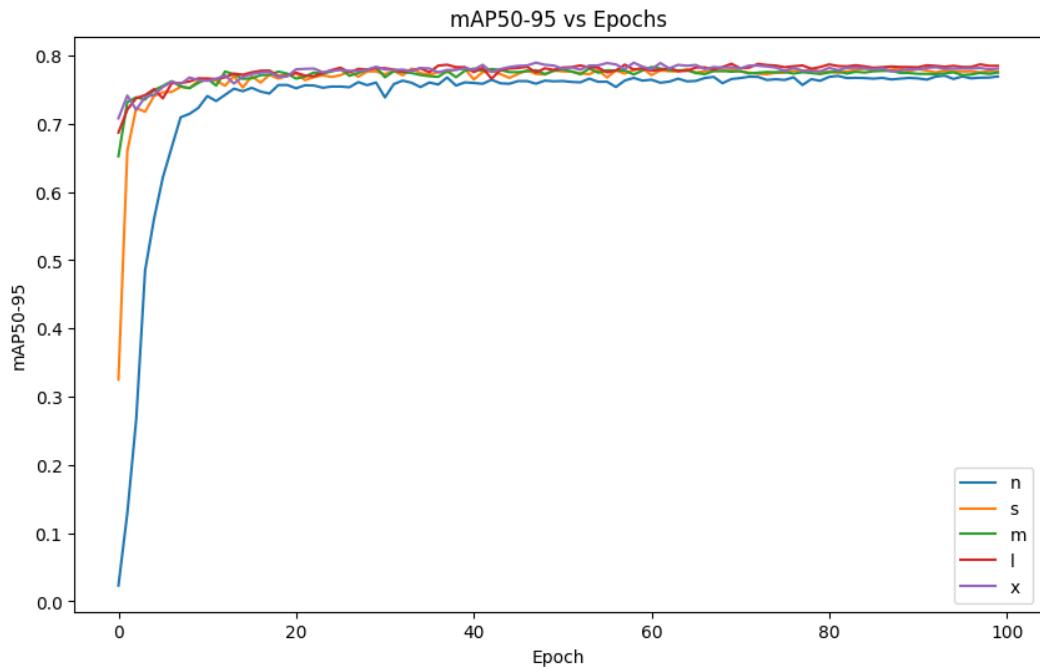


Hình 3.10. Kết quả mAP - 95 với mô hình phát hiện biển số

- ⇒ Nhận xét: Ta thấy kết quả trên mAP50 từ epoch 50 trở đi thì các mô hình có hiệu suất tương tự nhau, còn trên mAP50-95 thì các mô hình yolov8n và yolov8s có hiệu suất thấp hơn 3 mô hình còn lại và hiệu suất các mô hình giảm nhẹ từ epoch 60 trở đi.
- ⇒ Để cân bằng về tốc độ cũng như hiệu suất, ta sẽ chọn yolov8m là mô hình chính để thực hiện việc phát hiện biển số.

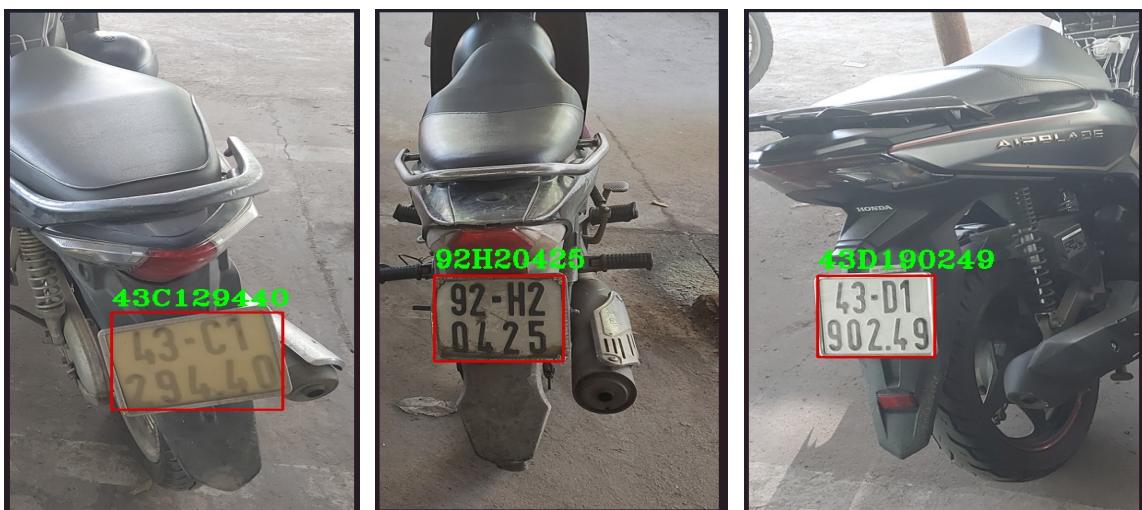


Hình 3.11. Kết quả mAP – 50 với mô hình nhận diện kí tự



Hình 3.12. Kết quả mAP – 95 với mô hình nhận dạng kí tự

- ⇒ Nhận xét: Ta thấy kết quả trên mAP50 từ epoch 50 trở đi thì các mô hình có hiệu suất tương tự nhau, còn trên mAP50-95 thì các mô hình yolov8n và yolov8s có hiệu suất thấp hơn 3 mô hình còn lại
- ⇒ Để cân bằng về tốc độ cũng như hiệu suất, ta sẽ chọn yolov8m là mô hình chính để thực hiện việc nhận dạng kí tự.



Hình 3.13. Kết quả sau khi kết hợp 2 mô hình YOLOv8

- Phương pháp này hạn chế và khắc phục được các yếu điểm của phương pháp dùng **Cascade** và **CNN** ở trên, tuy nhiên các trường hợp biển số bị nghiêng và tối quá nhiều thì dự đoán vẫn sai.

3.1.3. So sánh 2 phương pháp

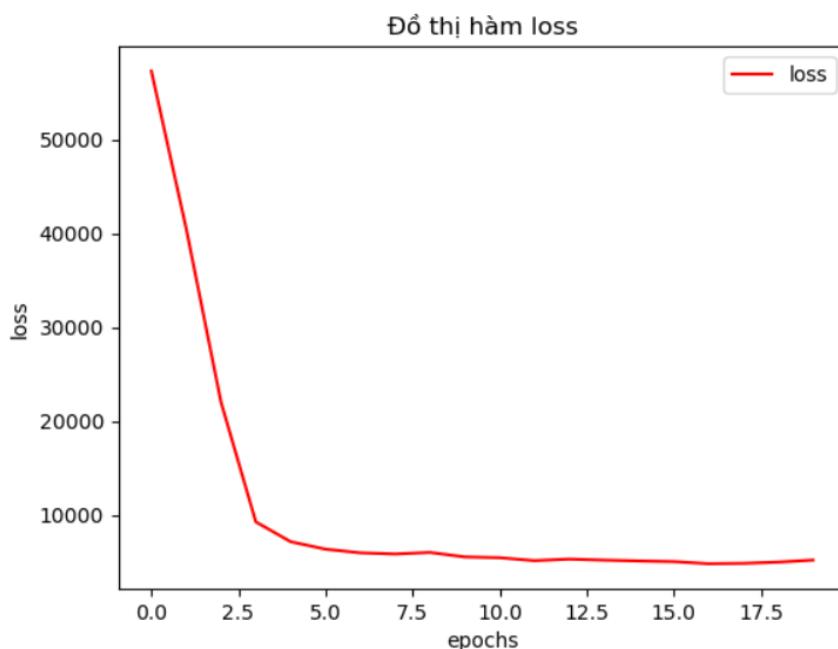
Thực hiện so sánh hiệu suất của hai phương pháp với dữ liệu test bao gồm 429 ảnh do nhóm thu thập.

- Thời gian thực thi:
 - Với phương pháp dùng **Cascade và CNN**: 160s trên cả tập test
 - Với phương pháp dùng Yolov8: 90s trên cả tập test
- Hiệu suất:
 - Với phương pháp dùng **Cascade và CNN**: Khi chạy trên tập dữ liệu test thì ta được độ chính xác thấp: 1.6%
 - Với phương pháp dùng Yolov8: Khi chạy trên tập dữ liệu test thì ta được độ chính xác cao: 98%

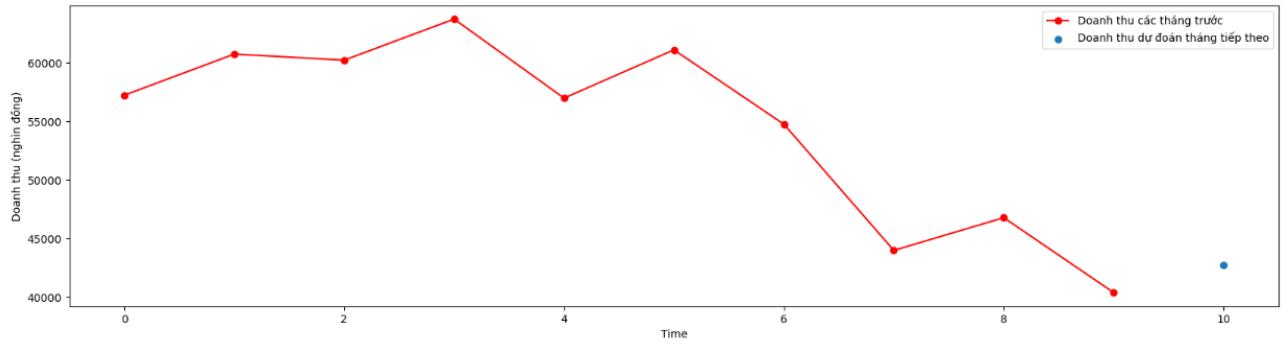
Từ các kết quả trên, ta thấy phương pháp dùng Yolov8 cải thiện đáng kể so với phương pháp dùng **Cascade và CNN**

3.2. Kết quả mô hình dự đoán doanh thu

Việc sử dụng mô hình RNN còn nhiều khó khăn do còn phụ thuộc khá nhiều yếu tố ngoại cảnh khác nhau

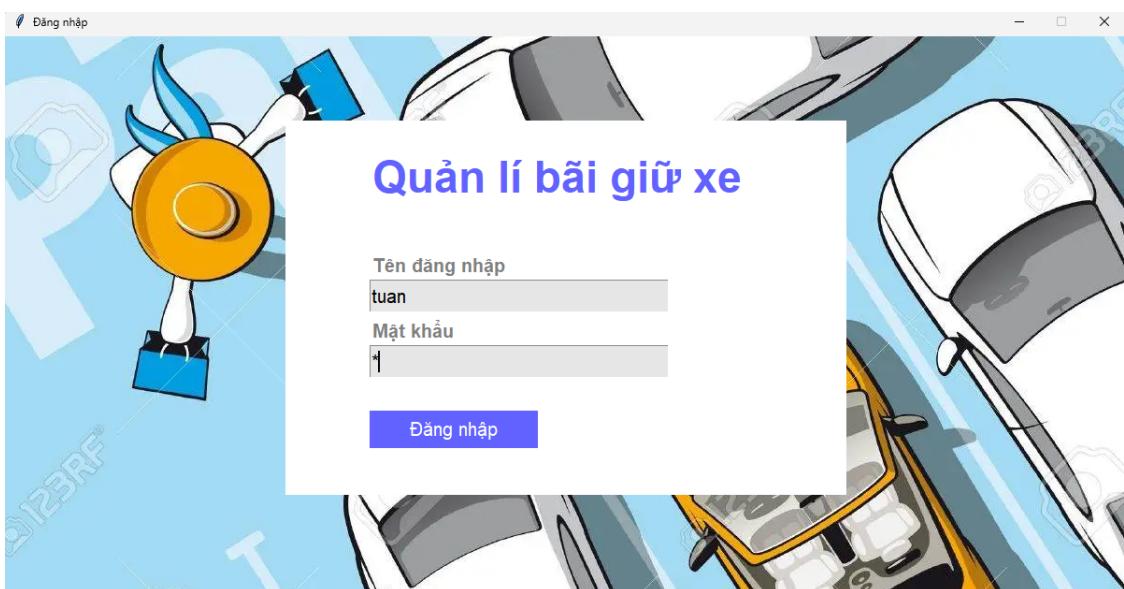


Hình 3.14. Đồ thị hàm loss của mô hình RNN



Hình 3.15. Kết quả dự đoán doanh thu của tháng tiếp theo

3.3. Kết quả chương trình



Hình 3.16. Giao diện đăng nhập



Hình 3.17. Giao diện chính của quản lý vào ra

Đăng kí

Thông tin khách hàng

Tên	<input type="text"/>
ID	<input type="text" value="6"/>
SDT	<input type="text"/>
CCCD	<input type="text"/>
Biển số xe	<input type="text" value="75H109874"/>

Hoàn tất

Đăng kí gửi xe

Giờ Ngày
 Tuần Tháng

Bảng giá

Giờ	<input type="text" value="6000"/>
Ngày	<input type="text" value="30000"/>
Tuần	<input type="text" value="500000"/>
Tháng	<input type="text" value="1500000"/>

Hình 3.18. Giao diện đăng kí thông tin xe khi chưa đăng kí

Đăng kí

Thông tin khách hàng

Tên	<input type="text" value="Nhat Tan"/>
ID	<input type="text" value="6"/>
SDT	<input type="text" value="0909123123"/>
CCCD	<input type="text" value="1234567123"/>
Biển số xe	<input type="text" value="68G166886"/>

Hoàn tất

Đăng kí gửi xe

Giờ Ngày
 Tuần Tháng

Bảng giá

Giờ	<input type="text" value="6000"/>
Ngày	<input type="text" value="30000"/>
Tuần	<input type="text" value="500000"/>
Tháng	<input type="text" value="1500000"/>

Hình 3.19. Giao diện của khi hoàn thành thủ tục đăng kí

Danh mục quản lý

Quản lý bãi đỗ xe

Thoát





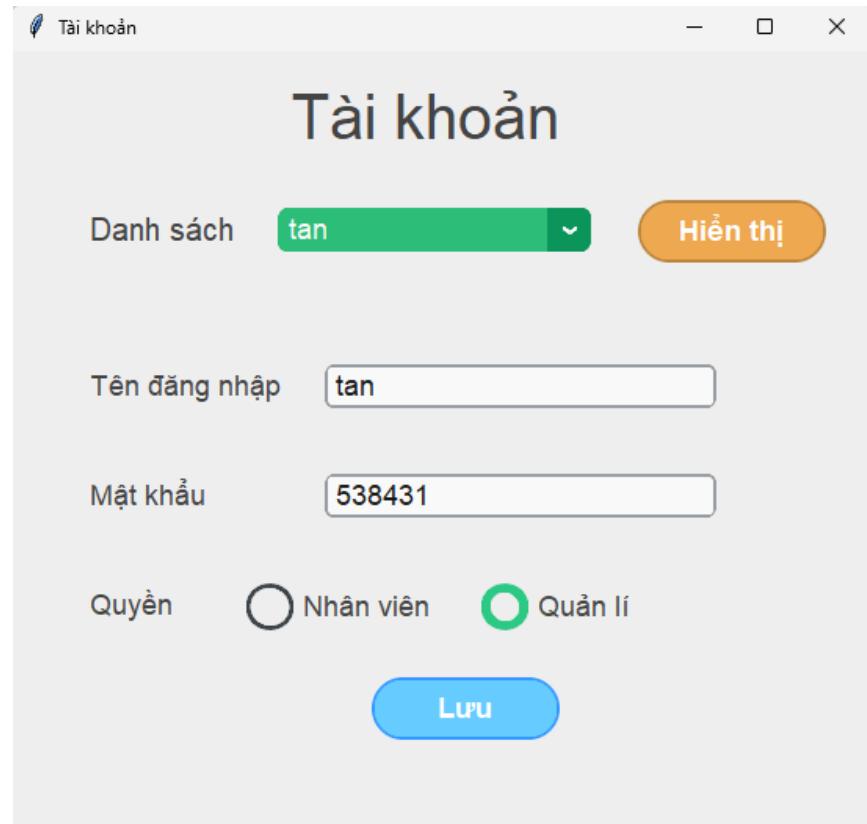

Quản lý doanh thu

Quản lý tài khoản

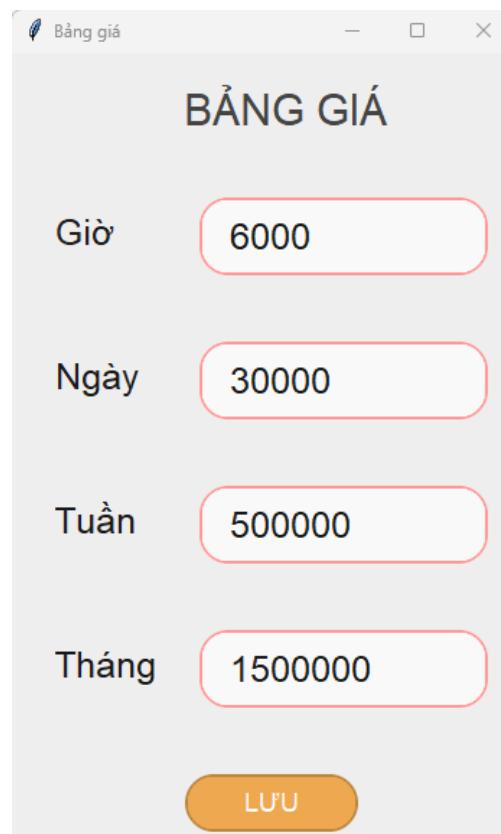
Quản lý xe

Quản lý giá tiền

Hình 3.20. Giao diện quản lý bãi xe



Hình 3.21. Giao diện quản lý tài khoản đăng nhập



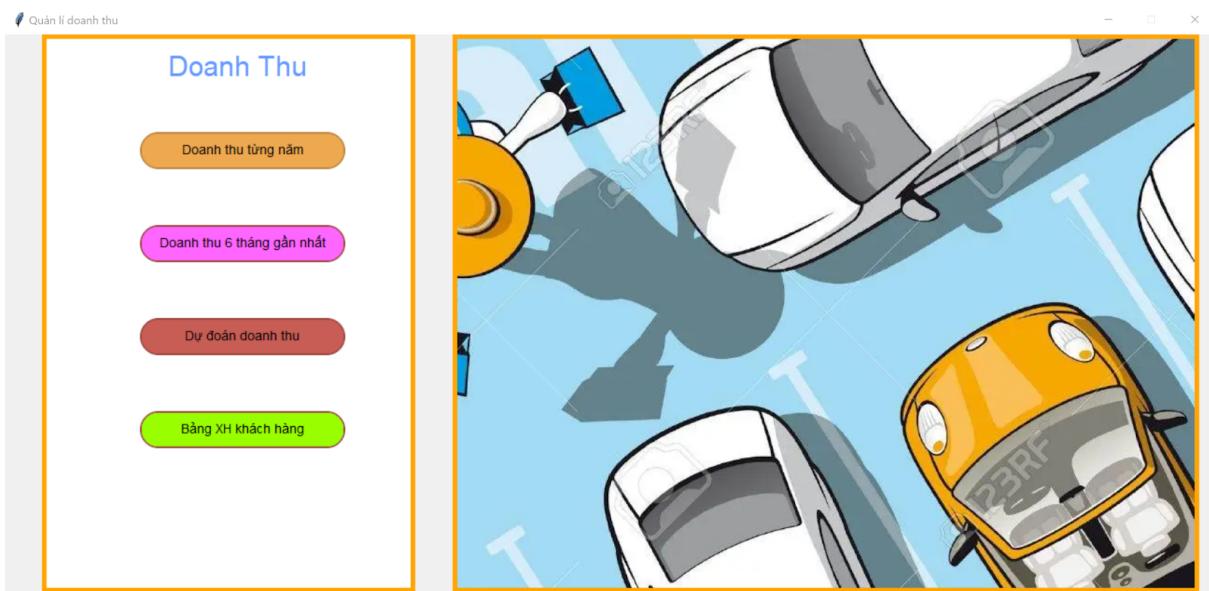
Hình 3.22. Giao diện quản lý giá cả của các khoảng thời gian đỗ xe



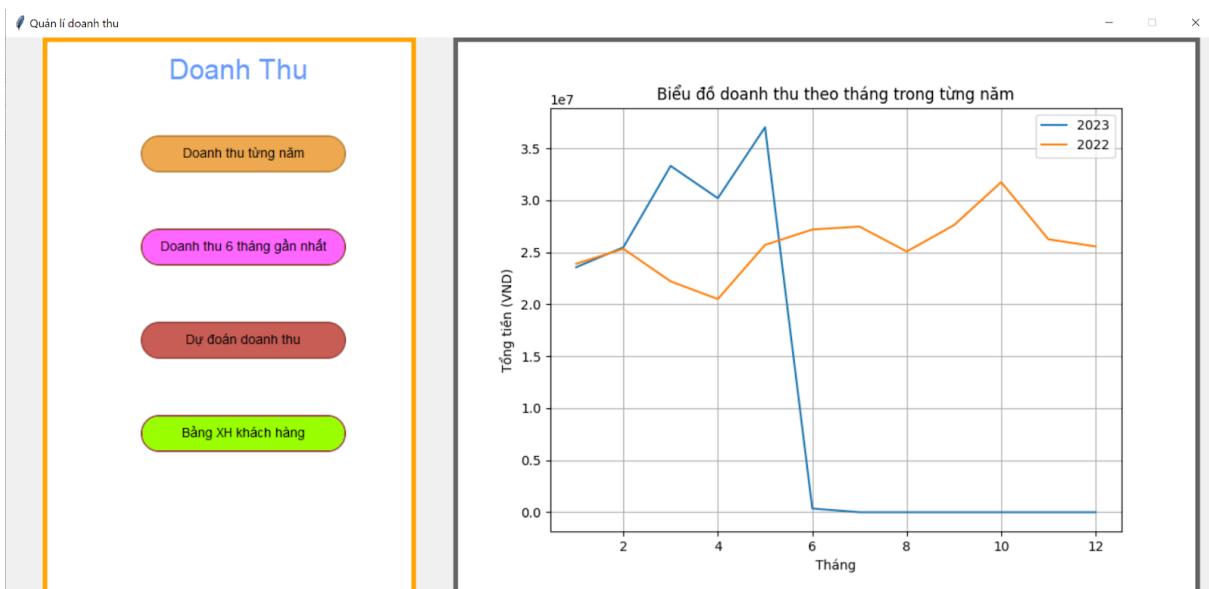
The screenshot shows a software interface titled "QUẢN LÍ XE" (Vehicle Management). The main area displays a table of parking records with the following columns: ID, ID chủ sở hữu (Owner ID), Tên chủ (Owner Name), SDT (Phone Number), CCCD (ID Card Number), Time-register (Registration Time), Period-time (Duration), and Tổng tiền (Total Price). The data includes various entries such as Tanhishi, Tuanhuu, and Nguyen Van Ngoc Hieu.

ID	ID chủ sở hữu	Tên chủ	SDT	CCCD	Time-register	Period-time	Tổng tiền
1	1	Tanhishi	0934744073	1512351253	2023-06-04 17:36:59	2h	6000 VND
2	2	Tuanhuu	0914201276	3123123124	2023-06-04 19:45:29	5h	15000 VND
3	2	Tuanhuu	0914201276	3123123124	2023-06-04 19:58:44	96.0h	120000 VND
4	1	Tanhishi	0934744073	1512351253	2023-06-04 19:59:58	2h	6000 VND
5	3	Tuanhuu	0912412312	3123123121	2023-06-04 20:10:59	48.0h	60000 VND
6	4	Nguyen Van Ngoc Hieu	0909100110	9182736451	2023-06-04 20:40:37	2h	6000 VND
7	5	Hieu Giap	0909111111	1234123124	2023-06-04 20:42:32	4h	12000 VND
8	3	Tuanhuu	0912412312	3123123121	2023-06-05 19:25:59	5.0h	30000 VND
9	3	Tuanhuu	0912412312	3123123121	2023-06-05 19:26:15	48.0h	60000 VND
10	3	Tuanhuu	0912412312	3123123121	2023-06-05 19:27:24	0.5h	3000.0 VND
11	6	Tanhishi123	0934182312	3123124111	2023-06-05 19:44:44	0.4	2400.0 VND
12	6	Tanhishi123	0934182312	3123124111	2023-06-05 19:45:21	0.6h	3600.0 VND

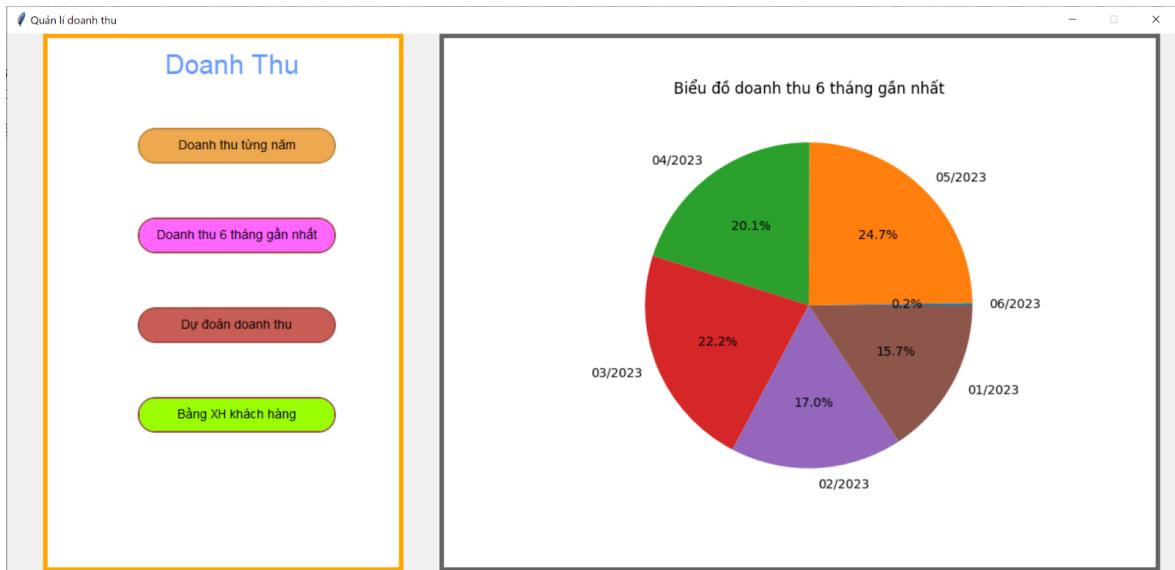
Hình 3.23. Giao diện quản lí xe và thời gian ra vào bãi



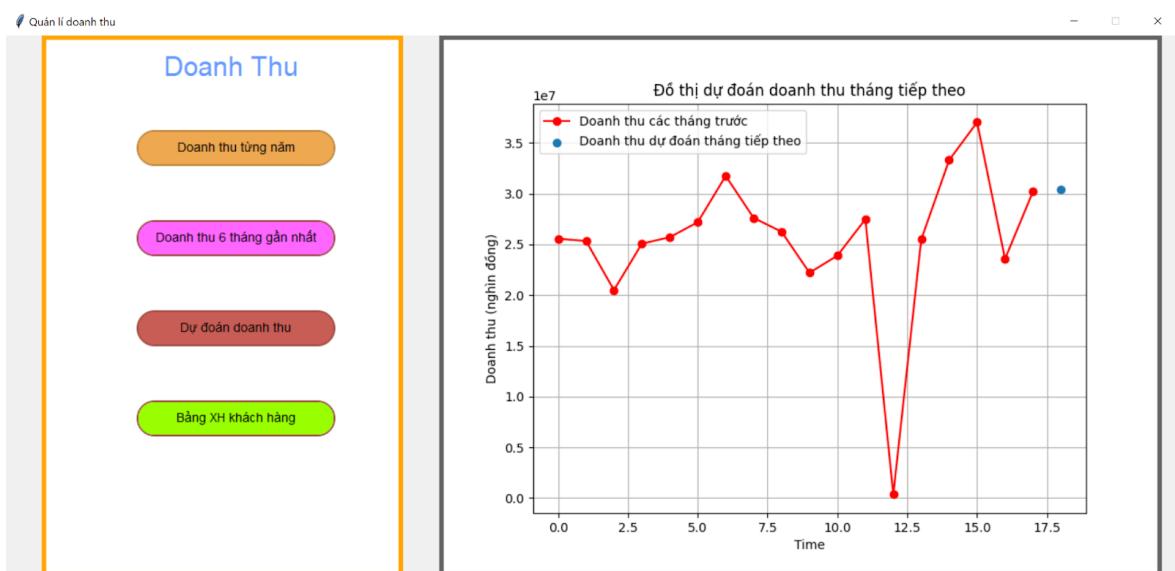
Hình 3.24. Giao diện quản lí doanh thu



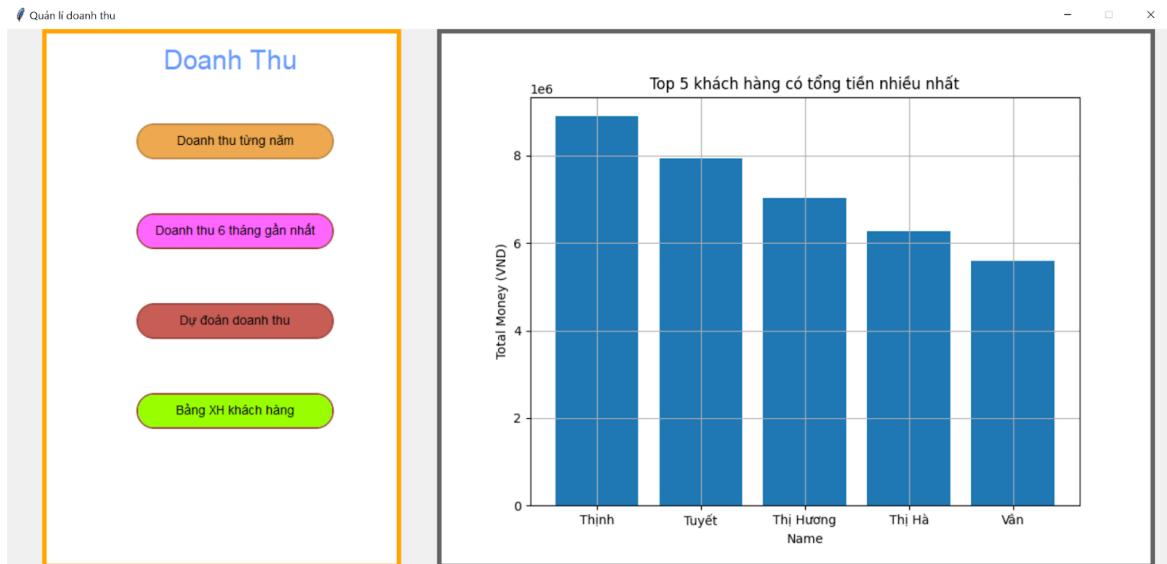
Hình 3.25. Giao diện biểu thị doanh thu của từng tháng trong các năm



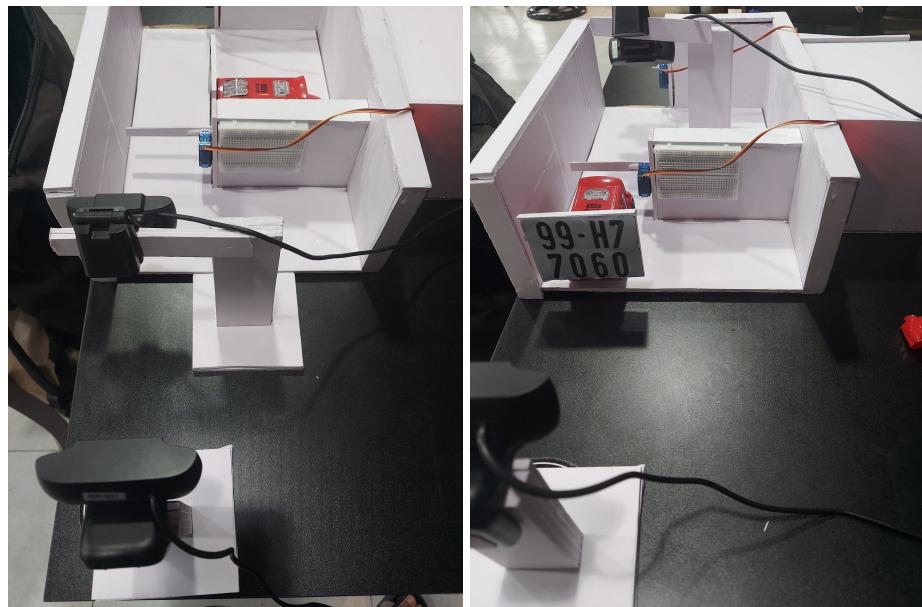
Hình 3.26. Giao diện doanh thu của 6 tháng gần nhất



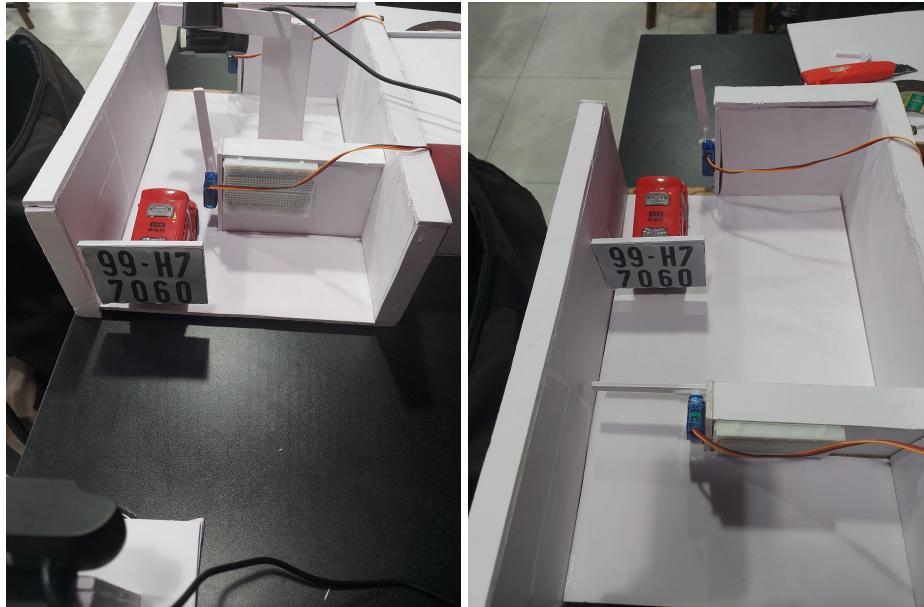
Hình 3.27. Giao diện dự đoán doanh thu



Hình 3.28. Giao diện top 5 khách hàng



Hình 3.29. Mô hình mô phỏng bãi đỗ xe



Hình 3.30. Mô hình mô phỏng bARRIER xe (t)

4. Kết luận

4.1. Kết luận

Thông qua đề tài lần này, nhóm đã hiểu hơn và áp dụng được các kiến thức trong bộ môn Vi Điều Khiển, Khoa Học Dữ Liệu, Mạng Nơron trong việc thiết lập phần cứng, phần mềm và huấn luyện, kiểm thử và sử dụng mô hình trong các vấn đề nhận dạng. Từ đó các vấn đề đặt ra trong đề tài đã được hoàn thành và đáp ứng được yêu cầu đưa ra.

Về phần mềm:

- **Phát hiện biển số:** Chức năng phát hiện bằng YOLOv8 có độ chính xác khác cao. Tuy nhiên, khi hình ảnh biển số quá to hoặc quá nhỏ so với kích thước ảnh thì việc nhận dạng diễn ra không tốt. Đối với mô hình Cascade để phát hiện biển số thì độ chính xác khá thấp và quá phụ thuộc vào góc độ, độ sáng của hình ảnh khiến cho việc phát hiện còn nhiều khó khăn.
- **Đọc các kí tự trên biển số:** Khi sử dụng YOLOv8 thì việc đọc các kí tự trên biển số có độ chính xác khá cao so với khi sử dụng mô hình CNN thì khá nhạy cảm với ánh sáng môi trường và độ nghiêng của ảnh và đáp ứng đủ yêu cầu đặt ra của đồ án. Tuy nhiên, với các biển số có độ nghiêng quá lớn sẽ ảnh hưởng đến thứ tự của các kí tự trên biển số hoặc ảnh hưởng đến độ chính xác của mô hình

- **Ứng dụng desktop:** đáp ứng được nhu cầu sử dụng và giải quyết các vấn đề đưa ra. Giao diện thân thiện và dễ dàng sử dụng với người dùng.

Về phần cứng:

- Việc lắp đặt phần cứng khác đơn giản và không mất quá nhiều thời gian. Dễ dàng di chuyển và cần quá nhiều nhân lực.

4.2. Hướng phát triển

Trong vấn đề nhận dạng, nhóm đề xuất ra các giải pháp sau:

- Với việc nhận dạng, có thể tăng độ phức tạp của tập dữ liệu huấn luyện giúp cho mô hình học được tốt hơn. Cụ thể, trong tập dữ liệu huấn luyện của mô hình phát hiện biển số, thêm các hình với các biển số ở góc nghiêng khác nhau và số có độ sáng khác nhau giúp cho độ chính xác của mô hình tăng lên. Cũng như trong mô hình đọc ký tự, việc tăng độ phức tạp cũng giúp tăng độ chính xác mô hình
- Sử dụng các mô hình khác để đưa ra được kết quả tốt hơn cho việc nhận dạng

Về phần cứng, nếu có thêm kinh phí thì có thể đầu tư vào các camera tốt hơn giúp cho dữ liệu hình ảnh đầu vào của mô hình được tốt hơn, tăng độ chính xác của mô hình.

5. Danh mục tài liệu tham khảo

[1] winter2897, “Real time Auto License Plate Recognition with Jetson Nano”

<https://github.com/winter2897/Real-time-Auto-License-Plate-Recognition-with-Jetson-Nano>

[2] mrzaizai2k, “VIETNAMESE LICENSE PLATE”

https://github.com/mrzaizai2k/VIETNAMESE_LICENSE_PLATE

[3] “YOLOv8 Github” <https://github.com/ultralytics/ultralytics>

[4] “YOLOv8 Documentation” <https://docs.ultralytics.com/>

[5] David Amos, “Python GUI tkinter”, <https://realpython.com/python-gui-tkinter/>

[6] teja00219, “Build a basic form GUI using custom tkinter module in python”,

<https://www.geeksforgeeks.org/build-a-basic-form-gui-using-customtkinter-module-in-python/>

[7] Marsmallotr , “License Plate Recognition”

<https://github.com/Marsmallotr/License-Plate-Recognition>

[8] Nguyễn Chiến Thắng, “YOLOv8 và train với dữ liệu cá nhân”,

<https://miae.vn/2023/05/19/thu-nghich-yolov8-va-train-voi-du-lieu-ca-nhan-mi-ai/>