



**HỌC VIỆN  
CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**  
Posts and Telecommunications Institute of Technology

# Hội đồng chấm đồ án tốt nghiệp

**Sinh viên khóa 2018**

Khoa CNTT1 - Học viện công nghệ Bưu chính Viễn Thông



**HỌC VIỆN  
CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**  
Posts and Telecommunications Institute of Technology

## **ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC**

# **ỨNG DỤNG NHẬN DẠNG KÝ TỰ QUANG CHO BÀI TOÁN NHẬN ĐIỆN BIẾN SỐ XE**

Người hướng dẫn:  
Sinh viên thực hiện:  
Mã sinh viên:  
Lớp:  
Hệ:

ThS. Đinh Xuân Trường  
Vũ Đức Đông  
B18DCCN161  
D18HTTT4  
Đại học chính quy

Chúng ta đang sống trong thời kỳ cách mạng công nghiệp 4.0

- Công nghệ phát triển mạnh mẽ
- Xử lý ảnh nói chung hay nhận dạng ký tự quang học nói riêng là một trong những công nghệ được quan tâm nhiều
- Ứng dụng trong một số lĩnh vực như: tài chính ngân hàng, giáo dục, y tế, giao thông, ...

Sự bùng nổ kinh tế, xã hội. Nhu cầu đặt ra là cần có hệ thống thông minh hỗ trợ tự động hóa các bãi đỗ xe

**Ứng dụng nhận dạng ký tự quang cho bài toán nhận diện biển số xe**

# NỘI DUNG CHÍNH

01

## GIỚI THIỆU CHUNG

Tổng quan đề tài, mục tiêu và phương hướng giải quyết

...

02

## CƠ SỞ LÝ THUYẾT

Kiến thức công nghệ liên quan sẽ sử dụng khi xây dựng chương trình

...

03

## XÂY DỰNG CHƯƠNG TRÌNH

Xây dựng mô hình dự đoán, website demo và đánh giá kết quả

...

04

## KẾT LUẬN VÀ PHƯƠNG HƯỚNG PHÁT TRIỂN

Tổng kết, đưa ra phương hướng phát triển

...



# 1

## GIỚI THIỆU CHUNG

## TỔNG QUAN ĐỀ TÀI

Cơ sở lý thuyết về xử lý ảnh, nhận dạng ký tự quang, học sâu, ...

Bài toán cụ thể là nhận diện biển số xe ô tô Việt Nam khi xe ra vào các bãi đỗ xe.



## MỤC TIÊU

Xây dựng Website hỗ trợ quản lý bãi đỗ xe thông minh

Đưa ra thông tin về chủ xe cũng như thông tin quản lý hiện trạng bãi đỗ xe

# PHƯƠNG HƯỚNG GIẢI QUYẾT

- Thu thập dữ liệu
- Sử dụng một tool mã nguồn mở labellmg để gán nhãn dữ liệu
- Sử dụng mô hình InceptionResNetV2 của mạng Nơ-ron tích chập CNN để huấn luyện mô hình
- Sử dụng thư viện Tesseract để nhận dạng ký tự biến số xe
- Sử dụng framework Flask để tạo nên Website hệ thống

# 2

## CƠ SỞ LÝ THUYẾT



# BIỂN SỐ XE Ô TÔ TẠI VIỆT NAM



Dạng dài: một hàng, 110 x 470 mm

Số lượng ký tự [7,9], 80x40 mm

Chữ số màu đen trên nền biển màu trắng

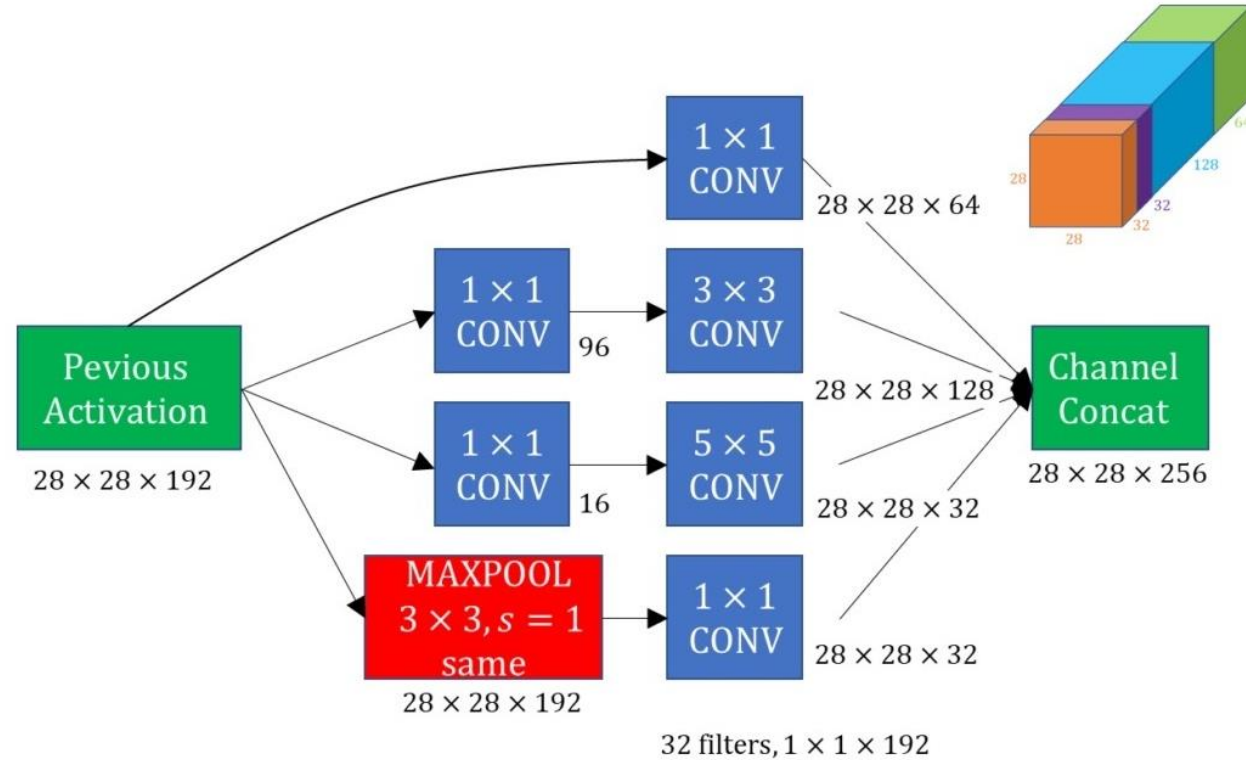
Sử dụng lần lượt một trong 20 chữ cái : A, B, C, D, E, F, G, H, K, L, M, N, P, S, T, U, V, X, Y, Z

# Mô hình InceptionResnetV2

InceptionResNetV2 là một mạng nơ-ron phức hợp được đào tạo trên hơn một triệu hình ảnh từ cơ sở dữ liệu ImageNet, có thể phân loại hình ảnh thành nhiều loại đối tượng như: ô tô, xe đạp, máy tính, bút, các loại động vật khác nhau, ...

InceptionResNet được xây dựng dựa trên những ưu điểm của mô hình Inception và ResNet

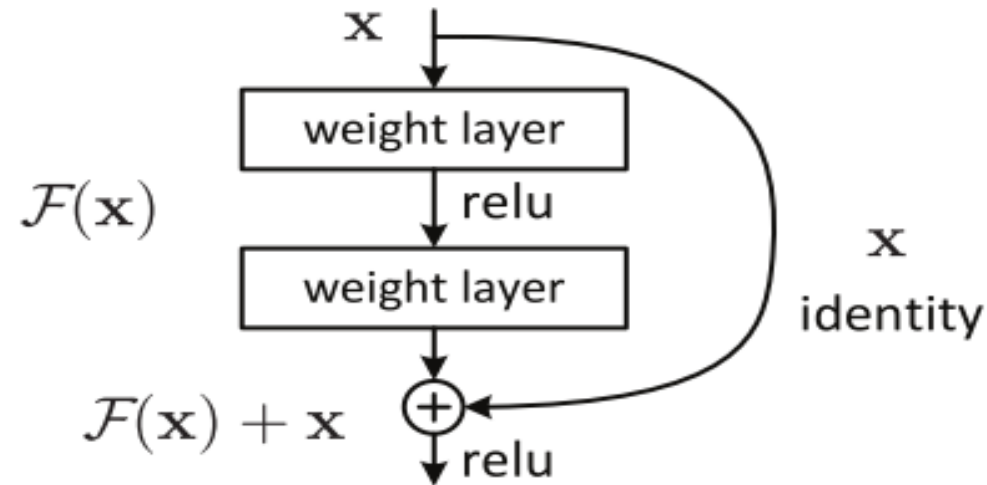
# Mô hình Inception



Hình 1: Khối inception của mô hình Inception

Sử dụng cùng lúc nhiều bộ lọc với khác nhau (1, 3, 5, 7, ...) sau đó ghép các output lại

# Mô hình Resnet



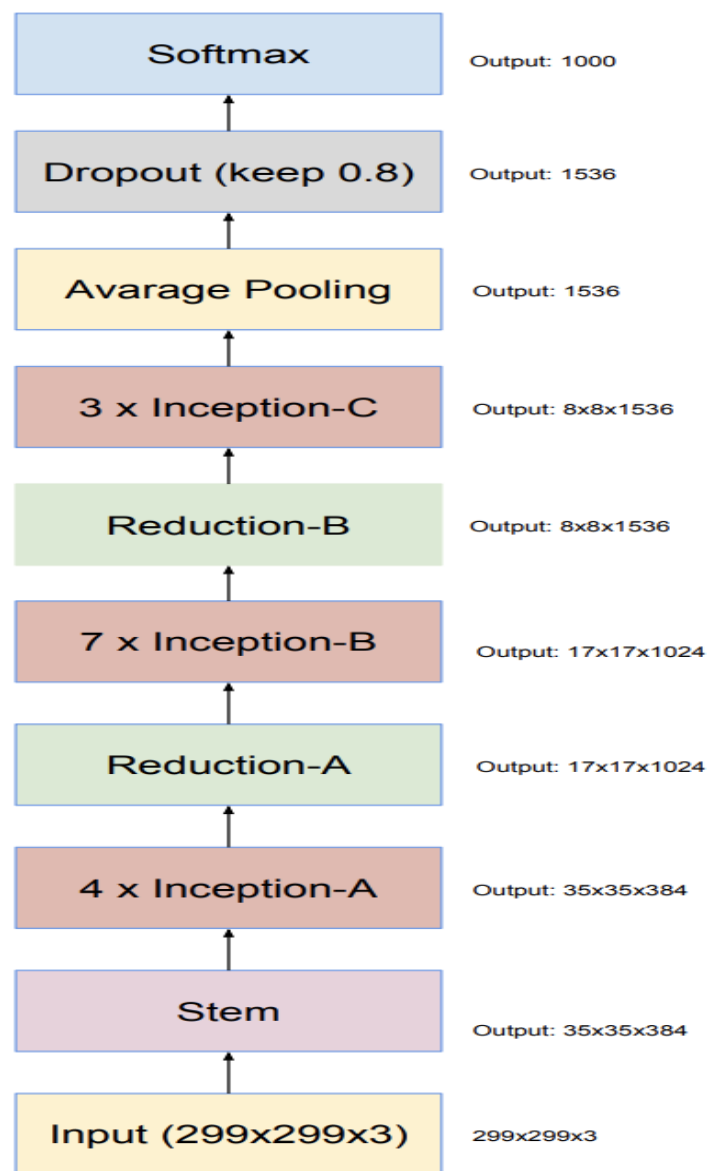
Hình 2: Minh họa Residual block

Sử dụng kết nối "tắt" đồng nhất để xuyên qua một hay nhiều lớp

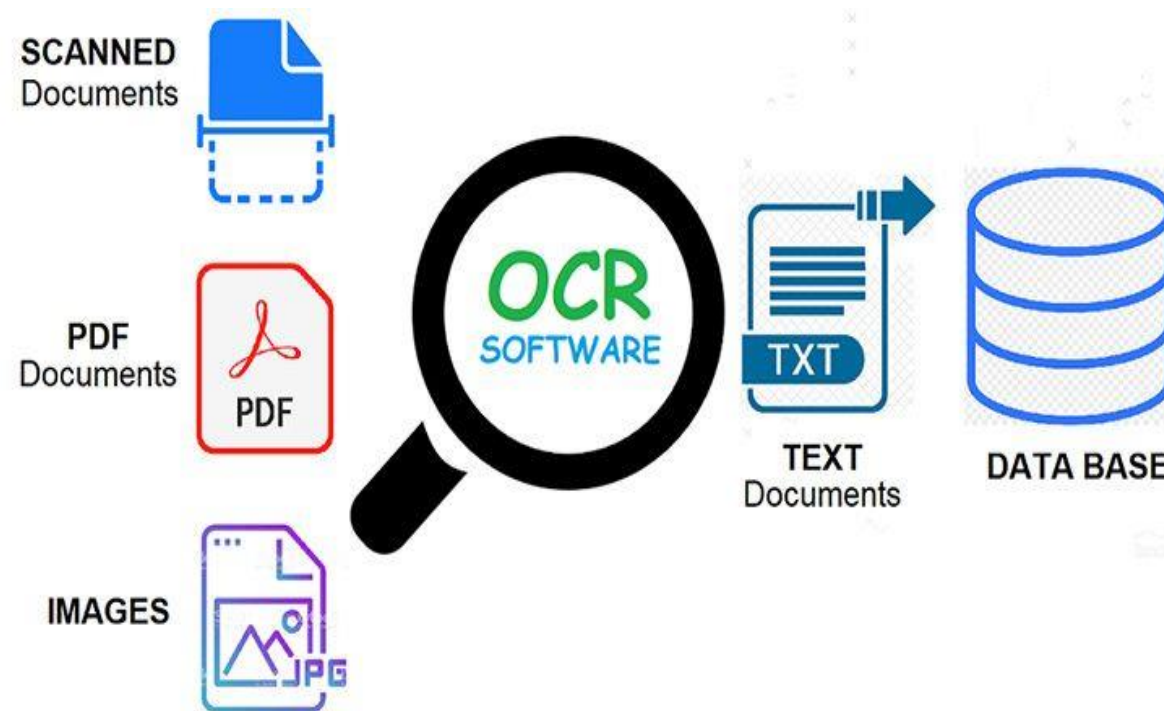
Mẫu chốt là cứ sau 2 layer, bổ sung input  $X$  vào đầu ra của layer

Huấn luyện các mô hình CNN có kích thước và độ phức tạp khủng hơn mà không lo bị mất mát đạo hàm

# Mô hình kiến trúc tổng thể của InceptionResNetV2



# NHẬN DẠNG KÝ TỰ QUANG HỌC



*Hình 3: Nhận dạng ký tự quang học OCR*

# THƯ VIỆN TESSERACT

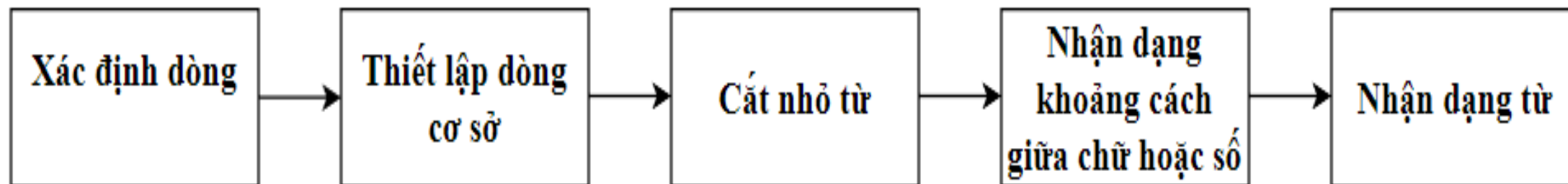
Tesseract là một OCR engine hàng đầu hiện nay

Tesseract được phân bố dưới dạng mã nguồn mở bởi HP và được phát triển bởi Google từ năm 2006

Ngôn ngữ	Tổng số ký tự (triệu)	Tổng số từ (triệu)	Lỗi ký tự (%)	Lỗi từ (%)
Tiếng Anh	39	4	0.5	3.72
Tiếng Nga	213	26	0.75	5.78
Tiếng Hoa (giản thể)	0.25	Không xác định	3.77	Không xác định
Tiếng Hindi	1.4	0.33	15.41	69.44

Phần mềm thương mại	Bộ nhận dạng Tesseract
Hỗ trợ hơn 100 ngôn ngữ	Hỗ trợ trên 40 ngôn ngữ và đang tăng dần
Có giao diện đồ họa	Không hỗ trợ giao diện đồ họa (dùng command line để gõ lệnh)
Hầu hết chỉ hỗ trợ trên Windows	Hỗ trợ trên Windows, Linux, MacOS
Độ chính xác cao mới đây	Độ chính xác khá cao từ năm 1995
Chi phí cao từ 130-500\$	Miễn phí (mã nguồn mở)

# TIỀN TRÌNH NHẬN DẠNG KÝ TỰ SỬ DỤNG TESSERACT



*Hình 4: Tiến trình nhận dạng ký tự của Tesseract*

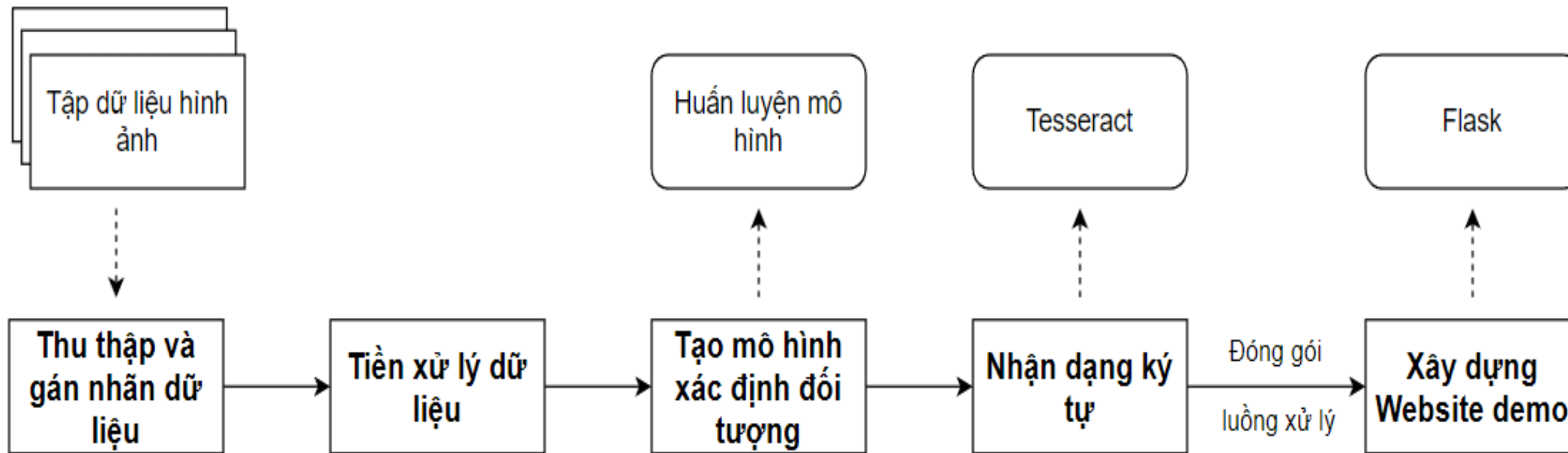


# 3

## XÂY DỰNG CHƯƠNG TRÌNH

# SƠ ĐỒ HỆ THỐNG

Hệ thống được xây dựng trải qua 5 bước chính



Hình 5: Sơ đồ hệ thống

# THU THẬP DỮ LIỆU

Bao gồm 658 bức ảnh chụp ô tô có biển số xe

500 ảnh



*thigiacmaytinj.com*

158 ảnh



*kaggle.com*

# GÁN NHÃN DỮ LIỆU



Gán nhãn cho tất cả 658 bức ảnh

Bounding box chứa thông tin: filepath, tọa độ xmin, xmax, ymin, ymax

# TIỀN XỬ LÝ DỮ LIỆU

Từ dữ liệu thô ban đầu, để có thể đưa dữ liệu vào mô hình huấn luyện và dễ xử lý hơn, ta cần trải qua bước tiền xử lý dữ liệu



# TIỀN XỬ LÝ DỮ LIỆU

Từ dữ liệu thô ban đầu, để có thể đưa dữ liệu vào mô hình huấn luyện và dễ xử lý hơn, ta cần trải qua bước tiền xử lý dữ liệu

Resize các ảnh  
về cùng kích  
thước 224x224





# TIỀN XỬ LÝ DỮ LIỆU

Từ dữ liệu thô ban đầu, để có thể đưa dữ liệu vào mô hình huấn luyện và dễ xử lý hơn, ta cần trải qua bước tiền xử lý dữ liệu

Chuyển  
ảnh sang  
dạng mảng

```
[[ [129.  98.  96.]
   [130.  99.  97.]
   [127.  96.  94.]
   ...
   [118. 126. 105.]
   [115. 124. 103.]
   [ 87.  99.  77.]]

[[ [130.  99.  97.]
   [130.  99.  97.]
   [128.  97.  95.]
   ...
   [ 28.  35.  17.]
   [ 85.  94.  75.]
   [141. 153. 133.]]

[[ [130.  99.  97.]
   [130.  99.  97.]
   [129.  98.  96.]
   ...
   [ 14.  21.   5.]
   [ 24.  31.  15.]
   [ 22.  31.  14.]]

...
```

Ảnh  
xám  
dạng  
mảng

```
[[ [0.5058824  0.38431373  0.3764706 ]
   [0.50980395  0.3882353  0.38039216]
   [0.49803922  0.3764706  0.36862746]
   ...
   [0.4627451  0.49411765  0.4117647 ]
   [0.4509804  0.4862745  0.40392157]
   [0.34117648  0.3882353  0.3019608 ]]

[[ [0.50980395  0.3882353  0.38039216]
   [0.50980395  0.3882353  0.38039216]
   [0.5019608  0.38039216  0.37254903]
   ...
   [0.10980392  0.13725491  0.06666667]
   [0.33333334  0.36862746  0.29411766]
   [0.5529412  0.6  0.52156866]]

[[ [0.50980395  0.3882353  0.38039216]
   [0.50980395  0.3882353  0.38039216]
   [0.5058824  0.38431373  0.3764706 ]
   ...
   [0.05490196  0.08235294  0.01960784]
   [0.09411765  0.12156863  0.05882353]
   [0.08627451  0.12156863  0.05490196]]

...
```

# TIỀN XỬ LÝ DỮ LIỆU

Từ dữ liệu thô ban đầu, để có thể đưa dữ liệu vào mô hình huấn luyện và dễ xử lý hơn, ta cần trải qua bước tiền xử lý dữ liệu

Chuẩn hóa tọa độ bounding box khớp với ảnh sau khi chuẩn hóa kích thước

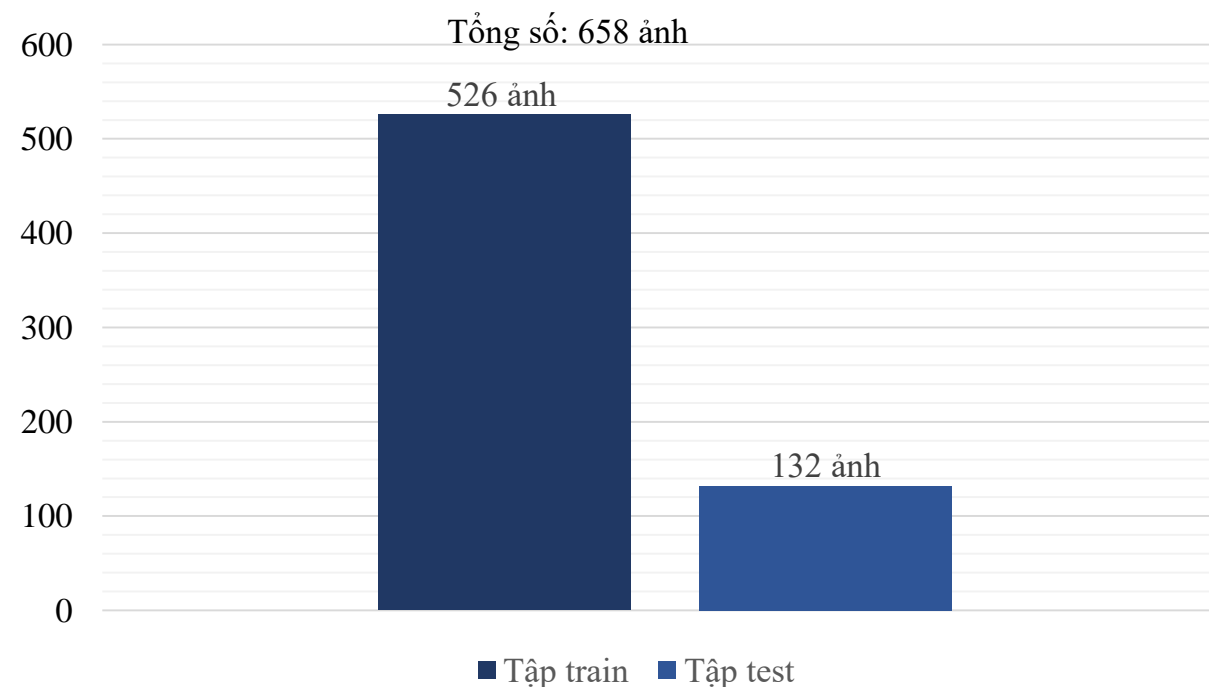
```
#Lấy chỉ số bounding box của ảnh từ file csv  
xmin,xmax,ymin,ymax = 226, 326, 201, 233  
#Tính toán tọa độ bounding box sau khi resize ảnh  
nxmin,nxmax = xmin/w,xmax/w  
nymin,nymax = ymin/h,ymax/h  
print(nxmin,nxmax,nymin,nymax)
```

```
0.4788135593220339 0.690677966101695 0.6633663366336634 0.768976897689769
```



# TIỀN XỬ LÝ DỮ LIỆU

Từ dữ liệu thô ban đầu, để có thể đưa dữ liệu vào mô hình huấn luyện và dễ xử lý hơn, ta cần trải qua bước tiền xử lý dữ liệu



# CÀI ĐẶT MÔ HÌNH DỰ ĐOÁN

Các tham số khởi tạo mô hình:

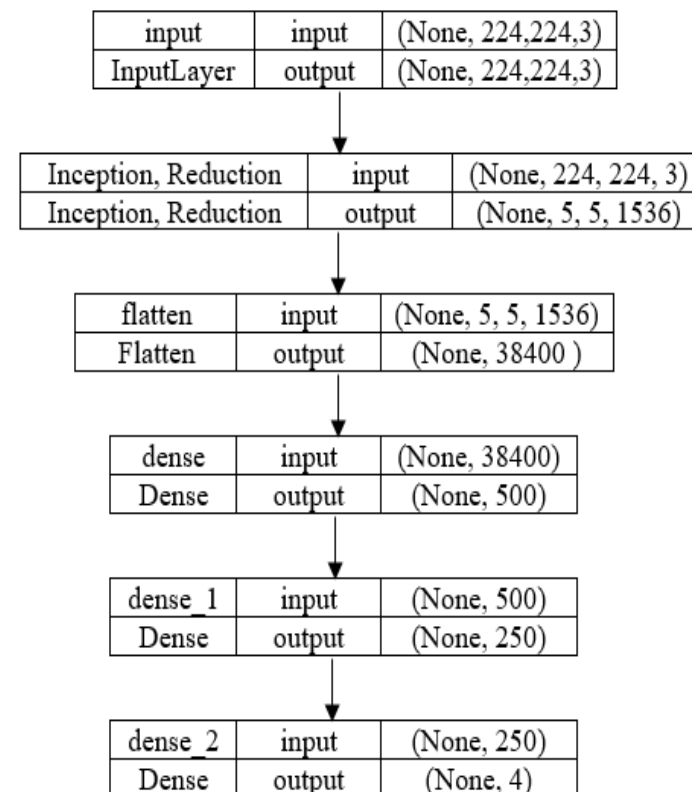
- `weights="imagenet"`
- `input_tensor=Input(shape=(224,224,3))`
- `loss='mse'`
- `optimizer = tf.keras.optimizers.Adam(learning_rate=1e-4)`
- `metrics = ["accuracy", tf.metrics.AUC()]`

Môi trường huấn luyện:

- Hệ điều hành: Windows 10
- Vi xử lý: Intel Core I5-9300H CPU @ 2.40GHz (8CPUs), ~2.4GHz
- Card đồ họa: NVIDIA GeForce GTX 1650
- RAM: 16GB

Các thông số huấn luyện mạng:

- Epochs = 200
- Batch\_size = 10



# ĐÁNH GIÁ MÔ HÌNH DỰ ĐOÁN

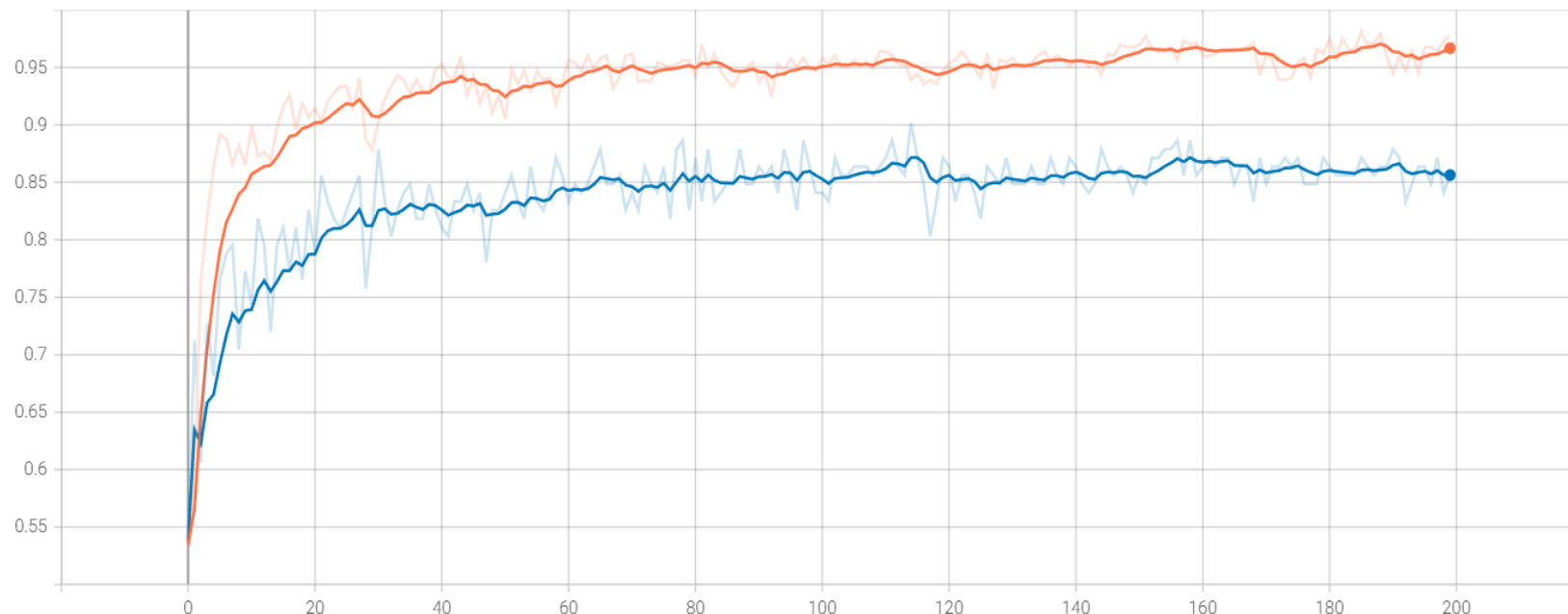
Tổng thời gian huấn luyện: 3 giờ

Thời gian huấn luyện cho mỗi Epochs là 52-54 giây

Biểu đồ thể hiện độ chính xác nhận dạng và giá trị hàm mất mát:

epoch\_accuracy

epoch\_accuracy  
tag: epoch\_accuracy



$$\text{Acc} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

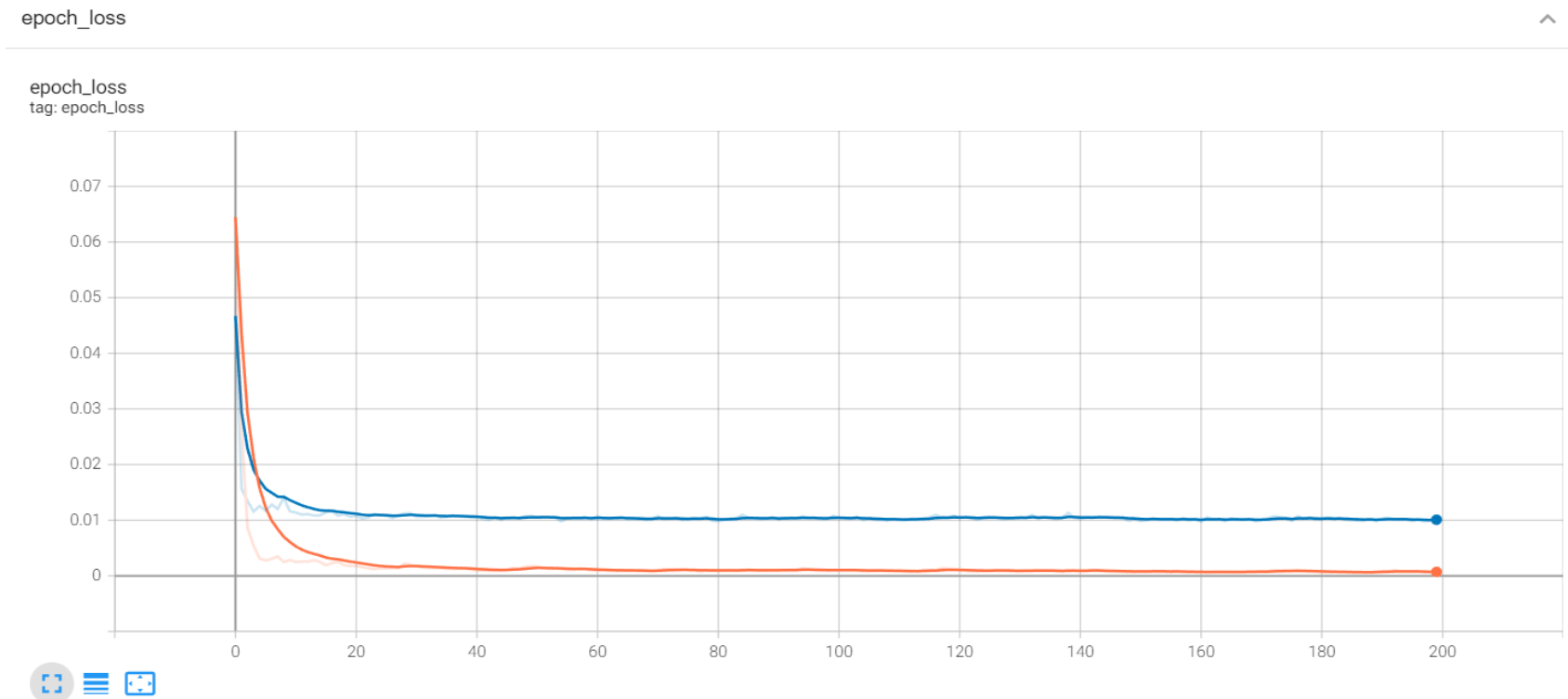


# ĐÁNH GIÁ MÔ HÌNH DỰ ĐOÁN

Tổng thời gian huấn luyện: 3 giờ

Thời gian huấn luyện cho mỗi Epochs là 52-54 giây

Biểu đồ thể hiện độ chính xác nhận dạng và giá trị hàm mất mát:



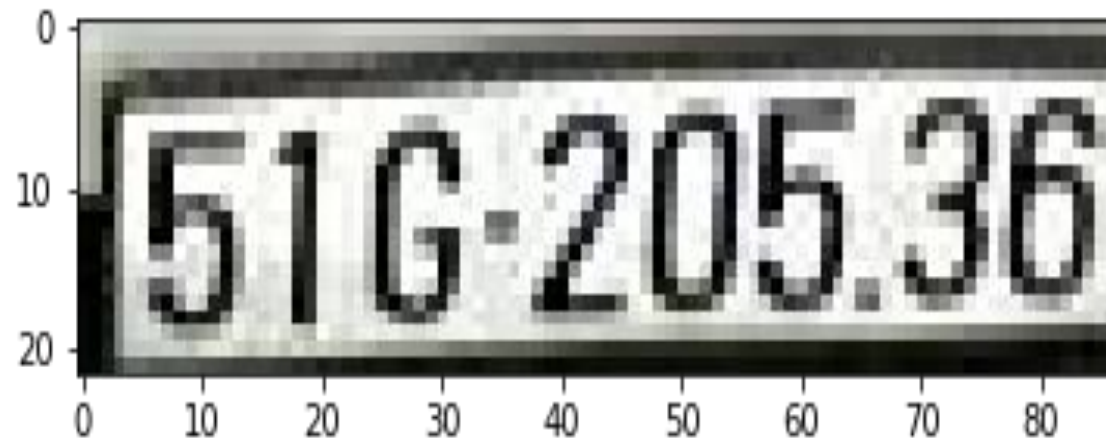
# NHẬN DẠNG KÝ TỰ BIỂN SỐ XE

Sau khi đã có thể xác định vùng ảnh chứa biển số trong bức ảnh bản đầu thông qua mô hình học sâu. Bước tiếp theo của chương trình sẽ là nhận dạng ra ký tự quang trong vùng ảnh đó, cụ thể là nhận diện ra các ký tự trong biển số xe



# NHẬN DẠNG KÝ TỰ BIỂN SỐ XE

Ảnh vùng biển số xe được cắt ra

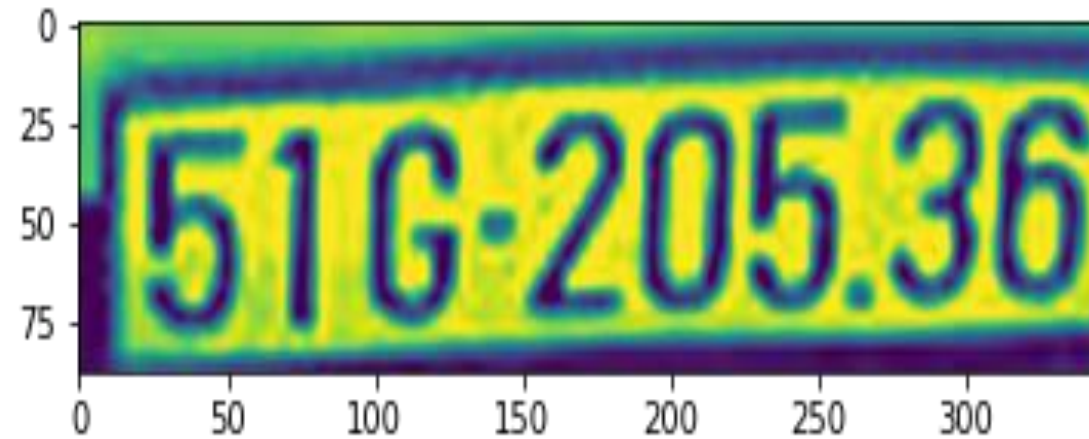


Tăng kích thước ảnh biển số đã cắt lên 2 lần cả chiều rộng và chiều cao

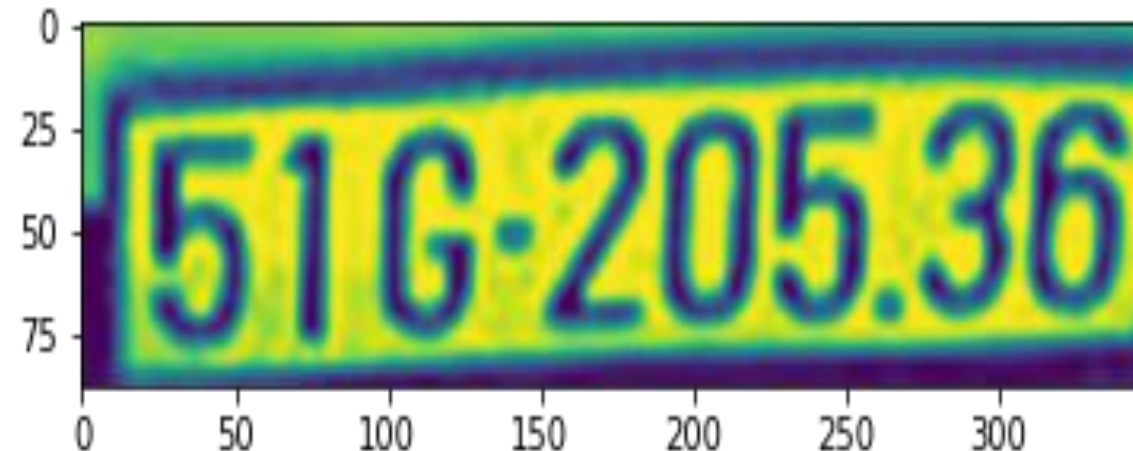


# NHẬN DẠNG KÝ TỰ BIỂN SỐ XE

Chuyển ảnh sang ảnh xám



Khử nhiễu, làm mịn ảnh sử dụng thuật toán GaussianBlur của thư viện OpenCV



## Tổng quan trang web

### Mục tiêu:

- Website giúp hỗ trợ quản lý bãi đỗ xe thông minh
- Xác vùng biển số xe, từ đó xác định ra biển số xe, thông tin trong CSDL

### Tính năng:

- Hiển thị ảnh chụp biển số xe, ký tự biển số
- Hiển thị thông tin chủ xe, tình trạng đăng ký đỗ xe tại bãi đỗ
- Xác định chi phí gửi xe nhanh chóng
- Cung cấp các thông số thống kê như: tổng số chỗ đỗ xe, số chỗ còn trống, tổng số lượt vào bãi đỗ xe trong ngày (đã đăng ký và vắng lai), tổng phí gửi xe trong ngày



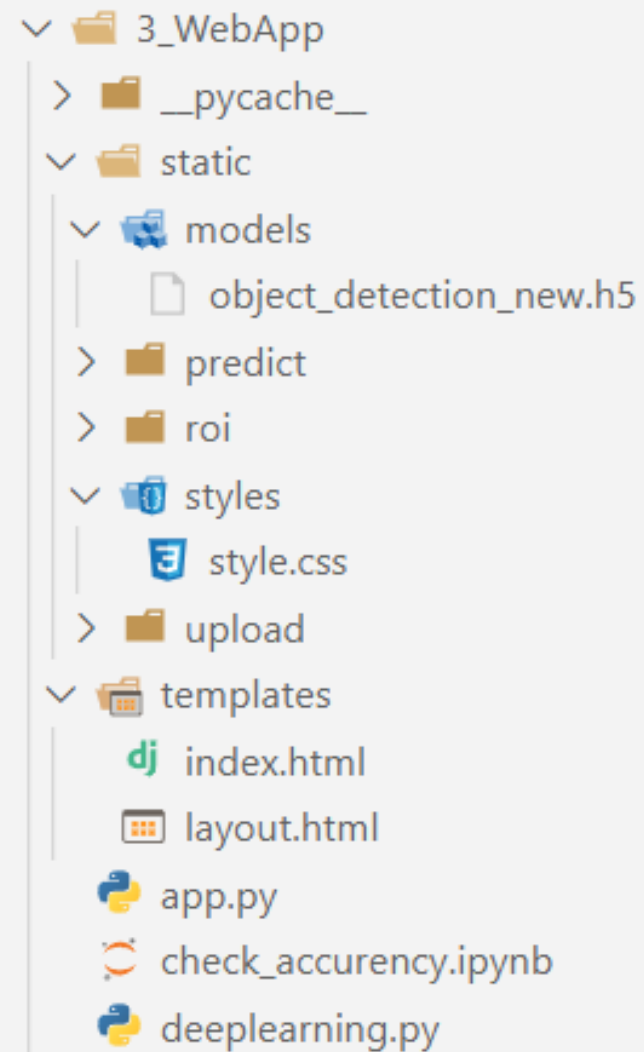
# XÂY DỰNG TRANG WEB DEMO

## Cơ sở dữ liệu

car_owner
id INT
name VARCHAR(250)
identity_id VARCHAR(45)
number_plate VARCHAR(45)
ticket_code VARCHAR(45)
ticket_type VARCHAR(250)
Indexes

parking_ledger
id INT
number_plate VARCHAR(45)
parking_fee VARCHAR(45)
ticket_type VARCHAR(250)
on_time DATETIME
way VARCHAR(45)
Indexes

## Cấu trúc thư mục



```
3_WebApp
├── __pycache__
├── static
│   ├── models
│   │   └── object_detection_new.h5
│   ├── predict
│   ├── roi
│   ├── styles
│   │   └── style.css
│   └── upload
├── templates
│   ├── index.html
│   └── layout.html
├── app.py
├── check_accurency.ipynb
└── deeplearning.py
```


# KẾT QUẢ CHƯƠNG TRÌNH

Giao diện trang web thu được có dạng

## THÔNG TIN BẾN ĐỖ XE

Tổng số chỗ	Số chỗ còn trống	Tổng lượt gửi trong ngày(vé ngày/vé đã đăng ký)	Tổng phí gửi xe đã thu trong ngày
100	42	6 (3/3)	30000 VND
Chọn tệp	Không có tệp nào được chọn		Vào Ra



Ảnh nhận diện biển số	Biển số	Loại vé	Chi phí
	51F61712	Vé tuần	0 VNĐ

## THÔNG TIN CHỦ XE

Họ và tên	CMND/CCCD	Biển xe	Mã vé
Trần Văn Nam	00341125432	51F61712	TK001

# KẾT QUẢ CHƯƠNG TRÌNH

## THÔNG TIN BẾN ĐỖ XE

5

Tổng số chỗ	Số chỗ còn trống	Tổng lượt gửi trong ngày(vé ngày/vé đã đăng ký)	Tổng phí gửi xe đã thu trong ngày
100	42	6 (3/3)	30000 VND

Chọn tệp Không có tệp nào được chọn

Vào Ra

1



2

Ảnh nhận diện biển số	Biển số
	51F61712

3

Loại vé	Chi phí
Vé tuần	0 VND

## THÔNG TIN CHỦ XE

4

Họ và tên	CMND/CCCD	Biển xe	Mã vé
Trần Văn Nam	00341125432	51F61712	TK001

# KẾT QUẢ CHƯƠNG TRÌNH

Tỉ lệ nhận diện ký tự chính xác: **97 %**

Tỉ lệ số biến nhận dạng đúng trên tổng số biến: **86 %**

Thời gian nhận diện trung bình: **314 ms**

Ký tự thực tế	Ký tự nhận dạng	Tỉ lệ chính xác	Thời gian (ms)
29A23950	29A23990	87,5 %	313
50LD04411	50LD04411	100 %	320
29A51796	29A51796	100 %	407
48A05177	48A05177	100 %	287
50Z1630	50Z1630	100 %	297
51A01204	51A01204	100 %	296
51A05277	51A05277	100 %	283
51A55433	51A55433	100 %	402
51A65474	51A65474	100 %	307
51A69172	151A69172	88,9 %	408
51A72078	51A72078	100 %	293
51A75932	51A75932	100 %	390
51A89714	51A89714	100 %	398
51A90905	51A90905	100 %	388
51A96141	51A96141	100 %	306
51B21666	51B21666	100 %	288
51F04877	51F04877	100 %	280
51F06532	51F06532	100 %	323
51F06609	51F06609	100 %	302
51F06948	51F06948	100 %	306
51F07973	51F07973	100 %	292
51F15380	51F15380	100 %	303
51F15585	51F15585	100 %	304
...	...	...	...
Trung bình	Tỉ lệ số biến nhận dạng đúng trên tổng số biến: <b>86%</b>	Tỉ lệ nhận diện ký tự chính xác: <b>97%</b>	Thời gian nhận diện: <b>314 ms</b>

# KẾT QUẢ CHƯƠNG TRÌNH

Kết quả nhận diện ký tự chính xác

Nhầm ký tự 5 thành 9

Ảnh nhận diện biển số      Biển số

---



51F24403

Ảnh nhận diện biển số      Biển số

---



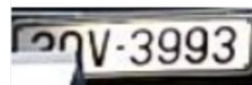
29A23990

Kết quả nhận diện khi biển số bị che



Ảnh nhận diện biển số

---



Biển số

---

FN3993



# 4

## KẾT LUẬN VÀ PHƯƠNG HƯỚNG PHÁT TRIỂN

## Thành quả:

- Tìm hiểu về xử lý ảnh, các khái niệm, thao tác tiền xử lý ảnh cơ bản.
- Nghiên cứu về mạng Nơ ron tích chập CNN và ứng dụng để tạo nên một mô hình học sâu xác định đối tượng, cụ thể ở đây là xác định vùng ảnh biển số xe
- Ứng dụng thư viện Tesseract vào việc nhận dạng ký tự quang học.
- Tiến hành thực nghiệp trên bộ dữ liệu ảnh chụp biển số xe ô tô, từ đó xây dựng nên một website nhỏ hỗ trợ quản lý bãi đỗ xe thông minh bằng cách sử dụng nhận dạng biển số xe ô tô, giúp rút ngắn thời gian chờ đợi, giảm nhân công, tăng năng suất.
- Phù hợp cho đối tượng sinh viên, người mới muốn tìm hiểu căn bản về xử lý ảnh, học sâu và nhận dạng ký tự quang học.

## Hạn chế:

- Đối một hệ thống có thể đưa ra sử dụng thực tế thì tỉ lệ chính xác vẫn còn hơi thấp
- Chưa tích hợp với camera
- Với những bức ảnh xô lệch, mờ, khác dữ liệu huấn luyện thì kết quả nhận diện chưa tốt



- Nâng cao tỉ lệ nhận diện chính xác, có thể áp dụng những thư viện hoặc thuật toán nhận diện ký tự quang khác như: VietOCR, CNN, ...
- Tích hợp với camera để hệ thống thực sự tự động, hoạt động theo thời gian thực
- Nâng cao chất lượng bộ dữ liệu và mô hình huấn luyện để có thể đa dạng đối tượng nhận diện hơn như: biển xe ô tô 2 dòng, biển xe máy, biển màu ...
- Kết hợp vào những chương trình, hệ thống khác để quản lý kho bãi hoặc sử dụng cho những mục đích quản lý, truy vết, tìm kiếm hiệu quả hơn.

- [1] Nguyễn Văn Tú, Hoàng Thị Lam và Nguyễn Thị Thanh Hà, "Nhận dạng chữ số viết tay dựa trên mạng Nơ ron tích chập sâu" Tạp chí Khoa học Tự nhiên và Công nghệ, vol. 14, no. 1, p. 118, 2019.
- [2] Shervine Amidi and Afshine Amidi, "Convolutional Neural Networks," Stanford University, California, 2020.
- [3] Nguyễn Quang Hoan, Phạm Ngọc Hưng và Nguyễn Đình Tài, "Nhận dạng ký tự viết tay sử dụng mạng Nơ ron tích chập" Tạp chí Khoa học và Công nghệ, vol. 24, no. 1, p. 5, 2019.
- [4] "InceptionResNetV2," MathWorks, 2017. [Online]. Available: <https://www.mathworks.com/help/deeplearning/ref/inceptionresnetv2.html>.
- [5] "Tổng hợp các loại biển số xe tại Việt Nam hiện nay," Thư viện pháp luật, 2022. [Online]. Available: <https://thuvienphapluat.vn/chinh-sach-phap-luat-moi/vn/thoi-su-phap-luat/tu-van-phap-luat/42297/tong-hop-cac-loai-bien-so-xe-tai-viet-nam-hien-nay>.



**CẢM ƠN!**

- Tỷ lệ nhận diện ký tự chính xác  $Acc = \frac{\text{Số lượng ký tự dự đoán đúng}}{\text{Tổng số các ký tự}}$
- Tỷ lệ dự đoán biến số xe đúng  $Acc = \frac{\text{Số lượng biến dự đoán đúng}}{\text{Tổng số biến}}$