

BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO CÔNG NGHỆ MỚI
ĐỀ TÀI NHẬN DIỆN KHUÔN MẶT ĐỀ ĐIỂM DANH

Môn học: Công nghệ mới

Giảng viên: Trương Văn Thông

Họ và tên		MSSV
Lê Văn Đức	Anh	19483571
Trần Công	Nguyên	19473081
Lê Trần Anh	Tuấn	19529271

TP. Hồ Chí Minh, ngày 09 tháng 12 năm 2022

MỞ ĐẦU

Khuôn mặt đóng vai trò quan trọng trong xã hội bởi nó cho thấy nhiều tác dụng như biểu thị cảm xúc , tuổi tác , giới tính, ... Và một tác dụng quan trọng hơn hết là để nhận biết người đối diện là ai . Bởi vậy bài toán nhận diện khuôn mặt ngày càng được quan tâm và ứng dụng vào 1 số công việc thực tiễn như điểm danh, chấm công ,điều tra ,...

Bài toán này cũng dần được áp dụng trong ngành giáo dục. Khi nhận thấy việc điểm danh cho từng sinh viên / học sinh bằng cách thủ công quá chậm chạp và mất thời gian. Trong khi những ứng dụng để điểm danh khác lại có một số khuyết điểm. Nhận diện khuôn mặt ngày càng trở thành ứng dụng quan trọng và tiện lợi trong việc điểm danh. Ứng dụng sẽ phát hiện người có đi học hay không . Nó giúp cho các thầy cô có thể kiểm soát chặt chẽ và mất ít thời gian hơn trong công tác điểm danh sinh viên.

Với sự phát triển nhanh chóng , thị giác máy tính đã đạt được những tiến bộ đáng kể trong những năm gần đây về nhận dạng và phát hiện đối tượng. Với việc xử lý hàng loạt những bài toán trong cuộc sống như nhận dạng hình ảnh, tô vẽ ảnh ,... . Hệ thống nhận diện khuôn mặt cũng có nhiều bước phát triển với nhiều mô hình được xây dựng bởi các chuyên gia như vgg16, resnet, facenet, googlenet ,... đạt hiệu quả cao.

MỤC LỤC

CHƯƠNG 1 TỔNG QUAN ĐỀ TÀI.....	3
1.1 Mục đích của đề tài.....	3
1.2 Đối tượng nghiên cứu.....	3
1.3 Phương pháp nghiên cứu.....	3
CHƯƠNG 2 CƠ SỞ LÝ THUYẾT.....	4
1. Face Detection :.....	4
2. Xác định danh tính :.....	4
3. REACT JS.....	4
3. MongoDB.....	6
5. Mô hình InceptionResnet-v2.....	8
6. Mô hình VGG16.....	9
CHƯƠNG 3 MÔ HÌNH XÂY DỰNG.....	11
3.1 Mô hình nhận diện InceptionResNet-v2 :.....	11
3.2 Mô hình nhận diện InceptionResNet-v2 :.....	12
3.3 Các thông số đánh giá giải thuật.....	12
CHƯƠNG 4 THỰC NGHIỆM.....	14
4.1 Dữ liệu đầu vào.....	14
4.1.1 Kết quả thu thập dữ liệu:.....	14
4.1.2 Tiền xử lý dữ liệu.....	14
4.2 Chọn và train mô hình:.....	14
4.3 Kết quả huấn luyện và đánh giá mô hình phân loại:.....	15
CHƯƠNG 5 DEMO.....	16
CHƯƠNG 6 KẾT LUẬN.....	17
TÀI LIỆU THAM KHẢO.....	18

CHƯƠNG 1 TỔNG QUAN ĐỀ TÀI

1.1 Mục đích của đề tài

Xây dựng hệ thống nhận diện khuôn mặt dùng để điểm danh thông qua việc train model để nhận diện khuôn mặt đó là ai. Với cơ sở dữ liệu đã được nhóm thu thập là các ảnh các thành viên trong nhóm và 1 unknow (nhiều ảnh của người không biết)

1.2 Đối tượng nghiên cứu

Nghiên cứu thông tin khuôn mặt của các sinh viên/ học sinh .

1.3 Phương pháp nghiên cứu

Phân tích các đặc trưng trong khuôn mặt và thử nghiệm model InceptionResNet-v2.

CHƯƠNG 2 CƠ SỞ LÝ THUYẾT

1. Face Detection :

Face detection hay facial detection là một công nghệ máy tính dựa trên trí tuệ nhân tạo (AI) được sử dụng để tìm và nhận dạng khuôn mặt người trong hình ảnh kỹ thuật số. Công nghệ nhận diện khuôn mặt có thể được áp dụng cho nhiều lĩnh vực khác nhau -- bao gồm bảo mật, sinh trắc học, thực thi pháp luật, giải trí và an toàn cá nhân -- để cung cấp khả năng giám sát và theo dõi mọi người trong thời gian thực.

Có nhiều phương pháp để phát hiện khuôn mặt như Harr cascades, HOG + Linear SVM, MTCNN, VGG16, yolo, ...

Chúng ta có thể thực hiện việc này qua các pre-train model có sẵn cũng như có thể tự build cho mình một mô hình mới. Nhưng việc này tốn nhiều thời gian và công sức do cần phải có 1 lượng lớn dữ liệu khuôn mặt (vài triệu) để đạt được mô hình càng tốt.

2. Xác định danh tính :

Nhận dạng khuôn mặt là một cách xác định hoặc xác nhận danh tính của một cá nhân bằng cách sử dụng khuôn mặt của họ. Hệ thống nhận dạng khuôn mặt có thể được sử dụng để xác định mọi người trong ảnh, video hoặc trong thời gian thực.

Nhận dạng khuôn mặt là một loại bảo mật sinh trắc học. Các dạng phần mềm sinh trắc học khác bao gồm nhận dạng giọng nói, nhận dạng dấu vân tay và nhận dạng võng mạc mắt hoặc mống mắt. Công nghệ này chủ yếu được sử dụng để bảo mật và thực thi pháp luật, mặc dù ngày càng có nhiều mối quan tâm đến các lĩnh vực sử dụng khác.

Một số phương pháp thường dùng (như 1 bài toán classification) : Model FaceNet dựa trên Siamese Network, Local Binary Patterns (LBPs), Eigenfaces, vgg16, resnet, ...

3. REACT JS

React là gì?

- React.JS là một thư viện Javascript dùng để xây dựng giao diện người dùng, nó không phải là 1 framework js nào hết.
- React hỗ trợ việc xây dựng những thành phần (components) UI có tính tương tác cao, có trạng thái và có thể sử dụng lại được.

- React được xây dựng xung quanh các component.
- React không chỉ hoạt động trên phía client, mà còn được render trên server và có thể kết nối với nhau...

Ưu điểm của ReactJS

- Thân thiện với SEO: SEO là phần quan trọng không thể thiếu đưa những thông tin trong website của bạn được lên top đầu của Google.
- Phù hợp với đa dạng thể loại website: Thư viện ReactJS ra đời khiến cho việc tạo ra một trang web dễ dàng hơn bởi vì người dùng không cần phải code nhiều như khi tạo trang web thuần khác sử dụng HTML, JavaScript.
- Debug dễ dàng: Ứng dụng Facebook đã phát hành một Chrome Extension dùng để debug trong quá trình phát triển ứng dụng. Điều này giúp tăng tốc quá trình phát triển của sản phẩm.
- Tái sử dụng các Component: Nếu người dùng có thể xây dựng một Component đủ tốt và linh hoạt để đáp ứng nhu cầu của từng dự án khác nhau, bạn chỉ tốn thời gian xây dựng ban đầu và hầu như có thể sử dụng lại toàn bộ ở các dự án sau
- Có thể sử dụng cho cả Mobile Application: Nếu như người dùng muốn phát triển thêm ứng dụng trên di động, thì họ có thể sử dụng thêm React Native-Một framework cũng cùng được phát triển trên nền tảng Facebook

Thành phần cơ bản của Reactjs là gì?

Thành phần cơ bản của React được gọi là components. Syntax để viết HTML sử dụng Javascript để render. Bạn có thể tạo ra một component bằng các gọi phương thức `createClass` của đối tượng React, điểm bắt đầu khi tiếp cận với thư viện này. Có thể lồng nhiều component vào nhau thông qua lệnh `return` của phương thức `render`.

Trong một chương trình thì có rất nhiều các component, để đơn giản việc quản lý các component đó người ta sử dụng `redux`, `redux` giống như 1 cái kho chứa các component và khi dùng component nào thì chỉ cần gọi nó ra. Virtual DOM không được tạo ra bởi Reactjs nhưng lại được sử dụng rất nhiều. Đây là một chuẩn của W3C được dùng để truy xuất code HTML hoặc XML.

Các Virtual DOM sẽ được tạo ra khi chạy chương trình, đó là nơi chứa các component. Sử dụng DOM sẽ tiết kiệm được hiệu suất làm việc, khi có thay đổi gì Reactjs đều tính toán trước và việc còn lại chỉ là thực hiện chúng lên DOM.

Cách thức hoạt động của React

React hoạt động dựa trên HTML và Javascript. Hầu hết các phần trong React được viết bởi JSX để giúp việc tạo React components dễ dàng hơn. Trong React, người dùng sẽ được trải nghiệm:

- Tạo đại diện cho Dom bằng hàm `Elements`. Cú pháp HTML của code trong trường hợp này khá tương tự XML components

Là sự kết hợp giữa HTML và JavaScript. Người dùng cần lưu ý một số vấn đề sau:

- GameScore là đối tượng có 2 cặp prop
- scores=(Game Score): Là thuộc tính điểm được nhận các giá trị từ GameScores
- Là một khối XML đã được render lại trang
- Là số đếm dùng để hiển thị biểu thức số tương ứng với giá trị của nó

3. MongoDB

a. Khái niệm.

- MongoDB là một hệ quản trị cơ sở dữ liệu mã nguồn mở thuộc học NoSQL. Nó được thiết kế theo kiểu hướng đối tượng, các bảng trong MongoDB được cấu trúc rất linh hoạt, cho phép các dữ liệu lưu trữ trên bảng không cần tuân theo một cấu trúc nhất định nào cả (điều này rất thích hợp để làm big data).
- MongoDB lưu trữ dữ liệu theo hướng tài liệu (document), các dữ liệu được lưu trữ trong document kiểu JSON nên truy vấn sẽ rất nhanh.

Ưu điểm của MongoDB

Open Source:

- MongoDB là phần mềm mã nguồn mở miễn phí, có cộng đồng phát triển rất lớn

Hiệu năng cao:

- Tốc độ truy vấn (find, update, insert, delete) của MongoDB nhanh hơn hẳn so với các hệ quản trị cơ sở dữ liệu quan hệ (RDBMS).
- Thử nghiệm cho thấy tốc độ insert, tốc độ insert của MongoDB có thể nhanh tới gấp 100 lần so với MySQL (So sánh hiệu suất của MongoDB với MySQL)

Tại sao MongoDB có hiệu năng cao như thế? có các lý do sau:

- MongoDB lưu dữ liệu dạng JSON, khi bạn insert nhiều đối tượng thì nó sẽ là insert một mảng JSON gần như với trường hợp insert 1 đối tượng
- Dữ liệu trong MongoDB không có sự ràng buộc lẫn nhau như trong RDBMS, khi insert, xóa hay update nó không cần phải mất thời gian kiểm tra xem có thỏa mãn các bảng liên quan như trong RDBMS.
- Dữ liệu trong MongoDB được đánh chỉ mục (đánh index) nên khi truy vấn nó sẽ tìm rất nhanh.
- Khi thực hiện insert, find... MongoDB sẽ khóa các thao tác khác lại, ví dụ khi nó thực hiện find(), trong quá trình find mà có thêm thao tác insert, update thì nó sẽ dừng hết lại để chờ find() xong đã.

Dữ liệu linh hoạt:

- MongoDB là document database, dữ liệu lưu dưới dạng JSON, không bị bó buộc về số lượng field, kiểu dữ liệu... bạn có thể insert thoải mái dữ liệu mà mình muốn.

Là Rich Query Language:

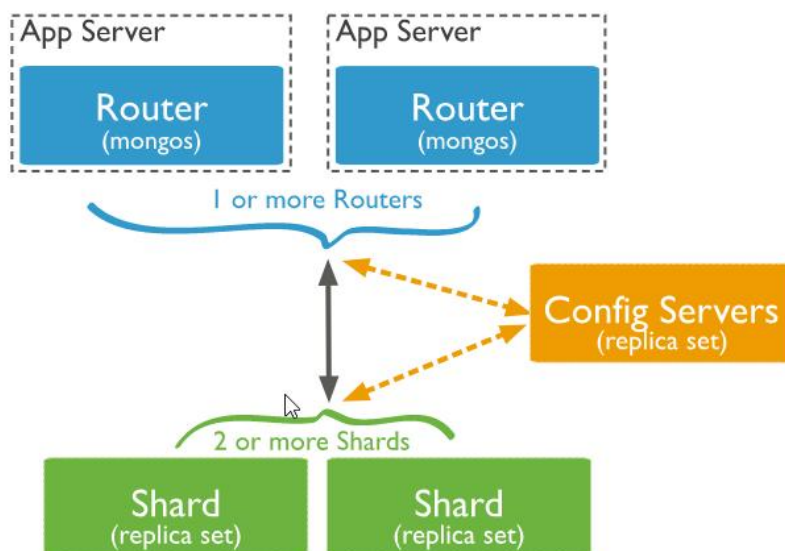
- MongoDB là một rich query language tức là nó có sẵn các method để thực hiện create, read, update, delete dữ liệu (CRUD)

Tính sẵn có:

- MongoDB hỗ trợ replica set nhằm đảm bảo việc sao lưu và khôi phục dữ liệu

Khả năng mở rộng Horizontal Scalability:

- Trong MongoDB có một khái niệm cluster là cụm các node chứa dữ liệu giao tiếp với nhau, khi muốn mở rộng hệ thống ta chỉ cần thêm một node vào vào cluster:



Nhược điểm của MongoDB

- MongoDB không có các tính chất ràng buộc như trong RDBMS → dễ bị làm sai dữ liệu
- Không hỗ trợ join giống như RDBMS nên khi viết function join trong code ta phải làm bằng tay khiến cho tốc độ truy vấn bị giảm.
- Sử dụng nhiều bộ nhớ: do dữ liệu lưu dưới dạng key-value, các collection chỉ khác về value do đó key sẽ bị lặp lại. Không hỗ trợ join nên sẽ bị dư thừa dữ liệu (trong RDBMS thì ta chỉ cần lưu 1 bản ghi rồi các bản ghi khác tham chiếu tới còn trong MongoDB thì không)

- Bị giới hạn kích thước bản ghi: mỗi document không được có kích thước > 16Mb và không mức độ các document con trong 1 document không được > 100

MongoDB doesn't support joins like a relational database. Yet one can use joins functionality by adding by coding it manually. But it may slow execution and affect performance.

Khi nào nên dùng MongoDB

MongoDB dùng cho các hệ thống:

Hệ thống realtime (thời gian thực) yêu cầu phản hồi nhanh

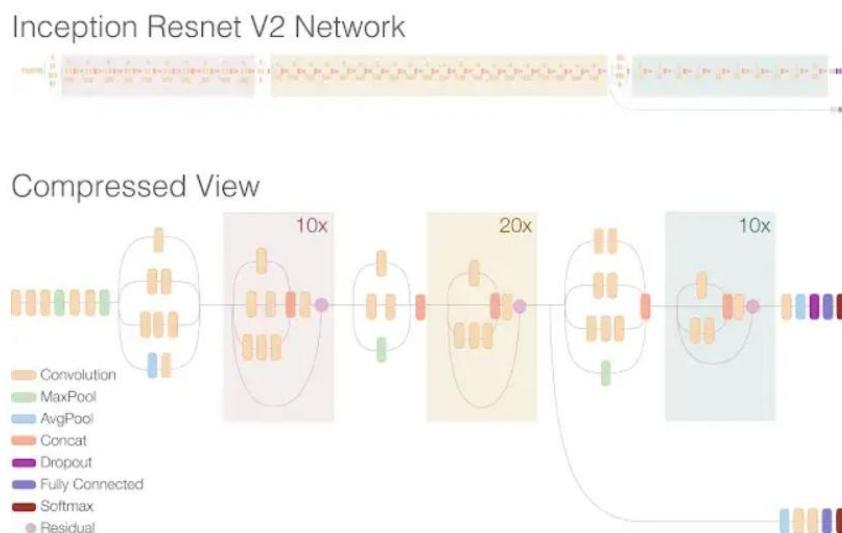
Các hệ thống bigdata với yêu cầu truy vấn nhanh.

Các hệ thống có tần suất write/insert lớn

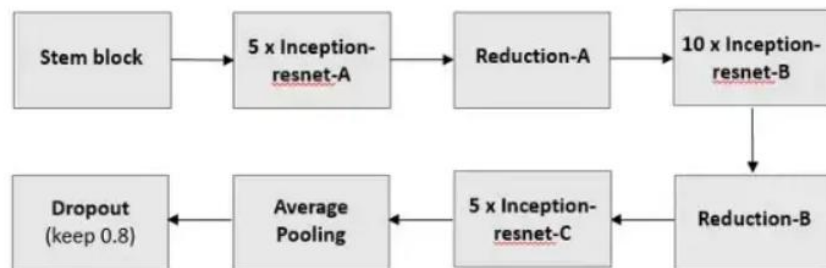
Sử dụng làm search engine.

5. Mô hình InceptionResnet-v2

Inception-ResNet-v2 là mạng thần kinh tích chập được đào tạo trên hơn một triệu hình ảnh từ cơ sở dữ liệu ImageNet. Mạng có độ sâu 164 lớp và có thể phân loại hình ảnh thành 1000 loại đối tượng, chẳng hạn như bàn phím, chuột, bút chì và nhiều loài động vật. Kết quả là, mạng đã học được các biểu diễn tính năng phong phú cho nhiều loại hình ảnh. Mạng có kích thước đầu vào tối đa hình ảnh là 299 x 299 và đầu ra là danh sách các xác suất ước tính của lớp.



Nó dựa trên định dạng dựa trên sự kết hợp của cấu trúc Inception và kết nối Residual. Trong khối Inception-Resnet, nhiều bộ lọc chập có kích thước được kết hợp với các kết nối còn lại. Việc sử dụng các kết nối còn lại không chỉ tránh được vấn đề xuống cấp do cấu trúc sâu gây ra mà còn giảm thời gian đào tạo. Hình này cho thấy kiến trúc mạng cơ bản của Inception-Resnet-v2.



The basic architecture of Inception-Resnet-v2.

Keras Implementation:

Inception-ResNet v2 model, with weights trained on ImageNet

```

application_inception_resnet_v2(include_top = TRUE,
    weights = "imagenet", input_tensor = NULL, input_shape = NULL,
    pooling = NULL, classes = 1000)

inception_resnet_v2_preprocess_input(x)
  
```

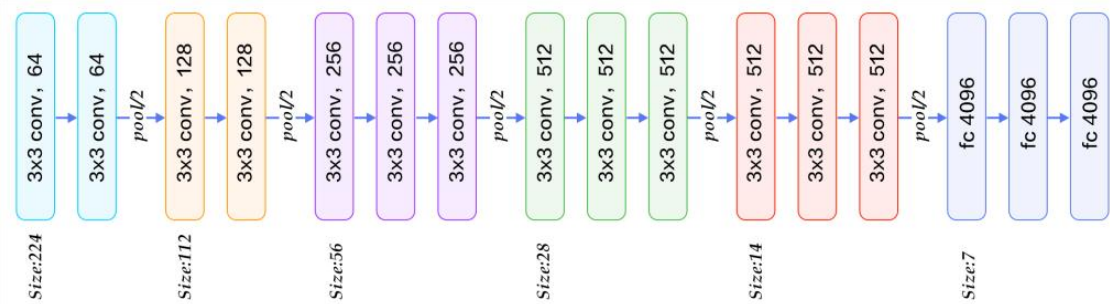
TensorFlow Implementation:

```

tf.keras.applications.InceptionResNetV2(
    include_top=True, weights='imagenet', input_tensor=None, input_shape=None,
    pooling=None, classes=1000, classifier_activation='softmax', **kwargs
)
  
```

6. Mô hình VGG16

VGG16 là mạng convolutional neural network được đề xuất bởi K. Simonyan and A. Zisserman, University of Oxford. Model sau khi train bởi mạng VGG16 đạt độ chính xác 92.7% top-5 test trong dữ liệu [ImageNet](#) gồm 14 triệu hình ảnh thuộc 1000 lớp khác nhau. Giờ áp dụng kiến thức ở trên để phân tích mạng VGG 16.

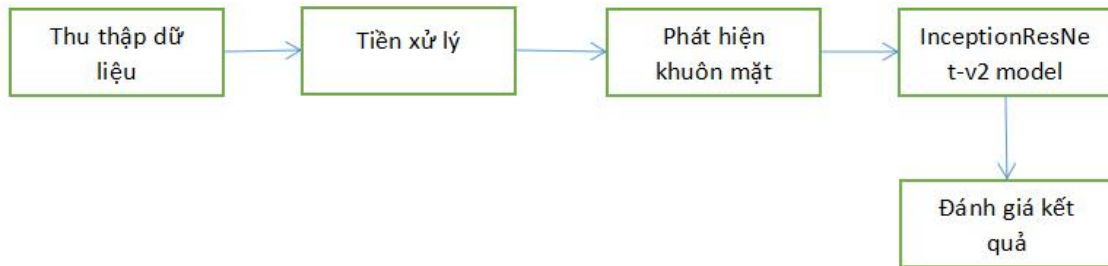


Phân tích:

- Convolutional layer: kích thước 3*3, padding=1, stride=1. Tại sao không ghi stride, padding mà vẫn biết? Vì mặc định sẽ là stride=1 và padding để cho output cùng width và height với input.
- Pool/2 : max pooling layer với size 2*2
- 3*3 conv, 64: thì 64 là số kernel áp dụng trong layer đấy, hay depth của output của layer đấy.
- Càng các convolutional layer sau thì kích thước width, height càng giảm nhưng depth càng tăng.
- Sau khá nhiều convolutional layer và pooling layer thì dữ liệu được flatten và cho vào fully connected layer.

CHƯƠNG 3 MÔ HÌNH XÂY DỰNG

3.1 Mô hình nhận diện InceptionResNet-v2 :



Bước 1 : Cắt ảnh từ clip thông qua tools (1 giây được 1 ảnh)

Bước 2 : Tiến hành detect face bằng tay hoặc công cụ có sẵn.

Bước 3 : Load ảnh lên thông qua chia package tránh tràn ram, sau đó tiền xử lý ảnh (chuyển ảnh xám , đầu vào model trùng với kích thước ảnh, ...)

Bước 4 : Chia tập dữ liệu thành train, test, val và tạo model InceptionResNet-v2 thông qua keras.applications

Bước 5 : Tiến hành train model và đánh giá kết quả.

Kiến trúc InceptionResNet-v2

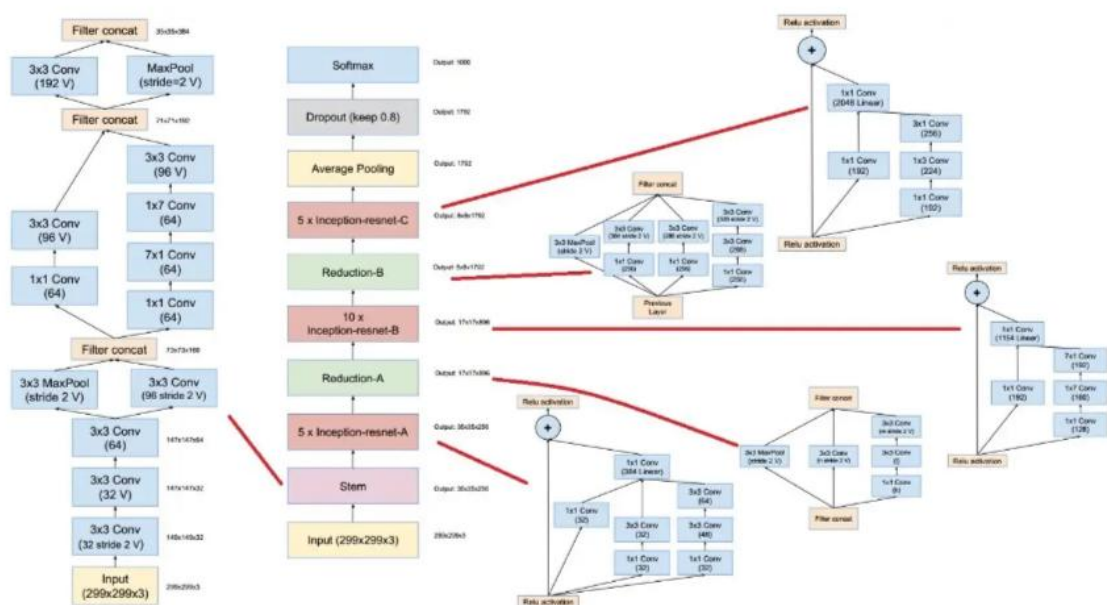
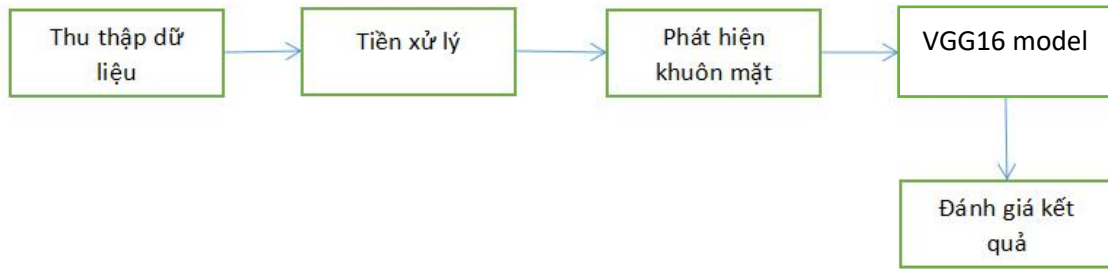


Image taken from [yeeephycho](#)

3.2 Mô hình nhận diện InceptionResNet-v2 :



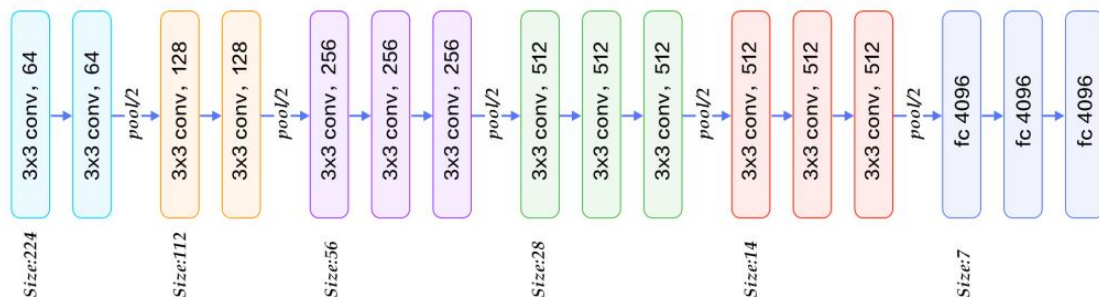
Bước 1 : Cắt ảnh từ clip thông qua tools (1 giây được 1 ảnh)

Bước 2 : Tiến hành detect face bằng tay hoặc công cụ có sẵn.

Bước 3 : Load ảnh lên thông qua chia package tránh tràn ram, sau đó tiền xử lý ảnh (chuyển ảnh xám , đầu vào model trùng với kích thước ảnh, ...)

Bước 4 : Chia tập dữ liệu thành train, test, val và tạo model VGG16 thông qua keras.applications

Bước 5 : Tiến hành train model và đánh giá kết quả.



3.3 Các thông số đánh giá giải thuật

Nghiên cứu sử dụng phương pháp đánh giá mô hình phân loại là dựa trên các chỉ số tính toán trong ma trận nhầm lẫn (Confusion Matrix) như bảng 3.1.

	Thực tế: Positive	Thực tế: Negative
Dự đoán: Positive	True Positive (TP)	False Negative (FN)
Dự đoán: Negative	False Positive (FP)	True Negative (TN)

Bảng 3.1 Ma trận nhầm lẫn (Confusion Matrix)

Hiệu quả của mô hình phân loại nhận xét được đánh giá dựa trên 4 chỉ số: Độ chính xác (Accuracy), Độ hội tụ (Precision), Độ bao phủ (Recall) và Giá trị trung bình điều hòa (F1). Ngoài ra, nghiên cứu này cũng xét đến yếu tố thời gian huấn luyện (Time) của từng mô hình.

Trong đó:

$$\text{Accuracy} = \frac{TN + TP}{TN + TP + FP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

CHƯƠNG 4 THỰC NGHIỆM

4.1 Dữ liệu đầu vào

4.1.1 Kết quả thu thập dữ liệu:

Bộ dữ liệu được nhóm em tạo nên bao gồm 5 lớp trong đó có 4 bạn sinh viên và 1 unknow(chứa nhiều ảnh không biết là ai). Mục tiêu là điểm danh xem sinh viên đó có đi học hay không sau khi sử dụng hệ thống nhận diện khuôn mặt.

Mô tả dữ liệu:

- Số lượng dữ liệu: 3685 ảnh

- Số lượng lớp : 6

Số lớp	Chú thích	Số ảnh
0_undefined	Nhiều người không biết	546
1_le-van-duc-anh	Lê Văn Đức Anh	1010
2_tran-cong-nguyen	Trần Công Nguyên	980
3_tran-huy-hoang	Trần Huy Hoàng	242
4_tran-minh-chien	Trần Minh Chiến	260
5_le-tran-anh-tuan	Lê Trần Anh Tuấn	647

4.1.2 Tiền xử lý dữ liệu

```
def pre_processing_image(img:np.ndarray):  
    img_result = cv2.cvtColor(img, cv2.COLOR_BGR2RGB) # Chuyển đ  
    img_result = cv2.resize(img, dsize=(IMAGE_SIZE, IMAGE_SIZE))  
    img_result = img_result/255 # [0->255] => [0->1]  
    return img_result
```

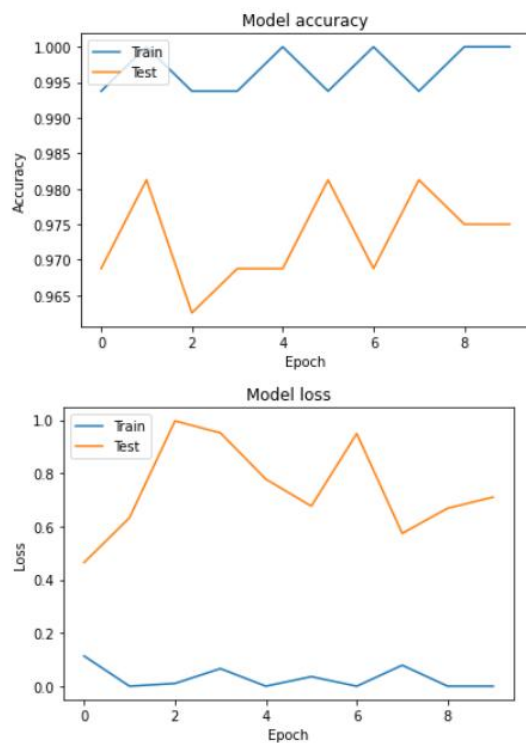
4.2 Chọn và train mô hình:

Tiến hành chia dữ liệu thành tập train, test và val với tập train = 80% và tập test=20% để huấn luyện mô hình

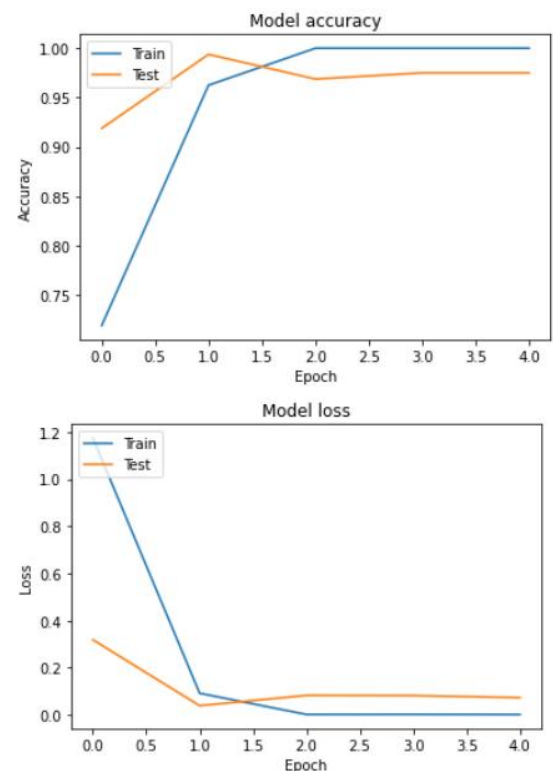
Trong nghiên cứu này sẽ dùng mô hình là InceptionResnet-v2 để thực hiện nhận diện khuôn mặt.

4.3 Kết quả huấn luyện và đánh giá mô hình phân loại:

STT	Tên mô hình	Accuracy	Loss
1	InceptionResNet-v2	0.99	<0.04
2	VGG16	0.99	<0.05



Inceptionresnet-v2



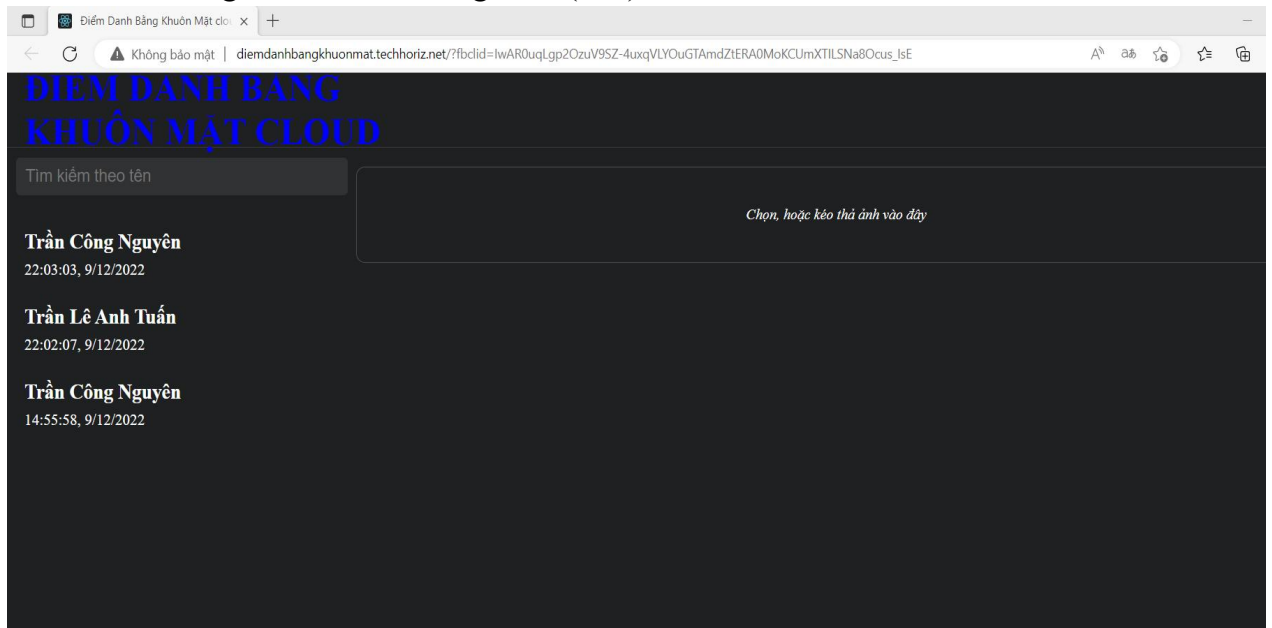
VGG16

Từ kết quả ta thấy được, mô hình InceptionResnet_v2 và VGG16 đều có độ chính xác cao và thích hợp để triển khai lên web.

CHƯƠNG 5 DEMO

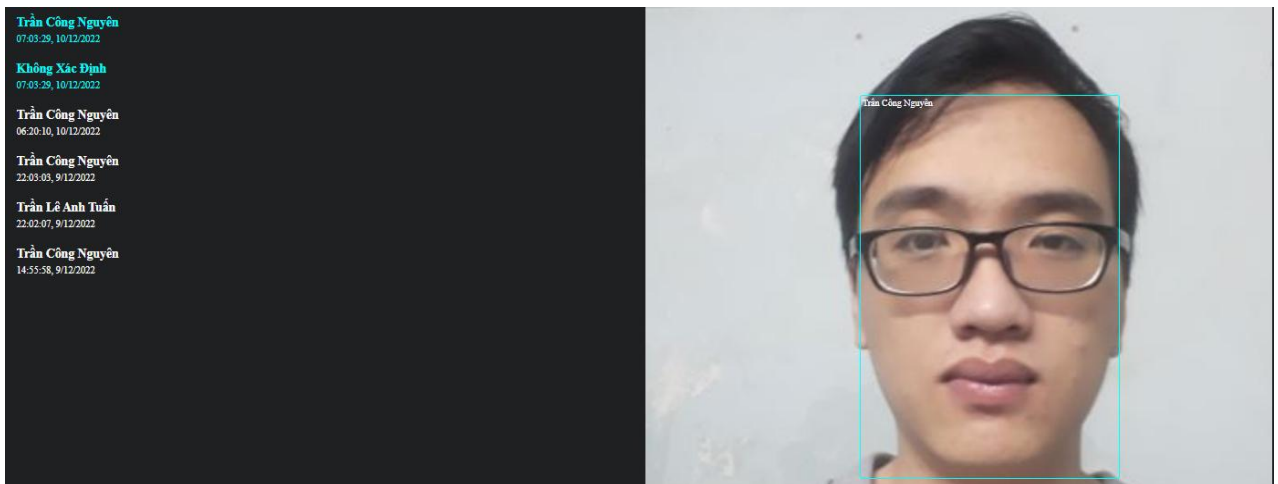
Giao diện trước khi nhận diện :

Bao gồm các chức năng chọn(kéo) ảnh, tìm kiếm theo tên sinh viên.



Giao diện sau khi nhận diện :

Hiện thị sinh viên được nhận diện là ai và lưu vào database.



CHƯƠNG 6 KẾT LUẬN

Đề tài này đã xây dựng ra mô hình thích hợp để điểm danh cho sinh viên là mô hình InceptionResNet-v2. Mô hình nhận diện khuôn mặt này đã được triển khai lên trang <http://diemdanhbangkhuonmat.techhoriz.net/> và trên máy ảo trên google cloud . Tuy nhiên, đề tài này vẫn còn nhiều hạn chế, do tập dữ liệu chưa tối ưu và 1 số chức năng còn thiếu sót như nhận diện bằng thời gian thực qua camera, phân quyền cho sinh viên và 1 số chức năng khác, chưa sử dụng nhiều mô hình . Nhóm sẽ tiếp tục cố gắng để có thể tiếp tục thực hiện trong thời gian tới hoặc trong những nghiên cứu tiếp theo.

Link demo đề tài : [Demo App Điểm Danh Bằng Khuôn Mặt - YouTube](#)

TÀI LIỆU THAM KHẢO

- [1]. https://keras.io/api/models/model_saving_apis/. (08/12/2022) (Save model)
- [2]. https://keras.io/api/models/model_training_apis/. (08/12/2022) (Build model)
- [3] More trainable inception-ResNet for face recognition Shuai Peng , Hongbo Huang , Weijun Chen , Liang Zhang , Weiwei Fang. (01/12/2022) (model inception-ResNet-v2)
- [4] https://vi.wikipedia.org/wiki/Hệ_thống_nhận_dạng_khuôn_mặt. (15/10/2022) (Tìm hiểu Face recognition)
- [5] <https://console.cloud.google.com/projectselector2/compute/instances?hl=vi&supportedpurview=project> (02/10/2022) (Tìm hiểu máy ảo trên cloud)
- [6] <https://medium.com/@zahraelhamraoui1997/inceptionresnetv2-simple-introduction-9a2000edcdb6> (02/10/2022) (InceptionResnet-v2)