

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN  
KHOA KHOA HỌC MÁY TÍNH**



**BÁO CÁO MÔN HỌC  
NHẬP MÔN THỊ GIÁC MÁY TÍNH  
CS231.M22.KHCL**

**ĐỀ TÀI:**

**NHẬN DIỆN NGƯỜI ĐEO KHẨU TRANG**

Giảng viên hướng dẫn: **TS Mai Tiến Dũng**

Sinh viên thực hiện: **Huỳnh Nguyễn Văn Khánh - 20521446**

**Trương Thị Thanh Thanh - 20520767**

**Đỗ Thị Thu Trang - 20520816**

**TP. HỒ CHÍ MINH, 06/2022**

## MỤC LỤC

<b>I. Giới thiệu đề tài:</b>	<b>1</b>
1. Ý tưởng giải quyết	1
2. Input & Output	1
<b>II. Thu thập dữ liệu:</b>	<b>2</b>
<b>III. Face Detection:</b>	<b>2</b>
1. Haarcascade	2
2. Kết quả	4
<b>IV. Mask Classification:</b>	<b>4</b>
1. Ý tưởng tổng quát của CNN	4
2. Keras	6
3. Kết quả	7
<b>V. Thực nghiệm:</b>	<b>7</b>
<b>VI. Tổng kết:</b>	<b>9</b>
1. Đánh giá	9
2. Ưu điểm	9
3. Nhược điểm	9
<b>VII. Nguồn tham khảo:</b>	<b>10</b>
<b>VIII. Bảng phân công công việc:</b>	<b>11</b>

## I. Giới thiệu đề tài:

Đại dịch COVID-19 là một đại dịch bệnh truyền nhiễm với tác nhân là virus SARS-CoV-2 diễn ra trên phạm vi toàn cầu. Khởi nguồn vào tháng 12 năm 2019, số ca nhiễm Covid 19 trên toàn thế giới đạt 544 triệu ca, trong đó có đến 6 triệu ca tử vong.

Tại Việt Nam, dịch bệnh đã được kiểm soát, mọi người đã quay lại với cuộc sống bình thường mới. Tuy nhiên, số ca nhiễm trong nước khá cao, trung bình 7 ngày đạt 904 ca nhiễm (ghi nhận từ ngày 7/7 đến ngày 13/7). Bên cạnh đó, tỷ lệ tiêm vaccin nhắc lại của nước ta chỉ đạt 30%, tỷ lệ tái nhiễm cao. Do đó, thực hiện quy tắc 5K là việc cần thiết, nhất là việc đeo khẩu trang.

Với những lý do trên, nhóm lựa chọn thực hiện đề tài “*Nhận diện người đeo khẩu trang*”.

### 1. Ý tưởng giải quyết

Với đề tài “Nhận diện người mang khẩu trang”, nhóm sẽ lần lượt giải quyết qua 2 bài toán nhỏ:

- **Face Detection:** Nhận diện khuôn mặt trên hình ảnh đầu vào.
- **Mask Classification:** Phân loại ảnh dựa trên khuôn mặt là kết quả của bài toán Face Detection thành 2 loại đeo khẩu trang hoặc không đeo khẩu trang.

### 2. Input & Output

- **Input:** Ảnh chân dung 1 người, chụp chính diện khuôn mặt, cận mặt.
- **Output:** Bức ảnh đã được đóng khung và xác định là có mang khẩu trang hay không (mask/ no mask).



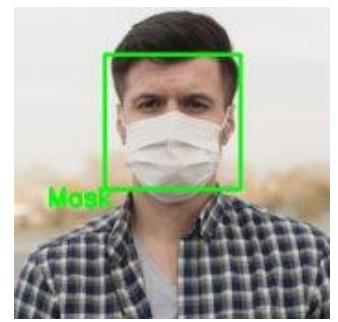
Input



Output



Input



Output

## II. Thu thập dữ liệu:

Bộ dữ liệu được nhóm thu thập từ nhiều nguồn. Hầu hết các bức ảnh được trích từ bộ dữ liệu, còn lại được lấy từ trên mạng và chụp trực tiếp. Bộ dữ liệu sẽ được chia làm 3 phần chính, mỗi phần sẽ gồm 2 file riêng biệt chứa các bức ảnh mô tả khuôn mặt đeo khẩu trang và khuôn mặt không đeo khẩu trang. 3 phần chính lần lượt là:

- **Train:** tập dữ liệu phục vụ quá trình huấn luyện mô hình.
- **Validation:** tập dữ liệu phục vụ cho việc kiểm thử độ chính xác của mô hình trong quá trình huấn luyện.
- **Test:** tập dữ liệu phục vụ quá trình thử nghiệm cuối cùng.

## III. Face Detection:

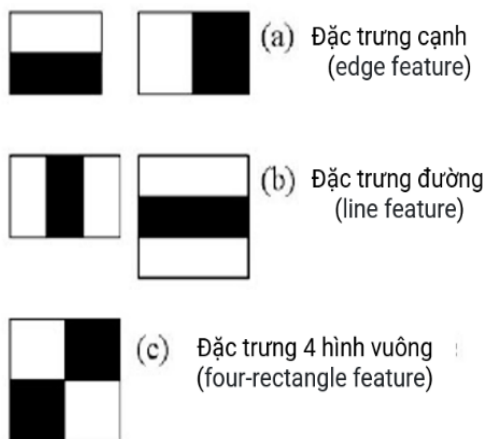
Face Detection là một bài toán cơ bản đã được xây dựng từ nhiều năm nay, có nhiều phương pháp được đưa ra như sử dụng template matching, neuron network,... Và với mục tiêu xác định được vị trí của khuôn mặt trong bức ảnh, nhóm lựa chọn thuật toán Haarcascade được cài đặt trong thư viện OpenCV.

### 1. Haarcascade

#### a. Giới thiệu Haarcascade

Haarcascade là thuật toán giúp nhận diện khuôn mặt bằng cách sử dụng các đặc trưng (tầng) Haar và sử dụng các đặc trưng đó qua nhiều lượt (cascade) để tạo thành một cỗ máy nhận diện khuôn mặt hoàn chỉnh.

#### b. Các loại đặc trưng Haar:



Sử dụng các đặc điểm hình chữ nhật (tính năng Haar) để xử lý hình ảnh và phát hiện khuôn mặt

Các đặc trưng được chạy lần lượt trên hình ảnh và giá trị thu được ở mỗi đặc trưng

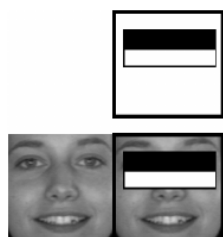
Giá trị của mỗi đặc trưng bằng tổng số pixel nằm trong phần màu đen trừ cho tổng số pixel nằm trong phần màu trắng.

$$f(x) = Tổng_{vùng\ đen}(pixel) - Tổng_{vùng\ trắng}(pixel)$$

Vậy khi được đặt lên một vùng ảnh, đặc trưng Haar sẽ tính toán và đưa ra giá trị đặc trưng  $f(x)$  của vùng ảnh đó.

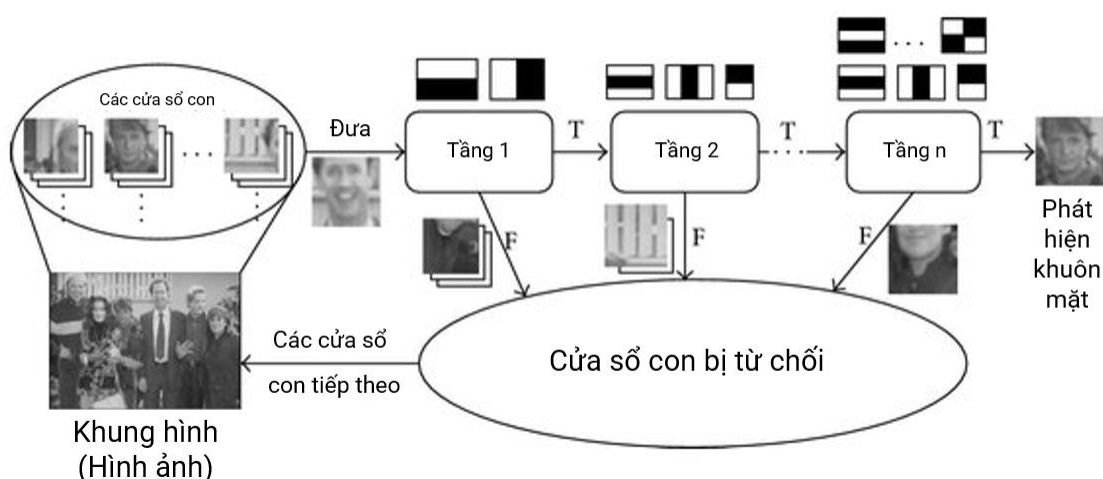
**Ví dụ:**

Đặc trưng Haar tính toán sự khác biệt về cường độ giữa 2 vùng mắt và má trên khuôn mặt dựa vào đặc trưng cạnh.



Đặc trưng Haar đo sự chênh lệch về cường độ giữa 2 vùng mắt và sống mũi trên khuôn mặt dựa vào đặc trưng đường. Đặc trưng Haar chỉ có thể nhìn cụ thể vào một vùng trong cửa sổ để nhận diện.

### c. Các bước nhận diện khuôn mặt:

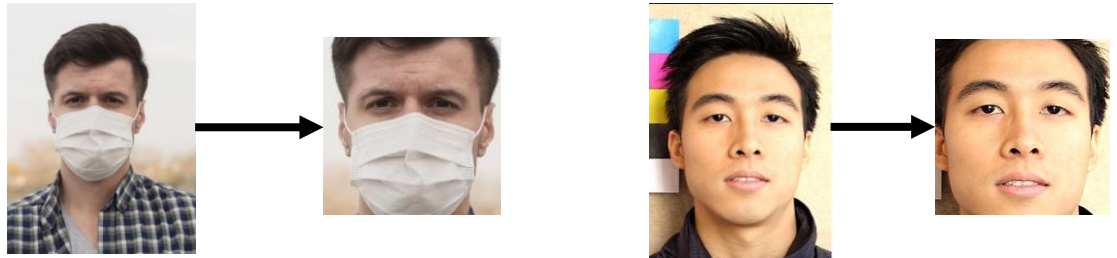


- **Bước 1:** Hình ảnh (đã được gửi đến bộ phân loại) được chia thành các phần nhỏ (hoặc các cửa sổ con).
- **Bước 2:** Chúng tôi đặt N không có bộ dò theo cách xếp tầng trong đó mỗi bộ phát hiện sự kết hợp của các loại đặc trưng khác nhau từ các hình ảnh (đường thẳng, cạnh, hình tròn, hình vuông) được truyền qua. Giả sử khi việc trích xuất đối tượng địa lý được thực hiện, mỗi phần phụ được gán một giá trị đặc trưng.

- **Bước 3:** Hình ảnh (hoặc hình ảnh phụ) có độ tin cậy cao nhất được phát hiện dưới dạng khuôn mặt và được gửi đến **bộ tích lũy** trong khi phần còn lại bị từ chối. Do đó, Cascade tìm nạp khung hình/hình ảnh tiếp theo nếu còn lại và bắt đầu lại quá trình.

## 2. Kết quả

Với đầu vào đã được nêu trên, kết quả của bài toán sau khi sử dụng thuật toán Haarcascade sẽ là hình ảnh khuôn mặt:



## IV. Mask Classification:

Ở bài toán Face Detection phía trên, khuôn mặt đã được xác định và đóng khung. Việc cần làm ở bài toán này là phân loại khuôn mặt vào một trong hai lớp: mask (đeo khẩu trang) và no-mask (không đeo khẩu trang). Đây là một bài toán phân loại, có thể áp dụng CNN (Convolutional Neural Network – Mạng Nơ-ron Tích chập) để giải quyết.

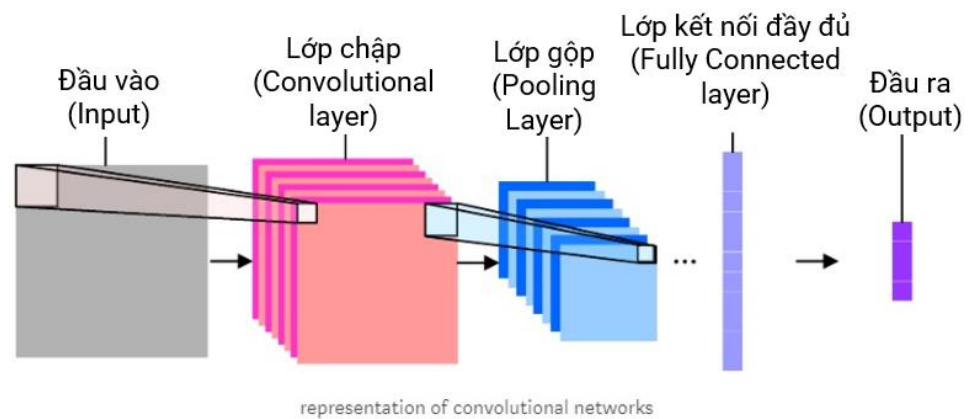
Trong mạng nơ-ron, mô hình mạng nơ-ron tích chập (CNN) là 1 trong những mô hình để nhận dạng và phân loại hình ảnh. Trong đó, xác định đối tượng và nhận dạng khuôn mặt là 1 trong số những lĩnh vực mà CNN được sử dụng rộng rãi.

### 1. Ý tưởng tổng quát của CNN

CNN phân loại hình ảnh bằng cách lấy 1 hình ảnh đầu vào, xử lý và phân loại nó theo các lớp nhất định (trong trường hợp này là mask và no-mask).

CNN sẽ áp dụng các bộ lọc lên ảnh trước khi huấn luyện mạng nơ-ron. Sau khi cho các tấm ảnh đi qua bộ lọc, những đặc trưng của tấm ảnh sẽ trở nên nổi bật, đồng cũng loại bỏ được các thông tin không cần thiết và chúng ta có thể dùng chúng để nhận diện hình ảnh.

Kiến trúc tổng quát của một mô hình CNN gồm có:



### a. Đầu vào (Input)

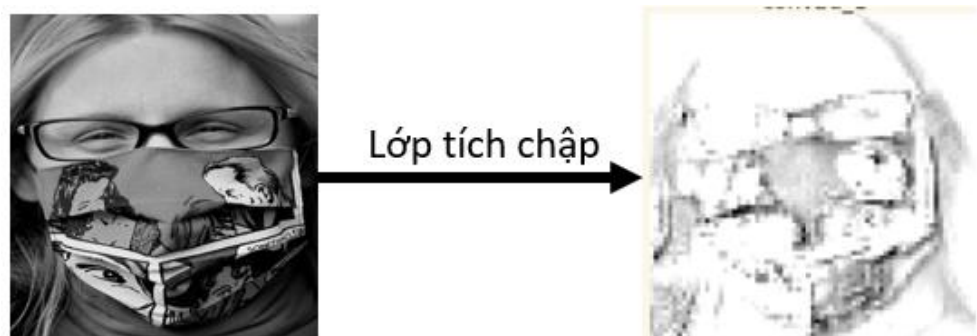
Máy tính coi hình ảnh đầu vào là 1 mảng pixel và kích thước của nó phụ thuộc vào độ phân giải của hình ảnh. Đầu vào sẽ là một mảng hai chiều nếu đó là ảnh xám, là một mảng 3 chiều (dài x rộng x 3) nếu đó là ảnh màu.

### b. Lớp tích chập (Convolution layer)

Dữ liệu đầu vào sẽ được tích chập với các bộ lọc (filter – có tên gọi khác là kernel hay nhân tích chập) có kích thước cho trước, tùy theo bộ lọc mà sau khi tích chập, chúng ta sẽ thu được các đặc trưng khác nhau từ ảnh (feature map). Ngoài ra các đặc trưng thu được còn phụ thuộc vào số lượng các bộ lọc tại lớp tích chập.

Giá trị của các bộ lọc không cố định mà sẽ được học qua quá trình huấn luyện, nhờ đó mà mô hình rút trích được các đặc trưng phù hợp.

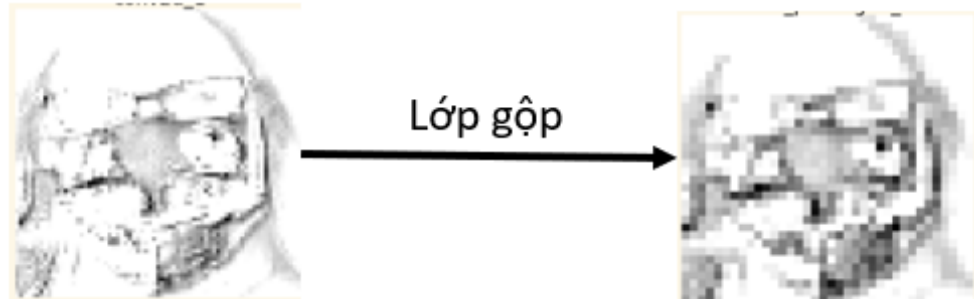
Một mô hình CNN có thể có nhiều lớp tích chập chồng lên nhau.





### c. Lớp gộp (Pooling layer)

Lớp gộp thường được dùng giữa các lớp chập, để giảm kích thước dữ liệu nhưng vẫn giữ được các thuộc tính quan trọng. Kích thước dữ liệu giảm giúp giảm việc tính toán trong mô hình.



### d. Lớp kết nối đầy đủ (Fully Connected layer)

Sau nhiều lớp tích chập và lớp gộp mô hình đã học được một số đặc trưng trên ảnh. Tuy nhiên lúc này dữ liệu vẫn ở dạng mảng 3 chiều. Trước khi truyền dữ liệu vào mạng nơ-ron (hay lớp kết nối đầy đủ), cần phải làm phẳng dữ liệu về dạng mảng một chiều.

Mạng nơ-ron sẽ đưa ra dự đoán nhãn cho ảnh (mask hay non-mask). Máy tính sẽ so sánh dự đoán này với nhãn đúng của các tấm ảnh trong tập huấn luyện và điều chỉnh tham số bộ lọc để cải thiện độ chính xác của mô hình.

Tùy thuộc vào yêu cầu bài toán mà có thể có một hoặc nhiều lớp kết nối đầy đủ.

## 2. Keras

Chúng ta hoàn toàn có thể xây dựng mô hình CNN từ đầu bằng Python, tuy nhiên sẽ khá tốn thời gian và công sức, vì đây là một thuật toán khá phức tạp. Chưa kể đến tùy thuộc vào người lập trình mà tốc độ của thuật toán có thể bị ảnh hưởng. Do đó, thông thường người ta sẽ sử dụng các thư viện hỗ trợ cho học sâu (deep learning).

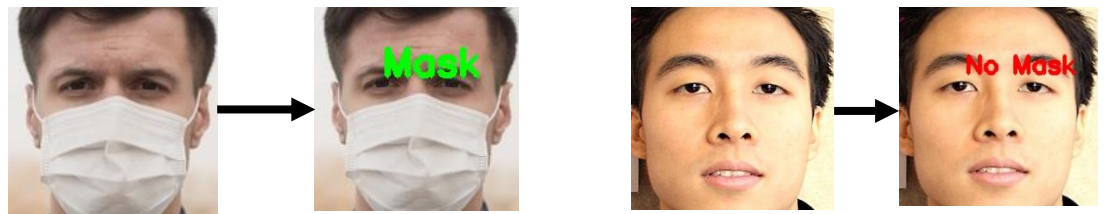


Keras là một framework mã nguồn mở cho học sâu được viết bằng Python, dễ sử dụng và thân thiện với người dùng. Nó có thể chạy trên nền của các framework khác như: tensorflow, theano, CNTK. Với các API bậc cao, dễ sử dụng, dễ mở rộng, keras giúp người dùng xây dựng các deep learning model một cách đơn giản.

Với keras việc xây dựng một mô hình học sâu (nói chung) và một mô hình mạng nơ-ron tích chập (nói riêng), giúp giảm nhẹ khối lượng công việc và tiết kiệm thời gian cho người lập trình.

### 3. Kết quả

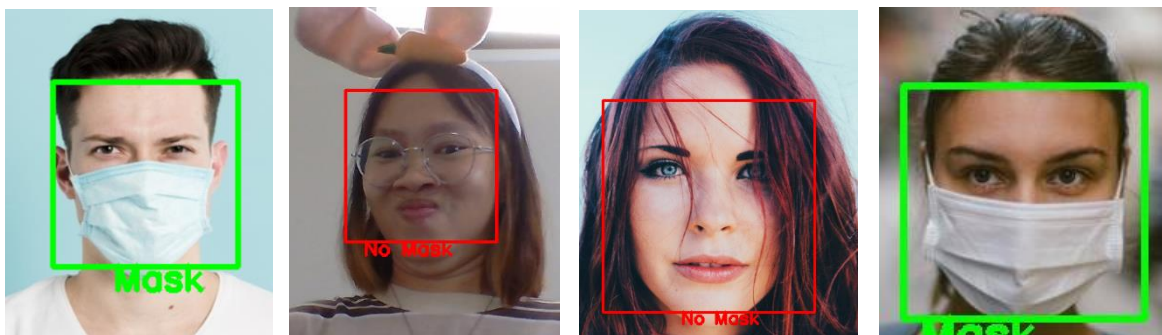
Với bức ảnh đầu vào là kết quả của bài toán Face Detection phía trên, đầu ra sẽ là khuôn mặt người đã được phân loại (mask/no mask).



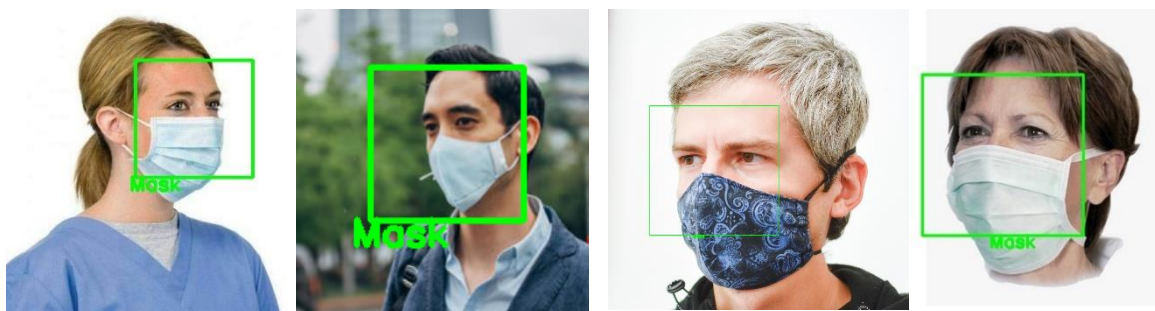
## V. Thực nghiệm:

Nhóm thực nghiệm trên bộ dữ liệu Test, bộ dữ liệu sẽ bao gồm những trường hợp đặc biệt so với Input đã nêu trên. Kết quả như sau:

Với những bức ảnh đúng yêu cầu Input, kết quả cho ra khá chính xác.



Với những bức ảnh không được chụp chính diện, được chụp với góc nghiêng nhỏ, mô hình cũng đã hoạt động tốt.



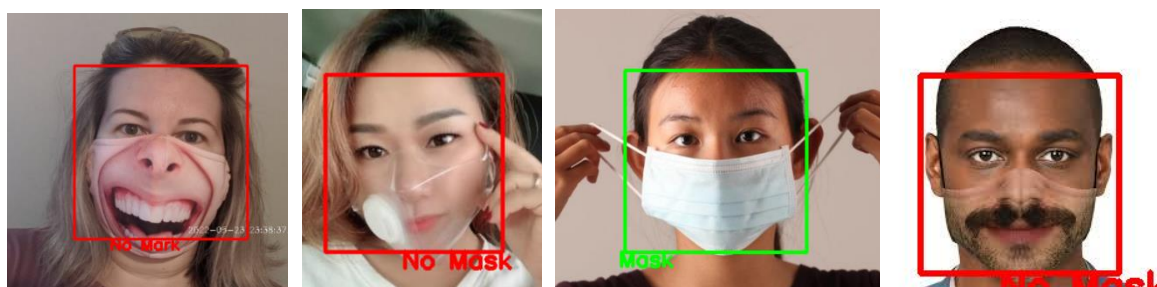
Ngoài ra, một số bức ảnh đeo khẩu trang đặc biệt, mặt nạ phòng độc hay những loại khẩu trang khác, mô hình cũng đã nhận ra được và cho ra những kết quả như sau:



Bên cạnh đó, mô hình có khả năng phát hiện những trường hợp mang khẩu trang sai cách, ví dụ như hở mũi, hở miệng, ...



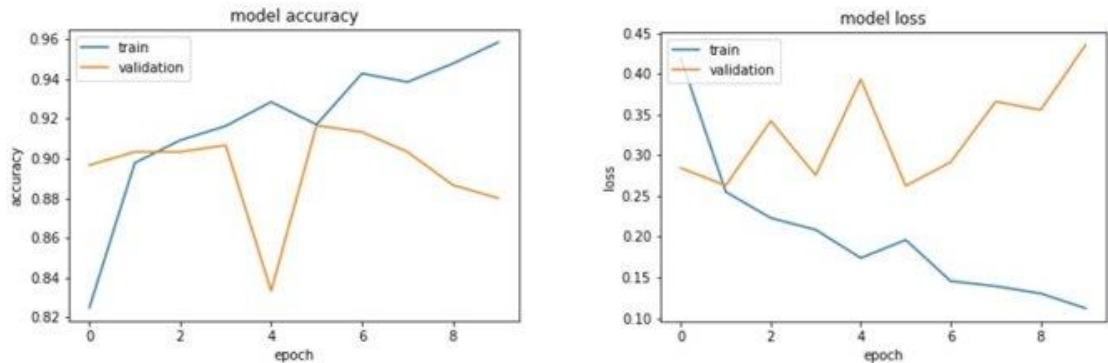
Tuy nhiên, mô hình vẫn còn nhiều sai sót, một số bức ảnh không được phân loại đúng như trường hợp mang khẩu trang trong suốt, ...



## VI. Tổng kết:

### 1. Đánh giá

Mô hình được đánh giá dựa trên độ chính xác (accuracy) và độ mất mát (loss). Sau 10 lần training, độ chính xác của mô hình đạt 88%, độ mất mát đạt 44%, con số này được tính dựa vào bộ dữ liệu Validation.



Để khách quan hơn, nhóm đã đánh giá dựa vào bộ dữ liệu Test. Với bộ dữ liệu này, độ chính xác đạt 86% và độ mất mát đạt 26%.

### 2. Ưu điểm

Với những kết quả nhận được sau phần thử nghiệm, mô hình đã đạt được những kết quả như sau:

- Model hoạt động tương đối tốt, có độ chính xác cao với yêu cầu input và output phía trên.
- Nhận diện được với những bức ảnh khuôn mặt có góc nghiêng nhỏ.
- Có khả năng nhận diện được người đeo khẩu trang sai như hở mũi, hở miệng, đều được xếp vào No -mask.

### 3. Nhược điểm

Bên cạnh những ưu điểm nêu trên, mô hình vẫn còn nhiều hạn chế cần cải thiện.

- Trong một số trường hợp như ảnh chụp cận mặt, khuôn mặt bị che phủ nhiều, mô hình không thể nhận diện được.
- Với một số loại khẩu trang như khẩu trang mặt người, khẩu trang trong suốt,...kết quả cho ra vẫn chưa chính xác.

## VII. Nguồn tham khảo:

- [1] Mohammad Farid Naufal, S. F. (2021). Comparative Analysis of Image Classification Algorithms for Face Mask Detection. Journal of Information Systems Engineering and Business Intelligence.
- [2] Pin Wang, E. F. (2020). Comparative Analysis of Image Classification Algorithms Based on Traditional Machine Learning and Deep Learning. Pattern Recognition Letters.
- [3] OpenCV: Cascade Classifier
- [4] Haar Cascade là gì? Hướng dẫn demo nhận diện khuôn mặt trong ảnh bằng Haar Cascade ONETECH Blogs
- [5] Phát hiện người đeo khẩu trang | Ứng dụng trong dịch Covid | Face Mask Detection (rabiloo.com)
- [6] Comparative analysis of image classification algorithms based on traditional machine learning and deep learning - ScienceDirect
- [7] #017 Face detection algorithms comparison - Master Data Science (datahacker.rs)
- [8] Thuật toán CNN - Convolutional Neural Network | TopDev
- [9] hongtin.net
- [10] Covid-19 Face Mask Detection Using TensorFlow, Keras and OpenCV | IEEE Conference Publication | IEEE Xplore
- [11] Face Mask Detection | Kaggle
- [12] Bộ Y tế - Cổng thông tin của Bộ Y tế về đại dịch COVID-19 (covid19.gov.vn)
- [13] Phát hiện người đeo khẩu trang trong thời gian thực (vjst.vn)

### **VIII. Bảng phân công công việc:**

- Dưới đây là bảng phân công công việc của nhóm.
- Đánh giá công việc được các thành viên trong nhóm đánh giá trong buổi họp tổng kết.

<b>STT</b>	<b>Họ và tên</b>	<b>Công việc</b>	<b>Đánh giá</b>
1	Huỳnh Nguyễn Vân Khánh	- Tìm hiểu đề tài, phương pháp giải quyết, chạy thử và tìm kiếm bộ dữ liệu. - Viết và chạy thử chương trình.	Hoàn thành
2	Trương Thị Thanh Thanh	- Tìm hiểu đề tài, phương pháp giải quyết, chạy thử và tìm kiếm bộ dữ liệu. - Viết và chạy thử chương trình.	Hoàn thành
3	Đỗ Thị Thu Trang	- Tìm hiểu đề tài, phương pháp giải quyết, chạy thử và tìm kiếm bộ dữ liệu. - Viết và chạy thử chương trình.	Hoàn thành