

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

KHOA KHOA HỌC MÁY TÍNH



# **BÁO CÁO ĐỒ ÁN CUỐI KỲ**

MÔN HỌC: MÁY HỌC

Đề tài: Nhận diện người có đeo khẩu trang hay không

Lớp: CS114.K21

Giảng viên hướng dẫn: GV: PGS.TS Lê Đình Duy - THS. Phạm Nguyễn Trường An

**Thành phố hồ chí minh, ngày 04, tháng 08, năm 2020**

## MỤC LỤC

<b>I/ GIỚI THIỆU ĐỀ TÀI:</b> .....	3
<b>II/CÁC BƯỚC THỰC HIỆN:</b> .....	3
1. Thu thập dữ liệu và gán nhãn:.....	3
2. Xử lý dữ liệu: .....	4
3. Chọn model và train:.....	5
4. Đánh giá:.....	6
5. Dự đoán: .....	7
6. Hướng phát triển: .....	8
<b>III.TÀI LIỆU THAM KHẢO:</b> .....	8

## I/ GIỚI THIỆU ĐỀ TÀI:

Trong tình trạng dịch covid đang ngày càng nghiêm trọng, thì việc mọi người đeo khẩu trang khi ra ngoài, hay khi tiếp xúc với người khác là điều hết sức quan trọng và cần thiết. Đặc biệt là tại các địa điểm có khả năng tụ tập, thu hút đông người tới như siêu thị, công sở, doanh nghiệp, trường học.

Một mô hình được tạo ra với yêu cầu là nhận diện được rằng một người đang đeo khẩu trang hay không đeo, để có thể đưa ra lời nhắc nhở, và yêu cầu cần thiết. Ở đây bài toàn của nhóm em chỉ làm về phần nhận diện một người có đeo khẩu trang hay không.

## II/CÁC BƯỚC THỰC HIỆN:

### 1. Thu thập dữ liệu và gán nhãn:

Lấy datasets trên nguồn:

<https://github.com/prajnasb/observations>

Ảnh down về chia làm 2 phần :

- with\_mask (đeo khẩu trang).
- without\_mask (không đeo khẩu trang).

Sau đó tiến hành gán nhãn dựa vào code:

<https://github.com/rainbow2125514/CS114.K21/blob/master/gannhan.ipynb>

Như thế sau bước này chúng ta đã có 1 tệp img.csv nhằm hỗ trợ các bước cho phần sau.

Sau đó tiến hành cắt phần mặt trong tám hình bằng 2 cách:

- cách 1: dùng CascadeClassifier.
- cách 2: cắt bằng tay.



## Sản phẩm thu được

### 2. Xử lý dữ liệu:

Quá trình cần 2 bước: nhận diện được khuôn mặt và xử lý dữ liệu.

- Bước 1: Nhận diện được khuôn mặt:
  - Cắt phần khuôn mặt trong tấm hình. (dùng CascadeClassifier) (đã làm).
- Bước 2: Xử lý dữ liệu:
  - Chuyển về hình trắng đen.
  - Resize dữ liệu về ma trận 150x150 và chuyển thành vector.

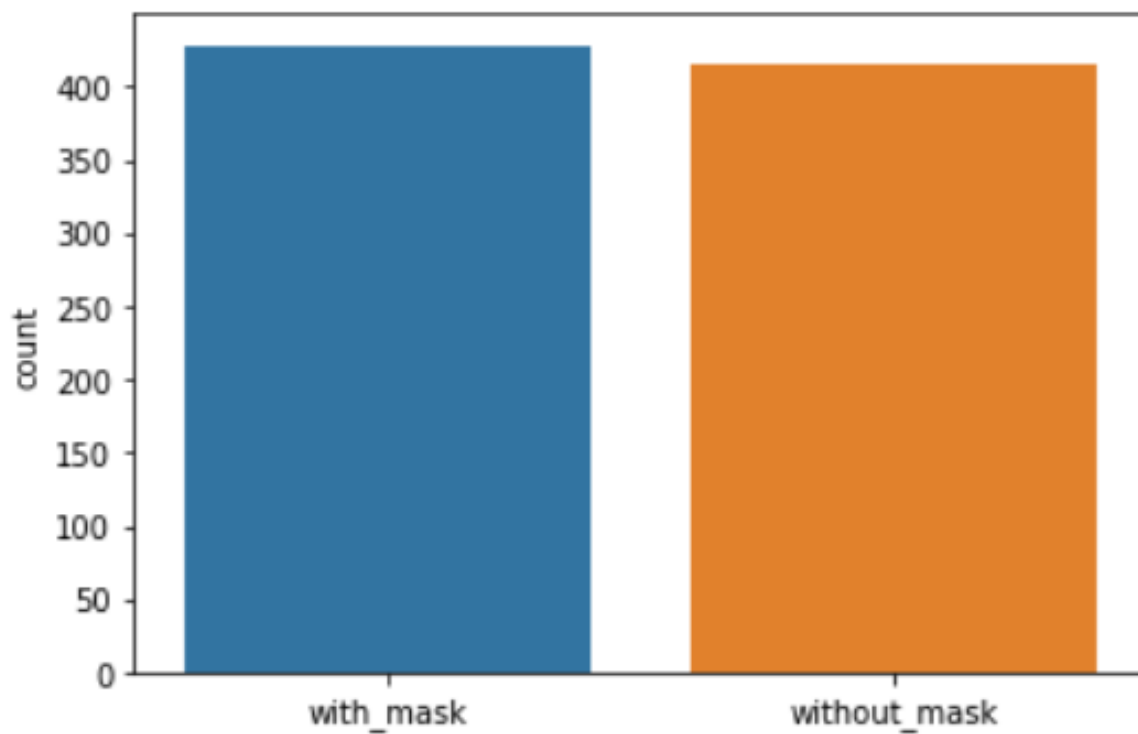
Sau khi xử lý, ta chia tập dataset thành 2 phần gồm: train 80% và test 20%



Tóm tắt các xử lý dữ liệu

### 3. Chọn model và train:

Tổng hợp dữ liệu sau khi xử lý:



Ta có số lượng cụ thể:

- With\_mask: 429 ảnh.
- Without\_mask: 416 ảnh.
- Tổng số lượng là: 845 ảnh.

➔ Chọn model SVC

#### 4. Đánh giá:

Đánh giá qua accuracy\_score, confusion matrix, precision score và recall score.

	Predicted: with_mask	Predicted: without_mask
True: with_mask	91	6
True: without_mask	4	69

Kết quả:

- Accuracy\_score: 0.9411764705882353
- Precision score: 0.9163851351351351
- Recall score: 0.9070642201834862



## 5. Dự đoán:



Sau đây là một vài kết quả dự đoán của mô hình:



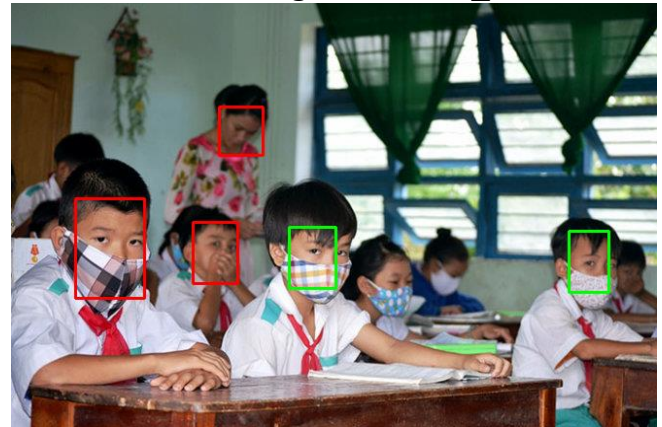
Sử dụng CascadeClassifier



Sử dụng mobilenet\_v2



Sử dụng CascadeClassifier



Sử dụng mobilenet\_v2

- Kết quả: Accuracy score: 90-95%
- Hạn chế:
  - Kết quả thu được tuy cao nhưng đó là kết quả khi chưa đánh giá qua bước face detection.
  - Hình ảnh quá mờ hoặc quá sáng sẽ không nhận diện được.
  - Đa phần nhận diện các khẩu trang đậm màu hoặc những loại khẩu trang lạ sẽ cho ra kết quả sai.

#### 6. Hướng phát triển:

- Cần nhiều datasets hơn.
- Sử dụng các deep model hoặc các model khác tốt hơn.
- Cần qua các bước hiệu chỉnh ảnh để có thể thu được kết quả tốt hơn.

### III.TÀI LIỆU THAM KHẢO:

<https://machinelearningcoban.com/2016/12/27/categories/>

<https://machinelearningcoban.com/2017/04/13/softmaxsmv/>

<https://www.pyimagesearch.com/2018/02/26/face-detection-with-opencv-and-deep-learning/>

<https://viblo.asia/u/ngovantien?page=2>



**Sinh viên thực hiện:**

<b>STT</b>	<b>Họ và tên</b>	<b>MSSV</b>
1	NGUYỄN VÕ HÙNG VỸ	18521683
2	NGUYỄN THỊ HÀ	18520691