

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

**KHOA CÔNG NGHỆ THÔNG TIN 1**



****

**BÀI TẬP LỚN**

**BỘ MÔN:** XỬ LÝ ẢNH

**CHUYÊN ĐỀ:** NHẬN DIỆN NGƯỜI ĐEO KHẨU TRANG

**NHÓM MÔN HỌC:** 03

**Giảng viên:** Đào Thị Thúy Quỳnh

**Sinh viên:** Đặng Thị Vân Anh

**Mã sinh viên:** B19DCCN010

***Hà Nội – 2022***

MỤC LỤC

[**I.** **Phát biểu bài toán** 3](#_Toc119776738)

[**II.** **Giới thiệu tổng quan về MobileNetV2** 3](#_Toc119776739)

[**III.** **Giới thiệu về tập dữ liệu** 5](#_Toc119776740)

[**IV.** **Viết chương trình** 6](#_Toc119776741)

[1. Khai báo thư viện cần sử dụng 6](#_Toc119776742)

[2. Khai báo đường dẫn đến thư mục chứa data 6](#_Toc119776743)

[3. Xây dựng hàm load dữ liệu, gắn nhãn và tiền xử lý dữ liệu 6](#_Toc119776744)

[4. Mã hóa nhãn, phân chia tập dữ liệu và tăng cường dữ liệu 7](#_Toc119776745)

[5. Xây dựng mô hình 7](#_Toc119776746)

[6. Huấn luyện mô hình, test mô hình và đánh giá 8](#_Toc119776747)

[7. Sử dụng mô hình 10](#_Toc119776748)

[**V.** **Kết luận** 12](#_Toc119776749)

1. **Phát biểu bài toán**

Đại dịch covid-19 đã và đang gây ra những hậu quả vô cùng nghiêm trọng ảnh hưởng đến đến sức khỏe cộng đồng Việt Nam và trên toàn thế giới. Kể từ khi đại dịch được phát hiện, đã có rất nhiều các biện pháp phòng ngừa mà người dân cần phải tuân theo, trong đó quan trọng nhất là giãn cách xã hội và đeo khẩu trang.

Trí tuệ nhân tạo hỗ trợ các phương pháp phát hiện dựa trên hình ảnh và video có thể phát hiện đối tượng với độ chính xác cao và phân loại xem một người có đeo hay không đeo khẩu trang. Nhận dạng người đeo khẩu trang có thể được thực hiện bằng các phương pháp học sâu và học máy, chẳng hạn như “Support Vector Machines” và “Decision Trees”, nhưng với các bộ dữ liệu khác nhau.

Phương pháp nhận biết ai đó có đeo khẩu trang hay không và vị trí khuôn mặt của người đó được gọi là phát hiện người đeo khẩu trang. Vấn đề được đan xen với phát hiện đối tượng chung, có nghĩa là xác định các lớp đối tượng. Phát hiện khuôn mặt là phát hiện một loại đối tượng cụ thể. Nhận dạng đối tượng và khuôn mặt có nhiều ứng dụng. Công việc liên quan trong việc áp dụng tính năng phát hiện khẩu trang là rất lớn, không chỉ sử dụng mạng thần kinh mà còn sử dụng các thuật toán máy học và áp dụng nhiều bộ phân loại khác nhau cho hệ thống.

Trong bài tập lớn này, sử dụng mô hình kiến trúc MobilNetV2 để huấn luyện. Mô hình đã được huấn luyện và thử nghiệm bằng cách sử dụng hai bộ dữ liệu độc lập về người đeo khẩu trang và không đeo khẩu trang. Để nhận dạng khuôn mặt từ các luồng video, sử dụng một mô hình kiến trúc MobileNetV2 được đào tạo trước. Một loạt các gói phương pháp học “máy/sâu” và “kỹ thuật xử lý hình ảnh” đã được sử dụng cùng với framework của OpenCV. Để cung cấp khả năng giám sát hiệu quả và cho phép chủ động, các bước khác nhau cũng đã được sử dụng, bao gồm “tăng cường dữ liệu, tải bộ phân loại (MobileNetV2), xây dựng lớp được kết nối đầy đủ (FC), xử lý trước và tải dữ liệu hình ảnh, áp dụng bộ phân loại, traning, validation và testing”.

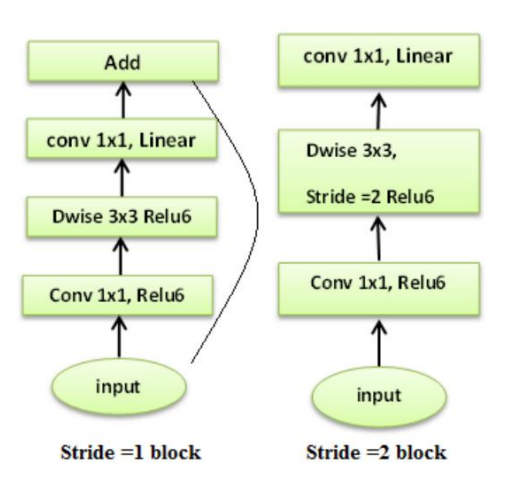
1. **Giới thiệu tổng quan về MobileNetV2**

MobileNetV2 là một kiến trúc sử dụng Depthwise Separable Convolutions( tích chập tách biệt chiều sâu). Ngoài ra còn đề xuất thêm: **Linear bottlenecks**, **Inverted Residual Block** (shortcut connections giữa các bottlenecks). Nó có hai loại block như trong Hình 1

Có 3 phần đối với mỗi block:

* Layer đầu là 1×1 convolution với ReLU6.
* Layer thứ hai, là depthwise convolution.
* Layer thứ 3 tiếp tục là 1×1 convolution nhưng không có activation function. Linear được sử dụng thay vì ReLu như bình thường.

Phương pháp phát hiện đối tượng là thực hiện phân loại để xác định lớp đầu vào, regression để điều chỉnh giới hạn của bounding box. Ngoại trừ các lớp được kết nối hoàn toàn cuối cùng, hầu hết backbone networks dùng để phát hiện vật thể là dùng cho việc phân loại. Backbone networks hoạt động như một trình trích xuất đặc trưng đơn giản cho các tác vụ phát hiện đối tượng, lấy hình ảnh làm đầu vào và tạo feature maps cho mỗi hình ảnh đầu vào.



*Hình 1. Các khối MobilNetV2*

Các predefined trained techniques thường được sử dụng để trích xuất các feature maps với các vấn đề về phân loại chất lượng cao. Phần này của mô hình được gọi là **Base model**.

**Base model** là mạng MobileNetV2 sử dụng trọng số " image net". ImageNet là một cơ sở dữ liệu hình ảnh đã được đào tạo trên hàng trăm nghìn hình ảnh và do đó, nó cực kỳ hữu ích cho việc phân loại hình ảnh. Các " bounding boxes " được đánh giá và so sánh với " ground truth boxes " trong quá trình đào tạo và các tham số có thể đào tạo được thay đổi khi cần trong quá trình lan truyền ngược. Một hạt nhân được sử dụng trong mỗi không gian đối tượng để tạo ra các kết quả hiển thị điểm số tương ứng cho từng pixel, cho dù một mục có tồn tại hay không, cũng như kích thước hộp giới hạn thích hợp.

MobileNetV2 có một lớp đầu ra bao gồm 1000 nơ-ron, tương ứng với 1000 danh mục mà nó đã được huấn luyện. Nhưng ở đây chúng ta chỉ muốn phân loại hai loại: with\_mark và without\_mark. Do đó, chúng ta cần một lớp đầu ra chỉ bao gồm hai nơ-ron.

MobileNet được tạo thành từ hai phần: **Base model** và bộ phân loại. Trong mô hình này, **Base model** được sử dụng lại, phần đầu được cắt bớt và sử dụng hai lớp được kết nối đầy đủ. Và điều chỉnh lại mạng bằng cách thêm 4 lớp nữa.

Cuối cùng, để phân loại xem một người có đeo khẩu trang hay không, ta dùng một lớp dense với 2 nơ-ron và softmax activation function.

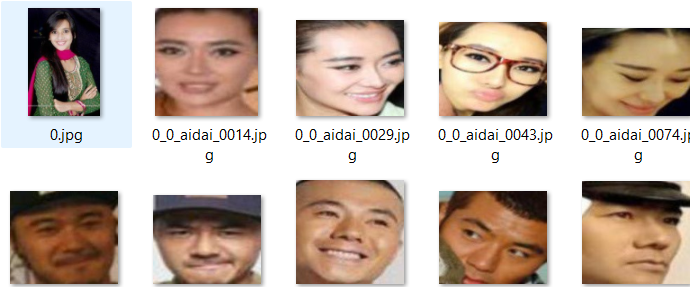
1. **Giới thiệu về tập dữ liệu**

Dữ liệu gồm có hai tập hình ảnh: một tập là hình ảnh người đeo khẩu trang (with\_mark) gồm có 1915 file ảnh, tập còn lại là người không đeo khẩu trang (without\_mark) gồm 1918 file ảnh.

Đây là tập dữ liệu được sưu tầm trên github.



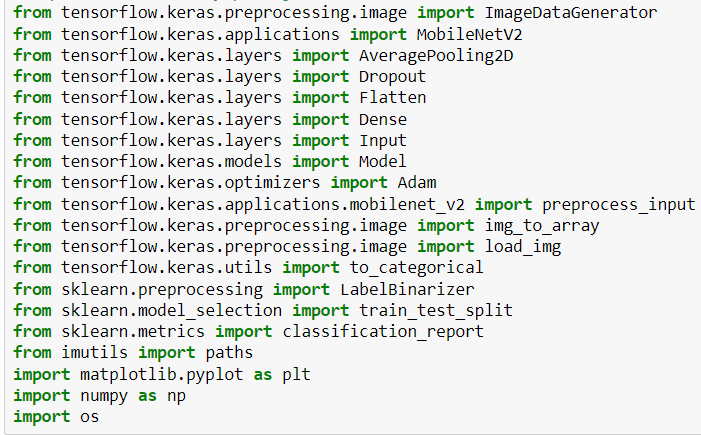
*Tập dữ liệu with\_mark*

**

*Tập dữ liệu without\_mark*

Dữ liệu huấn luyện bị giới hạn về kích thước nên việc tăng cường dữ liệu được sử dụng để mở rộng số lượng tập dữ liệu dành cho huấn luyện bằng cách thao tác hình ảnh trong tập dữ liệu. “Cắt, tương phản, lật ngang, xoay, thu phóng và làm mờ” đều được sử dụng để phong phú hình ảnh đào tạo. Bằng cách thay đổi tỷ lệ hình ảnh đầu vào bằng 224\*224 và chuyển đổi nó thành một kênh duy nhất, kích thước của mô hình hiện tại có thể được giảm xuống.

1. **Viết chương trình**
2. Khai báo thư viện cần sử dụng



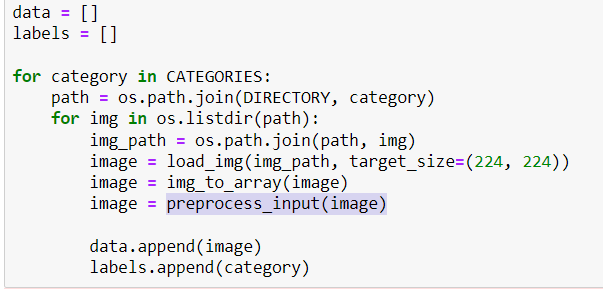
* Thư viện tensorflow dùng để tải dữ liệu hình ảnh, tăng cường dữ liệu, tải mô hình MobilNetV2, xây dựng một đầu layer mới
* Thư viện imutils dùng để triển khai và liệt kê các hình ảnh trong tập dữ liệu
* Thư viện matplotlib để vẽ đồ thị về kết quả huấn luyện

1. Khai báo đường dẫn đến thư mục chứa data



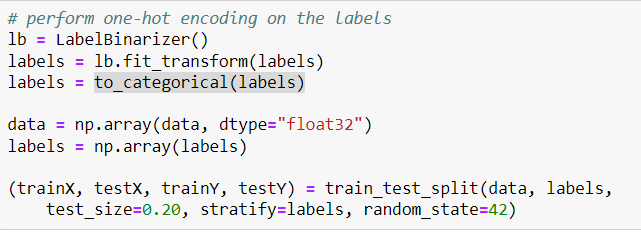
* Phân loại ảnh là bài toán học có giám sát, do đó dữ liệu huấn luyện và kiểm định phải được gắn nhãn. Ảnh được gắn nhãn tương ứng với tên thư mục chứa nó, ta có tên và thứ tự các nhãn tương ứng với tên và thứ tự cá thư mục chứa ảnh huấn luyện và kiểm định. Gồm có 2 nhãn là : with\_mark và without\_mark.

1. Xây dựng hàm load dữ liệu, gắn nhãn và tiền xử lý dữ liệu

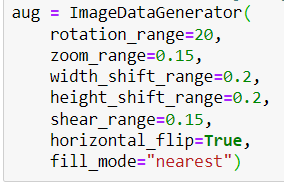


* Các bước tiền xử lý bao gồm: thay đổi kích thước của ảnh thành 224\*224 pixel, chuyển đổi sang định dạng mảng numpy và chia tỷ lệ cường độ pixel trong ảnh đầu vào thành phạm vị [-1, 1]

1. Mã hóa nhãn, phân chia tập dữ liệu và tăng cường dữ liệu



* Chuyển đổi nhãn thành dạng số để máy có thể đọc được. Vì chỉ có 2 nhãn nên ở đây dùng one-hot encoding. Với one-hot, ta chuyển đổi từng giá trị phân loại thành một cột phân loại mới và gán giá trị nhị phân là 1 hoặc 0 cho các cột đó. Mỗi giá trị số nguyên được biểu diễn dưới dạng một vectơ nhị phân
* Sử dụng method của scikit-learning để phân chia dữ liệu thành 2 phần: 80% để train và 20% để test.



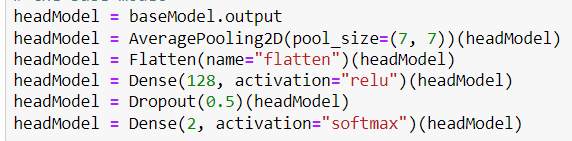
* Để cải thiện khả năng khái quát hóa và vì dữ liệu huấn luyện bị giới hạn về kích thước nên việc tăng cường dữ liệu được sử dụng để mở rộng số lượng tập dữ liệu dành cho huấn luyện bằng cách thao tác hình ảnh trong tập dữ liệu như: Cắt, tương phản, lật ngang, xoay, thu phóng và làm mờ.

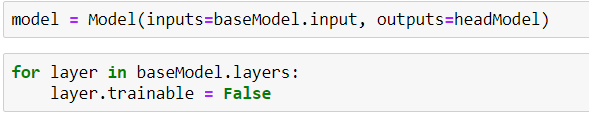
1. Xây dựng mô hình

* MobileNetV2 có một lớp đầu ra bao gồm 1000 nơ-ron, tương ứng với 1000 danh mục mà nó đã được huấn luyện. Nhưng ở đây chúng ta chỉ muốn phân loại hai loại: with\_mark và without\_mark. Do đó, chúng ta cần một lớp đầu ra chỉ bao gồm hai nơ-ron. Ở đây ta thực hiện việc tinh chỉnh(Fine-tuning) MobilNetV2 network, để có thể tận dụng lại các tri thức đã được huấn luyện và cải thiện lại trên tác vụ phân loại.
* Đầu tiên load MobilNetV2 network với các trọng số ImageNet đã được đào tạo trước và bỏ đi các layer FC ở đầu. MobilNetV2 network được sử dụng để trích xuất các feature maps với các vấn đề về phân loại. Phần này của mô hình được gọi là **Base model**.



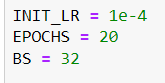
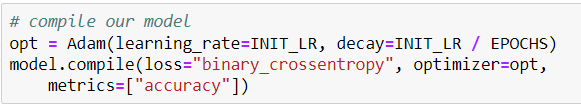
* Xây dựng một đầu FC mới bằng cách thêm 4 lớp nữa và gắn nó vào thay thế cho đầu cũ
  + AveragePooling2D dùng để lấy feature nổi bật (dùng average) và giúp giảm parameter khi training
  + Flatten dùng để làm phằng layer để fully connection
  + Dropout layer này dùng như regularization cho các layer hạn chế overfiting.
  + Cuối cùng là một lớp Dense với 2 nơ-ron tương ứng với đầu ra là 2



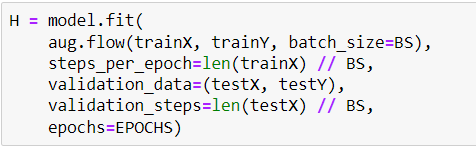


* Cuối cùng là đóng băng baseModel. Trọng số của baseModel sẽ không được cập nhật trong quá trình lan truyền ngược, trong khi trọng số của lớp đầu sẽ được điều chỉnh

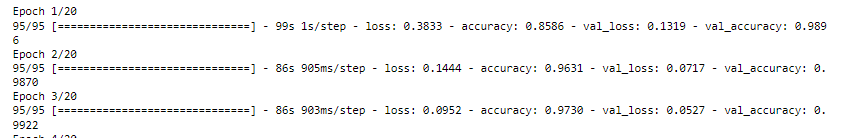
1. Huấn luyện mô hình, test mô hình và đánh giá

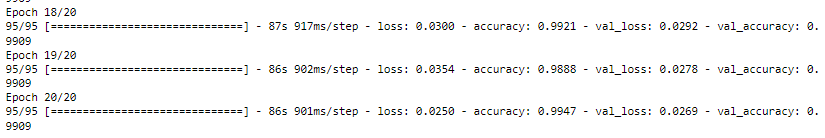


* Biên dịch mô hình với trình tối ưu hóa Adam, learning\_rate và entropy chéo nhị phân.
* Huấn luyện mô hình. Lưu ý tập dữ liệu tăng cường (aug) sẽ cung cấp các lô dữ liệu hình ảnh sau khi đã biến đổi

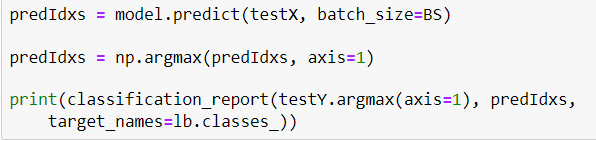


* Kết quả:

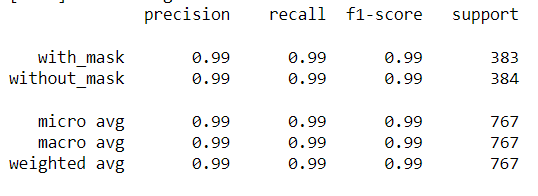




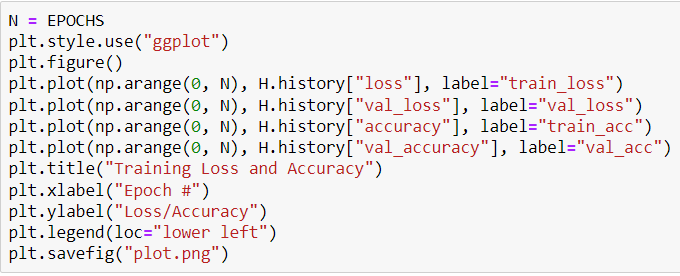
* Sau khi huấn luyện xong, ta sẽ đánh giá mô hình trên tập test



* Kết quả: tỷ lệ của các metric đều rất cao



* Vẽ đồ thị biểu diễn tỷ lệ loss và accuracy qua mỗi lần epoch



* Kết quả: độ chính xác đã tăng lên đến 99% ở lần epoch thứ 5. Sự tổn thất (loss) đã giảm xuống mức thấp nhất ở lần epoch thứ 10, gần như bằng 0.

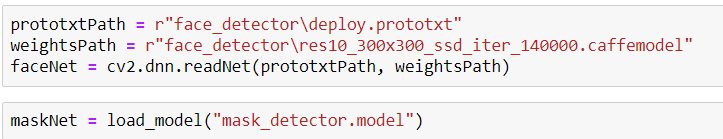


* Cuối cùng, lưu lại model đã huấn luyện:



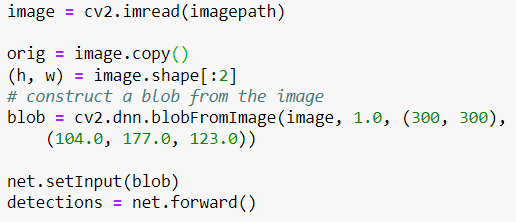
1. Sử dụng mô hình

Đầu tiên, load sử dụng DNN Face Detector để nhận diện khuôn mặt có sẵn và model mà ta đã train.

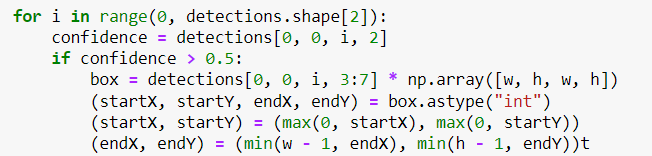


Xây dựng hàm load ảnh, xử lý hình ảnh đầu vào và thực hiện phân loại:

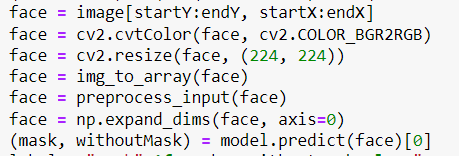
* Đầu tiên load ảnh và xử lý ảnh đầu vào: tạo bản sao và lấy kích thước khung để hiển thị và chia tỷ lệ. tat hay đổi kích thước thành 300\*300 pixel và thực hiện phép trừ trung bình
* Sau đó, thực hiện nhận diện khuân mặt để xác định vị trí của tất cả các khuân mặt trong ảnh.



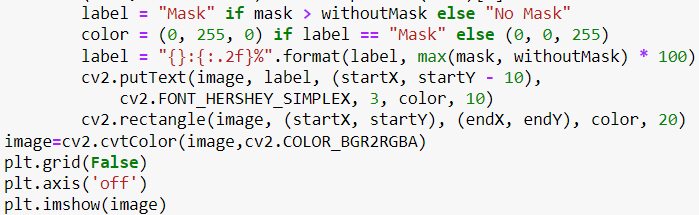
* Khi đã dự đoán được vị trí của từng khuân mặt, ta sẽ phải đảm bảo các khuân mặt đó có tỷ lệ chính xác > 0.5, lọc ra các đối tượng có xác suất thấp.
* Tiếp, tính toán giới hạn box cho từng khuân mặt và đảm bảo box nằm trong giới hạn của hình ảnh.



* Trích xuất các vùng xuất hiện khuân mặt, chuyển đổi nó từ BGR sang RGB, thay đổi kích thước thành 224\*224 pixel, … xử lý giống trong quá trình đào tạo, và thực hiện dự đoán



- Sau khi dự đoán ta sẽ chú thích và hiển thị kết quả: chú thích gồm có bounding box, tên nhãn, xác suất nhãn



* Kết quả:

|  |  |
| --- | --- |
|  | *Hình ảnh trước khi dự đoán* |
|  | *Hình ảnh sau khi dự đoán* |

1. **Kết luận**

Với mô hình được đào tạo ở trên đã đưa ra dự đoán khá chính xác và có thể giúp ích cho cộng đồng sức khỏe. Có thể ứng dụng trong bệnh viện hay những nơi công công khác. Tuy nhiên vẫn còn một số hạn chế như người ko đeo khẩu trang mà chỉ lấy tay che miệng hay quấn khăn quàng cổ cũng sẽ bị nhẫm lẫn là đeo khâu trang. Ta có thể sẽ phải tìm kiếm thêm bộ dữ liệu cho phần này và huấn luyện lại mô hình cho chính xác hơn.