**BÁO CÁO BÀI TẬP LỚN**

Môn Xử Lý Đa Chiều

Đề tài: Face Mask Detection

**Giảng viên hướng dẫn:** Nguyễn Thanh Bình

**Nhóm thực hiện:** 1711091 – Nguyễn Thị Thu Duyên

1711192 – Nguyễn Thị Thu Nhi



Thành phố Hồ Chí Minh – Năm 2020

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

Khoa Toán – Tin học

1. **Lý do chọn đề tài**

* Các khuyến nghị gần đây nhất liên quan đến khả năng chống lại sự lây lan của COVID19 bao gồm đeo khẩu trang như một biện pháp bảo vệ bổ sung. Vì vậy, việc phát triển một ứng dụng phát hiện người dân có đeo khẩu trang ở những nơi đông người hay không là rất cần thiết.
* Với sự phát triển của Face Mask Detection, chúng ta có thể phát hiện một người bị nhiễm có đeo khẩu trang hay không, ở đâu, khi nào có thể giúp xác minh mức độ lây nhiễm trong xã hội. Bên cạnh đó còn có nhiều lợi ích khác đi kèm như giảm nhân lực, chí phí chống dịch...

1. **Tổng quan mô hình**

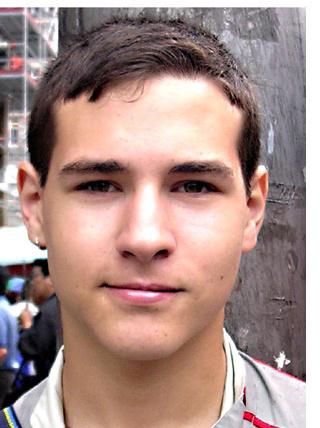
* Nhóm đã quyết định xây dựng mô hình Mạng nơ-ron liên kết (CNN) bằng cách sử dụng TensorFlow với thư viện Keras và OpenCV để phát hiện xem một người có đeo khẩu trang hay không.

1. **Giới thiệu về mô hình và thư viện chính sử dụng**
   * Convolutional Neural Networks (CNN) :
     + Là một trong những mô hình deep learning phổ biến nhất và có ảnh hưởng nhiều nhất trong cộng đồng Computer Vision.
     + CNN được dùng trong nhiều bài toán như nhận dạng ảnh, phân tích video, ảnh MRI, hoặc trong lĩnh vự xử lý ngôn ngữ tự nhiên. Nói chung, CNN có thể giải quyết tốt hầu hết các vấn đề này.
     + CNN bao gồm tập hợp các lớp cơ bản bao gồm: convolution layer với nonlinear layer, pooling layer, fully connected layer. Các lớp này liên kết với nhau theo một thứ tự nhất định.
     + Thông thường, một ảnh sẽ được lan truyền qua tầng convolution layer với nonlinear layer trước, sau đó các giá trị tính toán được sẽ lan truyền qua pooling layer. Bộ ba convolution layer, nonlinear layer và pooling layer có thể được lặp lại nhiều lần trong network. Sau cùng được lan truyền qua tầng fully connected layer và softmax để tính sác xuất ảnh đó chứa vật thể nào đó.
   * Keras, Tensorflow và OpenCV:
   * Kể từ 2012 khi [deep learning có bước đột phá lớn](https://machinelearningcoban.com/2018/06/22/deeplearning/" \l "dot-pha-), hàng loạt các thư viện hỗ trợ deep learning ra đời. Cùng với đó, ngày càng nhiều kiến trúc deep learning được đề xuất, khiến cho số lượng ứng dụng và các bài báo liên quan tới deep learning tăng lên chóng mặt.
   * Keras và Tensor là những thư viện deep learning được tạo ra từ một trong những công ty công nghệ lớn – Google.
   * TensorFlow:
     + Dùng cho tính toán số học sử dụng đồ thị luồng dữ liệu.
     + Khả năng tương thích và mở rộng tốt.
   * Keras:
     + Xây dựng model nhanh.
     + Hỗ trợ xây dựng CNN.
     + Tương đối dễ học, dễ dùng.
   * OpenCV:
     + Dùng trong xử lý về thị giác máy tính, machine learning, xử lý ảnh
     + Tốc độ tính toán nhanh
2. **Dữ liệu**

* Tập dữ liệu này bao gồm 3846 hình ảnh thuộc hai lớp:
  + with\_mask: 1916 hình ảnh



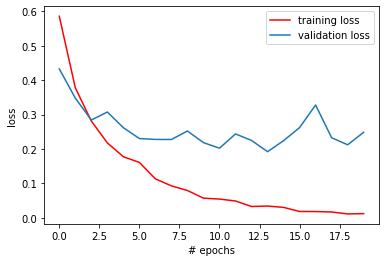
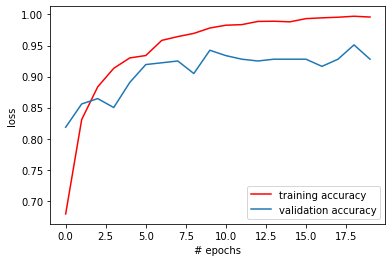
* + without\_mask: 1930 hình ảnh



* Hình ảnh được sử dụng là hình ảnh thực của những khuôn mặt. Hình ảnh được thu thập từ các nguồn sau:
  + Bing Search API
  + Bộ dữ liệu Kaggle
  + Bộ dữ liệu RMFD

1. **Quy trình**

* Bước 1: Xử lý trước dữ liệu
* Bước 2: Chuyển đổi hình ảnh thành hình ảnh thang độ xám
* Tập dữ liệu đang sử dụng bao gồm các hình ảnh có màu sắc khác nhau, kích thước khác nhau và hướng khác nhau.
* Do đó, cần chuyển đổi tất cả các hình ảnh thành thang độ xám vì nhóm cần chắc chắn rằng màu sắc không phải là điểm quan trọng để phát hiện khẩu trang.
* Sau đó, cần có tất cả các hình ảnh có cùng kích thước (64x64) trước khi áp dụng vào mạng nơ-ron.
* Bước 3: Tải các mảng numpy đã lưu
* Bước 4: Xây dựng mô hình tuần tự (Sequential model)
* Bây giờ, nhóm xây dựng mô hình CNN tuần tự của mình với các lớp khác nhau như Conv2D, MaxPooling2D, Flatten, Dropout và Dense.
* Trong lớp Dense cuối cùng, nhóm sử dụng hàm ‘softmax’ để xuất ra một vectơ cung cấp xác suất của mỗi lớp trong hai lớp.
* Vì dữ liệu có hai danh mục (có khẩu trang và không có khẩu trang) nên nhóm có thể sử dụng binary\_crossentropy.
* Bước 5: Đào tạo mô hình CNN
* Trong bước này, nhóm đưa hình ảnh của mình vào tập train và tập test với sequential model mà nhóm đã xây dựng bằng thư viện keras.
* Nhóm đã đào tạo mô hình trong 20 epoch (lặp lại). Tuy nhiên, có thể train thêm số lượng epoch để đạt được độ chính xác cao hơn để tránh xảy ra hiện tượng overfitting.
* Bước 6: Vẽ đồ thị accuracy và đường cong loss

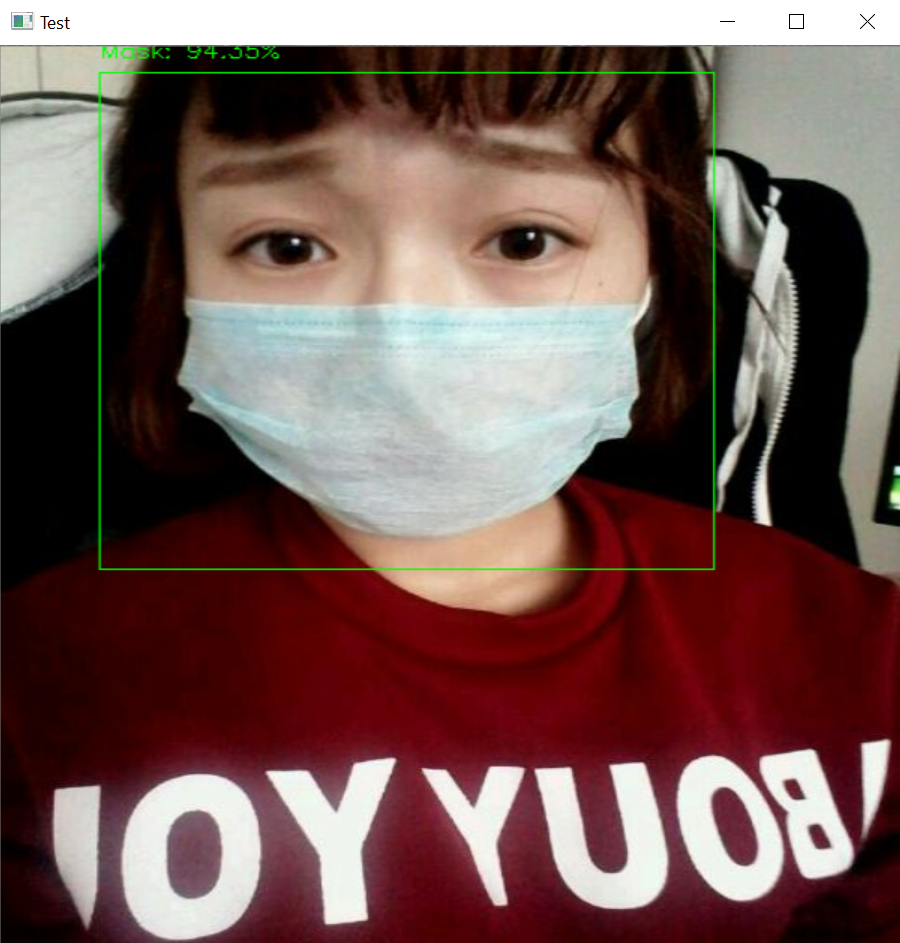
* Bước 7: Sử dụng mô hình
* Keras hỗ trợ lưu mô hình đã đào tạo để sử dụng sau này.
* Nó sẽ dự đoán khả năng xảy ra của hai lớp ([Mask, No mask]).
* Dựa trên xác suất, nhãn sẽ được chọn và hiển thị kết quả.
* Bước 8: Nhận diện khuôn mặt
* Để phát hiện xem một người có đeo khẩu trang hay không bằng cách sử dụng webcam của PC, nhóm cần triển khai tính năng nhận diện khuôn mặt.
* Trong trường hợp này, nhóm sử dụng Haar Feature-based Cascade Classifiers để phát hiện các đặc điểm của khuôn mặt.
* Bước 9: Phát hiện khuôn mặt đeo khẩu trang
* Cần gắn nhãn hai xác suất (0 cho No mask và 1 cho Mask)
* Sau đó, đặt màu hình chữ nhật giới hạn bằng cách sử dụng các giá trị RGB (ĐỎ và XANH LÁ).
* Ở bước cuối cùng, nhóm sử dụng thư viện OpenCV để chạy một vòng lặp vô hạn để sử dụng máy ảnh web, trong đó nhóm phát hiện khuôn mặt bằng Cascade Classifier.
* Mã webcam = cv2.VideoCapture (0) biểu thị việc sử dụng webcam.
* Mô hình sẽ dự đoán khả năng xảy ra của từng lớp trong số hai lớp ([Mask, No mask]).
* Dựa trên xác suất, nhãn sẽ được chọn và hiển thị kết quả.

1. **Kết quả**

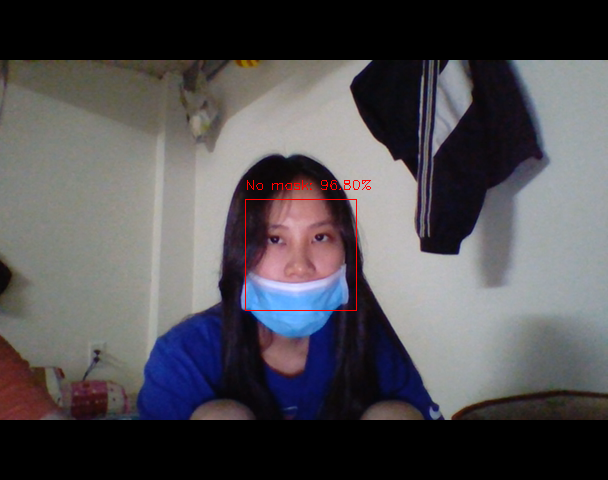
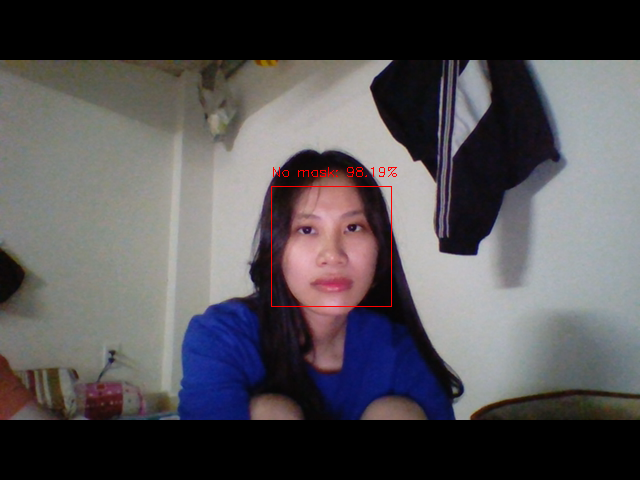
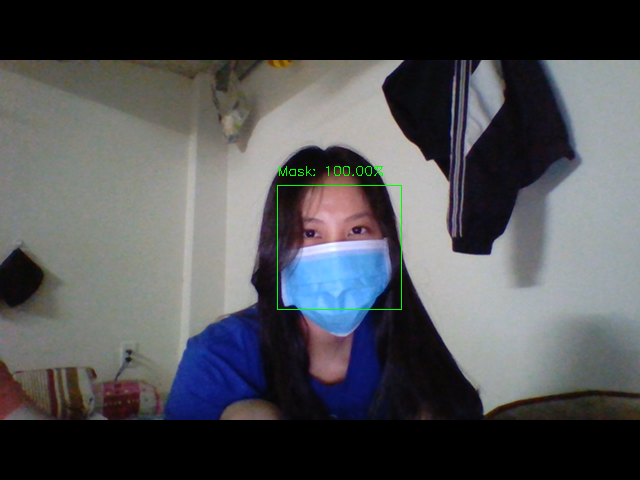
* Bảng kết quả thực nghiệm:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Tốc độ xử lý | Final loss | Final accuracy |
| Tập train | 11s 111ms – 12s 123ms/step | 0.0121 | 0.9958 |
| Tập valid | 0.2484 | 0.9280 |
| Tập test | 0s 27ms/step | 0.2254 | 0.9403 |

* Kết quả từ hình ảnh:



* Kết quả từ webcam:



* Điểm yếu:
  + Không nhận dạng được người đeo khẩu trang đen
  + Không nhận dạng được nhân vật hoạt hình
  + Bị ảnh hưởng bởi việc đeo kiếng
  + Không hoạt động được với góc nghiêng

1. **Hướng cải tiến**
   * Cải thiện mô hình để áp dụng cho nhân vật ảo và khuôn mặt góc nghiêng.
   * Khắc phục việc chưa nhận dạng được khẩu trang đen.
   * Trong tương lai, nhóm muốn thử việc kết hợp nhiều mô hình và phương pháp khác nhau như: Deep Dense Face Detector (DDFD), Deformable Part Model (DPM)…
2. **Nguồn tham khảo**

* <https://machinelearningcoban.com/2018/07/06/deeplearning/>
* <https://www.goeduhub.com/10548/detection-learning-project-detect-whether-person-wearing>
* <https://github.com/AIZOOTech/FaceMaskDetection>
* <https://github.com/chandrikadeb7/Face-Mask-Detection>