BÁO CÁO ĐỒ ÁN

KHOA HỌC MÁY TÍNH

Tên đề tài:

XÂY DỰNG MÔ HÌNH NHẬN DIỆN NGƯỜI DÙNG ĐEO KHẨU TRANG SỬ DỤNG MÔ HÌNH HỌC MÁY

***Giáo viên hướng dẫn:* Lê Hồng Anh**

***Sinh viên thực hiện*:**

**Đặng Thạch Sơn**

***Lớp*: Khoa học máy tính ứng dụng K63B**

**Nguyễn Đức Tư**

***Lớp*: Khoa học máy tính ứng dụng K63B**

**Hà Nội – 2021**

MỤC LỤC

[DANH MỤC HÌNH ẢNH 4](#_Toc90392636)

[LỜI MỞ ĐẦU 1](#_Toc90392637)

[CHƯƠNG I: TỔNG QUAN VỀ ĐỀ TÀI 2](#_Toc90392638)

[1: Lý do chọn đề tài 2](#_Toc90392639)

[2: Mục tiêu của đề tài 3](#_Toc90392640)

[3: Nội dung nghiên cứu. 3](#_Toc90392641)

[4: Phạm vi nghiên cứu. 3](#_Toc90392642)

[CHƯƠNG II: KHẢO SÁT HỆ THỐNG 4](#_Toc90392643)

[2.1: Nhiệm vụ cơ bản. 4](#_Toc90392644)

[2.2: Mô tả hệ thống. 4](#_Toc90392645)

[2.2.1: Bài toán nhận dạng khuôn mặt. 4](#_Toc90392646)

[2.2.2) Những khó khăn và thách thức trong bài toán nhận dạng khuôn mặt. 6](#_Toc90392647)

[2.2.3) Các ứng dụng của bài toán nhận diện khuôn mặt. 7](#_Toc90392648)

[2.3: Quy trình xử lý. 10](#_Toc90392649)

[2.3.1.Các chức năng tổng quát. 10](#_Toc90392650)

[2.3.2: Chi tiết về các chức năng. 10](#_Toc90392651)

[CHƯƠNG III: CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ 11](#_Toc90392652)

[3.1) Lý thuyết về học máy. 11](#_Toc90392653)

[3.2. Deep Learning. 14](#_Toc90392658)

[3.2. Một số thuật toán về nhận dạng khuôn mặt. 15](#_Toc90392659)

[3.3. Thuật toán CNN 17](#_Toc90392660)

[3.3.1. Cấu trúc của CNN. 17](#_Toc90392661)

[3.3.2: CNN với bài toán nhận diện khuôn mặt 20](#_Toc90392662)

[3.3.3 Phóng đại ảnh (Scale ảnh). 24](#_Toc90392663)

[3.3.4 Dịch chuyển ảnh (Translation) 25](#_Toc90392664)

[3.3.5 Xoay ảnh (Rotation) 25](#_Toc90392665)

[3.4. Các thư viện được sử dụng trong chương trình. 26](#_Toc90392666)

[3.4.1 Thư viện OpenCV. 26](#_Toc90392667)

[3.4.2 Thư viện Keras. 27](#_Toc90392668)

[3.4.3 Thư viện Sklearn. 28](#_Toc90392669)

[3.4.4 Thư viện NumPy 29](#_Toc90392670)

[3.4.5 Thư viện Pandas 29](#_Toc90392671)

[3.4.6 Thư viện TensorFlow 31](#_Toc90392672)

[3.4.7 Thư viện Pillow 32](#_Toc90392673)

[CHƯƠNG IV: XÂY DỰNG MÔ HÌNH VÀ THIẾT KẾ HỆ THỐNG 33](#_Toc90392674)

[4.1. Giới thiệu hệ thống 33](#_Toc90392675)

[4.2: Mô tả tập dữ liệu 34](#_Toc90392676)

[4.3: Thiết kế hệ thống 37](#_Toc90392677)

[4.4: Xây dựng mô hình 38](#_Toc90392678)

[4.4.1: Bài toán 1: Mô hình RETINAFACE: 38](#_Toc90392679)

[4.4.2: Bài toán 2: Sử dụng phương pháp Transfer Learning với mô hình Xception (đã huấn luyện trên ImageNet) 40](#_Toc90392680)

[4.4.3: Kết quả thử nghiệm trực tiếp bằng webcam: 47](#_Toc90392681)

[CHƯƠNG V: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN CỦA ĐỀ TÀI 49](#_Toc90392682)

[5.1: Kết luận: 49](#_Toc90392683)

[5.2: Hướng phát triển đề tài: 49](#_Toc90392684)

[TÀI LIỆU THAM KHẢO 50](#_Toc90392685)

|  |  |
| --- | --- |
| DANH MỤC HÌNH ẢNH | |
| Hình 1 | Nhận dạng mặt người |
| Hình 2 | Một số thông tin có thể tìm thấy trong một bức ảnh khuôn mặt. |
| Hình 3 | Quá trình xử lý ảnh |
| Hình 4 | Các bước cơ bản trong một hệ thống xử lý ảnh. |
| Hình 5 | minh họa về học máy |
| Hình 6 | mô tả học máy là gì? |
| Hình 7 | CNN ứng dụng trong nhiều mô hình |
| Hình 8 | Sự khác biệt giữa nhận dạng đối tượng và phát hiện đối tượng |
| Hình 9 | Một minh họa về một số bản đồ đặc trưng của các lớp DC12, conv22, conv32, conv42 và conv52 được đào tạo cho nhiệm vụ nhận dạng khuôn mặt |
| Hình 10 | Tổng quan về phương pháp nhận dạng khuôn mặt người bằng mạng CNN |
| Hình 11 | Thiết kế mạng CNN để xử lý ảnh |
| Hình 12 | Các giai đoạn dò tìm và hiệu chỉnh |
| Hình 13 | Kiến trúc mạng của mô hình DeepID 2 |
| Hình 14 | Giới thiệu về thư viện OpenCV |
| Hình 15 | Biểu đồ thể hiện tỷ lệ tập Train và Test của bộ dữ liệu |
| Hình 16 | Biểu đồ thể hiện kích thước bộ dữ liệu tập Train |
| Hình 17 | Biểu đồ thể hiện kích thước bộ dữ liệu tập Test |
| Hình 18 | Biểu đồ thể hiện kích thước bộ dữ liệu tập Validation |
| Hình 19 | Một số hình ảnh ngẫu nhiên của bộ dữ liệu |
| Hình 20 | Sơ đồ thực hiện chương trình |
| Hình 21 | Ảnh lấy từ webcam với mô hình RentinaFace |
| Hình 22 | Biểu đồ thể hiện độ chính xác của mô hình |
| Hình 23 | Hiển thị ra những kết quả dự đoán của mô hình |
| Hình 24 | Hình ảnh người dùng không đeo khẩu trang |
| Hình 25 | Hình ảnh người dùng đeo khẩu trang |

# LỜI MỞ ĐẦU

Công nghệ thông tin ngày càng phát triển và có vai trò hết sức quan trọng không thể thiếu trong cuộc sống hiện đại. Con người ngày càng tạo ra những cỗ máy thông minh có khả năng tự nhận biết và xử lí được các công việc một cách tự động, phục vụ cho lợi ích của con người. Trong những năm gần đây, một trong những bài toán nhận được nhiều sự quan tâm và tốn nhiều công sức nhất của lĩnh vực công nghệ thông tin, đó chính là bài toán nhận dạng. Tuy mới xuất hiện chưa lâu nhưng nó đã rất được quan tâm vì tính ứng dụng thực tế của bài toán cũng như sự phức tạp của nó.Bài toán nhận dạng có rất nhiều lĩnh vực như: nhận dạng vất chất(nước, lửa, đất, đá, gỗ..) nhận dạng chữ viết, nhận dạng giọng nói, nhận dang hình dáng, nhận dạng khuôn mặt.. trong đó phổ biến và được ứng dụng nhiều hơn cả là bài toán nhận diện khuôn mặt.Để nhận dạng được khuôn mặt, bước đầu tiên để nhận dạng là phát hiện ra khuôn mặt, điều này thực sự là quan trọng và hết sức khó khăn. Cho đến tận bây giờ, các nhà nghiên cứu vẫn chưa đạt được sự ưng ý trong việc giải quyết các khó khăn của bài toán và cho kết quả hoàn toàn đúng. Tuy nhiên, những gì đã đạt được cũng đủ để chúng ta áp dụng rộng rãi và đem lại những lợi ích to lớn trong cuộc sống.

# CHƯƠNG I: TỔNG QUAN VỀ ĐỀ TÀI

## 1: Lý do chọn đề tài

Trong những năm gần đây, xử lý hình ảnh đang được nghiên cứu và phát triển với tốc độ nhanh chóng bởi các trung tâm nghiên cứu, trường đại học và học viện. Trong đó nhận dạng và phân loại hình ảnh là một trong những lĩnh vực được theo đuổi một cách tích cực. Ý tưởng cốt lõi từ việc nhận dạng và phân loại hình ảnh là phân tích ảnh từ dữ liệu thu được bởi các cảm biến hình ảnh như camera, webcam. Hiện nay, cùng với sự phát triển của xã hội, vấn đề an ninh bảo mật đang được yêu cầu khắt khe tại mọi quốc gia trên thế giới. Các hệ thống xác định, nhận dạng con người được ra đời với độ tin cậy cao. Một trong những bài toán nhận dạng con người được quan tâm nhất hiện nay đó là nhận dạng qua khuôn mặt. Vì nhận dạng qua khuôn mặt là cách mà con người sử dụng để phân biệt nhau. Một số đề tài liên quan ví dụ như hệ thống chấm công tự động bằng khuôn mặt, điểm danh nhận dạng sinh viên bằng khuôn mặt, nhận dạng biển số xe…

Cùng với dịch bệnh COVID-19 vẫn đang diễn biến phức tạp, số người mắc mới chưa chịu dừng lại. Trước tình hình dịch bệnh như hiện nay, Việt Nam tiếp tục nỗ lực không ngừng để đẩy lùi đại dịch COVID-19. Vì vậy, hơn lúc nào hết chúng ta cần tiếp tục áp dụng các biện pháp phòng chống dịch hiệu quả, cần tiếp tục lan tỏa và thực hiện tốt Thông điệp 5K: “Khẩu trang – Khử khuẩn – Khoảng cách – Không tụ tập – Khai báo y” tế để giữ an toàn cho bản thân và xã hội trước đại dịch COVID-19. Nhưng vẫn có một số thành phần không chấp hành đúng biện pháp phòng chống dịch, khi nhìn thấy lực lượng kiểm tra thì mới đeo khẩu trang làm cho dịch bệnh ngày càng khó kiểm soát hơn. Trước tình hình đó, với mong muốn giúp ích cho xã hội để cho mọi người thực hiện đúng với Thông điệp 5K, nhóm chúng em quyết định áp dụng công nghệ thông tin để viết lên **“chương trình cảnh báo người dùng đeo khẩu trang sử dụng mô hình học máy”** nhằm muốn giúp ích cho việc quản lý phòng chống dịch COVID hiệu quả hơn.

Nhóm chúng em làm đề tài này nhằm nhắc nhở những người dân không đeo khẩu trang, đeo khẩu trang không đúng cách. Giúp người dân yên tâm hơn và nâng cao ý thức tuân thủ Thông điệp 5K, giảm thiểu lây lan bệnh dịch trong môi trường đông người, khép kín, dễ lây nhiễm ở nơi công cộng khó truy vết khi phát hiện ca mắc COVID-19.

## 2: Mục tiêu của đề tài

Mục tiêu chính của đề tài là xây dựng một **“chương trình cảnh báo người dùng đeo khẩu trang sử dụng mô hình học máy”** bằng nhận dạng khuôn mặt đáp ứng các chức năng sau:

1. Chức năng nhận diện khuôn mặt.

2. Chức năng kiểm tra người dùng đeo khẩu trang.

3. Chức năng cảnh báo và nhắc nhở người dùng đeo khẩu trang.

## 3: Nội dung nghiên cứu.

Mục tiêu xây dựng **“chương trình cảnh báo người dùng đeo khẩu trang sử dụng mô hình học máy”** như trên thì nhóm chúng em thực hiện những nội dung sau:

- Nghiên cứu và tìm hiểu các tài liệu liên quan về học máy.

- Xử lý ảnh đầu vào từ camera hoặc từ thư mục có sẵn.

- Kết nối các thư viện Keras, Sklearn, Numpy, Pandas, TensorFlow, OpenCV, Pillow…

- Lập trình và viết code cho hệ thống trên Colab.

- Chạy thử và điều chỉnh mô hình.

- Đánh giá kết quả thực hiện.

- Viết báo cáo.

## 4: Phạm vi nghiên cứu.

Ứng dụng chương trình này tại trường Đại Học Mỏ - Địa Chất – 18 Phố Viên

# CHƯƠNG II: KHẢO SÁT HỆ THỐNG

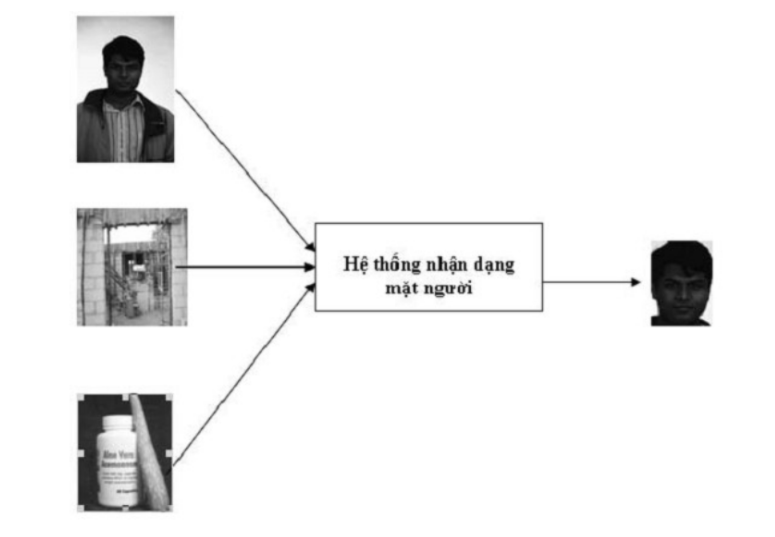
## 2.1: Nhiệm vụ cơ bản.

Nhiệm vụ chính của chương trình là nhận dạng khuôn mặt người dùng từ ảnh đầu vào (lấy trực tiếp trong webcam). huấn luyện có sẵn và đưa ra kết quả. Chương trình bắt đầu nhận dạng và nếu phát hiện người dùng không đeo khẩu trang thì sẽ đưa ra cảnh báo và nhắc nhở người dùng.

## 2.2: Mô tả hệ thống.

### 2.2.1: Bài toán nhận dạng khuôn mặt.

Ngày nay cùng với sự bùng nổ thông tin, sự phát triển công nghệ cao, sự giao tiếp giữa con người và máy tính đang thay đổi rất nhanh, giờ đây giao tiếp này không còn đơn thuần dùng những thiết bị cơ học như chuột, bàn phím… mà có thể thông qua các biểu hiện của khuôn mặt. Bên cạnh đó, công nghệ càng phát triển thì giá cả ngày càng giảm, thêm vào đó tốc độ xử lý của máy tính ngày càng cao, do đó hệ thống xử lý khuôn mặt đang được phát triển rất nhiều.



**Hình 1:** Nhận dạng mặt người

Trong số đó, có thể nói đến hệ thống phát hiện khuôn mặt, hệ thống này có thể giúp máy tính và con người giao tiếp với nhau tốt hơn. Những nghiên cứu trong hệ thống này chủ yếu dựa trên những thông tin trong ảnh để phát hiện vị trí khuôn mặt, làm bước đệm cho các ứng dụng tiếp theo. Rất nhiều nghiên cứu và ứng dụng được phát triển dựa trên hệ thống này. Đó là nhiệm vụ đầu tiên của bất kì hệ thống xử lý khuôn mặt nào. Tuy nhiên đây cũng là một thử thách rất lớn bởi phát hiện khuôn mặt dựa vào nhiều yếu tố như tỉ lệ, vị trí, hướng nhìn (từ trên xuống, quay,… ) kiểu chụp (chụp đối diện, chụp ngang,…). Ngoài ra, những cảm xúc của khuôn mặt, một số phần bị che, hoặc hướng ánh sáng cũng ảnh hưởng đến bài toán phát hiện khuôn mặt.

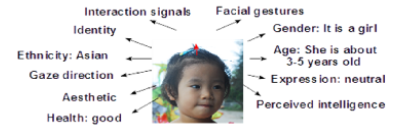
Hệ thống nhận dạng khuôn mặt là một ứng dụng máy tính tự động xác định hoặc nhận dạng một người nào đó từ một bức hình ảnh kỹ thuật số hoặc một khung hình video từ một nguồn video và còn là một loại phần mềm sinh trắc học ánh xạ. Hệ thống sẽ nhận diện các đặc điểm khuôn mặt của một cá nhân về mặt toán học và lưu trữ dữ liệu dưới dạng faceprint (dấu khuôn mặt). Công nghệ nhận diện khuôn mặt xác định 80 điểm nút trên khuôn mặt con người. Các điểm nút được sử dụng để đo các biến thể trên khuôn mặt người như: chiều dài, chiều rộng của mũi, miệng, độ sâu của hốc mắt và hình dạng của xương gò má và nhờ đó tạo nên “chìa khóa” cho khuôn mặt bạn và lưu lại trong phần mềm dưới dạng một thuật toán, mỗi khi có đối tượng, máy sẽ quét và phân tích dữ liệu xem khuôn mặt này có khớp với thuật toán đã lưu hay chưa. Một trong những cách để thực hiện điều này là so sánh các đặc điểm khuôn mặt chọn trước từ hình ảnh và một cơ sở dữ liệu về khuôn mặt. Bằng những thuật toán thông minh công nghệ này thực hiện so sánh phân tích dữ liệu giữa ảnh có sẵn với hình ảnh khuôn mặt người đó để cho ra kết quả.Với sự chính xác cao, công nghệ nhận diện khuôn mặt không chỉ được sử dụng trong lĩnh vực quân sự mà còn được ứng dụng trong quản trị doanh nghiệp. Cụ thể sử dụng trí tuệ nhân tạo nhận diện khuôn mặt để quản lý nhân sự, hỗ trợ hoạt động chấm công trong doanh nghiệp và có thể được so sánh với các dạng sinh trắc học khác như các hệ thống nhận dạng vân tay hay tròng mắt. nhận dạng khuôn mặt có nhiều ưu điểm:

- Một hệ thống nhận dạng khuôn mặt không đòi hỏi có sự tương tác trực tiếp giữa đối tượng được nhận dạng và hệ thống

 - Việc thu nhận dữ liệu (các bức ảnh về khuôn mặt con người) cho quá trình nhận dạng một con người dễ thực hiện hơn so với việc thu nhận các đặc điểm sinh trắc học khác (như việc thu nhận dấu vân tay, mống mắt ...)

- Dữ liệu và khuôn mặt phổ biến hơn so với các đặc trưng khác do sự bùng nổ các mạng xã hội Facebook, Twitter, Yahoo...), các dịch vụ chia sẻ dữ liệu da phương tiện (Youtube, Vimeo...) và sự phát triển nhanh chóng của các thiết bị thu nhận hình ảnh.

- Từ khuôn mặt của một người, ta có thể khai thác nhiều thông tin liên quan chứ không chỉ là danh tính của người đó, ví dụ như giới tính (Gender), màu da (Skin Color), hướng nhìn (Gaze Direction), chủng tộc, hành vi, sức khỏe, độ tuổi, cảm xúc…Ví dụ như hình 1.1 dưới đây:



**Hình 2:** Một số thông tin có thể tìm thấy trong một bức ảnh khuôn mặt.

### 2.2.2) Những khó khăn và thách thức trong bài toán nhận dạng khuôn mặt.

Bài toán nhận dạng mặt người được nghiên cứu từ những năm 70, người đầu tiên là Kanade. Tuy nhiên, đây là một bài toán khó nên những nghiên cứu hiện tại vẫn chưa đạt được kết quả mong muốn. Có thể kể đến những khó khăn của bài toán nhận dạng mặt người như dưới đây:

*a. Tư thế, góc chụp*: Ảnh chụp khuôn mặt có thể thay đổi rất nhiều bởi vì góc chụp giữa camera và khuôn mặt.Chẳng hạn như : chụp thẳng, chụp xéo bên trái hay xéo bên phải ,chụp từ trên xuống, chụp từ dưới lên,v.v…). Với các tư thế khác nhau, các thành phần trên khuôn mặt như mắt, mũi, miệng có thể bị khuất một phần hoặc thậm chí khuất hết

*b. Sự xuất hiện hoặc thiếu một số thành phần của khuôn mặt*: Các đặc trưng như râu mép, râu hàm, mắt kính,v.v… có thể xuất hiện hoặc không. Vấn đề này làm cho bài toán càng khó khăn hơn rất nhiều.

*c. Sự che khuất*: Khuôn mặt có thể bị che khuất bởi các đối tượng khác hoặc các vật thể khác.

*d. Hướng của ảnh*: Các ảnh của khuôn mặt có thể biến đổi rất nhiều với các góc quay khác nhau của trục camera. Chẳng hạn chụp với trục máy ảnh nghiêng làm cho khuôn mặt bị nghiêng so với trục của ảnh.

*e. Điều kiện của ảnh*: Ảnh được chụp trong các điều kiện khác nhau về: chiếu sang, về tính chất camera (máy kỹ thuật số, máy hồng ngoại,v.v…) ảnh hưởng rất nhiều đến chất lượng ảnh khuôn mặt.

### 2.2.3) Các ứng dụng của bài toán nhận diện khuôn mặt.

Ứng dụng của bài toán nhận dạng khuôn mặt có rất nhiều và đã được triển khai tương đối hiệu quả trong thực tế. Có thể kể đến một số ứng dụng điển hình sau đây**.**

**Xác minh tội phạm**

Dựa vào ảnh của một người, nhận dạng xem người đấy có phải là tội phạm hay không bằng cách so sánh với các tội phạm đang được lưu trữ. Hoặc có thể sử dụng camera để phát hiện tội phạm trong dám đông. Ứng dụng này giúp cơ quan an ninh quả lý con người tốt hơn.

**Camera chống trộm**

Các hệ thống camera sẽ xác định đâu là con người và theo dõi xem con người đó có làm gì phạm pháp hay không, ví dụ như lấy trộm đồ, xâm nhập bất hợp pháp vào một khu vực nào đó.

**Bảo mật**

Các ứng dụng về bảo mật rất đa dạng, một trong số đố là công nghệ nhận dạng mặt người của laptop, công nghệ này cho phép chủ nhân của máy tính chỉ cần ngồi trước máy là có thể đăng nhập được. Để sử dụng công nghệ này, ngườu dùng phải sử dụng một webcam để chụp ảnh khuôn mặt của mình và cho máy “học” thuộc các đặc điểm của khuôn mặt giúp cho quá trình đăng nhập sau này.

**Lưu trữ khuôn mặt**

Xác định mặt người có thể được ứng dụng trong các trạm rút tiền tự động (ATM) để lưu trữ khuôn mặt của người rút tiền. Hiện nay có những người bị người khác lấy trộm tiền nhưng lại báo với ngân hàng là mất thẻ và bị rút tiền trộm. Nếu lưu trữ được khuôn mặt của người rút tiền, ngân hàng có thể đối chứng và xử lý dễ dàng.

**Các ứng dụng khác**

* Điều khiển vào ra: văn phòng, công ty, trụ sở, máy tính, Paml,… Kết hợp thêm vân tay và mồng mắt. Cho phép nhân viên được ra vào nơi cần thiết.
* An ninh sân bat, xuất nhập cảnh (hiện nay cơ quan xuất nhập cảnh Mỹ đã áp dụng). Dùng camera quan sát để xác thực người nhập cảnh và kiểm tra xem ngườu đấy có phải là tội phạm hay phẩn tử khủng bố không.

Trong cuộc sống hiện nay với tình hình dịch covid đang rất căng thẳng thì ý thức phòng dịch của mỗi người dân là điều rất cần thiết. Số lượng người mắc tăng lên 1 cách đáng kể. Trong nhiều khu doanh nghiệp hay các khu dân cư, có 1 số ít những người chưa có ý thức phòng dịch còn lơ là trong thời điểm dịch bệnh đang căng thẳng, cụ thể hơn là có những người không tự giác đeo khẩu trang khi đi ra ngoài và đến nơi tập trung đông dân cư mà các lãnh đạo lại không có nhiều thời gian để trực tiếp theo dõi tất cả mọi người vì thế chúng ta cần có sự can thiệp của công nghệ để nâng cao hiệu quả phòng dịch, dễ dàng theo dõi và tiết kiệm thời gian. Và từ đấy nhóm chúng em đã tìm hiểu và ứng dụng 1 chương trình cần thiết phải có trong đời sống xã hội hiện nay nó là chương trình cảnh bảo người dùng không đeo khẩu trang sử dụng mô hình học máy để nâng cao í thức của người dân từ những hành động nhỏ nhất như là đeo khẩu trang.

Ảnh được xử lý

Xử lý ảnh

Ảnh đầu vào

Kết luận

**Hình 3:** Quá trình xử lý ảnh.

Camera nhận diện khuôn mặt

Truyền hình ảnh cần phân tích về máy chủ

Đối chiếu và nhận diện khuôn mặt người dùng với khối dữ liệu có sẵn trong hệ thống

**Hình 4:** Các bước cơ bản trong một hệ thống xử lý ảnh.

## 2.3: Quy trình xử lý.

### 2.3.1.Các chức năng tổng quát.

Hệ thống gồm 3 chức năng chính:

1. Chức năng nhận diện khuôn mặt.

2. Chức năng kiểm tra người dùng đeo khẩu trang.

3. Chức năng cảnh báo và nhắc nhở người dùng đeo khẩu trang.

### 2.3.2: Chi tiết về các chức năng.

#### 2.3.2.1: Chức năng nhận diện khuôn mặt.

**Mô tả:** Hệ thống camera tự động nhận diện khuôn mặt. Sau đó, dùng thuật toán nhận diện khuôn mặt của thư viện OpenCV để nhận diện khuôn mặt.

**- Trình tự thực hiện.**

1. Camera nhận diện hình ảnh khuôn mặt.

- **Kết quả:** Nhận diện khuôn mặt.

#### 2.3.2.2: Chức năng kiểm tra người dùng đeo khẩu trang.

**Mô tả:** Nhận diện khuôn mặt và dựa vào dữ liệu đã học để nhận biết xem người dùng có đeo khẩu trang hay không.

**- Trình tự thực hiện.**

1. Đối chiếu và nhận diện khuôn mặt người dùng với dựa trên dữ liệu mà hệ thống đã học được qua các tập dữ liệu có sẵn.

2. Hệ thống nhận diện người dùng không đeo khẩu trang hoặc đã đeo khẩu trang.

- **Kết quả:** Xác nhận xem người dùng đã đeo khẩu trang hay chưa.

#### 2.3.2.3: Chức năng nhắc nhở người dùng đeo khẩu trang.

**Mô tả:** Đưa ra cảnh báo và nhắc nhở đối với người dùng chưa đeo khẩu trang.

**- Trình tự thực hiện.**

1. Hệ thống hiển thị người dùng chưa đeo khẩu trang và lập tức nhắc nhở người dùng.

**- Kết quả:** Nhận diện và cảnh báo, nhắc nhở nếu người dùng chưa đeo khẩu trang.

# CHƯƠNG III: CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ

## 3.1) Lý thuyết về học máy.

**a) Machine learning là gì?**

Machine learning là gì? Machine learning là một lĩnh vực con của Trí tuệ nhân tạo (Artificial Intelligence) sử dụng các thuật toán cho phép máy tính có thể học từ dữ liệu để thực hiện các công việc thay vì được lập trình một cách rõ ràng.

[](https://nguyenvanhieu.vn/wp-content/uploads/2018/09/machine-learning-la-gi.jpg)

**Hình 5:** minh họa về học máy

Bạn đã hiểu rồi chứ? Chúng ta có thể làm cho máy tính học để làm công việc gì đó! Điều này nghe có vẻ khá trừu tượng. Nó có nghĩa rằng, chúng ta có thể lập trình cho các máy tính có khả năng tự học.

**b) Machine learning trên thực tế.**

Okay, chẳng có gì là hoàn hảo hết. Machine learning cũng có những giới hạn của nó. Chúng ta không thể nào xây dựng một cỗ máy thông minh để học dữ liệu từ cổ chí kim tới hiện tại. Tuy nhiên, đã có những ứng dụng thực tế mà machine learning làm rất tốt. Sau đây là các lĩnh vực phổ biến mà machine learing góp mặt:

**Xử lý ảnh:**

Bài toán xử lý ảnh (Image Processing) giải quyết các vấn đề phân tích thông tin từ hình ảnh hay thực hiện một số phép biến đổi. Một số ví dụ là:

**Gắn thẻ hình ảnh** (Image Tagging), giống như Facebook, một thuật toán tự động phát hiện khuôn mặt của bạn và bạn bè trên những bức ảnh. Về cơ bản, thuật toán này học từ những bức ảnh mà bạn tự gắn thẻ cho mình trước đó.

**Nhận dạng ký tự** (Optical Character Recognition), là một thuật toán chuyển dữ liệu trên giấy tờ, văn bản thành dữ liệu số hóa. Thuật toán phải học cách nhận biết ảnh chụp của một ký tự là ký tự nào.

**Ô tô tự lái** (Self-driving cars), một phần cơ chế sử dụng ở đây là xử lý ảnh. Một thuật toán machine learning giúp phát hiện các mép đường, biển báo hay các chướng ngại vật bằng cách xem xét từng khung hình video từ camera.

**Phân tích văn bản:**Phân tích văn bản (Text analysis) là công việc trích xuất hoặc phân lọi thông tin từ văn bản. Các văn bản ở đây có thể là các facebook posts, emails, các đoạn chats, tài liệu, …Một số ví dụ phổ biến là:

**Lọc spam** (Spam filtering), là một trong những ứng dụng phân loại văn bản được biết và sử dụng nhiều nhất. Ở đây, phân loại văn bản là xác định chủ đề cho một văn bản. Bộ lọc spam sẽ học cách phân loại một email có phải spam không dựa trên nội dung và tiêu đề của email.

**Phân tích ngữ nghĩa** (Sentiment Analysis), học cách phân loại một ý kiến là tích cực, trung tính hay tiêu cực dựa trên nội dung văn bản của người viết.

**Khai thác thông tin** (Information Extraction), từ một văn bản, học cách để trích xuất các thông tin hữu ích. Chẳng hạn như trích xuất địa chỉ, tên người, từ khóa, …

### Khai phá dữ liệu: Khai phá dữ liệu (Data mining) là quá trình khám phá ra các thông tin có giá trị hoặc đưa ra các dự đoán từ dữ liệu. Định nghĩa này có vẻ bao quát, nhưng bạn hãy nghĩ về việc tìm kiếm thông tin hữu ích từ một bảng dữ liệu rất lớn. Mỗi bản ghi sẽ là một đối tượng cần phải học, và mỗi cột là một đặc trưng. Chúng ta có thể dự đoán giá trị của một cột của bản ghi mới dựa trên các bản ghi đã học. Hoặc là phân nhóm các bản ghi của bản. Sau đây là những ứng dụng của khai phá dữ liệu:

**Phát hiện bất thường** (Anomaly detection), phát hiện các ngoại lệ, ví dụ như phát hiện gian lận thẻ tín dụng. Bạn có thể phát hiện một giao dịch là khả nghi dựa trên các giao dịch thông thường của người dùng đó.

**Phát hiện các quy luật** (Association rules), ví dụ, trong một siêu thị hay một trang thương mại điện tử. Bạn có thể khám phá ra khách hàng thường mua các món hàng nào cùng nhau. Dễ hiểu hơn, khách hàng của bạn khi mua món hàng A thường mua kèm món hàng nào? Các thông tin này rất hữu ích cho việc tiếp thị sản phẩm.

**Gom nhóm**(Grouping), ví dụ, trong các nền tảng SaaS, người dùng được phân nhóm theo hành vi hoặc thông tin hồ sơ của họ.

**Dự đoán**(Predictions), các cột giá trị (của một bản ghi mới trong database). Ví dụ, bạn có thể dự đoán giá của căn hộ dựa trên các dữ liệu về giá các căn hộ bạn đã có.

### Trò chơi điện tử & Robot:

* Trò chơi điện tử (Video games) và robot(Robotics) là lĩnh vực lớn có sự góp mặt của machine learning. Nếu ta có một nhân vật cần di chuyển và tránh các chướng ngại vật trong game. Machine learning có thể học và giải quyết công việc này thay bạn. Một kỹ thuật phổ biến được áp dụng trong trường hợp này là Học tăng cường (Reinforcement learning). Ở đó, máy sẽ học tăng cường với mục tiêu là giải quyết nhiệm vụ trên. Học tăng cường là tiêu cực nếu nó va phải chướng ngại vật, là tích cực nếu nó chạm tới đích.
* Một thành tựu gần đây nhất là cỗ máy Alpha Go của Google DeepMind đã đánh bại kỳ thủ cờ vậy số 1 thế giới. Trong khi cờ vây là một trò chơi có không gian trạng thái cực kỳ lớn.

**c)Machine learning làm việc ra sao?**

Một trong những cuốn sách đầu tiên tôi đọc hồi 10 năm trước là Cuốn sách này được viết năm 1997, nhưng kiến thức trong đó vẫn chuẩn xác cho tới ngày hôm nay.

Trong cuốn sách đó, tôi thích định nghĩa của tác giá về machine learning:

A computer program is said to learn to perform a task T from experience E, if its performance at task T, as measured by a performance metric P, improves with experience E over time

Tạm dịch: Một chương trình máy tính được cho là học để thực hiện một nhiệm vụ T từ kinh nghiệm E, nếu hiệu suất thực hiện công việc T của nó được đo bởi chỉ số hiệu suất P và được cải thiện bởi kinh nghiệm E theo thời gian.

[](https://nguyenvanhieu.vn/wp-content/uploads/2018/09/machine-learning.jpg)

**Hình 6:** mô tả học máy là gì?

Ví dụ: Một cỗ máy thực hiện chơi cờ (nhiệm vụ T), có thể học từ dữ liệu các ván cờ trước đó hoặc chơi với một chuyên gia (kinh nghiệm E). Khả năng chơi của cỗ máy là tỉ lệ số ván mà nó chiến thắng khi chơi với con người (hiệu suất P).

## 3.2. Deep Learning.

Khái niệm.

**- Học sâu** là một chức năng của [trí tuệ nhân tạo](https://vietnambiz.vn/tri-tue-nhan-tao-artificial-intelligence-ai-la-gi-20190924104039225.htm) (AI), bắt chước hoạt động của bộ não con người trong việc xử lí dữ liệu và tạo ra các mẫu để sử dụng cho việc ra quyết định.

Học sâu là tập con của [học máy](https://vietnambiz.vn/may-hoc-machine-learning-la-gi-ung-dung-thuc-tien-20190923225908014.htm)trong AI, có các mạng lưới có khả năng "học" mà không bị giám sát từ dữ liệu không có cấu trúc hoặc không được gắn nhãn.

* Học sâu đã phát triển cùng với thời đại kĩ thuật số, điều này đã mang lại sự bùng nổ dữ liệu dưới mọi hình thức và từ mọi khu vực trên thế giới. Dữ liệu này, gọi đơn giản là [dữ liệu lớn](https://vietnambiz.vn/du-lieu-lon-big-data-la-gi-nhung-kho-khan-khi-su-dung-du-lieu-lon-20190923114156622.htm), được lấy từ các nguồn như phương tiện truyền thông xã hội, công cụ tìm kiếm trên internet, nền tảng thương mại điện tử hoặc rạp chiếu phim trực tuyến,...
* Lượng dữ liệu khổng lồ này có thể truy cập dễ dàng và có thể được chia sẻ thông tin qua các ứng dụng Fintech như [điện toán đám mây](https://vietnambiz.vn/dien-toan-dam-may-cloud-computing-la-gi-20190909113101611.htm).
* Tuy nhiên, thông thường dữ liệu không có cấu trúc, rộng lớn đến mức có thể phải mất hàng thập kỉ để con người hiểu được nó và trích xuất thông tin liên quan. Các công ty nhận ra tiềm năng đáng kinh ngạc có thể có được từ việc tháo dỡ lượng thông tin phong phú này và ngày càng thích nghi với các hệ thống AI để được hỗ trợ tự động.
* Deep learning là một phần của một họ các phương pháp học máy rộng hơn dựa trên đại diện học của dữ liệu. Một quan sát (ví dụ như, một hình ảnh) có thể được biểu diễn bằng nhiều cách như một vector của các giá trị cường độ cho mỗi điểm ảnh, hoặc một cách trừu tượng hơn như là một tập hợp các cạnh, các khu vực hình dạng cụ thể, … Một vài đại diện làm khiến việc học các nhiệm vụ dễ dàng hơn (ví dụ, nhận dạng khuôn mặt hoặc biểu hiện cảm xúc trên khuân mặt) từ các ví dụ. Một trong những hứa hẹn của học sâu là thay thế các tính năng thủ công bằng các thuật toán hiệu quả đối với học không có giám sát hoặc nửa giám sát và tính năng phân cấp.

## 3.2. Một số thuật toán về nhận dạng khuôn mặt.

**1. One-shot learning**

- Ưu điểm: mỗi một người chỉ cần 1 vài, rất ít hoặc thậm chí chỉ 1 bức ảnh duy nhất (để khỏi tạo ra nhiều biến thể).

-Từ đầu vào là bức ảnh của một người, chúng ta sử dụng một kiến trúc thuật toán CNN đơn giản để dự báo người đó là ai.

- Nhược điểm: chúng ta phải huấn luyện lại thuật toán thường xuyên khi xuất hiện thêm một người mới vì shape của output thay đổi tăng lên 1. Rõ ràng là không tốt đối với các hệ thống nhận diện khuôn mặt của một công ty vì số lượng người luôn biến động theo thời gian.

**2. Learning similarity**

Learning similarity có thể trả ra nhiều hơn một ảnh là cùng loại với ảnh đầu vào tùy theo ngưỡng threshold.

Ngoài ra phương pháp này không bị phụ thuộc vào số lượng classes. Do đó không cần phải huấn luyện lại khi xuất hiện class mới.

Điểm mấu chốt là cần xây dựng được một model encoding đủ tốt để chiếu các bức ảnh lên một không gian eucledean n chiều. Sau đó sử dụng khoảng cách để quyết định nhãn của chúng.

Như vậy learning similarity có ưu điểm hơn so với one-shot learning khi không phải huấn luyện lại model khi mà vẫn tìm ra được ảnh tương đồng.

Vậy làm thế nào để học được biểu diễn của ảnh trong không gian euledean n chiều? Kiến trúc siam network sẽ giúp chúng ta thực hiện điều này một cách dễ dàng.

**3. Siam network**

Kiến trúc của Siam network dựa trên base network là một Convolutional neural network đã được loại bỏ output layer có tác dụng encoding ảnh thành véc tơ embedding. Đầu vào của mạng siam network là 2 bức ảnh bất kì được lựa chọn ngẫu nhiên từ dữ liệu ảnh. Output của Siam network là 2 véc tơ tương ứng với biểu diễn của 2 ảnh input. Sau đó chúng ta đưa 2 véc tơ vào hàm loss function để đo lường sự khác biệt giữa chúng. Thông thường hàm loss function là một hàm norm chuẩn bậc 2.

Khi sử dụng siam network chúng ta sẽ không cần phải lo lắng về vấn đề output shape thay đổi vì base network đã được loại bỏ layer cuối.

**4. facenet**

Facenet chính là một dạng siam network có tác dụng biểu diễn các bức ảnh trong một không gian eucledean chiều (thường là 128) sao cho khoảng cách giữa các véc tơ embedding càng nhỏ, mức độ tương đồng giữa chúng càng lớn.

Hầu hết các thuật toán nhận diện khuôn mặt trước facenet đều tìm cách biểu diễn khuôn mặt bằng một véc tơ embedding thông qua một layer bottle neck có tác dụng giảm chiều dữ liệu.

* Tuy nhiên hạn chế của các thuật toán này đó là số lượng chiều embedding tương đối lớn (thường >= 1000) và ảnh hưởng tới tốc độ của thuật toán. Thường chúng ta phải áp dụng thêm thuật toán PCA để giảm chiều dữ liệu để tăng tốc độ tính toán.
* Hàm loss function chỉ đo lường khoảng cách giữa 2 bức ảnh. Như vậy trong một đầu vào huấn luyện chỉ học được một trong hai khả năng là sự giống nhau nếu chúng cùng 1 class hoặc sự khác nhau nếu chúng khác class mà không học được cùng lúc sự giống nhau và khác nhau trên cùng một lượt huấn luyện.

Facenet đã giải quyết cả 2 vấn đề trên bằng các hiệu chỉnh nhỏ nhưng mang lại hiệu quả lớn:

* Base network áp dụng một mạng convolutional neural network và giảm chiều dữ liệu xuống chỉ còn 128 chiều. Do đó quá trình suy diễn và dự báo nhanh hơn và đồng thời độ chính xác vẫn được đảm bảo.

Sử dụng loss function là hàm triplot loss có khả năng học được đồng thời sự giống nhau giữa 2 bức ảnh cùng nhóm và phân biệt các bức ảnh không cùng nhóm. Do đó hiệu quả hơn rất nhiều so với các phương pháp trước đây

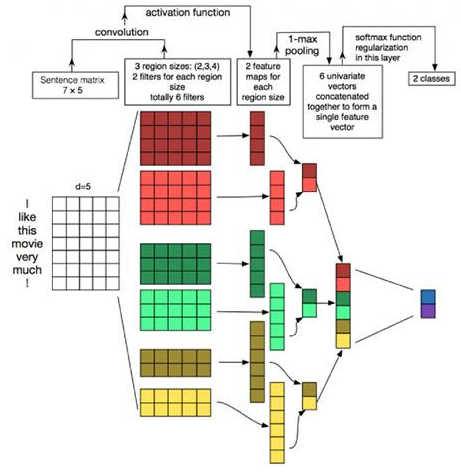
## 3.3. Thuật toán CNN

### 3.3.1. Cấu trúc của CNN.

Mạng CNN là một trong những tập hợp của lớp Convolution bị chồng lên nhau cũng như sử dụng hàm nonlinear activation như ReLU và tanh để kích hoạt trọng số trong node. Lớp này sau khi thông qua hàm thì sẽ được trọng số trong các node. Những lớp này sau khi đã thông qua hàm kích hoạt thì có thể tạo ra những thông tin trừu tượng hơn cho những lớp tiếp theo.

**Cấu trúc cơ bản của CNN thường bao gồm 3 phần chính là:**

* **Local receptive field**(trường cục bộ): Lớp này có nhiệm vụ tách lọc dữ liệu, thông tin ảnh và lựa chọn các vùng ảnh có giá trị sử dụng cao nhất.
* **Shared weights and bias** (trọng số chia sẻ): Lớp này giúp làm giảm tối đa lượng tham số có tác dụng chính của yếu tố này trong mạng CNN. Trong mỗi convolution sẽ có các feature map khác nhau và mỗi feature lại có khả năng giúp detect một vài feature trong ảnh.
* **Pooling layer**(lớp tổng hợp): Pooling layer là lớp cuối cùng và có tác dụng làm đơn giản các thông tin đầu ra. Có nghĩa là, sau khi đã hoàn tất tính toán và quét qua các lớp thì đến pooling layer để lược bớt các thông tin không cần thiết. Từ đó, cho ra kết quả theo như ý mà người dùng mong muốn.

****

Hình 7: CNN ứng dụng trong nhiều mô hình

**Mạng CNN sử dụng 3 ý tưởng cơ bản:**

* + **các trường tiếp nhận cục bộ** (local receptive field)
  + **trọng số chia sẻ** (shared weights)
  + **tổng hợp** (pooling).

**Những tham số trong CNN:**

1. Số các convolution layer: càng nhiều các convolution layer thì performance càng được cải thiện. Sau khoảng 3 hoặc 4 layer, các tác động được giảm một cách đáng kể
2. Filter size: thường filter theo size 5x5 hoặc 3x3
3. Pooling size: thường là 2x2 hoặc 4x4 cho ảnh đầu vào lớn
4. Cách cuối cùng là thực hiện nhiều lần việc train test để chọn ra được param tốt nhất.

**Một số mạng CNN nổi tiếng:**

* VGG-16 (hiện đã có VGG-19)
* Inception (GoogleLeNet)
* ResNet(viết tắt của residual network)
* Darknet-53
* Exception

### 3.3.2: CNN với bài toán nhận diện khuôn mặt

#### 3.3.2.1: Nhận dạng đối tượng

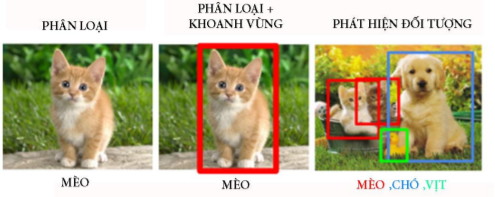
Một số khái niệm liên quan:

• Nhận dạng đối tượng (object recognition) là một thuật ngữ chung để mô tả một tập hợp các cách xử lý trong thị giác máy tính có liên quan đến việc xác định các đối tượng trong ảnh kỹ thuật số.

• Phân loại hình ảnh (image classification) liên quan đến việc dự đoán, phân loại các lớp thông tin của một đối tượng trong một hình ảnh.

• Khoanh vùng đối tượng (object localization) là việc xác định vị trí của một hoặc nhiều đối tượng trong một hình ảnh bằng hình chữ nhật xung quanh phạm vi của đối tượng bằng hộp chứa (bounding box).

• Phát hiện đối tượng (object detection) kết hợp cả hai nhiệm vụ nhận dạng đối tượng và khoanh vùng đối tượng. Phát hiện đối tượng là việc khoanh vùng và phân loại một hoặc nhiều đối tượng khác nhau trong một hình ảnh. Nhận dạng đối tượng khác với phát hiện đối tượng ở chỗ nhận dạng đối tượng mang ý nghĩa rộng hơn, bao gồm cả việc phân loại hình ảnh (đòi hỏi thuật toán xác định các lớp của đối tượng xuất hiện trong hình ảnh), cũng như phát hiện đối tượng (đòi hỏi thuật toán khoanh vùng, định vị được tất cả các đối tượng có trong hình ảnh)

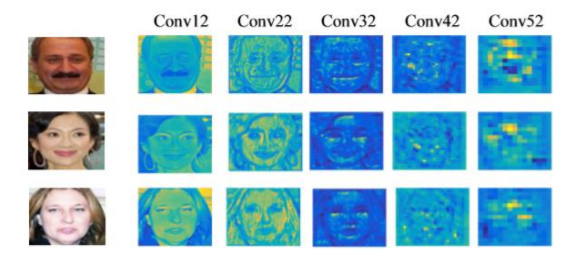


Hình 8: Sự khác biệt giữa nhận dạng đối tượng và phát hiện đối tượng

#### 3.3.2.2: Nhận dạng khuôn mặt.

Nhận dạng khuôn mặt là khả năng nhận diện, lưu trữ,so sánh và phân tích các mẫu dựa trên đường nét khuôn mặt để nhận dạng người từ hình ảnh hoặc video. Hệ thống thường sử dụng các công nghệ để thực hiện nhận dạng khuôn mặt như sử dụng sinh trắc học để ánh xạ các đặc điểm khuôn mặt từ ảnh hoặc video. Hệ thống so sánh thông tin này với một cơ sở dữ liệu đã lưu trữ về các khuôn mặt để tìm ra một kết quả khớp chính xác

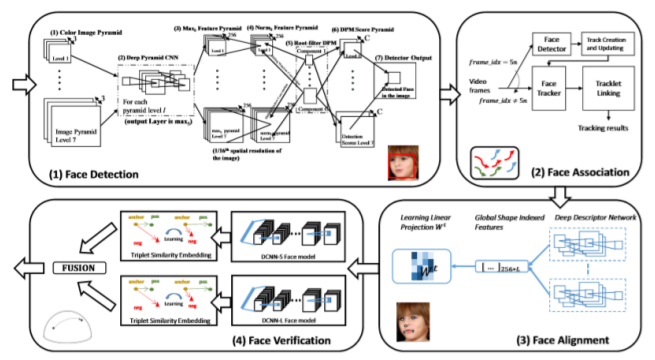
Nhận dạng khuôn mặt ngày càng được quan tâm trong nhiều lĩnh vực, như Trung Quốc có hệ thống Skynet để chấm điểm công dân, hay nhiều chuỗi cửa hàng lớn cũng sử dụng nhận dạng khuôn mặt khách hàng thân thiết để phân tích thói quen tiêu dùng. Trên thực tế, công nghệ nhận dạng khuôn mặt đã nhận được sự chú ý đáng kể vì nó có tiềm năng cho một loạt các ứng dụng liên quan đến thực thi pháp luật cũng như các doanh nghiệp khác.



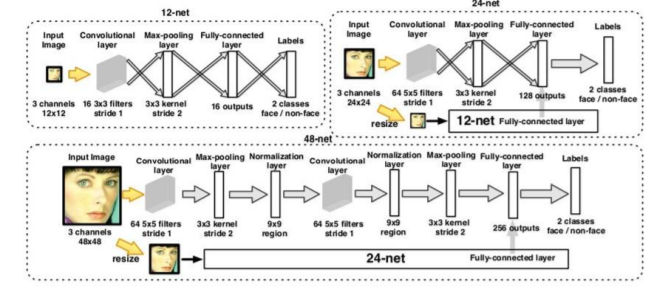
Hình 9: Một minh họa về một số bản đồ đặc trưng của các lớp DC12, conv22, conv32, conv42 và conv52 được đào tạo cho nhiệm vụ nhận dạng khuôn mặt.

#### 3.3.2.3: Cách thức hoạt động.

Phương pháp bao gồm các bước khác nhau để thực hiện nhận diện khuôn mặt tự động. Đầu tiên là thực hiện phát hiện khuôn mặt để khoanh vùng khuôn mặt trong từng khung hình ảnh và video. Sau đó, dữ liệu ở bước trên đuợc liên kết với các khuôn mặt được phát hiện với danh tính chung trên các video và căn chỉnh các khuôn mặt thành tọa độ bằng cách sử dụng các mốc được phát hiện. Cuối cùng, hệ thống thực hiện xác minh 31 khuôn mặt để tính toán độ tương tự giữa một cặp hình ảnh / video.

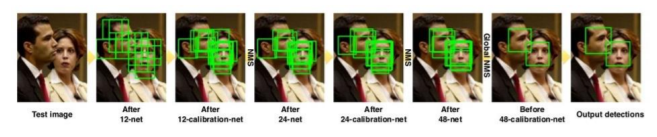


Hình10: Tổng quan về phương pháp nhận dạng khuôn mặt người bằng mạng CNN



Hình11 : Thiết kế mạng CNN để xử lý ảnh

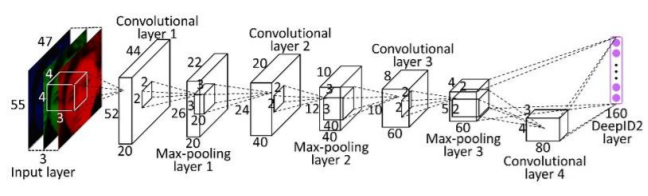
Từ mạng nơ ron trên, ta có thể thấy các bước xử lý ảnh từ trái sang phải. Lần lượt, ta có thể thấy hình ảnh các cửa sổ dò tìm (ô vuông màu xanh) đang giảm dần và hiệu chỉnh dần theo từng giai đoạn dò tìm:



Hình12: Các giai đoạn dò tìm và hiệu chỉnh

#### 3.3.2.4: Kết quả các thuật toán sử dụng mạng CNN.

Trong quá trình huấn luyện, 200 phần khuôn mặt được cắt với các vị trí, tỉ lệ và kênh màu sắc khác nhau. Mỗi phần khuôn mặt và phần đối xứng được đưa vào ConvNet. Hai vector 160 chiều được trích xuất từ phần khuôn mặt và phần đối xứng. Chọn 25 phần khuôn mặt tốt nhất



Hình13: Kiến trúc mạng của mô hình DeepID 2

### 3.3.3 Phóng đại ảnh (Scale ảnh).

Scale ảnh là việc chúng ta thay đổi kích thước dài, rộng của ảnh mà không làm thay đổi tính chất song song của các đoạn thẳng trên ảnh gốc so với các trục tọa độ X và Y. Trong opencv, chúng ta sẽ thay đổi kích thước của hình ảnh bằng hàm cv2.resize().

Theo định nghĩa về phép biến đổi hình học thì một biến đổi phóng đại các chiều (x, y) theo hệ số (a\_{1}, a\_{2}) sẽ có ma trận dịch chuyển M là ma trận đường chéo. Tức là ma trận vuông có đường chéo chính là [a1,a2] và các phần tử còn lại bằng 0. Khi đó phép dịch chuyển sẽ là:

T(x,y) = **M 󠆿**= =

### 3.3.4 Dịch chuyển ảnh (Translation)

Dịch chuyển ảnh thường được thực hiện trong trường hợp bạn muốn dịch chuyển ảnh đến các vị trí khác nhau. Ví dụ tới các góc trái, phải, ở giữa, bên trên, bên dưới. Phép dịch chuyển sẽ giữ nguyên tính chất song song của các đoạn thẳng sau dịch chuyển đối với các trục X hoặc Y nếu trước dịch chuyển chúng cũng song song với một trong hai trục này. Để dịch chuyển hình ảnh chúng ta phải xác định được (,) là các giá trị di chuyển ảnh theo trục x và y. Ma trận dịch chuyển **M** sẽ có dạng như bên dưới:

**M** =

Thật vậy. Giả sử mọi điểm ảnh đều nằm trên không gian 2 chiều. Khi đó ta coi chiều thứ 3 là một hằng số, chẳng hạn z=1. Khi đó phép biến đổi (x,y) bất kì theo ma trận dịch chuyển **M** sẽ là:

T(x,y) = **M 󠆿**= =

Như vậy mỗi điểm tọa độ (x,y) đã được dịch chuyển tới một tọa độ mới là (x +, y + )

### 3.3.5 Xoay ảnh (Rotation)

Xoay ảnh được hiểu là ta quay một bức ảnh theo một góc xác định quanh một điểm nào đó. Phép xoay sẽ không đảm bảo tính chất song song với các trục X hoặc Y như phép dịch chuyển nhưng nó sẽ bảo toàn độ lớn góc. Nếu 3 điểm bất kì tại ảnh gốc tạo thành một tam giác thì khi biến đổi qua phép xoay ảnh, chúng sẽ tạo thành một tam giác đồng dạng với tam giác ban đầu. Phép xoay của một hình ảnh tương ứng với một góc θ đạt được bằng một ma trận dịch chuyển **M** như sau:

**M** =

Ngoài ra OpenCV hỗ trợ một phép xoay phóng đại (scaled rotation) với khả năng vừa biến đổi ảnh theo phép xoay theo tâm xác định và điều chỉnh lại kích thước ảnh sau xoay. Như vậy bạn có thể xoay theo bất kì vùng nào mà bạn ưa chuộng hơn. Phép dịch chuyển ma trận được đưa ra như sau:

trong đó:

α=scale.cos(θ)β=scale.sin(θ)

(,) là tọa độ tâm của phép xoay và scale là độ phóng đại.

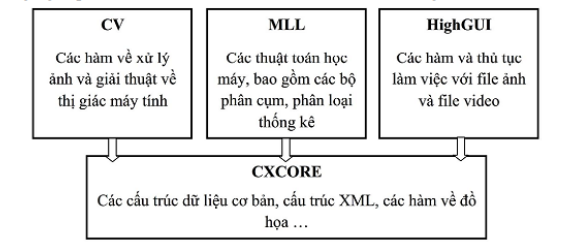
## 3.4. Các thư viện được sử dụng trong chương trình.

### 3.4.1 Thư viện OpenCV.

OpenCv (Open Source Computer Vision) là một thư viện mã nguồn mở về thị giác máy với hơn 500 hàm và hơn 2500 các thuật toán đã được tối ưu về XLA, và các vấn đề liên quan tới thị giác máy. OpenCv được thiết kế một cách tối ưu, sử dụng tối đa mạnh của các dòng chip đa lõi… để thực hiện các phép tính toán trong thời gian thực, nghĩa là tốc độ đáp ứng của nó thể đủ nhanh cho các ứng dụng thông thường.

OpenCv là thư viện được thiết kế để chạy trên nhiều nền tảng khác nhau (cross-platform),nghĩa là nó có thể chạy trên hệ điều hành Window, Linux, Mac, iOS… Việc sử dụng thư viện OpenCv tuân theo các quy định về sử dụng phần mềm mã nguồn mở BSD do đó bạn có thể sử dụng thư viện này một cách miễn phí cho các mục đích phi thương mại lẫn thương mại.

Dự án về OpenCv được khởi động từ những năm 1999, đến năm 2000 nó được giới thiệu trong một hội nghị của IEEE về các vấn đề trong thị giác máy và nhận dạng, tuy nhiên bản OpenCV 1.0 mãi tới tận năm 2006 mới chính thức được công bố và năm 2008 bản 1.1 (prerelease) mới được ra đời. Tháng 10 năm 2009, bản OpenCV thế hệ thứ hai ra đời (thường gọi là phiên bản 2.x), phiên bản này có giao diện của C++ (khác với phiên bản trước có giao diện của C) và có nhiều điểm khác biệt so với phiên bản thứ nhất.

Thư viện OpenCV ban đầu được sự hỗ trợ từ Intel, sau đó được hỗ trợ bởi Willow Garage, một phòng thí nghiệm chuyên nghiên cứu về công nghệ robot. Cho đến nay, OpenCV vẫn là thư viện mở, được phát triển bởi nguồn quỹ không lợi nhuận (none-profit foundation) và được sự hưởng ứng rất lớn của cộng đồng.

Hình 14: Giới thiệu về thư viện OpenCV

### 3.4.2 Thư viện Keras.

**Keras** là một thư viện nơ-ron mã nguồn mở được viết bằng ngôn ngữ lập trình Python. Nó có khả năng chạy trên đầu trang của Tensorflow, Microsoft Cognitive Toolkit, R, Theano hoặc StripeML. Được thiết kế để cho phép thử nghiệm nhanh với các mạng thần kinh sâu, nó tập trung vào việc thân thiện với người dùng, mô-đun và mở rộng.

**Keras** được coi là một thư viện ‘high-level’ với phần ‘low-level’ (còn được gọi là backend) có thể là TensorFlow, CNTK, hoặc Theano (sắp tới Theano sẽ không được duy trì nâng cấp nữa). Keras có cú pháp đơn giản hơn TensorFlow rất nhiều. Với mục đích giới thiệu về các mô hình nhiều hơn là các sử dụng các thư viện deep learning, tôi sẽ chọn Keras với TensorFlow là ‘backend’.

Một số tính năng của Keras:

* Keras ưu tiên trải nghiệm của người lập trình
* Keras đã được sử dụng rộng rãi trong doanh nghiệp và cộng đồng nghiên cứu
* Keras giúp dễ dàng biến các thiết kế thành sản phẩm
* Keras hỗ trợ huấn luyện trên nhiều GPU phân tán
* Keras hỗ trợ đa backend engines và không giới hạn bạn vào một hệ sinh thái.

### 3.4.3 Thư viện Sklearn.

Sklearn là thư viện về machine learning phổ biến nhất của Python. Nó viết sẵn các thuật toán lằng nhằng phức tạp, bạn chỉ việc nhét dữ liệu vào, chờ nó tính toán rồi lấy kết quả trên.

Thư viện này tích hợp rất nhiều thuật toán hiện đại và cố điển giúp bạn vừa học vừa tiến hành đưa ra các giải pháp hữu ích cho bài toán của bạn một cách đơn giản.

Scikit-learn (Sklearn) là thư viện mạnh mẽ nhất dành cho các thuật toán học máy được viết trên ngôn ngữ Python. Thư viện cung cấp một tập các công cụ xử lý các bài toán machine learning và statistical modeling gồm: classification, regression, clustering, và dimensionality reduction.

Thư viện được cấp phép bản quyền chuẩn FreeBSD và chạy được trên nhiều nền tảng Linux. Scikit-learn được sử dụng như một tài liệu để học tập.

Để cài đặt scikit-learn trước tiên phải cài thư viện SciPy (Scientific Python). Những thành phần gồm:

- Numpy: Gói thư viện xử lý dãy số và ma trận nhiều chiều

- SciPy: Gói các hàm tính toán logic khoa học

- Matplotlib: Biểu diễn dữ liệu dưới dạng đồ thị 2 chiều, 3 chiều

- IPython: Notebook dùng để tương tác trực quan với Python

- SymPy: Gói thư viện các kí tự toán học

- Pandas: Xử lý, phân tích dữ liệu dưới dạng bảng

### 3.4.4 Thư viện NumPy

Numpy (Numeric Python): là một thư viện toán học rât phổ biến và mạnh mẽ của Python. NumPy được trang bị các hàm số đã được tối ưu, cho phép làm việc hiệu quả với ma trận và mảng, đặc biệt là dữ liệu ma trận và mảng lớn với tốc độ xử lý nhanh hơn nhiều lần khi chỉ sử dụng Python đơn thuần.

Nếu bạn muốn trở thành một lập trình viên khoa học dữ liệu chuyên sâu, bạn cần phải nắm rõ numpy. Đây là một trong những thư viện hữu ích nhất của python, đặc biệt là nếu bạn đang tìm hiểu về các con số. Vì phần lớn Khoa học Dữ liệu và Máy học xoay quanh Thống kê, nên việc thực hành trở nên quan trọng hơn nhiều.

NumPy được phát triển bởi Jim Hugunin. Phiên bản ban đầu là Numarray được phát triển, có một số chức năng bổ sung. Năm 2005, Travis Oliphant đã tạo ra gói NumPy bằng cách kết hợp các tính năng của Numarray và gói Numeric.

Sử dụng NumPy, lập trình viên có thể thực hiện các thao tác sau:

* Các phép toán toán học và logic trên mảng.
* Các biến đổi Fourier và các quy trình để thao tác shape.
* Các phép toán liên quan đến đại số tuyến tính. NumPy tích hợp sẵn các hàm cho đại số tuyến tính và tạo số ngẫu nhiên.

NumPy - Sự thay thế hoàn hảo cho MatLab

NumPy thường được sử dụng cùng với các gói như SciPy (Python Scientific) và Mat-plotlib (thư viện vẽ đồ thị). Sự kết hợp này được sử dụng rộng rãi để thay thế cho MatLab, một nền tảng phổ biến cho tính toán kỹ thuật. Tuy nhiên, Python thay thế cho MatLab hiện được xem như một ngôn ngữ lập trình hoàn thiện và hiện đại hơn.Điều quan trọng hơn cả là Numpy là một thư viện mã nguồn mở, miễn phí so với MatLab là một thư viện mã nguồn đóng và phải trả phí .

### 3.4.5 Thư viện Pandas

Pandas là một thư viện mã nguồn mở được xây dựng dựa trên NumPy, sử dụng thao tác và phân tích dữ liệu, được thiết kế để cho phép bạn làm việc với dữ liệu được gắn nhãn hoặc quan hệ theo cách trực quan hơn.

* Có thể xử lý tập dữ liệu khác nhau về định dạng: chuỗi thời gian, bảng không đồng nhất, ma trận dữ liệu
* Khả năng import dữ liệu từ nhiều nguồn khác nhau như CSV, DB/SQL
* Có thể xử lý vô số phép toán cho tập dữ liệu: subsetting, slicing, filtering, merging, groupBy, re-ordering, and re-shaping,..
* Xử lý dữ liệu mất mát theo ý người dùng mong muốn: bỏ qua hoặc chuyển sang 0
* Xử lý, phân tích dữ liệu tốt như mô hình hoá và thống kê
* Tích hợp tốt với các thư viện khác của python
* Cung cấp hiệu suất tốt

Pandas là một thư viện Python toàn diện; một nguồn lực để thực hiện phân tích và thao tác dữ liệu; bất kỳ loại xử lý, phân tích, lọc và tổng hợp dữ liệu nào. Thư viện này được xây dựng dựa trên ngôn ngữ lập trình Python và có thể được sử dụng cho bất kỳ quy trình thu thập thông tin chi tiết từ dữ liệu nào. Trong nghiên cứu khoa học dữ liệu, Pandas là một trong những công cụ quan trọng trong việc hỗ trợ, xử lý và phân tích dữ liệu với mã nguồn mở nhanh, mạnh, linh hoạt và dễ sử dụng

Về cơ bản Pandas có thể được coi là ngôi nhà dữ liệu của người dùng. Thông qua thư viện này, bạn có thể làm quen với các dữ liệu của mình bằng cách sắp xếp, phân tích và biến đổi chúng. 1 số chức năng của pandas:

- Tính toán số liệu thống kê, trả lời các câu hỏi về dữ liệu như giá trị trung bình, tối đa, tối thiểu của mỗi cột. Cột A có tương quan với cột B không? Sự phân bố dữ liệu trong cột C trông như thế nào?...

- Làm sạch dữ liệu bằng cách thực hiện những việc như xóa các giá trị bị thiếu và lọc các hàng và cột theo một số tiêu chí.

- Trực quan hóa dữ liệu với sự trợ giúp từ Matplotlib, biểu đồ thanh, đường kẻ, biểu đồ,....

- Lưu trữ các dữ liệu đã được làm sạch, chuyển đổi chúng thành CSV, tệp hoặc các cơ sở dữ liệu.

Trước khi bắt đầu mô hình hóa các hình ảnh trực quan phức tạp, bạn cần hiểu rõ về bản chất của tập dữ liệu và Pandas là con đường tốt nhất để thực hiện điều đó.

Hai thành phần chính của Pandas là Seriesvà DataFrame: Một Series về cơ bản là một cột và một DataFrame là một bảng đa chiều được tạo thành từ một tập hợp các Chuỗi (Series).

### 3.4.6 Thư viện TensorFlow

Tensorflow là một thư viện có mã nguồn mở, được dùng để tính toán machine learning với quy mô lớn. TensorFlow kết hợp một loạt các mô hình và thuật toán machine learning cùng deep learning, từ đó làm cho chúng trở nên hữu ích bằng những phép toán. TensorFlow sử dụng [**Python**](https://wiki.tino.org/python-la-gi/) để cung cấp một API [**front-end**](https://wiki.tino.org/frontend-la-gi-backend-la-gi/) thuận tiện cho việc xây dựng các ứng dụng với [**framework**](https://wiki.tino.org/framework-la-gi/), đồng thời thực thi các ứng dụng đó bằng ngôn ngữ C++ để đạt hiệu suất cao hơn.

TensorFlow cho phép các nhà phát triển tạo một biểu đồ để thực hiện các tính toán. Mỗi nút trong biểu đồ đại diện cho một phép toán và mỗi kết nối đại diện cho dữ liệu. Do đó, thay vì xử lý các chi tiết nhỏ như tìm cách thích hợp để chuyển đầu ra của một chức năng với đầu vào của chức năng khác, nhà phát triển có thể tập trung vào logic tổng thể của ứng dụng.

TensorFlow cung cấp tất cả những điều này cho lập trình viên theo phương thức của ngôn ngữ Python. Vì Python khá dễ học và làm việc, ngoài ra còn cung cấp nhiều cách tiện lợi để ta hiểu được làm thế nào các high-level abstractions có thể kết hợp cùng nhau. Node và tensor trong TensorFlow là các đối tượng Python, và các ứng dụng TensorFlow bản thân chúng cũng là các ứng dụng Python.

Các operation toán học thực sự thì không được thi hành bằng Python. Các thư viện biến đổi có sẵn thông qua TensorFlow được viết bằng các binary C++ hiệu suất cao. Python chỉ điều hướng lưu lượng giữa các phần và cung cấp các high-level abstraction lập trình để nối chúng lại với nhau. TensorFlow 2.0, được ra mắt vào tháng 10 năm 2019, cải tiến framework theo nhiều cách dựa trên phản hồi của người dùng, để dễ dàng và hiệu quả hơn khi làm việc cùng nó (ví dụ: bằng cách sử dụng các Keras API liên quan đơn giản cho việc train model). Train phân tán dễ chạy hơn nhờ vào API mới và sự hỗ trợ cho TensorFlow Lite cho phép triển khai các mô hình trên khá nhiều nền tảng khác nhau. Tuy nhiên, nếu đã viết code trên các phiên bản trước đó của TensorFlow thì bạn phải viết lại, đôi lúc 1 ít, đôi lúc cũng khá đáng kể, để tận dụng tối đa các tính năng mới của TensorFlow 2.0. Và ở thời điểm hiện tại, TensorFlow chính là thư viện mã nguồn mở cho machine learning nổi tiếng nhất thế giới, được phát triển bởi các nhà nghiên cứu từ Google. Việc hỗ trợ mạnh mẽ các phép toán học để tính toán trong machine learning và deep learning đã giúp việc tiếp cận các bài toán trở nên đơn giản, nhanh chóng và tiện lợi hơn nhiều. Component của TensorFlow gồm tensor và graph và các thuật toán nổi bật được hỗ trợ bởi TensorFlow như: Deep learning classification, Deep learning wipe and deep,…

Kiến trúc TensorFlow hoạt động được chia thành 3 phần:

– Tiền xử lý dữ liệu

– Dựng model

– Train và ước tính model

Hiện nay,Tensorflow được xem như một trong những phương tiện trung gian giúp tính toán cho các số lượng có trong sản xuất và đồng thời trở thành một công cụ không thể thiếu trong Machine Learning. Từ đó, phục vụ cho nhu cầu học tập cũng như nghiên cứu một cách dễ dàng hơn.

### 3.4.7 Thư viện Pillow

Pillow là một fork từ thư viện PIL của Python được sử dụng để xử lý hình ảnh. So với PIL thì Pillow được cập nhật thường xuyên và đánh giá cao hơn. (PIL đã không được cập nhật từ năm 2009).

Pillow tự mô tả mình là “ngã ba PIL thân thiện”, PIL là Thư viện hình ảnh Python hiện không còn tồn tại. May mắn thay, Pillow vẫn còn rất nhiều và cung cấp chức năng chỉnh sửa hình ảnh toàn diện.

Về nguyên tắc, bạn có thể sử dụng nó làm cơ sở của một loại ứng dụng Photoshop nhẹ bằng cách sử dụng Tkinter hoặc PyQT, nhưng trường hợp sử dụng điển hình của nó là để xử lý back-end, ví dụ: tạo hình thu nhỏ và thêm biểu trưng hoặc hình mờ vào hình ảnh được tải lên trang web .

Mặc dù có khả năng mạnh mẽ và toàn diện nhưng nó cực kỳ dễ sử dụng và tôi sẽ giới thiệu cho hầu hết người dùng những tính năng hữu ích nhất của nó. Trong bài viết này, tôi sẽ đề cập đến:

• Cài đặt Gối

• Mở hình ảnh và truy xuất thông tin hình ảnh cơ bản

• Lưu bản sao của hình ảnh

• Thay đổi kích thước

• Tạo hình thu nhỏ

• Xoay

• Cắt xén

• Đặt các màu pixel riêng lẻ

• Cải thiện, màu sắc, độ tương phản, độ sáng và độ sắc nét

• Thêm biểu trưng hoặc hình mờ

# CHƯƠNG IV: XÂY DỰNG MÔ HÌNH VÀ THIẾT KẾ HỆ THỐNG

## 4.1. Giới thiệu hệ thống

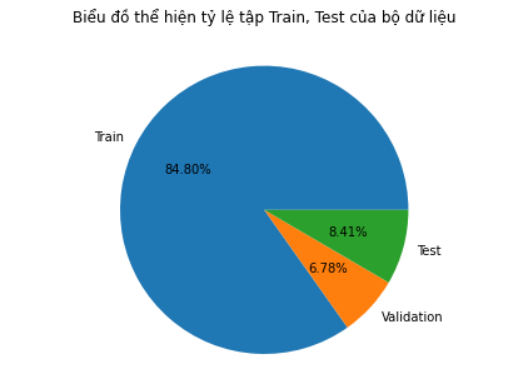
Nhiệm vụ chính của chương trình là nhận dạng khuôn mặt người dùng từ ảnh đầu vào (lấy trực tiếp trong webcam). Nhận dạng bằng cách dựa vào các dữ liệu đã huấn luyện có sẵn và đưa ra kết quả. Chương trình bắt đầu nhận dạng và nếu phát hiện người dùng không đeo khẩu trang thì sẽ đưa ra cảnh báo và nhắc nhở người dùng.

Về tính ứng dụng, hệ thống có thể sử dụng các điện thoại thông minh hoặc một máy tính để bàn cùng webcam để làm hệ thống nhận diện khuôn mặt. Hệ thống có thể trở thành một phần phụ trợ đối với các mô hình kinh doanh, như hệ thống nhận diện và nhắc nhở khách hàng đeo khẩu trang trong tình hình dịch bệnh phức tạp hiện nay.

## 4.2: Mô tả tập dữ liệu

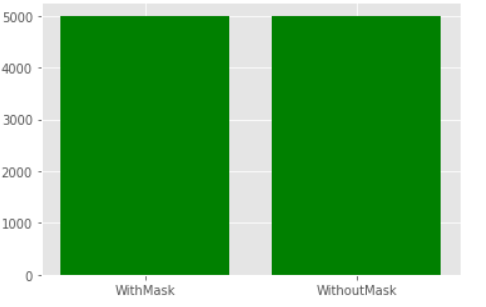
Bộ dữ liệu gồm có 3 tập Train, Validation và Test với 2 thuộc tính trong mỗi tập là:

* WithMask:là những hình ảnh được gắn nhãn là có khẩu trang
* WithoutMask: là những hình ảnh được gắn nhãn là không có khẩu trang

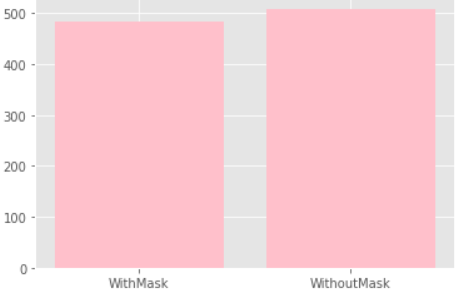


Hình15: Biểu đồ thể hiện tỷ lệ tập Train và Test của bộ dữ liệu

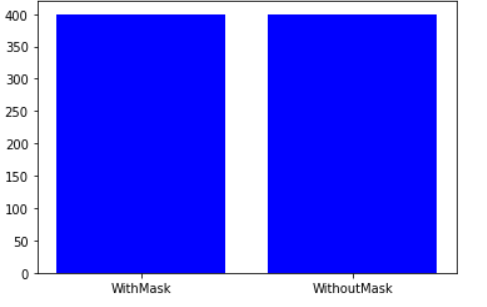
- Ở trong bộ dữ liệu gần 11000 ảnh này chúng em chia tập dữ liệu thành 90:10 tương ứng với 90% bộ dữ liệu là dùng để đưa bộ dữ liệu vào để học còn 10% dùng để chạy thử.



Hình16: Biểu đồ thể hiện kích thước bộ dữ liệu tập Train

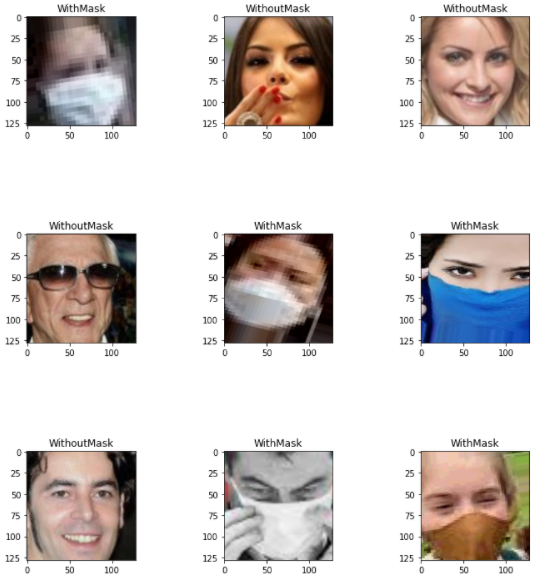


Hình17: Biểu đồ thể hiện kích thước bộ dữ liệu tập Test



Hình18: Biểu đồ thể hiện licks thước bộ dữ liệu tập Validation

- Ở trong mỗi tập Train, Validation và Test thì gồm có 50% withmark(hình ảnh được gán nhãn là đeo khẩu trang) và 50% Withoutmark( hình ảnh được gán nhãn là không đeo khẩu trang).



Hình 19: Một số hình ảnh ngẫu nhiên của bộ dữ liệu

## 4.3: Thiết kế hệ thống

Với các chức năng nêu trên, chương trình được chia thành 3 phần chính:

1. Chức năng nhận diện khuôn mặt.

2. Chức năng kiểm tra người dùng đeo khẩu trang.

3. Chức năng cảnh báo và nhắc nhở người dùng đeo khẩu trang.

Phát hiện khuôn mặt

Nhận Dạng

Cảnh báo và nhắc nhở

Kiểm tra

Sai

A

Đúng

Hiển thị kết quả

Hình 20: Sơ đồ thực hiện chương trình

## 4.4: Xây dựng mô hình

### 4.4.1: Bài toán 1: Mô hình RETINAFACE:

RetinaFace là một hệ thống nhận diện sử dụng dụng kỹ thuật học tập đa tác vụ để dự đoán đồng thời độ tin cậy, khuôn mặt, năm điểm mốc khuôn mặt, dự đoán mô hình khuôn mặt 3D và sự tương ứng của khuôn mặt với từng pixel. Bằng cạh sử dụng các mạng trong số nhỏ, RetinaFace có thể chạy thời gian thực trên một lõi CPU duy nhất cho hình ảnh có độ phân giải VGA.

* **Phương thức tái tạo mô hình 3D.**

Để tạo ra khuôn mặt 3D từ hình ảnh 2D, mô hình học có giám sát không thể triển khai do các mô hình 3D thực của 3D không dẽ dàng để thu thập được. Cùng với sự phát triển của các mô hình tái tạo 3D nổi bật là Mesh Decoder, RetinaFace đưa ra mô hình học tự giám sát mới avf khắc phục các yếu điểm.

* **Xác định vị trí khuôn mặt bằng kĩ thuật stingle-shot đa lớp**

Mô hình gồm 3 thành phần chính:

* Mạng đa tầng trích xuất đặc trưng. Mô hình mạng lấy hình ảnh đầu vào và xuất ra năm khu vực đặc trưng với các tỷ lệ khác nhau.
* Mô-dun phân tích bối cảnh: để tăng cường năng lựuc mô hình hóa, mạng tích chập biến dạng (DNC) được sử dụng trên mô-dun này trong các vũng đặc trưng
* Hàm tối ưu đa tác vụ của mô hình tầng trích xuất đặc trưng bao gồm: hàm tối ưu của lớp softmax đối với các việc phân loại khuôn mặt đầu vào. Hàm tối ưu ảu vùng bouding box của khuôn mặt. Hàm tối ưu nằm điểm landmarks của khuôn mặt. Hàm tối ưu điểm 3D mesh
* **Tái tạo và tối ưu tọa độ khuôn mặt 3D**

Việc tối ưu các tọa độ được dự đoán trong không gian 3 chiều sao cho khớp với ảnh đầu vào được thực hiện bằng hàm tối ưu đa tác vụ

Sau khi ảnh được cắt từ video, thì nó sẽ được định dạng kích thước giống kích của của bộ dữ liệu trong mô hình huấn luyện. Sau đó, ảnh này sẽ chuyển đổi thành những tham số có định dạng giống mô hình mẫu. Khi đó các tham số sinh ra được từ tập dữ liệu huấn luyện sẽ được sử dụng để thẩm định lại tính tích hợp của mô hình trên tập dữ liệu của hình ảnh vừa được trích xuất. Dựa vào kết quả thu được ta tiến hành hiển thị kết quả lên màn hình người dân có đeo khẩu trang hay không. Nếu người đó không đeo khẩu trang thì ngay lập tức được nhắc nhở đeo khẩu trang thông qua lời nói trực tiếp. Việc nhắc nhở này được thực thi qua sợ hồ trợ của thư viện “pyttxs3”, thư viện hỗ trợ chuyển văn bản thành dọng nói trong python.



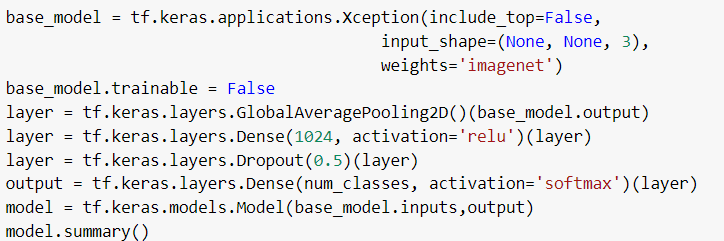
Hình 21: Ảnh lấy từ webcam với mô hình RentinaFace

### 4.4.2: Bài toán 2: Sử dụng phương pháp Transfer Learning với mô hình Xception (đã huấn luyện trên ImageNet)

“**Transfer learning** là việc ứng dụng kỹ năng/tri thức mình học được từ vấn đề này (source domain – Ds), với ứng dụng này (source task – Ts) sang vấn đề khác (target domain -Dt) với ứng dụng khác (target task – Tt) có liên quan. Transfer learning nhằm cải thiện việc học hàm f(.) cho ứng dụng Tt trên miền Dt”

Quá trình **Transfer learning** sẽ tận dụng lại các đặc trưng được học từ pre-trained model. Kiến trúc mô hình sử dụng transfer learning bao gồm 2 phần

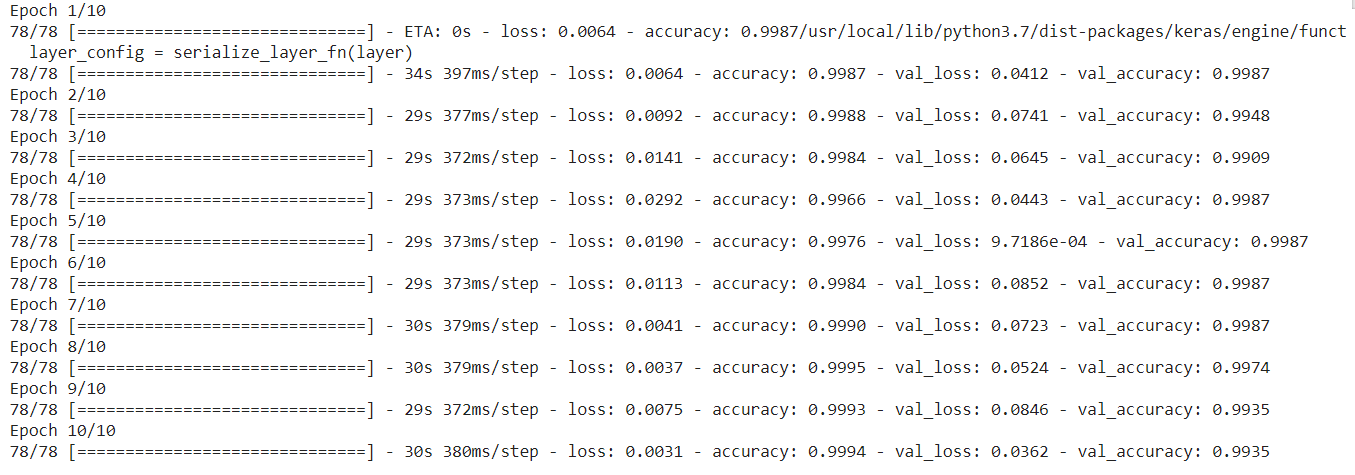
* Một mạng **Based network** có tác dụng trích lọc đặc trưng, based network này được trích xuất từ một phần của pre-trained model sau khi loại bỏ các top fully connected layers
* Các lớp **Fully Connected Layers** giúp giảm chiều dữ liệu và tính toán phân phối xác suất ở output. Bản chất Fully Connected Layers chính là một mạng **Multiple Layer Perceptron (MLP)** - một kiến trúc nguyên thủy nhất của thuật toán neural network. Tùy vào các bài toán cụ thể sẽ điều chỉnh số lượng các units ở output
* **Sử dụng Transfer Learning với mô hình Xception**



Từ tập huấn luyện đã có sẵn sử dụng kèm theo với phương pháp Transfer Learning với mô hình Xception để đưa vào huấn luyện cho mô hình để đưa ra được kết quả tốt nhất.

Ở đây include\_top = False để loại bỏ lớp Fully Connected trên cùng.

* **Thiết lập các tham số và huấn luyện.**

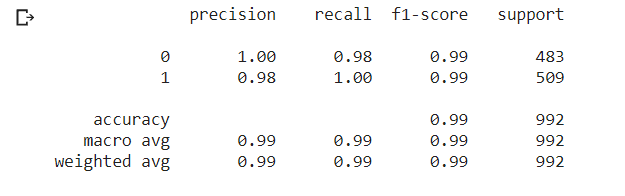


Thử nghiệm mô hình với epochs 10 và đưa ra được kết quả về độ chính xác của mô hình, với epochs càng cao thì độ chính xác của mô hình càng chuẩn.



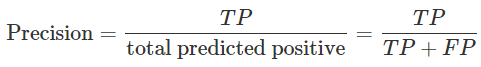
Hình 22: Biểu đồ thể hiện độ chính xác của mô hình

* **Kết quả về độ chính xác của mô hình qua một số độ đo.**



Kết quả độ chính xác của mô hình với những độ đo:

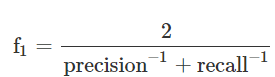
* Precision: Độ chuẩn xác trả lời cho câu hỏi trong các trường hợp được dự báo là dương tính thì có bao nhiêu trường hợp là đúng ? Và tất nhiên độ chuẩn xác càng cao thì mô hình của chúng ta càng tốt trong việc phân loại hồ sơ BAD. Công thức của độ chuẩn xác được tính trên nhóm dương tính như sau:



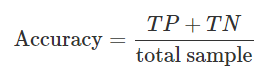
* Recall: Độ phủ đo lường tỷ lệ dự báo chính xác các trường hợp dương tính trên toàn bộ các mẫu thuộc nhóm dương tính. Công thức của recall như sau:



* F1-score: f1 Score là trung bình điều hòa giữa độ chuẩn xác và độ phủ. Do đó nó là chỉ số đại diện trong việc đánh giá tỷ lệ dự báo đúng của các trường hợp mẫu dương tính (hồ sơ BAD).



* Accuracy: Khi xây dựng mô hình phân loại chúng ta sẽ muốn biết một cách khái quát tỷ lệ các trường hợp được dự báo đúng trên tổng số các trường hợp là bao nhiêu. Tỷ lệ đó được gọi là độ chính xác. Độ chính xác giúp ta đánh giá hiệu quả dự báo của mô hình trên một bộ dữ liệu. Độ chính xác càng cao thì mô hình của chúng ta càng tốt. Độ chính xác được tính bằng tổng số trường hợp dự báo đúng trên cả âm tính và dương tính chia cho tổng số mẫu:



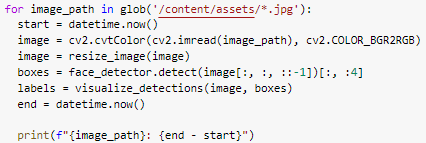


Hình 23: Hiển thị ra những kết quả dự đoán của mô hình

Với độ chính xác lên đến 99% thì mô hình này cho kết quả dự đoán rất chính xác:

* Ground Truth = 0, Predicted = 0, thì cho ra kết quả là người dùng có đeo khẩu trang.
* Ground Truth = 1, Predicted = 1, thì cho ra kết quả là người dùng không đeo khẩu trang.
* **Thử nghiệm mô hình**

**Thử nghiệm với hình ảnh**



Đầu tiên chúng ta đưa một số hình ảnh vào để chạy thử xem kết quả mà mô hình dự đoán như thế nào với:

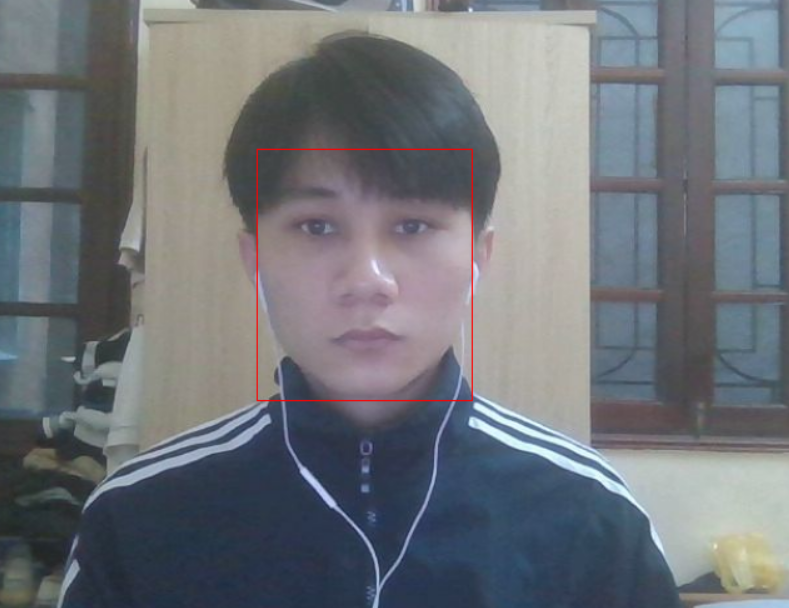
* Khung hiện màu đỏ là không đeo khẩu trang
* Khung hiện màu xanh lá là đeo khẩu trang

Dưới đây là kết quả của một số hình ảnh khi đưa vào chạy thử:





### 4.4.3: Kết quả thử nghiệm trực tiếp bằng webcam:

Sau khỉ đưa vào Test với những hình ảnh có sẵn thì chúng ta tiếp tục mang nó vào chạy thử với Webcam để xem kết quả đánh giá của mô hình cho ra như thế nào:

Hình 24: Hình ảnh người dùng không đeo khẩu trang



Hình 25: Hình ảnh người dùng đeo khẩu trang

# CHƯƠNG V: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN CỦA ĐỀ TÀI

## 5.1: Kết luận:

Bài báo cáo của bọn em đã xây dựng thành công chương trình phát hiện và nhắc nhở người không đeo khẩu trang nhằm hỗ trợ công tác giám sát người dân thực hiện đúng quy định của chính phủ về việc đeo khẩu trang nơi công cộng. Nhất là vào thời điểm hiện tại, khi tốc độ truyền nhiễm của dịch bệnh chưa có dấu hiệu suy giảm.

Chương trình được viết bằng ngôn ngữ python và sử dụng 1 số thư viện mã nguồn mở OpenCV, Tensorflow, Keras,... Để xây dựng hệ thống giám sát và nhắc nhở người dân đeo khẩu trang ở những nơi đông người như: trung tâm thương mại, trường học,...

Qua bài toán chúng em có biết thêm được một số công nghệ và một số thư viện để áp dụng trong bài toán, biết cách xây dựng các biểu đồ để đánh giá và so sánh các thuộc tính ở trong bài toán.

## 5.2: Hướng phát triển đề tài:

* Xây dựng thêm chức năng lưu dữ liệu của người dùng không đeo khẩu trang
* Xây dựng thêm việc đo nhiệt độ cơ thể của người dân để nắm được sức khỏe hiện tại của người đó.
* Đưa đề tài vào các mô hình thực tế như website hoặc app để dẽ tiếp cận với người dùng

Trong quá trình thực hiện đề tài này, chúng em đã cố gắng để tìm hiểu nhưng vì lượng kiến thức còn có hạn nên chúng em chỉ có thể làm được một số chức năng. Hơn nữa các công nghệ trên đối với chúng em cũng là khá mới mẻ nên chúng em phải tiến hành nghiên cứu nhiều hơn nhưng do thời gian nghiên cứu còn hạn chế nên sản phẩm còn chưa được hoàn thiện nên chúng em rất mong nhận được sự thông cảm của Thầy/Cô giáo.

Chúng em xin chân thành cảm ơn!

# TÀI LIỆU THAM KHẢO

1. **VnExpress, Đại dịch thay đổi thế giới năm 2020**

<https://vnexpress.net/dai-dich-thay-doi-the-gioi-nam-2020-4207154.html>

1. **Thị giác máy tính.**

<https://thigiacmaytinh.com/nhan-dien-khuon-mat-p1-face-detection/>

1. **Thuật toán CNN**

<https://topdev.vn/blog/thuat-toan-cnn-convolutional-neural-network/>

1. **Dữ Liệu từ nền tảng Kaggle**

<https://www.kaggle.com/ashishjangra27/face-mask-12k-images-dataset/>

1. **Model exception đã huấn luyện trên ImageNet**

<https://storage.googleapis.com/tensorflow/keras-applications/xception/xception_weights_tf_dim_ordering_tf_kernels_notop.h5>

1. **Thư viện OpenCV**

<https://topdev.vn/blog/opencv-la-gi-hoc-computer-vision-khong-kho/>

1. **Thư viện tensorflow**

<https://timviec365.vn/blog/tensorflow-la-gi-thu-vien-ma-nguon-mo-cho-machine-learning-new13179.html>

1. **Deep learning, Thư viện keras**

<https://itzone.com.vn/vi/article/gioi-thieu-ve-deep-learning-thu-vien-keras/>

1. **Thư viện numpy**

<https://codelearn.io/sharing/tim-hieu-thu-vien-numpy-trong-python>

1. **Thư viện matplotlib**

<https://viblo.asia/p/gioi-thieu-ve-matplotlib-mot-thu-vien-rat-huu-ich-cua-python-dung-de-ve-do-thi-yMnKMN6gZ7P>

1. **Thư viện Sklearn**

<https://helpex.vn/article/moi-thu-ban-can-biet-ve-thu-vien-scikit-learn-python-608fc7db101a480c64e9a0cd>