

SỞ GIÁO DỤC VÀ ĐÀO TẠO TP. ĐÀ NẴNG
TRƯỜNG THPT CHUYÊN LÊ QUÝ ĐÔN



**HỆ THỐNG TÍCH HỢP AI NHẬN DIỆN NGƯỜI ĐEO KHẨU TRANG
VÀ NHẬN DẠNG KHUÔN MẶT**

Mã đề tài: 49.KHKT

MỤC LỤC

A. GIỚI THIỆU CHUNG:	4
1. Giới thiệu ý tưởng:	4
2. Mục tiêu và nhiệm vụ:	4
3. Đối tượng nghiên cứu:	4
4. Phạm vi và giới hạn nghiên cứu:	5
5. Ý nghĩa của nghiên cứu:	5
a. Đối với bản thân học sinh:	5
b. Đối với xã hội:	5
B. TỔNG QUAN	5
1. Virus SARS-CoV-2:	6
2. Khuyến cáo của Bộ Y tế:	7
3. Một số thống kê đáng chú ý:	7
a. Trên thế giới:	7
b. Tại Việt Nam:	9
c. Không đeo khẩu trang khi tiếp xúc gần:	10
C. CÔNG NGHỆ - KỸ THUẬT SỬ DỤNG:	12
1. Thành phần cấu tạo và ngôn ngữ sử dụng:	12
2. Thư viện và model AI sử dụng:	12
3. Cơ sở dữ liệu:	12
4. Một số khái niệm về AI và ML:	12
a. Trí tuệ nhân tạo:	12
b. Machine learning:	14
5. Thư viện Tensorflow:	15
6. Thư viện OpenCV:	16
8. Thư viện Numpy:	16
9. Trình biên tập mã Visual Studio Code:	17
a. Khái niệm:	17
b. Tính năng:	17
c. Lí do nên sử dụng Visual Studio Code:	17
8. Công nghệ và thuật toán sử dụng trong nhận diện khẩu trang:	19
a. Object Detection:	19
b. Supervised learning:	20

c. Feature Pyramid Network:	21
d. MobileNet:	22
9. Thuật toán và công nghệ sử dụng trong nhận dạng khuôn mặt:	23
a. Nhận dạng khuôn mặt:	23
b. Unsupervised learning:	24
c. PCA (Principal Component Analysis):	25
d. Nhận diện khuôn mặt bằng Eigenface:	27
10. Nhận diện khẩu trang:	31
a. Tạo file TFRecord từ dataset để dễ dàng cho việc training:	31
b. Chạy training:	31
c. Tensorboard:	31
d. Chương trình chính:	31
11. Nhận dạng khuôn mặt:	31
a. Chuẩn bị ảnh để train:	31
b. Xử lý hình ảnh trước khi train:	32
c. Chạy training:	32
d. Chạy nhận dạng:	33
12. Thiết kế ứng dụng web:	33
a. Sử dụng cơ sở dữ liệu phpmyadmin:	33
b. Kết nối cơ sở dữ liệu bằng python:	33
c. Thiết kế web bằng HTML5 và CSS:	34
d. Xử lý dữ liệu bằng PHP:	34
e. Cách hoạt động của hệ thống: [06/05]	34
E. KẾT QUẢ THỰC HIỆN ĐỀ TÀI:	35
1. Chức năng hiện có:	35
a. Một số hình ảnh nghiên cứu và thử nghiệm:	35
b. Kết quả đạt được:	37
2. Ưu điểm:	38
3. Nhược điểm:	38
F. HƯỚNG PHÁT TRIỂN:	38
G. TÀI LIỆU THAM KHẢO:	39

A. GIỚI THIỆU CHUNG:

1. Giới thiệu ý tưởng:

Bệnh dịch Covid-19 vẫn đang hoành hành khắp nơi trên thế giới, để lại nhiều hậu quả nặng nề trong nhiều khía cạnh của cuộc sống. Ở Việt Nam, tuy chính quyền đã có nhiều biện pháp quyết liệt để phòng chống dịch, nhưng cũng không thể tránh khỏi những trường hợp người dân thiếu ý thức gây lây lan dịch bệnh trong cộng đồng. Một trong những nguyên nhân dẫn đến tình trạng này đó là việc người dân không chịu đeo khẩu trang khi đi ra nơi công cộng.

Nhận thấy những khó khăn trong công tác quản lý và giám sát việc đeo khẩu trang của người dân tại nơi công cộng, chúng em lên ý tưởng về việc sử dụng trí tuệ nhân tạo để nhận diện và phát hiện danh tính những học sinh không mang khẩu trang nơi công cộng, với mong muốn áp dụng công nghệ hiện đại vào việc nâng cao ý thức của người dân trong việc phòng chống dịch bệnh. Trước hết, chúng em sẽ thí điểm trong quy mô trường học và có thể mở rộng sau này.

2. Mục tiêu và nhiệm vụ:

Ngay từ khi xây dựng ý tưởng, nhóm đã đề ra mục tiêu đó là áp dụng công nghệ nhận diện vật thể và nhận dạng khuôn mặt để phát triển một "trí tuệ nhân tạo" có khả năng giám sát việc đeo khẩu trang của học sinh tại trường học. Với ý tưởng này, chúng em mong muốn được đóng góp một phần khả năng vào công cuộc phòng chống dịch bệnh, giảm thiểu khả năng lây nhiễm trong trường học và nâng cao ý thức phòng chống dịch của học sinh.

Để thực hiện mục tiêu đề ra, nhiệm vụ của nhóm chúng em bao gồm:

- Tìm hiểu về công nghệ nhận diện vật thể và nhận dạng khuôn mặt.
- Thu thập dữ liệu hình ảnh phục vụ cho việc xây dựng mô hình "trí tuệ nhân tạo".
- Xây dựng website để quản lý và theo dõi.

3. Đối tượng nghiên cứu:

Học sinh, sinh viên trong trường học.

4. Phạm vi và giới hạn nghiên cứu:

Phạm vi nghiên cứu: các bài báo cáo, các công trình nghiên cứu và mã nguồn mở của công nghệ nhận diện vật thể và nhận dạng khuôn mặt được đăng tải trên Internet.

Giới hạn nghiên cứu:

- Nghiên cứu công nghệ và thuật toán để nhận diện khẩu trang và nhận dạng danh tính.
- Thử nghiệm trong quy mô trường học.

5. Ý nghĩa của nghiên cứu:

a. Đối với bản thân học sinh:

Thông qua việc nghiên cứu và thực hiện đề tài, bản thân chúng em đã trau dồi và tích lũy được rất nhiều kiến thức cho bản thân:

- Đặt nền tảng vững chắc cho khả năng tự học lập trình, nghiên cứu công nghệ mới, tiếp cận kịp thời với xu hướng “trí tuệ nhân tạo” trên toàn cầu, tạo tiền đề cho phát triển tương lai học tập ở các bậc cao hơn.
- Hiểu rõ về sự nguy hiểm, mức độ lây lan của dịch bệnh Covid – 19. Nâng cao ý thức của chúng em về việc đeo khẩu trang nói riêng và phòng chống dịch bệnh nói chung.

b. Đối với xã hội:

Đề tài của chúng em sẽ góp phần trong công cuộc phòng chống dịch bệnh và ngăn cản sự lây lan dịch bệnh trong cộng đồng. Tuy chưa được hoàn chỉnh nhưng sản phẩm sẽ tạo sự khích lệ, động viên cũng như nêu ra vấn đề mà các bạn học sinh nói riêng và toàn xã hội nói chung cần đặc biệt quan tâm trong những năm gần đây.

B. TỔNG QUAN

1. Virus SARS-CoV-2:

Virus corona gây hội chứng hô hấp cấp tính nặng 2, viết tắt **SARS-CoV-2** (*Severe acute respiratory syndrome corona virus 2*), trước đây có tên là **virus corona mới 2019 (2019-nCoV)** và cũng được gọi là **virus corona ở người 2019 (HCoV-19 hoặc hCoV-19)**, là một chủng coronavirus gây ra bệnh viêm đường hô hấp cấp do virus corona 2019 (COVID-19), xuất hiện lần đầu tiên vào tháng 12 năm 2019, trong đợt bùng phát đại dịch COVID-19 ở thành phố Vũ Hán, Trung Quốc và bắt đầu lây lan nhanh chóng, sau đó trở thành một đại dịch toàn cầu. Vào ngày 12 tháng 1 năm 2020, nó được Tổ chức Y tế Thế giới gọi tên là 2019-nCoV, dựa trên một phương thức đặt tên cho virus corona mới. Đến ngày 11 tháng 2 năm 2020, Ủy ban Quốc tế về Phân loại Virus (ICTV) quyết định đặt tên chính thức cho chủng virus corona mới này là SARS-CoV-2 khi họ phân tích rằng nó cùng loài với virus SARS từng gây ra đại dịch năm 2003 nhưng là một chủng khác của loài. Virus này là một loại virus corona RNA liên kết đơn chính nghĩa. Trong khoảng thời gian đầu của đại dịch COVID-19, các nhân viên nghiên cứu đã phát hiện chủng virus này sau khi họ tiến hành đo lường kiểm tra axit nucleic và dò tra trình tự bộ gen ở mẫu vật lấy từ người bệnh.

Virus corona đã biết gây ra cảm mạo cùng với các triệu chứng khá nghiêm trọng giống như Hội chứng hô hấp Trung Đông (MERS) và Hội chứng hô hấp cấp tính nghiêm trọng (SARS). SARS-CoV-2 là phân dạng của virus corona mà từ trước đây chưa bao giờ phát hiện ở trong cơ thể người. Tháng 12 năm 2019 tới nay, thành phố Vũ Hán của tỉnh Hồ Bắc, Trung Quốc liên tục trông coi tra xét bệnh cúm trải rộng và bệnh tật tương quan, phát hiện nhiều trường hợp bệnh viêm phổi mang tính virus nổi dậy, tất cả cùng chẩn đoán là viêm phổi mang tính virus/truyền nhiễm phân phổi. Ủy ban Sức khỏe Vệ sinh Nhà nước Trung Quốc nhận định đây là bệnh truyền nhiễm loại B, chiếu theo quản lý loại A.

Virus corona chủng mới chủ yếu lây lan qua các giọt bắn trong không khí khi một cá nhân bị nhiễm bệnh ho hoặc hắt hơi trong phạm vi khoảng 0,91 m đến 1,8 m. Trong số 41 trường hợp ban đầu, hai phần ba có tiền sử tiếp xúc với Chợ

bán buôn hải sản Hoa Nam. Tháng 5 năm 2020, một nghiên cứu tại Đại học Hong Kong - Trung Quốc cũng cho biết virus này cũng lây qua mắt cao gấp 100 lần so với SARS

2. Khuyến cáo của Bộ Y tế:

Để chủ động phòng, chống dịch COVID-19 trong trạng thái “bình thường mới”, Bộ Y tế mong muốn và kêu gọi mỗi người dân Việt Nam cùng nhau thực hiện Chung sống an toàn với đại dịch COVID-19. Bộ Y tế gửi đến Bạn “Thông điệp 5K: Khẩu trang – Khử khuẩn – Khoảng cách – Không tụ tập – Khai báo y tế” với các nội dung chính sau đây:

- **KHẨU TRANG:** Đeo khẩu trang vải thường xuyên tại nơi công cộng, nơi tập trung đông người; đeo khẩu trang y tế tại các cơ sở y tế, khu cách ly.
- **KHỬ KHUẨN:** Rửa tay thường xuyên bằng xà phòng hoặc dung dịch sát khuẩn tay. Vệ sinh các bề mặt/ vật dụng thường xuyên tiếp xúc (tay nắm cửa, điện thoại, máy tính bảng, mặt bàn, ghế...). Giữ vệ sinh, lau rửa và để nhà cửa thông thoáng.
- **KHOẢNG CÁCH:** Giữ khoảng cách khi tiếp xúc với người khác.
- **KHÔNG TỤ TẬP** đông người.
- **KHAI BÁO Y TẾ:** thực hiện khai báo y tế trên App NCOVI; cài đặt ứng dụng BlueZone tại địa chỉ <https://www.bluezone.gov.vn> để được cảnh báo nguy cơ lây nhiễm COVID-19.

3. Một số thống kê đáng chú ý:

a. Trên thế giới:

Theo trang thống kê trực tuyến worldometers.info, tính đến sáng ngày 12/10/2021 (giờ Việt Nam), thế giới ghi nhận có tổng cộng 238.942.670 ca nhiễm COVID-19, trong đó 4.871.568 ca tử vong và 216.186.074 ca bình phục.

Mỹ tiếp tục là quốc gia chịu ảnh hưởng nghiêm trọng nhất bởi dịch bệnh trên toàn thế giới. Tính đến nay, nước này ghi nhận có 45.258.845 ca nhiễm COVID-19, trong đó 734.164 ca tử vong vì dịch bệnh.

Tại châu Âu, hiện tại châu lục này ghi nhận 60.530.487 ca mắc COVID-19, trong đó 1.246.625 ca tử vong. Hết ngày 11/10, châu lục này ghi nhận đã có thêm 128.807 ca nhiễm mới và 2.007 ca tử vong vì COVID-19. Tính đến nay, Anh ghi nhận có 8.193.769 ca nhiễm và 137.763 ca tử vong vì COVID-19. Nga ghi nhận số ca lây nhiễm mới nhiều thứ 2 trong khu vực, với 29.409 ca. Các quốc gia Pháp, Tây Ban Nha, Italy, Đức... lần lượt xếp sau Anh và Nga về mức độ ảnh hưởng của đại dịch COVID-19.

Châu Á hiện vẫn đang là tâm dịch COVID-19 trên toàn thế giới. Tính đến nay, châu lục này đã có tổng cộng 77.235.108 ca nhiễm và 1.140.258 ca tử vong vì COVID-19 tính đến thời điểm hiện tại. Riêng tại châu Á có 74.087.590 ca được điều trị khỏi; 2.007.260 ca đang được điều trị tích cực và chỉ còn 30.251 ca bệnh nặng.

Ấn Độ hiện vẫn là quốc gia chịu tác động nặng nề nhất vì COVID-19 trong khu vực, tổng số bệnh nhân và số trường hợp mắc và không qua khỏi do dịch bệnh nguy hiểm này lên lần lượt là 33.984.466 ca và 450.991 ca. Tính đến nay, Thổ Nhĩ Kỳ ghi nhận có 7.475.115 ca nhiễm COVID-19 và 66.368 ca tử vong vì dịch bệnh, là quốc gia xếp thứ 2 về mức độ chịu ảnh hưởng bởi dịch bệnh tại châu lục.

Tại Bắc Mỹ, khu vực này ghi nhận có 54.414.245 ca nhiễm COVID-19, trong đó 1.105.721 ca tử vong vì dịch bệnh. Các nước chịu tác động nặng nề nhất bởi đại dịch tại khu vực bao gồm: Mỹ, Mexico, Canada, Cuba, Guatemala, Costa Rica, Panama...

Tại Nam Mỹ, khu vực này hiện đã có 38.038.108 ca, trong đó 1.161.156 ca tử vong vì đại dịch. Các quốc gia Brazil, Argentina, Colombia, Peru, Chile... lần lượt đứng đầu khu vực vì mức độ ảnh hưởng của đại dịch COVID19. Brazil hiện vẫn là quốc gia chịu ảnh hưởng nặng

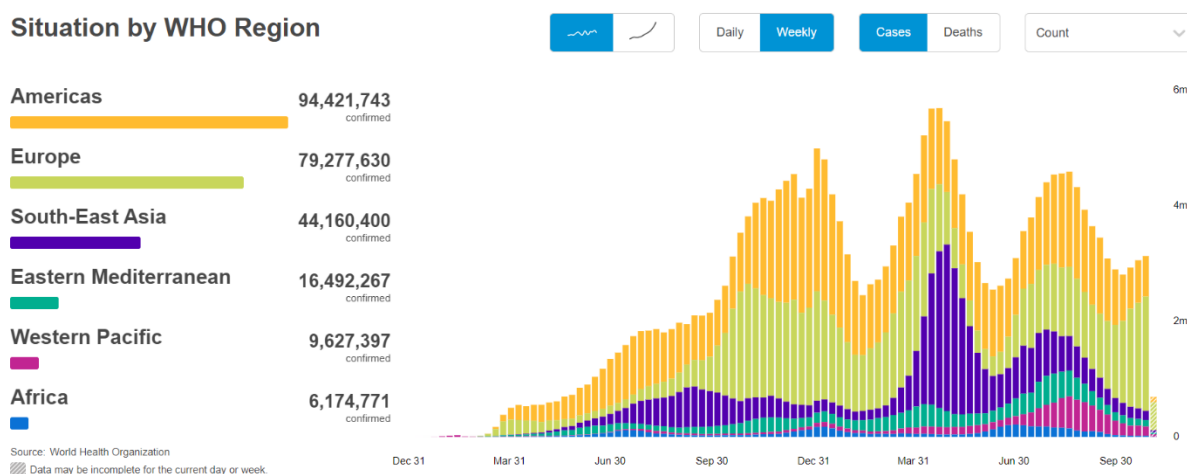
nề nhất bởi dịch bệnh tại khu vực và thứ 3 thế giới. Tính đến nay, quốc gia này ghi nhận có 21.582.738 ca nhiễm, trong đó 601.213 ca tử vong vì COVID-19.

Tại châu Phi, tính đến nay, châu lục này ghi nhận 8.465.365 ca nhiễm, trong đó 214.584 ca tử vong vì dịch bệnh. Nam Phi tiếp tục là quốc gia chịu ảnh hưởng nặng nề nhất bởi dịch bệnh. Nước này ghi nhận có 2.912.346 ca nhiễm COVID-19, trong đó 88.346 ca tử vong vì dịch bệnh. Xếp sau Nam Phi về mức độ ảnh hưởng bởi đại dịch COVID-19 bao gồm, Morocco, Tunisia, Ethiopia, Libya, Ai Cập...

Châu Đại dương ghi nhận có 258.636 ca nhiễm COVID-19, trong đó 3.209 ca tử vong vì dịch bệnh. Australia, Fiji và French Polynesia vẫn đang dẫn đầu khu vực về mức độ ảnh hưởng bởi đại dịch.

Tại ASEAN, tính đến nay, toàn khối ASEAN ghi nhận có tổng cộng hơn 12,5 triệu ca mắc COVID-19, trong đó 269.043 ca tử vong.

Nhìn chung, tình hình dịch bệnh tại Đông Nam Á đang tiếp tục xu thế hạ nhiệt. Số ca mắc mới và tử vong tại các nước đi ngang trong mấy ngày gần đây.



b. Tại Việt Nam:

Tình hình dịch COVID-19 tại Việt Nam

- Kể từ đầu dịch đến nay Việt Nam có 984.805 ca mắc COVID-19, đứng thứ 38/223 quốc gia và vùng lãnh thổ, trong khi với tỷ lệ số ca

nhễm/1 triệu dân, Việt Nam đứng thứ 151/223 quốc gia và vùng lãnh thổ (bình quân cứ 1 triệu người có 9.996 ca nhiễm).

- Đợt dịch thứ 4 (từ ngày 27/4/2021 đến nay):
 - Số ca nhiễm mới ghi nhận trong nước là 979.840 ca, trong đó có 839.983 bệnh nhân đã được công bố khỏi bệnh.
 - Có 4 tỉnh, thành phố không có ca lây nhiễm thứ phát trên địa bàn trong 14 ngày qua: Ninh Bình, Yên Bái, Bắc Kạn, Cao Bằng.
 - Các địa phương ghi nhận số nhiễm tích lũy cao trong đợt dịch này: TP HCM (441.216), Bình Dương (240.347), Đồng Nai (74.065), Long An (35.990), Tiền Giang (18.703).



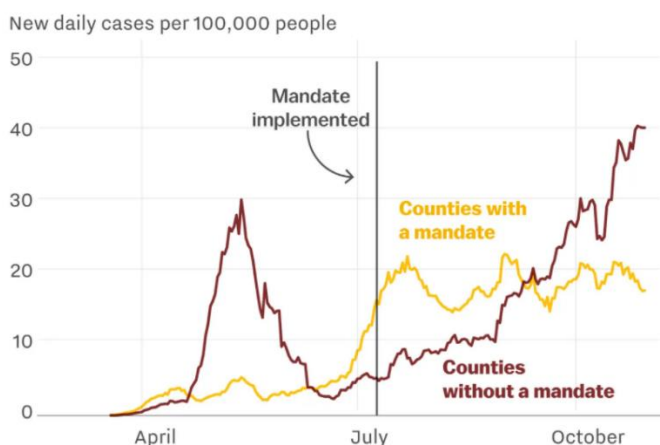
c. Những số liệu khác về việc đeo khẩu trang:

- Khảo sát của tổ chức Ipsos trên 29000 người ở 15 quốc gia trong tháng 4 năm 2020 bao gồm Việt Nam cho thấy:

- + Tỷ lệ đeo khẩu trang nơi công cộng ở Việt Nam đạt mức khá cao, ở mức 91%
- + Số người được hỏi cho biết họ không thấy lợi ích của việc đeo khẩu trang là 10%.
- + Khoảng 80% người vẫn sẽ đeo khẩu trang kể cả khi dịch bệnh kết thúc để bảo vệ những người khác.
- Khảo sát của đại học St.Louis, Mỹ được thực hiện trên 265 sinh viên chỉ ra rằng:
 - + Tỷ lệ lây nhiễm khi cả hai đeo khẩu trang khi tiếp xúc gần chỉ ở mức 7,7%.
 - + Tỷ lệ lây nhiễm khi một trong hai người không đeo khẩu trang lên tới 33%.
 - + Và với thêm mỗi một người không đeo khẩu trang thì tỷ lệ sẽ tăng thêm 40%.
- Theo khảo sát của 2 nhà nghiên cứu của trường đại học Kansas của Mỹ, 24 quận trong tổng số 105 quận của bang Kansas đồng ý thi hành điều luật đeo khẩu trang bắt buộc ở nơi công cộng chứng kiến số ca nhiễm giữ nguyên, trong đó số ca nhiễm ở 81 bang còn lại có số ca nhiễm gần như gấp đôi.

Case counts stabilized in Kansas counties that implemented a mask mandate

Without a statewide mandate, 21 Kansas counties implemented a mandate while the others opted out. Here is what happened next:



C. CÔNG NGHỆ - KỸ THUẬT SỬ DỤNG:

1. Thành phần cấu tạo và ngôn ngữ sử dụng:

- Hệ thống gồm 3 phần chính: nhận diện việc mang khẩu trang, nhận dạng khuôn mặt của người không mang khẩu trang và trang web để theo dõi.
- Ngôn ngữ lập trình chính: Python và HTML
- Trình biên tập mã: Visual Studio Code

2. Thư viện và model AI sử dụng:

- Thư viện: Tensorflow, OpenCV, Numpy
- Model AI: SSD MobileNet V2 FPNLite 320x320. Đây là model có độ chính xác tương đối cao, thời gian train không quá lâu và tốc độ nhận diện nhanh.

3. Cơ sở dữ liệu:

Nhóm chúng em đã sử dụng nhiều nguồn cơ sở dữ liệu để đào tạo cho mô hình AI nhận dạng khẩu trang:

- Bộ dữ liệu *Face Mask Detection* từ trang Kaggle. Bộ dữ liệu bao gồm 853 bức ảnh với 3 label đi kèm (có khẩu trang, không có khẩu trang, đeo khẩu trang sai cách).
- Bộ dữ liệu tự tạo gồm 145 bức ảnh và gồm 3 label.
- Như vậy, hiện tại bộ dữ liệu của nhóm dùng để nhận diện khẩu trang là **998 ảnh** có label tương ứng.

Đối với nhận dạng danh tính, chúng em sử dụng phần mềm để thu thập khoảng 100 ảnh của mỗi đối tượng cần nhận dạng.

4. Một số khái niệm về AI và ML:

a. Trí tuệ nhân tạo:

Trong khoa học máy tính, **trí tuệ nhân tạo** hay **AI** (*artificial intelligence*), đôi khi được gọi là **trí thông minh nhân tạo**, là trí thông minh được thể hiện bằng máy móc, trái ngược với **trí thông minh tự nhiên** của con người. Thông thường, thuật ngữ "trí tuệ nhân tạo" thường được sử dụng để mô tả các máy móc (hoặc máy tính) có khả năng bắt chước các

chức năng "nhận thức" mà con người thường phải liên kết với tâm trí, như "học tập" và "giải quyết vấn đề".

Khi máy móc ngày càng tăng khả năng, các nhiệm vụ được coi là cần "trí thông minh" thường bị loại bỏ khỏi định nghĩa về AI, một hiện tượng được gọi là hiệu ứng AI. Một câu châm ngôn trong Định lý của Tesler nói rằng "AI là bất cứ điều gì chưa được thực hiện." Ví dụ, nhận dạng ký tự quang học thường bị loại trừ khỏi những thứ được coi là AI, đã trở thành một công nghệ thông thường. Khả năng máy hiện đại thường được phân loại như AI bao gồm thành công hiểu lời nói của con người, cạnh tranh ở mức cao nhất trong trò chơi chiến lược (chẳng hạn như cờ vua và Go), xe hoạt động độc lập, định tuyến thông minh trong mạng phân phối nội dung, và mô phỏng quân sự.

Trí tuệ nhân tạo có thể được phân thành ba loại hệ thống khác nhau: trí tuệ nhân tạo phân tích, lấy cảm hứng từ con người và nhân tạo. AI phân tích chỉ có các đặc điểm phù hợp với trí tuệ nhận thức; tạo ra một đại diện nhận thức về thế giới và sử dụng học tập dựa trên kinh nghiệm trong quá khứ để thông báo các quyết định trong tương lai. AI lấy cảm hứng từ con người có các yếu tố từ trí tuệ nhận thức và cảm xúc; hiểu cảm xúc của con người, ngoài các yếu tố nhận thức và xem xét chúng trong việc ra quyết định. AI nhân cách hóa cho thấy các đặc điểm của tất cả các loại năng lực (nghĩa là trí tuệ nhận thức, cảm xúc và xã hội), có khả năng tự ý thức và tự nhận thức được trong các tương tác.

Trí tuệ nhân tạo được thành lập như một môn học thuật vào năm 1956, và trong những năm sau đó đã trải qua nhiều làn sóng lạc quan, sau đó là sự thất vọng và mất kinh phí (được gọi là "mùa đông AI"), tiếp theo là cách tiếp cận mới, thành công và tài trợ mới. Trong phần lớn lịch sử của mình, nghiên cứu AI đã được chia thành các trường con thường không liên lạc được với nhau. Các trường con này dựa trên các cân nhắc kỹ thuật, chẳng hạn như các mục tiêu cụ thể (ví dụ: "robot học" hoặc "học máy"),

việc sử dụng các công cụ cụ thể ("logic" hoặc mạng lưới thần kinh nhân tạo) hoặc sự khác biệt triết học sâu sắc. Các ngành con cũng được dựa trên các yếu tố xã hội (các tổ chức cụ thể hoặc công việc của các nhà nghiên cứu cụ thể).

Lĩnh vực này được thành lập dựa trên tuyên bố rằng trí thông minh của con người "có thể được mô tả chính xác đến mức một cỗ máy có thể được chế tạo để mô phỏng nó". Điều này làm dấy lên những tranh luận triết học về bản chất của tâm trí và đạo đức khi tạo ra những sinh vật nhân tạo có trí thông minh giống con người, đó là những vấn đề đã được thần thoại, viễn tưởng và triết học từ thời cổ đại đề cập tới. Một số người cũng coi AI là mối nguy hiểm cho nhân loại nếu tiến triển của nó không suy giảm. Những người khác tin rằng AI, không giống như các cuộc cách mạng công nghệ trước đây, sẽ tạo ra nguy cơ thất nghiệp hàng loạt.

Trong thế kỷ 21, các kỹ thuật AI đã trải qua sự hồi sinh sau những tiến bộ đồng thời về sức mạnh máy tính, dữ liệu lớn và hiểu biết lý thuyết; và kỹ thuật AI đã trở thành một phần thiết yếu của ngành công nghệ, giúp giải quyết nhiều vấn đề thách thức trong học máy, công nghệ phần mềm và nghiên cứu vận hành.

b. Machine learning:

Học máy (*machine learning*) là một lĩnh vực của trí tuệ nhân tạo liên quan đến việc nghiên cứu và xây dựng các kỹ thuật cho phép các hệ thống "học" tự động từ dữ liệu để giải quyết những vấn đề cụ thể. Ví dụ như các máy có thể "học" cách phân loại thư điện tử xem có phải thư rác (spam) hay không và tự động xếp thư vào thư mục tương ứng. Học máy rất gần với suy diễn thống kê (statistical inference) tuy có khác nhau về thuật ngữ.

Học máy có liên quan lớn đến thống kê, vì cả hai lĩnh vực đều nghiên cứu việc phân tích dữ liệu, nhưng khác với thống kê, học máy tập trung vào sự phức tạp của các giải thuật trong việc thực thi tính toán. Nhiều bài

toán suy luận được xếp vào loại bài toán NP-khó, vì thế một phần của học máy là nghiên cứu sự phát triển các giải thuật suy luận xấp xỉ mà có thể xử lý được.

Học máy có hiện nay được áp dụng rộng rãi bao gồm máy truy tìm dữ liệu, chẩn đoán y khoa, phát hiện thẻ tín dụng giả, phân tích thị trường chứng khoán, phân loại các chuỗi DNA, nhận dạng tiếng nói và chữ viết, dịch tự động, chơi trò chơi và cử động rô-bốt (*robot locomotion*).

5. 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 để cung cấp một API front-end thuận tiện cho việc xây dựng các ứng dụng với framework, 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 cho phép bạn xây dựng biểu đồ và cấu trúc luồng dữ liệu để mô tả cách dữ liệu di chuyển qua biểu đồ hoặc di chuyển qua một seri mà các node đang xử lý. Mỗi một node trong đồ thị đại diện cho một operation toán học, có thể gọi đây là mảng dữ liệu đa chiều hay tensor.

TensorFlow sẽ cung cấp tất cả thông tin cho lập trình viên bằng ngôn ngữ lập trình Python. Python có nhiệm vụ điều phối các luồng công việc và kết nối chúng lại với nhau. Các node và tensor có trong TensorFlow cũng là những đối tượng của Python.

6. Thư viện OpenCV:

OpenCV (được viết tắt từ *Open Computer Vision*) là một bộ công cụ phần mềm để xử lý hình ảnh, video, phân tích và học máy theo thời gian thực. Bộ công cụ này có hơn 2500 thuật toán được sử dụng cổ điển và hiện đại được tối ưu hóa cho thị giác máy tính và học máy.

OpenCV hỗ trợ nhiều ngôn ngữ lập trình như C, C++, Java, Python,...tối ưu hóa thiết kế đa nền tảng và có khả năng hoạt động trên nhiều phần cứng khác nhau. Được thiết kế để tận dụng tất cả các khả năng có sẵn trong phần cứng, OpenCV giúp đảm bảo mang lại hiệu suất tốt nhất cho các ứng dụng máy tính sử dụng bộ công cụ này.

Ứng dụng của thư viện OpenCV:

- Các hoạt động xử lý hình ảnh
- Xây dựng GUI
- Phân tích video
- Tái tạo 3D
- Phát hiện đối tượng
- Học máy
- Phân tích hình dạng
- Nhận dạng khuôn mặt và đối tượng
- Phát hiện và nhận dạng văn bản
- Nhiếp ảnh điện toán

8. Thư viện Numpy:

Numpy là một thư viện lõi phục vụ cho khoa học máy tính của Python, hỗ trợ cho việc tính toán các mảng nhiều chiều, có kích thước lớn với các hàm đã được tối ưu áp dụng lên các mảng nhiều chiều đó. Numpy đặc biệt hữu ích khi thực hiện các hàm liên quan tới Đại Số Tuyến Tính.

9. Trình biên tập mã Visual Studio Code:

a. Khái niệm:

Visual Studio Code là một trình biên tập mã được phát triển bởi Microsoft dành cho Windows, Linux và macOS. Nó hỗ trợ chức năng debug, đi kèm với Git, có syntax highlighting, tự hoàn thành mã thông minh, snippets, và cải tiến mã nguồn. Nó cũng cho phép tùy chỉnh, do đó, người dùng có thể thay đổi theme, phím tắt, và cá tùy chọn khác. Nó miễn phí và là phần mềm mã nguồn mở, mặc dù gói tải xuống chính thì là có giấy phép.

Visual Studio Code được dựa trên Electron, một nền tảng được sử dụng để triển khai các ứng dụng Node.js máy tính cá nhân chạy trên động cơ bố trí Blink. Mặc dù nó sử dụng nền tảng Electron nhưng phần mềm này không phải là một bản khác của Atom, nó thực ra được dựa trên trình biên tập của Visual Studio Online (tên mã là "Monaco").

b. Tính năng:

Visual Studio Code là một trình biên tập mã. Nó hỗ trợ nhiều ngôn ngữ và chức năng tùy vào ngôn ngữ sử dụng theo như trong bảng sau. Nhiều chức năng của Visual Studio Code không hiển thị ra trong các menu tùy chọn hay giao diện người dùng. Thay vào đó, chúng được gọi thông qua khung nhập lệnh hoặc qua một tập tin .json (ví dụ như tập tin tùy chỉnh của người dùng). Khung nhập lệnh là một giao diện theo dòng lệnh. Tuy nhiên, nó biến mất khi người dùng nhấp bất cứ nơi nào khác, hoặc nhấn tổ hợp phím để tương tác với một cái gì đó ở bên ngoài đó. Tương tự như vậy với những dòng lệnh tốn nhiều thời gian để xử lý. Khi thực hiện những điều trên thì quá trình xử lý dòng lệnh đó sẽ bị hủy.

c. Lý do nên sử dụng Visual Studio Code:

- Hoạt động đa nền tảng:

VS Code được thiết kế để có thể chạy tốt trên macOS, Linux và Windows. Vì thế, bạn có thể thoải mái sử dụng trình soạn thảo này mà không cần lo lắng về vấn đề lựa chọn nền tảng tương thích.

- Đơn giản trong việc chỉnh sửa, xây dựng và sửa lỗi:

Như đã nói ở trên, tốc độ là một trong những ưu thế vượt trội của VS Code. Nó có thể hỗ trợ rất nhiều các loại ngôn ngữ lập trình khác nhau như Java, Python, C++, C#, Golang hay PHP. Điều này cho phép bạn có thể ứng dụng rất nhiều tính năng thú vị như đánh dấu cú pháp, tự động thụt lề, đối sánh dấu ngoặc,... cho công việc của mình.

Hệ thống phím tắt sinh động, trực quan giúp thao tác trở nên nhanh chóng hơn, từ đó năng suất và hiệu quả lập trình cũng được tăng lên. Ngoài ra, tính năng nhắc mã IntelliSense chuyên nghiệp trên VS Code sẽ tự động phát hiện đoạn mã không đầy đủ và thông báo cho lập trình viên. Hoặc nó có thể tự động bổ sung cú pháp còn thiếu khi bạn quên không khai báo biến.

Gỡ lỗi cũng là một tính năng nổi bật của VS Code. Nó giúp bạn xem mã nguồn, kiểm tra các biến, Call Stack và thực thi lệnh trong bảng điều khiển. Bên cạnh đó, trình soạn thảo này còn hỗ trợ Git phục vụ cho việc kiểm soát nguồn mà không đòi hỏi phải rời khỏi trình chỉnh sửa.

- Sự hỗ trợ của cộng đồng Github:

Visual Studio Code là một dự án mã nguồn mở nên bạn hoàn toàn có thể tham gia đóng góp và tìm kiếm sự giúp đỡ từ cộng đồng GitHub.

- Kiến trúc mạnh mẽ và có thể mở rộng:

VS Code được xây dựng dựa trên Electron, kết hợp các công nghệ web như Node.js và JavaScript với tốc độ và sự linh hoạt của các ứng dụng gốc. Ngoài ra, kiến trúc dịch vụ công cụ của VS Code giúp nó tích hợp được với các công nghệ hỗ trợ Visual Studio như Roslyn cho .NET, TypeScript, công cụ gỡ lỗi Visual Studio,...

Nhờ những ưu điểm nổi bật của mình, VS Code trở thành một trong những sự lựa chọn hàng đầu của rất nhiều lập trình viên, giúp đảm bảo tiến độ và đạt hiệu quả công việc cao.

8. Công nghệ và thuật toán sử dụng trong nhận diện khẩu trang:

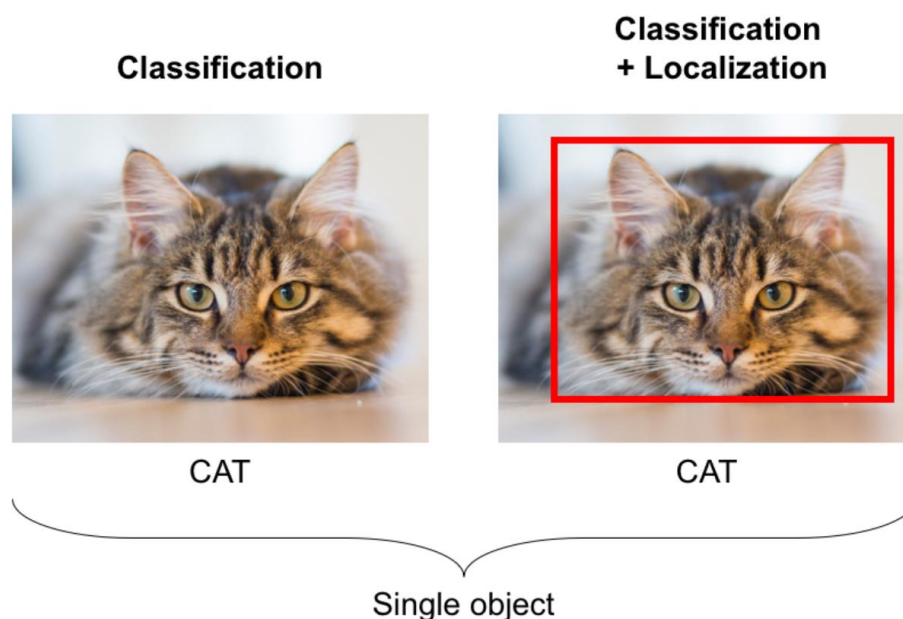
a. Object Detection:

Nhận dạng đối tượng là một thuật ngữ chung để mô tả một tập hợp các nhiệm vụ thị giác máy tính có liên quan liên quan đến việc xác định các đối tượng trong ảnh kỹ thuật số.

Phân loại hình ảnh liên quan đến việc dự đoán lớp của một đối tượng trong một hình ảnh. Định vị vật thể đề cập đến việc xác định vị trí của một hoặc nhiều đối tượng trong một hình ảnh và vẽ bounding box xung quanh chúng. Phát hiện đối tượng kết hợp hai nhiệm vụ trên và thực hiện cho một hoặc nhiều đối tượng trong hình ảnh. Chúng ta có thể phân biệt giữa ba nhiệm vụ thị giác máy tính cơ bản trên thông qua input và output của chúng như sau:

- **Phân loại hình ảnh:** Dự đoán nhãn của một đối tượng trong một hình ảnh.
 - Input: Một hình ảnh với một đối tượng, chẳng hạn như một bức ảnh.
 - Output: Nhãn lớp (ví dụ: một hoặc nhiều số nguyên được ánh xạ tới nhãn lớp).
- **Định vị đối tượng:** Xác định vị trí hiện diện của các đối tượng trong ảnh và cho biết vị trí của chúng bằng bounding box.
 - Input: Một hình ảnh có một hoặc nhiều đối tượng, chẳng hạn như một bức ảnh.
 - Output: Một hoặc nhiều bounding box được xác định bởi tọa độ tâm, chiều rộng và chiều cao.
- **Phát hiện đối tượng:** Xác định vị trí hiện diện của các đối tượng trong bounding box và nhãn của các đối tượng nằm trong một hình ảnh.
 - Input: Một hình ảnh có một hoặc nhiều đối tượng, chẳng hạn như một bức ảnh.

- Output: Một hoặc nhiều bounding box và nhãn cho mỗi bounding box.

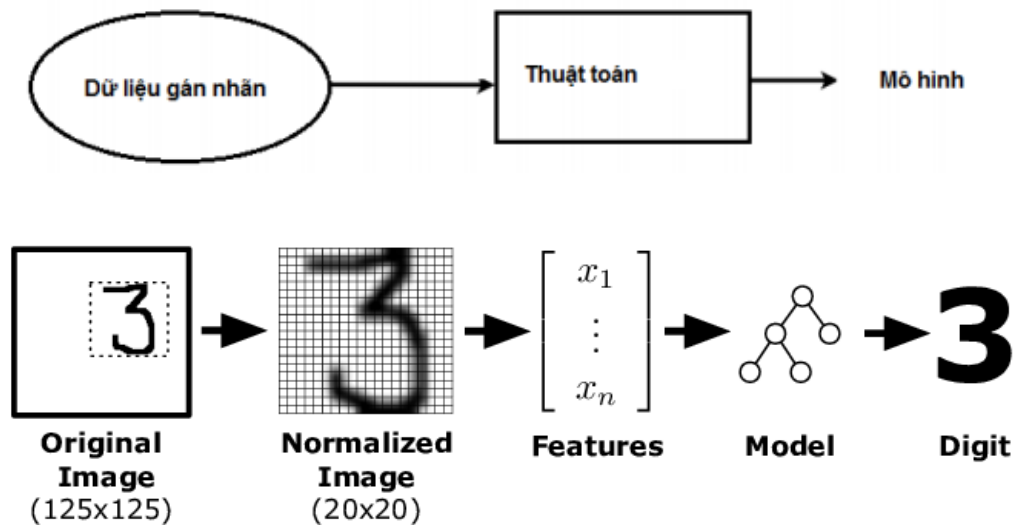


b. Supervised learning:

Supervised learning là thuật toán dự đoán đầu ra (outcome) của một dữ liệu mới (new input) dựa trên các cặp (*input, outcome*) đã biết từ trước. Cặp dữ liệu này còn được gọi là (*data, label*), tức (*dữ liệu, nhãn*). Supervised learning là nhóm phổ biến nhất trong các thuật toán Machine Learning.

Một cách toán học, Supervised learning là khi có một tập hợp biến đầu vào $X=\{x_1, x_2, \dots, x_N\}$ và một tập hợp nhãn tương ứng $Y=\{y_1, y_2, \dots, y_N\}$, trong đó x_i, y_i là các vector. Các cặp dữ liệu biết trước $(x_i, y_i) \in X \times Y$ được gọi là tập *training data* (dữ liệu huấn luyện). Từ tập training data này, cần tạo ra một hàm số ánh xạ mỗi phần tử từ tập X sang một phần tử (xấp xỉ) tương ứng của tập Y : $y_i \approx f(x_i), \forall i=1, 2, \dots, N$

Mục đích là xấp xỉ hàm số f thật tốt để khi có một dữ liệu x mới, có thể tính được nhãn tương ứng của nó $y=f(x)$.

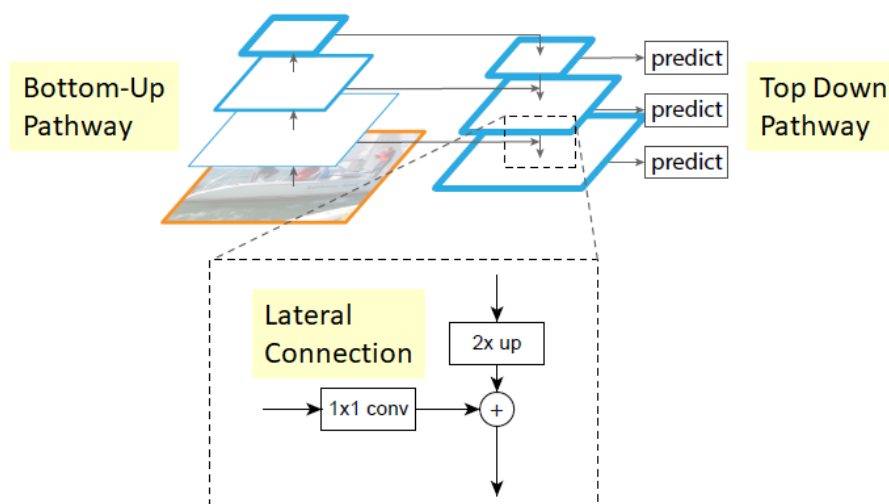


c. Feature Pyramid Network:

Dò tìm các đối tượng có kích thước nhỏ là một vấn đề đáng được giải quyết để nâng cao độ chính xác. Và FPN là mô hình mạng được thiết kế ra dựa trên khái niệm pyramid để giải quyết vấn đề này.

Mô hình FPN kết hợp thông tin của mô hình theo hướng *bottom-up* kết hợp với *top-down* để dò tìm đối tượng (trong khi đó, các thuật toán khác chỉ thường sử dụng *bottom-up*). Khi chúng ta ở bottom và đi lên (up), độ phân giải sẽ giảm, nhưng giá trị ngữ nghĩa (semantic value) sẽ tăng lên.

Trong quá trình xây dựng lại các layer từ top xuống bottom, sẽ gặp một vấn đề khá nghiêm trọng là bị mất mát thông tin của các đối tượng. Ví dụ một đối tượng nhỏ khi lên top sẽ không thấy nó, và từ top đi ngược lại sẽ không thể tái tạo lại đối tượng nhỏ đó. Để giải quyết vấn đề này, tạo các kết nối (skip connection) giữa các reconstruction layer và các feature map để giúp quá trình detector dự đoán các vị trí của đối tượng thực hiện tốt hơn (hạn chế tốt nhất việc mất mát thông tin).



d. MobileNet:

Sự phát triển về mặt học thuật của thị giác máy tính đã tạo ra rất nhiều các models khác nhau với độ chính xác được đo lường trên bộ dữ liệu ImageNet ngày càng được cải thiện. Tuy nhiên không phải toàn bộ trong số chúng đều có thể sử dụng được trên các thiết bị gặp hạn chế về tài nguyên tính toán. Để phát triển được những ứng dụng AI trên các thiết bị như mobile, IoT thì cần hiểu về tài nguyên của những thiết bị này để lựa chọn model phù hợp cho chúng. Những mô hình ưa chuộng được sử dụng thường là những model có số lượng tính toán ít và độ chính xác cao. MobileNet là một trong những lớp mô hình như vậy.

MobileNetV2 cũng sử dụng những kết nối tắt như ở mạng ResNet. Các khối ở layer trước được cộng trực tiếp vào layer liền sau. Nếu coi layer liền trước là x , sau khi đi qua các xử lý tích chập hai chiều ta thu được kết quả $f(x)$ thì output cuối cùng là một residual block có giá trị $x+f(x)$.

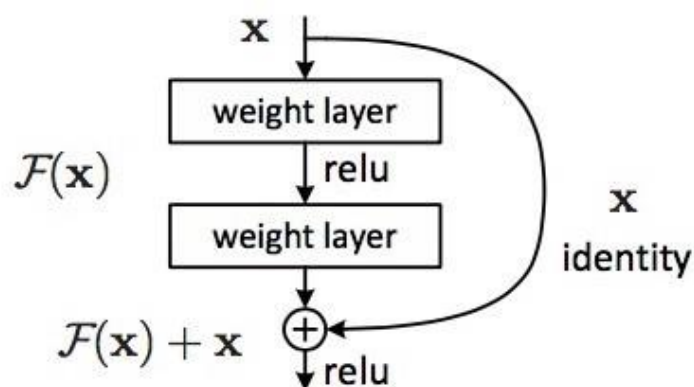
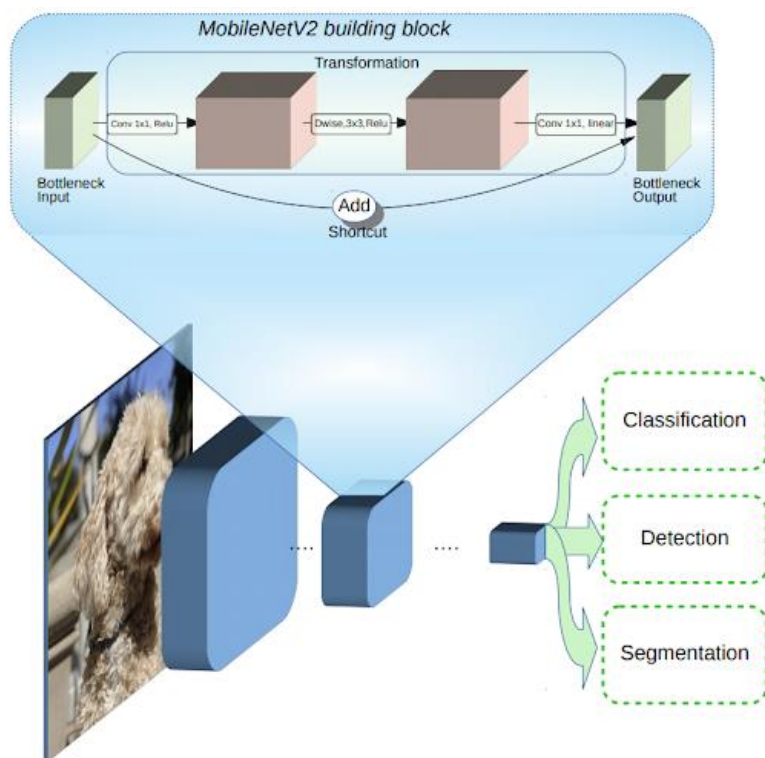


Figure 2. Residual learning: a building block.

Tuy nhiên kết nối tắt ở MobileNetV2 được điều chỉnh sao cho số kênh (hoặc chiều sâu) ở input và output của mỗi block residual được thắt hẹp lại. Chính vì thế nó được gọi là các bottleneck layers.



9. Thuật toán và công nghệ sử dụng trong nhận dạng khuôn mặt:

a. Nhận dạng khuôn mặt:

Nhận dạng khuôn mặt là việc phân loại khuôn mặt của ai trong ảnh đầu vào. Điều này khác với tính năng **nhận diện khuôn mặt**, là việc xác định xem có khuôn mặt nào trong hình ảnh đầu vào hay không. Với nhận dạng khuôn mặt, chúng ta cần một cơ sở dữ liệu hiện có về các khuôn mặt. Với một hình ảnh mới của một khuôn mặt, cần báo cáo tên của người đó.

Một cách đơn giản để thực hiện điều này là lấy hình ảnh mới, làm phẳng nó thành một vector và tính toán khoảng cách Euclide giữa nó và tất cả các hình ảnh phẳng khác trong dataset.

Có một số nhược điểm của cách tiếp cận này. Trước hết, nếu có một cơ sở dữ liệu lớn về các khuôn mặt, thì việc so sánh từng khuôn mặt này sẽ mất một khoảng thời gian. Tập dữ liệu càng lớn, thuật toán càng chậm. Nhưng nhiều khuôn mặt hơn sẽ tạo ra kết quả tốt hơn. Chúng em muốn một hệ thống vừa nhanh vừa chính xác. Đối với điều này thì sẽ sử dụng mạng nơ-ron. Có thể train mạng của mình trên dataset và sử dụng nó cho nhiệm vụ nhận dạng khuôn mặt.

Có một vấn đề khi sử dụng trực tiếp mạng nơ-ron: hình ảnh có thể lớn! Nếu có một $m \times n$ hình ảnh thì sẽ phải làm phẳng nó thành một hình ảnh duy nhất $min \times 1$ vector để đưa vào mạng nơ-ron. Đối với kích thước hình ảnh lớn, điều này có thể ảnh hưởng đến tốc độ. Điều này liên quan đến vấn đề thứ hai với việc sử dụng hình ảnh theo cách tiếp cận ngây thơ: có nhiều chiều dữ liệu. Một đầu vào mới có thể có rất nhiều nhiễu và việc so sánh từng pixel bằng phép trừ ma trận và khoảng cách Euclide có thể cho nhiều rủi ro và phân loại sai.

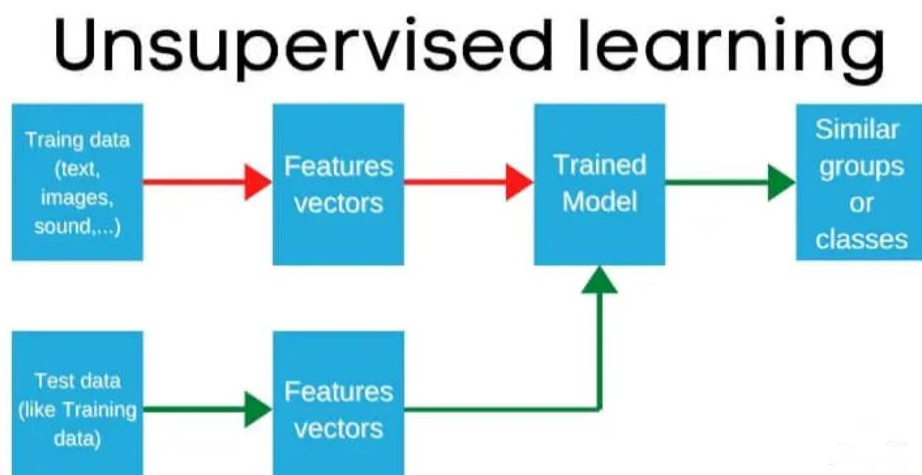
Những vấn đề này là lý do tại sao không nên sử dụng phương pháp “ngây thơ”. Thay vào đó, lấy các hình ảnh có nhiều chiều của mình và giảm chiều của nó trong khi vẫn giữ lại *bản chất* hoặc *các phần quan trọng* của hình ảnh.

b. Unsupervised learning:

Học không có giám sát (unsupervised learning) là một phương pháp của ngành học máy nhằm tìm ra một mô hình mà phù hợp với các quan sát. Nó khác biệt với học có giám sát ở chỗ là đầu ra đúng tương ứng cho mỗi đầu vào là không biết trước. Trong học không có giám sát, một tập dữ liệu đầu vào được thu thập. Học không có giám sát thường đối xử

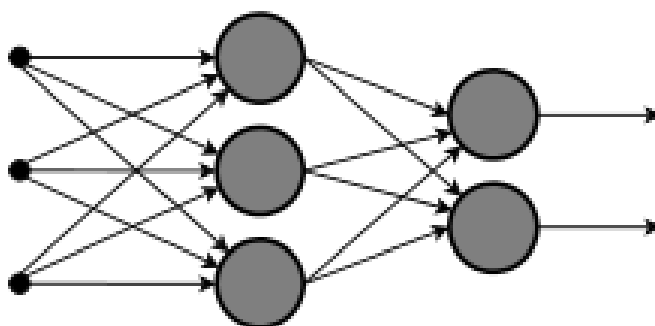
với các đối tượng đầu vào như là một tập các biến ngẫu nhiên. Sau đó, một mô hình mật độ kết hợp sẽ được xây dựng cho tập dữ liệu đó.

Một cách toán học, Unsupervised learning là khi chúng ta chỉ có dữ liệu vào X mà không biết *nhãn* Y tương ứng.



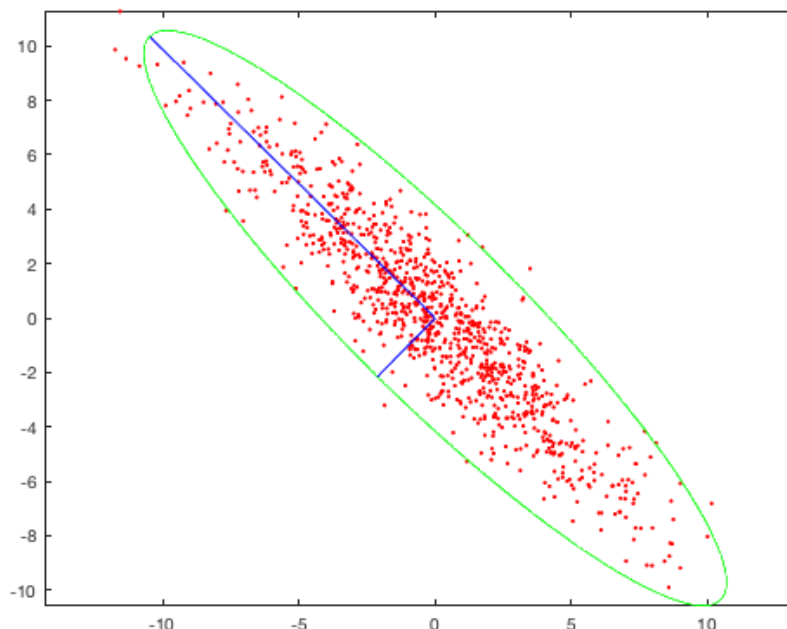
c. PCA (Principal Component Analysis):

Giảm chiều dữ liệu là một loại của unsupervised learning khi đó chúng ta lấy dữ liệu có không gian chiều-cao thành không gian chiều-thấp để biểu diễn ở dạng chiều-thấp đồng thời giữ lại một số thuộc tính có ý nghĩa của dữ liệu gốc, có ý tưởng là gần với chiều nội tại (intrinsic dimension).



Một kỹ thuật giảm kích thước được gọi là **principal component analysis (PCA)**. Ý tưởng đằng sau PCA là chúng tôi muốn chọn siêu phẳng sao cho khi tất cả các điểm được chiếu lên nó, chúng sẽ được trải rộng một cách tối đa. Nói cách khác, muốn *trục của phương sai cực đại*. Trục tiềm năng là trục x hoặc trục y , nhưng, trong cả hai trường hợp, đó

không phải là trục tốt nhất. Tuy nhiên, nếu chọn một đường cắt qua dữ liệu theo đường chéo, đó là trục mà dữ liệu sẽ được lan truyền nhiều nhất!

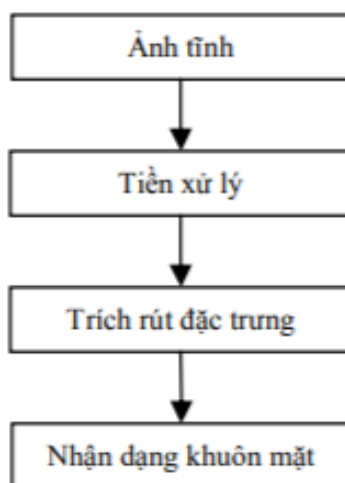


Trục màu xanh lam dài hơn là trục chính xác! Nếu chiếu các điểm của mình lên trục này, chúng sẽ được trải rộng tối đa. Để tìm ra trục này, có thể mượn một thuật ngữ từ đại số tuyến tính gọi là **eigenvectors** ! Đây là nơi **eigenfaces** lấy tên của nó! Về cơ bản, tính toán ma trận hiệp phương sai của dữ liệu và xem xét các tập số **riêng** lớn nhất của ma trận hiệp phương sai đó. Đó là các *trục chính* và các trục mà ta chiếu dữ liệu của mình lên để giảm kích thước. Sử dụng cách tiếp cận này, có thể lấy dữ liệu chiều cao và giảm nó xuống một chiều thấp hơn bằng cách chọn các phương sai lớn nhất của ma trận hiệp phương sai và chiếu lên các phương sai đó.

Vì đang tính toán các trục của mức chênh lệch tối đa, phải giữ lại các khía cạnh quan trọng nhất trong dữ liệu của mình. Trình phân loại dễ dàng tách các khuôn mặt hơn khi dữ liệu được trải ra thay vì nhóm lại với nhau.

Điều này liên quan như thế nào đến thách thức nhận dạng khuôn mặt? Có thể khái niệm hóa $m \times n$ hình ảnh như điểm trong **min**-không gian chiều. Sau đó, sử dụng PCA để giảm dung lượng từ **min** thành một cái gì đó nhỏ hơn nhiều. Điều này sẽ giúp tăng tốc độ tính toán và tránh hiện tượng nhiễu.

d. Nhận diện khuôn mặt bằng Eigenface:



Hệ thống nhận dạng khuôn mặt

- Tiền xử lý

Quá trình tiền xử lý đối với khuôn mặt nhằm nâng cao chất lượng ảnh, chuẩn hóa dữ liệu, kích thước ảnh. Các ảnh trong nghiên cứu này là có chất lượng tương đối tốt nên ta không cần dùng các thuật toán nâng cao chất lượng ảnh mà ta chỉ cần chuẩn hóa ảnh (Normalize Image). Việc chuẩn hóa này khiến độ lệch giữa 2 điểm ảnh được giảm xuống làm quá trình rút đặc trưng thêm chính xác.

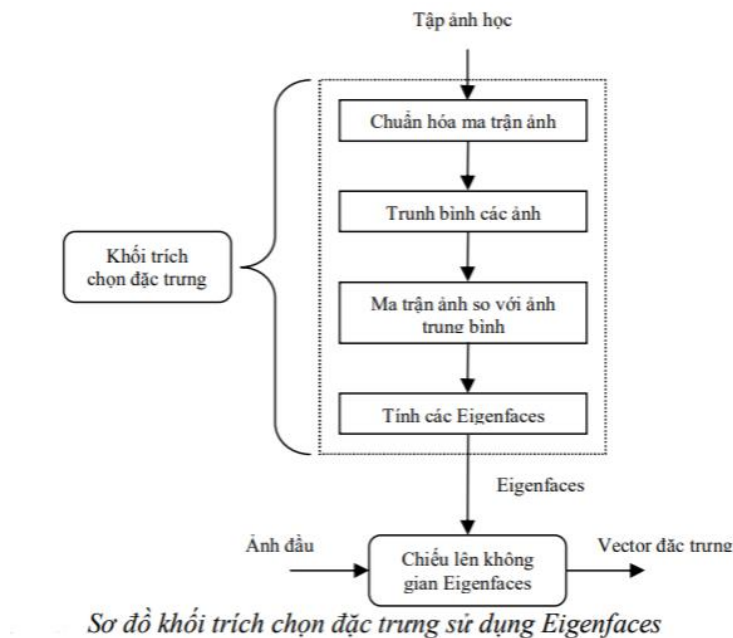
- Trích rút đặc trưng

Trích rút đặc trưng là kỹ thuật sử dụng các thuật toán để lấy ra những thông tin mang những đặc điểm riêng biệt của một người.

- Các khâu trong quá trình trích chọn đặc trưng:

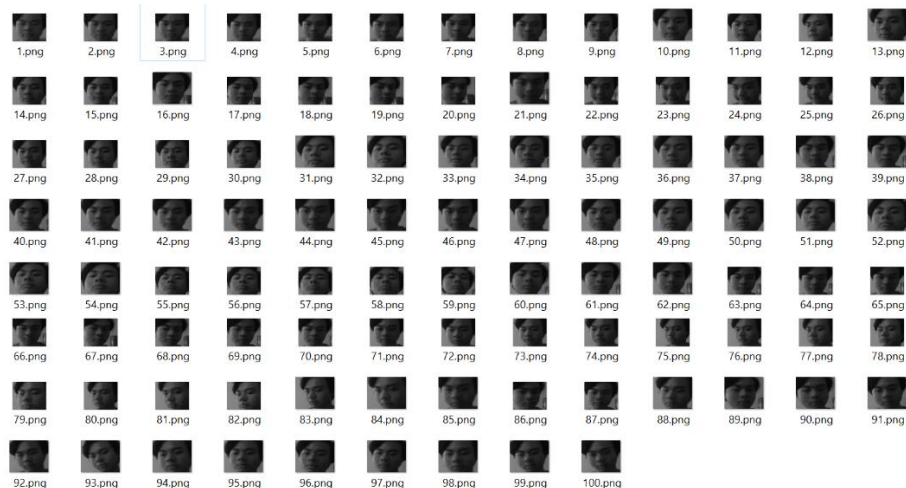
- Đầu vào: Ảnh đã được chuẩn hóa.

- Đầu ra: vector đặc trưng của ảnh đầu vào. Quá trình trích rút đặc trưng được thể hiện qua lưu đồ sau:



- Tính toán eigenface:

- Mô hình được xây dựng dựa trên bộ dữ liệu gồm 1 người và khoảng 100 bức ảnh:



- Chuẩn hóa ma trận ảnh:

Chuẩn hóa vector là khâu đầu tiên cần thực hiện khi sử dụng thuật toán PCA. Sự phân bố của một vector trong không gian được thể hiện qua trung bình (mean) và chuẩn (standard) của vector đó.

Cho vector , trung bình và chuẩn vector $X = [x_1, x_2, x_3, \dots]$ được tính như sau:

$$\text{mean}(X) = \bar{X} = \frac{1}{N} \sum_{i=1}^N x_i$$

$$\text{std}(X) = \sqrt{\frac{\sum_{i=1}^N (x_i - \bar{X})^2}{N-1}}$$

Theo từ một ảnh Γ ban đầu, sẽ có trung bình $\bar{\Gamma}$ và chuẩn $\text{std}(\Gamma)$ của Γ

Phép chuẩn hóa ma trận ảnh được thực hiện theo công thức sau:

$$\Gamma'(x,y) = \frac{\Gamma(x,y) - \bar{\Gamma}}{\text{std}(\Gamma)}$$

Kết quả quá trình chuẩn hóa được thể hiện như sau:



Ảnh đầu vào



Ảnh sau khi chuẩn hóa

- Tính trung bình tập ảnh đầu vào
 - Bước này cho ta xác định được gốc tọa độ của không gian vector biểu diễn các ảnh. Ảnh trung bình được tính bằng trung bình cộng

của tất cả các ảnh. Ảnh trung bình này thể hiện những điểm giống nhau nhất giữa tất cả các khuôn mặt.

- Gọi ảnh số học là \mathbf{M} , mỗi ảnh có kích thước $\mathbf{R} \times \mathbf{C}$

$$S = \{\Gamma_1, \Gamma_2, \dots, \Gamma_M\}$$

- Trong trung bình \mathbf{M} ảnh trên được tính theo công thức:

$$\Psi = \frac{1}{M} \sum_{i=1}^M \Gamma_i$$

- Eigenface

- Theo lý thuyết, eigenface chính là các vector riêng ứng với các giá trị riêng có giá trị lớn nhất của ma trận C:

$$C = \frac{1}{M} \sum_{i=1}^M \Phi_i \Phi_i^T = A.A^T$$

- Vì ma trận C có kích thước quá lớn ($N \times N$) nên để tìm eigenvector u_i của ma trận ta tìm eigenvector và eigenvalue của ma trận L :

$$L = A^T A \quad \text{với} \quad L_{m,n} = \Phi_m^T \Phi_n$$

- Ma trận L có kích thước $M \times M \ll N \times N$ nên việc tính eigenvector sẽ nhanh chóng hơn.
- Gọi v_i là eigenvector của L thì eigenvector của C là:

$$u_i = \sum_{k=1}^M v_{ik} \Phi_k, i = \overline{1, M}$$

- Sau khi tìm được các eigenface, các ảnh trong tập cơ sở dữ liệu sẽ được chiếu lên không gian các eigenface này để tạo ra vector đặc tính. Vector này có kích thước nhỏ hơn nhiều so với ảnh nhưng vẫn mang nhiều nhất thông tin chứa ảnh.

10. Nhận diện khẩu trang:

a. Tạo file TFRecord từ dataset để dễ dàng cho việc training:

```
!python {SCRIPTS_PATH + '/generate_tfrecord.py'} -x {IMAGE_PATH + '/train'} -l
{ANNOTATION_PATH + '/label_map.pbtxt'} -o {ANNOTATION_PATH + '/train.record'}
!python {SCRIPTS_PATH + '/generate_tfrecord.py'} -x {IMAGE_PATH + '/test'} -l
{ANNOTATION_PATH + '/label_map.pbtxt'} -o {ANNOTATION_PATH + '/test.record'}
```

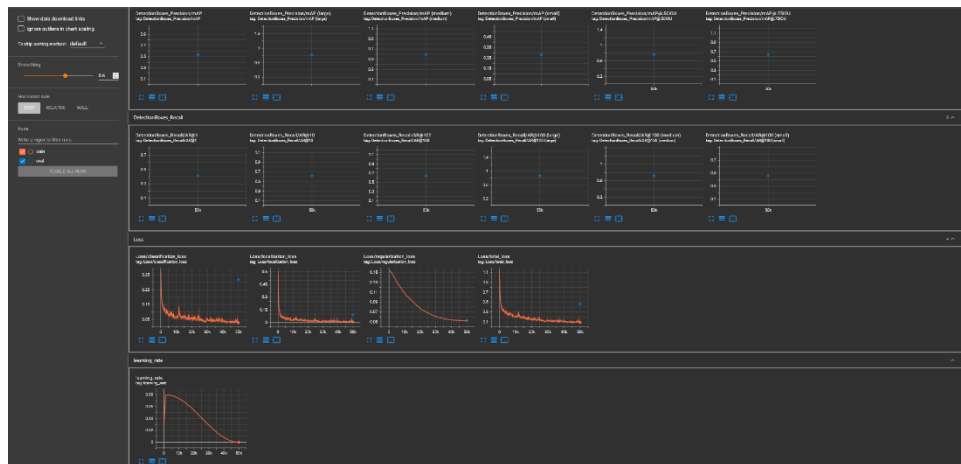
b. Chạy training:

```
(c) Microsoft Corporation. All rights reserved.

C:\Users\PC>python Tensorflow/models/research/object_detection/model_main_tf2.py
--model_dir=Tensorflow/workspace/models/my_ssd_mobnet --pipeline_config_path=Te
nsorflow/workspace/models/my_ssd_mobnet/pipeline.config --num_train_steps=50000
```

c. Tensorboard:

Tensorflow có cung cấp tensorboard để người dùng tiện theo dõi quá trình train và nhiều thông số khác để dễ dàng đánh giá model.



d. Chương trình chính:

```
while True:
    ret, frame = cap.read()
    #frame = increase_brightness(frame, value=10)
    image_np = np.array(frame)

    input_tensor = tf.convert_to_tensor(np.expand_dims(image_np, 0), dtype=tf.float32)
    detections = detect_fn(input_tensor)

    num_detections = int(detections.pop('num_detections'))
    detections = {key: value[0, :num_detections].numpy()
                  for key, value in detections.items()}
    detections['num_detections'] = num_detections

    # detection_classes should be ints.
    detections['detection_classes'] = detections['detection_classes'].astype(np.int64)

    label_id_offset = 1
    image_np_with_detections = image_np.copy()
```

11. Nhận dạng khuôn mặt:

a. Chuẩn bị ảnh để train:

Dùng camera máy tính để thu thập ảnh (100 bức).

```
def capture_training_images(self):
    video_capture = cv2.VideoCapture(0)
    while True:
        self.count_timer += 1
        ret, frame = video_capture.read()
        inImg = np.array(frame)
        outImg = self.process_image(inImg)
        cv2.imshow('Video', outImg)

        # When everything is done, release the capture on pressing 'q'
        if cv2.waitKey(1) & 0xFF == ord('q'):
            video_capture.release()
            cv2.destroyAllWindows()
            return
```

b. Xử lý hình ảnh trước khi train:

Chuyển màu ảnh từ RGB sang màu Gray.

```
def process_image(self, inImg):
    frame = cv2.flip(inImg, 1)
    resized_width, resized_height = (112, 92)
    if self.count_captures < NUM_TRAINING:
        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
        gray_resized = cv2.resize(gray, (int(gray.shape[1]/RESIZE_FACTOR), int(gray.shape[0]/RESIZE_FACTOR)))
        faces = self.face_cascade.detectMultiScale(
            gray_resized,
            scaleFactor=1.1,
            minNeighbors=5,
            minSize=(30, 30),
            flags=cv2.CASCADE_SCALE_IMAGE
        )
```

c. Chạy training:

```
def eigen_train_data(self):
    imgs = []
    tags = []
    index = 0

    for (subdirs, dirs, files) in os.walk(self.face_dir):
        for subdir in dirs:
            img_path = os.path.join(self.face_dir, subdir)
            for fn in os.listdir(img_path):
                path = img_path + '/' + fn
                tag = index
                imgs.append(cv2.imread(path, 0))
                tags.append(int(tag))
            index += 1
    (imgs, tags) = [np.array(item) for item in [imgs, tags]]

    self.model.train(imgs, tags)
    self.model.save('eigen_trained_data.xml')
    print ("Training completed successfully")
    return
```


d. Chạy nhận dạng:

```
def process_image(self, inImg):
    frame = cv2.flip(inImg, 1)
    resized_width, resized_height = (112, 92)
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    gray_resized = cv2.resize(gray, (int(gray.shape[1]/RESIZE_FACTOR), int(gray.shape[0]/RESIZE_FACTOR)))
    faces = self.face_cascade.detectMultiScale(
        gray_resized,
        scaleFactor=1.1,
        minNeighbors=5,
        minSize=(30, 30),
        flags=cv2.CASCADE_SCALE_IMAGE
    )
    persons = []
    for i in range(len(faces)):
        face_i = faces[i]
        x = face_i[0] * RESIZE_FACTOR
        y = face_i[1] * RESIZE_FACTOR
        w = face_i[2] * RESIZE_FACTOR
        h = face_i[3] * RESIZE_FACTOR
        face = gray[y:y+h, x:x+w]
        face_resized = cv2.resize(face, (resized_width, resized_height))
        confidence = self.model.predict(face_resized)
        if confidence[1] < 3500:
            person = self.names[confidence[0]]
            cv2.rectangle(frame, (x,y), (x+w, y+h), (255, 0, 0), 3)
            cv2.putText(frame, '%s - %.0f' % (person, confidence[1]), (x-10, y-10), cv2.FONT_HERSHEY_PLAIN, 1, (0, 255, 0))
        else:
            person = 'Unknown'
            cv2.rectangle(frame, (x,y), (x+w, y+h), (0, 0, 255), 3)
            cv2.putText(frame, person, (x-10, y-10), cv2.FONT_HERSHEY_PLAIN, 1, (0, 255, 0))
        persons.append(person)
    return (frame, persons)
```

12. Thiết kế ứng dụng web:

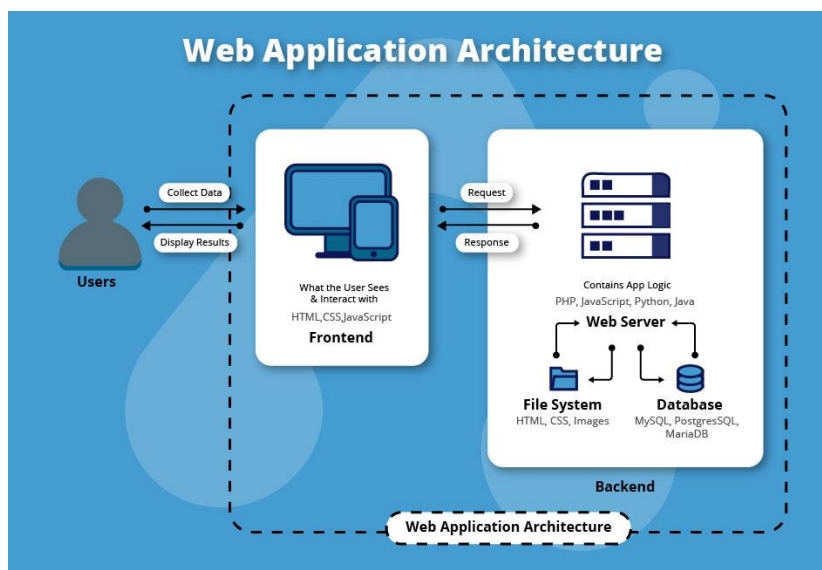
a. Sử dụng cơ sở dữ liệu phpmyadmin:

phpMyAdmin là một công cụ nguồn mở miễn phí được viết bằng ngôn ngữ lập trình PHP để xử lý các tác vụ quản trị của MySQL thông qua một trình duyệt web. Nó có thể thực hiện nhiều tác vụ như tạo, sửa đổi hoặc xóa bỏ cơ sở dữ liệu, bảng, các trường hoặc bản ghi; thực hiện theo báo cáo SQL; hoặc quản lý người dùng và cấp phép.

MySQL là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở (gọi tắt là RDBMS) hoạt động theo mô hình client-server. Với RDBMS là viết tắt của **Relational Database Management System**. MySQL được tích hợp apache, PHP. MySQL quản lý dữ liệu thông qua các cơ sở dữ liệu. Mỗi cơ sở dữ liệu có thể có nhiều bảng quan hệ chứa dữ liệu. MySQL cũng có cùng một cách truy xuất và mã lệnh tương tự với ngôn ngữ SQL.

b. Kết nối cơ sở dữ liệu bằng python:

Python cung cấp một thư viện giúp kết nối và truy cập vào cơ sở dữ liệu đó là thư viện Mysqldb. Mysqldb là một interface dùng để kết nối từ Python tới một máy chủ cơ sở dữ liệu MySQL. Nó triển khai Python Database API v2.0 và được xây dựng dựa trên MySQL C API (tức là các API được viết bằng ngôn ngữ C dành cho MySQL)



Dữ liệu từ chương trình của bọn em là mã định danh của học sinh sẽ được đưa lên cơ sở dữ liệu và từ đây những dữ liệu này sẽ được đối chiếu, xử lý và hiển thị trên web những học sinh không đeo khẩu trang.

c. Thiết kế web bằng HTML5 và CSS:

Ngôn ngữ HTML5 được sử dụng bởi đây là ngôn ngữ thiết kế cốt lõi của web, được cải thiện cho đa phương tiện mới nhất trong khi vẫn giữ nó dễ dàng đọc được bởi con người và luôn hiểu được bởi các thiết bị và các chương trình máy tính như trình duyệt web, phân tích cú pháp,....

d. Xử lý dữ liệu bằng PHP:

Ngôn ngữ PHP được dùng để kết nối với máy chủ sau đó lấy dữ liệu từ cơ sở dữ liệu và tiến hành xử lý cuối cùng là hiển thị lên web.

e. Cách hoạt động của hệ thống:

Mã định danh của học sinh không đeo khẩu trang được lấy từ phần nhận dạng khuôn mặt sau đó được xử lý và gửi lên cơ sở dữ liệu

```
import MySQLdb

db = MySQLdb.connect("localhost","root","","khkt_database")
insertrec = db.cursor()
send_id = '1.0.20a502'
# sqlquery="insert into class_list(id, uname, class_id) values (%s, 'Tran Dinh Phuoc', '11A5')"
sqlquery="insert into 11a5(id, date_time, only_date) values (%s, curtime(), curdate())"
insertrec.execute(sqlquery, [send_id])
db.commit()
db.close()
```




Gửi dữ liệu bằng Mysqldb python

Dữ liệu trên database sẽ được đối chiếu và xử lý sau đó hiển thị lên web

```

<?php
function get_id($val){
    $dom = 0;
    $str = "";
    $array = str_split($val);
    for ($i = 0; $i < strlen($val); $i++){
        if ($dom < 2){
            if ($val[$i] == "-"){
                $dom += 1;
            }
        }
        else {
            $str .= $val[$i];
        }
    }
    return $str;
}

$conn = mysqli_connect("localhost","root","","khkt_database");
if ($conn->connect_error){
    die("Connection failed:".$conn->connect_error);
}

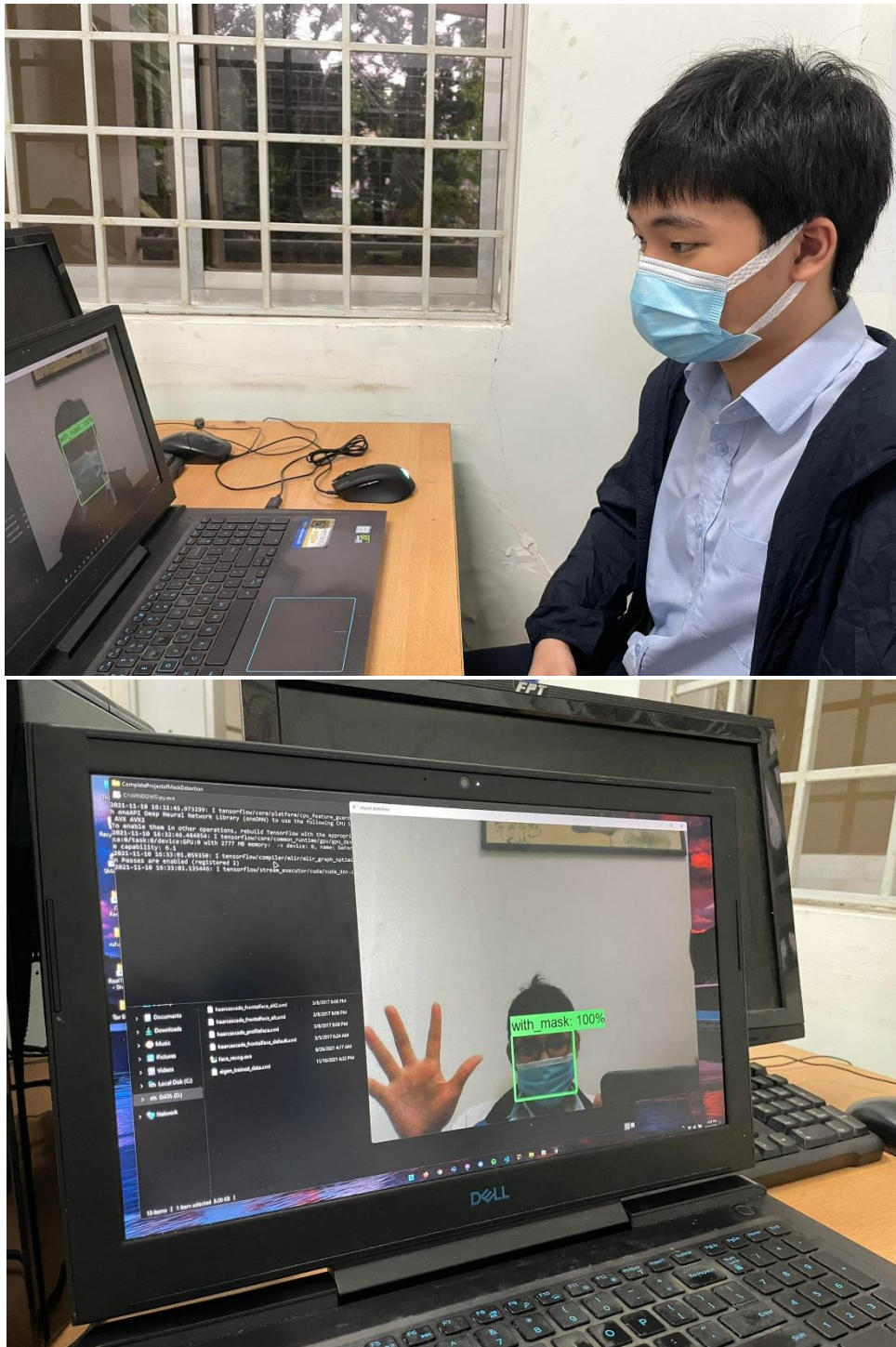
if (isset($_POST["search"])){
    $select_text = $_POST["formDate"];
    $select_name = $_POST["name_form"];
    if ($select_name != ""){
        $sql = "SELECT id, date_time, only_date from lla5";
        $result = $conn->query($sql);
        if ($result->num_rows > 0){
            while($row = $result->fetch_assoc()){
                $id = $row["id"];
                $row["id"] = get_id($row["id"]);
                $id_dtb = $row["id"];
                if ($select_name == $id_dtb || $select_text == $row["only_date"]){
                    $sql2 = "SELECT uname from class_list where id = '$id_dtb' ";
                    $result2 = $conn->query($sql2);
                    $ddate = $row["only_date"];
                    while($row2 = $result2->fetch_assoc()){
                        echo "<tr><td>". $row["id"]. "</td><td>". $row2["uname"]. "</td><td>". $row["date_time"]. "</td><td>". <img src='./AfterCrop/'. $ddate. '/'. $id. '.jpg' width=
                    }
                }
                // $id_anh = $row["ma_anh"]
            }
            // echo "<tr><td>". $row["id"]. "</td><td>". $row["uname"]. "</td><td>". $row["date_time"]. "</td><td>".   |
| 20a503 | Dinh Thanh Nhat  | 2021-11-12 16:00:16 |  |
| 20a504 | Tran Dinh Phuoc  | 2021-11-12 16:00:38 |  |

Dữ liệu được hiển thị trên web

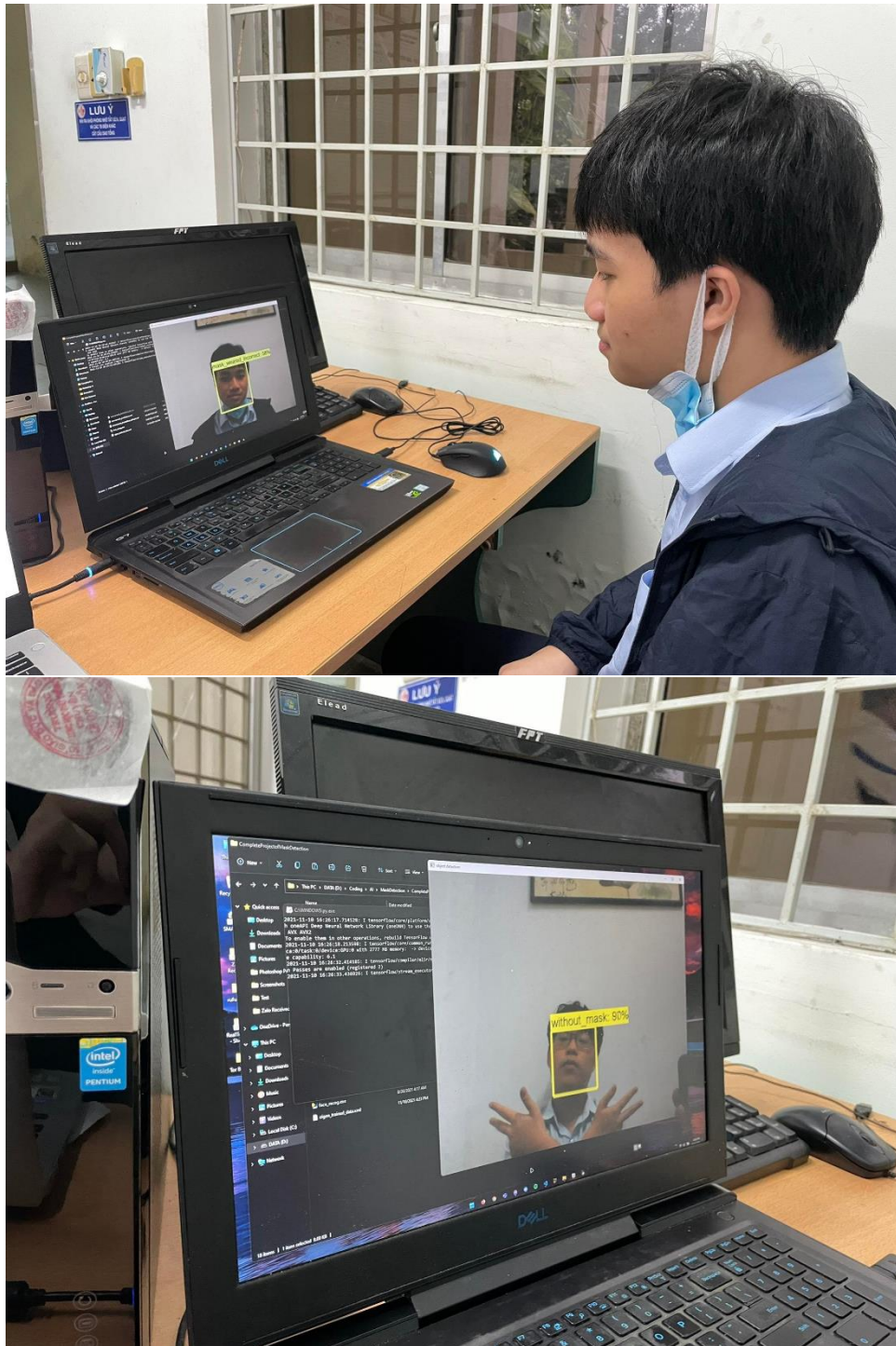
## E. KẾT QUẢ THỰC HIỆN ĐỀ TÀI:

### 1. Chức năng hiện có:

a. Một số hình ảnh nghiên cứu và thử nghiệm:







*b. Kết quả đạt được:*

- Một trang web demo đơn giản trên máy tính
- Nhận diện được người ngồi trước màn hình máy tính với 3 nhãn bao gồm (khẩu trang, không đeo khẩu trang và đeo khẩu trang sai cách)

- Nhận diện được danh tính của người không mang khẩu trang. Sau đó hình ảnh sẽ được gửi lên database và được hiển thị trên website.
- Phát âm thanh cảnh báo khi có người không mang khẩu trang.

## **2. Ưu điểm:**

- Dễ hiểu và dễ sử dụng, thân thiện với người dùng.
- Mức độ xác định chính xác việc mang khẩu trang và nhận danh tính tương đối cao.
- Nhận diện được nhiều kiểu loại khẩu trang các nhau.
- Bộ dữ liệu có đa dạng chủng tộc nên độ chính xác tương đối giống nhau với bất kì đối tượng nào.
- Tốc độ nhận diện nhanh với cả nhận diện khẩu trang và danh tính.

## **3. Nhược điểm:**

- Tuy độ chính xác của AI nhận diện khẩu trang tương đối cao nhưng bị ảnh hưởng bởi điều kiện ánh sáng (khuất sáng).
- Thuật toán EigenFace có sẵn của thư viện OpenCV chưa thật sự có độ chính xác quá tốt.
- Trang web chỉ chạy dưới dạng localhost.
- Dataset vẫn chưa đủ lớn để có độ chính xác cao hơn.

## **F. HƯỚNG PHÁT TRIỂN:**

- Tìm kiếm thêm bộ dữ liệu để cải tiến độ chính xác, đặc biệt là trong nhiều điều kiện ánh sáng khuất.
- Thay đổi thuật toán nhận dạng danh tính, sử dụng Local Binary Patterns Histograms để thay thế.
- Đưa phần mềm lên Raspberry Pi để có thể thực nghiệm thực tế.
- Mở rộng đối tượng và địa điểm sử dụng như văn phòng công ty, khu công nghiệp,...

## G. TÀI LIỆU THAM KHẢO:

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

<https://www.phamduytung.com/blog/2018-12-06-what-do-we-learn-from-single-shot-object-detection>

<https://phamdinhhkhanh.github.io/2019/09/29/OverviewObjectDetection.html>

<https://phamdinhhkhanh.github.io/2020/09/19/MobileNet.html>

<https://vi.wikipedia.org/wiki/SARS-CoV-2>

[https://vi.wikipedia.org/Học máy](https://vi.wikipedia.org/Học_máy)

<https://wiki.tino.org/tensorflow-la-gi/>

<https://thuvienphapluat.vn/tintuc/vn/thoi-su-phap-luat/thoi-su/27262/cap-nhat-moi-nhat-tinh-hinh-virus-corona-covid-19>

<https://covid19.who.int/>

<https://freetuts.net/hoc-php>

<https://vi.wikipedia.org/wiki/PHP>

<https://cafedev.vn/tu-hoc-python-ket-noi-mysqldb-trong-python/>

[http://tailieuso.udn.vn/bitstream/TTHL\\_125/4836/3/Tomtat.pdf](http://tailieuso.udn.vn/bitstream/TTHL_125/4836/3/Tomtat.pdf)

<https://thorphanh.github.io/blog/2018/04/18/eigenface/>

[https://vi.wikipedia.org/wiki/Visual Studio Code](https://vi.wikipedia.org/wiki/Visual_Studio_Code)

<https://vn.got-it.ai/blog/visual-studio-code-la-gi-loi-the-tu-viec-su-dung-visual-studio-code>

<https://viblo.asia/p/gioi-thieu-ve-numpy-mot-thu-vien-chu-yeu-phuc-vu-cho-khoa-hoc-may-tinh-cua-python-maGK7kz9Kj2>

<https://wiki.tino.org/opencv-la-gi/>

<https://www.cdc.gov/mmwr/volumes/70/wr/mm7036a3.htm>

<https://www.vox.com/science-and-health/21546014/mask-mandates-coronavirus-covid-19>

<https://www.ipsos.com/en-lb/more-people-say-theyre-wearing-masks-protect-themselves-covid-19-march>