

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA ĐIỆN TỬ VIỄN THÔNG

ĐỒ ÁN TỐT NGHIỆP

NGÀNH: ĐIỆN TỬ VIỄN THÔNG
CHUYÊN NGÀNH: KỸ THUẬT MÁY TÍNH

ĐỀ TÀI:
NHẬN DẠNG KHUÔN MẶT ỨNG DỤNG TRONG
HỆ THỐNG CHẤM CÔNG

Người hướng dẫn: **TS. TRẦN THỊ MINH HẠNH**

Sinh viên thực hiện: **ĐẶNG CÔNG MINH**

Số thẻ sinh viên: **106180032**

Lớp: **18DT1**

Đà Nẵng, 12/2022

TÓM TẮT

Tên đề tài: Nhận dạng khuôn mặt ứng dụng trong hệ thống chấm công

Sinh viên thực hiện: Đặng Công Minh

Số thẻ SV: 106180032 Lớp: 18DT1

Nhận dạng khuôn mặt người là một lĩnh vực quan trọng và có nhiều ứng dụng thực tế, như trong các hệ thống bảo mật, xác minh nhân dạng, phương thức giao tiếp người máy mới, hay trong lĩnh vực giải trí,... Trong thời đại dịch covid 19 vẫn còn đang tiềm tàng thì cần nhiều biện pháp để ngăn chặn và phòng ngừa dịch bệnh. Từ hai nhận thức trên đề tài nhận dạng khuôn mặt ứng dụng trong hệ thống chấm công có thể phát hiện và nhận dạng ngay cả khi khuôn mặt đeo khẩu trang được hình thành. Các kết quả thử nghiệm trong đồ án hoàn toàn trung thực và có thể đem vào ứng dụng thực tế đặc biệt là trong công việc chấm công

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên:Đặng Công Minh..... Số thẻ sinh viên: ...106180032.....

Lớp:...18DT1 ... Khoa:....Điện tử viễn thông..... Ngành: ...Kỹ thuật máy tính.....

1. Tên đề tài đồ án:

...Nhận dạng khuôn mặt ứng dụng trong hệ thống chấm công

.....

2. Đề tài thuộc dạng: ☐ Có ký kết thỏa thuận sở hữu trí tuệ đối với kết quả thực hiện

3. Các số liệu và dữ liệu ban đầu:

.....

.....

.....

4. Nội dung các phần thuyết minh và tính toán:

.....

.....

.....

.....

.....

5. Các bản vẽ, đồ thị (ghi rõ các loại và kích thước bản vẽ):

.....

.....

.....

.....

6. Họ tên người hướng dẫn: Trần Thị Minh Hạnh

7. Ngày giao nhiệm vụ đồ án: ...05.../...09.../...2022...

8. Ngày hoàn thành đồ án: ..15.../..12.../...2022..

Đà Nẵng, ngày 16 tháng 12 năm 2022

Trưởng Bộ môn Kỹ thuật máy tính

Người hướng dẫn

LỜI NÓI ĐẦU

Xin chân thành cảm ơn các thầy cô trong khoa Điện Tử Viễn Thông đại học Bách khoa Đà Nẵng đã xây dựng những kiến thức nền tảng vững chắc để hỗ trợ hoàn thành đồ án. Cảm ơn cô Trần Thị Minh Hạnh đã trực tiếp hỗ trợ và giúp đỡ trong quá trình xây dựng và hoàn thiện đồ án tốt nghiệp.

CAM ĐOAN

Tôi xin cam đoan rằng đồ án tốt nghiệp với đề tài... là nghiên cứu độc lập của tôi. Đồng thời những số liệu được cung cấp từ báo cáo đều là của cá nhân và đây là kết quả nghiên cứu hoàn toàn trung thực, không sao chép từ bất kì một công trình nghiên cứu khác nào. Những tài liệu trích dẫn đều đã được ghi rõ nguồn gốc.

Tôi xin chịu hoàn toàn trách nhiệm trước nhà trường nếu trường hợp phát hiện ra bất cứ sai phạm hay vấn đề sao chép nào trong đề tài này.

Sinh viên thực hiện

Đặng Công Minh

MỤC LỤC

Tóm tắt	
Nhiệm vụ đồ án	
Lời nói đầu	iv
Cam đoan	v
Mục lục	vi
Danh sách các bảng biểu, hình vẽ và sơ đồ	x
Danh sách các cụm từ viết tắt	xiii
MỞ ĐẦU	1
CHƯƠNG 1 GIỚI THIỆU ĐỀ TÀI NHẬN DẠNG KHUÔN MẶT ỨNG DỤNG TRONG HỆ THỐNG CHẤM CÔNG.....	2
1.1 Giới thiệu chương	2
1.2 Đặt vấn đề	2
1.3 Mục tiêu đề tài	3
1.4 Thách thức và ứng dụng.....	4
1.5 Mô hình đề xuất nghiên cứu	4
1.6 Kết luận chương.....	5
CHƯƠNG 2 TỔNG QUAN LÝ THUYẾT	6
2.1 Giới thiệu chương	6
2.2 Học sâu trong thị giác máy tính	6
2.2.1 Học sâu (Deep Learning)	6
2.2.2 Mạng nơron tích chập (CNN)	7
2.2.3 Neural network.....	13
2.2.4 Hàm kích hoạt ReLU	14
2.2.5 Kỹ thuật Dropout.....	15
2.3 Các kiến trúc mạng CNN.....	16
2.3.1 Kiến trúc mạng ResNet	16

2.3.2	Kiến trúc mạng Inception-V1.....	17
2.3.3	Kiến trúc mạng DarkNet	18
2.4	Thuật toán YOLO	19
2.4.1	YOLO (You Only Look One).....	19
2.4.2	Anchor box.....	21
2.4.1	IOU (INTERSECTION OVER UNION)	22
2.4.2	Non-max suppression.....	22
2.5	Mô hình FaceNet.....	23
2.5.1	FaceNet.....	23
2.5.2	Hàm mất mát Triplet	24
2.5.3	Lựa chọn ảnh đầu vào	26
2.5.4	Learning Similarity	26
2.6	Các thang đo đánh giá chỉ số	27
2.6.1	Độ chính xác	27
2.6.2	Recall và Precision	28
2.7	Kết luận chương.....	29
CHƯƠNG 3 MÔ HÌNH NHẬN DẠNG KHUÔN MẶT		30
3.1	Giới thiệu chương	30
3.2	Hướng thực hiện và nghiên cứu.....	30
3.3	Sơ đồ khối của mô hình tổng thể	31
3.4	Sơ đồ khối mô hình phát hiện khuôn mặt	32
3.5	Sơ đồ khối mô hình nhận dạng danh tính	33
3.5.1	Phân loại áp dụng Learning similarity	34
3.5.2	Phân loại áp dụng Neural network	34
3.6	Tổng quan các tập dữ liệu sử dụng	35
3.6.1	Tập dữ liệu phát hiện khuôn mặt.....	35
3.6.2	Tập dữ liệu nhận dạng danh tính	36
3.7	Phạm vi giải quyết vấn đề.....	37
3.8	Tổng kết chương	38

CHƯƠNG 4	KẾT QUẢ ĐÁNH GIÁ MÔ HÌNH	39
4.1	Giới thiệu chương	39
4.2	Mô hình phát hiện và phân loại khuôn mặt	39
4.2.1	Quá trình huấn luyện mô hình lần 1	39
4.2.1.1	Quy trình gán nhãn dữ liệu	39
4.2.1.2	Đánh giá kết quả lần 1	41
4.2.2	Quá trình huấn luyện mô hình lần 2	43
4.2.2.1	Gán nhãn dữ liệu bán tự động	43
4.2.2.2	Thay đổi các tham số	43
4.2.2.3	Đánh giá kết quả lần 2	44
4.2.3	Tổng kết kết quả	46
4.3	Mô hình nhận dạng khuôn mặt	47
4.3.1	Quy trình gán nhãn dữ liệu	47
4.3.1.1	Tập dữ liệu thực tế	48
4.3.1.2	Tập dữ liệu người nổi tiếng	48
4.3.1.3	Tập dữ liệu người đeo khẩu trang	48
4.3.2	Huấn luyện mô hình	49
4.3.2.1	Phân loại áp dụng learning similarity	50
4.3.2.2	Phân loại áp dụng neural network	50
4.3.3	Kết quả thu được	51
4.3.3.1	Kết quả trên tập dữ liệu thực tế	51
4.3.3.2	Kết quả trên tập dữ liệu người nổi tiếng	53
4.3.4	Nhận xét kết quả	55
4.3.4.1	So sánh giữa 2 tập dữ liệu thực tế và người nổi tiếng	55
4.3.4.2	So sánh giữa 2 tập dữ liệu đeo khẩu trang và không đeo khẩu trang	55
4.3.4.3	So sánh giữa 2 phương pháp huấn luyện	55
4.4	Mô hình hoàn thiện	56
4.5	Kết luận chương	58
KẾT LUẬN		56

TÀI LIỆU THAM KHẢO.....
PHỤ LỤC.....

DANH SÁCH CÁC BẢNG, HÌNH VẼ

Bảng 1 Bảng so sánh các thang đo giữa hai lần huấn luyện	46
Bảng 2 Đánh giá kết quả trên tập dữ liệu thực tế phương pháp learning similarity ...	51
Bảng 3 Đánh giá kết quả trên tập dữ liệu thực tế phương pháp neural network	51
Bảng 4 Đánh giá kết quả trên dữ liệu người nổi tiếng phương pháp learning similarity	53
Bảng 5 Bảng đánh giá kết quả trên dữ liệu người nổi tiếng phương pháp neural network	53
Hình 1.1 Minh họa khả năng thuận tiện của nhận dạng khuôn mặt	2
Hình 1.2 Minh họa ứng dụng của nhận diện khuôn mặt trong thức tế	3
Hình 1.3 Minh họa khi chấm công mà khuôn mặt có đeo khẩu trang	4
Hình 1.4 Tổng quan mô hình lựa chọn và cách hoạt động	5
Hình 2.1 Minh họa 1 mô hình học sâu dùng để phân loại	7
Hình 2.2 Một trình tự CNN để phân loại các chữ số viết tay	8
Hình 2.3 Làm phẳng (flattening) một ma trận 3x3 thành một véc tơ 9x1	8
Hình 2.4 Minh họa một hình RGB có kích thước 4x4x3	9
Hình 2.5 Tích chập một hình ảnh 5x5x1 với một bộ lọc 3x3x1 để có được một hình ảnh chập 3x3x1	10
Hình 2.6 Minh họa cách di chuyển của bộ lọc	10
Hình 2.7 Một hình ảnh 5x5x1 được đệm thành một hình ảnh 6x6x1	11
Hình 2.8 Ma trận gộp 3x3 từ một ma trận tích chập 5x5	12
Hình 2.9 Các kiểu phép gộp	12
Hình 2.10 Minh họa đồ thị hàm ReLU	14
Hình 2.11 Minh họa việc áp dụng dropout trong neural network	15
Hình 2.12 Kết nối ‘tắt’ trong ResNet	16
Hình 2.13 Kiến trúc GoogleNet - Inception version 1	17

Hình 2.14 Kiến trúc mạng DarkNet53	19
Hình 2.15 Minh hoạ thuật toán YOLO hoạt động.....	20
Hình 2.16 Feature map của YOLO với từng kích thước khác nhau	20
Hình 2.17 Minh hoạ cho cách anchor box hoạt động.....	21
Hình 2.18 Minh hoạ cách tính IOU	22
Hình 2.19 Từ 3 bounding box ban đầu cùng bao quanh chiếc xe đã giảm xuống còn một bounding box cuối cùng	23
Hình 2.20 Minh hoạ các anchor, positive và negative	24
Hình 2.21 Minh hoạ phương pháp learning similarity	27
Hình 2.22 Minh hoạ các chỉ số precision và recall	28
Hình 3.1 Tổng quan mô hình nhận dạng khuôn mặt	31
Hình 3.2 Sơ đồ khối mô hình phát hiện khuôn mặt.....	32
Hình 3.3 Sơ đồ khối phương pháp nghiên cứu huấn luyện mô hình nhận dạng danh tính	33
Hình 3.4 Sơ đồ khối phân loại áp dụng learning similarity	34
Hình 3.5 Sơ đồ khối phân loại áp dụng neural network.....	35
Hình 3.6a và 3.6b Minh hoạ dữ liệu phát hiện khuôn mặt	36
Hình 3.7 Dữ liệu tập thực tế không đeo khẩu trang	36
Hình 3.8 Dữ liệu tập người nổi tiếng không đeo khẩu trang.....	37
Hình 4.1 Minh hoạ công cụ hỗ trợ gán nhãn dữ liệu.....	40
Hình 4.2 Biểu đồ số lượng các đối tượng sau khi được gán nhãn lần 1.....	40
Hình 4.3 Biểu đồ trên là các hàm mất mát và thang đo trong mô hình phát hiện khuôn mặt	42
Hình 4.4 Ma trận nhầm lẫn của huấn luyện lần 1.....	43
Hình 4.5 Biểu đồ số lượng các đối tượng sau khi được gán nhãn lần 2.....	44
Hình 4.6 Kết quả ma trận nhầm lẫn của huấn luyện lần 2	44
Hình 4.7 Kết quả của các thang đo recall, precision, mAP 0.5 và mAP 0.5:0.95.....	45
Hình 4.8 Kết quả của hàm mất mát trong tập huấn luyện và đánh giá	46
Hình 4.9 Hộp giới hạn không đồng nhất ở huấn luyện lần 1	47

Hình 4.10 Hộp giới hạn đồng nhất ở huấn luyện lần 2	47
Hình 4.11 Minh họa việc gán nhãn tự động	48
Hình 4.12 Các loại khẩu trang có sẵn.....	49
Hình 4.13 Minh họa khẩu trang được đeo theo dạng sinh học.....	49
Hình 4.14 Minh họa phân cấp các tập dữ liệu	49
Hình 4.15 Thông số chi tiết model neural network cho tập dữ liệu người nổi tiếng ...	50
Hình 4.16 Thông số huấn luyện tốt nhất của tập dữ liệu không đeo khẩu trang và chưa có tăng cường dữ liệu	52
Hình 4.17 Thông số huấn luyện tốt nhất của tập dữ liệu đeo khẩu trang và chưa có tăng cường dữ liệu	52
Hình 4.18 Thông số huấn luyện tốt nhất của tập dữ liệu không đeo khẩu trang và có tăng cường dữ liệu	54
Hình 4.19 Thông số huấn luyện tốt nhất của tập dữ liệu đeo khẩu trang và có tăng cường dữ liệu	54
Hình 4.20 Một số hình ảnh demo của tập dữ liệu thực tế	56
Hình 4.21 Một số demo của hình ảnh người nổi tiếng	57
Hình 4.22 Một số hình ảnh demo kết quả sau cùng	57

DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT

CHỮ VIẾT TẮT:

CNN : Convolution neural network

NN: Neural network

YOLO: You only look one

SSD: Single Shot Multibox Detector

MTCNN: Multi-task Cascaded Convolutional Networks

IOU: Intersection Over Union

RCNN: Regional convolutional neural network

AI: Artificial Intelligence

VGG: Visual Geometry Group

MỞ ĐẦU

Những năm gần đây, tình hình dịch bệnh đã trở thành vấn đề rất phức tạp và nóng hổi cần ưu tiên hàng đầu đối với con người. Chủng virus corona hay còn gọi là covid-19 đã và đang hoành hành trên toàn thế giới là một trong những vấn đề cấp bách của con người đương thời cần đối mặt, bên cạnh đó còn có các vấn đề về ô nhiễm môi trường, khói bụi không khí, sự nóng lên toàn cầu... Dịch bệnh đã ảnh hưởng tới nước ta trong các phương diện về kinh tế, y tế, giao thông, giáo dục, con người... Kinh tế tăng trưởng chậm, y tế quá tải, giao thông đình trệ, giáo dục bất cập là những tác động rõ ràng của covid-19 lên đất nước và thế giới thế nên để hạn chế dịch bệnh ngoài những chính sách đúng đắn của chính phủ thì cũng cần ý thức của người dân tuân thủ quy định. Một trong những quy định là việc đeo khẩu trang y tế, khẩu trang chuyên dụng tại nơi đông người là những yêu cầu bắt buộc với mọi người dân. Thế nhưng vấn đề này vẫn còn tranh cãi khi một số bộ phận thiếu ý thức không chấp hành tại nơi công cộng.

Từ những mong muốn đó đã xuất hiện ý tưởng về mô hình phát hiện và nhận dạng người đeo khẩu trang dựa trên bài toán thị giác máy tính. Bài toán giúp giải quyết các vấn đề không đeo khẩu trang ở nơi công cộng với mục đích là cảnh báo và nhắc nhở. Các bước xây dựng mô hình dựa trên học sâu. Từ các ý tưởng ban đầu đến khi hoàn thành sản phẩm có tính ứng dụng vào các thiết bị cảnh báo, nhắc nhở, xử phạt các trường hợp không tuân thủ quy định phòng dịch ở nơi đông người, các thiết bị tự động chấm công, mở cửa tự động...

CHƯƠNG 1 GIỚI THIỆU ĐỀ TÀI NHẬN DẠNG KHUÔN MẶT ỨNG DỤNG TRONG HỆ THỐNG CHĂM CÔNG

1.1 Giới thiệu chương

Với mục đích phát triển các dự án giúp đỡ cộng đồng nên vấn đề về một máy tính có thể nhận dạng được khuôn mặt người đeo khẩu trang là hướng nghiên cứu thiết thực và có tính ứng dụng trong thời gian dịch covid còn đang hoành hành. Mở rộng hơn ở phạm vi nghiên cứu mô hình có thể áp dụng trong các hệ thống chăm công khuôn mặt khi có thể nhận dạng được danh tính của nhân viên khi người đó vẫn đang đeo khẩu trang. Qua quá trình tìm hiểu và ứng dụng những kiến thức để hoàn thiện bài toán nhận dạng khuôn mặt người đeo khẩu trang góp phần thúc đẩy khả năng nhận dạng của máy tính đối với người đeo khẩu trang.

1.2 Đặt vấn đề

Hệ thống chăm công khuôn mặt là một bài toán không mới lạ trong lĩnh vực trí tuệ nhân tạo nhưng một hệ thống chăm công có thể nhận biết được người đó khi người đó đeo khẩu trang là một vấn đề mới. Trong thời kì dịch bệnh covid còn hoành hành, việc đeo khẩu trang như một yếu tố đề phòng dịch bệnh lây lan nên một hệ thống chăm công tích hợp việc nhận dạng khi đeo khẩu trang mang tính cấp thiết.

Bài toán đã đặt ra những thách thức trong việc có thể phát hiện được khuôn mặt người đeo khẩu trang, thách thức khi nguồn dữ liệu về người đeo khẩu trang đang khan hiếm hoặc không được chia sẻ rộng rãi. Ngoài việc có thể phát hiện khuôn mặt đeo khẩu trang thì nhận dạng danh tính khi khuôn mặt đeo khẩu trang cũng là những thách thức lớn vì việc đeo khẩu trang đã che hầu hết các đặc điểm trên khuôn mặt.



Hình 1.1 Minh họa khả năng thuận tiện của nhận dạng khuôn mặt

1.3 Mục tiêu đề tài

Mục tiêu đầu tiên của bài toán là phát hiện được khuôn mặt người đeo khẩu trang. Việc áp dụng các mô hình đã được huấn luyện sẵn để phát hiện khuôn mặt không phải là việc quá khó khăn, nhưng một mô hình phát hiện khuôn mặt đeo khẩu trang là vấn đề mới. Các mô hình lâu đời như SSD, MTCNN, Harr Cascade, ... có thể giải quyết được vấn đề nhưng khả năng phát hiện được khuôn mặt người đeo khẩu trang nhưng vẫn chưa triệt để.

Vấn đề tiếp theo là việc phải phân loại được các khuôn mặt đã phát hiện được. Việc phân loại có thể sử dụng những kiến trúc chuyên dùng phân loại cho hình ảnh như vgg, mobilenet, resnet ...

Cuối cùng, việc nhận dạng danh tính khi người đó có đeo khẩu trang hay không đeo khẩu trang. Các mô hình nhận dạng như FaceNet, CosFace, ArcFace có thể giải quyết được yêu cầu của bài toán

Mục tiêu sau cùng của đề tài là có thể phát hiện và nhận dạng khuôn mặt của một người dù người đó có đeo khẩu trang hay không đeo khẩu trang. Từ đó tăng khả năng ứng dụng vào các hệ thống chăm công khuôn mặt.



Hình 1.2 Minh họa ứng dụng của nhận diện khuôn mặt trong thức tế

1.4 Thách thức và ứng dụng

Thách thức của đề tài nằm ở nguồn dữ liệu người đeo khẩu trang không được chia sẻ rộng rãi, cũng như nguồn dữ liệu đã được gán nhãn sẵn. Đặc biệt, với yêu cầu thêm một đối tượng đeo khẩu trang sai cách thì nguồn dữ liệu này càng khan hiếm

Thêm vào đó, việc nhận dạng danh tính được một đối tượng khi đeo khẩu trang là vấn đề khó thực hiện khi việc nhận dạng chỉ phụ thuộc vào các đặc trưng trên vùng mắt của đối tượng

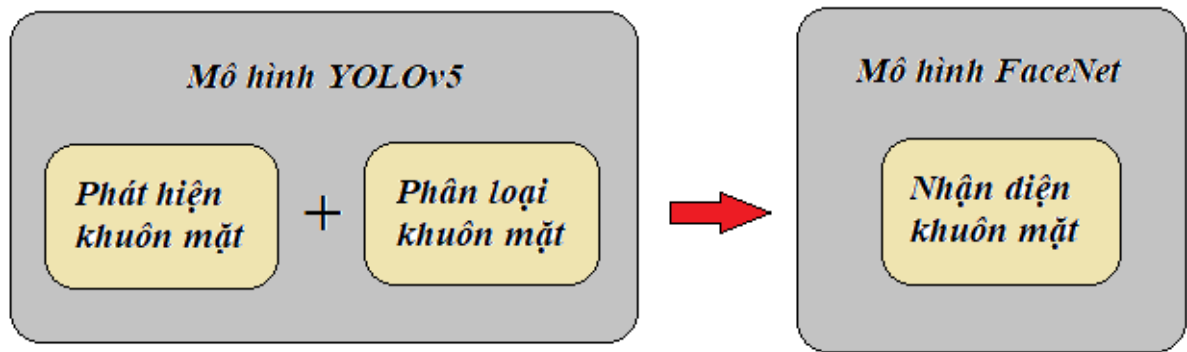


Hình 1.3 Minh họa khi chăm công mà khuôn mặt có đeo khẩu trang

Ứng dụng của hệ thống có thể áp dụng vào các hệ thống chăm công khuôn mặt hoặc các địa điểm tụ tập bắt buộc người đeo khẩu trang như bệnh viện, bưu điện, văn phòng hành chính ... Để áp dụng được vào việc chăm công thì yêu cầu về độ chính xác, thời gian thực thi ở thực tế nhanh và tốt cũng là một trong những thách thức của bài toán

1.5 Mô hình đề xuất nghiên cứu

Do yêu cầu về việc sẽ cần độ chính xác và hoạt động tốt trên thời gian thực tế nên mô hình họ các YOLO từ v1 tới v5 là mô hình được lựa chọn. Vấn đề phát hiện và phân loại sẽ được áp dụng trong một mô hình duy nhất sẽ giảm quá trình kết nối các mô hình. Mô hình nhận dạng danh tính khuôn mặt sẽ áp dụng mô hình FaceNet để nhận dạng danh tính người đeo khẩu trang lẫn không đeo khẩu trang



Hình 1.4 Tổng quan mô hình lựa chọn và cách hoạt động

Việc lựa chọn YOLO thay cho các mô hình phát hiện khuôn mặt khác vì lý do khả năng hoạt động trên thời gian thực tế tốt, độ chính xác cao và khả năng thực thi nhanh. Ngoài ra việc thay thế mô hình phát hiện khuôn mặt MTCNN của FaceNet bằng YOLO vì YOLO phân loại và phát hiện cùng lúc, giảm thiểu quá trình huấn luyện mô hình

Mô hình FaceNet được lựa chọn vì khả năng giảm tải việc huấn luyện khi dữ liệu biến động vì sử dụng các embedding và lý thuyết dễ dàng áp dụng

1.6 Kết luận chương

Chương 1 đã trình bày mục tiêu đề tài cũng như các vấn đề thách thức gặp phải khi thực hiện bài toán nhận dạng khuôn mặt người đeo khẩu trang tích hợp trong hệ thống chăm công. Ở chương 2 sẽ trình bày các phần lý thuyết liên quan được áp dụng trong đồ án

CHƯƠNG 2 TỔNG QUAN LÝ THUYẾT

2.1 Giới thiệu chương

Trong chương này sẽ trình bày tổng quan các lý thuyết liên quan được áp dụng vào bài toán. Lý thuyết sẽ tập trung vào các mô hình phát hiện, phân loại và nhận dạng danh tính, các hàm mất mát, các phần lý thuyết tổng quan về học sâu trong thị giác máy tính.

2.2 Học sâu trong thị giác máy tính

2.2.1 Học sâu (*Deep Learning*)

AI(Artificial Intelligence) bao gồm nhiều lĩnh vực nghiên cứu, từ thuật toán di truyền đến các hệ thống chuyên gia và cung cấp phạm vi cho các lập luận về những gì cấu thành AI.

Trong lĩnh vực nghiên cứu AI, Machine Learning đã đạt được thành công đáng kể trong những năm gần đây – cho phép máy tính vượt qua hoặc tiến gần đến việc kết hợp hiệu suất của con người trong các lĩnh vực từ nhận dạng khuôn mặt đến nhận dạng giọng nói và ngôn ngữ.

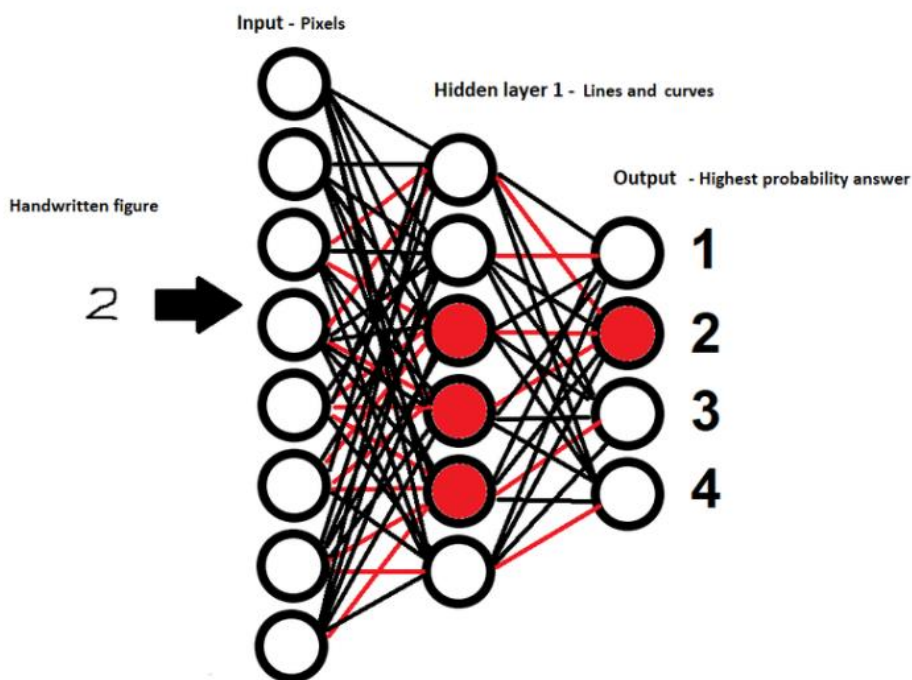
Machine Learning là quá trình dạy máy tính thực hiện một nhiệm vụ, thay vì lập trình nó làm thế nào để thực hiện nhiệm vụ đó từng bước một.

Khi kết thúc đào tạo, một hệ thống Machine Learning sẽ có thể đưa ra dự đoán chính xác khi được cung cấp dữ liệu.

Điều đó nghe có vẻ khô khan, nhưng những dự đoán đó có thể trả lời liệu một miếng trái cây trong ảnh là chuối hay táo, nếu một người đang băng qua trước một chiếc xe tự lái, cho dù việc sử dụng sách từ trong câu liên quan đến bìu mềm hoặc đặt phòng khách sạn, cho dù email là thư rác hay nhận dạng giọng nói đủ chính xác để tạo chú thích cho video YouTube.

Machine Learning thường được chia thành học có giám sát, trong đó máy tính học bằng ví dụ từ dữ liệu được gắn nhãn và học không giám sát, trong đó các máy tính nhóm các dữ liệu tương tự và xác định chính xác sự bất thường.

Deep Learning là một tập hợp con của Machine Learning, có khả năng khác biệt ở một số khía cạnh quan trọng so với Machine Learning nông truyền thống, cho phép máy tính giải quyết một loạt các vấn đề phức tạp không thể giải quyết được.



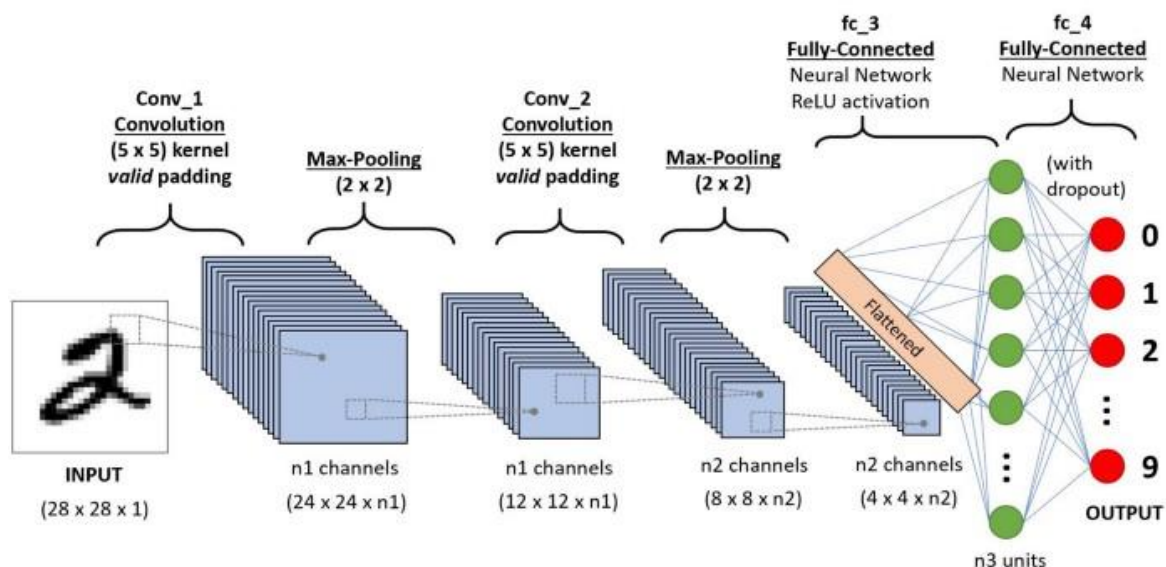
Hình 2.1 Minh họa 1 mô hình học sâu dùng để phân loại

2.2.2 Mạng nơ-ron tích chập (CNN)

Tích chập đóng một vai trò quan trọng và xuất hiện từ sớm trong lịch sử xử lý tín hiệu số. Việc tìm ra các bộ lọc phù hợp cho mỗi loại tín hiệu và mỗi bài toán đã được nghiên cứu và giảng dạy rất nhiều trong các giáo trình kỹ thuật.

Mục tiêu của lĩnh vực này là cho phép máy móc nhìn thế giới giống như con người, nhận thức nó theo cách tương tự con người và thậm chí sử dụng kiến thức đó cho vô số nhiệm vụ như nhận dạng hình ảnh & video, phân tích & phân loại hình ảnh, giải trí truyền thông, hệ thống khuyến nghị (recommendation system), xử lý ngôn ngữ tự nhiên (natural language processing), v.v. Những tiến bộ trong Thị giác Máy tính với Học sâu đã được xây dựng và hoàn thiện theo thời gian, chủ yếu qua một thuật toán cụ thể - Mạng nơ-ron tích chập (Convolutional Neural Network) [1].

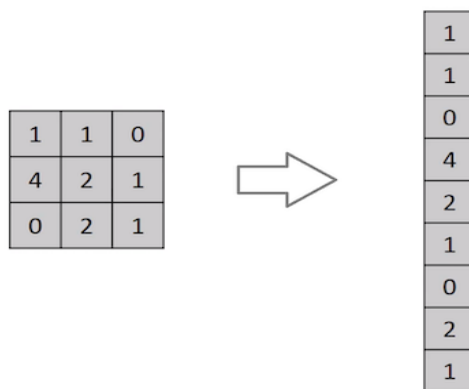
Cuối những năm 1980s, Yann Lecun đề xuất một mô hình tích chập hai chiều cho dữ liệu ảnh và thu lại thành công lớn trong bài toán phân loại chữ số viết tay. Bằng việc sử dụng rất nhiều dữ liệu và thay các tầng nối kín (neural network layer) [2] trong mạng perceptron đa tầng bởi tích chập hai chiều, các bộ lọc phù hợp với bài toán và dữ liệu có thể được học để mang lại kết quả phân lớp tốt nhất.



Hình 2.2 Một trình tự CNN để phân loại các chữ số viết tay

Mạng nơron tích chập (còn gọi là ConvNet / CNN) là một thuật toán Deep Learning có thể lấy hình ảnh đầu vào, gán độ quan trọng (các trọng số - weights và độ lệch - bias có thể học được) cho các đặc trưng/đối tượng khác nhau trong hình ảnh và có thể phân biệt được từng đặc trưng/đối tượng này với nhau. Công việc tiền xử lý được yêu cầu cho mạng nơron tích chập thì ít hơn nhiều so với các thuật toán phân loại khác. Trong các phương thức sơ khai, các bộ lọc được thiết kế bằng tay (hand - engineered), với một quá trình huấn luyện để chọn ra các bộ lọc/đặc trưng phù hợp thì mạng nơron tích chập lại có khả năng tự học để chọn ra các bộ lọc/ đặc trưng tối ưu nhất.

Kiến trúc của nơron tích chập tương tự như mô hình kết nối của các nơron trong bộ não con người và được lấy cảm hứng từ hệ thống vỏ thị giác trong bộ não (visual cortex). Các nơ-ron riêng lẻ chỉ phản ứng với các kích thích trong một khu vực hạn chế của trường thị giác được gọi là Trường tiếp nhận (Receptive Field). Một tập hợp các trường như vậy chồng lên nhau để bao phủ toàn bộ khu vực thị giác.

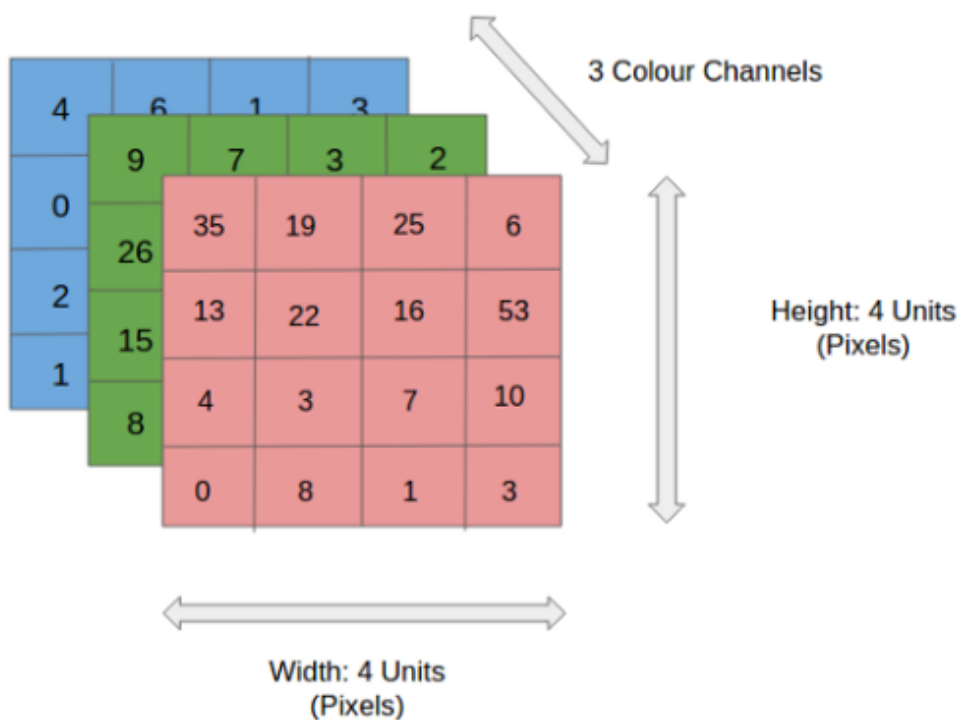


Hình 2.3 Làm phẳng (flattening) một ma trận 3x3 thành một véc tơ 9x1

Một hình ảnh không gì khác là một ma trận các giá trị pixel. Vậy tại sao chúng ta không làm phẳng hình ảnh (ví dụ: ma trận hình ảnh 3×3 thành một vector 19×1) và đưa nó vào một mạng nơ-ron đa lớp cho mục đích phân loại? Không thực sự đơn giản như vậy...

Trong trường hợp của hình ảnh nhị phân cơ bản (binary images), phương pháp nơ-ron đa lớp có thể cho ra độ chính xác trung bình khi phân loại nhưng sẽ không còn chính xác khi phân tích các hình ảnh phức tạp - nơi mà các pixel có một mức độ phụ thuộc lẫn nhau.

Phương pháp nơ-ron tích chập có năng lực để hiểu được các mức độ phụ thuộc không gian và tạm thời giữa các pixel trong dữ liệu ảnh thông qua các bộ lọc ứng dụng đặc trưng. Kiến trúc này có hiệu suất tốt hơn cho tập dữ liệu dạng hình ảnh do làm giảm được số lượng tham số liên quan và khả năng tái sử dụng (reusability) các trọng số. Nói cách khác, mô hình mạng nơ-ron tích chập có thể được huấn luyện để hiểu được sự tinh tế của hình ảnh tốt hơn các mô hình khác.

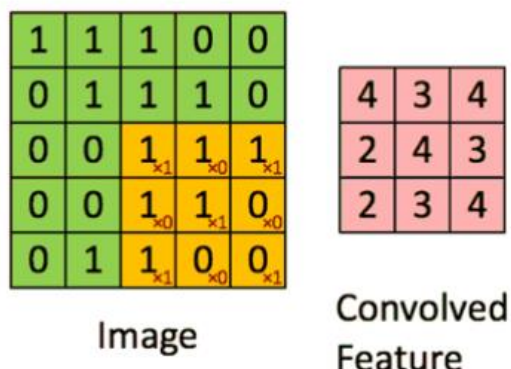


Hình 2.4 Minh họa một hình RGB có kích thước $4 \times 4 \times 3$

Ở hình bên, chúng ta có một hình ảnh RGB được phân tách theo 3 mặt phẳng (kênh) màu - Đỏ, Xanh lục và Xanh dương. Có một số không gian màu khác cho hình ảnh như Grayscale, RGB, HSV, CMYK, v.v.

Bạn có thể thấy chi phí tính toán sẽ lớn như thế nào khi hình ảnh đạt đến kích thước, giả sử 8K ($7680 \text{ pixel} \times 4320 \text{ pixel}$). Vai trò của nơ-ron tích chập là giảm chiều hình ảnh thành một dạng dễ xử lý hơn, mà không đánh mất đi các đặc trưng quan trọng của hình

ảnh để có được một dự đoán tốt (good prediction). Điều này rất quan trọng khi chúng ta thiết kế một kiến trúc không chỉ tốt về việc học tập các đặc trưng mà còn có thể tương thích với các bộ dữ liệu có kích thước lớn.



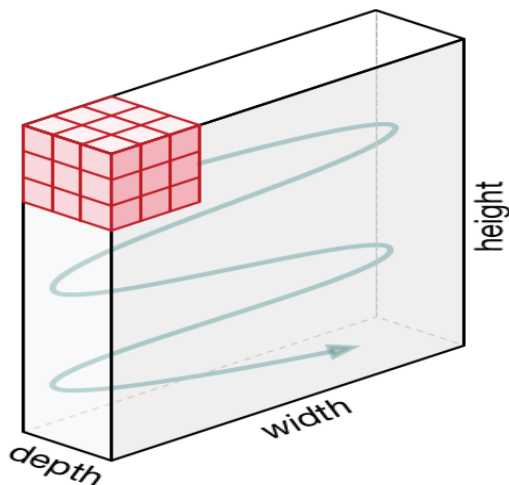
Hình 2.5 Tích chập một hình ảnh $5 \times 5 \times 1$ với một bộ lọc $3 \times 3 \times 1$ để có được một hình ảnh chập $3 \times 3 \times 1$

Kích thước hình ảnh = 5 (Chiều cao) x 5 (Chiều rộng) x 1 (Số lượng kênh, ví dụ: RGB).

Ở hình bên, phần màu xanh lục là hình ảnh đầu vào $15 \times 5 \times 1$ của chúng ta, ta gọi hình ảnh đầu vào này là I

Phần tử liên quan đến việc thực hiện thao tác tích chập trong phần đầu tiên của lớp tích chập được gọi là Bộ lọc (Kernel / Filter) , K, được thể hiện bằng màu vàng. Chúng ta chọn K là một ma trận $13 \times 3 \times 1$.

Bộ lọc di chuyển 9 lần vì độ dài dải trượt (Stride Length) = 1 (tức không bị trượt), với mỗi lần di chuyển sẽ thực hiện một phép nhân ma trận giữa bộ lọc K và tỉ lệ P của bức ảnh tương ứng với vị trí mà bộ lọc lúc đó đang đi qua.

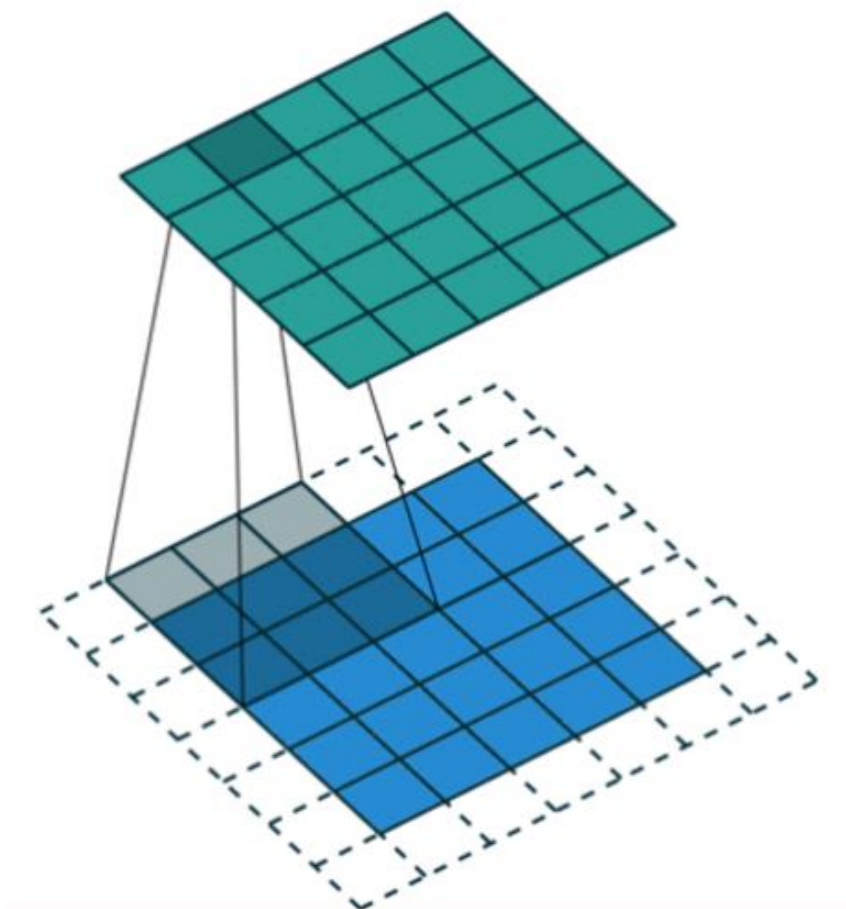


Hình 2.6 Minh họa cách di chuyển của bộ lọc

Bộ lọc di chuyển sang phải với Giá trị trượt cố định cho đến khi hoàn thành việc quét theo chiều rộng. Tiếp tục, nó nhảy xuống phía đầu bên trái của hình ảnh với cùng Giá trị trượt và lặp lại quá trình cho đến khi toàn bộ hình ảnh được duyệt qua.

Mục tiêu của phép tính tích chập là trích xuất các đặc trưng cấp cao như các cạnh (edges), từ hình ảnh đầu vào. Mạng nơron tích chập không nhất thiết chỉ giới hạn trong một lớp tích chập. Thông thường, lớp tích chập đầu tiên chịu trách nhiệm nắm bắt các đặc trưng cấp thấp như màu sắc (colors), hướng dốc (gradient orientation), v.v. Với các lớp tích chập được thêm vào, mô hình cũng nắm bắt các đặc trưng cấp cao, mang đến cho chúng ta một mạng lưới nơron tích chập có sự hiểu biết toàn diện về hình ảnh trong bộ dữ liệu, tương tự như cách chúng ta - con người hiểu về hình ảnh.

Có hai loại kết quả cho phép tính chập - một loại trong đó kết quả tích chập bị giảm chiều so với đầu vào và loại khác trong đó chiều của kết quả tích chập được tăng hoặc giữ nguyên. Điều này được thực hiện bằng cách áp dụng phép Đệm hợp lệ (Valid Padding) trong trường hợp trước, hoặc phép Đệm tương tự (Same Padding) trong trường hợp sau.



Hình 2.7 Một hình ảnh 5x5x1 được đệm thành một hình ảnh 6x6x1

3.0	3.0	3.0
3.0	3.0	3.0
3.0	2.0	3.0

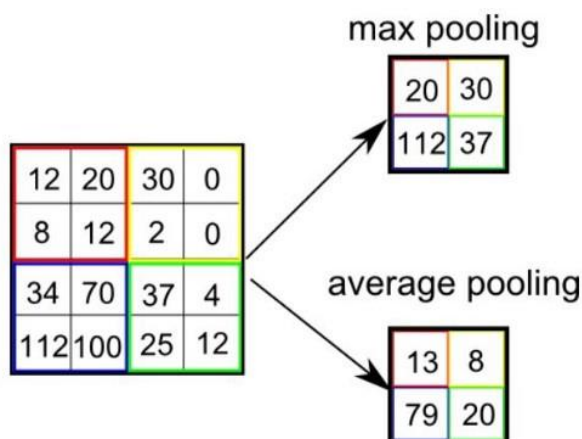
3	3	2	1	0
0	0	1	3	1
3	1	2	2	3
2	0	0	2	2
2	0	0	0	1

Hình 2.8 Ma trận gộp 3x3 từ một ma trận tích chập 5x5

Tương tự như lớp tích chập (Convolutional Layer), lớp gộp (pooling) chịu trách nhiệm để là giảm chiều kết quả tích chập (Convolved Feature). Điều này nhằm mục đích để giảm chi phí tính toán cần phải có để xử lý dữ liệu thông qua việc giảm kích thước tính năng đầu vào. Hơn nữa, nó rất hữu ích để trích xuất các đặc trưng cốt lõi, cái thường bất biến trước các phép xoay và phép trượt, do đó làm cho quá trình huấn luyện mô hình hiệu quả hơn.

Có hai loại phép gộp: Gộp cực đại (Max Pooling) và Gộp trung bình (Average Pooling). Phép gộp cực đại trả về giá trị lớn nhất từ phần hình ảnh được bao phủ bởi bộ lọc. Trong khi đó, phép gộp trung bình trả về giá trị trung bình của tất cả các giá trị từ phần hình ảnh được bao phủ bởi bộ lọc.

Phép gộp cực đại cũng hoạt động như một công cụ khử nhiễu. Nó loại bỏ các nguồn nhiễu và thực hiện khử nhiễu song song với giảm kích thước. Mặt khác, phép gộp trung bình chỉ đơn giản thực hiện giảm kích thước như một cơ chế khử nhiễu. Do đó, chúng ta có thể nói rằng phép gộp cực đại hoạt động tốt hơn rất nhiều so với phép gộp trung bình.



Hình 2.9 Các kiểu phép gộp

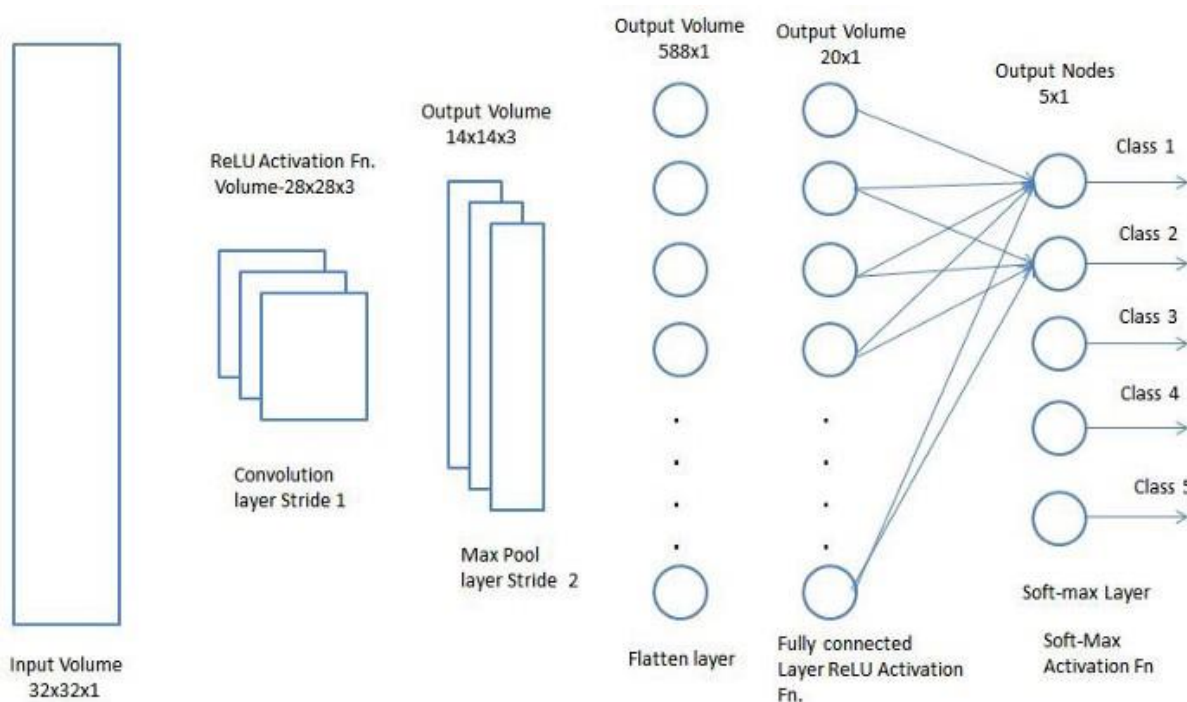
Lớp tích chập (Convolutional Layer) và lớp gộp (Pooling layer), kết hợp với nhau tạo thành lớp thứ i của mạng nơron tích chập. Tùy thuộc vào độ phức tạp của ảnh, số lượng các lớp như vậy có thể được tăng lên để có thể bắt được các đặc trưng ở mức độ chi tiết hơn nữa, nhưng chi phí cho sức mạnh tính toán cũng sẽ nhiều hơn.

Sau khi đi qua quá trình trên, chúng ta đã thiết lập thành công mô hình nơron tích chập để học các đặc trưng. Tiếp theo, chúng ta sẽ làm phẳng (flattening) đầu ra cuối và đưa nó vào mạng nơron thông thường cho mục đích phân loại.

Có nhiều kiến trúc CNN khác nhau có sẵn, sẽ là chìa khóa trong việc xây dựng các thuật toán tạo sức mạnh tính toán cho AI trong tương lai gần. Một số thuật toán đã được liệt kê dưới đây:

- LeNet
- AlexNet
- VGGNet
- GoogLeNet
- ResNet
- ZFNet

2.2.3 Neural network



Hình 2.10 Minh họa 1 lớp neural network

Sử dụng mạng nơron kết nối đầy đủ là cách làm phổ biến nhất để học các tổ hợp phi tuyến từ các đặc trưng trích xuất từ kết quả ma trận tích chập đầu ra. Mạng nơron kết nối đầy đủ có thể học được các đặc trưng trong không gian phi tuyến này.

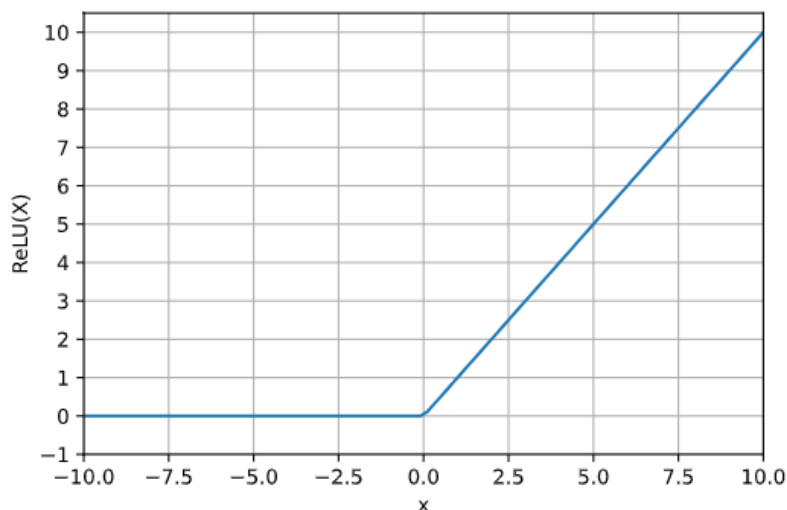
Chúng ta đã chuyển đổi hình ảnh đầu vào thành một dạng thích hợp cho mạng nơron đa lớp. Chúng ta sẽ làm phẳng hình ảnh đầu vào này thành một vector cột. Vector đầu ra đã được làm phẳng sẽ được đưa vào một mạng nơron suy luận tiến (feedforward) và phương pháp truyền ngược (backpropagation) được áp dụng cho quá trình huấn luyện. Qua một loạt lần lặp, mô hình có thể phân biệt giữa các đặc trưng cốt lõi và các đặc trưng không quan trọng trong hình ảnh và phân loại chúng bằng kỹ thuật Phân loại Softmax (softmax classification).

2.2.4 Hàm kích hoạt ReLU

Công thức

$$f(x)=\max(0,x) \quad (2.1)$$

Phân tích



Hình 2.10 Minh họa đồ thị hàm ReLU

Hàm ReLU [3] đang được sử dụng khá nhiều trong những năm gần đây khi huấn luyện các mạng neuron. ReLU đơn giản lọc các giá trị < 0 . Nhìn vào công thức chúng ta dễ dàng hiểu được cách hoạt động của nó. Một số ưu điểm khá vượt trội của nó so với Sigmoid và Tanh:

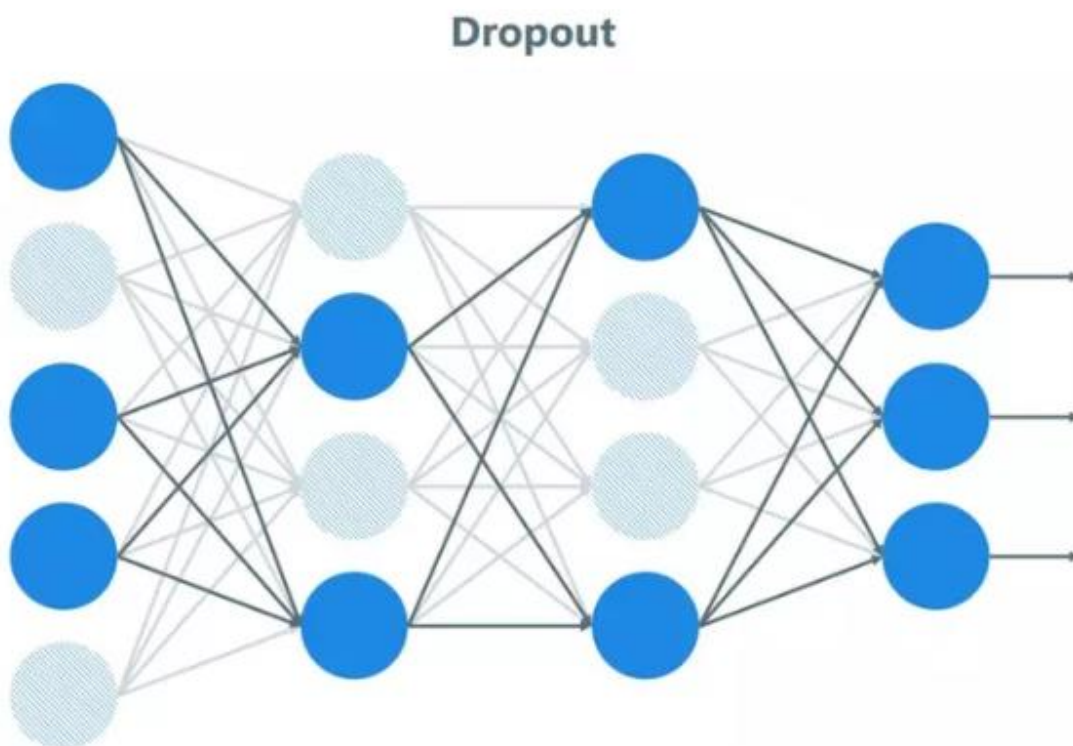
- Tốc độ hội tụ nhanh hơn hẳn. ReLU có tốc độ hội tụ nhanh gấp 6 lần Tanh. Điều này có thể do ReLU không bị bão hòa ở 2 đầu như Sigmoid và Tanh.
- Tính toán nhanh hơn. Tanh và Sigmoid sử dụng hàm exp và công thức phức tạp hơn ReLU rất nhiều do vậy sẽ tốn nhiều chi phí hơn để tính toán.

Tuy nhiên ReLU cũng có một nhược điểm: Với các node có giá trị nhỏ hơn 0, qua ReLU activation sẽ thành 0, hiện tượng này gọi là “Dying ReLU”. Nếu các node bị chuyển thành 0 thì sẽ không có ý nghĩa với bước linear activation ở lớp tiếp theo và các

hệ số tương ứng từ node đáy cũng không được cập nhật với gradient descent. => Leaky ReLU ra đời. Ngoài ra khi learning rate lớn, các trọng số (weights) có thể thay đổi theo cách làm tắt cả neuron dừng việc cập nhật.

2.2.5 Kỹ thuật Dropout

Hiểu 1 cách đơn giản thì Dropout [4] là việc bỏ qua các đơn vị (tức là 1 nút mạng) trong quá trình đào tạo 1 cách ngẫu nhiên. Bằng việc bỏ qua này thì đơn vị đó sẽ không được xem xét trong quá trình forward và backward. Theo đó, p được gọi là xác suất giữ lại 1 nút mạng trong mỗi giai đoạn huấn luyện, vì thế xác suất nó bị loại bỏ là $(1 - p)$.



Hình 2.11 Minh họa việc áp dụng dropout trong neural network

Nếu 1 lớp neural network có quá nhiều tham số và chiếm hầu hết tham số, các nút mạng trong lớp đó quá phụ thuộc lẫn nhau trong quá trình huấn luyện thì sẽ hạn chế sức mạnh của mỗi nút, dẫn đến việc kết hợp quá mức.

Ưu điểm của việc áp dụng dropout:

- Dropout sẽ được học thêm các tính năng mạnh mẽ hữu ích
- Nó gần như tăng gấp đôi số epochs cần thiết để hội tụ. Tuy nhiên, thời gian cho mỗi epoch là ít hơn.
- Ta có H đơn vị ẩn, với xác suất bỏ học cho mỗi đơn vị là $(1 - p)$ thì ta có thể có 2^H mô hình có thể có. Nhưng trong giai đoạn test, tất cả các nút mạng phải được xét đến, và mỗi activation sẽ giảm đi 1 hệ số p

2.3 Các kiến trúc mạng CNN

2.3.1 Kiến trúc mạng ResNet

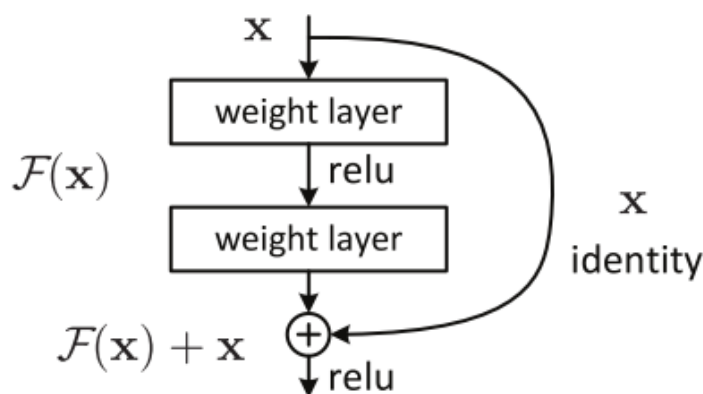
ResNet (Residual Network) [5] được giới thiệu đến công chúng vào năm 2015 và thậm chí đã giành được vị trí thứ 1 trong cuộc thi ILSVRC 2015 với tỉ lệ lỗi top 5 chỉ 3.57%. Không những thế nó còn đứng vị trí đầu tiên trong cuộc thi ILSVRC and COCO 2015 với ImageNet Detection, ImageNet localization, Coco detection và Coco segmentation. Hiện tại thì có rất nhiều biến thể của kiến trúc ResNet với số lớp khác nhau như ResNet-18, ResNet-34, ResNet-50, ResNet-101, ResNet-152,... Với tên là ResNet theo sau là một số chỉ kiến trúc ResNet với số lớp nhất định.

Trước hết thì Backpropagation Algorithm là một kỹ thuật thường được sử dụng trong quá trình training. Ý tưởng chung của thuật toán là sẽ đi từ output layer đến input layer và tính toán gradient của cost function tương ứng cho từng parameter (weight) của mạng. Gradient Descent sau đó được sử dụng để cập nhật các parameter đó.

Toàn bộ quá trình trên sẽ được lặp đi lặp lại cho tới khi mà các parameter của network được hội tụ. Thông thường chúng ta sẽ có một hyperparameter (số Epoch - số lần mà training set được duyệt qua một lần và weights được cập nhật) định nghĩa cho số lượng vòng lặp để thực hiện quá trình này. Nếu số lượng vòng lặp quá nhỏ thì ta gặp phải trường hợp mạng có thể sẽ không cho ra kết quả tốt và ngược lại thời gian training sẽ lâu nếu số lượng vòng lặp quá lớn.

Tuy nhiên, trong thực tế Gradients thường sẽ có giá trị nhỏ dần khi đi xuống các layer thấp hơn. Dẫn đến kết quả là các cập nhật thực hiện bởi Gradients Descent không làm thay đổi nhiều weights của các layer đó và làm chúng không thể hội tụ và mạng sẽ không thu được kết quả tốt. Hiện tượng như vậy gọi là Vanishing Gradients.

Cho nên giải pháp mà ResNet đưa ra là sử dụng kết nối "tắt" đồng nhất để xuyên qua một hay nhiều lớp. Một khối như vậy được gọi là một Residual Block, như trong hình



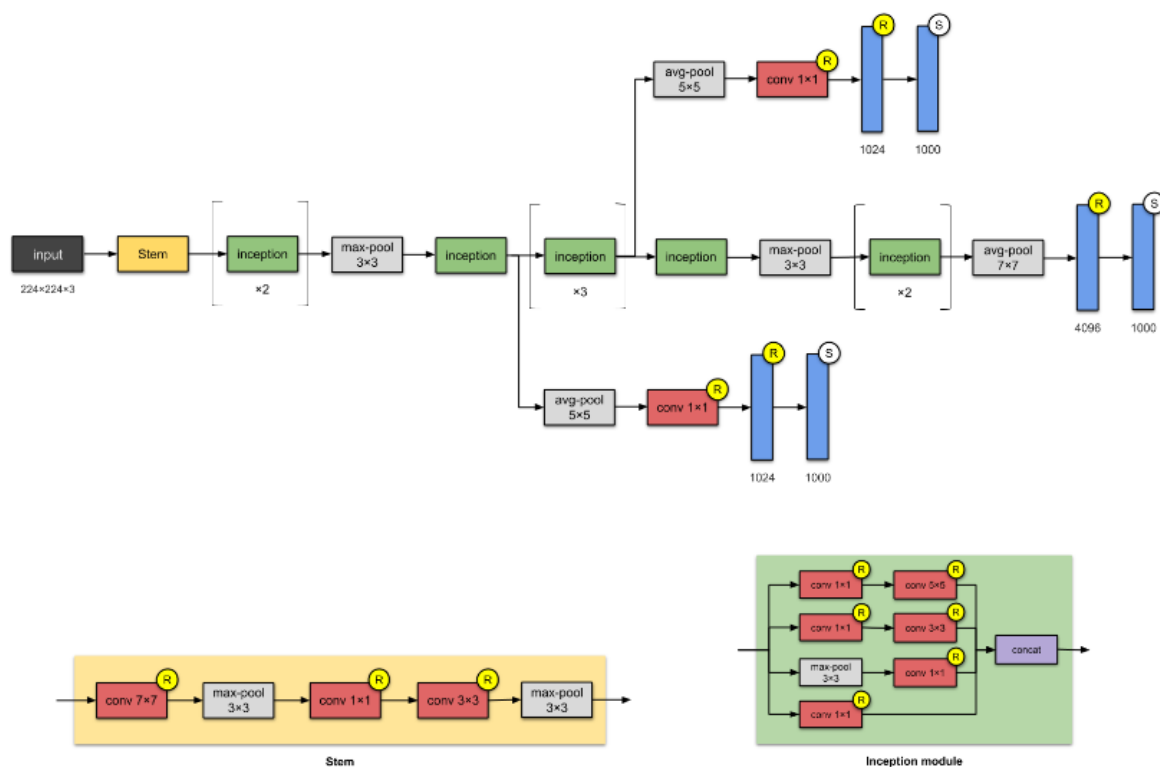
Hình 2.12 Kết nối 'tắt' trong ResNet

ResNet gần như tương tự với các mạng gồm có convolution, pooling, activation và fully-connected layer. Ảnh bên trên hiển thị khối dư được sử dụng trong mạng. Xuất hiện một mũi tên cong xuất phát từ đầu và kết thúc tại cuối khối dư. Hay nói cách khác là sẽ bổ sung Input X vào đầu ra của layer, hay chính là phép cộng mà ta thấy trong hình minh họa, việc này sẽ chống lại việc đạo hàm bằng 0, do vẫn còn cộng thêm X. Với $H(x)$ là giá trị dự đoán, $F(x)$ là giá trị thật (nhãn), chúng ta muốn $H(x)$ bằng hoặc xấp xỉ $F(x)$

Thực tế, ResNet không phải là kiến trúc đầu tiên sử dụng các kết nối tắt, Highway Network là một ví dụ. Trong thử nghiệm cho thấy Highway Network hoạt động không tốt hơn ResNet. Giải pháp ResNet đưa ra đơn giản hơn và tập trung vào cải thiện thông tin thông qua độ dốc của mạng. Sau ResNet hàng loạt biến thể của kiến trúc này được giới thiệu. Thử nghiệm cho thấy những kiến trúc này có thể được huấn luyện mạng nơ ron với độ sâu hàng nghìn lớp và nó nhanh chóng trở thành kiến trúc phổ biến nhất trong Computer Vision.

2.3.2 Kiến trúc mạng Inception-V1

Mạng Inception-V1 [6] đã dành chiến thắng ở cuộc thi ImageNet vào năm 2015. Kiến trúc này đã giải quyết một câu hỏi lớn trong mạng CNN đó là sử dụng kernel_size với kích thước bao nhiêu thì hợp lý. Các kiến trúc mạng nơ ron trước đó đều sử dụng các bộ lọc với đa dạng các kích thước 11x11, 5x5, 3x3 cho tới nhỏ nhất là 1x1. Một khám phá được đưa ra bởi bài báo đó là việc cùng kết hợp đồng thời các bộ lọc này vào cùng một block có thể mang lại hiệu quả đó chính là kiến trúc khối Inception.



Hình 2.13 Kiến trúc GoogleNet - Inception version 1

Khối Inception:

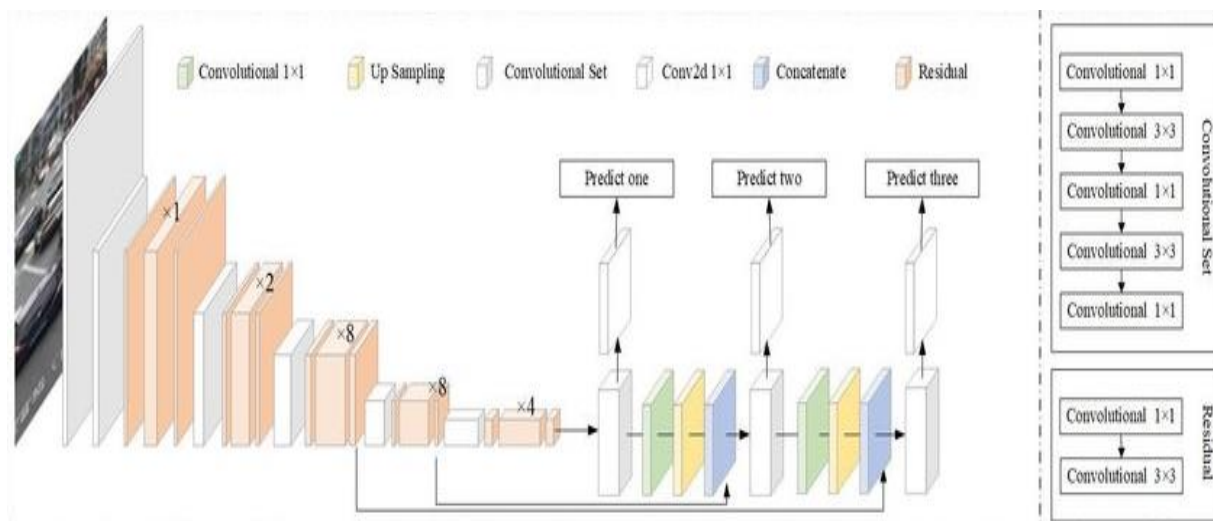
- Khối Inception sẽ bao gồm 4 nhánh song song. Các bộ lọc kích thước lần lượt là 1×1 , 3×3 , 5×5 được áp dụng trong Inception Module giúp trích lọc được đa dạng đặc trưng trên những vùng nhận thức có kích thước khác nhau.
- Ở đầu các nhánh 1, 2, 4 từ trên xuống, phép tích chập 1×1 được sử dụng trên từng điểm ảnh như một kết nối neural network nhằm mục đích giảm độ sâu kênh và số lượng tham số của mô hình. Ví dụ: Ở block trước chúng ta có kích thước $\text{width} \times \text{height} \times \text{channels} = 12 \times 12 \times 256$. Sau khi áp dụng 32 bộ lọc kích thước 1×1 sẽ không làm thay đổi width, height và độ sâu giảm xuống 32, output shape lúc này có kích thước là $12 \times 12 \times 32$. Ở layer liên sau, khi thực hiện tích chập trên toàn bộ độ sâu, chúng ta chỉ khởi tạo các bộ lọc có độ sâu 32 thay vì 256. Do đó số lượng tham số giảm đi một cách đáng kể.
- Nhánh thứ 3 từ trên xuống chúng ta giảm chiều dữ liệu bằng một layer max-pooling kích thước 3×3 và sau đó áp dụng bộ lọc kích thước 1×1 để thay đổi số kênh.
- Các nhánh áp dụng padding và stride sao cho đầu ra có cùng kích cỡ chiều dài và chiều rộng. Cuối cùng ta concatenate toàn bộ kết quả đầu ra của các khối theo kênh để thu được output có kích thước bằng với input.

Khối Inception được lặp lại 7 lần trong kiến trúc Inception-V1. Toàn bộ mạng bao gồm 22 Layers, lớn hơn gần gấp đôi so với VGG-16 [7]. Nhờ áp dụng tích chập 1×1 giúp tiết kiệm số lượng tham số xuống chỉ còn 5 triệu, ít hơn gần 27 lần so với VGG-16.

2.3.3 Kiến trúc mạng DarkNet

Mạng DarkNet [8] thường được gắn liền với YOLO. Trong YOLO, DarkNet thường là mạng CNN cơ bản để trích đặc trưng. Qua nhiều thế hệ YOLO thì nhiều phiên bản như DarkNet19, DarkNet53,... ra đời.

Mạng DarkNet53 gồm 53 convolutional layers kết nối liên tiếp, mỗi layer được theo sau bởi một batch normalization và một activation Leaky ReLU. Để giảm kích thước của output sau mỗi convolution layer, giảm số lượng bằng các filter với kích thước là 2. Mẹo này có tác dụng giảm thiểu số lượng tham số cho mô hình.



Hình 2.14 Kiến trúc mạng DarkNet53

Darknet-53 mạnh hơn nhiều so với Darknet-19 nhưng vẫn hiệu quả (efficient) hơn ResNet-101 và ResNet-152 ở mặt số phép tính toán trong khi hiệu suất ngang nhau. Ưu điểm so với các mạng ResNet như:

- Darknet-53 có performance tốt hơn ResNet-101 và nhanh hơn 1.5 lần.
- Darknet-53 có performance ngang với ResNet-152 và nhanh hơn 2 lần.

Điều này cho thấy network structure của Darknet-53 sử dụng GPU tốt hơn, giúp cho việc đánh giá tốt hiệu quả hơn và nhanh hơn.

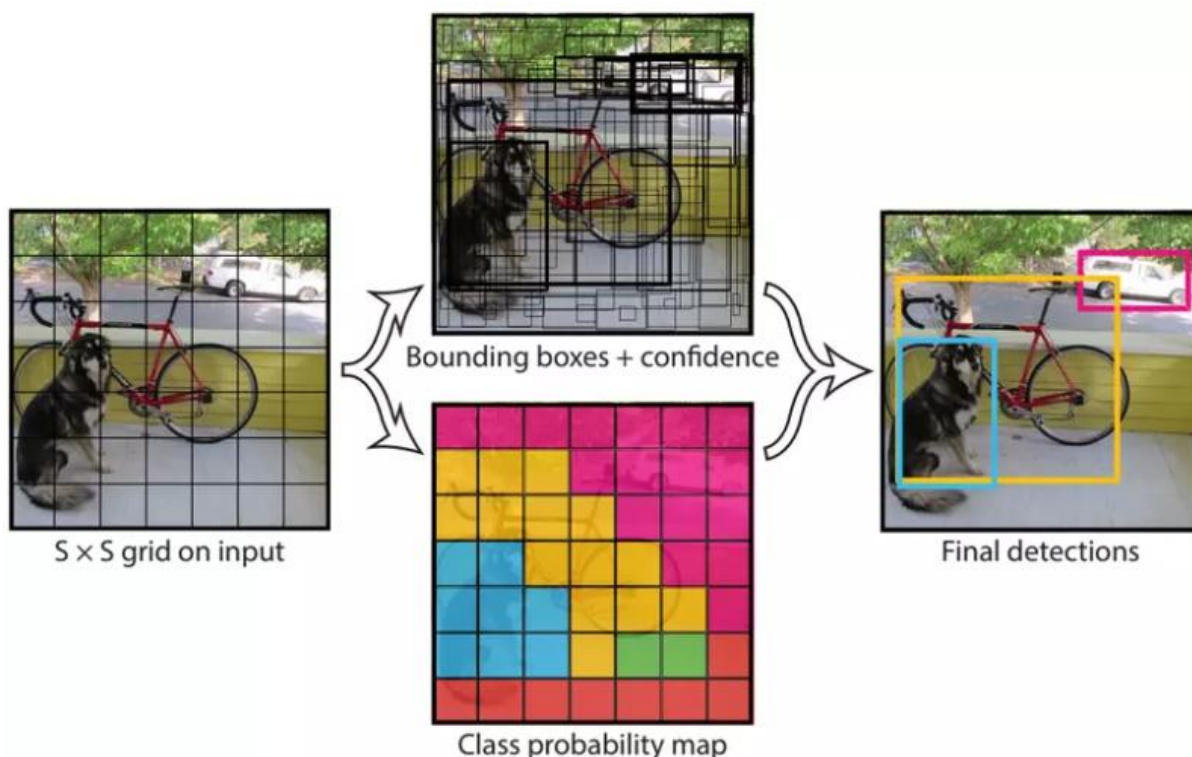
2.4 Thuật toán YOLO

2.4.1 YOLO (You Only Look One)

Object Detection là một bài toán quan trọng trong lĩnh vực Computer Vision, thuật toán Object Detection được chia thành 2 nhóm chính:

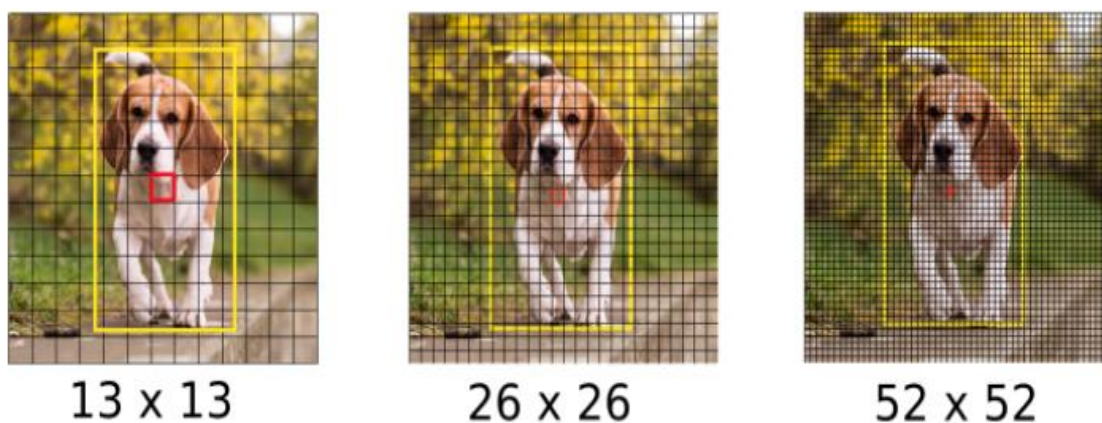
- Họ các mô hình RCNN [9] (Region-Based Convolutional Neural Networks) để giải quyết các bài toán về định vị và nhận diện vật thể.
- Họ các mô hình về YOLO [10] (You Only Look Once) dùng để nhận dạng đối tượng được thiết kế để nhận diện các vật thể real-time

YOLO là một mô hình mạng CNN cho việc phát hiện, nhận dạng, phân loại đối tượng. YOLO được tạo ra từ việc kết hợp giữa các convolutional layers và connected layers. Trong đó các convolutional layers sẽ trích xuất ra các feature của ảnh, còn full-connected layers sẽ dự đoán ra xác suất đó và tọa độ của đối tượng.



Hình 2.15 Minh họa thuật toán YOLO hoạt động

Đầu vào của mô hình là một ảnh, mô hình sẽ nhận dạng ảnh đó có đối tượng nào hay không, sau đó sẽ xác định tọa độ của đối tượng trong bức ảnh. Ảnh đầu vào được chia thành thành $S \times S$ ô thường thì sẽ là 3×3 , 7×7 , 9×9 ... việc chia ô này có ảnh hưởng tới việc mô hình phát hiện đối tượng. Với Input là 1 ảnh, đầu ra mô hình là một ma trận 3 chiều có kích $S \times S \times (5 \times N + M)$ với số lượng tham số mỗi ô là $(5 \times N + M)$ với N và M lần lượt là số lượng Box và Class mà mỗi ô cần dự đoán. Ví dụ với hình ảnh trên chia thành 7×7 ô, mỗi ô cần dự đoán 2 bounding box và 3 object : con chó, ô tô, xe đạp thì output là $7 \times 7 \times 13$, mỗi ô sẽ có 13 tham số, kết quả trả về $(7 \times 7 \times 2 = 98)$ bounding box. Chúng ta sẽ cùng giải thích con số $(5 \times N + M)$ được tính như thế nào.



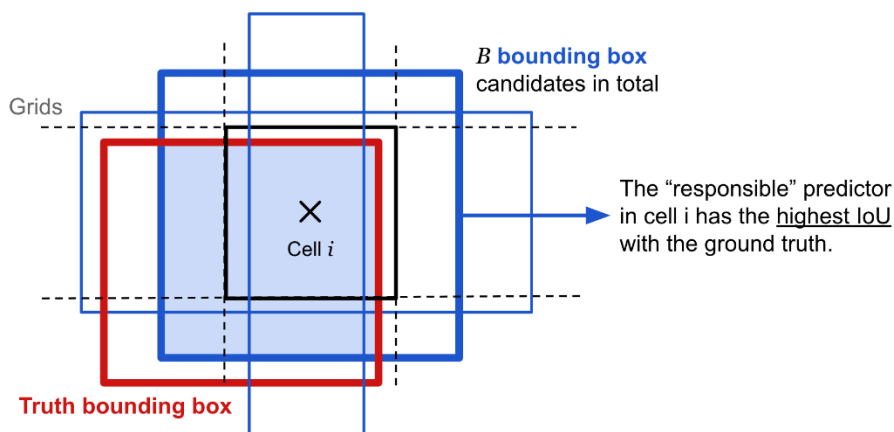
Hình 2.16 Feature map của YOLO với từng kích thước khác nhau

Dự đoán mỗi bounding box gồm 5 thành phần : (x, y, w, h, prediction) với (x, y) là tọa độ tâm của bounding box, (w, h) lần lượt là chiều rộng và chiều cao của bounding box, prediction được định nghĩa $\text{Pr}(\text{Object}) * \text{IOU}(\text{pred}, \text{truth})$ xin trình bày sau. Với hình ảnh trên như ta tính mỗi ô sẽ có 13 tham số, ta có thể hiểu đơn giản như sau tham số thứ 1 sẽ chỉ ra ô đó có chứa đối tượng nào hay không $P(\text{Object})$, tham số 2, 3, 4, 5 sẽ trả về x, y, w, h của Box1. Tham số 6, 7, 8, 9, 10 tương tự sẽ Box2, tham số 11, 12, 13 lần lượt là xác suất ô đó có chứa object1($P(\text{chó}|\text{object})$, object2($P(\text{ô tô}|\text{object})$), object3($P(\text{xe đạp}|\text{object})$)). Lưu ý rằng tâm của bounding box nằm ở ô nào thì ô đó sẽ chứa đối tượng, cho dù đối tượng có thể ở các ô khác thì cũng sẽ trả về là 0. Vì vậy việc mà 1 ô chứa 2 hay nhiều tâm của bounding box hay đối tượng thì sẽ không thể detect được, đó là một hạn chế của mô hình YOLO1, vậy ta cần phải tăng số lượng ô chia trong 1 ảnh lên đó là lí do vì sao mình nói việc chia ô có thể làm ảnh hưởng tới việc mô hình phát hiện đối tượng.

2.4.2 Anchor box

Để tìm được bounding box cho vật thể, YOLO sẽ cần các anchor box làm cơ sở ước lượng. Những anchor box [11] này sẽ được xác định trước và sẽ bao quanh vật thể một cách tương đối chính xác. Sau này thuật toán regression bounding box sẽ tinh chỉnh lại anchor box để tạo ra bounding box dự đoán cho vật thể. Trong một mô hình YOLO:

Mỗi một vật thể trong hình ảnh huấn luyện được phân bố về một anchor box. Trong trường hợp có từ 2 anchor boxes trở lên cùng bao quanh vật thể thì ta sẽ xác định anchor box mà có IoU với ground truth bounding box là cao nhất.



Hình 2.17 Minh họa cho cách anchor box hoạt động


Mỗi một vật thể trong hình ảnh huấn luyện được phân bố về một cell trên feature map mà chứa điểm mid point của vật thể. Chẳng hạn như hình chú chó trong hình 3 sẽ được phân về cho cell màu đỏ vì điểm mid point của ảnh chú chó rơi vào đúng cell này. Từ cell ta sẽ xác định các anchor boxes bao quanh hình ảnh chú chó.

Như vậy khi xác định một vật thể ta sẽ cần xác định 2 thành phần gắn liền với nó là (cell, anchor box). Không chỉ riêng mình cell hoặc chỉ mình anchor box.

Một số trường hợp 2 vật thể bị trùng mid point, mặc dù rất hiếm khi xảy ra, thuật toán sẽ rất khó xác định được class cho chúng.

2.4.1 *IOU (INTERSECTION OVER UNION)*

Trên ta có đề cập prediction được định nghĩa $Pr(\text{Object}) * IOU(\text{pred}, \text{truth})$, ta sẽ làm IOU (INTERSECTION OVER UNION) [12] rõ hơn IOU(pred, truth) là gì. IOU (INTERSECTION OVER UNION) là hàm đánh giá độ chính xác của object detector trên tập dữ liệu cụ thể.

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$


Hình 2.18 Minh họa cách tính IOU

Trong đó Area of Overlap là diện tích phần giao nhau giữa predicted bounding box với ground-truth bounding box, còn Area of Union là diện tích phần hợp giữa predicted bounding box với ground-truth bounding box. Những bounding box được đánh nhãn bằng tay trong tập training set và test set. Nếu $IOU > 0.5$ thì prediction được đánh giá là tốt.

2.4.2 *Non-max suppression*

Do thuật toán YOLO dự báo ra rất nhiều bounding box trên một bức ảnh nên đối với những cell có vị trí gần nhau, khả năng các khung hình bị overlap là rất cao. Trong trường hợp đó YOLO sẽ cần đến non-max suppression [13] để giảm bớt số lượng các khung hình được sinh ra một cách đáng kể.



Hình 2.19 Từ 3 bounding box ban đầu cùng bao quanh chiếc xe đã giảm xuống còn một bounding box cuối cùng

Các bước của non-max suppression:

Step 1: Đầu tiên chúng ta sẽ tìm cách giảm bớt số lượng các bounding box bằng cách lọc bỏ toàn bộ những bounding box có xác suất chứa vật thể nhỏ hơn một ngưỡng threshold nào đó, thường là 0.5

Step 2: Đối với các bounding box giao nhau, non-max suppression sẽ lựa chọn ra một bounding box có xác suất chứa vật thể là lớn nhất. Sau đó tính toán chỉ số giao thoa IoU với các bounding box còn lại.

Nếu chỉ số này lớn hơn ngưỡng threshold thì điều đó chứng tỏ 2 bounding boxes đang overlap nhau rất cao. Ta sẽ xóa các bounding box có xác suất thấp hơn và giữ lại bounding box có xác suất cao nhất. Cuối cùng, ta thu được một bounding box duy nhất cho một vật thể.

2.5 Mô hình FaceNet

2.5.1 FaceNet

FaceNet chính là một dạng siam network có tác dụng biểu diễn các bức ảnh trong một không gian euclidean n 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 [14] 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:

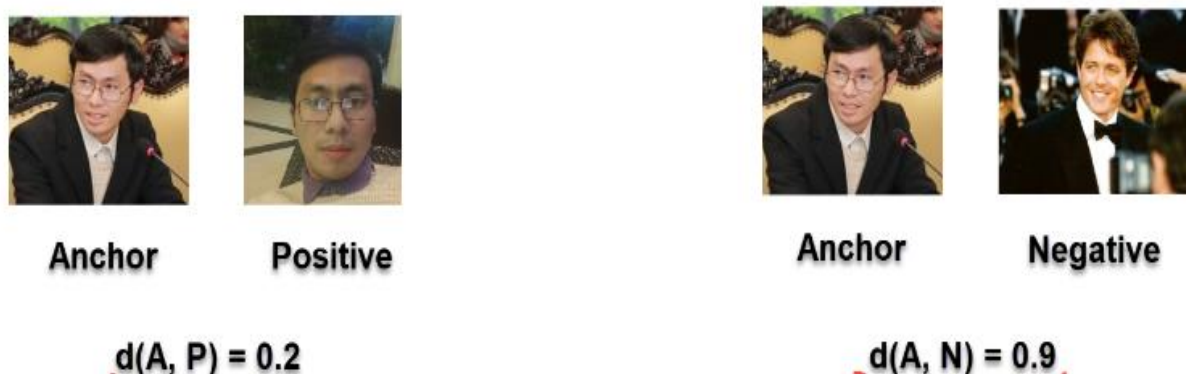
Base network áp dụng một mạng convolutional neural network và giảm chiều dữ liệu xuống chỉ còn 128 chiều. Do đó quá trình suy diễn và dự báo nhanh hơn và đồng thời độ chính xác vẫn được đảm bảo.

Sử dụng loss function là hàm triplet 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.

2.5.2 Hàm mất mát Triplet

Trong FaceNet, quá trình encoding của mạng convolutional neural network đã giúp ta mã hóa bức ảnh về 128 chiều. Sau đó những véc tơ này sẽ làm đầu vào cho hàm loss function đánh giá khoảng cách giữa các véc tơ.

Để áp dụng triple loss, chúng ta cần lấy ra 3 bức ảnh trong đó có một bức ảnh là anchor. Anchor image cũng có tác dụng gần như vậy. Trong 3 ảnh thì ảnh anchor được cố định trước. Chúng ta sẽ lựa chọn 2 ảnh còn lại sao cho một ảnh là negative (của một người khác với anchor) và một ảnh là positive (cùng một người với anchor).



Hình 2.20 Minh họa các anchor, positive và negative

Kí hiệu ảnh Anchor, Positive, Negative lần lượt là A,P,N.

Mục tiêu của hàm loss function là tối thiểu hóa khoảng cách giữa 2 ảnh khi chúng là negative và tối đa hóa khoảng cách khi chúng là positive. Như vậy chúng ta cần lựa chọn các bộ 3 ảnh sao cho:

- Ảnh Anchor và Positive khác nhau nhất: cần lựa chọn để khoảng cách $d(A,P)$ lớn. Điều này cũng tương tự như bạn lựa chọn một ảnh của mình hồi nhỏ so với hiện tại để thuật toán học khó hơn. Nhưng nếu nhận biết được thì nó sẽ thông minh hơn.
- Ảnh Anchor và Negative giống nhau nhất: cần lựa chọn để khoảng cách $d(A,N)$ nhỏ. Điều này tương tự như việc thuật toán phân biệt được ảnh của một người anh em giống bạn với bạn.

Triplot loss function luôn lấy 3 bức ảnh làm input và trong mọi trường hợp ta kì vọng:

$$d(A, P) < d(A, N) \quad (2.2)$$

Để làm cho khoảng cách giữa vế trái và vế phải lớn hơn, chúng ta sẽ cộng thêm vào vế trái một hệ số α không âm rất nhỏ. Khi đó (2) trở thành:

$$\mathcal{L}(\mathbf{A}, \mathbf{P}, \mathbf{N}) = \sum_{i=0}^n \|f(\mathbf{A}_i) - f(\mathbf{P}_i)\|_2^2 - \|f(\mathbf{A}_i) - f(\mathbf{N}_i)\|_2^2 + \alpha$$

Trong đó n là số lượng các bộ 3 hình ảnh được đưa vào huấn luyện.

Sẽ không ảnh hưởng gì nếu ta nhận diện đúng ảnh Negative và Positive là cùng cặp hay khác cặp với Anchor. Mục tiêu của chúng ta là giảm thiểu các trường hợp mô hình nhận diện sai ảnh Negative thành Positive nhất có thể. Do đó để loại bỏ ảnh hưởng của các trường hợp nhận diện đúng Negative và Positive lên hàm loss function. Ta sẽ điều chỉnh giá trị đóng góp của nó vào hàm loss function về 0.

Tức là nếu:

$$\|f(\mathbf{A}) - f(\mathbf{P})\|_2^2 - \|f(\mathbf{A}) - f(\mathbf{N})\|_2^2 + \alpha \leq 0$$

sẽ được điều chỉnh về 0. Khi đó hàm loss function trở thành:

$$\mathcal{L}(\mathbf{A}, \mathbf{P}, \mathbf{N}) = \sum_{i=0}^n \max(\|f(\mathbf{A}_i) - f(\mathbf{P}_i)\|_2^2 - \|f(\mathbf{A}_i) - f(\mathbf{N}_i)\|_2^2 + \alpha, 0) \quad (2.3)$$

Như vậy khi áp dụng Triple loss vào các mô hình convolutional neural network chúng ta có thể tạo ra các biểu diễn véc tơ tốt nhất cho mỗi một bức ảnh. Những biểu diễn véc tơ này sẽ phân biệt tốt các ảnh Negative rất giống ảnh Positive. Và đồng thời các bức ảnh thuộc cùng một label sẽ trở nên gần nhau hơn trong không gian chiều euclidean.

Một chú ý quan trọng khi huấn luyện mô hình siam network với triplet function đó là chúng ta luôn phải xác định trước cặp ảnh (A,P) thuộc về cùng một người. Ảnh N sẽ được lựa chọn ngẫu nhiên từ các bức ảnh thuộc các nhãn còn lại. Do đó cần thu thập ít nhất 2 bức ảnh/1 người để có thể chuẩn bị được dữ liệu huấn luyện.

2.5.3 *Lựa chọn ảnh đầu vào*

Nếu lựa chọn triple input một cách ngẫu nhiên có thể ảnh hưởng cho bất đẳng thức (2.3) dễ dàng xảy ra vì trong các ảnh ngẫu nhiên, khả năng giống nhau giữa 2 ảnh là rất khó. Hầu hết các trường hợp đều thỏa mãn bất đẳng thức (2.3) và không gây ảnh hưởng đến giá trị của loss function do giá trị của chúng được set về 0. Như vậy việc học những bức ảnh Negative quá khác biệt với Anchor sẽ không có nhiều ý nghĩa.

Để mô hình khó học hơn và đồng thời cũng giúp mô hình phân biệt chuẩn xác hơn mức độ giống và khác nhau giữa các khuôn mặt, chúng ta cần lựa chọn các input theo bộ 3 khó học (hard triplets).

Ý tưởng là chúng ta cần tìm ra bộ ba (A,N,P) sao cho (2.3) là gần đạt được đẳng thức (xảy ra dấu =) nhất. Tức là $d(A,P)$ lớn nhất và $d(A,N)$ nhỏ nhất. Hay nói cách khác với mỗi Anchor A cần xác định:

Hard Positive: Bức ảnh Positive có khoảng cách xa nhất với Anchor tương ứng với nghiệm:

$$\operatorname{argmax}_{P_i}(d(A_i, P_i))$$

Hard Negative: Bức ảnh Negative có khoảng cách gần nhất với Anchor tương ứng với nghiệm:

$$\operatorname{argmin}_{P_i}(d(A_i, P_i))$$

Với i, j là nhãn của người trong ảnh.

Việc tính toán các trường hợp Hard Positive và Hard Negative có thể được thực hiện offline và lưu vào checkpoint hoặc có thể tính toán online trên mỗi mini-batch.

Chiến lược lựa chọn Triple images sẽ có ảnh hưởng rất lớn tới chất lượng của mô hình FaceNet. Nếu lựa chọn Triplet images tốt, FaceNet sẽ hội tụ nhanh hơn và đồng thời kết quả dự báo chuẩn xác hơn. Lựa chọn ngẫu nhiên dễ dẫn tới thuật toán không hội tụ.

2.5.4 *Learning Similarity*

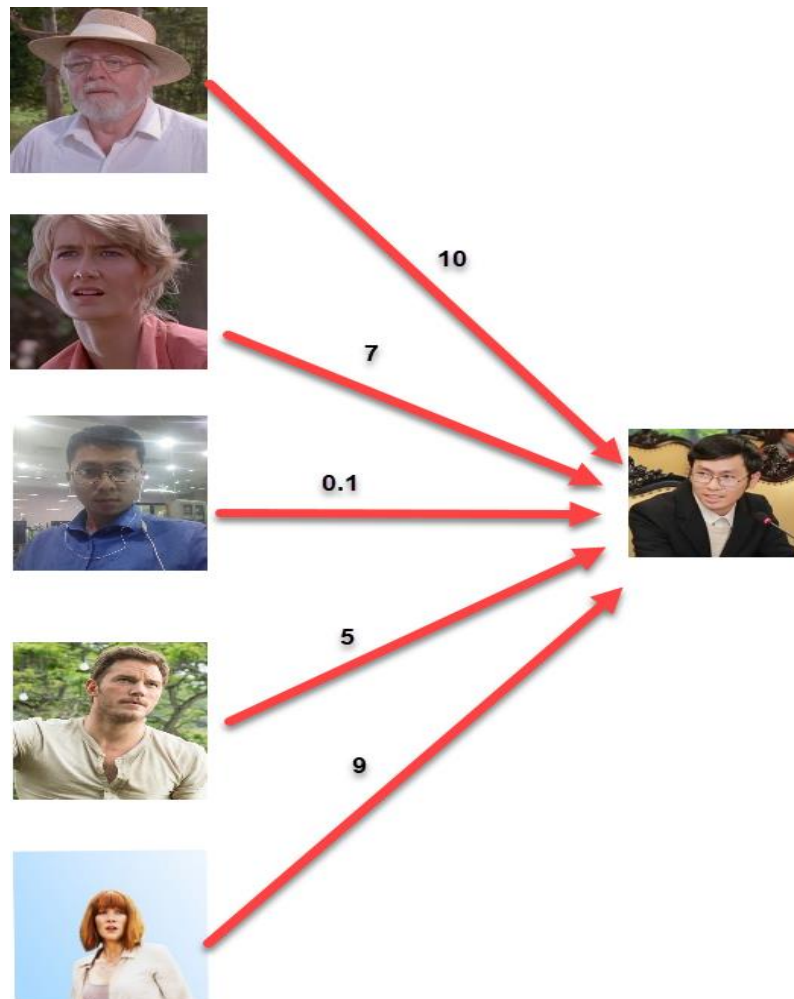
Phương pháp này dựa trên một phép đo khoảng cách giữa 2 bức ảnh, thông thường là các norm chuẩn l1 hoặc l2 sao cho nếu 2 bức ảnh thuộc cùng một người thì khoảng cách là nhỏ nhất và nếu không thuộc thì khoảng cách sẽ lớn hơn.

Nếu $d(\text{img1}, \text{img2}) \leq \tau \rightarrow \text{giống}$, ngược lại $(\text{img1}, \text{img2}) > \tau \rightarrow \text{khác}$

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 euclidean n chiều. Sau đó sử dụng khoảng cách để quyết định nhãn của chúng.



Hình 2.21 Minh họa phương pháp learning similarity

2.6 Các thang đo đánh giá chỉ số

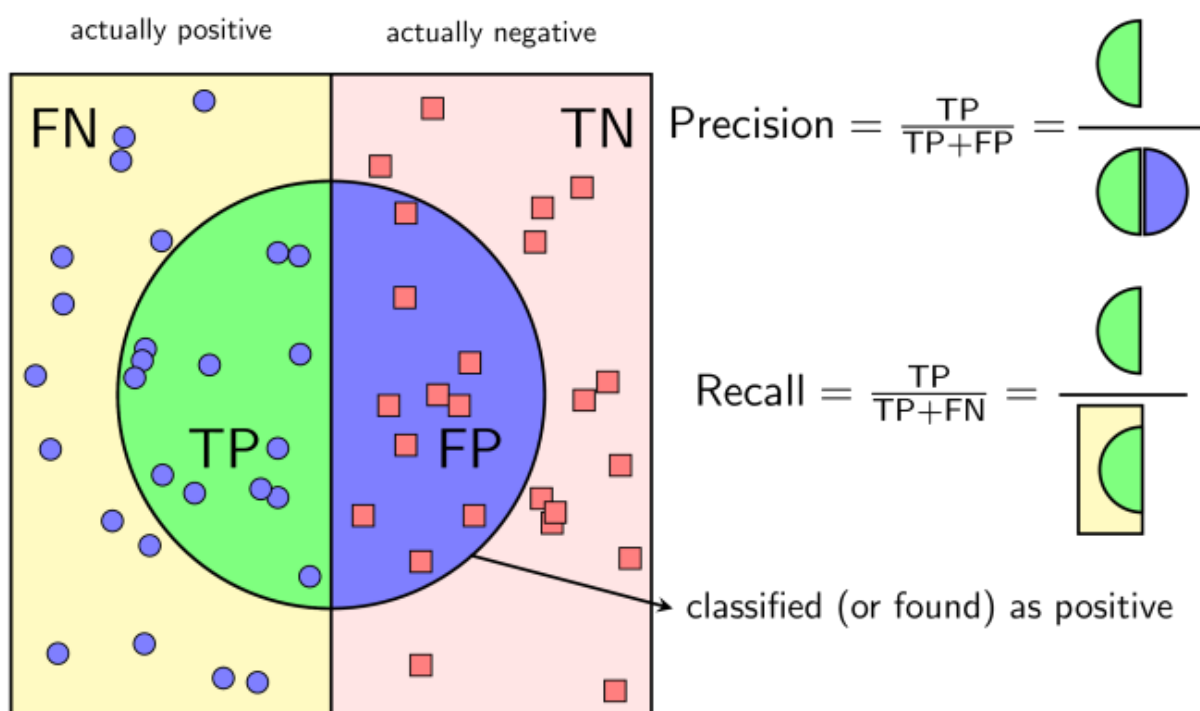
2.6.1 Độ chính xác

Khi xây dựng một mô hình Machine Learning, chúng ta cần một phép đánh giá để xem mô hình sử dụng có hiệu quả không và để so sánh khả năng của các mô hình.

Có rất nhiều cách đánh giá một mô hình phân lớp. Tùy vào những bài toán khác nhau mà chúng ta sử dụng các phương pháp khác nhau. Các phương pháp thường được sử dụng là: accuracy score, confusion matrix, ROC curve, Area Under the Curve, Precision and Recall, F1 score, Top R error, etc.

Cách đơn giản và hay được sử dụng nhất là accuracy (độ chính xác). Cách đánh giá này đơn giản tính tỉ lệ giữa số điểm được dự đoán đúng và tổng số điểm trong tập dữ liệu kiểm thử.

2.6.2 Recall và Precision



Hình 2.22 Minh họa các chỉ số precision và recall

Khi đó, Precision được định nghĩa là tỉ lệ số điểm Positive mô hình dự đoán đúng trên tổng số điểm mô hình dự đoán là Positive. Recall được định nghĩa là tỉ lệ số điểm Positive mô hình dự đoán đúng trên tổng số điểm thật sự là Positive (hay tổng số điểm được gán nhãn là Positive ban đầu).

Precision càng cao, tức là số điểm mô hình dự đoán là positive đều là positive càng nhiều. Precision = 1, tức là tất cả số điểm mô hình dự đoán là Positive đều đúng, hay không có điểm nào có nhãn là Negative mà mô hình dự đoán nhầm là Positive.

Recall càng cao, tức là số điểm là positive bị bỏ sót càng ít. Recall = 1, tức là tất cả số điểm có nhãn là Positive đều được mô hình nhận ra.

Để hiểu rõ hơn về hai chỉ số này, ta có thể tưởng tượng một ví dụ như sau. Khi một người nghĩ là mình đang mắc bệnh gì đó, họ thường đi đến bệnh viện để làm các xét nghiệm để bác sĩ chẩn đoán xem kết quả là dương tính hay là âm tính. Ta có hai trường

hợp về tình trạng bệnh là mắc bệnh hoặc không mắc bệnh. Ta có hai trường hợp về kết quả chẩn đoán là dương tính và âm tính.

Khi đó, precision là tỉ lệ người được chẩn đoán là dương tính thật sự mắc bệnh trên tổng số người được chẩn đoán là dương tính. Nếu precision = 0.9, thì cứ 100 người được chẩn đoán là dương tính thì sẽ thật sự có 90 người mắc bệnh. Precision càng cao thì xác suất người được chẩn đoán là dương tính có khả năng mắc bệnh càng cao.

Recall là tỉ lệ người được chẩn đoán là dương tính thật sự mắc bệnh trên tổng số người thật sự mắc bệnh. Nếu recall = 0.9, thì cứ 100 người mắc bệnh thì sẽ chẩn đoán 90 người dương tính. Recall càng cao thì xác suất người mắc bệnh được chẩn đoán là dương tính càng cao.

2.7 Kết luận chương

Trong chương 2 đã trình bày lý thuyết về các mô hình áp dụng trong bài toán. Ngoài ra các khái niệm cơ bản của học sâu và mạng nơron tích chập là những nền tảng cơ bản trong thị giác máy tính cũng được trình bày. Chương 3 sẽ đi sâu vào bài toán nhận diện khuôn mặt

CHƯƠNG 3 MÔ HÌNH NHẬN DẠNG KHUÔN MẶT

3.1 Giới thiệu chương

Trong chương sẽ trình bày tổng quan các phương pháp thực hiện để tiến hành nhận dạng danh tính người đeo khẩu trang và không đeo khẩu trang. Mô hình tổng thể cũng như các mô hình nhỏ hơn như mô hình phát hiện và mô hình nhận dạng danh tính sẽ được trình bày dưới dạng sơ đồ khối. Ngoài ra phạm vi khả năng mà đề tài nghiên cứu và thực hiện có thể giải quyết được cũng được trình bày

3.2 Hướng thực hiện và nghiên cứu

Áp dụng các phân lý thuyết của học sâu để giải quyết vấn đề bài toán nhận dạng khuôn mặt ứng dụng trong hệ thống chăm công. Bài toán được chia ra làm 3 phần nhỏ hơn bao gồm trình tự thực hiện lần lượt là:

- Phát hiện khuôn mặt ở trong ảnh hoặc video
- Phân loại khuôn mặt là đeo khẩu trang đúng, đeo khẩu trang sai và không đeo khẩu trang
- Nhận dạng danh tính khi khuôn mặt đeo khẩu trang đúng. Nhận dạng danh tính và cùng lúc nhắc nhở khi khuôn mặt không đeo khẩu trang. Nhắc nhở đeo khẩu trang đúng khi khuôn mặt đeo khẩu trang không đúng

Trong quá trình thực hiện khâu phát hiện và phân loại khuôn mặt đều áp dụng trên cùng một mô hình là YOLOv5 nên sẽ gọi tắt là mô hình phát hiện khuôn mặt. Quá trình huấn luyện và làm dữ liệu mô hình hoàn toàn từ những bước đầu tiên như thu thập dữ liệu, gán nhãn dữ liệu, lựa chọn tham số, huấn luyện mô hình và đánh giá mô hình. Để giảm tải quá trình gán nhãn dữ liệu thì một phần dữ liệu đã được gán nhãn sẵn và cũng như giảm tải quá trình huấn luyện sẽ sử dụng mô hình có sẵn là YOLO5s để tiếp tục huấn luyện dựa trên mô hình có sẵn đó.

Quá trình thực hiện nhận dạng danh tính sẽ thực hiện trên mô hình FaceNet để tạo các embedding. Mô hình áp dụng trên 2 tập dữ liệu là tập dữ liệu thực tế và dữ liệu người nổi tiếng (trình bày ở mục 4.3.1 a và b). Mô hình nhận dạng danh tính được xây dựng riêng theo thành 2 mô hình nhỏ theo 2 hướng là hướng nhận dạng đối với người đeo khẩu trang đúng và hướng nhận dạng đối với người không đeo khẩu trang. Mỗi hướng đi sẽ giải quyết 1 bài toán nhỏ hơn.

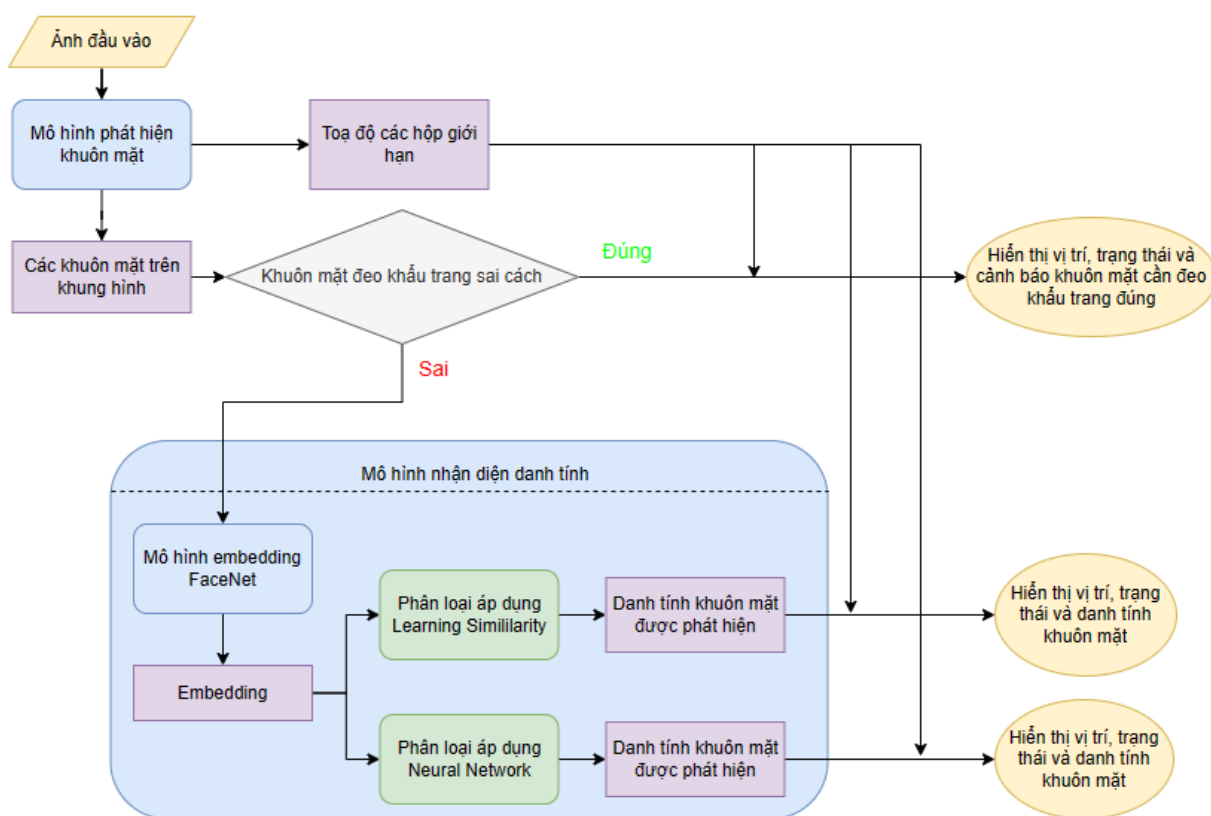
Trong nghiên cứu và thực hiện nhận dạng danh tính khuôn mặt dựa trên mô hình FaceNet thì sẽ không huấn luyện lại mô hình embedding của FaceNet. Các phương pháp

thực hiện đều thực hiện trên mô hình phân loại đầu ra của các embedding FaceNet, nghĩa là sẽ tập trung vào việc phân loại kết quả đầu ra của mô hình chứ không huấn luyện từ đầu mô hình. Lý do vì lượng dữ liệu cần để huấn luyện là rất lớn để mô hình hoạt động tốt và cần tài nguyên lớn để huấn luyện trong khi việc dựa trên mô hình có sẵn để phân loại có thể giảm tải lượng công việc và tài nguyên cần thiết

Hai phương pháp áp dụng lần lượt vào việc phân loại các embedding là learning similarity và neural network. Các nghiên cứu chi tiết, cách thực hiện thu thập dữ liệu và huấn luyện mô hình nhận dạng danh tính được trình bày chi tiết ở mục 4.3

3.3 Sơ đồ khối của mô hình tổng thể

Đề tài nhận dạng người đeo khẩu trang sẽ dựa trên bài toán nhận dạng khuôn mặt. Ứng dụng đề tài sẽ nhận dạng khuôn mặt trên hình là của đối tượng nào và nhận biết khuôn mặt đó có đeo khẩu trang hay không. Với các công việc phát hiện khuôn mặt sử dụng mô hình YOLO5s và mô hình nhận dạng danh tính sử dụng FaceNet kết hợp với các phương pháp phân loại khác nhau.



Hình 3.1 Tổng quan mô hình nhận dạng khuôn mặt

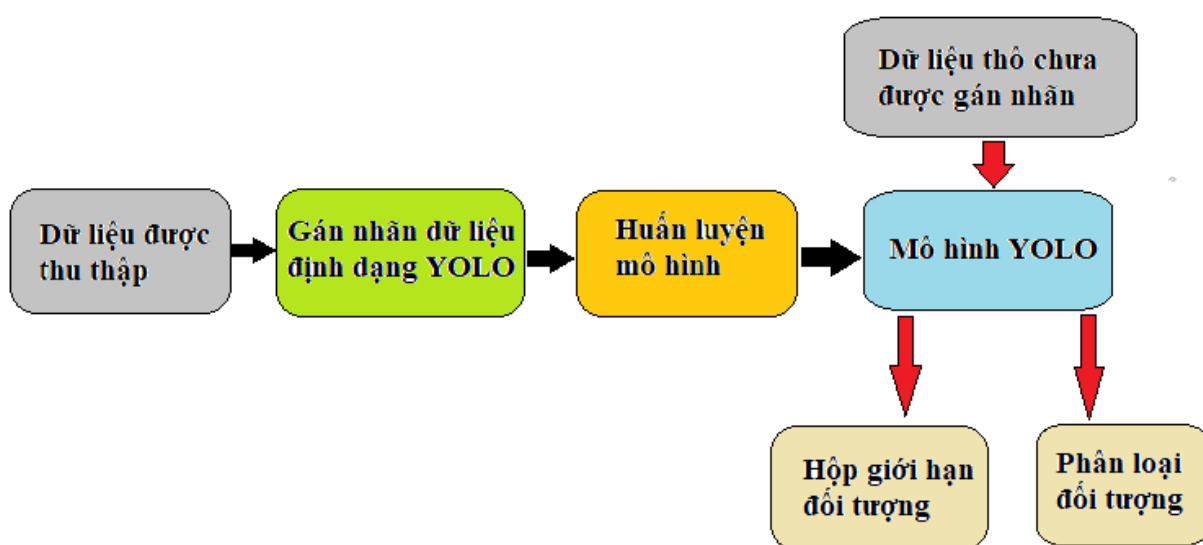
Ở hình 3.1, mô hình phát hiện khuôn mặt chính là mô hình YOLO với công việc thực hiện hai chức năng chính là phát hiện khuôn mặt và phân loại khuôn mặt. YOLO là thuật toán nổi tiếng trong việc phát hiện đối tượng dựa trên thời gian thực để trả về các hộp giới hạn (bounding box) và đối tượng được phân loại. Trong đề tài các hộp giới hạn sẽ

chứa bất kì khuôn mặt nào cho dù đối tượng có đeo khẩu trang hay không từ đó nhận biết khuôn mặt đeo khẩu trang đúng, đeo khẩu trang sai hay không đeo khẩu trang. Đầu ra của YOLO là một tập hợp các nhãn gán của đối tượng và toạ độ hộp giới hạn của đối tượng đó

Sau khi qua mô hình phát hiện khuôn mặt, dữ liệu các khuôn mặt được đi tới mô hình nhận diện danh tính là sự kết hợp mô hình embedding FaceNet và phân loại áp dụng nhiều phương pháp khác nhau. Mô hình FaceNet dựa trên ý tưởng mã hoá các khuôn mặt về một chuỗi các giá trị từ đó việc thực hiện phân loại các khuôn mặt dựa trên chuỗi giá trị đó. Việc mã hoá thường được đưa về một chuỗi giá trị có số chiều là 128 hoặc 512 (trong paper gốc các tác giả thực hiện mã hoá thành 128 giá trị, các mô hình huấn luyện sau này được mã hoá thành 512 giá trị) và áp dụng một hàm mất mát dùng để phân cụm là Triplet Loss để huấn luyện mô hình mã hoá. Đề tài nghiên cứu sẽ thực hiện hai phương pháp phân loại là learning similarity và neural network để phân loại các chuỗi giá trị mã hoá.

Dữ liệu cuối cùng sẽ kết hợp các kết quả tới từ cả hai mô hình là hộp giới hạn, trạng thái khuôn mặt có đeo khẩu trang hay không và danh tính của khuôn mặt đó. Nếu khuôn mặt đeo khẩu trang sai cách thì mô hình sẽ không tiến hành nhận diện danh tính mà chỉ hiển thị ngay kết quả của mô hình phát hiện và tiến hành cảnh báo khuôn mặt đeo khẩu trang đúng cách. Trong trường hợp khuôn mặt đeo khẩu trang đúng và không đeo khẩu trang mô hình sẽ trả kết quả vị trí khuôn mặt trên ảnh, hộp giới hạn và tên danh tính của khuôn mặt đó.

3.4 Sơ đồ khối mô hình phát hiện khuôn mặt



Hình 3.2 Sơ đồ khối mô hình phát hiện khuôn mặt

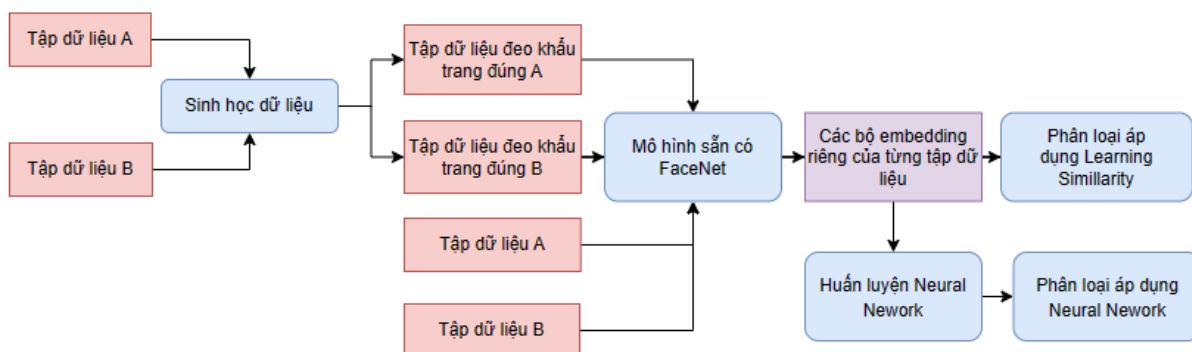
Mô hình được áp dụng và sử dụng trong bài toán phát hiện khuôn mặt là YOLOv5. Mô hình nổi tiếng với bài toán thời gian thực để phát hiện đối tượng với tên đầy đủ là ‘You Only Look One’, được phát triển qua nhiều phiên bản với các cái tên như v1, v2, v3.. .Trong đề tài phiên bản v5 được áp dụng để phát hiện khuôn mặt và phân loại các khuôn mặt đeo khẩu trang và không đeo khẩu trang. Trong phiên bản v5 còn rất nhiều kiến trúc mạng được hỗ trợ cho riêng từng thiết bị, trong đề tài kiến trúc mạng YOLO5s được sử dụng

Mô hình YOLO huấn luyện sẵn có 80 đối tượng được huấn luyện. Do yêu cầu của bài toán khác với mô hình đã huấn luyện sẵn nên sẽ thực hiện huấn luyện tiếp dựa trên mô hình có sẵn với việc thay đổi các thông số có sẵn để phục vụ đúng với yêu cầu của bài toán. Giảm từ 80 đối tượng xuống thành 3 đối tượng là đeo khẩu trang đúng, đeo khẩu trang sai và không đeo khẩu trang. Mục đích sử dụng mô hình có sẵn nhằm mục đích giảm thời gian huấn luyện cũng giảm số lượng dữ liệu dùng cho quá trình huấn luyện.

Ở hình 3.12, sau khi quá trình huấn luyện mô hình hoàn tất ta sẽ có mô hình YOLO hoàn thiện. Quá trình kiểm thử sẽ đưa từng ảnh qua mô hình YOLO và mô hình sẽ dự đoán về hai phần là hộp giới hạn chứa các khuôn mặt và một giá trị tương trưng cho từng đối tượng phân loại. Các kết quả này sẽ đưa vào mô hình nhận diện danh tính ở phía sau

3.5 Sơ đồ khối mô hình nhận dạng danh tính

Mô hình nhận dạng danh tính là thực hiện tổ hợp giữa các tập dữ liệu thu thập được với các phương pháp tiến hành nghiên cứu. Tùy vào từng loại tập dữ liệu các phương pháp đi kèm mà trường hợp sẽ được ứng dụng vào các tình huống thực tế



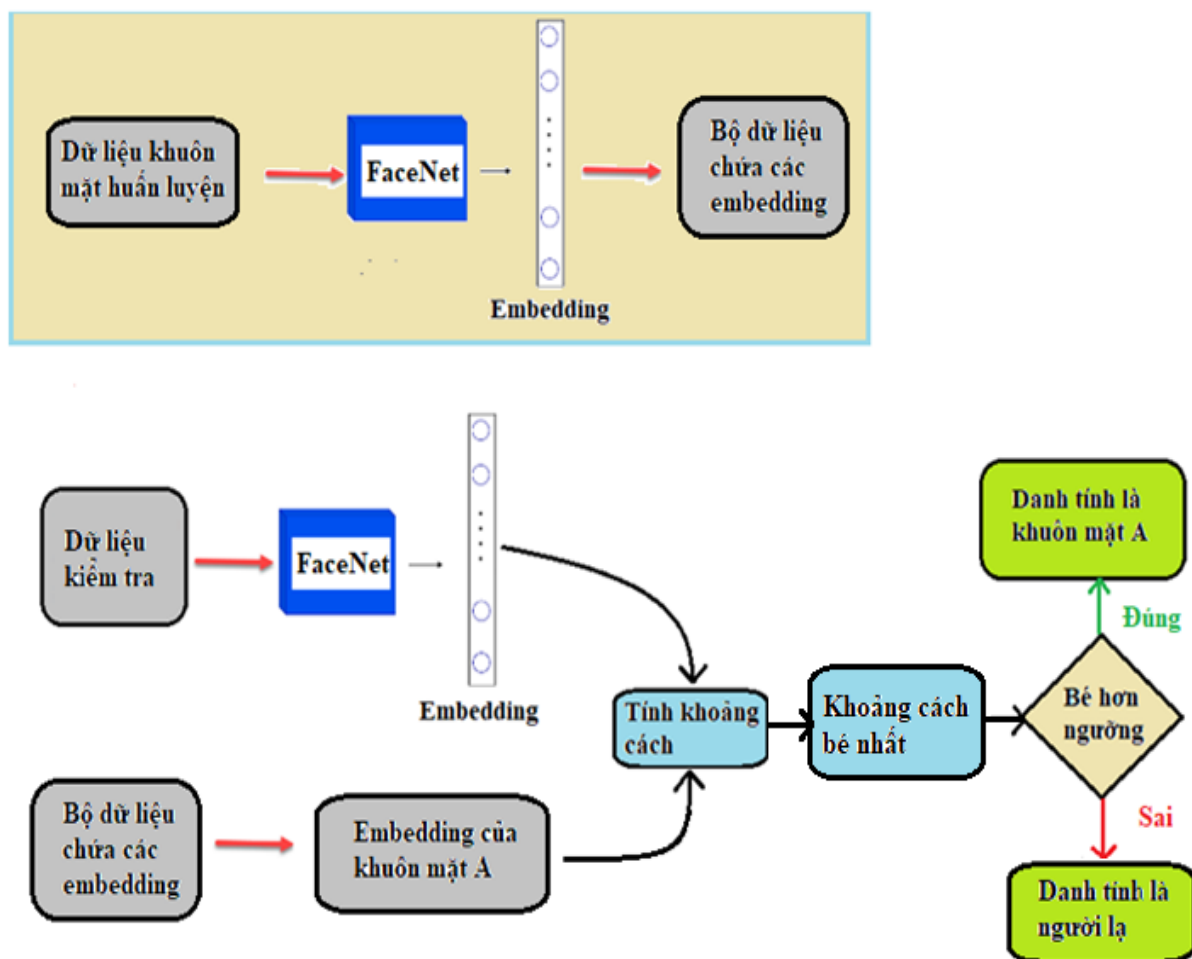
Hình 3.3 Sơ đồ khối phương pháp nghiên cứu huấn luyện mô hình nhận dạng danh tính

Trong hình 3.3, tập dữ liệu A là tập dữ liệu thực tế, tập dữ liệu B là tập dữ liệu người nổi tiếng (chi tiết về hai tập dữ liệu ở mục 3.6.2). Do dữ liệu không có khuôn mặt đeo khẩu trang nên sẽ tiến hành sinh học dữ liệu đeo khẩu trang, cách thức sinh học được

trình bày ở mục 4.3.1.3. Sẽ không tiến hành huấn luyện lại mô hình FaceNet mà áp dụng sẵn mô hình đã được huấn luyện. Mô hình FaceNet sẽ mã hoá các đặc trưng của khuôn mặt thành các embedding để áp dụng các phương pháp phân loại. Sau đó tùy vào phương pháp phân loại mà các công việc thực hiện sẽ khác nhau

3.5.1 Phân loại áp dụng *Learning similarity*

Phương pháp không cần xây dựng và huấn luyện mô hình nên learning similarity dễ dàng trong việc xây dựng và triển khai. Dựa trên khoảng cách ecudid 2 và việc đặt ngưỡng phương pháp dễ dàng và phân loại nhanh trong trường hợp lượng dữ liệu ít

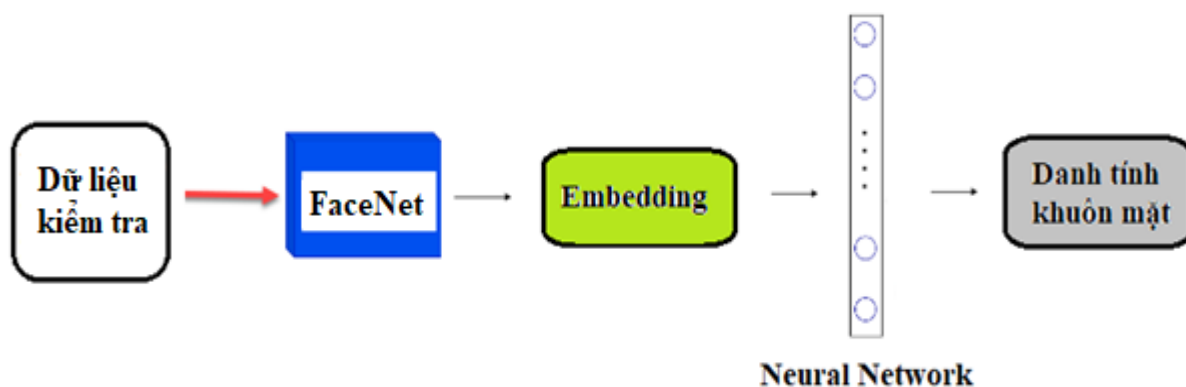


Hình 3.4 Sơ đồ khối phân loại áp dụng *learning similarity*

Ở hình 3.4, để tiện cho công việc kiểm tra thì các khuôn mặt đem đi huấn luyện sẽ được mã hoá sẵn và lưu trữ thành các file để tiện xử lý và giảm thời gian phân loại kết quả sau này.

3.5.2 Phân loại áp dụng *Neural network*

Phương pháp giải quyết bài toán đòi hỏi khả năng xây dựng và huấn luyện mô hình. Nhưng bù lại việc tối ưu mô hình sẽ cải thiện kết quả, đặc biệt đối với trường hợp khuôn mặt đeo khẩu trang và số lượng khuôn mặt lớn



Hình 3.5 Sơ đồ khối phân loại áp dụng neural network

3.6 Tổng quan các tập dữ liệu sử dụng

Việc thực hiện bài toán chia làm hai mô hình nên tùy thuộc vào từng mô hình mà cần tập dữ liệu tương ứng. Tập thứ nhất là dữ liệu phát hiện khuôn mặt là các ảnh chụp bình thường và được gán nhãn theo định dạng YOLO để huấn luyện mô hình. Tập dữ liệu thứ hai là các khuôn mặt đã được cắt ra từ khung hình để phục vụ quá trình nhận dạng danh tính của khuôn mặt đó.

3.6.1 Tập dữ liệu phát hiện khuôn mặt

Được thu thập để phục vụ mô hình phát hiện khuôn mặt. Lượng dữ liệu được định dạng theo kiểu YOLO và thường là 1 bức ảnh chứa 1 hoặc nhiều khuôn mặt. Các khuôn mặt không nhất thiết phải cùng 1 đối tượng trên cùng 1 bức ảnh mà có thể có nhiều đối tượng được gán nhãn trên cùng 1 bức ảnh. Ngoài ra các đối tượng che khuất nhau 1 vùng vẫn có thể được gán nhãn và phát hiện bình thường





Hình 3.6a và 3.6b Minh họa dữ liệu phát hiện khuôn mặt

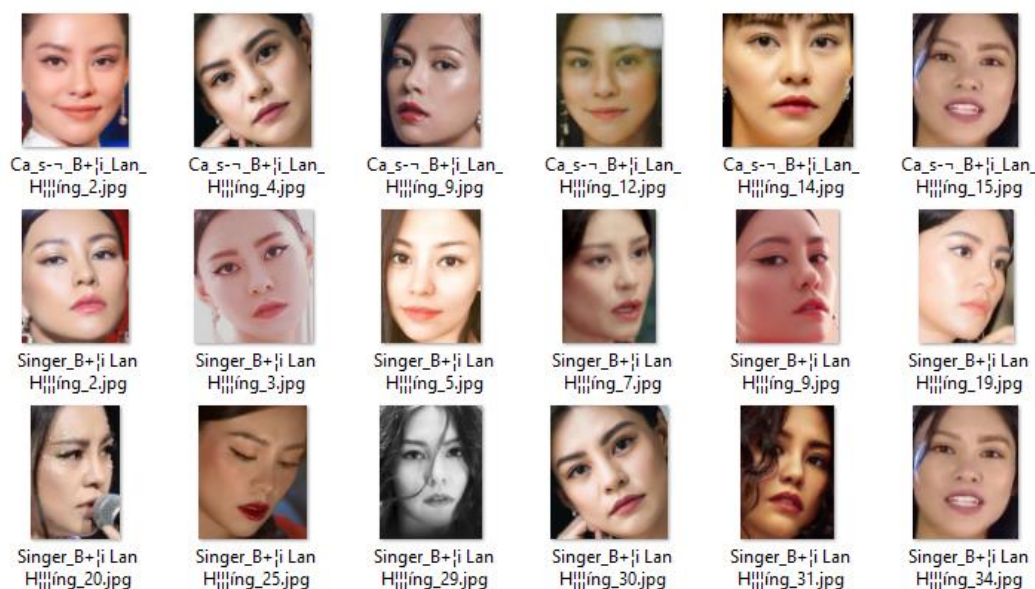
3.6.2 Tập dữ liệu nhận dạng danh tính

Trong quá trình nghiên cứu và thu thập dữ liệu thì để ứng dụng và tăng độ khách quan cho từng trường hợp thì tập dữ liệu nhận dạng danh tính sẽ chia ra làm tập dữ liệu thực tế và dữ liệu người nổi tiếng. Dữ liệu thực tế mang tính thực tiễn gần sát với ứng dụng vào thực tế để chăm công. Tập dữ liệu người nổi tiếng mang tính khách quan và đa dạng nhằm kiểm chứng các phương pháp nghiên cứu

Ngoài ra trong mỗi tập dữ liệu thực tế và dữ liệu người nổi tiếng còn có các tập dữ liệu nhỏ của người đeo khẩu trang và không đeo khẩu trang để giải quyết vấn đề nhận dạng danh tính khi khuôn mặt đó vẫn đeo khẩu trang



Hình 3.7 Dữ liệu tập thực tế không đeo khẩu trang



Hình 3.8 Dữ liệu tập người nổi tiếng không đeo khẩu trang

Về các tập dữ liệu đeo khẩu trang đều được sinh học dựa trên tập dữ liệu không đeo khẩu trang. Chung quy lại, có tất cả bốn tập dữ liệu hai tập dữ liệu không đeo khẩu trang được minh họa ở Hình 3.7 và 3.8 và hai tập dữ liệu đeo khẩu trang. Cách thức sinh học dữ liệu đeo khẩu trang và minh họa dữ liệu được trình bày ở mục 4.3.1.3

3.7 Phạm vi giải quyết vấn đề

Mô hình tổng thể có thể nhận dạng khuôn mặt kể cả khi người đó có đeo khẩu trang, ứng dụng vào trong các hệ thống chăm công hoặc các hệ thống giám sát phát hiện người đeo không đeo khẩu trang để nhắc nhở

Trong phạm vi nghiên cứu nhận dạng khuôn mặt mô hình có thể giải quyết vấn đề tùy vào phương pháp phân loại danh tính của khuôn mặt được lựa chọn như:

- Phát hiện được cả khuôn mặt đeo khẩu trang đúng và đeo khẩu trang không đúng
- Nhận dạng được danh tính ngay cả khi khuôn mặt đeo khẩu trang
- Nhận biết được người lạ không có trong tập dữ liệu
- Không cần huấn luyện lại mô hình đối với phương pháp phân loại áp dụng learning similarity
- Dễ dàng thêm dữ liệu mới vào tập dữ liệu với phương pháp phân loại áp dụng learning similarity
- Khả năng phân loại chính xác cao với 100 đối tượng ở phương pháp phân loại áp dụng neural network

3.8 Tổng kết chương

Chương 3 đã trình bày các hướng nghiên cứu cũng như tổng thể từng mô hình nhỏ trong bài toán hoạt động, giới thiệu được khả năng và phạm vi đề tài có thể giải quyết được. Các kết quả và thông tin chi tiết hơn từng khâu thực hiện được trình bày rõ hơn ở chương 4

CHƯƠNG 4 KẾT QUẢ ĐÁNH GIÁ MÔ HÌNH

4.1 Giới thiệu chương

Trong chương 4 các kết quả thu được trong quy trình thực hiện các đề xuất ở chương 3 sẽ được trình bày. Các kết quả thu được như việc thực hiện đánh nhãn dữ liệu, độ chính xác của từng mô hình phát hiện và nhận dạng khuôn mặt, confusion matrix của mô hình phát hiện khuôn mặt và kết quả khi được đưa vào ứng dụng thực tế

4.2 Mô hình phát hiện và phân loại khuôn mặt

4.2.1 Quá trình huấn luyện mô hình lần 1

4.2.1.1 Quy trình gán nhãn dữ liệu

Lượng dữ liệu sử dụng để phục vụ quá trình xây dựng mô hình là 4.13 GB, với 10.265 bức ảnh được tổng hợp từ các nguồn khác nhau như github, kaggle, google,... Lượng dữ liệu bao gồm 3 đối tượng và được chia ra làm tập 3 tập dữ liệu huấn luyện, đánh giá và kiểm thử. Số lượng dữ liệu được chia cho các tập dữ liệu như sau:

- 8790 ảnh cho huấn luyện
- 390 ảnh cho đánh giá
- 1085 ảnh cho việc kiểm thử

Trong mô hình, tên và số thứ tự của 3 đối tượng được gán nhãn như sau:

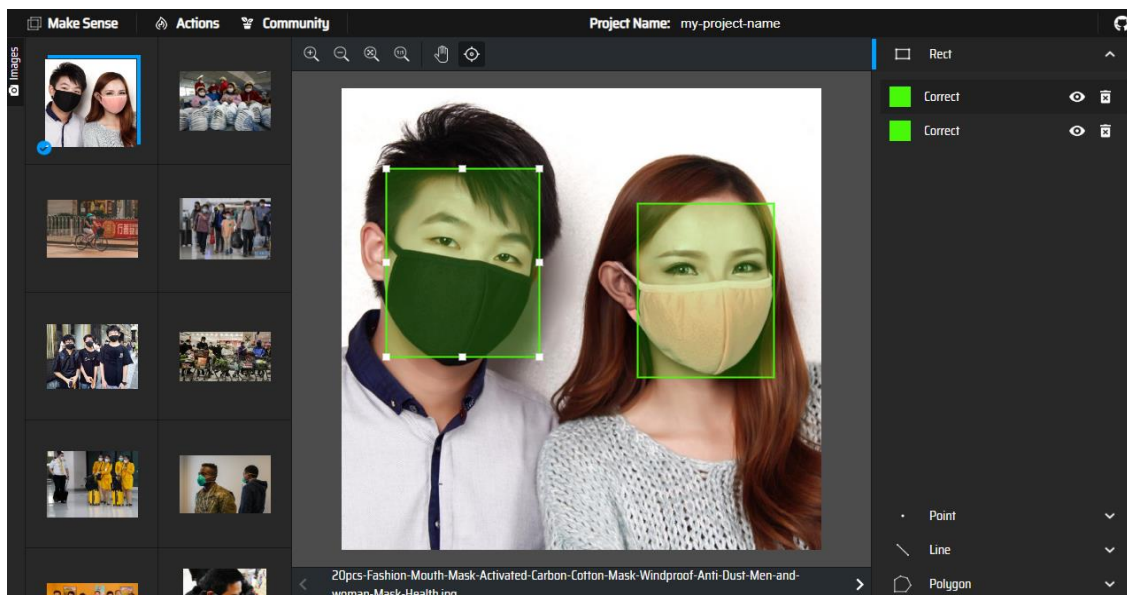
- Đối tượng không đeo khẩu trang với số thự tự là 0
- Đối tượng đeo khẩu trang đúng với số thự tự là 1
- Đối tượng đeo khẩu trang sai với số thự tự là 2

YOLOv5 được tổ chức file lưu trữ dữ liệu theo dạng tệp ảnh và tệp đuôi txt chứa các tọa độ của đối tượng như sau:

```
▼ Dataset
  ▼ images
    ▶ test
    ▶ train
    ▶ valid
  ▼ labels
    ▶ test
    ▶ train
    ▶ valid
  <> CheckData.py
```

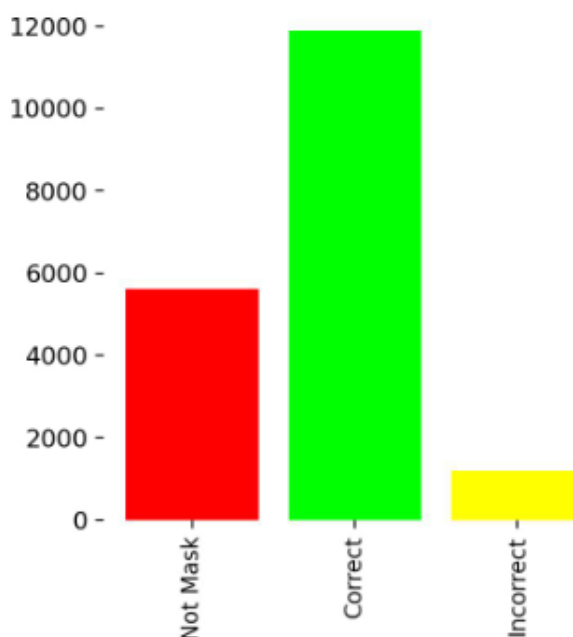

Trong đó yêu cầu tên ảnh và tên của nhãn cho ảnh đó phải giống nhau và được phân chia trong cùng thư mục như nhau. Ví dụ ảnh ‘Face 1.jpg’ ở trong images/test thì ở thư mục labels/test sẽ có file ‘Face 1.txt’ tương ứng để đánh dấu tọa độ các hộp chứa khuôn mặt.

Để giảm tải công việc gán nhãn sẽ áp dụng công hỗ trợ ở trên Makesense



Hình 4.1 Minh họa công cụ hỗ trợ gán nhãn dữ liệu

Sau khi đánh nhãn cho toàn bộ dữ liệu, dữ liệu sẽ được tải lại xuống máy cá nhân và đẩy lại lên Kaggle để tiến hành quá trình huấn luyện mô hình. Thống kê sơ bộ số lượng số mẫu đối tượng được gán nhãn lần 1 ở Hình 4.2



Hình 4.2 Biểu đồ số lượng các đối tượng sau khi được gán nhãn lần 1

Trong đó not mask, correct và incorrect lần lượt là các đối tượng không đeo khẩu trang, đeo khẩu trang đúng và đeo khẩu trang sai. Ở các phần sau, các từ viết tắt này cũng được áp dụng để đơn giản hoá cách gọi tên các đối tượng

Quá trình thực hiện huấn luyện mô hình được thực hiện trên Kaggle là trang web hỗ trợ GPU cho quá trình huấn luyện, thư viện hỗ trợ là Pytorch trong ngôn ngữ lập trình Python. Với việc áp dụng mô hình sẵn có của YOLOv5

Ngoài việc sử dụng các thông số mặc định của mô hình YOLOv5 thì có sự thay đổi của các tham số sau:

- Kích thước ảnh resize về chung 1 kích thước là 640x640
- Batch size sử dụng là 64
- Số epochs áp dụng khi huấn luyện là 50 epoch
- Mô hình phát hiện khuôn mặt sử dụng tiền trọng số và kiến trúc CNN của mô hình YOLO5s
- Đường dẫn dữ liệu và thay đổi lại số đối tượng để huấn luyện đúng yêu cầu của bài toán

Thông số thay đổi trong file quy định các đối tượng như sau:

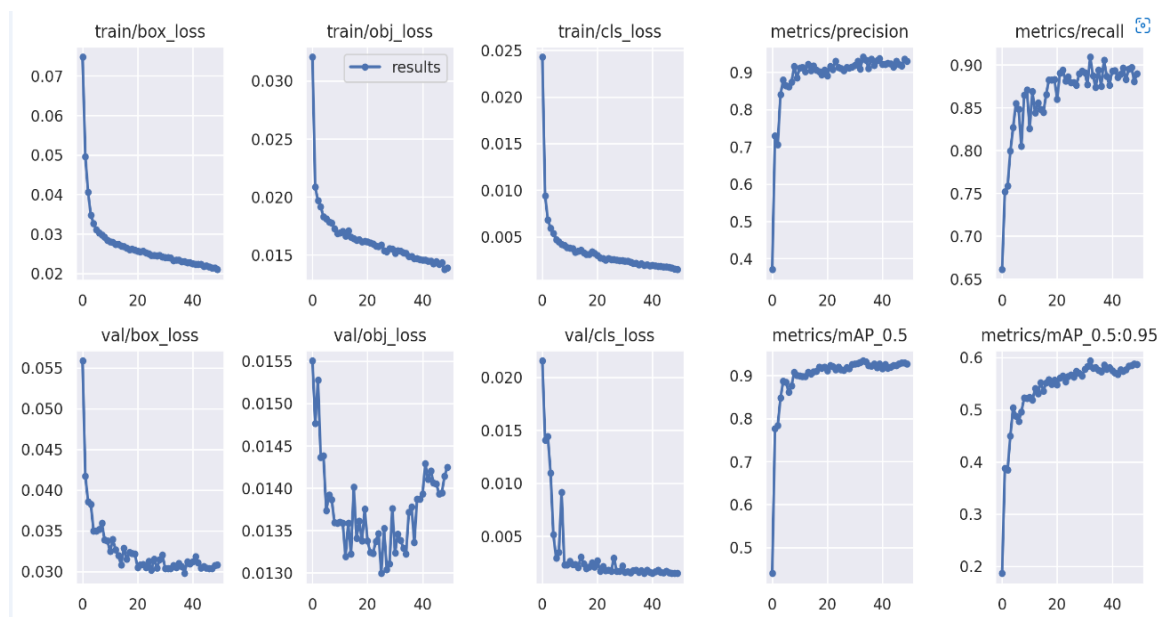
```
path: ../../input/facemaskYOLOv2/Dữ liệuset
train: images/train
val: images/valid
test: images/test
# Classes
nc: 3 # number of classes
names: ['Not Mask', 'Correct', 'Incorrect'] # class names
```

Trong đó path là đường dẫn thư mục gốc được lưu trên Kaggle, train là đường dẫn dữ liệu và thư mục chứa tệp đã được đánh nhãn cho tập dữ liệu dùng để huấn luyện, tương tự với 2 val và test là cho 2 tệp đánh giá và kiểm thử. Nc là số đối tượng có ở mô hình phát hiện là 3 đối tượng và names là tên lần lượt của đối tượng được xếp đúng theo vị trí

4.2.1.2 Đánh giá kết quả lần 1

Kết quả của các chỉ số trong hàm mất mát của 2 tập huấn luyện và đánh giá sau khi thực hiện huấn luyện lần đầu được thể hiện ở Hình 4.3. Ngoài ra các chỉ số mAP, precision và recall cũng được hiển thị. Các giá trị precision và recall nằm trong khoảng từ 0.85 tới 0.95 chứng tỏ mô hình hoạt động tốt. Chỉ số mAP_0.5 đạt tới giá trị khoảng 0.95 sau 50 epochs nhưng để tăng độ tin cậy thì ngưỡng IoU thường lớn hơn 0.5 nên ở

mAP_0.5:0.95 chỉ số nằm khoảng 0.6 chứng tỏ với các ngưỡng IoU tăng dần thì mô hình hoạt động kém hiệu quả nhanh chóng.



Hình 4.3 Biểu đồ trên là các hàm mất mát và thang đo trong mô hình phát hiện khuôn mặt

Trong YOLO ta có 3 hàm loss khác nhau như sau

- *Classification loss*: chỉ tính cho những ô vuông được đánh nhãn là có đối tượng. Classification loss tại những ô vuông đó được tính bằng độ lỗi bình phương giữa nhãn được dự đoán và nhãn đúng của nó. Tương đương là cls_loss trong biểu đồ trên
- *Localization loss*: dùng để tính giá trị lỗi cho hộp giới hạn được dự đoán bao gồm offset x,y và chiều dài, rộng so với nhãn chính xác. Tương đương là box_loss
- *Confidence loss*: thể hiện độ lỗi giữa dự đoán hộp giới hạn đó chứa đối tượng so với nhãn thực tế tại ô vuông đó. Độ lỗi này tính nên cả những ô vuông chứa đối tượng và không chứa đối tượng. Tương đương là obj_loss

Các giá trị của từng hàm mất mát trong tập đánh giá và huấn luyện đều đang có xu hướng tiếp tục giảm chứng tỏ mô hình có thể train tiếp để tăng các thang đo. Nhưng hàm confidence loss của tập đánh giá có xu hướng tăng. Dự đoán do lượng dữ liệu chia đánh giá ít hoặc do các hộp giới hạn không đồng nhất do có một phần dữ liệu đã được đánh gán nhãn sẵn.

Kết quả ma trận nhầm lẫn được trình bày ở Hình 4.4 và dựa trên tập kiểm thử. Với độ chính xác xấp xỉ 92 % trong đó, đối tượng đeo khẩu trang có tỉ lệ chính xác thấp nhất với 88% trong khi đối tượng đeo khẩu trang không đúng với tỉ lệ chính xác 99%. Nhưng trong thực tế khi demo thì khả năng phát hiện đối tượng chính xác thấp nhất là đối tượng đeo khẩu trang không đúng và đối tượng không đeo khẩu trang thì khả năng chính xác cao hơn.

		Thực tế			
		Not Mask	Correct	Incorrect	BG FP
Dự đoán	Not Mask	0.88	0	0	0.28
	Correct	0.02	0.96	0	0.7
	Incorrect	0	0	0.99	0.02
	BG FN	0.09	0.09	0	0

Hình 4.4 Ma trận nhầm lẫn của huấn luyện lần 1

Nhận định trong quá trình học đối tượng không đeo khẩu trang có sự đa dạng về dữ liệu nên khả năng khi vào thực tế mô hình có thể dự đoán chính xác cao với từng đối tượng khác nhau. Để tránh trường hợp mất cân bằng dữ liệu thì đối tượng đeo khẩu trang không đúng được tăng cường lượng dữ liệu lên nhưng dữ liệu vẫn còn quá ít và còn bị nhiễu nhiễu

Vào thực tế khả năng nhận nhầm đối tượng đeo khẩu trang không đúng đối tượng đeo khẩu trang đúng vẫn có và thường chỉ gặp ở các trường hợp đeo khẩu trang hở mũi.

4.2.2 Quá trình huấn luyện mô hình lần 2

4.2.2.1 Gán nhãn dữ liệu bán tự động

Do các chỉ số trong mô hình phát hiện khuôn mặt chưa cao và gặp lỗi trong quá trình hiển thị kết quả của đối tượng đeo khẩu trang không đúng nên số tập dữ liệu sẽ được gán nhãn dữ liệu lại bằng chính mô hình phát hiện khuôn mặt. Toàn bộ hộp giới hạn và số thực tự của đối tượng sẽ được dự đoán bằng mô hình phát hiện khuôn mặt đã được huấn luyện và tiến hành chú thích lại theo định dạng của YOLO. Toàn bộ chú thích này sẽ tải lên lại công cụ CVAT để tiến hành căn chỉnh lại hộp giới hạn và gán nhãn lại số thực tự của đối tượng.

Lượng dữ liệu của đối tượng đeo khẩu trang không đúng sẽ được tăng cường dữ liệu để giảm thiểu sự tác động của mất cân bằng dữ liệu

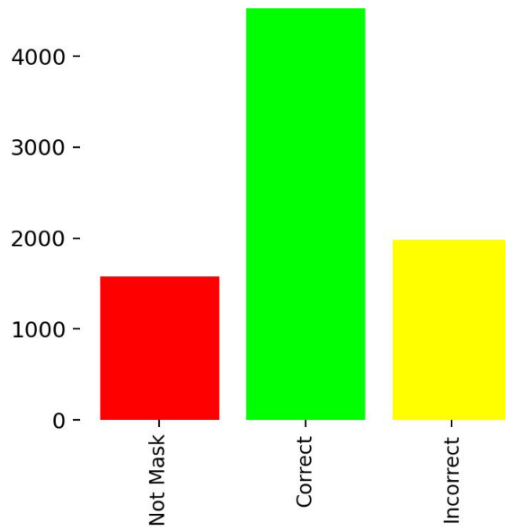
4.2.2.2 Thay đổi các tham số

Các thông số huấn luyện ở lần hai sẽ được giữ nguyên như lần đầu nhưng sẽ tăng số epoch lên thành 100 và thay đổi optimizer từ SGD thành AdamW để hàm mất được cải thiện khả năng hội tụ hơn

Sau khi thay đổi về tập dữ liệu và các tham số mới sẽ tiến hành huấn luyện lần 2 dựa trên bộ tiền trọng số của YOLO5s với lượng dữ liệu khoảng 6200 hình ảnh dùng để huấn luyện và đánh giá. Số lượng dữ liệu được chia cho các tập dữ liệu như sau:

- 5200 ảnh cho huấn luyện
- 1000 ảnh cho đánh giá

Thống kê sơ bộ số lượng số mẫu đối tượng được gán nhãn lần 2 ở Hình 4.5



Hình 4.5 Biểu đồ số lượng các đối tượng sau khi được gán nhãn lần 2

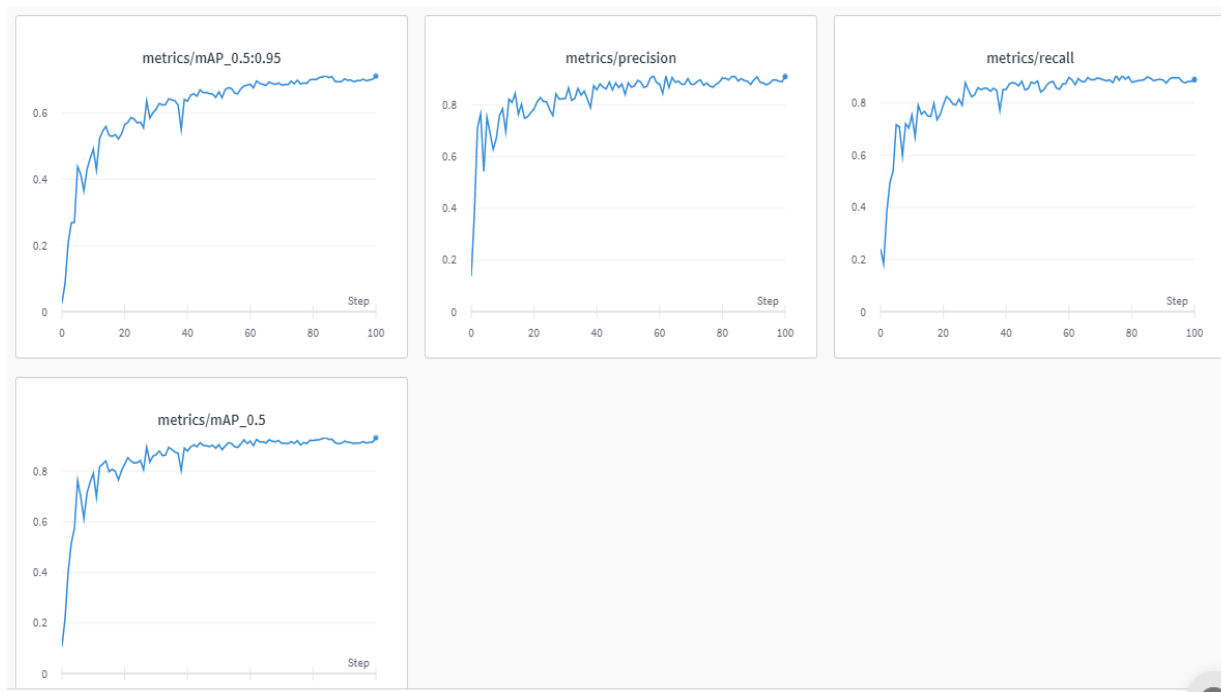
4.2.2.3 Đánh giá kết quả lần 2

		Thực tế			
		Not Mask	Correct	Incorrect	BG FP
Dự đoán	Not Mask	0.97	0.01	0.05	0.3
	Correct	0.01	0.97	0.09	0.61
	Incorrect	0.01	0.01	0.8	0.09
	BG FN	0.01	0.01	0.05	0

Hình 4.6 Kết quả ma trận nhầm lẫn của huấn luyện lần 2

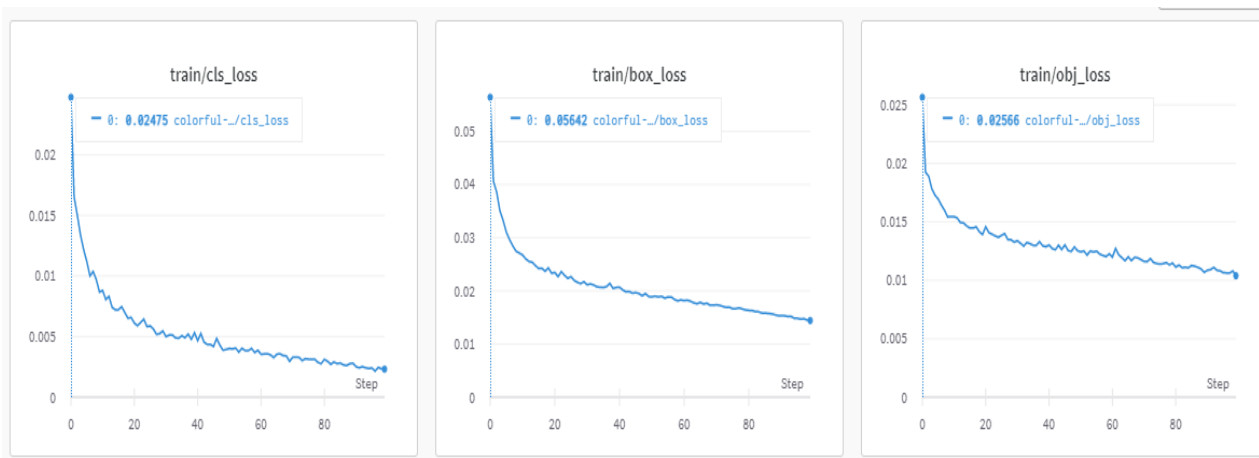
Việc dự đoán đúng 2 đối tượng đeo khẩu trang đúng và không đeo khẩu trang gần như là tuyệt đối khi chỉ số recall đạt tới 97%. Nhưng đối tượng đeo khẩu trang không đúng chỉ dự đoán được 80% đối tượng mang ra đánh giá. Việc đánh giá giữa hai mô

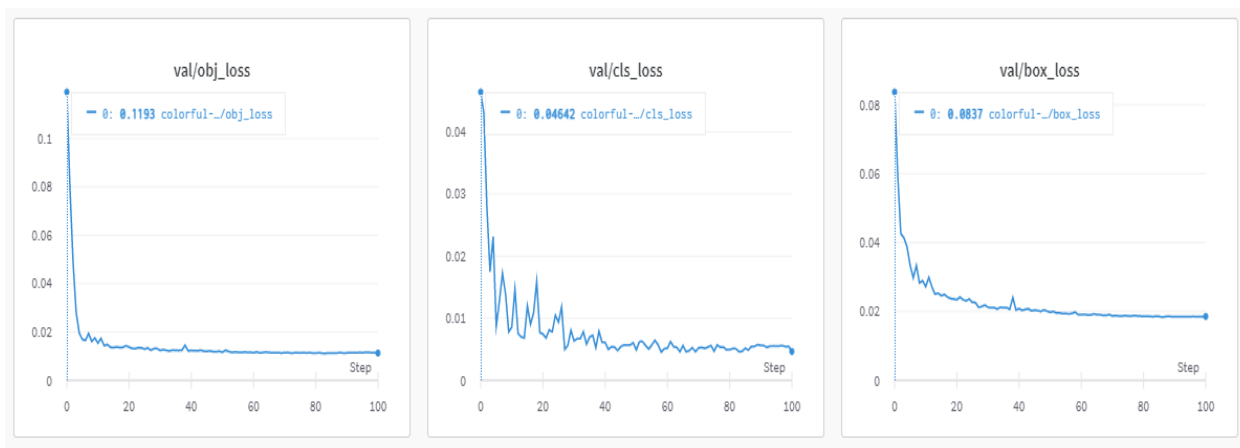
hình huấn luyện lần 1 và lần 2 thì chỉ số lần 2 kém hiệu quả hơn nhưng khi vào bài toán thực tế việc gặp lỗi ở đối tượng đeo khẩu trang không đúng thì không còn nữa. Chứng tỏ việc tăng cường dữ liệu đối tượng đeo khẩu trang không đúng đã cải thiện kết quả nhưng sự đa dạng trong dữ liệu còn kém nên mô hình phát hiện và phân loại được đối tượng này kém hơn so với các đối tượng khác.



Hình 4.7 Kết quả của các thang đo recall, precision, mAP 0.5 và mAP 0.5:0.95

Kết quả của mô hình đã cải thiện hơn so với huấn luyện lần 1. Đặc biệt chỉ số mAP 0.5:0.95 đã tăng từ 0.59 tới 0.71. Chỉ số AP 0.5 tới 0.95 của hai đối tượng là không đeo khẩu trang và đeo khẩu trang đúng có xu thế cải thiện tốt khi lần lượt là 0.73 và 0.76, trong khi đối tượng đeo khẩu trang sai cách do vẫn thiếu sự đa dạng dữ liệu nên khi vào tập đánh giá kết quả không được tốt như 2 đối tượng còn lại khi chỉ số chỉ là 0.64





Hình 4.8 Kết quả của hàm mất mát trong tập huấn luyện và đánh giá

Dựa vào đồ thị của hàm mất mát trên tập đánh giá không thấy hiện tượng overfitting nhưng có xu hướng đi ngang sau khoảng 30 epochs đầu nên mô hình khó có cải thiện bằng việc tăng số lượng epoch để huấn luyện. Hướng cải thiện mô hình hoặc tăng cường số dữ liệu huấn luyện mới có khả năng cải thiện tốt các chỉ số thang đo như mAP 0.5:0.95

4.2.3 Tổng kết kết quả

Bảng 1 Bảng so sánh các thang đo giữa hai lần huấn luyện

Huấn luyện	Accuracy	Recall	Precision	mAP 0.5	mAP 0.5:0.95
Lần 1	94,33%	0.88	0.935	0.93	0.58
Lần 2	91,33%	0.9	0.9	0.93	0.71

Sau khi so sánh và nhận xét thì việc huấn luyện lần 2 vượt trội hơn lần 1 ở các thang đo quan trọng như mAP 0.5:0.95, dù các chỉ số còn lại của 2 lần huấn luyện như nhau. Đặc biệt việc huấn luyện lần 2 khắc phục được dự đoán đối tượng đeo khẩu trang sai cách của huấn luyện lần 1.

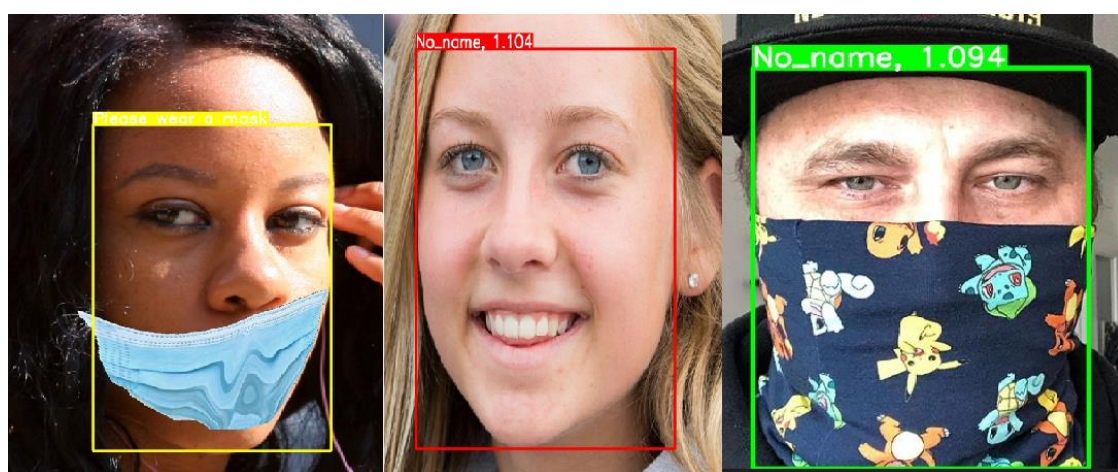
Lý do chính để lần 2 tốt hơn lần 1 dù lần 2 huấn luyện với ít dữ liệu hơn:

- Dữ liệu đã được đồng nhất về hộp giới hạn, minh họa ở Hình 4.9 và Hình 4.10
- Giảm thiểu tối đa của việc ảnh hưởng mất cân bằng dữ liệu, so sánh giữa Hình 4.2 và Hình 4.5

Các mô hình phát hiện và phân loại ở các phần sau đều sẽ áp dụng kết quả của mô hình huấn luyện lần 2 để hỗ trợ cho mô hình nhận diện danh tính và mô hình tổng thể sau khi huấn luyện sau cùng cũng sẽ áp dụng mô hình phát hiện huấn luyện lần 2 để phát hiện khuôn mặt



Hình 4.9 Hộp giới hạn không đồng nhất ở huấn luyện lần 1



Hình 4.10 Hộp giới hạn đồng nhất ở huấn luyện lần 2

4.3 Mô hình nhận dạng khuôn mặt

Bài toán quy về nhận dạng khuôn mặt đã được phát hiện từ mô hình phát hiện khuôn mặt, với việc nhận dạng khuôn mặt đó là của ai trong tập dữ liệu nhận dạng. Việc phát hiện khuôn mặt ra khỏi khuôn hình, phân loại khuôn mặt sẽ áp YOLO và thực hiện nhận dạng bằng các embedding của mô hình FaceNet.

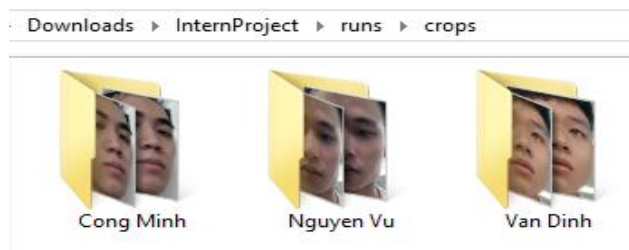
4.3.1 Quy trình gán nhãn dữ liệu

Sau khi thu thập dữ liệu, dữ liệu sẽ được chia ra thành tập dữ liệu thực tế và người nổi tiếng. Hai tập dữ liệu tùy vào phương pháp phân loại mà được chia nhỏ ra. Chia thành mã hoá và kiểm thử đối với phân loại áp dụng learning similarity và chia thành huấn luyện, đánh giá và kiểm thử đối với phương pháp áp dụng neural network. Song song với hai tập dữ liệu thực tế và người nổi tiếng sẽ có thêm hai tập dữ liệu thực tế và người nổi tiếng đeo khẩu trang nhằm nhận dạng danh tính khi người đó đeo khẩu trang

4.3.1.1 Tập dữ liệu thực tế

Chuẩn bị dữ liệu bằng cách quay 1 video chứa các góc cạnh khuôn mặt của đối tượng. Sau đó sẽ thực hiện lấy từng khung hình và lưu lại làm tập dữ liệu. Lượng dữ liệu được chia làm 80% cho huấn luyện hoặc mã hoá và 20% để đánh giá kết quả

Do dữ liệu cần được phát hiện khuôn mặt nên áp dụng mô hình phát hiện khuôn mặt đã huấn luyện để phát hiện. Việc gán nhãn dữ liệu sẽ thực hiện tự động bằng cách gán tên cho từng tập dữ liệu là tên của đối tượng đó



Hình 4.11 Minh họa việc gán nhãn tự động

Lượng dữ liệu có khoảng hơn 30 đối tượng, mỗi đối tượng có khoảng 50 ảnh và tiến hành đánh giá kết quả phân loại dựa trên 2 cách là learning similarity và neural network

4.3.1.2 Tập dữ liệu người nổi tiếng

Do quá trình đánh giá trên dữ liệu thực tế sẽ không khách quan khi cả tập huấn luyện và kiểm thử sử dụng trên cùng 1 video và khi kiểm thử sẽ đưa vào thực tế để kiểm chứng. Nên quy trình tiến hành tạo ra 1 tập dữ liệu khác khách quan hơn dựa trên việc cào dữ liệu của 100 người nổi tiếng Việt Nam ở các nguồn trên internet.

Lượng dữ liệu sẽ được tự động gán nhãn dựa trên tệp cào về. Sau đó tiến hành bằng mắt để kiểm chứng kết quả. Số lượng hình ảnh của đối tượng sẽ không đồng đều khi một vài đối tượng sẽ có khoảng 6 ảnh trong khi đối tượng khác sẽ có khoảng hơn 20 ảnh.

4.3.1.3 Tập dữ liệu người đeo khẩu trang

Để giải quyết bài toán nhận dạng khi người đó đeo khẩu trang thì cần tạo luồng dữ liệu của người đeo khẩu trang. Việc tạo khuôn mặt đeo khẩu trang sẽ được tạo bằng cách sinh học dựa trên lượng dữ liệu của khuôn mặt không đeo khẩu trang

Việc áp dụng đeo khẩu trang sinh học bằng cách dựa trên thư viện OpenCV để hỗ trợ lấy các điểm Landmark. Dựa vào các điểm Landmark để biết khuôn mặt đang xoay trái, xoay phải hay chính dạng để áp dụng các hướng khẩu trang cho hợp lí

Quá trình thực hiện được hỗ trợ bởi nhóm tác giả MaskTheFace. Ngoài ra nhóm tác giả hỗ trợ việc áp dụng nhiều loại khẩu trang khác nhau và màu sắc, độ sáng cũng có thể thay đổi để hợp hơn với yêu cầu bài toán

Trong thực tế khẩu trang số 1 và 3 từ trái sang ở Hình 4.12 được áp dụng nhiều nhất nên lượng dữ liệu sẽ được thêm vào hai loại khẩu trang này để phục vụ huấn luyện sau này

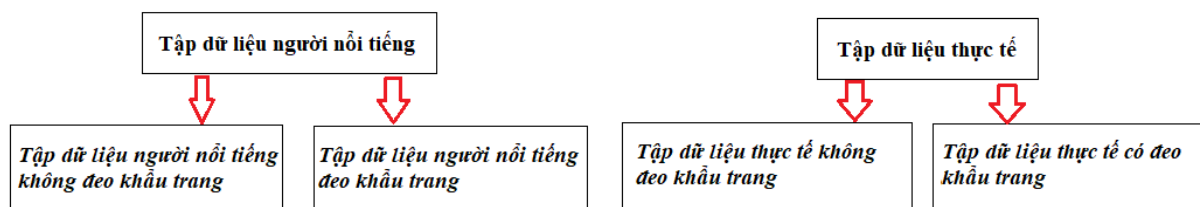


Hình 4.12 Các loại khẩu trang có sẵn



Hình 4.13 Minh họa khẩu trang được đeo theo dạng sinh học

Tập dữ liệu người đeo khẩu trang cũng được chia làm 2 phần nhỏ cho tập dữ liệu thực tế và dữ liệu người nổi tiếng nên có tất cả 4 tập dữ liệu được đem đi huấn luyện



Hình 4.14 Minh họa phân cấp các tập dữ liệu

4.3.2 Huấn luyện mô hình

Quá trình xây dựng mô hình áp dụng sẵn mô hình đã được huấn luyện trong Pytorch.

Trong mô hình đã được huấn luyện sử dụng kiến trúc InceptionResnetV1 và có thể sử dụng một trong hai mô hình có sẵn là 'vggface2' với khoảng 8631 đối tượng đầu ra và 'casia-webface' với khoảng 10575 đối tượng đầu ra. Tất cả dữ liệu là các khuôn mặt đã được phát hiện để tiến hành mã hoá thành các vector với 512 giá trị, được định dạng là Tensor.

Tất cả các phương pháp thử nghiệm đều áp dụng mô hình có sẵn vggface2 để mã hoá khuôn mặt thành các embedding. Lý do là vì để thống nhất và dễ dàng cho công việc so sánh khi đánh giá mô hình

Để thực hiện quá trình sau này không cần mã hoá lại các khuôn mặt đã được phát hiện thì tất cả vector đã được mã hoá sẽ lưu dưới dạng mảng gồm các vector và tên đối tượng tương ứng của vector đó.

4.3.2.1 Phân loại áp dụng learning similarity

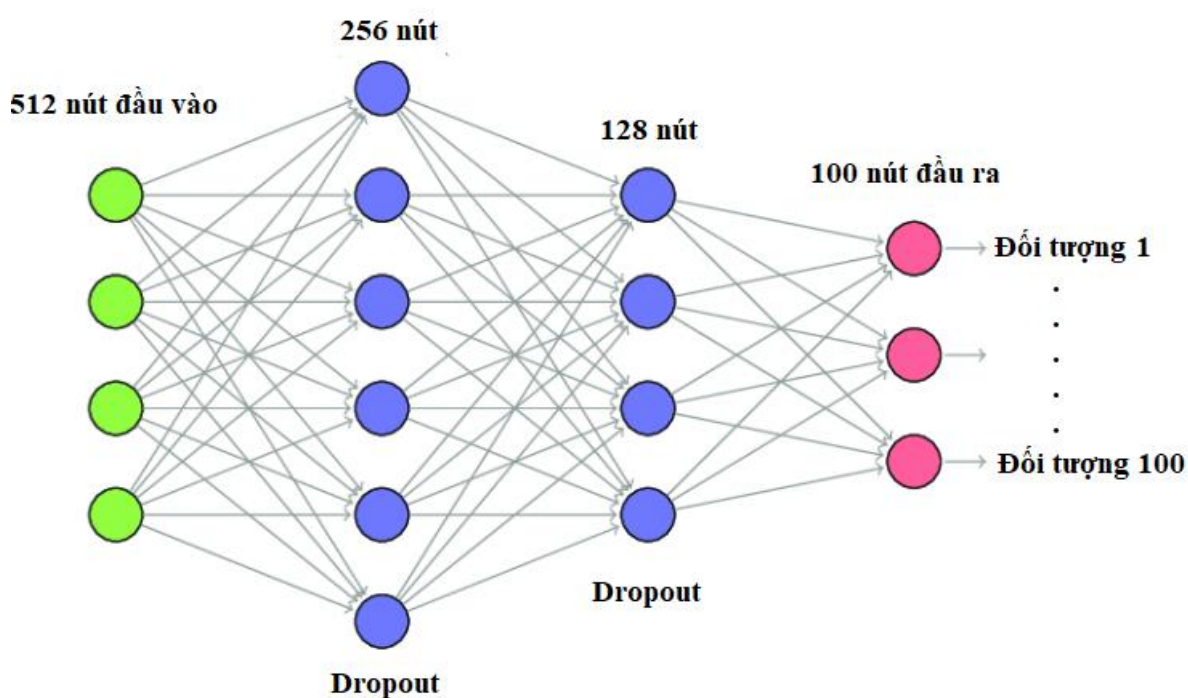
Trong learning similarity thì khâu huấn luyện mô hình thường không thực hiện hoặc có thể quy về là công thực hiện mã hoá dữ liệu thành 1 tệp lưu trữ để phục vụ kiểm thử sau này. Learning similarity tương tự như KNN và dựa trên khoảng cách euclid 2 để tính khoảng cách. K thường được chọn là bằng 1. Ngưỡng để phân loại sẽ khảo sát và tìm ra kết quả tối ưu nhất

Nhìn chung tham số cần tìm trong phương pháp learning similarity là ngưỡng riêng cho từng tập dữ liệu đeo khẩu trang và không đeo khẩu trang.

4.3.2.2 Phân loại áp dụng neural network

Áp dụng neural network để phân loại thì kiến trúc sẽ áp dụng đơn giản gồm 1 input layer, 2 hidden layer và 1 output layer. Hàm kích hoạt ReLU và thêm Drop Out được áp dụng để cải thiện kết quả đầu ra

Đầu vào của neural network là các vector đã được mã hoá bằng mô hình FaceNet tức sẽ có 512 nút đầu vào. Số nút đầu ra sẽ phụ thuộc vào số người cần nhận dạng trong tập dữ liệu (31 đối với tập dữ liệu thực tế và 100 đối với dữ liệu người nổi tiếng)



Hình 4.15 Thông số chi tiết model neural network cho tập dữ liệu người nổi tiếng

Quá trình huấn luyện thì hàm mất mát được chọn là CrossEntropy, optimizer sẽ áp dụng khảo sát đồng thời trên cả SGD, Adam và AdamW, learning rate sẽ thay đổi trong khoảng từ 0.0001 tới 0.01

Ngoài ra các tham số khác như epoch và batch size thì tùy vào tình trạng hàm mất mát của quá trình huấn luyện và thông số của GPU huấn luyện để linh động đưa ra quyết định

Toàn bộ dữ liệu và code được gửi lên công cụ Kaggle để phục vụ quá trình huấn luyện. Ngoài ra áp dụng thêm công cụ Wandb để dễ dàng quan sát thêm dự thay đổi của hàm mất mát, độ chính xác của tập huấn luyện, đánh giá và kiểm thử trên từng epoch

4.3.3 Kết quả thu được

4.3.3.1 Kết quả trên tập dữ liệu thực tế

Phương pháp learning similarity

Kết quả khi thử trên tập đánh giá thường có độ chính xác tuyệt đối với tập dữ liệu người không đeo khẩu trang. Và giảm sâu đối với lượng người không đeo khẩu trang. Đối với hiện tượng trên do tất cả dữ liệu đều trích từ 1 video nên tập huấn luyện và đánh giá sẽ không khách quan

Bảng 2 Đánh giá kết quả trên tập dữ liệu thực tế phương pháp learning similarity

	Learning similarity			
	Threshold=0.6	Threshold=0.7	Threshold=0.8	Threshold=0.9
Mask Data	99,56%	99,56%	99,56%	99,56%
Not Mask Data	99,56%	99,56%	99,56%	99,56%

Phương pháp neural network

Tương tự learning similarity thì neural network sau khi huấn luyện kết quả cũng gần đạt ngưỡng tuyệt đối. Tập dữ liệu đeo khẩu trang cũng có khả năng giảm sâu sau khi huấn luyện.

Bảng 3 Đánh giá kết quả trên tập dữ liệu thực tế phương pháp neural network

	Fully Connected (Last Layer: L2)		
Mask Data	80,73%(AdamW, lr=0.01, ep=500)	82,16%(AdamW, lr=0.005, ep=40)	
Not Mask Data	98,41% (Adam, lr=0.01, ep=300)	97,12% (SGD, lr=0.01, ep=300)	

Có augmentation data

Không augmentation data

Nhận dạng khuôn mặt ứng dụng trong chấm công



Hình 4.16 Thông số huấn luyện tốt nhất của tập dữ liệu không đeo khẩu trang và chưa có tăng cường dữ liệu



Hình 4.17 Thông số huấn luyện tốt nhất của tập dữ liệu đeo khẩu trang và chưa có tăng cường dữ liệu

4.3.3.2 Kết quả trên tập dữ liệu người nổi tiếng

Phương pháp learning similarity

Kết quả tốt nhất khi thực hiện phương pháp learning similarity thấp hơn phương pháp neural network nhưng với ưu điểm không cần huấn luyện mô hình và dễ dàng thêm đối tượng mới vào tập dữ liệu thì phương pháp vẫn có những ưu điểm riêng. Độ tin cậy của ngưỡng càng thấp nên thực tế thường mức ngưỡng 0.6 để tạo độ tin cậy cho mô hình.

Bảng 4 Đánh giá kết quả trên dữ liệu người nổi tiếng phương pháp learning similarity

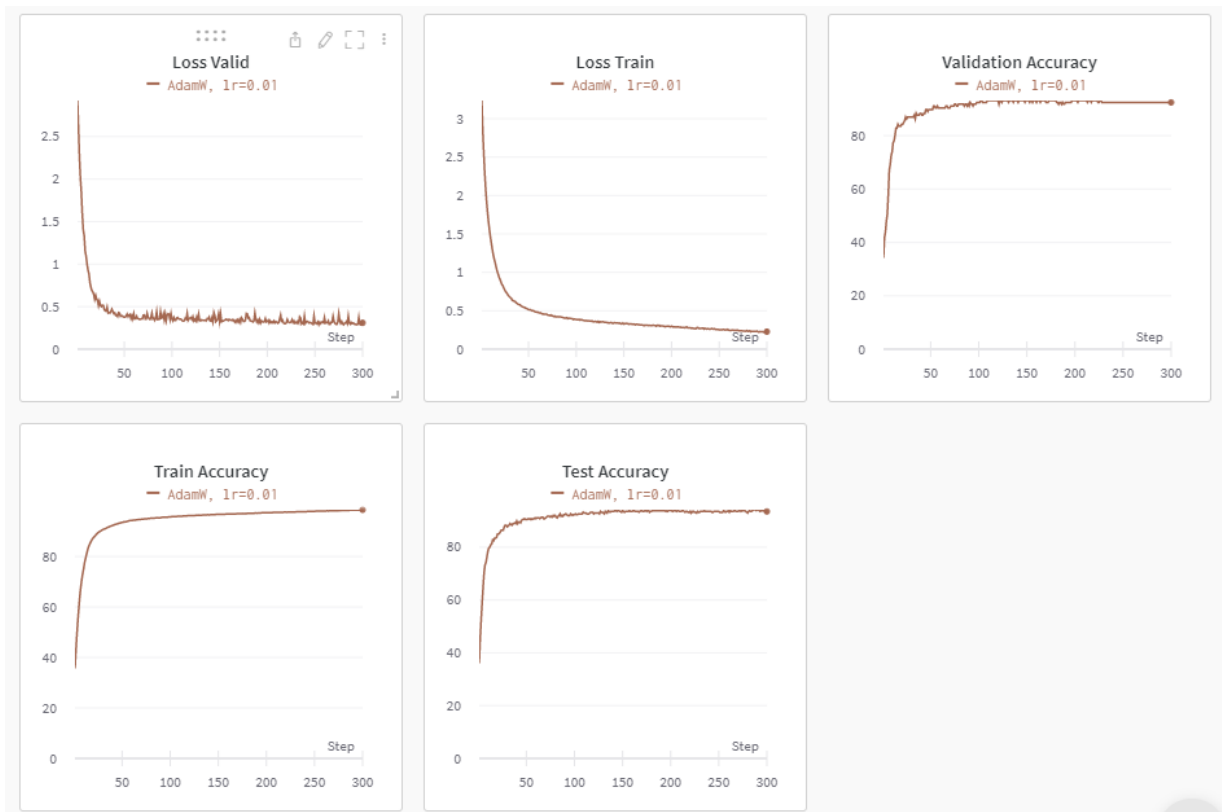
	Learning similarity			
	Threshold=0.6	Threshold=0.7	Threshold=0.8	Threshold=0.9
Mask Data	50.56%	66%	69.60%	70.20%
Not Mask Data	54.15%	81.84%	88%	89.80%

Phương pháp neural network

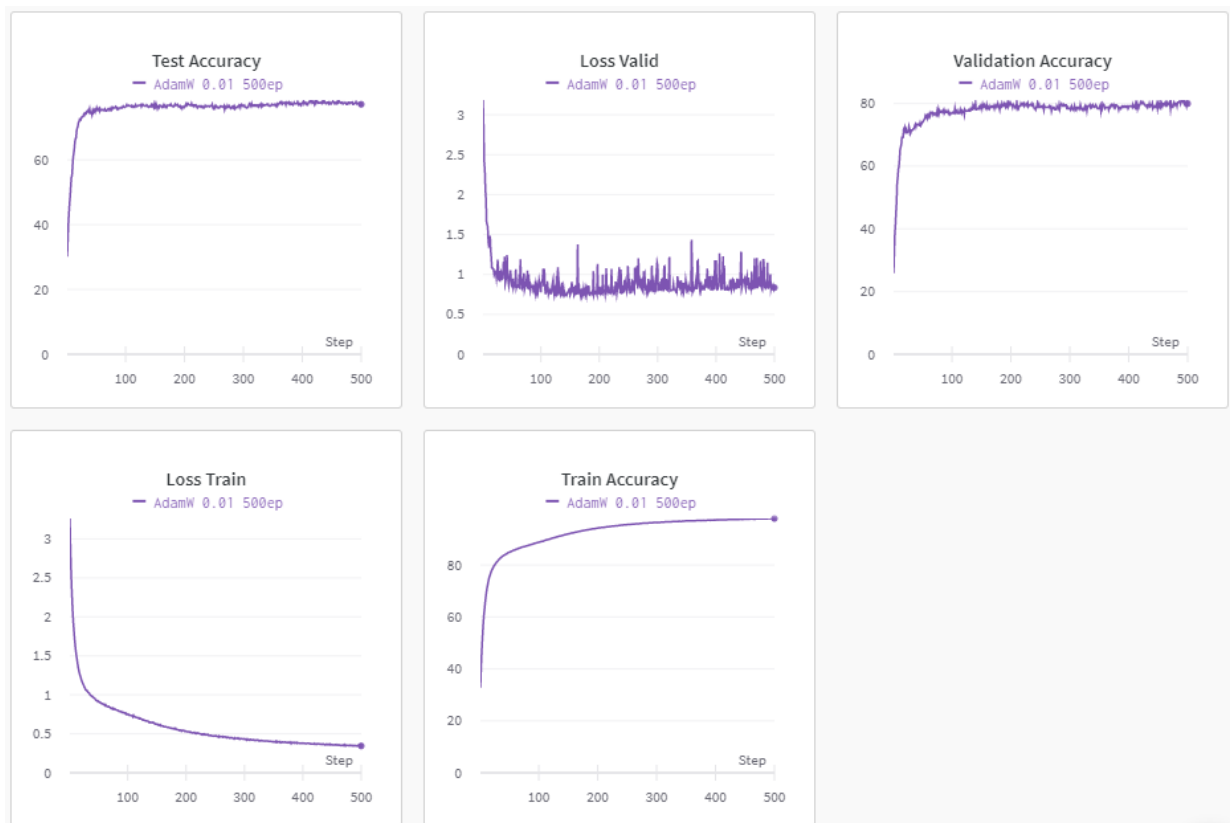
Kết quả tốt nhất sau khi thực hiện huấn luyện mô hình với 100 classes người nổi tiếng việt nam với phương pháp áp dụng là neural network. Thang đo sử dụng độ chính xác của tập dữ liệu, trong đó các kết quả thường tốt hơn ở AdamW khi việc điều chỉnh tốc độ học không cấp thiết. SGD hiệu quả không thua kém Adam và AdamW nhưng việc lựa chọn tốc độ học rất quan trọng trong hàm tối ưu hoá này.

Bảng 5 Bảng đánh giá kết quả trên dữ liệu người nổi tiếng phương pháp neural network

	Fully Connected (Last Layer: L2)	
	Có augmentation data	Không augmentation data
Mask Data	77,15%(AdamW, lr=0.01, ep=500)	75,46%(AdamW, lr=0.005, ep=40)
Not Mask Data	93,54% (Adam, lr=0.01, ep=300)	92,92% (SGD, lr=0.01, ep=300)



Hình 4.18 Thông số huấn luyện tốt nhất của tập dữ liệu không đeo khẩu trang và có tăng cường dữ liệu



Hình 4.19 Thông số huấn luyện tốt nhất của tập dữ liệu đeo khẩu trang và có tăng cường dữ liệu

4.3.4 Nhận xét kết quả

4.3.4.1 So sánh giữa 2 tập dữ liệu thực tế và người nổi tiếng

Như đã trình bày, tập dữ liệu thực tế thường có kết quả tuyệt đối với tập dữ liệu người không đeo khẩu trang và sụt giảm đối với tập người đeo khẩu trang. Trong khi tập dữ liệu người nổi tiếng cũng có bị sụt giảm giữa người không đeo khẩu trang và người đeo khẩu trang nhưng việc huấn luyện mô hình thường không đạt kết quả quá cao.

Khi áp dụng vào thực tiễn để kiểm thử, cả hai tập dữ liệu đều hoạt động được nhưng tập dữ liệu người nổi tiếng thường nhận dạng tốt hơn đối với các trường hợp dữ liệu không có trong huấn luyện. Còn bên tập dữ liệu thực tế nhận dạng tốt với các trường hợp tương đồng về dữ liệu nhưng khi các dữ liệu có sự thay đổi nhẹ thì tập tỏ ra kém hiệu quả hơn

Tùy vào phương thức và lượng dữ liệu có thể thu thập mà ta có thể linh động lựa chọn cách tạo dữ liệu giữa hai tập dữ liệu để phù hợp với việc ứng dụng

4.3.4.2 So sánh giữa 2 tập dữ liệu đeo khẩu trang và không đeo khẩu trang

Mô hình tỏ ra học kém hiệu quả hơn đối với tập dữ liệu đeo khẩu trang. Lý do là các embedding của mô hình FaceNet đều được huấn luyện trên dữ liệu người không đeo khẩu trang nên việc nhận dạng người đeo khẩu trang khó khăn hơn. Việc áp dụng các embedding để phân loại tập dữ liệu người nổi tiếng không đeo khẩu trang không đạt được kết quả cao, vì lý do một phần dữ liệu FaceNet huấn luyện đều trên người châu Âu và lượng dữ liệu người châu Á nói chung, Việt Nam nói riêng đều rất ít

Để có mô hình tổng quan hơn đòi hỏi cần đi huấn luyện cả các embedding của FaceNet trên lượng dữ liệu của người châu Á và Việt Nam. Có thể tích hợp thêm dữ liệu người đeo khẩu trang vào huấn luyện nhưng cách này không khách quan. Hướng xử lý cho người đeo khẩu trang là tách lớp tích chập cuối làm 2 nhánh giống nhau. Nhánh 1 tập trung vào phân loại người đeo khẩu trang, nhánh 2 tập trung vào phân loại danh tính người cần nhận dạng. Hàm mất mát sẽ tính bằng tổng của 2 nhánh và đánh 1 hệ số alpha vào nhánh đeo khẩu trang để giảm thiểu mô hình học quá nhiều vào nhánh 1

4.3.4.3 So sánh giữa 2 phương pháp huấn luyện

Hai phương pháp huấn luyện đều có những ưu nhược điểm riêng nên tùy vào ứng dụng thực tế ta có thể linh động chọn phương pháp huấn luyện

Ưu điểm của learning similarity là phương pháp đơn giản, không cần huấn luyện và xây dựng mô hình, khả năng phát hiện người lạ dễ dàng và việc thêm dữ liệu mới không gặp khó khăn. Nhưng gặp nhược điểm rất lớn là gặp với các trường hợp đối tượng nhận dạng danh tính càng nhiều thì không thể phân loại chính xác, tỉ lệ chính xác không cao

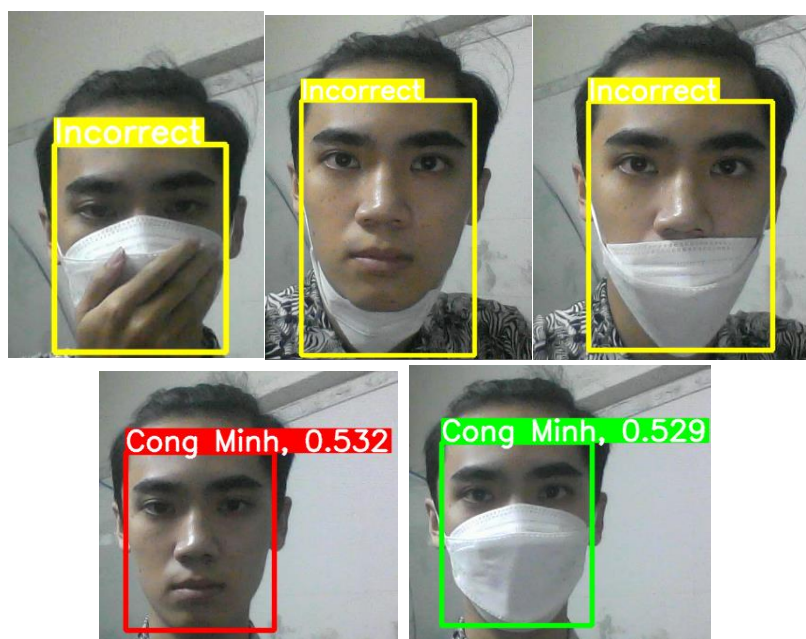
bằng các phương pháp khác và phụ thuộc rất nhiều vào mô hình embedding phía trước. Ngoài ra lượng dữ liệu càng lớn thì thời gian thực thi phương pháp càng lâu do có càng nhiều embedding cần tính toán khoảng cách với dữ liệu dự đoán

Ưu điểm của neural network là huấn luyện mô hình sẽ mang ra độ chính xác cao, khả năng phát hiện được người lạ, tốc độ thực thi nhanh, không quá phụ thuộc vào mô hình embedding trước vì mô hình cũng tự huấn luyện để nhận dạng được. Nhược điểm là gặp khó khăn khi thêm dữ liệu mới là cần huấn luyện lại từ đầu không hợp với các trường hợp tập dữ liệu biến động theo thời gian.

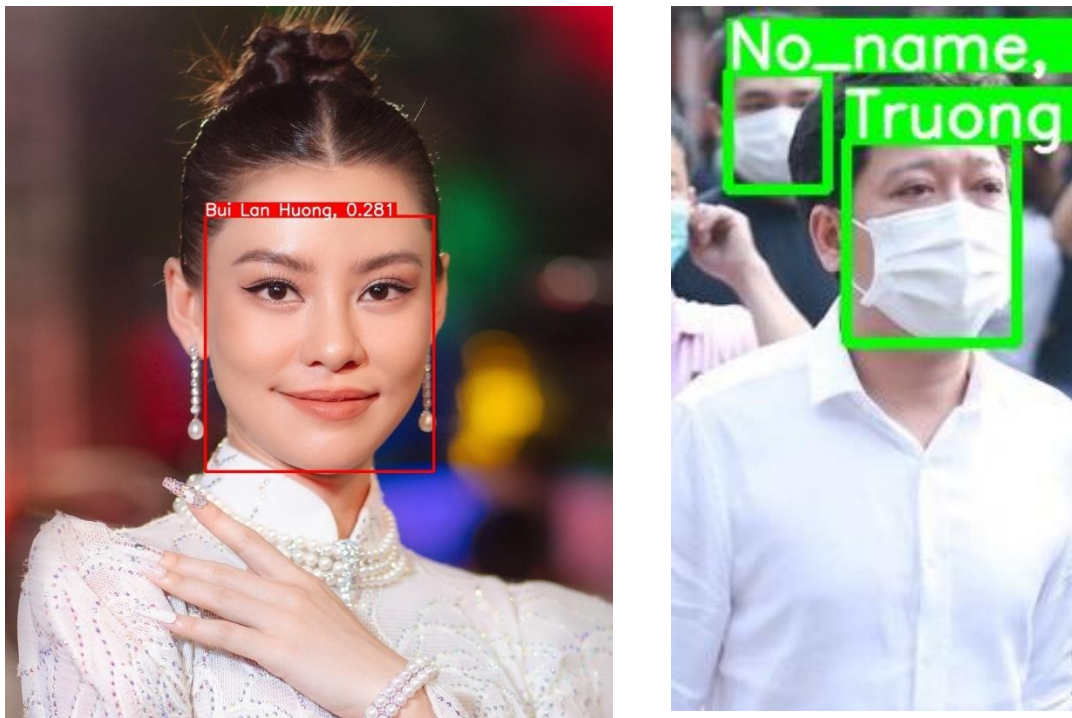
4.4 Mô hình hoàn thiện

Về tổng thể mô hình phát hiện khuôn mặt được sử dụng trong thư viện FaceNet-pytorch là MTCNN và làm việc dựa trên thư viện Python PIL. Còn trong mô hình của bài toán áp dụng mô hình YOLOv5 để phát hiện và phân loại làm việc dựa trên OpenCV. Nên cần chuyển dạng dữ liệu trước khi qua mô hình nhận dạng gương mặt sử dụng FaceNet

Quá trình áp dụng trong chăm công sẽ tùy vào số lượng đối tượng mà áp dụng các phương pháp phân loại khác nhau. Nếu số lượng dữ liệu khuôn mặt ít sẽ áp dụng learning similarity để phân loại. Trong trường hợp dữ liệu quá nhiều thì để tăng độ chính xác ta cần huấn luyện trên neural network. Hình 4.20 sẽ demo việc nhận dạng khuôn mặt khi không đeo khẩu trang và đeo khẩu trang của tập dữ liệu thực tế. Hình 4.21 là việc nhận diện khuôn mặt của các người nổi tiếng được lấy ngẫu nhiên ở trên mạng. Hình 4.22 là khả năng đưa mô hình vào thực tế và có nhiều khuôn mặt với các trạng thái và danh tính khác nhau.



Hình 4.20 Một số hình ảnh demo của tập dữ liệu thực tế



Hình 4.21 Một số demo của hình ảnh người nổi tiếng



Hình 4.22 Một số hình ảnh demo kết quả sau cùng

4.5 Kết luận chương

Chương đã trình bày các phương pháp đã thực hiện đối với từng mô hình phát hiện, phân loại và nhận dạng danh tính. Kết quả thực nghiệm trên từng tập dữ liệu khác nhau và kết luận sau cùng cho các phương pháp đã nghiên cứu

KẾT LUẬN

Kết quả sau khi xây dựng mô hình là hoàn thiện được bài toán nhận diện khuôn mặt với khả năng có thể phát hiện và nhận dạng ngay cả khi khuôn mặt đang đeo khẩu trang. Mô hình được xây và hoạt động tốt với khả năng chính xác và phát hiện được nhiều khuôn mặt trên cùng một khung hình. Ngoài mô hình chung, mô hình riêng phát hiện khuôn mặt có thể phân loại cả người đeo khẩu trang đúng, người đeo khẩu trang sai quy định và người không đeo khẩu trang dựa trên các khuôn mặt được phát hiện, ứng dụng này có thể áp dụng vào các công ty khép kín cần kiểm soát dịch bệnh covid tránh lây lan như khu vực công xưởng, khu vực hành chính, bưu điện, ngân hàng... Mô hình phát hiện khuôn mặt có thể áp dụng cùng mô hình nhận dạng danh tính để ra mô hình nhận dạng khuôn mặt áp dụng tiêu biểu vào các hệ thống chăm công. Các nghiên cứu trong đồ án đã hoàn thành được các mục tiêu đặt ra ban đầu là nhận dạng khuôn mặt ngay cả khi khuôn mặt đang đeo khẩu trang.

Hướng phát triển là mô hình có thể tích hợp thêm các trường hợp khác ngoài đeo khẩu trang như đối tượng có mang kính, đối tượng đội mũ... Trong mô hình phát hiện khuôn mặt do hạn chế về thời gian nên phần dữ liệu được gán nhãn chưa nhiều nên có thể tăng cường dữ liệu để cải thiện mô hình. Về phần dữ liệu trong mô hình nhận diện danh tính, có các cải tiến về cách làm dữ liệu là chỉ áp dụng một bức hình để nhận diện. Phương pháp đòi hỏi nhận diện chỉ trong một bức hình thì cần một mô hình mã hoá đủ tốt để tạo ra nhiều embedding như embedding mang khẩu trang, embedding đeo kính, embedding khuôn mặt xoay trái... Mô hình mã hoá này cần có một yếu tố then chốt là lượng dữ liệu cần có rất nhiều là của người Việt Nam để lấy các đặc trưng của người Việt

TÀI LIỆU THAM KHẢO

- [1] R. N. Keiron O'Shea, "An Introduction to Convolutional Neural Networks".
- [2] B. Mehlig, "Machine learning with neural networks".
- [3] A. F. Agarap, "Deep Learning using Rectified Linear Units (ReLU)".
- [4] G. H. A. K. I. S. R. S. Nitish Srivastava, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting".
- [5] X. Z. S. R. J. S. Kaiming He, "Deep Residual Learning for Image Recognition".
- [6] W. L. Y. J. P. S. S. R. D. A. D. E. V. V. A. R. Christian Szegedy, "Going Deeper with Convolutions".
- [7] A. Z. Karen Simonyan, "Very Deep Convolutional Networks for Large-Scale Image Recognition".
- [8] A. F. Joseph Redmon, "YOLOv3: An Incremental Improvement".
- [9] J. D. T. D. J. M. Ross Girshick, "Rich feature hierarchies for accurate object detection and semantic segmentation".
- [10] S. D. R. G. A. F. Joseph Redmon, "You Only Look Once: Unified, Real-Time Object Detection".
- [11] J. W. J. P. L. Z. Yuanyi Zhong, "Anchor Box Optimization for Object Detection".
- [12] N. T. J. G. A. S. I. R. S. S. Hamid Rezatofighi, "Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression".
- [13] R. B. B. S. Jan Hosang, "Learning non-maximum suppression".
- [14] D. K. J. P. Florian Schroff, "FaceNet: A Unified Embedding for Face Recognition and Clustering".
- [15] U. Licensing, "Github," [Online]. Available: <https://github.com/ultralytics/yolov5>.
- [16] P. Đ. Khánh, "Khoa học dữ liệu," [Online]. Available: <https://phamdinhhkhanh.github.io/2020/03/12/faceNetAlgorithm.html>.

PHỤ LỤC

Các trang web hỗ trợ đồ án như sau:

Hỗ trợ gán nhãn dữ liệu lần 1: Web MakeSense [link](#)

Hỗ trợ gán nhãn dữ liệu lần 2: Web CVAT [link](#)

Hỗ trợ huấn luyện mô hình: Web Kaggle [link](#)

Các nguồn dữ liệu được dùng trong đồ án như sau:

Dữ liệu phát hiện khuôn mặt lần 1 [link](#)

Dữ liệu phát hiện khuôn mặt lần 2 [link](#)

Dữ liệu nhận dạng danh tính thực tế [link](#)

Dữ liệu nhận dạng danh tính người nổi tiếng [link](#)

Dữ liệu nhận dạng danh tính thực tế đeo khẩu trang [link](#)

Dữ liệu nhận dạng danh tính người nổi tiếng đeo khẩu trang [link](#)

Các nguồn code được dùng trong đồ án như sau:

Nguồn code được dùng để huấn luyện mô hình phát hiện và chạy mô hình tổng thể [link](#)

Nguồn code được dùng để huấn luyện mô hình nhận dạng danh tính bằng neural network [link](#)

Nguồn code sinh dữ liệu đeo khẩu trang [link](#)