ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

_☆_

ĐỒ ÁN CUỐI KỲ

NHẬN DIỆN NGƯỜI CÓ ĐEO KHẨU TRANG

Môn: MÁY HỌC – CS114.M11.KHCL

Giảng viên hướng dẫn:

PGs. TS Lê Đình Duy

ThS Phạm Nguyễn Trường An

Sinh viên thực hiện:

Nguyễn Thành Trung - 19522432

Trần Hồ Thiên Phước - 19522057

TP. HỒ CHÍ MINH - 01/2022

Nội dung

PHẦN 1. TỔNG QUAN

I Mô tả bài toán

1 Ngữ cảnh ứng dụng

Đại dịch covid-19 đã mang lại cho xã hội loài người ở thế kỷ 21 rất nhiều khó khan về sức khỏe, tinh thần và kinh tế. Hiện nay các nhà khoa cũng chỉ mới điều chế ra vaccine để giảm tỉ lệ lây nhiễm của virus trong cộng đồng. Vì vậy việc lây nhiễm vẫn có nguy cơ khá cao trong cộng đồng nếu có người không tuân thủ các quy tắc phòng dịch.

Hình 1. Người dân nô nức trên phố đi bộ vào đầu năm 2021.

Qua hình 1, việc ở nhà quá lâu trong một khoảng thời gian dài cũng làm cho con người muốn ra ngoài đi lại ở những nơi công cộng cho giải tỏa những bí bách trong thời kì giãn cách xã hội. Việc này đẩy nguy cơ dịch bùng lại lên nguy cơ khá cao.

Vấn đề cần thiết lúc này là quản lý việc đeo khẩu trang của người dân khi tham gia sinh hoạt, làm việc, đi lại và vui chơi ở nơi công cộng như trung tâm mua sắm, công ty, các cửa hàng tiện lợi, siêu thị, nhà ga, sân bay, các quán ăn, quán cafe, các tạp hóa,...

2 Bài toán

Với đề tài này chúng em muốn xây dựng một model máy học có thể tự thực hiện việc nhận dạng một hoặc nhiều người có đeo khẩu trang hay không, hay có hiểu là giải quyết bài toán phân lớp (classification) cho 2 nhãn *đeo/không đeo* qua hình ảnh đầu vào với:

a Input

- Bức ảnh một đám đông được trích xuất từ camera giám sát hoặc các thiết bị ghi hình có cùng góc quay.

Hình 2. Đám đông đang di chuyển trên hành lang công cộng.

b Output

- Những người trong ảnh có đeo khẩu trang hay không, cùng với xác suất dự đoán tương ứng.

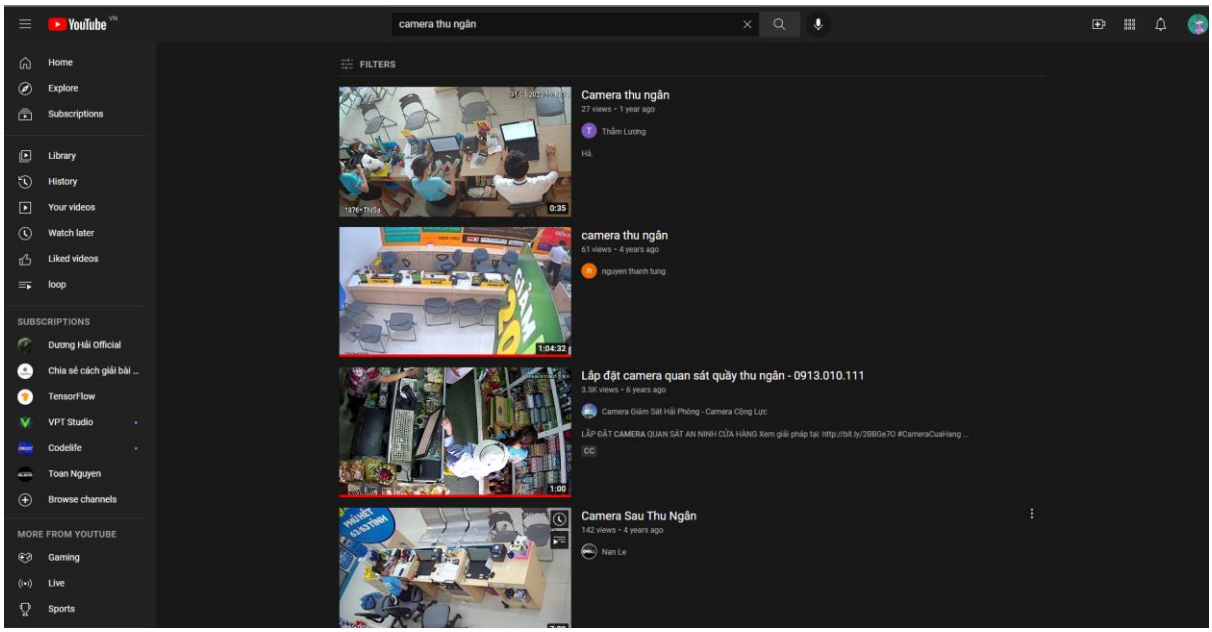
Hình 3. Kết quả dự đoán người đeo/không đeo khẩu trang trong đám đông.

II Thu thập dữ liệu

Từ việc xác định bài toán trên nhóm nhận thấy rằng bộ dữ liệu nên là các hình ảnh phải được chụp từ góc cao, phải thấy được các bộ phận trên khuôn mặt.

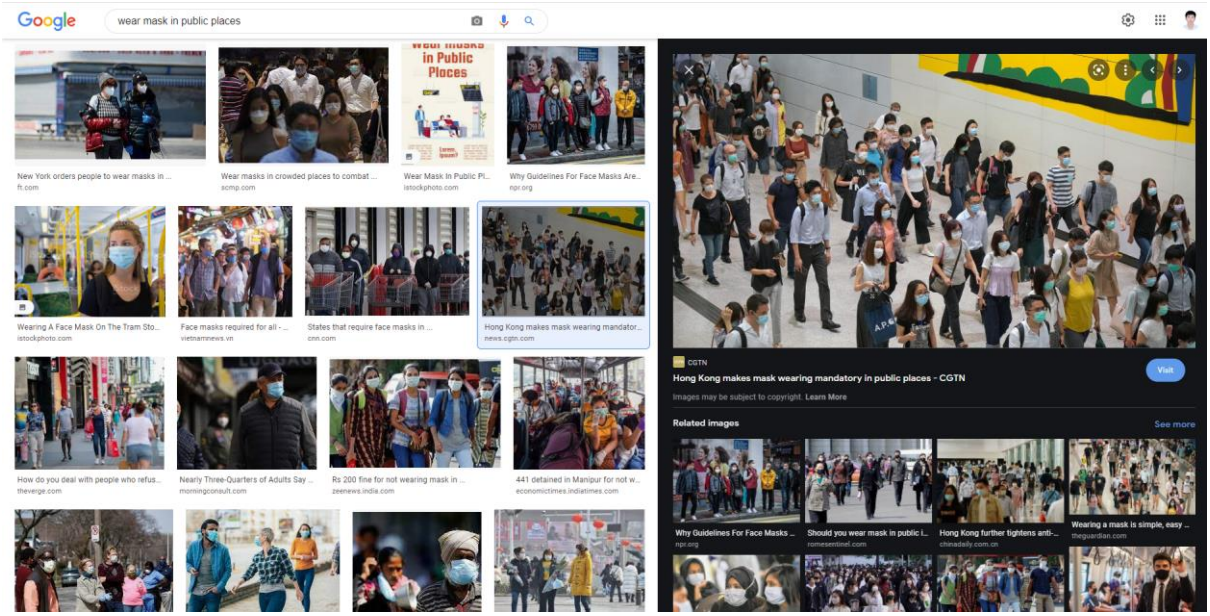
Đầu tiên, nhóm đi hỏi xin các tạp hóa, các quán xá gần nhà nhưng không được câu trả lời mong muốn. Lúc này, nhóm em đã xin những người quen ở xa. Sau khi nhận được các video trích xuất từ camera giám sát, nhóm đã thực hiện chuyển các video về thành ảnh rồi lọc lấy các ảnh đúng tiêu chí được nêu ở Phần 2 mục 1 bằng ứng dụng DVDVideoSoft Free Studio. Vì ở xa nên nhóm cũng chỉ mô tả qua lời nói dẫn đến số video được người thân gửi chỉ lọc ra được 150 ảnh qua 3 videos dài trung bình khoảng 40 phút nhưng mất tới 4 ngày để thu thập. Nhận thấy cách thu thập này không hiệu quả về mặt thời gian nên nhóm có sử dụng thêm 600 ảnh từ cuộc thi Data-Centric AI Competition được tổ chức vào cuối năm 2021 vừa rồi bởi FPT.

Nhóm em cải thiện số lượng ảnh trong bộ dữ liệu bằng cách thu thập các video có trên Youtube với các từ khóa tìm kiếm như 'camera thu ngân', 'demo camera'.. và chuyển đổi các video sang ảnh. Từ khoảng 20 video dài ngắn khác nhau nhóm thu được 497 ảnh, tuy nhiên việc chuyển đổi ảnh này cũng không có tính đa dạng về background, màu sắc, độ sáng tối trong ảnh vì số hình ảnh được chọn chỉ diễn ra tại cùng một thời điểm cho một video.



Hình 4. Thu thập ảnh qua Youtube.

Vì thế nhóm quyết định tăng tính đa dạng dữ liệu của bộ dữ liệu bằng cách thu thập thêm ảnh trên Google qua việc search với từ khóa "*wear mask in public places*", rồi lọc lấy các ảnh phù hợp.



Hình 5. Thu thập ảnh trên Google.

Nhờ đó, số lượng ảnh cuối cùng nhóm thu thập được là 1461 ảnh.

PHẦN 2 XÂY DỰNG BỘ DỮ LIỆU

I Cách thức xây dựng bộ dữ liệu

1 Tiêu chí thu thập ảnh

Ảnh được lấy có các tiêu chí sau:

- Là ảnh chụp từ góc cao.
- Ảnh chụp nơi cộng đồng.
- Ảnh thấy được rõ mặt, hoặc ảnh có thể xác định được chân mày hoặc mắt từ vùng trán xuống tới cằm.
- Ảnh bị che khuất một phần nhưng thấy được các phần còn lại của vùng từ chân mày xuống tới cằm hoặc miệng.

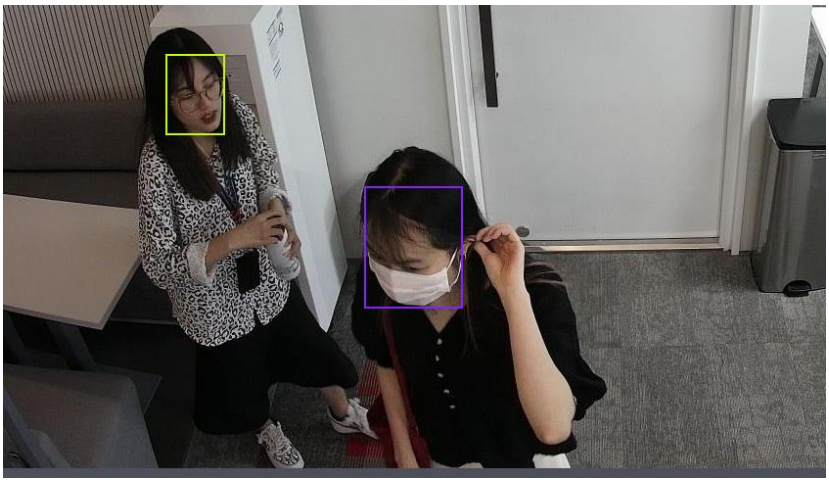
Ảnh không được lấy có các đặc điểm sau:

- Ảnh bị mờ, không xác định được rõ bộ phận nào của mặt chỉ thấy được màu da.
- Ảnh có tất cả người đứng gần nhưng chỉ chụp được phần trán tới mũi.
- Ảnh chụp người đứng quá xa.

2 Quy tắc kẻ bounding box và gán nhãn

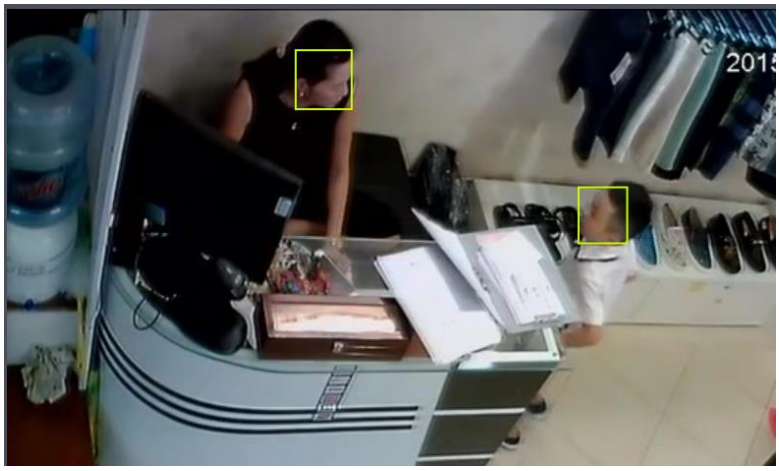
Vì trong ảnh có nhiều người nên việc kẻ bounding box và gán nhãn cũng dựa trên nhiều tiêu chí ảnh được/ không được lấy, gồm các quy tắc như sau:

- Bounding box kẻ từ vùng đỉnh trán xuống tới cằm, không lấy phần tai ở những khuôn mặt thấy rõ.
- Ảnh người bị che khuất một phần nhưng phần còn lại vẫn thấy được như tiêu chí ở phần lấy ảnh thì ta sẽ kẻ bounding box phần thấy được đó.



Hình 6. Kẻ bounding box cho khuôn mặt/phần mặt được thấy rõ.

- Những khuôn mặt được chụp từ góc nghiêng, khuất đi vùng mặt trước thì nhóm sẽ xét nếu vùng tai được nhìn rõ thì sẽ kẻ bounding box, không thì bỏ qua.



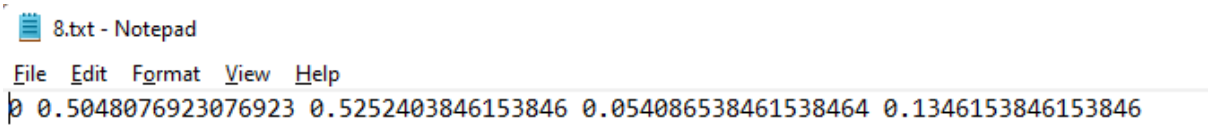
Hình 7. Kẻ bounding box cho các khuôn mặt chụp từ góc nghiêng.

- Trường hợp khuôn mặt được chụp thấy được tất cả các bộ phận trên khuôn mặt nhưng bị mờ/ không thấy rõ thì nhóm sẽ xác định vùng chân mày, mắt có thể phân biệt với màu sắc của da không, nếu có thì thực hiện kẻ bounding box, không thì bỏ qua.

- Không kẻ bounding box với các khuôn mặt không thấy rõ bộ phận chân mày hoặc mắt, các khuôn mặt quá nhỏ.
- Việc gán nhãn cho từng bounding box nhóm đã thống nhất 0 cho nhãn *không đeo* và 2 cho nhãn *đeo*.

Nhận xét chung:

- Ảnh sau khi được gán nhãn sẽ trả về kèm thêm một tệp có định dạng txt, với nội dung: <object-class> <x> <y> <width> <height>



Hình 8. Định dạng tệp nhãn cho mỗi hình

Trong đó:

<object-class>: nhãn của từng đối tượng (*đeo/không đeo*)

<x> , <y>: tọa độ tâm bounding box (theo chuẩn hóa tỉ lệ *size-bounding-box/ size-image*)

<width>, <height>: chiều rộng, chiều cao của object-class (theo chuẩn hóa tỉ lệ *size-bounding-box/ size-image*)

- Thời gian chuẩn bị của nhóm cho đề tài không nhiều nhưng việc thu thập ảnh đạt tiêu chí tốn khá nhiều thời gian nên số lượng ảnh trong bộ dữ liệu còn khá hạn chế và sự đa dạng giữa các ảnh cũng không cao.

II Thao tác xử lý dữ liệu sử dụng Roboflow

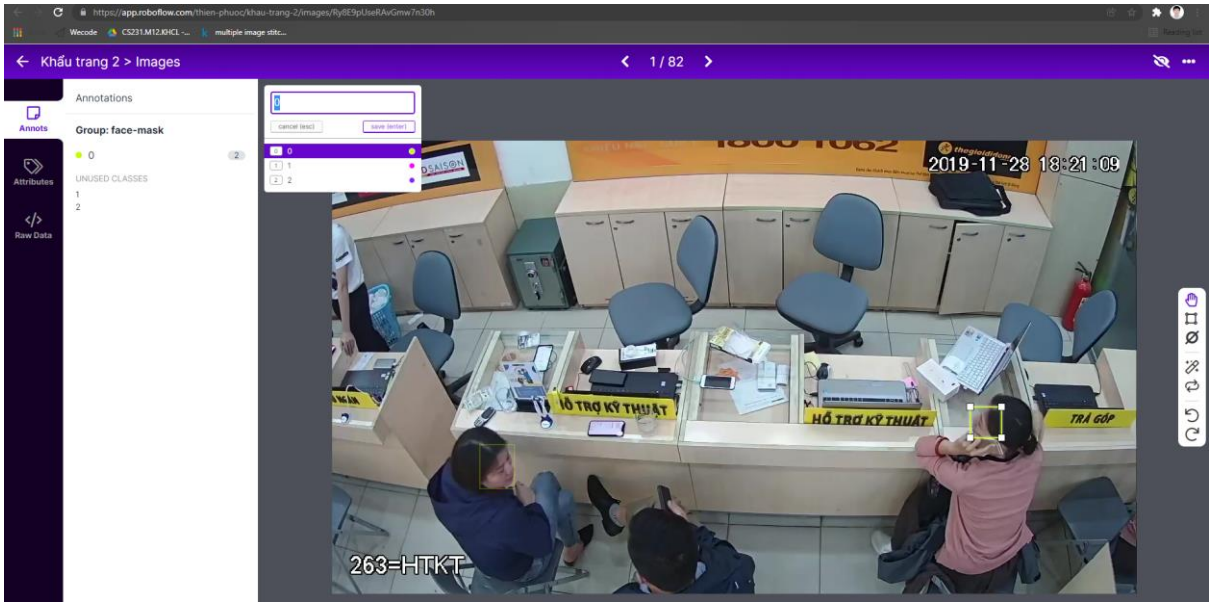
Sau khi thực hiện thu thập và gán nhãn ảnh. Nhóm chuyển dữ liệu lên một ứng dụng có tên Roboflow, ứng dụng này khá hữu ích trong việc tự động chia tỉ lệ train/validation/test, các bước tiền xử lý và tăng cường dữ liệu. Các quá trình trên đều hiển thị trước khi ta tiến hành huấn luyện model.

1 Roboflow

Roboflow được ra đời vào tháng 1 năm 2020 bởi Brad Dwyer và Joseph Nelson, cùng với sự hỗ trợ của những nhà sáng lập Segment, Paypal, Firebase, Stripe, v.v.. .

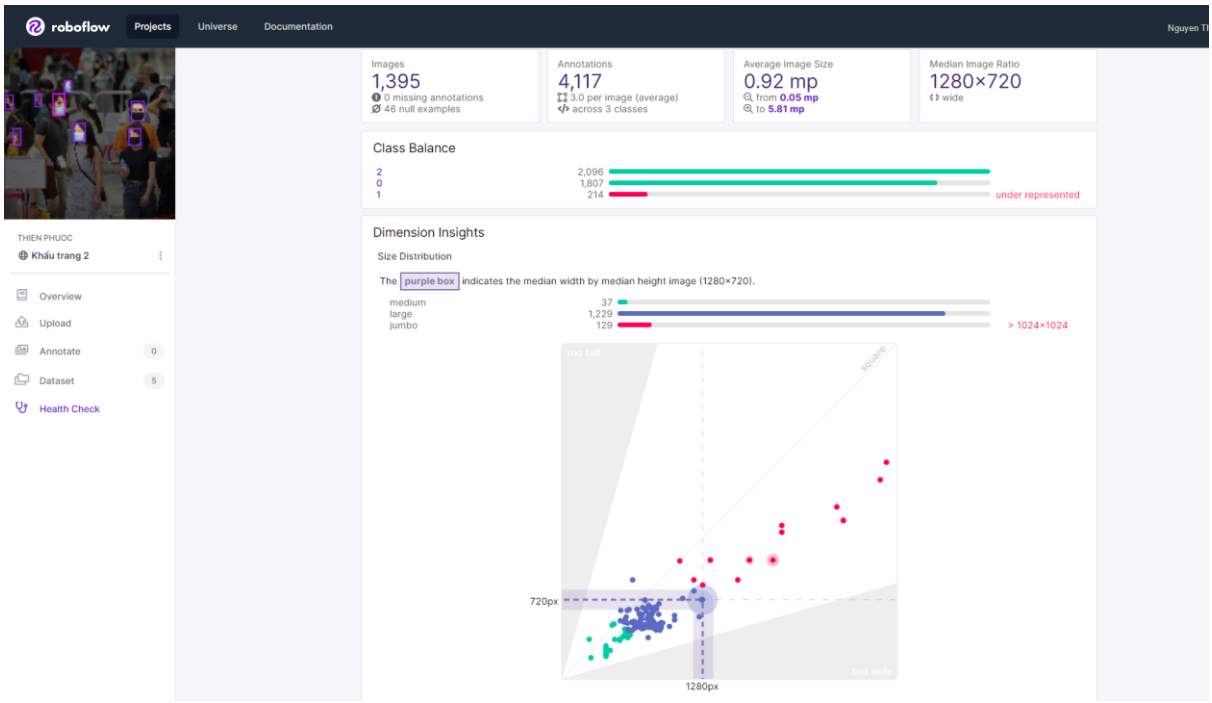
Roboflow là một công cụ hỗ trợ xử lý dữ liệu khá hữu ích và nhanh. Nó tiết kiệm thời gian cho nhóm ở nhiều công việc như:

a Kẻ, điều chỉnh các đường kẻ bounding box, tên nhãn



Hình 9. Điều chỉnh trực tiếp các bouding box và tên nhãn trên Roboflow

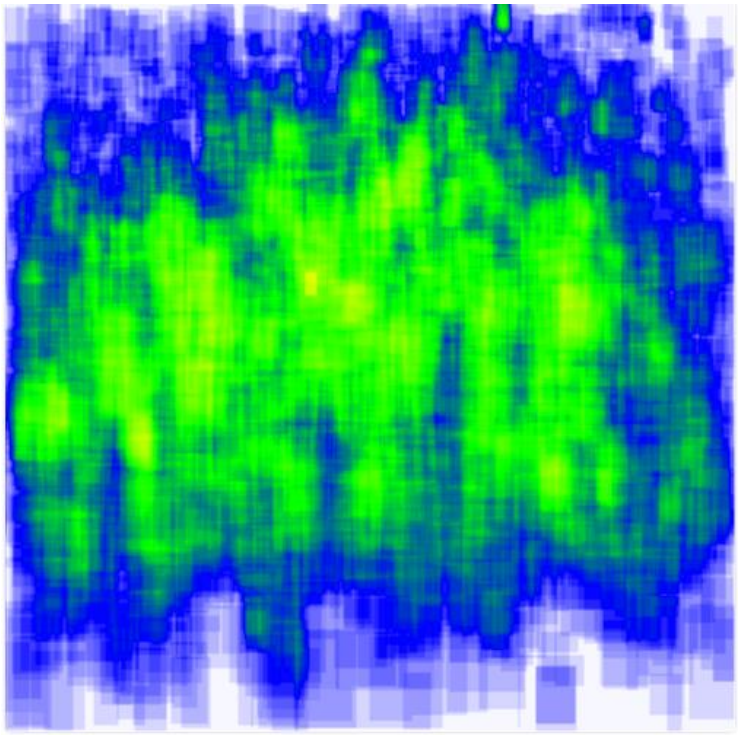
b Thống kê số lượng ảnh, số lượng nhãn và các thông số liên quan ở phần Health Check



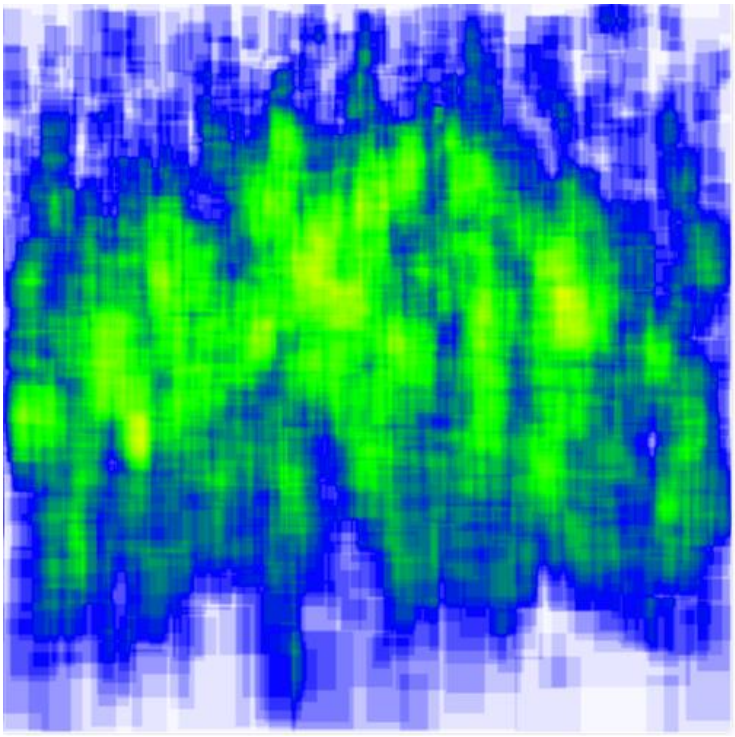
Hình 10. Thống kê tự động các thông tin bộ dữ liệu trên Roboflow

c Phân tích mức độ đa dạng của bộ dữ liệu, của từng nhãn qua biểu đồ Heatmap

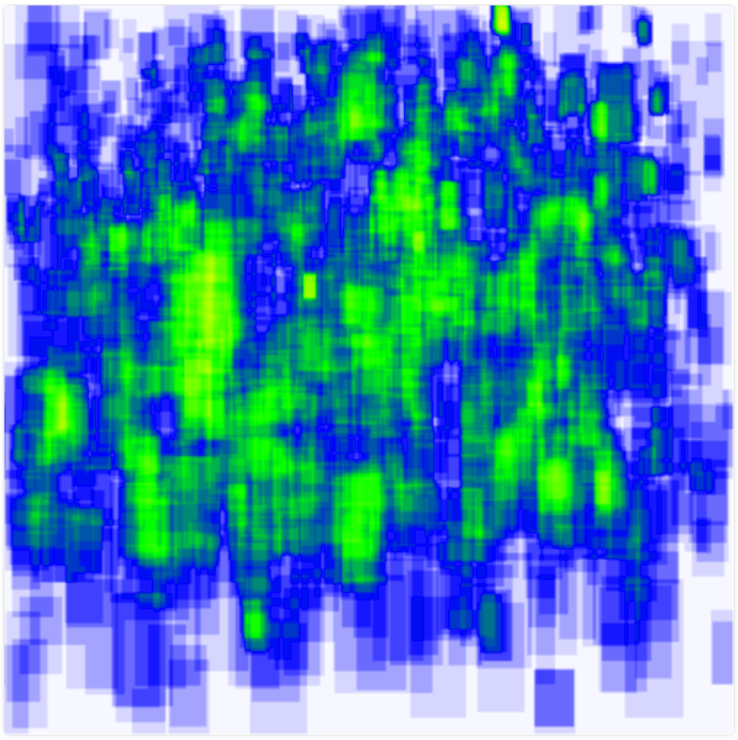
Heatmap là một biểu đồ nhiệt thể hiện sự phân bổ các bounding box trên ảnh trong một bộ dữ liệu qua các vùng màu xanh lá. Vùng màu xanh lá này phân bổ càng rộng, càng đều thì sự phân bổ các bounding box càng đa dạng, bộ dữ liệu càng tốt.



Hình 11. Heatmap của bộ dữ liệu khẩu trang mà nhóm thực hiện.



Hình 12. Heatmap nhãn *đeo* trong bộ dữ liệu của nhóm.

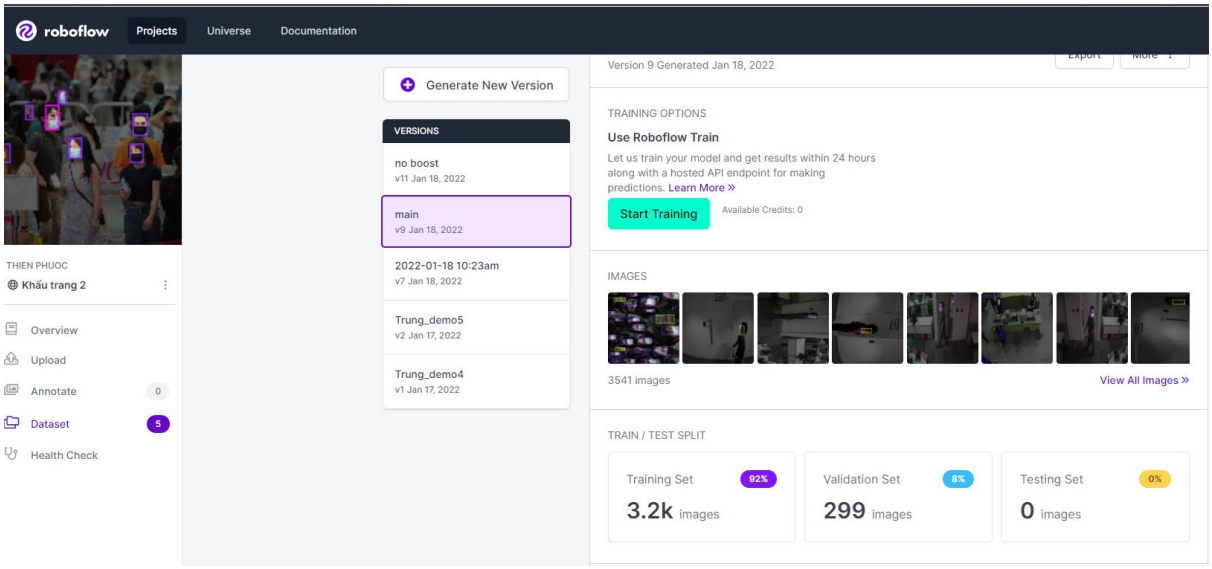


Hình 13. Heatmap nhãn *không đeo* trong bộ dữ liệu của nhóm.

Nhận xét chung: Roboflow mang lại những thông kê dữ liệu khá trực quan về các thông tin chi tiết của ảnh và tính đa dạng trong bộ dữ liệu.

2 Chia tỉ lệ

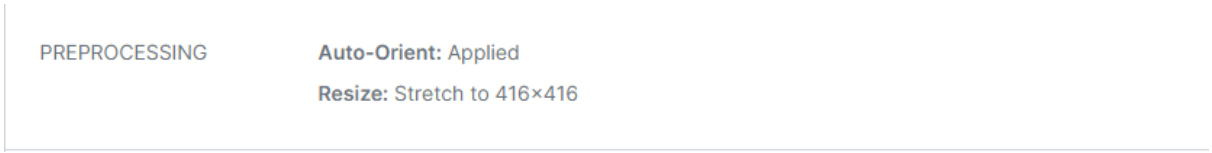
Để thực hiện phân chia dữ liệu hiệu quả, tránh tình trạng overfit trên bộ dữ liệu có số lượng chưa được nhiều, chưa được quá đa dạng và kiểm tra được model có tốt hay không, cùng với đó nhóm sử dụng mô hình YOLOv5 để huấn luyện nên tập validation là quan trọng trong việc đánh giá. Vì vậy, nhóm chỉ dùng 66 ảnh cho việc test, được lưu riêng ở trên drive và không tải lên Roboflow, 1395 ảnh nhóm tải lên Roboflow chia theo tỉ lệ 80/20 cho train/val. Sau khi Roboflow chia ngẫu nhiên ảnh, nhóm thực hiện lọc thủ công lấy những ảnh có đặc trưng riêng, những ảnh không có tính trùng lặp trong cả bộ dữ liệu với các ảnh còn lại sang tập validation, rồi chuyển những ảnh gần giống nhau sang tập train.



Hình 14. Chia dữ liệu ngẫu nhiên theo tỉ lệ 80/20 sử dụng Roboflow

3 Tiền xử lý dữ liệu sử dụng Roboflow

Dữ liệu sau khi được chia trên Roboflow sẽ tiếp tục đến bước tiền xử lý dữ liệu bằng cách chuyển các ảnh về kích thước 414x416. Để đồng bộ tất cả ảnh, tránh tình trạng nhiễu hay lỗi. Việc chuyển ảnh về kích thước nhỏ hơn cũng giúp quá trình huấn luyện mô hình diễn ra nhanh hơn.



Hình 15. Các lựa chọn cho bước tiền xử lý dữ liệu trên Roboflow.

Nhóm có áp dụng thêm lựa chọn ‘Auto-Orient’ trong bước tiền xử lý. Với ‘Auto-Orient’, các bounding box sẽ luôn được định hướng theo đối tượng đã được kẻ bounding box trước đó khi ảnh bị xoay. ‘Auto-Orient’ khá hữu ích cho việc tăng cường dữ liệu sử dụng phép xoay.

4 Tăng cường dữ liệu sử dụng Roboflow

Ở hình 13, biểu đồ Heatmap cho nhãn *không đeo* là còn khá thưa, điều này cho thấy sự phân bổ vị trí các bounding box của nhãn không đeo còn chưa đều và rộng, dữ liệu lúc này chưa tốt. Sẽ có 2 cách phổ biến để khắc phục vấn đề này, đó là:

- Bổ sung dữ liệu bằng việc thu thập thêm ảnh. Tuy nhiên với cách này thì nhóm không đủ thời gian để thu thập được số lượng ảnh có cải thiện Heatmap cho nhãn *không đeo*.

- Tăng cường dữ liệu bằng cách sử dụng các phép biến đổi (flip, rotate, hue, saturation, brightness,…) trên các ảnh có sẵn. Nhóm chọn cách này và tiến hành thực hiên trên Roboflow.

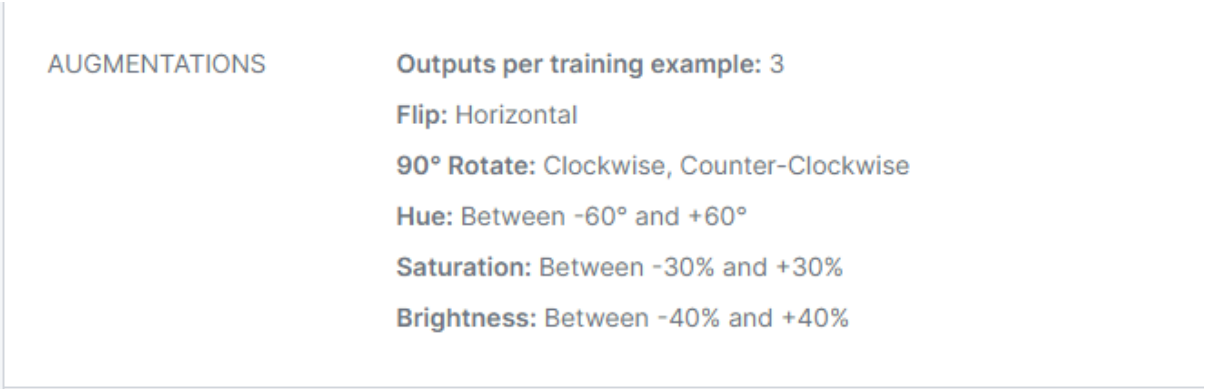
Với Augmentations trên Roboflow, các thông số như hình 16 là lựa chọn mang lại kết quả tốt nhất trong tất cả các lần lựa chọn các thông số một cách ngẫu nhiên của nhóm, vì vậy kết quả của việc tăng cường còn mang khá nhiều tính chủ quan.

- Flip- Horizontal: Lật ngang các đối tượng trong ảnh.
- 90^o Rotate- Clockwise, Counter-Clockwise: Xoay ảnh sang ngang. Giúp mô hình không bị nhầm lẫn với hướng của máy ảnh.

→ Flip, 90^o Rotate: đa dạng vị trí các bounding box.

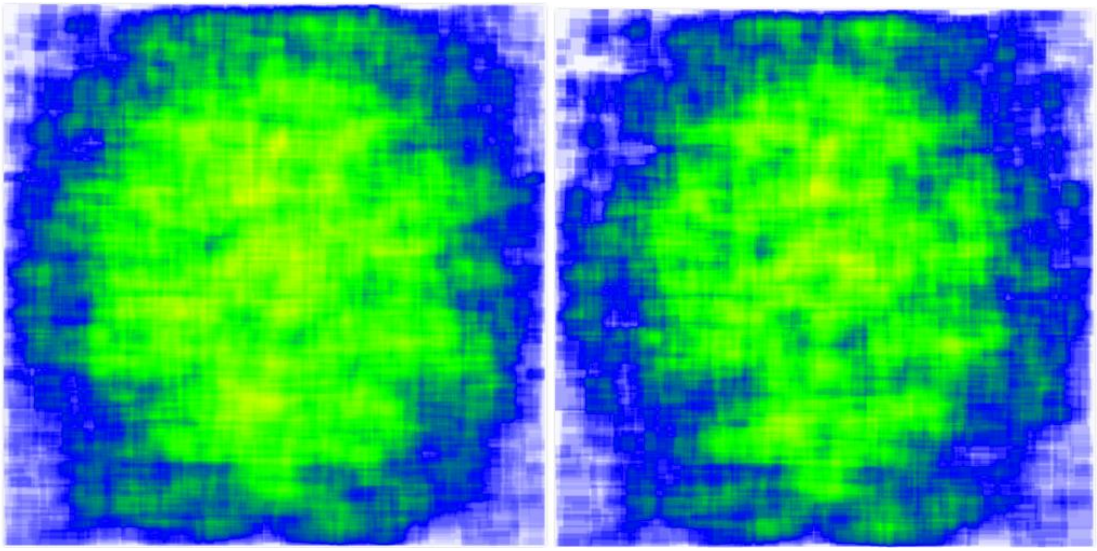
- Hue: thay đổi màu sắc trong khoảng -60o đến +60o của ảnh.
- Saturation: thay đổi cường độ màu sắc trong khoảng -30% đến +30%.
- Brightness: thay đổi độ sáng-tối của nền ảnh trong khoảng -40% đến 40%.

→ Hue, Saturation, Brightness: đa dạng màu sắc, cường độ màu sắc của các ảnh và độ sáng-tối nền ảnh.

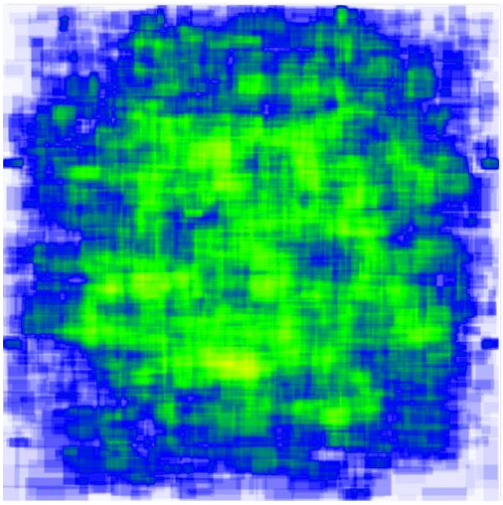


Hình 16. Tăng cường dữ liệu trên Roboflow.

Với việc sử dụng cách tăng cường dữ liệu như trên, kết quả biểu đồ Heatmap nhóm nhận được sau cùng đã cải thiện rõ rệt:



(a) (b)



(c)

Hình 17. Các Heatmap của bộ dữ liệu đã được tăng cường.

- Hình 17-a: Heatmap cho bộ dữ liệu chung của 2 nhãn, vùng phân bổ màu xanh lá trải khá đều và rộng.
- Hình 17-b: Heatmap biểu diễn sự phân bổ các bounding box của nhãn *đeo* qua vùng xanh lá cũng khá tốt.
- Hình 17-c: Heatmap biểu diễn sự phân bổ các bounding box của nhãn không *đeo* lúc này vẫn còn chưa tốt nhưng đã cải thiện so với hình 13.

Sau khi xây dựng được bộ dữ liệu, nhóm tiến hành huấn luyện và đánh giá mô hình.

PHẦN 3 TRAINING VÀ ĐÁNH GIÁ

I Hướng tiếp cận và kế hoạch:

Bài toán nhận diện đối tượng là một bài toán rất quen thuộc trong lĩnh vực computer vision. Hiện nay có rất nhiều mô hình nhận diện đã được các chuyên gia hàng đầu nghiên cứu và xây dựng riêng để giải quyết cho bài toán này một cách hiệu quả nhất. Vì thế việc sử dụng các mô hình đã được xây dựng là một sự lựa chọn tốt tiết kiệm thời gian, công sức mà vẫn mang lại kết quả tốt nhất. Thay vào đó nhóm sẽ tập trung xây dựng bộ data tốt nhất có thể vì data có vai trò rất quan trọng ảnh hưởng đến kết quả của bài toán

Các mô hình nhận diện hiện nay được phân thành hai nhóm chính bao gồm:

- One-stage : Các model được gọi là one-stage khi trong quá trình thiết kế model không hề có bước nào để xác định vật thể có trong tấm ảnh (dự đoán các vùng box có thể chứa đối tượng trong ảnh). Bởi vì không có bước này nên những model thuộc pipeline này sẽ có tốc độ xử lý nhanh hơn so với two-stage. Tuy nhiên, "độ chính xác" của model thường kém hơn so với two-stage object detection. Một số model điển hình : YOLO, Retinanet, SSD,...
- Two-stage : Khác với one-stage object detection thì two-stage có thêm một mạng con (SubNetwork) gọi là RPN (Region Proposal Network) với nhiệm vụ là tìm ra vùng có khả năng chứa vật thể từ bức ảnh, gọi là vùng đề xuất RoI (Region of Interest). Sau đó, dựa vào từng vùng đề xuất này để dự đoán và phân loại. Một số model điển hình : RCNN, Fast RCNN, Faster-RCNN...

Nhóm lựa chọn sử dụng các mô hình thuộc nhóm One-stage vì khả năng xử lý nhanh chóng rất thích hợp cho ngữ cảnh của bài toán nhận diện khẩu trang cho các camera giám sát (xử lý real time).

Vì các mô hình có sẵn đã được nghiên cứu và có những đánh giá tổng quan trên một số trang web và diễn đàn, nhóm đã lựa chọn mô hình YOLOv5 để huấn luyện.

Theo những thông tin so sánh và đánh giá về YOLOv5 mà nhóm tìm hiểu được, YOLOv5 tuy không phải là mô hình One-stage tốt nhất nhưng YOLOv5 phù hợp cho kế hoạch của nhóm, cụ thể:

- YOLOv5 dễ sử dụng, có nhiều hướng dẫn và mẫu ví dụ code, điều này giúp ích cho nhóm trong việc tiếp cận và làm quen với bài toán object detection khi chưa có kinh nghiệm

- YOLOv5 có kích thước nhỏ, thời gian training thấp phù hợp với môi trường làm việc với bộ nhớ có hạn trên google colab của nhóm và việc thử nghiệm nhiều lần để kiểm tra độ hiệu quả của bộ dữ liệu mà nhóm xây dựng.

Độ chính xác tốt không quá thua kém so với các thuật toán Two-Stage.

Tóm lại:

- Nhóm sẽ sử dụng mô hình đã được nghiên cứu thuộc nhóm One-Stage để phù hợp với ngữ cảnh bài toán
- Mô hình được sử dụng là YOLOv5 với 2 phiên bản được sử dụng cho 2 giai đoạn: YOLOv5s (size nhỏ) sử dụng cho việc thử nghiệm kiểm tra bộ dữ liệu, YOLOv5l (size lớn) sử dụng cho việc cải thiện kết quả.
- Tập trung xây dựng bộ dữ liệu thử nghiệm nhiều lần để đạt được kết quả về bộ data sau đó sử dụng mô hình tốt hơn để đạt được kết quả cuối cùng.

II Thuật toán:

YOLOV5:

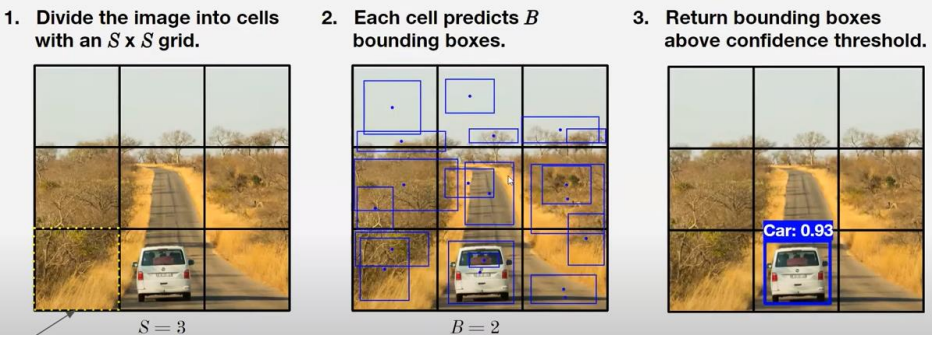
YOLOv5 được phát hành bởi Glenn Jocher vào tháng 9/2020 được phát triển dựa trên YOLOv4 và EfficientDet.

Tuy nhiên YOLOv5 được coi là mô hình không chính thống vì vẫn chưa có bài báo khoa học cụ thể nào viết về mô hình này.

YOLOv5 bao gồm 4 phiên bản ứng với 4 size: YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x. Phiên bản càng lớn mô hình càng phức tạp, thời gian training, detecting lâu hơn nhưng mang lại độ chính xác cao hơn.

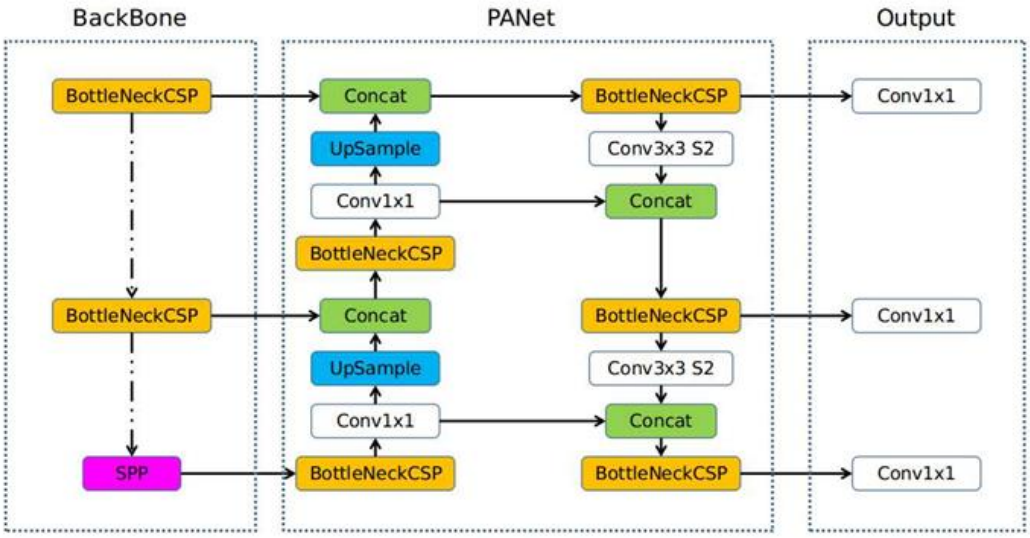
Ý tưởng:

- Chia ảnh thành SxS các box nhỏ
- Mỗi box có nhiệm vụ dự đoán B các bounding box có thể chứa đối tượng
- Trả về những bounding box có điểm confidence cao ngưỡng confidence đã thiết lập trước.



Cấu trúc chia làm ba phần:

- Backbone: là vùng mạng dùng để thu thập các đặc trưng của hình ảnh (YOLOv5 sử dụng CSP: https://arxiv.org/pdf/1911.11929v1.pdf)
- Neck: là vùng dùng để kết hợp các feature và chuyển nó đến vùng đưa ra dự đoán (YOLOv5 sử dụng PANet https://arxiv.org/pdf/1803.01534.pdf)
- Head: là vùng dùng phụ trách đưa ra dự đoán bounding box và class (sử dụng các lớp Convolution network)



Hình 18. Cấu trúc YOLOv5.

III Training:

Nhóm thực hiện training trên hai phiên bản của mô hình YOLOv5 là YOLOv5s và YOLOv5l

Các thông số cài đặt:

	Img size	Batch size	Epochs
YOLOv5s	416	17	200
YOLOv5l	416	17	200

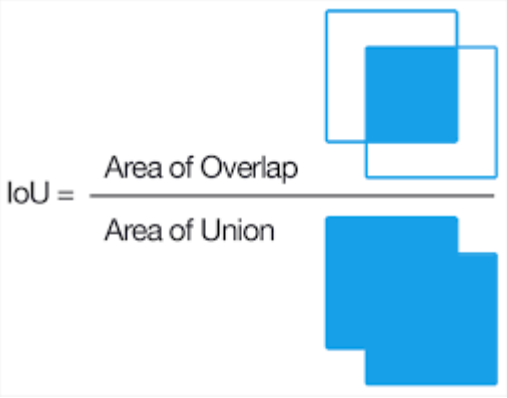
Thời gian training và kích thước mô hình trên mô trường gg colab gpu Tesla T4:

	Thời gian training (hours)	Kích thước mô hình (MB)
YOLOv5s	1.618	14.2MB
YOLOv5l	4.529	92.8MB

IV Phương thức đánh giá:

IOU:

Được tính bằng tỉ số diện tích của phần giao và phần hợp giữa bounding box đúng trong bộ data và bounding box mô hình dự đoán được.

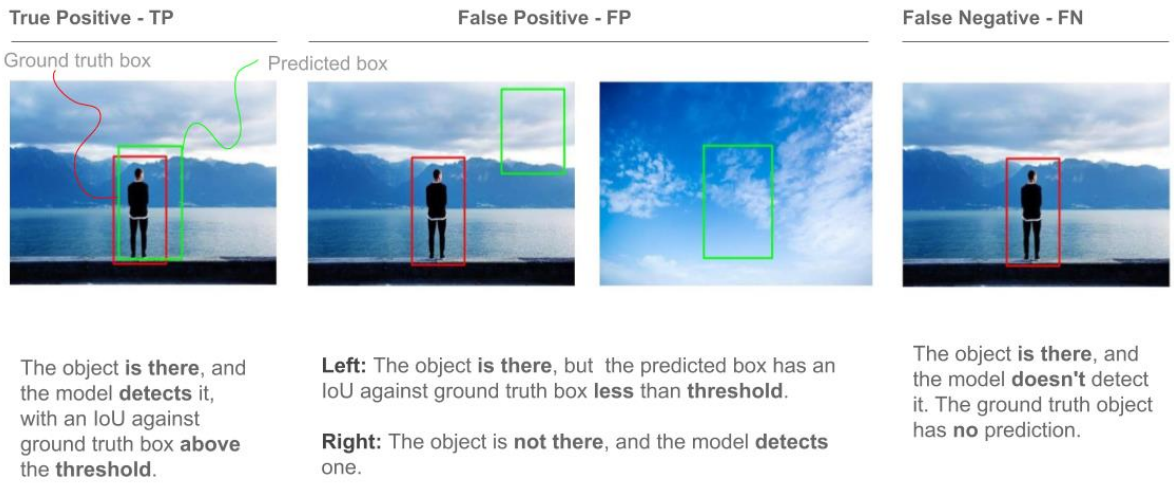


Hình 19. Định nghĩa về IoU bằng công thức

IoU là điểm số dùng để phân loại các dự đoán của mô hình:

- Dự đoán có điểm IoU thấp hơn ngưỡng IoU cho trước sẽ được xem là True Positive (dự đoán đúng)
- Dự đoán có điểm IoU thấp hơn ngưỡng IoU cho trước sẽ được xem là False Positive.

- Những vùng có đối tượng nhưng không được dự đoán là False Negative.



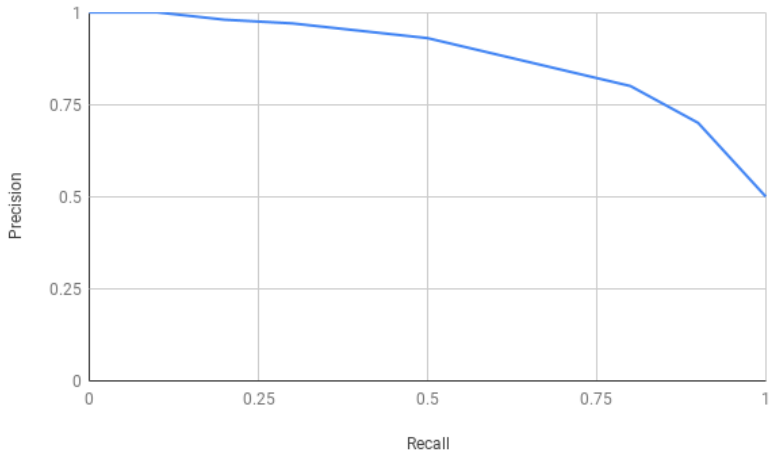
Hình 20. Các trường hợp dự đoán dùng IoU.

Precision và Recall :

- Precision: thể hiện độ chính xác của dự đoán, precision càng cao dự đoán của mô hình càng chính xác.
- Precision được tính bằng tổng số dự đoán được cho là TP chia cho tổng số dự đoán (TP/(TP+FP))
- Recall: thể hiện độ phủ của dự đoán, recall càng cao khả năng bỏ sót dự đoán càng thấp.
- Recall được tính bằng tổng số dự đoán được cho là TP chia cho tất cả dự đoán đúng thực sự có (TP/(TP+FN)).

mAP:

Sau khi đã tính được Precision và Recall, chúng ta tiến hành thực hiện thay đổi ngưỡng IOU và quan sát giá trị của Precision và Recall. Đặt Recall làm trục x, Precision làm trục y => được cong (Recall-Precision Curve)**.**



Hình 21. Đường cong Recall-Precision Curve.

Gía trị AP: là điểm số tóm tắt đường cong P-R, AP càng cao mô hình dự đoán càng hiệu quả cả về độ chính xác lẫn độ phủ(trên 1 nhãn).

AP được tính: tổng n-1 giá trị độ chệnh lệch của Recall tại ngưỡng IOU hiện tại và ngưỡng IOU kế tiếp nhân cho Precision tại ngưỡng hiện tại.

$$AP = \sum_{k=0}^{k=n-1} [Recalls(k) - Recalls(k+1)] * Precisions(k)$$

$$Recalls(n) = 0, Precisions(n) = 1$$

$$n = Number\ of\ thresholds.$$

mAP là điểm số đánh giá tốt nhất trên một mô hình, là trung bình cộng của n giá trị AP tính được trên mỗi class

mAP cho biết được sự hiệu quả của mô hình trên toàn bộ các class cần dự đoán trong mô hình.

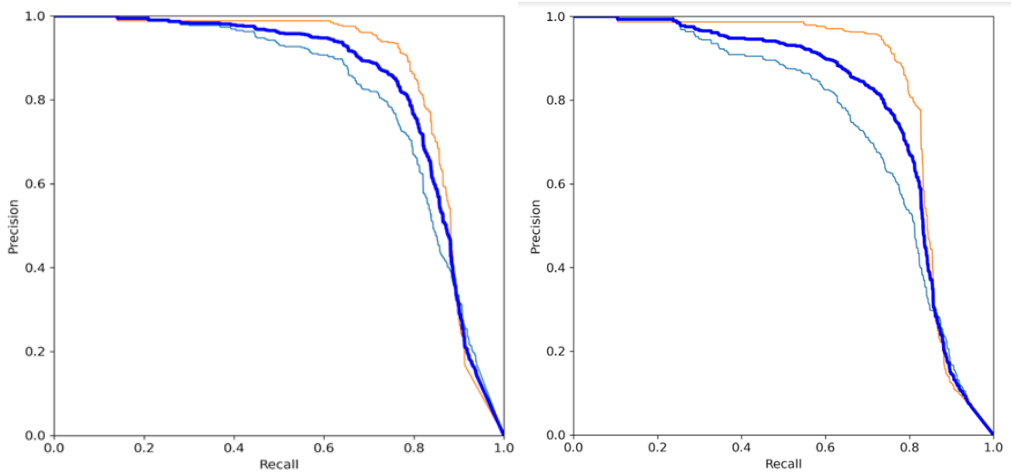
$$mAP = \frac{1}{n} \sum_{k=1}^{k=n} AP_k$$

$$AP_k = \text{the AP of class } k$$

$$n = \text{the number of classes}$$

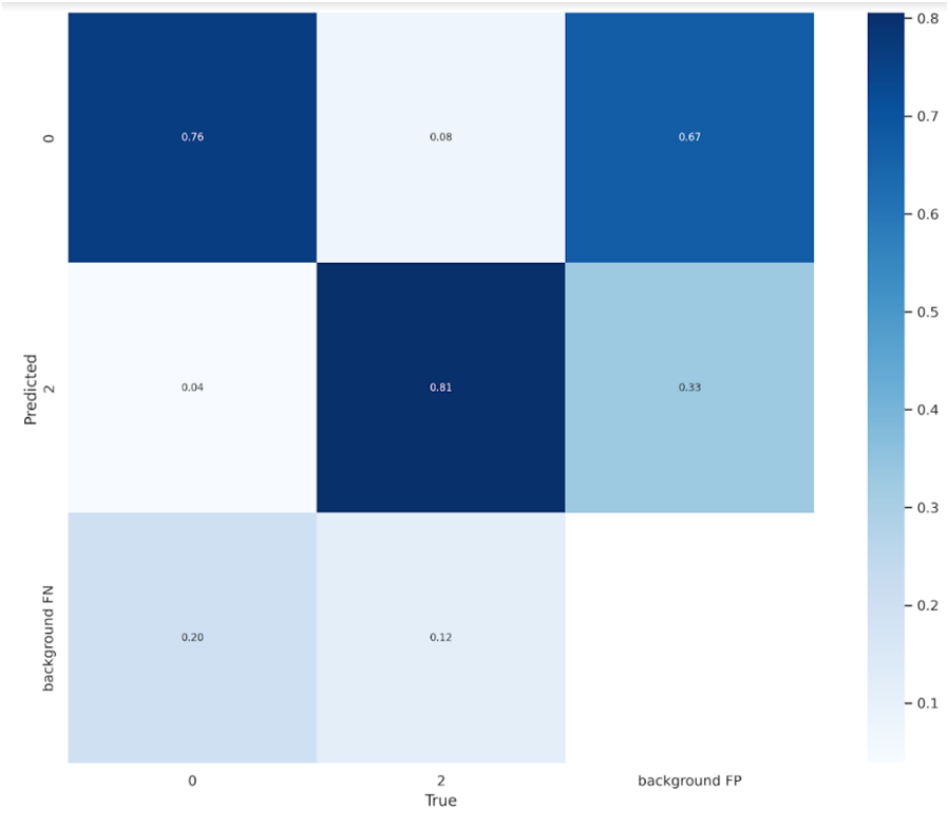
V Kết quả:
Số liệu:

Tên mô hình	Precision	Recall	mAP@0.5	mAP@0.5:0.95
YOLOv5s	0.832	0.715	0.79	0.407
YOLOv5l	0.86	0.753	0.836	0.45

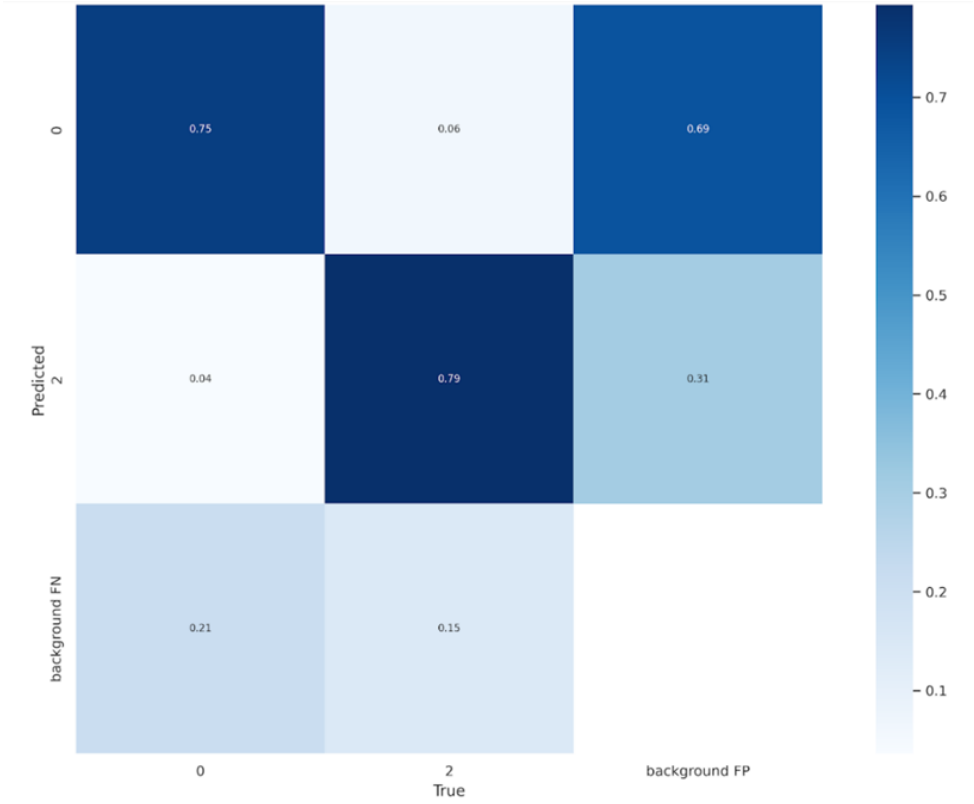


Biểu đồ đường cong P-R: YOLOv5l (bên trái) và YOLOv5s (bên phải)

Confusion matrix là ma trận biểu hiễn các thống kê về dự đoán đúng, sai theo từng class giúp phát hiện lỗi sai của mô hình: YOLOv5l (Hình 22) và YOLOv5s (Hình 23).

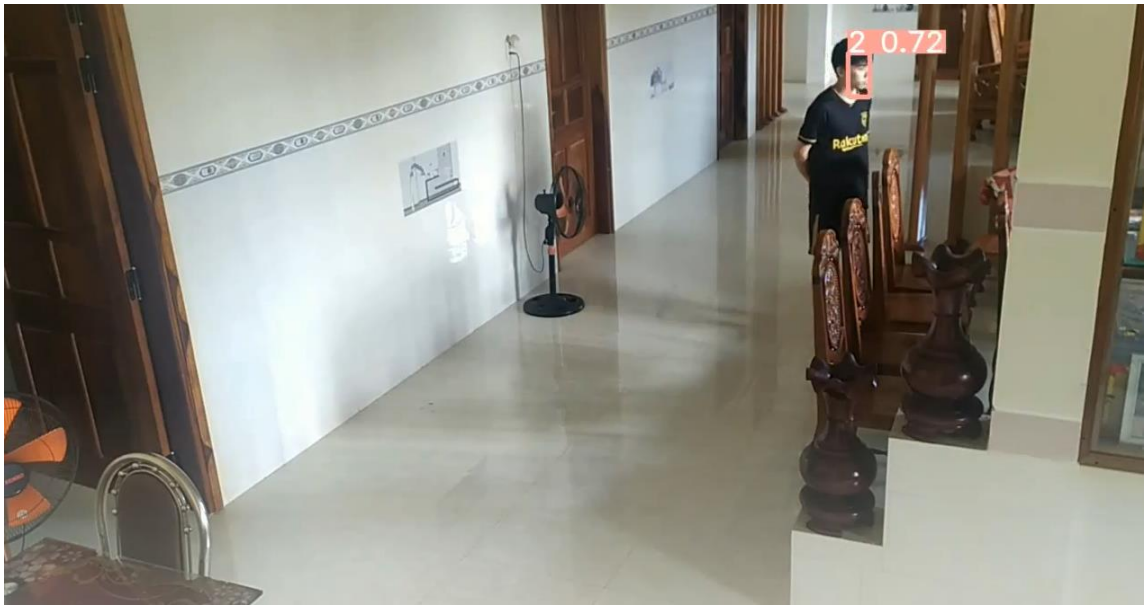


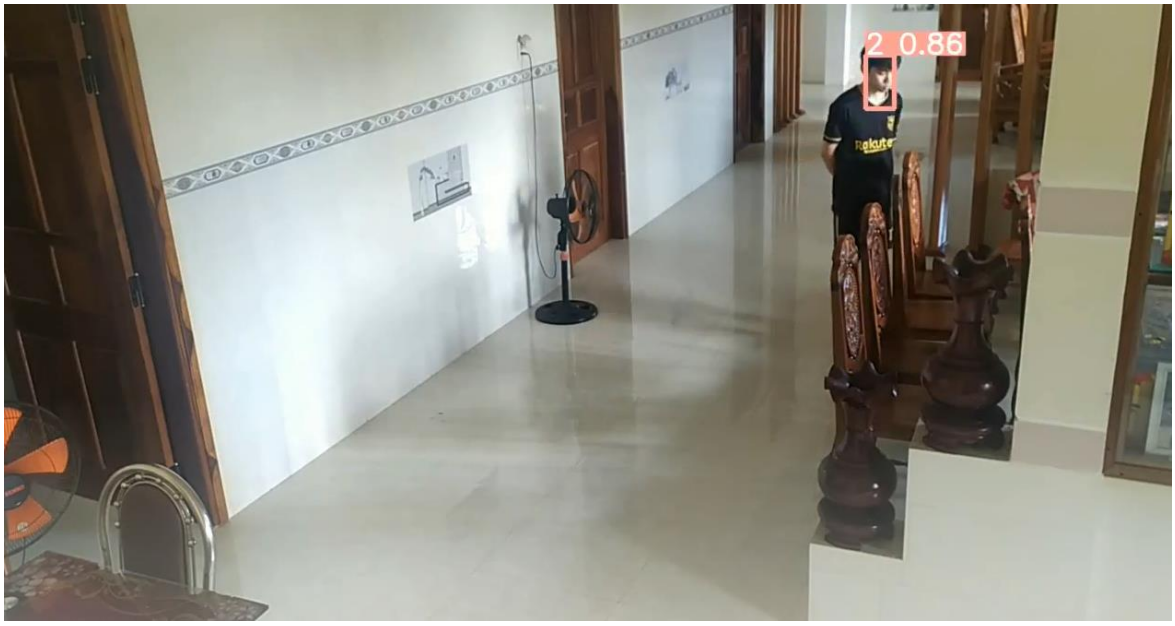
Hình 22. Confusion matrix của YOLOv5l.

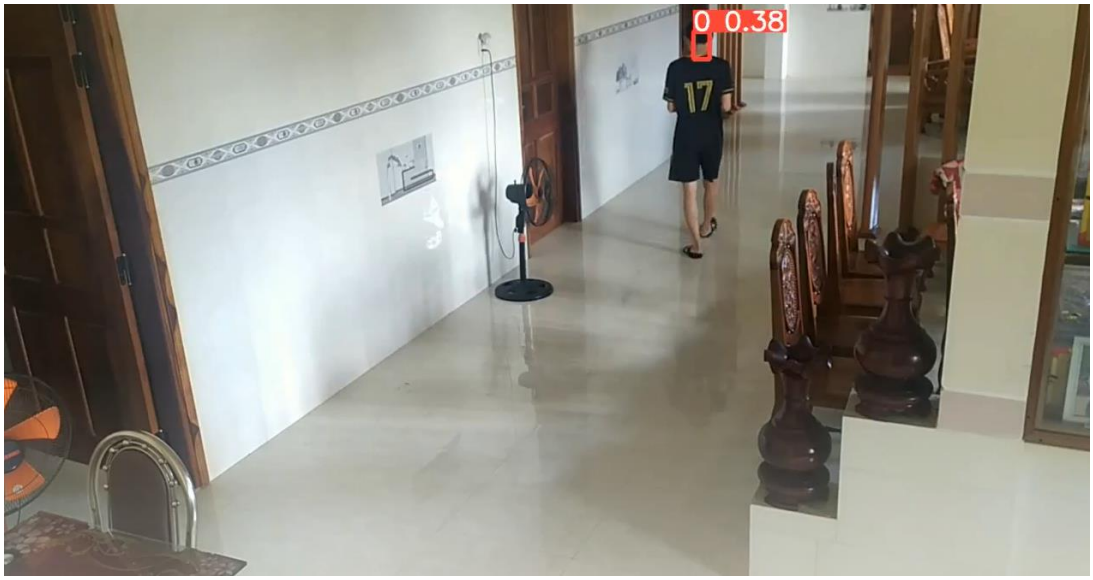


Hình 23. Confusion matrix của YOLOv5s.

Hình ảnh một số lỗi sai test trên YOLOv5l:







Kết quả hình ảnh những dự đoán tốt:

Kết quả video demo:

Video test trên YOLOv5l:
https://drive.google.com/file/d/1MVPR5cv0PnfBdILVh1fr24sIpVcKZobR/view?usp=sharing

Video test trên YOLOv5s:

https://drive.google.com/file/d/1pu3gxIMcxfFczklhjad2a5GvFMSNDiyM/view?usp=sharing

Nhận xét:

- Về điểm số đánh giá ta thấy YOLOv5l đều nhỉnh hơn YOLOv5s về mọi mặt cụ thể: Precision tăng 3,36%, Recall tăng 5,31%, mAP@0.5 tăng 5,82%, mAP@0.5:0.95 tăng 10,57%. Điều này cho thấy việc sử dụng mô hình YOLOv5l tuy có thời gian training lâu hơn, có cấu trúc phức tạp hơn nhưng mang lại kết quả tốt hơn.
- Thông qua biểu đồ P-R của cả hai mô hình ta thấy những dự đoán cho class "đeo khẩu trang" tốt hơn class "không đeo khẩu trang". Điều này cho thấy data về nhãn "không đeo khẩu trang" còn thiếu đa dạng và cần được bổ sung cải thiện
- Thông qua confusion matrix ta thấy mô hình thường nhầm lẫn giữa nhãn "không đeo khẩu trang" và background. Lý do: nhóm đã thực hiện gán nhãn vẽ bounding box cho một số gương mặt xa và mờ gây hiện tượng nhầm lẫn nhãn "không đeo khẩu trang" với các hình ảnh khác có màu sắc và hình dạng gần giống.
- Thông qua hình ảnh ta thấy một số trường hợp nhầm lẫn giữa nhãn "không đeo khẩu trang" và "đeo khẩu trang" khi mặt bị sáng quá mức. Lý do: do việc tăng cường dữ liệu thông qua việc tăng sáng quá mức vô tình gây ra nhiễu lỗi trong bộ data.
- Một số vật có hình dạng gần giống khẩu trang bị nhận dạng nhầm (cụ thể: cổ áo)
- Mô hình bỏ sót một số gương mặt đeo khẩu trang với màu sắc đặc biệt và người đội nón. Nguyên nhân: bộ dữ liệu không đủ đa dạng về những trường hợp như trên.

Cải thiện kết quả mô hình:

- Nghiên cứu một số thuật toán được đánh giá tốt hơn để thử nghiệm trên mô hình.
- Bổ sung data để giải quyết một số lỗi:
- Hình ảnh về những gương mặt có đội nón
- Hình ảnh về nhừng gương mặt đeo khẩu trang với màu sắc đa dạng hơn
- Hình ảnh về gương mặt không đeo khẩu trang ở nhiều góc quay và khoảng cách hơn.
- Hình ảnh không có bounding box về một số vật thể có hình dạng dễ gây nhầm lẫn (gáy, cổ áo,...)
- Loại bỏ nhưng bounding box về gương mặt nhưng quá mờ và xa.
- Điều chỉnh tăng cường dữ liệu phù hợp tránh gây nhiễu

- Nghiên cứu thực hiện thêm các phương pháp tăng cường trên bộ dữ liệu.

Link tham khảo

Phương pháp đánh giá mô hình object detection:

https://blog.paperspace.com/mean-average-precision/

https://jonathan-hui.medium.com/map-mean-average-precision-for-object-detection-45c121a31173

So sánh các mô hình object detection:

https://viblo.asia/p/object-detection-speed-and-accuracy-faster-r-cnn-r-fcn-ssd-fpn-retinanet-and-yolov3-ByEZkJRWKQ0

So sánh giữa các phiên bản YOLOv5:

https://github.com/ultralytics/yolov5

Ý tưởng về YOLO:

https://aicurious.io/posts/tim-hieu-yolo-cho-phat-hien-vat-tu-v1-den-v3/

Tìm hiểu về YOLOv5:

https://blog.roboflow.com/yolov4-versus-yolov5/