Tạp chí Khoa học Trường Đại học Cần Thơ
Phần A: Khoa học Tự nhiên, Công nghệ và Môi trường
website: ctujsvn.ctu.edu.vn

DOI:10.22144/ctu.jvn.2022.217

NGHIÊN CỨU THUẬT TOÁN THỊ GIÁC MÁY TÍNH ĐỂ THEO DÕI VIỆC THỰC HIỆN CÁC QUY ĐỊNH VỀ PHÒNG NGỪA COVID-19 SỬ DỤNG KỸ THUẬT HỌC SÂU

Nguyễn Đức Thiện[1], Lư Tất Thắng[2], Nguyễn Văn Thặng[3] và Trương Quốc Bảo[4*]

[1]*Công ty Cổ phần Công trình đô thị Sóc Trăng*

[2]*VNPT Đồng Tháp*

[3]*Trường Cao đẳng Nghề Kiên Giang*

[4]*Trường Bách Khoa, Trường Đại học Cần Thơ*

**Người chịu trách nhiệm về bài viết: Trương Quốc Bảo (email: tqbao@ctu.edu.vn)*

Thông tin chung:
Ngày nhận bài: 27/11/2021
Ngày nhận bài sửa: 22/06/2022
Ngày duyệt đăng: 19/07/2022

Title:
A a study on computer vision algorithm for monitoring the implementation of regulations on the prevention Covid-19 using deep learning

Từ khóa:
Chuyển đổi Bird's-eye view, học sâu, phát hiện đeo khẩu trang, thị giác máy tính, ước lượng khoảng cách, YOLOv5

Keywords:
Bird eye view transformation, computer vision, distance estmation, deep learning, face mask detection, YOLOv5

ABSTRACT

This study is aimed to detect and to check compliance with regulations on wearing face masks and keeping social distance in crowded places. Deep learning in object detection through image input was used. The YOLO model, which is state-of-the-art algorithm is used to build a model to detect the correct or incorrect wearing of masks. In addition, using this approach can detect people, check keeping social distance by using the Euclidean algorithm to calculate distance between bounding box around persons who were detected in the image, combined with the Bird's-eye view transformation algorithm. The test uses a dataset consisting of 40 images, with two people classified by actual standing distance from each other: greater than or equal to 2 m and less than 2 m. At the same time, each person in the image with a different mask-wearing state is actually classified into three classes: wearing correct or in correct mask and without wearing mask. The testing results reached 90% for the group with the standing distance less or greater than 2 m. The mask-wearing identification test had the following results: 86.67% for the object is wearing correct mask, 76.67% for without wearing mask and 65% for wearing wrong mask.

TÓM TẮT

Bài báo này được thực hiện nhằm nghiên cứu phát hiện và kiểm tra việc tuân thủ các quy định về đeo khẩu trang, giữ khoảng cách xã hội ở các địa điểm đông đúc. Mô hình YOLO được sử dụng để xây dựng thuật toán phát hiện đeo khẩu trang đúng hay không đúng quy định, đồng thời kiểm tra việc giữ khoảng cách xã hội bằng việc sử dụng thuật toán tính khoảng cách Euclid giữa các khung bao quanh người được phát hiện trong hình ảnh, kết hợp thuật toán chuyển đổi Bird's-eye view. Tập dữ liệu được sử dụng bao gồm 40 hình ảnh với hai đối tượng người được phân loại thực tế theo khoảng cách đứng với nhau: lớn hơn hoặc bằng 2 m và nhỏ hơn 2 m. Đồng thời, mỗi đối tượng người trong hình ảnh được phân loại thành ba lớp: đeo khẩu trang đúng hay không đúng và không đeo khẩu trang. Kết quả thử nghiệm khoảng cách đối tượng đạt 90% và nhận diện đối tượng đeo khẩu trang có kết quả như sau: 86,67% đeo khẩu trang đúng, 76,67% không đeo khẩu trang và 65% đeo khẩu trang sai.

1. GIỚI THIỆU

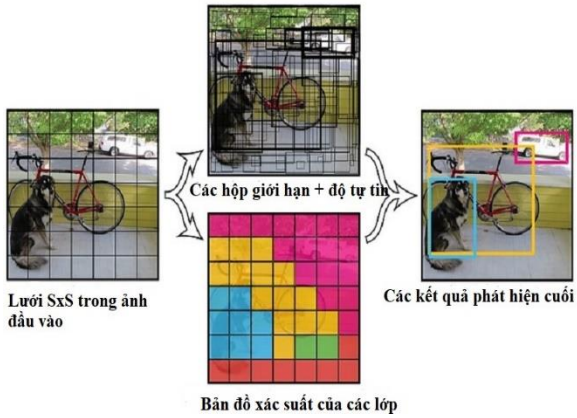
Thị giác máy tính là tập hợp con của trí tuệ nhân tạo sử dụng sức mạnh của máy tính để trích xuất thông tin có ý nghĩa từ các tập dữ liệu được cung cấp, các tập dữ liệu đó có thể là hình ảnh, video... Đại dịch Covid-19 đã tác động rất lớn đến các lĩnh vực kinh tế - xã hội ở Việt Nam nói riêng và thế giới nói chung. Mặc dù đã có những tiến bộ trong việc điều trị, nhiều loại vaccine đã ra đời, việc tiêm chủng đã được đẩy mạnh, tuy nhiên tình hình dịch bệnh vẫn còn tiếp diễn do khả năng tiến hóa và lây lan mạnh của virus. Vì vậy, việc thích nghi, kết hợp tuân thủ các biện pháp phòng, chống dịch theo hướng dẫn của Bộ Y tế, cụ thể là thực hiện tốt thông điệp 5K - khẩu trang, khử khuẩn, khoảng cách, không tụ tập và khai báo y tế là hết sức cần thiết, góp phần phòng, chống, không làm tái phát nguồn lây lan của dịch bệnh.

Để đảm bảo an toàn, tiết kiệm về thời gian, công sức các tổ chức xã hội cũng như doanh nghiệp đang hướng đến việc tự động hóa việc phát hiện vi phạm quy định phòng chống dịch Covid-19, đặc biệt là quy định về đeo khẩu trang và giữ khoảng cách xã hội (Kumar & Shetty, 2021), việc ứng dụng thị giác máy tính trong phát hiện đeo khẩu trang, phát hiện người làm cơ sở để tính khoảng cách xã hội là hết sức cần thiết. Một vài giải pháp đã được đề xuất và nghiên cứu, ứng dụng phát hiện khẩu trang, phát hiện vi phạm khoảng cách xã hội sử dụng mạng CNN (Convolutional Neural Networks) như xây dựng mạng CNN dựa trên keras để phát hiện khẩu trang, mô hình YOLOv3 huấn luyện sẵn để phát hiện người và tính khoảng cách bằng công thức Eculid (Kumar & Shetty, 2021); phát hiện khuôn mặt đeo khẩu trang theo thời gian thực, so sánh hiệu suất giữa mô hình MTCNN + ResNet18 và YOLOv5 trên tập dữ liệu là video (Ding et al., 2021); theo dõi khoảng cách xã hội sử dụng mô hình YOLOv5 nhận diện đối tượng người trong video, sử dụng công thức Eculid để tính khoảng cách (Shukla et al., 2021). Tuy nhiên, hiện tại chưa có một nghiên cứu hay báo cáo về việc sử dụng đồng thời thuật toán YOLOv5 trong việc nhận diện đeo khẩu trang và phát hiện người, áp dụng chuyển đổi Bird's-eye view trong tính khoảng cách Eculid. Trong các thuật toán được sử dụng để phát hiện đối tượng trong hình ảnh, YOLO nổi bật hơn nhờ tốc độ xử lý nhanh, độ chính xác cao, YOLOv5 duy trì được tốc độ 52 FPS trên video thử nghiệm, trong khi MTCNN + ResNet18 chỉ đạt 6 FPS, YOLO có khả năng phát hiện ổn định hơn, do học cách phát hiện trực tiếp "khuôn mặt với khẩu trang được đeo đúng cách" và "khuôn mặt với khẩu trang đeo không đúng cách" thay cho việc phát hiện khuôn mặt đầu tiên sau đó phân loại xem khuôn mặt có đeo khẩu trang hay không như ở MTCNN + ResNet18. Nghiên cứu này được thực hiện nhằm phát hiện việc đeo khẩu trang đúng quy định hay không thông qua việc huấn luyện mô hình YOLOv5 trên tập dữ liệu tùy chỉnh, đồng thời sử dụng các mô hình đã được huấn luyện sẵn để phát hiện và theo dõi đối tượng người trong ảnh, trích xuất tọa độ điểm cơ bản kết hợp sử dụng chuyển đổi Bird's-eye view, thuật toán Eculid làm cơ sở để tính khoảng cách giữa các đối tượng, khoảng cách xã hội cần tuân thủ là 2 m theo thông điệp 5K của Bộ Y tế về giữ khoảng cách khi tiếp xúc với người khác.

2. PHƯƠNG PHÁP NGHIÊN CỨU

2.1. Phát hiện vật thể với YOLO

Kiến trúc YOLO coi bài toán phát hiện vật thể như một bài toán hồi quy (*regression*). Từ hình ảnh đầu vào, qua một mạng gồm các lớp tích chập (*convolution*), tổng hợp (*pooling*) và kết nối đầy đủ (*fully connected*) có thể cho được kết quả đầu ra. Kiến trúc này có thể được tối ưu để chạy trên GPU với một lần chuyển tiếp (*forward pass*), vì thế nó đạt được tốc độ rất cao. Thuật toán YOLO cho độ chính xác cao và nhanh trong những bài toán phát hiện vật thể, do đó nó vô cùng thích hợp cho các ứng dụng thị giác máy tính.



Hình 1. Mô hình tổng quát của YOLO

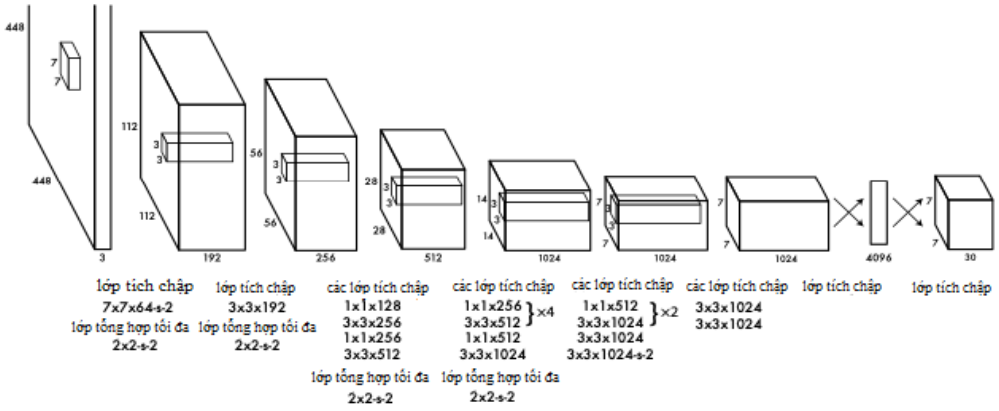
(Redmon et al., 2016)

YOLO chia ảnh thành một lưới các ô (*grid cell*) với kích thước SxS (mặc định là 7x7). Với mỗi ô (*grid cell*), mô hình sẽ đưa ra dự đoán cho B khung bao (*bounding box*). Ứng với mỗi hộp (*box*) trong B khung bao này sẽ là 5 tham số x, y, w, h, confidence, lần lượt là tọa độ tâm (x, y), chiều rộng (w), chiều cao (h) và độ tự tin (confidence) của dự đoán. Với mỗi ô (*grid cell*) trong lưới SxS, mô hình cũng dự

đoán xác suất rơi vào mỗi class theo công thức (1) (Hình 1) (Redmon et al., 2016).

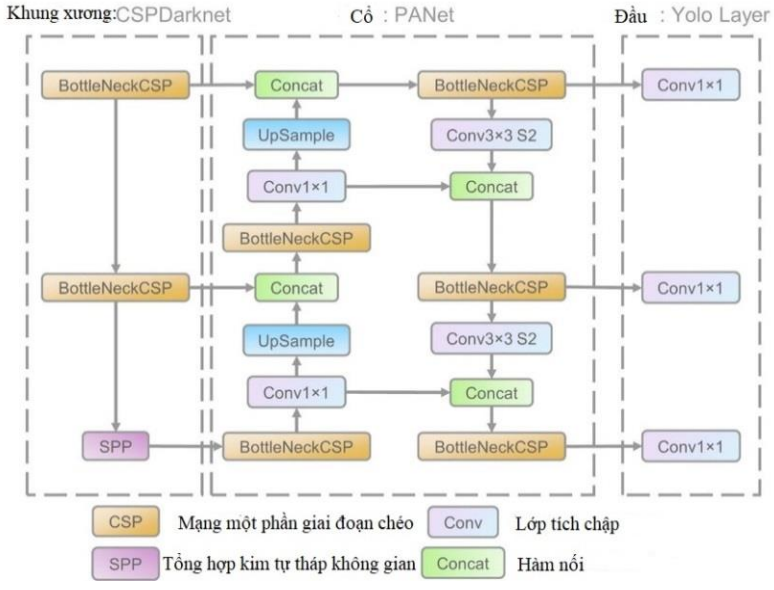
Trong thời gian qua, có nhiều phiên bản cải tiến so với kiến trúc YOLO ban đầu (Hình 2). Trong nghiên cứu này, phiên bản thứ năm YOLOv5 phát triển bởi công ty Ultralytics được sử dụng (Jocher, 2020). Kiến trúc mạng YOLOv5 được trình bày như Hình 3. Phiên bản này sử dụng Cross Stage Partial Network (*CSPNet*) (Wang et al., 2020) làm mô hình xương sống và Path Aggregation Network (*PANet*) (Liu et al., 2018) như phần để tổng hợp tính năng. Những cải tiến này đã dẫn đến việc trích xuất các đặc tính tốt hơn và tăng trung bình của điểm chính xác trung bình (mAP). Hiệu suất của một số mô hình huấn luyện sẵn của YOLOv5 được trình bày ở Bảng 1 (Ding et al., 2021).

$$\Pr(Class_i|Object) * \Pr(Object) * IOU\frac{truth}{pred} = \Pr(Class_i) * IOU\frac{truth}{pred} \quad (1)$$



Hình 2. Kiến trúc của mạng YOLO

(Redmon et al., 2016)



Hình 3. Tổng quan kiến trúc của YOLOv5

(Ding et al., 2021)

Bảng 1. Hiệu suất của một số mô hình huấn luyện sẵn của YOLOv5

Mô hình	Kích thước (pixels)	mAP^{val} 0,5:0,95	mAP^{val} 0,5	Tốc độ CPU b1 (ms)	Tốc độ V100 b1 (ms)	Tốc độ V100 b32 (ms)	Số lượng tham số (M)	FLOPs @640 (B)
YOLOv5n	640	28,4	46,0	45	6,3	0,6	1,9	4,5
YOLOv5s	640	37,2	56,0	98	6,4	0,9	7,2	16,5
YOLOv5m	640	45,2	63,9	224	8,2	1,7	21,2	49,0
YOLOv5l	640	48,8	67,2	430	10,1	2,7	46,5	109,1
YOLOv5x	640	50,7	68,9	766	12,1	4,8	86,7	205,7

(Jocher, 2020)

Bảng 1 cho thấy:

– **Kích thước (pixels)**: kích thước của ảnh trong tập huấn luyện. Giá trị 640 có nghĩa là kích thước ảnh là 640x640 px.

– **mAPval 0,5:0,95**: giá trị trung bình của độ chính xác trung bình (mean Average precision) tại IoU có giá trị từ 0,5 đến 0,95, xác định trên tập dữ liệu kiểm định (validation) tính bằng đơn vị phần trăm (%).

– **mAPval 0,5**: giá trị trung bình của độ chính xác trung bình (mean Average precision) tại IoU có giá trị 0,5, xác định trên tập dữ liệu kiểm định (validation) tính bằng đơn vị phần trăm (%).

– **Tốc độ CPU b1 (ms), Tốc độ V100 b1 (ms), Tốc độ V100 b32 (ms)**: đo lường thời gian xử lý cho mỗi dữ liệu hình ảnh trong tập dữ liệu kiểm định, chạy trên các phần cứng khác nhau từ CPU đến GPU NVIDIA Tesla V100. Đơn vị được tính bằng mili giây (ms).

– **Số lượng tham số(M)** (weights and biases): trong mô hình học sâu. Đơn vị tính là triệu (million). Mô hình YOLOv5s có giá trị số lượng tham số=7,2 nghĩa là 7,2 triệu tham số.

– **FLOPs @640 (B)**: chỉ số đo lường số lượng các phép toán với số thực dấu chấm động cần được thực hiện trên một mạng trong 1 giây (floating point operation per second) với kích thước hình ảnh đầu vào là 640x640 px, đơn vị tính là tỷ (billion) phép toán tử. Mô hình YOLOv5n thực hiện 4,5 tỷ phép tính với số thực dấu chấm động trong 1s.

Bảng 1 cho thấy mô hình YOLOv5n có độ chính xác thấp nhất so với các mô hình còn lại, nhưng tốc độ xử lý nhanh. Mô hình YOLOv5x có độ chính xác cao nhất, nhưng tốc độ xử lý chậm do mô hình có nhiều tham số và phép tính cần xử lý. Mô hình YOLOv5n, YOLOv5s phù hợp với các bài toán đòi hỏi về tốc độ xử lý theo thời gian thực, nhưng độ chính xác chỉ yêu cầu ở mức trung bình.

Môi trường huấn luyện (Jocher, 2020)

– Tất cả các mô hình được huấn luyện 300 epochs với các cài đặt và siêu tham số mặc định.

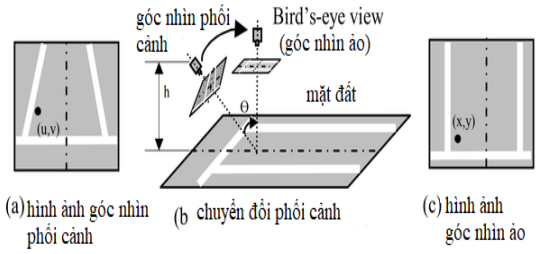
– Giá trị mAPval với mô hình đơn trên tập dữ liệu COCO val2017.

– Tốc độ được tính trung bình trên tập dữ liệu COCO val bằng cách sử dụng phiên bản AWS p3.2xlarge, không bao gồm thời gian NMS (~ 1 ms / img).

– Tăng cường thời gian kiểm tra (Test Time Augmentation) bao gồm tăng cường phản xạ và tỷ lệ.

2.2. Tính khoảng cách vật thể với phép chuyển đổi Bird's-eye view

Với khung cảnh phối cảnh nghiêng thông thường, tọa độ của các đối tượng quan sát bị ảnh hưởng bởi khoảng cách đến camera, do đó nếu tính toán khoảng cách dựa trên các tọa độ này thì sai số sẽ rất lớn. Bird's-eye view về cơ bản là góc nhìn từ trên xuống của một khung cảnh, giúp cải thiện chất lượng của việc tính khoảng cách giữa các đối tượng được theo dõi.



Hình 4. Minh họa về sự chuyển đổi góc nhìn trong bãi đậu xe

(Luo et al., 2010)

Hình ảnh chế độ xem bird's-eye là hình chiếu phối cảnh của hình ảnh gốc như trong Hình 4 (Luo et al., 2010). Mối quan hệ giữa (x, y) của hình ảnh xem qua bird's-eye view và (u, v) của hình ảnh gốc có thể được xây dựng bằng ma trận đồng nhất (*homography matrix*) 3 * 3, với x = x ' /w ' and y = y ' /w '.

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (2)$$

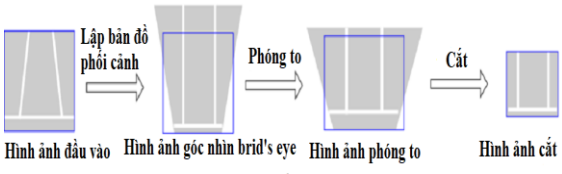
(O) **(H)** **(I)**

(I): Ma trận tọa độ đồng nhất của các điểm ảnh trên hình ảnh gốc, (x_g, y_g) là tọa độ của điểm ảnh trên hình ảnh gốc hai chiều với tọa độ đồng nhất là (x_g, y_g, 1) hay có thể được viết (u/w, v/w, 1) hoặc (u, v, w).

(H): Ma trận chuyển đổi đồng nhất.

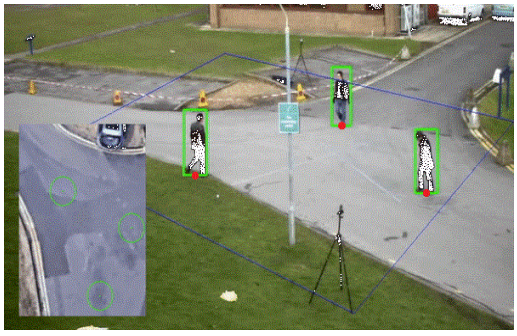
(O): Ma trận tọa độ đồng nhất của các điểm ảnh sau khi đã chuyển đổi thông qua việc nhân hai ma trận (H) và (I).

Sau khi nhận được ma trận chuyển đổi, đơn giản cách tiếp cận của hệ thống Bird's-eye view bao gồm 3 bước lập bản đồ phối cảnh, cắt và phóng to. Quy trình chuyển đổi được trình bày như Hình 5 (Luo et al., 2010).



Hình 5. Quy trình chuyển đổi đơn giản

(Luo et al., 2010)



Hình 6. Minh họa về việc chuyển đổi bird's-eye view, tính toán điểm cơ bản cho mỗi đối tượng người được phát hiện trong ảnh

Đối với mỗi người được phát hiện, 2 điểm cần thiết để xây dựng một hộp bao xung quanh đối tượng sẽ được trả về. Các điểm là góc trên cùng bên trái của hộp và góc dưới cùng bên phải. Từ những điểm này, tính toán tâm của hộp bằng cách lấy điểm giữa và tọa độ của điểm nằm ở tâm dưới cùng của hộp hay điểm cơ bản là đại diện tốt nhất cho tọa độ của một người trong một hình ảnh (Hình 6).

Dựa trên tập hợp tọa độ điểm cơ bản của các đối tượng người được phát hiện, tính khoảng cách giữa các cặp điểm với nhau. Các khoảng cách tính được có đơn vị là pixel, được tính bằng khoảng cách Eculid.

$$dist\left((x_i, y_i), (x_j, y_j)\right) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (3)$$

$$i, j = 1, \ldots, N$$

dist((x_i,y_i),(x_j,y_j)): Khoảng cách Eculid giữa 2 điểm cơ bản có tọa độ (x_i,y_i) và (x_j,y_j).

x_i,y_i: Tọa độ x, y của điểm cơ bản thứ i.

x_j,y_j: Tọa độ x, y của điểm cơ bản thứ j.

N: Số lượng các điểm cơ bản được tìm thấy

3. KẾT QUẢ VÀ THẢO LUẬN

3.1. Nhận diện người, tính khoảng cách xã hội

Tất cả các mô hình của YOLOv5 đã được huấn luyện với tập dữ liệu MS COCO (Common Objects in Context) phiên bản 2017 (Lin et al., 2014). Tập dữ liệu trên chứa 123.287 hình ảnh với tổng cộng 886.284 nhãn vật thể trong toàn bộ ảnh. Các mô hình YOLOv5 được huấn luyện để phát hiện 80 loại đối tượng khác nhau trong tập dữ liệu này. Danh sách vật thể nhận diện bao gồm người, xe ô tô, biển báo giao thông…

Hình 7. Lựa chọn 4 điểm để hình thành khung chuyển đổi với kích thước hình ảnh 1280x720 px

(Ghi chú: lần lượt các điểm (245, 320) – top-lef, (977, 320) – top-right, (1202, 720) – bottom-right, (16,720) – bottom-left)

Hình 8. Minh hoạ kết quả chuyển đổi bird's-eye view, chuyển đổi tọa độ điểm cơ bản của các đối tượng người theo góc nhìn mới

Nghiên cứu này sử dụng mô hình YOLOv5s đã được huấn luyện sẵn kết hợp ngôn ngữ lập trình python, thư viện OpenCV, Pytorch để phát hiện các vật thể trong hình ảnh, chỉ lấy những đối tượng có nhãn là người với chỉ số dự đoán $> \mathbf{0,55}$. Điểm giữa cạnh dưới của khung bao đối tượng được lấy làm cơ sở để tính khoảng cách xã hội. Kết quả hình ảnh đầu ra: khung bao quanh đối tượng màu xanh lá cây ứng với khoảng cách xã hội tối thiểu được đảm bảo, khung bao quanh các đối tượng người chuyển màu đỏ khi khoảng cách giữa các đối tượng nhỏ hơn khoảng cách cần đảm bảo. Một ví dụ minh họa được trình bày ở Hình 7 và Hình 8.

Để xác định được khoảng cách xã hội tối thiểu ta cần tuân thủ, tiến hành thực hiện các bước hiệu chỉnh. Thí nghiệm được bố trí với 2 người trong khung hình với camera chụp ảnh được đặt nghiêng và cố định tại một vị trí trong quá trình thu thập hình ảnh, kích thước hình ảnh đầu vào [1280 x 720 px–96 dpi]. Hai người trong khung hình lần lượt đổi các vị trí khác nhau đảm bảo giữ khoảng cách 2 m với nhau. Tiến hành chuyển đổi bird's-eye view, khoảng cách được tính từ điểm cơ bản ở các hình ảnh đầu vào đã thu thập được, lấy giá trị nhỏ nhất của các giá trị khoảng cách thu được, tính được giá trị khoảng cách tối thiểu cần đảm bảo theo quy định về khoảng cách xã hội. Vì vậy, các khoảng cách nhỏ hơn khoảng cách tối thiểu sẽ được tính là vi phạm (Hình 9).

Hình 9. Một số kết quả nhận diện người trong hình ảnh và khoảng cách xã hội

(Ghi chú: Khung bao quanh đối tượng chuyển màu đỏ cho biết khoảng cách xã hội bị vi phạm)

Thí nghiệm hiệu chỉnh thực tế này có số lượng mẫu thử N = 20 hình ảnh, mỗi hình ảnh chứa hình ảnh của 2 đối tượng người đứng cách nhau một khoảng cách, sử dụng thước để đo là 2 m trong thực tế. Trên mỗi hình ảnh đưa vào nhận dạng và tính khoảng cách từ các điểm cơ bản, 1 giá trị d_i đơn vị là pixels được thu. Giá trị nhỏ nhất của các giá trị d_i trên N số lượng mẫu thu được D_{min} được lấy. Với các giá trị đã đo được trong thực tế là 2 m, ta có thể kết luận được giá trị khoảng cách D_{min} xấp xỉ với khoảng cách 2 m trong thực tế. Kết quả được trình bày ở Bảng 2.

$$D_{min} = \min_i \{d_i\}_{i=1}^{N} \quad (4)$$

D_{min}: Khoảng cách nhỏ nhất tính từ điểm cơ bản (pixels).

d_i: Khoảng cách giữa 2 đối tượng người sau khi đã chuyển đổi bird's-eye view và tính khoảng cách bằng công thức Eculid trong mẫu thử thứ i (pixels).

N: Số lượng mẫu thử.

Bảng 2. Khoảng cách nhỏ nhất từ điểm cơ bản tính được giữa 2 người tương đương với khoảng cách 2 m trong thực tế

Số ảnh 2 người giữ khoảng cách 2 m trong thực tế (N)	Khoảng cách nhỏ nhất tính từ điểm cơ bản (D_{min}) (pixels)	Khoảng cách thực tế giữa 2 người (mm)
20	403,151565274844	2.000

Để đánh giá được độ chính xác của giá trị khoảng cách nhỏ nhất tính từ điểm cơ bản (Bảng 2) thu được thử thí nghiệm hiệu chỉnh, ta tiến hành thử nghiệm với tập hợp các dữ liệu mới, độc lập với dữ liệu sử dụng để hiệu chỉnh, chứa các hình ảnh của 2 đối tượng người đứng ở nhiều vị trí với khoảng cách khác nhau.

Tập dữ liệu thử nghiệm bao gồm 40 hình ảnh của 2 đối tượng người, đứng ở các vị trí khác nhau trong khung hình của máy chụp hình cố định về độ cao cũng như góc nghiêng trong thí nghiệm hiệu chỉnh. Các khoảng cách được đo đạc thực tế sau đó phân loại thành 2 nhóm chính. Nhóm 1 bao gồm 20 ảnh các đối tượng đứng cách nhau >= 2 m, nhóm 2 bao gồm 20 hình các đối tượng đứng cách nhau < 2 m. Kết quả ở Bảng 3 thu được từ việc chuyển đổi và tính khoảng cách giữa các điểm cơ bản giữa các đối tượng người được nhận diện trong tập dữ liệu thử nghiệm, khi khoảng cách nhỏ hơn khoảng cách trung bình tính có được từ hiệu chỉnh camera (Bảng 2) thì tính là nhỏ hơn 2 m, phân loại vào vi phạm khoảng cách xã hội.

Bảng 3. Kết quả phân loại từ chương trình với tập hợp hình ảnh đã được đo đạc trong thực tế

	Khoảng cách >= 2 m	Khoảng cách < 2 m
Thực tế	20	20
Dự đoán	18	18
Tỉ lệ	90%	90%

Kết quả dự đoán phân loại ở Bảng 3 tương đối tốt, tuy nhiên các yếu tố ảnh hưởng đến độ chính xác của việc tính khoảng cách phải kể đến sai số việc chọn tọa độ 4 điểm trong chuyển đổi Brid's eye-view. Một nguyên nhân khách quan khác khi cùng một vị trí đứng, mô hình sẽ tiến hành nhận dạng, vẽ khung bao sao cho gần như toàn bộ vật thể sẽ nội tiếp trong khung bao, vì vậy diện tích khung bao sẽ thay đổi với tư thế của đối tượng khi đứng yên, tay thả lỏng sẽ khác khi dang tay, chống tay lên phần hông, mở rộng chân, các tư thế khi đang di chuyển…, kéo theo tọa độ điểm cơ bản sẽ thay đổi, gây ra sai lệch trong tính toán khoảng cách Eculid. Ngoài ra, sai số của việc đo đạc thực tế cũng ảnh hưởng đến độ chính xác của kết quả.

3.2. Nhận diện việc đeo khẩu trang

Các mô hình YOLOv5 huấn luyện sẵn không phát hiện được đối tượng tùy chỉnh trong nghiên cứu này là khuôn mặt có đeo khẩu trang, không đeo khẩu trang và đeo sai. Vì vậy, việc huấn luyện lại với dữ liệu đầu vào là tập hợp các hình ảnh, kèm theo các file chú thích (*annotations*) chứa các thông tin về đối tượng cần phát hiện trong ảnh hay còn được gọi là dán nhãn đối tượng được tiến hành.

Tập dữ liệu về phát hiện khẩu trang được sử dụng trong nghiên cứu là tập dữ liệu đã được gán nhãn dữ liệu (*data labelling*) trên Kaggle tên là "Face Mask Detection" (Larxel, 2020), bao gồm 853 hình ảnh thuộc về 3 đối tượng: đeo khẩu trang (*with_mask*), không đeo khẩu trang (*without_mask*) và đeo sai (*mask_weared_incorrect*). Đối tượng đeo khẩu trang sai được xác định trong hình ảnh như sau: khuôn mặt người đeo khẩu trang để hở toàn bộ phần mũi (Hình 12(a)). Kích thước của ảnh đầu vào là [416x416] px. Từ tập dữ liệu 853 ảnh, để tăng độ chính xác, 683 hình ảnh được áp dụng tăng cường dữ liệu (*Data Augmentation*) có thể ngẫu nhiên thêm một phiên bản với kỹ thuật độ mờ Gauss ngẫu nhiên từ 0 đến 3,5 pixel (Hình 10), 170 hình ảnh được giữ

nguyên để sử dụng cho quá trình kiểm thử mô hình. Tổng số hình ảnh là 1.406, trong đó 1.236 hình ảnh được sử dụng trong quá trình huấn luyện mô hình (Hình 11).

Hình 10. Hình ảnh trong tập huấn luyện được tăng cường

(Ghi chú: (a) Hình ảnh đã tăng cường, (b) Hình ảnh gốc)

Hình 11. Mẫu dữ liệu huấn luyện trên tập dữ liệu Kaggle Face Mask Detection

Để việc huấn luyện mô hình diễn ra nhanh chóng, hiệu quả, máy tính có cấu hình mạnh được sử dụng, đặc biệt máy cần có GPU cho khả năng tính toán lớn. Trong nghiên cứu này, việc xây dựng, huấn luyện mô hình YOLOv5s được tiến hành trên Google Colaboratory, thường được gọi là "Google Colab" hoặc đơn giản là "Colab". Đây là một dự án nghiên cứu để tạo mẫu các mô hình máy học trên các tùy chọn phần cứng như GPU và TPU cung cấp một trình soạn thảo Jupyter serverless để phát triển tương tác. Google Colab được sử dụng miễn phí như các sản phẩm khác của G suite (Bisong, 2019).

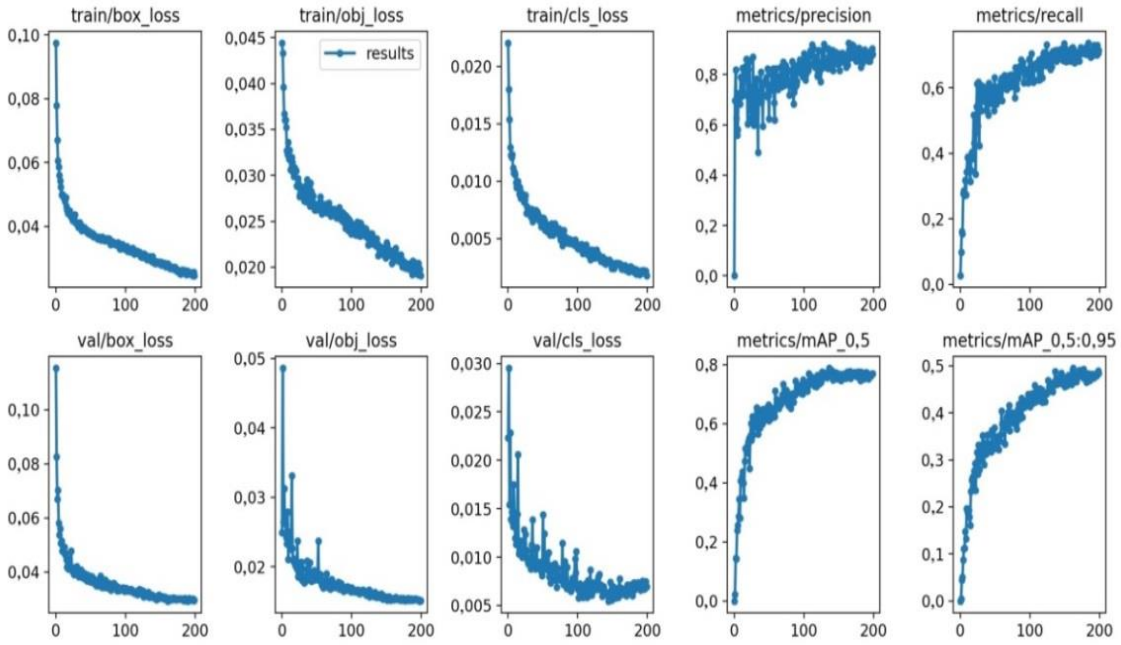
Mô hình YOLOv5s được huấn luyện trên môi trường Google Colab với 200 epoch trên tập dữ liệu hình ảnh tùy chỉnh. Kết quả huấn luyện nhận dạng của mô hình được trình bày ở Bảng 4 và Hình 13.

Bảng 4. Kết quả huấn luyện mô hình YOLOv5s

Độ tin cậy	Độ nhạy	mAP@0,5	mAP@0,5:0,95
0,88	0,715	0,77	0,484

Hình 12. So sánh các trạng thái đeo khẩu trang

(Ghi chú: (a) Đeo khẩu trang sai, (b) Đeo khẩu trang đúng)



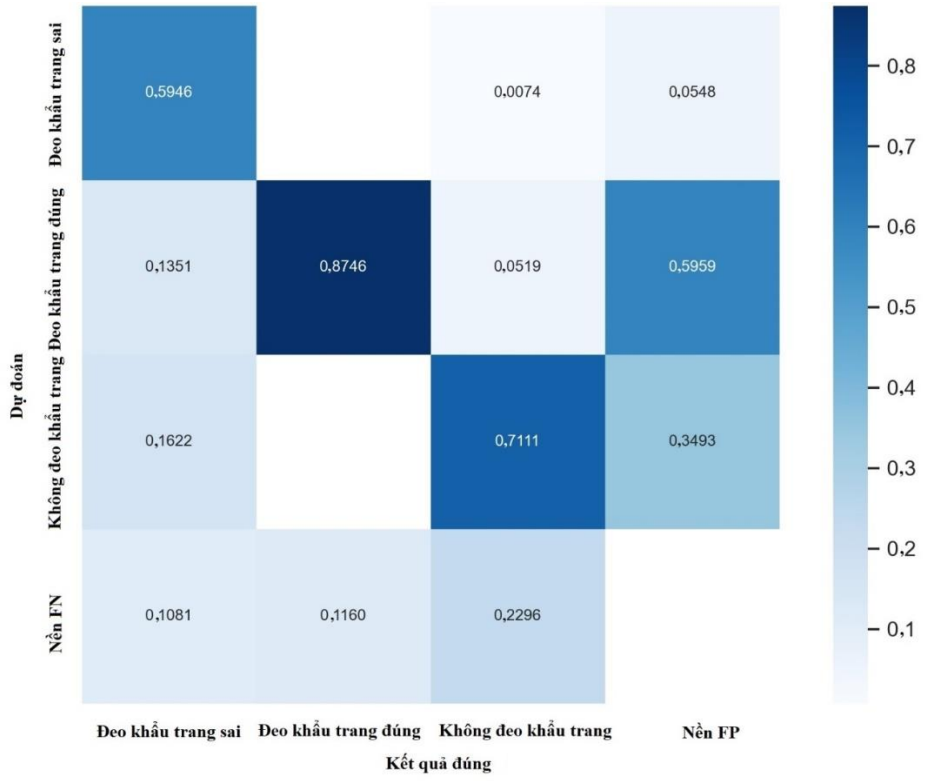
Hình 13. Kết quả huấn luyện mô hình nhận diện khuôn mặt đeo khẩu trang và không đeo khẩu trang

Các biểu đồ trong Hình 13 cho thấy sau 200 vòng lặp huấn luyện, các chỉ số mất mát giảm, độ tin cậy (*precision*), độ nhạy (*recall*), giá trị trung bình của độ chính xác trung bình (*mAPval 0,5; mAPval 0,5:0,95*) của mô hình đạt được các chỉ số trình bày trong Bảng 4.

Ba loại mất mát được trình bày trong Hình 12 là ất mát hộp bao vật thể (*box_loss*), mất mát vật thể (*obj_loss*) và mất mát phân loại (*cls_loss*). Mất hộp bao vật thể biểu thị mức độ tốt của thuật toán có thể xác định vị trí trung tâm của một đối tượng và hộp giới hạn được dự đoán bao phủ một đối tượng tốt như thế nào. Mất mát vật thể về cơ bản là thước đo xác suất một đối tượng tồn tại trong một khu vực quan tâm được đề xuất, về khách quan điều này có nghĩa là hình ảnh có khả năng chứa một đối tượng. Mất mát phân loại cung cấp ý tưởng về cách thuật toán có thể dự đoán đúng lớp của một đối tượng đã cho. Các chỉ số mất mát càng thấp thì độ tin cậy của mô hình đã huấn luyện càng cao.

Theo kết quả Bảng 4, độ tin cậy, độ nhạy, mAP@0,5, mAP@0,5:0,95 của mô hình đạt chỉ số tương đối tốt. Điều này cho thấy việc huấn luyện với tập dữ liệu được tăng cường đạt hiệu quả, do mô hình được sử dụng để huấn luyện là YOLOv5s, với cấu trúc đơn giản, số lượng tham số thấp hơn, độ chính xác thấp, ưu tiên về tốc độ xử lý so với các mô hình YOLOv5m, YOLOv5l, YOLOv5x. Mô hình có khả năng nhận diện tốt với các hình ảnh đầu vào trong thực tế.

Ứng dụng nhận diện được xây dựng dựa trên mô hình đã huấn luyện với cài đặt độ tự tin (*confidence*) trong dự đoán của các đối tượng là **0,25** và IOU = **0,45**. Ngôn ngữ lập trình python, thư viện OpenCV, Pytorch được sử dụng. Kết quả hình ảnh là khung bao màu xanh lá cây đối với việc đeo khẩu trang đúng cách, khung màu cam chỉ việc đeo khẩu trang sai và khung màu đỏ là không đeo khẩu trang. Mức độ tự tin khi mô hình đưa ra dự đoán đạt từ **0,65** đến **0,95** (Hình 14).



Hình 14. Ma trận nhận dạng các đối tượng trong tập dữ liệu thử nghiệm gồm 170 hình ảnh

Môi trường chạy thử nghiệm là máy tính cá nhân với cấu hình chính CPU Core-i5, không có GPU, 8 GB RAM. Chương trình thực nghiệm cho kết quả về việc kiểm tra đeo khẩu trang được trình bày ở Bảng 5.

Ma trận nhận dạng ở Hình 14 cho thấy tỷ lệ nhận dạng đúng đối với từng loại đối tượng so sánh giữa nhãn dự đoán và nhãn trong thực tế. Dựa vào giá trị của đường chéo ta có kết quả như Bảng 5.

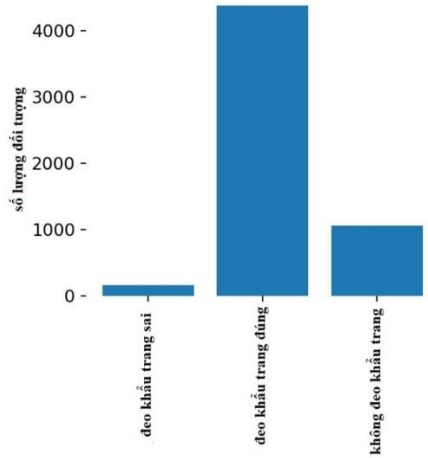
Nền FP (*False Positive*) cho biết tỷ lệ dự đoán sai lệch khi mô hình dự đoán đối tượng là một phần hình nền trong hình ảnh nhưng đối tượng đó không thuộc hình nền.

Nền FN (*False Negative*) cho biết tỷ lệ sai lệch một cách gián tiếp khi mô hình dự đoán đối tượng không phải là một phần của hình nền nhưng kết quả đúng là đối tượng đó thuộc một phần của hình nền trong hình ảnh.

Bảng 5. Kết quả nhận dạng các đối tượng trong tập dữ liệu thử nghiệm

Đeo khẩu trang sai	Đeo khẩu trang đúng	Không đeo khẩu trang
59,46%	87,46%	71,11%

Kết quả nhận diện (Bảng 5) của nhãn đối tượng đeo khẩu trang sai tương đối thấp do dữ liệu về đối tượng đeo khẩu trang sai trong tập huấn luyện của mạng YOLOv5s thấp hơn nhiều so với các nhãn còn lại (Hình 15), dẫn đến việc nhận dạng nhầm sang các nhãn đối tượng khác. Mô hình có khả năng dự đoán được nhiều đối tượng khác nhau trong cùng 1 hình ảnh (Hình 16).

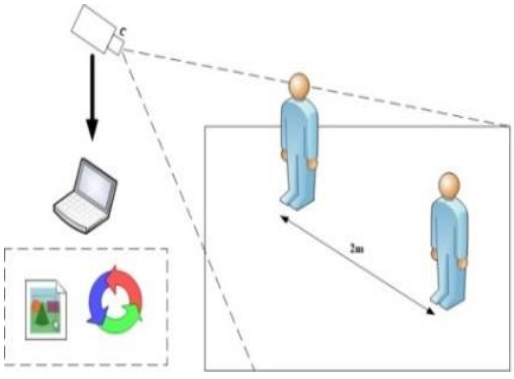


Hình 15. Thống kê về nhãn các đối tượng trong quá trình huấn luyện

Hình 16. Nhận diện được nhiều đối tượng thuộc các nhãn khác nhau

3.3. Kết hợp tính khoảng cách xã hội và phát hiện việc đeo khẩu trang

Hai mô hình được sử dụng, trong đó một mô hình YOLOv5s đã huấn luyện trong việc phát hiện việc đeo khẩu trang, kết hợp với mô hình YOLOv5s đã được huấn luyện sẵn với tập dữ liệu MS COCO (Lin et al., 2014) để xây dựng ứng dụng có khả năng phát hiện việc vi phạm khoảng cách xã hội và phát hiện việc đeo khẩu trang (Hình 17).



Hình 17. Mô hình thí nghiệm của ứng dụng

Việc sử dụng 2 mô hình thay vì huấn luyện một mô hình nhận diện 4 lớp bao gồm đối tượng người, đeo khẩu trang đúng, đeo khẩu trang sai, không đeo khẩu trang giúp tận dụng được các mô hình đã được huấn luyện sẵn, đồng thời tiết kiệm được tài nguyên phần cứng, thời gian huấn luyện. Khi có sự thay đổi, bổ sung dữ liệu trong tập huấn luyện, hoặc thử nghiệm kết hợp với mô hình khác để thực hiện một trong hai yêu cầu tính khoảng cách xã hội hay nhận diện đeo khẩu trang, lúc đó không cần phải huấn luyện lại toàn bộ. Tuy nhiên, nhược điểm của phương pháp này là tốc độ xử lý sẽ không thể nhanh hơn khi sử dụng một mô hình cho cả 4 lớp đối tượng nhận dạng, vì mỗi hình ảnh đầu vào phải trải qua 2 lần nhận dạng ở 2 mô hình độc lập nối tiếp nhau.

Phương pháp thông thường để kết hợp 2 mô hình độc lập là lập trình tuần tự, hình ảnh đầu vào lần lượt đi qua từng mô hình để phân loại, sau đó tổng hợp kết quả ra một hình ảnh đầu ra. Ưu điểm của phương pháp trên là lập trình đơn giản, tuy nhiên tốc độ xử lý sẽ phụ thuộc vào tổng thời gian xử lý hình ảnh của mỗi mô hình.

Hình 18. Sơ đồ xử lý hình ảnh của ứng dụng

Nhằm khắc phục một phần hạn chế của việc kết hợp 2 mô hình, tăng tốc độ xử lý, ứng dụng sử dụng kỹ thuật lập trình đa luồng (*Multithreaded Programming*) (Hình 18). Với cùng một hình ảnh đầu vào, hai mô hình YOLOv5s sẽ xử lý trên 2 luồng riêng biệt, chạy song song với nhau trong ứng dụng. Khi cả hai luồng đã xử lý xong, kết quả cuối cùng sẽ được tổng hợp trên một hình ảnh đầu ra (Hình 19).

Để đánh giá hiệu quả của việc kết hợp hai mô hình YOLOv5s khi sử dụng phương pháp lập trình tuần tự và áp dụng kỹ thuật lập trình đa luồng, ta tiến hành thử nghiệm với tập dữ liệu 40 hình ảnh đã được sử dụng trong thử nghiệm phân loại khoảng cách xã hội (mục 3.1). Mô hình đang sử dụng trong bài báo là YOLOv5s thuộc bài toán phát hiện vật thể (*Object detection*), do đó trong một hình ảnh có chứa nhiều vật thể khác nhau, mô hình sẽ xác định vị trí của vật thể bằng cách vẽ khung bao xung quanh vị trí của vật thể trong hình ảnh, đồng thời phân loại đối tượng đó thuộc nhãn nào. Trở lại với tập dữ liệu được sử dụng trong thử nghiệm là 40 hình ảnh, mỗi hình ảnh có 2 người, mỗi người đeo khẩu trang thuộc 1 trong 3 trạng thái: đeo khẩu trang đúng, đeo khẩu trang sai, không đeo khẩu trang. Vì vậy, tổng số đối tượng trong toàn bộ tập thử nghiệm để mô hình nhận diện đeo khẩu trang phát hiện và phân loại là $2 \times 40 = 80$.

Kết quả thử nghiệm phân loại đối tượng ở cả 2 phương pháp là giống nhau, do sử dụng chung 2 mô hình YOLOv5s. Kết quả phân loại khoảng cách xã hội giống với thử nghiệm về phân loại khoảng cách

xã hội (mục 3.1) được thể hiện trong Bảng 3. Kết quả nhận diện đeo khẩu trang thể hiện trong Bảng 6.

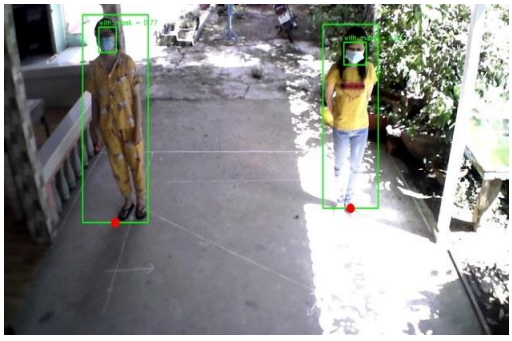
Bảng 6. Kết quả nhận diện đối tượng đeo khẩu trang

Nhãn đối tượng	Dự đoán	Thực tế	Tỷ lệ
Đeo khẩu trang đúng	26	30	86,67%
Không đeo khẩu trang	23	30	76,67%
Đeo khẩu trang sai	13	20	65,0%

Kết quả nhận diện (Bảng 6) và hình ảnh trong tập dữ liệu thử nghiệm cho thấy mô hình nhận diện tốt khi khuôn mặt người hướng về phía chính diện của máy ảnh, các trường hợp khuôn mặt hay cụ thể là tư thế đứng, nhìn ngang so với máy ảnh, diện tích phần khuôn mặt được ghi nhận vào hình ảnh càng nhỏ, thì tỷ lệ mô hình không phát hiện được càng lớn dẫn đến không nhận diện trong một số trường hợp.

Bảng 7. Kết quả so sánh thời gian xử lý trung bình giữa chương trình sử dụng phương pháp tuần tự và kỹ thuật lập trình đa luồng

	Phương pháp tuần tự	Kỹ thuật đa luồng
Thời gian xử lý trung bình với 1 hình ảnh (s)	0,55	0,28



Hình 19. Nhận diện đồng thời khoảng cách xã hội và việc đeo khẩu trang

Do kết quả phân loại ở hai phương pháp là giống nhau nên để đánh giá tính hiệu quả, ta cần đo thời gian xử lý trung bình với mỗi hình ảnh trong tập dữ liệu thử nghiệm, bằng cách chèn thêm phần mã đo thời gian trong mỗi chương trình. Bảng 7 thể hiện kết quả so sánh thời gian được thực hiện bởi 2 phương pháp, tính bằng giây (s).

Kết quả ở Bảng 7 cho thấy các mô hình YOLOv5s có tốc độ xử lý tương đối nhanh, khi áp dụng kỹ thuật lập trình đa luồng thời gian xử lý trên mỗi hình ảnh được giảm đi đáng kể so với phương pháp lập trình tuần tự.

4. KẾT LUẬN

4.1. Kết luận

Nghiên cứu đã cung cấp một phương thức theo dõi và phát hiện việc thực hiện các biện pháp phòng chống dịch Covid-19 thông qua việc giữ khoảng cách xã hội và đeo khẩu trang. Dữ liệu đầu vào là hình ảnh kết hợp kỹ thuật học sâu với thuật toán YOLO cho độ chính xác cao và tốc độ xử lý nhanh. Việc phát hiện mang khẩu trang đúng quy định được thực hiện bởi mô hình YOLOv5s được huấn luyện lại để nhận diện 3 nhãn đối tượng là: đeo khẩu trang, không đeo khẩu trang và đeo sai với mAp@0,5 đạt 0,77. Mô hình YOLOv5s huấn luyện sẵn trên tập dữ liệu MS COCO được sử dụng để phát hiện đối tượng người trong hình ảnh, kết hợp với các thuật toán để tính được khoảng cách giữa các đối tượng.

4.1. Đề xuất

Thứ nhất, nghiên cứu tăng độ dự đoán chính xác, tốc độ phát hiện đối tượng trên dữ liệu đầu vào là video, dữ liệu thời gian thực từ webcam, camera cần tiếp tục thực hiện.

Thứ hai, trong nghiên cứu đo khoảng cách xã hội giữa các đối tượng người, thay vì sử dụng phương pháp tính khoảng cách trung bình theo pixel giữa các cặp tọa độ cơ bản sau đó suy luận ra khoảng cách thực tế, thì nghiên cứu chuyển sang sử dụng phương pháp phân lớp đối tượng dựa trên việc huấn luyện mạng nơ-ron nhân tạo (Artificial Neural Network - ANN) với dữ liệu đầu vào là các cặp tọa độ cơ bản sau khi đã chuyển đổi bird's-eye view để đưa ra dự đoán với 2 nhãn: đủ khoảng cách xã hội và không đủ khoảng cách xã hội.

Thứ ba, đối với nghiên cứu nhận diện đeo khẩu trang theo quy định, ta cần bổ sung dữ liệu thuộc 2 tập hợp có nhãn là: đeo khẩu trang sai quy định, không đeo khẩu trang, nhằm tăng độ chính xác của mô hình trong quá trình huấn luyện cũng như thực nghiệm.

TÀI LIỆU THAM KHẢO

Bisong, E. (2019). Google Colaboratory. In E. Bisong (Ed.), *Building Machine Learning and Deep Learning Models on Google Cloud Platform* (pp. 59–64). Apress. https://doi.org/10.1007/978-1-4842-4470-8_7

Ding, Y., Li, Z., & Yastremsky, D. (2021). *Real-time Face Mask Detection in Video Data*. arXiv:2105.01816 [cs.CV] 5 May 2021. http://arxiv.org/abs/2105.01816

Jocher, G. (2020). *YOLOv5*. https://doi.org/10.5281/zenodo.5563715

Kumar, G., & Shetty, S. (2021). Application Development for Mask Detection and Social Distancing Violation Detection using Convolutional Neural Networks. *Proceedings of the 23rd International Conference on Enterprise Information Systems*, 760–767. https://doi.org/10.5220/0010483107600767

Larxel. (2020). Face Mask Detection | Kaggle. In *Www.Kaggle.Com*. https://www.kaggle.com/andrewmvd/face-mask-detection

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common Objects in Context. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics): Vol. 8693 LNCS* (Issue PART 5, pp. 740–755). https://doi.org/10.1007/978-3-319-10602-1_48

Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J. (2018). Path Aggregation Network for Instance Segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 8759–8768. https://doi.org/10.1109/CVPR.2018.00913

Luo, L. B., Koh, I. S., Min, K. Y., Wang, J., & Chong, J. W. (2010). Low-cost implementation of bird's-eye view system for camera-on-vehicle. *2010 Digest of Technical Papers International Conference on Consumer Electronics (ICCE)*, 311–312. https://doi.org/10.1109/ICCE.2010.5418845

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779–788. https://doi.org/10.1109/CVPR.2016.91

Shukla, R., Mahapatra, A. K., & Selvin Peter, P. J. (2021). Social distancing tracker using YOLOv5. *Turkish Journal of Physiotherapy and Rehabilitation*, *32*(2), 1785-1793.

Wang, C. Y., Mark Liao, H. Y., Wu, Y. H., Chen, P. Y., Hsieh, J. W., & Yeh, I. H. (2020). CSPNet: A New Backbone that can Enhance Learning Capability of CNN. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 1571–1580. https://doi.org/10.1109/CVPRW50498.2020.00