PHÁT HIỆN NGƯỜI TRONG VIDEO GIÁM SÁT VỚI YOLOV8

Lê Quang Chiến

Khoa Công nghệ Thông tin, Trường Đại học Khoa học, Đại học Huế

Email: lqchien@husc.edu.vn

Ngày nhận bài: 9/6/2023; ngày hoàn thành phản biện: 23/6/2023; ngày duyệt đăng: 26/6/2023

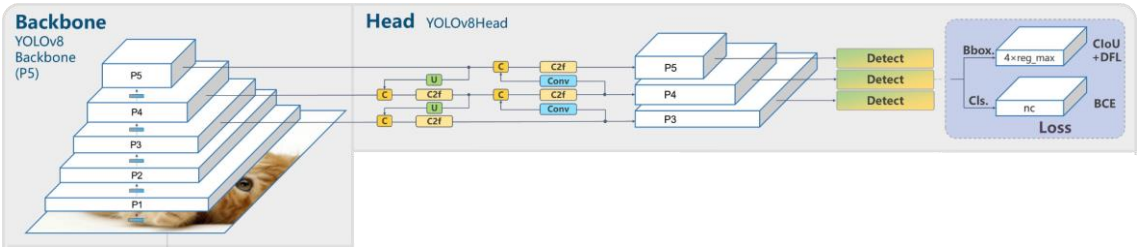
TÓM TẮT

Bài báo này trình bày một mô hình tổng quát để nhận dạng tự động người dựa trên các bức ảnh thu được từ camera giám sát. Mô hình này có thể được sử dụng cho việc học chuyển đổi và nghiên cứu sâu hơn, cũng như tinh chỉnh để triển khai cho các dự án cụ thể trong thực tế. Đầu tiên, chúng tôi huấn luyện mô hình trên tập dữ liệu được cung cấp sẵn để mô hình làm quen với đặc trưng của đối tượng quan tâm. Tiếp theo, chúng tôi thực hiện quá trình học chuyển với các tham số đã học này trên tập dữ liệu được thu thập từ môi trường "thế giới thực". Quá trình này giúp mô hình được tinh chỉnh để thích nghi tốt hơn với các điều kiện thách thức trong thực tế. Thêm vào đó, để tối đa hóa hiệu suất của mô hình, chúng tôi khai thác YOLOv8, một trong những mô hình phát hiện đối tượng tiên tiến đã được công bố gần đây. Các kết quả thí nghiệm cho thấy tính khả thi của cách tiếp cận này khi được sử dụng cho giám sát video trong thành phố thông minh.

Từ khóa: Phát hiện đối tượng, YOLO, học chuyển đổi, giám sát video.

1. MỞ ĐẦU

Trong những năm gần đây, bài toán phát hiện người trong các video giám sát đang trở nên cấp thiết trong nhiều lĩnh vực: giáo dục, an ninh và quản lý giao thông. Ở Việt Nam, bài toán này càng trở nên cấp thiết hơn khi số lượng camera giám sát được lắp đặt tại các đô thị, khu công nghiệp, bệnh viện, và trường học đang ngày càng tăng lên. Vì vậy, việc sử dụng video giám sát giúp cho việc quản lý giáo dục, an ninh, an toàn giao thông và chống tội phạm trở nên hiệu quả hơn. Tuy nhiên, để có thể giám sát và xác định được hành vi của con người một cách nhanh chóng và chính xác, thì việc kết hợp hệ thống camera giám sát này với thuật toán phát hiện đối tượng để phát hiện và xác định người hiệu quả là vô cùng cần thiết.



Hình 1. Kiến trúc tổng quan của YOLOv8.

Dựa trên những công bố gần đây, YOLOv8 [3] là mô hình kiến trúc một giai đoạn đạt được hiệu quả cao trên cả hai khía cạnh độ chính xác và tốc độ xử lý. Trong bài báo này, chúng tôi sẽ tìm hiểu và đánh giá trên cả hai khía cạnh này. Các đánh giá của các mô hình này được thực hiện trên tập dữ liệu thu được từ camera giám sát thực tế kết hợp với dữ liệu được thu thập từ internet. Các kết quả thí nghiệm chỉ ra được tính khả thi của cách tiếp cận này cho bài toán phát hiện người trong các hệ thống giám sát thông minh.

2. PHƯƠNG PHÁP NGHIÊN CỨU

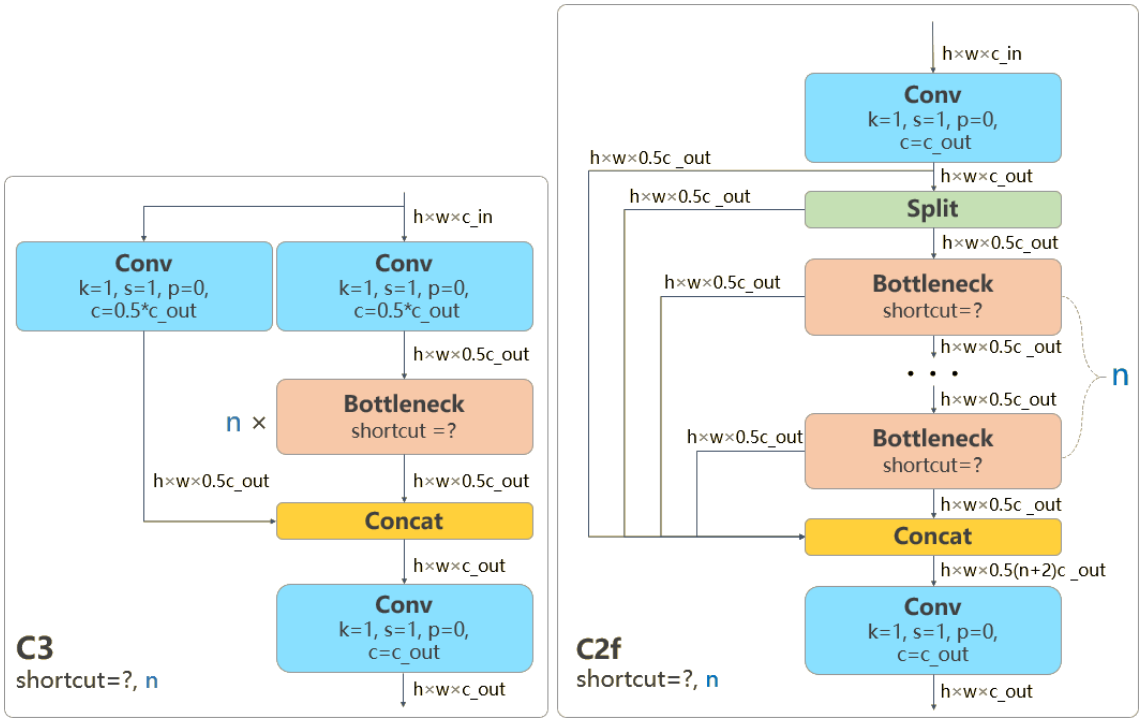
Phát hiện đối tượng là một trong những bài toán quan trọng trong thị giác máy tính với mục tiêu là để nhận biết một đối tượng là gì và ở đâu. Hiện tại, kiến trúc phát hiện đối tượng một giai đoạn là sự lựa chọn thực tế để xây dựng các mô hình với tốc độ suy luận nhanh. Đại diện nổi bật cho kiểu kiến trúc này là YOLO (You Only Look Once). Kiến trúc YOLO giúp cho nhiệm vụ phát hiện đối tượng đạt hiệu suất tốt hơn cả về độ chính xác và tốc độ xử lý. Cho đến hiện tại, kiến trúc YOLO đã tiến hóa qua nhiều phiên bản: từ YOLOv1 đến YOLOv8. Trong bài báo này, chúng tôi tìm hiểu những điểm mới ở phiên bản YOLOv8 và phương pháp học chuyển đổi với mô hình YOLOv8 đã được huấn luyện trước đó.

2.1. Kiến trúc YOLOv8

Kiến trúc của YOLOv8 được xây dựng dựa trên phiên bản trước của mô hình phát hiện đối tượng YOLOv5. Kiến trúc này có thể được chia thành hai phần chính: xương sống (backbone) và phần đầu (head) (Xem hình 1).

Phần backbone của YOLOv8 về cơ bản giống với phần backbone của YOLOv5, điểm khác là mô-đun C3 được thay thế bằng mô-đun C2f dựa trên ý tưởng CSP (Cross-Stage-Partial [1]) (Xem hình 2). Trong đó, mô-đun C2f được xây dựng dựa trên sự kết hợp đồng thời C3 và ELAN (Efficient Layer Aggregation Network [2]). Mục đích của ý tưởng này là giúp YOLOv8 có thể thu được nhiều thông tin về luồng gradient hơn trong khi vẫn đảm bảo bộ trọng số nhẹ. Ở phần cuối của backbone, mô-đun SPPF phổ biến nhất vẫn được sử dụng và ba lớp MaxPool có kích thước 5×5 được truyền nối tiếp, sau

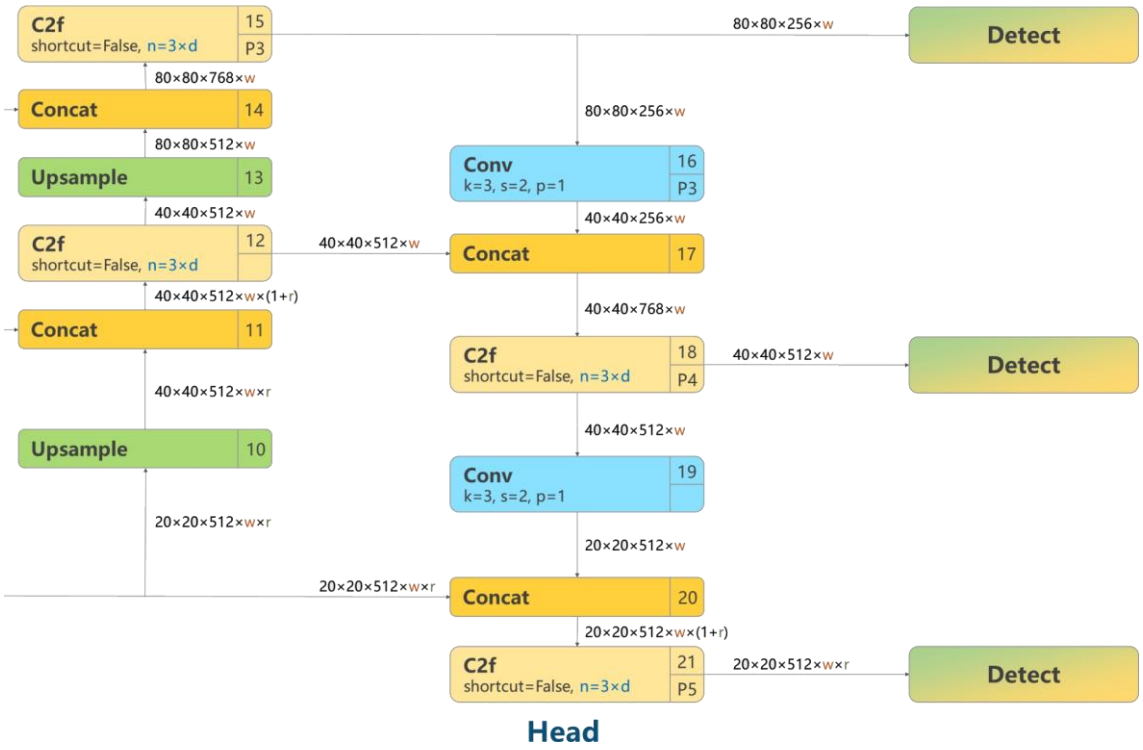
đó mỗi lớp được ghép nối, mục tiêu là để đảm bảo độ chính xác ở các tỷ lệ khác nhau đồng thời đảm bảo thu gọn đáng kể số lượng tham số.



Hình 2. So sánh mô-đun C3 của YOLOv5 và mô-đun C2f của YOLOv8.

Trong phần head, phương pháp hợp nhất đặc trưng được YOLOv8 sử dụng vẫn là PAN (Path Aggregation Network [5]) và FPN (Feature Pyramid Networks [4]). Hai thuật toán này giúp tăng cường khả năng hợp nhất và sử dụng thông tin lớp đặc trưng ở các tỷ lệ khác nhau. Ngoài ra, hai mô-đun upsampling kết hợp xen kẽ với các mô-đun C2f để tạo thành các mô-đun Detect ở các tỷ lệ khác nhau (Xem hình 3).

Ngoài một số cập nhật về kiến trúc, YOLOv8 sử dụng phương pháp Anchor-Free Detection để dự đoán trực tiếp tâm của đối tượng thay vì độ lệch (offset) từ anchor box đã biết. Các anchor box là một tập hợp các hộp được xác định trước với chiều cao và chiều rộng cụ thể. Chúng được sử dụng để phát hiện các lớp đối tượng có tỷ lệ co giãn và tỷ lệ kích thước mong muốn. Chúng được chọn dựa trên kích thước của các đối tượng trong tập dữ liệu huấn luyện. Trong quá trình phát hiện, các anchor box này sẽ được lát trên toàn bộ ảnh. Mô hình mạng sẽ dự đoán xác suất và các thuộc tính khác, như nền, IoU (Intersection over Union) và offset cho mỗi anchor box được lát. Các dự đoán này sau đó sẽ được sử dụng để tinh chỉnh cho từng anchor box riêng lẻ. Thêm vào đó, các anchor box này có thể được định nghĩa tương ứng với các kích thước đối tượng khác nhau. Phương pháp Anchor-Free Detection giúp làm giảm số lượng dự đoán, qua đó tăng tốc thuật toán NMS (Non-Maximum Suppression), một bước xử lý hậu kỳ phức tạp để sàng lọc các phát hiện ứng viên sau khi suy luận.



Hình 3. Kiến trúc chi tiết phần Head của YOLOv8.

Thêm vào đó, để mô hình có thể thích ứng với những biến thể của đối tượng trong thực tế, YOLOv8 tăng cường dữ liệu hình ảnh trong quá trình huấn luyện. Tại mỗi thời điểm, mô hình sẽ thấy một biến thể hơi khác nhau của hình ảnh mà nó đã được cung cấp. Một trong những phương pháp tăng cường dữ liệu như vậy được gọi là Mosaic Augmentation. Đây là một kỹ thuật gia tăng dữ liệu đơn giản, trong đó đầu vào của mô hình được tạo thành bằng việc ghép bốn hình ảnh khác nhau. Cách gia tăng dữ liệu này có thể khiến cho mô hình học được các đối tượng ở các vị trí khác nhau trong điều kiện bị che khuất một phần. Tuy nhiên, cách gia tăng này được chứng minh bằng thực nghiệm là làm giảm hiệu quả của mô hình nếu được thực hiện trong toàn bộ quá trình luyện tập. Vì vậy, cách gia tăng dữ liệu này sẽ không được sử dụng trong 10 vòng lặp huấn luyện cuối cùng.

2.2. Học chuyển đổi với YOLOv8

Học chuyển đổi đề cập đến việc sử dụng một mô hình CNN đã được huấn luyện trước trên một lượng lớn dữ liệu để có thể giải quyết nhiệm vụ T1, thay thế một hoặc nhiều lớp trong mô hình CNN này và sau đó huấn luyện lại mô hình này trên một tập dữ liệu khác, nhỏ hơn để có thể để giải quyết nhiệm vụ T2. Ví dụ như, giả sử chúng ta đã huấn luyện một mô hình CNN có thể phát hiện được nhiều loại đối tượng trên một tập dữ liệu rất lớn. Sau đó, chúng ta hiệu chỉnh lớp đầu ra chỉ để phát hiện một loại đối tượng cụ thể và huấn luyện lại mô hình này với một tập dữ liệu nhỏ hơn nhiều. Do vậy,

học chuyển là một cách hữu ích để nhanh chóng huấn luyện lại một mô hình trên dữ liệu mới mà không cần phải huấn luyện lại toàn bộ mạng.

Có hai cách tiếp cận chính khi nói đến phương pháp học chuyển đổi: tinh chỉnh và trích xuất đặc trưng cố định. Cả hai đều xoay quanh việc thay đổi trọng số trong mạng trong khi huấn luyện, tuy nhiên cách thức mà hai cách tiếp cận thực hiện việc thay đổi này là hoàn toàn khác nhau. Khi sử dụng trích xuất đặc trưng cố định, cách tiếp cận này sẽ đóng băng các lớp đầu tiên trong mạng và chỉ huấn luyện các lớp cuối cùng. Ngược lại, khi sử dụng tinh chỉnh, mạng được huấn luyện bình thường, trong đó tất cả các trọng số được phép thay đổi. Cả hai cách tiếp cận này vẫn thay thế một số lớp trong mạng trước khi huấn luyện.

Trong nghiên cứu này, chúng tôi áp dụng phương pháp học chuyển đổi dựa trên các trọng số được huấn luyện trước từ tập dữ liệu Microsoft Common Objects in Context (MS-COCO) để cải thiện hiệu suất của mô hình. Tập dữ liệu MS-COCO là một bộ dữ liệu hình ảnh quy mô lớn có chứa các chú thích cho phép người dùng huấn luyện các mô hình thị giác máy tính để nhận dạng, gắn nhãn và mô tả các đối tượng. Ngoài ra, tập dữ liệu MS-COCO bổ sung cho quy trình học chuyển đổi trong đó dữ liệu được áp dụng cho một mô hình đóng vai trò là điểm ban đầu cho một mô hình khác. Tập dữ liệu MS-COCO còn là một tiêu chuẩn quan trọng cho thị giác máy tính để huấn luyện, kiểm tra và tinh chỉnh mô hình phát hiện đối tượng.

3. KẾT QUẢ VÀ THẢO LUẬN

Trong phần này, chúng tôi trình bày các thiết lập thí nghiệm để thực hiện đánh giá năm phiên bản của mô hình YOLOv8: YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l và YOLOv8x. Bên cạnh đó, chúng tôi cũng báo cáo các kết quả thí nghiệm đã tiến hành. Cuối cùng, chúng tôi phân tích, thảo luận dựa trên các kết quả đã báo cáo để đánh giá hiệu quả của từng phiên bản khi áp dụng cho hệ thống giám sát thực tế.

3.1. Các thiết lập thí nghiệm

Chúng tôi thực hiện đánh giá các phiên bản YOLOv8 trước và sau học chuyển đổi trên tập dữ liệu được thu thập từ các camera giám sát ở trong phòng học của một trường mẫu giáo. Trước khi học chuyển đổi, chúng tôi sử dụng các phiên bản YOLOv8 đã được huấn luyện với tập dữ liệu MS-COCO [6]. Tập dữ liệu này bao gồm hơn 330.000 hình ảnh, với hơn 200.000 hình ảnh được gán nhãn dữ liệu phục vụ cho các bài toán: phát hiện, phân đoạn và chú thích đối tượng. Tập dữ liệu này bao gồm 80 lớp đối tượng khác nhau, chứa các đối tượng phổ biến như người, ô tô, xe đạp và động vật, cũng như một số đối tượng đặc biệt như ô, túi xách và trang bị thể thao. Các phiên bản của YOLOv8 trước khi học chuyển đổi được huấn luyện từ đầu với hơn 118.000 ảnh dữ liệu.

Trong phần học chuyển đổi, chúng tôi sử dụng tập dữ liệu được thu thập từ các camera giám sát trong các lớp học mẫu giáo (Xem hình 4). Tập dữ liệu gồm 72 ảnh được gắn nhãn chỉ cho một lớp đối tượng là con người và được chia thành 2 tập huấn luyện (30 ảnh) và tập kiểm thử (42 ảnh). Trong pha huấn luyện của việc học chuyển đổi, chúng tôi sử dụng cùng chung một thiết lập gồm: (1) kích thước đầu vào: 640×640; (2) kích thước batch: 4; (3) số vòng lặp cần thực thi: 200; (4) optimizer sử dụng thuật toán Adam với momentum 0.937; (5) learning rate: 0.001 và (6) weight decay: 0.0005.

Hình 4. Một số hình ảnh minh họa tập dữ liệu được thu thập từ camera giám sát.

Chúng tôi cung cấp các đánh giá liên quan dựa trên các độ đo như: Precision, Recall, F1-score, mAP và tốc độ xử lý của các phiên bản. Các kết quả được báo cáo trong bài báo này đều được đánh giá trên PC với các tham số cơ bản (CPU: Intel(R) Xeon(R) CPU @ 2.20GHz; GPU: NVIDIA Tesla T4 with 16 GB GDDR6).

3.2. Các kết quả thí nghiệm

3.2.1. Tốc độ xử lý

Bảng 1 thể hiện các kết quả các kết quả thí nghiệm khi đánh giá tốc độ xử lý của các mô hình YOLO. Tốc độ xử lý được tính bằng thời gian (ms) được dùng để thực hiện quá trình suy luận trên một ảnh.

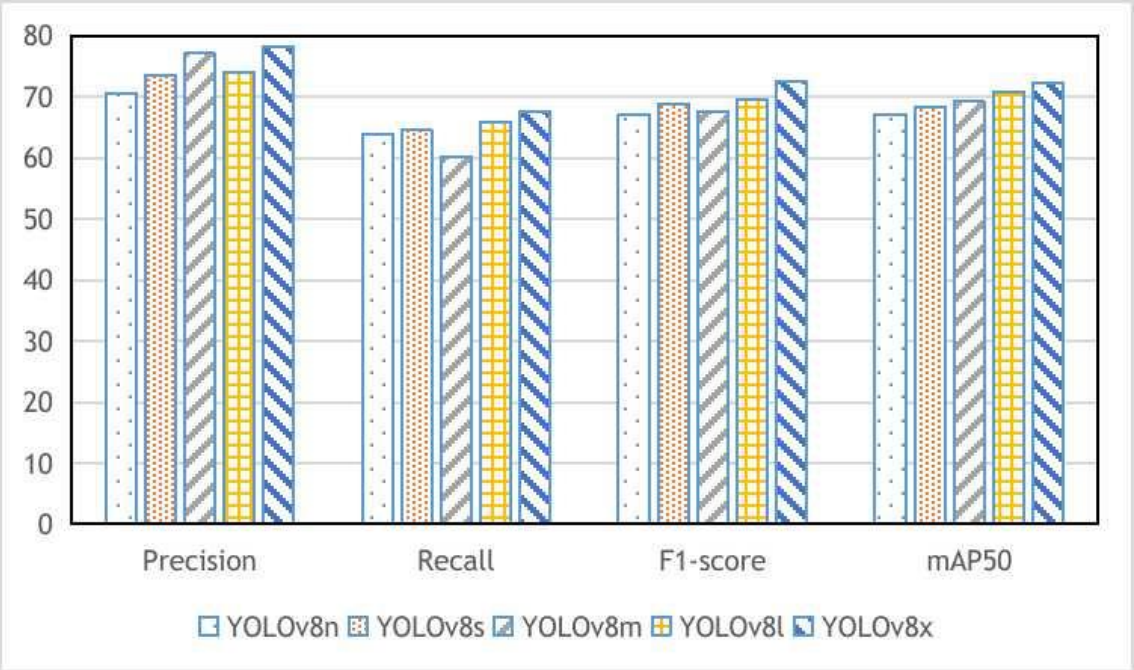
Bảng 1. Tốc độ xử lý của các mô hình YOLO

Độ đo	YOLOv8n	YOLOv8s	YOLOv8m	YOLOv8l	YOLOv8x
ms (GPU)	8.6	16.2	24.5	28.4	36.1
ms (CPU)	206.4	562.4	1382.3	2665.0	3988.4

3.2.2. Độ chính xác

Hình 5 thể hiện phần đánh giá các độ đo: Precision, Recall, F1-score và mAP50 của năm phiên bản của mô hình YOLOv8, tức là YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l và YOLOv8x. Phần đánh giá được thực nghiệm trên tập dữ liệu được thu thập

từ các lớp học mẫu giáo. Đây là tập dữ liệu được ghi hình bởi các camera giám sát thông thường. Tập dữ liệu này thể hiện sự đa dạng trong góc nhìn với nhiều thách thức nảy sinh từ môi trường thực tế: ánh sáng, bị che khuất, tư thế người khác nhau. Mục đích của tập dữ liệu này là để phục vụ cho các hệ thống giám sát giáo dục trong các điều kiện khác nhau.



Hình 5. Precision, Recall, F1-score và mAP của các phiên bản YOLOv8 trên tập dữ liệu giám sát.

3.3. Phân tích và thảo luận

Các kết quả được báo cáo trong Bảng 1 thể hiện tốc độ xử lý của các mô hình YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l và YOLOv8x. Kết quả thí nghiệm cho thấy cả 5 phiên bản của YOLOv8 đều có thể đạt đến khả năng xử lý theo thời gian thực với sự hỗ trợ tính toán của GPU. Kết quả này cũng khẳng định cả 5 phiên bản của mô hình YOLOv8 đều có thể đáp ứng các ứng dụng đòi hỏi tốc độ xử lý theo thời gian thực. Khi không có sự hỗ trợ của GPU, để có thể tăng tốc quá trình suy luận của YOLOv8 chúng ta có thể áp dụng một số kỹ thuật tối ưu kiến trúc mô hình, hoặc giảm kích thước đầu vào cũng như thông lượng truyền đến mô hình.

Hình 5 cho thấy kết quả so sánh của 5 phiên bản của mô hình YOLOv8 được áp dụng phương pháp học chuyển đổi trên tập dữ liệu thực tế để phát hiện người từ các camera giám sát. YOLOv8x có F1-score (72.5%) và mAP50 (72.2%) cao nhất, tuy vậy mô hình này phải đánh đổi với tốc độ xử lý thấp và khó áp dụng cho các hệ thống với yêu cầu thấp về cấu hình phần cứng. YOLOv8n có F1-score (67.0%) và mAP50 (67.2%) thấp nhất trong 5 phiên bản nhưng lại có tốc độ xử lý nhanh nhất. Với việc chuẩn bị một tập

dữ liệu phù hợp thì YOLOv8n cũng có thể đáp ứng được cả hai đòi hỏi về độ chính xác và tốc độ xử lý.

(a) Đối tượng bị che khuất một phần (b) Đối tượng ở khoảng cách xa

Hình 6. Một số thách thức liên quan đến kích thước đối tượng và bị che khuất một phần. Các bounding box màu xanh biểu diễn cho ground truth, màu đỏ biểu diễn cho kết quả dự đoán của mô hình YOLOv8n.

Với lượng dữ liệu chuẩn bị cho quá trình huấn luyện là khá ít (30 ảnh) nhưng hiệu quả mà các phiên bản của mô hình YOLOv8 mang lại là rất khả thi. Chỉ với 200 vòng lặp huấn luyện, mô hình YOLOv8x (kiến trúc lớn nhất trong 5 phiên bản) có thể hội tụ trong vòng 1 giờ. Rõ ràng, thời gian huấn luyện đã được rút ngắn rất nhiều khi so với việc huấn luyện từ đầu trên tập dữ liệu MS-COCO.

Bên cạnh việc tăng tốc quá trình huấn luyện khi tập dữ liệu để chuyển đổi tác vụ là nhỏ, thì hiệu quả dự đoán của mô hình cũng sẽ bị ảnh hưởng. Lý giải cho kết quả thí nghiệm này, chúng tôi nhận thấy rằng còn có nhiều thách thức tác động lên hiệu suất của mô hình. Thứ nhất là đối tượng nhỏ do ở khoảng cách xa hoặc bị che khuất như được chỉ ra ở hình 6. Trong tình huống đầu, các cháu học sinh đang nằm ngủ và bị che khuất bởi các tấm chăn (mền). Tình huống thứ hai lại có nhiều cháu ở ngoài phòng và các cháu ngồi dọc với hướng quan sát của camera khiến cho nhiều cháu bị che khuất. Để có thể nhận biết tốt các đối tượng trong trường hợp này, chúng ta có thể sử dụng các dấu hiệu khác để nhận biết như khuôn mặt nói riêng và đầu người nói chung chẳng hạn.

Hình 7. Một số thách thức liên quan đến tư thế đối tượng. Các bounding box màu xanh biểu diễn cho ground truth, màu đỏ biểu diễn cho kết quả dự đoán của mô hình YOLOv8n.

Bên cạnh đó, tư thế của đối tượng cũng ảnh hưởng lớn đến việc ra quyết định chính xác của mô hình. Trong cả hai tình huống (Xem hình 7), hấu hết các đối tượng đều ở tư thế ngồi và quay lưng lại với camera. Để khắc phục thách thức này, chúng ta cũng có thể cho mô hình được huấn luyện thêm với tập dữ liệu mới có nhiều đối tượng với tư thế ngồi.

Các báo cáo thí nghiệm cho thấy việc thực thi nhiệm vụ phát hiện đối tượng trong ngữ cảnh của camera giám sát có thể đáp ứng tốc độ xử lý thời gian thực. Bên cạnh đó, để đảm bảo được độ chính xác phù hợp với các bài toán thực tế, việc sử dụng các phiên bản YOLOv8s và YOLOv8m là thích hợp hơn so với các phiên bản còn lại. Tuy nhiên, chúng ta cần xác định rằng các ứng dụng khác nhau có các yêu cầu khác nhau về tốc độ và độ chính xác. Do vậy, khi triển khai nhiệm vụ giám sát cho từng ngữ cảnh, chúng ta cần phải cân bằng hai yếu tố này.

4. KẾT LUẬN

Mục đích của nghiên cứu này giới thiệu một mô hình tổng quát cho bài toán phát hiện người từ các video giám sát. Trong đó, chúng tôi giới thiệu phương pháp học chuyển đổi áp dụng cho mô hình YOLOv8, mô hình mới nhất trong họ mô hình YOLO. Bên cạnh đó, nghiên cứu này cũng thực hiện đánh giá sự phù hợp của việc áp dụng các mô hình YOLO cho nhiệm vụ phát hiện đối tượng thời gian thực. Năm phiên bản của mô hình YOLOv8 đã được lựa chọn để đánh giá trên độ chính xác và tốc độ xử lý. Kết

quả cho thấy các phiên bản này của mô hình YOLOv8 đều đáp ứng được tốc độ xử lý theo thời gian thực với sự hỗ trợ của GPU. Tuy nhiên, trong các hệ thống giám sát với cấu hình phần cứng không đảm bảo thì YOLOv8 có thể linh hoạt được sử dụng với các kiến trúc nhỏ hơn, tức là YOLOv8n và YOLOv8s.

Việc đạt được sự cân bằng giữa tốc độ xử lý và độ chính xác phụ thuộc rất lớn vào yêu cầu của từng ứng dụng thực tế. Điều này dẫn đến việc lựa chọn mô hình ban đầu phù hợp là rất quan trọng để có được sự cân bằng cần thiết cho một ứng dụng cụ thể. Nghiên cứu này có thể giúp ích cho việc triển khai các hệ thống giám sát đạt được hiệu quả cao nhất.

TÀI LIỆU THAM KHẢO

[1]. Wang, C. Y., Liao, H. Y. M., Wu, Y. H., Chen, P. Y., Hsieh, J. W., & Yeh, I. H. (2020). CSPNet: A new backbone that can enhance learning capability of CNN. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, pp. 390-391.

[2]. Wang, C. Y., Bochkovskiy, A., & Liao, H. Y. M. (2023). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7464-7475.

[3]. https://github.com/ultralytics/yolov8

[4]. Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2117-2125.

[5]. Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J. (2018). Path aggregation network for instance segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 8759-8768.

[6]. Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13, pp. 740-755. Springer International Publishing.

HUMAN DETECTION IN SURVEILLANCE VIDEOS USING YOLOV8

Le Quang Chien

Faculty of Information Technology, University of Sciences, Hue University

Email: lqchien@husc.edu.vn

ABSTRACT

The paper presents a generalized model for automatic detection of people using images from surveillance cameras. The model can be used for transfer learning and fine-tuned for specific projects in the real world. The first step is to train the model on a provided dataset to familiarize it with the object of interest. Then, transfer learning is performed with the learned parameters on a dataset collected from a "real world" environment to better adapt to challenging conditions. To maximize the model's performance, the current state-of-the-art single-shot detector YOLOv8 is utilized. The experimental results demonstrate the feasibility of the approach for video surveillance in smart cities.

Keywords: Object detection, YOLO, Transfer learning, Video surveillance.

Lê Quang Chiến sinh ngày 15/09/1983 tại Thừa Thiên Huế. Năm 2005, ông tốt nghiệp cử nhân chuyên ngành Tin học tại trường Đại học Khoa học, Đại học Huế. Năm 2007, ông nhận bằng thạc sĩ chuyên ngành khoa học máy tính tại trường Đại học Khoa học, Đại học Huế. Năm 2016, ông nhận học vị tiến sĩ chuyên ngành Tin học tại trường SOKENDAI (The Graduate University for Advanced Studies), Nhật Bản. Hiện nay, ông đang công tác tại khoa Công nghệ Thông tin, trường Đại học Khoa học, Đại học Huế.

Lĩnh vực nghiên cứu: Xử lý và nhận dạng ảnh, xử lý video, học máy, thị giác máy tính.