

ỨNG DỤNG YOLOV8 NHẬN DIỆN BỆNH TRÊN TÔM

YOLOV8 APPLICATION TO IDENTIFY DISEASES IN SHRIMP

Viên Thanh Nhã^{1,*}, Nguyễn Thị Kim Phụng¹, Đặng Ngô Duy Cát¹

Trần Vĩnh Phúc², Lê Đình Phú Cường³,

Phạm Thành Công⁴, Lê Đình Hồng Mạnh⁴

DOI: <http://doi.org/10.57001/huiv5804.2024.209>

TÓM TẮT

Trong nghiên cứu này, chúng tôi nghiên cứu nhận diện tôm trên mô hình nhận diện YOLOv8 đã đánh giá được cho tốc độ chính xác cao và tốc độ nhận diện nhanh cũng như tìm hiểu về kiến trúc của các tầng và so sánh kiến trúc mô hình YOLOv5. Quá trình đào tạo hai mô hình YOLOv8 và YOLOv5 cho nhận diện tôm có dấu hiệu bình thường hay bất thường diễn ra theo từng bước, (1) thu thập dữ liệu đầu vào với dữ liệu được thu thập là 2170 ảnh, (2) tiền xử lý dữ liệu loại bỏ những ảnh mờ và gán nhãn đối tượng, (3) đào tạo mô hình, đánh giá hiệu suất, và so sánh các thông số, kết quả đào tạo giữa hai mô hình để chọn ra mô hình có khả năng nhận diện tốt nhất. Cuối cùng, chuyển đổi mô hình sang các định dạng khác để nhận diện ảnh, ghi hình Video đối tượng tôm trên Website và ứng dụng di động Android.

Từ khóa: YOLOv8, YOLOv5, tôm, phát hiện đối tượng, hộp giới hạn.

ABSTRACT

In this study, we present the topic of shrimp identification on the YOLOv8 recognition model, which has been evaluated for high accuracy and fast recognition speed as well as learning about the architecture of the layers and Compare the YOLOv5 model architecture. The process of training two models YOLOv8 and YOLOv5 to identify shrimp with normal or abnormal signs takes place in steps, (1) collecting input data with collected data of 2170 images, (2) preprocess the data to remove blurry images and label objects, (3) train the model, evaluate performance, and compare parameters and training results between two models to select the best model. best recognition ability. Finally, convert the model to other formats to recognize images and record videos of shrimp objects on the Website and Android mobile applications.

Keywords: YOLOv8, YOLOv5, shrimp, object detection, bounding box.

¹Phân hiệu Trường Đại học Thủy Lợi

²Khoa Công nghệ thông tin, Trường Đại học Đà Lạt

³Trường Đại học Yersin Đà Lạt

⁴Trường Đại học Công nghiệp Hà Nội

*Email: vienthanhnha@tlu.edu.vn

Ngày nhận bài: 03/4/2024

Ngày nhận bài sửa sau phản biện: 10/5/2024

Ngày chấp nhận đăng: 25/6/2024

1. GIỚI THIỆU

Nhận diện đối tượng (Object Detection) là lĩnh vực quan trọng của thị giác máy tính sử dụng nhiều các thuật toán khác nhau dựa vào kiến trúc mạng nơ-ron tích chập để

phân tích trích xuất các đặc trưng hình ảnh [1]. Để có khả năng nhận diện và phân loại [1] đối tượng trong ảnh, video thì mô hình được đào tạo trên một lượng lớn dữ liệu hình ảnh hoặc đoạn video được cắt thành từng ảnh có gắn nhãn đối tượng mà mô hình cần học, giúp mô hình học được các đặc điểm của từng đối tượng. Khi mô hình thực hiện dự đoán ảnh hay video hoàn toàn mới mô hình dùng thuật toán để so sánh các đặc điểm có trong ảnh với dữ liệu đã được đào tạo để xác định vị trí, tỉ lệ các đối tượng thuộc lớp nào trong ảnh [2].

Nhận diện đối tượng đang được ứng dụng trong các dự án thực tế yêu cầu thời gian thực tế như nhận diện khuôn mặt [3], biển số xe, phát hiện chuyển động, xe tự lái, theo dõi giám sát hoạt động, chuẩn đoán bệnh,... Mô hình nhận diện đối tượng cũng đang phát triển cải tiến thêm với nhiều chức năng riêng biệt như ước lượng tư thế (pose estimation) dự đoán, ước lượng các tư thế thay đổi ở nhiều vị trí, góc quay về đối tượng có trong ảnh, video hay trong môi trường không gian 3D; mô hình theo dõi (tracking) xác định vị trí theo dõi hành vi di chuyển của đối tượng trong video hoặc từ khung hình này sang khung khác; phân đoạn hình ảnh (image segmentation) khi mô hình đã xác định được vị trí và lớp đối tượng mô hình tiến hành tô màu đối tượng theo hình dạng và từng pixel thuộc về đối tượng trong ảnh; hộp giới hạn định hướng (Oriented Bounding Box) giúp hộp giới hạn (Bounding Box) có thể xoay [4,5] theo nhiều chiều hướng khác nhau cho phù hợp với hình dáng đối tượng.

YOLOv8 là một mô hình nhận diện đối tượng mới đây, được đánh giá với có độ chính xác cao, tốc độ xử lý nhanh, dễ triển khai và khả năng linh hoạt, có thêm nhiều tính năng nhận diện khác nhau cũng như có hỗ trợ chuyển đổi triển khai mô hình đã đào tạo lên các ứng dụng trên Website, ứng dụng Android hay hỗ trợ nhúng mô hình vào Raspberry Pi. Dựa vào những khả năng vượt trội trên ta tiến hành thử nghiệm và đánh giá khả năng dự đoán chính xác đối tượng và tốc độ của mô hình này dựa trên đề tài nghiên cứu nhận diện các dấu hiệu bình thường và bất thường trên tôm do không có nhiều nghiên cứu sử dụng mô hình nhận diện của YOLOv8 để nhận diện các dấu hiệu bất thường trên tôm. Vì thế mô hình YOLOv8 được triển khai để đánh giá chính xác [1] khả năng của mô hình trong dự đoán đối tượng trên ảnh, video hay trong thời gian thực.

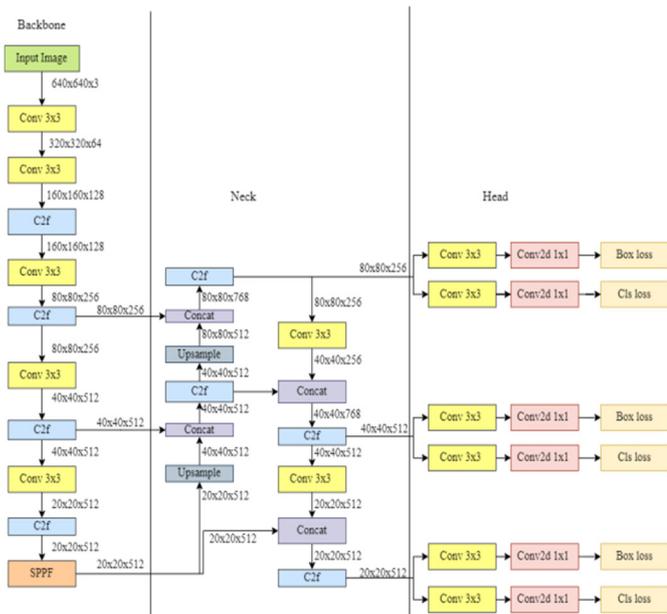
Ngoài ra, Việt Nam được biết đến là một trong những nước sản xuất tôm hàng đầu với hơn 700 nghìn ha nuôi tôm trải khắp cả nước [6] và tập trung nhiều ở các vùng đồng bằng sông Cửu Long và ngành sản xuất tôm đóng vai trò quan trọng trong ngành thủy hải sản với ước tính sản lượng tôm được sản xuất lên đến 1.014 nghìn tấn tôm trong năm 2022. Tuy nhiên, do biến đổi khí hậu và thời tiết thay đổi thất thường như mưa trái mùa, nhiệt độ tăng liên tục qua các năm dẫn đến tôm mắc nhiều loại bệnh khác nhau như bệnh đốm trắng (WSSV-White spot syndrome virus), đen mang (black gill), hoại tử cơ (IMNV - Infectious Myonecrosis Virus),... đều gây chết tôm làm thiệt hại cho nông dân nuôi trồng.



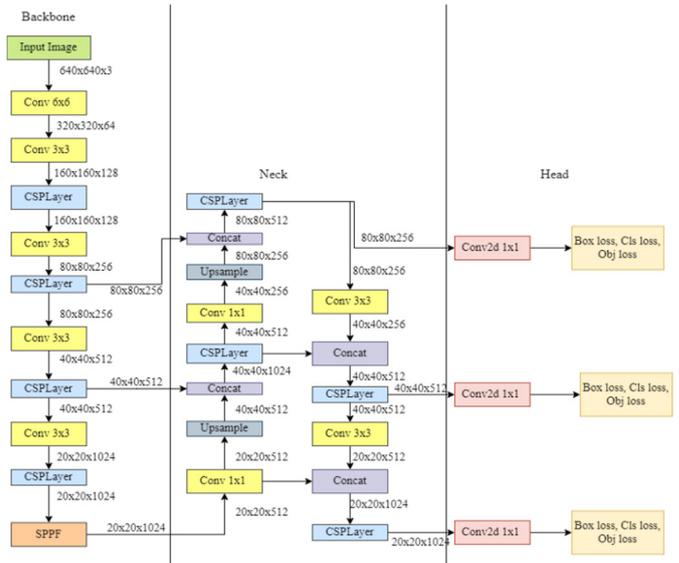
Hình 1. Tôm bị đen mang

2. MÔ HÌNH THUẬT TOÁN NHẬN DIỆN TÔM

YOLOv8 là một trong các phiên bản mã nguồn mở cho phép người dùng điều chỉnh sửa đổi lại kiến trúc hoặc thay thế các hàm khác cho phù hợp với mô hình dự án người dùng và kiến trúc YOLOv8 được phát triển xây dựng dựa trên kiến trúc YOLOv5 được đánh giá có tốc độ nhận diện nhanh trong thời gian thực tác giả cho ra mắt phiên YOLOv8 để cải thiện độ chính xác trong dự đoán so với phiên bản trước. Mô hình YOLOv8 có kiến trúc được chia ra làm 3 tầng: backbone, neck và head với vai trò riêng biệt.

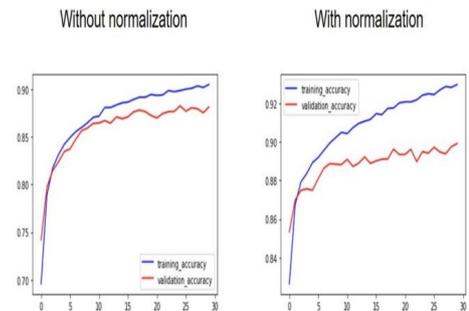


Hình 2. Mô hình kiến trúc YOLOv8



Hình 3. Mô hình kiến trúc YOLOv5

Đối với tầng đầu tiên Backbone có nhiệm vụ xử lý các dữ liệu ảnh đầu vào bằng cách trích xuất các đặc của đối tượng trong ảnh tiếp tục sử dụng kiến trúc CSPDarknet-53 ở cả hai phiên bản với lớp tích chập (Convolutional Layer) là lớp đầu tiên của kiến trúc ảnh đầu vào được biến đổi thành ma trận với các ô lưới chứa các giá trị sau đó lớp tích chập sẽ thực hiện bộ lọc (kernel) trượt trên bề mặt ma trận để lấy các giá trị đặc trưng của ảnh sau đó các giá trị trên được kết hợp lại tạo thành biểu đồ ma trận đặc trưng [1]. Chuẩn hóa lô (Batch Normalization) giúp chuẩn hóa đầu ra của convolutional layer trong biểu đồ đặc trưng bằng cách giảm các giá trị về khoảng giá trị nhất định giúp tăng tốc độ học và hội tụ của mô hình, giảm phụ thuộc vào giá trị ban đầu khiến quá trình học của mô hình ổn định hơn tránh hiện tượng overfitting.

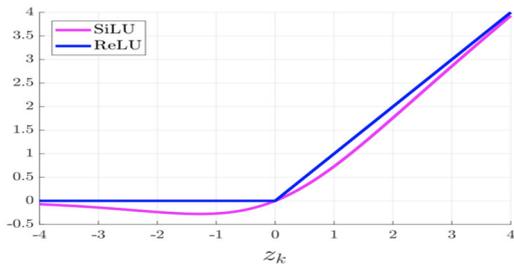


Hình 4. Biểu đồ sử dụng và không sử dụng chuẩn hóa dữ liệu

Hàm kích hoạt (Activation Function) được sử dụng trong YOLOv8 bởi SiLU (Sigmoid Linear Unit) được sử dụng sau lớp chuẩn hóa có độ hiệu quả cao khi giải quyết bài toán phức tạp khi biểu diễn các đặc trưng phi tuyến tính với đạo hàm đơn giản có thể nhận các giá trị âm thay vì biến đổi các giá trị âm thành 0 dẫn đến mất thông tin làm mô hình không học được tiếp các giá trị đó ở các tầng Convolution khác của hàm kích hoạt được sử dụng nhiều như hàm ReLU (Rectified Linear Unit).

Đặc biệt, hàm SiLU có công thức là:

$$SiLU(x) = \frac{x}{1 + e^{-x}} \tag{1}$$



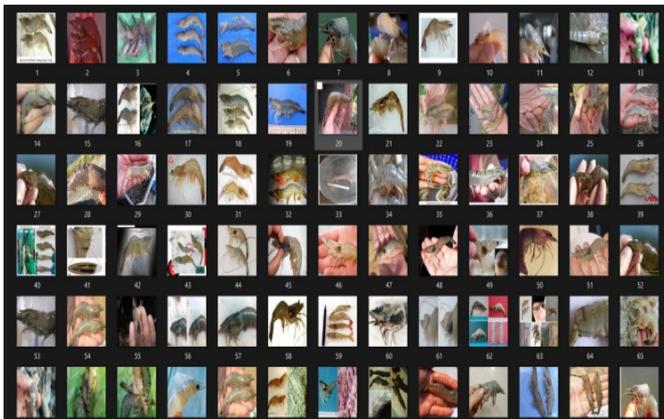
Hình 5. Biểu đồ so sánh hàm kích hoạt ReLU và SiLU

Neck có nhiệm vụ kết hợp các feature map lại với nhau từ backbone sau đó neck sẽ tạo biểu đồ đặc trưng đa cấp độ với các kích thước khác nhau sau đó sẽ đem vào phần head để tiến hành dự đoán ảnh ở tỉ lệ kích thước khác nhau. Để có thể kết hợp các feature map có kích thước khác nhau mô hình sử dụng upsample để tăng kích thước ảnh lên để có thể kết hợp các feature map lại với nhau.

Head ở YOLOv8 sử dụng phương pháp decoupled head thì mô hình sẽ chia làm hai phần có vai trò nhiệm vụ độc lập nhau không chia sẻ các giá trị tham số như phiên bản YOLOv5 để giúp mô hình tăng hiệu suất và độ chính xác trong dự đoán đối tượng và để hàm cls loss thêm vai trò xác định vị trí đối tượng trong ảnh thay cho hàm obj loss. Phần head được hiểu là đầu ra cuối cùng của mô hình kiến trúc sau khi đã trích xuất các đặc trưng của ảnh thì head có trách nhiệm dự đoán vị trí và tạo hộp giới hạn (Bounding Box) bao quanh đối tượng thông qua thông số tọa độ, kích thước mà mô hình phát hiện. Sau đó mô hình tiến hành dự đoán xác suất thuộc về từng lớp đối tượng có thể có trong bounding box.

3. PHƯƠNG PHÁP ĐỀ XUẤT

3.1. Tiền xử lý dữ liệu đầu vào



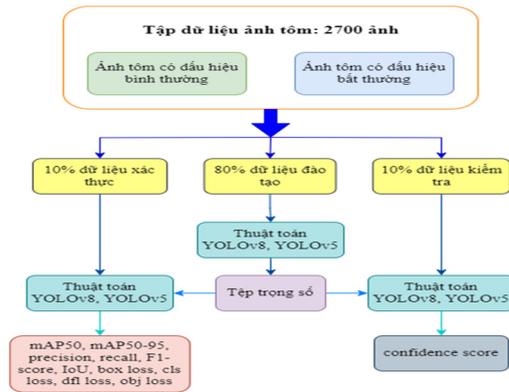
Hình 6. Tập dữ liệu ảnh tôm

Để mô hình có thể nhận diện đối tượng ta mong muốn ta cần thu thập càng nhiều dữ liệu ảnh tôm với nhiều đa dạng các góc độ, độ sáng, điều kiện môi trường và các dấu hiệu bất thường trên thân tôm, loại tôm qua đó giúp mô hình có thể nhận diện được nhiều trường hợp về tôm. Nguồn gốc của ảnh tôm được thu thập trên mạng như Google, shutterstock, các hội nhóm diễn đàn trên Facebook. Sau khi đã thu thập dữ liệu tiến hành điều chỉnh kích thước ảnh đầu vào trùng với kích thước ảnh mà mô hình có thể đào

tạo tốt nhất là 640*640 sau đó loại bỏ những ảnh quá mờ không rõ để giúp mô hình học được những đặc trưng, chi tiết trên đối tượng tôm cho ra kết quả đào tạo và dự đoán trên ảnh không qua đào tạo tốt nhất. Sau đó tiến hành gán nhãn đối tượng tôm cho 2 tập huấn luyện (train set), tập đánh giá (evaluation set) và tập kiểm tra (test set) dùng để đánh giá kết quả dự đoán thực tế của mô hình.

3.2. Phân chia tập dữ liệu và đào tạo mô hình

Sử dụng thư viện split-folders chia tập dữ liệu chính thành 3 tập dữ liệu là tập huấn luyện (train set), tập đánh giá (evaluation set) và tập kiểm tra (test set) theo như hình 7. Đối tập dữ liệu train sử dụng cho đào tạo mô hình, tập validate đánh giá mô hình trong quá trình đang đào tạo, tập test đánh giá kết quả mô hình sau khi đào tạo.



Hình 7. Quá trình đào tạo đánh giá của hai mô hình

Bảng 1. Các tham số đào tạo và kiểm tra của 2 mô hình

Tham số	YOLOv5	YOLOv8
Classes	2	2
Image size	640	640
Epochs	100	100
Batch	8	8
Mosaic	1,0	1,0
Activation Function	SiLU	SiLU

Các giá trị để đánh giá quá trình đào tạo mô hình trong mỗi lần lặp có trọng số tốt dựa vào công thức từ (2) đến (6). Trong đó TP (True Positive), FP (False Positive), TN (True Negative), FN (False Negative) với các giá trị trên có vai trò quan trọng để tính các tham số cho việc đánh giá mô hình, IoU (Intersection over Union) tính trên phần giao chia hợp giữa groundtruth box và predicted box và giá trị này quyết định đối tượng thuộc một trong bốn giá trị kể trên dựa vào IoU threshold (ngưỡng IoU), mAP50 là giá trị độ chính xác trung bình ở ngưỡng IoU 50, mAP50-95 giá trị độ chính xác trung bình ở ngưỡng IoU từ 50-95, Precision tỉ lệ độ chính xác mô hình trong dự đoán, Recall tỉ lệ độ chính xác mô hình so với thực tế và F1-score đánh giá hiệu suất của mô hình [1].

$$\text{precision} = \frac{TP}{TP + FP} \tag{2}$$

$$\text{recall} = \frac{TP}{TP + FN} \tag{3}$$

$$F1 - score = 2 \times \frac{precision \times recall}{precision + recall} \quad (4)$$

$$mAP50 = \frac{1}{N} \times \sum AP(IoU = 50) \quad (5)$$

$$mAP50 - 95 = \frac{1}{N} \times \sum AP(IoU = 50:95) \quad (6)$$

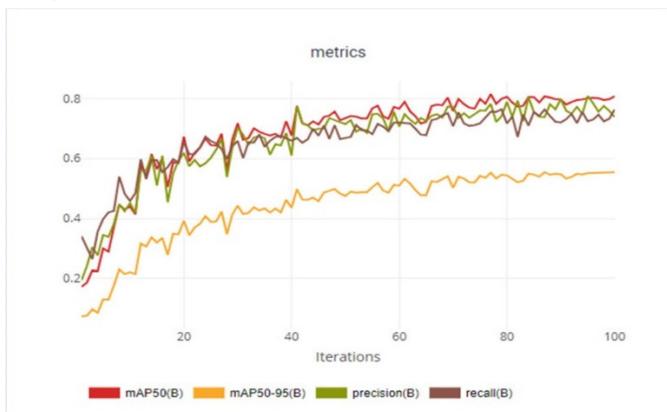
Quá trình huấn luyện và kiểm tra đánh giá khả năng nhận diện của mô hình đều thực hiện trên máy tính xách tay (Laptop) mô tả tại bảng 2. Sử dụng môi trường Python 3.11.5, Framework Pytorch phiên bản 2.1.1 và các thư viện cần thiết trong quá trình huấn luyện và hiển thị đánh giá kết quả huấn luyện như numpy, matplotlib, pillow, pandas, seaborn,...

Bảng 2. Thông tin phần cứng đào tạo và kiểm tra 2 mô hình

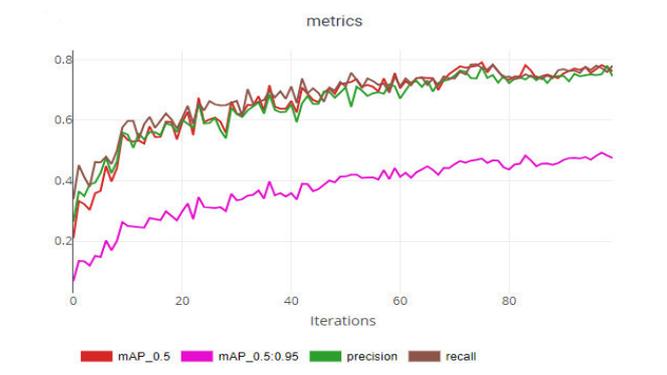
Cấu hình laptop	Thông số
Hệ điều hành	Windows 11
CPU	I7 10750H
GPU	GTX 1660Ti 6GB
RAM	16GB

4. KẾT QUẢ THỰC NGHIỆM

Hình 8 và 9 cho thấy trong quá trình đào tạo có tỉ lệ Precision và Recall ngang nhau giữa hai mô hình, tuy nhiên, tại Bảng 3 mô hình YOLOv5 có tốc độ đào tạo là 5h56 nhanh hơn so với YOLOv8 là 17h05 nhưng YOLOv8 lại có tỉ lệ chính xác giữa các tham số còn lại cao hơn YOLOv5.



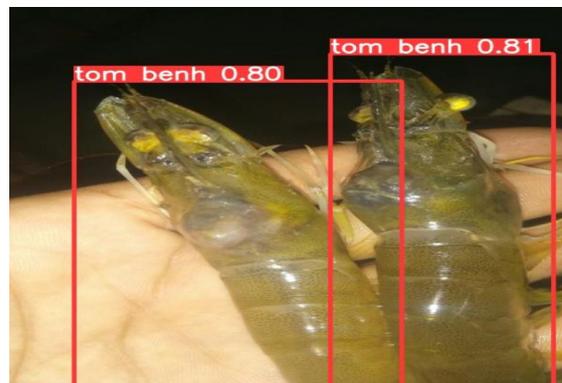
Hình 8. Đồ thị tỉ lệ độ chính xác bởi 100 lần đào tạo của YOLOv8



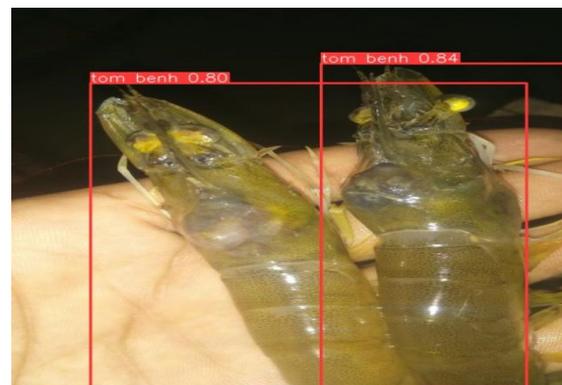
Hình 9. Đồ thị tỉ lệ độ chính xác bởi 100 lần đào tạo của YOLOv5

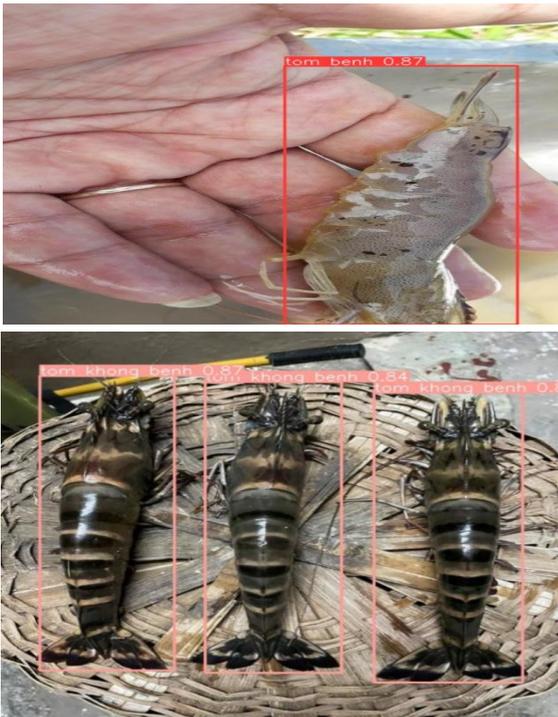
Bảng 3. Kết quả 100 lần đào tạo của hai mô hình.

Model	Run time	Precision	Recall	mAP50	mAP50-95	F1-score
YOLOv8	17h05	0,739	0,763	0,807	0,554	0,753
YOLOv5	5h56	0,751	0,772	0,782	0,493	0,731



YOLOv5





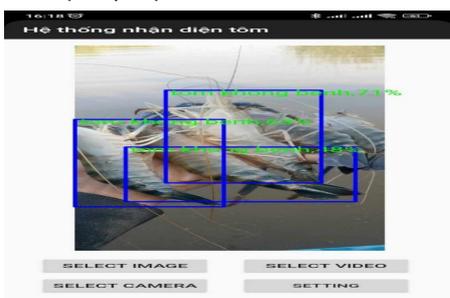
YOLOv8

Hình 10. Kết quả kiểm tra trên tập test của hai mô hình

Sau khi đánh giá được các thông số trong quá trình đào tạo mô hình cùng khả năng nhận diện trong tập kiểm tra (tập test) ta thấy mô hình YOLOv8 độ chính xác trong dự đoán ảnh mới cao hơn so với YOLOv5 [7] nên ta sử dụng tập trọng số của mô hình YOLOv8 để tiến hành chuyển đổi tập best.pt (chứa thông số dự đoán tốt nhất) sang định dạng model.json của TensorflowJS trong bộ thư viện Tensorflow sử dụng cho nhận diện trên Website và định dạng sang best.onnx cho nhận diện trên ứng dụng Android.



Hình 11. Giao diện nhận diện tôm bởi Website



Hình 12. Giao diện nhận diện tôm bởi ứng dụng Android

5. KẾT LUẬN

Mô hình nhận diện YOLOv8 đã cho ra kết quả dự đoán có độ chính xác cao hơn so với mô hình YOLOv5 cũng như đã ứng dụng mô hình nhận diện đối tượng tôm trên Website và ứng dụng Android. Tuy độ chính xác trong đào tạo nhỏ hơn 75% nhưng nó cũng đã cho thấy mô hình chúng tôi vẫn có thể phát triển tiếp mô hình và bổ sung thêm các chức năng cho Website và ứng dụng Android trong tương lai.

TÀI LIỆU THAM KHẢO

- [1]. Phuc D. H., Vu L. A. U., Quy H. D., "Research and development of an intrusion warning system using advanced artificial intelligence algorithms," *Journal of Transportation Science and Technology*, 23, 1, 22-35, 2023.
- [2]. Khang N. T. T., Giang N. H., Truc N. T. T., "Research on Using the Detr Model with the Problem of Object Detection in Aerial Images," in *Proceeding of The National Conference on Fundamental and Applied IT Research (FAIR)*, Danang, 28-29/9/2023.
- [3]. <https://tuhoclaptrinhsite.wordpress.com/author/phanminhtoanhcm/>
- [4]. <https://thuvienso.hcmute.edu.vn/dien-dien-tu-vien-thong/tat-ca-tai-lieu-dien-dien-tu-vien-thong-479-0.html?catetl=0&fft=all&ft=all&page=4&subcatetl=0&vt=tainhieunhat>
- [5]. Nghia T. G., Huy C. Q., Cuong T. V. , "The analysis and search human information system from surveillance cameras," in *Proceeding of The National Conference on Fundamental and Applied IT Research (FAIR)*, Hanoi, 03-04/11/2022.
- [6]. <https://tailieu.vn/doc/bao-cao-tang-cuong-nganh-van-tai-hang-hoa-duong-bo-viet-nam-huong-den-giam-chi-phi-logistics-va-pha-2391585.html>.
- [7]. Anh T. V., Khiem H. T., Dong K. T., Nghi L. T. , Hanh T. H. , Phong D. H., "Research on the capability of the GB and SVR machine learning models in mapping land subsidence susceptibility in the Ca Mau region, Vietnam," *Journal of Hydro-Meteorology*, 757, 60-73, 2024.

AUTHORS INFORMATION

**Vien Thanh Nha¹, Nguyen Thi Kim Phung¹, Dang Ngo Duy Cat¹,
Tran Vinh Phuc², Le Dinh Phu Cuong³, Pham Thanh Cong⁴,
Le Dinh Hong Manh⁴**

¹Thuy Loi University - Southern Campus, Vietnam

²Faculty of Information Technology, Da Lat University, Vietnam

³Yersin University, Vietnam

⁴Hanoi University of Industry, Vietnam