ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

NHẬN DIỆN VÀ PHÂN LOẠI CÔN TRÙNG TRÊN BẪY VÀNG

Học phần: Học máy

Lý Trường Phước

Mã sinh viên: 21020934

I. Giới thiệu đề tài

Bẫy dính màu vàng được sử dụng phổ biến trong nhà kính để giám sát côn trùng gây hại như Trialeurodes vaporariorum và Bemisia tabaci. Tuy nhiên, việc đếm và phân loại côn trùng hiện nay chủ yếu thủ công, tốn thời gian và dễ sai sót.Đề tài "Nhận diện và phân loại côn trùng trên bẫy vàng bằng học máy" nhằm xây dựng mô hình tự động, giúp nâng cao hiệu quả giám sát sâu bệnh trong nông nghiệp công nghệ cao.

II. Mô tả, phân tích và xây dựng bộ dữ liệu

A. Mô tả, phân tích dữ liệu

Bộ dữ liệu “Yellow Sticky Traps” được cung cấp bởi nhóm nghiên cứu Hà Lan, gồm các ảnh chụp từ bẫy côn trùng dùng trong nhà kính.

- Số lượng ảnh: 284 ảnh định dạng .jpg.
- Kích thước ảnh gốc: 5184 × 3456 pixels.
- Định dạng nhãn: Pascal VOC (.xml), được gán bởi công cụ LabelImg.
- Số lớp được gán nhãn chính:
    - Whiteflies (WF) – Bọ phấn trắng: 5807 mẫu (≈ 75.35%)
    - Macrolophus (MR) – Bọ mắt lưới xanh: 1619 mẫu
    - Nesidiocoris (NC) – Bọ mắt lưới nâu: 688 mẫu
    - (Có lớp phụ Thysanoptera nhưng bị loại do quá ít mẫu)

Đặc điểm thống kê:

- Phân bố lớp mất cân bằng: Lớp WF chiếm đa số, gây thách thức trong huấn luyện.
- Số lượng bounding box: Trung bình mỗi ảnh có khoảng 10–20 côn trùng.
- Tổng số mẫu sau gán nhãn: ≈ 8114 bounding boxes.

B. Xây dựng bộ dữ liệu

Trong bài toán nhận diện và phân loại côn trùng, chất lượng đầu vào ảnh hưởng mạnh đến hiệu suất mô hình học sâu. Để chuẩn hóa và tối ưu hóa dữ liệu đầu vào, tôi sử dụng nền tảng Roboflow để thực hiện các bước tiền xử lý tự động và hiệu quả:

Bước 1: Thay đổi kích thước ảnh

Toàn bộ ảnh gốc có độ phân giải cao (5184 × 3456 pixels), tuy mang lại chi tiết tốt nhưng gây tốn tài nguyên bộ nhớ và tính toán trong quá trình huấn luyện. Do đó, ảnh được resize về kích thước chuẩn 640 × 640 pixels, vì:

- Đây là kích thước tiêu chuẩn được tối ưu cho các mô hình object detection hiện đại như YOLOv5/YOLOv8.
- Giúp cân bằng giữa chi tiết ảnh và tốc độ huấn luyện, tiết kiệm bộ nhớ GPU.
- Cho phép huấn luyện đồng nhất khi batch size lớn, tránh lỗi "out of memory" khi xử lý ảnh độ phân giải cao.

Bước 2: Tăng cường dữ liệu (Data Augmentation)

Nhằm tăng độ đa dạng và khả năng khái quát của mô hình, toàn bộ dữ liệu được tăng cường bằng cách áp dụng tất cả các kỹ thuật augmentation miễn phí có sẵn trên Roboflow, bao gồm:

- Flip ngang / dọc (Horizontal/Vertical Flip): tăng khả năng nhận diện khi côn trùng xoay ngược.
- Rotate: giúp mô hình học tốt hơn với các hướng xuất hiện khác nhau của vật thể.
- Brightness / Exposure: tăng cường độ sáng, mô phỏng điều kiện ánh sáng đa dạng.
- Blur: làm mờ ảnh nhẹ, giảm nhiễu mô hình với hình ảnh không rõ nét.
- Noise: thêm nhiễu ngẫu nhiên, mô phỏng cảm biến máy ảnh kém chất lượng.
- Cutout: làm mất một phần ảnh nhằm nâng cao khả năng mô hình nhận diện khi côn trùng bị che khuất.
- Auto-orient / Resize & Pad: đảm bảo ảnh đầu vào luôn đúng định dạng đầu vào yêu cầu.

Việc sử dụng đầy đủ các phương pháp tăng cường dữ liệu này giúp nhân rộng số lượng mẫu huấn luyện một cách tự nhiên, đồng thời giảm hiện tượng overfitting trên tập train.

Bước 3: Chia dữ liệu

Sau khi xử lý và tăng cường, bộ dữ liệu được chia theo tỷ lệ:

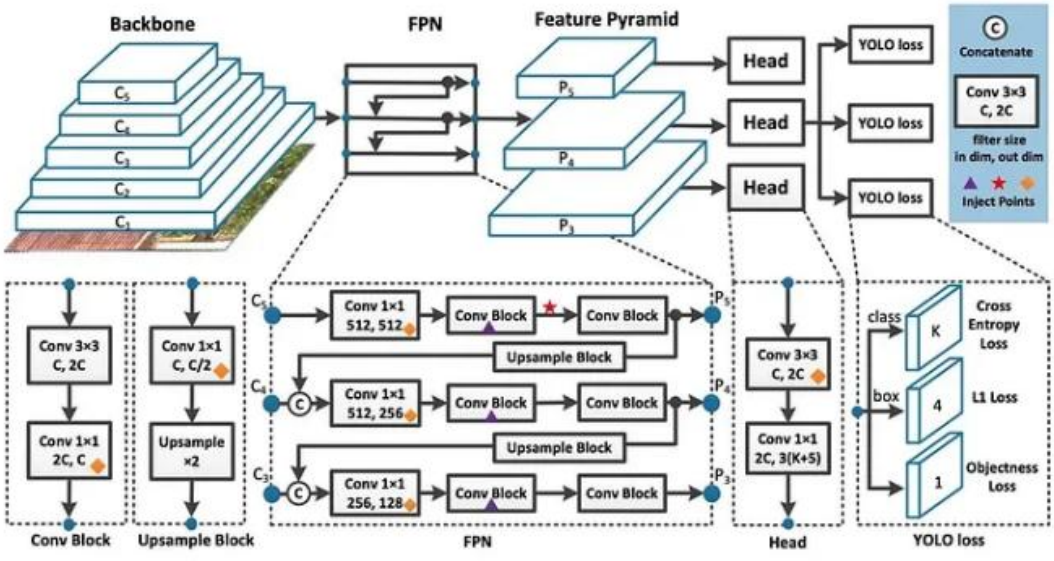
- 70% dùng để huấn luyện (train),
- 20% dùng để kiểm định (validation),
- 10% dùng để kiểm thử (test),

III. Xây dựng mô hình

Để giải quyết bài toán nhận diện và phân loại côn trùng trên bẫy vàng, đề tài lựa chọn ba mô hình học sâu tiêu biểu, đại diện cho hai hướng tiếp cận chính trong bài toán object detection: mô hình một giai đoạn (one-stage) và mô hình hai giai đoạn (two-stage). Việc kết hợp này cho phép so sánh toàn diện giữa hiệu suất, độ chính xác và tốc độ xử lý, từ đó lựa chọn phương án tối ưu cho ứng dụng thực tế.

1. YOLOv8n (Ultralytics)

YOLOv8 (You Only Look Once version 8) là một trong những mô hình nhận diện vật thể tiên tiến nhất hiện nay, do Ultralytics phát triển. Mô hình này được tối ưu hóa cả về kiến trúc và thuật toán để đạt hiệu quả cao trong các bài toán object detection, đặc biệt là trên các bộ dữ liệu có nhiều đối tượng nhỏ và mật độ cao – đúng với đặc trưng của ảnh bẫy dính côn trùng.



Hình 3.1. Cấu trúc YOLOv8

Các ưu điểm khiến YOLOv8 phù hợp cho bài toán này:

- Anchor-free: Mô hình loại bỏ khái niệm anchor boxes, giúp phát hiện tốt hơn các đối tượng nhỏ và không bị giới hạn bởi preset anchor.
- Hiệu suất cao: Kết hợp backbone C2f hiện đại, giúp cải thiện khả năng trích xuất đặc trưng.
- Hỗ trợ tốt với ảnh có bounding box chồng lấn – tình trạng phổ biến trong ảnh côn trùng.
- Tối ưu cho inference: Tốc độ nhanh, dễ triển khai trên các nền tảng như Kaggle, Roboflow, hoặc thiết bị thực tế.

⇨ Lý do chọn phiên bản YOLOv8n:

YOLOv8n là phiên bản "nano", nhẹ nhất trong dòng YOLOv8, phù hợp cho các lý do sau:

- Tài nguyên hạn chế: Huấn luyện trên Kaggle với GPU giới hạn (ví dụ: T4/16GB), mô hình nhẹ giúp giảm nguy cơ tràn bộ nhớ.
- Tốc độ huấn luyện nhanh, phù hợp với số lượng ảnh không lớn (284 ảnh gốc).
- Khả năng khái quát tốt, dù kích thước nhỏ nhưng vẫn đạt độ chính xác chấp nhận được với các bài toán đơn lớp hoặc ba lớp như hiện tại.

Dữ liệu đầu vào cho mô hình YOLOv8 bao gồm:

- Ảnh đầu vào: Định dạng .jpg, đã resize về 640×640 pixels thông qua Roboflow.
- Nhãn (labels): Định dạng YOLO (txt) gồm các dòng: [class_id x_center y_center width height] – các giá trị chuẩn hóa từ 0–1.
- Số lớp huấn luyện: 3 lớp chính: Whiteflies, Macrolophus, Nesidiocoris.

Sau khi huấn luyện, mô hình YOLOv8n tạo ra:

- Tệp trọng số mô hình: .pt (ví dụ: best.pt, last.pt) – lưu mô hình tốt nhất và cuối cùng.
- Biểu đồ loss/metrics: File .png hoặc log hiển thị quá trình giảm loss và cải thiện độ chính xác.
- Kết quả dự đoán (inference): Hình ảnh có bounding box + nhãn lớp + độ tự tin (confidence score).

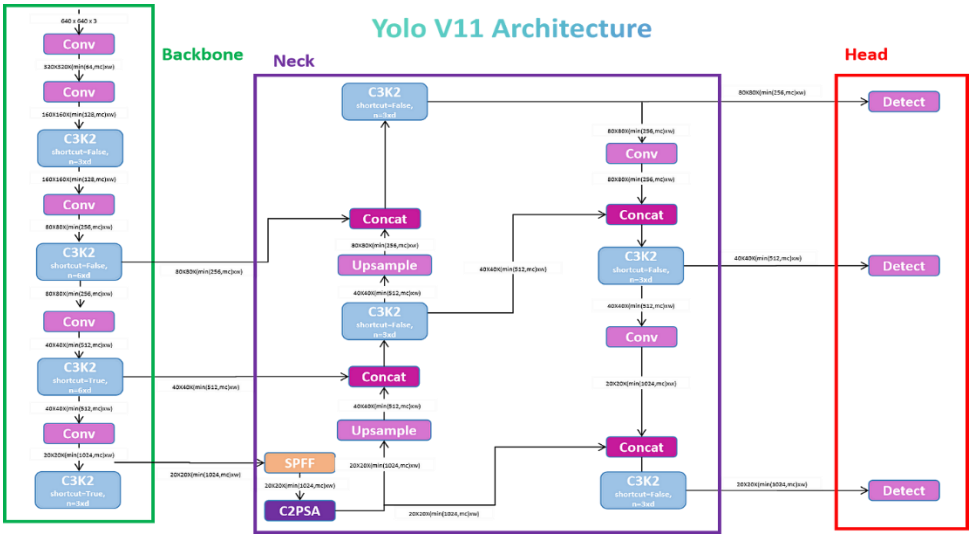
Các chỉ số được sử dụng để đánh giá hiệu suất mô hình gồm:

- Precision (Độ chính xác): Tỉ lệ phát hiện đúng trên tổng số dự đoán.
- Recall (Khả năng bao phủ): Tỉ lệ phát hiện đúng trên tổng số đối tượng thực tế.

- mAP@0.5: Mean Average Precision với ngưỡng IoU = 0.5 – chỉ số phổ biến để đánh giá object detection.
- mAP@0.5:0.95: Chỉ số trung bình mAP trên nhiều ngưỡng IoU – đánh giá độ bền mô hình.
- Loss: Gồm classification loss, box regression loss, objectness loss – phản ánh mức độ hội tụ của mô hình.

2. YOLOv11n (Ultralytics)

YOLOv11 là phiên bản kế nhiệm của YOLOv8 do Ultralytics phát triển. Với kiến trúc cải tiến và thuật toán huấn luyện nâng cao, YOLOv11 được kỳ vọng cải thiện độ chính xác trong nhận diện vật thể nhỏ, chồng lấn – rất phù hợp với ảnh bẫy dính côn trùng có nhiều đối tượng nhỏ, phân bố dày đặc.



Hình 3.2. Cấu trúc YOLOv11

So với YOLOv8, YOLOv11 có một số điểm cải tiến:

- Kiến trúc backbone mới (ví dụ: cải tiến từ C2f sang kiến trúc hiệu quả hơn).
- Cải thiện module head để tăng độ chính xác phân loại và định vị.
- Chiến lược huấn luyện tối ưu hơn, giúp nhanh hội tụ và chống overfitting tốt hơn trên tập dữ liệu nhỏ.
- Mặc định sử dụng augmentation mạnh hơn.

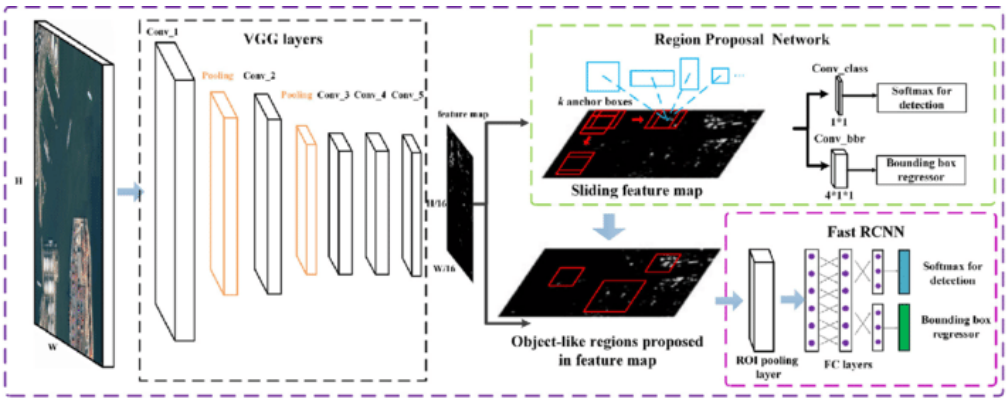
YOLOv11n là phiên bản "nano", nhẹ nhất trong dòng YOLOv11, được chọn vì:

- Giữ ưu điểm tốc độ nhanh, ít tài nguyên, tương đương YOLOv8n.
- Đánh giá khả năng cải tiến thực tế so với YOLOv8n trên cùng dữ liệu.
- Phù hợp để triển khai thử nghiệm hoặc thực tế trong môi trường giới hạn phần cứng.

Đầu vào, đầu ra các giá trị đánh giá của mô hình YOLOv11 tương tự như YOLOv8

3. Faster R-CNN

Faster R-CNN (Fast Region-based Convolutional Neural Network) là một trong những mô hình nhận diện vật thể tiên phong, được giới thiệu bởi Ren et al. năm 2015. Đây là phiên bản cải tiến của R-CNN và Fast R-CNN, với điểm nổi bật là cơ chế Region Proposal Network (RPN) – một mạng nơ-ron tích chập được huấn luyện song song để tự động sinh ra các vùng chứa vật thể (region proposals), giúp giảm thiểu độ trễ khi xử lý so với các phiên bản trước.



Hình 3.3. cấu trúc Faster R_CNN

Cấu trúc tổng quát của Faster R-CNN gồm ba bước chính:

1. Trích xuất đặc trưng (Feature Extraction): sử dụng một backbone CNN như ResNet hoặc VGG để rút trích đặc trưng từ ảnh đầu vào.
2. Sinh vùng đề xuất (Region Proposal Network - RPN): quét các vùng đặc trưng để đề xuất những vùng có khả năng chứa vật thể.
3. Nhận diện và phân loại: mỗi vùng đề xuất được cắt (RoI pooling), sau đó đưa qua fully connected layers để xác định loại vật thể và vẽ bounding box chính xác.

Lý do lựa chọn Faster R-CNN trong đề tài:

- Tối ưu về độ chính xác: Dù không phải là mô hình nhanh nhất, Faster R-CNN đạt độ chính xác rất cao trong nhiều bài toán nhận diện vật thể phức tạp, đặc biệt khi các đối tượng nhỏ, dày đặc, hoặc chồng lấn – rất phù hợp với ảnh bẫy dính côn trùng.
- Mô hình đơn giản, dễ triển khai: Dễ huấn luyện trên tập dữ liệu nhỏ với ít tinh chỉnh phức tạp hơn so với các mô hình transformer-based như RT-DETR.
- Hợp lý về mặt tính toán: Với tập dữ liệu không quá lớn, việc sử dụng một mô hình two-stage như Faster R-CNN giúp cân bằng giữa tài nguyên và độ chính

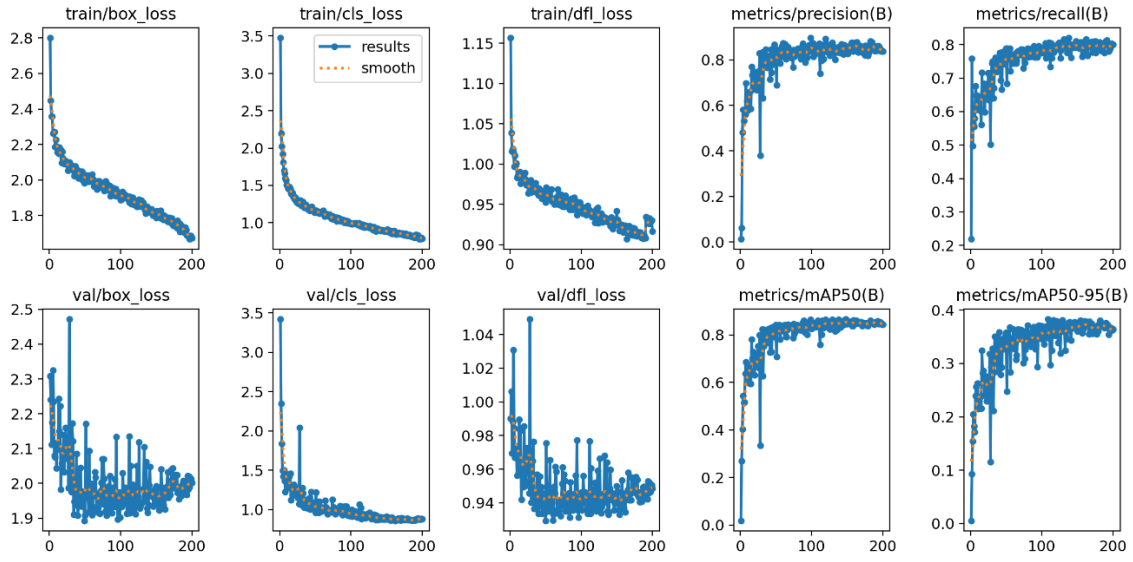
xác, không yêu cầu GPU quá mạnh khi so với các mô hình real-time tối ưu hóa hiệu suất cao như YOLO.

Bảng 3.1. So sánh Faster R-CNN với YOLOv8 và YOLOv11

Tiêu chí	**Faster R-CNN**	**YOLOv8**	**YOLOv11 (hypothetical)**
Cấu trúc	Hai giai đoạn (two-stage)	Một giai đoạn (one-stage)	Một giai đoạn
Tốc độ	Chậm hơn	Rất nhanh, gần real-time	Nhanh hơn YOLOv8 (giả định)
Độ chính xác vật thể nhỏ	Rất tốt	Tốt, nhưng khó khăn với vật thể nhỏ	Có thể cải thiện
Dễ triển khai	Cao	Trung bình	Thấp hơn do cấu trúc phức tạp hơn
Phù hợp với ảnh độ phân giải cao, côn trùng chồng lấn	Rất phù hợp	Có thể bị ảnh hưởng	Tùy biến mạnh nhưng cần nhiều tinh chỉnh

IV. Kết quả, đánh giá so sánh các mô hình

A.Mô hình YOLOV8



Hình 4.1 Các biểu đồ Training and Validation Loss and Metrics Curves

Nhận xét :

Loss giảm đều:

- Cả train loss và validation loss (box, class, dfl) đều giảm ổn định theo số epoch, cho thấy quá trình học hiệu quả và không bị overfit nghiêm trọng.

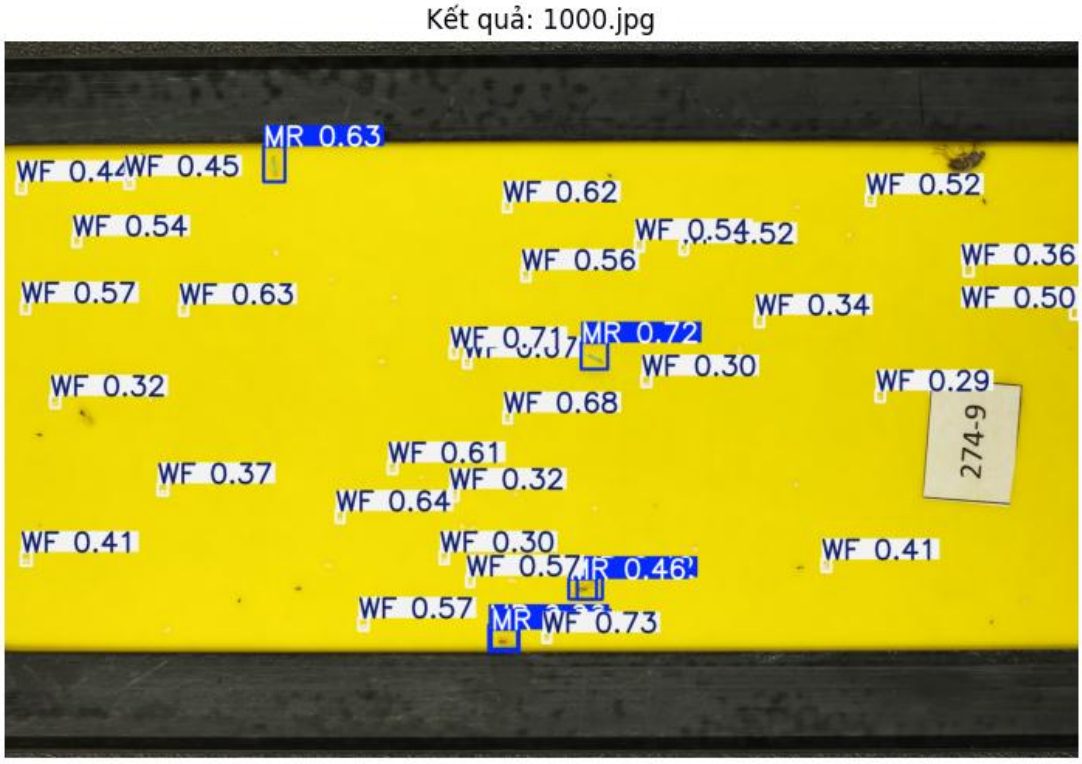
Precision và Recall tăng dần:

- Precision (độ chính xác) tiệm cận ~0.9, Recall (tỷ lệ phát hiện đúng) tăng lên ~0.8, cho thấy mô hình học tốt và ít bỏ sót vật thể.

mAP50 ~ 0.9, mAP50-95 ~ 0.38:

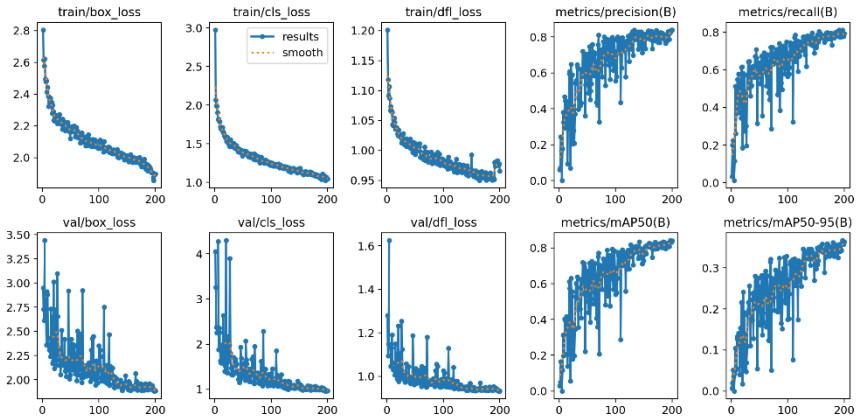
- mAP@0.5 cao (~0.9): mô hình phát hiện vật thể tốt khi chỉ yêu cầu IOU 0.5.
- mAP@[0.5:0.95] thấp hơn (~0.38): cho thấy độ chính xác định vị còn hạn chế ở các mức IOU cao, có thể do vật thể nhỏ hoặc chồng lấn nhiều.

⇨ Mô hình YOLOv8 đã huấn luyện thành công, học ổn định, không overfit, precision và recall tốt. Tuy nhiên, mAP50–95 chưa cao có thể do độ phức tạp trong phân biệt các loài côn trùng nhỏ, chồng lấn, và số mẫu chưa đủ lớn.



Hình 4.2 Kết quả nhận diện sau quá trình train YOLOV8n

B.Mô hình YOLOV11



Hình 4.3. Các biểu đồ Training and Validation Loss and Metrics Curves

Nhận xét

1. Loss giảm đều, hội tụ tốt:

- Cả ba loại loss trong training (box_loss, cls_loss, dfl_loss) và validation đều có xu hướng giảm dần đều, cho thấy mô hình học ổn định.
- Tuy nhiên, validation loss dao động mạnh hơn, đặc biệt là val/cls_loss (lên đến >4), cho thấy dữ liệu validation có độ nhiễu cao hoặc mô hình chưa khái quát hóa tốt hoàn toàn.

2. Precision và Recall tiến triển tốt:

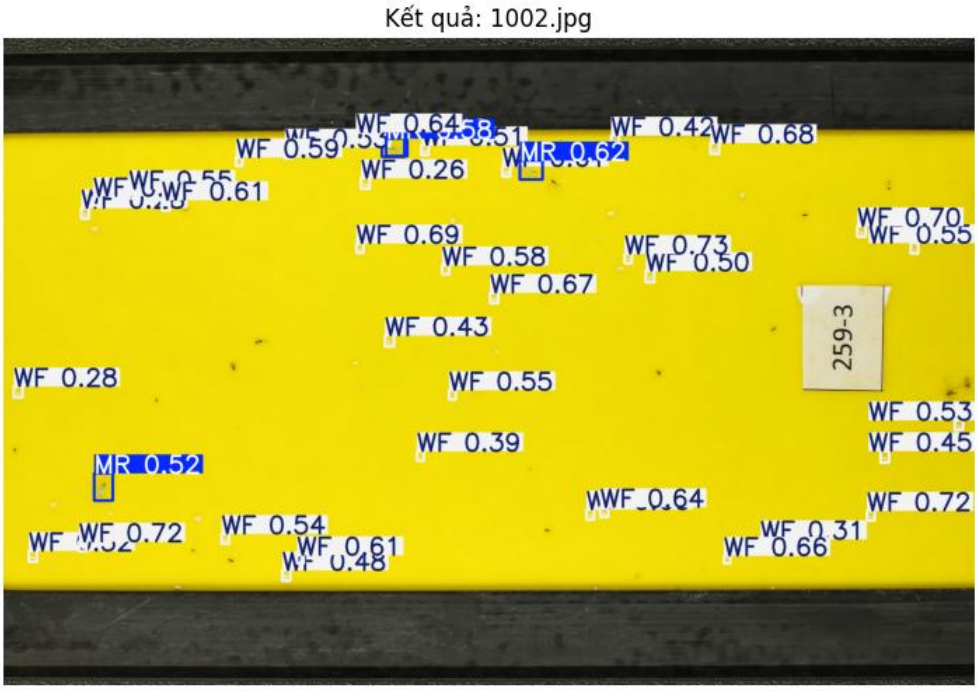
- Precision và Recall đều tiệm cận ~0.85, chứng tỏ mô hình có khả năng phát hiện và phân loại tốt các vật thể mục tiêu, bao gồm cả những đối tượng nhỏ hoặc chồng lấn.

3. Hiệu suất mAP:

- mAP50(B) tiệm cận ~0.85, cho thấy độ chính xác cao khi cho phép sai số vị trí lớn (IOU = 0.5).
- mAP50-95(B) đạt khoảng ~0.36, cao hơn YOLOv8 một chút, cho thấy YOLOv11 có thể tăng khả năng định vị chính xác hơn ở nhiều mức IOU khác nhau.

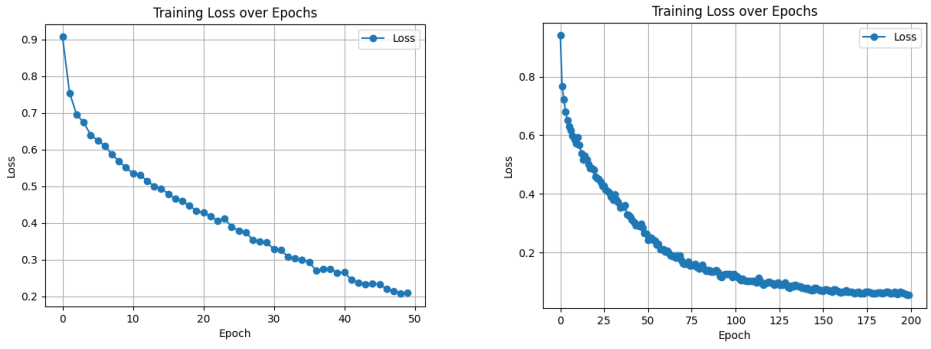
⇨ YOLOv11 đạt hiệu suất cao với loss giảm ổn định và mAP tốt. Precision/Recall cao cho thấy mô hình phát hiện hiệu quả. Tuy val/cls_loss dao động, kết quả cuối cùng vẫn thể hiện sự cải thiện nhẹ về độ chính xác so với YOLOv8, chứng

tỏ mô hình này có tiềm năng cao hơn về mặt tổng quát hóa và nhận diện vật thể phức tạp trong bối cảnh dữ liệu bẫy côn trùng.



Hình 4.3 Kết quả nhận diện sau quá trình train YOLOV11n

C.Mô Hình Faster R-CNN

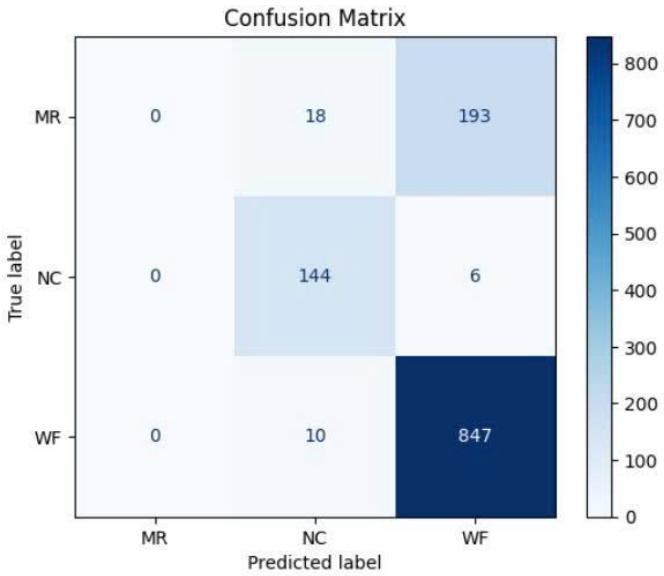


Hình 4.4 Đồ thị Training Loss over Epochs

Nhận xét:

- Cả hai biểu đồ (50 epochs và 200 epochs) đều cho thấy loss giảm ổn định và liên tục, không có hiện tượng dừng sớm hay tăng trở lại → chứng tỏ quá trình huấn luyện diễn ra tốt và mô hình đang học dần dần.

- Với 200 epochs, loss hội tụ về khoảng 0.1, thể hiện mức lỗi huấn luyện rất thấp → mạng đã học tốt đặc trưng của dữ liệu.



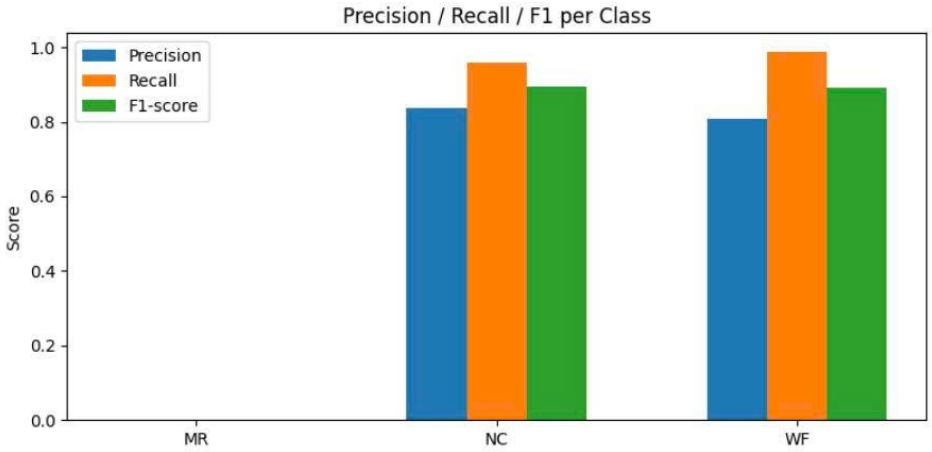
Hình 4.5 Confusion Matrix

Nhận xét:

- Lớp WF có độ chính xác rất cao (847 đúng trên ~857 mẫu).
- Lớp NC cũng tốt với 144/150 mẫu đúng.
- Tuy nhiên, lớp MR bị phân loại sai hoàn toàn, chủ yếu nhầm thành WF → cho thấy lớp này quá ít mẫu, khó phân biệt, hoặc đặc trưng không rõ ràng.

Bảng 4.1 Bảng đánhg giá

True Label	Dự đoán đúng	Dự đoán sai
MR	0	→ 193 bị nhầm thành WF, 18 thành NC
NC	144	→ 6 nhầm WF
WF	847	→ 10 nhầm NC

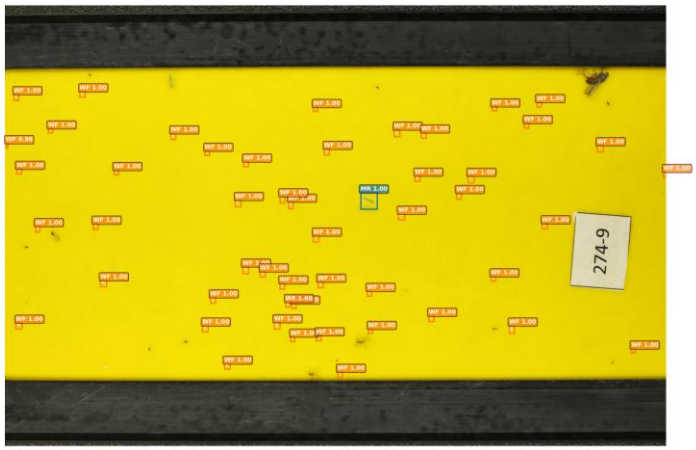


Hình 4.6. Biểu đồ Precision / Recall / F1-score per Class

- WF và NC đều đạt:
    - Precision ~ 0.80–0.85
    - Recall ~ 0.95–0.98
    - F1-score cao ~ 0.90
- MR không hiển thị → đồng nhất với confusion matrix: lớp này bị lỗi hoàn toàn trong nhận diện.

Nhận xét:

- Mô hình hoạt động rất tốt với lớp WF và NC, có thể ứng dụng thực tế trong phát hiện hai loài phổ biến nhất.
- Cần xử lý lại dữ liệu với lớp MR, ví dụ tăng cường dữ liệu (augmentation), sampling lại, hoặc tách riêng training cho lớp này nếu vẫn muốn giữ.



Hình 4.3 Kết quả nhận diện sau quá trình train Fastr R-CNN

Tổng kết mô hình Faster R-CNN được huấn luyện tốt, loss giảm đều và hội tụ rõ rệt. Các lớp côn trùng NC và WF được nhận diện chính xác với độ chính xác và độ bao phủ cao. Tuy nhiên, mô hình gặp khó khăn nghiêm trọng với lớp MR, do mất cân bằng dữ liệu hoặc đặc trưng chưa đủ mạnh. Kết quả phù hợp với kỳ vọng của mô hình hai giai đoạn: chính xác cao, nhưng nhạy cảm với mất cân bằng lớp.

Bảng 4.2 So sánh các mô hình đã huấn luyện

Tiêu chí	**YOLOv8**	**YOLOv11**	**Faster R-CNN**
Cấu trúc mô hình	One-stage (một giai đoạn)	One-stage (tối ưu hóa mới – giả định nâng cấp)	Two-stage (hai giai đoạn)
Thời gian huấn luyện	Nhanh, dễ triển khai	Trung bình đến nhanh (tùy cải tiến)	Chậm hơn do kiến trúc hai bước
Hội tụ loss	Nhanh, ổn định trong ~200 epochs	Nhanh, nhưng dao động ở val loss, cần tinh chỉnh thêm	Ổn định, hội tụ tốt, loss về ~0.1 sau 200 epochs
Hiệu suất tổng thể (mAP50)	~0.90	~0.85	Không rõ mAP, nhưng F1 cao với lớp chính (WF, NC)
mAP50–95	~0.38	~0.36	Không đánh giá trực tiếp, dùng F1 để thay thế
Độ chính xác lớp phổ biến (WF/NC)	Rất tốt (~0.85–0.90)	Tốt (~0.85–0.90)	Rất tốt (F1-score ~0.90 cho WF, NC)
Xử lý lớp nhỏ (MR)	Có xu hướng nhầm, nhưng vẫn nhận diện được	Có cải thiện về recall lớp nhỏ	**Gặp lỗi hoàn toàn**, MR bị phân loại sai hoàn toàn
Phù hợp dữ liệu côn trùng nhỏ, chồng lấn	Tốt	Tốt hơn YOLOv8 về recall	Tốt trong lý thuyết, nhưng thực tế cần xử lý dữ liệu tốt hơn
Khả năng triển khai thực tế	Rất cao (real-time)	Cao (tùy độ ổn định mô hình mới)	Trung bình (chậm hơn, phù hợp phân tích chuyên sâu)