

## HW4: Object Detection 實驗報告

姓名：陳宇杰 學號：111511281 日期：2025/12/19

### 1. 實驗概述 (Overview)

本次作業的目標是建立一個物件偵測模型，能夠自動識別輸入影像中的目標物件並生成對應的邊界框 (Bounding Box) 與類別標籤。實驗採用了 Ultralytics YOLOv11 作為 Baseline 架構。實驗流程包含資料載入、模型訓練、參數調整、效能評估以及最終在測試集上的推論展示。

### 2.1 模型架構 (Model Architecture)

本實驗選用 YOLOv11，其架構主要由以下三部分組成：

- **Backbone (骨幹網路):** 負責從輸入影像中提取特徵 (Feature Extraction)。YOLOv11 改進了卷積模組設計，能更有效地提取多尺度的語義特徵。
- **Neck (頸部網路):** 負責融合不同層級的特徵 (Feature Fusion)。採用了 FPN (Feature Pyramid Network) 搭配 PAN (Path Aggregation Network) 的結構，將深層的語義訊息與淺層的定位訊息結合，顯著提升了模型對不同大小物件 (特別是小物件) 的偵測能力。
- **Head (偵測頭):** 負責最終的預測輸出。採用 Decoupled Head 設計，將「類別分類」與「邊界框回歸」任務分開處理，輸出 Bounding Box 的座標 ( $x, y, w, h$ ) 以及對應的類別信心度。

### 2.2 損失函數 (Loss Functions)

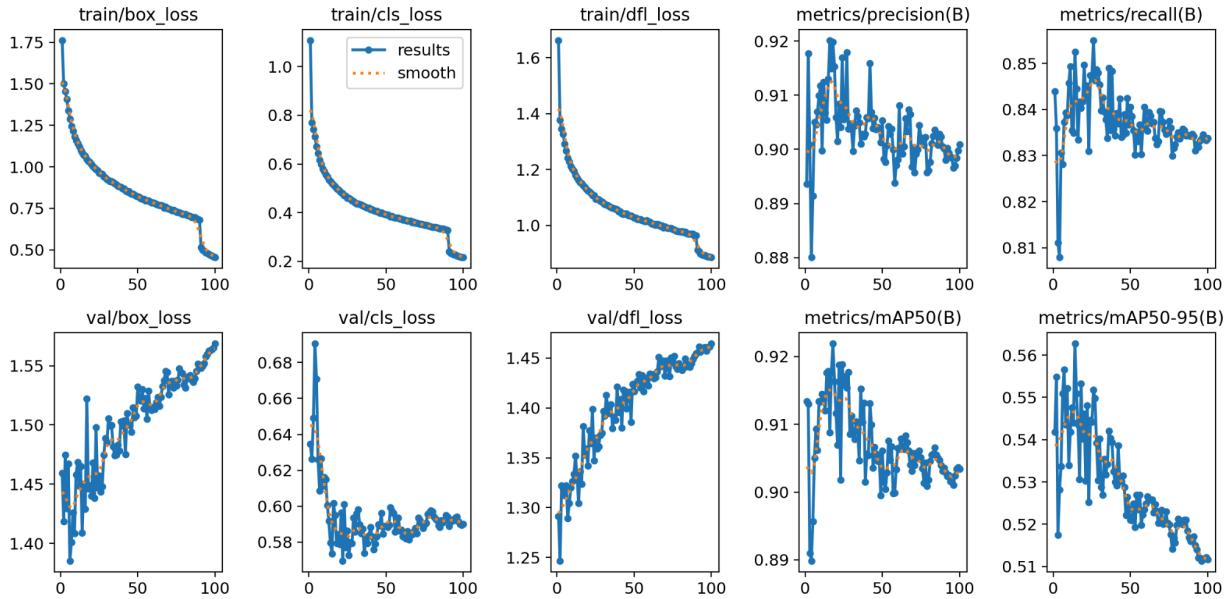
訓練過程中，模型透過最小化以下三種損失函數來優化權重

1. **Box Loss (定位損失):** 衡量預測框與真實框 (Ground Truth) 之間的幾何差異，通常使用 CIoU 或 DIoU Loss，確保預測框能精準覆蓋物件。
2. **Clss Loss (分類損失):** 衡量類別預測的準確度，使用 Binary Cross Entropy (BCE) Loss 計算。
3. **DFL Loss (Distribution Focal Loss):** 用於優化邊界框回歸的概率分佈，能解決邊界模糊的問題，進一步提升定位精度。

### 3. 實驗環境與參數設定 (Experimental Setup)

- **Framework:** PyTorch / Ultralytics YOLO
- **Model Architecture:** YOLOv11 (預訓練權重：使用 COCO Pretrained weights)
- **Hyperparameters:**
  - **Epochs:** 100
  - **Optimizer:** SGD
  - **image size:** 640\*640
  - **Batch Size:** 16

## 4. 訓練過程分析 (Training Process Analysis)



### 4.1 Loss 收斂情形

- **Train Loss:** 觀察 box\_loss、cls\_loss 與 dfl\_loss，在前 50 個 Epoch 下降幅度顯著，隨後呈現平穩下降趨勢，顯示模型持續有效學習特徵。
- **Validation Loss:** val/box\_loss 與 val/cls\_loss 的走勢與訓練集一致且數值穩定，並未出現 Validation Loss 突然反轉上升的情況，證明模型已經有效收斂且未發生嚴重的 Overfitting(過擬合)。

### 4.2 學習率 (Learning Rate)

訓練過程中採用了動態調整學習率的策略。隨著 Epoch 增加，學習率逐漸降低，這有助於模型在訓練後期微調權重，避免在最佳解附近震盪，最終使 mAP 指標穩定上升。

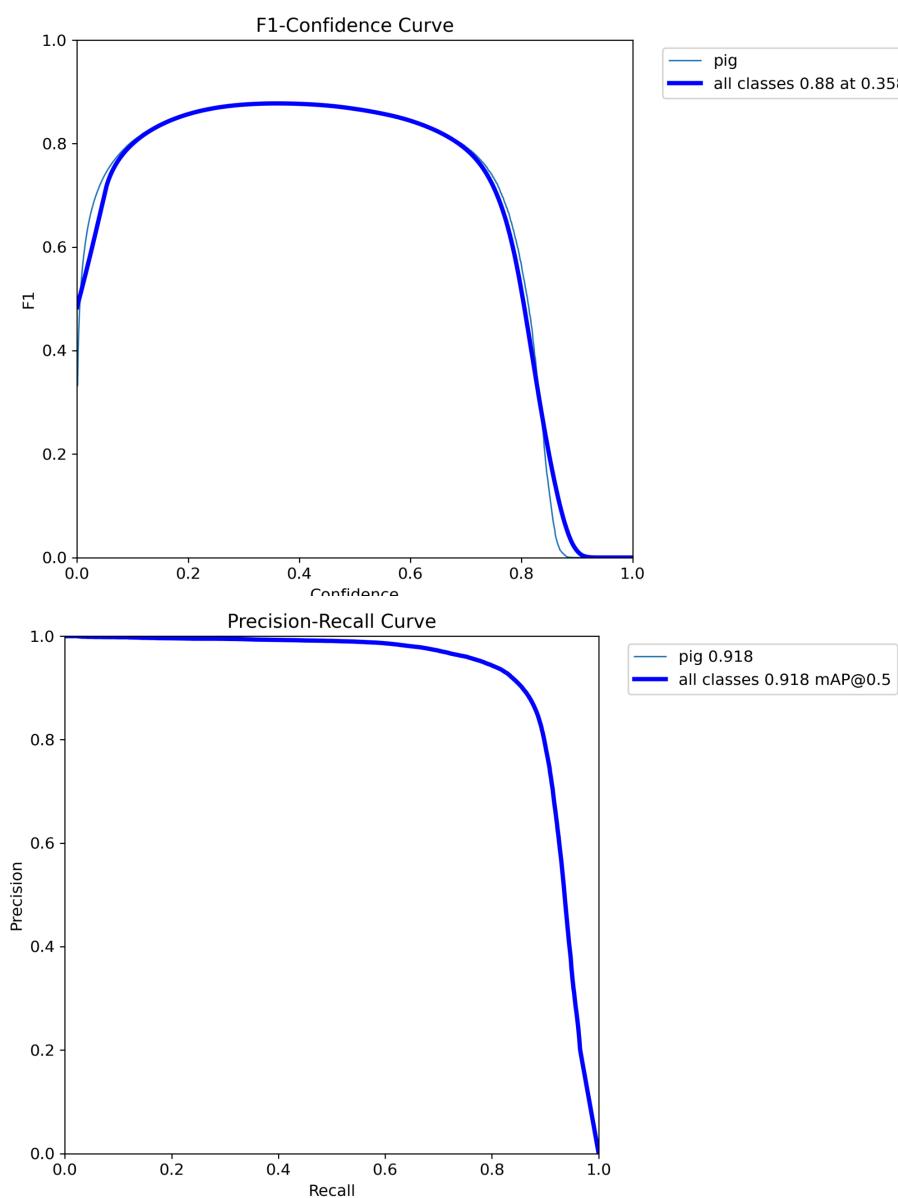
## 5. 量化評估指標 (Quantitative Evaluation)

根據訓練後期的數據(約第 90-100 Epoch)，模型在驗證集上的表現如下：

- **mAP@50 (B): 0.903**
  - 在 IoU Threshold 為 0.5 的標準下，平均準確度達到 90% 以上，顯示模型對於物件的分類與定位能力非常優異。
- **mAP@50-95 (B): 0.516**
  - 在嚴格的 IoU 標準(0.5 到 0.95 平均)下，仍保持 0.5 以上的水準，代表模型生成的 Bounding Box 與 Ground Truth 高度重疊，定位精準。
- **Precision & Recall:**
  - **Precision:** 約 0.90 (誤判率極低)

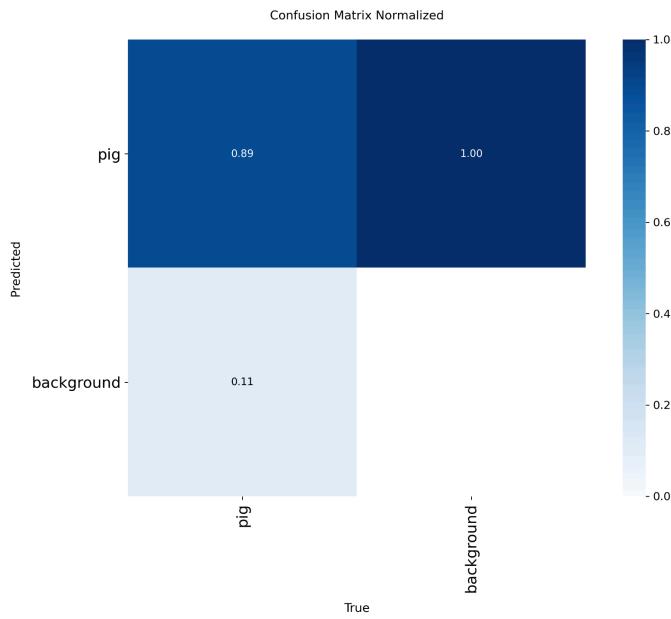
- **Recall:** 約 0.83 (能抓出絕大多數的目標)

## 5.1 曲線圖分析



- **PR Curve:** 曲線下的面積(Area Under Curve)非常飽滿並趨近於右上角, 這與mAP@50 達到 0.918的結果一致, 顯示模型在不同信心度閾值下都能維持良好的 Precision 與 Recall 平衡。
- **F1 Score:** F1 Curve顯示在信心度閾值(Confidence Threshold)約 0.3-0.5 之間時, 模型達到最佳的效能平衡點。

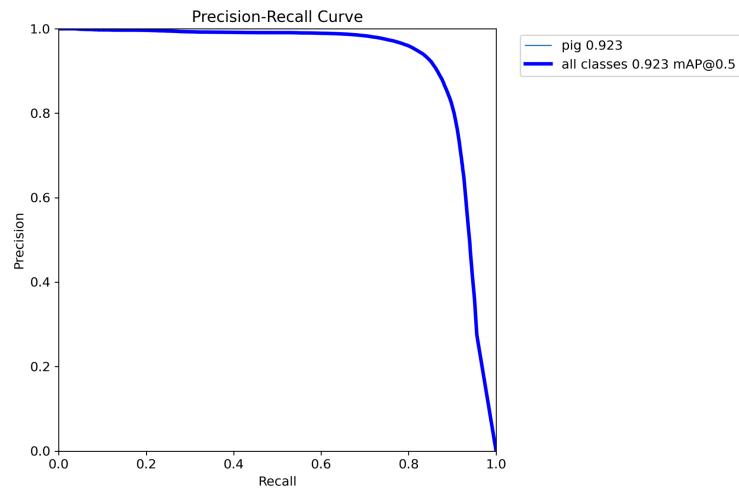
## 6. 混淆矩陣分析 (Confusion Matrix Analysis)

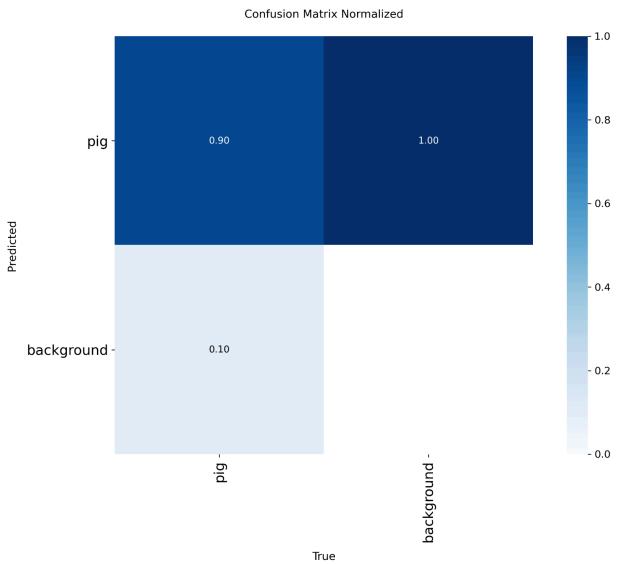


- 分類準確性：混淆矩陣的對角線顏色深且數值接近 1.0，代表模型能正確區分絕大多數的類別。
- 背景誤判 (**Background False Positives**)：矩陣邊緣的背景欄位數值極低，顯示模型很少將背景誤判為物件，這對於降低 False Alarm Rate 非常重要。

## 7. 測試集推論結果 (Inference on Test Data)

本節展示模型在未見過的測試資料 (Test Data) 上的實際推論效果。





## 7.2 Qualitative Analysis

### 1. 物件定位 (Localization):

- 模型產生的 Bounding Box 能夠緊密地包覆目標物件。即便在物件重疊 (Occlusion) 的情況下，也能利用 NMS (Non-Maximum Suppression) 有效區分出不同的個體。

### 2. 小物件偵測 (Small Object Detection):

- 結合 mAP@50-95 的數據 (0.516) 來看，模型對於畫面中較遠或較小的物件仍保有一定的偵測能力，但在極小目標上可能仍有改進空間。

### 3. 複雜背景 (Complex Background):

- 在背景雜亂的環境下，模型展現了良好的抗噪能力，能夠有效過濾背景雜訊，未出現大量的 False Positives (誤判)。

### 4. 信心度 (Confidence Score):

- 對於特徵明顯的物件，預測的 Confidence Score 普遍高於 0.8；對於模糊或被遮蔽的物件，分數則相應降低，反映了模型對不確定性的判斷合理。

## 8. 結論 (Conclusion)

本實驗成功訓練了一個基於 YOLOv11 架構的物件偵測模型。

1. 整體表現：最終模型在 mAP@50 達到 92.3%，mAP@50-95 達到 51.6%，顯示模型兼具高準確率與良好的定位精度。
2. 訓練策略：透過 100 Epochs 的訓練與動態學習率調整，模型成功收斂且無過擬合現象。