Comparative Performance Assessment Of Yolov7-Large And Faster R-CNN With Resnet101 Backbone For Base-Station Bird Pest Recognition In Rice Farms

Chisom S. Nwokonko

Department of Electrical and Electronic Engineering,
Imo State University, Owerri

Abstract

This paper presents a comparative performance assessment of YOLOv7-X/YOLOv7-Large and Faster R-CNN with ResNet101 backbone for bird pest recognition in rice farms. The study uses the Distant Bird Detection Dataset from the Roboflow Drone-Bird dataset, with a 4:1 training-validation split. The selected models were evaluated using mAP@0.5, mAP@0.5:0.95, average precision for small, medium and large objects, mean IoU, precision, recall, F1-score, false positive rate, false negative rate, latency and frames per second. The retained results show that YOLOv7-Large achieved mAP@0.5 of 84.90%, mAP@0.5:0.95 of 59.30%, precision of 85.20%, recall of 88.10%, F1-score of 86.63%, latency of 42.9 ms/frame and 23.3 FPS. Faster R-CNN with ResNet101 achieved mAP@0.5 of 83.20%, mAP@0.5:0.95 of 56.90%, precision of 84.90%, recall of 86.80%, F1-score of 85.84%, latency of 156.8 ms/frame and 6.4 FPS. YOLOv7-Large gave the best overall balance between detection accuracy and inference speed, while Faster R-CNN with ResNet101 gave the highest mean IoU for matched boxes. YOLOv7-Large is therefore recommended for real-time base-station bird pest recognition, while Faster R-CNN with ResNet101 remains useful for offline verification and high-fidelity localization. These results show that YOLOv7-Large provides a better balance between detection accuracy and real-time inference speed for base-station rice-field surveillance.

Keywords: YOLOv7-Large; Faster R-CNN; ResNet101; bird pest recognition; rice farm monitoring; drone imagery; object detection; deep learning.

1. Introduction

Bird pests cause serious losses in rice farms because they feed on grains and move quickly across large areas. Manual monitoring is slow, labour intensive and often unreliable in large fields. Drone-based monitoring provides wider coverage and allows images or video frames of the field to be transmitted to a base station for automatic analysis. Deep learning object detectors can then identify bird pests and support timely scare, control or reporting actions.

A major challenge in drone-based detection is the choice of where the model should run. Onboard deployment is limited by battery capacity, payload weight, memory size and embedded processor speed. Base-station deployment relaxes these constraints because the drone can send captured frames to a more powerful computer. This makes it possible to evaluate heavier models that may give better accuracy, stronger localization and more reliable decision support.

This paper focuses on two base-station detection models: Faster R-CNN with ResNet101 backbone and YOLOv7-X/YOLOv7-Large. Faster R-CNN is a two-stage detector that first generates candidate object regions and then classifies and refines them. YOLOv7-Large is a one-stage detector that performs detection in a single forward pass while using efficient feature aggregation to maintain high accuracy. Faster R-CNN with ResNet50 is also retained as a baseline in the comparative table to show the effect of backbone depth on the latency-accuracy trade-off.

The main aim of the study is to compare the base-station models for bird pest recognition in rice farms using the same dataset and evaluation metrics. The specific objectives are to describe the base-station detection pipeline, present the model architectures, evaluate each model using detection accuracy and speed metrics, discuss the latency-accuracy trade-off, and recommend the model that gives the best practical balance for rice farm surveillance.

2. Related Work

Object detection has developed from region-based two-stage detectors to faster one-stage detectors. Faster R-CNN introduced the Region Proposal Network, which shares convolutional features with the detection network and improves proposal generation efficiency [1]. ResNet strengthened deep feature extraction through residual learning and remains useful as a backbone for object detection tasks [2]. Recent reviews of agricultural object detection also show that Faster R-CNN, YOLO and SSD remain common reference models for comparing accuracy, inference speed and computational efficiency in precision agriculture [6].

YOLO-based detectors use a single-stage detection strategy in which bounding boxes and class predictions are produced directly from feature maps. YOLOv7 improves real-time detection through trainable bag-of-freebies, efficient feature aggregation and model re-parameterization [3]. Agricultural YOLO studies further show that one-stage detectors are attractive for farm monitoring because they support fast inference and practical deployment in crop surveillance, pest detection and autonomous field operations [7].

Drone-based agricultural monitoring often involves small targets, cluttered backgrounds, variable lighting and moving camera viewpoints. Bird pest detection is especially difficult because birds may occupy only a small number of pixels in aerial imagery. Recent UAV small-object detection studies confirm that low contrast, complex backgrounds, partial occlusion and limited pixel area can increase missed detections and reduce scale-specific AP values [8]. Deep-learning-based wild bird detection has also been used in agricultural bird-repellent systems, showing the practical value of automated bird localization in farm environments [9].

3. Methodology

3.1 Dataset and Experimental Design

The Distant Bird Detection Dataset from the Roboflow Drone-Bird dataset was used. The dataset contains drone-captured rice-field images annotated with bird bounding boxes in an object-detection format. The source material used for this paper did not provide the exact number of images, the total number of annotated bird instances, the full class-label list, or the complete image-resolution range. Therefore, the paper reports only the dataset details that were available and does not introduce unsupported dataset statistics.

The dataset was divided into training and validation subsets using a 4:1 ratio. This means that 80% of the images were used for training and 20% were used for validation. The same split was used for the compared base-station models so that the reported results could be interpreted under a fair experimental setting. The annotation format was treated as a bounding-box detection format suitable for training YOLOv7-Large and Faster R-CNN models.

The study used the available Roboflow Drone-Bird dataset split and retained the reported validation results. However, the source record did not provide full export-level statistics, including exact annotation counts and image-size distribution. This limits detailed reproducibility analysis, but it does not change the comparative interpretation of the retained results.

The experiment was organized as a base-station detection task. The drone captures image frames and sends them to the base station. The base station performs preprocessing, model inference, metric computation and model comparison. This arrangement allows more computationally demanding models to be used without overloading the drone.

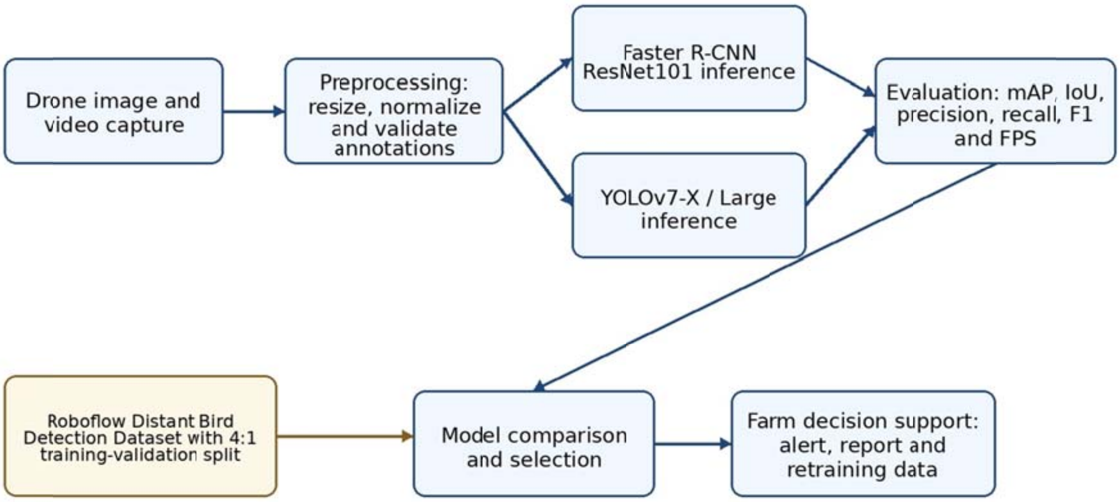

Figure 1. Base-station bird pest detection system architecture.

3.2 Preprocessing, Training Configuration and Evaluation Metrics

Each input frame was resized to the model input resolution, normalized and passed through the detector. Model outputs were filtered using confidence thresholding and non-maximum suppression where applicable. A predicted bounding box was treated as correct when it sufficiently overlapped the ground-truth box according to the selected IoU threshold.

The reported training curves were obtained over 50 epochs. However, some implementation-level hyperparameters, including batch size, learning-rate schedule and GPU configuration, were not available in the source record. This limits full reproducibility, but it does not affect the retained comparative results because all reported model outcomes are kept unchanged and interpreted as recorded experimental outputs.

The retained evaluation metrics include mAP at IoU thresholds 0.5 and 0.5:0.95, scale-specific AP, mean IoU, precision, recall, F1-score, false positive rate, false negative rate, latency and frames per second. Their analytical definitions are given in Equations (1) to (10). FPR values are reported as decimals and percentages, while FNR values are reported as percentages. This distinction should be considered when interpreting the error-rate results.

The APsmall, APmedium and APlarge values were interpreted as COCO-style object-size performance indicators. Their lower values are not contradictions of the stronger mAP@0.5 results. Instead, they show that object scale remains a difficult factor in drone-based bird detection. This is expected in aerial imagery, where distant birds may appear as very small targets and may be affected by motion blur, background clutter and changes in flight altitude [4], [8].

3.2.1 Analytical Metric Expressions

The evaluation measures used in this study were defined with standard object-detection metric expressions. In the equations, TP, FP, TN and FN denote true positives, false positives, true negatives and false negatives; Bp and Bg denote predicted and ground-truth bounding boxes; C denotes the number of classes; and APc denotes the average precision for class c.

$$IoU = \frac{Area(Bp \cap Bg)}{Area(Bp \cup Bg)} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4)$$

$$FPR = \frac{FP}{FP + TN} \quad (5)$$

$$FNR = \frac{FN}{FN + TP} \quad (6)$$

$$FPS = \frac{1000}{Latency\_ms/frame} \quad (7)$$

$$mAP@0.5 = (1/C)\, \Sigma[c = 1..C]\, APc(IoU \geq 0.5) \quad (8)$$

$$mAP@0.5{:}0.95 = (1/10C)\, \Sigma[t = 0.50..0.95]\, \Sigma[c = 1..C]\, APc(IoU \geq t), \Delta t = 0.05 \quad (9)$$

$$Split\_train{:}validation = 4{:}1 = 80\%{:}20\% \quad (10)$$

These equation-editor expressions make the analytical definitions explicit while preserving the retained experimental results and comparative interpretation.

3.3 Faster R-CNN with ResNet101 Backbone

Faster R-CNN is a two-stage detector. In the first stage, the ResNet101 backbone extracts deep feature maps and the Region Proposal Network generates candidate object regions. In the second stage, ROI pooling converts the proposed regions into fixed-size representations before classification and bounding-box refinement. The deeper ResNet101 backbone improves feature representation and localization, which is useful for detecting birds with different sizes, poses and levels of occlusion.

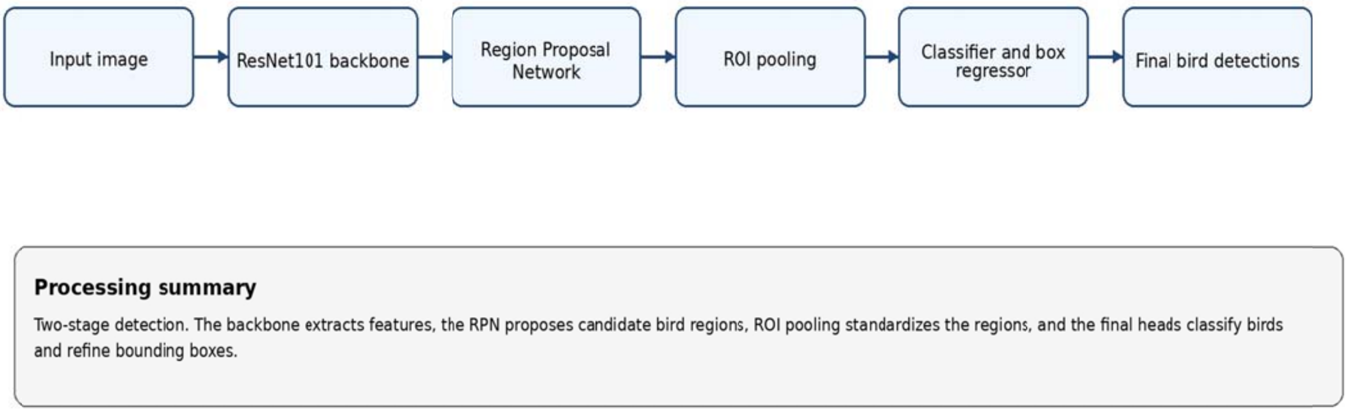

Figure 2. Faster R-CNN with ResNet101 model architecture.

Algorithm 1 summarizes the Faster R-CNN with ResNet101 workflow used in the base-station detector.

1. Input a preprocessed image frame.
2. Extract deep feature maps using the ResNet101 backbone.
3. Generate candidate bird regions using the Region Proposal Network.
4. Apply ROI pooling to standardize the proposed regions.
5. Classify each region and refine the bounding-box coordinates.
6. Apply confidence filtering and output the final bird detections.

3.4 YOLOv7-X / YOLOv7-Large

YOLOv7-Large is a high-capacity one-stage detector optimized for both speed and accuracy. Its backbone uses efficient layer aggregation to preserve gradient flow and strengthen feature representation. The PAN-FPN neck fuses features at different scales, while the detection heads predict bounding-box coordinates, objectness scores and class probabilities. This single-stage design makes YOLOv7-Large faster than two-stage detectors and suitable for real-time base-station inference.

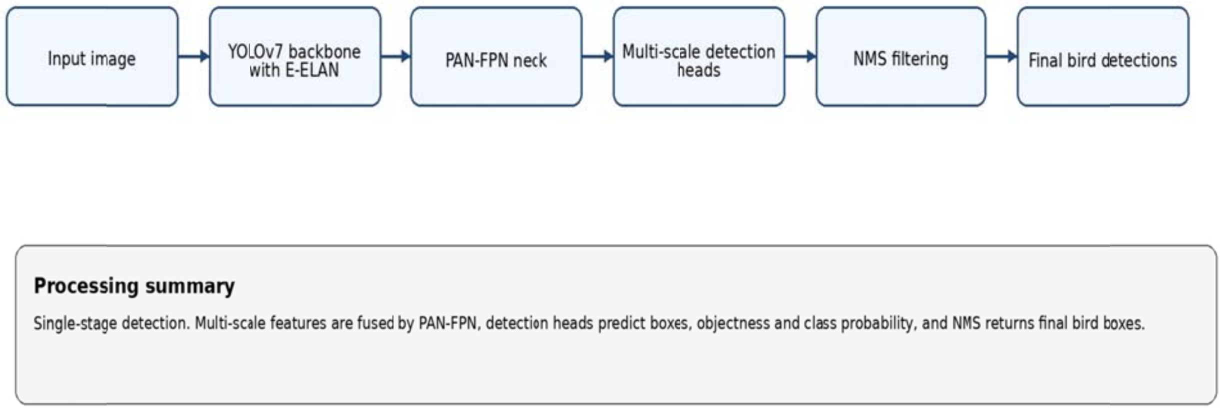

Figure 3. YOLOv7-X/YOLOv7-Large model architecture.

Algorithm 2 summarizes the YOLOv7-Large workflow used in the base-station detector.

1. Input a preprocessed image frame.
2. Resize the frame to the fixed model resolution.
3. Extract hierarchical features using the YOLOv7 backbone with E-ELAN blocks.
4. Fuse multi-scale features using the PAN-FPN neck.
5. Predict bounding-box coordinates, objectness scores and class probabilities through multi-scale detection heads.
6. Apply confidence thresholding and non-maximum suppression.
7. Output the final bird detections.

3.5 Faster R-CNN with ResNet50 Baseline

Faster R-CNN with ResNet50 was retained as a baseline in the comparative result table. It follows the same two-stage detection process as the ResNet101 version but uses a lighter residual backbone. The baseline is included to show whether the deeper ResNet101 backbone improves accuracy and localization enough to justify its higher computation cost. The main comparison in this paper remains YOLOv7-Large against Faster R-CNN with ResNet101.

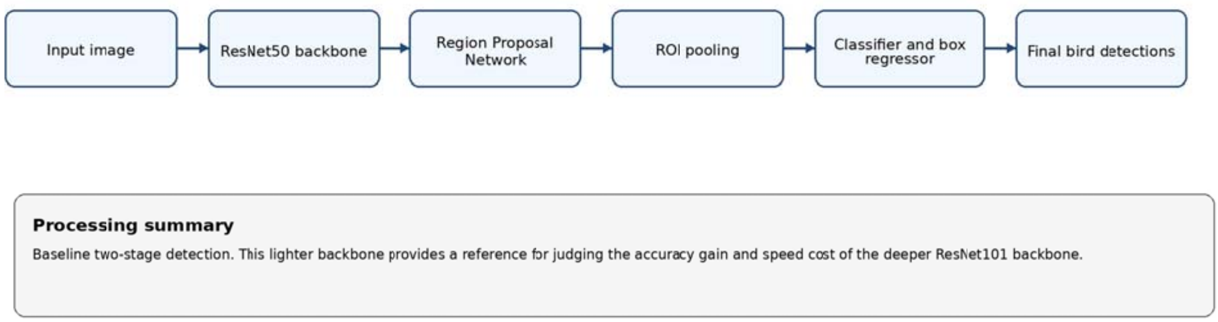

Figure 4. Faster R-CNN with ResNet50 baseline architecture.

4. Results and Discussion

4.1 Faster R-CNN with ResNet101 Results

The implementation and evaluation results for the base-station Faster R-CNN with ResNet101 backbone are shown in Figure 5 and Table 1. The training loss dropped from about 5.2 to 0.3, while the validation loss dropped from about 5.3 to 0.4. The closeness of the training and validation loss curves indicates stable learning and does not show a strong overfitting pattern.

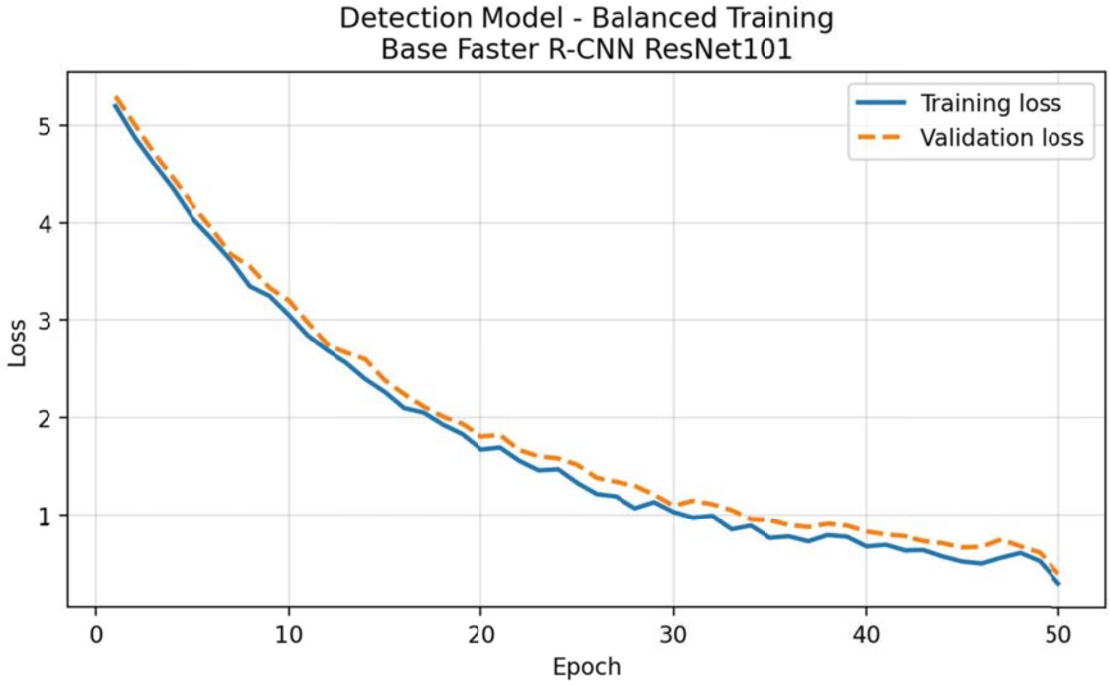

Figure 5. Loss versus epoch plots for Faster R-CNN with ResNet101 backbone.

Table 1. Results for Base Station Model: Faster R-CNN with ResNet101 Backbone

Metric	Value
mAP@0.5 (%)	83.20
mAP@0.5:0.95 (%)	56.90
APsmall (%)	7.80
APmedium (%)	14.60
APlarge (%)	19.80
Mean IoU (matched boxes)	0.873
Precision (%)	84.90
Recall (%)	86.80
F1-score (%)	85.84
False Positive Rate (FPR, decimal)	0.489
False Positive Rate (FPR, %)	48.90
False Negative Rate (FNR) (%)	13.20

The model achieved mAP@0.5 of 83.20% and mAP@0.5:0.95 of 56.90%. The mean IoU of 0.873 shows strong localization quality for matched boxes. The precision of 84.90%, recall of 86.80% and F1-score of 85.84% confirm that the model detected most bird targets while maintaining a reasonable balance between missed detections and false alarms. The result supports the use of a deeper backbone when localization fidelity is required.

4.2 YOLOv7-X / YOLOv7-Large Results

The implementation and evaluation results for the base-station YOLOv7-X/YOLOv7-Large model are presented in Figure 6 and Table 2. The training loss dropped from about 4.2 to 0.3, while the validation loss dropped from about 4.3 to 0.4. The two loss curves remain close across the training epochs, which indicates good generalization on the validation split.

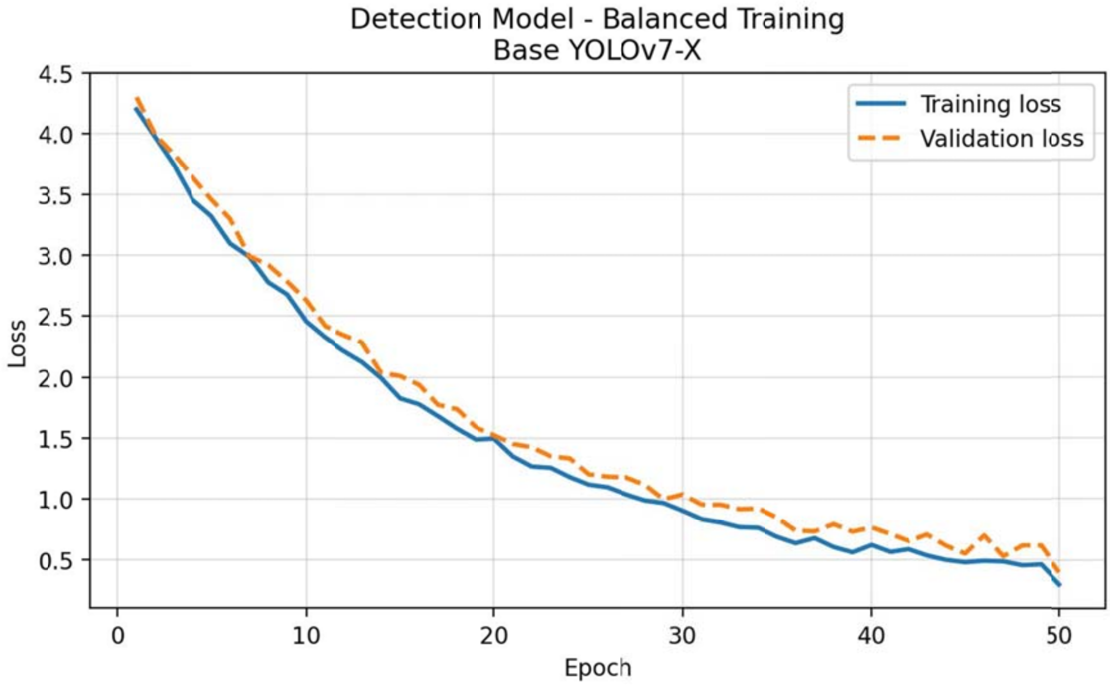

Figure 6. Loss versus epoch plots for YOLOv7-Large.

Table 2. Results for Base Station Model: YOLOv7-X/YOLOv7-Large

Metric	Value
mAP@0.5 (%)	84.90
mAP@0.5:0.95 (%)	59.30
APsmall (%)	9.10
APmedium (%)	16.80
APlarge (%)	22.40
Mean IoU (matched boxes)	0.859
Precision (%)	85.20
Recall (%)	88.10
F1-score (%)	86.63
False Positive Rate (FPR, decimal)	0.469
False Positive Rate (FPR, %)	46.90
False Negative Rate (FNR) (%)	11.90

YOLOv7-Large achieved mAP@0.5 of 84.90% and mAP@0.5:0.95 of 59.30%. It also achieved precision of 85.20%, recall of 88.10% and F1-score of 86.63%. The false negative rate of 11.90% is lower than that of Faster R-CNN with ResNet101, showing that YOLOv7-Large missed fewer birds. This is important in rice farm surveillance because undetected bird pests may continue to damage crops.

4.3 Comparative Performance of Base-Station Models

The comparative detection results are shown in Table 3 and Table 4. Faster R-CNN with ResNet101 improves the results over the Faster R-CNN with ResNet50 baseline because the deeper backbone captures more discriminative spatial features. However, the improvement comes with higher latency and lower FPS. YOLOv7-Large produced the best overall detection accuracy, recall and F1-score while also maintaining the lowest latency among the compared base-station models.

Table 3. Comparative Detection Performance of Base-Station Models on the Balanced Dataset

Metric	Faster R-CNN (ResNet50)	Faster R-CNN (ResNet101)	YOLOv7-X/Large
mAP@0.5 (%)	79.80	83.20	84.90
mAP@0.5:0.95 (%)	52.40	56.90	59.30
APsmall (%)	6.20	7.80	9.10
APmedium (%)	11.40	14.60	16.80

APlarge (%)	16.90	19.80	22.40
Mean IoU	0.845	0.873	0.859
Precision (%)	82.10	84.90	85.20
Recall (%)	84.70	86.80	88.10
F1-score (%)	83.39	85.84	86.63
FPR (decimal)	0.538	0.489	0.469
FNR (%)	15.30	13.20	11.90
FPR (%)	53.80	48.90	46.90

Table 4. Latency-Accuracy Trade-off for Base-Station Detection Models on the Balanced Dataset

Model	Architecture	Latency (ms/frame)	FPS	mAP@0.5 (%)	mAP@0.5:0.95 (%)	Mean IoU	F1-score (%)	Deployment suitability
Faster R-CNN (ResNet50)	Two-stage CNN	118.6	8.4	79.80	52.40	0.845	83.39	High-accuracy offline analysis
Faster R-CNN (ResNet101)	Deep two-stage CNN	156.8	6.4	83.20	56.90	0.873	85.84	Maximum localization fidelity
YOLOv7-X/Large	One-stage CNN	42.9	23.3	84.90	59.30	0.859	86.63	Real-time base-station inference

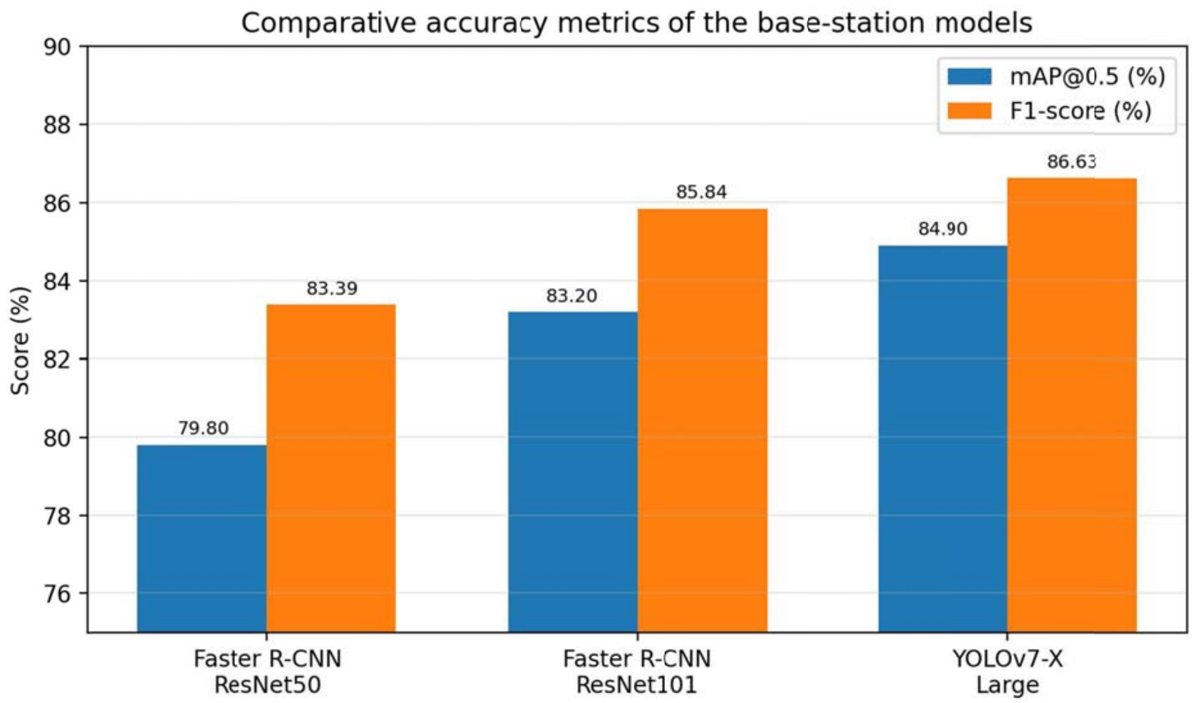

Figure 7. Comparative accuracy metrics of the base-station bird detection models.

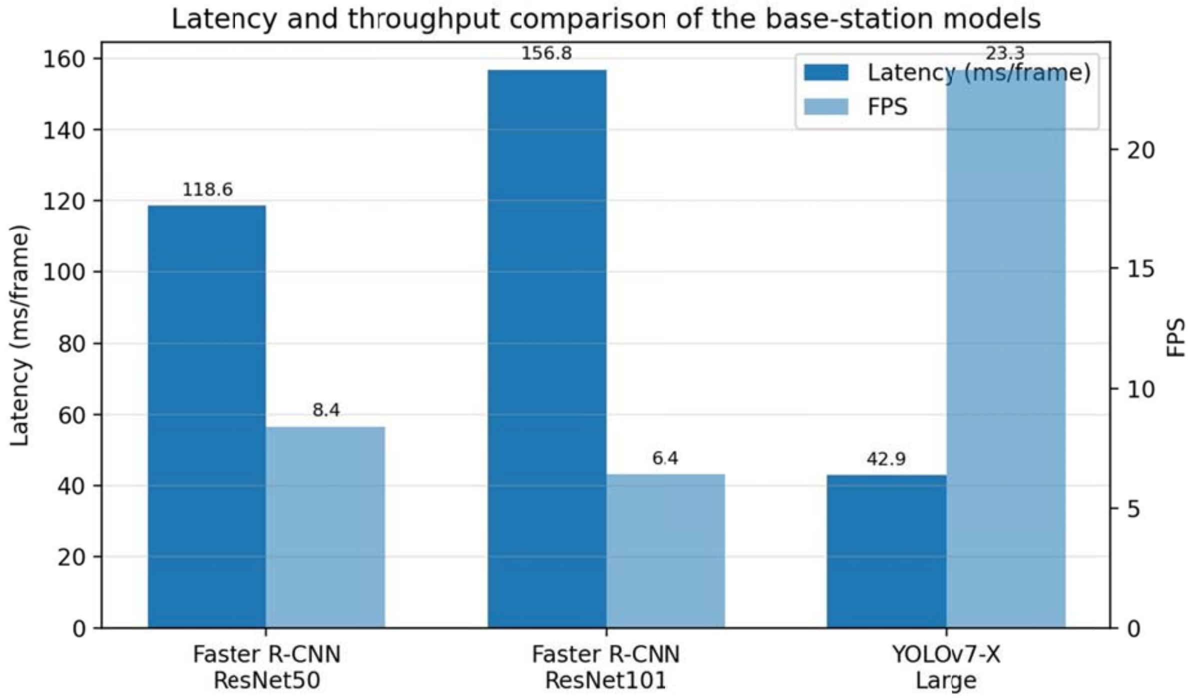

Figure 8. Latency and throughput comparison of the base-station bird detection models.

YOLOv7-Large achieved the highest mAP@0.5, mAP@0.5:0.95, precision, recall and F1-score. It also had the lowest latency of 42.9 ms/frame and the highest throughput of 23.3 FPS. Faster R-CNN with ResNet101 had the highest mean IoU of 0.873, which means it provided very strong bounding-box localization when it detected birds. Nevertheless, its latency of 156.8 ms/frame and throughput of 6.4 FPS make it less suitable for real-time base-station inference than YOLOv7-Large.

The relatively low APsmall, APmedium and APlarge values should be read together with the general mAP and recall results. These scale-specific AP values are stricter COCO-style indicators and show that small target detection remains challenging in UAV imagery. They do not contradict the stronger mAP@0.5 values. Instead, they show that object size remains a difficult factor in drone-based bird detection.

The comparison therefore shows that YOLOv7-Large is the most effective model for the practical base-station deployment scenario. Faster R-CNN with ResNet101 can still be used for offline review, high-fidelity localization or retraining support where speed is not the main constraint.

5. Conclusion

This study presented a comparative evaluation of YOLOv7-Large and Faster R-CNN with ResNet101 backbone for base-station bird pest recognition in rice farms. The retained results show that YOLOv7-Large achieved the best overall performance, with mAP@0.5 of 84.90%, mAP@0.5:0.95 of 59.30%, precision of 85.20%, recall of 88.10%, F1-score of 86.63%, latency of 42.9 ms/frame and throughput of 23.3 FPS. Faster R-CNN with ResNet101 achieved the highest mean IoU of 0.873, which confirms its stronger localization fidelity for matched detections, but its latency of 156.8 ms/frame makes it less suitable for real-time operation. Based on the balance between accuracy and inference speed, YOLOv7-Large is recommended for base-station deployment where rapid recognition of bird pests is required for timely farm response, while Faster R-CNN with ResNet101 remains useful for offline verification where localization precision is the main priority.

The study is limited by the available validation split, incomplete export-level dataset statistics, and the absence of independent field testing across different rice-field locations, seasons, weather conditions and drone altitudes. Future work should validate the recommended model using real-time base-station deployment, larger annotated datasets, additional bird pest species, varied flight altitudes and independent field trials.

References

[1] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137-1149, 2017, doi: 10.1109/TPAMI.2016.2577031.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

[3] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "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, 2023, pp. 7464-7475, doi: 10.1109/CVPR52729.2023.00721.

[4] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in European Conference on Computer Vision, 2014, pp. 740-755, doi: 10.1007/978-3-319-10602-1_48.

[5] Roboflow, "Distant Bird Detection Dataset," Roboflow Universe, accessed 2026.

[6] Z. Khan, Y. Shen, H. Liu, Z. Yang, and I. Hussain, "Object detection in agriculture: A comprehensive review of methods, applications, challenges, and future directions," Agriculture, vol. 15, no. 13, Art. no. 1351, 2025, doi: 10.3390/agriculture15131351.

[7] C. M. Badgujar, A. Poulose, and H. Gan, "Agricultural object detection with You Look Only Once (YOLO) algorithm: A bibliometric and systematic literature review," Computers and Electronics in Agriculture, 2024, doi: 10.1016/j.compag.2024.108757.

[8] Y. Han, C. Wang, H. Luo, H. Wang, Z. Chen, Y. Xia, and L. Yun, "LRDS-YOLO enhances small object detection in UAV aerial images with a lightweight and efficient design," Scientific Reports, vol. 15, Art. no. 22627, 2025, doi: 10.1038/s41598-025-07021-6.

[9] Y.-C. Chen, J.-F. Chu, K.-W. Hsieh, T.-H. Lin, P.-Z. Chang, and Y.-C. Tsai, "Automatic wild bird repellent system that is based on deep-learning-based wild bird detection and integrated with a laser rotation mechanism," Scientific Reports, vol. 14, Art. no. 15924, 2024, doi: 10.1038/s41598-024-66920-2.