Science and Technology Publishing (SCI & TECH)
ISSN: 2632-1017
Vol. 10 Issue 1, January - 2026

Base-Station Faster R-CNN with ResNet50 for Rice Bird Pest Detection Using UAV Aerial Images

Eke Godwin Kelechi[1]

Computer Engineering Department,
Federal Polytechnic Nekede Owerri, FPNO
Imo State, Nigeria.
Email: eke.kelechy@gmail.com

Itoro Kingsley Akpabio[2]

Department of Electrical/Electronic Engineering
University of Uyo , Akwa Ibom State, Nigeria
Emai: itoroakpabio@uniuyo.edu.ng

Kingsley Bassey Clement[3]

Department of Electrical/ Electronic Engineering
University of Uyo, Akwa Ibom State, Nigeria
kingsleyclement@uniuyo.edu.ng

Abstract

This paper presents the training and evaluation of a base-station Faster R-CNN model with a ResNet50 backbone for rice bird pest detection using UAV aerial images. The model is designed for base-station-assisted monitoring, where the UAV captures aerial frames and the base station performs more detailed object detection. The retained dataset split was 80% for training and 20% for validation. The training and validation losses decreased from about 4.6 to 0.3 and from about 4.6 to 0.4, respectively, by about epoch 50. The model achieved mAP@0.5 of 79.80%, mAP@0.5:0.95 of 52.40%, mean IoU of 0.845, precision of 82.10%, recall of 84.70%, and F1-score of 83.39%. However, the low APsmall value of 6.20% shows that small bird targets remain difficult to detect in UAV aerial images. The results show that Faster R-CNN with ResNet50 can support base-station validation of bird detections in smart rice farms, while further improvement is required for small-object detection.

Keywords: *Faster R-CNN, ResNet50, rice bird pest detection, UAV aerial images, base station, computer vision, object detection.*

1. Introduction

Bird pest damage is a major challenge in rice farming because birds can feed on rice grains during vulnerable growth stages and reduce yield when monitoring and deterrent actions are delayed. Manual field patrol is labour intensive and may not provide continuous coverage across large rice farms. UAV-based monitoring provides a practical way to capture aerial images of field conditions and to support faster detection of bird presence in the farm environment.

Deep learning has improved the ability of computer vision systems to identify objects in complex scenes. For rice farm monitoring, object detection is more useful than simple image classification because the system must determine whether birds are present and also localize them within the aerial frame. This makes region-based detection models suitable for base-station processing, especially when the base station has more memory and processing power than the onboard embedded device.

Lightweight onboard models are useful when low latency and low power consumption are the main requirements. However, a base station can host a deeper detector that gives stronger localization, better verification, and more reliable dataset curation. Faster R-CNN with ResNet50 is therefore used in this study as an accuracy-focused base-

station model. The model complements the onboard UAV pipeline by validating difficult frames, correcting false alarms, and generating improved labelled samples for future retraining.

This study focuses on a Faster R-CNN detector with a ResNet50 backbone for rice bird pest detection at the base station. The base station receives UAV image frames and performs high-accuracy bird detection. The detected bounding boxes and confidence scores can support farm alerts, deterrent activation, field supervision, and continuous improvement of the detection dataset.

This paper makes four main contributions. First, it presents a harmonized base-station detection framework for rice bird pest monitoring using UAV aerial images. Second, it develops a system model architecture that shows the relationship among UAV image capture, preprocessing, wireless transmission, base-station detection, alert generation, and dataset curation. Third, it provides a Faster R-CNN with ResNet50 model architecture that clearly describes the backbone, region proposal network, proposal filtering, RoIAlign, and detection head. Finally, it presents a completed methodology and results discussion that retains the reported model performance values and explains their practical meaning for rice farm monitoring.

2. Related Work

Object detection models are widely used where the task requires both recognition and localization. Faster R-CNN introduced a region proposal network that shares convolutional features with the detection network, making two-stage object detection more effective than earlier region-based pipelines. ResNet50 is commonly used as a backbone because its residual connections allow deeper feature extraction while reducing the difficulty of training deep convolutional networks.

For agricultural monitoring, UAV imagery is useful because it can cover large fields and capture information from different viewing positions. However, aerial images may contain small objects, background clutter, motion blur, scale variation, and changing illumination. These challenges are important in bird pest detection because birds may appear as small targets within wide-area rice field images. Recent UAV object detection studies also report that aerial objects are often smaller and more irregularly distributed than objects in ordinary ground-level images. This makes small-object detection and robust localization important for practical deployment.

One-stage models such as YOLO and SSD are often selected for speed, while two-stage models such as Faster R-CNN are usually selected when localization accuracy and proposal refinement are important. In this work, the base-station deployment makes it practical to use Faster R-CNN with ResNet50 because the computation is handled away from the drone. This allows the UAV to conserve onboard resources while the base station performs more detailed detection and verification.

3. Methodology

The proposed methodology uses UAV aerial images as input to a base-station deep learning detector. The UAV captures image frames from rice fields and sends selected frames to the ground station. At the base station, Faster R-CNN with ResNet50 detects birds by generating region proposals, classifying candidate regions, and refining bounding-box coordinates. The model output consists of bird bounding boxes, predicted class labels, and confidence scores.

3.1 System Model Architecture

The system model is shown in Figure 1. The UAV camera captures aerial images of the rice field. The drone performs basic preprocessing such as frame sampling, resizing, and noise reduction before transmitting selected frames to the base station. The base station receives the frames and applies the Faster R-CNN with ResNet50 detector. The output is used for farm monitoring, alert generation, and deterrent decision support.

The feedback loop in the system is important. Verified detections, false positives, false negatives, and difficult bird images are stored at the base station for dataset curation. These curated samples can later be used to improve the

onboard and base-station models. In this way, the base station does not only detect birds; it also supports continuous model improvement and reduces repeated detection errors over time.

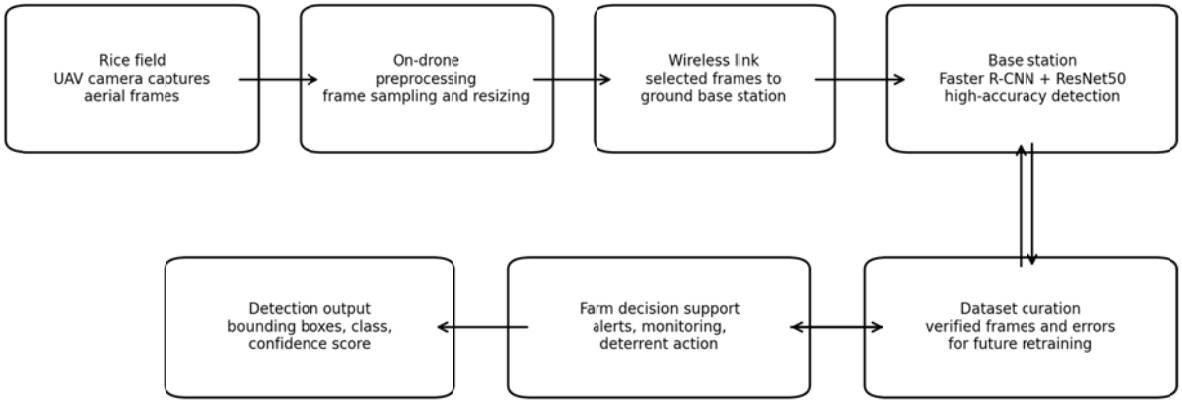

Figure 1. System model architecture for UAV-based rice bird pest detection.

3.2 Dataset Preparation

The UAV aerial image dataset was prepared for supervised object detection. Images containing rice field scenes were annotated with bird bounding boxes and class labels. The retained study reported an 80% training and 20% validation split. The total number of images, total number of annotated bird instances, UAV flight height, image resolution, and exact annotation format were not explicitly reported in the retained results. For this reason, these values are not invented in this paper. The dataset description is therefore limited to the information supported by the retained study.

During preprocessing, the images were organized in a detector-compatible format so that each bird instance could be represented by a bounding box and a class label. The base-station pipeline supports later dataset curation because false positives, false negatives, and difficult small-object cases can be reviewed and added to future training sets.

Table 1. Dataset preparation and splitting procedure.

Dataset item	Description	Purpose
Training set	80% of the annotated UAV images	Used to update Faster R-CNN with ResNet50 parameters.
Validation set	20% of the annotated UAV images	Used to monitor generalization and validation loss.
Labels	Bird bounding boxes and class labels	Used to compute classification and localization losses.
Curated samples	Verified detections and difficult cases	Used to support future retraining and model improvement.

3.3 Faster R-CNN with ResNet50 Model Architecture

The model architecture used in this study is presented in Figure 2. Faster R-CNN is a two-stage detector. Stage 1 uses the region proposal network to generate candidate bird regions from the ResNet50 feature map. Stage 2 uses RoIAlign and the detection head to classify each proposal and refine its bounding-box coordinates. ResNet50 is used as the backbone because it can learn deep hierarchical visual features from UAV frames.

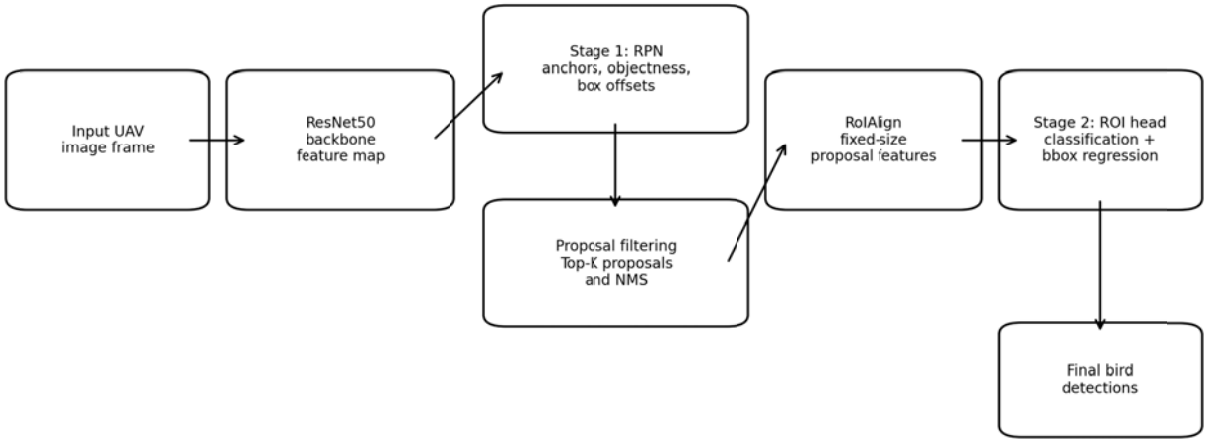

Figure 2. Faster R-CNN with ResNet50 model architecture.

Let I represent an input UAV image. The ResNet50 feature extractor maps the input image to a convolutional feature map F as follows:

$$F = f_{ResNet50}(I; \theta_b) \tag{1}$$

where θ_b represents the backbone parameters. The region proposal network slides over F and predicts an objectness score and bounding-box offsets for each anchor. For an anchor box a = (x_a, y_a, w_a, h_a) and a ground-truth box g = (x, y, w, h), the bounding-box regression targets are defined as:

$$t_x = \frac{x - x_a}{w_a}, \quad t_y = \frac{y - y_a}{h_a} \tag{2}$$

$$t_w = log(\frac{w}{w_a}), \quad t_h = log(\frac{h}{h_a}) \tag{3}$$

The RPN loss combines binary objectness classification and Smooth L1 bounding-box regression:

$$L_{RPN} = \frac{1}{N_{cls}} \Sigma L_{cls}(p_i, p_i*) + \lambda \frac{1}{N_{reg}} \Sigma p_i * L_{reg}(t_i, t_i*) \tag{4}$$

where N_{cls} is the classification normalizing term, N_{reg} is the regression normalizing term, p_i is the predicted probability that anchor i contains an object, p_i* is the anchor label, t_i is the predicted regression vector, t_i* is the target regression vector, and λ controls the balance between classification and localization losses. Candidate proposals are filtered using non-maximum suppression and passed to RoIAlign to generate fixed-size proposal features:

$$z_j = RoIAlign(F, b_j) \tag{5}$$

where b_j is the j-th candidate proposal and z_j is its fixed-size feature representation. The detection head uses the proposal features to predict the bird class probability and refined bounding box. The total detection loss is:

$$L_{total} = L_{RPN} + L_{ROI\_cls} + L_{ROI\_reg} \tag{6}$$

Where L_{ROI_cls} is the ROI classification loss and L_{ROI_reg} is the ROI bounding-box regression loss. The final detection output is expressed as:

$$D = \{(b_k, c_k, s_k)\}_{k=1}^{K} \tag{7}$$

where b_k is the predicted bounding box, c_k is the predicted class, s_k is the confidence score, and K is the number of final detections after filtering.

3.4 Training Configuration and Detection Workflow

The base-station detector was trained using the annotated UAV images for 50 epochs, as reflected in the retained loss curve. The workflow follows feature extraction, anchor generation, RPN prediction, proposal selection, RoIAlign, ROI classification, bounding-box refinement, and parameter update. Non-maximum suppression is applied to remove duplicate proposals before the final detections are returned.

The implementation details are summarized in Table 2. The values reflect the retained study where they were available. Standard Faster R-CNN training settings were adopted where specific implementation details were not explicitly reported in the retained study. This wording is used to avoid unsupported claims about parameters that were not available in the source result description.

Table 2. Base-station Faster R-CNN with ResNet50 training and inference configuration.

Component	Configuration
Detector	Faster R-CNN
Backbone	ResNet50
Deployment level	Base station
Dataset split	80% training and 20% validation
Training duration	50 epochs
Optimization approach	Supervised Faster R-CNN training using classification and bounding-box regression losses
RPN-level non-maximum suppression	IoU threshold of 0.7
Inference proposals	Top-ranked proposals filtered before final detection
Output	Bounding box, predicted class, and confidence score
Optimizer	Not explicitly reported in the retained study.
Learning rate	Not explicitly reported in the retained study.
Batch size	Not explicitly reported in the retained study.
Input image size	Not explicitly reported in the retained study.
Training platform/GPU	Not explicitly reported in the retained study.

3.5 Evaluation Metrics

The model was evaluated using detection and localization metrics. The main metrics are mean average precision at IoU threshold 0.5, mean average precision averaged across IoU thresholds from 0.5 to 0.95, mean IoU for matched boxes, precision, recall, F1-score, false positive rate, and false negative rate. Precision measures the proportion of predicted bird detections that are correct. Recall measures the proportion of actual bird instances detected by the model. F1-score combines precision and recall into one balanced measure.

$$Precision = \frac{TP}{TP + FP} \tag{8}$$

$$Recall = \frac{TP}{TP + FN} \tag{9}$$

$$F1 - score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \tag{10}$$

$$FPR = \frac{FP}{FP + TN}, \quad FNR = \frac{FN}{TP + FN} \tag{11}$$

where TP, FP, TN, and FN denote true positives, false positives, true negatives, and false negatives, respectively.

4. Results and Discussion

4.1 Training and Validation Loss Behaviour

The training and validation loss behaviour of the base-station Faster R-CNN with ResNet50 detector is shown in Figure 3. The results are retained from the uploaded study. The training loss reduced from about 4.6 to 0.3 by about epoch 50, while the validation loss reduced from about 4.6 to 0.4. The closeness of the training and validation curves shows stable learning and does not indicate severe overfitting. This means that the model learned useful visual patterns for bird detection rather than only memorizing the training samples.

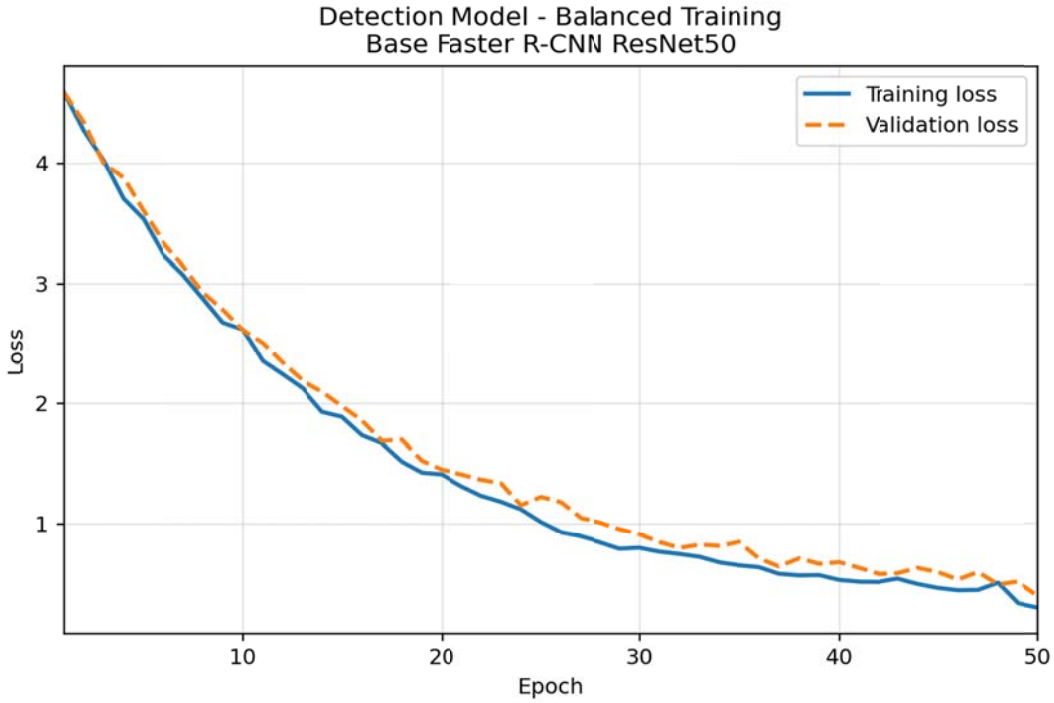

Figure 3. Loss versus epoch plots for Faster R-CNN with ResNet50 backbone model.

4.2 Detection Performance of the Base-Station Model

The retained detection results are presented in Table 3. The model achieved mAP@0.5 of 79.80%, which shows good bird detection ability at the commonly used IoU threshold of 0.5. The mAP@0.5:0.95 value of 52.40% gives a stricter assessment because it averages detection quality over several IoU thresholds. The difference between the two mAP values shows that the detector performs well at moderate localization tolerance but becomes less accurate when stricter bounding-box overlap is required. The mean IoU of 0.845 for matched boxes still shows that many predicted boxes were well aligned with annotated bird locations.

Table 3. Results for base-station model using Faster R-CNN with ResNet50 backbone.

Metric	Value
mAP@0.5 (%)	79.80
mAP@0.5:0.95 (%)	52.40
APsmall (%)	6.20
APmedium (%)	11.40
APlarge (%)	16.90
Mean IoU (matched boxes)	0.845
Precision (%)	82.10
Recall (%)	84.70
F1-score (%)	83.39
False Positive Rate (FPR)	0.538
False Negative Rate (FNR)	0.153

The precision of 82.10% indicates that most detections returned by the model were correct. The recall of 84.70% indicates that the model detected most of the bird instances present in the validation data. The F1-score of 83.39% confirms that the model maintained a reasonable balance between precision and recall. The reported false positive rate is 0.538, while the false negative rate is 0.153. These values show that missed detections and false alarms should still be reduced before large-scale field deployment.

The AP values for small, medium, and large objects show the effect of object scale in UAV images. The APsmall value of 6.20% is low and is the main performance limitation of the detector. This suggests that birds appearing as very small targets in UAV frames were difficult for the detector to localize. The difficulty may be caused by high flight altitude, low pixel coverage, background similarity, motion blur, and a limited number of small-object

samples in training. Future work should therefore test feature pyramid backbones, higher-resolution training, improved anchor scales, stronger image augmentation, and more annotated small-bird samples.

4.3 Practical Implications for Smart Rice Farms

The results show that a base-station Faster R-CNN with ResNet50 detector can serve as the high-accuracy perception unit in a UAV-based rice bird pest monitoring system. The base station can validate suspicious frames from the drone, support near-real-time alerts, and provide reliable labels for continuous model improvement. In practice, the model can be connected to deterrent systems, farm dashboards, or decision-support platforms that notify farmers when bird activity is detected in the rice field.

The base-station deployment strategy is suitable because the detector is more computationally demanding than lightweight onboard models. The drone can focus on image capture and basic preprocessing, while the base station performs detailed detection. This separation improves the balance between UAV energy consumption, detection accuracy, and operational usefulness.

5. Conclusion

This paper presented a revised and harmonized base-station Faster R-CNN with ResNet50 framework for rice bird pest detection using UAV aerial images. The model is designed for base-station deployment, where more computational resources are available for accurate object localization and validation of UAV-captured frames.

The retained results show that the training loss decreased from about 4.6 to 0.3 and the validation loss decreased from about 4.6 to 0.4 by about epoch 50. The detector achieved mAP@0.5 of 79.80%, mAP@0.5:0.95 of 52.40%, mean IoU of 0.845, precision of 82.10%, recall of 84.70%, and F1-score of 83.39%. These values show that the Faster R-CNN with ResNet50 model is effective for base-station bird pest detection in UAV-based smart rice farm monitoring.

Future work should improve the detection of small birds by increasing the number of annotated small-object samples, using higher-resolution UAV frames, testing feature pyramid backbones, optimizing anchor scales, and tuning detection thresholds. Additional studies should also report inference speed, UAV flight height, image resolution, training hardware, and full dataset size to improve reproducibility.

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] R. Girshick, "Fast R-CNN," in Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1440-1448, doi: 10.1109/ICCV.2015.169.

[4] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2117-2125, doi: 10.1109/CVPR.2017.106.

[5] M. Tan, R. Pang, and Q. V. Le, "EfficientDet: Scalable and Efficient Object Detection," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 10781-10790, doi: 10.1109/CVPR42600.2020.01079.

[6] D. Avola, L. Cinque, A. Diko, A. Fagioli, G. L. Foresti, A. Mecca, D. Pannone, and C. Piciarelli, "MS-Faster R-CNN: Multi-stream backbone for improved Faster R-CNN object detection and aerial tracking from UAV images," Remote Sensing, vol. 13, no. 9, 1670, 2021, doi: 10.3390/rs13091670.

[7] A. Rejeb, A. Abdollahi, K. Rejeb, and H. Treiblmaier, "Drones in agriculture: A review and bibliometric analysis," Computers and Electronics in Agriculture, vol. 198, 107017, 2022, doi: 10.1016/j.compag.2022.107017.

[8] Z. Li, Y. Wang, N. Zhang, Y. Zhang, Z. Zhao, D. Xu, G. Ben, and Y. Gao, "Deep learning-based object detection techniques for remote sensing images: A survey," Remote Sensing, vol. 14, no. 10, 2385, 2022, doi: 10.3390/rs14102385.

[9] G. Tang, Z. Guo, Y. Yang, Y. Zhu, and W. Ding, "A survey of object detection for UAVs based on deep learning," Remote Sensing, vol. 16, no. 1, 149, 2024, doi: 10.3390/rs16010149.

[10] Z. Cao, C. Xu, L. Yang, and J. Zhang, "Real-time object detection based on UAV remote sensing: A review," Drones, vol. 7, no. 10, 620, 2023, doi: 10.3390/drones7100620.