RESEARCH ARTICLE

RSP Science Hub

International Research Journal on Advanced Science Hub
2582-4376
www.rspsciencehub.com
Vol. 07, Issue 09 September

Check for updates

http://dx.doi.org/10.47392/IRJASH.2025.085

Comparative Analysis of Object Detection Models for Peacock Detection: Evaluating Their Performance and Key Points

Ankith Manchale[1], Mohith G K[2], Dr. Suganeshwari G[3]
[1,2]Student, Vellore Institute of Technology, Chennai India.
[3]Associate Professor, Vellore Institute of Technology, Chennai India.
Email ID: *ankith.manchale2022@vitstudent.ac.in[1], mohith.gk2022@vitstudent.ac.in[2], suganeshwari.g@vit.ac.in[3]*

Article history
Received: 14 August 2025
Accepted: 28 August 2025
Published:04 September 2025

Keywords:
Object Detection, Peacock, Yolo, Bird Detection

Abstract
Peacocks being a major contributor in the damage caused in the agriculture sector, an imminent requirement of efficient detection system for timely intervention. Object Detection being an important application of Computer Vision aims in precisely finding the location and identifying the object in images and videos. With increased advancements in the deep learning field and with various models, choosing the best model that not only performs accurate object detection but is also evaluated based on its inference time. This research aims in conducting an analysis on various models in market on the application of peacock detection evaluating them based on accuracy, precision, recall and F1-score. The Yolo11 yields the highest result with the accuracy of 84.9%. The detailed comparison with various evaluation metrics gives the efficient solution in mitigating this problem in-hand saving the agriculture from further damage and incurring future losses.

1. Introduction

Peacock detection in farming fields plays a very significant role in agriculture sector. Peacocks can help the farmers in intensifying the soil's fertility and also aid in natural pest control, it also could cause harm to the produce if not monitored timely. A decrease of 40% in the agriculture is seen with peahens and peacocks left in the field during the harvest. In the advancing field of Deep learning, the Computer Vision domain gained various improvements in its object detection models, with a huge number of models in the market, it's a complicated task to choose the highly efficient model[1]. Safeguarding the produce by farmers, various methods is practiced to veer off the peacocks from the farms. The detection of peacocks using various object detection models is majorly used to locate and identify peacocks in the farms. [2] To meet these challenges researchers are increasingly developing models and advancing improving versions of them. The flexibility, and the independence from third parties makes it easy to implement these solutions in real-time. Among the commonly used models YOLO, RCNN, Faster RCNN, and various other models are implemented in this research and compared using multiple evaluation criteria. The evaluation of the research will be a breakthrough in the field of object detection in the birds species domain focusing on the current problems and meeting the demands of the society, the research will provide a foundation to newer technologies and help the current third party users to make full use of the research helping them thrive a better decision for the protection and development of the their local area.[3]

2. Literature Survey

In this the recent years, object detection major component of Computer Vision, helping the computers to mimic vision like humans, marking objects and identifying their locations. Over the years, there has been an immense advancement in the object detection algorithms and a notable increase in the available models on the market. Object Detection is divided into two segments based on how many times the input is passed through the network i.e, one shot detector and two shot detector. Single stage detectors with a basic design analyze all parts of an image at once to look for objects. The most significant features of two-stage detectors are two passes and elaborate construction. to offer different things to do in various areas. In recent years, the process of identifying objects in different applications has been sharply revised to make it more accurate and efficient. Many current systems that recognize objects instantly such as YOLO and Faster R-CNN, are widely applied. Because of their adaptability and reliability, these algorithms are put to work in healthcare, smart cities, surveillance and for driverless cars.[1] Traditional ways of detecting objects were slow because many images needed to be analyzed to recognize them. As a result of region-based strategies, applications could not respond in real time and were slow due to high computations.
Yolo made detection much faster by using a one-shot detection method instead. Real-time recognition of items in changing environments is made simple with YOLO, as it subdivides pictures into cells and processes predictions inside each cell instead of at the whole image level. After debuting, the YOLO system has seen several updates, each making it better at handling problems and speeding up detection. Starting with YOLOv1 and continuing through the most recent YOLOv8, this review aims to describe every version, discussing each advance and difference in detail [5].

3. Methodology

This research aims in analysis of various object detections models and compares them using multiple evaluation metrics. The dataset consists of 1472 images of size 640x640. The dataset is then read from its path and corresponding labels. These paths and labels are stored in respected variables. This variable is then considered for training and testing in the following stages. Image Processing functionality loads each image, converts it to grayscale, and reshapes dimensions for use in various models implemented. The various models used for object detection are listed below. The algorithm of the model is also explained for a better and depth understanding on the complexity and working of the model.

3.1. RCNN

RCNN, which expands as Region-based Convolutional Neural Network was introduced by Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. The main objective of RCNN is to perform object detection by combining region proposals and convolutional neural networks (CNNs) in an end-to-end framework. RCNN processes an image to identify object regions, and then classifies these regions using a CNN, resulting in accurate object detection [7].
Algorithm followed by RCNN

- **Region Proposal:** RCNN uses a technique called Selective Search to generate potential object regions from the given input image. This technique splits the image into superpixels and merges them to form region proposals. [8]
- **Feature Extraction:** Each region proposition is passed through a pre-trained CNN (like AlexNet) to extract deep features from the image.
- **Object Classification:** The extracted features are then forwarded to a Support Vector Machine (SVM) to classify the regions into their respective object categories. [12]
- **Bounding Box Regression:** After classification, a linear regression model is used to fine-tune the bounding boxes and improve their accuracy.

3.2. Faster-RCNN

Faster R-CNN, which stands for “Faster Region Convolutional Neural Network” is an advanced object detection variant of the R-CNN family, introduced by Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. The main objective of the Faster R-CNN network is to come up with a unified structure that not only detects objects in an image but also finds the position of objects accurately in the image. It features the advantages of convolutional neural networks (CNNs), and region proposal networks(RPNs) into a single network, which drastically improves the speed and accuracy of the model [9].

3.2.1. Algorithm Followed by RCNN

- **Selective Search Algorithm:** Used to extract multiple candidate region recommendations from the input image. In the initial sub-segmentation, this technique will generate a large number of potential regions. Ten, comparable regions are merged to create larger areas with an algorithm that is greedy. The final region proposals are composed of these regions.

As a vector output, the CNN component distorts the suggestions and abstracts unique properties. To identify objects of relevance in the proposal, the retrieved features are passed into an SVM (Support Vector Machine). [13]

3.3. Masked RCNN

Mask R-CNN, which stands Mask Region-Convolutional Neural Network, is an continuation of Faster R-CNN, introduced by Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross B. Girshick. The main motive of the Mask R-CNN is not only detecting objects and locate them but also to split them at the pixel level, giving each object in the picture a comprehensive mask. By including a branch that creates object masks together with the object identification task, Mask R-CNN expands on Faster R-CNN [10].

3.3.1. Algorithm Followed by Mask R-CNN

- **Region Proposal Network (RPN):** Just like Faster R CNN, Mask R-CNN uses a Region Proposal Network (RPN) to suggest potential object areas from the input picture. The RPN forms a set of bounding boxes that most likely contain objects, using the feature maps abstracted by the backbone network.
- **RoI Align:** Mask R-CNN replaces RoI Pooling in faster R-CNN with RoI Align, which accurately avoids quantization and extract features for every region suggestion. Both the mask prediction and classification tasks become more accurate as a result.
- **Object Classification and Bounding Box Regression:** Mask R-CNN, like Faster R-CNN, classifies each region proposition and improves localization by fine-tuning the bounding box coordinates. The class label for the object in the region is then predicted by the network.
- **Mask Generation:** The branch in Mask R-CNN that makes pixel-wise segmentation masks for every object is a crucial addition. This branch creates a binary mask for every object in the image using the region suggestions from the RPN. Each mask offers a fine-grained segmentation by accurately capturing the contour of the item in high detail.
- **Final Results:** Mask R-CNN produces the final predictions, which include a segmentation mask for each object in the picture, the class label, and a revised bounding box. [14]

3.4. YOLOv8

YOLOv8 is the latest iteration of the YOLO family, carrying on the tradition of increasing speed and accuracy while optimizing for real-time object recognition. With an emphasis on improved quality outcomes, quicker inference, and broader use cases, YOLOv8 integrates a number of state of-the-art innovations [11].

Algorithm followed by YOLOv8:

- **Advanced Backbone (EfficientNet or Similar):** YOLOv8 uses a more efficient backbone network for quicker feature extraction, often integrating EfficientNet or other architectures for best performance.
- **Transformer Integration:** For improved contextual understanding, YOLOv8 adds transformers to the network design. This response improves object detection, even in settings that are complicated and crowded.
- **Dynamic Anchor Assignation:** By dynamically modifying anchor boxes during training to better fit the data, YOLOv8 improves the anchor box design and aids in accuracy for objects of different sizes.
- **Improved Augmentation Techniques:** YOLOv8 incorporates advanced data augmentation techniques, including more diverse and realistic augmentation strategies, to help the model generalize better on unseen data.
- **Efficient Inference:** YOLOv8 introduces techniques like knowledge distillation and quantization to speed up inference while retaining high accuracy, making it more suitable for edge devices and real-time applications.
- **Attention Mechanisms:** YOLOv8 applies

spatial and channel attention mechanisms to allow the model to focus on important regions and features, improving both detection performance and robustness.

3.5. YOLOV9

YOLOv9 is a step beyond the scope of real-time object detection by performing faster, making more accurate predictions and operating more resiliently on different applications. It is an extension of success of YOLOv8 and includes even more innovatory methods and developments to improve performance. Next-gen Backbone (Vision Transformer Hybrid): YOLOv9 unites the capabilities of CNNs and Vision Transformers (ViTs) in a hybrid backbone design in a bid to enable better feature extraction. The design enables the network to process both local and global information in the images in a better way [13] .

3.5.1. Features of YOLOV9

- **Self-Supervised Learning:** YOLOv9 is based on self-supervised learning, so that the model will be able to make the most out of unlabeled data. This enhances generalization and, as such, dependence on vast labeled data to develop the model, a fact that makes it more flexible. [15]
- **Adaptive Anchor Clustering:** YOLOv9 improves the assignment of anchor boxes with the dynamic and adaptive clustering method on training. This assists the model to choose anchors more intelligently in accordance with the distribution in the object that enhances the performance of the model across different scales of objects
- **Augmentation 2.0:** Having the advanced augmentation pipeline with the synthetic data generation, YOLOv9 is capable of coping with a variety of real-world variations more efficiently, and its generalization is quite high, particularly in regard to edge cases. [16]
- **Edge Optimizations:** YOLOv9 also proposes faster inference with better optimizations by offering more effective inference techniques such as model pruning and knowledge distillation to guarantee the high speed of model inference even on the edges as well as cloud environments.
- **Better Attention Mechanisms:** YOLOv9 has superior detection because of its strong attention mechanisms, making the model decide where to apply its attention on problematic areas, diminishing false positives and having stronger resistance to dense settings. [16]

3.6. YOLOV11

YOLOv11 brings the YOLO architecture to a new era, and one that focuses on versatility, scalability, and performance at the next level. This release has been built to cover both high-accuracy surveillance and low-latency autonomous use cases and we are testing the limits of detection accuracy and at the same time ensuring that we continue to be real-time optimized. High-performance Backbone (EfficientNetV2 and ViT Hybrid): YOLOv11 uses an even stronger backbone methodology, combining the high-performance and high speed approach of EfficientNetV2 with global coherent representation of Vision Transformers, which may be very useful in dense or complex scenes [15]

Some of the features of the model are given below:

- **Multimodal Data Fusion:** YOLOv11 does not limit visual data since it uses multimodal input, such as LiDAR and radar data, to find objects in dimly lit situations or behind boundaries to deal with autonomous driving and robotics. [17]
- **Dynamic Anchor Optimization:** YOLOv11 learns to place the anchor boxes in a more optimized way with the help of dynamic algorithm where anchors are customised on a real-time basis depending on the features of a given object leading to greater accuracy in different cases, including small objects and big buildings.
- **Independent Data Augmentation:** YOLOv11 uses advanced technologies in augmentation such as MixUp and CutMix, to ensure it produces stronger training data, enabling the model to generalize to a broad variety of contexts and setting.
- **Hardware Acceleration:** With inference optimization, being a crucial part of the inference process, YOLOv11 offers sophisticated hardware-accelerated inference with perfect hardware-acceleration methods, including TensorRT and GPU optimizations, to optimize the results to the maximum even when it is

required on the restricted devices. [18]

- **Cross-Scale Attention Networks:** YOLOv11 involves a cross-scale attention mechanism so that the model may focus on important features at different scales to better support the detection of objects of various sizes and locations within images.

3.7. YOLOV12

The ending of the technology on object detection is the YOLOv12 as it is both very fast and very close to perfection with regard to all the various activities. The design intended high end-use and the license maximized high-definition object detection on the basis of a balancing architecture to be used in most demanding environments. Super-efficient Backbone (MobileNetV3 and ViT Fusion): it is founded on the confidential features of YOLOv12 which include lightweight computation with MobileNetV3 and high-power global feature extraction with Vision Transformer. This hybrid backbone gives you an assurance that not only availing of YOLOv12 in an edge device would perform well, but also in a cloud device [16].

Some features of Yolo12 Model are as follows:

- **AI-Helped Preprocessing:** In order to implement it even more conveniently in the real setting, YOLOv12 introduces the AI-based preprocessing pipeline that will help normalize this image based on the conditions of the environment (the quality of the lighting conditions, the presence of noise, etc.). [19]
- **Smart Anchor Learning:** YOLOv12 goes even further than anchor learning by making anchor mechanism dynamic during training to not only learn a set of anchor box configuration varying over training. This results in more accuracy in prediction of a greater range of object sizes and shape.
- **Augmentation:** The model employs the use of high-level augmentation that it synthesizes training data. This is useful in improving model robustness especially using rare classes of objects. [20]
- **Real-time and Low Latency:** YOLOv12 has many real time optimization mechanisms, including quantization-aware training. Besides, it responds with a low latency inference application on mobile devices or even real-time video perception.
- **Multi-Resolution Feature pyramid Networks (FPN):** YOLOv12 implements multi-resolution FPNs, influencing the superior performance rates of the model on the capabilities to recognize the objects of various resolutions within the same image, which significantly improves the detection of objects in images of a differing size

4. Implementation And Result

This section provides the experimental analysis of multiple object detection models, that is, Faster R-CNN, Masked R-CNN, YOLO v8, YOLO v9, YO YOv11, and YO YOv12, on the dataset on peacock detection. The basic Recurrent Neural Network (RNN) as a baseline classifier was employed in binary classification as well. They tested the models on a consistent dataset of 62 validation images, and the key performance metrics were the F1-score, precision, and the recall. YOLOv11 performed better, having precision of 0.8499, recall of 0.5028, and F1-score of 0.7130, than all the other models as far as the overall performance is concerned. This implies that YOLOv11 is the most effective when it comes to balance between accuracy of predictions and effective object coverage. It is reliable in the minimisation of the false positives in the detection process especially due to the high precision. Having a precision of 0.8119, many of the discovered real time possibilities of YOLOv8, recall of 0.5178, and F1-score of 0.7807. Among the models, it had to yield the fastest inference speed (13 ms per image), which is why it is best suited to any real-time applications where reaction speed is of the essence. The F1-score achieved on YOLOv9 was 0.7019, although the model has led to moderate precision (0.7019) and recall (0.5392). This model provided a normally balanced detection scheme and that worked well in both the human as well as the peacock classes. Faster R-CNN, as well as Masked R-CNN, are two typical detection models the competitive precision scores of which were 0.7868 and 0.6193 respectively. The recall did better on Faster R-CNN (0.7329) than on Masked R-CNN (0.7466). Hence, Masked R-CNN gained F1-score of 0.6770, and Faster R-CNN achieved F1-score of 0.7589. These models only exhibited uniform localization; however, their predictions at class level differed. provides the experimental analysis Table 1 shows Result & Model

Table 1 Result & Model

Result Evaluation of Peacock Detection Models			
Model	**Precision**	**Recall**	**F1-Score**
Faster R-CNN	0.7868	0.7329	0.7589
Masked R-CNN	0.6193	0.7466	0.677
R-CNN	0.5392	1	0.7007
YOLOv8	0.8119	0.5178	0.7807
YOLOv9	0.7019	0.5392	0.7019
YOLOv11	0.8499	0.5028	0.713
YOLOv12	0.838	0.5079	0.758

In binary classification (presence vs. absence), the RNN classifier achieved an F1-score score of 0.7007, precision of 0.5392, and a recall of 1.0000. The recall of the RNN is considerable signifying that this is high on overestimating existence of objects thus high overall detection but less exactness. The F1-confidence curve demonstrates the relationship between F1-score and confidence threshold with the change of the latter, which may be referred to as the precision-recall tradeoff of YOLOv11. Figure 1 shows Confidence Precision Curve of YOLOv11 Figure 2 shows Confidence Precision Curve of YOLOv11

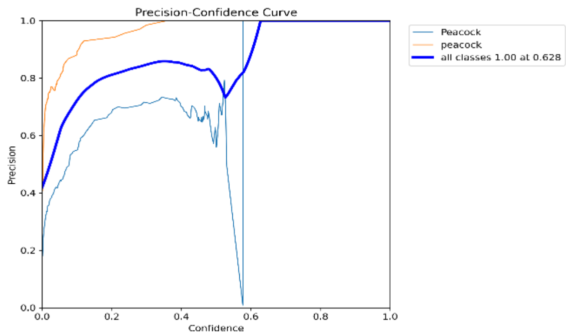

Figure 1 Confidence Precision Curve of YOLOv11

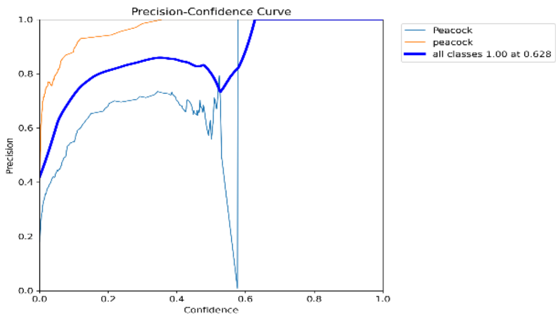

Figure 2 Confidence Precision Curve of YOLOv11

The precision-confidence curve is illustrated in the YOLOv11 and depicted the highest precision of 0.818 in the peacock class, which means a low number of false positives in the peacock class. Figure 3 shows Consider this Recall Confidence Curve of Yolov11

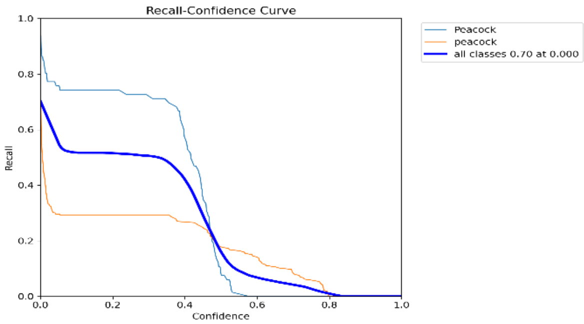

Figure 3 Consider this Recall Confidence Curve of Yolov11

The recall curve indicates that the overall detection rate of the peacocks is very high, as the value in the recall curve ranges between 0.4786 to 0.793, ensuring the practical potential of the model in the detection of objects. Figure 4 shows Rec Curve of YOLOv11

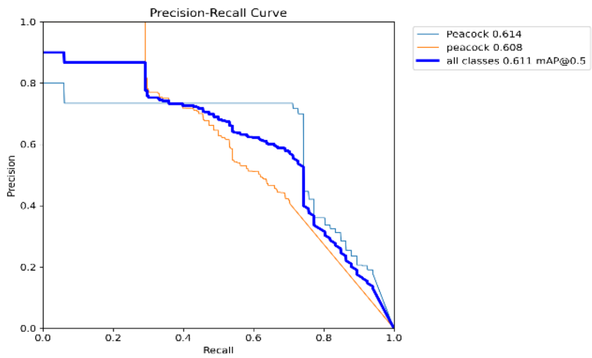

Figure 4 Rec Curve of YOLOv11

This graph shows the change in precision and recalls together with thresholds and proves the overall efficacy of YOLOv11. Figure 5 shows Identification of Sample Images

Figure 5 Identification of Sample Images

The samples of detection reveal that the YOLOv11 can indeed accomplish to specify and classify peacocks in various conditions even when used in real-time period through a live camera.

Conclusion

The present work features a comprehensive assessment of a number of recent state-of-the-art deep learning approaches to the problem of real-time peacock detection with the view to facilitating its use in wildlife surveys, smart agriculture, and automated tracking of wildlife in nature parks. The architectures to be tested will be region-based (Faster R-CNN, Masked R-CNN, R-CNN) and the YOLO-based (YOLOv8 to YOLOv12). The results in the evaluation give adequate reason to choose the best model especially because of separating the peacocks with other birds species-- this is very essential in avoiding misdetection during practical deployment activated the most tests (1.0), meaning that it did not miss any instances, but it has a lot of false positives, which should not be used when it is imperative that specific classification is made. YOLOv8 offered the overall best tradeoff, having high values of the F1-score (0.7807) and precision (0.8119), and becoming a highly credible choice for real-time application. YOLOv12 and Faster R-CNN were also able to compete with F1-scores of 0.758 and 0.7589 respectively. All in all, the results of this evaluation would be used to build an ideal model which not only identifies the peacocks correctly but also reduce mismarking of other birds. The YOLOv8 and YOLOv11 are especially well balanced in terms of accuracy versus efficiency and hence are especially capable to be used in mobile and edge devices in environments where resources are limited.

Future Work

The need to view peacock in the agricultural landscape involves many future research opportunities. A major one will be the incorporation of in real-time surveillance systems the farmers will be able to react to spotted presence of peacocks and enhance the crop protection measures. Also, the investigation of state-of-the-art machine learning strategies, e.g., transfer learning, can make the model more adaptive by being able to detect novel peacock behaviours in different conditions of farming. The data collected should be multimodal by incorporating physical media, e.g. by including visual data with audio clues, including the calls of the peacocks. Another urgent direction is creating less demanding models that would run on mobile devices or IoT implementations to make them easier to access by the farmers. These systems will require field trials to test the real-life situation and this will be facilitated by the feedback of farmers to make the necessary adjustments. Finally, the cooperation with the agricultural stakeholders (farmers, scientists) will be essential to finding working solutions to the problem of peacock management. Following such directions, future research could strongly improve peacock detection systems and make the agricultural industry more sustainable.

References

[1]. https://www.downtoearth. org.in/ wildlife-biodiversity /researchers-quantify-damage-caused-by-peafowl-to-farmers-62441:text= Farmers % 20are%20hit %20hardest%20as,trampling%20on%20an d%20dislodging%20seedlings.

[2]. Thangaraj, Rajasekaran, et al. "Bird Species Detection Using Deep Learning Techniques." 2025 3rd International Conference on Intelligent Systems,

Advanced Computing and Communication (ISACC). IEEE, 2025.

[3]. Vijayakumar, Ajantha, and Subramaniyaswamy Vairavasundaram. "Yolo-based object detection models: A review and its applications." Multimedia Tools and Applications 83.35 (2024): 83535-83574.

[4]. Gui, Shengxi, et al. "Remote sensing object detection in the deep learning era—a review." Remote Sensing 16.2 (2024): 327.

[5]. Sun, Yibo, Zhe Sun, and Weitong Chen. "The evolution of object detection methods." Engineering Applications of Artificial Intelligence 133 (2024): 108458.

[6]. Ariza-Sentís, Mar, et al. "Object detection and tracking in Precision Farming: A systematic review." Computers and Electronics in Agriculture 219 (2024): 108757.

[7]. Chen, Wei, et al. "A review of object detection: Datasets, performance evaluation, architecture, applications and current trends." Multimedia Tools and Applications 83.24 (2024): 65603-65661.

[8]. Wang, Chien-Yao, and Hong-Yuan Mark Liao. "YOLOv1 to YOLOv10: The fastest and most accurate real-time object detection systems." APSIPA Transactions on Signal and Information Processing 13.1 (2024).

[9]. Alazeb, Abdulwahab, et al. "Remote intelligent perception system for multi-object detection." Frontiers in Neurorobotics 18 (2024): 1398703.

[10]. Diwan, Tausif, G. Anirudh, and Jitendra V. Tembhurne. "Object detection using YOLO: challenges, architectural successors, datasets and applications." multimedia Tools and Applications 82.6 (2023): 9243-9275.

[11]. Liu, Chengji, et al. "Object detection based on YOLO network." 2018 IEEE 4th information technology and mechatronics engineering conference (ITOEC). IEEE, 2018.

[12]. Vijayakumar, Ajantha, and Subramaniyaswamy Vairavasundaram. "Yolo-based object detection models: A review and its applications." Multimedia Tools and Applications 83.35 (2024): 83535-83574.

[13]. Shandilya, Shishir Kumar, et al. "YOLO-based segmented dataset for drone vs. bird detection for deep and machine learning algorithms." Data in Brief 50 (2023): 109355.

[14]. Mpouziotas, Dimitrios, et al. "Automated wildlife bird detection from drone footage using computer vision techniques." Applied Sciences 13.13 (2023): 7787.

[15]. Sun, Zi-Wei, et al. "Flying bird object detection algorithm in surveillance video based on motion information." IEEE Transactions on Instrumentation and Measurement 73 (2023): 1-15.

[16]. Chen, Xian, et al. "An efficient method for monitoring birds based on object detection and multi-object tracking networks." Animals 13.10 (2023): 1713.

[17]. Varghese, Rejin, and M. Sambath. "Yolov8: A novel object detection algorithm with enhanced performance and robustness." 2024 International conference on advances in data engineering and intelligent computing systems (ADICS). IEEE, 2024.

[18]. He, Lu-hao, et al. "Research on object detection and recognition in remote sensing images based on YOLOv11." Scientific Reports 15.1 (2025): 14032.

[19]. Mao, Makara, and Min Hong. "YOLO object detection for real-time fabric defect inspection in the textile industry: A review of YOLOv1 to YOLOv11." Sensors (Basel, Switzerland) 25.7 (2025): 2270.

[20]. Lee, Yong-Hwan, and Youngseop Kim. "Comparison of CNN and YOLO for Object Detection." Journal of the semiconductor & display technology 19.1 (2020): 85-92.