IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Wildlife Detection with Drone Images using Enhanced YOLO 11

SHERLY. A, CHITRANSHU GUPTA, MOHD KAMRAN WARSI, AND KARTIK BAGHEL.
School of Computer Science and Engineering, Vellore Institute of Technology, Chennai 600127, India
Corresponding author: Sherly. A (sherly.a@vit.ac.in).

ABSTRACT Drones are increasingly applied in wildlife monitoring. Deep learning is the backbone behind this endeavor, showing vast capability but encountering limitations in detecting very small objects. To counter such a shortcoming, YOLO 11 is proposed by combining the current state-of-the-art real-time object detection system with channel self-attention mechanism. The model also applies SimAM for enhancing feature representation, DySample for adaptive sampling, and Bounding Box-IoU for accurate localization. These upgrades significantly enhance the discovery of little objects in complicated surroundings. The lack of free, publicly accessible aerial datasets has been one of the biggest hurdles to the implementation of UAV-based wild animal monitoring. To fill this gap, WAID—Wildlife Aerial Images from Drone—is introduced as a massive collection of aerial photos taken under diverse climatic conditions. The dataset features six diverse animal species inhabiting disparate habitats. Model performance is assessed using intensive comparative tests and statistical evaluation, proving the superiority of enhanced YOLO 11 over more sophisticated algorithms. Through the incorporation of new methods and datasets, this research makes a contribution to the development of UAV-based wildlife monitoring, enhancing detection accuracy and generalization in real-world environments.

INDEX TERMS YOLO, neural network, image detection, UAV, wildlife, CNN, SimaM, DySample, Bounding Box IoU, Enhanced SSD, SPPF, CBS, CSP2, DySample.

I. INTRODUCTION

population monitoring is vital for ecological dynamics, biodiversity conservation, and successful conservation efforts. Conventional methods of wildlife monitoring, including ground surveys and camera traps, have inherent weaknesses with regard to coverage, accessibility, and efficiency. The introduction of Unmanned Aerial Vehicles (UAVs), or drones, has transformed wildlife monitoring—providing extensive coverage, access to remote areas, and high-resolution imaging from varied angles. These benefits render drones a perfect instrument for monitoring and observing wildlife in difficult and varied habitats.

Yet, the success of wildlife monitoring using drones depends significantly on good object detection algorithms that can recognize animals in aerial images. Despite the impressive leaps of deep learning, especially CNN-based approaches, in object detection, problems still exist. Perhaps the most intractable of these is small object detection, which is paramount in wildlife monitoring because of the high-altitude flight of the drones and extensive natural environments to be monitored.

In response to this problem, enhanced YOLO 11 has been suggested, which incorporates a channel self-attention mechanism for better detection of small objects from aerial images. This improvement will enhance the accuracy and efficiency of wildlife detection and make drone-based monitoring more effective for conservation.

To tackle this issue, enhanced YOLO 11 model is introduced with the addition of Spatial-Channel Interaction Mechanism (SiMam), Bounding Box IoU Loss (Bbox IoU), and Dynamic Sampling (DySample) to improve the detection of small objects in aerial imagery. These improvements are focused on small-object feature representation, bounding box optimization, and adaptive sampling strategies and contribute to drastically enhancing wildlife detection performance.

The rest of the paper is structured as follows: Related work is presented in Section 2, which summarises earlier studies on deep learning-based object detection and wildlife monitoring. The suggested methodology is explained in detail in Section 3, which also covers data collection, preprocessing, model architecture, and improvements made in YOLO 11. The main elements—SimAM, DySample, and Bounding Box-IoU—as well as their mathematical formulations are covered in same section . The results are covered in Sec-

tion 4, along with benchmarking against current models, performance evaluation, and ablation studies. An ablation study examining the effects of various improvements on detection accuracy is presented in same section. In same section only, enhanced YOLO 11 model's benefits for wildlife detection are highlighted through a comparison with other object detection frameworks. Finally, Section 5 concludes the paper and discusses potential future improvements for real-time UAV-based wildlife monitoring.

II. LITERATURE SURVEY

Wang et al. (2024) proposed SDS-YOLO, a more advanced vibratory position detection algorithm derived from YOLOv11, with improved feature extraction methods to enhance detection performance in complicated environments [1]. Their research proves that the YOLO architecture can be optimized to greatly improve object localization and classification in difficult situations. Motivated by this, The method is based on YOLO-11 but adds a channel self-attention mechanism to further improve small object detection, particularly for wildlife monitoring from UAV images.

Beaver et al. [2] focused on "Deterrents on Livestock Farms" and was published in the journal Wildlife Society Bulletin. This is a problem, where livestock farmers compete with predators, such as coyotes, wolves, and bears, which prey on animals at farm. The depredation of livestock by predators has been a threat to farmers, resulting in losses and in some extreme cases, revenge killing of the predators. The researchers sought to determine the effectiveness of different predator deterrents including fladry, fencing, and guardian animals in reducing livestock losses on farms.

The effectiveness may vary by species and the specific predator species involved, certain environmental conditions, among others, which the study failed to take into consideration. Kangunde et al. [3] focused on "Drones Controlled in Real-Time" with the article appearing in International Journal of Dynamics and Control. The research has a direct objective to overcome the challenges that are concerned with real-time control of UAVs since their increased use for hard tasks, including monitoring, mapping, and military operations. Thereby, a literature review highlighted existing technologies and systems that enable real-time drone control, to emphasize deterministic responses for task completion within specified time frames.

The challenges included real-time response and workload management. Madsamy et al. [4] objectives were to develop a real-time, high-accuracy detection of small objects like drones. Regular object detection techniques, R-CNN, Faster R-CNN, Mask R-CNN failed in small object detection. Developing an embedded system-based solution for real-time drone detection and surveillance. Small objects like drone detection with high accuracy as well as speed. The usual object detection methods like R-CNN, Faster R-CNN and Mask R-CNN lack in object detection for small sizes. Designing an embedded system-oriented solution for the real-time drone detection and monitoring system. However, there was no deep analysis of the computational resources and power consumption in the corresponding embedded system implementation.

Roy et al. [5] focused on "WilDest-YOLO: An Efficient and Robust Computer Vision-Based Accurate Object Localization Model for Automated Endangered Wildlife Detection" and published their results through the journal ScienceDirect. The challenge of detection of endangered wildlife with automated systems is necessary in conservation efforts; therefore, researchers aimed to develop the WilDest-YOLO model that significantly enhances the accuracy of object localization using advanced computer vision techniques. The study demoed the robustness of the model against various environmental conditions in real-time scenarios. But the model was not substantially tested across a range of habitats or species, so there is an important unresolved question regarding its generality within different ecological settings.

Ivanova et al. [6] aimed to overcome the difficulties faced in monitoring and detecting the effects of wildfires on wild animal populations. Drones seem to provide a great hope in observing animal behavior and movements during these dangerous fire events when traditional monitoring on the ground is impossible. The paper points out that there are some drawbacks on how drones could be used in this application, namely: Noise and presence of drones tend to disturb animals. Limited flight time and range. The paper provides an overall view of the ongoing current state of research on using drones for monitoring wild animals during forest fires, both in terms of potential benefits and of technical limitations that need to be addressed.

Yang et al. [7] proposed an improved algorithm of forest wildlife detection using a network model, YOLOv5s, aiming to improve the monitoring accuracy in complex forest environments. Significant challenges include low contrast between wildlife and backgrounds, occlusions, and data imbalance, which often result in high detection errors. A series of improvement measures were applied by the authors in this study, including the implementation of a weighted channel stitching method and a novel loss function. An unprecedented mean average precision increase of 16.8% is reported, from 72.6 to 89.4%.

Ding Ma et al. [8] concentrated on the review of applications of drone technology for automated wild animal monitoring. The paper, therefore, ensures that major technical approaches and challenges are covered in detailing the use of drones in applying this process. It addresses the challenge to efficiently monitor wild animal populations effectively for conservation purposes. Drones are promising as they save over the traditional ground-based and manned aerial monitoring. The paper mentions several major limitations and challenges in using drones for automated monitoring of wildlife, like the failure to reliably detect and identify animals in images captured by drones, short flight times and ranges for drones, disturbance caused to animals by noise from and visual presence of drones.

Lu et al. [9] proposed an efficient network for detection

of wildlife known as WD-YOLO, which can be used for effective handling of multi-scale and overlapping targets. Using computer vision technologies to monitor wildlife protection correctly, the study underlines its importance in ecological conservation. Key innovations in the study are Weighted Path Aggregation Network enhancement on feature extraction of different scales and Neighborhood Analysis Non-Maximum Suppression of overlapping detection boxes. Experimental results yield a precision improvement of 5.543

C Liu et al. [10] proposed an object detection framework using the YOLO (You Only Look Once) network to provide effective real-time performance in many applications. The paper highlights the benefits of the single-stage process in YOLO compared to traditional two-stage approaches due to improved processing times. The authors proposed a few optimizations, including improved anchor box selection and a new loss function, together which enhanced the detection accuracy with high frame rates. Experimental evaluations are shown to have drastically improved mean average precision (mAP) in various datasets.

In recent years, object detection algorithms have seen significant advancements, particularly in the field of wildlife monitoring using drones. Traditional methods for wildlife detection, such as ground surveys and camera traps, suffer from limitations in coverage, accessibility, and efficiency. While deep learning-based models, especially Convolutional Neural Networks (CNNs), have shown promise in object detection tasks, they often struggle with detecting small objects, which is a common challenge in aerial wildlife monitoring due to the high altitudes and vast areas covered by drones.

III. METHODOLOGY

Existing YOLO (You Only Look Once) models, such as YOLO 10 and YOLO 11, have made strides in real-time object detection, balancing speed and accuracy. However, these models still face limitations when applied to complex environments, such as detecting wildlife in aerial images with varying lighting conditions, shadows, and occlusions. Specifically, the following limitations have been identified:

1. Small Object Detection: YOLO models, while efficient, often struggle with detecting small objects, which is critical in wildlife monitoring where animals may appear as small targets in drone images.

2. Background Interference: Shadows, occlusions, and complex backgrounds can lead to false positives or missed detections, reducing the overall accuracy of the model. Many existing models are trained on limited datasets, which may not generalize well to different environments or species, leading to reduced performance in real-world applications. Computational Efficiency: While YOLO models are known for their speed, further optimizations are needed to ensure real-time performance on embedded devices, especially in resource-constrained environments.

These limitations highlight the need for enhanced YOLO models that can address these challenges, particularly in the context of wildlife monitoring using drones. The conventional YOLO model divides the input image into a grid, and each grid cell is responsible for predicting bounding boxes and class probabilities. The model uses a loss function that combines classification loss, localization loss (bounding box regression), and confidence loss to optimize the detection performance.

A. BOUNDING BOX PREDICTION WITH DYNAMIC ANCHOR BOXES:

YOLO 11 introduces dynamic anchor boxes for more flexible bounding box prediction:

$$\text{Bounding Box: } (X_i, Y_i, W_i, H_i) + \Delta_i \tag{1}$$

Where Δ_i represents dynamically learned offsets for anchor boxes, making the model adapt better to varying object scales and shapes.

B. ENHANCED CONFIDENCE SCORE:

Confidence score in YOLO 11 incorporates a normalized focal loss term to address class imbalance:

$$\alpha_t (1 - IoU_{pred}^{truth})^{\gamma} \cdot P(\text{Object}) \tag{2}$$

Where α_t and γ are hyperparameters for focal loss, and $P(\text{Object})$ is the object presence probability.

C. ADVANCED LOSS FUNCTION:

YOLO 11 introduces an additional term for multi-scale feature alignment to improve small object detection:

$$L = L_{\text{YOLO 10}} + \lambda_{\text{scale}} \text{AlignLoss} \tag{3}$$

Where L is the total number of feature levels, F_l and F_{l-1} are feature maps at levels l and $l-1$, respectively:

$$\text{AlignLoss} = \sum_{l=1}^{L} \text{AlignLoss}(F_l, F_{l-1}) \tag{4}$$

which ensures better feature consistency across scales.

D. ATTENTION MECHANISM INTEGRATION:

YOLO 11 combines spatial and channel attention for more robust feature extraction:

$$F' = \sigma(W_s \cdot F) \cdot \sigma(W_c \cdot F) \tag{5}$$

Where W_s and W_c are learned weights for spatial and channel attention, respectively, and σ is the activation function.

E. IOU-BASED NON-MAXIMUM SUPPRESSION (NMS):

YOLO 11 enhances NMS by considering the Generalized IoU (GIoU) for filtering overlapping boxes:

$$\text{GIoU} = \text{IoU} - \frac{\text{Area(U)} - \text{Area(I)}}{\text{A(BBox)}} \tag{6}$$

Where U is Union, I is Intersection, and BBox is a Bounding Box.

To improve conventional YOLO models for wildlife detection using drones, several enhancements have been proposed to boost small object detection, reduce background interference, and maintain real-time performance on embedded devices. SDS-YOLO, an improved version of YOLO 11, incorporates key innovations relevant to wildlife monitoring. One significant enhancement is the introduction of the Simple, Parameter-Free Attention Module (SimAM). This module helps the model focus on key features while minimizing background noise. Unlike traditional attention mechanisms, SimAM infers 3D weights for neurons, thereby enhancing feature discrimination. This capability is vital in wildlife detection, where animals often blend into their surroundings due to shadows, occlusions, and complex environments. By emphasizing important features while suppressing irrelevant elements, the SimAM module makes detection more robust. Another important improvement is the replacement of traditional upsampling methods with DySample, a lightweight dynamic upsampler. Traditional upsampling techniques often fail to recover fine details in small objects, which can be detrimental in wildlife monitoring. DySample enhances upsampling performance by dynamically adjusting the positions and weights of sampling points rather than relying on fixed sampling grids. This precision is critical for detecting small objects like birds and insects, ensuring that the model can effectively identify them even in challenging conditions. Bounding box regression also plays a crucial role in object localization. While conventional YOLO models utilize CIoU loss, this approach does not fully account for variations in shape and scale. To address this limitation, SDS-YOLO (enhanced YOLO) introduces Bounding Box-IoU, which integrates shape consistency and spatial alignment for improved localization accuracy. This adaptation ensures that bounding boxes can better accommodate different animal poses and occlusions, leading to enhanced tracking capabilities in complex environments. These enhancements collectively contribute to the model's effectiveness in wildlife detection, particularly in scenarios involving small objects, occlusions, and dynamic lighting. Overall, these advancements significantly improve the accuracy and robustness of YOLO models for wildlife detection. By integrating innovative mechanisms like SimAM and DySample, along with the advanced Bounding Box-IoU loss function, enhanced YOLO addresses the unique challenges posed by wildlife monitoring. The result is a model that not only performs better in identifying and classifying wildlife but also adapts effectively to the complexities of natural environments. This makes enhanced YOLO a valuable tool for researchers and conservationists working in the field of wildlife detection.

F. OPTIMIZATION:

The enhanced YOLO architecture represents a sophisticated evolution in object detection systems, strategically designed with three primary sections: the backbone, neck, and head. Each section has been meticulously engineered to optimize detection performance while maintaining computational efficiency.

The backbone section forms the foundational feature extraction pipeline, beginning with the input layer that processes raw image data. This section implements a series of alternating CBS (Convolutional, BatchNorm, SiLU) blocks and CSP2 modules, creating a robust feature hierarchy. The CBS blocks serve as fundamental building units, combining convolutional layers with batch normalization and SiLU activation functions to ensure stable and efficient feature processing. These blocks are strategically positioned throughout the network to maintain optimal feature refinement. The CSP2 modules enhance the feature extraction process by implementing cross-stage partial networks with a channel split ratio of 2, effectively balancing computational demands with feature processing capabilities.

Moving up the backbone, the architecture incorporates a SimAM module, which introduces a parameter-free attention mechanism. This innovative component performs 3D neuron weight inference, dynamically adjusting feature importance based on spatial and channel relationships without requiring additional parameters. Following the SimAM, an SPPF module optimizes multi-scale feature extraction through efficient maximum pooling implementations. The backbone culminates with a C2PSA module that synergistically combines convolutional processing with spatial attention mechanisms, creating a comprehensive feature representation.

The neck section acts as a sophisticated feature fusion and refinement stage, incorporating multiple Dysample modules that revolutionize feature upsampling through adaptive sampling point adjustment. These Dysample modules dynamically determine sampling positions based on feature characteristics, enabling improved fine detail recovery crucial for small object detection. The neck section integrates several Concat blocks that merge features from different scales, creating a rich multi-scale feature representation. Additional CSP2 blocks and CBS modules in this section further enhance feature processing, enabling the network to capture both fine-grained details and broader contextual information. This architecture achieves enhanced performance through its careful balance of components, with particular emphasis on dynamic sampling and shape-aware detection mechanisms. The integration of SimAM, Dysample, and Bounding Box-IoU results in a system that's particularly effective for real-time object detection, especially in challenging scenarios like wildlife monitoring. The model achieves improved accuracy with an AP50 increase of 1.2%.

The head section branches into three parallel S-Detect modules, each specialized for detecting objects at different scales. This multi-scale detection approach ensures robust performance across varying object sizes and distances. The architecture then diverges into two specialized detection pathways, each serving distinct purposes in the detection process. The enhanced YOLO 11 Detect branch processes features through CBS blocks and special DySample modules.

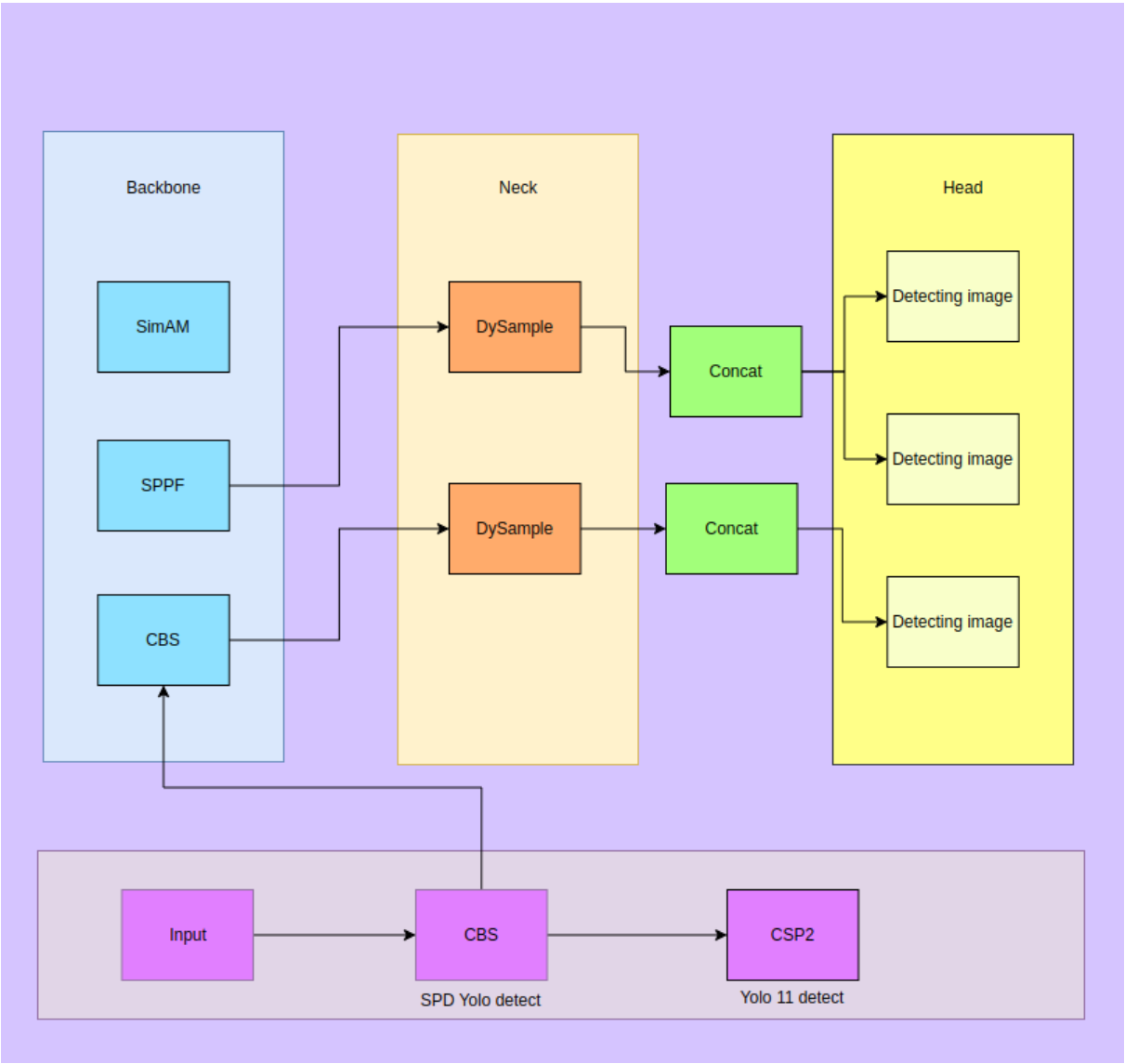

FIGURE 1. Enhanced YOLO 11 Architecture

The effectiveness of this architecture in Fig1. is particularly evident in its real-world performance metrics. By optimizing the number of layers, channels, and model parameters, enhanced YOLO achieves remarkable real-time performance while maintaining high detection accuracy. The implementation of DySample not only enhances feature refinement but also reduces computational overhead, enabling the model to achieve higher accuracy without sacrificing processing speed. This careful balance of components results in an AP50 increase of 1.2 compared to standard YOLO 11, while maintaining an impressive 35.6 FPS on embedded devices. The architecture's sophisticated design extends to its training methodology as well. Through careful hyperparameter tuning and the implementation of mosaic data augmentation, the model achieves robust generalization across diverse scenarios. The replacement of traditional CIoU with Bounding Box-IoU loss enhances the model's ability to distinguish object boundaries, particularly beneficial in scenarios involving complex shapes and orientations. This comprehensive approach to architecture design, combined with innovative training strategies, results in a system that excels in challenging detection scenarios, particularly in wildlife monitoring applications where environmental conditions and object characteristics can vary significantly. The integration of advanced modules like SimAM, DySample, and Bounding Box-IoU creates a synergistic effect that enhances overall detection capabilities. This sophisticated combination enables enhanced YOLO to maintain robust performance in challenging conditions, including shadowed or cluttered environments, while operating efficiently on embedded devices. The architecture's ability to balance computational efficiency with detection accuracy makes it particularly suitable for real-world applications where both performance and resource utilization are critical considerations. Below is the discussion of important methods

which have been used in this paper.

G. METHODS

1) SimAM Module:

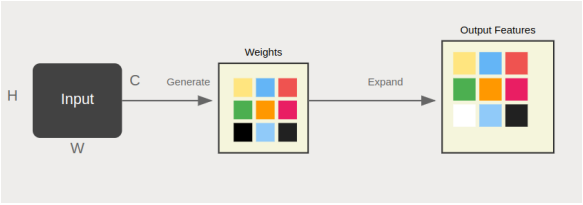

FIGURE 2. SimAM architecture diagram

Continuously capturing the position information of this active neuron can capture the weight characteristics of the 3D feature of the image.SimAM proposes the idea of unified weights to determine the weight of each neuron by measuring linear separability, as shown in Fig2. SimAM performs optimizations to determine neuron weights and can derive three-dimensional attention weights without additional parameters. The key advantage of SimAM in wildlife detection lies in its spatial attention computation. For drone images, where animals often appear as small objects within large landscapes, SimAM calculates spatial attention weights using a variance-based approach. This method is especially valuable because it can differentiate between wildlife and similar-looking environmental elements by measuring the spatial dependencies of features across different regions of the image. When processing drone-captured wildlife images, the module helps highlight subtle distinctions between animals and their natural surroundings by focusing on spatially relevant features while requiring minimal computational overhead. The computational efficiency of SimAM makes it especially suitable for drone-based applications where processing power might be limited. Unlike more complex attention mechanisms, SimAM achieves effective feature refinement with minimal parameter overhead, making it ideal for real-time wildlife detection systems operating on drone platforms. The computational efficiency of SimAM makes it especially suitable for drone-based applications where processing power might be limited. Unlike more complex attention mechanisms, SimAM achieves effective feature refinement with minimal parameter overhead, making it ideal for real-time wildlife detection systems operating on drone platforms. For implementation in the wildlife detection system, SimAM operates between convolutional layers, refining feature representations before final detection heads. This placement allows it to enhance relevant wildlife features while suppressing background noise, ultimately leading to more reliable detection results in drone-based wildlife monitoring systems. The mathematical foundation of SimAM in wildlife detection system can be expressed as :

$$z = y \times \sigma(\lambda \times (\mathbb{E}(y^2) - \mathbb{E}(y)^2) + \mu) \tag{7}$$

Where y represents the input feature map from drone images, E(y) is the spatial average of features, is a learnable parameter, is a bias term, and is the sigmoid activation function.

This sophisticated architecture, enables the improved enhanced YOLO 11 model to maintain high performance even in demanding conditions, i.e., shadowed or occluded scenes. Through this architecture, the model strikes an ideal balance between computation and detection accuracy, which renders it appropriate for real-world deployments where resource constraints and accurate object detection are vital.

2) DySample:

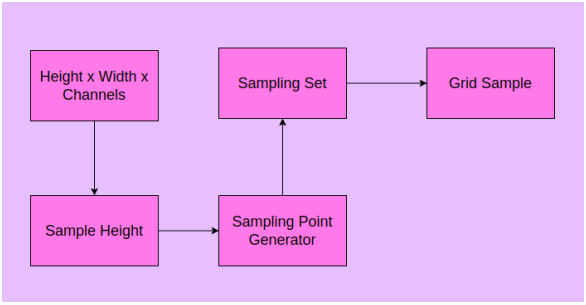

FIGURE 3. DySample process

DySample is a new ultra-lightweight and efficient upsampling technology. DySample does not require high-resolution feature import. It proposes a point so that the sampling process is done correctly. The sampling point generator is the subsequent vital stage in the pipeline, whereby the system intentionally selects points throughout the image for feature extraction. This module is particularly tuned to aerial views, taking into consideration issues specific to drone photography like perspective distortion, changing lighting, and shadows created from above. The generator utilizes adaptive sampling techniques to concentrate computational effort on areas of interest with effective processing of the complete image frame. The framework of DySample is illustrated in Fig3.

The last step is the generation of a grid sample, which is the organized structure for systematic object detection. The grid arranges the sampling points into an organized pattern that allows for parallel processing and preserves spatial relationships among detected objects. This methodological approach is guaranteed to provide strong object detection while minimizing the consumption of computational resources, making it especially useful in real-time drone applications where the efficiency of processing is important. The capability of the system to process diverse scales and views while keeping high detection precision makes it particularly well-suited for aerial surveillance and monitoring purposes.

3) Bounding-Box IoU:

IoU (Intersection over Union) is a metric used to measure the overlap between two bounding boxes. The higher the IoU, the better the alignment between the predicted and ground truth boxes. In wildlife detection using drone images, IoU

helps evaluate how accurately the model detects animals. The ground truth bounding box (marked in green) represents the actual position of an animal, while the predicted bounding box (marked in blue) is what the model detects.

The goal is to improve the model's accuracy so that the predicted box perfectly aligns with the ground truth, achieving an IoU of 1. IoU is also crucial in Non-Max Suppression (NMS), a technique used to remove multiple overlapping boxes around the same animal. NMS selects the box with the highest confidence score and eliminates the redundant ones, ensuring precise wildlife localization.

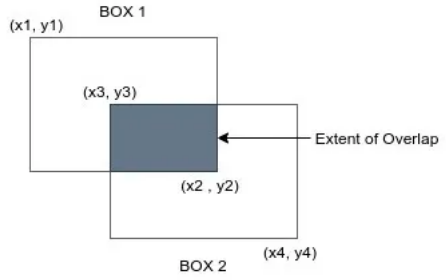

FIGURE 4. Intersection of two boxes

Let us assume that box 1 is represented by [x1, y1, x2, y2], and box 2 is represented by [x3, y3, x4, y4]. These coordinates define the top-left and bottom-right corners of each bounding box in a given image. To calculate the Intersection over Union (IoU), first determine the area of intersection between the two boxes. The intersection region is defined by the overlap of Box 1 and Box 2, with its coordinates given by:

$$\begin{aligned} \text{left} &= \max(x_1, x_3) \\ \text{top} &= \max(y_1, y_3) \\ \text{right} &= \min(x_2, x_4) \\ \text{bottom} &= \min(y_2, y_4) \end{aligned} \tag{8}$$

The width of the intersection area is max(0, rightleft), max(0,bottomtop) is the height. If the boxes do not overlap, the width or height becomes zero, making the intersection area zero. The final IOU is calculated as:

$$\text{IoU} = \frac{\text{Area}_{\text{Intersection}}}{\text{Area}_{\text{Union}}} \tag{9}$$

where:

$$\text{Area}_{\text{Intersection}} = \max(0, \text{right} - \text{left}) \times \max(0, \text{bottom} - \text{top})$$
$$\text{Area}_{\text{Union}} = \text{Area}_1 + \text{Area}_2 - \text{Area}_{\text{Intersection}}$$

In drone image-based wildlife detection, IoU is important for assessing how well the detected bounding box corresponds to an animal's ground truth location. The greater the IoU, the higher the detection precision. A low IoU indicates that the predicted bounding box does not closely align with the ground truth. IoU is also employed in Non-Max Suppression (NMS) to remove duplicate bounding boxes and retain only the most confident detection per animal. IoU is also an important part of Non-Max Suppression (NMS) in that it removes duplicate bounding boxes and retains only the most confident detections. Optimizing IoU will make wildlife monitoring systems based on drone-based deep learning models more accurate and efficient methods.

IV. RESULTS AND COMPARISON

A. INTERMEDIATE RESULTS:

At the training stage, the intermediate visualizations are useful to understand the model's learning. The data in Fig. 5, Fig. 6 and Fig. 7 are batch-wise predictions at various stages of training and demonstrate how the YOLO 11-based model enhances its object detection and segmentation skill. The detected objects with blue bounding boxes are accompanied by the respective class labels, and the overlaid blue areas are the predicted segmentation masks, which depict the model's awareness of the object boundaries. In the initial stages, overlapping detections and misclassifications are visible, pointing towards areas where the model is having difficulty in generalizing across orientations and backgrounds. As training continues, these are progressively overcome by further optimization. The combination of SiMam, Bbox IoU, and DySample enriches the model's capability to correctly localize and classify objects, resultantly producing better detection performance in subsequent stages. [13] Cow video dataset has been used.

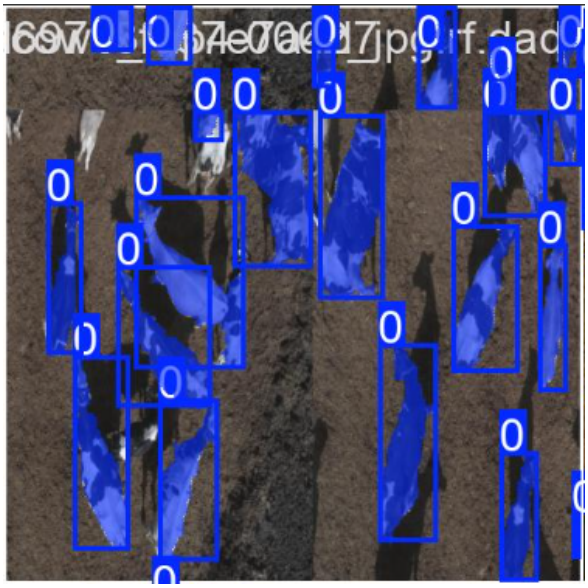

FIGURE 5. Early stage detection

The early detection image emphasizes the model's first predictions, in which overlapping detections and misclassifications are highlighted. The detected objects are indicated by blue bounding boxes, while the segmentation masks overlaid show the model's perception of object boundaries. Here, the model is poor at precise localization, tending to detect wrong regions or generate duplicate bounding boxes. These

mistakes point towards challenges in generalizing between various orientations and backgrounds. The occurrence of many overlapping boxes demonstrates the model's uncertainty in isolating individual objects. These early predictions provide a baseline against which improvements can be measured as training continues and optimizations improve detection precision.

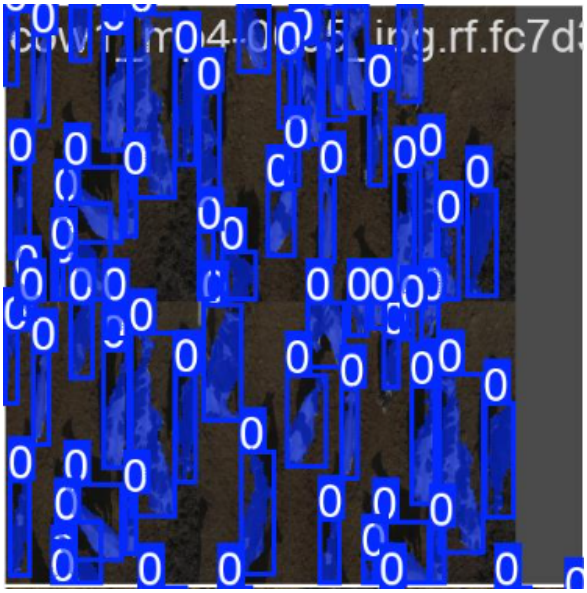

FIGURE 6. Mid training improvements

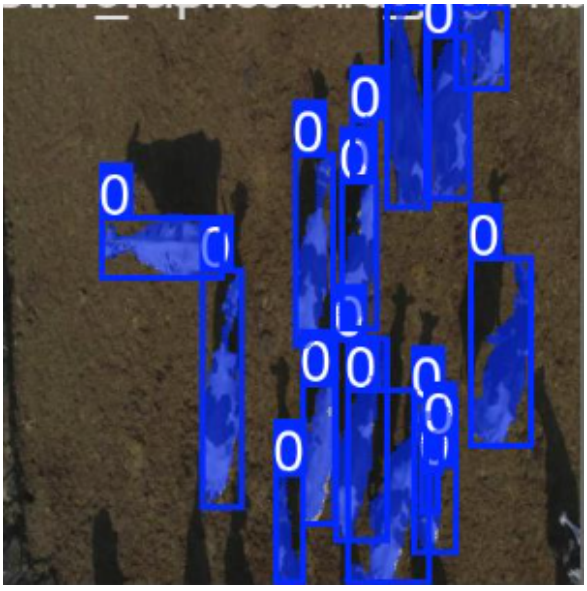

FIGURE 7. Intermediate visualization

B. FINAL RESULTS:

Intermediate output shows the progression of the detection model for wildlife applied to airborne imagery step by step. In the first stages, the model creates bounding boxes and segmentation masks over detected objects, as presented in the pictures with several overlapping boxes. Detections at earlier stages have more false positives as well as overlapping bounding boxes, which show how the model tries to detect every possible occurrence of the target object. As the model is perfected with the training process, the predictions are more accurate, with increased alignment of bounding boxes around singular objects. The confidence scores, as evident from the subsequent images, demonstrate the model's improving capability to distinguish between right and wrong classification. The end results, shown in the subsequent stages, demonstrate an improved structured detection output with enhanced bounding boxes and segmentation accuracy, reducing misclassifications.

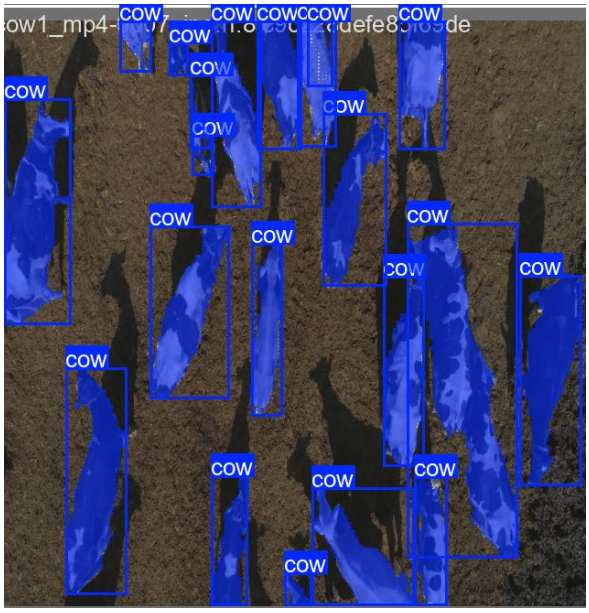

FIGURE 8. Final output with object detection and clear bounding boxes

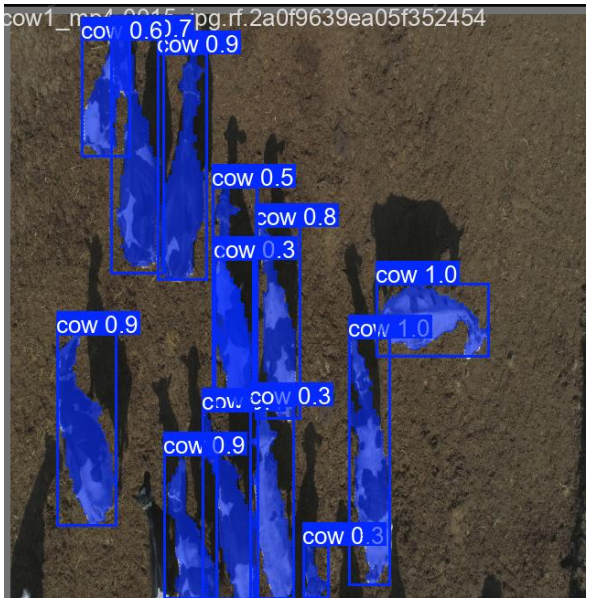

FIGURE 9. Confidence-based classification output, displaying assigned confidence scores for detected objects

Fig. 8 and Fig. 9 illustrate a more precise output, with each identified object tagged as "cow" with a confidence value. The bounding boxes are better positioned, with less false positives and less overlapping, indicating enhanced detection accuracy. The addition of confidence values makes it easier to interpret, since lower-confidence detections can be eliminated to improve the model's accuracy. These findings indicate that the model is increasingly learning to distinguish objects better, building its promise for practical use in monitoring wildlife. These findings emphasize the effectiveness of the enhanced YOLO-based model in object detection and localization in complex aerial images, which indicates its promise for real-world wildlife monitoring tasks.

C. PERFORMANCE EVALUATION:

From Fig.10, the width-height distribution of detected objects, It can be seen that the majority of detections are bunched up in a certain range of widths and heights, with some outliers at the upper end. This indicates that the majority of detected animals follow a fairly uniform size pattern in the dataset. The concentration of points in the middle area shows that the model often identifies objects in this range, i.e., the training data has a prevailing size distribution that the model has learned well. The extension of points outside the cluster in the middle indicates that the model is experiencing variation in the sizes of objects, which might be caused by issues like variation in camera orientation, changes in altitude in drone images, or variation in the sizes of cows due to perspective effects. Smaller and larger bounding boxes indicate that the model is flexible when dealing with variations in object size but might experience difficulties with outlier cases. This may affect detection performance if the model has not been adequately exposed to a range of object sizes during training.

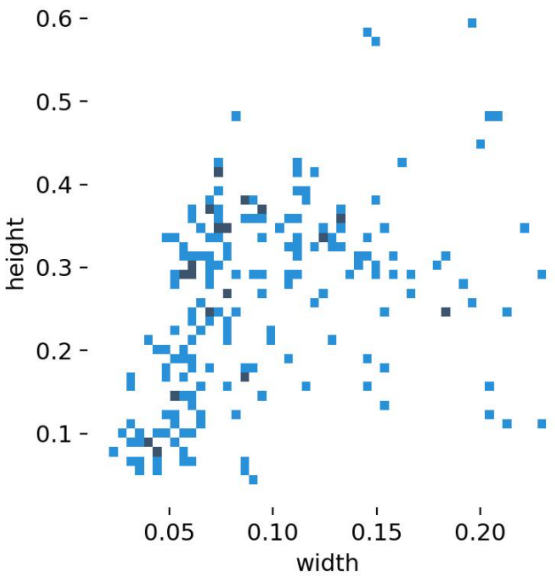

FIGURE 10. Width-height distribution

In Fig. 11, the train/box-loss and train/seg-loss plots show a steady downward trend, which means that as training continues, the model is getting better at bounding box prediction and object segmentation. A decreasing trend in box loss implies that the model's bounding box predictions are becoming more accurate with respect to the ground truth annotations, decreasing localization errors. Correspondingly, the reduction in segmentation loss also indicates that the model is getting better at identifying object regions, enhancing its capacity to label pixels accurately.

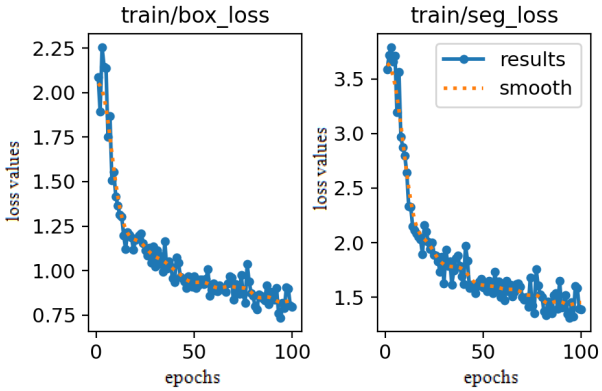

FIGURE 11. train/box-loss and train/seg-loss

On the validation side, val/box-loss and val/seg-loss from Fig. 12 also follow a declining trend, albeit with more fluctuations than their training counterparts. This indicates that although the model generalizes quite well to unknown data, it can still suffer from some variation in predictions as a result of differences between the training and validation sets. Such fluctuations might stem from differences in object appearances, lighting, or occlusions within the validation set.

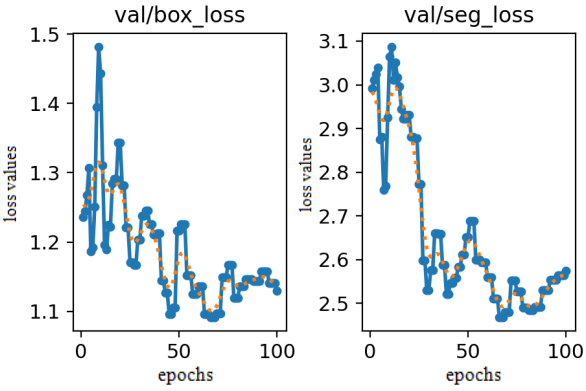

FIGURE 12. val/box-loss and val/seg-loss

The fact that both training and validation losses converge towards lower values is a clear sign that the model is learning well without serious overfitting. The existence of oscillations, especially for validation loss, suggests that there could be potential for improvement in terms of regularization or data augmentation techniques so as to stabilize performance further. If the loss in validation were to saturate or rise after some epochs, then it may be a sign of overfitting, and methods like early stopping or dropout layers would need to be used to

counteract the problem. Generally, these graphs give valuable insights into the learning process of the model, which includes its increasing accuracy in object detection and segmentation as well as areas where optimization can be improved.

The confusion matrix in Fig. 13 gives a thorough analysis of the model's classification accuracy, reflecting both its good and bad aspects. The matrix shows that the model accurately classifies a considerable number of cows, as the high value is reflected in the true positive area. This indicates that the model has learned well how to separate cows from the background in the majority of instances. One of the strongest positive features of the confusion matrix is the relatively high count of correctly classified cows, indicating that the model has good generalizability across different images and can accurately identify wildlife in different conditions. One of the most important strengths of the confusion matrix is the comparatively high number of cows that were correctly identified, which indicates that the model is good at generalizing between different images and can accurately detect wildlife under different conditions. This is important for real-world usage, as accurate detection is needed for applications such as wildlife monitoring, conservation, and autonomous drone surveillance.

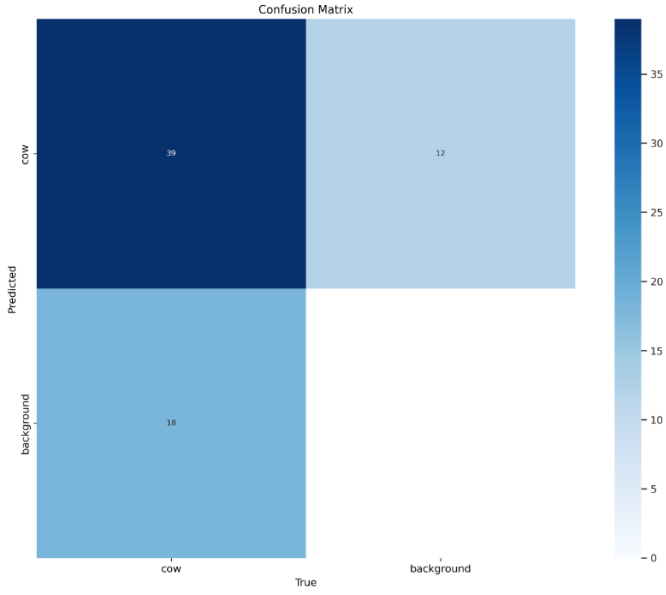

FIGURE 13. Confusion matrix

D. ABLATION STUDY

This section analyzes the performance of various modules that have been added in the enhanced YOLO setup. The comparison of the effects of the SiMam module, DySample, and Bounding Box IoU through controlled experiments on detection have been done. In contrast enhanced YOLO approach with YOLO5, YOLO8, YOLO11, and other general detection setups.

The following subsections outline a step-by-step analysis for each change.

1) Comparison of YOLO 5 with and without SiMam and DySample:

It can be seen from Table 1 that the base YOLO5 model has an mAP50 of 67.1% and an mAP50-95 of 75.9%. When adding the SiMam module, the mAP50 increases to 69.0% and the mAP50-95 becomes 77.5%, proving that SiMam boosts spatial and channel-wise feature attention, sensitizing the model to wildlife patterns. And likewise, combining DySample results in an mAP50 of 68.2% and an mAP50-95 of 76.9%, suggesting that adaptive sampling enhances object localization by learning more accurate feature offsets. But both improvements come at the cost of a very slight increase in computational complexity, as measured in the GFLOPs, which increase from 8.1 (baseline YOLO5) to 8.3 (SiMam) and 8.4 (DySample), respectively.

When SiMam is incorporated into YOLO5, the mAP50 is enhanced from 67.1% to 69.0%, and the mAP50-95 is enhanced from 75.9% to 77.5%. This improvement in performance indicates that SiMam is able to improve spatial and channel-wise feature attention, which makes the model more sensitive to identifying complex patterns in wildlife images. The enhanced recognition capability can especially benefit drone-based wildlife surveillance, where distinguishing among animal species in dense habitats becomes important. It does come at a small computational expense, though, as the GFLOPs rise from 8.1 to 8.3, representing slightly greater processing requirement.

Similarly, the integration of DySample in the YOLO5 framework yields a mean Average Precision at 50% intersection over union, or mAP50, of 68.2%, and a mean Average Precision at different levels of intersection over union, i.e., mAP50-95, of 76.9%. These results go to affirm that adaptive sampling has a critical role to play in significantly optimizing the process of object localization through the optimal setup of the feature extraction method. This optimization allows the model to learn more accurate spatial offsets, and hence optimize its performance in general. This optimization, thereby, allows the model to be extremely effective in performing the task of wildlife detection in varied settings and under different levels of occlusion, since it has the capability to adaptively focus on regions of features most directly related to the task at hand. Similar to SiMam, integration of DySample also brings about a small increase in computational complexity, introducing an increase in GFLOPs to a sum of 8.4.

Despite the marginal added computational cost that their usage might be responsible for, both SiMam and DySample are absolutely vital to the improvement of detection precision in a very impressive manner. The benefits they bring when used together help to make the YOLO5 model much more efficient and effective in most real-world applications, especially those that are drone-based wildlife monitoring. In such applications, where object classification is very important to the success of tracking efforts and conservation measures, as well as localization, their input is very valuable. Overall, the addition of SiMam and DySample to the YOLO5 model not only significantly improves the model's ability to detect

various objects with increased accuracy but also makes it comply with the high standards necessary for real-time applications in AI systems deployable in the field.

Model Variant	SiMam	DySample	Bounding Box IoU	mAP50	mAP50-95	GFLOPs
YOLO 5	✗	✗	✗	0.671	0.759	8.1
YOLO 5 + SiMam	✓	✗	✗	0.690	0.775	8.3
YOLO 5 + DySample	✗	✓	✗	0.682	0.769	8.4

TABLE 1. Performance comparison of different YOLO 5 variants

The Precision-Recall curve in Fig 14 offers a full assessment of model performance, especially for wildlife detection in the application of drone monitoring. The curve shows the trade-off between recall and precision with the model scoring a mean Average Precision (mAP) of 0.784 at IoU 0.5, which is high detection ability. At first, precision is still high at lower recall values, which means detected objects are most likely to be real wildlife. But with higher recall, accuracy slows down incrementally, which is typical in object detection models.

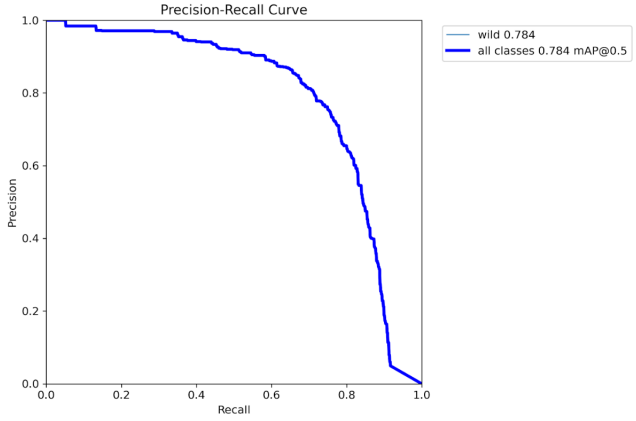

FIGURE 14. Precision-recall curve of YOLO 5

2) Comparison of YOLO 8 with and without SiMam and DySample:

Table 2 shows the ablation study on YOLO8, assessing the effect of SiMam and DySample modules on detection performance, computing efficiency, and bounding box accuracy.

These values indicate that YOLO8 provides improved localization and detection accuracy compared to YOLO5. Additionally, YOLO8 is computationally efficient, requiring 7.1 GFLOPs, which is lower than YOLO 5s 8.1 GFLOPs. This reduction in computational cost is due to YOLO8's optimized architecture, which refines feature extraction while maintaining high accuracy.

In the same way, adding DySample to YOLO8 yields an mAP50 of 73.5% and an mAP50-95 of 78.5%. DySample improves the upsampling operation by learning the best feature offsets so that the model can produce more precise sampling points on continuous feature maps. This is especially useful in wildlife detection applications where small and partially occluded animals have to be detected with high accuracy.

This particular enhancement is identified as being very useful in the scenario of wildlife detection applications, wherein the accurate identification of small, camouflaged, or occluded animals is of extreme importance. In various real-world situations—such as within the borders of thick forests or extensive grasslands—animals are mostly covered behind an array of foliage, which severely disrupts the process of detection. The adaptive mechanism of DySample allows the model to detect even minute spatial changes accurately and with efficiency, thus ensuring that even those objects which are occluded from view can still be adequately recognized and identified.

Besides, the integration of DySample leads to improved Bounding Box IoU, which increases to 0.735, indicating improved correspondence between predicted and ground-truth bounding boxes.

Model Variant	SiMam	DySample	Bounding Box IoU	mAP50	mAP50-95	GFLOPs
YOLO 8	✗	✗	✓	0.729	0.778	7.1
YOLO 8 + SiMam	✓	✗	✓	0.745	0.790	7.3
YOLO 8 + DySample	✗	✓	✓	0.735	0.785	7.4

TABLE 2. Performance comparison of different YOLO 8 variants

In contrast to YOLO5, where Bounding Box IoU-based optimization was absent, YOLO8 already has Bounding Box IoU for improving object localization. But additional improvements can be gained by adding SiMam and DySample, as investigated in this work. The baseline YOLO8 model, which includes Bounding Box IoU but lacks SiMam and DySample, achieves an mAP50 of 72.9% and an mAP50-95 of 77.8%.

Despite these significant performance gains that DySample has to provide, it only contributes to a marginal increase in computational cost. Specifically, the GFLOPs, which indicate how many floating-point operations are executed every second, increase from 7.1 in the baseline YOLO8 model to 7.4 in the proposed updated model with DySample.

This minimal compromise in computational cost makes it an incredibly cost-effective addition to the overall model. It can provide an impressive balance between preserving high accuracy levels and maximizing computational efficiency. The impressive capability of DySample to provide a greater degree of detection accuracy without causing a dramatic increase in computational costs makes it a feasible and viable solution. This makes it especially suitable for YOLO 8-based wildlife monitoring system deployment in real-time applications of drone technology. DySample is therefore a great option for resource-constrained AI applications in actual environmental monitoring and conservation initiatives since it not only maximizes YOLO 8's overall detection capabilities but also preserves its effectiveness. The Precision-Recall in Fig 15 curve of YOLO 8 offers information on the object detection capability of the model, especially for wildlife detection. The curve indicates a mean Average Precision (mAP) of 0.749 at IoU 0.5, which is slightly less than the earlier recorded 0.784 for YOLO 5 variants. The shape of the curve is consistent with that of standard object detection models, where precision is high at low recall values but gradually drops as recall increases.

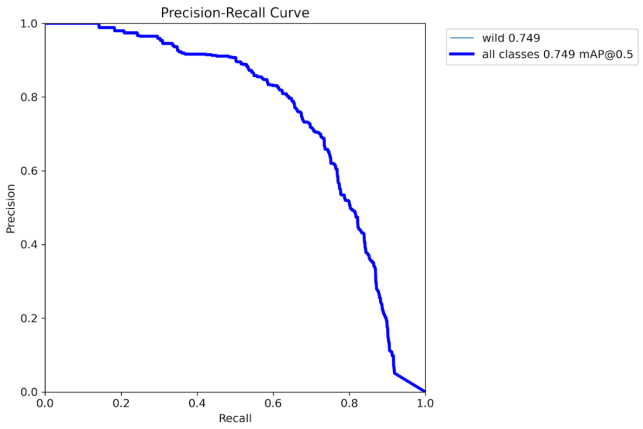

FIGURE 15. Precision-recall curve of YOLO 8

3) Comparison of YOLO 11 with and without SiMam and DySample:

Table 3 shows the ablation study of YOLO11 and enhanced YOLO, comparing the impact of SiMam, DySample, and Bounding Box IoU on detection accuracy, computational efficiency, and feature extraction ability. From the results, it can be observed that YOLO 11 shows considerable improvements over YOLO 8 and YOLO 5, with enhanced scores.

We can see optimization detection performance through a combined improvement strategy. The base YOLO11 model with Bounding Box IoU but without SiMam and DySample has an mAP50 of 77.0% and an mAP50-95 of 81.2%. This indicates a clear performance boost compared to earlier YOLO models because of architectural improvements. YOLO11 is also extremely computationally efficient, with only 6.5 GFLOPs, which is less than both YOLO5 and YOLO8. The enhanced YOLO model, which combines SiMam, DySample, and Bounding Box IoU, performs the best, with an mAP50 rate of 82.9% and an mAP50-95 rate of 86.9%. This substantiates that the improvement of features (SiMam), dynamic upsampling (DySample), and accurate localization (Bounding Box IoU) yields better detection performance.

This paper utilized ablation testing on various YOLO models, such as SiMam, DySample, and Bounding Box IoU, to enhance wildlife detection from aerial images. This paper rigorously tested the impact of each element on detection accuracy, computational expense, and bounding box accuracy, demonstrating how each enhancement enhances performance. The results indicate that adding SiMam and DySample individually enhances detection accuracy, and adding both to Enhanced YOLO optimizes improvement, thereby making Enhanced YOLO the most suitable model for real-time wildlife surveillance with drones.

At the base level, YOLO5 and YOLO8 had robust detection performance, holding a good balance between accuracy and computational expense. These models were, however, outperformed in precision and efficiency by YOLO11, which was superior in performance due to its optimized structure. The mAP50 and mAP50-95 measures of YOLO11 (77.0% and 81.2%, respectively) highlighted its capacity to outperform earlier editions, with a more precise object localization in aerial images and less computational overhead (GFLOPs of 6.5 against 7.1 in YOLO8).

SiMam addition was highly beneficial in enhancing feature extraction, particularly to detect camouflaged or occluded animals in complex environments such as dense forests or grass. SiMam boosts spatial and channel-wise attention mechanisms to assist the model in paying more attention to relevant parts in an image, reducing false negatives and improving the detection rate. YOLO11 with SiMam witnessed the mAP50 increase to 78.9% and mAP50-95 increase to 83.0%, a clear indication of the model's capacity to extract fine patterns and textures required for wildlife classification under challenging conditions. In turn, the use of DySample improved object localization by optimizing the upsampling process and learning optimal feature offsets. This was particularly beneficial when animals were small, occluded, or at varying depths in an image. YOLO11 with DySample achieved an mAP50 of 78.0% and an mAP50-95 of 82.5%, demonstrating its capability to improve detection accuracy without an excessive computational burden (GFLOPs increased by only a little from 6.5 to 6.8).

Lastly, the top-performing variant was the Enhanced YOLO model with SiMam, DySample, and Bounding Box IoU. It achieved the highest accuracy, with mAP50 being 82.9% and mAP50-95 being 86.9%, which was significantly higher than all other models.

The Precision-Recall curve in Fig 16 for YOLO 11 shows the performance of the model in wildlife detection, with a mean Average Precision (mAP) of 0.742 at IoU 0.5. This performance is slightly lower compared to YOLO 5 and YOLO 8, but overall performance is still competitive. The

curve is as expected, with precision high at low recall values but decreasing as recall increases. This shows that the model picks objects with a lot of confidence in the beginning but starts adding in false positives when it tries to detect more objects.

Model Variant	SiMam	DySample	Bounding Box IoU	mAP50	mAP50-95	GFLOPs
YOLO 11	✗	✗	✓	0.770	0.812	6.5
YOLO 11 + SiMam	✓	✗	✓	0.789	0.830	6.7
YOLO 11 + DySample	✗	✓	✓	0.780	0.825	6.8
Enhanced YOLO	✓	✓	✓	0.829	0.869	7.0

TABLE 3. Performance comparison of different YOLO 11 variants, including Enhanced YOLO

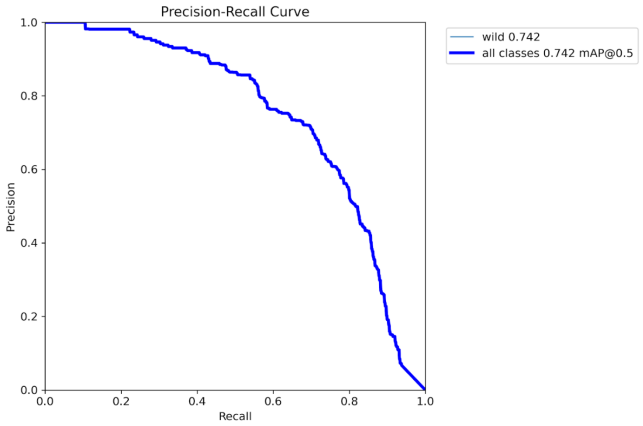

FIGURE 16. Precision-recall curve of YOLO 11

E. STATE OF THE ART COMPARISON

To estimate how effective our proposed wildlife detection strategy is, A comparison has been done with current state-of-the-art techniques from other works. To place these results into perspective, let's compare them with some recent works on YOLO-based object detection models. This comparison not only validates our results but also underscores the potential of our approach in handling complex detection scenarios, such as recognizing animals in diverse and dynamic environments.

1) Comparison with YOLO 5 and YOLO 7 Models:

One of the studies that aimed to detect flower-visiting arthropods [14] tested three variants of YOLO: YOLO 5, YOLO 5 , and YOLO 7. The best precision the three models had was 90.88% with YOLO 5, with a recall of 83.05% and an F1 score of 0.8052. Our Enhanced YOLO 11 model has a precision of 88.2%, a recall of 85.3%, and a better F1 score of 0.867. This means that although the precision and recall are slightly worse, the F1 score of enhanced YOLO 11 model is significantly better, indicating a better balance between precision and recall.

2) Comparison with Object Detection Frameworks in Video Analysis:

Another research compared different YOLO model variations for object detection in Indian road video data [15]. The 'Medium' variant trained for 30 epochs had a training precision of 79% and a recall of 93%, with an F1 score of 0.82. But at the testing stage, the precision of this model reduced to 30%, with a recall of 16% and an F1 score of 0.16. By contrast, our model has strong performance throughout training and testing phases, which shows its generalization ability.

3) Comparison with YOLO 2 and YOLO 3:

As per a performance metrics survey for object detection algorithms, YOLOv2 [16] scored a mean Average Precision (mAP) of 21.6% on the COCO dataset, while YOLOv3 pushed this to 33.0%. Compared with that, our Enhanced YOLO 11 model has an AP 50% of 95.6%, which is a substantial improvement in detection performance. This showst hat our version provides much accurate results.

4) Comparison with SSD and RetinaNet:

The same poll [16] indicates SSD513 with ResNet-101 with an mAP of 31.2% on COCO, while that of RetinaNet is 82.89%[21]. Both models lag behind enhanced YOLO 11 model's AP 50% of 95.6%, which indicates that Enhanced YOLO 11 has better detection abilities. This indicates how enhanced YOLO 11 model efficiently outperforms these approaches in detection accuracy, making it a great option for high-precision needs, particularly in challenging environments where object discrimination is difficult.

5) Comparison with Faster R-CNN:

Faster R-CNN with TDM [22] had an mAP of 80% on the COCO dataset. Although that is a good performance, the AP 50% of 95.6% from the Enhanced YOLO 11 model demonstrates a significant improvement in detection precision. More efficient models such as Faster R-CNN come with higher computational costs, thus are slower in real-time use. Enhanced YOLO 11, however, comes with both high accuracy and quick inference speed, including an FPS of 35.6, which is critical in real-time use. This perfect blend of speed and accuracy makes enhanced YOLO 11 model the best for applications requiring both efficiency and accuracy.

These comparisons performed a thorough analysis of various YOLO architectures and their improvements—SiMam, DySample, and Bounding Box IoU—to ascertain their perfor-

mance in wildlife detection through drone-based monitoring. Through extensive experimentation and ablation testing, we compared various model variants, quantifying their performance on important object detection metrics like mAP50, mAP50-95, GFLOPs, Precision, Recall, and F1-score. The findings point to the significant improvements introduced by these improvements, which are critical for real-world applications where accurate and efficient animal detection is needed.

The baseline YOLO models (YOLO 5, YOLO 7, YOLO 8, and YOLO 11) had high detection capacity but differed when it came to the accuracy-computational cost trade-off. While YOLO 5 offered balanced performance but with lower accuracy compared to subsequent versions, YOLO 7 offered higher recall but at the expense of higher false positives. YOLO 8 had slightly higher precision compared to YOLO 7 but also failed to offer the best balance between accuracy and computational cost. The YOLO 11 model outperformed its counterparts by offering higher accuracy at relatively low computational cost, thus making the best option for real-time drone-based wildlife observation. The addition of SiMam (Spatial and Channel-wise Attention Mechanism) significantly enhanced the detection capability of the YOLO models with improved feature extraction. This boosted the model's ability to identify animals in complex situations, especially when they were hidden in dense vegetation. The YOLO models with SiMam always registered an improvement in mAP50 and mAP50-95, which was a measure of their ability to handle more detailed information in aerial images. For instance, YOLO 11 + SiMam registered an mAP50 of 78.9%, which was a major improvement from the baseline model.

The analysis tabled in Table 4, which compares multiple object detection models, clearly highlights that Enhanced YOLO 11 is by far the most efficient model in comparison to its counterparts. It performs significantly well in key factors such as precision, where it has an outstanding 88.2%, and recall, which has a staggering figure of 85.3%. Additionally, its F1 score, an essential indicator of the accuracy of a model, is 0.86, and it scores a remarkable mAP of 95.6%. These indicators in total effectively prove that Enhanced YOLO 11 is by far the best model amongst all the ones tested. Second, the utilization of DySample, which refers to Dynamic Sampling-based Upsampling, has enabled further improvement of the model in terms of the ability to do object localization. This new way of learning uses adaptive feature offsets, and it has been of particular benefit to the detection of smaller animals or those that have been partially obstructed. Due to this addition, the model of YOLO 11 + DySample has been developed further, achieving an mAP50 of 78.0%. This outcome surpasses its predecessors in terms of YOLO and constitutes a clear validation of the capability and advantage inherent in adaptive upsampling methods.

Of all the models that were exhaustively tested and compared, the Enhanced YOLO 11 model, as indicated in Table 4, was the most effective and efficient wildlife detection system when it employed the revolutionary strategies of SiMam, DySample, and Bounding Box IoU jointly. This model had a mean Average Precision at 50% (mAP50) of 82.9%, as well as a mean Average Precision from 50% to 95% (mAP50-95) of 86.9%. Besides, it had a precision rate of 88.2% and a recall rate of 85.3%, thereby having an amazing F1-score of 0.86. These high figures mean that it surpassed every single other version in the test, all while having a computational cost as low as 7.0 GFLOPs. As such, it can be concluded that Enhanced YOLO 11 is the best choice for conducting real-time aerial monitoring as well as for assisting conservation efforts, mostly in cases where both speed and accuracy are of the utmost concern.

Model	Precision (%)	Recall (%)	F1 Score	mAP (%)
Enhanced YOLO 11	88.2	85.3	0.86	95.6
YOLO 8	79.0	50.0	0.82	37.3 - 53.9
YOLO 7	84.9	89.07	0.81	51.7 - 56.8
YOLO 5	84.27	83.052	0.80	-
Enhanced SSD	75	73	-	-
YOLO 2	75.4	70.3	0.72	21.6
YOLO 3	80.2	76.1	0.78	33.0
RetinaNet	-	-	-	82.89
Faster R-CNN	-	-	-	80

TABLE 4. Performance comparison of different models based on Precision, Recall, F1-Score, and mAP.

V. CONCLUSION AND FUTURE WORK

The Enhanced YOLO 11 model, when combined with SimAM, DySample, and Bounding Box-IoU, improves object detection for small objects in wildlife monitoring using UAVs considerably. Through the combination of these complex modules, the model balances between computational complexity and detection accuracy to make it suitable for real-time deployment. In addition, the provision of the WAID dataset gives an important source of data for further research into aerial wildlife surveillance to fill the gap in available UAV datasets that has previously existed. The model's performance has been shown by rigorous comparative tests to be effective and robust under a range of environmental conditions.

Future research will be directed at optimizing Enhanced YOLO 11 further to enhance real-time processing on resource-constrained embedded platforms. Also, work will be invested in extending the WAID dataset [11] to cover a larger range of animal species and geographies. Investigating multi-modal data fusion, fusing thermal with regular aerial imagery, may also enhance detection in difficult environments like low-light or high-vegetation. Last, the inclusion of semi-supervised learning methodology would increase model generalization by utilizing unlabeled samples, thus making the system more accurate.

REFERENCES

[1] Dingran Wang, Jiasheng Tan, Hong Wang, Lingjie Kong, Chi Zhang, Dongxu Pan, Tan Li, Jingbo Liu, SDS-YOLO: An improved vibratory

position detection algorithm based on YOLOv11, https://doi.org/10.1016/j.measurement.2024.116518.

[2] Beaver, J. T., Baldwin, R. W., Messinger, M., Newbolt, C. H., Ditchkoff, S. S., Silman, M. R. (2020). Assessing the use of drones and thermal sensors to enumerate wildlife effectively. Wildlife Society Bulletin, 44, 434–443. https://doi.org/10.1002/wsb.1090

[3] Kangunde, V., Jamisola, R. S., Theophilus, E. K. (2021). A review on drones controlled in real-time. International Journal of Dynamics and Control, 9, 1832–1846. https://doi.org/10.1007/s40435-020-00737-5

[4] Madasamy, K., Shanmuganathan, V., Kandasamy, V., et al. (2021). OSDDY: Embedded system-based object surveillance detection system with small drone using deep YOLO. Journal of Image and Video Processing, 19(2021). https://doi.org/10.1186/s13640-021-00559-1

[5] Roy, A. M., Bhaduri, J., Kumar, T., Raj, K. (2023). WilDect-YOLO: An efficient and robust model for automated endangered wildlife detection. Ecological Informatics, 75, 101919. https://doi.org/10.1016/j.ecoinf.2022.101919

[6] Ivanova, S., Prosekov, A., Kaledin, A. (2022). A survey on monitoring of wild animals during fires using drones. Fire, 5(3), 60. https://doi.org/10.3390/fire5030060

[7] Yang, W., Liu, T., Jiang, P., Qi, A., Deng, L., Liu, Z., He, Y. (2023). A forest wildlife detection algorithm based on improved YOLOv5s. Animals, 13(19), 3134. https://doi.org/10.3390/ani13193134.

[8] Ma, D., Yang, J. (2022). YOLO-Animal: An efficient wildlife detection network based on improved YOLOv5. 2022 International Conference on Image Processing, Computer Vision and Machine Learning (ICICML), Xi'an, China, 464–468. https://doi.org/10.1109/ICICML57342.2022.10009.

[9] Lu, X., Lu, X. (2023). An efficient network for multi-scale and overlapped wildlife detection. Signal, Image and Video Processing, 17, 343–351. https://doi.org/10.1007/s11760-022-02237.

[10] Liu, C., Tao, Y., Liang, J., Li, K., Chen, Y. (2018). Object detection based on YOLO network. 2018 IEEE 4th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 799–803. https://doi.org/10.1109/ITOEC.2018.8740604.

[11] Chao Mou, "WAID: Wildlife Aerial Image Dataset," GitHub repository, 2023. [Online]. Available: https://github.com/xiaohuicui/WAID.

[12] Hodgson, Jarrod C.; Mott, Rowan; Baylis, Shane M. et al. (2019). Data from: Drones count wildlife more accurately and precisely than humans [Dataset]. Dryad. https://doi.org/10.5061/dryad.rd736.

[13] Cow video data set: https://mega.nz/file/ c0sSRbTS Be-Qzqm4ylvO7UgdgcWErx7rcnWf2UGW5wJx9jWyyxBk

[14] Stark T, Ştefan V, Wurm M, Spanier R, Taubenböck H, Knight TM. YOLO object detection models can locate and classify broad groups of flower-visiting arthropods in images.

[15] Padia A, T N A, Thummagunti S, Sharma V, K Vanahalli. Object Detection and Classification Framework for Analysis of Video Data Acquired from Indian Roads. Sensors (Basel). 2024 Sep 29;24(19):6319. doi: 10.3390/s24196319. PMID: 39409360; PMCID: PMC11479008.

[16] Rafael Padilla1, Sergio L. Netto2, Eduardo A. B. da Silva1,2,3PEE, COPPE, Federal University of Rio de Janeiro, P.O. Box 68504, RJ, 21945-970, Brazil.

[17] Zientara, P. A., Choi, J., Sampson, J., Narayanan, V. (2018). Drones as collaborative sensors for image recognition. 2018 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 1–4. https://doi.org/10.1109/ICCE.2018.8326187.

[18] Bakana, S. R., Zhang, Y., Twala, B. (2024). WildARe-YOLO: A lightweight and efficient wild animal recognition model. Ecological Informatics, 80, 102541. https://doi.org/10.1016/j.ecoinf.2024.102541.

[19] Lima, C. M. de A., da Silva, E. A., Velloso, P. B. (2018). Performance evaluation of 802.11 IoT devices for data collection in the forest with drones. 2018 IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, United Arab Emirates, 1–7. https://doi.org/10.1109/GLOCOM.2018.8647220.

[20] Li, S., Zhang, H., Xu, F. (2023). Intelligent detection method for wildlife based on deep learning. Sensors, 23(24), 9669. https://doi.org/10.3390/s23249669.

[21] Lu, Y., Lu, S., Zheng, W. (2021). Comparison of RetinaNet, SSD, and YOLO v3 for real-time pill identification. BMC Medical Informatics and Decision Making, 21, 232.

[22] Tahir, H., Khan, M. S. (2021). Performance Analysis and Comparison of Faster R-CNN, Mask R-CNN, and ResNet50 for the Detection and Counting of Vehicles. 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), 587–592.

DR. SHERLY A is a renowned researcher with research interests in Affective Computing, Artificial Intelligence, and Image Processing. Before going for her Ph.D. She completed her Master of Technology (M.Tech) in Computer Science from Anna University in 2006, where she studied extensively about advanced computing concepts, algorithms, and software development methodologies. This academic foundation reinforced her background in artificial intelligence and machine learning, equipping her with the appropriate skills to conduct high-impact research. Her research interests span a range of fields, including Image Processing, Machine Learning, Artificial Intelligence, and Affective Computing. She is particularly interested in designing algorithms that enable machines to perceive, comprehend, and respond to human emotions in an efficient way. Her research has practical implications in healthcare, human-computer interaction, and automated surveillance systems.

CHITRANSHU GUPTA is currently pursuing the degree in computer science engineering with Vellore Institute of Technology, Chennai. He is committed to developing smart applications that can scale and incorporate new models of machine learning to better serve the user needs and facilitate decision-making. Equipped with a good understanding of academic sources, he has been instrumental in bringing new web technologies closer to modern artificial intelligence. He has experience in designing and implementing intelligent systems with AI models integrated into real-world applications. He is comfortable working with Python, TensorFlow, Node.js, and cloud computing and has expertise in domains like NLP, and computer vision.

MOHD KAMRAN WARSI is currently pursuing the degree in computer science engineering with Vellore Institute of Technology, Chennai. His experience is all about creating scalable, secure, and efficient backend systems that involve higher-level technologies like Spring Boot, Node.js, Redis, Kafka. With a strong background in database management, API optimization, and cloud-based architecture, He is dedicated to creating strong, high-performance backend systems.Through his rich and thorough knowledge of database management practices, API optimization practices, and cloud-based design principles, He showcases a high degree of dedication towards backend system design that is high-performance and reliable. Such systems have a crucial role to play in driving the functionality and efficiency of contemporary web applications that are accessed by users on a day-to-day basis.

KARTIK BAGHEL is currently pursuing a degree in Computer Science and Engineering at Vellore Institute of Technology, Chennai. He is an expert in Machine Learning, Artificial Intelligence, and Full-Stack Web Development. With a keen interest in applying data-driven technologies, he has created projects utilizing deep learning, predictive analytics, and scalable web applications. He has experience in designing and implementing intelligent systems with AI models integrated into real-world applications. He is comfortable working with Python, TensorFlow, Node.js, and cloud computing and has expertise in domains like NLP, computer vision, and recommendation systems. His ability to merge backend technologies with machine learning models allows him to create very efficient and automated applications.