A Deep Learning-Based Real-Time Surveillance Framework for Mitigating Herbivorous Animal Intrusions in Rice Nursery Fields

Muhammad Juman Jhatial, Riaz Ahmed Shaikh, Rafaqat Hussain Arain
Institute of Computer Science, Shah Abdul Latif University, Khairpur Mirs, Sindh, Pakistan

***Abstract*—The rice nursery stage is a very critical stage of the rice growing cycle, as intrusion of herbivorous animals can lead to substantial economic losses and yield losses. Manual guarding and physical fencing are inefficient and ineffective. Real-Time Animal Intrusion Detection Network (RAID-Net) is an end-to-end intelligent agricultural surveillance system that consists of an integration of YOLO and a multi-task CNN to detect and recognize herbivorous animal intrusion in open-field rice nursery fields in real time. To this end, a novel and perfectly balanced dataset of 14,000 images with five different herbivore species (Buffalo, Camel, Cow, Goat, and Sheep) was curated, and a variety of seven YOLO models and five CNN architectures were benchmarked to identify the optimal detection and classification backbones. The model design includes a lightweight YOLOv8 Nano object detector and a custom-designed Multi-task Convolutional Neural Network (CNN) for species classification and behavior detection grazing and moving. The results of the Customized Multi-task CNN on cropped animal regions showed 97.72% species classification accuracy and 98.14% behaviour recognition accuracy. The RAID-Net with YOLO detection-aware alert pipeline yielded 90.05% species accuracy, 94.31% behaviour accuracy, and 82.73% hybrid accuracy, with an end-to-end latency of 17.61 ms and an inference rate of 56.79 FPS on edge-compatible hardware with an alert precision of 98% and recall of 89%. The system differentiates between high-threat grazing and low-threat moving behaviors, automatically activating a siren deterrent and sending instant WhatsApp/SMS alerts to farmers, providing an automated, scalable, and very accurate solution for precision agriculture and crop protection.**

Keywords*—Precision agriculture; deep learning; Yolo model; convolutional neural networks

I. Introduction

Rice farming is the basic livelihood activity for food security in Pakistan and plays a significant role in rural economies. The nursery stage is crucial for the establishment of seedlings, with a duration of 20-30 days.

At this time, rice plants are delicate and prone to grazing and trampling by domestic herbivorous animals [1]. Out of many herbivorous animals, five such animals were chosen for this study as they are seen with regularity in rice nursery fields and have a marked effect on the damage to the crop (Buffalo, Camel, Cow, goat, and Sheep). Illustrated in Fig. 1.

Fig. 1. Dataset images.

The rice nursery stage is one of the most critical periods of the rice production cycle, especially in open field conditions where the intrusions of herbivorous animals may lead to damage to young seedlings, which may be irreversible and fast. Domestic animals like buffalo, camel, cow, goat, and sheep often intrude into the nursery fields for food and have strong grazing ability, which easily strips newly planted rice plants, resulting in huge economic losses for farmers in many rural agricultural areas. Current protection measures are often traditional, reactive, and labour-intensive, such as manual guarding, fencing, and scare devices [2]. They are labour-intensive, time-consuming, and difficult to scale up to large areas of agriculture, particularly when animals can be in fields without being monitored and cannot be seen at night or in unsuitable weather. Recently, computer vision and smart agriculture technologies have emerged with potential, but most of the proposed methods are either object detection or species classification and fail to consider behavior analysis, severity assessment of intrusions, and automatic alert generation. Hence, there is a critical need for a specialized end-to-end intelligent surveillance framework that can detect, identify, and assess threats based on species behavior for proactively protecting the rice nursery field in real-time and enhancing agricultural resilience.

A. Computer Vision and Deep Learning

The past few years have seen tremendous progress in computer vision and deep learning, which has led to the development of robust object detection and image classification algorithms that can perform these tasks in real time with high accuracy in challenging scenarios. In agriculture, intelligent systems can also be used to replace manual field supervision using the detection and tracking of animals like cows, goats, and buffalo in rice paddy fields, as shown in Fig. 2, which is a promising solution for agricultural monitoring.

Fig. 2. Computer vision system monitoring a rice paddy field.

However, the existing agricultural monitoring frameworks are still limited in their functions. Many systems are limited to the mere detection of animal movement, and not necessarily on the basis of what it means in terms of animal behavior, which makes it challenging to distinguish harmless animal movement from destructive grazing behavior. On the other hand, there are also studies that focus on species classification but do not have the capability of spatial localization because they work on images that have been pre-segmented or cropped instead of identifying the animals in large-scale real scenes in the agricultural field. Consequently, existing methods focus on detection or classification only, and there is a significant research gap in how to combine spatial localization with species identification and behaviour understanding into a single entity. Considering this shortcoming, this study presents a comprehensive deep learning-based surveillance system, especially for protecting the rice nursery field. An annotated herbivore intrusion dataset of five domestic animals was built to facilitate training and evaluation of the model. Various detection models and CNN architectures based on YOLO were extensively benchmarked to choose the best configurations considering the accuracy, inference speed, and computational efficiency. Based on the above results, a new hybrid model fused YOLO for detection and a multi-task CNN for classification, which can simultaneously localize, identify the species, and recognize the behaviors of the animals, was constructed. Moreover, a behavior-aware threat assessment mechanism was added to distinguish the difference in movement between the animal moving and the destructive use of the animal to graze, enabling context-aware decision-making. Last but not least, a real-time alert and deterrence pipeline was put in place, comprising automated siren activation and mobile notifications to enable farmers to respond immediately, rather than relying on passive observation, leading to a shift from passive monitoring to proactive crop protection.

II. Related Work

The integration of artificial intelligence (AI), machine learning (ML), and deep learning (DL) in agriculture has emerged as an effective solution to challenges related to population growth, climate change, and resource scarcity, improving crop management, resource optimization, and pest control. In this comprehensive review, the authors investigate the potential of machine learning and deep learning applications for agricultural use. Food security is still defined largely by crops, livestock, and fisheries, but these are subjected to serious challenges including climate variability, scarcity of resources, and sustainability issues. Recent research has shown that AI can be used to solve problems, from simple machine learning to more complex deep learning and vision-language models, in various fields like crop disease detection, livestock health monitoring, and aquatic species management. Yet, challenges such as data variability, scalability constraints, and deployment constraints underscore the importance of multimodal data integration and flexible AI solutions for a variety of farming contexts.

A. AI in Agriculture: A Survey of Deep Learning Techniques for Crops, Fisheries and Livestock.

Consequently, deep learning has emerged as a significant technology in precision agriculture, especially when it comes to the early detection of crop diseases that pose threats to yield and quality. This study used multi-modal approaches on potato, tomato, grape, apple, and peach leaves and found that VGG16, MobileNetV2, Xception, and ResNet all performed quite well, with the metrics of Accuracy, Precision, Recall, and F1-Score. The findings indicate that VGG16 achieved 99% efficiency, suggesting that scalable, non-invasive AI systems could provide valuable plant disease management tools, all supported by farmer trust in AI systems.

Judging from the title, the authors have a very creative methodology for enhancing agricultural disease detection. Judging from the title, the authors have a very creative methodology for enhancing agricultural disease detection.

Food sustainability is a growing global challenge due to rapid urbanization and globalization, and AI systems can provide new solutions to improve productivity and food security. Crop Diseases like chlorosis continue to pose a significant threat in India, where more than half of the population relies on agriculture for their livelihood, despite significant strides in the past, including the Green Revolution. The existing detection method has shortcomings in accuracy, and deep learning technology is a more reliable and scalable solution for disease detection in the context of variable environments and climates.

B. Leveraging Deep Learning and Neural Networks for Real-Time Crop Monitoring in Smart Agricultural Systems Srinivas Kalyan Yellanki

Precision agriculture and plant disease diagnosis rely heavily on early detection and accurate diagnosis, especially when plant diseases are a significant threat to food security worldwide. In recent years, studies have emerged that show the ability of computer vision and deep learning methods such as CNNs, transformer vision, and GANs combined with imaging technologies like RGB, multispectral, and hyperspectral sensors. A great deal of research reviews have documented progress and identified challenges associated with the use of these models and have offered useful insights on how these models are applicable for sustainable agricultural production. The use of deep learning and computer vision has been proposed as a comprehensive review of the techniques, models, and trends in precision agriculture for the purposes of plant disease detection. Deep learning and computer vision in plant disease detection: a comprehensive review of techniques, models and trends in precision agriculture. Artificial Intelligence (AI) and Deep Learning (DL) are two technologies that have the promise of revolutionizing agriculture and offering new solutions to some of its biggest problems, including productivity, sustainability, labour shortages, and environmental monitoring. Intelligent systems, such as computer vision, machine learning, and edge computing, have been developed to perform various tasks like crop classification, weed and pest detection, yield estimation, soil health analysis, and automatic irrigation control, etc., in recent years [1, 2]. Although these all yield different results, deep learning, particularly using Convolutional Neural Networks (CNNs), has been very effective for image processing applications such as plant disease recognition, fruit grading, phenol typing, livestock monitoring, and more [3] [4]. For object detection in real time, there are some popular frameworks like YOLO (You Only Look Once), Faster R-CNN, and SSD (Single Shot Multi-Box Detector) which are used for agriculture-related objects like plant, pest, fruit, and animal object detection [5]. The models can be used in precision agriculture: automatic identification and localization of certain entities in images or real video. Furthermore, AI-driven Unmanned Aerial Vehicles (UAVs), autonomous tractors, and robotic harvesters have started to change the dynamics of field operations, making them more data-informed and independent in how they are conducted. Moreover, Recurrent Neural Networks (RNNs) and LSTM models are applied to time series prediction tasks such as prediction of the weather, crop yield, irrigation scheduling, etc. [6]. But there remains a whole host of issues to be resolved, including: availability of large annotated data sets; regional and seasonal generalizability of models; ability to deploy models on low-cost edge devices suitable to rural environments. While there have been significant advances in crop and disease detection, the ability to detect livestock and/or animal incursions in some high-risk sites, e.g., rice nurseries, has not been well explored. This points to the need to develop more specialized and domain-specific deep learning systems to address the technical and practical issues in the agri-environment. 2.2 Animal detection or monitoring systems in place. The traditional methods of animal counting are still challenging in livestock management, and drone-based methods are also not easy. This study presents the Mask YOLOv7, an advanced object detection and instance segmentation model, for detecting and counting cattle in both controlled and uncontrolled settings, such as feedlots and open ranges. The experiment results reveal that the model has high accuracy of 93% and 95%, respectively, which makes it possible to apply it in livestock inventories and monitoring and also in intelligent farm management applications. Drone Vision System for Automated Cattle Detection and Counting using Mask YOLOv7 is an innovative technology that aims to monitor cattle from a distance using an unmanned aerial vehicle (UAV). Despite the importance of cows in daily life and nutrition, especially in India, maintaining healthy cows can be a big problem, and lameness is one of the most costly problems in dairy herds. This study introduces a Smart Cow Health Monitoring System (ScHeMoS) that uses an IoT sensor system and, in particular, the LSTM deep learning model to analyze the behavior and physiological parameters of the cow for the early detection of lameness. The model showed good accuracy of 92.45% and an F1 score of 0.92 on the self-collected set, suggesting its potential feasibility for practical farm management and animal welfare practices.

C. Schemos Smart Cow Health Monitoring System: An IoT-Based Cow Hoof Detection And Healthcare Alert System Using LSTM Network

Regularly checking on a pet is very important to lower health risks, but checking on external behavior can be misleading. In this study, a video-based monitoring system that extracts behavioral features from the image, detects anomalies, and alerts to deviations from common behavior is proposed using Mask R-CNN. Hamster experiments showed the effectiveness of the system in recognizing abnormal and normal movements, and future work plans involve increasing the sensitivity of the system to move towards recognizing more subtle movements. Monitoring systems for animals based on behavioral information and internal state information. Regularly checking on pets is important to minimize health risks, but often external behavior is a poor indicator of internal condition. In this study, a video-based monitoring system with Mask R-CNN was proposed to obtain behavioral features and detect anomalies, making alerts whenever there are anomalies in the behaviors. Experimental tests with hamsters showed the efficacy of the system to discriminate between "normal" and "abnormal" behaviors, and further developments would involve increasing the sensitivity of the system to subtle movements.

D. Research on Behavior Recognition and Online Monitoring System for Liaoning Cashmere Goats Based on Deep Learning

Automated cameras (e.g., camera traps) are also deployed for wildlife monitoring, but there are some limitations in their effectiveness, including imperfect detection, limited storage, and power supply. This work introduces a customizable camera system with YOLOv5 integrated into a Raspberry Pi, which is used for continuous monitoring of fast-moving organisms in field conditions and is demonstrated in mason bees. The system was proven to be reliable, with an average detection precision of 78%, and it was found to be a low-cost, modular, and researcher-friendly solution to commercial camera traps. Monitoring high-speed animals, creating a specific camera system, and developing an evaluation toolset.

In the past few years, the detection and monitoring of animals has become a rising concern, especially in the fields of livestock management, wildlife conservation, and agricultural protection. There are a large number of works that used deep learning methods, including convolutional neural networks (CNNs) and object detection methods like YOLO (You Only Look Once), SSD (Single Shot Detector), and Faster R-CNN to detect animals in images or video streams [1],[2]. These are usually used to monitor open pasture areas [3], forest areas [4], or urban wildlife zones [5]. Although they have shown promising results in detecting animals like cattle, deer, or wild boars, most of the existing models are developed for generalized detection tasks, e.g., animal/non-animal or livestock/wildlife.

Moreover, the current animal monitoring systems are typically designed for controlled conditions, high-quality video resolution, and background stability, which are seldom observed in the real agricultural environment [6]. Many models are unable to be optimized for inference speed on edge devices, making it difficult to obtain a good accuracy rate. There is a lack of lightweight architectures and applications that are mobile-friendly and suitable for real-time application, especially in rural rice fields [7].

Furthermore, the problem of species-level identification of herbivorous animals (buffalo, cow, goat, sheep, and camel), which are visually similar in some aspects, is significant for prioritizing alerts based on the severity of crop damage, and few studies address this issue. In the same way, in the majority of works, behavioral analysis, including animal movement tracking or prediction of grazing, is still not well studied. Some work has used GPS collars or radio frequency identification (RFID) to track livestock [8], but such methods are not scalable and not suitable for free-roaming livestock entering rice fields. There is also a lack of real-time farm monitoring solutions, including the use of early warnings via SMS or mobile app, which further diminishes the usefulness of these solutions in practice [9].

III. Materials and Methods

A real-time deep learning-based surveillance framework, RAID-Net, is proposed to protect rice nursery fields from domestic herbivorous animal intrusions like buffalo, camels, cows, goats, and sheep. The architecture is built on a multi-stage pipeline architecture with integration of object detection, behavioral analysis, and automated response mechanisms, as shown in Fig. 3.

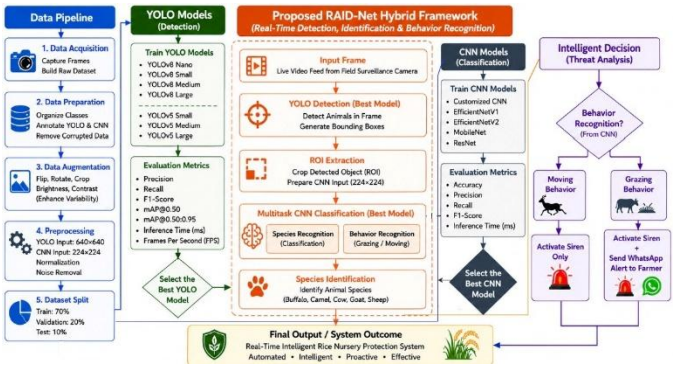


Fig. 3. Flowchart of research methodology and development pipeline.

This figure shows the five major steps in the research lifecycle: (Phase 1), Data Pipeline (acquisition and augmentation), YOLO testing and benchmarking (Phase 2), and CNN testing and benchmarking (Phase 3). It also explains the incorporation of these models in the RAID-Net (Phase 4). The final logic-based alert generation system (Phase 5) for protection of rice nursery fields.

A. Phase 1: Data Pipeline

The data pipeline can be used to systematically move images from acquisition to preprocessing to model inference, which makes it easy to detect and identify herbivorous animals (Fig. 4).

B. Data Collection

The data set contains a total of 14000 RGB images from various sources, such as rural field areas, open grazing areas, and publicly available data sets [14, 15, 16, 17]. The images are classified into 5 classes: Buffalo, Camel, Cow, Goat, and Sheep, and each class contains approximately 2800 images, which is a nice balance for training and testing of the model. Shown in Tables I and II.

TABLE I. Dataset Acquisition Characteristics

Source	Geographic location	Acquisition	Device source	Day/Night	Images
Rural field dataset	Larkana, Sindh, Pakistan	Field	Digital camera	Day	3,420
Open grazing dataset	Dadu, Sindh, Pakistan	Field	Smartphone	Day	3,680
Public Dataset A	Kaggle.com	Public dataset	Internet	Variable	3,510
Public Dataset B	Shutterstock.com	Public dataset	Internet	Variable	3,390
Total					**14000**

Fig. 4. Raw images.

TABLE II. CLASS-WISE DISTRIBUTION OF DATASET

Animal Class	Number of Images
Buffalo	2800
Camel	2700
Cow	2900
Goat	2700
Sheep	2800
Total	14000

C. Dataset Annotation and Labelling

The full set of data is systematically annotated and labelled for object detection using YOLO as well as CNN for classification. In the case of the YOLO models, the instances of animals in the whole dataset were manually cropped, and the class labels were manually assigned to the cropped objects. By having this annotation and labelling procedure, the data preparation and class representation for both the detection and the classification phases of the proposed framework were consistent [10], [11]. The dataset labelling is illustrated in Fig. 5 for classification models.

Fig. 5. Dataset labelling.

D. Dataset Splitting for Model Development.

To support proper model development, we divided the dataset as follows: 70% for training, 20% for validation, and 10% for testing [12], [13]. This is illustrated in Fig. 6 and Table III.

Fig. 6. Dataset splitting.

TABLE III. DATASET DISTRIBUTION FOR TRAINING, VALIDATION AND TESTING

Dataset Split	Percentage	Number of Images
Training set	70%	9800
Validation set	20%	2800
Testing set	10%	1400
Total		14000

E. Images Preprocessing and Augmentation

Pre-processing of images was done to prepare them for model training. This involved resizing the images to 640X640 for the YOLO model and 240X240 for the CNN model, and then normalizing the images. To improve the model's robustness and generalization, as shown in Fig. 7, data augmentation methods like horizontal flipping, brightness variations, and random cropping techniques were employed. Horizontal flipping, 15% brightness, and resizing are preprocessing steps [18],[19],[20].

Fig. 7. Data preprocessing and data augmentation.

F. Phase 2: Detection Module (YOLO)

The Detection Module in the proposed RAID-Net framework is based on the YOLO architecture [21]; Fig. 8 shows the YOLO Architecture below.

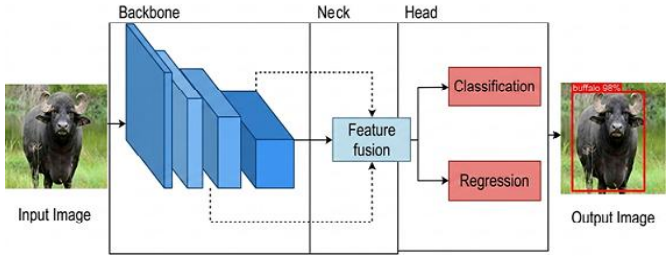


Fig. 8. YOLO architecture.

The system starts with a high-resolution camera that takes images from the field, which are then resized to 640×640 pixels and then represented as:

$$I \in R^{640\times640\times3}$$

The YOLOv8-Nano detector works on the input image and outputs a list of bounding boxes, confidences, and class probabilities:

$$f_{YOLO}(I) = \{B_i, C_i, S_i\}_{i=1}^{N}$$

where, B_i denotes the bounding box coordinates, C_i represents the confidence score, and S_i indicates the class probability for the i^{th} detected object.

The composite loss function is used for the YOLO model:

$$L_{YOLO} = L_{BOX} + L_{OBJECT} + L_{CLS}$$

where, L_{BOX} represents bounding box regression loss, L_{OBJECT} denotes objectness loss, and L_{CLS} corresponds to classification loss. Together, they optimize the model to get accurate localization and reliable detection. Real-time agricultural surveillance of herbivorous animals that can be detected and effectively classified.

G. Phase 3: Multi-Task CNN Module (Species and Behavior Classification)

The Multi-Task CNN Classification Module is shown in Fig. 9 of the CNN model architecture [22].

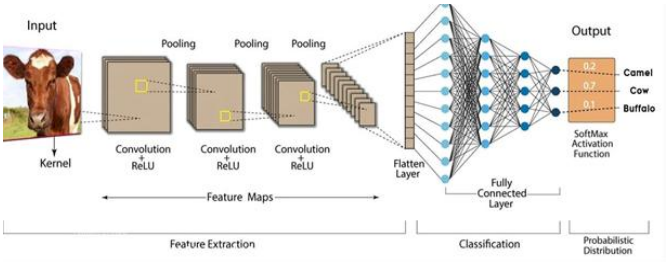


Fig. 9. CNN Architecture.

CNN-based classification stage for the RAID-Net framework:

$$F = CNN(ROI), \quad P_s = Softmax(W_s F + b_s), P_b = Softmax(W_b F + b_b)$$

$$L_{total} = \lambda 1 L_{species} + \lambda 2 L_{behavior}$$

F = feature vector from the cropped ROI (240 × 240 pixels)

P_s, P_b = predicted probabilities for species and behavior

W_s, b_s, W_b, b_b = classifier parameters

L_{total} = weighted multi − task loss balancing species and behavior learning

H. RAID-Net and Alert Phases

$$\text{Alert} = \begin{cases} Higt - Priority\ if\ S_{SPECIES} = Animal, S_{Behaviour} = Grazing, C_{Hybrid} \geq T_{Threshold} \\ Meduim - Priority\ if\ S_{SPECIES} = Animal, S_{Behaviour} = Moving, C_{Hybrid} \geq T_{Threshold} \end{cases}$$

$$\text{Response} = \begin{cases} Grazing: Acitvate\ Siren + Send\ WhatsApp\ Alert \\ Moving\ Activate\ Siren \end{cases}$$

$S_{species}$, $S_{behavior}$ = predicted species and behavior

C_{hybrid} = hybrid confidence score from fusion layer

$T_{threshold}$ = minimum confidence to trigger an alert

The outputs are connected to the real-world system, such as the siren hardware and WhatsApp API, allowing real-time deterrence and remote farmer notification. This is a combination of AI and rules, providing both reliability in operations and readability in agricultural field deployments. The RAID-Net architecture is shown in Fig. 10.

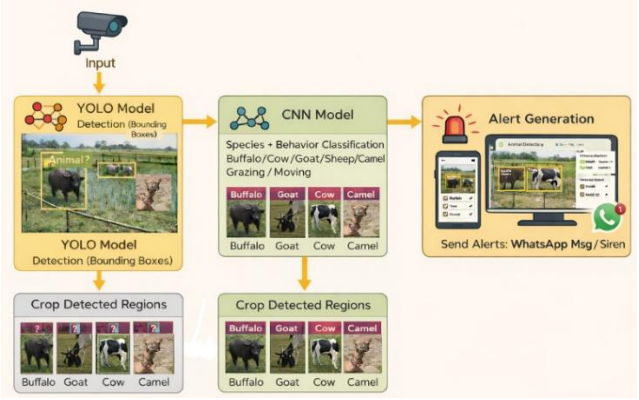


Fig. 10. Architectural framework of RAID-Net.

The diagram shows the actual operation of the surveillance system from the acquisition of the video stream from the input camera. It illustrates the multi-step process of passing the output of a YOLO model for object localization through a CNN model for fine-grained species and behavior classification, and then to either automated WhatsApp alerts or audible sirens, depending on the intrusion detected.

IV. Experimental Setup

To ensure the flexibility and scalability of the experimental setting, the proposed RAID-Net framework was implemented on cloud-based GPU platforms as well as a local computational system to facilitate reproducibility of the experiment. The implementation of YOLO based detection models and integration experiments in a hybrid setup consisted of using Google Colab with Python 3 and NVIDIA Tesla T4 GPU acceleration, which proved to be the most effective for training the YOLOv5 and YOLOv8 architectures with PyTorch and the Ultralytics framework. The multi-task CNN classification experiments were conducted on Kaggle Notebooks using GPU-enabled environments (P100/T4) that are optimized for deep learning tasks, with TensorFlow and Keras APIs for developing the models and hyperparameter tuning [23]. We also validated and tested the model on a local system with an Intel Core i5 (12th Generation) CPU, 16 GB of RAM, 250 GB SSD storage, and an NVIDIA graphics card for model inference, lightweight training, and system-level integration. It integrates computing infrastructure to ensure stability in handling the datasets, faster speed for model training, and repeatability of the performance evaluation across different hardware configurations.

TABLE IV. YOLO Training Configuration

Category	Parameters	Values
Model Settings	Base Model	YOLOv8x
	Task	Object Detection
	Image Size	640 × 640
	Epochs	100
Optimization	Optimizer	SGD
	Initial Learning Rate	0.01
	Momentum	0.937
	Weight Decay	0.0005
Hardware / Device	GPU	NVIDIA(CUDA enabled)
	CPU	Intel Core i5 12th Gen
	RAM	16 GB
	Storage	250 GB SSD
	Workers	8
Data Augmentation	HSV Hue	0.015
	HSV Saturation	0.7
	HSV Value	0.4
	Horizontal Flip	0.5
	Mosaic	1.0
Dataset	Data Configuration	data.yaml

This specific YOLOv8x model was optimized for 100 epochs with 16 images per batch and an input resolution of 640×640 pixels. The optimizer used was SGD with an initial learning rate of 0.01, a momentum of 0.937, and a weight decay of 0.0005. To enhance generalization, data augmentation methods were used, such as HSV color augmentation, horizontal flipping, and mosaic augmentation. All experiments were carried out on an Intel Core i5 (12th Generation) system with 16 GB of RAM and NVIDIA GPU acceleration, as shown in Table IV.

TABLE V. CNN MODEL CONFIGURATION AND TRAINING PARAMETERS

Category	Parameter	Value
Architecture	Backbone	Custom CNN
	Input Size	224 × 224
	Species Classes	5
	Behavior Classes	2
	Dropout	0.5
Optimization	Optimizer	Adam
	Learning Rate	0.001
	Weight Decay	0.0001
Loss	Species Loss	Cross Entropy
	Behavior Loss	Cross Entropy
	λ_1 (Species)	1.0
	λ_2 (Behavior)	1.0
Training	Epochs	100
	Batch Size	16
Hardware	GPU	NVIDIA CUDA
	RAM	16 GB

Table V illustrates that the proposed custom CNN model has an input size of 224 x 224. It has 5 classes for species and 2 classes for behavior. To avoid overfitting, a dropout layer with a rate of 0.5 is used [24,25]. The network is trained with the Adam optimizer, learning rate = 0.001 and weight decay = 0.0001. Both the loss function for the species and the loss function for the behavior classification are cross-entropy with λ_1 = 1.0 for species and λ_2 = 1.0 for behaviour classification. Training is done for 100 epochs, using a batch size of 16. All experiments are run on a system with an NVIDIA CUDA GPU and 16 GB of RAM.

Table VI shows that the CNN model has three convolutional blocks with an input size of 224×224×3. A block consists of a convolutional layer (3×3) and a ReLU activation followed by a max-pooling layer (2×2). There are 32, 64, and 128 filters in the convolutional layers, respectively. The feature maps are then flattened (86,528 units) and fed into a fully connected dense layer with 256 units, with a dropout layer with a rate of 0.5 to avoid overfitting. Finally, two parallel dense layers are used as outputs: species classification (5 classes) and behaviour classification (2 classes), both with Softmax activation. The total number of trainable parameters is 22,246,471 (~84.86 MB).

TABLE VI. PARAMETERS OF MULTI-TASK CNN MODEL

Layer (Type)	Output Shape	Parameters	Description
InputLayer	(224×224×3)	0	Input image
Conv2D (3×3, 32)	(222×222×32)	896	ReLU activation
MaxPooling2D (2×2)	(111×111×32)	0	Pooling layer
Conv2D (3×3, 64)	(109×109×64)	18,496	ReLU activation
MaxPooling2D (2×2)	(54×54×64)	0	Pooling layer
Conv2D (3×3, 128)	(52×52×128)	73,856	ReLU activation
MaxPooling2D (2×2)	(26×26×128)	0	Pooling layer
Flatten	86,528	0	Flatten feature maps
Dense (256)	256	22,151,424	Fully connected layer, ReLU
Dropout	256	0	Dropout 0.5
Dense (5)	5	1,285	Species classification (Softmax)
Dense (2)	2	514	Behavior classification (Softmax)

Total parameters: 22,246,471 (~84.86 MB)
Trainable parameters: 22,246,471
Non-trainable: 0

1) YOLO model benchmarking and selection: YOLOv8 Nano is the most suitable for real-time field surveillance with the fastest inference time, 193.25ms per frame, and the highest FPS, 5.17fps. YOLOv8 Large and YOLOv5 Large have much higher inference times, making them unsuitable for real-time applications. In summary, YOLOv8 Nano provides the closest compromise between accuracy of detection and the speed of the system, and is the ideal choice for the proposed herbivorous animal surveillance framework. As indicated in the below Table VII.

TABLE VII. SELECTED YOLO MODEL METRICS

Model	Pre	Rec	F1-Score	mAP@0.5	mAP@0.5:0.95	Inference Time (ms)	FPS
YOLOv8 Nano	0.8133	0.8159	0.7947	0.5821	0.3198	193.25	5.17
YOLOv8 Large	0.7298	0.7178	0.7144	0.5304	0.2894	2617.14	0.38
YOLOv8 Medium	0.7984	0.7722	0.7723	0.5241	0.2970	861.29	1.16
YOLOv8 Small	0.7917	0.7828	0.7732	0.5724	0.3537	363.09	2.75
YOLOv5 Large	0.7919	0.6683	0.6862	0.4693	0.2679	2574.43	0.39
Yolov5 meduim	0.87142	0.871429	0.861282	0.544692	0.317363	730.0422	1.369784
Yolov5small	0.60094	0.574359	0.501412	0.011558	0.001967	318.9177	3.135605

TABLE VIII. SELECTED MULT-TASK CNN METRICS

Model	Sp.Acc	Sp.Pre	Sp.Rec	Sp.F1	Be.Acc	Be.Pre	Be.Rec	Be.F1	Inference(ms)	FPS
CUSTOMIZECNN	0.9772	0.9775	0.9771	0.9771	0.9814	0.9816	0.9812	0.9814	3.18	314
EFFICIENTNETV1	0.2627	0.1319	0.2498	0.1536	0.4721	0.2361	0.5000	0.3207	6.77	147.7
EFFICIENTNETV2	0.2517	0.2470	0.2457	0.1908	0.5507	0.7050	0.5733	0.4779	2.68	372.7
MOBILENET	0.9949	0.9951	0.9950	0.9950	0.9949	0.9951	0.9947	0.9949	4.06	246.5
RESNET	0.9975	0.9975	0.9975	0.9975	0.9949	0.9950	0.9948	0.9949	3.30	303.3

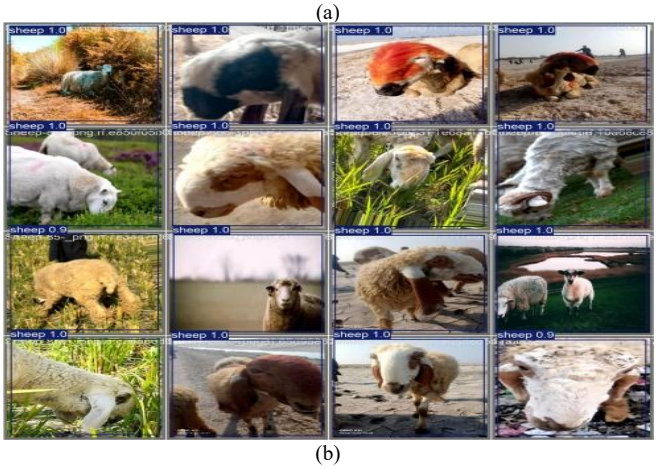
The best overall performance was the YOLOv8-Nano model, which was chosen as the backbone of the detection network in RAID-Net.

2) Selected multi-task CNN model results: In the multi-task classification experiment, CNN architectures were tested. Table VIII shows that the customized CNN model has the best inference time and frame per second (FPS). The Customized CNN is determined to be the best classification backbone in RAID-Net.

B. Results and Analysis

The image in Fig. 11 (a) shows sample outputs of an animal detection system where different herbivorous animals are identified using bounding boxes. Each box identifies the presence of animals like camel, cow, goat, sheep, and buffalo in various scenes.

(a)



(b)

Fig. 11. (a) The training images; (b) The validation images.

The image in Fig. 11 shows sample outputs of an animal detection system where different herbivorous animals are identified using bounding boxes. Every box identifies the presence of animals like camel, cow, goat, sheep, and buffalo in various scenes.

1) Accuracy comparisons of all yolo models: Fig. 12 is a grouped bar chart that compares the detection accuracy of different livestock species (buffalo, camel, cow, goat, and sheep) using various models of YOLOv5 and YOLOv8, which have different configurations of model sizes (Nano, Small, Medium, and Large). The visualization shows, in comparison, how strong the feature extraction and object localization are in each architecture. It is worth noting that the YOLOv5 Medium model achieves the best performance for these classes in terms of Precision, Recall, and F1-Score, while the YOLOv8 Nano and Small models are extremely competitive in terms of accuracy while having lower model complexity.

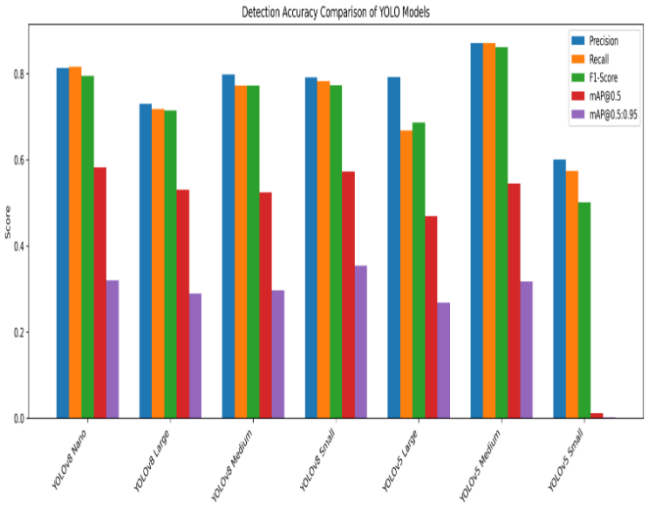


Fig. 12. Detection accuracy comparison.

2) Computational efficiency and speed: Fig. 13 shows the computational efficiency of the tested YOLO models with the help of a dual-axis graph. The primary y-axis (blue bars) represents the inference time for buffalo, camel, cow, goat, and sheep in milliseconds (ms), and the secondary y-axis (the yellow line plot) represents the processing speed (in frames per second: FPS). The figures show clearly that the complexity and the processing speed are inversely related. YOLOv8 Nano not only has the highest FPS and the shortest inference time, but it

also has good deployment potential for live monitoring in the farm.

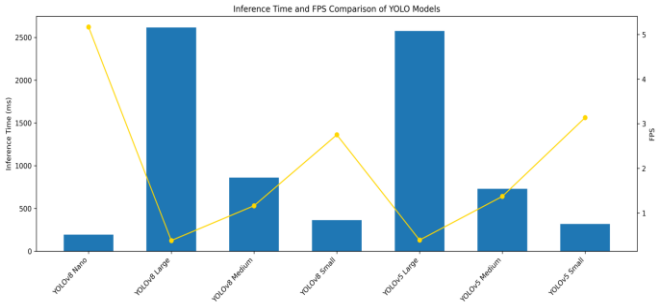


Fig. 13. Computational efficiency and speed.

3) Holistic performance: This radar graph shows a multivariate comparison of the YOLOv5 and YOLOv8 models for buffalo, camel, cow, goat, and sheep detection. The axes range from Precision, Recall, F1-Score, mAP@0.5, mAP@0.5:0.95, Inference_ms, and FPS. The shaded polygons represent the overall performance curve of each model variant, enabling a quick evaluation of the model's detection ability and processing speed. Such a holistic approach is essential to enable the proper selection of the most suitable model architecture that can give high-fidelity identification of objects while maintaining fast inference times required for continuous livestock surveillance. Shown in Fig. 14.

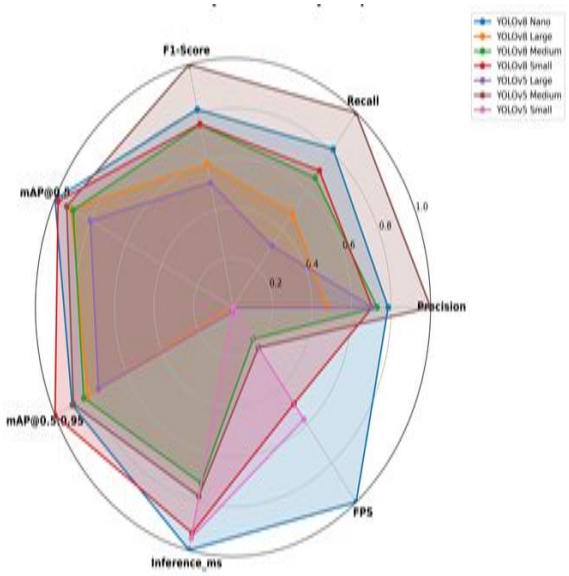


Fig. 14. Holistic performance profiling.

4) CNN model classification performance: These charts demonstrate in Fig. 15 an almost perfect accuracy (close to 1.0) in species and behavior detection of the Custom and ResNet models compared to any of the EfficientNet variants.

5) CNN models inference time and frames per second: This visualization shows that the EfficientNetV2 model has a peak speed of more than 350 FPS, but at the cost of classification accuracy compared to the Custom model, as shown in Fig. 16.

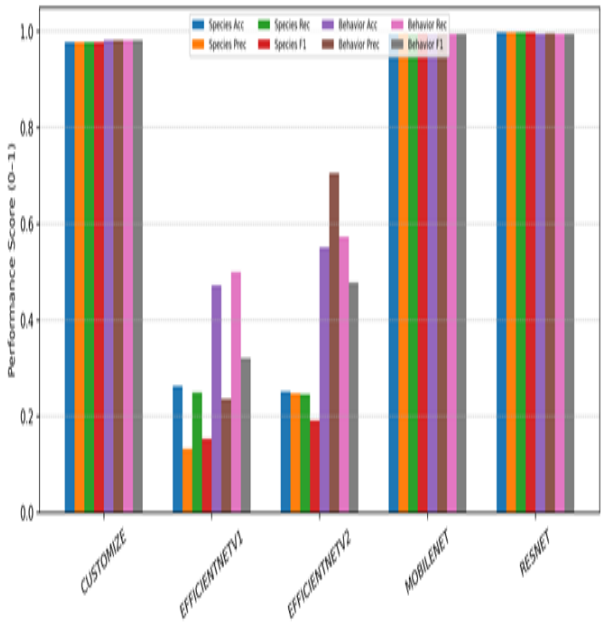


Fig. 15. Classification performance of CNN models.

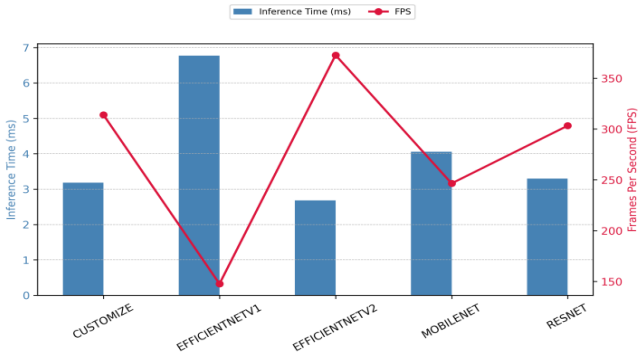


Fig. 16. CNN inference time and FPS.

6) Classification models accuracy and efficiency radar: This summary radar chart demonstrates that overall, the customized models are the most consistent and accurate on all behavior and species metrics. Shown in Fig. 17.

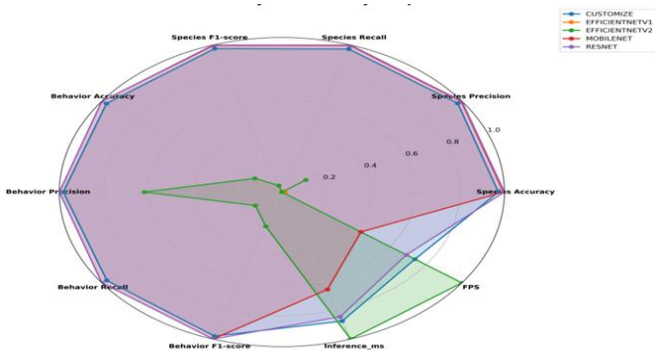


Fig. 17. CNN accuracy and efficiency radar.

7) RAID-Net results: The primary success of the integrated system is compactly summarized in this bar chart: With high Species Accuracy (0.9005) and Behavior Accuracy (0.9431), an

outstanding Alert Precision of 0.9892 is achieved. As shown in Fig. 18, the Customized Multi-task CNN performed 97.72% species classification and 98.14% behaviour recognition accuracy. The complete RAID-Net pipeline, however, has a Hybrid Accuracy of 82.73%, which indicates the overall accuracy of the system at the object detection, ROI localization, classification, and decision-level alert processing stages. Thus, the standalone CNN results are not to be considered the end-to-end accuracy of RAID-Net.

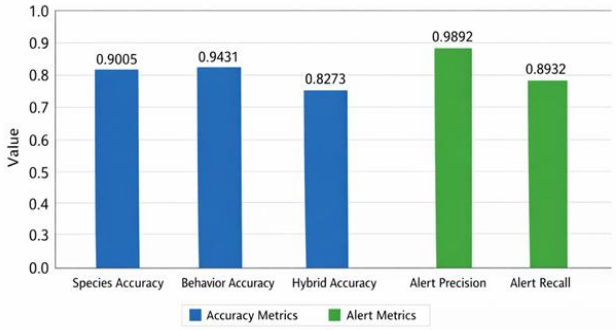


Fig. 18. Performance metrics of the RAID-NET.

Real-time performance is ensured in the system with an average value of 17.61ms and an average FPS of 56.79, suitable for live surveillance applications (Fig. 19).

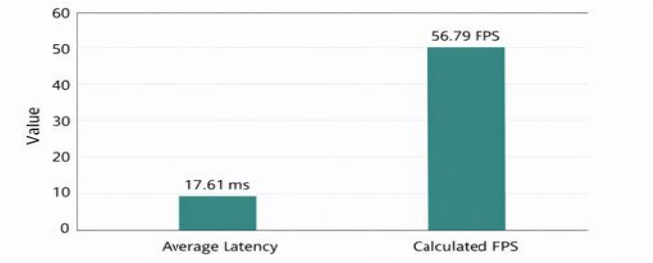


Fig. 19. Latency and FPS performance of the RAID-Net.

This visualization shows the overall performance of your custom framework, showcasing well-balanced high accuracy in hybrid, reliability in alerts, and computational latency.

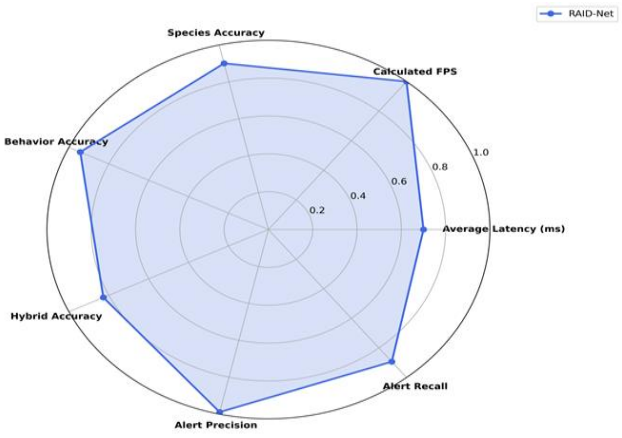


Fig. 20. RAID-Net performance metrics radar chart.

The radar chart shows the overall performance of the RAID-Net model, illustrated in Fig. 20. Species and behavior accuracy are high, and alert precision/recall and computational efficiency are also high.

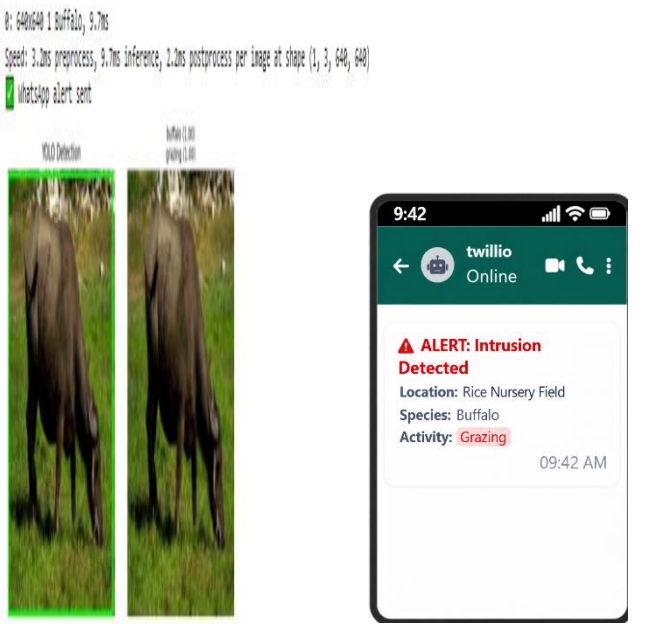
8) RAID-Net performance: The RAID-Net was tested in terms of performance and its feasibility in agricultural applications. The main performance characteristics of the system are summarized in Table IX. The model's Species Accuracy was found to be 90.05%, showing that it was able to identify herbivorous animals with a high level of accuracy, while the Behavior Accuracy was determined to be 94.31%, demonstrating the model's ability to accurately identify animal behavior. The overall Hybrid Accuracy was found to be 82.73%, as a result of the integration of species and behavior recognition

TABLE IX. PERFORMANCE METRICS OF THE RAID-NET

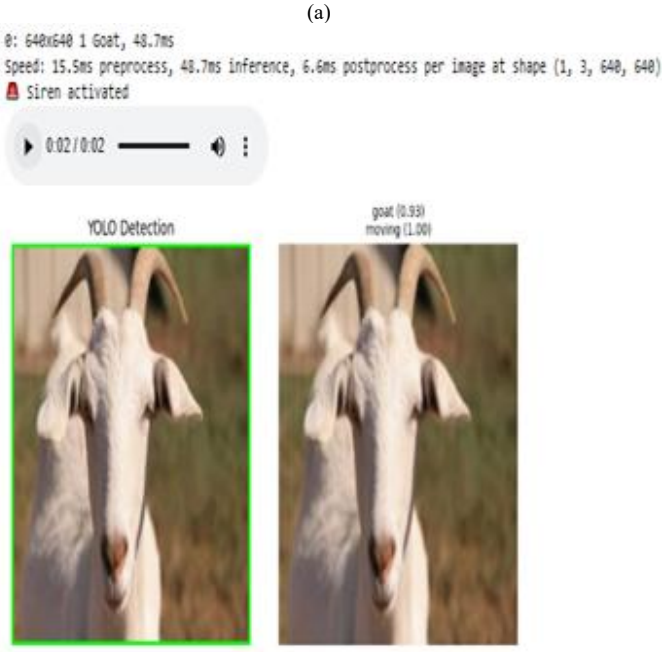
Metric	Value
Average Latency (ms)	17.61
Calculated FPS	56.79
Species Accuracy	0.9005
Behavior Accuracy	0.9431
Hybrid Accuracy	0.8273
Alert Precision	0.9892
Alert Recall	0.8932

These performance metrics are shown in Fig. 18. The accuracy-related metrics Species, Behavior, and Hybrid are represented in blue, while the alert-related metrics Precision and Recall are represented in green. The high Alert Precision of 98.92% indicates that the majority of alerts that the system issues are valid, while the Alert Recall of 89.32% indicates that the majority of relevant intrusion events are detected. The system can be used for real-time monitoring for suitable live agricultural applications with an average latency of 17.61ms and calculated FPS of 56.79.

9) Experimental evaluation of RAID-Net for grazing detection and moving animal tracking: Real-time detection and automated response mechanisms showcase the end-to-end capabilities of the proposed framework. The system successfully identified buffalo that are grazing with a high confidence score and also calculated the inference speed to achieve low-latency performance, as shown in Fig. 21(a). When such an intrusion is detected, the framework automatically sends an SMS to a mobile device through the Twilio API, sending a precise alert to the farm manager over WhatsApp, indicating the location, type of species, and what was seen on the farm. Moreover, the explanation of how the intrusion detection and audio warning system works is given practically in the rice nursery field in Fig. 21(b). In this case, the model detects a goat intrusion and triggers an audible alarm very quickly, which means an immediate physical deterrent to help prevent the crops from being damaged.



(a)



(b)

Fig. 21. (a) Real-time detection and automated WhatsApp alert, (b) Intrusion detection and audio warning system.

V. Ablation Study

To assess the contribution of each component in the RAID-NET, three different configurations were tested: YOLO only, CNN only, and RAID-Net. The summary of key performance metrics for the RAID-NET model is shown in Table VII. Results show the RAID-Net model is always superior to single-component models, giving the highest FPS with high accuracy in species recognition and behavior recognition. Specifically, the hybrid configuration shows that the integration of the detection system based on YOLO with multi-task CNN classification and severity modelling system achieves the Species Accuracy of 90.05%, Behavior Accuracy of 94.31%, and Hybrid Accuracy of 82.73%, respectively, showing efficient inference and reliable recognition.

VI. Limitation

Although the RAID-NET has great potential, some drawbacks still exist. First, the model has been trained and tested largely with daytime data, which may limit its performance at night. Secondly, the current system has no thermal imaging function that would help in detecting people in low-light or obstructed conditions. Lastly, the system has been assessed only in one field test, and the scalability and robustness of the system in various agricultural settings need to be proven. Future studies are needed to address these limitations to improve the versatility and reliability of the model for use in practical farming.

VII. Future Work

These are just some of the future enhancements that are planned for the RAID-NET, and they will help the system become more accurate, scalable, and adaptable. In conclusion, the RAID-NET is a promising solution that has the potential to revolutionize agricultural monitoring in the future. The creation of a vision of the environment and an assessment of the thresholds for confidence. Environmental visual-condition analysis and assessment of thresholds of confidence.

VIII. Conclusion

In this study, the hybrid deep learning-based real-time, behavior-aware herbivorous animal intrusion detection system in rice nursery fields, called RAID-Net, is proposed. The system can detect a variety of species and distinguish their actions by using YOLO-based detection, multi-task CNN classification, and severity modeling, which have high detection accuracy and can effectively reduce false alarms. The outcomes show that RAID-Net is a promising tool to improve the efficiency of agricultural surveillance, giving earlier alerts for crop protection, and reducing labour costs. Moreover, it is an extensible and edge-deployment-friendly framework and offers a smart and feasible protection solution for modern agriculture. Overall, RAID-Net is a strong artificial intelligence-based solution to protect rice nursery fields from herbivorous intrusion, which will help achieve sustainable and technology-driven agriculture.

References

[1] Çimen, M. E. (2025). Comparison of Deep Learning and YOLOv8 Models for Fox Detection Around the Henhouse. Journal of Smart Systems Research, 5(2), 76-90.

[2] Liu, Li, G., Sun, J., Guan, M., Sun, S., Shi, G., & Zhu, C. (2022). A New Method for non-destructive identification and Tracking of multi-object behaviors in beef cattle based on deep learning. Animals, 14(17), 2464.

[3] Chappidi, J., & Sundaram, D. M. (2024). Novel Animal Detection System: Cascaded YOLOv8 with Adaptive Preprocessing and Feature Extraction. IEEE Access.

[4] Othman, S., Mavani, N. R., Hussain, M. A., Abd Rahman, N., & Ali, J. M. (2023). Artificial intelligence-based techniques for adulteration and defect detection in the food and agricultural industry: A review. Journal of Agriculture and Food Research, 100590.

[5] Albahar, M. (2023). A Survey on Deep Learning and Its Impact on Agriculture: Challenges and Opportunities. Agriculture, 13(3), 540.

[6] Mesías-Ruiz, G. A., Pérez-Ortiz, M., Dorado, J., de Castro, A. I., & Peña, J. M. (2023). Boosting precision crop protection towards agriculture 5.0

via machine learning and emerging technologies: A contextual review. Frontiers in Plant Science, 14.

[7] Dracatos, P. M., Lück, S., & Douchkov, D. K. (2023). Diversifying resistance mechanisms in cereal crops using microphenomics. Plant Phenomics, 5, 0023.

[8] Ezugwu, A. E., Oyelade, O. N., Ikotun, A. M., Agushaka, J. O., & Ho, Y. S. (2023). Machine Learning Research Trends in Africa: A 30-Year Overview with Bibliometric Analysis Review. Archives of Computational Methods in Engineering, 1-31.

[9] Vidhyalatha, T., Sreeram, Y., & Purushotham, E. (2022). Animal Intrusion Detection using Deep Learning and Transfer Learning Approaches. International Journal of Human Computations & Intelligence, 1(4), 51-60.

[10] Ukwuoma, C. C., Qin, Z., Yussif, S. B., Happy, M. N., Nneji, G. U., Urama, G. C., ... & Agobah, H. (2022). Animal species detection and classification framework based on modified multi-scale attention mechanism and feature pyramid network. Scientific African, 16, e01151.

[11] Gowda, T., Sahana, R., Raksha, H., Bindushree, R. A Review on Animal Encroachment Detection in Crop Fields Using Deep Learning and IoT.

[12] Sandeep, T., Manushree, B., Rahul, S., & Bharath, T. (2022). Crop Protection from animals using CNN.

[13] Fu.Wang, Zhang, C., Xu, P., & Ruan, W. (2022). Deep learning and its adversarial robustness: A brief introduction. In Handbook On Computer Learning And Intelligence: Volume 2: Deep Learning, Intelligent Control and Evolutionary Computation (pp. 547-584).

[14] Bochkovskiy, Alexey, Chien-Yao Wang, and Hong-Yuan Mark Liao. "Yolov4: Optimal speed and accuracy of object detection." arXiv preprint arXiv:2004.10934 (2020).

[15] Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." arXiv preprint arXiv:2010.11929 (2020).

[16] Tan, Mingxing, Ruoming Pang, and Quoc V. Le. "Efficientdet: Scalable and efficient object detection." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020.

[17] Lin, Tsung-Yi, et al. "Focal loss for dense object detection (pp. 1)." IEEE transactions on pattern analysis and machine intelligence (2018).

[18] Dai, Jifeng, et al. "Deformable convolutional networks." Proceedings of the IEEE international conference on computer vision. 2017.

[19] Gu, H., Wang, Y., Hong, S., & Gui, G. (2019). Blind channel identification aided generalized automatic modulation recognition based on deep learning. IEEE Access, 7, 110722-110729.

[20] Jhatial, M. J., Shaikh, R. A., Shaikh, N. A., Rajper, S., Arain, R. H., Chandio, G. H., & Shaikh, K. H. (2022). Deep Learning-Based Rice Leaf Diseases Detection Using Yolov5. Sukkur IBA Journal of Computing and Mathematical Sciences, 6(1), 49-61.

[21] Zhang, X., Zhou, X., M., & Sun, J. (2018). Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 6848-6856)

[22] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 2961-2969).

[23] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).

[24] Dai, J., Li, Y., He, K., & Sun, J. (2016). R-fcn: Object detection via region-based fully convolutional networks. Advances in neural information processing systems, 29.

[25] Sana, S., Gudla, S., Kandi, H., Mishra, P., Sambana, B., Hameed, S. S., & Ramesh, Y. Amalgam enhanced to Computer Vision-Deep Learning and Internet of Things for Aerial View Cultivating Monitoring. MMTC Communications-Frontiers.