Edge-AI Based Smart Pet Monitoring Framework: A Rabbit Case Study

Vinit Masale[1], Suyog Mamankar[2], Prathamesh Sawant[3], Mhaboob Ali[4], Prof. Jyoti Shrote[5]
[1,2,3,4]*School of Information Technology, Indira University, Pune*
[5]*Professor, School of Information Technology, Indira University, Pune*
doi.org/10.64643/ijirt.208534-459

***Abstract*—Continuous Monitoring of animals is hard when the people who take care of them are not around. Changes in how they are eaten, how much they drink, the way they stand and what they do during the day can show if their health is changing or if their normal routine is different. This paper talks about an Edge AI-based Smart Pet Monitoring Framework. The goal is watching the pet's behaviour and the environment around them in time. The system keeps privacy in mind. It uses computer vision, IoT, sensors and computing that happens close to the data not in the cloud This reduces the need for internet access. A Raspberry Pi serves as the edge device running a Python and OpenCV image processing chain that uses a YOLOv9 detector to spot feeding, drinking, posture and activity states. Temperature, humidity, ultrasonic distance and water-level sensors give environment data while an Arduino control layer handles alerts using an LCD, buzzer, LEDs and a servo. Local inference gives real-time processing and improves privacy, data ownership and response time. We tested than 200 images in a controlled setting and found it works but the test is still early. This framework offers a flexible base for smart pet monitoring, which could help at home in shelters and, for vets once more tests are done.**

***Index Terms*—Edge AI, IoT, Smart Pet Monitoring, Animal Behaviour Analysis, Computer Vision, YOLOv9, Raspberry Pi, Multimodal Sensing, Privacy-Aware Monitoring.**

I. INTRODUCTION

Many people now treat pets like cats, dogs, rabbits and birds as family members. Because we love our pets, we want to watch them. People who watch their pets every day can spot signs of trouble like when a pet is sick or not drinking enough water. However, it is very hard for busy families to watch their pets every single minute. This is where new technology helps. Using artificial intelligence and cheap sensors gives us a great way to help. For example, smart programs like YOLO (You Only Look Once) can watch video to see what an animal is doing. At the time small sensors can check things, like the temperature or if the water bowl is full. Embedding these capabilities directly in the home, without dependence on a remote server, allows always-on monitoring that keeps sensitive data local, which motivates an edge-computing rather than a cloud-computing approach to pet monitoring.

Existing approaches, reviewed in Section II, tend to address either vision-based behaviour recognition or environmental sensing in isolation, and rarely combine the two into a single context-aware system. This paper proposes an edge-computing framework for smart pet monitoring that unites both streams and demonstrates it using a domestic rabbit as a case study. A Raspberry Pi running a YOLOv9 detector observes the animal through a camera and infers its behaviour locally, while an Arduino-based module measures temperature, humidity, and water level and drives local alert devices. The framework combines these two data streams and performs all video processing on-device, resulting in a system that is low-cost, privacy-preserving, and context-aware.

The remainder of the paper is organised as follows. Section II reviews the relevant literature across four related areas. Section III presents the proposed system and its architecture. Section IV describes the implementation and working of the framework. Section V reports the results obtained and concludes the paper, and Section VI outlines the future scope of the work.

II. LITERATURE REVIEW

A. Vision-Based Animal Behaviour Monitoring

Vision-based recognition of animal behaviour is a well-established research direction. Chen et al. used YOLOv3-Tiny on a Raspberry Pi to recognise pet-cat behaviours such as sleeping and eating from camera video alone [1], while Alameer et al. showed that a similar camera-based approach can reliably classify pig postures and drinking behaviour relevant to animal health [3]. Kim and Moon found that fusing camera detections with a wearable accelerometer improved dog-behaviour recognition over either source alone, though this required the animal to wear a device [2]. A broader review by Rajagukguk et al. confirms that YOLO- and CNN-LSTM-based detectors dominate this field, but that multimodal fusion remains an open challenge [7]. Overall, vision-based recognition is mature, but most systems still treat behaviour as a single-sensor problem, disconnected from the animal's environment.

B. Edge AI, YOLO, and Raspberry Pi-Based Monitoring

A second strand of literature addresses deploying detection models on constrained edge hardware. Lu and Wang's MAR-YOLOv9 improved on base YOLOv9 accuracy while reducing model size, confirming a favourable accuracy-to-cost balance for edge use [8]. Abdulhaq and Ahmed showed that lightweight detectors can reach real-time performance on Raspberry Pi-class embedded platforms once optimised [5]. In wildlife monitoring, Campbell described edge-AI systems performing on-site inference on Raspberry Pi, Jetson Nano, and Coral TPU hardware, reducing latency, bandwidth use, and exposure of sensitive visual data compared with cloud processing [4]. Together, these studies confirm that YOLOv9-class detectors suit Raspberry Pi hardware, and that on-device inference is an established way to reduce both latency and privacy risk.

C. IoT-Based Environmental and Pet Monitoring

A parallel body of work has examined IoT-based sensing for pet care and animal environments. Reviewing thirty-six studies from 2010 to 2024, Pico-Valencia and Holgado-Terriza found that the “Internet of Pets” literature is dominated by remote-feeding prototypes, with health monitoring and continuous surveillance comparatively underexplored [11]. In the livestock domain, Zhao et al. observed that wearable, imaging, and environmental sensors are usually deployed and reported in isolation, each capturing only a partial view of an animal's condition [10]. These findings suggest that IoT-based environmental sensing is technically mature but rarely integrated with camera-based behaviour recognition, leaving room for a combined low-cost platform such as the one proposed here.

D. Multimodal Sensing, Context-Aware Monitoring, and Privacy

The final strand of literature concerns combining multiple data sources for context-aware interpretation, along with the resulting privacy implications. Sabolek and Jović found that AI-based behavioural monitoring in companion-animal care is active but remains fragmented and poorly integrated into routine use [6]. Focusing on rabbits, Abdel-Wareth et al. identified pregnancy monitoring, health assessment, behavioural tracking, and environmental control as complementary streams that together support more proactive management than any single stream [9]. Zhao et al. similarly argue that multimodal fusion enables earlier, more reliable welfare detection than a single modality [10], while Campbell's account of edge-AI wildlife monitoring reinforces that on-device inference limits exposure of sensitive visual data [4]. This literature motivates fusing vision-based behaviour detection with environmental sensor data locally, as pursued here.

III. SYSTEM ARCHITECTURE

The proposed system consists of two cooperating layers, shown in Fig. 1: an Edge-AI vision layer on a Raspberry Pi that detects behaviour from camera video, and an Arduino-based sensor/control layer that monitors the immediate environment. A camera continuously captures the enclosure, and the Raspberry Pi runs Python and OpenCV to process each frame with a YOLOv9 detector that locates the rabbit together with reference objects such as the food bowl and water source. At the same time, the Arduino reads a DHT11 temperature/humidity sensor and an HC-SR04 ultrasonic sensor for water-level data and transmits these readings to the Raspberry Pi over a serial connection.

The two data streams are combined in a multimodal, context-aware analysis stage that produces an event or alert decision, output locally through an LCD display, a buzzer, and LEDs, without raw video or personal data leaving the device. Splitting the system this way allows the computationally demanding vision task to run on the Raspberry Pi while the lightweight, time-critical tasks of sensor polling and indicator control run on the Arduino, so that each device is sized appropriately for its role.

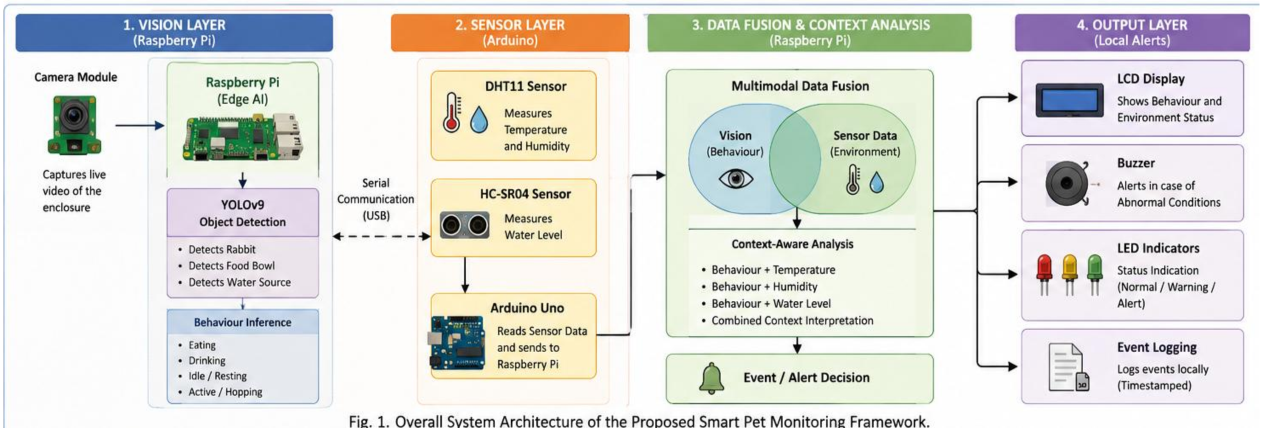

Fig. 1. Overall system architecture of the proposed Edge-AI smart pet monitoring framework

A. Hardware Architecture

Fig. 2 details the hardware architecture and physical layout within the rabbit enclosure. The camera is mounted to maintain a clear view of the enclosure so that the food bowl and water container fall within its field of view, and the HC-SR04 sensor is mounted above the water container to estimate its fill level. The DHT11 sensor and the Arduino board are positioned along the enclosure wall, with the Arduino driving the LCD, LED, and buzzer indicators.

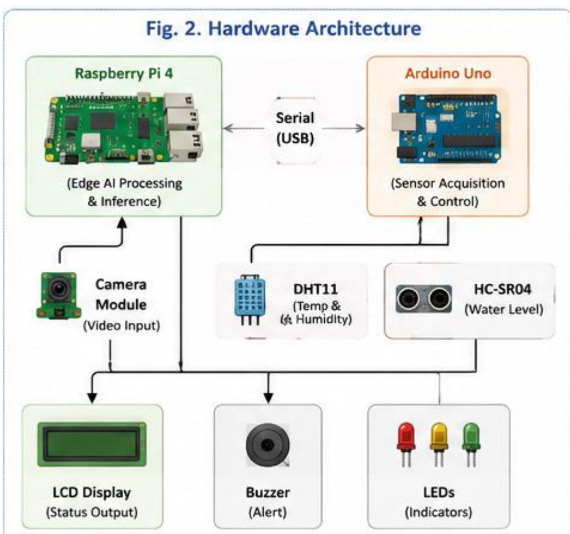

Fig. 2. Hardware architecture showing the Raspberry Pi, Arduino, camera, sensors, and local output devices.

The Arduino communicates sensor readings to the Raspberry Pi, positioned outside the enclosure, over a serial (USB) connection, while the camera supplies its video feed to the Raspberry Pi directly. This division keeps the Raspberry Pi dedicated to edge-AI processing and inference, while the Arduino handles sensor acquisition and local control.

B. Edge-AI Software Architecture and Processing Pipeline

The software pipeline, summarised in Fig. 3, proceeds from frame acquisition through preprocessing, YOLOv9-based object detection, and behaviour inference, before sensor data acquisition and multimodal fusion produce a context-aware decision and alert.

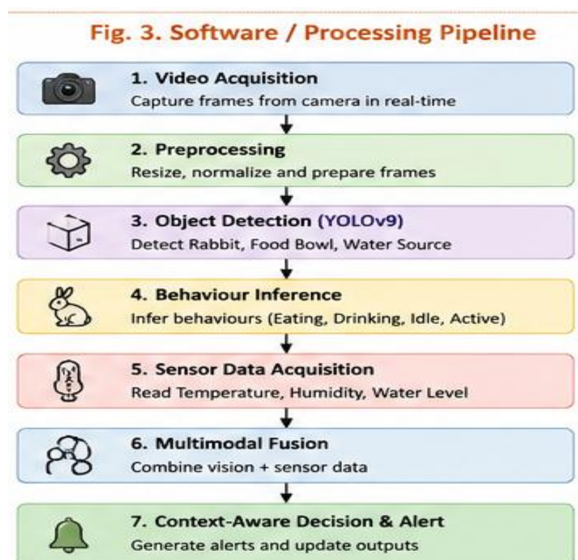

Fig. 3. Edge-AI processing pipeline for video acquisition, YOLOv9 detection, and behavioural inference.

YOLOv9, a one-stage detector, was selected for this pipeline because of its favourable accuracy-to-speed trade-off relative to earlier YOLO variants, which makes it suitable for embedded deployment on a Raspberry Pi [8]. The pipeline is designed to repeat this sequence on each incoming frame, so that behaviour inference and sensor acquisition proceed in a continuous cycle.

C. Multimodal Data Fusion and Context-Aware Decision Process

The framework's principal contribution is contextual interpretation: a detected behaviour is escalated to an event or alert only after being compared with the current sensor reading, as illustrated in Fig. 4. For example, the rabbit being detected near the food bowl together with a confirmed food supply is logged as a feeding event, whereas a prolonged period of inactivity combined with a rise in ambient temperature is logged as a possible heat-stress alert rather than a diagnosis. This combination reduces false alarms that a single modality would otherwise produce for instance, misreading a quiet drinking pause as distress. All visual inference takes place on the Raspberry Pi, and only high-level events are logged locally or, optionally, forwarded to a caregiver application, consistent with the edge-AI principle of keeping sensitive data on-device without requiring continuous internet connectivity.

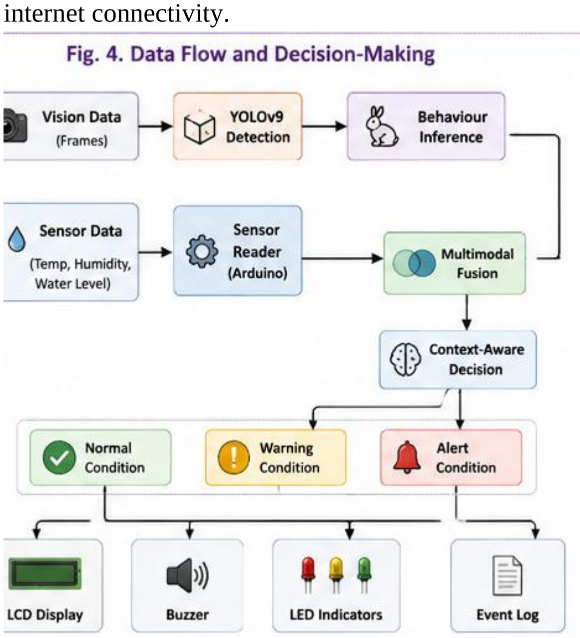

Fig. 4. Data flow and context-aware decision-making process.

IV. IMPLEMENTATION AND WORKING

A. Video Acquisition and Preprocessing

Video frames are captured continuously from the camera module and passed to the Raspberry Pi, where OpenCV is used to resize and normalise each frame before it is submitted to the detector. This preprocessing step keeps the input format consistent with what YOLOv9 expects and reduces the computational cost of subsequent inference on the constrained hardware.

B. YOLOv9-Based Object Detection

Each pre-processed frame is passed through the YOLOv9 detector, which localises the rabbit together with reference objects such as the food bowl and water source. YOLOv9 was chosen for this stage because of its accuracy-to-speed trade-off relative to earlier YOLO variants, as noted in Section III-C, which makes it appropriate for real-time use on a Raspberry Pi [8].

C. Behaviour Inference

A spatial-rule stage interprets the bounding-box positions and proximities produced by the detector to classify a candidate behaviour, such as eating, drinking, resting, or locomotion. This candidate label forms the vision-side input to the multimodal fusion stage described below.

D. Environmental Sensor Acquisition

In parallel with the vision pipeline, the Arduino continuously reads the DHT11 sensor for temperature and humidity and the HC-SR04 sensor for water level, and transmits these readings to the Raspberry Pi over the serial connection described in Section III-B. This acquisition runs independently of the vision pipeline, so that sensor readings remain current regardless of the state of the detection cycle.

E. Multimodal Fusion

The candidate behaviour label and the current sensor readings are combined in the multimodal fusion stage. As described in Section III-D, a detected behaviour is only escalated to an event or alert once it has been checked against the corresponding sensor context — for example, cross-referencing a detected feeding behaviour with the food bowl's status, or cross-referencing prolonged inactivity with ambient temperature. This step is what distinguishes the framework from vision-only or sensor-only monitoring approaches.

F. Local Alert Generation and Event Logging

Once the context-aware decision stage produces an event or alert, the outcome is communicated locally through the LCD display, buzzer, and LED indicators, and the event can be logged on-device with a timestamp for later review. Because inference and decision-making are performed entirely on the Raspberry Pi, no raw video or personal data needs to leave the device; only high-level event information is retained or optionally shared with a caregiver application.

V. RESULTS AND CONCLUSION

The proposed smart pet-monitoring framework was qualitatively assessed with the help of images of rabbits and sensor readings. The vision module, based on YOLOv9, was able to detect the rabbit, the food bowl, and the water source. The representative detection output is shown in Fig. 5.

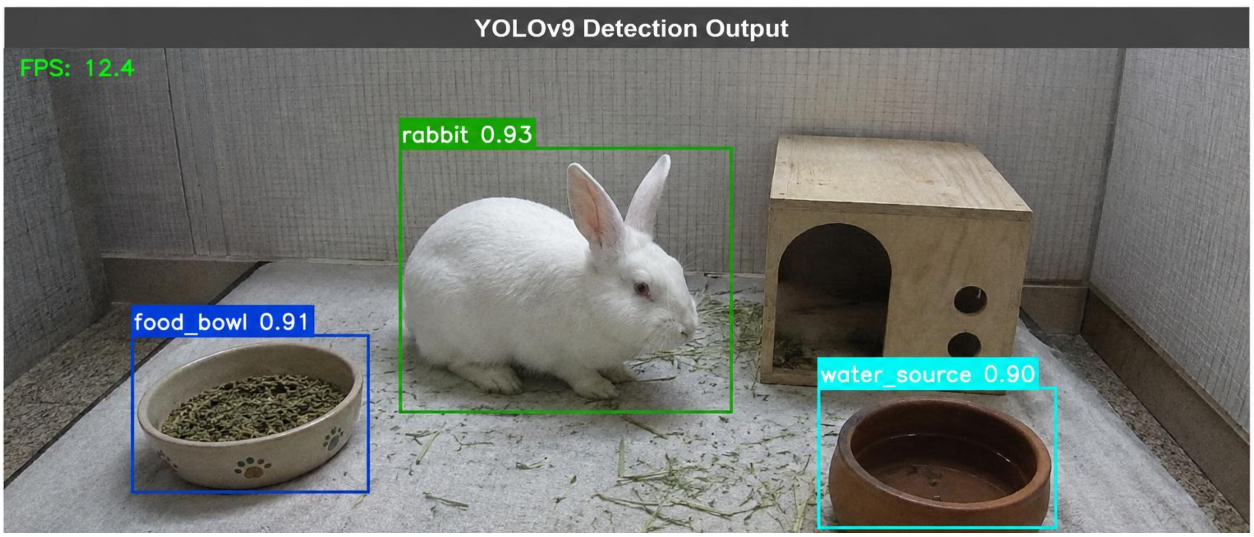

Fig. 5. YOLOv9-based detection of the rabbit and reference objects.

The sensing layer, which was based on the Arduino microcontroller, was able to accurately measure the temperature, humidity, and water level with the as feeding and resting, as illustrated in Fig. 6. DHT11 and HC-SR04 sensors, respectively, and send the data to the Raspberry Pi.

The behaviour was inferred by the spatial rules, such

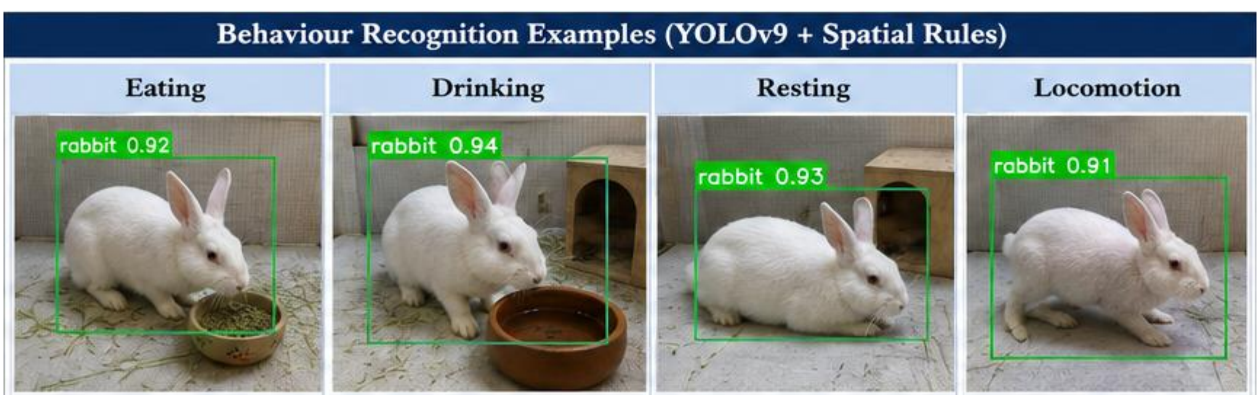

Fig. 6. Representative behavioural states identified using spatial-rule inference.

The multimodal fusion of visual and sensor data resulted in context-aware events, which were used to trigger local LCD, buzzer and LED alerts as desired. The corresponding alert output is shown in Fig. 7.

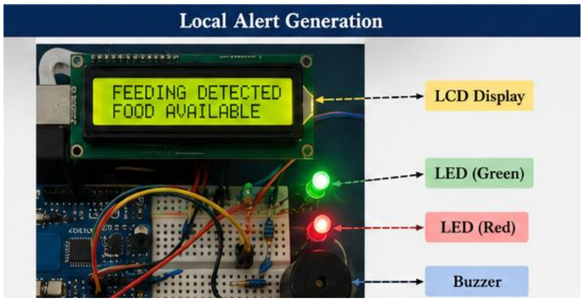

Fig. 7. Alert output using LCD, LEDs, and buzzer.

The study overall shows that it is possible to integrate Edge AI and IoT-based sensing to monitor companion animals in a low-cost and privacy-preserving manner. By processing video locally on the Raspberry Pi, sensitive data is not sent outside the home. The current evaluation is qualitative and based on a limited dataset, but the framework offers a solid basis for future quantitative testing, larger and more diverse datasets, better behaviour recognition, and real-world field deployment.

VI. FUTURE SCOPE

The proposed approach can be further extended by building a bigger and more diversified data set of rabbits to further increase the accuracy of detection and to evaluate the performance of the proposed approach. For future work, the development of advanced behaviour recognition in the framework of pose estimation and sequence-based models, such as CNN-LSTM, can be considered. Raspberry PI devices can be used with edge optimisation techniques such as quantization and pruning to achieve real time performance and power efficiency. Precise sensors and sensor-fusion can be employed to enhance sensor integration. Can be expanded to low light monitoring, mobile/companion web notifications. Lastly, reliability, false alarms, and the potential of it as a non-diagnostic animal monitoring and decision support system can be assessed in real world field deployments and via caregiver and/or veterinary evaluation.

REFERENCES

[1] R.-C. Chen, V. S. Saravanarajan, and H.-T. Hung, “Monitoring the behaviours of pet cat based on YOLO model and Raspberry Pi,” *Int. J. Appl. Sci. Eng.*, vol. 18, no. 5, pp. 1–12, Sep. 2021. Article link

[2] J. Kim and N. Moon, “Dog behavior recognition based on multimodal data from a camera and wearable device,” Appl. Sci., vol. 12, no. 6, Art. no. 3199, 2022, doi: 10.3390/app12063199. DOI link

[3] A. Alameer, I. Kyriazakis, and J. Bacardit, “Automated recognition of postures and drinking behaviour for the detection of compromised health in pigs,” Sci. Rep., vol. 10, Art. no. 13665, Aug. 2020, doi: 10.1038/s41598-020-70688-6. DOI link

[4] L. Campbell, “Building edge-AI wildlife monitoring systems with real-time animal detection and classification using computer vision,” preprint, 2025. ResearchGate link

[5] K. Abdulhaq and A. A. Ahmed, “Real-time object detection and recognition in embedded systems using open-source computer vision frameworks,” Int. J. Electr. Eng. Sustainability, vol. 3, no. 1, pp. 103–118, Mar. 2025, doi: 10.65998/ijees. v3i1.113. Article link

[6] S. Devikar, M. Hinge, and S. S. Shevkari, “Human vs. machine: A review of AI’s impact on language learning interaction or replacing human interaction,” International Journal for Multidisciplinary Research, vol. 7, no. 3, 2025, doi: 10.36948/ijfmr. 2025.v07i03.49459.

[7] R. A. Rajagukguk et al., “Deep learning for visual animal monitoring (detection, tracking, pose estimation, and behavior classification): A comprehensive review,” Smart Agric. Technol., vol. 12, Art. no. 101539, Oct. 2025, doi: 10.1016/j.atech.2025.101539. DOI link

[8] D. Lu and Y. Wang, “MAR-YOLOv9: A multi-dataset object detection method for agricultural fields based on YOLOv9,” PLOS ONE, vol. 19, no. 10, Art. no. e0307643, Oct. 2024, doi: 10.1371/journal.pone.0307643. DOI link

[9] A. A. A. Abdel-Wareth, A. A. Ahmed, M. Salahuddin, and J. Lohakare, “Application of artificial intelligence in rabbit husbandry: From reproductive monitoring to precision farming,” Front. Vet. Sci., vol. 12, Art. no. 1679630, Oct. 2025, doi: 10.3389/fvets.2025.1679630. DOI link

[10] P. Bhamare, S. Shevkari, and P. Mane, “How machine learning with Python and artificial intelligence benefits Formula 1 teams,” International Journal of Innovative Research in Technology, vol. 12, no. 6, 2025.

[11] P. Pico-Valencia and J. A. Holgado-Terriza, “The Internet of Things empowering the Internet of Pets—An outlook from the academic and scientific experience,” Appl. Sci., vol. 15, no. 4, Art. no. 1722, Feb. 2025, doi: 10.3390/app15041722. DOI link