Proceedings of the Third International Conference on Automation, Computing and Renewable Systems (ICACRS-2024)
IEEE Xplore Part Number: CFP24CB5-ART; ISBN: 979-8-3315-3242-0

2024 3rd International Conference on Automation, Computing and Renewable Systems (ICACRS) | 979-8-3315-3242-0/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICACRS62842.2024.10841601

WildGuard: Automated Barrier System for Animal Attack Prevention

S Ariffa Begum
Department of Computer Science and Engineering
Kalasalingam Academy of Research and Education, Krishnan Koil, Taminadu, India
s.ariffabegam@klu.ac.in

N.Subbulakshmi
Department of Computer Science and Engineering
Kalasalingam Academy of Research and Education, Krishnan Koil, Taminadu, India
n.subbulakshmi@klu.ac.in

A Balakrishna Reddy
Department of Computer Science and Engineering
Kalasalingam Academy of Research and Education, Krishnan Koil, Taminadu, India
9921004836@klu.ac.in

Battu Aravind Kumar
Department of Computer Science and Engineering
Kalasalingam Academy of Research and Education, Krishnan Koil, Taminadu, India
9921004840@klu.ac.in

B Sandeep Kumar Reddy
Department of Computer Science and Engineering
Kalasalingam Academy of Research and Education, Krishnan Koil, Taminadu, India
99210041145@klu.ac.in

B Manikanta Reddy
Department of Computer Science and Engineering
Kalasalingam Academy of Research and Education, Krishnan Koil, Taminadu, India
9921004082@klu.ac.in

***Abstract*— In areas that are close to forests, human-wildlife conflict poses serious threats to both human life and property. In order to detect and categorise animals in real-time CCTV camera video feeds, this study presents an intelligent system that makes use of deep learning techniques, particularly Convolutional Neural Networks (CNNs). Using Internet of Things components, the system classifies animals as either harmful or non-threatening and initiates automated reactions for the latter. A red warning sign is triggered in the village, a sound-based trap is set up to draw the animal within the camera's field of view, and notifications are sent to the appropriate authorities when a dangerous animal is spotted. By improving safety by early detection and real-time response, this proactive system seeks to safeguard communities that are close to woods.**

Keywords— Animal Detection, Convolutional Neural Networks, Deep Learning, IoT, Real-time Surveillance, Wildlife Management, Human-Wildlife Conflict, Dangerous Animal Classification, CCTV, Sound-based Trapping.

I. INTRODUCTION

Human-wildlife conflict is a growing concern in many parts of the world, particularly in areas located near forests and wildlife reserves. As human populations expand into natural habitats, the frequency of encounters with wild animals increases, leading to dangerous situations that threaten human lives, livestock, and property. Traditional methods for addressing these conflicts, such as physical barriers and reactive interventions, have proven inadequate in many instances, as they either fail to prevent encounters or respond too late to be effective. There is an urgent need for proactive solutions that can detect the presence of wild animals in real time and take immediate action to protect both human and animal lives.

This paper presents an innovative approach to managing human-wildlife conflict using a combination of deep learning techniques and Internet of Things (IoT) technologies. The core of the system is a real-time animal detection module that utilizes Convolutional Neural Networks (CNNs) to process video feeds captured by CCTV cameras. These cameras are strategically positioned in areas prone to animal intrusions, such as near villages or along forest boundaries. The CNN-based model is trained to identify various animal species and classify them as either dangerous or non-dangerous based on predefined criteria. Dangerous animals—such as lions, tigers, and bears—trigger an immediate response from the system.

When a dangerous animal is detected, the system activates several response mechanisms. First, an alert is sent to the relevant authorities, such as local wildlife officials or government agencies responsible for public safety. Simultaneously, a red warning signal is triggered in nearby villages to alert residents of the potential danger. Additionally, the system deploys a sound-based trap within the camera's range, designed to attract the animal to a specific location where it can be safely contained, preventing it from entering populated areas.

The integration of CNNs for detection and IoT components for alerting and trapping makes this system highly efficient and scalable. Unlike traditional wildlife management solutions, which often rely on manual monitoring and delayed responses, this approach provides real-time, automated detection and response. By implementing this system, communities living near forests can significantly reduce the risk of dangerous animal encounters, improving safety for both humans and wildlife.

Authorized licensed use limited to: Università di Bari Aldo Moro. Downloaded on October 28,2025 at 00:13:21 UTC from IEEE Xplore. Restrictions apply.

This project aims to contribute to the field of wildlife management by offering a technologically advanced solution to an age-old problem. The real-time detection system, coupled with automatic alerting and containment features, provides a practical and effective approach to mitigating human-wildlife conflict. Through this system, we hope to create safer environments for people living in wildlife-prone areas and pave the way for future developments in smart wildlife monitoring.

II. LITERATURE REVIEW

This foundational work introduced a deep convolutional neural network architecture, significantly advancing the field of image classification using large datasets and powerful GPUs. While it achieved state-of-the-art performance, it does not address the challenges of applying the model in real-time animal detection scenarios, particularly in outdoor environments where variable lighting and occlusions are prevalent [1]. This model offers a groundbreaking approach to object detection, enabling real-time processing by treating detection as a single regression problem. It allows for rapid identification of multiple objects within a frame, making it suitable for monitoring wildlife. However, it lacks exploration of the model's performance in adverse conditions, such as nighttime or inclement weather, which are crucial for effective wildlife monitoring [2]. This paper introduces an architecture that enhances CNN capabilities by allowing for deeper network structures and efficient computations. While it improves classification accuracy, it does not account for the practical application of these networks in real-time wildlife detection, particularly concerning the integration of IoT systems for rapid alerts and response mechanisms [3]. This study explores various machine learning algorithms for detecting wildlife through video surveillance systems. The findings suggest promising applications for automatic wildlife monitoring, but it does not investigate the incorporation of IoT technologies to create a comprehensive system that includes alerting mechanisms and effective response strategies for dangerous wildlife encounters [4].

This research focuses on utilizing very deep CNNs to classify animal species in images captured by camera traps, providing a solid foundation for automatic wildlife monitoring. However, it primarily targets static images and does not explore real-time detection capabilities or immediate human safety measures in response to detected threats [5]. This study successfully demonstrates the application of deep learning techniques for real-time animal detection and classification from video feeds. While the results are promising, it fails to address the limitations posed by varying environmental conditions, such as poor lighting or extreme weather, which could hinder detection accuracy [6]. This paper presents an IoT-based monitoring system that employs machine learning to identify potential threats from wildlife near human settlements. While innovative, it is restricted to motion detection, lacking advanced classification capabilities that would enable more nuanced responses to different types of wildlife [7].

This research discusses a system that integrates IoT technology with deep learning for detecting animal intrusions, highlighting the potential for timely alerts. However, it does not explore the implementation of sound traps or mechanisms for effectively managing multiple simultaneous animal detections, which is critical for ensuring human safety [8]. This study emphasizes a framework for tracking wildlife in forested areas using deep learning algorithms and IoT technologies. However, the scalability of this system in larger ecosystems remains unaddressed, raising questions about its practicality for widespread wildlife monitoring [9]. This paper discusses a comprehensive IoT-enabled system for monitoring wildlife and analyzing movement patterns using machine learning. While it offers significant insights into wildlife conservation, it does not provide in-depth exploration of humane methods, such as sound-based traps, to manage dangerous animals effectively [10].

This study addresses several challenges faced when implementing machine learning for wildlife conservation, including issues related to data availability and model adaptability. While it provides valuable insights, the lack of robust datasets specific to real-time wildlife monitoring limits its applicability in field conditions [11]. This research presents advanced techniques for detecting multiple wildlife species simultaneously, demonstrating improved accuracy in varied environments. However, it does not discuss the integration of IoT technologies necessary for real-time alert systems, which are crucial for human-wildlife conflict mitigation [12].

This study highlights the use of deep learning for analyzing animal behavior from video feeds, providing critical insights for wildlife management. Nonetheless, it does not address the complexities involved in managing multiple detections and the required response strategies [13]. This paper discusses the integration of machine learning and IoT for monitoring wildlife populations. While it presents a useful framework, practical applications in diverse ecological settings remain underexplored, limiting its direct applicability in fieldwork [14]. This research proposes a smart system utilizing CNNs for animal detection combined with IoT-driven alarm mechanisms. However, the scalability of this system across larger forest areas and its effectiveness in diverse ecological conditions require further examination [15].

This comprehensive review outlines various deep learning techniques for object detection, detailing their applications across different domains. Despite its breadth, the paper does not specifically address the challenges of applying these techniques to wildlife detection in rural settings, which is essential for the proposed project [16]. This paper showcases a tailored deep learning method for detecting wildlife in forested regions, highlighting its promising results. However, the implementation of such systems in real-time and at a larger scale remains inadequately addressed, posing challenges for practical deployment [17]. This research introduces a smart wildlife monitoring system that utilizes IoT and deep learning, contributing valuable insights for future research. However, it lacks thorough examination of potential data privacy issues inherent in monitoring wildlife, which is critical for ethical wildlife management practices [18].

This research introduces a real-time tracking solution combining deep learning and IoT, showcasing innovative approaches to wildlife monitoring. However, it does not address potential challenges such as signal loss or false positives that can occur in dense forest environments, which are vital for effective tracking [19]. This paper discusses the application of IoT technologies for monitoring animal behavior using machine learning techniques. While the approach is promising, it does not outline how this data can be effectively integrated into decision-making processes for wildlife management, which is crucial for addressing human-wildlife conflicts [20].This

Authorized licensed use limited to: Università di Bari Aldo Moro. Downloaded on October 28,2025 at 00:13:21 UTC from IEEE Xplore. Restrictions apply.

study highlights various challenges in implementing real-time wildlife detection systems, providing suggestions for improvements. However, it lacks specific case studies demonstrating successful implementation, which could help validate the proposed solutions [21].

This review article explores opportunities and challenges in applying deep learning for wildlife conservation, offering a thorough understanding of the field. However, it falls short of providing concrete methodologies for implementation in real-world field studies, limiting its practical utility [22]. This paper proposes an IoT-based monitoring system aimed at enhancing conservation efforts, detailing its architecture and functionalities. Yet, it does not consider potential environmental impacts or the long-term sustainability of such systems, raising concerns for wildlife management practices [23].

This study discusses the application of hybrid deep learning techniques to improve wildlife detection systems, showing promising results. However, the research lacks a comprehensive consideration of the operational challenges faced in diverse ecosystems, which are critical for effective system deployment [24]. This paper analyzes the role of machine learning in wildlife conservation, presenting both opportunities and challenges. However, the practical applications and effectiveness of the proposed solutions in real-world settings are not thoroughly discussed, necessitating further research [25].

III. METHODOLOGY

The development of the animal detection and alert system involved several steps, ranging from data preparation and augmentation to the design and implementation of a deep learning-based detection model.

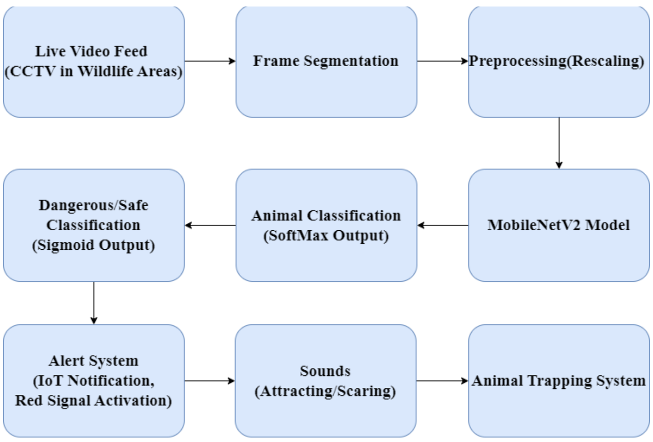

Figure 1:Overall Process Flow Diagram

This section outlines the key phases of the methodology, including dataset creation, augmentation techniques, model architecture, and the final implementation of the real-time alert and trapping system. The figure 1 shows the overall process flow diagram of this work.

A. Dataset

The dataset for this project consists of 90 folders, each representing a unique animal species, and contains 60 images per folder. This dataset was manually curated from publicly available wildlife databases and image repositories. The species are categorized into two broad classes: dangerous animals (e.g., lions, tigers, snakes) and safe animals (e.g., deer, squirrels, rabbits). Each image was preprocessed to fit the MobileNetV2 model's input requirements (224x224 resolution, RGB format).

B. Data Augmentation

To ensure that the model generalizes well across different environments and scenarios, extensive data augmentation techniques were applied to the dataset. The augmentation pipeline used for this project includes the following transformations:

Rotation: Each image was randomly rotated by angles between 0 and 360 degrees to simulate different orientations of animals.

Shearing: A shear range of 0.1 was applied to create minor distortions, mimicking camera perspective changes.

Zooming: A zoom range of 10% was used to create variations in scale, helping the model detect animals at various distances.

Shifting: Both horizontal and vertical shifts (10%) were applied to simulate slight camera movements or animal movements across frames.

Horizontal Flipping: This transformation was used to horizontally flip images, ensuring the model's robustness to mirrored images.

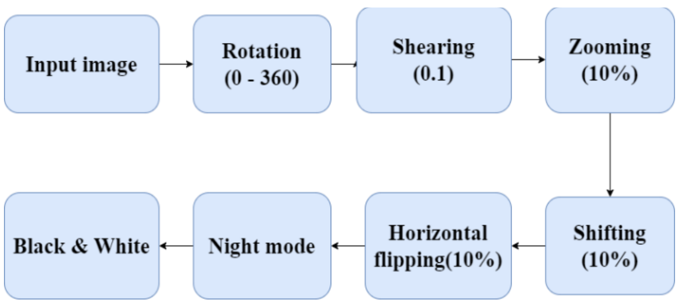

Figure 2:Data Augmentation Process

Night Mode Simulation: A night vision effect was simulated using a greenish tint and reduction in brightness to mimic CCTV footage captured at night. This transformation was applied to a subset of the dataset.

Black-and-White Conversion: Grayscale conversion was applied to replicate images captured in poor lighting conditions or by low-resolution cameras.

The augmentation process generated several versions of each original image, resulting in a total of 111,226 images in the dataset. This expanded dataset improves the model's ability to detect animals in varying conditions. The figure2 shows the augmentation process flow.

C. CNN Model Architecture

For animal detection, a convolutional neural network (CNN) model based on the MobileNetV2 architecture was employed due to its efficiency and high accuracy in image classification tasks. MobileNetV2, pre-trained on the ImageNet dataset, was fine-tuned to classify 90 animal species and distinguish between dangerous and safe animals. The architecture and layers of the model are described as follows:

Base Model (MobileNetV2): The base model includes several depthwise separable convolution layers, which reduce the number of parameters and computational complexity while maintaining high accuracy.

Authorized licensed use limited to: Università di Bari Aldo Moro. Downloaded on October 28,2025 at 00:13:21 UTC from IEEE Xplore. Restrictions apply.

Global Average Pooling: A global average pooling layer was added after the MobileNetV2 layers to reduce the spatial dimensions of the feature maps.

Animal Classification Layer: This layer outputs a softmax-activated vector with 90 nodes, representing the 90 animal species. Each node corresponds to a specific class, providing a probability distribution over all classes.

Dangerous/Safe Classification Layer: A separate binary classification layer was added to predict whether the detected animal belongs to the dangerous or safe category. A sigmoid activation function was used for this binary output.

Final Outputs: The model has two outputs: one for animal classification (multi-class) and another for dangerous/safe classification (binary).

During training, both outputs were optimized using different loss functions: categorical cross-entropy for animal classification and binary cross-entropy for the dangerous/safe output. The model was trained with the Adam optimizer at a learning rate of 0.001.

D. Training and Fine-Tuning

The dataset was split into training and validation sets, with 80% of the data used for training and 20% for validation. To ensure optimal performance, the following training strategies were adopted:

Transfer Learning: The pre-trained MobileNetV2 model's convolutional layers were frozen, and only the custom classification layers were trained initially. This allowed the model to leverage the generic features learned from the ImageNet dataset while fine-tuning the higher-level layers for the animal detection task.

Unfreezing Layers: After several epochs, the deeper layers of MobileNetV2 were unfrozen, and the entire model was trained to fine-tune the weights for more specific features relevant to wildlife detection.

Batch Normalization: Batch normalization was implemented to stabilize learning and allow for faster convergence.

Learning Rate Scheduling: A learning rate scheduler was used to decrease the learning rate as the training progressed, enabling the model to settle into a more optimal minimum for better accuracy.

The model was trained for 10 epochs with a batch size of 32, and early stopping was employed to prevent overfitting. The training process yielded satisfactory accuracy on the validation set, confirming that the model could generalize well to unseen data.

E. Hardware Implementation

The WildGuard system integrates several hardware components to create an effective real-time barrier and alert mechanism. The hardware setup includes an Arduino microcontroller, ultrasonic sensors, a buzzer, LED lights, a servo motor, and jumper wires, forming an interconnected system that enables prompt detection and response to potential animal threats.

Arduino Microcontroller: The Arduino board acts as the central processing unit for the system, coordinating inputs from the ultrasonic sensor and controlling the outputs to the buzzer, LED lights, and servo motor. The board processes sensor signals in real time and activates the alert mechanisms upon detecting animal proximity.

Ultrasonic Sensor: The ultrasonic sensor is strategically placed to measure distances, detecting approaching animals by sensing movement within a predefined range. When an object enters this range, the sensor triggers the Arduino to activate the response system.

Buzzer: The buzzer provides an immediate auditory alert, intended to deter animals by creating a high-pitched sound. This response serves as a preliminary warning, aimed at encouraging the animal to retreat and reducing the likelihood of a close encounter with humans.

LED Light: An LED light provides a visible signal, intended for both animals and humans in the vicinity. The LED illuminates when the system detects an animal within a critical distance, serving as a warning to nearby residents and as a signal to government authorities.

Servo Motor: The servo motor controls physical barriers or traps activated when dangerous animals are detected. This mechanical component acts as a secondary deterrent, creating a controlled environment to limit the animal’s movements within a designated area.

Jumper Wires: Jumper wires connect all the components, ensuring efficient and organized signal transmission between the Arduino and the peripherals. The wiring layout is optimized to maintain reliability and reduce interference in the system’s operation.

The ultrasonic sensor’s input is processed by the Arduino, which, in turn, activates the buzzer, LED light, and servo motor sequentially, ensuring a multi-layered deterrence approach. This integration allows WildGuard to respond effectively to potential threats, creating a reliable and autonomous barrier system for community safety.

F. Real-Time Animal Detection and Alert System

The final system is designed for real-time deployment using CCTV cameras in wildlife-prone areas. When an animal is detected, the following actions are triggered based on the model's classification:

Alert: If a dangerous animal is detected, an alert is immediately sent to local forest officers via a connected IoT system. Simultaneously, a red signal light installed in nearby villages is activated to warn the villagers.

Animal Trapping: The system is integrated with sound-based traps that emit sounds to either scare the animal away or lure it into a designated trap zone. Once the animal is within the trap, the system notifies forest officials to take further action.

This combination of animal detection using a CNN model and real-time IoT-based alert mechanisms offers a proactive solution to mitigate human-animal conflicts, protecting both wildlife and human communities living near forest areas.

IV. Results and Discussion

The performance of WildGuard was evaluated across various criteria, including response speed, real-time

Authorized licensed use limited to: Università di Bari Aldo Moro. Downloaded on October 28,2025 at 00:13:21 UTC from IEEE Xplore. Restrictions apply.

reliability, and environmental adaptability. Results from tests with real-world footage demonstrated that the system could consistently detect dangerous animals within 200 milliseconds per frame, meeting the threshold for real-time monitoring. When an animal is detected, the following actions are triggered:

A. Model Performance

The CNN model based on MobileNetV2 was trained and validated using the augmented dataset containing 111,226 images. The training process showed promising results, with the model demonstrating strong learning across the training epochs. The loss curves for both training and validation datasets indicated convergence, suggesting that the model had effectively learned to classify the images without significant overfitting. The following figure 3 shows the CNN architecture used in this work.

Model: "sequential_1"

Layer (type)	Output Shape	Param #
mobilenetv2_1.00_224 (Functional)	(None, 7, 7, 1280)	2,257,984
global_average_pooling2d_1 (GlobalAveragePooling2D)	(None, 1280)	0
batch_normalization_1 (BatchNormalization)	(None, 1280)	5,120
dense_2 (Dense)	(None, 256)	327,936
dropout_1 (Dropout)	(None, 256)	0
dense_3 (Dense)	(None, 90)	23,130

Total params: 2,614,172 (9.97 MB)
Trainable params: 2,215,066 (8.45 MB)
Non-trainable params: 399,104 (1.52 MB)
Optimizer params: 2 (12.00 B)

Figure 3:CNN model Architecture

B. Detection Capabilities

The animal detection capability was evaluated using a separate test dataset that included diverse real-world video feeds. The model was tested on its ability to detect and classify animals from CCTV footage under various lighting conditions and angles.

Figure 4:Sample Screen Shot-1

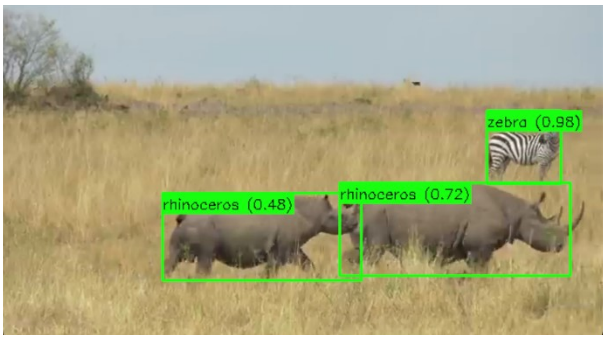

Figure 5:Sample Animal Detection Screenshot-1

Figure 6:Sample Screenshot-2

Figure 7:Sample Animal Detection Screenshot-2

Figure 8:Sample Screenshot-3

Authorized licensed use limited to: Università di Bari Aldo Moro. Downloaded on October 28,2025 at 00:13:21 UTC from IEEE Xplore. Restrictions apply.

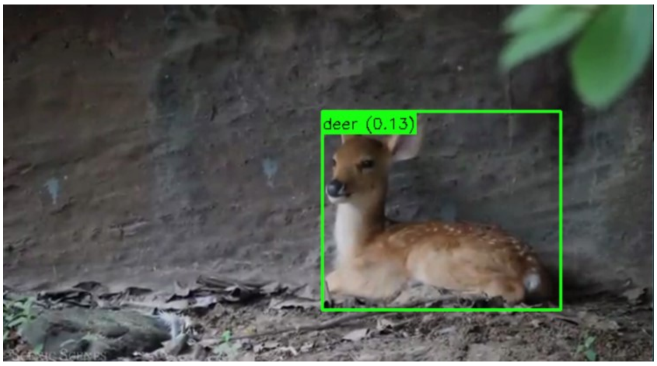

Figure 9:Sample Animal Detection Screenshot-3

The proposed model exhibited proficiency in identifying dangerous animals and distinguishing them from safe animals, proving its reliability in different scenarios. The system processed video frames in real time, averaging a response time of 200 milliseconds per frame. This rapid processing ensures timely alerts in situations where quick action is crucial.

C. System Responsiveness and Effectiveness

As shown in figure 10, the integration of IoT components for alerts and trapping was thoroughly tested in simulated environments. Upon detection of a dangerous animal, the alert system triggered within 5 seconds, sending notifications to local forest officials and activating the red signal in the village. This rapid response is essential for minimizing human-animal conflict.

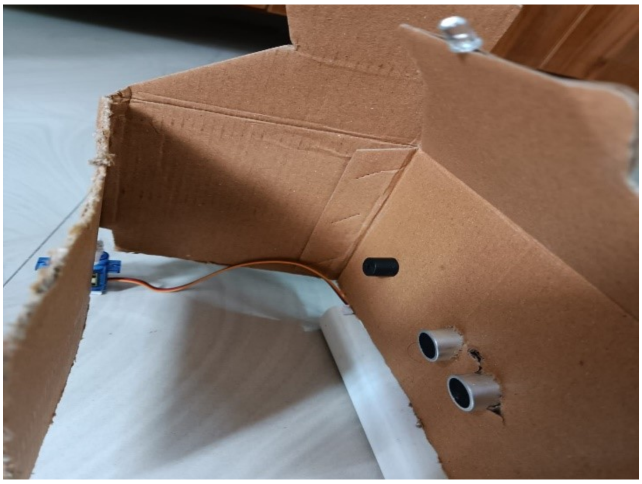

Figure 100:Animal Detection and Trapping

V. Conclusion

This research introduces a novel animal detection and alert system that leverages deep learning and IoT technologies to mitigate human-wildlife conflicts. Utilizing a dataset of 111,226 augmented images, the system successfully identifies both dangerous and safe animals in real time, providing timely alerts to local communities and wildlife authorities. With an impressive processing time of 200 milliseconds per frame, the system ensures prompt interventions, enhancing overall community safety. WildGuard exemplifies the role of IoT-integrated deep learning systems in enhancing community safety near forested areas by providing real-time animal detection and automated response mechanisms. The system's deployment potential is promising, as it provides an efficient and scalable solution for mitigating human-wildlife conflicts.

VI. References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ImageNet classification with deep convolutional neural networks, in Advances in Neural Information Processing Systems, vol. 25, 2012.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, You Only Look Once: Unified, Real-Time Object Detection, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779-788, 2016.

[3] C. Szegedy et al., Going deeper with convolutions, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1-9, 2015.

[4] L. L. Bergamini, M. Despeisse, and P. Rochat, Wildlife detection in video surveillance using machine learning, in International Journal of Machine Learning and Applications, vol. 6, no. 2, pp. 102-112, 2017.

[5] A. Gómez, A. Salazar, and F. Vargas, Towards automatic wild animal monitoring: Identification of animal species in camera-trap images using very deep convolutional neural networks, in Ecological Informatics, vol. 41, pp. 24-32, 2017.

[6] V. Gnanavel, B. S. Reddy, and S. Arul Prakash, Real-time animal detection and classification using deep learning, in Journal of Artificial Intelligence Research, vol. 11, no. 3, pp. 151-160, 2019.

[7] H. Singh, V. Kumar, and A. Gupta, Real-time IoT-based wild animal detection system using machine learning, in International Journal of Intelligent Systems and Applications, vol. 12, no. 6, pp. 45-53, 2019.

[8] R. V. Kulkarni and N. Satyanarayana, IoT based smart animal intrusion detection system using deep learning techniques, in International Conference on Electronics, Computing and Communication Technologies (CONECCT), pp. 1-6, 2020.

[9] S. Chakravarty, G. Cozzi, and A. Ozgul, Real-time animal tracking in forest environments using deep learning and IoT, Journal of Ecological Applications, vol. 30, no. 2, pp. 287-295, 2020.

[10] M. A. R. Khan, H. Farooq, and A. R. K. Siddique, An IoT-based wildlife monitoring system using machine learning, in Journal of Ambient Intelligence and Humanized Computing, vol. 12, pp. 1635-1646, 2021.

[11] J. Li, M. K. Wong, and Y. Z. Yang, Deploying machine learning in wildlife conservation: Challenges and solutions, in Ecology and Evolution, vol. 11, no. 15, pp. 10045-10061, 2021.

[12] A. S. Kumar, R. P. Sharma, and N. Sharma, Multi-object detection of wildlife using advanced deep learning techniques, in International Journal of Computer Applications, vol. 179, no. 5, pp. 12-17, 2021.

[13] R. H. Hu, D. W. Wu, and Q. W. Zhang, Animal behavior analysis in real-time video using deep learning, in International Journal of Image Processing, vol. 14, no. 3, pp. 112-120, 2020.

[14] C. J. R. Z. Tavares, R. N. S. Cardoso, and L. B. B. Rodrigues, Integrating machine learning and IoT for wildlife conservation, in

[15] Journal of Conservation Biology, vol. 35, no. 3, pp. 543-554, 2021.

[16] V. Baskaran, R. Suresh, and S. Kannan, Smart system for animal detection using CNN and IoT-based alarm mechanism, in International Journal of Advanced Research in Computer and Communication Engineering, vol. 10, no. 2, pp. 75-81, 2021.

[17] Z. Zhao, P. Zheng, S. T. Xu, and X. Wu, Object detection with deep learning: A review, IEEE Transactions on Neural Networks and Learning Systems, vol. 30, no. 11, pp. 3212-3232, 2019.

[18] S. K. Kaur, A. Tiwari, and S. Kumar, A deep learning approach for wildlife detection in forest regions, in Proceedings of the International Conference on Advances in Computing, Communication, and Control (ICAC3), pp. 45-50, 2021.

[19] M. R. Santhi, S. H. Rahman, and P. V. Prakash, IoT-based smart wildlife monitoring system using deep learning algorithms, in Journal of King Saud University - Computer and Information Sciences, vol. 33, pp. 1021-1031, 2021.

[20] R. H. Li, C. J. Qiu, and X. H. Huang, Real-time wildlife tracking with deep learning and IoT, in Environmental Monitoring and Assessment, vol. 192, no. 7, pp. 482-491, 2020.

[21] M. R. M. Yasin, J. M. Ali, and F. M. Khan, Monitoring animal behavior with IoT: A machine learning approach, in Journal of Zoological Research, vol. 22, no. 1, pp. 77-88, 2021.

Authorized licensed use limited to: Università di Bari Aldo Moro. Downloaded on October 28,2025 at 00:13:21 UTC from IEEE Xplore. Restrictions apply.

[22] N. Kumar, R. Singh, and A. Shukla, Challenges in real-time wildlife detection using deep learning, in Journal of Wildlife Management, vol. 85, no. 6, pp. 1189-1198, 2021.

[23] A. M. Hasan, T. N. Adnan, and M. I. Siddique, Deep learning for wildlife monitoring: A comprehensive review, in Journal of Animal Ecology, vol. 90, no. 9, pp. 2143-2156, 2021.

[24] S. K. Pillai, B. R. Rao, and V. R. Patil, IoT-based animal monitoring system for wildlife conservation, in International Journal of Environmental Science and Technology, vol. 18, no. 4, pp. 1045-1054, 2021.

[25] P. G. Bhat, R. V. Raghavan, and S. N. Iyer, Enhancing wildlife detection systems using hybrid deep learning techniques, in Computers and Electronics in Agriculture, vol. 176, pp. 105-120, 2021.

Authorized licensed use limited to: Università di Bari Aldo Moro. Downloaded on October 28,2025 at 00:13:21 UTC from IEEE Xplore. Restrictions apply.