BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **25**, No 1
Sofia • 2025 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2025-0002

Federated Learning and Blockchain-Based Collaborative Framework for Real-Time Wild Life Monitoring

Preetha Jagannathan[1], *Kalaivanan Saravanan*[2], *Subramaniyam Deepajothi*[3], *Sharmila Vadivel*[4]

[1]*Department of Information Technology, Kongunadu College of Engineering and Technology, Trichy, Tamil Nadu,621215, India*
[2]*Department of Information Science and Engineering, New Horizon College of Engineering, Bangalore, 560103, India*
[3]*Department of Computing Technologies, SRM College of Engineering and Technology, Kattankulathur, Tamil Nadu, 603203, India*
[4]*Department of Computer Science and Engineering, K.S.R College of Engineering, Namakkal, Tamil Nadu, 637215, India*
E-mails: psgpreetha@gmail.com (corresponding author) saravklvn@gmail.com deepajothi.a@gmail.com sachinsv06@gmail.com

***Abstract*: *Effective wildlife monitoring in hilly and rural areas can protect communities and diminish human-wildlife conflicts. A collaborative framework may overcome challenges like inadequate data integrity and security, declining detection accuracy over time, and delays in critical decision-making. The proposed study aims to develop a real-time wildlife monitoring framework using Federated Learning and blockchain to improve conservation strategies. Min-max normalization enhances training data and Elastic Weight Consolidation (EWC) for real-time adaptation. The improvised YOLOv8+EWC enables real-time classification and continual learning and prevents catastrophic forgetting. It also automates actions based on detection results using smart contracts and ensures secure, transparent data management with blockchain. Compared to existing classifiers such as Deep Neural Network, Dense-YOLO4, and WilDect: YOLO, YOLOv8+EWC performs exceptionally well across several metrics, accomplishing an accuracy of 98.91%. Thus, the proposed model enables reliable decision-making by providing accurate, real-time information about wildlife.*

***Keywords*: *Blockchain, Elastic Weight Consolidation (EWC), Federated averaging, Federated learning, YOLOv8.*

1. Introduction

Wildlife surveillance is crucial for preserving localities in mountainous regions, where interactions between humans and wildlife are often more frequent due to the proximity of ecological areas. Effective monitoring can update conservation strategies using technology and promote safer interactions between wildlife and local communities. This ultimately mitigates the risk of wildlife-related conflicts, such as attacks or property damage, and supports biodiversity [1, 2]. Typically, wildlife tracking is investigated by professionals who collect data through feedback forms, tracking systems, and diverse observational approaches. While these methods offer valuable perceptions, they are often labor-intensive and time-consuming, making them resource-demanding. The comprehensive field research and hands-on data examination required can limit the regularity and extent of monitoring activities, possibly delaying prompt conservation efforts [3].

To address the challenges of traditional wildlife monitoring, Machine Learning (ML) [4, 5] and Deep Learning (DL) frameworks are progressively utilized to automate the analysis of extensive datasets [6, 7]. DL models, particularly Convolutional Neural Networks (CNNs), have acquired considerable focus in research literature because of their exceptional recognition capability. By utilizing CNNs, researchers proficiently handle significant quantities of the recorded image and video content from drone and camera traps [8, 9]. This technology allows for rapid recognition and sorting of various species, significantly minimizing the time and effort needed for manual analysis. Thus, using CNNs enhances the reliability of species detection and facilitates prompt conservation initiatives and improved wildlife management. This change toward automated observation represents a notable breakthrough in wildlife research and conservation strategies [10, 11]. Despite these advancements, there are still challenges, such as a lack of robust data integrity and security, decreased detection accuracy over time, delay in critical decision-making, and notable deficiency in collaboration among stakeholders. To address these issues, we propose a blockchain-enabled Federated Learning (FL) platform that enhances wildlife monitoring and conservation strategies.

1.1. Research questions

The subsequent research questions are examined to evaluate the significance of the proposed research work:

Q1: What strategy can be implemented to ensure data integrity and authenticity of wildlife monitoring systems?

Q2: How can we reduce reliance on human intervention for decision-making and alerts in responding to wildlife events?

Q3: What approach can be used to develop models that adapt to new data or changing environments in real-time?

Q4: What can be done to create frameworks that facilitate effective collaboration among researchers, conservationists, and local communities?

Q5: What steps can be taken to improve models that lack consistency in performance across different environments and species?

To answer these questions, we propose a real-time framework integrating blockchain and Federated Learning (FL) to secure wildlife monitoring data to solve Q1, automate actions based on detection results through smart contracts to reduce manual intervention for solving Q2, implement Elastic Weight Consolidation (EWC) for real-time adaptation to solve Q3, establish a decentralized platform for enhanced collaboration to facilitate better communication and data sharing for solving Q4 and employ an advanced data preprocessing strategy to improve the training dataset for solving Q5.

1.2. Research contributions

As far as we have done an exhaustive survey on related works, the proposed real-time wildlife monitoring framework is the first to integrate blockchain, FL, and CNN for wildlife monitoring. The study's key contributions are listed below.

- We employ blockchain technology to secure wildlife monitoring data, effectively addressing the need for data integrity and transparency. This approach ensures the information is accessible and trustworthy for all stakeholders, fostering greater collaboration and informed decision-making in conservation efforts.
- The proposed framework automates actions based on detection results through smart contracts, minimizing the need for manual intervention. This accelerates response efforts and improves overall effectiveness, enabling faster and more effective decision-making in critical situations.
- By implementing an on-device continual learning technique, Elastic Weight Consolidation (EWC), we ensure the proposed model remains accurate despite environmental changes. This approach allows continuous updates, enhancing the framework's flexibility and performance.
- We have created a decentralized platform for collaboration that facilitates better communication and data sharing among researchers and conservationists, leading to more effective conservation strategies. By fostering a community-driven approach, we further strengthen the impact of conservation efforts.
- We use an advanced data preprocessing strategy to enhance the training dataset, improving model robustness and generalization capabilities. This approach improves performance and ensures the framework can adjust effectively to diverse real-world scenarios.

The arrangement of the research paper is as follows: Section 2 reviews a few wildlife monitoring-based frameworks, Section 3 outlines the structure of the proposed real-time wildlife monitoring framework, Section 4 presents the performance analysis of the proposed wildlife monitoring framework, and Section 5 provides the conclusion of the research work.

2. Literature survey

This section examines a few frameworks focused on wildlife activity monitoring. A DL model, KI-CLIP, was developed to track endangered wildlife with limited data and low computational cost [12]. Though the model achieved over 97% recognition accuracy, it struggled to detect small targets. An animal species recognition model was suggested to improve the detection accuracy grounded on YOLOv2 [13]. Though the improved YOLOv2 model outperformed the base YOLOv2 by 12% in speed, the researchers further aimed to enhance the detection speed. To further enhance developments to automate the identification, classification, and counting of wildlife species in camera trap videos, the research in [14] used Faster R-CNN and Inception-ResNet-v2. However, this approach required a large amount of training data, which was challenging to obtain for rare species. To combat this, the authors [15] introduced WilDect-YOLO for real-time detection of endangered wildlife. It integrated residual blocks in CSPDarknet53 and DenseNet blocks to improve feature extraction and preservation. It also used Spatial Pyramid Pooling (SPP) and a modified Path Aggregation Network (PANet) for enhanced feature integration. Further advancements included a system using CNNs to detect wildlife intrusions and send alerts to mitigate human-wildlife conflicts [16]. Although this system showed promising results in accurately detecting intrusions, its success depended on factors including data quality, model accuracy, and alert mechanism efficiency.

An automated action detection system was developed that combined SWIFT (Segmentation with Filtering of Tracklets) for detecting and tracking wildlife and MAROON (Mask-Guided Action Recognition) for recognizing their actions [17]. However, the system didn't detect interactions between multiple animals. To address this limitation, a hybrid Deep Neural Network (DNN) model [18] was created, combining Visual Geometry Group 19 (VGG 19) and Bidirectional Long Short-Term Memory (Bi-LSTM) to distinguish wild animal movements and create alarm messages to ensure safety. The authors used a dataset of 40,000 images across 25 classes. The paper [19] highlighted continual learning to improve inference reliability for on-site wildlife monitoring to enhance performance further. The model achieved a 10% higher F1-score on-site compared to off-site processing. Consequently, we recognized several research gaps by analyzing the existing wildlife monitoring frameworks. The existing frameworks lack robust data integrity and security, making them vulnerable to tampering and compromising trustworthiness while also relying heavily on human intervention for real-time responses, leading to delays in critical decision-making. Additionally, traditional models often struggle with adaptability, resulting in decreased detection accuracy over time, and there is a notable deficiency in collaboration among stakeholders, as current systems fail to notice deficiency, hindering conservation efforts. To tackle these limitations, our proposed real-time wildlife monitoring framework integrates FL and blockchain to enhance wildlife monitoring and conservation strategies significantly.

3. Proposed real-time wildlife monitoring framework

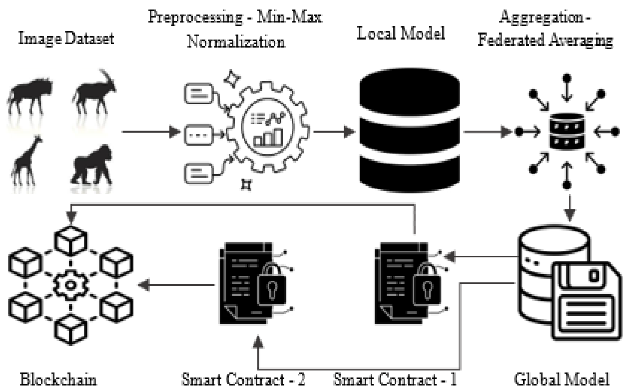

Fig. 1. Illustration of the proposed real-time wildlife monitoring framework

The proposed real-time wildlife monitoring framework integrating blockchain and FL is shown in Fig. 1. It uses an advanced preprocessing process to improve the training dataset and incorporates EWC for real-time adaptation without catastrophic forgetting. The improvised YOLOv8+EWC allows for real-time recognition and classification of wildlife, enables continual learning, and prevents the loss of previously learned information. The framework automates actions based on detection results via smart contracts, reducing manual intervention. Integrating blockchain technology establishes a secure, transparent data management system and a decentralized platform for enhanced collaboration and data sharing.

3.1. Dataset description

The Animals Detection Images Dataset, sourced from Google Open Images V6+, comprises a collection of wild animal species and annotations. It features 21 animal classes – dog, cat, zebra, lion, leopard, cheetah, tiger, bear, brown bear, butterfly, canary, crocodile, polar bear, bull, camel, crab, chicken, centipede, cattle, caterpillar, and duck. For our study, we focus on six: lion, cheetah, leopard, tiger, crocodile, and bear. These species are far more dangerous and require quick alerts to ensure the safety of both wildlife and human communities. Fig. 2 presents a few selected images from the dataset.
Dataset Link: **https://www.kaggle.com/datasets/antoreepjana/animals-detection-images-dataset/data**

Fig. 2. Reference images from the dataset: top row (left to right) – crocodile, bear, tiger; bottom row (left to right) – lion, cheetah, leopard

3.2. Image preprocessing through min-max normalization

Normalizing image data improves the proposed model’s convergence and ensures consistency across different image types. We utilize Min-Max Normalization [20] to scale pixel values between [0, 1] by dividing pixel values by 255. Next equation is used to normalize the features,

(1) $$A_{\mathrm{nw}} = \frac{A - A_{\mathrm{mn}}}{A_{\mathrm{mx}} - A_{\mathrm{mn}}},$$

where A is the obtained feature values, A_{mn} is the minimum value in A, and A_{mx} is the maximum value in A. By preserving the relative distribution of values, min-max normalization helps the framework identify subtle patterns in the data more effectively, resulting in more accurate classification and better overall performance in wildlife monitoring tasks.

3.3. Local model training (improvised YOLOv8+EWC)

EWC enables continual learning and helps prevent the loss of previously learned information. YOLOv8 offers better accuracy in wildlife species detection, with higher mean Average Precision (mAP) and increased speed. We incorporate the EWC component into the YOLOv8 model so that the previously trained features of wildlife are retained. Our model's innovative use of EWC permits real-time adaptation, efficient edge device deployment, and seamless integration with FL.

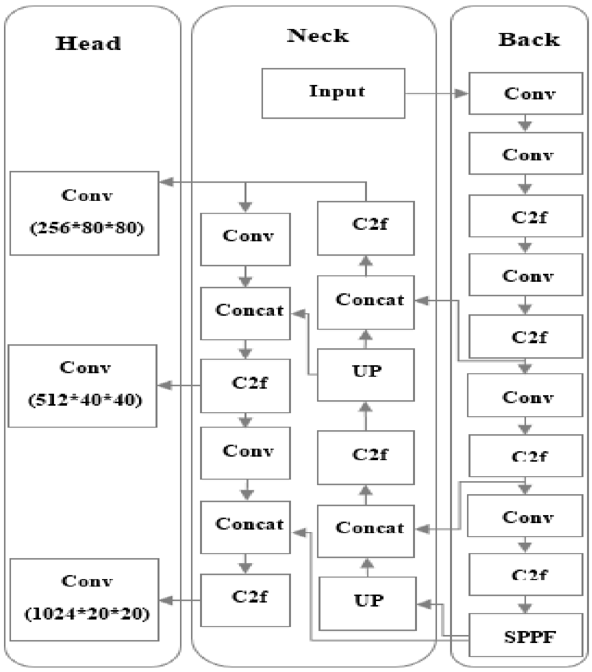

Fig. 3. Structure of the YOLOv8 architecture

YOLOv8 [21] employs an anchor-free strategy that identifies object centers instead of relying on predefined anchor boxes. This approach ensures a faster post-processing phase through simplified non-maximum filtering. YOLOv8's training integrates techniques like online image and mosaic augmentation, improving the framework's capability to distinguish wildlife in diverse scenarios. Moreover, YOLOv8 simplifies the neck segment by directly concatenating features, as shown in Fig. 3. Learning a task involves fine-tuning the weights and biases $\emptyset$ of linear projections to enhance performance. Over-parameterization implies there is a solution for workload V, $\emptyset_V^*$, that is in proximity to the earlier solution for workload

U, $\emptyset_U^*$. While learning workload V, EWC [22] maintains the performance in workload U by confining the constraints to remain within a low-error zone for workload U concentrated on $\emptyset_U^*$. Adjusting the parameters is equivalent to identifying the likely values given some data X. We compute conditional probability $p(\emptyset|X)$ through Bayes' rule with the prior probability of parameters $p(\emptyset)$ and the probability of data $p(X|\emptyset)$,

(2) $$\log p(\emptyset|X) = \log p(X|\emptyset) + \log p(\emptyset) - \log p(X).$$

When the data is divided into two independent sets, one for task $U(X_U)$ and the other for task $V(X_V)$, we can rearrange Equation (2) as the next equation,

(3) $$\log p(\emptyset|X) = \log p(X_V|\emptyset) + \log p(\emptyset|X_U) - \log p(X_V).$$

All the information related to a task U is taken in by the posterior distribution $p(\emptyset|X_U)$, which reveals which parameters are vital to task U and are crucial for implementing EWC. We approximate the posterior as a Gaussian distribution, where the parameters represent the mean $\emptyset_U^{\approx}$ and a diagonal precision is derived from the transverse elements of the Fisher information matrix Y. Based on this calculation, the function L to be minimized in EWC is expressed in equation

(4) $$L(\emptyset) = L_V(\emptyset) + \sum_Z \frac{\lambda}{2} Y_i (\emptyset_z - \emptyset_{U,z}^{\approx})^2,$$

where $L_V(\emptyset)$ is the loss specific to task V, λ determines the importance of the previous task compared to the new one, and z tags each parameter. When transitioning to a third task, task W, EWC aims to maintain the network parameters close to the learned parameters from both task U and V. This is achieved either with two isolated penalties or by combining them into a single penalty.

3.4. Federated Learning (FL) – global model aggregation

The central server aggregates local model updates to create a global model, which is redistributed to the edge devices. Local models in FL focus on data privacy and efficiency by processing data on individual devices. This decentralized tactic also improves responsiveness as the updates occur swiftly without extensive data transfers to a central server. The global model improves accuracy by combining knowledge from different locations. We use FL to aggregate model updates from multiple edge devices using Federated Averaging (FedAvg) [23]. It works in rounds to solve the optimization problem outlined in equation (5). In each round, the global model is sent to a random user group, who then adjust the model on their local data for a set number of epochs. When the process ends, the user sends back updates that reflect the changes they made. The server collects these updates until a timeout and then combines them using weighted averaging based on how much data each client has. Finally, the server updates the global model with the averaged results. The pseudocode for FedAvg applied in the proposed framework is presented in Pseudocode 1.

Let's assume a_0 is the initial model parameters, φ represents the client update computed based on its local training, ϑ denotes the learning rate, g indicates the number of participated clients, j represents the fraction of clients, E_b is a private dataset held by each client of the user b, ℓ is the loss function, and h signifies the number of local epochs:

(5) $$\min_a f(a) := \sum_{b=1}^{B} \frac{c_b}{c} D_b(a),$$

where a represents the global model parameters for training, B denotes the total number of users, c_b indicates the data sample quantity held by the user (b), c is the total data sample quantity across all users, and $D_b(a)$ refers to the local objective function for each client.

Pseudocode 1. Pseudocode of Federated Averaging

Objective: To update the global model

Input: B and c_b

Output: φ

Step 1. central server do:

Step 2. initialize a_0

Step 3. for every round $t = 0, 1, 2, 3 \ldots$

Step 4. $g \leftarrow \max(j * B, 1)$

Step 5. $f_t \leftarrow$ (random set of g clients)

Step 6. for each client $b \in f_t$

Step 7. $\varphi_t^b \leftarrow \mathrm{ClientUpdate}\ (b, a_t)$

Step 8. $\varphi_t \leftarrow \sum_{i \in f_t} \frac{c_b}{c} \varphi_t^b$

Step 9. $a_{t+1} \leftarrow a_t + \varphi_t$

Step 10. client update (b, a_t)

Step 11. $a \leftarrow a_t$

Step 12. $K \leftarrow$ (divide E_b into sets of size K)

Step 13. for all local epoch h from 1 to H

Step 14. for $k \in K$

Step 15. $a \leftarrow a - \vartheta \nabla \ell(a; k)$

Step 16. $\varphi \leftarrow a - a_t$

Step 17. return φ to the server

3.5. Global model architecture – MobileNetV3

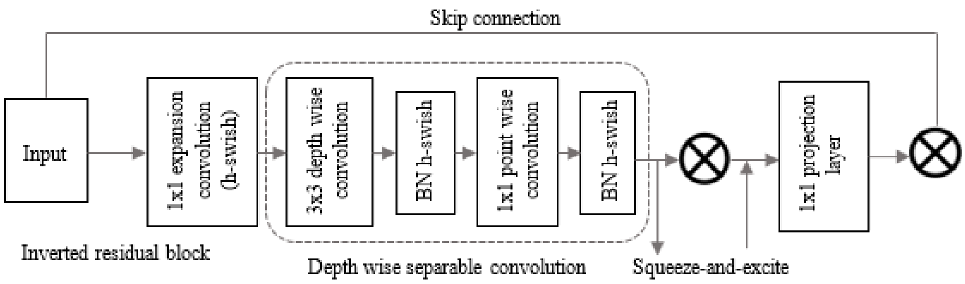

Fig. 4. Design of the MobileNetV3 architecture

We deploy MobileNetV3 [24] as the architecture for the global model to enable scalable and efficient object detection and classification with FL. In MobileNetV3, the depth-wise convolutional layer is the central building block, as shown in Fig. 4. The layer replaces the traditional layer with a segmented layer to minimize the model scale. It consists of two components: depth-wise convolution, which implements one filter for every input channel, and a 1×1 pointwise convolution, which creates new feature maps through the linear aggregations of the input channels.

3.6. Smart contracts and blockchain technology

Our research uses smart contract [25] automation for wildlife monitoring, as well as blockchain [26] to ensure secure, privacy-preserving, and decentralized data storage.

3.6.1. Smart contracts

Smart contracts automate the process of triggering alerts and actions in the proposed real-time wildlife monitoring framework. When wildlife is detected in human habitats, these smart contracts promptly trigger predefined actions, such as notifying the relevant authorities or conservation teams. The pseudocode for the smart contract utilized in the proposed framework is presented in Pseudocode 2.

Pseudocode 2. Pseudocode of Smart Contract

Objective: Smart contract for wildlife monitoring

Input: Wildlife species

Output: Wildlife detection alert

Step 1. CONTRACT WildlifeMonitor
Step 2. STRUCT AnimalDetection
Step 3. STRING species
Step 4. STRING location
Step 5. UINT256 timestamp
Step 6. BOOLEAN endangered
Step 7. BOOLEAN alerted
Step 8. MAPPING detection count → AnimalDetection detections
Step 9. UINT256 detection count
Step 10. EVENT DetectionLogged(detectionId, species, location, timestamp)
Step 11. EVENT AlertTriggered(detectionId, species, location)
Step 12. FUNCTION logDetection(species, location, endangered)
Step 13. INCREMENT detection count
Step 14. CREATE new detection = AnimalDetection(species, location, CURRENT_TIMESTAMP, endangered, FALSE)
Step 15. STORE new detection in detections[detectionCount]
Step 16. EMIT DetectionLogged(detection count, species, location, CURRENT_TIMESTAMP)
Step 17. IF endangered THEN
Step 18. CALL triggerAlert(detectionCount, species, location)
Step 19. ELSE IF species IS NULL THEN
Step 20. ELSE
Step 21. FUNCTION triggerAlert(detectionId, species, location)
Step 22. SET detections[detectionId].alerted = TRUE
Step 23. EMIT AlertTriggered(detectionId, species, location)
Step 24. FUNCTION getDetection(id)
Step 25. GET detection = detections[id]

Step 26. RETURN (detection.species, detection.location, detection. timestamp, detection. endangered, detection.alerted)

Step 27. END CONTRACT

3.6.2. Blockchain

We store animal detections, including cheetahs, leopards, lions, and tigers, along with important data points on the blockchain. This ensures invariability and transparency, ensuring a reliable and secure record of wildlife activity. The process, illustrated in Figs 5 and 6, begins with deploying the smart contract. After this, a smart contract confirmation form is generated to verify the submission.

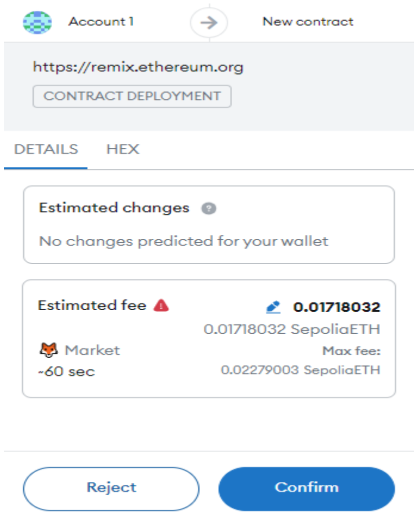

Fig. 5. Smart contract deployment form

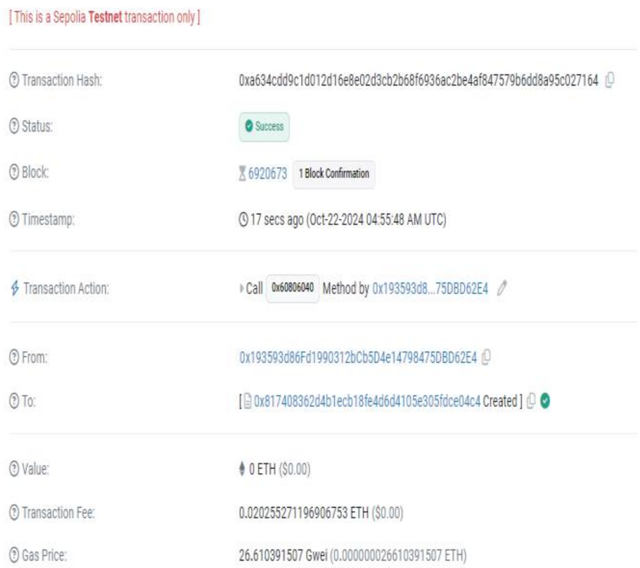

Fig. 6. Smart contract confirmation form

4. Results and discussion

In this section, we assess the operation of the proposed real-time wildlife monitoring framework. The frontend framework uses React to build the user interface, and a library like web3.js interacts with Ethereum smart contracts for blockchain integration. During implementation, the Sepolia testnet blockchain was chosen to test and deploy the framework. This approach ensured the model would operate smoothly when deployed in a real-world environment.

4.1. Output of the proposed wildlife monitoring blockchain unit

Fig. 7 illustrates the form representing the alert triggered when a wildlife species (bear) is detected. The ‘logs record the species, location, and timestamp.

[block:6920786 txIndex:25] from: 0x193...d62e4 to: WildlifeMonitor.logDetection(string,string,bool) 0x817...e04c4 value: 0 wei data: 0xd68...00000 logs: 2 hash: 0xb6f...09165
status 0x1 Transaction mined and execution succeed
transaction hash 0xe3f6eed80f1b7e2a5137c62cf363762cda33e0b2d50e2a063e4cea6be9538a2f
block hash 0xb6f45c5b9c5ca0553972cad7e2c191d90886f2b8ed4d4d806dbe9fa62fe09165
block number 6920786
from 0x193593d86fd1990312bcb5d4e14798475dbd62e4
to WildlifeMonitor.logDetection(string,string,bool) 0x817408362d4b1ecb18fe4d6d4105e305fdce04c4
gas 145660 gas
transaction cost 144504 gas
input 0xd68...00000
decoded input {
"string _species": "Bear",
"string _location": "Nagercoil",
"bool _endangered": true
}
decoded output -
logs [
{
"from": "0x817408362d4b1ecb18fe4d6d4105e305fdce04c4",
"topic": "0x3e99c227055d1cdcbe8b0d2c6e6ef7791482b5aaeeb2ea12e4b075602d53d61f",
"event": "DetectionLogged",
"args": {
"0": "1",
"1": "Bear",
"2": "Nagercoil",
"3": "1729574460",
"detectionId": "1",
"species": "Bear",
"location": "Nagercoil",
"timestamp": "1729574460"
}
},
{
"from": "0x817408362d4b1ecb18fe4d6d4105e305fdce04c4",
"topic": "0x0d41529466bbba3703731dc8f3d847f5262fd1377336e8bebce6f1413fe9b6f7",
"event": "AlertTriggered",

Fig. 7. Wildlife alert form

4.2. Performance analysis of the improvised YOLOv8+EWC

Fig. 8. Wildlife classification through the improvised YOLOv8+EWC

Fig. 8 displays the wildlife species classified with annotations by the improvised YOLOv8+EWC. It shows incomparable performance in wildlife detection and classification on the Animals Detection Images Dataset across various metrics, including accuracy, precision, recall (True Positive Rate (TPR)), specificity (True Negative Rate (TNR)), F-measure, and Matthews Correlation Coefficient (MCC), as shown in Table 1. Furthermore, it establishes a low error rate, with a False Positive Rate (FPR) of 1.25% and a False Negative Rate (FNR) of 1.5%.

Table 1. Performance metrics of the proposed YOLOv8+EWC

Metrics	Values
Accuracy	98.91%
Precision	98.75%
TPR	98.5%
F-measure	98.62%
TNR	98.5%
MCC	97%

Fig. 9 displays the accuracy and loss curves for the proposed YOLOv8+EWC, presenting insights into its performance during 40 epochs of training. The training and test curves converge smoothly from the first to the last epoch, exhibiting negligible fluctuations. The curves help evaluate the proposed YOLOv8+EWC's performance and its ability to adapt to the Animals Detection Images Dataset.

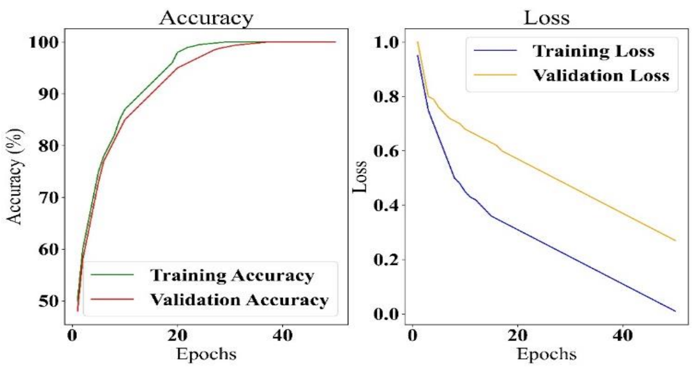

Fig. 9. Accuracy and loss curve of the proposed YOLOv8+EWC for 40 epochs

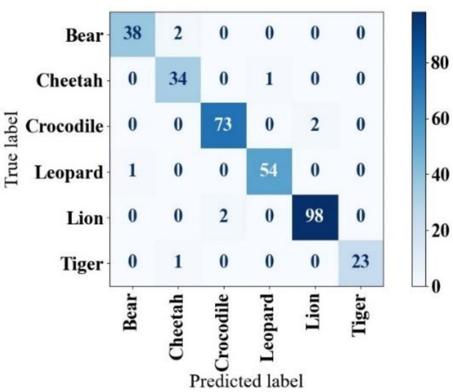

Fig. 10. Assessment of the proposed YOLOv8+EWC with the confusion matrix

Fig. 10 shows the confusion matrix for evaluating the proposed YOLOv8+EWC, distinguishing wildlife species. The true labels are represented by each column, and the forecast labels are denoted by each row. The matrix provides insight into the accuracy of the classification of wildlife species.

Fig. 11 displays the different gas consumption levels (gwei) for the six classes (species). We tracked how each class performed in terms of gas consumption.

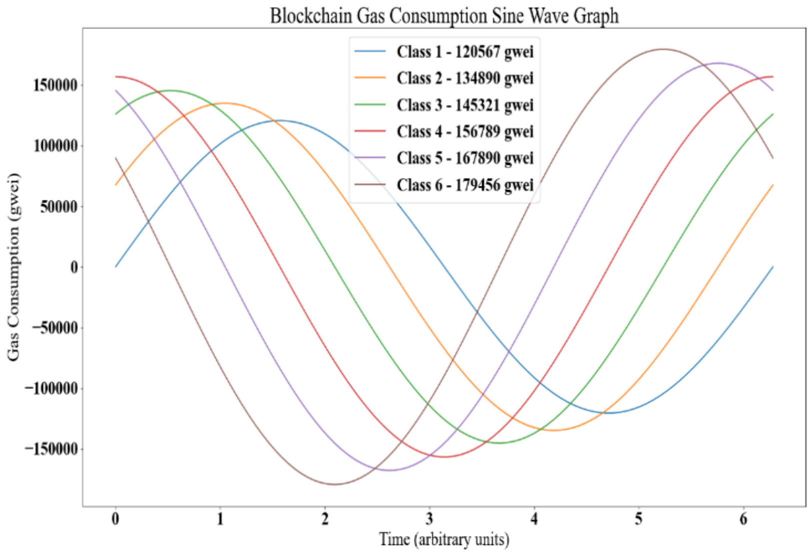

Fig. 11. Tracking of gas consumption levels

4.3. Performance analysis of the FL unit

The FL unit in the proposed real-time wildlife monitoring framework also shows unique performance in wildlife monitoring across various metrics, including accuracy, precision, TPR, TNR, F-measure, and MCC, as shown in Table 2. Moreover, it has a low incidence of errors, with an FPR of 0.0025 and an FNR of 0.031.

Table 2. Performance metrics of the FL unit

Metrics	Values
Accuracy	99.28%
Precision	98.5%
TPR	99%
F-measure	98.75%
TNR	99.5%
MCC	98.5%

Fig. 12 displays the accuracy and loss curves for the FL unit, presenting insights into its performance during training for ten communication rounds. The training and test curves converge smoothly from the first to the last communication round, indicating minimal fluctuations.

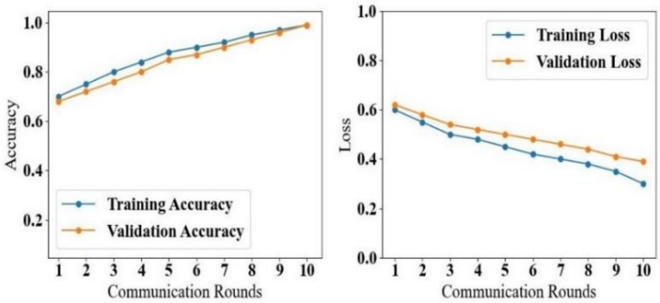

Fig. 12. Accuracy and loss curve of the FL unit

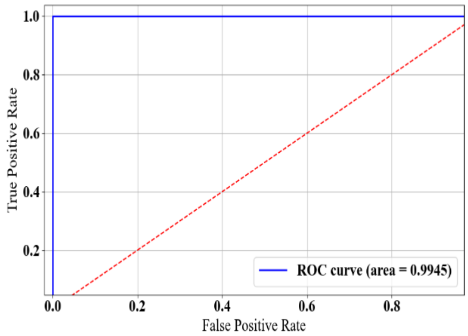

Fig. 13. Area under the Receiver Operating Characteristic Curve (AUC-ROC) of the FL unit

The graph in Fig. 13 depicts AUC, which measures the area under the ROC curve. Higher AUC values indicate better performance of the FL unit.

4.4. Comparative assessment of the proposed and existing frameworks

In this segment, the process efficiency of the proposed real-time wildlife monitoring framework is assessed by comparing it with the existing frameworks.

Table 3. Comparison of performance metrics for the proposed YOLOv8+EWC and existing classifiers

Classifiers / Metrics	Precision	Recall	F1-score
Dense-YOLO4	93.5	96.4	94.9
WilDect-YOLO	97.18	98	97.87
DNN	81	80	81
Improvised YOLOv8+EWC (Proposed)	98.75	98.5	98.62

Table 3 demonstrates that the improvised YOLOv8+EWC outperforms existing classifiers in terms of various performance metrics. The comparison highlights the improvised YOLOv8+EWC's precision, F1-score, and recall metrics against those of existing classifiers, including DNN, WilDect-YOLO, and Dense–YOLO4.

4.5. Discussion

The proposed framework aims to deliver a real-time system for improving wildlife monitoring through the integration of blockchain, CNN, and FL. This framework has several applications, including integrating blockchain, CNN, and FL, which promote the development of smart wildlife monitoring and conservation strategies. The model specifically addresses challenges such as inadequate data integrity and security through blockchain integration and dependence on human intervention for timely responses via smart contracts. The model achieved the best detection accuracies of 98.75% with the improvised YOLOv8+EWC and 99.28% with the FL unit. The proposed YOLOv8+EWC outperformed existing classifiers in terms of various performance metrics. The accuracy and loss curves of both the FL unit and the improvised YOLOv8+EWC show the best performance during training. The ROC curve area of 0.9945 for the FL unit indicated its effectiveness. The classification of wildlife species was further validated through a confusion matrix analysis. The proposed framework could protect hilly communities by enabling real-time wildlife monitoring near residential areas. Sensors and cameras would detect and classify animals approaching settlements, triggering alerts to warn residents, reducing the risk of human-wildlife conflicts, and ensuring safety for both people and animals.

5. Conclusion

In this study, a novel real-time blockchain framework is developed to protect communities in hilly regions using an automated wildlife monitoring system. We enhanced real-time wildlife classification in the proposed framework, as demonstrated by robust performance results and the accuracy and loss curves of the improvised YOLOv8+EWC. Our model's efficiency was also collaborated by high-performance metrics and the notable AUC value of the FL unit. Additionally, integrating smart contracts and blockchain leads to more effective conservation strategies. Blockchain securely stores data from the wildlife monitoring system, providing an immutable record of animal sightings and habitat changes. This hybrid approach upgrades the performance of DL techniques in real-time applications, preserving both wildlife and local communities. Hence, the proposed monitoring framework signifies an extensive advancement in the realm of smart wildlife conservation through automated monitoring. In the future, the proposed framework will be executed on a larger dataset with diverse wildlife species, broadening its ecosystem applicability. Integrating real-time analytics and edge computing will improve performance in remote areas with limited connectivity. Collaboration with conservation organizations will enable deployment in protected regions, aiding in poaching prevention and biodiversity monitoring. These advancements will enhance the proposed framework's global impact on wildlife conservation.

Statements and Declarations

Author contributions: All authors contributed to the conception of the problem setting and overall design of the work. P. J, K. S, S. D, built the conceptualization and methodology, S. D, V. S implemented the work, validation was performed by P. J and K. S and writing was done by P. J, K. S, and S.D. This version was revised and improved by all authors, who also read and approved the final manuscript.

Funding: No funding was received for conducting this study.

Availability of data and materials: The data set is available as public in the Kaggle repository

Conflict of interest: The authors declare that they have no conflict of interest.

Ethical approval: The research is original and all the figures and tables are created by the authors of this manuscript.

Consent to participate: Not applicable.

Consent for publication: All authors agree with the submission of the manuscript to this journal and possible publication afterward**.**

References

1. Delplanque, A., R. Lamprey, S. Foucher, J. Theau, P. Lejeune. Surveying Wildlife and Livestock in Uganda with Aerial Cameras: Deep Learning Reduces the Workload of Human Interpretation by over 70%. – Frontiers in Ecology and Evolution, Vol. **11**, 2023, 1270857.
2. Wood, C. M., A. B. Cruz, S. Kahl. Pairing a User-Friendly Machine-Learning Animal Sound Detector with Passive Acoustic Surveys for Occupancy Modeling of an Endangered Primate. – American Journal of Primatology, Vol. **85**, 2023, No 8, e23507.
3. Tanzib, H. M., A. Zaman, M. R. Abir, S. Akter, S. Mursalin, S. S. Khan. Synchronizing Object Detection: Applications, Advancements, and Existing Challenges. – IEEE Access, 2024.
4. Shinichi, N., L. Malgorzata, F. Roxane, T. Jessica, L. Xun, E. Andrew, J. R. Neil, O'Brien K. Justine, P. J. Benjamin, V. S. Monique, S. Arcot, K. T. Richard. Rapid Literature Mapping on the Recent Use of Machine Learning for Wildlife Imagery. – Peer Community Journal, Vol. **3**, 2023.
5. Aguilar-Lazcano, C. A., I. E. Espinosa-Curiel, J. A. Ríos-Martínez, F. A. Madera-Ramírez, H. Pérez-Espinosa. Machine Learning-Based Sensor Data Fusion for Animal Monitoring: A Scoping Review. – Sensors, Vol. **23**, 2023, No 12, 5732.
6. Popek, Ł., R. Perz, G. Galiński. Comparison of Different Methods of Animal Detection and Recognition on Thermal Camera Images. – Electronics, Vol. **12**, 2023, No 2, 270.
7. Jeantet, L., E. Dufourq. Improving Deep Learning Acoustic Classifiers with Contextual Information for Wildlife Monitoring. – Ecological Informatics, Vol. **77**, 2023, 102256.
8. Binta Islam, S., D. Valles, T. J. Hibbitts, W. A. Ryberg, D. K. Walkup, M. R. Forstner. Animal Species Recognition with Deep Convolutional Neural Networks from Ecological Camera Trap Images. – Animals, Vol. **13**, 2023, No 9, 1526.
9. Rančić, K., B. Blagojević, A. Bezdan, B. Ivošević, B. Tubić, M. Vranešević, B. Pejak, V. Crnojević, O. Marko. Animal Detection and Counting from UAV Images Using Convolutional Neural Networks. – Drones, Vol. **7**, 2023, No 3, 179.
10. Delplanque, A., S. Foucher, J. Théau, E. Bussière, C. Vermeulen, P. Lejeune. From Crowd to Herd Counting: How to Precisely Detect and Count African Mammals Using Aerial Imagery and Deep Learning? – ISPRS Journal of Photogrammetry and Remote Sensing, Vol. **197**, 2023, pp. 167-180.
11. Sharma, S., K. Sato, B. P. Gautam. A Methodological Literature Review of Acoustic Wildlife Monitoring Using Artificial Intelligence Tools and Techniques. – Sustainability, Vol. **15**, 2023, No 9, 7128.
12. Mou, C., A. Liang, C. Hu, F. Meng, B. Han, F. Xu. Monitoring Endangered and Rare Wildlife in the Field: A Foundation Deep Learning Model Integrating Human Knowledge for Incremental Recognition with Few Data and Low Cost. – Animals, Vol. **13**, 2023, No 20, 3168.
13. Ibraheam, M., K. F. Li, F. Gebali. An Accurate and Fast Animal Species Detection System for Embedded Devices. – IEEE Access, Vol. **11**, 2023, pp. 23462-23473.
14. Simões, F., C. Bouveyron, F. Precioso. DeepWILD: Wildlife Identification, Localisation and Estimation on Camera Trap Videos Using Deep Learning. – Ecological Informatics, Vol. **75**, 2023, 102095.
15. Roy, A. M., J. Bhaduri, T. Kumar, K. Raj. WilDect-YOLO: An Efficient and Robust Computer Vision-Based Accurate Object Localization Model for Automated Endangered Wildlife Detection. – Ecological Informatics, Vol. **75**, 2023, 101919.

16. R a v i, G., S. A f r o z, K. Y a m u n a, S. A f s h a. CNN Based Wildlife Intrusion Detection and Alert System. – International Transactions on Electrical Engineering and Computer Science, Vol. **2**, 2023, No 1, pp. 30-36.
17. S c h i n d l e r, F., V. S t e i n h a g e, S. T. van B e e c k C a l k o e n, M. H e u r i c h. Action Detection for Wildlife Monitoring with Camera Traps Based on Segmentation with Filtering of Trackless (SWIFT) and Mask-Guided Action Recognition (MAROON). – Applied Sciences, Vol. **14**, 2024, No 2, 514.
18. S a m r e e n, S., L. A k h i l a, R. R a v a l i, K. A k s h i t h a. Developing Hybrid Deep Neural Networks for Detecting the Movement of Wild Animals and Generating Alarm Messages. – International Journal of Computing and Artificial Intelligence, Vol. **5**, 2024, No 2, pp. 119-123.
19. V e l a s c o-M o n t e r o, D., J. F e r n á n d e z-B e r n i, R. C a r m o n a-G a l á n, A. S a n g l a s, F. P a l o m a r e s. Reliable and Efficient Integration of AI into Camera Traps for Smart Wildlife Monitoring Based on Continual Learning. – Ecological Informatics, Vol. **83**, 2024, 102815.
20. K a l b h o r, M., S. S h i n d e, D. E. P o p e s c u, D. J. H e m a n t h. Hybridization of Deep Learning Pre-Trained Models with Machine Learning Classifiers and Fuzzy Min-Max Neural Network for Cervical Cancer Diagnosis. – Diagnostics, Vol. **13**, 2023, No 7, 1363.
21. L a l i n i a, M., A. S a h a f i. Colorectal Polyp Detection in Colonoscopy Images Using YOLO-V8 Network. – Signal, Image, and Video Processing, Vol. **18**, 2024, No 3, pp. 2047-2058.
22. Z h a n g, Y., Z. L u, F. Z h a n g, H. W a n g, S. L i. Machine Unlearning by Reversing the Continual Learning. – Applied Sciences, Vol. **13**, 2023, No 16, 9341.
23. L a z z a r i n i, R., H. T i a n f i e l d, V. C h a r i s s i s. Federated Learning for IoT Intrusion Detection. – AI, 2023, No 4, pp. 509-530.
24. D a h o u, A., A. O. A s e e r i, A. M a b r o u k, R. A. I b r a h i m, M. A. A l-B e t a r, M. A. E l a z i z. Optimal Skin Cancer Detection Model Using Transfer Learning and Dynamic-Opposite Hunger Games Search. – Diagnostics, 2023, No 13, 1579.
25. K o l l u, V. N., V. J a n a r t h a n a n, M. K a r u p u s a m y, M. R a m a c h a n d r a n. Cloud-Based Smart Contract Analysis in FinTech Using IoT-Integrated Federated Learning in Intrusion Detection. – Data, 2023, No 8, 83.
26. Z h u, J., J. C a o, D. S a x e n a, S. J i a n g, H. F e r r a d i. Blockchain-Empowered Federated Learning: Challenges, Solutions, and Future Directions. – ACM Computing Surveys, Vol. **55**, 2023, No 11, pp. 1-31.

Received: 07.11.2024. Revised version: 11.12.2024, Accepted: 17.12.2024