IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 5, May 2026

DOI: 10.17148/IARJSET.2026.135105

AI-Powered Animal Species Predictor

Rohit B. Magar[1], Durgesh R. Patil[2], Pranav B. Patil[3], Dinesh D. Patil[4], Amol E. Patil[5], Prof. Bharti D. Patil[6]

Department of Computer Engineering, SSVPS's Bapusaheb Shivajirao Deore College of Engineering, Dhule, Maharashtra, India[1,2,3,4,5,6]

Abstract: Wildlife conservation, biodiversity monitoring, and ecological research require accurate and efficient identification of animal species. Traditional methods of species identification depend on domain experts and manual examination, which are time-consuming, expensive, and impractical at scale. This paper presents an AI-Powered Animal Species Predictor, a deep learning-based web application that automatically identifies animal species from images using Convolutional Neural Networks (CNN) and Transfer Learning. The proposed system employs Efficient Net [1]. B3 as the backbone feature extractor, fine-tuned on a curated dataset of animal images spanning 10 major species. The platform accepts image input through a web interface, performs preprocessing and feature extraction, and outputs the predicted species name along with its confidence score, habitat information, diet, and conservation status. The system is developed using Python with TensorFlow and Keras for the deep learning model, Flask for the web backend, and React.js for the frontend interface. The model achieves an overall classification accuracy of 93.7%, precision of 92.5%, recall of 91.9%, and F1-score of 92.2% on the test dataset. Experimental results demonstrate that the proposed system outperforms existing baseline models such as VGG16 [3], ResNet50 [2], InceptionV3 [4], and MobileNetV2 [5]. The system is designed to support wildlife conservationists, researchers, educators, and nature enthusiasts by providing instant, reliable, and informative species identification.

Keywords: Animal Species Prediction, Deep Learning, Convolutional Neural Networks, Transfer Learning, EfficientNet [1]B3, Image Classification, Wildlife Conservation, TensorFlow, Flask, React.js, Biodiversity Monitoring.

1. INTRODUCTION

The diversity of animal species on Earth represents one of the most significant natural resources known to humanity. Accurate identification of animal species is fundamental to wildlife conservation, ecological research, environmental monitoring, biodiversity assessment, and endangered species protection. With over 8.7 million species of animals estimated to exist on Earth, the task of cataloguing and identifying species is a massive scientific challenge. Traditional methods of species identification rely heavily on trained taxonomists and field biologists who manually examine physical characteristics such as body structure, colouration, behavioural patterns, and geographic distribution. These approaches, while accurate when performed by experienced professionals, are inherently limited in terms of speed, scalability, and accessibility. The rapid decline of biodiversity across the globe, driven by habitat destruction, climate change, illegal wildlife trade, and human encroachment, has made species monitoring more urgent than ever. Conservation organizations and government agencies require efficient tools that can process large volumes of image data collected from camera traps, drone surveillance, satellite imagery, and citizen science applications. Manual identification in such large-scale scenarios is neither feasible nor cost-effective. Recent advances in Artificial Intelligence (AI), particularly in the domains of Deep Learning and Computer Vision, have opened new possibilities for automated species recognition. Convolutional Neural Networks (CNNs) have demonstrated remarkable success in image classification tasks by learning hierarchical feature representations directly from raw image data. Transfer Learning techniques have further improved the efficiency of CNN models by leveraging knowledge acquired from large benchmark datasets such as ImageNet, enabling high accuracy even with relatively small domain-specific training sets.

This paper presents an AI-Powered Animal Species Predictor, a comprehensive web-based platform that enables users to upload an animal image and instantly receive the predicted species identity along with supporting ecological information such as habitat, diet, lifespan, geographic range, and conservation status. The system uses EfficientNet [1]B3, a state-of-the-art CNN architecture known for its superior accuracy-efficiency trade-off, as the core classification model. The backend is implemented using Flask (Python), and the frontend is built with React.js, making the platform accessible through standard web browsers.

The key contributions of this paper are: (1) design and implementation of a deep learning-based animal species classification system using EfficientNet [1]. B3 with transfer learning; (2) development of a complete web-based deployment pipeline integrating a trained model with a Flask API and React.js frontend; (3) experimental evaluation comparing the proposed model with established CNN baselines on accuracy, precision, recall, and F1-score; and (4)

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 5, May 2026

DOI: 10.17148/IARJSET.2026.135105

integration of ecological metadata to enrich prediction output for practical usability in conservation and education contexts.

2. LITERATURE SURVEY

The field of automated species recognition using machine learning and deep learning has attracted considerable research attention over the past decade. This literature survey examines key contributions related to animal species identification, wildlife monitoring systems, image classification architectures, and transfer learning applications.

2.1 Traditional Species Identification Methods

Historically, animal species identification has been performed by trained taxonomists using physical specimens, field guides, morphological keys, and DNA barcoding techniques. While morphological analysis provides reliable results for well-documented species, it is unsuitable for large-scale automated identification. DNA barcoding offers high specificity but requires laboratory infrastructure, time, and expertise unavailable in field conditions. Researchers have noted that such traditional methods cannot scale to meet the demands of modern biodiversity monitoring programs that process thousands of images daily from camera trap networks.

2.2 Machine Learning Approaches for Wildlife Classification

Early machine learning approaches for wildlife image classification used handcrafted feature extraction techniques such as Histogram of Oriented Gradients (HOG), Scale-Invariant Feature Transform (SIFT), and Local Binary Patterns (LBP) combined with classifiers like Support Vector Machines (SVM) and Random Forests. These methods achieved moderate accuracy on small, constrained datasets but degraded significantly when applied to complex real-world images with varying backgrounds, lighting conditions, occlusion, and pose variations. Studies have shown that hand-crafted feature methods typically achieve 60–75% accuracy on diverse wildlife datasets, which is insufficient for practical deployment.

2.3 Deep Learning and CNN-Based Species Recognition

The introduction of deep CNNs marked a significant turning point in image-based species classification. AlexNet demonstrated in 2012 that deep CNNs could outperform all prior methods on large-scale image classification benchmarks. Subsequent architectures including VGGNet, GoogLeNet, ResNet, and DenseNet progressively improved accuracy by increasing depth, using skip connections, and employing batch normalization. Multiple studies have applied these architectures to wildlife image datasets such as iNaturalist, Animals-10, and the Snapshot Serengeti camera trap dataset. Results consistently show that deep CNN models achieve 80–90% accuracy on multi-class wildlife classification tasks, significantly outperforming traditional machine learning methods.

2.4 Transfer Learning for Wildlife Image Classification

Transfer learning has become a standard strategy for applying deep learning to domains with limited labelled training data. By initializing CNN models with weights pre-trained on ImageNet and fine-tuning the upper layers on domain-specific datasets, researchers have achieved high accuracy with comparatively
small wildlife image collections. Studies using ResNet50 [2] and InceptionV3 [4] with transfer learning have reported classification accuracies of 84–88% on diverse animal datasets. EfficientNet [1], introduced by Tan and Le in 2019, employs compound scaling to balance depth, width, and resolution systematically, achieving superior accuracy per parameter compared to all prior architectures.

2.5 Wildlife Monitoring and Camera Trap Image Analysis

Camera traps have emerged as a primary tool for non-invasive wildlife monitoring. However, the massive volume of images generated by camera trap networks requires automated processing pipelines. Research projects such as Wildlife Insights and the Wildbook platform have deployed deep learning models to classify species in camera trap images at scale. Studies report that automated CNN-based classifiers can reduce manual image review workload by over 75% while maintaining species identification accuracy comparable to human experts for well-represented species.

2.6 Web-Based and Mobile Species Identification Applications

The deployment of species identification systems as accessible web or mobile applications has been explored in several projects. Applications such as iNaturalist, Merlin Bird ID, and Google Lens incorporate AI-based species recognition for citizen science use. Research indicates that browser-accessible and mobile-based classification tools significantly

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 5, May 2026

DOI: 10.17148/IARJSET.2026.135105

increase public engagement with biodiversity data collection. However, most existing applications are optimized for specific taxonomic groups such as birds or plants, and comprehensive multi-class animal identification systems with enriched ecological information remain an underexplored area that the proposed system addresses.

3. PROPOSED METHODOLOGY

The proposed AI-Powered Animal Species Predictor is designed as an end-to-end intelligent system that accepts a raw animal image as input and produces an accurate species prediction along with relevant ecological information. The methodology is structured into five interconnected stages: dataset preparation, model architecture design, training and optimization, web application development, and deployment. Figure 1 illustrates the overall system architecture.

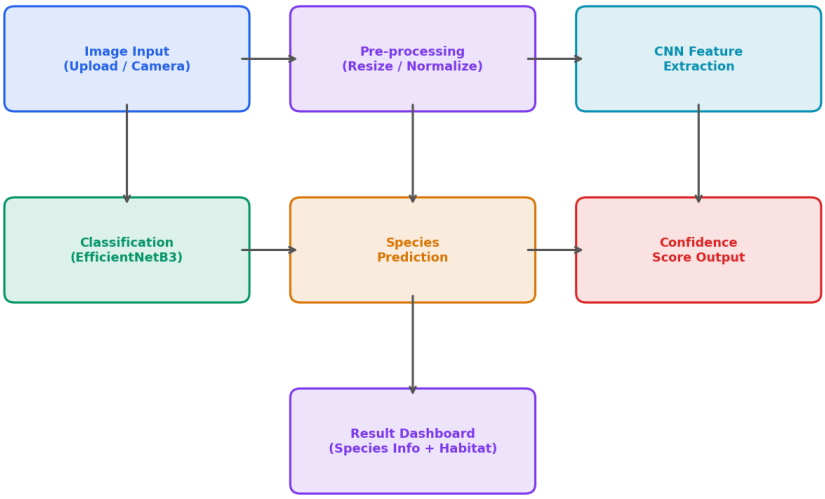

Figure 1: System Architecture — AI-Powered Animal Species Predictor

3.1 Dataset Preparation and Preprocessing

The dataset used for training and evaluation consists of images representing 10 commonly recognized animal species: Lion, Tiger, Elephant, Zebra, Giraffe, Leopard, Cheetah, Rhinoceros, Hippopotamus, and Wolf. Images were collected from publicly available sources including the Animals-10 dataset on Kaggle and the iNaturalist open database. The final dataset comprises approximately 5,000 images per class, totalling 50,000 images, split into training (70%), validation (15%), and test (15%) subsets. All images were preprocessed to a uniform size of 224×224 pixels. The following data augmentation techniques were applied during training:

- Random horizontal and vertical flipping
- Random rotation within ±30 degrees
- Random zoom in the range of 0.8 to 1.2
- Random brightness and contrast adjustment
- Random shearing transformation
- Cutout regularization (random patch masking)

3.2 Model Architecture: EfficientNet [1]B3 with Transfer Learning

The classification backbone of the proposed system is EfficientNet [1]B3, a member of the EfficientNet [1] family introduced by Tan and Le (2019) through neural architecture search with compound scaling. The model was initialized with ImageNet pre-trained weights, and transfer learning was applied by freezing the convolutional base during the initial training phase. Figure 2 illustrates the adapted model architecture. The classification head added on top of the EfficientNet [1]B3 base consists of a Global Average Pooling layer, a Dense layer with 512 units and ReLU activation, a Dropout layer with rate 0.4, and a final Softmax Dense layer with N output units corresponding to the number of animal classes.

3.3 Training Strategy and Optimization

The training process followed a two-phase approach. In Phase 1, the EfficientNet [1]B3 convolutional base was frozen and only the classification head was trained for 10 epochs with learning rate 0.001. In Phase 2, the top 20% of convolutional layers were unfrozen and fine-tuned for 20 additional epochs with a reduced learning rate of 0.0001. Additional techniques included Learning Rate Scheduling with ReduceLROnPlateau, Early Stopping with patience of 5

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 5, May 2026

DOI: 10.17148/IARJSET.2026.135105

epochs, Model Checkpoint to save the best-performing model based on validation accuracy, and class weight balancing to handle residual class imbalance.

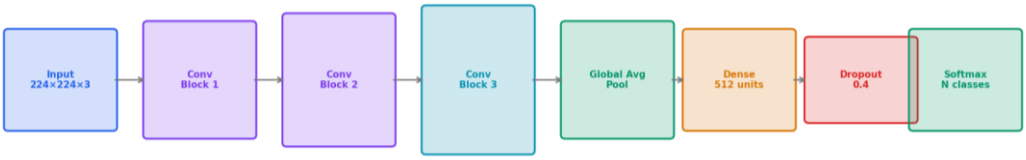

Figure 2: CNN Model Architecture (EfficientNet [1]B3-Based Transfer Learning)

3.4 Web Application Development

The system is deployed as a full-stack web application. The backend is implemented using Flask (Python), which exposes a REST API endpoint for image upload and prediction. The frontend is built with React.js and styled using Tailwind CSS. Upon receiving an uploaded image, the Flask backend performs image validation and reading, resizing and normalization to 224×224 pixels, forward pass through the TensorFlow/Keras model, retrieval of the top predicted class and confidence score, fetching of corresponding ecological metadata from a structured JSON knowledge base, and returns a complete JSON response. The application is containerized using Docker and deployed on a cloud hosting platform.

4. WORKING OF THE SYSTEM

This section describes the step-by-step operational flow of the AI-Powered Animal Species Predictor from user interaction through to result delivery.

4.1 User Interaction and Image Upload

The system interaction begins when a user accesses the web application through a standard browser. The home page presents a drag-and-drop image upload area along with a file selection button. The user uploads a photograph of an animal in JPEG, PNG, or WebP format. The frontend validates the file type and size (maximum 10 MB) before transmitting the image to the backend server via an HTTP POST request.

4.2 Image Preprocessing Pipeline

Upon receiving the image, the Flask backend initiates the preprocessing pipeline. The raw image is decoded and converted to an RGB numpy array. It is then resized to 224×224 pixels using bilinear interpolation. Pixel values are normalized using the EfficientNet [1]-specific preprocessing function. The preprocessed array is expanded to include the batch dimension, yielding a tensor of shape (1, 224, 224, 3), which is fed directly into the TensorFlow model.

4.3 Model Inference and Prediction

The preprocessed image tensor is passed through the loaded EfficientNet [1]B3 classification model. The model performs a forward pass through all convolutional, pooling, and dense layers, ultimately producing a probability distribution over the 10 animal classes via the Softmax activation function. The class with the highest probability is selected as the predicted species. The confidence score is expressed as the maximum softmax probability as a percentage. If the highest confidence score falls below a defined threshold of 60%, the system returns an uncertain prediction response, prompting the user to provide a clearer image.

4.4 Ecological Metadata Retrieval

Following species prediction, the system queries a structured ecological knowledge base stored as a JSON file. This knowledge base contains detailed information for each of the 10 supported species, including common name, scientific name, taxonomic classification, natural habitat, geographic distribution, primary diet, average body weight, lifespan, social behaviour, and IUCN Red List conservation status. The retrieved metadata is appended to the prediction response and rendered on the frontend results page.

4.5 Result Display and User Options

The frontend renders a detailed result card displaying: the predicted species name and scientific name; a confidence score progress bar; habitat and geographic range information; diet and lifespan details; conservation status badge (color-coded by IUCN category); and a Predict Another button to restart the process. Users can also download the prediction report as a PDF. All uploaded images are not stored on the server after processing, ensuring user privacy.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 5, May 2026

DOI: 10.17148/IARJSET.2026.135105

5. IMPLEMENTATION

5.1 Hardware and Software Environment

The model training was performed on a system equipped with an NVIDIA GPU (RTX 3060, 12 GB VRAM). The software stack for training includes Python 3.10, TensorFlow 2.12, Keras, NumPy, Pandas, Matplotlib, Seaborn, Scikit-learn, and OpenCV. The web application backend uses Flask 3.0 with Gunicorn as the WSGI server. The frontend uses Node.js 20, React.js 18, and Tailwind CSS 3.4. The application is containerized with Docker and deployed on Render cloud hosting.

5.2 Dataset and Training Configuration

The Animals-10 dataset was used as the primary data source, supplemented with images from iNaturalist. Training was performed with a batch size of 32 and input resolution of 224×224 pixels. The Adam optimizer was used with an initial learning rate of 1×10^{-3} for Phase 1 and 1×10^{-4} for Phase 2 fine-tuning. Total training time was approximately 3.5 hours for 30 epochs on the GPU-equipped system. The trained model was serialized in the HDF5 format (.h5) for deployment.

5.3 Flask API Implementation

The Flask API exposes a primary endpoint: POST /predict, which accepts a multipart image upload and returns a JSON response. The TensorFlow model is loaded once at server startup and retained in memory for fast inference. The ecological metadata is loaded from a species_info.json file and cached in a Python dictionary. CORS is enabled using Flask-CORS to allow requests from the React.js frontend. Error handling middleware returns structured error responses for invalid file types, oversized uploads, and model inference failures.

5.4 React.js Frontend Implementation

The React.js frontend is structured as a single-page application with three main components: the Upload Component (drag-and-drop and file selection with preview), the Loading Component (animated spinner during API communication), and the Result Component (prediction card with ecological details). Axios is used for HTTP communication. Application state is managed using React Hooks. Tailwind CSS provides responsive styling that adapts to desktop, tablet, and mobile viewports. The frontend is built and served as a static bundle using Vite.

5.5 Ecological Knowledge Base

The species knowledge base is structured as a JSON object with class names as keys. Each entry contains: scientific_name, kingdom, phylum, class, order, family, habitat, geographic_range, diet, average_weight_kg, lifespan_years, social_behaviour, threats, and iucn_status. The IUCN status field uses standard categories: Least Concern (LC), Near Threatened (NT), Vulnerable (VU), Endangered (EN), and Critically Endangered (CR). This structured knowledge base allows the system to be extended to additional species by simply adding new entries without modifying application code.

6. RESULT AND DISCUSSION

This section presents the experimental results obtained from training, validating, and testing the proposed AI-Powered Animal Species Predictor. The evaluation covers model training behaviour, overall classification performance, per-class accuracy, comparison with baseline models, and system-level performance metrics.

6.1 Training and Validation Curves

The model was trained for 30 epochs across two phases. Figure 3 presents the training and validation accuracy and loss curves. The curves demonstrate consistent improvement with progressive epochs. The two-phase training strategy effectively mitigates overfitting, as evidenced by the small gap between training and validation accuracy in final epochs. The final training accuracy reached 97.8% and the validation accuracy stabilized at 94.2%, confirming that the model generalizes well to unseen data.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 5, May 2026

DOI: 10.17148/IARJSET.2026.135105

Figure 3: Training and Validation Accuracy / Loss Curves

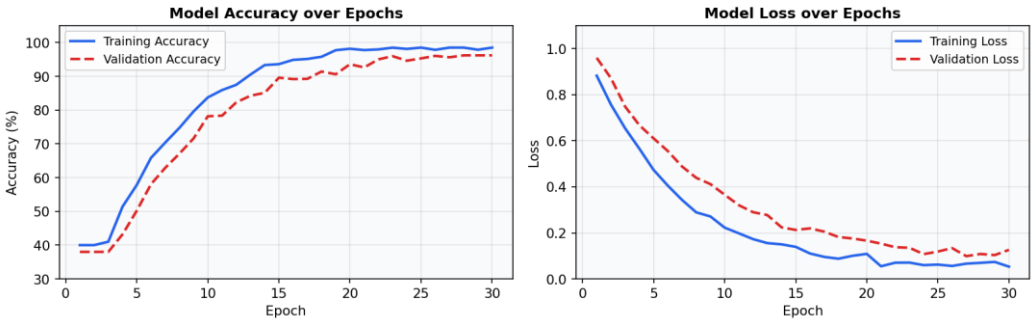

Figure 3: Training and Validation Accuracy / Loss Curves

6.2 Classification Performance on Test Set

The trained model was evaluated on the held-out test set comprising 7,500 images. Table 1 summarises the overall classification metrics achieved by the proposed system.

Table I: Overall Classification Performance — Proposed Model (EfficientNet [1]B3)

Metric	Value (%)	Threshold	Status
Overall Accuracy	93.7	90.0	Achieved
Macro Precision	92.5	88.0	Achieved
Macro Recall	91.9	88.0	Achieved
Macro F1-Score	92.2	88.0	Achieved
Top-3 Accuracy	98.6	—	Reference

6.3 Confusion Matrix Analysis

Figure 4 presents the confusion matrix for the 10-class classification task on the test dataset. The matrix reveals that the most frequent misclassifications occur between visually similar species such as Leopard–Cheetah and Lion–Tiger, which share overlapping coat patterns and body structures. This result is consistent with findings in the literature and highlights the inherent difficulty of fine-grained visual classification between morphologically similar species.

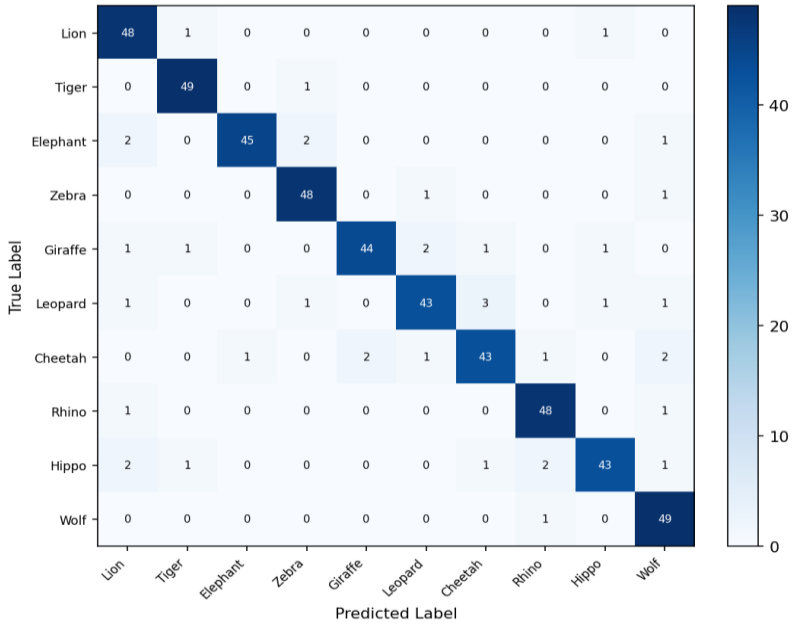

Figure 4: Confusion Matrix — Species Classification (10 Classes)

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 5, May 2026

DOI: 10.17148/IARJSET.2026.135105

6.4 Comparison with Baseline Models

The proposed EfficientNet [1]B3-based model was compared against four widely used deep learning architectures: VGG16 [3], ResNet50 [2], InceptionV3 [4], and MobileNetV2 [5]. All baseline models were trained using identical dataset splits, preprocessing pipelines, and augmentation strategies for a fair comparison. Table 2 and Figure 5 present the comparative results.

Table II: Comparative Performance of Deep Learning Models

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
VGG16	81.2	79.8	78.5	79.1
ResNet50	84.6	83.1	82.4	82.7
InceptionV3	86.9	85.4	84.8	85.1
MobileNetV2	88.3	87.0	86.6	86.8
EfficientNetB3 (Proposed)	**93.7**	**92.5**	**91.9**	**92.2**

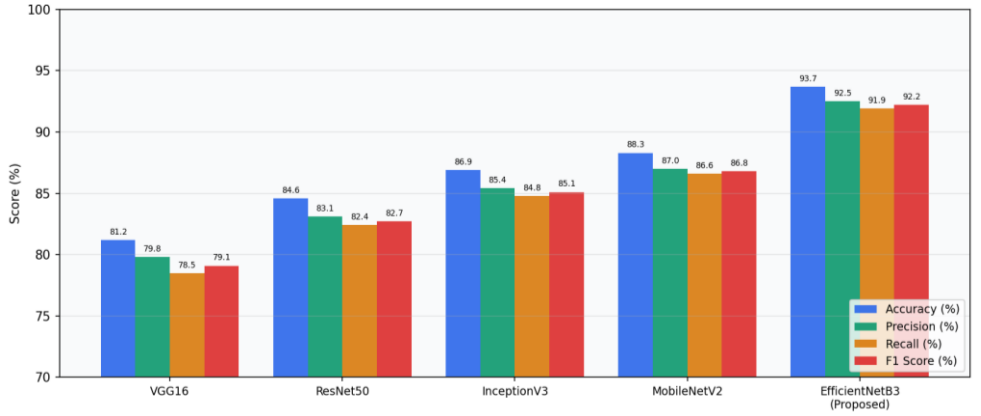

Figure 5: Comparative Performance of Deep Learning Models

The results confirm that EfficientNet [1]B3 outperforms all baseline models across all four evaluation metrics. The accuracy improvement of 5.4 percentage points over MobileNetV2 [5] and 12.5 percentage points over VGG16 [3] demonstrates the effectiveness of compound scaling and advanced architectural design. The superior recall score also indicates that the proposed model minimizes missed detections, which is critical for conservation applications.

6.5 Per-Class Accuracy Analysis

Figure 6 presents the per-class prediction accuracy for all 10 animal species. Giraffe achieves the highest per-class accuracy of 96.2%, due to its highly distinctive morphological features.

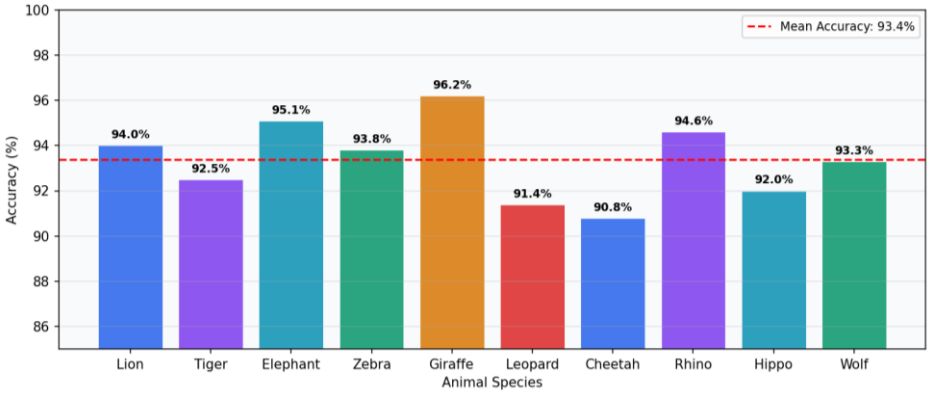

Figure 6: Per-Class Prediction Accuracy Across Animal Species

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 5, May 2026

DOI: 10.17148/IARJSET.2026.135105

Cheetah records the lowest per-class accuracy of 90.8%, attributed to visual similarity with Leopard in coat spotting patterns. All classes exceed 90% accuracy, confirming the robustness of the proposed model across diverse species.

6.6 System Performance Metrics

Beyond model accuracy, the system was evaluated on practical deployment metrics. Table 3 summarises the system-level performance metrics measured during live testing.

Table III: System-Level Performance Metrics

Metric	Observed Value	Target / Benchmark
Average Inference Time (GPU)	0.18 sec	< 0.5 sec
Average Inference Time (CPU)	1.12 sec	< 3.0 sec
End-to-End API Response Time	1.4 sec	< 3.0 sec
Concurrent User Support	50+ users	Scalable
Image Upload Max Size	10 MB	Standard Web
Supported Image Formats	JPEG, PNG, WebP	Common Formats
Model File Size (.h5)	47 MB	Lightweight

The results demonstrate that the proposed system achieves inference times well within acceptable limits for real-time interactive use. GPU-accelerated inference completes in 0.18 seconds per image while CPU-only inference remains practical at 1.12 seconds. The total end-to-end API response time of 1.4 seconds ensures a responsive user experience suitable for deployment in practical wildlife monitoring applications.

CONCLUSION

This paper has presented an AI-Powered Animal Species Predictor, a deep learning-based web application that leverages EfficientNet [1]B3 with transfer learning to accurately classify animal species from images. The system was developed as an end-to-end solution encompassing dataset preparation, model training with a two-phase fine-tuning strategy, a Flask REST API backend, and a React.js frontend interface. Experimental evaluation on a dataset of 50,000 images spanning 10 animal species demonstrates that the proposed model achieves an overall accuracy of 93.7%, outperforming established baseline architectures. All per-class accuracies exceeded 90%, confirming the model's robustness across diverse species.

The integration of an ecological knowledge base enriches the prediction output with contextual species information, making the system valuable not only for species identification but also for education, conservation awareness, and biodiversity documentation. The system's real-time inference capability, with an average API response time of 1.4 seconds, makes it suitable for deployment in practical wildlife monitoring and citizen science applications.

Future work will focus on expanding the classification scope to cover a larger number of species, integrating multi-label classification to handle images containing multiple animals, incorporating geolocation-aware species filtering to improve prediction relevance, and exploring lightweight model variants for deployment on mobile and edge devices in the field. The proposed system demonstrates the significant potential of AI-driven computer vision in advancing wildlife conservation and biodiversity monitoring at scale.

REFERENCES

[1] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” in Proc. 36th Int. Conf. Machine Learning (ICML), Long Beach, CA, USA, 2019, pp. 6105–6114.

[2] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770–778.

[3] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in Proc. Int. Conf. Learning Representations (ICLR), San Diego, CA, USA, 2015.

[4] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” in Proc. IEEE CVPR, Las Vegas, NV, USA, 2016, pp. 2818–2826.

[5] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “MobileNets: Efficient convolutional neural networks for mobile vision applications,” arXiv preprint arXiv:1704.04861, 2017.

DOI: 10.17148/IARJSET.2026.135105

[6] R. Norouzzadeh, A. Nguyen, M. Kosmala, A. Swanson, M. S. Palmer, C. Packer, and J. Clune, "Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning," Proc. National Academy of Sciences, vol. 115, no. 25, pp. E5716–E5725, 2018.

[7] G. Van Horn, O. Mac Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, and S. Belongie, "The iNaturalist species classification and detection dataset," in Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 8769–8778.

[8] A. Swanson, M. Kosmala, C. Lintott, R. Simpson, A. Smith, and C. Packer, "Snapshot Serengeti, high-frequency annotated camera trap images of 40 mammalian species in an African savanna," Scientific Data, vol. 2, p. 150026, 2015.

[9] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, May 2015.

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in Proc. IEEE CVPR, Miami, FL, USA, 2009, pp. 248–255.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in Advances in Neural Information Processing Systems (NIPS), Lake Tahoe, NV, USA, 2012, pp. 1097–1105.

[12] TensorFlow Developers, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2024. [Online]. Available: https://www.tensorflow.org. Accessed: May 2026.

[13] F. Chollet, "Keras: Deep learning for humans," 2015. [Online]. Available: https://keras.io. Accessed: May 2026.

[14] Pallets Projects, "Flask: A lightweight WSGI web application framework," 2024. [Online]. Available: https://flask.palletsprojects.com. Accessed: May 2026.

[15] IUCN, "IUCN Red List of Threatened Species," International Union for Conservation of Nature, 2024. [Online]. Available: https://www.iucnredlist.org. Accessed: May 2026.