Ecological Informatics 90 (2025) 103193

Contents lists available at ScienceDirect

Ecological Informatics

journal homepage: www.elsevier.com/locate/ecolinf

BirdRecon: A free open source tool for image based bird species recognition

Hari Kishan Kondaveeti [a], Nabin Kumar Upadhaya [a], Dheeraj Sai Tukkugudam [a], Rahul Panigrahi [a], Sirivella Madhan Chandra Mouli [a], Valli Kumari Vatsavayi [b], Nagendra Panini Challa [a,*]

[a] School of Computer Science & Engineering, VIT-AP University, Amaravathi, 522237, Andhra Pradesh, India

[b] Department of Computer Science & Systems Engineering, AUCE (A), Andhra University, Visakhapatnam, 530003, Andhra Pradesh, India

ARTICLE INFO

Dataset link: https://gts.ai/392dataset-download/birds-525-species-image-classification/

Keywords:
Bird species recognition
Deep ensemble learning
Open source tool
Mobile application
Web application
Google gemini

ABSTRACT

Automated bird species recognition is a critical, yet challenging task, particularly for the systems aimed at supporting ornithologists, conservationists, and bird enthusiasts. This study introduces BirdRecon, an open-source bird species recognition system developed to enhance birdwatching, ornithological research, and biodiversity conservation. The system leverages a soft voting ensemble of four pretrained deep learning models—DenseNet201, EfficientNetB7, InceptionV3, and ResNet50V2—to improve classification accuracy and robustness. To address class imbalance problem and enhance generalization, data augmentation is applied and an early stopping optimization strategy is used to prevent overfitting during training. A benchmark dataset comprising 525 bird species with over 84,000 training images is used to evaluate the system. The experimental results demonstrate that the proposed ensemble model achieves a classification accuracy of 99.6%, precision of 99.7%, and recall of 99.6%, outperforming the existing state-of-the-art methods by a margin of 0.51%.

BirdRecon is implemented as both a web and mobile application, offering real-time bird species identification with multilingual support (English, Hindi, and Telugu) and additional features, such as species descriptions through Google Gemini and visual references from Wikimedia Commons. The open-source nature of the system, available on GitHub, promotes collaboration and further advancements. With its user-friendly design and practical deployment capability on resource-constrained devices, BirdRecon serves as a valuable tool for researchers, conservationists, and birdwatchers, contributing to biodiversity monitoring and conservation efforts.

1. Introduction

Birds are vital to ecosystems worldwide and perform essential functions such as pollination, seed dispersal, pest control, and nutrient recycling. Birdwatching promotes mental well-being by reducing stress, improving mindfulness, fostering connection with nature, improving mood and focus, and offering relaxation and a sense of awe through the observation of wildlife (Randler et al., 2023; Grimalt et al., 2023; Peterson et al., 2024). Accurate identification of bird species is critical for conserving biodiversity and monitoring the environment, as it allows researchers to track changes in bird populations and understand how external factors such as climate change impact them Kati and Sekercioglu (2006) and Sullivan et al. (2009). Proper identification supports various activities, including monitoring environmental conditions, studying migration patterns to assess climate variability, and cataloging species to assess biodiversity (Bibby, 1999; Charmantier and Gienapp, 2014; Gregory et al., 2003).

Traditional bird identification methods are based on manual observations and expert knowledge, making them time-consuming and resource-intensive (Jasim et al., 2022). The wide diversity of species, variations within species, and similarities between different species complicate the identification process. In addition, environmental factors, such as limited visibility and time constraints, can hinder effective bird observations (Nichols and Williams, 2006). Advances in technology have led to the development of numerous automated bird species classification methods that aim to improve both accuracy and efficiency, while reducing time, effort, and costs (Kahl et al., 2021; Kumar et al., 2023). Automated recognition systems use modern Artificial Intelligence (AI) technologies to enhance scientific research, engage the public (Green et al., 2020; Fraisl et al., 2024), and play a crucial role in monitoring and protecting threatened and endangered species (Kwok et al., 2019; Fang et al., 2019; Ullah et al., 2024), ultimately promoting our understanding of species diversity.

* Corresponding author.
E-mail address: nagendra.challa@vitap.ac.in (N.P. Challa).

https://doi.org/10.1016/j.ecoinf.2025.103193
Received 26 June 2024; Received in revised form 30 March 2025; Accepted 6 May 2025
Available online 26 May 2025

Visual feature-based bird classification methods identify bird species from images captured remotely using monitoring equipment such as camera traps and mobile cameras (Edney and Wood, 2021). Deep learning, particularly Convolutional Neural Networks (CNNs), have been used in visual recognition tasks by learning features directly from data (Qin et al., 2020). Huang and Basanta (2019) developed a skipped CNN model achieved 100% accuracy in identifying 27 endemic Taiwanese birds. Skip connections in CNNs act as shortcuts, allowing information to bypass layers and ensuring an uninterrupted data flow. They solve vanishing gradient problems, enabling better learning by directly linking layers. By combining early and deep features, they enhanced pattern recognition and improved performance.

Transfer learning has improved bird species recognition by using pretrained convolutional neural network (CNN) models, saving time and improving performance (Wu et al., 2021; Kumar et al., 2022; Kondaveeti et al., 2023). Huang and Basanta (2021) achieved an accuracy of 98.39% in detecting endemic birds using the InceptionResNetV2 model. Kumar and Kondaveerti (2023) fine-tuned various pre-trained models on the same dataset and achieved an accuracy of 87.13% with the EfficientNet model. Cai (2023) used EfficientNetB0 and achieved an accuracy of 87% to classify images of 525 bird species. Vo et al. (2023) employed the YOLOv5 object detection algorithm along with deep transfer learning models, including VGG19, Inception V3, and EfficientNetB3, to develop an efficient system for bird detection and species classification, achieving 98% accuracy for both tasks. Mochurad and Svystovych (2024) used the EfficientNetB5 model and achieved an accuracy of 98.86%. SS et al. (2024) used ResNet50, MobileNetV3, EfficientNetB0, and Wide-ResNet50V2, and EfficientNetB0 achieved the highest accuracy of 99.09%. Kumar and Kondaveeti (2024) proposed a method that combines transfer learning with a hybrid hyperparameter optimization scheme, achieving an accuracy of 99.12% with EfficientNetB0.

Fine-Grained Image Classification (FGIC) methods help recognize subtle differences between similar bird species using techniques such as local part learning, discriminative features, and deep learning. These techniques extract specific patterns, such as feather textures and colors, to improve accuracy. FGIC methods such as Mask-CNN (Wei et al., 2018) and attention-guided augmentation (Wang et al., 2023) have demonstrated strong performance, achieving 85%–87% accuracy. Ferreira et al. (2020) used a Raspberry Pi-based bird monitoring system to collect images of three bird species and achieved 90% accuracy.

Together, the above studies attempted to improve accuracy, scalability, and efficiency. However, despite achieving good performance, the single models used in the above studies often face challenges, such as overfitting, bias, or high variance, which can lead to unreliable predictions (Pichler and Hartig, 2023). Ensemble deep learning addresses these issues by combining multiple models, thereby improving predictive accuracy and generalization and providing more reliable results (Ganaie et al., 2022; Mohammed and Kora, 2023).

To address the limitations of relying on single models, the proposed methodology offers an effective solution by integrating multiple models using the ensemble technique. We considered four pre-trained models, DenseNet201, EfficientNetB7, InceptionV3, and ResNet50V2, and used a soft voting ensemble technique to improve recognition accuracy. This approach allows models to take advantage of their individual strengths, leading to enhanced recognition accuracy. By averaging the confidence scores generated by each model, the ensemble produced a final prediction that was more robust than relying on the output of a single model.

Several researchers have introduced mobile and web applications for automatic bird species recognition. For instance, Gupta et al. (2021) developed a mobile application that achieved 98.61% offline accuracy, whereas Hermadi et al. (2022) created a web-based system using TensorFlow.js that achieved 97% accuracy. Other applications, such as the Birds Predictor (Huang and Basanta, 2019), Internet of Birds (Huang and Basanta, 2019), and BirdGuide (De Silva et al., 2019), have also shown promising performances ranging from 97% to 99%. Several popular applications have emerged in this field, including Merlin Bird ID (stagmerlinapp2, 2024), BirdNet (2024), eBird (2023), iNaturalist (2024), and Birda (2023). However, despite their benefits, these applications have several limitations. They often depend on a specific list of species and the absence of publicly available codes restricts community collaboration and further development. Furthermore, these applications lack effective user feedback mechanisms and do not integrate latest AI technologies.

Our work also aimed to develop an open-source practical bird species recognition system with a user-friendly design, focused on real-world applications. Unlike other tools, our code is freely available on GitHub, making it accessible to researchers and bird enthusiasts, even to those with limited coding skills. This openness promotes collaboration and provides constructive feedback to enhance the system. A comparison of the five bird recognition applications in Table 1 emphasizes the unique features of BirdRecon. Our application does not require users to download data from specific areas. Instead, it can be adapted to recognize a wide range of bird species locally/globally, conserving device storage and is ideal for users with limited space or frequent travelers.

1.1. Potential contributions

The bird identification system developed in this study makes several notable contributions to this field.

- Development of BirdRecon, an open source bird species recognition system designed to enhance birdwatching and ornithological research.
- BirdRecon features web and Android applications with real-time feedback and intuitive interfaces.
- The developed system depends on an ensemble approach, combining multiple pre-trained models (DenseNet201, EfficientNetB7, InceptionV3, ResNet50V2) through soft voting to improve the accuracy of recognition.
- The system features a user-friendly design focused on real-world applications, which makes it easy for users to identify bird species.
- The system can be adapted to recognize a wide range of bird species, eliminating the need for multiple downloads and conserving device storage.
- The system provides real-time identification of bird species, supporting multiple languages (English, Hindi, and Telugu) and making it an invaluable resource for users in diverse and biodiverse regions.
- The system offers comprehensive informational resources such as Google Gemini (Team et al., 2023) and Wikimedia Commons (Wikimedia Commons, 2023), enriching the user experience by providing additional information on the bird species encountered.
- The code is freely available on GitHub, making it accessible to researchers and bird enthusiasts, even those with limited coding skills. This openness promotes collaboration and constructive feedback to enhance the system.
- The system encourages collaboration and community involvement, allowing users to contribute to the development and improvement of the system.

2. Proposed methodology

2.1. Dataset description

The dataset employed for bird species classification experiments was sourced from the AI dataset collection (GTS, 2025). This dataset contains images of 525 different bird species organized into training, validation, and test sets, comprising 84,635 training images, 2625

Table 1
Comparison of bird recognition applications (Specific features related to the application are highlighted in **bold text**).

Features	Merlin	BirdNet	iNaturalist	Birda	BirdRecon (Our Application)
Support images/audio/video	Images	Images, Audio	Images, Audio, Video	Images	Images
Species description	Yes	Yes	Yes	Yes	Yes
Multi-language support	Yes	Yes	Yes	Yes	Yes
Need additional location-based data	Yes	Yes	Yes	No	No
Public availability of Code	No	No	No	No	Yes
Customizable	No	No	No	No	Yes
Google Gemini integration	No	No	No	No	Yes
Wikimedia Commons integration	No	No	No	No	Yes
Species coverage	North America, Europe, South America	North America, Europe, Asia	Global	Global	**Customizable (local/global)**

validation images, and 2625 test images. To address the class imbalance problem, a data augmentation technique was employed to artificially increase the diversity of the underrepresented classes in the training dataset.

2.2. Data augmentation

To improve the ability of the model to generalize and achieve a better performance, data augmentation techniques were applied to improve the training dataset. These techniques include the use of random shearing to distort the images slightly, zooming in or out to vary the image scales, and horizontal flipping, which was implemented with a probability of 100% to ensure consistent variation. In addition, rotation transformations were applied to the images, allowing rotations up to 20° in either direction. Through these methods, the training data were augmented with different variations, helping the model to better adapt to different scenarios and improve its overall robustness during the learning process.

2.3. Transfer learning and model customization

Four pre-trained models, namely DenseNet201, EfficientNetB7, InceptionV3, and ResNet50V2, were used for the classification of bird species, leveraging their existing capabilities in feature extraction, which are based on the ImageNet dataset. These four pre-trained models were employed to classify bird species, taking advantage of their ImageNet-based feature extraction capabilities to achieve this goal.

2.3.1. Custom layer addition

Transfer learning was implemented by enhancing each pretrained model with new layers. By introducing the following layers, the pre-trained model was fine-tuned for the bird species classification task, adapting the learned features to a specific dataset and improving predictive accuracy.

GlobalAveragePooling2D Layer: This layer was added to reduce the spatial dimensions of the feature map by condensing them into a single vector for each feature map. It effectively reduces the number of parameters, optimizes computational efficiency, and prepares the extracted features for the classification layers.

- Dense Layer with Softmax Activation: A dense layer is incorporated to act as the classification layer. It was specifically designed to produce a probability distribution across 525 classes of bird species, ensuring that the final predictions were interpretable as confidence scores for each class. These probability scores play a crucial role in ensemble learning using soft voting. The ensemble averages the confidence scores of each model to make a final prediction.

2.4. Model training

Each model underwent training for a maximum of 15 epochs, using a batch size of 32. The Stochastic Gradient Descent (SGD) optimizer is employed because of its efficiency in handling large datasets, whereas the categorical cross-entropy loss function is utilized because it is well-suited for multiclass classification tasks. To prevent overfitting and conserve computational resources, an early stopping mechanism was implemented, halting the training if the validation performance did not improve. During training, individual model accuracies were refined, and an ensemble approach using soft voting combined the predictions of the four pre-trained models, leading to enhanced classification performance and robustness.

2.5. Soft voting ensemble

The soft voting ensemble combines probability predictions from multiple models by averaging the predicted probabilities for each class. The probability of ensemble class j is calculated by averaging the predictions of all models, as shown in Eq. (1), and the final prediction of the ensemble is given by Eq. (2).

$$P_{ensemble}(class_j) = \frac{1}{N}\sum_{i=1}^{N} P_i(class_j) \tag{1}$$

$$\text{Final prediction} = \arg\max_j \left[P_{ensemble}(class_j)\right] \tag{2}$$

where:

$P_{ensemble}(class_j)$ = final probability of ensemble for class j
N = number of models in the ensemble
$P_i(class_j)$ = probability predicted by model i for class j

2.6. Model deployment

For practical deployment, the trained models are saved in the HDF5 format, which simplifies further development and retraining. In addition, the models were converted and saved in TensorFlow Lite format to allow efficient deployment on edge devices. This ensures that the software can be utilized in environments with limited computational resources. This strategic approach increases the accessibility and usability of the application, enabling users to take advantage of bird recognition capabilities on a variety of devices without the need for extensive computational power.

2.7. User interface development

We developed a user-friendly web and mobile interface that allows users to interact effectively with our system. We enabled users to upload bird images, a file-upload option available on web applications. Furthermore, our mobile application allows users to capture bird images using smartphone cameras. Our mobile application includes features such as camera integration, gallery selection, and image preprocessing before sending data to the back-end server.

3. Experimental setup

This study used a benchmark dataset (GTS, 2025) that included images of 525 different bird species. This dataset was categorized into three groups: 84,635 training images, 2625 validations, and 2625 test images. Each image was resize 224 × 224 × 224 pixels and saved in JPEG format. To ensure a balanced representation, five images per species are included in both the validation and test sets. The number of training images per species varied from 130 to 263, thereby providing a diverse set of examples. To address class imbalance, we applied data augmentation techniques such as random shearing, zooming, flipping, and rotation, increasing diversity in underrepresented classes and reducing bias. In addition, the images were rescaled to normalize the pixel values within the range of [0, 1].

Our experiments were conducted using Google Colab Pro, a cloud-based environment that provides an efficient platform for executing Python code. The system configuration used in this study was based on Microsoft Windows 11 Home (version 10.0.26100), manufactured by Microsoft Corporation, configured as a standalone workstation equipped with an Intel64 family microprocessor. In terms of runtime configuration within Google Colab Pro, we leveraged an NVIDIA Tesla T4 GPU with 12 GB of RAM to enhance processing capabilities, making it well-suited for deep learning tasks.

We executed our experiments using Python and employed various libraries and packages to facilitate our experiments, including TensorFlow, a popular open-source machine learning library used for building and training deep learning models; Keras, a high-level neural network API that can run on top of TensorFlow, providing an easy-to-use interface for building and training deep learning models; NumPy, a library for efficient numerical computation in Python, providing support for large, multidimensional arrays and matrices; pandas, a library for data manipulation and analysis in Python, providing data structures and functions for efficiently handling structured data; and Flask, a micro web framework for Python, which was used to build the backend server for mobile applications. Additional relevant libraries, such as OpenCV and Matplotlib, have also been incorporated to support specific tasks, such as image processing. For transparency and reproducibility, we created a complete code for the mobile application and backend server accessible through a GitHub repository. This allows other researchers to replicate our experiments and build upon the existing work.

We used the following hyperparameters during the analysis: the batch size was set to 16 and the number of epochs was limited to 15, with early stopping criteria based on validation loss to prevent overfitting. In addition, we applied data augmentation techniques including random shearing, zooming, horizontal flipping, and rotations to enhance the robustness of the training dataset.

The experimental analysis involved evaluating the performance of four individual deep learning models, DenseNet201, EfficientNetB7, InceptionV3, and ResNet50V2, along with various ensemble combinations of these models. The purpose was to determine the strength of using ensemble methods to improve the accuracy and overall performance metrics of individual models. Here are the descriptions of each performance metric, along with their ranges of values and formulae.

Accuracy: Measures the proportion of correctly classified instances out of all instances in the dataset.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Precision: Measures the proportion of true positives out of all positive predictions made by the model.

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall: Measures the proportion of true positives out of all actual positive instances in the dataset.

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1 Score: The harmonic mean of precision and recall provides a balanced measure of both metrics.

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

In the above formulae, we used True Positives (TP) to denote correctly predicted instances, True Negatives (TN) for correctly predicted non-instances, False Positives (FP) for incorrectly predicted instances, and False Negatives (FN) for incorrectly predicted non-instances, and n represents the number of classes in our multiclass classification setup. All measures range from 0 to 1 (or 0% to 100%), where 1 (or 100%) indicates perfect performance, and 0 indicates complete failure.

4. Result analysis

Initially, the individual models were trained and tested on the dataset, and their performance was evaluated using accuracy, precision, recall, and F1-score metrics. Subsequently, 2-model, 3-model, and 4-model ensemble combinations are implemented. For 2-model ensembles, combinations such as DenseNet201 + EfficientNetB7, InceptionV3 + ResNet50V2, and EfficientNetB7 + ResNet50V2 were evaluated. For 3-model ensembles, combinations such as DenseNet201 + EfficientNetB7 + InceptionV3 and EfficientNetB7 + InceptionV3 + ResNet50V2 were tested. Finally, an ensemble of all four models (4-Model Ensemble) was created to maximize synergy and improve robustness.

The ensemble methods used soft voting (averaging the probability outputs of the models) to classify test samples. Performance metrics were calculated similarly for these ensembles to assess their effectiveness compared with the standalone models. Table 2 compares the performance metrics of the selected pretrained models and their ensemble combinations.

Individual model performance

Among the individual models, EfficientNetB7 achieved the best performance with an accuracy of 98.3%, precision of 97.7%, recall of 97.4%, and F1-score of 97.3%. This was followed by DenseNet201 and ResNet50V2, which exhibited identical performance metrics with accuracies of 97.1% and 97.2%, respectively. InceptionV3 displayed the lowest performance with an accuracy of 96.3%.

2-Model ensemble performance

The 2-model Ensemble combinations showed an improvement in performance over the individual models, benefiting from the complementary strengths of the paired models. For example, DenseNet201 + EfficientNetB7 and EfficientNetB7 + ResNet50V2 both achieved accuracies of 97. 5% and an F1 score of 96. 7%. However, the combination of InceptionV3 and ResNet50V2 was 96%. The 5% accuracy did not exceed that of superior individual models. The decrease in accuracy can be due to the soft voting process, where the predictions of weaker models overpower the strong model-confident predictions. This leads to less accurate decision making. Other factors contributing to the decline may include disagreements between models, the negative impact of combining diverse models, and the equal weighting of all models. Further research is required to fully understand these results.

3-Model ensemble performance

The 3-model ensembles stabilized the performance across metrics, with combinations such as DenseNet201 + EfficientNetB7 + InceptionV3 and EfficientNetB7 + InceptionV3 + ResNet50V2 achieving accuracies of 97.0%. Although this represented a slight improvement over the individual models, these ensembles did not provide a significant boost, with performance remaining consistent with the averages of their constituent models.

Table 2
Performance metrics of pre-trained models and their ensembles on benchmark dataset (GTS, 2025).

Model	Accuracy	Precision	Recall	F1-Score
DenseNet201	97.1%	96.5%	96.3%	96.3%
EfficientNetB7	98.3%	97.7%	97.4%	97.2%
InceptionV3	96.3%	95.6%	95.8%	95.4%
ResNet50V2	97.2%	96.9%	96.7%	96.8%
2-Model Ensemble (DenseNet201 + EfficientNetB7)	97.5%	97.0%	96.5%	96.7%
2-Model Ensemble (InceptionV3 + ResNet50V2)	96.5%	96.0%	95.5%	95.7%
2-Model Ensemble (EfficientNetB7 + ResNet50V2)	97.5%	97.0%	96.5%	96.7%
3-Model Ensemble (DenseNet201 + EfficientNetB7 + InceptionV3)	97.0%	96.5%	96.0%	96.2%
3-Model Ensemble (EfficientNetB7 + InceptionV3 + ResNet50V2)	97.0%	96.5%	96.0%	96.2%
3-Model Ensemble (DenseNet201 + InceptionV3 + ResNet50V2)	96.7%	96.2%	95.7%	95.9%
4-Model Ensemble (All Models) - Proposed solution	**99.6%**	**99.7%**	**99.6%**	**99.6%**

Table 3
Comparative analysis of performances of existing studies with proposed method on benchmark dataset (GTS, 2025).

Model	Accuracy (%)
EfficientNetB0 (Cai, 2023)	89.0
CNN with Skip Connections, VGG16 (Farman et al., 2023)	92.0
Weighted Average Ensemble (EfficientNetB5 + VGG19) (Polisetty and Chokkalingam, 2024)	97.18
YOLOv5+EfficientNetB3 (Vo et al., 2023)	98.0
EfficientNetB5 (Mochurad and Svystovych, 2024)	98.86
ResNet50, MobileNetV3, EfficientNetB0, ResNet50V2 (SS et al., 2024)	99.09
Proposed solution	**99.60**

4-Model ensemble performance

The 4-Model ensemble (DenseNet201 + EfficientNetB7 + InceptionV3 + ResNet50V2) achieved the best definitive results, with an accuracy of 99.6%, a precision of 99.7%, a recall of 99.6%, and an F1-score of 99.6%. This ensemble demonstrates the effective synergistic capabilities of combining diverse architectures, yielding significant performance improvements over individual models and smaller ensemble combinations.

Performance comparison with existing studies

We performed a comparative analysis of the performance of the proposed solution with several notable studies such as EfficientNetB0 (Cai, 2023), CNN with Skip Connections and VGG16 (Farman et al., 2023), Weighted Average Ensemble (EfficientNetB5 + VGG19) (Polisetty and Chokkalingam, 2024), YOLOv5 + EfficientNetB3 (Vo et al., 2023), EfficientNetB5 (Mochurad and Svystovych, 2024), individual models such as ResNet50, MobileNetV3, EfficientNetB0, and ResNet50V2 (SS et al., 2024), and the proposed solution. The results demonstrated a clear progression in accuracy as more advanced techniques, architectures, and ensemble methods were employed. Based on the accuracy comparison shown in Table 3, the proposed solution achieves superior performance with an accuracy of 99.60%, outperforming all the existing approaches. This comparison demonstrates that the proposed solution offers a notable improvement over the existing methods, with a margin of 0.51% over the next best-performing model.

EfficientNetB0 achieves an accuracy of 89.0%, which is relatively lower than that of the other models (Cai, 2023). This result reflects the limitations of using a lightweight model, such as EfficientNetB0, for complex tasks, such as bird species classification, where high accuracy is critical. CNN with skip connections and VGG16 improved this, achieving an accuracy of 92.0% (Farman et al., 2023). The inclusion of skip connections improves the ability of the model to capture features; however, it still falls short of the performance achieved by more sophisticated ensemble and transfer learning approaches.

The weighted average ensemble, which combines EfficientNetB5 and VGG19, significantly increases the accuracy to 97. 18% (Polisetty and Chokkalingam, 2024). This result highlights the effectiveness of ensemble learning in leveraging the strengths of multiple models to improve classification performance. Similarly, the combination of YOLOv5 and EfficientNetB3 achieved an accuracy of 98.0% (Vo et al., 2023), demonstrating the importance of integrating object detection and classification techniques for tasks involving visually complex datasets, such as bird species. EfficientNetB5, which is a more advanced version of the EfficientNet family, achieved an accuracy of 98. 86% (Mochurad and Svystovych, 2024), thereby demonstrating its superior feature extraction capabilities. In the study by SS et al. (2024), individual models including ResNet50, MobileNetV3, EfficientNetB0, and ResNet50V2 were evaluated separately, with the highest accuracy among them reaching 99.09%.

However, the proposed solution achieves the highest accuracy of 99.60%, outperforming these models. This performance is attributed to the integration of techniques, including data augmentation, ensemble learning with soft voting, and optimization strategies such as early stopping.

5. Conclusion, privacy considerations, and future directions

The bird species recognition system developed in this project is available as a web application and a mobile application, offering significant capabilities that improve ornithological research and improve bird watcher experience. This system provides real-time identification of bird species and supports multiple languages, making it an invaluable resource for users in diverse and biodiverse regions like India. Using user-friendly interfaces and comprehensive informational resources, this system effectively bridges the gap between amateur birdwatchers and professional researchers. Using data from Google Gemini and visually similar images from Wikimedia Commons, the system enriches the user experience by providing additional information on bird species encountered, fostering a deeper engagement with avian biodiversity.

This study represents a foundational step in the development of the BirdRecon platform for the bird species identification. At this stage, the application is still under development and is not currently available for public use. However, the code for the project is open-sourced, enabling fellow researchers to use the existing functionality and update it according to their specific requirements. This allows researchers to leverage foundational work without needing to develop a similar platform from scratch, thereby fostering collaboration and innovation within the community.

To improve classification accuracy, the system could benefit from integrating user-provided location and habitat information. Considering data such as GPS coordinates, specific habitat types (e.g. forest, wetland, and urban area), and contextual factors such as weather conditions can help narrow down potential species matches. In addition, users can provide details on the date and time of their sightings, bird behavior, and physical characteristics, including estimated size, predominant colors, distinctive markings, and even the option to provide multiple images from different angles. This comprehensive approach to data collection would enable the system to make more informed decisions, ultimately leading to greater precision in bird identification.

However, the system has several limitations that need to be addressed to reach its full potential. Identifying bird species is particularly challenging under poor lighting and adverse weather conditions. This limitation can be mitigated by implementing robust data augmentation techniques that simulate such conditions, thereby improving the reliability and accuracy of the model. However, it is important to note that expecting the software to perform accurately under extreme conditions may not be entirely realistic, as bird species identification, particularly when differentiating subspecies, relies heavily on distinguishing fine color patterns on body parts. These subtle color variations are critical for the final decision. Currently, although the Android application is functional, the web platform is still under development and there is no version available for iOS devices. This restriction limits accessibility for iOS users and other mobile platforms. In addition, the system relies solely on visual identification and currently does not include the ability to recognize bird sounds, which is essential for accurate identification in some situations. Although the system supports a few languages, this restricts its reach, preventing many potential users around the world from accessing the system. Furthermore, the system was not initially designed as a commercial product, raising concerns regarding its long-term sustainability and widespread use.

Several future directions can improve the functionality and effectiveness of this system. Developing iOS and other mobile versions would greatly expand accessibility to a wider audience. Adding bird sound recognition features will create a more robust identification solution, while supporting a broader range of international languages will foster inclusivity and attract users from various linguistic backgrounds. Exploring commercialization opportunities or partnerships could help ensure the long-term maintenance and growth of this application.

Incorporating Explainable Artificial Intelligence (XAI) techniques can also help users better understand how the system makes identification decisions. By clarifying the characteristics that influence the identification results, users can develop greater confidence in the system. Employing techniques that focus on identifying subtle differences between similar bird species can improve the accuracy of the application, making it easier for users to distinguish closely related birds.

Adopting advanced technologies, such as vision transformers, can significantly improve the ability of the system to effectively analyze images. These technologies capture complex patterns and relationships within visual data, leading to improved classification results. Collaborating with conservation organizations would be invaluable in improving the system and ensuring its practical effectiveness in real-world conservation efforts.

Encouraging regular data collection and participation in citizen science, where users share their bird sightings, can also make the model more adaptable and realistic. Citizen scientists can provide valuable real-time information that enriches the dataset and enhances the training and validation processes of the application. Together, these efforts can transform the application and website into comprehensive tools that support global bird conservation and research, ultimately contributing to the preservation of avian biodiversity.

Since the code is open-source, we encourage researchers to use this interface to deploy their own models, whether for regional, country-level, or global-level bird species recognition. This flexibility allows for customized adaptations to specific datasets or regions. For users utilizing open-source code to deploy their models, we emphasize the importance of addressing key ethical and compliance considerations in their implementation. Users must ensure that image ownership rights are respected and clearly defined, particularly for uploads that involve user-generated content. Additionally, these users are responsible for ensuring adherence to regulations, such as the General Data Protection Regulation (GDPR) or other relevant local data-sharing laws. Proper measures must be taken to protect user data, including implementing consent mechanisms, secure storage, and transparent data handling procedures.

As we continue to develop the BirdRecon platform, our future updates will aim to provide built-in mechanisms to address these concerns. The planned improvements include compliance with data sharing regulations by integrating a privacy policy and service terms to provide transparency in data handling. Enhanced features will also be introduced to handle user-generated content responsibly, including secure, temporary image storage, and user control over uploaded data. Furthermore, our objective is to improve the performance and adaptability of the platform to include additional features for identifying bird species in complex scenarios, such as low light conditions or occlusions. By iteratively refining BirdRecon, our aim is to create a highly reliable and user-friendly platform that adheres to ethical practices and global standards.

CRediT authorship contribution statement

Hari Kishan Kondaveeti: Writing – review & editing, Writing – original draft, Supervision, Methodology, Conceptualization. **Nabin Kumar Upadhaya:** Writing – review & editing, Writing – original draft, Software. **Dheeraj Sai Tukkugudam:** Writing – review & editing, Writing – original draft, Software, Methodology. **Rahul Panigrahi:** Writing – review & editing, Writing – original draft, Software. **Sirivella Madhan Chandra Mouli:** Writing – review & editing, Writing – original draft, Software. **Valli Kumari Vatsavayi:** Writing – review & editing, Writing – original draft, Conceptualization. **Nagendra Panini Challa:** Writing – review & editing, Writing – original draft, Conceptualization.

Software availability

Name of software: BirdRecon

Availability and cost: BirdRecon is available for free and can be downloaded from https://bit.ly/BirdRecon.

Description and instructions: The repository includes a README file with a description and instructions.

License: GNU General Public License v3.0

Developers: Nabin Kumar Upadhaya, Dheeraj Sai Tukkugudam, Rahul Panigrahi, S Madhan Chandra Mouli

e-mail: nabinkumar2150@gmail.com; kishan.kondaveeti@vitap.ac.in

Year first available: 2024

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

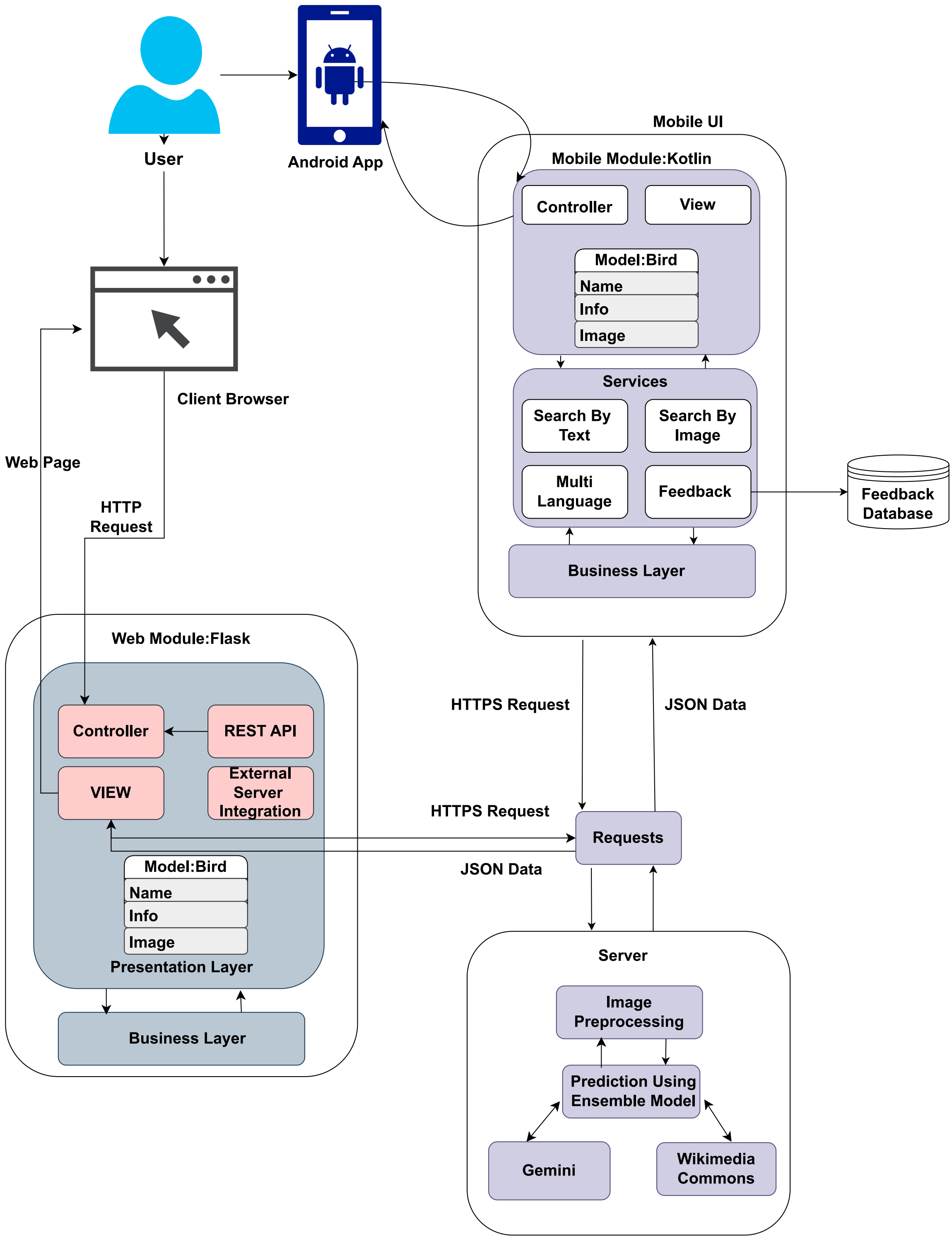

Fig. A.1. Software architecture.

Appendix A. Software description

A.1. Software architecture

Our bird species recognition system is designed to provide users with a seamless and intuitive experience on both mobile and web platforms. The system architecture is divided into two main components: the client-side application and the server-side processing. This separation allows for better scalability, maintainability, and performance optimization. Fig. A.1 illustrates the architecture of the developed software. This figure illustrates the interaction between the client and server-side components.

The client-side application consists of two user interfaces: an Android application and a web-based interface. The Android application is designed to allow users to easily capture or select images of the birds they encounter in the wild. This application provides two primary

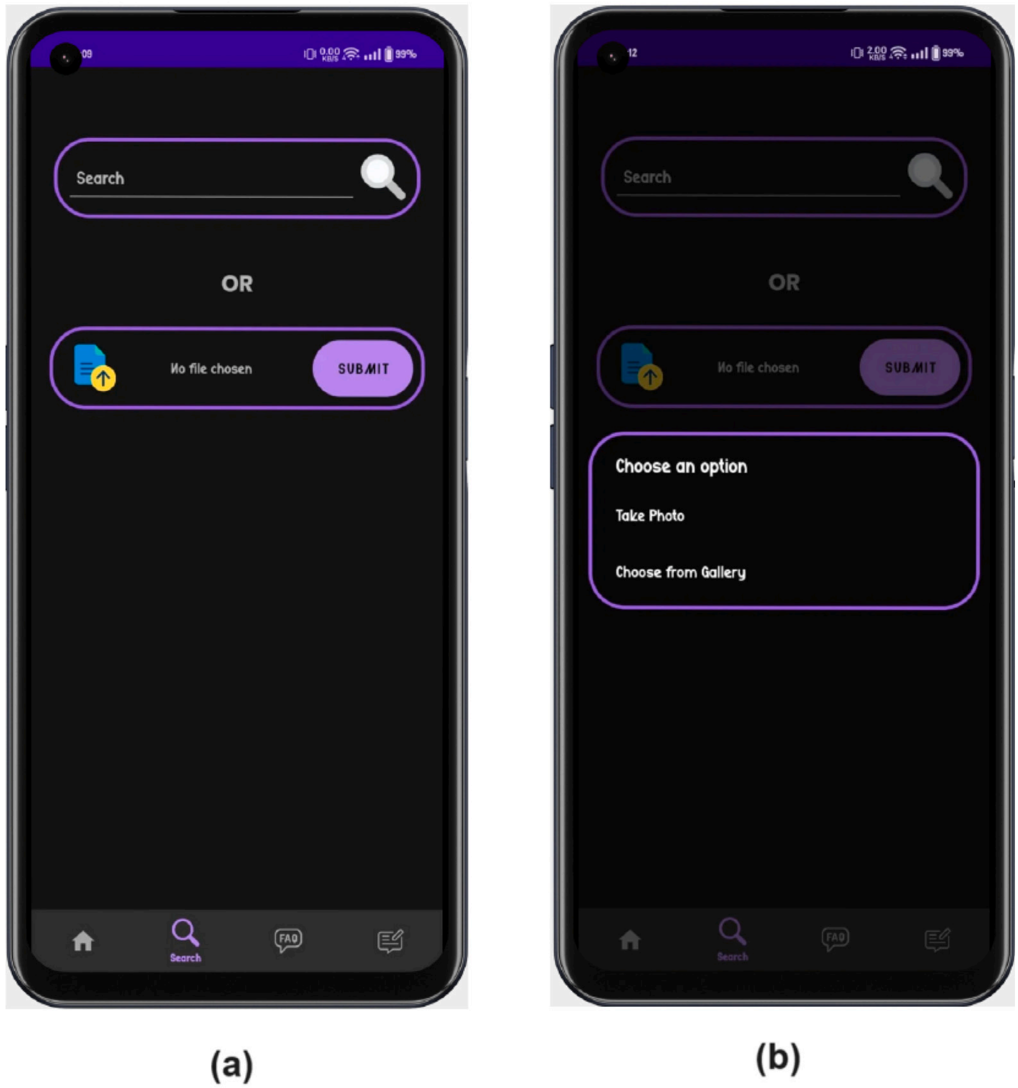

Fig. A.2. Mobile application with two search options. (a) Search by name and search by image. (b) Search by image with two options.

functionalities: image capture and image selection. The image capture feature enables users to capture photographs of birds directly within the application using their device's camera. This eliminates the need to switch between different applications, thereby making the process of capturing bird images more convenient and efficient. Alternatively, users can select an existing image from their device gallery, providing flexibility for those who may have already taken photos of birds using another camera or application.

Once an image is captured or selected, the application preprocesses the image data. This preprocessing step involves converting the image into a Base64 format, which is a standard encoding scheme that represents binary data in ASCII string format. The conversion to Base64 is necessary to ensure efficient and secure transmission of image data from the client-side application to the server-side processing component over a network connection.

The Android application utilizes the Volley networking library to facilitate communication between the client-side application and server-side processing. Volley is chosen for its simplicity, performance, and ability to handle large amounts of data effectively. The application sends the pre-processed image data to the server using the POST method. The POST method is used because it allows for the secure transmission of data and is suitable for sending larger amounts of information, such as image files.

Upon receiving data from the client-side application, the server-side processing component, which is implemented using the Flask web framework, handles the request. The flask is chosen for its lightweight nature, ease of use, and ability to be seamlessly integrated with other technologies. The server first decodes the received Base64-encoded image data back to its original binary format, making it suitable for further processing.

The server then feeds the decoded image data into a trained deep-learning model for bird species recognition. This study used a dataset (GTS, 2025) of 525 bird species with 84,635 training images, 2625 validation images, and 2625 test images. Four pretrained models, DenseNet201, EfficientNetB7, InceptionV3, and ResNet50V2, were customized with specialized layers for efficient bird classification. These models were combined as an ensemble with soft voting, where the final prediction was the average of the individual model predictions.

The model used in this system is the TensorFlow Lite model, which is specifically optimized for performance on servers and mobile devices. TensorFlow Lite is a lightweight version of the popular TensorFlow framework designed to run efficiently on resource-constrained devices without compromising accuracy. TensorFlow Lite ensures fast and accurate predictions even when handling a large number of requests simultaneously. During the inference phase, the server loads the pre-trained TensorFlow Lite model into a TensorFlow Lite interpreter,

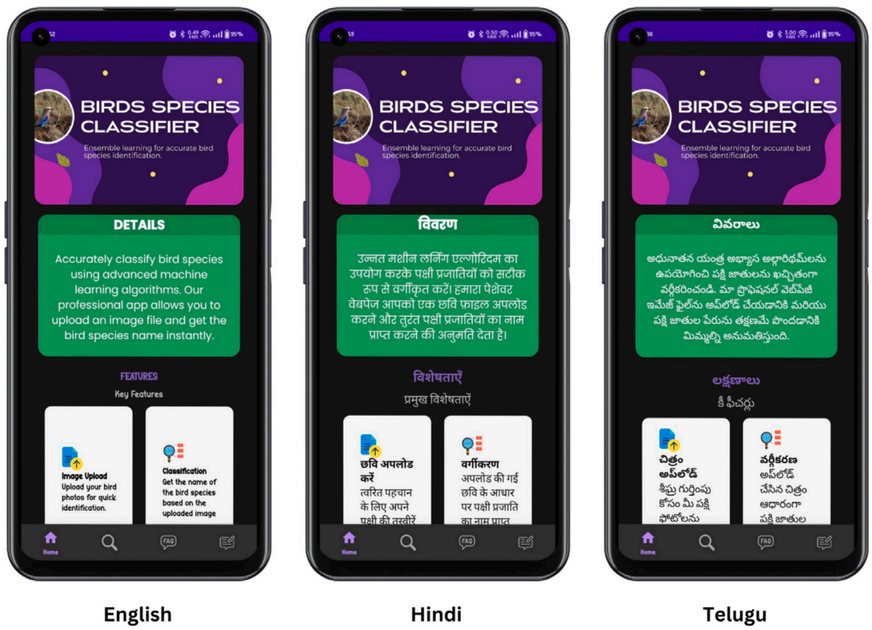

Fig. A.3. Mobile application with multilingual user interfaces.

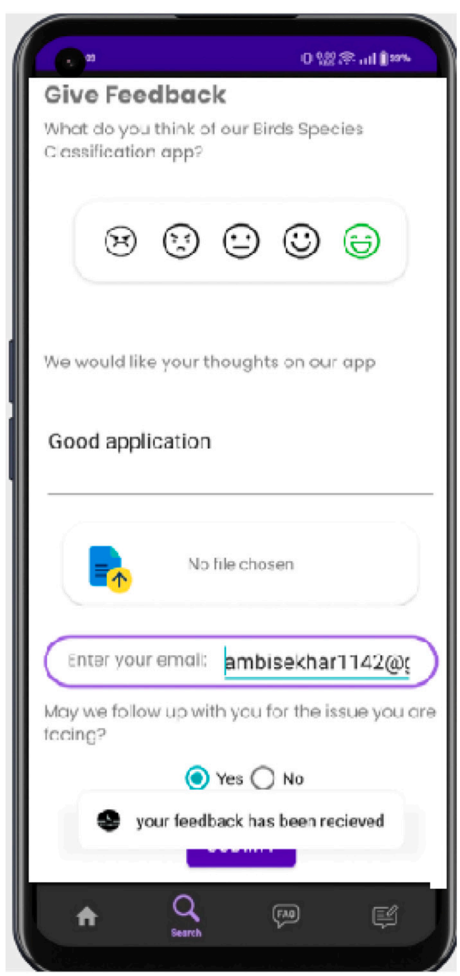

Fig. A.4. Mobile application with feedback form.

which is responsible for executing the model and generating predictions based on the input image data.

Once the model generates a prediction for the bird species present in the image, the server-side processing component retrieves additional information regarding the identified species to provide users with a more comprehensive understanding. To achieve this, the system utilizes two external services: Google Gemini for content generation, and Wikipedia Commons for image retrieval.

Google Gemini is a powerful natural language processing tool that can generate relevant and accurate textual content based on a given topic or context. In this case, the server sends the predicted bird species information to Google Gemini, which then generates a detailed summary or description of the bird. The generated content may include information and other relevant facts regarding the identified bird species.

To enhance the visual experience for users, the server also retrieves a relevant image of the bird species identified from the Wikipedia Commons. The Wikipedia Commons is a vast repository of freely usable images, including a wide variety of bird photographs contributed by the community. Using the Wikipedia Commons API, the server can search for and retrieve high-quality images of predicted bird species, providing users with a visual reference to aid in their identification and learning processes.

Finally, the server compiles all relevant information, the predicted bird species, the Google Gemini generated textual content, and the retrieved image URL from the Wikipedia Commons in a structured JavaScript Object Notation (JSON) format. JSON is chosen for its simplicity, readability, and compatibility with various client-side platforms. The server then sends this JSON response back to the client-side application, which parses the received data and presents them to the user in a user-friendly and visually appealing manner.

The web-based interface is also developed using Flask and followed a process similar to that of the Android application. Users can upload bird images directly through the web interface, which are then processed by the server-side component in the same manner as described above. The Web interface provides an alternative access point for users who may not have an Android device or may prefer to use a web browser for bird species recognition.

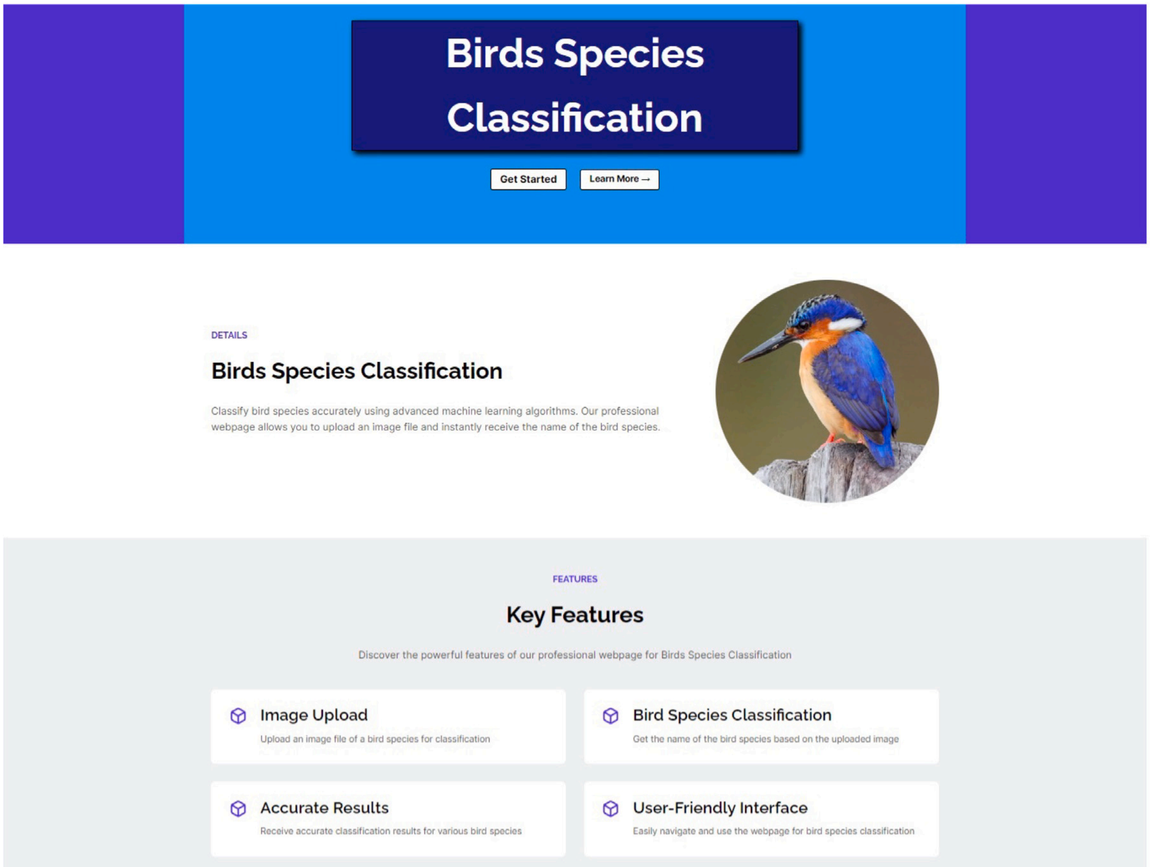

Fig. B.5. Web interface.

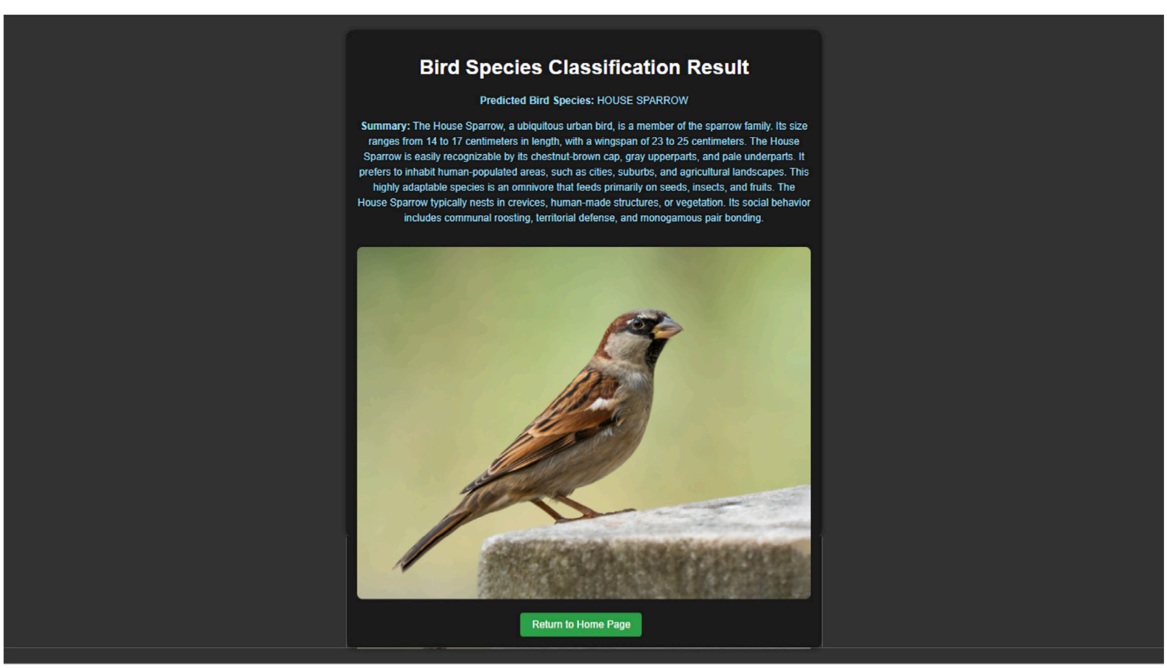

Fig. B.6. Result displayed after inputting an image of a House Sparrow. Bird species image courtesy by wiki commons (Rhododendrites, 2022).

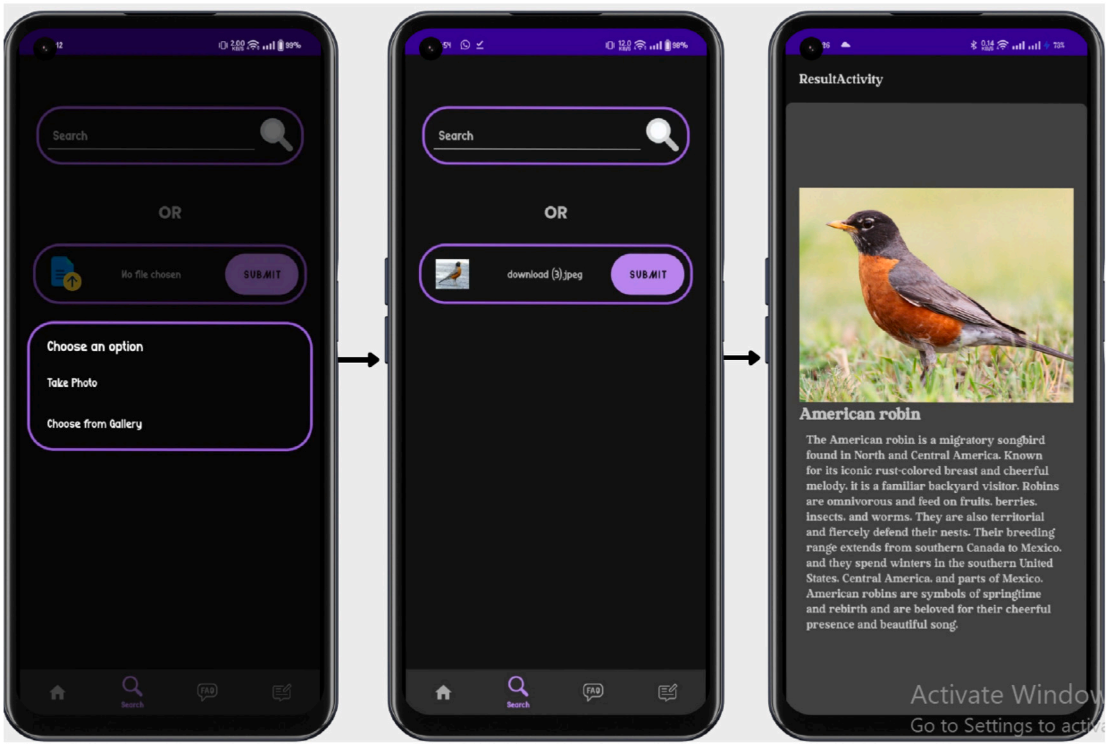

Fig. B.7. Result displayed after inputting an image of an American Robin to the mobile application from the gallery. Bird species image courtesy by wiki commons (Mdf, 2005).

A.2. Software functionalities

Our application is designed to serve ornithologists, bird enthusiasts, and casual birdwatchers by providing an automatic bird species recognition system. The primary goals are to facilitate research, support conservation efforts, and enhance the birdwatching experience.

The system is available on Android as a mobile application for both casual and professional use in the field, and also features a Flask-based web interface that can be accessed via web browsers for broader accessibility and ease of updates.

A.2.1. Search functionality

The application features robust search functionality, allowing users to upload an image to search for bird species or to capture an image using a mobile phone camera via the mobile application. Users can capture bird images directly through the mobile application or upload existing photos from the image gallery. The Web interface allows users to upload photos from memory. The system processes these images in real time to identify bird species using deep ensemble learning for accurate identification.

In addition, users can search for bird species by entering the name of the bird. Fig. A.2 shows the mobile application that features two types of search options.

A.2.2. Additional species related information

To enhance user experience, the application provides additional species related information with the support of Google Gemini, along with additional images fetched from Wikipedia Commons to provide visual context.

A.2.3. Multilingual support

Our application supports English, Telugu, and Hindi, thus enhancing usability for a wider range of users. Fig. A.3 shows the user interface of the mobile application with multilingual user interfaces, demonstrating its ability to cater to diverse linguistic preferences.

A.2.4. User feedback

Users can provide feedback, including thoughts, comments, and suggestions, through a built-in feedback option that facilitates user satisfaction assessments and issue reporting. Fig. A.4 shows the feedback feature of the mobile application.

A.2.5. FAQs section

Another valuable feature is the FAQ section, which provides basic information and answers to frequently asked queries, helping users to quickly find the answers they need.

Appendix B. Illustrative examples

Two user-friendly interfaces were created for bird enthusiasts to easily identify bird species and to access brief information along with a sample image in the gallery or by capturing a photograph.

Ornithologists and bird watchers can use the web interface for their studies. Fig. B.5 shows the web interface of our application, which is designed to be user-friendly and efficient. For this purpose, a clear image of a bird should be uploaded to the website.

When a user uploads an image of a bird, such as a House Sparrow, the application preprocesses the image by converting it to Base64 for secure transmission. Using the Volley library, the application sends encoded data to a server via a POST request. The server, implemented with Flask, decodes the image and processes it using a TensorFlow Lite model to identify bird species. In addition, the server retrieves additional information from Google Gemini and related images from Wikipedia Commons. The combined data, including bird identification, text, and an image, are returned to the application in JSON format for a user-friendly display. Fig. B.6 shows the result when an image of a House Sparrow is provided as input.

Fig. B.7 shows the result when an image of an American Robin is provided as input to the mobile application.

Data availability

The data set used in this study is available for free and can be downloaded from https://gts.ai/392dataset-download/birds-525-species-image-classification/.

References

Bibby, C.J., 1999. Making the most of birds as environmental indicators. Ostrich 70 (1), 81–88.

Birda, 2023. Birda: Bird identification and observation platform. https://birda.org/. (Accessed 14 October 2023).

BirdNet, 2024. BirdNET sound ID - the easiest way to identify birds by sound. — birdnet.cornell.edu. https://birdnet.cornell.edu/. (Accessed 11 June 2024).

Cai, R., 2023. Automating bird species classification: A deep learning approach with CNNs. In: Journal of Physics: Conference Series. Vol. 2664, IOP Publishing, 012007.

Charmantier, A., Gienapp, P., 2014. Climate change and timing of avian breeding and migration: evolutionary versus plastic changes. Evol. Appl. 7 (1), 15–28.

De Silva, P., Senevirathne, M., Samarasinghe, S., Manawadu, U.A., 2019. BirdGuide: Developing a mobile platform to identify the categories of Sri Lankan bird species. In: Proceedings of the 6th International Conference on Multidisciplinary Approaches. ICMA.

eBird, 2023. Ebird: Explore your area. https://ebird.org/ebird/home. (Accessed 14 October 2023).

Edney, A.J., Wood, M.J., 2021. Applications of digital imaging and analysis in seabird monitoring and research. Ibis 163 (2), 317–337.

Fang, F., Tambe, M., Dilkina, B., Plumptre, A.J., 2019. Artificial Intelligence and Conservation. Cambridge University Press.

Farman, H., Ahmed, S., Imran, M., Noureen, Z., Ahmed, M., 2023. Deep learning based bird species identification and classification using images. J. Comput. Biomed. Inform. 6 (01), 79–96.

Ferreira, A.C., Silva, L.R., Renna, F., Brandl, H.B., Renoult, J.P., Farine, D.R., Covas, R., Doutrelant, C., 2020. Deep learning-based methods for individual recognition in small birds. Methods Ecol. Evol. 11 (9), 1072–1085.

Fraisl, D., See, L., Fritz, S., Haklay, M., McCallum, I., 2024. Leveraging the collaborative power of AI and citizen science for sustainable development. Nat. Sustain. 1–8.

Ganaie, M.A., Hu, M., Malik, A.K., Tanveer, M., Suganthan, P.N., 2022. Ensemble deep learning: A review. Eng. Appl. Artif. Intell. 115, 105151.

Green, S.E., Rees, J.P., Stephens, P.A., Hill, R.A., Giordano, A.J., 2020. Innovations in camera trapping technology and approaches: The integration of citizen science and artificial intelligence. Animals 10 (1), 132.

Gregory, R.D., Noble, D., Field, R., Marchant, J., Raven, M., Gibbons, D., 2003. Using birds as indicators of biodiversity. Ornis Hung. 12 (13), 11–24.

Grimalt, R., Murawiec, S., Tryjanowski, P., 2023. Jizz in birdwatching activity and clinical practice: how it works and why? Arch. Psychiatry Psychother. 25 (2), 6.

GTS, 2025. Birds 525 species- image classification. URL: https://gts.ai/dataset-download/birds-525-species-image-classification/.

Gupta, D., et al., 2021. Mobile application for bird species identification using transfer learning. In: 2021 IEEE International Conference on Artificial Intelligence in Engineering and Technology. IICAIET, IEEE, pp. 1–6.

Hermadi, I., Dhira, D., et al., 2022. Development of a protected birds identification system using a convolutional neural network.. Biodiversitas: J. Biol. Divers. 23 (5).

Huang, Y.-P., Basanta, H., 2019. Bird image retrieval and recognition using a deep learning platform. IEEE Access 7, 66980–66989.

Huang, Y.-P., Basanta, H., 2021. Recognition of endemic bird species using deep learning models. Ieee Access 9, 102975–102984.

iNaturalist, 2024. Inaturalist — inaturalist.org. https://www.inaturalist.org/. (Accessed 11 June 2024).

Jasim, H.A., Ahmed, S.R., Ibrahim, A.A., Duru, A.D., 2022. Classify bird species audio by augment convolutional neural network. In: 2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications. HORA, IEEE, pp. 1–6.

Kahl, S., Wood, C.M., Eibl, M., Klinck, H., 2021. BirdNET: A deep learning solution for avian diversity monitoring. Ecol. Inform. 61, 101236.

Kati, V.I., Sekercioglu, C.H., 2006. Diversity, ecological structure, and conservation of the landbird community of Dadia reserve, Greece. Diversity and Distributions 12 (5), 620–629.

Kondaveeti, H.K., Sanjay, K.S., Shyam, K., Aniruth, R., Gopi, S.C., Kumar, S.V., 2023. Transfer learning for bird species identification. In: 2023 2nd International Conference on Computational Systems and Communication. ICCSC, IEEE, pp. 1–6.

Kumar, S.V., Kondaveerti, H.K., 2023. A comparative study on deep learning techniques for bird species recognition. In: 2023 3rd International Conference on Intelligent Communication and Computational Techniques. ICCT, IEEE, pp. 1–6.

Kumar, S.V., Kondaveeti, H.K., 2024. Bird species recognition using transfer learning with a hybrid hyperparameter optimization scheme (HHOS). Ecol. Inform. 102510.

Kumar, S., Kondaveeti, H.K., Simhadri, C.G., Reddy, M.Y., 2023. Automatic bird species recognition using audio and image data: A short review. In: 2023 IEEE International Conference on Contemporary Computing and Communications (InC4). Vol. 1, IEEE, pp. 1–6.

Kumar, M., Yadav, A.K., Kumar, M., Yadav, D., 2022. Bird species classification from images using deep learning. In: International Conference on Computer Vision and Image Processing. Springer, pp. 388–401.

Kwok, R., et al., 2019. AI empowers conservation biology. Nature 567 (7746), 133–134.

Mdf, 2005. American Robin (Turdus migratorius) in Humber Bay Park (East), (Toronto, Canada). URL: https://en.wikipedia.org/wiki/File:Turdus-migratorius-002.jpg. (Accessed 26 June 2024).

Mochurad, L., Svystovych, S., 2024. A new efficient classifier for bird classification based on transfer learning. J. Eng. 2024 (1), 8254130.

Mohammed, A., Kora, R., 2023. A comprehensive review on ensemble deep learning: Opportunities and challenges. J. King Saud Univ.- Comput. Inf. Sci. 35 (2), 757–774.

Nichols, J.D., Williams, B.K., 2006. Monitoring for conservation. Trends Ecol. Evolut. 21 (12), 668–673.

Peterson, M.N., Larson, L.R., Hipp, A., Beall, J.M., Lerose, C., Desrochers, H., Lauder, S., Torres, S., Tarr, N.A., Stukes, K., Stevenson, K., Martin, K.L., 2024. Birdwatching linked to increased psychological well-being on college campuses: A pilot-scale experimental study. J. Environ. Psychol. 96, 102306. http://dx.doi.org/10.1016/j.jenvp.2024.102306, URL: https://www.sciencedirect.com/science/article/pii/S0272494424000793.

Pichler, M., Hartig, F., 2023. Machine learning and deep learning—A review for ecologists. Methods Ecol. Evol. 14 (4), 994–1016.

Polisetty, V.R.M., Chokkalingam, S., 2024. Enhancing bird species classification: A weighted average transfer learning ensemble approach. In: 2024 2nd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT). IEEE, pp. 1401–1406.

Qin, J., Pan, W., Xiang, X., Tan, Y., Hou, G., 2020. A biological image classification method based on improved CNN. Ecol. Inform. 58, 101093.

Randler, C., Staller, N., Kalb, N., Tryjanowski, P., 2023. Charismatic species and birdwatching: Advanced birders prefer small, shy, dull, and rare species. Anthrozoös 36 (3), 427–445.

Rhododendrites, 2022. Male house sparrow in prospect park. URL: https://en.wikipedia.org/wiki/File:House_sparrow_male_in_Prospect_Park_(53532).jpg.

SS, P.D., Jayanth, M., Sangeetha, V., 2024. Multiclass bird species identification using deep learning techniques. In: 2024 IEEE International Conference on Electronics, Computing and Communication Technologies. CONECCT, IEEE, pp. 1–6.

stagmerlinapp2, 2024. Merlin bird ID - home — merlin.allaboutbirds.org. https://merlin.allaboutbirds.org/. (Accessed 11 June 2024).

Sullivan, B.L., Wood, C.L., Iliff, M.J., Bonney, R.E., Fink, D., Kelling, S., 2009. eBird: A citizen-based bird observation network in the biological sciences. Biol. Cons. 142 (10), 2282–2292.

Team, G., Anil, R., Borgeaud, S., Wu, Y., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A.M., Hauth, A., et al., 2023. Gemini: a family of highly capable multimodal models. arXiv preprint arXiv:2312.11805.

Ullah, F., Saqib, S., Xiong, Y.-C., 2024. Integrating artificial intelligence in biodiversity conservation: bridging classical and modern approaches. Biodivers. Conserv. 1–21.

Vo, H.-T., Thien, N.N., Mui, K.C., 2023. Bird detection and species classification: using YOLOv5 and deep transfer learning models. Int. J. Adv. Comput. Sci. Appl. 14 (7).

Wang, K., Yang, F., Chen, Z., Chen, Y., Zhang, Y., 2023. A fine-grained bird classification method based on attention and decoupled knowledge distillation. Animals 13 (2), 264.

Wei, X.-S., Xie, C.-W., Wu, J., Shen, C., 2018. Mask-CNN: Localizing parts and selecting descriptors for fine-grained bird species categorization. Pattern Recognit. 76, 704–714.

Wikimedia Commons, 2023. Wikimedia commons: Main page. https://commons.wikimedia.org/wiki/Main_Page. (Accessed 14 October 2023).

Wu, P., Wu, G., Wu, X., Yi, X., Xiong, N., 2021. Birds classification based on deep transfer learning. In: Smart Computing and Communication: 5th International Conference, SmartCom 2020, Paris, France, December 29–31, 2020, Proceedings 5. Springer, pp. 173–183.