Advanced Deep Learning Techniques for Wildlife Image Classification: A Case Study Using EfficientNet and Augmented Animal Datasets

Submitted for

CSET211 - Statistical Machine Learning

Submitted by:

E23CSEU0525 - Supratim Saha

E23CSEU0577 - Sambhav Jain

Submitted to:

Prashant Kapil

July-Dec 2024

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Contents

Abstract

This project explores the application of advanced deep learning techniques for wildlife image classification, leveraging the "iamsouravbanerjee/animal-image-dataset-90-different-animals." The pipeline includes comprehensive data preprocessing, augmentation strategies, and the use of the EfficientNetB3 architecture to achieve accurate classification of 90 distinct animal species. Key steps include resizing and normalizing image data, encoding labels, and implementing data augmentation to enhance model robustness. The model is evaluated using performance metrics such as accuracy, confusion matrix, and precision-recall analysis. The results demonstrate the effectiveness of EfficientNetB3 in handling complex wildlife datasets, providing valuable insights for ecological monitoring and conservation. Supplementary materials and source code are available on GitHub.

Introduction

Wildlife monitoring and conservation have become critical areas of research due to the increasing threats to biodiversity caused by habitat destruction, climate change, and poaching. Accurate identification and classification of animal species play a vital role in enabling effective wildlife management and ecological studies. Traditional methods of wildlife detection, such as field surveys and manual image annotation, are labor-intensive and prone to errors.

In this project, we leverage advancements in deep learning to develop a robust and scalable solution for wildlife image classification. Using the "iamsouravbanerjee/animal-image-dataset-90-different-animals," which includes images of 90 distinct animal classes, we preprocess the data and train a convolutional neural network (CNN) based on the EfficientNetB3 architecture. This model is known for its high accuracy and efficiency, making it suitable for large-scale image classification tasks.

The methodology involves preprocessing and augmenting the dataset to enhance generalization, followed by training the model with an 80-20 train-test split. The performance of the model is evaluated using metrics such as classification accuracy, confusion matrix, and precision-recall analysis. This report aims to demonstrate the effectiveness of deep learning techniques in addressing the challenges of wildlife image classification and explore their potential for real-world applications.

Related Survey

Recent advancements in deep learning have significantly improved animal image classification. Studies such as those by Kumar et al. (2023) and Gourisaria et al. (2023) emphasize the effectiveness of transfer learning with CNN architectures like VGG16, EfficientNet, and ResNet, achieving accuracies as high as 99%. Zhang et al. (2022) demonstrated parameter optimization's role in improving traditional CNN models, achieving 85% accuracy.

Hossain et al. (2020) and Chandrakar et al. (2021) explored advanced techniques like genetic algorithms and fine-tuned CNNs for vertebrate classification, achieving high precision and recall. Singh et al. (2021) showcased the adaptability of CNNs in identifying specific categories like dog breeds using pre-trained Xception architecture.

These studies highlight the importance of preprocessing, transfer learning, and model optimization, forming the foundation for our use of EfficientNetB3 and data augmentation strategies.

Datasets

The dataset used in this project is the *"iamsouravbanerjee/animal-image-dataset-90-different-animals"* from Kaggle. This dataset contains a diverse collection of images representing 90 distinct animal species, making it suitable for training and evaluating deep learning models for wildlife image classification.

Data Preprocessing

To prepare the dataset for training, several preprocessing steps were applied to ensure data consistency and improve model performance:

- **Image Resizing:** All images were resized to 224x224 pixels to maintain a uniform input shape compatible with the EfficientNetB3 architecture.

- **Normalization:** Pixel values were normalized to the range [0, 1] by dividing each pixel value by 255. This step ensures that the model converges faster during training.
- **Data Augmentation:** To enhance the model's ability to generalize, data augmentation techniques such as rotation, width/height shift, shear, zoom, and horizontal flips were applied. This created variability in the training data without requiring additional images.
- **Label Encoding:** The animal class labels were converted into numerical format using a label encoder, making them suitable for input to the neural network.

These preprocessing steps were crucial for addressing challenges such as varying image resolutions, imbalanced datasets, and overfitting during training. They also improved the robustness and scalability of the final model.

Methodology

The proposed methodology for wildlife image classification involves the following steps:

- **Data Collection:** Images from the "iamsouravbanerjee/animal-image-dataset-90-different-animals" were used, comprising 90 animal classes.
- **Data Preprocessing:** Images were resized to 224x224 pixels, normalized to [0, 1], and augmented using transformations like rotation, zoom, and flips.
- **Model Architecture:** The EfficientNetB3 model was chosen for its balance of accuracy and computational efficiency. Additional layers, including Global Average Pooling, Dense, and Dropout, were added to improve performance.
- **Training Process:** The dataset was split into training (80%) and testing (20%) sets. Early stopping and learning rate reduction were applied to optimize the training process.
- **Evaluation:** The trained model was evaluated using metrics such as accuracy, confusion matrix, and classification reports.

This methodology ensures robust model training and generalization across diverse animal classes.

Hardware and Software Requirements

The project was developed and executed using the following hardware and software:

Hardware

- **Laptop:** ASUS TUF F15
- **Processor:** Intel i7 13th Gen
- **GPU:** NVIDIA RTX 4060
- **RAM:** 16 GB

Software

- **Programming Language:** Python 3.9
- **Libraries and Frameworks:** TensorFlow, Keras, OpenCV, Plotly, Pandas, NumPy, Matplotlib, Seaborn
- **Development Environment:** Jupyter Notebook
- **Dataset Source:** Kaggle

Performance Metrics

To evaluate the effectiveness of the classification model, the following performance metrics were used:

- **Accuracy:** Measures the proportion of correctly classified images out of the total images.
- **Confusion Matrix:** Provides a detailed breakdown of true positives, true negatives, false positives, and false negatives.

- **Precision, Recall, and F1-Score:**
  - **Precision:** The proportion of true positive predictions among all positive predictions.
  - **Recall:** The proportion of true positives identified out of all actual positives.
  - **F1-Score:** The harmonic mean of precision and recall, balancing the trade-off between the two.
- **Loss Metrics:** Training and validation loss were monitored to assess model convergence and detect overfitting.

These metrics provide a comprehensive assessment of the model's performance across diverse animal classes.

Results and Analysis

The EfficientNetB3 model demonstrated high performance in classifying 90 animal species.

Performance Summary

- **Accuracy:** Training accuracy reached 98%, with a validation accuracy of 96%.
- **Confusion Matrix:** Most predictions were correct, with minimal misclassifications primarily between visually similar classes.
- **Classification Report:** The model achieved an average precision of 95%, recall of 96%, and F1-score of 95.5%.

Visual and Error Analysis

Sample predictions showed accurate classifications, while errors were linked to overlapping features or poor-quality images.

Conclusions and Future Works

This project successfully demonstrated the application of advanced deep learning techniques for wildlife image classification. By leveraging the EfficientNetB3 architecture and employing robust preprocessing and augmentation strategies, the model achieved high accuracy (96%) across 90 animal classes. The results validate the potential of deep learning in addressing challenges related to biodiversity monitoring and conservation.

Future Works

- Expanding the dataset to include more diverse and challenging animal images to improve model robustness.
- Exploring advanced techniques such as attention mechanisms and ensemble methods to further enhance classification accuracy.
- Deploying the model for real-time wildlife monitoring and integrating it into conservation workflows.
- Addressing misclassification errors by incorporating domain-specific features and improving dataset quality.

These improvements can elevate the model's practicality, ensuring its effectiveness in real-world applications for ecological and conservation efforts.