Animal Multiclassification

Computer Vision & Deep Learning Project

Artificial Intelligence MSc 2023/2024

Student: Emil Alizada

Matricola: VR510345

Date: 9 July 2024

Table of Contents

1. Introduction 3-4

2. Motivation 5-6

3. Objectives 7-8

4. Methodology 9-11

4.1 Data Loading and Preprocessing 9

4.2 Data Augmentation 9

4.3 Model Definition 10

4.4 Learning Rate Scheduler 11

5. Experiments and Results 12-14

5.1 Model Training 12

5.2 Model Evaluation 12

5.3 Training and Validation Loss 13

5.4 Detailed Evaluation 13

5.5 Qualitative Results 14

6. Conclusion 14

7. References 15

1. Introduction

The Animal Multiclassification Project uses powerful deep learning techniques to reliably categorize photos of different animal species into predetermined categories. Numerous applications of this classification method exist, such as ecological study, wildlife monitoring, and improved comprehension of animal behavior. This study uses Convolutional Neural Networks (CNNs) to develop a dependable system that can distinguish between different animal species.

The project uses a thorough methodology that includes numerous important stages, such as building the model, loading and preprocessing data, augmentation of data, training, and assessment. Each stage is meticulously prepared to allow the development of a viable categorization model. The dataset contains photos of animals from many categories. Images are preprocessed to standardize their dimensions and scale their pixel values before being fed into neural networks. This step is crucial as it ensures uniformity in the input data, facilitating more efficient and accurate training of the model.

A data augmentation strategy is utilized to increase the robustness and generalization capabilities of the model. These techniques consist of arbitrary flips, zooms, shifts, and rotations. The model gains the ability to identify animals from a variety of viewpoints and angles by artificially increasing the diversity of the training data, which helps it become more flexible in response to variations in real-world circumstances. This project uses two different models: a pre-trained ResNet50v2 model and a custom CNN model.

The custom CNN model is created from scratch and includes numerous convolutional layers that extract relevant characteristics from the input photos. Activation functions, pooling layers, and dropout layers are all applied after each convolutional layer. Batch normalization is used to increase training stability and performance. The model is built using the

Adam optimizer with categorical cross-entropy loss, which ensures efficient training and convergence.

The ResNet50v2 model, a highly acclaimed pretrained model, is fine-tuned for the animal classification task. The base layers of the ResNet50v2 are frozen to retain the knowledge acquired during pretraining on large-scale datasets. Custom dense layers are added on top to adapt the model to the specific animal classification task. This approach leverages the powerful feature extraction capabilities of ResNet50v2, leading to improved performance.

The training phase is where the models learn to recognize and classify creatures. The CNN models get augmented data in batches. During training, model parameters like weights and biases are modified to reduce the loss function, which assesses the difference between predicted and actual classifications. A learning rate scheduler is used to dynamically alter the learning rate, increasing the training process's efficiency and convergence.

Metrics including accuracy, precision, recall, and F1-score are used to evaluate the models' performance on an alternative test set. These metrics provide a thorough insight of the models' classification capabilities. Confusion matrices and classification reports are also provided to provide specific information about the models' strengths and weaknesses across different animal groups.

The successful completion of this project has significant implications for various fields. In wildlife monitoring, accurate animal classification can aid in tracking population dynamics and identifying endangered species. Ecological studies can benefit from automated classification systems, providing valuable data on animal behavior and interactions. Furthermore, enhanced understanding of animal species through such technologies can contribute to conservation efforts and scientific research.

2. Motivation

The urgent need to reliably categorize animal species for a range of significant applications is the driving force behind the Animal Multiclassification Project. With the help of technology developments, this project seeks to address important issues and significantly progress the following fields:

1. Monitoring and conserving wildlife

Exact taxonomy of animal species is necessary for efficient wildlife observation and preservation initiatives. Through accurate identification and tracking of various species, conservationists can:

- Keep an eye on wildlife populations. Maintain an eye on population levels, movement trends, and breeding practices to safeguard the survival and well-being of species.

- Preserve Endangered Species: Identify and concentrate conservation efforts on species that are in danger of going extinct as soon as possible to allow for prompt interventions and habitat preservation measures.

2. Ecological Studies

Understanding animal behavior and distribution is crucial for ecological research. Accurate classification enables researchers to:

- Studying the relationships between different species and their environments might help you understand ecosystems better. This will provide insights into ecological dynamics.

- Forecast Environmental Shifts: Keep an eye on how animal populations react to environmental shifts like habitat degradation and climate change. This information is crucial for developing predictive models.

3. Automation in Animal Research

Automating the process of animal classification presents numerous benefits for scientific research and practical applications:

• Save Time and Resources: By reducing the amount of time and manual work needed to identify and classify animal species, researchers may concentrate on data analysis and interpretation.

• Improve Data Accuracy: Reduce human error in species identification, which will produce more trustworthy and accurate study findings.

4. Broader Impact and Future Prospects

The successful implementation of the Animal Multiclassification Project holds promise for a wide range of future applications:

• Conservation Technology: For real-time wildlife monitoring and preservation, incorporate cutting-edge classification algorithms into conservation technology like drones and video traps.

• Citizen Science Projects: Equip citizen scientists with the means to precisely identify and document animal encounters, enabling them to make contributions to ecological study.

3. Objectives

Creating a dependable and precise Convolutional Neural Network (CNN) model for classifying animal pictures into predetermined groups is the main objective of this project. A few distinct goals that each concentrate on an important facet of the model building process assist this overall goal:

1. **Data Acquisition and Preparation**

Objective: To compile and preprocess a comprehensive dataset of animal images, ensuring it is suitable for training a CNN model.

- **Data Collection:** Gather a diverse and extensive dataset of animal photos from various sources, covering multiple species and categories.
- **Image Preprocessing:** Standardize the images by resizing them to a uniform dimension and normalizing pixel values, ensuring efficient processing and improved model performance.

2. **Data Augmentation**

Objective: To enhance model robustness and increase the diversity of the training dataset using data augmentation techniques.

- **Rotations:** Apply random rotations to images to make the model invariant to orientation changes.
- **Shifts:** Perform horizontal and vertical shifts to simulate real-world scenarios where animals might not be perfectly centered.
- **Zooms:** Implement random zoom operations to make the model resilient to variations in image scale.
- **Flips:** Introduce random horizontal flips to improve the model's ability to generalize across different viewing angles.

3. **Model Architecture Design**

Objective: To create effective CNN architectures for both custom and pretrained models, enabling them to learn and generalize features from animal images.

- Personalized CNN Model: Construct a personalized CNN that combines numerous convolutional layers, pooling layers, and dropout layers to effectively extract and learn complex information from the images.
- Pretrained ResNet50v2 Model: To optimize a pretrained ResNet50v2 model for the animal classification problem, add custom dense layers on top of it to take use of its potent feature extraction capabilities.

4. **Model Training**

Objective: To optimize the training process and train the CNN models using the augmented dataset and learning rate schedulers.

- The training procedure involves using the enhanced dataset to train the CNN models, modifying model parameters like weights and biases to minimize the loss function.
- Learning Rate Schedulers: During training, dynamically change the learning rate using learning rate schedulers such as ReduceLROnPlateau to ensure optimal convergence and avoid overshooting.

5. **Model Evaluation**

Objective: Using a range of performance criteria, assess the trained models on a test dataset and generate detailed assessment reports.

- **Performance measures:** Evaluate the models' capacity for classification using measures including accuracy, precision, recall, and F1-score.
- **Confusion Matrix:** Create a confusion matrix to see how well the models perform across various animal groups and to pinpoint their advantages and disadvantages.
- **Classification Reports:** Produce detailed classification reports that provide insights into each model's performance, highlighting areas for potential improvement.

4. Methodology

4.1 Preprocessing and Data Loading

Objective: To prepare the dataset for effective training of the CNN models by ensuring it is standardized, and appropriately divided.

- **Data Collection:** The dataset comprises images of animals classified into various species. It is divided into training, validation, and separate test sets to facilitate comprehensive model evaluation. The model is exposed to a broad range of animal groups thanks to the diversified dataset.

- **Image Resizing**: All images are resized to the standard size of 224 by 224 pixels. The CNN models require a uniform input size for efficient processing and better performance, and standardization is the only way to achieve this.
- **Normalization:** Pixel values are subjected to a range of 0 to 1. Normalization speeds up the model's convergence during training and increases the efficacy of the training process by ensuring that the data is on the same scale.
- **Dataset splitting:** To enhance model generalization, the dataset is logically split into training and validation sets.

4.2 Data Augmentation

Objective: By using a variety of data augmentation approaches, to increase the diversity of the training dataset and enhance the generalization of the model.

To artificially boost the training dataset's variability and strengthen the model's resilience to real-world circumstances, data augmentation techniques are utilized:

- **Rotation:** Images are randomly rotated within a specified range, making the model invariant to orientation changes and ensuring it can recognize animals regardless of their angle.
- **Zoom:** Random zoom operations are applied, both zooming in and out, to make the model resilient to variations in the scale of the images, ensuring accurate classification of animals at different distances.
- **Shifts:** Horizontal and vertical shifts are randomly performed to simulate real-world scenarios where animals might not be perfectly centered in the images. This enhances the model's ability to generalize across different image compositions.
- **Flips:** Random horizontal flips are introduced to make the model invariant to the direction of the animals, ensuring it can accurately classify animals regardless of their orientation.

4.3 Model Definition

Objective: To design and implement effective CNN architectures for both custom and pretrained models, enabling them to learn and generalize features from animal images.

Custom CNN Model:

To carefully extract complex characteristics from the input images, a multiple convolutional block custom CNN model is created.

- **Convolutional Layers:** These layers apply various filters to the input images to detect essential features such as edges, textures, and shapes.
- **Activation Functions:** Rectified Linear Unit (ReLU) activation functions introduce non-linearity, allowing the model to learn complex patterns.
- **Pooling Layers:** Max-pooling layers are used to downsample the feature maps, reducing their dimensions while retaining critical features, thus enhancing computational efficiency and robustness to spatial variations.
- **Batch Normalization:** Batch normalization layers are included to stabilize and speed up the training process by normalizing the output of the previous layers.
- **Dropout Layers:** Dropout layers are incorporated to prevent overfitting by randomly setting a fraction of the input units to zero during training, promoting generalization.
- **Compilation:** The model is compiled with the Adam optimizer and categorical cross-entropy loss function, ensuring efficient and effective training.

Pretrained ResNet50v2 Model:

The pretrained ResNet50v2 model takes advantage of a large-scale dataset to extract powerful features.

- **Freezing Base Layers:** ResNet50v2's base layers are frozen to retain the features gained during pretraining.
- **Custom Dense Layers:** Custom dense layers are added on top of the frozen base layers to adapt the model for the specific animal classification task.

- **Compilation:** Like the custom CNN, the ResNet50v2 model is compiled with the Adam optimizer and categorical cross-entropy loss, ensuring consistency in training approaches.

4.4 Learning Rate Scheduler

Objective: During training, dynamically modify the learning rate to improve convergence of the model and avoid overshooting.

• **ReduceLROnPlateau**: Based on the validation loss, the learning rate is modified using the ReduceLROnPlateau learning rate scheduler. The learning rate is lowered to prevent the model from overshooting the ideal answer and to ensure that it converges successfully when the validation loss stops declining.

5. Experiments and Results

5.1 Model Training

Objective: To train the CNN models effectively using the augmented dataset.

The training process is a critical phase where the models learn to classify animal images accurately. The steps involved in the training process include:

- **Batch Training:** To ensure effective use of computational resources and enable gradient-based optimization, the augmented dataset is fed into the CNN models in batches
- **Epochs:** The models are trained over several epochs, each consisting of multiple iterations where the entire training dataset is passed through the model.
- **Parameter Optimization:** During training, the model parameters, including weights and biases, are continuously adjusted to minimize the loss function. The Adam optimizer is utilized for its efficiency and adaptive learning rate capabilities, which help accelerate convergence.

- **Early Stopping:** Early stopping is used to avoid overfitting. In order to preserve the model's capacity for generalization, this method tracks the validation loss and stops training if no progress is seen after a predetermined number of epochs

5.2 Model Evaluation

Objective: Using a variety of assessment criteria, evaluate how well the trained models perform on the test dataset. After training, the models are evaluated on a separate test dataset to determine their classification performance:

- **Accuracy:** Measures the overall correctness of the model's predictions.
- **Precision:** Calculates the proportion of true positive predictions among all positive predictions made by the model.
- **Recall:** Determines the proportion of true positive predictions among all actual positive instances in the dataset.
- **F1-Score:** Combines precision and recall into a single metric that balances both concerns.
- **Confusion Matrix:** Provides a detailed breakdown of the model's performance across different animal classes, highlighting true positives, false positives, true negatives, and false negatives.
- **Classification Report:** Summarizes the precision, recall, and F1-score for each class, offering insights into the strengths and weaknesses of the models.

5.3 Training and Validation Loss

Objective: To visualize the training progress and detect potential overfitting.

The training and validation loss curves are plotted to provide a clear picture of the models' performances over time:

- **Training Loss:** Tracks the model's performance on the training dataset across epochs.
- **Validation Loss:** Monitors the model's performance on the validation dataset across epochs.

- **Convergence:** Ideally, both training and validation losses should decrease and converge to a low value, indicating effective learning.
- **Overfitting Detection:** An increase in validation loss while training loss continues to decrease signals potential overfitting, where the model performs well on training data but poorly on unseen data.

5.4 Detailed Evaluation

Objective: To conduct a thorough evaluation of the models' performances and identify areas for improvement.

A detailed evaluation involves analyzing the classification reports and confusion matrices for each model:

- **Precision, Recall, and F1-Score:** These metrics are examined for each animal class to pinpoint areas where the models excel or need improvement.
- **Confusion Matrix Analysis:** The confusion matrix reveals how well the models distinguish between different animal classes, highlighting common misclassifications and potential areas for dataset enhancement or model tuning.

5.5 Qualitative Results

Objective: To gain qualitative insights into the models' predictions through visual analysis.

Qualitative analysis involves visualizing the models' predictions on random test images to understand their performance better:

- **Random Test Images:** A selection of random test images is used to compare the models' predictions against the true labels.
- **Error Analysis:** Consistent errors or biases in the models' predictions are identified. For example, if certain animal classes are frequently misclassified, this may indicate a need for further model tuning or dataset balancing.
- **Visualization:** Displaying the images along with the predicted and true labels helps in diagnosing specific issues and understanding the model's decision-making process.

6. Conclusion

The Animal Multiclassification Project has shown that both custom and pretrained CNN models can effectively classify images of various animal species. By implementing thorough data preprocessing and augmentation, designing robust model architectures, and optimizing training processes, the project achieved meaningful results in animal classification.

Despite moderate performance, the insights gained highlight several areas for improvement. Balancing the dataset and employing more advanced data augmentation techniques are crucial next steps to enhance model robustness and accuracy. Additionally, fine-tuning the models, particularly the pretrained ResNet50v2, can further optimize their performance for specific classification tasks.

Moving forward, expanding the dataset with more diverse animal images and exploring different model architectures and hyperparameters will be vital. These efforts will contribute to the development of even more precise and reliable animal classification systems.

Overall, this project underscores the potential of deep learning in advancing animal classification, with significant implications for wildlife monitoring, ecological research, and various technological applications. The findings provide a solid foundation for future work, paving the way for ongoing improvements and broader applications of CNN-based classification models.

7. Bibliography

- Dataset Source: https://www.kaggle.com/datasets/alessiocorrado99/animals10
- Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the International Conference on Learning Representations (ICLR).
- https://arxiv.org/abs/2210.07973
- Animal Classification Project, GitHub Link: https://github.com/alizade34/AnimalClassification
- ChatGPT. (2024). Assistance with syntax and algorithms for code. OpenAI.