Automated Wildlife Monitoring - Progress Report

A. Introduction and Problem Statement

Entire fields of study, including biology, ecology, and wildlife conservation, depend heavily on the correct classification of animals. Automated species identification has long been needed because of their various morphologies, extensive distribution, and complex ecological roles. In these fields, correctly classifying animal species is still a challenging undertaking despite advances. [9] This problem is critical because it helps scientists understand and track animal species diversity, which is important for managing ecosystems and developing conservation plans. However, it can be quite difficult to identify between different animal species because of the intrinsic variety within species and the intricate ecological connections. A potential solution to this problem is deep learning, a potent method that has been shown to perform well in image classification tasks across a wide range of domains. By harnessing deep learning algorithms, we can significantly improve the accuracy and efficiency of animal species classification, thereby enhancing our understanding of biodiversity and supporting conservation efforts.

We chose a general strategy for tackling the issue. The proposed strategy is defined as follows:

- **Automated Wildlife Monitoring Development:** The project's goal is to create an automated monitoring system by utilizing machine learning and training dataset using models. Large amounts of labelled data and images from three different datasets will be utilized for training our model.
- **Model Selection:** VGG16, ResNet18, and GoogLeNet are three well-known convolutional neural network (CNN) architectures that have been carefully chosen for the project's objectives and difficulties.
- **Dataset Selection:** Three different datasets are selected to offer a strong basis for investigation, with differing volumes and levels of complexity.
- **Pre-processing and Model Training:** Pre-processing procedures including feature extraction, normalization, and data cleaning will be applied to the chosen datasets. These pre-processed datasets will then be used to train the models, and transfer learning and other approaches will be used to improve accuracy and efficiency.
- **Evaluation and Optimization:** Each model's performance will be assessed using metrics including recall, F1 score, accuracy, precision and ROC curve. Confusion matrices will also be used to conduct a thorough analysis of the models' performance.

Obtaining datasets on wildlife from Kaggle presents considerable difficulties for automated monitoring systems. These datasets frequently fall short of adequately representing the wide range of species found in natural settings due to a lack of specificity and variety. They might not include all range of environmental circumstances and settings found in the wild. A significant amount of processing power is needed to train models on these datasets, particularly for sizable datasets with many of classes or high-resolution photos. Moreover, modifying models to efficiently generalize across different datasets while minimizing over-fitting continues to be a significant obstacle.

Model durability can be increased and over-fitting can be avoided by utilizing strategies like regularization and feature engineering. Furthermore, scalability and computational efficiency of the established models must be guaranteed for large-scale monitoring projects to be implemented in the actual world.

The project promises insightful information about ecosystem health and biodiversity. [4] Comprehensive analysis using measures like accuracy, precision, recall, and F1 score will be used in the evaluation, along with visualization tools like confusion matrices and ROC curves. Ongoing optimization guarantees the prototype's performance in actual animal monitoring situations, opening the door for significant conservation initiatives.

B. Proposed Methodologies

Datasets

The three selected datasets comprise highly rated animal datasets sourced from Kaggle. Dataset-1 consists of 10 distinct classes, [6] each containing an average of approximately 2600 images per class, totaling 26000 images. Dataset-2 includes 20 classes evenly distributed, with 60 images allocated per class, resulting in a total of 1200 images. [2] Dataset-3 encompasses 30 classes with varying distributions, culminating in a total of 4500 images. [**?**] [1]

CNN Models

The project employed three widely-used convolutional neural network (CNN) architectures: ResNet18, VGG16, and GoogleNet, to develop and evaluate a total of 12 model instances. Among these, nine instances underwent training from scratch using three distinct datasets for each CNN architecture. Additionally, three transfer learning models were trained specifically using Dataset 1. Hence, the analysis encompasses the performance of these 12 model instances. The choice of these three architectures was guided by an analysis of popular CNN models tailored for object identification/classification, considering their performance and training duration. [10]

ResNet18 is a simple architecture with 18 layers which

Figure 1. Image Samples

includes skip connections to prevent vanishing gradients. VGG16 is a very deep architecture with 16 layers and uses small filters, max-pooling, and many parameters for feature extraction. GoogleNet is a complex architecture with diverse filter sizes, pooling layers, and inception modules for efficient feature extraction and pattern recognition.

ResNet18's faster execution compared to other ResNet versions, along with its ability to maintain low error rates deeper into the network with fewer flops, adds to its appeal. GoogleNet is chosen for its design, which notably improves the model's capacity to detect intricate patterns and structures in images, rendering it well-suited for tasks demanding heightened accuracy, such as this project. In contrast, VGG16's utilization of 3x3 convolutional filters throughout the network stands out as a pivotal feature, reducing parameter count and enhancing computational efficiency.

C. Attempts At Solving The Problem

Failed attempt: Initially dataset of 30 classes produced a model that was 100% accurate but also showed an extremely high test loss. This seeming contradiction showed a typical example of over-fitting, a dangerous machine learning trap in which the model fails to generalize well to new, unseen data because it has grown too specialized in identifying patterns within the training data. Even though the model was able to achieve 100% accuracy on the training set, its fundamental restriction was exposed when it was unable to perform effectively on the test set. When a model overfits, it performs poorly when given new instances because it effectively memorizes the training data instead of understanding the underlying patterns and relationships.

Successful Attempt: The models performed better on the test set for the 10 class dataset because they showed a more comprehensive understanding of the fundamental trends. Similar to that, for the dataset with 20 classes, the improved models showed more flexibility to the wider range of classes and a successful result was achieved by strategically changing the 30 classes dataset in response to the difficulties faced during the initial training phase. Even though the model's accuracy may not have been as high as it was in the beginning, this modification successfully dealt with the overfitting problem that was seen in the dataset with 30 classes. The models were exposed to a wider variety of characteristics and variations by expanding the dataset or providing a more diverse group of instances, which improved their generalization skills.This shows the importance of careful dataset selection and model evaluation to achieve significant outcomes in machine learning tasks.

Possible/Preliminary Results: Different CNN architectures and datasets show different levels of accuracy, according to the performance study. Interestingly, GoogleNet frequently performs better than ResNet and VGG16, with the highest accuracies in the majority of situations, indicating that it is more efficient at catching the complexity seen in the datasets of animal images. But accuracy tends to decline with the number of classes, suggesting that it becomes more difficult to differentiate between more groups. Reasonable loss reductions are shown in the training factors, suggesting efficient learning without significant overfitting. Training effectiveness is increased when GPU acceleration is enabled via CUDA. Performance could be enhanced by investigating other designs and fine-tuning hyperparameters.

D. Future Improvements

The following improvements can be made to the program in the upcoming phases where we will implement:

- **Data Augmentation:** We plan to boost our model's performance by introducing data augmentation techniques such as rotation, flipping, scaling, and color jittering. This approach will diversify our training data, compensating for the smaller size of Dataset 2, and help the model generalize better across varied wildlife images, ultimately improving accuracy.

- **Early Stopping:** To counteract over0fitting, especially in cases with smaller datasets and numerous classes, we'll implement early stopping. By halting training when validation loss begins to rise, we can avoid overfitting and ensure our model remains generalized, thus yielding a more accurate outcome.

- **Hyperparameter Tuning:** Adjusting the model's hyperparameters, including the learning rate, optimizer type, batch size, and dropout rates, can significantly impact its performance. We intend to use methods like grid or random search to identify the optimal hyperparameter settings, enhancing the model's accuracy and overall performance.

References

[1] Asaniczka. Mammals Image Classification Dataset - 45 Animals. https://www.kaggle.com/datasets/asaniczka/mammals-image-classification-dataset-45-animals, 2024. Accessed: 2024-03-24. 1

[2] Sourav Banerjee. Animal Image Dataset: 90 Different Animals. https://www.kaggle.com/datasets/iamsouravbanerjee/animal-image-dataset-90-different-animals, 2024. Accessed: 2024-03-24. 1

[3] Sreedevi C K and Saritha E. Automated wildlife monitoring using deep learning. In *Proceedings of the International Conference on Systems, Energy & Environment (ICSEE) 2019*, GCE Kannur, Kerala, July 2019. Available at SSRN: https://ssrn.com/abstract=3447740 or http://dx.doi.org/10.2139/ssrn.3447740. 3

[4] Michael L. Casazza, Austen A. Lorenz, Cory T. Overton, Elliott L. Matchett, Andrea L. Mott, Desmond A. Mackell, and Fiona McDuie. Aims for wildlife: Developing an automated interactive monitoring system to integrate real-time movement and environmental data for true adaptive management. *Journal of Environmental Management*, 345:118636, 2023. 1

[5] Ruilong Chen, Ruth Little, Lyudmila Mihaylova, Richard Delahay, and Ruth Cox. Wildlife surveillance using deep learning methods. *Ecology and Evolution*, 9(17):9453–9466, 2019. 3

[6] Alessio Corrado. Animals-10 dataset. https://www.kaggle.com/datasets/alessiocorrado99/animals10, 2019. Accessed: 2023-02-07. 1

[7] Vladimir Dyo, Stephen A. Ellwood, David W. Macdonald, Andrew Markham, Cecilia Mascolo, Bence Pásztor, Salvatore Scellato, Niki Trigoni, Ricklef Wohlers, and Kharsim Yousef. Evolution and sustainability of a wildlife monitoring sensor network. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, SenSys '10, page 127–140, New York, NY, USA, 2010. Association for Computing Machinery. 3

[8] Sergio Marrocoli, Martin Reinhardt Nielsen, David Morgan, Tom van Loon, Lars Kulik, and Hjalmar Kühl. Using wildlife indicators to facilitate wildlife monitoring in hunter-self monitoring schemes. *Ecological Indicators*, 105:254–263, 2019. 3

[9] Hung Nguyen, Sarah J. Maclagan, Tu Dinh Nguyen, Thin Nguyen, Paul Flemons, Kylie Andrews, Euan G. Ritchie, and Dinh Phung. Animal recognition and identification with deep convolutional neural networks for automated wildlife monitoring. In *2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 40–49, 2017. 1

[10] Shagun Sharma and Kalpna Guleria. Deep learning models for image classification: Comparison and applications. In *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, pages 1733–1738, 2022. 1

[11] U.S. Department of Agriculture, Forest Service. Advances in technology for automating analyses of wildlife monitoring videos. https://www.fs.usda.gov/research/pnw/news/highlights/advances-technology-automating-analyses-wildlife-monitoring-videos, 2023. Accessed: 2023-02-07. 3

[12] Jens Wawerla and Greg Mori. Robot identification of human activities using trajectory information. https://www.cs.sfu.ca/~mori/research/papers/wawerla_bearcam_jmva08.pdf, 2008. Accessed: 2023-02-07. 3

[13] Worley. Using automation to make wildlife monitoring easier. https://www.worley.com/en/solutions/case-studies/digital-and-technology/using-automation-to-make-wildlife-monitoring-easier, 2023. Accessed: 2023-02-07. 3

[14] Meilun Zhou, Jared A. Elmore, Sathishkumar Samiappan, Kristine O. Evans, Morgan B. Pfeiffer, Bradley F. Blackwell, and Raymond B. Iglay. Improving animal monitoring using small unmanned aircraft systems (suas) and deep learning networks. *Sensors*, 21(17), 2021. 3