Wildlife Classification using Convolutional Neural Networks (CNN)

Vijay Mane
Department of Electronics & Telecommunication Engineering
Vishwakarma Institute of Technology, Pune.
vijay.mane@vit.edu

Tejas Patil
Department of Electronics & Telecommunication Engineering
Vishwakarma Institute of Technology, Pune.
tejas.patil21@vit.edu

Pranjali Nikude
Department of Electronics & Telecommunication Engineering
Vishwakarma Institute of Technology, Pune.
ishwar.pranjali21@vit.edu

Prashik Tambe
Department of Electronics & Telecommunication Engineering
Vishwakarma Institute of Technology, Pune.
prashik.tambe21@vit.edu

2024 International Conference on Inventive Computation Technologies (ICICT) | 979-8-3503-5929-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICICT60155.2024.10544702

Abstract: The rapid advancements in computer vision and deep learning techniques have opened new avenues for wildlife monitoring and conservation efforts. This project addresses the critical need for efficient and automated wildlife detection and classification using Convolutional Neural Networks (CNN). The primary objective of this research study is to develop a robust CNN-based model capable of accurately identifying and categorizing diverse wildlife species from images. The dataset utilized for training and evaluation comprises a comprehensive collection of high-resolution images, encompassing various environmental conditions and wildlife habitats. The Convolutional Neural Network architecture employed is tailored to capture intricate patterns and features inherent in wildlife images. Training the model involves optimizing key hyperparameters, utilizing data augmentation strategies, and fine-tuning the network to enhance its generalization capabilities. The evaluation of the proposed model involves assessing its accuracy, validation loss etc. on a separate test dataset. Additionally, a detailed analysis of the model's performance on individual wildlife classes is presented. The project aims to contribute to wildlife conservation by providing a reliable and automated tool for monitoring and identifying species, thereby facilitating timely interventions and resource allocation. The outcomes of this research hold promise for real-world applications, including ecological studies, wildlife management, and the development of monitoring systems. The integration of advanced technologies in wildlife conservation not only improves the efficiency of data collection but also fosters a deeper understanding of ecological dynamics, ultimately contributing to the preservation of biodiversity.

Keywords: Convolutional Neural Networks (CNN), Computer Vision, Deep Learning, Image Classification, Wildlife Species Identification, Data Augmentation.

I. INTRODUCTION

Convolutional Neural Networks (CNNs), which are well-known for their simplicity and precision, have set new benchmarks in visual image tasks. These deep neural networks excel at image categorization by utilizing relatively simple algorithms. Their main advantage is their ability to learn features on their own without prior human understanding. CNNs are widely used in image and video identification, and recommendation systems, demonstrating their versatility and impact across multiple areas. [1]

A network with stationary camera traps is created by installing sensor cameras atop trees in a specific region to monitor wildlife. Whenever motion is recognized, the camera traps are set to record brief recordings of the animals' actions together with data on their surroundings, including temperature, humidity, light intensity, and position. Camera-trap networks are necessary for the disturbance-free gathering of wildlife data.

This work introduces a CNN-based method for wildlife animal detection using camera-trap images. Overcoming challenges in object segmentation from dynamic backgrounds, this approach focuses on motion-based analysis from image sequences. Unlike conventional methods, which struggle with complex scenes featuring swinging trees, waving water, and changing weather, the proposed model is designed to efficiently detect animals in such dynamic environments. [6] Even though it is quite old, CNN offers overall better results and accuracy. Also, CNN tries to access the data in a different way than the other variants which is considered to be more human-like.

The main objective of the system is to detect species of animals recorded in the camera, know if the particular species is endangered or needs conservation, and take quick actions accordingly.

II. LITERATURE REVIEW

Amplitude spectra make it easier for people to locate animals in photos, even when it is obscured or set against cluttered backdrops. This occurs because amplitude spectra provide the brain with information about the texture and form of objects. Research indicates that individuals can locate animals more quickly when amplitude spectra are normal than jumbled ones. Although the precise mechanism by which amplitude spectra aid in vision is still unknown, scientists believe it may be because it enables the brain to see objects more rapidly by allowing it to focus on their basic forms rather than all their finer features [1].

The study by N. Banupriya et al. demonstrates the use of state-of-the-art technology in the sector by applying deep learning algorithms for animal detection. The study uses deep learning algorithms to analyze photos and video frames with an emphasis on computer vision and image recognition with the goal of identifying and classifying different animal species. This technical approach promises improved accuracy and efficiency in animal identification systems, marking a substantial shift from conventional methods [2].

Parham et al. (2023) present an innovative approach to animal detection and identification in wildlife images. This particular paper proposed a 5-component detection pipeline that efficiently locates, classifies, and segments animals of interest, ultimately generating annotations of interest (AoI) with species and viewpoint labels. This pipeline demonstrates promising results on the Wildlife Image and Localization Dataset (WILD), achieving high accuracy in animal detection, species classification, and AoI prediction. The proposed method holds the potential to enhance animal censusing studies and provide valuable ecological insights for conservation efforts[3].

Anankware et al. (2016) conducted a comprehensive survey to identify and classify common edible insects in Ghana, emphasizing their potential as a valuable source of nutrition and food security. Through extensive fieldwork and interviews with local communities, nine readily consumed insect species were documented. The study highlights the widespread consumption of insects across various ethnic groups in Ghana, demonstrating their cultural significance and nutritional value. These insects are rich in protein, vitamins, minerals, and essential amino acids, offering a promising alternative to traditional protein sources[4].

To overcome the drawbacks of conventional techniques, KnausgŤard et al. (2020) present a novel deep-learning approach for identifying and categorizing temperate fish species without pre-filtering. The two-stage method uses a CNN with SE architecture for classification and YOLO for fish detection. The proposed method performs remarkably well when tested on the Fish4Knowledge dataset, achieving 99.27% detection and 83.68% classification accuracy, which increases to 87.74% with image augmentation.[5].

A deep convolutional neural network (DCNN) based method for identifying wild animals in natural settings is presented by Verma and Gupta (2018). Their technique first finds candidate animal proposals using a multi-level graph cut algorithm, and then it uses a DCNN for classification. The method used by the authors shows how successful DCNNs are at identifying wild animals. Their approach is a useful tool for many applications, including ecological monitoring and conservation biology, since it can attain high accuracy on difficult datasets[6].

Trowbridge and Mintzes (1999) explored the prevalence of alternative conceptions in animal classification among students of different age groups, which examined the understanding of three key concepts: vertebrate vs. invertebrate, warm-blooded vs. cold-blooded, and mammal vs. non-mammal. Results indicated that alternative conceptions persisted across all grade levels, highlighting the challenges in students' grasp of animal classification[7].

This paper proposes a deep learning approach for identifying animal species in camera-trap images using very deep convolutional neural networks (CNNs), particularly residual networks. The method incorporates data augmentation, multi-scale classification, and top-k predictions to address pose variations, illumination changes, and occlusion challenges. Achieving 88.9% accuracy for top species and 98.1% for top-5, this method outperforms previous approaches and paves the way for automated wildlife monitoring with significant implications for conservation and research. This is helpful because analyzing large amounts of camera-trap data manually is time-consuming.[8]

Existing methods for detecting animals, like those based on motion, struggle with complex environments. This paper by Munian et al. proposes a new approach for automobiles. It combines a Histogram of Oriented Gradients (HOG) with Convolutional Neural Networks (CNNs). CNNs are powerful machine-learning tools that can learn complex patterns from data, potentially leading to more accurate animal detection on roads.[25]

Manual analysis of camera trap images for wildlife monitoring is time-consuming. This paper uses deep neural networks (DNNs) to automate species identification from camera trap images in Texas. DNNs excel at recognizing patterns in data, making them suitable for this task. This approach has the potential to significantly improve efficiency and accuracy in wildlife conservation efforts.[26]

III. METHODOLOGY

Data collection and pre-processing:

A freely available dataset is downloaded from Kaggle, which has 5 different animal categories in it. The entire dataset consists of around two thousand different images including all five categories. All the images in the dataset are resized to a convenient size that makes the image visible as well as keeps its size limited. Also, the dataset includes images of animals from different geographic regions which increases the overall accuracy of the model. All the images are further converted into arrays so that all the operations can be performed on them. Later this data in the form of arrays is stored in a pickle file to be used in the next process. After pre-processing, the image data is divided into two pickle files X and Y using numpy arrays. X contains the dimensions of each image while Y contains the labels for each image.

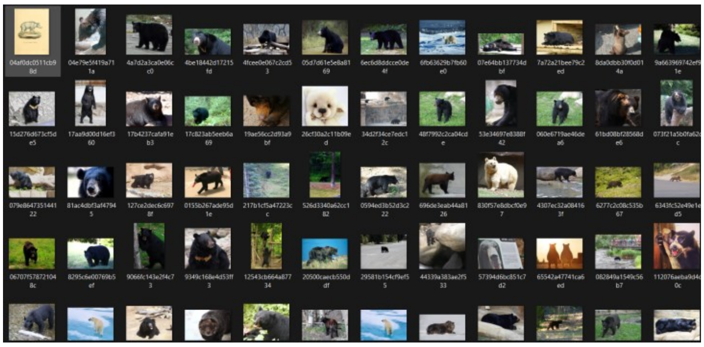

Fig 1: Bear Dataset

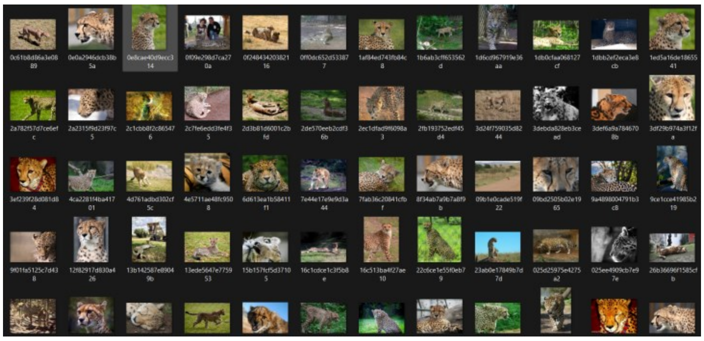

Fig 2: Cheetah Dataset

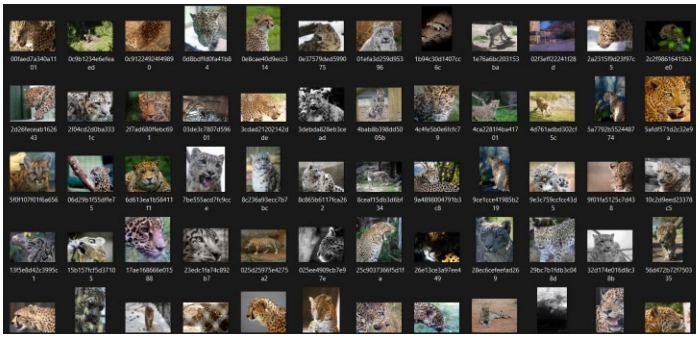

Fig 3: Leopard Dataset

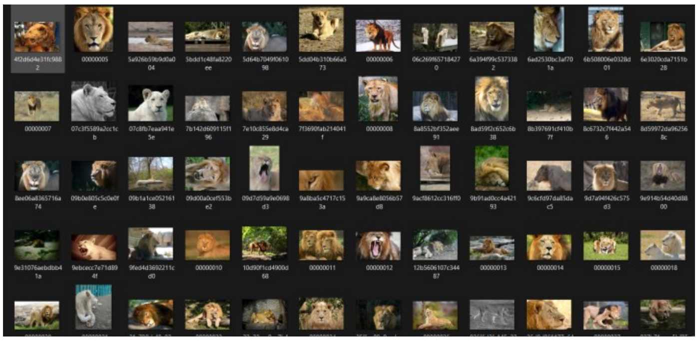

Fig 4: Lion Dataset

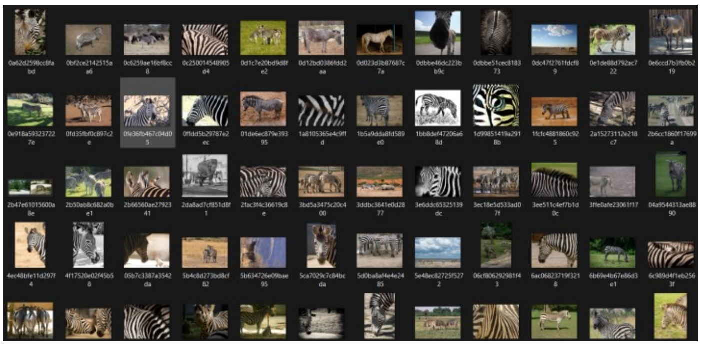

Fig 5: Zebra Dataset

Data Augmentation:

After pre-processing, the data is augmented. Various operations are performed on the images of the dataset. The images are rotated, scaled, flipped etc. Performing all these operations on the images play a big role in increasing the accuracy of the model

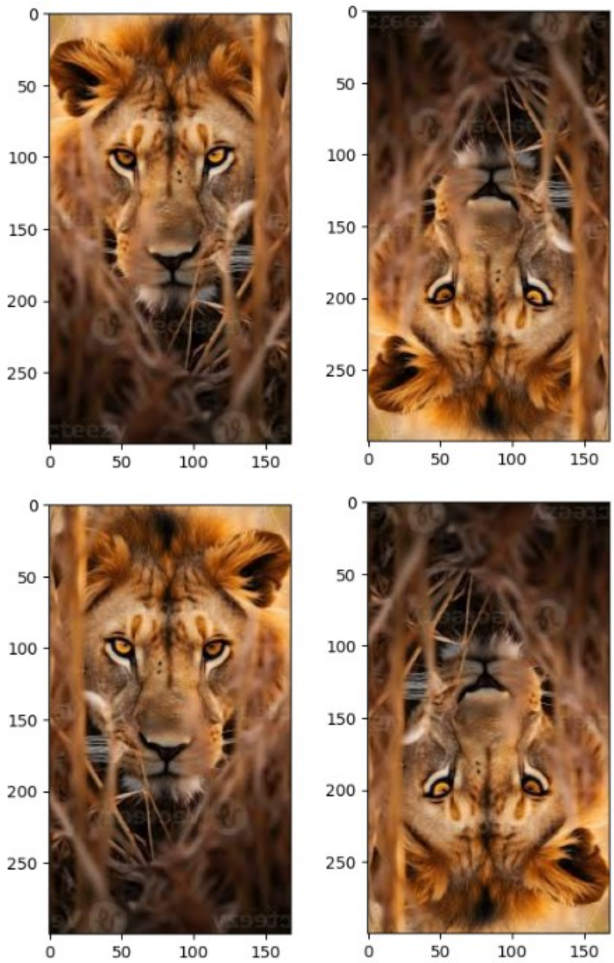

Fig 6: Rotating, scaling and flipping of image

Training of Data:

A new notebook file is created to train the data. The pre-processed data is loaded from the two pickle files. To make further operations on the data, the X array(the array that includes the dimensions of the images) is divided by 255, making all the values lay between 0 and 1. Various models are created using different layers in different proportions to train the data. All these models are graphically studied using the tensorboard visualization toolkit provided by the tensorflow library. The model with high accuracy is selected to train the data.

Convolutional Layer:

A convolutional layer is a class of deep-learning models designed for processing images. Conv2D is a 2-Dimensional conventional layer suitable for processing images. 64 filters were used in this convolutional layer. The size of each convolutional kernel(filter) is 3x3. All these 64 filters are used to detect different features in input data or to locate particular patterns in the input. These filters are slid over the input data and at each position, the filter undergoes element-wise multiplication followed by summation, producing a single value as output. 2 such convolutional 2D layers along with maxpooling layers are added to the model. Here is the formula used in the Convolution Layer.

$$Y^k_{ij} = m\sum n\sum X_{i+m,j+n} \cdot F^k_{mn} + b^k$$

- Y^k_{ij} is the output of the k^{th} filter at position (i,j).
- F^k is the k^{th} filter kernel
- b^k is the bias for the k^{th} filter
- The sums over m and n iterate over the spatial dimensions of the filter.

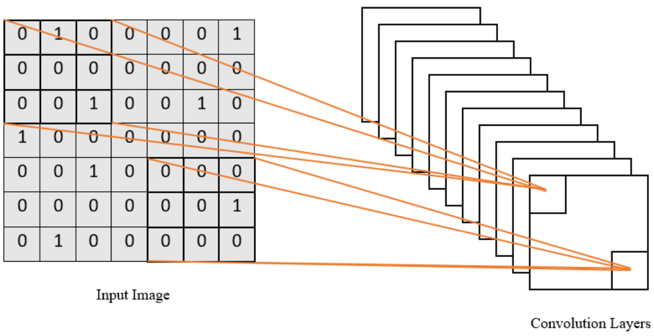

Fig 7: Convolutional Layer

Max Pooling:

MaxPooling is a downsampling operation commonly used in CNNs. It is a technique to reduce the spatial dimensions of a given input volume for a neural network. It involves dividing the input into non-overlapping regions and retaining the maximum value within each region. It accomplishes this by down sampling the input, so decreasing its size while maintaining crucial information.

By eliminating extraneous information, this down sampling technique reduces dimensionality while maintaining dominant features, improving computational efficiency and avoiding overfitting. By choosing the most important data from each pooling window, max pooling also adds a degree of translation invariance and promotes reliable feature detection independent of positional changes in the input. To achieve the best model performance, information retention and dimension reduction should be carefully balanced. This is because of the model's intrinsic information loss caused by rejecting non-maximal values, as well as the choice of pooling size and stride. The size of pooling window was decided to be 2x2. A total of 2 maxpooling layers along with con2D layers are used in the model. Here is the formula used in the pooling layer.

$$P^k_{ij} = \max_{(m,n)\in[2x2]} \cdot Z^k_{2i+m,2j+n}$$

This operation selects the maximum value from a 2x2 block.

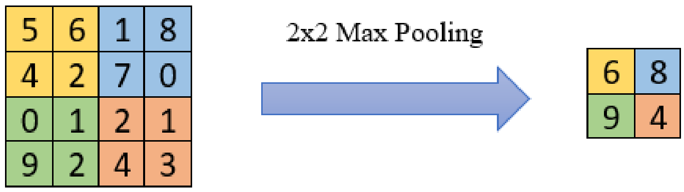

Fig 8: Maxpooling Layer

Flattening Layer:

In order to prepare the data for additional processing in the ensuing dense layers, the Flattening Layer, also known as the Global Pooling Layer, is a fundamental component of CNNs. It does this by converting multi-dimensional feature maps into a one-dimensional format, minimising computational load, and preserving important features that were extracted from earlier convolutional operations. After using all these layers the model is compiled.

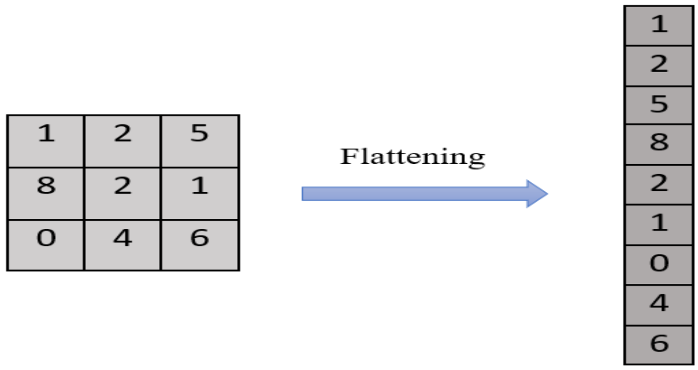

Fig 9: Flattening Layer

Output Layer

In a neural network, the output layer is in charge of generating the final predictions or outputs based on the learnt representations from earlier layers. Whether it's for classification, regression, or other prediction tasks, the network's configuration, activation functions, and loss functions are customised for the particular task at hand.

There are various ways to increase the accuracy of the model. Increasing the size of dataset for the model to train on is one of the easiest way to increase the accuracy. The model can be changed by using different layers and changing the number of layers in the model.

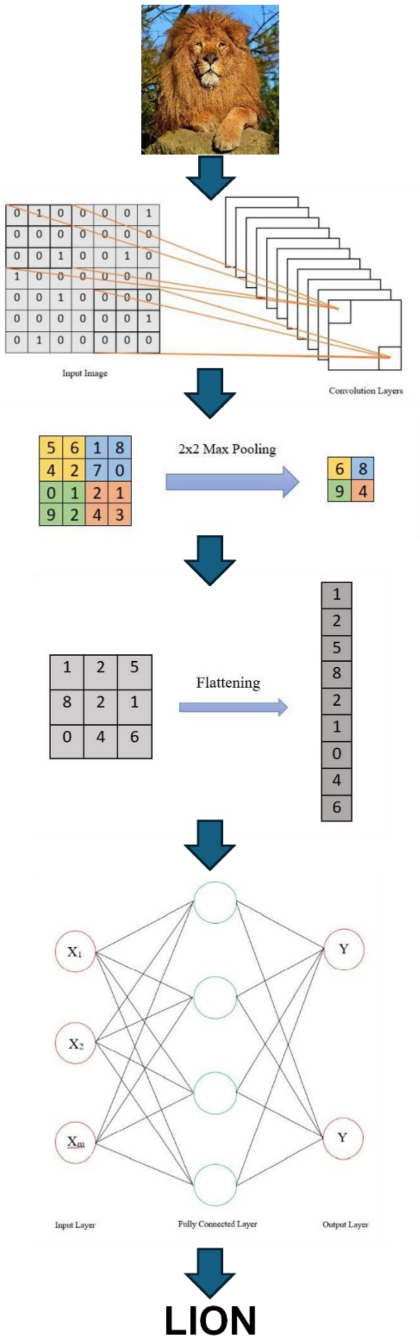

Fig 10. Flow Diagram

Tools Used

- numpy
- cv2
- os
- random
- matplotlib
- pickle
- tensorflow
- tensorboard
- keras
- gradule

Architecture

The proposed CNN-based model for wildlife categorization uses a dataset of photos from five animal categories that have been preprocessed with scaling and augmentation techniques to improve variability. The model architecture consists of several layers, including convolutional layers with 64 filters for feature identification, maxpooling layers for downsampling, and a flattening layer for formatting. Training entails optimizing hyperparameters and selecting the most accurate model, which is monitored using TensorBoard visualization. Evaluation parameters such as training and validation losses and accuracies indicate successful model optimization and generalization, with training accuracy reaching almost 97% and validation accuracy approaching 93%.

These results show the model's effectiveness at automating species identification, making it a good candidate for real-time wildlife monitoring. Expanding the dataset with different categories of animals and experimenting with advanced techniques such as attention mechanisms and ensemble learning could result in enhanced performance and interpretability.

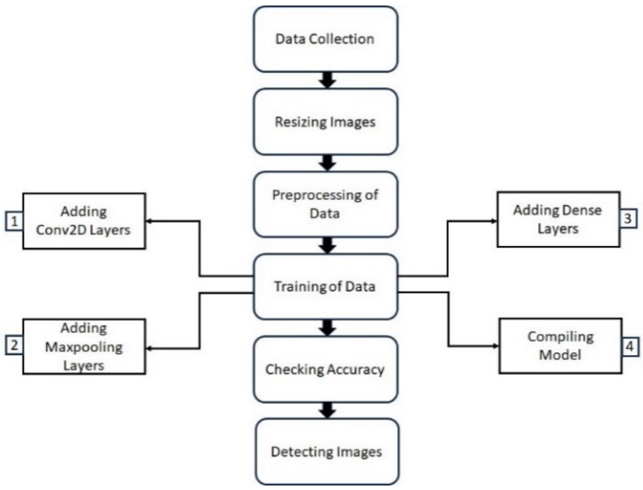

Fig 11. Block Diagram of the system

IV. RESULTS AND DISCUSSION

The model's training accuracy of almost 97% (Table 1) was one of the encouraging results of the training phase. This high accuracy shows that the complex patterns and characteristics found in the training dataset were successfully. Concurrently, the corresponding training loss shows that the model was successfully optimized during the training phase, converging to approximately 6%. As the network processed the training data iteratively, the decreasing training loss is a sign that the network is improving its predicting accuracy and minimizing errors.

This model demonstrated an impressive 93% validation accuracy (Table 1). This demonstrates how well it can classify a wide range of objects or features in unknown data. It could be useful in handling a variety of situations. However, caution regarding any problems that could affect the segmentation quality, such as unequal class representation or complex real-world circumstances should be exercised. These are important criteria to take into account when evaluating the model's performance.

	(10 epoch)
Training Loss	11%
Training Accuracy	97%
Validation Loss	18%
Validation Accuracy	93%

Table 1: Training Results

Training loss: The training loss is like a score that checks how well a computer is learning from a bunch of examples. By plotting a curve of this loss after each epoch, it helps to track and aim to decrease errors, ensuring the computer gets better at its task.

Validation loss: Validation loss is like a measure of how often a computer model makes mistakes when tested on a different set of examples. It helps us see how well the model can handle new, unseen data.

Training accuracy: Training accuracy shows how good a machine learning model is on the training data, indicating the percentage of correct predictions it makes during learning. It reflects how well the model learns from the given examples in the training set.

Validation accuracy: Validation accuracy tells us how good a machine learning model is at making correct predictions on new data it hasn't seen before. It's like a percentage that shows how well the model can handle fresh examples in a separate validation set.

CNNs, unlike existing and traditional machine learning algorithms that rely on specified features, generate hierarchical representations utilising convolutional layers (such as max pooling, output layer, and flattening layer), allowing them to recognise subtle patterns in images. Existing methods, such as manually built feature extraction followed by classification algorithms such as Support Vector Machines (SVMs) or decision trees, necessitate extensive preprocessing and feature engineering, which can be computationally demanding and time-consuming. In contrast, CNNs may automatically learn hierarchical representations of features directly from raw data, removing the need for manual feature engineering and preprocessing. This automated feature learning process not only eliminates computational overhead but also increases efficiency by leveraging parallel processing and hardware acceleration techniques such as Graphics Processing Units (GPUs) or specialized hardware like Tensor Processing Units (TPUs).While classical algorithms may be easier to interpret and require less labelled data, CNNs are more adaptable and flexible to many tasks and domains, making them the ideal choice for image classification and recognition in a variety of applications.

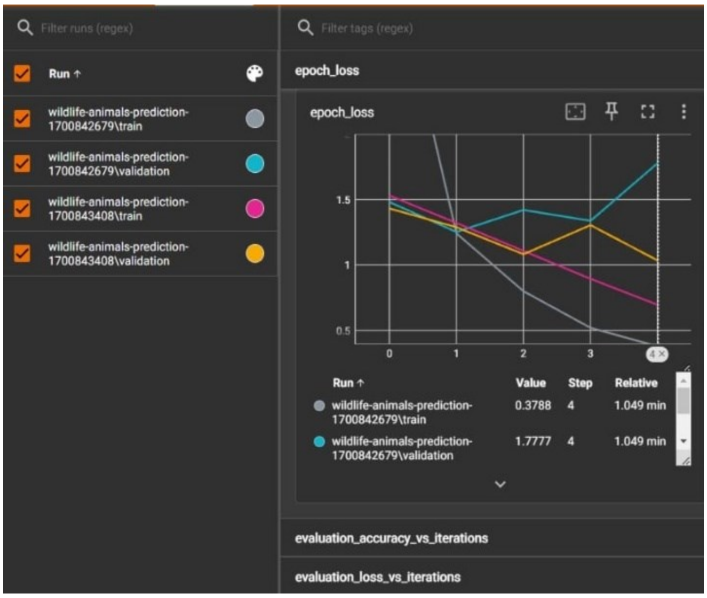

Fig 12: Graphs of training models - loss

Fig 13: Graphs of training models - accuracy

Fig 14: Output of the model - Zebra

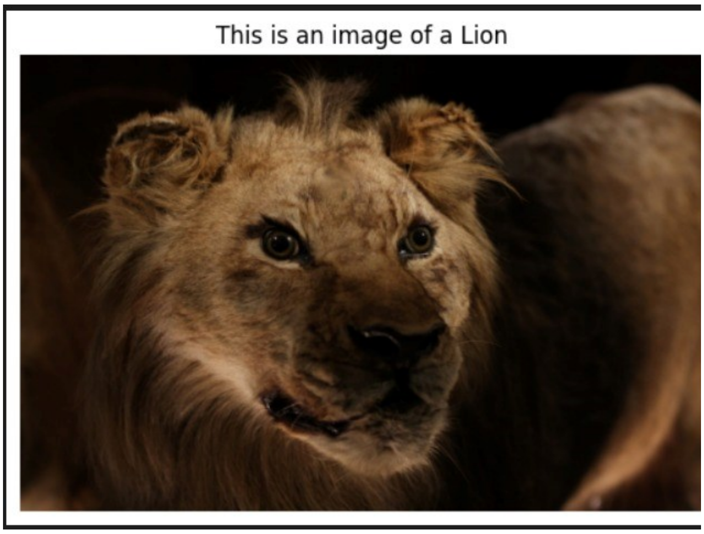

Fig 15: Output of the model - Lion

5. CONCLUSION & FUTURE SCOPE

During testing, the CNN, which had been fine-tuned and optimised with TensorFlow and Keras, displayed considerable accuracy, validation loss indicating its efficacy in automated species identification. In the future, dataset can be expanded for more accuracy and precision like adding more images and categories of animals, fine-tune hyperparameters, and investigate advanced techniques like as attention mechanisms and ensemble learning to further improve the model's performance and interpretability.

Furthermore, the incorporation of proposed model into real-time monitoring systems holds potential for real-time species identification. Using TensorFlow and Keras not only aided the smooth construction of the model, but also offers up possibilities for deploying it on edge devices, decreasing reliance on centralised compute resources. Ethical factors, such as privacy concerns and collaboration with conservation organisations, will be critical in ensuring that automated wildlife monitoring systems are used responsibly and sustainably. CNNs offer a wide range of applications in wildlife protection. It may be used to recognise and count animals in camera trap photos, which can help track population changes and identify poaching hotspots. CNN can be used to map different types of habitats, which can aid in assessing ecosystem health and identifying regions at danger of degradation. CNNs can also be used to detect sickness in animals, track disease spread, and identify animals that enter human-occupied regions, which can help reduce human-wildlife conflict.

REFERENCES

[1] Gaspar, Carl M., and Guillaume A. Rousselet. "How do amplitude spectra influence rapid animal detection?." Vision research 49, no. 24 (2009): 3001-3012.

[2] Banupriya, N., S. Saranya, Rashmi Swaminathan, Sanchithaa Harikumar, and Sukitha Palanisamy. "Animal detection using deep learning algorithm." J. Crit. Rev 7, no. 1 (2020): 434-439.

[3] Parham, Jason, Charles Stewart, Jonathan Crall, Daniel Rubenstein, Jason Holmberg, and Tanya Berger-Wolf. "An animal detection pipeline for identification." In 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1075-1083. IEEE, 2018.

[4] Anankware, Jacob P., Enoch A. Osekre, Daniel Obeng-Ofori, and Canute Khamala M. "Identification and classification of common edible insects in Ghana." (2016).

[5] Knausgård, Kristian Muri, Arne Wiklund, Tonje Knutsen Sørdalen, Kim Tallaksen Halvorsen, Alf Ring Kleiven, Lei Jiao, and Morten Goodwin. "Temperate fish detection and classification: a deep learning-based approach." Applied Intelligence (2022): 1-14.

[6] Verma, Gyanendra K., and Pragya Gupta. "Wild animal detection using deep convolutional neural network." In Proceedings of 2nd International Conference on Computer Vision & Image Processing: CVIP 2017, Volume 2, pp. 327-338. Springer Singapore, 2018.

[7] Trowbridge, John E., and Joel J. Mintzes. "Alternative conceptions in animal classification: A cross-age study." Journal of Research in Science Teaching 25, no. 7 (1988): 547-571.

[8] Villa, Alexander Gomez, Augusto Salazar, and Francisco Vargas. "Towards automatic wild animal monitoring: Identification of animal species in camera-trap images using very deep convolutional neural networks." Ecological informatics 41 (2017): 24-32.

[9] Schindler, Frank, and Volker Steinhage. "Identification of animals and recognition of their actions in wildlife videos using deep learning techniques." Ecological Informatics 61 (2021): 101215.

[10] Peng, Jinbang, Dongliang Wang, Xiaohan Liao, Quanqin Shao, Zhigang Sun, Huanyin Yue, and Huping Ye. "Wild animal survey using UAS imagery and deep learning: modified Faster R-CNN for kiang detection in Tibetan Plateau." ISPRS Journal of Photogrammetry and Remote Sensing 169 (2020): 364-376.

[11] Nguyen, Hung, 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), pp. 40-49. IEEE, 2017.

[12] Kellenberger, Benjamin, Michele Volpi, and Devis Tuia. "Fast animal detection in UAV images using convolutional neural networks." In *2017 IEEE international geoscience and remote sensing symposium (IGARSS)*, pp. 866-869. IEEE, 2017.

[13] Chen, Guobin, Tony X. Han, Zhihai He, Roland Kays, and Tavis Forrester. "Deep convolutional neural network based species recognition for wild animal monitoring." In *2014 IEEE international conference on image processing (ICIP)*, pp. 858-862. IEEE, 2014.

[14] Chandrakar, Ramakant, Rohit Raja, and Rohit Miri. "Animal detection based on deep convolutional neural networks with genetic segmentation." *Multimedia Tools and Applications* (2021): 1-14.

[15] Maiti, Rajarshi, Yi Hou, Colleen Cassady St Clair, and Hong Zhang. "Use of convolutional neural networks to automate the detection of wildlife from remote cameras." In *2015 IEEE 7th International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, pp. 42-47. IEEE, 2015.

[16] O'Shea, Keiron, and Ryan Nash. "An introduction to convolutional neural networks." *arXiv preprint arXiv:1511.08458* (2015).

[17] Surya, T., and S. Selvaperumal. "The IoT-based real-time image processing for animal recognition and classification using deep convolutional neural network (DCNN)." *Microprocessors and Microsystems* 95 (2022): 104693.

[18] Weinstein, Ben G. "Scene-specific convolutional neural networks for video-based biodiversity detection." *Methods in Ecology and Evolution* 9, no. 6 (2018): 1435-1441.

[19] Silva, Luís Carlos, Maricéia BS Pádua, Leonardo M. Ogusuku, Marcelo Keese Albertini, Renato Pimentel, and Andre R. Backes. "Wild boar recognition using convolutional neural networks." *Concurrency and Computation: Practice and Experience* 33, no. 22 (2021): e6010.

[20] Sanghvi, Kavish, Adwait Aralkar, Saurabh Sanghvi, and Ishani Saha. "Fauna image classification using convolutional neural network." *International journal of future generation communication and networking* 13, no. 1s (2020): 08-16.

[21] CK, Sreedevi. "Automated Wildlife Monitoring Using Deep Learning." In *proceedings of the International Conference on Systems, Energy & Environment (ICSEE)*. 2019.

[22] Zhong, Ming, Ruth Taylor, Naomi Bates, Damian Christey, Hari Basnet, Jennifer Flippin, Shane Palkovitz, Rahul Dodhia, and Juan Lavista Ferres. "Acoustic detection of regionally rare bird species through deep convolutional neural networks." *Ecological Informatics* 64 (2021): 101333.

[23] Zhou, Meilun, 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, no. 17 (2021): 5697.

[24] Gujjar, Sanjana V., and UG Scholar Shradha Bhat. "Wildlife Detection Using Convolutional Neural Network." (2023).

[25] Munian, Yuvaraj, Antonio Martinez-Molina, and Miltiadis Alamaniotis. "Intelligent system for detection of wild animals using HOG and CNN in automobile applications." In 2020 11th International Conference on Information, Intelligence, Systems and Applications (IISA, pp. 1-8. IEEE, 2020.

[26] Islam, Sazida B., and Damian Valles. "Identification of wild species in texas from camera-trap images using deep neural network for conservation monitoring." In 2020 10th Annual Computing and Communication Workshop and Conference (CCWC), pp. 0537-0542. IEEE, 2020.