Инженерный вестник Дона, №6 (2025)
ivdon.ru/ru/magazine/archive/n6y2025/10145

Unveiling hidden Patterns in Classifying Wildlife Images using Convolutional neural networks for Species Identification in Conservation Initiatives

D. Lwagula, I. Mirimu

National University of Science and Technology MISiS

Abstract: This study is a testament to the potential of convolutional neural networks in softmax activation to classify mantis, honey badger, and weasel samples. The model was able to predict highly with low misclassification and had the potential to reduce environmental variance by minimizing it using data augmentation. The research shows how deep learning networks would be used in the automation of taxonomic classification, which in turn would help species identification through images and large-scale conservation monitoring.
Keywords: deep learning, machine learning, convolutional neural networks, dataset, softmax function, image classification, wildlife, data augmentations.

Introduction

Convolutional Neural Networks (CNNs) revolutionized image classification over the past decade, dominating traditional machine [1] learning approaches to ease . Their relentless effectiveness is a result of their inherent capacity to learn in an autonomous manner to discover discriminating features directly from raw images [2, 3], forming resilient representations that form a basis of their enhanced performance. To date, this broad application is poorly comprehended at the mechanism level. This lack of interpretability presents a critical barrier to broader use in mission-critical applications like autonomous systems and medical diagnosis, where explainability is crucially [4] essential to ensure accountability and trust . This study suggests a novel methodological strategy that seeks to reveal the underlying processes in convolutional nueral networks by visualizing the complex computational processes in intermediate feature layers and classifier mechanisms, thereby [5] gaining insightful knowledge about the internal structure of these powerful neural networks .

Literature Review

The use of convolutional neural networks to automatically identify camera trap images in order to identify wildlife species and conservation status is explained in this article. By avoiding traditional gradient flow and processing restrictions with models like visual geometry group network(VGGNet), residual neural network (ResNet), AlexNet [6] and convolutional neural networks have largely revolutionized computer vision according to the study. The system pre-processes many wildlife image datasets and uses deep neural algorithms to function well in a wide range of conservation application scenarios. The study also uses explainable artificial inteligence [7] frameworks to address the difficulty of effectively handling growing amounts of animal picture data for conservation, as well as to enhance the interpretability of the model .

Methodology

Classification of images of three other animal species honey badgers, mantises, and weasels is addressed in this work. The images in the dataset are divided into training and validation datasets according to one [8] deep learning-based approach. Using metrics of convolutional nueral network model evaluation, data preprocessing, and model evaluation, one goal is to classify the images into one of these three classes. The three categories of animals within the data set are honey badgers, mantises, and weasels.There are a number of photos in each category taken across different locations.The training set is used to train the convolutional neural network model on patterns that correspond to each category. By testing the model's performance on previously unseen data, a validation set is used to ascertain the capability of the trained model [9] to generalize. The data set is balanced since every one of the classes occurs approximately equally in the training set and in the validation set. Rescaling and data augmentation are two basic mathematical procedures that were used in preprocessing and image preparation for deep learning models. Rescaling: By reevaluating all of the pixel values to fall inside [0, 1], the neural network can train more [10] efficiently. An image is provided with I pixel values I_{ij} thus far, where i and j represent the spatial resolution of the pixel. The following is a mathematical description of the rescaling process:

$$I'_{ij} = \frac{I_{ij}}{255}, \tag{1}$$

where I'_{ij} represents the re-evaluated pixel value.

Data Augmentation: To enrich the training set with diversity and decrease the likelihood of overfitting, a few data augmentation strategies were utilized. Each such operation can be mathematically expressed based on the following methods.

Rotation: Each image is subjected to a rotation matrix $\mathcal{R}(\theta)$ [11], where θ can be any real number within [−20∘,20]. For each image I, one has to use the following formula in order to determine its coordinate transformation in (x, y):

$$\left[\frac{x'}{y'}\right] = R(\theta).\left[\frac{x}{y}\right], \tag{2}$$

where the rotation matrix $R(\theta)$ is defined as:

$$R(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}, \tag{3}$$

Shift in Height and Width: This translation process is unpredictable. It applies a shift of δ_x and δ_y to the image. The formular below represents this method

$$I'(x, y) = I(x - \delta_x, y - \delta_y)\ , \tag{4}$$

where δ_x , δ_y are randomly selected within [-20%, 20%] of the image dimensions.

Zooming and shearing apply affine transformations, causing the image to be skewed by an angle θ (shear) or scaled by a factor $\propto$ $(zoom)$

$$I'(x, y) = \propto \cdot\ I(x, y)\ \text{(zoom)}, \tag{5}$$

Since the model has seen numerous versions of the images during training, these changes help the model to generalize more effectively.

Convolutional Neural Network (CNN) Architecture

To achieve hierarchical features from images, convolutional neural networks (CNNs) having many layers are utilized [12]. Convolution and pooling are the two fundamental operations of a convolutional nueral network. Convolution layer generates a feature map F through a filter W applied on image I. The subsequent operation convolves the image with the filter.

$$F(x,y) = (I * W)(x,y) = \sum_{i=0}^{k-1} \sum_{j=0}^{k-1} I(x+i, y+j) W(i,j), \tag{6}$$

where k is the size of the filter, and $W(i,j)$ represents the filter values. By lowering the network's parameter count, max pooling operation technique lowers its computational cost by choosing the highest value in each patch, max pooling is utilized to minimize the image's spatial dimensions .

$$\mathrm{P}(x,y) = \max_{i,j \in patch} I(x+i, y+j) , \tag{7}$$

Once the convolutional and pooling layers have finished extracting information, the network is fully connected. The softmax layer, using the softmax function, creates class probabilities [12] and provides the output].

$$P(y_j|X) = \frac{e^{z_j}}{\sum_k e^{z_k}}, \tag{8}$$

where z_j is the output of the final fully connected layer corresponding to class j, and k runs over all possible classes. The class with the highest probability is chosen as the model's prediction.

Model Compilation and Training

The categorical cross-entropy loss function, measuring the difference between the target class probability and the predicted class distribution, is used for multi-class classification. The loss function L for a single sample is given by:

$$L = -\sum_{i=1}^{N} y_i \log(\hat{y}_i), \tag{9}$$

where $\hat{y}_i$ is the estimated probability of class I, N is the number of classes, and y_i is the actual label (1 for the true class, 0 for others). The Adam optimizer incorporates the best of both the momentum and RMSProp optimizers and reduces the category cross-entropy loss. For the weights θ, the update rule is:

$$\theta_{t+1} = \theta_t - \eta \cdot \frac{m_t}{\sqrt{v_t + \epsilon}}, \tag{10}$$

where m_t and v_t are estimates of the first and second moments of the gradients, respectively, and η is the learning rate. A small constant called $\in$ prevents dividing the gradients by zero.For evaluating the classification results, the confusion matrix is utilized. Overall performance measurement is provided by the matrix which compares the predicted labels with the actual labels. Here is how it works:

$$C = \begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix}, \tag{11}$$

where FP is False Positives, FN is False Negatives, TN is True Negatives, TP is True Positives, and FN is False Negatives, which are wrongly predicted negative samples, rightly predicted negative samples, and correctly predicted positive samples, respectively.

Results

The performance of complicated algorithms was highly variable in terms of processing time, F1 score, recall, accuracy, and precision. These figures for the principal algorithms we tested in this study are presented in Table 1.

The categorization model is 93% correct and does well in each of the three classes. As can be observed, with concurrent F1-scores of 0.97 and 0.96, the weasels and badgers are predicted with a very high rate of accuracy, and that the Mantis class is close behind at 0.89. For all three animal classes, the stable and strong performance of the model is also confirmed through the overall average precision (0.95) and recall (0.93). The F1-score, recall, and precision for each of the aforementioned classes are as follows:

Classification report

Table № 1

Class	Precision	Recall	F1-Score	Support
Badger	0.94	1.00	0.97	5
Mantis	1.00	0.80	0.89	5
Weasel	0.92	1.00	0.96	5
Overall				
Accuracy		0.93		15
Macro Avg	0.95	0.93	0.94	15
Weighted Avg	0.95	0.93	0.94	15

Precision: Measures the accuracy of positive predictions.

$$precision = \frac{TP}{TP+FP}, \quad (12)$$

Recall: Measures the ability of the model to capture positive instances.

$$Recall = \frac{TP}{TP+FN}, \quad (13)$$

F1-Score: Harmonic mean of precision and recall, providing a balance between them.

$$F1 - Score = 2 \cdot \frac{Precision \cdot Recall}{Precision+Recall}, \quad (14)$$

When it comes to managing the trade-off between false positives and false negatives, these indicators offer a full view of the model's performance.

Fig. 1. - Predictions for mantis, honey badger and weasel Classes

Discussion

The experiment proved that Convolutional Neural Networks (CNNs) perform best to label wildlife species at an average classification accuracy of 93% from mantises, honey badgers, and weasels. High model precision and recall are a sign that the model had indeed harvested discriminative features regardless of intra-species variability, occlusion, and lighting variance. Model performance could mostly be credited to the augmentation data, improving generalization with the simulation of realistic variation of images in the camera trap ones. Overfitting was avoided and unseen-data performance was maintained consistent with methods including rotation, scaling, and shifting. However, mantises have a comparatively lower F1-score (0.89) when compared to those of weasels (0.96) and honey badgers (0.97) and could perhaps be due to the fact that they look much like the other arthropods beyond this dataset or that there are lesser morphological differences between them. Label-fine-grained convolutional nueral network structures or attention are worth trying in future work in refining classification among very visually similar species. Further preprocessing of labeling, such as histogram equalization or background subtraction, would also improve accuracy, according to the performance of the labeling model across varying environmental conditions. Further testing of the model's robustness for conservation applications would involve evaluating its performance on actual, uncurated camera trap data.

Conclusion

The project effectively showed that Convoulutional nueral networks could classify images of wildlife animals to monitor the environment. In order to avoid species bias, it emphasized training data variation. Transfer learning was shown to enhance current models of animal tracking and how neural network architecture modification would be an influential impact on classification performance. In short, the establishment of effective conservation policies through data collection and monitoring in natural environments will be greatly facilitated by advanced artificial intelligence methods.

References

1. Alsaadi F., Abbadi S. Wildlife conservation through the application of deep learning: A survey of techniques and models. International Journal of Computer Science and Network Security. 2020. Vol. 20. №5. pp. 121-130.

2. He K., Zhang X., Ren S., Sun J. Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015. pp. 770-778. URL: doi.org/10.1109/CVPR.2016.90.

3. Islam S. M., Khan S., Rahman M. Deep learning techniques for image-based species identification: A survey. Journal of Computer Vision and Image Processing. 2020. Vol. 10. №4. pp. 71-81.

4. Krizhevsky A., Sutskever I., Hinton G. E. ImageNet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems. 2012. Vol. 25. pp. 1097-1105.

5. Li Y., Gu X., Yang J. Deep learning in visual recognition tasks: A comprehensive review. IEEE Transactions on Neural Networks. 2006. Vol. 17. №1. Pp. 100-118. URL: doi.org/10.1109/TNN.2005.852122.

6. LeCun Y., Bengio Y., Hinton G. Deep learning. Nature. 1998. Vol.521. № 7553. pp. 436-444. URL: doi.org/10.1038/nature14539.

7. Scott M., Hooft B., McDonald K. Automated species identification in conservation with deep learning models: Trends and applications. Ecology and Evolution. 2024. Vol. 14. №1. pp. 123-135. URL: doi.org/10.1002/ece3.9956.

8. Simonyan K., Zisserman A. Very deep convolutional networks for large-scale image recognition. Proceedings of the International Conference on Learning Representations (ICLR). 2014. pp. 1-14. URL: arxiv.org/abs/1409.1556.

9. Sultana F., Rehman R., Ahmad M. Deep learning for wildlife monitoring: Species detection and classification from camera trap images. Journal of Animal Conservation. 2018. Vol. 21. №2. pp. 102-112.

10. Ioffe S., Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. Proceedings of the 32nd International Conference on Machine Learning. 2015. pp. 448-456.

11. Badrinarayanan V., Kendall A., Cipolla R. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2017. Vol. 39. №12. pp. 2481-2495. URL: doi.org/10.1109/TPAMI.2016.2644615.

12. Zhang X., Zhou X., Chen Y. ResNet-50 based deep convolutional neural network for real-time object detection. Proceedings of the IEEE International Conference on Computer Vision Workshops. 2018. pp. 190-197. URL: doi.org/10.1109/ICCVW.2017.29.

Дата поступления: 8.04.2025

Дата публикации: 25.05.2025