IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

ANIMAL SPECIES DETECTION USING CONVOLUTIONAL NEURAL NETWORK

[1]Dr. Leena Giri G, [2]Geetha N R, [3]Kausthub Babu, [4] Sushma G C, [5]K Dheeraj

2,3,4,&5 Students, Department of Computer Science and Engineering, Dr Ambedkar Institute of Technology, Bengaluru, Karnataka, India

1 Associate Professor, Department of Computer Science and Engineering, Dr Ambedkar Institute of Technology, Bengaluru, Karnataka, India

Abstract:The project, "Animal Species Detection using Convolutional Neural Network," aims to create a sophisticated system for identifying and classifying animal species from both static images and live video streams. Utilizing Convolutional Neural Networks (CNN) and the YOLO v8 object detection algorithm, the system promises accurate, real-time animal detection and recognition. Inputs from various sources will yield outputs with bounding boxes, species names, and corresponding audio tracks. Developed with the Streamlit framework for an interactive front end, the project will leverage the Roboflow platform for data set curation, model training, and deployment. Notifications via Telegram will provide real-time detection updates, and the front end will offer detailed information on each detected species. This project is poised to significantly advance animal conservation, wildlife monitoring, and environmental research by offering an efficient and accurate tool for animal species detection and identification.

***Keywords*–CNN, YOLO, Roboflow.**

I. INTRODUCTION

In recent years, advancements in deep learning and computer vision have enabled significant progress in animal species detection. This project leverages convolutional neural networks (CNNs) to accurately identify animal species from static images, video feeds, and live camera footage. The system outputs the detected animal's name along with the precision rate, an audio clip of the animal's name and sound, detailed information about the species, and a Telegram notification with the animal's name and the detection time and date.

The project's main goal is to develop a robust and efficient model capable of real-time recognition of a diverse range of animal species. This involves training a CNN model on an extensive dataset of animal images to ensure high accuracy and reliability. Additionally, the project integrates an audio component to enhance the user experience by providing the corresponding sound of the identified animal. This feature offers an immersive and educational tool for wildlife enthusiasts, researchers, and educators, and it is particularly beneficial for visually impaired individuals by allowing them to identify animals through sound, thereby enhancing their interaction with nature.

The proposed system includes several key components: model training and validation, real-time detection from images, videos, and live feeds, and audio output integration. By utilizing YOLO (You Only Look Once) in machine learning and computer vision, this project not only demonstrates the practical applications of CNNs but also contributes to wildlife conservation through improved species recognition capabilities.

Overall, this project aims to develop a user-friendly application that facilitates animal species identification, providing both visual and auditory information to enhance understanding and appreciation of wildlife. It also supports visually impaired individuals in learning about animals, promoting inclusivity and accessibility in wildlife education.

1.1 Existing System

In the existing landscape of animal species detection systems, there are notable limitations that our research seeks to address. Currently, most systems lack the capability for live feed input, making real-time monitoring challenging. Moreover, there is a notable absence of alert mechanisms, such as Telegram notifications, which are crucial for timely responses to wildlife sightings. Additionally, visually impaired users are underserved, as existing systems do not provide auditory feedback to enhance accessibility.

Regarding the architectural frameworks commonly employed in animal species detection, VGG models like VGG16 and VGG19 have demonstrated effectiveness, particularly in scenarios where datasets are large and species are well-represented. Their depth, characterized by multiple convolutional layers, enables intricate feature extraction, contributing to accurate species classification.

Similarly, ResNet models such as ResNet-50 and ResNet-101 have made significant strides in overcoming challenges like the vanishing gradient problem. By introducing skip connections, these architectures facilitate learning from both shallow and deep layers, enhancing the model's ability to classify animals accurately.

In addition to VGG and ResNet models, Faster R-CNN and Mask R-CNN have emerged as powerful tools for animal detection tasks. These models offer capabilities beyond classification, enabling localization and segmentation of individual animals within images. Such fine-grained analysis is invaluable for tasks like population estimation and tracking specific individuals.

In real-world applications, CNN-based systems play pivotal roles across diverse domains. In wildlife conservation efforts, these systems are deployed in camera trap networks to automate animal identification and monitor endangered species populations. By providing continuous data streams, they inform conservation strategies and aid in preserving biodiversity.

In agricultural settings, CNN models find utility in identifying livestock species, monitoring their health, and automating disease detection and animal counting tasks. This automation streamlines farm management practices, ensuring optimal animal welfare and productivity.

Moreover, in ecological research, automated species detection facilitates the analysis of large datasets of camera trap images. This enables researchers to gain insights into animal behavior, habitat utilization patterns, and interspecies interactions, fostering a deeper understanding of ecological dynamics.

As we delve into our research, we aim to bridge existing gaps in animal species detection systems, particularly in terms of real-time monitoring, alert mechanisms, and accessibility for visually impaired users. By leveraging the strengths of established architectural frameworks and incorporating novel features, our endeavor seeks to advance the field and contribute to its practical applications in conservation, agriculture, and ecological research.

1.2 Proposed System

The existing system lacks several critical features that our proposed project aims to address comprehensively. Key limitations include the absence of live feed input options and the lack of Telegram alerts for real-time notifications when wild animals are detected. Additionally, there is no provision for auditory feedback to assist visually impaired users in the existing system.

To overcome these deficiencies, our proposed project, titled "Animal Species Detection Using Convolutional Neural Network," integrates cutting-edge technologies such as Convolutional Neural Networks (CNN) and the YOLO v8 object detection algorithm. By leveraging these advanced tools, our system endeavors to accurately identify and classify a wide array of animal species in real-time, enhancing usability and interactivity.

II. LITERATURE SURVEY

One-dimensional (1D) CNNs were effectively used for bearing problem diagnosis and detection because of their excellent efficacy in vibration signal processing [5, 6]. A 1D CNN bearing defect diagnostic model working on time domain signals was studied by Zhang et al. [7]. Abdeljaber et al. [8] fed raw signals of time into a 1D CNN and applied it to real-time structural damage detection in bleachers. Su et al. [9] proposed ResNet to directly process the raw signal of time domain for fault diagnosis of a high-speed train bogie. Wang et al. [10] proposed a multi-attention one-dimensional convolutional neural network (MA1DCNN) to diagnose wheelset bearing faults. Fast Fourier transform (FFT) was used by Zhao et al. [11] to convert 1D time domain signals into frequency domain images, which were then input into models for defect diagnosis such as BiLSTM, LeNet, AlexNet, ResNet18, and others. The discrete Fourier transform (DFT) was utilized by Janssens et al. [12] to convert signals from the time domain into the frequency domain, which was then fed into a CNN for problem identification.

Even with its use in fault diagnosis, the 1D CNN model still has the following shortcomings.
(1) Given that CNN was first created to address the learning challenges associated with two-dimensional (2D) images, its benefits cannot be completely realized when 1D signals are used as the input.
(2) The valuable fault feature information is lost when the time domain signal is processed directly using 1D CNN. The precise fault characteristics are outside the scope of the 1D CNN model.

Two-dimensional images are far more effective and efficient in diagnosing faults since they frequently carry a lot of fault information. Deep learning is capable of automatically extracting features from the pictures that describe the kind of deep-level bearing faults. To identify bearing fault states, a 2D shape conversion of the 1D vibration data is followed by image classificatio Deep learning is capable of automatically extracting features from the pictures that describe the kind of deep-level bearing faults. n[13]. In order to provide the model with statistical variables derived from vibration data, Bhadane et al. [14] constructed a 2D CNN for the purpose of classifying bearing defects. Hoang et al. [15] converted the original time domain signals into 2D gray-scale images based on the time series as input to CNN for fault diagnosis. Wang et al. [16] used FFT to segment the 1D raw signals, turn them into frequency domain signals, and then create 2D images from the frequency domain signals. Ultimately, the enhanced LeNet-5 model, which was trained on the 2D images, was able to quickly assess the bearing's reliability and project how long it would last. In order to diagnose faults, Wen et al. [17] suggested converting the original time domain signals into 2D grayscale images, which would then be entered into an upgraded LeNet-5 model.
In contrast to the 2D transformations found in the aforementioned literature, the STFT can be used to transform 1D signals and produce 2D time-frequency pictures. In addition to having more fault information, the time-frequency pictures also have information in the frequency and time domains. Compared with time series signals, time–frequency images are much easier to extract information in noisy environments, increasing the overall efficiency. Time–frequency domain inputs are notably superior to time domain inputs, as has been shown in the study of the defect diagnostics. The widespread usage of STFT in rotating machinery defect diagnosis highlights the technology's significance in real-world applications. Therefore, the time–frequency images are fed into the proposed CNN model for fault diagnosis, leading to better results achieved with significantly fewer learnable parameters. We used the STFT to generate 2D images from 1D signals, followed by fault diagnosis using a CNN. And furthermore, we construct a new network for bearing fault diagnosis based onSTFT and CNN. The application of this combined approach shows promising results in real-world fault detection scenarios.

III. THEORETICAL FUNDAMENTALS

3.1 Convolutional Neural Network

Traditional CNN is used in computer vision and is very good at extracting feature information from images. A CNN is a deep learning technique that is particularly well-suited for the exammination of visual data. The layers that make up a CNN are often categorized into 3: Convolutional Layers, Pooling Layers, and Fully Connected Layers. The CNN's complexity rises as data moves through these layers, enabling it to detect progressively more abstract characteristics and greater areas of a picture. Figure 1 represents the general CNN structure.

The convolution function is given as follows:

$$H_j^{l+1} = \sum_{i \subset x_j} H_j^l * w_{ij}^{l+1} + b_j^{l+1} \quad (1)$$

where H_j^{l+1}denotes the jth feature map of the neuron at layer *l+1*, * denotes the convolution function, w_{ij}^{l+1}denotes the convolution kernel connecting the jth feature map of the neuron at layer l+1 and the ith feature map of the neuron at layer l, b_j^{l+1}denotes the bias, and $x_j$denotes the image of the input CNN.

There is a linear process in the convolution layer. A nonlinear activation function is introduced to the model to improve its classification performance. The Sigmoid function, Tanh function and ReLU function, which are frequently used are defined as follows:

$$f_{sigmoid}(x) = \frac{1}{1+e^{-x}} \quad (2)$$

$$f_{Tanh}(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \quad (3)$$

$$f_{ReLU}(x) = \begin{cases} x, x \geq 0 \\ 0, x < 0 \end{cases} = \max(0, x) \quad (4)$$

Pooling layers take part in reducing the network parameters, and it is given as follows:

$$H_j^{l+1} = \mathrm{f}(\beta_j^{l+1}\, down\,(H_j^l) + b_j^{l+1}) \quad (5)$$

where down(.) denotes a subsampling function, β denotes the multiplicative bias.

The two most popular pooling techniques are average and maximal pooling. While average pooling averages the window values and outputs them, maximum pooling produces the window's maximum value. We employ the greatest pooling layer in this study. Figure 2 displays a schematic of the Maximum pooling technique. In the example, the step size is two and the convolution kernel is two by two in size.

The feature data that was previously extracted is classified using the fully connected layer; this process is represented as follows:

$$y^k = f(w^k x^{k-1} + b^k) \quad (6)$$

where *k* is the *k-th* layer network, $x^{k-1}$is the input of the *(k-1)-th* fully connected layer, the $y^k$is the output of the *k-th* fully connected layer, $w^k$is the weight coefficient, $b^k$is the bias, and *f* is the classification function.

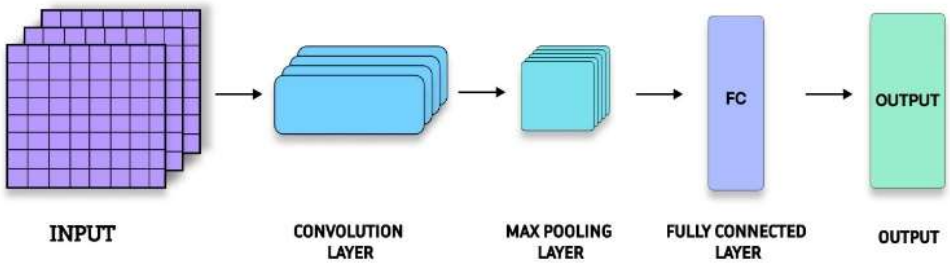
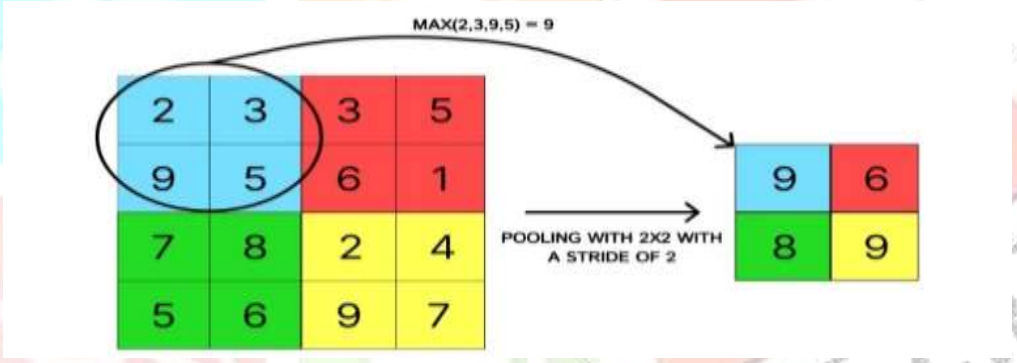


Figure 1 General CNN Structire

Figure 2 Maximum Pooling Method



IV. METHODOLOGY

4.1 Procedures of the proposed method

The journey of our system commences with the initiation of the process, where users engage by uploading their desired media, whether it be images, videos, or even live feeds. This media is then seamlessly processed through the YOLO model, renowned for its real-time object detection capabilities. Within this critical stage, the YOLO model diligently scrutinizes the content, discerning and categorizing objects present. Upon completion, the system transitions to a pivotal juncture, evaluating whether any species, or animals, have been successfully identified. In the event of a positive detection, the system springs into action, crafting a tailored alert. This alert not only includes an audio snippet associated with the detected species but also dispatches a prompt notification via the Telegram messaging platform, ensuring real-time awareness. Conversely, should the system find no trace of any species, it gracefully concludes its task, signifying the absence of relevant species within the uploaded media. Finally, as the process wraps up, it exits the stage, having fulfilled its duty of handling any detected species or acknowledging their absence, thus drawing this captivating journey to a close.

4.2 Details of the CNN model

The suggested CNN, which has four fully connected layers (FC), two maximum pooling layers (MP), one flatten layer, and five convolutional layers (C), is depicted in Figure 3. The original signals are converted into images and fed into the proposed CNN model to classify the images. In this work, the suggested CNN

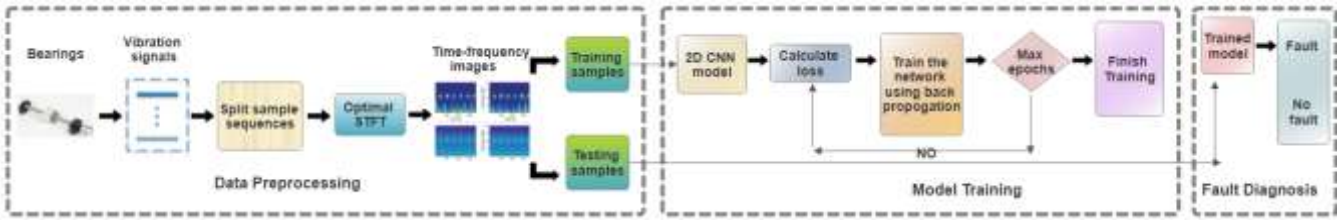


Figure 3 Flow Diagram of the proposed Model

model is used to complete the fault diagnosis task.

Table 1 displays the specific structural parameters for every layer in the CNN model. There are four components to the model. The first part consists of 32 convolutional kernels of size 5 ×5 followed by a 2 × 2 maximum pooling layers. The second part has a a two-layer stack of 32 convolutional kernels of size 3 × 3 followed by a 2 × 2 maximum pooling layer. The third part has a two-layer stack of 32 convolutional kernels of size 3 × 3 followed by a flatten layer. Maximum pooling is applied after the first convolutional layer, while stacking is applied after the remaining two convolutional layers. The fourth part is a four-layer full connection layer with input dimensions of 256, 1024, 128, and 2 respectively.

Additionally, the benefits of this model can be summed up as follows:

(1) A 5 × 5 convolution kernel is used in the first convolution layer to extract information from the time-frequency image's greater neighborhood range and produce superior features. In order to expand the receptive field, collect more data, and provide the network's later layers more information, huge convolutional kernels are employed in the first layer. Additionally, these kernels are more effective at suppressing high-frequency noise. [18]

(2) Use two 3 × 3 convolution kernels instead of one 5 × 5 convolution kernel. Gaining more nonlinear expression capabilities requires two activation functions for each of the two 3 × 3 convolution layers. Less parameters can lower the computational effort while two layered convolutional layers can enhance the feature extraction capability.

Table 1 Structural Parameters of the CNN Model

Layers	Parameters
C1	Conv2D(5 X 5 X 32)
MP1	MaxPool2D(2 X 2)
C2	Conv2D(3 X 3 X 32)
C3	Conv2D(3 X 3 X 32)
MP2	MaxPool2D(2 X 2)
C4	Conv2D(3 X 3 X 32)
C5	Conv2D(3 X 3 X 32)
Flatten Layer	
FC1	Input Dimensions = 256
FC2	Input Dimensions = 1024
FC3	Input Dimensions = 128
FC4	Input Dimensions = 2

V. CONCLUSION AND FUTURE WORK

The incorporation of Convolutional Neural Networks (CNNs) in ecological research for animal species detection marks a significant leap forward in wildlife monitoring and conservation endeavors. By harnessing the processing capabilities of live images and video feeds, these neural networks demonstrate the ability to accurately identify various animal species. Beyond merely labeling the detected species, the system elevates user engagement and educational value by providing not only the name of the animal but also an accompanying audio track, potentially featuring the animal's unique sounds. Additionally, the system offers real-time notifications via Telegram, furnishing users with detailed information such as the animal's name, date, time, and additional insights gleaned from the model's analysis.

The efficacy of such a system hinges on two key factors: the precision of the CNN in classifying a diverse array of species and the richness of the dataset used for its training. This robust classification capability holds immense promise for biodiversity assessments and the exploration of ecological dynamics.

In essence, CNN-based animal species detection presents a promising avenue for ecological research, furnishing a scalable and efficient means of wildlife monitoring that contributes to the preservation of biodiversity.

Looking ahead, there exist numerous avenues for further enhancement, encompassing aspects of performance, functionality, user experience, and scalability. These include refining the YOLO model through training on custom datasets, fortifying the system's security with user authentication features, and enhancing the Streamlit interface for seamless interaction. Deployment on cloud platforms and integration with IoT devices offer avenues for broader accessibility and automated actions, while advanced logging and monitoring mechanisms ensure robust performance and error handling. Furthermore, multilingual support and customizable detection parameters cater to diverse user needs, making the system more inclusive and adaptable.

Through the implementation of these future enhancements, the project stands poised to evolve into a more comprehensive, versatile, and user-centric tool, capable of addressing a myriad of ecological challenges and providing an enriching experience for users across various domains.

REFERENCES

[1] Yo-Ping Huang 1, 2, (Senior Member, IEEE), and Haobijam Basanta, "*Bird Image Retrieval and Recognition Using a Deep Learning Platform 89*" IEEE Access PP(99):1-1, 2019.

[2] Zhangkai Ni, Lin Ma, Huanqiang Zeng, Jing Chen, Canhui Cai, and Kai-Kuang Ma, "*ESIM: Edge similarity for screen content image quality assessment*" IEEE Transactions on Image Processing, vol. 26, no. 10, pp. 4818–4831, 2017.

[3] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. Slic "*Superpixels compared to state-of-the-art superpixel methods*." Trans. on Pattern Analysis and Machine Intelligence, 34(11):2271–2282, 2017.

[4] Xiaogang Wang, The Chinese University of Hong Kong, Foundations and Trends R, in Signal Processing, Vol. 8, No. 4 (2014) 217–382 c 2016 X. Wang DOI, "*Deep Learning in Animals Recognition, Detection, and Segmentation*".

[5] X. Chen, S. Xiang, C.-L. Liu, and C.-H. Pan, "*Vehicle detection in satellite images by hybrid deep convolutional neural networks*," IEEE Geoscience and remote sensing letters, vol. 11, pp. 1797-1801, 2014.

[6] Viola, Paul, and Michael Jones. "*Rapid object detection using a boosted cascade of simple features*." Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on. Vol. 1. IEEE, 2001.

[7] Angelova, Anelia, and Shenghuo Zhu. "*Efficient object detection and segmentation for fine-grained recognition.*" Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2013.

[8] Wohlhart, P., & Lepetit, V. "*Learning descriptors for Animals recognition and 3D pose estimation*." IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp. 3109-3118, 2015.

[9] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed,D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. "*Going deeper with convolutions*". IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp. 1–9, 2015.