Wild Animal Classification and Alert System

Parvathi Salera J[*1], Mrs. Namitha M V [*2], Radhika J G[*3], Priyanka S R[*4], Ruchitha A R[*5]

[*1,3,4,5]Research Scholar, Dept. of CS&E , JNNCE, Shimoga, India.

[*2]Assistant Prof ., Dept. of CS&E , JNNCE, Shimoga, India.

ABSTRACT

In recent years, the attack of wild animals on humans is increasing at a faster pace especially in the roads adjacent to reserved forests. Although there are a lot of systems that detect and classify wild animals, none takes any further step to rescue people in danger by taking required steps. Hence, we have proposed a model based on CNN in conjunction with YOLO(You Only Look Once) to successfully detect an animal, classify it and send appropriate message to the concerned authorities. A total of 15 animal classes is used to train the model which yielded an accuracy of 98%.

Keywords: Wild animals, CNN, YOLO

I. INTRODUCTION

Animals in the wild are under serious threat due to continual hunting and poaching. Also due to the loss of wild habitat, animals are moving out of forests and into the cities and hence posing a danger to human lives. Hence it is of paramount importance to monitor the wild animal's movement and take necessary action especially in the forest highways or forest routes. The proposed project mainly involves capturing the animals in live, detecting them, classifying them to identify if they are wild or no and then sending an alert message to the local guards or the respected authorities. There are several models which are based on CNNs which were proposed previously to detect the animals in real time but faces a problem of unclear images captured during live streaming which may affect the overall performance of the algorithm. The proposed model makes use of Convolution Neural Network (CNN) to detect the animals supplemented by using a real time object detection system called You Only Look Once (YOLO).

II. LITERATURE REVIEW

Durmus Ozdemir et al., [1] When it comes to insects, there are approximately 21 different criteria to determine order level, from the number of wings, body shape, number of feet, head shape and 32 insect orders have been identified in nature. Faster R-CNN, Inception V3, SSD Mobile NET, YoloV4 deep learning methods are used for the comparative analysis to classify and detect insects. In order to group insects, the insect images in the training dataset were labeled and images collected were subjected to preprocessing. The user can select the insect image by taking a photo from the device's image gallery or the camera. Then, the comparison was made with the deep learning file embedded in the program, and the classification process was performed. After the classification, insect team name and prediction percentage were shown to the user. When the test results of the compared deep learning models were evaluated in general, the insects were successfully detected and classified in order level. Faster R-CNN gave the best results with an accuracy of 81%.

Nidhal K. El Abbadi et al., [2] used a Deep Convolution Neural Network to detect and classify animals. A ConvNet architecture has a number of layers which transforms the image volume into an output volume. In the network, there are different kinds of Layers like Convolution, ReLU, Pool and Fully Connected (FC). The 2D array from the last layer was converted into 1D vector by flattening process. Then Flattened matrix from the pooling layer is fed as an input to the Fully Connected Layer to classify the image. Each neuron in the network was fully connected to all the neurons in the previous layer, but neurons in a single layer function was completely independent and didn't share any connection. The final FC layer is named (output layer) and it represents the class scores in classification settings. After several tests have been done in the proposed model, the accuracy of detection and classification of animals reached to 97.5%.

Sonain Jamil et al., [3] The dataset contained four classes, Snow Leopard, Marco polo sheep, Himalayan bear, and other animals. Many D-CNN like AlexNet, ResNet-50, VGG-19, and inception v3 were used to extract features. A confusion matrix was the parameter to check performance of classification problem. 100 different images of each class were used. Dataset had 70% of the data used for training the model, while 30% was used for testing. The experimental results verify that inception v3 integrated with kNN outperforms other D-CNNs. It also has more accuracy of 98.3% with a classification error of 2%, which is quite negligible.

Tibor TRNOVSZKY et al., [4] proposed a CNN to classify the animal input images. Firstly, preprocessing was done by converting the raw data into sensible format, then feature extraction was carried out by algorithms such as PCA, LDA, LBPH and finally classification was done by SVM which was used for comparison. The created animal database includes five classes of animals such as fox, wolf, bear, hog and deer. Each animal had 100 different images. The best recognition rate with an accuracy of 98% using proposed CNN for the performed experiments with the ratio of – 90 % of training images and 10 % test images was achieved. On the other hand, the worst recognition rate with an accuracy of 78 % for the experiments which is – 40 % of training images and 60 % of the test images was obtained.

Takuro Matsui et al., [5] There is a need for post processing tool that removes the fences from images. Conventional methods are inaccurate. Therefore, novel methods of DCNN with classical domain knowledge in image processing was combined. Training requires both fence and corresponding non fence ground truth images. Hence synthesis of natural fences from real images was carried out. Spatial filter processing-improves performance of CNN. Fences may have irregular size, texture and colors. In automatic De-Fencing algorithm, it is assumed that most of the fences had near regular shape. The proposed method could automatically detect fences and recover the missing regions without any human intervention. Fence detection is a regression problem, it can also deal with irregular fences. In the image completion phase, synthesis of natural fence images takes place in order to train the system to work with wide range of datasets. Experimental results demonstrate that the method is effective for a broad range of fence images.

	Paper name	Author name	Algorithms used	Methodology	Limitations
1	Comparison of Deep Learning Techniques for Classification of the Insects in Order Level with Mobile Software Application	Durmus Ozdemir, Musa Selman Kunduraci	Faster R-CNN, Inception V3, SSD Mobile NET, Yolo V4 Deep learning methods for the comparative analysis for classifying and detection of insects	A mobile-based decision support software with a deep learning model to classify and detect insects at the order level.	The algorithm cannot classify if the insect photograph does not accurately capture its characteristic features and it takes longer time for classification.
2	An Automated Vertebrate Animals Classification Using Deep Convolution Neural Networks	Nidhal K. El Abbadi, Elham Mohammed Thabit A. ALSAADI	Deep Convolution Neural Network to detect and classify the animals which belong to vertebrate classes in digital images	Vertebrate animals have been used as the basis for identifying and categorising them into one of five groups. Mammals, Amphibians, Reptiles, Birds, Fish.	The image size that performs better is 50X50 and increasing number of epochs more than 100 will not have significant effect on accuracy, but it leads to increase in training time.
3	Deep Learning and Computer Vision - based a Novel Framework for Himalayan Bear, Marco Polo Sheep and Snow Leopard Detection	Sonain Jamil, Fawad, Muhammad Sohail Abbas	Deep Convolution Neural Network(DCNN), and k Nearest Neighbors (KNN) to detect the animals	Extraction of features with a convolutional layer. The classification with fully-connected (FC) and softmax layers using classifier algorithm	Novel Framework is only for Himalayan Bear, Marco Polo Sheep and Snow Leopard Detection only.
4	Animal Recognition System Based on Convolutional Neural Network	Tibor TRNOVSZKY, Patrik KAMENCAY, Richard ORJESEK, Miroslav BENCO, Peter SYKORA	Convolutional Neural Network, Principal Component Analysis, Linear Discriminant Analysis, Local Binary Patterns Histograms and Support Vector Machine	CNN is used to classify animals. This method is compared with well-known image recognition methods. In all experiments, all animal images were aligned and normalized based on the positions of animal eyes.	When the image is divided into more windows the classification results should be better. On the other hand, the computation complexity will increase.
5	Single-Image Fence Removal Using Deep Convolutional Neural Network	Takuro Matsui, D Masaaki Ikehara	Deep Convolution Neural Network, automatic De-Fencing algorithm, Classification and Regression algorithm	Fence Detection using Color estimation repeated use of elements and Fence Removal methods using Defencing methods such as video based and image based .	Image de-fencing is challenging because real-world fences have various type of shapes, textures and colors.

III. SYSTEM DESIGN

The flow of how the system works is depicted in the following figure:

1. The user will input the image of animal the camera trap would have captured.

2. Image processing will take place using Keras library of Deep Learning.

3. TensorFlow is used at the backend to extract the features.

4. Based on feature extraction a machine learning model will get generated and trained.

5. Then the test images have to be provided to compare the train and test data

6. Animals will get classified and detected based on trained model.

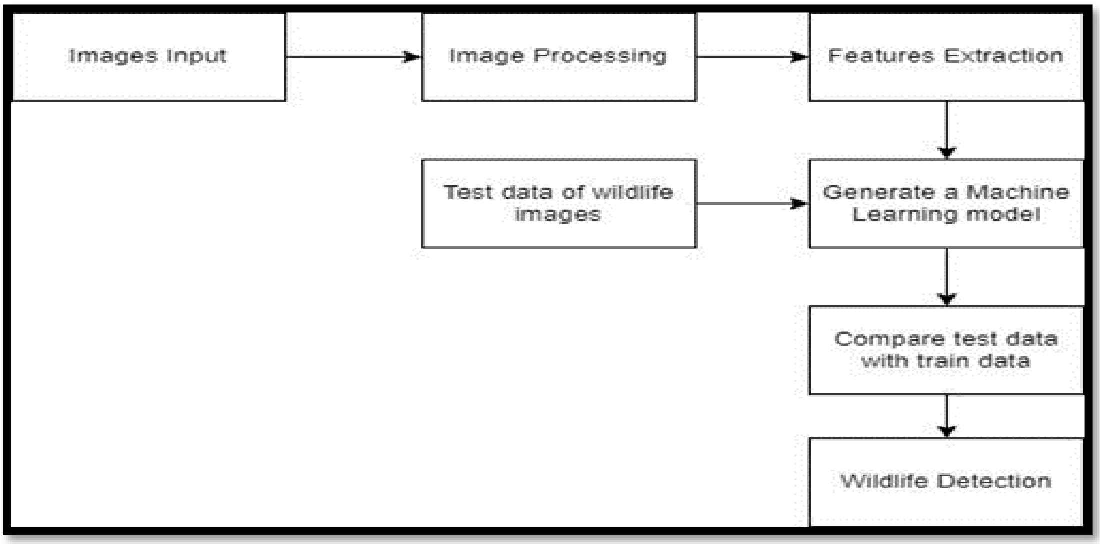


Fig: Flow Chart of proposed model

IV. DATASET

The dataset for the proposed project is taken from Kaggle. The images of one species of animal are stored in a folder and many such folders are used. The dataset used for the project includes 15 classes of animals – giraffe, tiger, bear, lion, elephant, deer, wolf, bull, monkey, leopard, rhinoceros, hippo, cattle, buffaloe and goat. The various ways in which an animal can be captured such as standing, sleeping, hunting, being hunted, climbing a tree, eating food, being in a group etc are used.

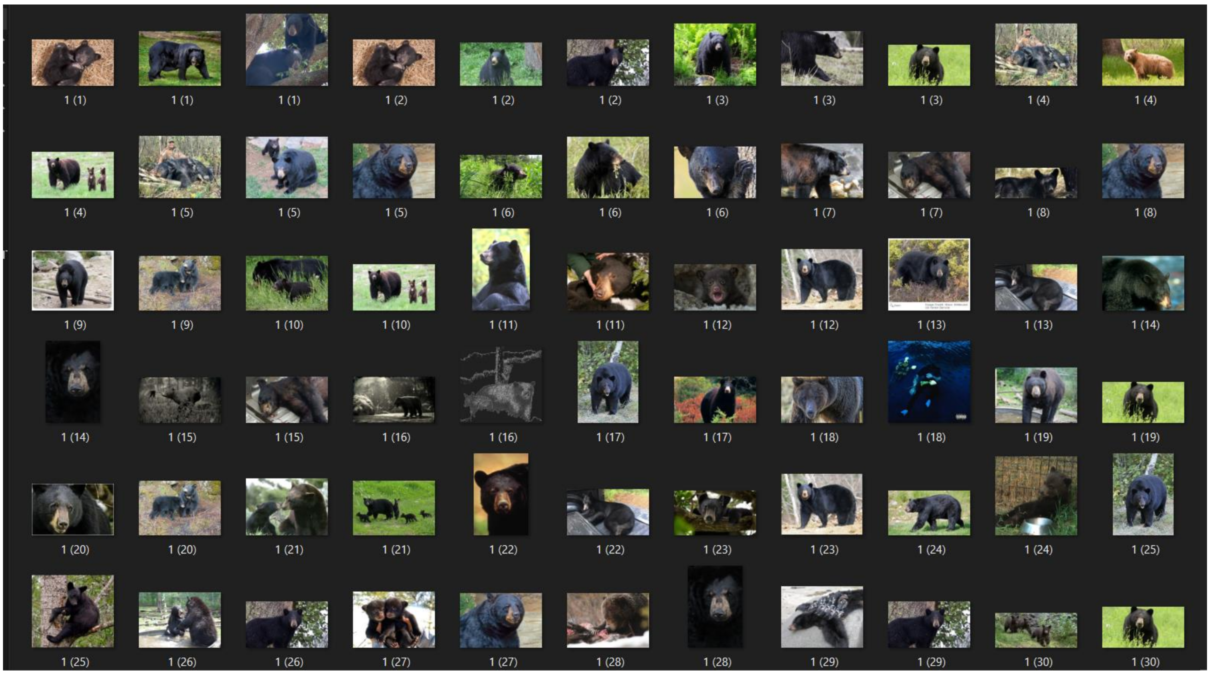


Fig: Dataset

V. IMAGE PREPROCESSING

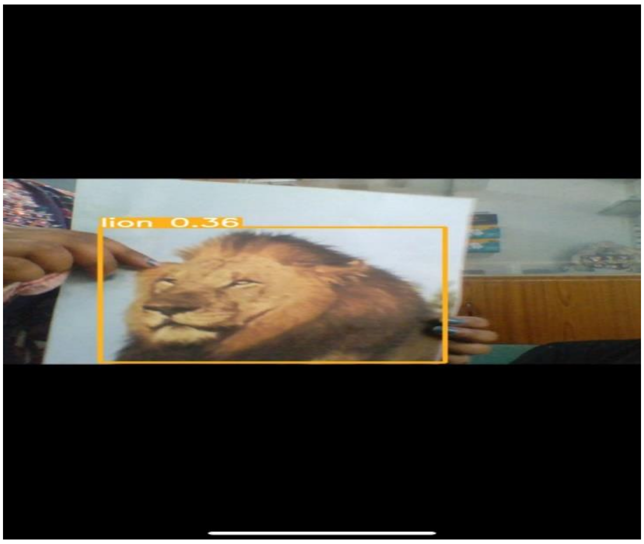
Image pre-processing is a crucial step in wild animal classification system as it helps preparing the images for analysis and improves the accuracy of the classification results. In a wild animal classification system, image pre-processing involves several steps as explained below. As it is known, when reading a colour image file, OpenCV reads it as BGR. The only difference between the RGB and BGR image is that, the red, green and blue channels are read in the reverse order. Once a coloured image is read, it is converted into a grayscale image as it reduces the algorithmic complexities related to computation. In contrary to 3 channels in RGB images, grayscale images consist of only a single channel. The image is then resized as models can be easily trained on smaller sized images and aids in easier computation.

VI. METHODOLOGY

The project implementation makes use of several python libraries used for the applications of machine learning and computer vision. The various libraries used are os, cv2, numpy, tensorflow and keras. A dictionary is made use of to map each of the name of the animals to the corresponding numerical value. A mapper is used to map each of the animal name to the corresponding class label. The next step is to train the model. The size of the images used for training the model is set to 225*225 and then the training images and corresponding class labels are loaded from the specified path. The loaded images from the directory and the labels will be stored in a list and a full path to the current subdirectory is created. If the path does not correspond to the directory, the particular iteration will be skipped. The loaded images are pre-processed as explained above. The core of the project is to create a CNN for image classification. Initially a sequential model is created which is a linear stack of layers. A total of 5 convolution layers are used with two of them having 64 filters and rest three with 128 filters of size 3*3 an ReLu activation function. Each convolution layer is followed by a max pooling layer of size 2*2. Finally a flatten layer is added to convert the output of the convolution layers to a ID vector. Then, to avoid overfitting, a dropout layer is added to drop out 50% of the units. Finally, a fully connected layer with 512 units is added along with ReLu activation function. Tensorflow is configured to use GPU for training the model. The model is compiled by specifying optimizer, loss function and evaluation metric to use during training for the required number of epochs. The trained model is saved for later use. The dataset is divided in the ratio of 75:25 for training and testing respectively.Yolo v4 is used to detect animals in the image that also consists of background.Once the image is captured, if the model detects it as a wild animal, then an email is sent to the respected authorities.

VII. RESULTS

After training the model with 15 animals, the tested model provided the best result of 98% accuracy for the number of epochs of 30. The figures below represent the detection of an animal along with the accuracy with which they are detected.



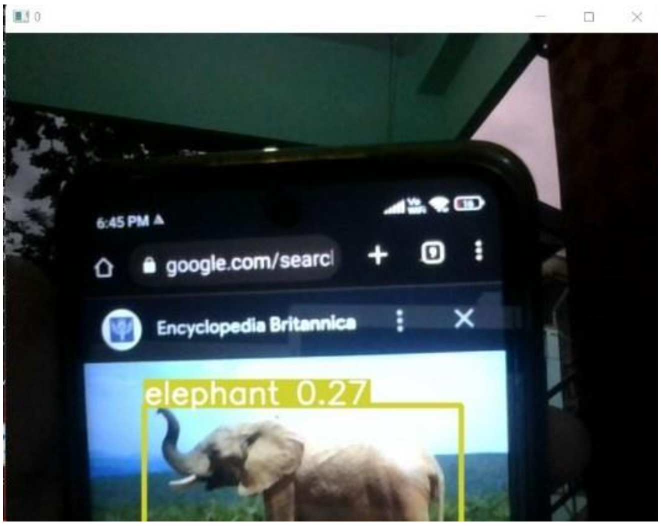


Fig: Detection of animals

VIII. REFERENCES

[1] Durmus Ozdemir and Musa Selman Kundurachi, "Comparison of Deep Learning Techniques for Classification of the Insects in Order Level With Mobile Software Application", Creative Commons Attribution, ISSN: 2169-3536, Vol. 10, 2022, 35675-35684.

[2] Nidhal K. El Abbadi and Elham Mohammed Thabit A. ALSAADI, "An Automated Vertebrate Animals Classification Using Deep Convolution Neural Networks", 2020 International Conference on Computer Science and Software Engineering (CSASE), Duhok, Kurdistan Region – Iraq, ISBN: 978-1-7281-5249-3, April 2020, 72-77.

[3] Sonain Jamil, Fawad, Muhammad Sohail Abbas, Faisal Habib, Muhammad Umair and Muhammad Jamil Khan , "Deep Learning and Computer Vision-based a Novel Framework for Himalayan Bear, Marco Polo Sheep and Snow Leopard Detection", 2020 International Conference on Information Science and Communication Technology, ISBN: 978-1-7281-6899-9, 2020.

[4] Tibor TRNOVSZKY, Patrik KAMENCAY, Richard ORJESEK, Miroslav BENCO and Peter SYKORA, "Animal Recognition System Based on Convolutional Neural Network" ADVANCES IN ELECTRICAL AND ELECTRONIC ENGINEERING, Vol. 15, Issue 3, 2017.

[5] T. Matsui and M. Ikehara, "Single-Image Fence Removal Using Deep Convolutional Neural Network," Vol. 8, pp. 38846-38854, 2020.