Automated biometric system for individual identification of Mugger crocodile (Crocodylus palustris): A Deep learning approach

***Abstract*-**In the field of animal behaviour and ecology, individual identification plays a critical role. Traditionally, invasive methods have been employed for this purpose. This study explores adapting a deep learning approach for non-invasive individual identification of free-ranging mugger crocodiles (Crocodylus palustris) utilising Unmanned Aerial Vehicle (UAV) imagery and the YOLOv8 deep learning model. A key aspect of this adaptation involves transitioning from the previously used YOLOv5l model, which operates within a Python 2.7 environment, to the latest YOLOv8 model requiring a Python 3 environment. Our dataset has been expanded from 143 to 160 individual mugger crocodiles for training. The data was split into a 90% training set and a 10% validation set. Training will be conducted on the university's supercomputer, Param Shavak, configured with a Python 3 environment suitable for YOLOv8. Future steps involve exploring data augmentation techniques and transfer learning to enhance model performance.

***Keywords* -** Convolutional neural network, Deep learning Free-ranging, Individual identification, Mugger crocodile, Unmanned aerial vehicle, Deep learning, YOLOv8, UAV imagery, Individual identification, Mugger crocodile

Introduction

Individual identification of animals plays a vital role in behavioural and ecological studies. The research by Desai and others used deep learning to identify muggers based on dorsal scute patterns instead of the traditional methods involving capturing the animals and tagging them. The new approach achieved TPR (re-identification of trained muggers) and TNR (differentiating untrained muggers as 'unknown') values of 88.8% and 89.6%, respectively. The trained model showed 100% TNR for the non-mugger species it was tested on. The images were captured using UAVs.[2]

Methodology

The original study leverages existing research that employed YOLOv5l, a convolutional neural network (CNN) model, for individual mugger crocodile identification using UAV imagery[1]. We aim to adapt this approach using the latest YOLOv8 model.

a) Dataset Preparation

The project began by merging an existing dataset of 143 individual mugger crocodiles with an expanded dataset containing 160 individuals. This resulted in a dataset encompassing a total of 303 mugger crocodiles. Each individual has multiple images focused on their dorsal body region.

b) Computing Environment

Param Shavak, the supercomputer at our university, will be utilised for training the YOLOv8 model. We configured a Python 3 environment suitable for YOLOv8 on the supercomputer.

c) Training Data Split

The combined dataset was divided into a 90% training set and a 10% validation set. The training set will be used to train the YOLOv8 model, and the validation set will be used to monitor the model's performance during training and prevent overfitting.

d)Yolov8 model:(YOLO-You Only Look Once)

Yolov8 is the latest SOTA model(state-of-the-art model) which can be used for object detection, image classification and instance segmentation tasks. It is built by Ultralytics. It has shown highest accuracy on databases of COCO and roboflow 100.

Yolov8 uses anchor box free prediction which means it directly predicts the centre of object instead of offset from anchor box. The Yolov8 model has an advanced backbone and Neck architecture. The model we used is yolov8l for detection.

Backbone: The backbone of YOLOv8 is a modified version of the CSPDarknet53 architecture1. This architecture consists of 53 convolutional layers1. The first 6x6 Conv is replaced with 3x3 Conv in the Backbone. Two Conv layers (No.10 and No.14 in the YOLOv5 config) are deleted2.

Neck: The neck connects the backbone and the head3. In YOLOv8, state-of-the-art neck architectures are utilized2. The traditional Conv (convolution) layers in the neck module of YOLOv8 are replaced with lightweight GSConv4. The dimension of No.13 Upsample output should be 80×80×512×w, not 80×80×256×w; the dimension of No.14 Concat output should be 80×80×768×w, not 80×80×512×w; the dimension of No.17 Concat output should be 80×80×768×w, not 80×80×512×w2.

Head: The head of YOLOv8 consists of multiple convolutional layers followed by a series of fully connected layers1. These layers are responsible for predicting bounding boxes, objectness scores, and class probabilities for the objects detected in an image1. The first 1x1 Conv is replaced with 3x3 Conv in the Bottleneck (this may change depending on input data).

6. Results:

Waiting for freeing up of computation resources at the institution to begin training.

7. Discussions:

Waiting for response of using GPU at full potential from the institution.

8. Conclusion:

Waiting for response of using GPU at full potential from the institution.

References

[1]https://www.researchgate.net/publication/369914392_DC-YOLOv8_Small_Size_Object_Detection_Algorithm_Based_on_Camera_Sensor

[2]https://www.sciencedirect.com/science/article/abs/pii/S1574954122003247?via%3Dihub

[3]https://github.com/ultralytics/ultralytics

[4]https://docs.ultralytics.com/models/yolov8/