Wildlife Detection & Identification through Machine-Learning AI Image Analysis

Abhik Biswas [1]
Faculty Advisor: Dr. Maria Diuk-Wasser [2]

[1]Data Science Institute, Columbia University [2]Ecology, Evolutionary and Environmental Biology Lab, Columbia University

Abstract

This research focuses on leveraging images from the Urban Wildlife Information Network (UWIN) to investigate the effects of urbanization on wildlife and disease vectors across NYC. By utilizing nearly 50 wildlife cameras placed in parks and greenspaces, the project generates a vast amount of data in the form of hundreds of thousands of images. To efficiently process this data, machine learning models like Megadetector are being implemented to automate the detection and identification of species. The research involves training these models on over 200,000 manually processed images and developing a pipeline for continuous model re-training as new data is collected, enabling scalable and accurate analysis. YOLOv5 model was used to carry out the process of generating training data, and subsequently the task of species identification.

Figure 1. A camera trap image showing white-tailed deers

Data & Methods

The data for this study is sourced from nearly 50 wildlife cameras strategically placed across NYC parks and greenspaces, capturing images along an urbanization gradient from Brooklyn to Nassau County. These cameras have generated hundreds of thousands of images, with over 200,000 of them manually processed to identify and catalog local wildlife species. This annotated dataset is crucial for training machine learning models to automate future wildlife identification. Additionally, metadata from these images, including species information and environmental factors, is uploaded to the Urban Wildlife Information Network (UWIN) database, which supports comparative research across cities globally.

Generating Training Data

Since each image had labels associated with it, we needed to generate the bounding box coordinates as a downstream task.

- The MegaDetector v5 Model, developed by Microsoft's AI for Good Lab was used to generate the bounding box coordinates. This model has 121M parameters, and can detect 3 classes: Animals (1), Humans (2), and Vehicles (3).
- The coordinates obtained from this were (x_{min}, y_{min}, h, w), where (x_{min}, y_{min}) is the lower left corner of the rectangle, and h & w are the height and width of the bounding box respectively. All those bounding boxes with a confidence score ≥ 0.2 were considered to be significant.

MegaDetector Outputs

Figure 2. Inference from the MegaDetector Model

You Only Look Once (YOLO)v5 Model

YOLOv5 is a state-of-the-art, real-time object detection model developed by Ultralytics. It builds upon the YOLO (You Only Look Once) family of models, offering improved speed, accuracy, and flexibility. YOLOv5 utilizes techniques like Mosaic data augmentation, auto-learning bounding box anchors, and Class Weighted Image Classification (CWIC) to enhance performance. The model comes in different sizes (YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x), balancing trade-offs between speed and accuracy. Implemented in PyTorch, YOLOv5 supports transfer learning and hyperparameter tuning, with a focus on simplicity and scalability, making it suitable for various applications, from autonomous driving to security systems. In this case, YOLOv5s model, with weights trained on the COCO dataset was used as a starting point.

Input Data

Figure 3. Inputs to the YOLOv5 Model - only those images were chosen as inputs that had only one annotation. The bounding box coordinates were obtained from the MegaDetector Model, and transformed to the YOLOv5 Format - $(c_x, c_y, \text{width}, \text{height})$

Model Performance & Outputs

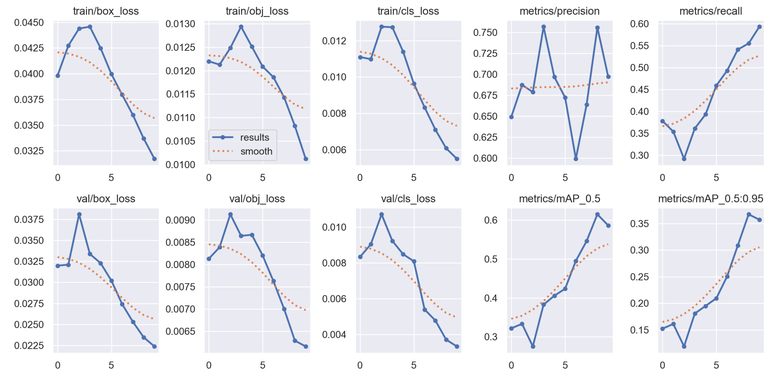

Figure 4. Various training and validation metrics, the top row represents training metrics, and the bottom row, validation metrics. From **left to right**: (**top row**) - box loss, object loss, cross-entropy loss, precision & recall, (**bottom row**) - box loss, object loss, cross-entropy loss, mean Average Precision at **IOU** threshold of 0.5, and mean Average Precision over different **IoU** thresholds, ranging from 0.5 to 0.95. The model was trained for 10 epochs.

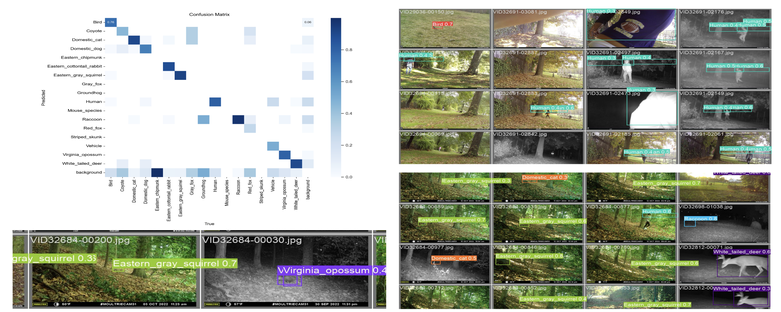

Figure 5. Inferences by the YOLOv5 Model. The species predicted with the highest confidence are indicated, along with the confidence of prediction. The confusion matrix for the predictions is present on the top left corner.

Future Scope

- Mannual Annotations - Mannual annotations of multiple objects in an image, along with more accurate bounding boxes could lead to potentially better model performances.
- Newer Models - Modern model structures like YOLOv8 can be leveraged, depending on the availability of resources to obtain better predictions.
- Given that we have a large number of samples, we can leverage GPUs to train these models from scratch.