**DATA612/MSML612 Assignment 2 Proposal**

**By Cole Bromfield and Dennis Reyes**

**Problem statement**:

Wild animals can present a distinct threat to safety and security, whether of individuals, livestock, or assets. As such, early detection is crucial. A Convolutional Neural Network (CNN) model can provide helpful aid in this endeavor, detecting the presence of a wild animal, in addition to whether it presents a danger. The differentiation of dangerous animals and harmless ones is left up to the user and may differ by his or her goals for the technology, as certain species may present a higher level of danger to human life, while others may pose a greater threat to livestock or assets.

**Background/Significance**:

Object detection, a fundamental computer vision task, entails identifying and categorizing objects within digital images. In the early days of object detection, due to limited image representation techniques, most algorithms relied on manually crafted features (Borah, 2020). Traditional methods employed sliding window-based approaches and image classifiers for each segment. However, these approaches were computationally intensive and often lacked accuracy.

In the evolution of object detection, traditional methods relied on handcrafted features and sliding window-based approaches. However, the breakthrough came with deep convolutional networks. These neural networks, equipped with multiple layers, automatically learn powerful feature representations from image data (Borah). This shift opened up new possibilities and significantly improved accuracy in object detection tasks.

**Applications**:

Object detection has a wide range of applications, including autonomous driving, robotics, and surveillance (“7 Real-Life Use Cases”, 2024). Dangerous animal detection can be most easily applied to surveillance, with early warning systems notifying a user when a surveillance camera has detected an animal which presents a danger. As previously stated, the differentiation of dangerous and non-dangerous animals is flexible (although the model must be re-trained with any redefinition)—in the case of a yard where children play, the “dangerous” label may be applied to bears and wolves, whereas for the protection of livestock, foxes and coyotes may be given the “dangerous” label. In still another use case, sharks and orcas could be the target for underwater cameras.

**Motivation of Using Deep Learning:**

Object detection, which relies on large, high-dimensional datasets, benefits significantly from deep learning. Convolutional Neural Networks (CNNs) have excelled in image recognition tasks by automatically learning relevant features from images. These learned features play a crucial role in object localization and classification. Compared to traditional methods, deep learning-based techniques offer superior accuracy and speed. By capturing both low-level and high-level image features, deep learning models enhance detection performance. Moreover, deep learning models learn directly from raw data, eliminating the need for manually crafted features. This end-to-end learning approach enables models to adapt to complex patterns and variations in object appearance.

**Plan of execution**:

We will train a simple CNN model (such as the ones covered in Assignment 1) using the Keras CIFAR-100 dataset. We will examine the 100 unique labels within this dataset, and map it to three new labels: “dangerous”, “not dangerous”, or “no animal”. With 3 classes (as opposed to 100), this model should require significantly shorter computational time than a model trained to classify animals directly by species. In terms of framework, we will use Keras to obtain the train/test data and Scikit-learn to build and train the model. If the results are numerized (No animal = 0, non-dangerous animal = 1, dangerous animal = 2), a simple distance metric (such as Euclidean) can be utilized to determine the accuracy of the model. Removing the “no animal” class altogether, a local accuracy metric can inform us how accurately the model differentiates animals between dangerous and non-dangerous.

References

*7 Real-Life Use Cases of Object Detection*. (2024). Folio3; Folio3 Software, Inc.

https://www.folio3.ai/blog/use-cases-of-object-detection/

Borah, C. (2020, November 11). Evolution of object detection. *Analytics Vidhya*.

https://medium.com/analytics-vidhya/evolution-of-object-detection-582259d2aa9b