Machine Learning Eng. Capstone Project

Wildlife Image Classifier

Description

Wildlife cameras are common for determining where animals live, travel or for counting their numbers in an area. These cameras typically take pictures when triggered by a motion sensor. The images are then either stored or sent to the customer via a cellular network. A wildlife image classifier is proposed for the purpose of cataloging the images for faster viewing later based on animal type. This is similar to [1] and [2].

Proposed Project

To train an image classier using transfer learning from an existing model, such as MobileNetV2, to identify wildlife not part of the original dataset or more accurately identify them than the base model. Create and publish a blog post describing the process, lessons learned and results of the project.

Tasks

The following are some of the tasks required to complete the development of the wildlife classifier:

1. Obtain and preprocess the dataset from Oregon Wildlife dataset.
2. Perform data cleaning and augmentation, such as scaling, rotation, mirror, etc.
3. Decide upon a base model, such as MobileNetV2, to use as a basis for transfer learning.
4. Setup and train a new model using transfer learning, Figure 1, using TensorFlow.
5. Compare the classification percentage, of the test dataset, for the transfer learning model with that of the base model.

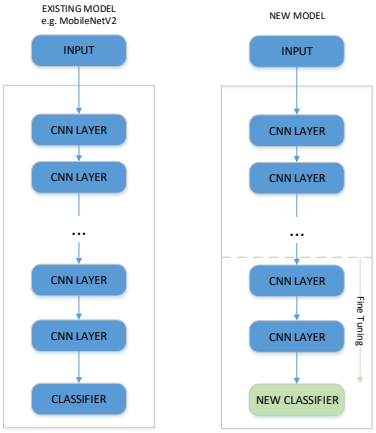

Figure 1

Evaluation

In order to evaluate the model, it is proposed to load a pre-existing model, such as MobileNetV2, and measure accuracy of the test data set. Then apply transfer learning to the core model and test it on the same test images to evaluate potential accuracy improvements.

Deliverables

The deliverables for this project shall include:

1. A Github/Kaggle repository of the work done
2. A blog post written showing the steps taken, key learning points, and results.

Future Work

Time permitting, this model would be deployed on an edge device, such as the NVIDIA Jetson Nano. Then, a demonstration of the ability to capture images from the camera and process them real-time on the Nano platform should be demonstrated.

Resources

Getting Started with AI on Jetson Nano, NVIDIA.com

https://github.com/dusty-nv/jetson-inference, Deploying Deep Learning, NVIDIA

Courses

A Complete Guide on TensorFlow 2.0 using Keras API, Udemy.com

Books

Deep Learning with Python; Francois Chollet

Datasets

https://github.com/visipedia/iwildcam_comp

https://www.kaggle.com/virtualdvid/oregon-wildlife

References

[1] https://www.pnas.org/content/115/25/E5716

[2] https://www.kaggle.com/c/iwildcam2018/overview