AI-Based Bird Species Identification from Trail Camera Footage

This is a Chickadee my favorite songbird! When they chirp it sounds like they are saying cheeseburger. https://www.youtube.com/watch?v=E25czyNMhqE

Abstract

Wildlife conservation relies on accurately identifying bird populations and behaviors, often requiring manual analysis of trail camera footage. This process is time-consuming and inefficient, making it difficult for researchers to analyze large datasets. To address this, I propose leveraging computer vision and deep learning to automate bird species identification from images.

Instead of training a model from scratch, this project will use a pre-trained convolutional neural network (CNN), such as MobileNet or ResNet, to classify birds in trail camera images. OpenCV will be used for image preprocessing (background subtraction, filtering, and noise reduction), while TensorFlow will handle model inference and optimization.

Additionally, this project will compare CPU vs. GPU performance to assess how parallel computing speeds up image classification. If my trail camera does not collect enough images, I will use an online dataset (such as Cornell Lab of Ornithology, iNaturalist, or Kaggle datasets)

as a backup. Success will be measured by how accurately and quickly the model identifies bird species. The expected outcome is a scalable and efficient pipeline for automated bird identification, making species monitoring faster and more accessible for conservation efforts.

Schedule

WEEK	TASK
FEB 5	Submit Project Proposal (Milestone 1)
FEB 6 - FEB 12	Set up environment: Install TensorFlow, OpenCV, and test basic scripts to ensure everything runs smoothly
FEB 13 - FEB 19	Learn OpenCV: Use image preprocessing techniques like resizing, filtering, and background subtraction to clean up trail camera images
FEB 20 - FEB 26	Use a pre-trained MobileNet model: Run test images through the AI model and classify birds
FEB 26	Submit Part 1: Software Exploration
FEB 27 - MAR 10	Automate classification: Process multiple images at once and improve model accuracy
MAR 11 - MAR 24	Benchmark CPU vs. GPU performance: Measure processing speed and test optimization strategies
MAR 25 - APR 7	Experiment with ResNet as an alternative to MobileNet (optional, depending on time)
APR 8 - APR 14	Final report and project presentation
APR 15	Final Project Due

Part 1: Software Exploration

For the software exploration phase, I will focus on OpenCV and TensorFlow, two powerful tools for image processing and deep learning.

- OpenCV will be used for image preprocessing, such as background subtraction, filtering, and feature extraction, to enhance the quality of trail camera images before classification.

- TensorFlow will be used to deploy a pre-trained deep learning model (such as MobileNet or ResNet) for bird species recognition, eliminating the need for training a model from scratch.

Exploration Steps

1. Set up TensorFlow and OpenCV on an HPC/GPU environment.
2. Collect and preprocess sample bird images from my trail camera.
3. Backup Plan: If my trail camera does not capture enough data, I will use an online bird dataset (such as Cornell Lab of Ornithology, iNaturalist, or Kaggle datasets) to supplement the images.
4. Run a pre-trained MobileNet model on sample images to test classification accuracy.

Part 2: Benchmark and Optimization

To benchmark performance, I will test:

- How long it takes to classify an image using CPU vs. GPU. (if we have GPU access which I think we do)
- How image preprocessing (e.g., background removal, denoising) affects classification accuracy.
- The effects of parallel processing techniques like batch inference on large datasets.

Optimization Goals:

Test OpenCV optimizations to improve processing speed.
Compare CPU vs. GPU processing times and determine speedup factors.