Sprint 3 Formal Feedback
Reviewer: Jimmy Nguyen
Reviewee: Alec Campodnico

Team Name: Angry Birds
Project: Bird Chirps

Libraries Noted:

- TensorFlow
- OpenCV
- NumPy
- OpenPyXL
- JSON
- Tkinter

Readability:
Improvements: The code remains readable with clear variable naming and consistent formatting. The use of comments before major blocks (e.g., preprocessing, frame processing) helps outline the process steps.
Areas for Improvement: While the variable names and step comments aid readability, the logic behind certain operations, particularly within complex functions like thresholding, could benefit from more detailed comments. Explaining why specific values are chosen or the rationale behind certain thresholds can make the code more accessible to others.

Efficiency:
Improvements: The script smartly employs TensorFlow Lite for model inference, continuing its approach towards efficiency. The frame_divisor concept for processing selected frames is a smart way to manage computational load, especially relevant for real-time or resource-constrained environments.
Areas for Improvement: Consider exploring more advanced techniques for dynamic frame skipping based on motion detection or scene changes, which might further optimize processing without compromising the detection performance.

Modularity:
Areas for Improvemen: Further compartmentalization could be achieved by encapsulating the GUI setup, video processing, and output generation (including Excel writing) into separate classes or modules. This would enhance maintainability and potentially facilitate unit testing.

Error Handling:

Areas for Improvement: Robust error handling is still needed. For instance, the script should include try-except blocks around file operations (reading configurations, writing Excel files) and TensorFlow operations to catch and handle exceptions gracefully. Additionally, validating user inputs from the GUI (e.g., frame skip interval) to prevent crashes due to invalid data.

Overall possible future suggestions:

Performance Metrics: Implementing a way to track and display performance metrics (e.g., frames processed per second) in real-time could provide valuable insights for optimization.
Configurability: Enhancing the configuration file to include more parameters (e.g., GUI window size, output formats) could make the script more versatile for different use cases.
GUI Usability: Adding features to the GUI, like progress indicators or pause/stop buttons, could improve the user experience, especially for lengthy processing tasks.