BiVo: An Open-Source Foundation for Remote Bird Vocalizations

Kevin Imlay, Daniel Mercado, Yasmin Vega-Nuno, Anqi Wang

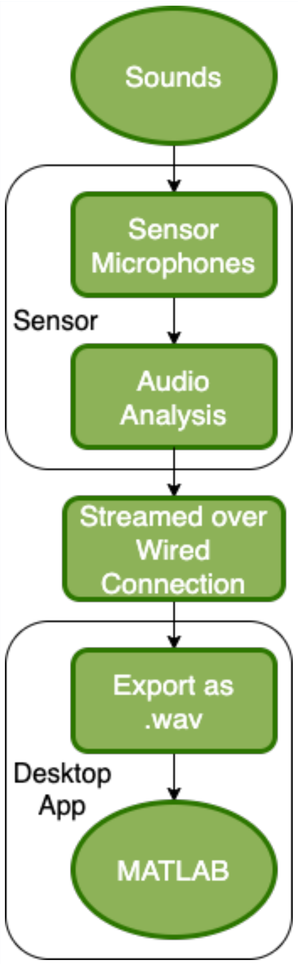
Department of Computer Science, Northern Arizona University
Project Sponsor: Paul Flikkema, PhD., School of Informatics, Computing, and Cyber Systems
Project Mentor: Andrew Abraham, School of Informatics, Computing, and Cyber Systems

College of Engineering, Informatics, and Applied Sciences

Motivation

Dr. Paul Flikkema's research team is developing an artificial intelligence tool to help scientists classify birds based on audio snippets. This tool needs to be trained using audio of bird vocalizations containing background noise such as weather and other animals. Current wildlife recording tools lack extensibility and fluent workflows for scientists, making collection of audio cumbersome. These deficiencies inspired us to develop BiVo.

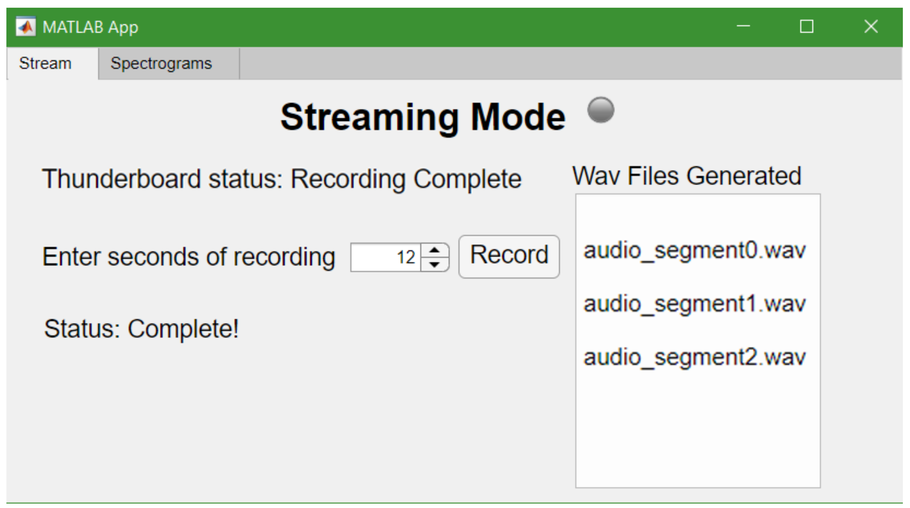
Architecture



- Data starts as sounds in the environment.
- Microphones on the sensor record these sounds.
- The recorded audio is analyzed for bird vocalizations.
- Audio containing bird vocalizations is streamed to the desktop application.
- The audio is exported as .wav files to the user's computer.
- The audio is brought into MATLAB for the user to further use.

Our Solution

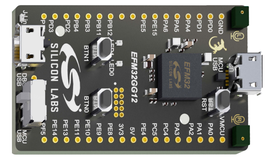
BiVo simplifies the recording of bird vocalizations and bringing them into analysis. Audio is streamed straight from the sensor to the user's computer, removing the need for daily swapping of SD cards.



BiVo is open source, and its sensors are developed on a common development board to allow for easy sourcing and addition of hardware. The desktop application provides a MATLAB interface for the user and a Python backend to allow easy and fast additions in functionality.

Technologies

- GUI and System Control
- Board Development
- Front End Serial Communication
- Embedded Sensor System
- Front End Data Management
- Source Control



- Thunderboard EFM32GG12: Sensor board

Challenges

- Toolchain issues with importing libraries and unpredictable behaviors.
- Bad library dependencies on broken code.
- Restructuring project due to time limitations.

Future Work

This system is only in the early development phase, so there are many opportunities for additional features. A few planned future features are:

- Bluetooth networking of sensors and sending data to the cloud using a cellular-enabled modem.
- Standalone modes to allow the sensors to be left in the field unattended.
- Time intervals of recording to save energy and target usage to specific times of the day.
- Additional storage to allow for storage of audio on the sensor itself.