Harsha Jagarlamudi, Kelly Kong

DSC 180B: A09

Mentor: Ryan Kastner, Jacob Ayers

4 February 2022

Project Report Checkpoint

1. Introduction

Climate change is a global forefront issue that affects all species. Tracking the biodiversity of species and their population can help gauge how much climate change has affected these species. Within our domain, the focus is on birds, so we use bird audio to measure their biodiversity and to potentially monitor endangered birds. The intent behind this project is that by being able to passively identify and differentiate bird species in a reliable way, it will be possible to gain a better understanding of the environments they belong to. We can determine what bird species belong to what habitats and get a stronger grasp of changes to an ecosystem based on the species we believe to be present there.

However, the problem is that a lot of machine learning models specifically for acoustic species IDing need labeled data that require manual input and time. Within our section, we saw this as we all had to manually label over 200 bird clips. This process can be very time-consuming, can be prone to inconsistencies between different human annotators, and can have human input errors that would be less expected of an automated solution.

2. Methods

The cross-correlation based template-matching project involves Digital Signal Processing for audio data segmentation. This technique takes a sample of audio and compares it to a set of other audio, outputting a score array based on similarity to the sample. Then, an algorithm is run to determine the best scoring audio clips which are hopefully similar to the sample and can be used as training data for scalable models. Our intention with this project is to create a reliable way to automatically label audio data by species and cut down the time investment and manual input needed to create strong training data for machine learning models that work on acoustic species IDing.

The cross-correlation pipeline is Python-based and uses the packages PyHa, Librosa, and SciPy. PyHa is a package designed to convert audio-based “weak” labels to strong intraclip labels. It provides a pipeline to compare automated moment-to-moment labels to human labels by combining Python and Piha.

The first step in performing cross-correlation is deciding on data to run the pipeline on. In our case, to make sure cross-correlation was working as intended initially, we chose to run the pipeline on ScreamingPiha audio data. The first function we call is load_audio, which takes in a clip path and loads the audio data at 12kHz into a variable that is returned.

Following this, we want to do simple exploratory data analysis on this clip and create a spectrogram to take a look at. This function, spectrogram, takes the output of load_audio as an input and short-time Fourier transforms the audio. This transformed audio is saved to a variable, used to display a spectrogram of the data, and then returned as an output.

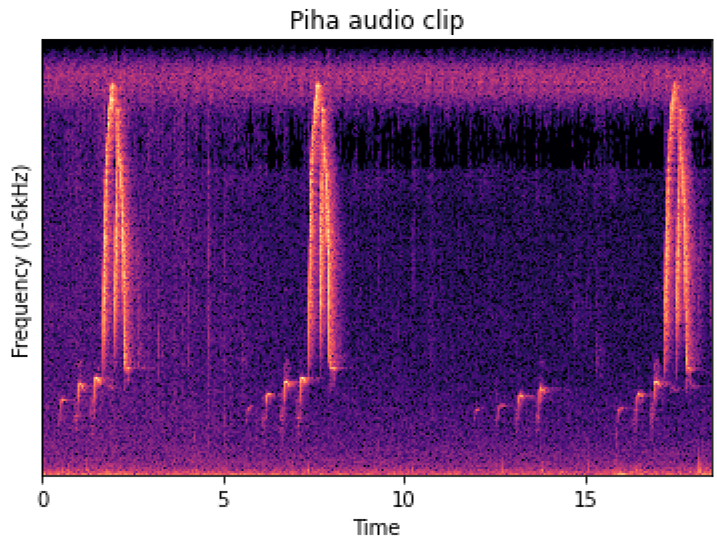

Figure 1: Screaming Piha Audio Clip Spectrogram

Next, we want to use an audio file from the same species to create an audio template. In our case, we use the function, template, that takes in the output of load_audio. A section of the audio file that contains the species sound is short-time Fourier transformed and saved as a template. This template is then plotted, displayed, and returned as the output of the function.

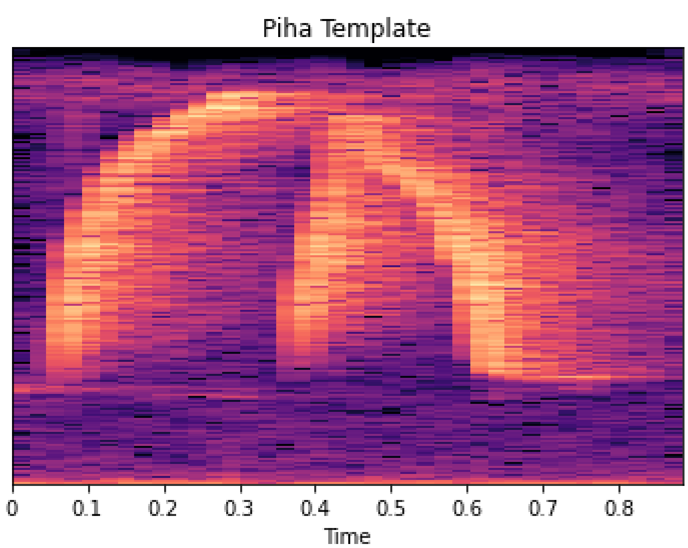

Figure 2: Screaming Piha Audio Template

Finally, we want to perform the actual cross-correlation. For this, we use the function, correlation, which takes in as input the clip path,and the output of load_audio, spectrogram, and template. This function defines isolation parameters and uses the scipy.correlate2d function to perform correlation on the transformed audio using our template. Using PyHa's steinberg_isolate function, we created a resulting dataframe and were able to display a spectrogram of peaks alongside a local score array that represented our cross correlation model's prediction on the audio.

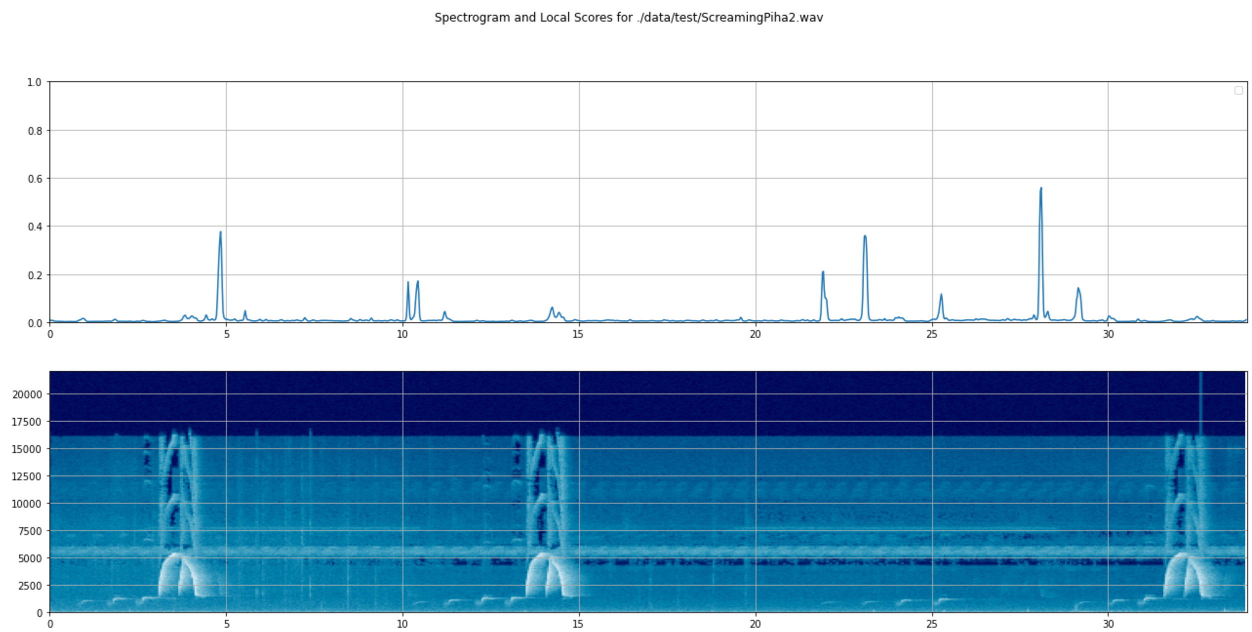

Figure 3: 2D Cross Correlation Spectrogram and Local Score Array

3. Future Steps

Moving forward, we want to improve the speed of the cross-correlation algorithm to run it more effectively on larger sets of data. In addition, we want to manually evaluate its effectiveness on different sets of data beyond the Screaming Piha, potentially with different isolation parameters. The species we specifically want to look into come from the Scripps and Peru audio sets. These species are the California Gnatcatcher, California Towhee, and American Crow from the Scripps dataset and the Sepia-Capped Flycatcher, Screaming Piha, Southern Mealy Amazon, and General "Macaw" from the Peru dataset. In addition, we intend to make the pipeline both more fluid and simpler to use so that any template can be passed in alongside full audio data and generate local score arrays with little manual input.