MSU ECE REU Program : Semi-Automatic Detection of Red-winged Blackbirds

Allison Busa

August 2018

1 Abstract

I set out to discover if there was a distinguishing feature of male Red-winged Blackbird songs that could be used to detect the species in audio recordings. During the course of this program, I found that the modulation of power of different frequencies of the song, the trill part of the song, was a constant pattern present through all Red-winged Blackbird songs. Furthermore, this modulation occurs at range of 20 to 80 Hz for the recordings of United States birds tested and in one region, multiple frequencies of modulation in this range occur. I then used this information to create a MATLAB program with a GUI that can semi-automatically find the frequency of modulation. I also created two MATLAB programs that can search for Red-winged Blackbird songs in a long audio file; one using parameters that the user sets by typing them in and the other by inputting a reference signal and the GUI to specify parameters.

Table of Contents

1. Introduction
   2. Background Information
   3. Methods and Results
   4. Analysis
   5. Conclusion
   6. References

2 Introduction

In this paper, I describe a method for semi-automatic detection of Red-winged Blackbird male mating songs from the United States in both short and long audio files.

Knowing if a species is present in an area is important for park rangers, biologists[19], environmentalists[18] and construction workers; however, the current methods of determining if a species is present in an region is expensive in resources and time. For example, when a construction worker needs to know how many birds reside in the area she wants to build on, she has to stop construction for days or weeks and hire expert ornithologists in that time to look for the birds[17]. In another case, it took four years for a study on how Red-winged Blackbirds divided territory on only a .15km2 to .25km2 space[25]. Finally, the United States National Park Service uses aircrafts to count the number of large mammals in a park and observers for the National Park Service spend hours at a time recording data[36].

For the above reasons, I set out, with help from my mentor , Rob Maher, to create a new way to find signals in long audio files, with a focus on signals of the songs of male Red-winged Blackbirds. By finding the signals of a species in an audio file, an individual can know if a species is present in a region without disrupting the animals in the region and without hours of active devotion to recording data.

For the duration of this program, I set out to discover and create. I set out to discover the key distinguishing factor of Red-winged Blackbird male mating songs from other bird songs and from other sounds, how to abstract that characteristic into a medium discernible by a computer and if and how that characteristic differs for Red-winged Blackbirds of different regions of the United States. I set out to create a program that could detect that distinguishing characteristic of a Red-winged Blackbird and a program that would use the formerly mentioned program to detect Red-winged Blackbird songs in long audio recordings (defined as > 10 minutes) with at least 60 % of the results truly being Red-winged Blackbirds. Recurring obstacles in this project were variability in the trends, length and value of the data, overlapping sound that distorts a signal and inherent noise that distorts a signal.

At the end of the program, I had discovered that the key feature of male Red-winged Blackbird songs is the pattern of modulation in amplitude for a range of frequencies, which is distinct from other songs because of its shape, and varies for different Red-winged Blackbirds in one area but in different regions of the United States, the frequency of the modulation mostly stays between the bounds of 30 to 85 Hz. I also created a program that calculates the frequency of modulation, with the help of the program user through a GUI, and another program that locates the occurrences of male Red-winged Blackbird songs, providing 59.5 to 85.8 % of time occurrences of actual Red-winged Blackbirds correctly.

The remainder of this report is organized as follows. Section 3 provides background information about mathematical and general Electrical Engineering concepts, simple acoustical concepts and signal processing concepts used for this work. Section 4 describes the steps of the investigation in chronological order. In Section 4, work that involves my final programs starts at 4.5.4. Also, in Section 4, work that involves testing these programs on birds from different regions starts at 4.9. Section 5 and Section 6 contain my analysis and conclusion, respectively. "

3 Background Information

3.1 About Sound

When you make a noise, you create a force that displaces the surrounding air molecules with a certain velocity. The oscillating part of a sound wave is the pressure (or density, depending on how you look at it) of a unit of volume of air as well as the velocity at which the volume of air moves. The movement is oscillatory because the force is not being applied outwards forever; when the density of the air molecules gets to its peak, the surrounding air molecules push against each other, causing an equal and opposite force. At the peak, the sound is has the greatest pressure and this pressure is measured in deciBels, which is a logarithmic way of describing a ratio. Sound intensity is measured as such $I(dB) = 10\log_1 0[\frac{I}{I_0}$. From this,we see that sound pressure needs to be 10 times as high for a sound to be twice as loud. Furthermore, according to the Inverse square law, $I \sim \frac{1}{r(2}$, the intensity changes proportionally to the inverse of the distance from the sound source, squared. Based on this, moving twice as far from a sound makes the sound one quarter as loud[28].

What we perceive as sound, loud or quiet, is this change in pressure hitting a little drum in our ears that is attached to nerves. When the drum vibrates, the nerves get stimulated and send a signal to our brains. However, our ears cannot pick up all frequencies. Usually, the typical range of hearing is between 20 to 20,000 Hz.

In this paper, I will be describing ways to computational find similarities between sounds. Two sounds are similar if they exhibit similar frequencies or frequency patterns. Similar sound pressure does not make two sounds similar : if you talked at the same volume that the radio was playing at, you would not sound like the radio. However, if you heard a guitar and a piano play a middle C note, which is characterized as a sound wave at a 261.2Hz frequency [22], then you would notice a similarity. More complex sounds like speech patterns or bird calls exhibit more characteristics, like change in frequency over time, or pauses.

3.2 About the Sounds of Red-winged Blackbirds

Red-winged Blackbirds produce four types of sounds, they produce the male mating song, the female mating song, a flight and feeding call and an intense alarm call. The male's call is common in wetlands and classified as a "1-second song [that] starts with an abrupt note that turns into a musical trill"[26]. This Red-winged Blackbird sound is the one that I chose to focus on and from hereon in the paper, I will refer to it as the Red-winged Blackbird song. Furthermore, it has been proven that birds, like humans, have regional dialects[16].

3.3 Review of Mathematical and Electrical Engineering Concepts

3.3.1 Discrete Fourier Transform

The Discrete Fourier Transform is one of the most important mathematical tools in Signal Processing because it allows an individual to go between data about the pressure of the signal over time and about the power as a function of different frequencies.

The Fourier Transform stems from the concept of projection in Linear Algebra : you take a function or a set of function that is in one coordinate system and move the function or functions so that they lie on a different coordinate system. The original coordinate system in this case is time and power and the new coordinate system is frequency and power. Like in basic Linear Algebra, you are decomposing a signal into its basis vector components (with the components being frequencies instead of Euclidean distance in the X, Y, Z directions). Our basis functions for the Fourier Transform will be $e^{-2\pi i\omega x}$ and the powers can be calculated using $F(\omega_n) = \int_{-\infty}^{\infty} e^{-2\pi i\omega_n x}dx$. The Discrete Fourier Transform is used to find the power levels for a finite set of frequencies, which can be generated from the equation $f_k = f(k\delta)$ where k represents whole integers and δ represents the reciprocal of the sampling rate, or for signal processing how many samples are collected per second. According to the Nyquist theorem, we must not exceed the frequency of $\frac{1}{2\delta}$, because then samples of the maximum frequency would be sampled at less than two points per cycle, producing inaccurate results.

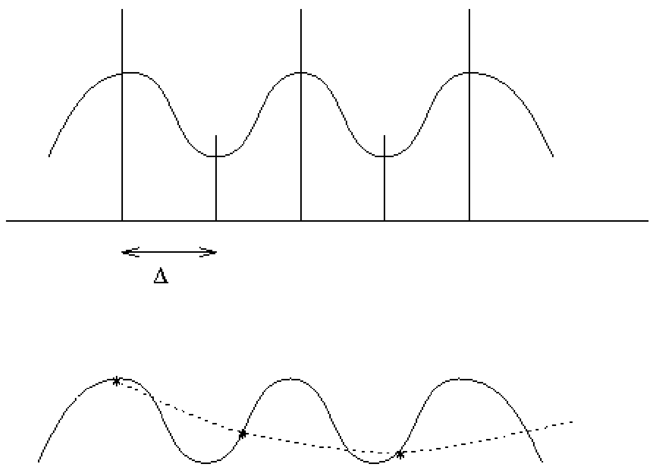


Figure 1: Graphic portraying Nyquist Thereom

[24] This being the case, we can only generate as many outputs as N inputs, with the limits corresponding to the upper and lower Nyquist frequencies (the limits that dictate how many samples per cycle will be too late and how many samples per cycle will be too much.) Considering all this, we can make the continuous Fourier Transform from above discrete.

$$\int_{-\infty}^{\infty} e^{-2\pi i\omega_n x}dx \approx \delta_{k=0}^{N-1} F e^{-2\pi ik\frac{n}{N}}$$

. This is the equation for the Discrete Fourier Transform. [24]

3.3.2 Convolution

If you have two functions, convolving one function onto another is placing one function over all the instances in which the other is present.

$$y(t) = \int_{-\infty}^{\infty} f(\lambda)h(t-\lambda)d\lambda$$

[7] This is the equation for convolution. It shows that the convolution of signals f and h with parameters of λ and t-λ is equal to the integral of the multiplication of these two signals with those parameters, integrated with respect to λ.

It could also be interpreted as the area of overlap of two vectors, u and v, under the points as v slides across u. In this sense, convolution could be thought of as multiplying polynomials whose coefficients are the elements of u and v. [8]

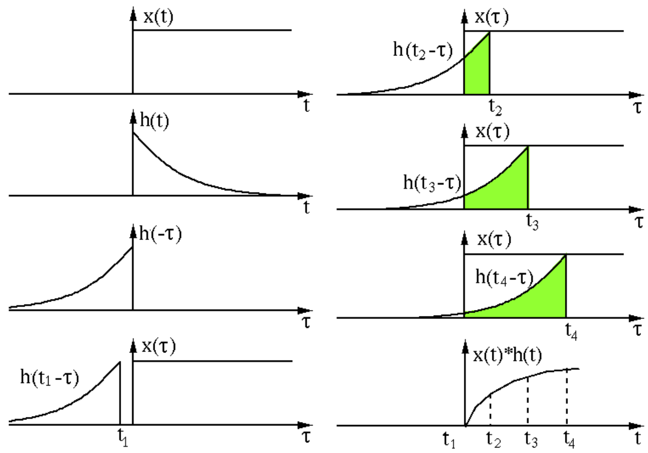


Figure 2: Graphic portraying Convolution

Another interpretation of convolution is that the function h(t-λ) is a memory weight. The convolution function receives an input in the form of a time interval, but while calculating the output, it remembers being at a certain value before (dictated by h(t-λ)) and will lean towards that value in the output. [32]

These last two interpretations feed into a function of convolution, which is smoothing a signal. If you convolve a jagged signal with a smooth sine wave, the output signal will create the signal while remembering the sine wave.

3.3.3 Euclidean Distance

Euclidean distance is what normally is referred to as just 'distance' and represents the shortest path between two points. When using euclidean distance to match a pattern, the distance between the values of two sets of data, two signals in this case, at each time interval are calculated. A low Euclidean distance is a sign of similar patterns.[13]

3.3.4 Cross Correlation

Cross correlation is used to determine how similar two signals are to each other and how much one signal has to be shifted in order to be maximally similar. Autocorrelation is cross correlation of a signal with itself and it is used to detect periodicity in a signal. Cross correlation is similar mathematically to convolution.

$$Convolution \rightarrow u(t) * v(t) = \int_{-\infty}^{\infty} u \star (\tau) v(\tau + t) d\tau$$

$$CrossCorrelation \rightarrow u(t) \otimes v(t) = \int_{-\infty}^{\infty} u \star (\tau - t) v(\tau) d\tau$$

where $u\star$ denotes a complex conjugate. [1]

3.3.5 Cross Power Spectral Density

Energy spectral density describes how the energy of a signal is distributed among the different frequencies. Power spectral density describes how to power of a signal is distributed among the different frequencies.[29]

Cross power spectral density is the Discrete Time Fourier Transform of the cross-correlation of the two signals. [15]

3.3.6 Magnitude-Squared Coherence

An estimate of magnitude-squared coherence is a function of frequency that takes in the power spectral densities of two signals as well as their cross spectral power densities, and outputs a vector of values spanning from 0 to 1. The values indicate how well the first function corresponds to the second function for each frequency. [21]

3.3.7 Band Pass Filters

In electrical engineering, a band pass filter keeps the signal the same between two intervals and lowers, or in other words, attenuates, the rest. Band pass filtering is useful in audio signal processing when looking for a certain signal; a certain signal is usually limited to only one frequency range, and it saves memory and computational power to discard of the frequencies that are not of interest. Even if you do not know what range your signal in an audio file is in, it is helpful to remove everything under 100 Hz, as this is usually noise.

3.3.8 Octave Band Filters

An octave band filter splits a signal into bands which have a highest frequency that is twice the lowest frequency [30]. A $\frac{1}{3}$ octave band filter takes those split bands and divides them even further into three bands. In order to create a $\frac{1}{3}$ octave band filter that is centered around one frequency, calculate the lower frequency and higher frequency as:

$$lowerFrequency = \frac{centerFrequency}{2^{\frac{1}{6}}}$$

$$higherFrequency = centerFrequency * 2^{\frac{1}{6}}$$

. [9]

3.3.9 RMS Envelope

An envelope of a signal outlines the overall shape of a signal. A Root Mean Squared (RMS) envelope is an envelope that is created by squaring each sample, calculating the average squared value for a given number of samples, and taking the square root of the average. This is done for all the samples in a dataset. The resulting RMS envelope can be used in place of the original dataset and has the advantage of reducing the number of samples in the data and smoothing out the data.

Another mathematical approach to calculating the RMS envelope which is less computationally taxing is to convolve the data squared with a vector of 1's that is the length of the window that you want to calculate the average over, divide by the length of the window, and take the square root of the resulting vector.

$$RMSEnvelope = \sqrt{\frac{Data^2 * [11111]}{5}}$$

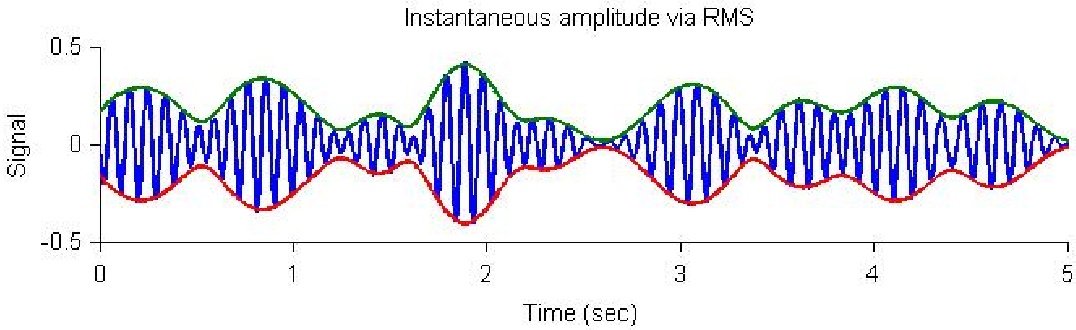


Figure 3: Graphic Portraying RMS Envelope

[20]

3.3.10 Median Filter

Median filtering is a filtering technique that sorts the samples in a time frame, takes the median and replaces any outliers with the average.[33]

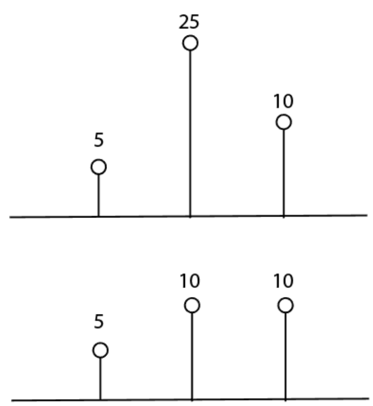


Figure 4: Graphing portraying a Median Filter

3.3.11 Hysteresis in Control Systems

In the context of control systems, hysteresis can be understood as a range or as the difference between the switch point and the reset point.[6] The range or points are values in a set of parameters which you are measuring; for example, water pressure, temperature or flow. The switch point is the highest value in the range and the reset point is the lowest value in the range. When the parameter you are measuring reaches the switch point, you allow a process to begin. This could be thought of as turning on a light or completing a circuit. When the parameter's levels drop down to the reset point, you stop the process- you turn the light off.

One example that highlights the usefulness of hysteresis is a filter in a tank of water. When a sensor detects that there is a lot foreign particles in the tank of water, the filter turns on. When the filter has brought the number of particles down to an acceptable level, the filter turns off.

3.4 Signal Processing Concepts

Sampling Rate - samples per second

Stereo vs Monochannel - An audio recorder can either have a monochannel (one channel) input, meaning that it has only one microphone. It can also have stereo (two channels), meaning that it has two microphones, usually corresponding to the left and right.

16 bit vs 32 bit - Either a value is stored with 16 bits of data or 32 bits of data. 32 bits is more precise, but it also costs more memory. [10]

.WAV files vs .MP3 files - .WAV files directly contain pressure levels for each time sample. .MP3 files are compressed to save bit space, and the lost data bits have to be estimated for certain time intervals. [23]

Clipping - The pressure of the sound in the recording surpasses the maximum recordable pressure on the audio recorder and the data becomes cut, or clipped, at the extrema.

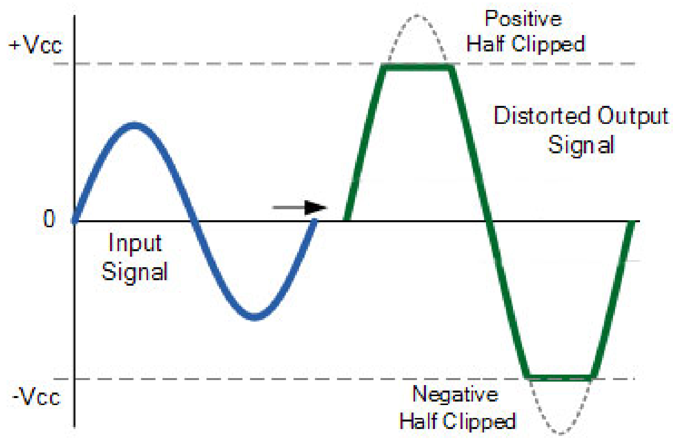


Figure 5: Graphic portraying Clipping

[4]

3.4.1 Spectrograms

A spectrogram is a graph that shows the power levels of different frequencies over time. The x-axis is time, the y-axis is frequency and the third axis, which is portrayed with color, is power.

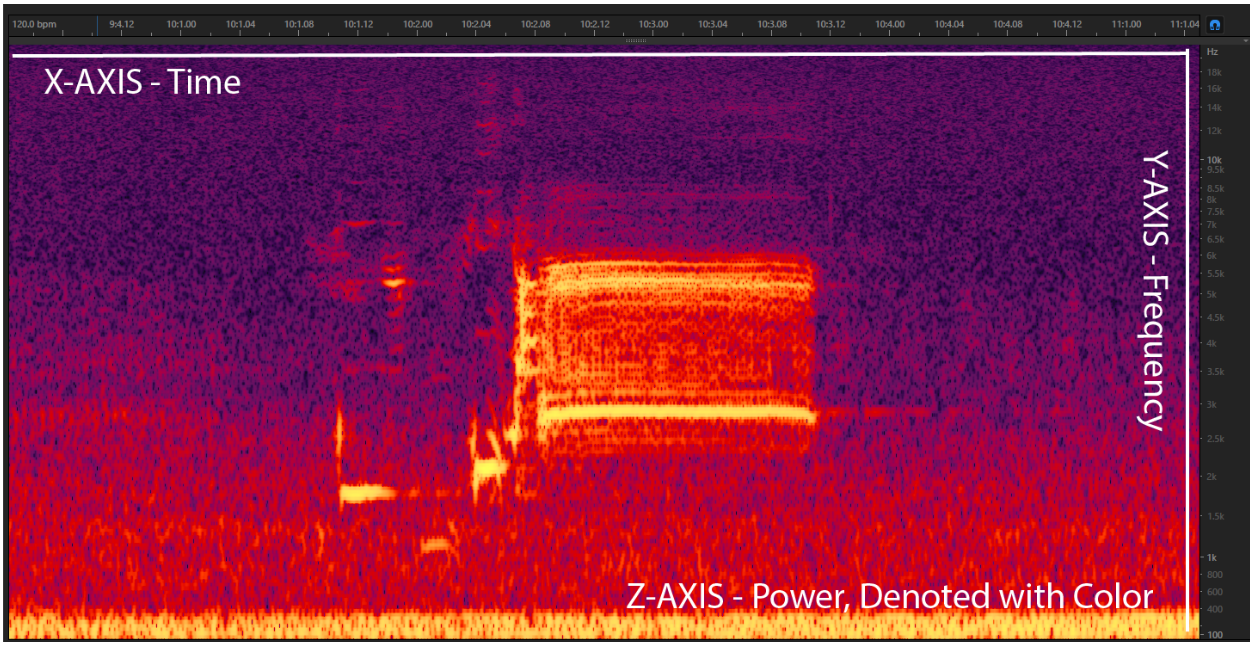


Figure 6: Typical Spectrogram

3.5 Audio Signal Processing Tools

3.5.1 Audio Recorders and Recording Sessions

We record audio signal data with an audio recorder. Audio recorders can either be big, for long term recordings, or small, for a couple of hours worth of recording or less. During this project, I mostly used the Roland Edirol.

In order to set up an audio recorder, there are a few things that you need to prepare:

- We set the sampling rate usually to 44.1kHz bc of the Nyquist rate, which states that the sampling rate should be twice the maximum frequency response.

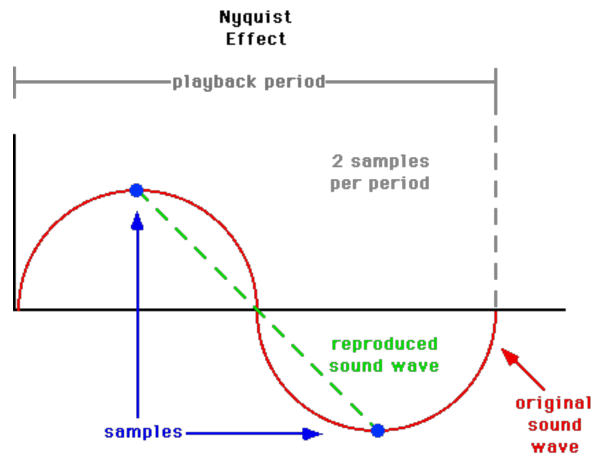


Figure 7: Graphic Portraying Nyquist Effect

[5]

- In order to prepare the recorder, we press the red middle button. This tells the audio recorder to start measuring pressure, but not to start recording, yet. Now, we can see and set the input level.

Figure 8: Graphic Showing Set-Up of Edirol

[14]

- We set the input level high enough so that we see in the input level bar that the pressure is varying through the whole bar, and not just at the bottom. We also want to avoid clipping. If we see that the pressure level is reaching the clipping box, then we turn the input level down.

- When the input level is set, we press the red middle button again. It should now have started recording. You can tell it is recording because the timer is running.

- When you are certain that it is recording, state the time, date and location to the recorder, for later use.

We can either do attended or unattended recordings.

- Unattended: Put the recorder in a discrete spot and leave. Long recordings are unattended.

- Attended: Stay throughout the whole recording, taking notes of the sounds you hear and the times that you heard them. This is good for comparing what the audio recorder can pick up, to what you can actually hear. Humans unconsciously take part in the cocktail effect, and we direct our hearing to one thing, whereas an audio recorder picks up everything.

3.5.2 Adobe Audition CC

Adobe Audition is a useful software for visualizing signals. It can be used to distort audio and create soundtracks, like for example in movies. For audio signal processing, we can use the functions that visualize the power of the audio file in dB and the spectrogram. In order to view the spectrogram, click the 'View' tab and click 'Show Spectral Frequency Display'.

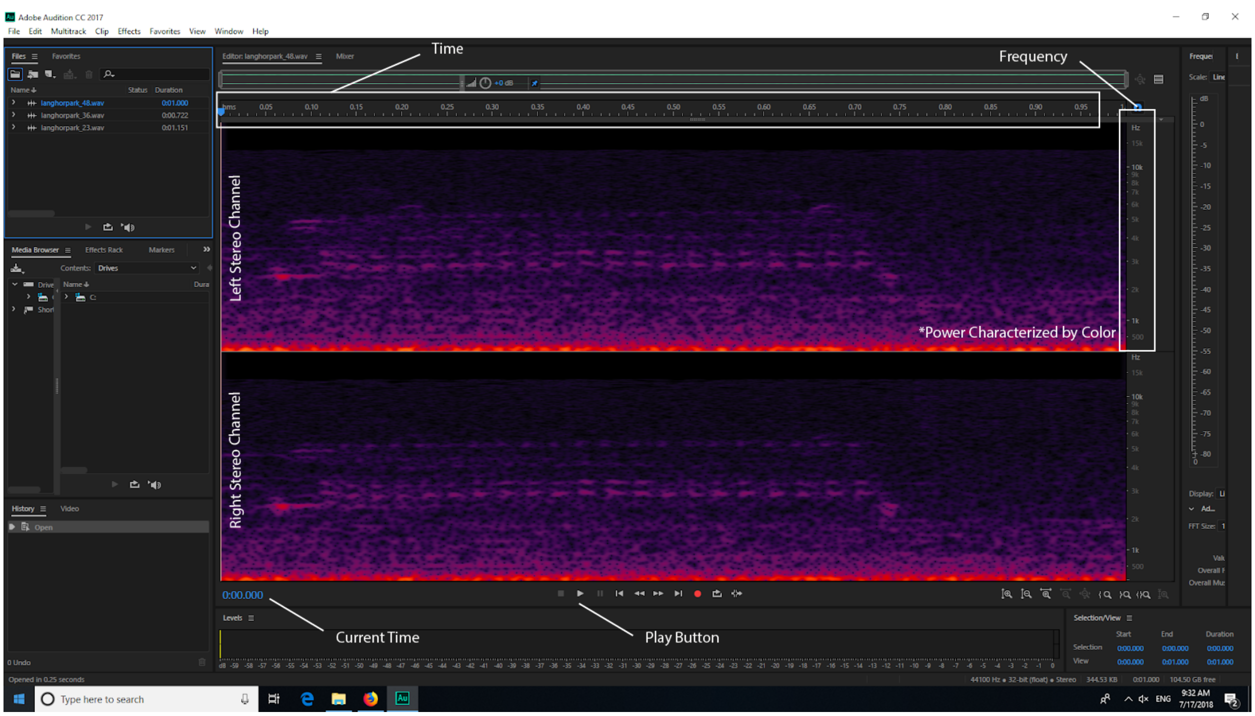


Figure 9: Typical Spectrogram

3.6 Sources for Red-winged Blackbird Recordings Online

– *Cornell Lab of Ornithology* - the Cornell Lab of Ornithology is a non-profit organization and part of Cornell University. The lab collects audio recordings and video recordings and will allow you to use their media if you create an account for their lab and request a license. [2]

– *Xeno-Canto* - Xeno-Canto is an open site in which bird enthusiasts from around the world upload their recordings of bird sounds. [35]

– *Acoustic Atlas* - Acoustic Atlas is a database curated by the Montana State University and contains natural sounds of Montana and the surrounding regions. [34]

3.7 Information about Red-winged Blackbirds Found at the end of the Program

On August 7th, three days before the end of the program, I found a couple of sources that focus on the same subject that I had been studying throughout my 10 weeks. Here, I will provide brief summaries of their contents and later on relate their findings to my own.

Stereotypy of Some Parameters of Red-winged Blackbird Songs

In this 1980 paper, the authors study how the different parts of a Red-winged Blackbird song for a particular male vary during the span of a one breeding season. One of the parameters used in the study is described as trill modulation rate and is the same characteristic that I have been studying these past 10 weeks. [3] *The Contribution of Temporal Song Cues to Species Recognition in the Red-winged Blackbird*

This paper describes to what degrees the pulse recognition rate (PRR), akin to the trill modulation rate, as well as sound structure of pulses and the duration of the trill are necessary for a Red-winged Blackbird to believe that it hears another Red-winged Blackbird.[11]

4 Methods and Results

Note: All graphs, tables and programs can be found at https://github.com/AllisonBusa/MSU_ECE_REU_RedwingedBlackbird

4.1 Ideation and First Analysis

In this section I describe how I can to my project goal and the initial testing that I did to discover the features of my test signals, songs of Red-winged Blackbirds. I include initial spectrograms and graphs depicting changes in power as well as my initial ideas for creating a program.

For this research program, I was able to choose what my research goal would be. In the first week of the program, I studied different articles about audio signal processing, bioacoustics and studies that merged the two areas. I was interested to see what more there was to learn about natural sound and how Electrical and Computer Engineering could be used to attain this goal. In the end, I ended up picking a project that semi-automatically identifies Red-winged Blackbirds in audio files. I was curious to see what distinguished a Red-winged Blackbird song as a signal and how that could be used to achieve other goals, like finding the time and place that a Red-winged Blackbird was at an area, or finding the population density of Red-winged Blackbirds in an area.

The first part of my project involved conducting recordings of Red-winged Blackbirds. I obtained my first recording of Red-winged Blackbird songs at Langhor Park in Bozeman, MT, where I conducted a 15 minute attended recording with the Edirol. Afterwards, I uploaded the file to Audition and followed along with the audio as I looked at the spectrogram. I saw that a Red-winged Blackbird song has a very distinct pattern that first looks mesh-like and is followed by arches that span for about half a second and for a few different frequencies. I counted 56 instances of Red-winged Blackbird songs and extracted all these times to new audio files.

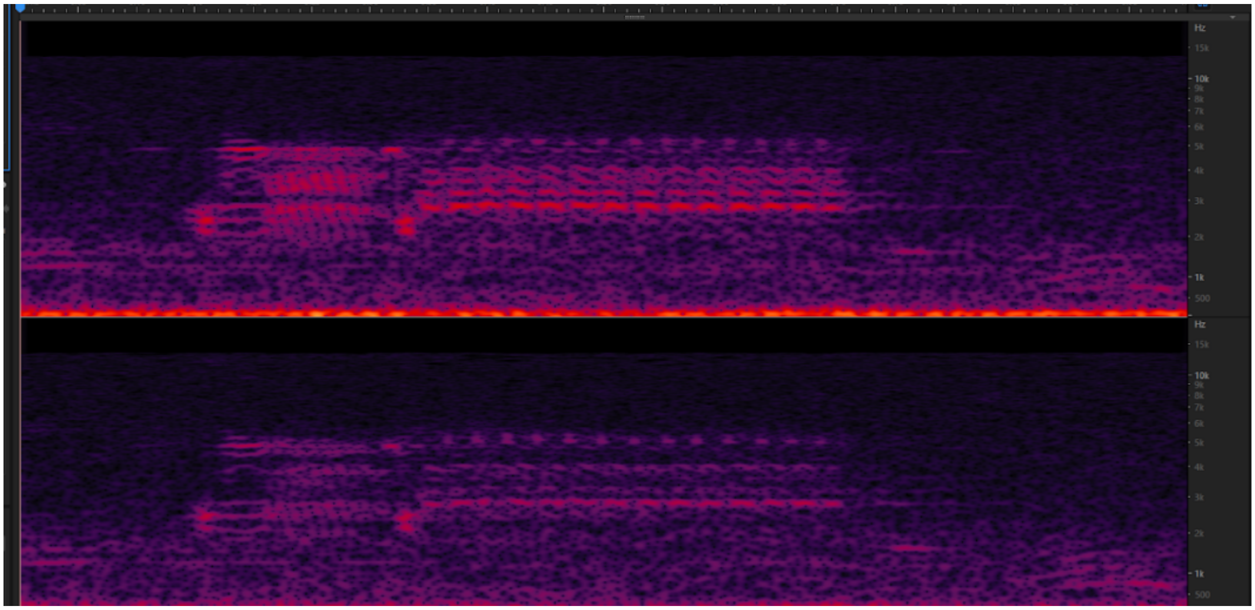


Figure 10: Spectrogram from Langhor Park Recording

My next step was to analyze the 56 new Red-winged Blackbird files I had. I also decided to analyze along with it a similar bird song and a different-sounding bird song which I recorded at Cherry River in Bozeman, MT.

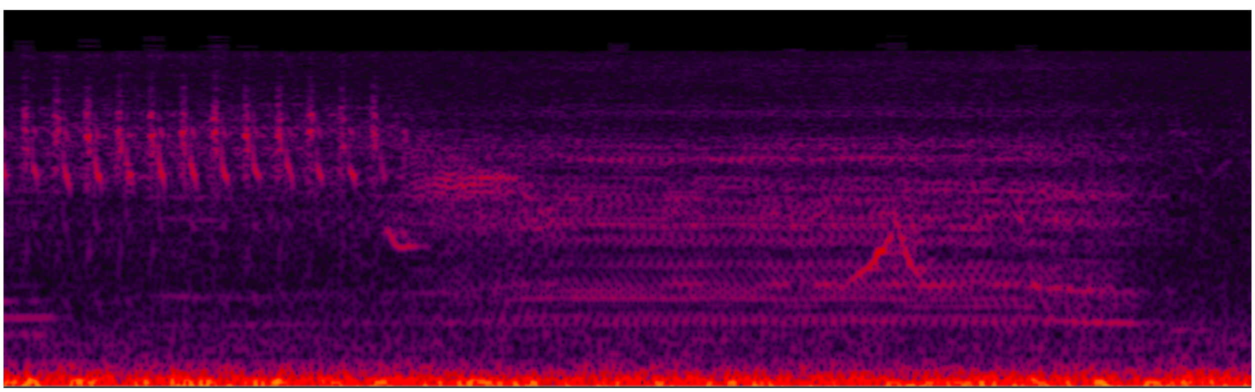


Figure 11: Spectrogram from Similar Bird

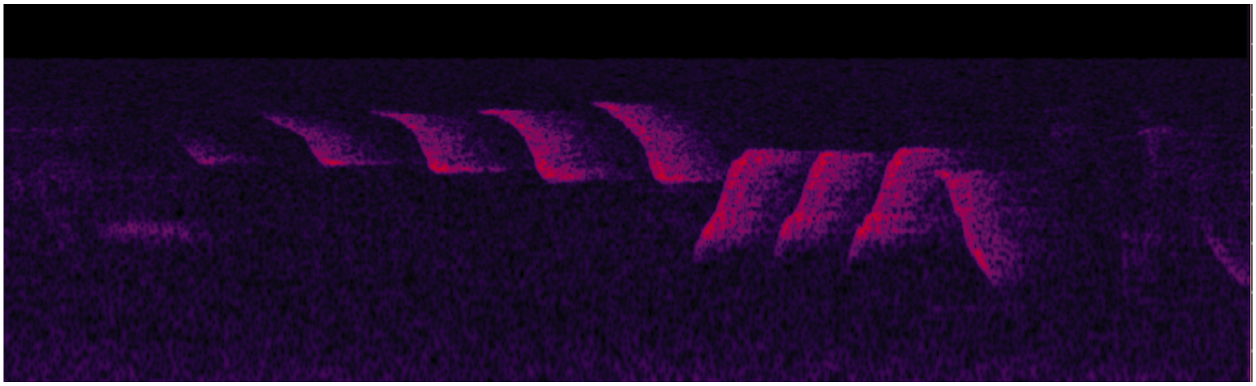


Figure 12: Spectrogram from Different Bird

I also found a one second segment of background noise from the Langhor Park file and convert that to an audio file to use. I wanted to see if there was some way that the arch pattern could be extracted further in a way that I could use and if there was any other nuances to the song which I could find by conducting tests on the songs and visualizing them in different ways. In MATLAB, I created graphs that showed the maximum pressure over time for certain frequencies, the power levels of all the frequencies present for equally spaced instances in the recording, and the total maximum power over time.

4.1.1 Graph 1: Power of 6 different frequencies over time

There seemed to be either a spike or a dense, uniform region at the point where the bird song starts. It seemed to be random whether it is a spike or a uniform region. Moreover, at which frequency this spike was most prominent and if it occurred at all was also variable. The oscillation was distinct.

In the two later discovered research papers, the spike is referred to as an introductory note and the oscillation as either the trill modulation rate or the pulse recognition rate.

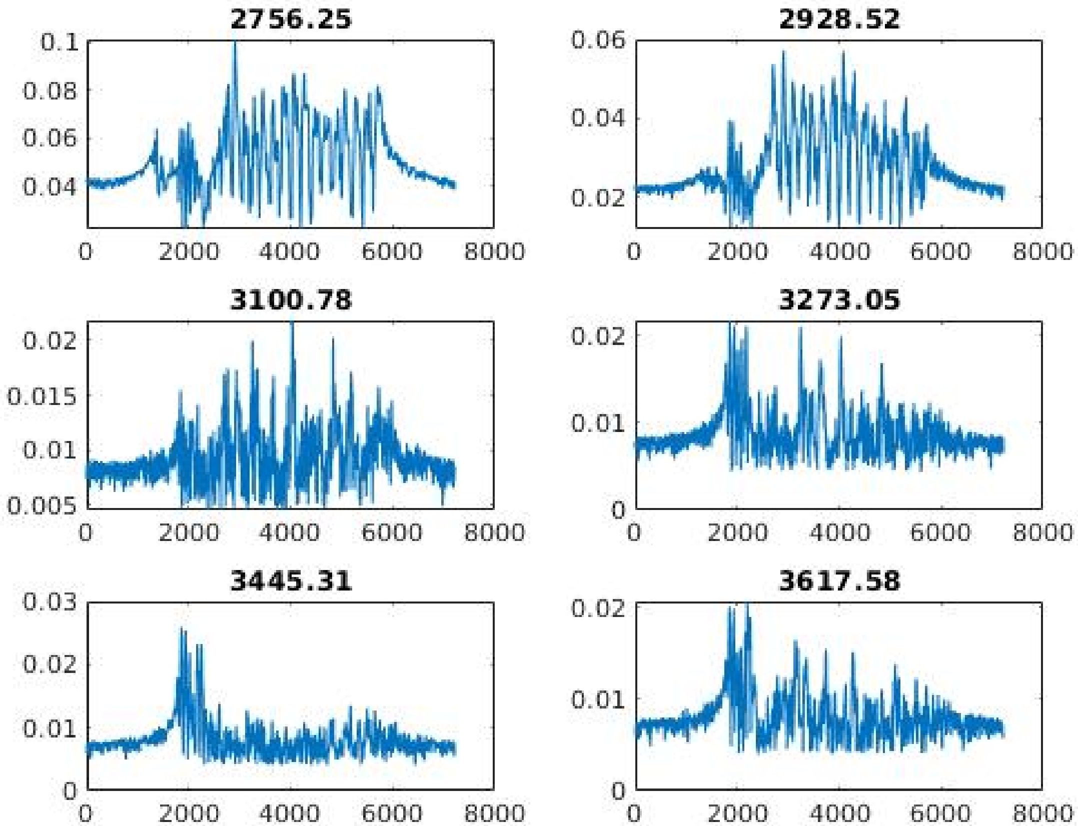


Figure 13: Powers of Different Frequencies over Time

4.1.2 Graph 2: The power levels of all the frequencies at one instant for 6 instances of time

There seemed to be a time at which all the power is contained under a certain threshold. This behavior did not apply for the background noise, similar birds or different birds.

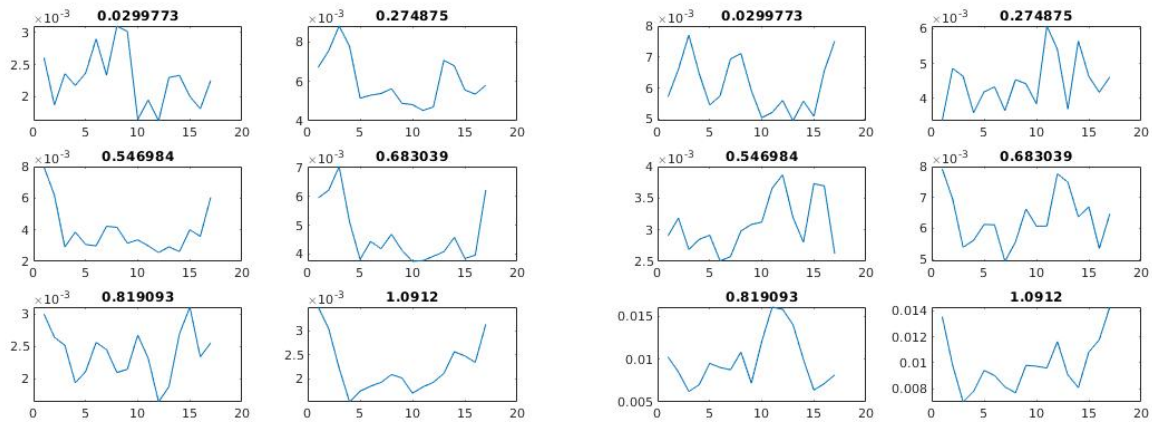


Figure 14: Langhor Park, file 22

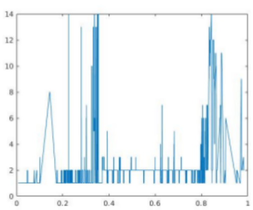
Figure 15: Cherry River, Similar Bird

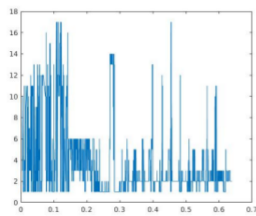
Figure 16: Powers Levels for Different Frequencies at 6 Instances of Time

Therefore, it could also be a distinguishing factor. However, again, the time at which this occurred was not consistent. I assumed it occurred at the time which the spectrogram shows arches but I had not continued on probing this distinction, because it did not seem as useful as the pattern from the graphs above.

4.1.3 Graph 3 : Maximum Power over Time

From the spectrogram, my initial reaction was to look for jumps in the maximum power that resemble the shape of an arch. First, I tried looking manually in the code for where the maximum power jumped from one frequency to the recorded frequency above it and then again back down. However, this proved to be inefficient and, instead, I plotted the powers over time of the individual frequencies.





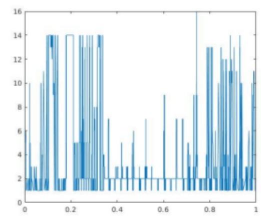
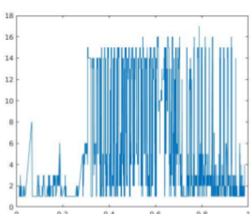
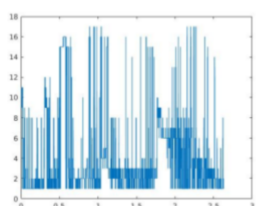


Figure 10: Langhor Park

Figure 11: Langhor Park

Figure 12: Langhor Park





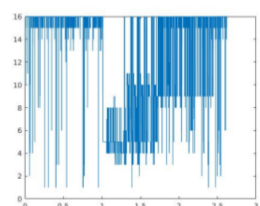


Figure 13: Langhor Park

Figure 14: Denver

Figure 15: Mexico

Figure 17: Powers Maximum through the whole Recording

From these graphs, I can see that the arch part of the song is characterized by periodic oscillations in the power level of one frequency. All of the graphs portray low power until the song, with the exception of the file from Mexico. I attribute this to loud background noise. I also realized that there is a series of horizontal lines that characterize the first part of the song. I looked at the values of the series of horizontal lines that characterize the first part of the song and found that the powers at these times aren't consistent, but oscillate by minuscule amounts (ie: 0.001 kPa).

4.1.4 Planning a Program

I went back to read articles about audio signal processing and bioacoustics to find examples of other similar programs to the one I was trying to create. I found one paper in which the author describes creating a program that also finds bird songs automatically in long audio files[17]. I created a system based of off his that would search for Red-winged Blackbird songs. The program would be two-fold: in the first part, I would rapidly convert the signal into the frequency domain and in the second, I would compare short segments of the audio file with a reference signal. In order to achieve the first part of the program I tried implementing down sampling and a manual Discrete Fourier Transform, calculating the instantaneous frequency with *instfreq()*, using *pspectrum()* to convert into the frequency domain, the Hilbert Huang transform *hht()*, and the Fourier

Synchrosqueezed Transform, *fsst()*. All these methods did not work effectively because they either took long, took up too much memory or, in the case of the Hilbert Huang Transform and Fourier Synchosqueezed Transform, were not understood properly. I decided from this that the Fast Fourier Transform, *fft()*, with a large window would be the most effective method. However, as I later discover, I did not know need to convert to the frequency domain, rendering the twofold program and the use of the *fft()* obsolete.

4.2 Initial Testing of Pattern Recognition

In this section I describe how I return to my initial results and creating my first program to detect different patterns and trends of Red-winged Blackbird songs. The concept of this program was not successful. I also describe how I arrive at my final method of filtering data.

In order to understand how to compare a reference signal to a potential Red-winged Blackbird signal, which I will refer to as a candidate signal, I returned to the results of the first graphs and began to test how I could reliably distinguish the pattern that I saw from the graphs before. I experimented with different ways of identifying the peaks in the signal and their characteristics. I will refer to these peaks later as bumps.

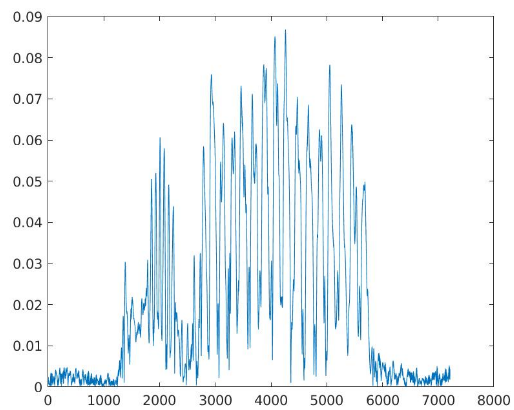


Figure 18: Sample Red-winged Blackbird Filtered and Smoothed Graph

4.2.1 Analyzing the peaks

In order to find the peaks themselves I used the function *findpeaks()* set with a 'MinPeakHeight' of $\frac{1}{3}$ of the maximum power and with a variety of different 'MinPeakDistances's. This method did not reliably find all the peaks due to varying sizes of the peaks and signals which were not clean having large spikes.

Nonetheless, for the peaks that were calculated correctly, I tested the frequency of these peaks using *diff(peakidx)* where peakidx was the index of the peaks calculated from *findpeaks()*. I tested to find any correlation using *mean()* and *std()* on the result from *diff(peakidx)* , but none could be recognizable.

4.2.2 Initial Program

After these tests I decided to create a program that would identify multiple aspects of a reference signal, so that variations in maximum peaks and candidate signals that overlapped with other noise would be accounted for. The program would identify multiple aspects of a reference signal, create a maximum number of points based on the number of aspects and test all of these aspects on a candidate signal, giving it a point if it was close to the value of the aspect of the reference signal. I implemented this program using these characteristics:

- 5 most prevalent frequencies
- 5 frequencies with the highest powers

- number of peaks that overcome *isoutlier()*'s upper threshold
- If there's a peak in the first 1/3 of the recording that exceeds the mean of the power of the first 30 samples by two.This tested if there was a significant increase in power.
- Using the same method, testing for a drop in power at the end of the recording.
- Duration from the first and last peak (above two points)
- If the oscillations between the first and the last peak are greater than outside those time limits

On Red-winged Blackbird songs, the success rate was $\frac{7}{18}$ and on non-Blackbird songs it returned "not a match" for $\frac{4}{5}$.

This poor success rate encouraged me to return to the spectrogram and brainstorm new ways to detect the signal. From the spectrogram, I knew that there was a prominent power that oscillated up and down, like an arch. I manually looked through the data to see at which frequencies this oscillation occurred. When I thought that I found it, I took the difference between the values to find the change in power. Then, I created a logical matrix that shows increasing power as 1 and decreasing as 0. From this matrix I did see an arch pattern, but it isn't consistent and I could not compare the whole logical matrix to another because the intervals between arches were different. I could not find a way to proceed after that point.

2778	2779	2780	2781	2782	2783	2784	2785	2786	2787	2788	2789	2790	279
0	1	0	1	1	0	1	1	0	1	0	1	1	0
1	0	1	0	0	1	0	1	0	0	1	0	0	1
0	1	0	1	0	0	1	0	1	0	0	1	0	0

Figure 19: Table Showing Increasing and Decreasing Power of Three Frequencies

Therefore, I returned to the initial analysis of the songs and decided to focus on extracting only the right range of frequencies. I filtered the whole frequency spectrum of the signal with $\frac{1}{3}$ octave filters and graphed the average power level of each band. There seemed to be an increase in power for the right frequency band (around 2500-3000 Hz), but it was not an outlier so I could not extract it with *isoutlier()* or *min()* and it was not always prominent.

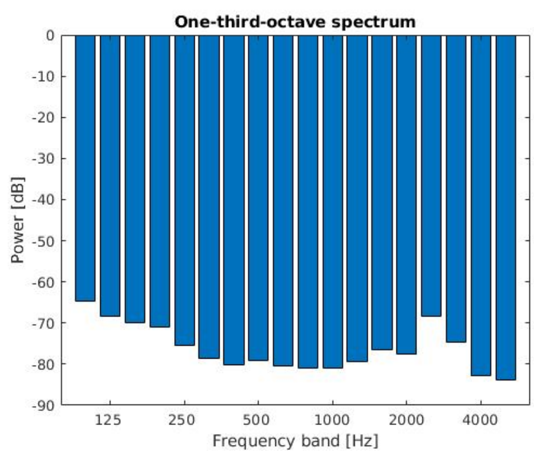


Figure 20: Power Levels of $\frac{1}{3}$ Octave Band Filters

In order to get the pattern that I saw from graphing the maximum power for a certain frequency for a larger range, I decided to filter all the original pressure data except for the data in the frequency range of 3000-3500 Hz. I accomplished this using the designoctfilter from a MATLAB file exchange program. [9]

I then attempted again to find the part of the data in which the signal is present. I graphed the *changepts()* of the data and it matched 6 songs correctly and 7 incorrectly.

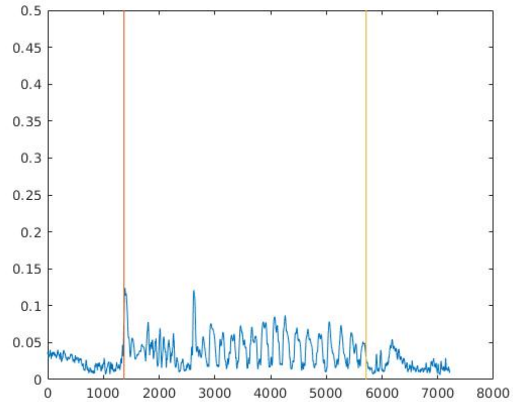


Figure 21: Identified Start and Stop Times Using *changepts()*

4.3 Experimenting with Different Methods of Analyzing Filtered Data

In this section I describe how I experimented with different analytical tools to detect the power modulation depicted in the graphs from my initial testing.

Since I could reproducibly create the right pattern with a range of different frequencies, and thus counteracting slight variations of clarity between frequencies, I could focus again on detecting a candidate in the segment of data.

4.3.1 Euclidean Distance

If you take the Euclidean distance of two signals at each time interval, the segment that has the smallest Euclidean distances indicates that the patterns at those parts are similar. I tested comparing the Euclidean distances first using *findsignal()*. If I used a reference signal, it would find all the signals correctly, but because it did not have a threshold of what the minimum Euclidean distance can be, it will find a signal in every time segment that you give it, Red-winged Blackbird or noise. I also tried using a 37 Hz sine wave that I created using the sine wave system creator from the Digital Signal Processing toolbox. With this method, I was not finding the signal or only finding a very small portion of it. Using the Dynamic Time-Warp parameter and all the normalization parameters produced similar results. I then tried creating my own functions which would find the Euclidean distance of all the points. For this I wanted to implement *pdist()*, but the memory would become filled or the process would take too long to run.

4.3.2 Magnitude Squared Coherence

I tried implementing MATLAB's mscohere() function on a reference signal and a candidate signal.

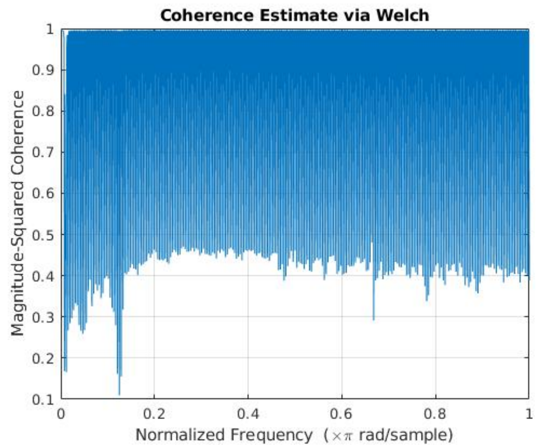


Figure 22: Magnitude-Squared Coherence of Filtered Signal

I left this idea when I decided on using frequency modulation.

4.3.3 Cross Power Spectral Density

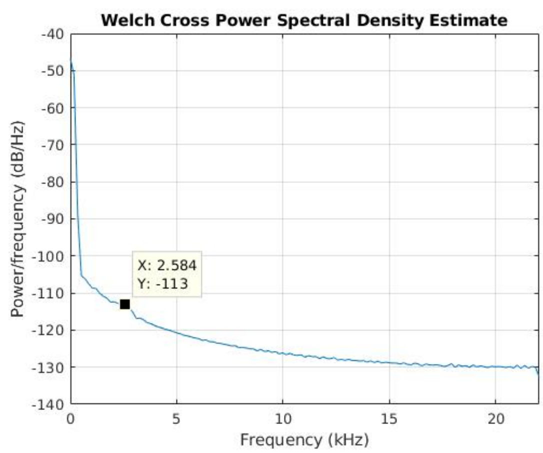


Figure 23: Cross Spectral Power Density Estimate of Filtered Signal

I had trouble interpreting this one too, but it seemed to indicate where the frequency modulation occurred, but not what it was. Cross Correlation I tried running *xcorr()* with the data as inputs.

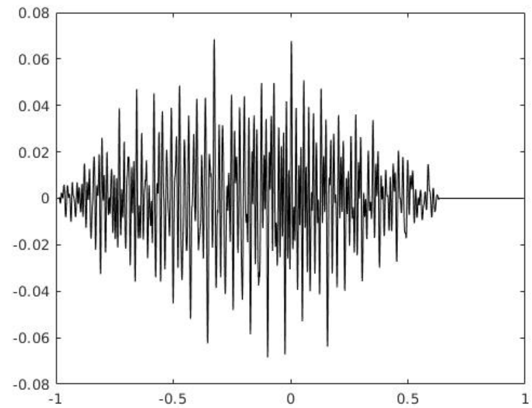
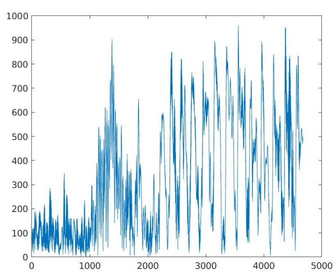
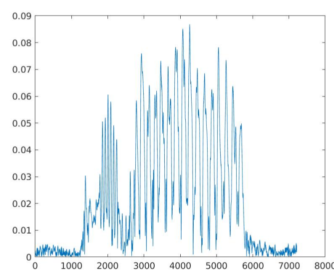


Figure 24: Example of Output of *xcorr()* of Two Signals

I tried running *xcorr()* on the maximum power of two signals at frequencies of 2576Hz and 3270Hz.





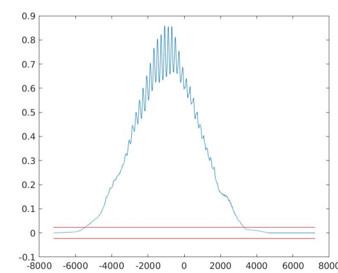
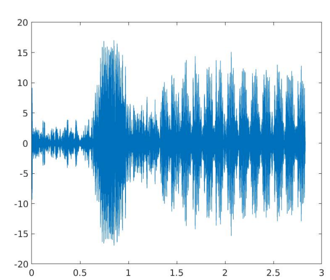


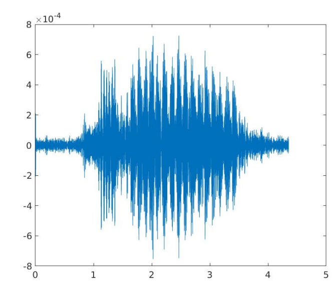
Figure 25: Langhor Park File # 22, Frequency of 3270 Hz

Figure 26: Langhor Park File # 34, Frequency of 3270 Hz

Figure 27: Cross Correlation of the Two Signals

Then, I tried running the function *xcorr()* on the data filtered with an octave filter





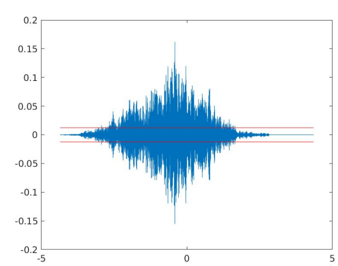


Figure 28: Langhor Park File # 22, Octave Filter with Center Frequency 3250 Hz

Figure 29: Langhor Park File # 34, Octave Filter with Center Frequency 3250 Hz

Figure 30: Cross Correlation of the Two Signals

4.3.4 Audio Fingerprinting

My first attempt of using audio fingerprinting was to use audfprint with Python, using Dan Ellis's Github audfprint program.[12] I created a database of 9 Red-winged Blackbird song as a reference and ran the program to match the whole Langhor Park recording, the recording of noise, a recording of a Red-winged Blackbird from Denver, downloaded from Xeno-Canto, and the recordings of the similar and different birds from Cherry River. All of the results were correct except for the the Denver recording.

Then I read An *Industrial-Strength Audio Search Algorithm* by Avery Li-Chun Wang[31] and Cheng Yang's, "MACS: Music Audio Characteristic Sequence Indexing For Similarity Retrieval"[37]. Based on Avery Li-Chun Wang's paper, I created a program that identifies points in the spectrogram.

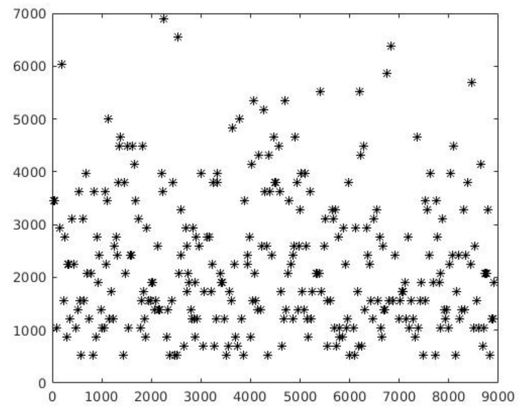


Figure 31: "Constellation Map" based on Shazam Music Recognition Program

I did not go as far as to create a program that uses combinatorial hashing to identify and store data about the distances of the signals, as is used in Shazam, because I could not find an abundance of sources describing how to implement it, it seemed out of the scope of the time frame, and frequency modulation, which I was experimenting with at the same time, seemed more promising.

4.3.5 Conclusion

In the end, I decided to pursue measuring the frequency modulation using the peaks as from before. I decided to do this because the pattern of the oscillation was as reliable as the spectrogram showing arches and because it seemed like a simpler way to approach the problem then some of the methods that I have listed above.

4.4 Calculating the Peaks and Patterns Manually

In this section I describe how I collect reliable results to compare my program outputs to.

4.4.1 Calculating Peaks

If I was going to focus on the modulation of the frequency, I decided that I needed to more precisely know what I was looking for. Therefore, I measured the frequency of the modulation, again using the *findpeaks()* method from earlier. When this method seemed to be unreliable, I counted the number of peaks manually for 22 Red-winged Blackbird files. Then I estimated the sample range for which the oscillations occurred by looking at the graph and converting the range to seconds, which I later found out that I did incorrectly. However, the method was consistent, so I ended up with values which were not correct numerically but were correct in their ratio with each other. From this, I found all of the Red-winged Blackbird files to be in between 0.33 and 0.54.

Then, I improved two aspects of this method: finding the start time and end time of the signal and calculating the time interval at which the oscillation occurred. The first I did by graphing the patterns and using MATLAB's data cursor to point to the start and end times. I recorded the start and end times into my research journal, and found the change in samples by subtracting the start time from the end time. My new method for calculating the time interval was to calculate a new sampling rate and then dividing the number of samples by that number.

4.4.2 Calculating New Sampling Rate

At the time that I was testing this new method of finding the change in time of a signal, I had begun to put the signal through an RMS envelope. Therefore, I calculated a new sampling rate to align with the smaller number of samples:

$$\frac{length(originalData)}{samplingFrequency} = totalSeconds \tag{1}$$

$$totalSeconds = \frac{length(envelope)}{newSamplingFrequency} \tag{2}$$

$$newSamplingFrequency = \frac{length(envelope)}{totalSeconds} \tag{3}$$

4.4.3 Calculating Cycles per Second

Example Calculation:

$$newSamplingFrequency = 4.4767e4\frac{samples}{sec} \tag{4}$$

$$2260\frac{samples}{newSamplingFrequency} = 0.5048sec \tag{5}$$

$$\frac{42cycles}{0.5048sec} = 83.1949\frac{cycles}{sec} \tag{6}$$

4.4.4 Calculation Results

Out of 11 Langhor park records, all except 1 were between [38.38 40.17]. The one outlier had a frequency of 64.3016.

Out of 8 foreign Blackbird songs, 2 were in the range mentioned above and the range for these songs was [25.62 61.28].

4.4.5 Measuring Patterns

I saw that the difference in height between the first bump (the mesh-like part of the signal in the spectrogram) and the second set of bumps (the arches in the spectrogram) caused problems so I tried creating a program that was supposed to measure the heights of the two bumps. This program turned out to be inconclusive, so I created 59 filtered octave graphs of Red-winged Blackbird songs (6 from outside sources) and manually kept track of characteristics like : 'Did the second bump have constant spread out oscillations?', 'Is there an initial bump?', 'Is the initial bump the highest part of the graph?', 'Is the pattern recognizable? Or Somewhat recognizable? Not at all?'. 3 of the external Blackbird songs were not mating songs, so I had to remove them. I went back and looked at the Audition spectrograms of the unrecognizable songs and found that they were either very low or there were other bird songs overlapping it- showing that getting clear audio songs would be a barrier to finding the correct frequency modulation. Later on, I found out that of the Red-winged Blackbird signals that had less distinguishable patterns, changing the center frequency and convolution value would produce graphs that had clearly distinguishable patterns as well. I also acknowledged that the first bump, which I described as a mesh earlier on, did not seem to always appear. I confirmed this when I looked at graphs of songs from outside sources.

These manual calculations confirmed for me that finding this pattern was indeed an effective method.

Question	Amount answered positive (out of 56)
'Did the second bump have constant spread out oscillations?'	40
'Is there an initial bump?'	40
'Is the initial bump the highest part of the graph?'	20
'Is the pattern recognizable?'	31
' Somewhat recognizable?'	11
'Not at all?'	14

Table 1: Results from Visually Analyzing Filtered Graphs

4.5 Creating a System that Detects the Frequency of the Modulation

In this section I describe my initial pivot that led to my final program. I narrate my discovery of using the *periodogram()* function and attempting to create a program that uses *periodogram()* to find the frequencies of data at different windows in a long audio file.

My first program scanned a file of a Red-winged Blackbird song that was filtered and smoothed, and used *periodogram()* to calculate the frequency of the oscillations.

4.5.1 Implementation in a Long Audio File

The program scanned a long audio file, and returned the intervals at which there were frequencies in two ranges over a set time period. I chose the initial ranges and time periods I set as parameters from running the program on a few Red-winged Blackbird files and making my own estimates. The testing for the intervals happened with a for loop that would find each range of the whole vector of filtered and smoothed data that was specified with the index of the loop. It would then calculate the frequency of the data using the *periodogram()* function.

4.5.2 Confirming that Red-winged Blackbirds Produce a song at a Reproducible Frequency of Modulation

Although I proved that the pattern was constant, I needed to prove that the frequency of the oscillation, which I thought defined the pattern, was also constant.

To do this, I collected 18 Red-winged Blackbird songs from Xeno-Canto and the AcousticAtlas, and 8 random bird songs from Xeno-Canto. I created filtered graphs for all of these and confirmed that Red-winged Blackbirds do all have a distinct pattern and that the initial bump is for most a spike with no oscillation. I found that 9 Blackbird songs were also descending, which interferes with the threshold reliability.

I tried searching for more reliable functions that existed that measured periodicity and found *damp()*, which returns the natural frequency of a dataset. I ran this on different intervals, such as the whole file, only the segment when the song oscillates around a threshold, the interval of the second bump and the indices above the threshold. These results were not as reliable as I would have liked– what's more, when I ran the function today on data which I believed to be of a similar length, the function would run for over 5 minutes without producing a result.

I found a function that can replicate an oscillation using a Hilbert Transform, but I could not find a way to implement it in my program.[27]

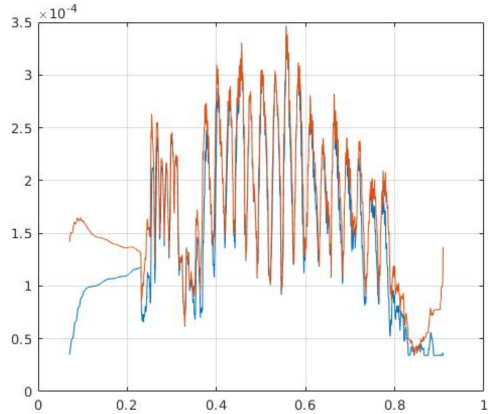


Figure 32: Result from Hilbert Transform

At this point, I decided that I had to answer the question "Can you reproducibly show that Red-winged Blackbirds produce a song with a distinct amplitude modulation frequency?"

4.6 Creating an On-Off System

In this section I describe implementing algorithms that set different intervals of the data to either 1 or 0 in order to make the *periodogram()* more effective. I then implement it to find the frequencies in a long audio file.

Based on the advice of my mentor, Rob Maher, I created a program used to detect peaks that would turn all the pressure data values to 0 if they were below a threshold and all those above a threshold to 1. I then put this binary matrix through the *periodogram()* function and take the frequency at which the maximum power occured to be the frequency of the oscillation. I had also tried using *xcorr()* as well, but the *periodogram()* function was more intuitive. The function of using an on-off system as opposed to using *periodogram()* right away on the filtered and smoothed pressure data was that I was interested in only the oscillation of the pattern and the on-off system exaggerates it.

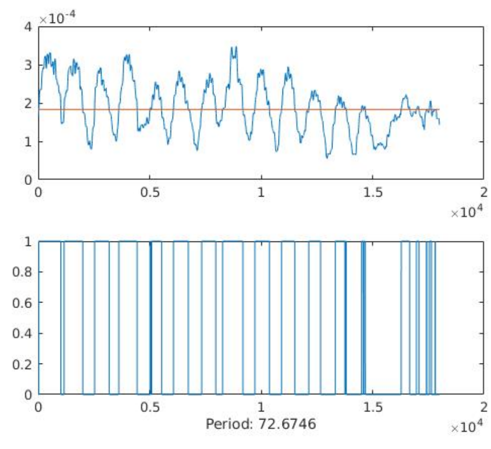


Figure 33: Visualizing the Exaggeration of the On-Off System

4.6.1 A Reproducible Program

I set out to make a program that would compare 7 random bird files with 10 Blackbird songs I found online and 14 of my own Blackbird song files. It would do so utilizing the threshold program and graphs of the filtered and smoothed data.

With the graphs of the filtered and smoothed data, I used the MATLAB data cursor to pick the start time, end time and thresholds. I set the time intervals and thresholds individually for each graph so that the result would be reliable. To accelerate the process, I created a list of time intervals and thresholds from the graphs I already had of these songs and found out that when I ran them through the program, the intervals and amplitudes of the oscillations were different (amplitudes were different by an order of 2).

I tested the program on an initial set of files from Langhor Park, but I was not receiving consistent results. I suspected that these issues stemmed from the fact that I normalized some of my graphs and accidently removed the first 20,000 samples of the data automatically by keeping a function I left over from my last program that searched for signals in a long audio file, in order to account for my speech. I remade new graphs and calculated the time intervals and thresholds for each. At this point, I saw from calculating thresholds and time intervals that both the foreign and Bozeman Blackbird songs were showing a constant set of frequencies between [131 137]. Then, I compiled the program in a for loop that would take cell arrays of the file names, corresponding thresholds and corresponding time intervals, and save all the graphs with the frequencies to a folder and create mat file containing all the names and frequencies.

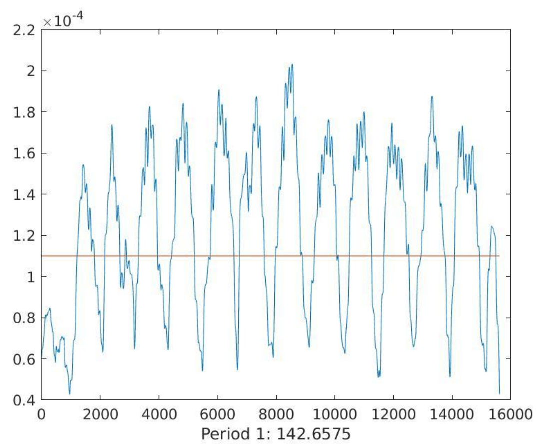


Figure 34: Output of the Improved Threshold System

In order to be able to use a for loop, I had to convert all the files to one file type. Not realizing that this would mess up the data, I converted all the files to mp3 and when I ran the program, the amplitudes were off by a magnitude of 2 again and the time intervals were shifted.

Instead of going through the whole process of setting the thresholds and time intervals beforehand again, I changed my program to make it user interactive. You run the program, and the first graph shows up with a GUI created with *inputdlg()* that asks you to input the threshold, start time and end time. You can use the tools on the MATLAB graph to select coordinates that seem to you to show oscillations (this is what I was doing before) and you can select start and stop times a little out of range, because the program only looks at the interval of time between the first point above the threshold and the last. When you choose your points and click enter, the resultant graph will show up, but the graph will also get stored to your folder and the value will be added to a mat file.

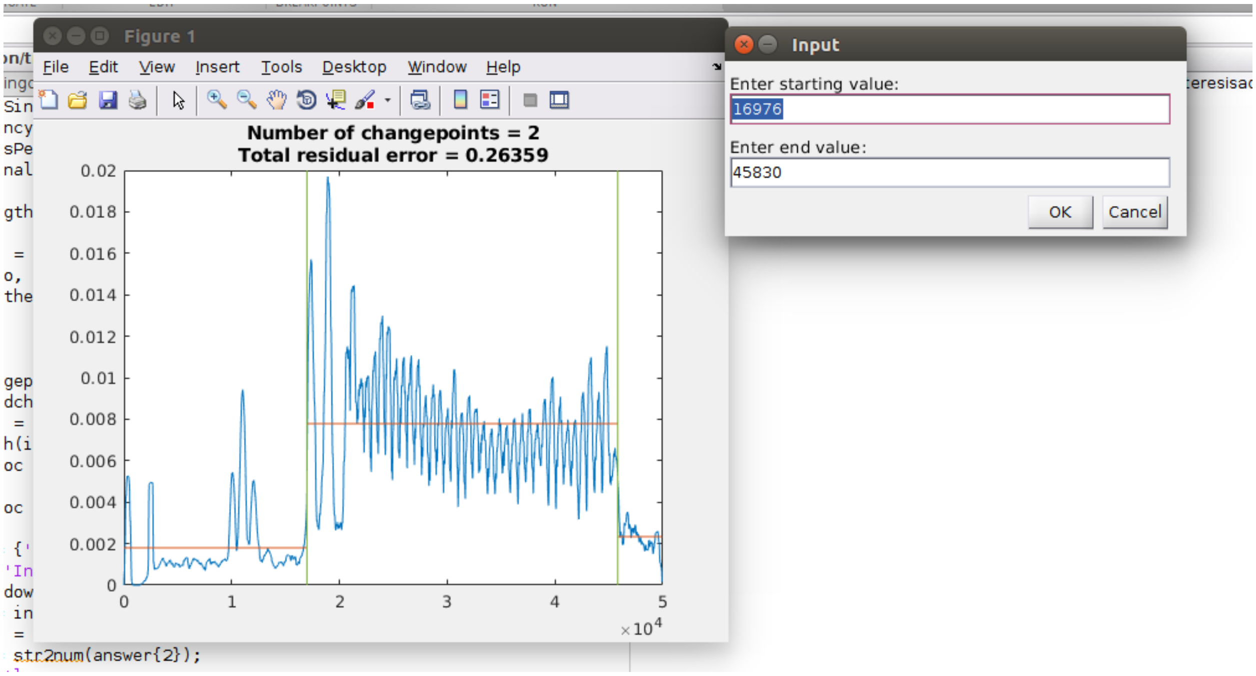


Figure 35: My first GUI implementation

I ran this in a for loop for 30 files . My results are on the next page.

Random Bird Calls	
'alectorisrufa.wav'	71.3287
'collaredforestfallcon.wav'	73.2421875
'Conopophagalineataantagonistic.wav'	269.1650391
'europeangoldfinch.wav'	33.64562988
'greatercrestedterns.wav'	68.63708496
'greattit.wav'	1289.0625
'spotcrownedwoodpecker.wav'	585.9375
Foreign Bird Calls	
'AcousticAtlasRWBB.wav'	559.8632813
'AtlasRWBB2.wav'	137.2741699
'DenverRWBBshort.wav'	131.8908691
'georgia.wav'	263.7817383
'mexicoRWBBshort.wav'	703.125
'michigan.wav'	146.484375
'northcarolina.wav'	263.7817383
'sanfranRWBB.wav'	366.0644531
'southcaliRWBB.wav'	145.3491211
'tennessee.wav'	131.8909
'AcousticAtlasYellowstone.wav'	71.77734375
Bozeman Bird Calls	
'langhor3.wav'	191.40625
'langhorpark_11.wav'	137.2741699
'langhorpark_014.wav'	131.8908691
'langhorpark_016noise.wav'	131.8908691
'langhorpark_22nice.wav'	1
'langhorpark_25.wav'	142.6574707
'langhorpark_26.wav'	269.1650391
'langhorpark_28.wav'	131.8908691
'langhorpark_32.wav'	131.8908691
'langhorpark_34.wav'	131.8908691
'langhorpark_37.wav'	145.3491211
'langhorpark_38.wav'	137.2741699

Figure 36: First Output of GUI system frequencies: outliers are highlighted in yellow

At this time, I knew some of the outliers were caused by my setting points incorrectly and I believed that a good part of the other outliers are caused by the graphs not being completely smooth yet and other points being used to calculate the frequency. For example, the song from Georgia :

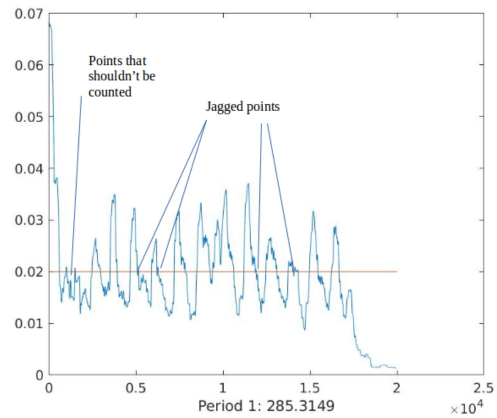


Figure 37: Portrayal of Flaws in First GUI System

4.6.2 Testing in a Long Audio File

I implemented the program I used before to find the frequency of segments of a long audio file, with the exception that at each iteration of the for loop, the data was converted to a binary vector using the threshold method. I had set the threshold to be 0.002, which seemed to be a common threshold from the threshold graphs created above.

On the first try of this program, it returned 500 potential signals in a 15 minute recording with 56 songs. Then I improved it to find the oscillations of the mesh, which were constant in the Langhor Park recordings, and take those time intervals, offset them by 2500 samples and look for the oscillations of the second bump. I also set the threshold to be the center of the data of each window. This new program said that it found 21 songs in the same 15 minute recording and out of those 21, 10 were right. After obtaining these results, I spent time debugging and trying to tweak the rules of my program to get better results.

The first test that I did was try to find a threshold that would adjust itself to every song. I tried implementing thresholds that measured 75% of the first peak, the upper limit of *isoutlier()* and the center of *isoutlier()*. I tried this on 15 graphs and compared it with the graphs that had 0.002kPa as a threshold.

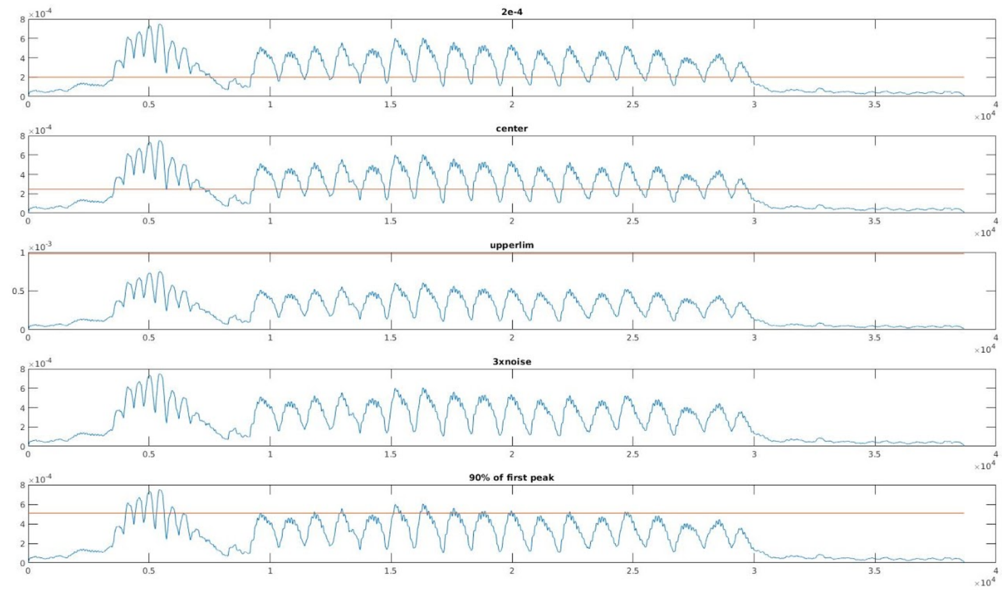


Figure 38: The Different Thresholds that I Experimented with Using

Next, I graphed the 11 false positives and found a lot of noise and a repeated signal that looks like this:

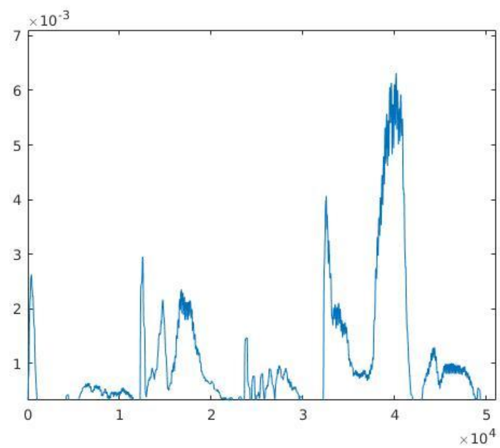


Figure 39: Example of False Positive : this is my voice

I tried tweaking small parts of the program to get more accurate results, such as: the length of the windows for the first and second bumps, the overlap length, widening the frequency ranges, changing the program to check for both bumps in one window and changing the thresholds in the program. None of these seemed to help, because they did not solve the real problem.

So, I ran the program with all its new parts on the original Red-winged Blackbird songs and, surprisingly, they were not being recognized anymore. Also, when I was checking the frequencies, I kept getting inconsistent results. This led me to the conclusion that I needed to fix my program on a more fundamental level.

4.6.3 First Implementation of Hysteresis

Rob Maher suggested that I use hysteresis to overcome the problems I was having as related above with the Georgia file example. The way that the first version of the system worked was that it would check if every point in the data was between two thresholds and it would revalue it to 0 if it was not and 1 if it was. I thought that the function of using hysteresis was to account for small variations in jagged data that would oscillation between 1 and 0. However this is not how hysteresis works, instead its mechanism is described in the *Background Information* of this paper.

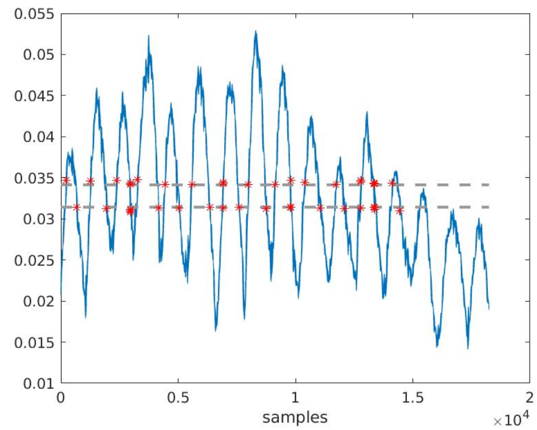


Figure 40: First Implementation of Hysteresis: Gray lines represent the threshold and the red dots represent the start and end points of 'on' sections

Using this function, the returned frequencies were between 76 -79 Hz, which was about twice what I calculated manually. Looking back this is entirely expectable, because the program, when it looks for all the points between two thresholds, will find two sections per peak. Therefore, it was as if there were two peaks for all the instances at which there should have been one.

4.6.4 Frequency Modulation Detection using Correct Hysteresis

All of the previous tests and advice from Rob Maher helped me come to a final program that would accurate detect the frequency of modulation.

Rob Maher had helped me see that I was interpreting hysteresis wrong in the previous program I wrote about above. It would find all the points between two thresholds, when in reality, I wanted it to find all the points between the times in which the data reaches the upper threshold and goes back down to the lower threshold.

I changed my program so that it would look for peaks this way instead and then tested it on sample signals.

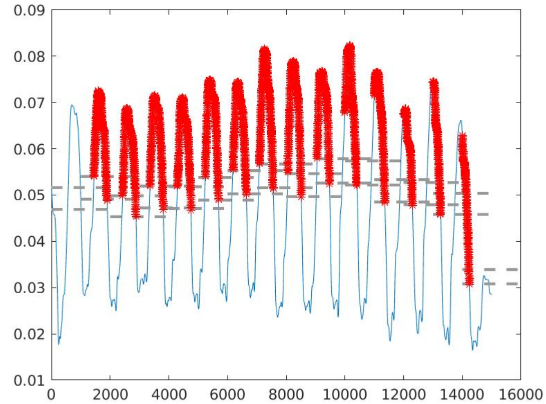


Figure 41: Final Implementation of Hysteresis: Gray lines represent the thresholds and the red dots represent the 'on' sections

First I tested this on the individual Langhor Park recordings whose frequency I calculated manually. My results were all the 36 - 45 Hz range, matching my manual calculations.

4.6.5 Other Results

I tested the accuracy of this program on 44 recordings, which would show a 'Match' if there was a segment with a modulation frequency in the range 32 Hz to 42 Hz and 'No Match' if there was not. The way I determined accuracy was by determining if the program produced a True result or a False result. True was defined as being representative of the spectrogram : showing 'No Match' for instances which did not show a Red-winged Blackbird and showing 'Match' for instances in which there was a Red-winged Blackbird. False was defined as being the opposite of True: showing 'Match' for instances in which the spectrogram showed no Red-winged Blackbird and showing 'No Match' when the spectrogram showed a Red-winged Blackbird.

Source of Blackbird Songs	Amount of Songs	Amount Detected	True of False
Sourdough (Blackbird) *see next section	26	22 (Match)	T
		6 (No Match)	F
Langhor Park (Blackbird)	11	10 (Match)	T
		1 (No Match)	F
Other Birds (not Blackbird)	7	7 (No Match)	T

Table 2: Results from Testing Accuracy of Hysteresis Program

$$\frac{39TrueMatches}{44TotalMatches} = 88.64\%$$

4.7 Additional Red-winged Blackbird Recordings Added to Database

In this section I describe different sources of audio files that I use.

4.7.1 Sourdough

In the section above, Red-winged Blackbird recordings from Sourdough are mentioned. These are files of Red-winged Blackbirds from a 24 hour recording conducted at Sourdough Trail in Bozeman, MT by Rob Maher, another REU students Justin Ortgies, and myself.

I used three 25 minute sections of the Sourdough (approx. 8 - 8:25 pm, 5-5:30 am,5:30-6am), there was one outstanding peculiarity about the songs. In the first recording from 8-8:30 pm, the songs seemed to only have one or two prominent bands of oscillations, which were lower than the Langhor Park prominent bands on the frequency scale.

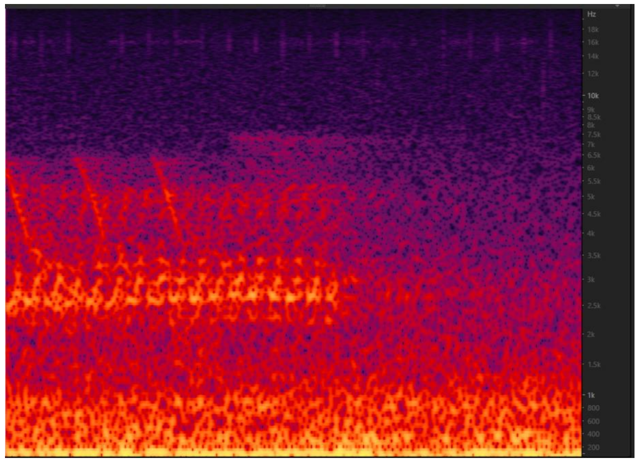


Figure 42: Subdued Sourdough Song

Then, starting at around 5:20 am, the Red-winged Blackbirds start singing in the frequency range that I expected them to.

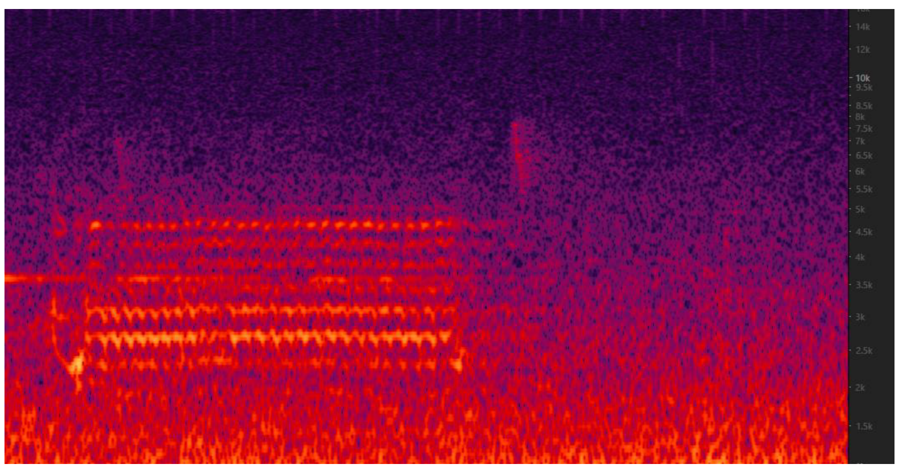


Figure 43: Expected Sourdough Song

In the 5:30 - 6am interval, 26 Red-winged Blackbird songs look how I expect them to and then after the 27th, they go back to being at a lower frequency and only one or two bands.

My initial hypothesis for this was that there were other birds singing at the frequency range that a Red-winged Blackbird would usually sing at and the Blackbirds adapted their song when other birds were around.

Although I was unsure of the cause, for the project, this discovery prompted me to experiment with having the program try multiple frequency ranges and multiple convolution settings for one recording.

According to the information that I have found from Temporal Cues and Species Recognition, the peak ranges for the song should be in the same frequency range for all the Red-winged Blackbirds in a region so that all the members can distinguish it[3]. Since the bands present in the first spectrogram and the most prominent (most brightly colored) bands of the second spectrogram hover around 2.5kHz, this behavior seems to be acceptable for the species.

4.7.2 Cornell Lab of Ornithology

I had applied for and gotten a license to use the Cornell Lab of Ornithology's Red-winged Blackbird audio recordings. Any further audio files mentioned that are not from Langhor Park, Sourdough Trail or the seven different bird songs I collected before, are from the Lab of Ornithology's database.

4.8 Implementing the System in a Long Audio File

In this section I describe implementing the final algorithm of the 'On-Off' system in the program that detects frequencies in long audio files, how I improve said program and the results of my tests of it.

In order to improve the program I had before for detecting segments of Red-winged Blackbirds in long audio recordings, I first improved the way that the data was parsed. I split the data up into sections of 20,000 samples before putting the sections through a for loop which checked for frequency. The sections had an overlap of 2,000 samples and they were created and implemented as such :

```
%% Reshaping Data
v = data; %reassigning data
cs = 20000; %row length
sh = 18000; % row length - overlap
A = v(bsxfun(@plus,(1:cs),(0:sh:length(v)-cs)'));
% Implementing in a for loop
for j= 1:size(A,1)
```

This reduced both the time of computation and the RAM.

4.8.1 Testing the Modulation Recognition Program in Long Audio Files

For the user, the way this program works is that she sets her parameters and hits 'Run'. The .mat file with all the frequency and other information becomes available and the graphs pop up and automatically get saved to a folder. This process, when using a wav file, takes about 24 seconds for an 18 minute wav file recording.

4.8.2 Test on file 189395

I tested this program on file 189395 from the Cornell Lab of Ornithology, it is a recording of mostly Red-winged Blackbirds from California. In the recording, there are male mating songs, female mating songs, flight songs, few instances of other birds and the recorder talking.

My program found 86 Red-winged Blackbird songs. Using the program's times at which the songs occurred, I followed the spectrogram to test the accuracy of the program. I found:

- 30 additional songs that were at the foreground. By this I mean that the song was substantially loud and did not sound like it came from a larger distance.
- 12 -18 False Positives. At first, when I listened to the recording by ear, it seemed that there were more false positives but when I looked at the spectrogram, the sound looked like a Red-winged Blackbird song.

$$\frac{86songs - NumberWrong}{86songs + 30AdditionalFound} = 85.84 - 85.89\%$$

Looking back at the recording, I believe this number is artificially high. Because most of the sounds in the recording were Blackbirds, there was not a lot of room for false errors. Moreover, I set my frequency bounds to be 32 to 50, eight more than I would have for a Langhor Park or Sourdough Recording.

4.8.3 Test on Recording from Acoustic Atlas

Next, I tested this program on an hour long recording from Utah, which I found on the Acoustic Atlas. This recording was filled with other bird sounds that overlapped with Red-winged Blackbird calls. I set the parameters to be the same as in the California file and received 421 potential candidates. Again, through listening to the file and looking at the Audition spectrogram, I saw that the results from this trial were very poor. When I went back to the results matrix, I saw that the number of bumps for many candidates was very low, and many had only seven bumps. By looking at many results from before, I knew that 7 was too low of a number for number of bumps and so I reinstated the number of bumps as a parameter to check for Red-winged Blackbird songs. Contrary to before, when I checked for bumps in a certain range, now I set the threshold for bumps to be greater than or equal to 10. Running the program again, the number of candidates decreased to 274.

I set the function a third time, with a center frequency of 3250, convolution size of 400, reference frequency of 37.681 Hz and variability of +/- 3, meaning that the frequencies of the candidates can only vary +/- 3 from the reference frequency. I also set the number of bumps to be greater than 10. Now, the program produced 184 candidates, which I checked by plotting and saving graphs of each candidate to a file and checking the filtered graphs manually. I tallied the graphs as either 'Yes' that I was sure it was a Red-winged Blackbird, 'Maybe' that I was not sure but it was possible, and 'No' that I was sure the graph did not show a Red-winged Blackbird.

Type	Tally
Yes	94
Maybe	31
No	59

Table 3: Results from Testing Accuracy on Long, Noisy Audio File

Assuming that half of the 'Maybe' candidates are right, then 94+15.5 / 184 = 59.5 % of the candidates are correct.

However, out of the 59 marked wrong, 19 of the wrong graphs had a constant shape to them .

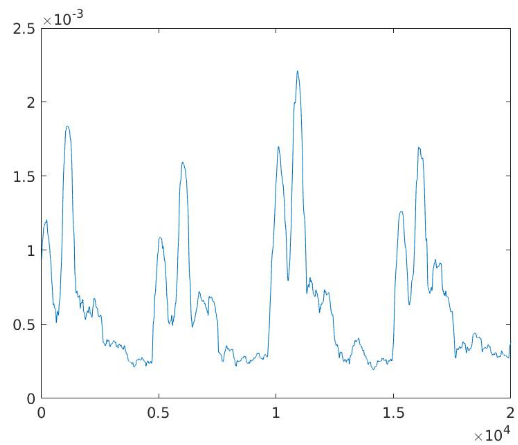


Figure 44: Example of Repeated False Positive Pattern

4.9 Frequency Ranges for Red-winged Blackbirds from Different Regions

In this section I describe how I use my programs to discover nuances of Red-Winged Blackbird songs in general and of those in different regions of the United States.

I created multiple databases of Red-winged Blackbirds from specific states, like California, New York and Massachusetts. To create this database, I downloaded files from the Cornell Lab of Ornithology that contained songs from New York, California, Maryland, Massachusetts, Ontario, Oregon, Manitoba, Indiana and Cuba. Then, I extracted clean instances of Red-winged Blackbird songs. Here, I define clean to be of prominent power with respect to its surroundings and with no other bird calls running through the song in the spectrogram.

Then, using the GUI I created and snippets of other code, I created a program that would loop through all the files in a database, present the GUI to the user, find the frequency of the recording and store the name of the file, the frequency, the number of bumps, the center frequency and the convolution size as a row in a matrix. See GitHub reference for full output.

When I isolated the file names and the frequencies, I saw two patterns. Either all the songs from one Lab of Ornithology recording were the same, differing by only 2 Hz at most, or there were groupings of frequencies that would differ by 2 Hz. This I can only deduce to the fact that when multiple Red-winged Blackbirds sing their mating song in a region, they sing at different frequencies from each other to differentiate themselves. Furthermore, all the groupings of frequencies stay within the frequency range of 30 to 85, with occasionally some calls in the 20 Hz and 90 Hz range. According to the Temporal Cues and Species Recognition paper, the range for Red-winged Blackbirds from Ithaca is 40 -107 Hz and the range for California Red-winged Blackbirds is 20 to 200 Hz.[3] For all the following tables, I have grouped all Red-winged Blackbird songs from one recording together with one color and all the similar frequencies in a recording with another color.

4.9.1 Test on Oregon Files

Filename	Frequency	Number of Bumps	Center Frequency	Convolution Size
'/home/allison/Downloads/oregonclean/106833_44k_01.wav'	34.99145508	28	3159.631147541	200
'/home/allison/Downloads/oregonclean/106833_44k_02.wav'	34.99145508	11	3004.0983606557	440
'/home/allison/Downloads/oregonclean/11955_44k_01.wav'	45.75805664	16	2400	271
'/home/allison/Downloads/oregonclean/11955_44k_02.wav'	45.75805664	19	2400	400
'/home/allison/Downloads/oregonclean/11955_44k_03.wav'	45.75805664	21	2400	249
'/home/allison/Downloads/oregonclean/11962_44k_01.wav'	32.29980469	8	3193.4426229508	448
'/home/allison/Downloads/oregonclean/11962_44k_02.wav'	32.29980469	8	3250	400
'/home/allison/Downloads/oregonclean/11962_44k_03.wav'	32.29980469	7	3500	290
'/home/allison/Downloads/oregonclean/11962_44k_04.wav'	30.95397949	17	3500	407
'/home/allison/Downloads/oregonclean/11962_44k_05.wav'	30.95397949	16	3500	400
'/home/allison/Downloads/oregonclean/11962_44k_06.wav'	30.95397949	14	3250	400
'/home/allison/Downloads/oregonclean/11962_44k_07.wav'	32.29980469	14	3500	511
'/home/allison/Downloads/oregonclean/129029_44k_01.wav'	40.37475586	12	3250	400
'/home/allison/Downloads/oregonclean/129029_44k_02.wav'	40.37475586	9	3250	400
'/home/allison/Downloads/oregonclean/129029_44k_03.wav'	40.37475586	11	3000	585

Figure 45: Results of Testing Frequencies of Red-winged Blackbirds from Oregon

The recordings from Oregon all show that within a recording, the frequency is fairly constant. These recordings also show that the center frequency and convolution size that produce the clearest signal are not constant for songs in the same recording. Based on the Stereotypy of Some Parameters of Red-winged Blackbird Song and Temporal Cues and Species Recognition papers, this suggests that all the songs from each recording represent only one out of the two to seven variations of songs that a single Red-winged Blackbird can sing , which vary in modulation frequency[11].

4.9.2 Test on California Files

'/home/allison/Downloads/caliclean/181509_44k_04.wav'	64.599609375
'/home/allison/Downloads/caliclean/181510_44k_01.wav'	37.683105469
'/home/allison/Downloads/caliclean/181510_44k_02.wav'	34.991455078
'/home/allison/Downloads/caliclean/181510_44k_03.wav'	29.608154297
'/home/allison/Downloads/caliclean/181510_44k_04.wav'	34.991455078
'/home/allison/Downloads/caliclean/181510_44k_05.wav'	36.337280273
'/home/allison/Downloads/caliclean/181510_44k_06.wav'	64.599609375
'/home/allison/Downloads/caliclean/189395_44k_01.wav'	41.720581055
'/home/allison/Downloads/caliclean/189395_44k_02.wav'	41.720581055
'/home/allison/Downloads/caliclean/189395_44k_03.wav'	40.374755859
'/home/allison/Downloads/caliclean/189395_44k_04.wav'	43.06640625
'/home/allison/Downloads/caliclean/189395_44k_05.wav'	41.720581055
'/home/allison/Downloads/caliclean/189395_44k_06.wav'	43.06640625
'/home/allison/Downloads/caliclean/189395_44k_07.wav'	43.06640625
'/home/allison/Downloads/caliclean/189395_44k_08.wav'	44.412231445
'/home/allison/Downloads/caliclean/189395_44k_09.wav'	41.720581055
'/home/allison/Downloads/caliclean/189395_44k_10.wav'	41.720581055
'/home/allison/Downloads/caliclean/189395_44k_11.wav'	37.683105469
'/home/allison/Downloads/caliclean/189395_44k_12.wav'	43.06640625
'/home/allison/Downloads/caliclean/189395_44k_13.wav'	40.374755859
'/home/allison/Downloads/caliclean/189395_44k_14.wav'	41.720581055
'/home/allison/Downloads/caliclean/189395_44k_15.wav'	43.06640625
'/home/allison/Downloads/caliclean/189395_44k_16.wav'	43.06640625
'/home/allison/Downloads/caliclean/189395_44k_17.wav'	41.720581055
'/home/allison/Downloads/caliclean/189395_44k_18.wav'	41.720581055
'/home/allison/Downloads/caliclean/189395_44k_19.wav'	44.412231445
'/home/allison/Downloads/caliclean/189395_44k_20.wav'	44.412231445
'/home/allison/Downloads/caliclean/189395_44k_21.wav'	44.412231445
'/home/allison/Downloads/caliclean/189395_44k_22.wav'	41.720581055
'/home/allison/Downloads/caliclean/189395_44k_23.wav'	43.06640625
'/home/allison/Downloads/caliclean/189395_44k_24.wav'	43.06640625
'/home/allison/Downloads/caliclean/189395_44k_25.wav'	43.06640625
'/home/allison/Downloads/caliclean/189395_44k_26.wav'	43.06640625
'/home/allison/Downloads/caliclean/190982_44k_01.wav'	55.178833008
'/home/allison/Downloads/caliclean/190982_44k_02.wav'	53.833007813
'/home/allison/Downloads/caliclean/192406_44k_01.wav'	43.06640625
'/home/allison/Downloads/caliclean/192406_44k_02.wav'	41.720581055
'/home/allison/Downloads/caliclean/192406_44k_03.wav'	40.374755859
'/home/allison/Downloads/caliclean/192406_44k_04.wav'	40.374755859
'/home/allison/Downloads/caliclean/192406_44k_05.wav'	40.374755859
'/home/allison/Downloads/caliclean/192406_44k_06.wav'	21.533203125
'/home/allison/Downloads/caliclean/192406_44k_07.wav'	40.374755859
'/home/allison/Downloads/caliclean/192406_44k_08.wav'	43.06640625
'/home/allison/Downloads/caliclean/192406_44k_09.wav'	43.06640625
'/home/allison/Downloads/caliclean/192406_44k_10.wav'	40.374755859
'/home/allison/Downloads/caliclean/22990_44k_01.wav'	37.683105469
'/home/allison/Downloads/caliclean/22990_44k_02.wav'	36.337280273
'/home/allison/Downloads/caliclean/22990_44k_03.wav'	36.337280273
'/home/allison/Downloads/caliclean/22990_44k_04.wav'	37.683105469

Figure 46: Results of Testing Frequencies of Red-winged Blackbirds from California

I mapped the ranges of the resulting frequencies from the California recordings to the areas from which they came.

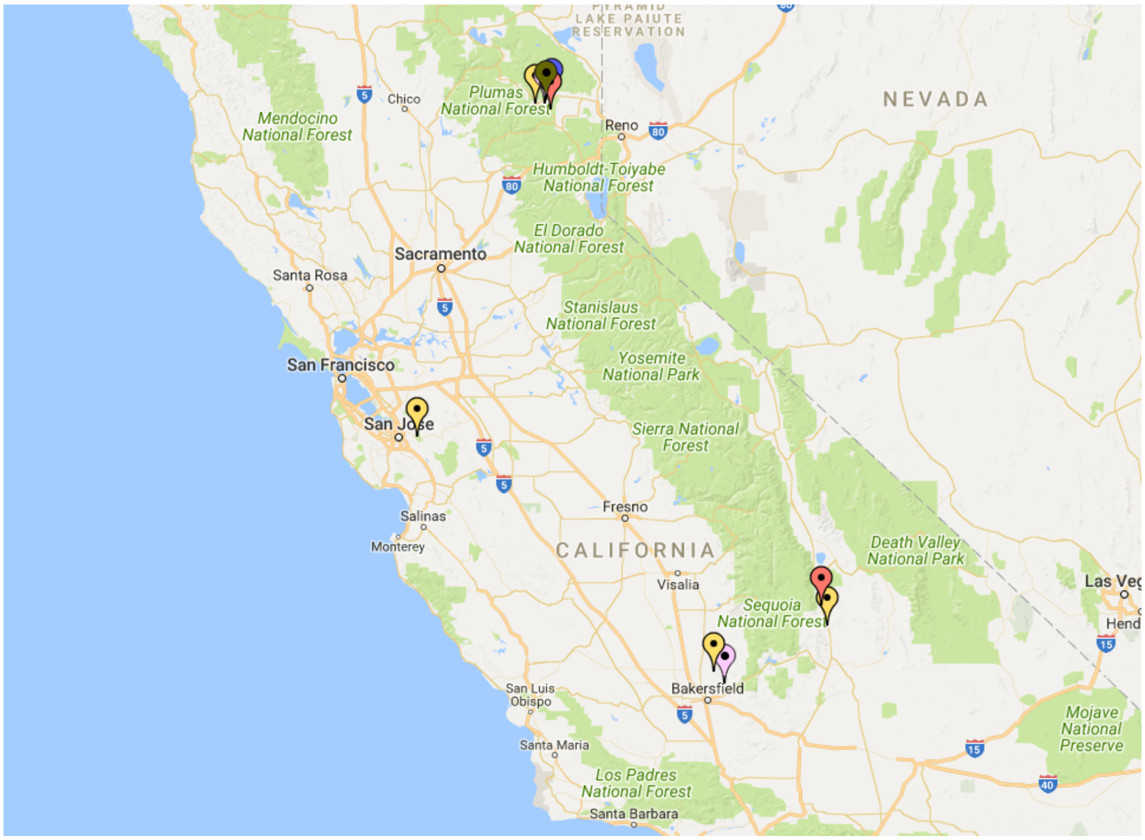


Figure 47: Mapping the California Results to Regions
Color: Center Frequency (Hz)
Olive: 29
Red: 36.5
Yellow: 43
Purple: 54
Blue: 64

This data set is incomplete, so it is unreasonable to make judgements about all the frequencies present in a region. However, all regions contain frequencies of around 43 Hz. This could either be because that frequency is common to California Red-winged Blackbirds, out of chance the frequency happens to be prevalent to the recordings I found, or the frequency is prevalent for Red-winged Blackbirds of all regions.

I later discovered that 43 Hz is in the 20- 200 Hz range that the *Temporal Cues and Species Recognition* paper defined[3]. Therefore, it is natural that 43 Hz would be in this range. Further testing is still necessary to discover if the 43 Hz song is more common than songs of other frequencies for California Red-winged Blackbirds.

4.9.3 Test on New York Files

All of the following recordings come from Tompkins, Ithaca and the dates of the recordings, in order, are 4/30/1952, 4/29/1952, 4/22/1952, 5/23/1952, 3/6/1956.

Filename	Frequency
'/home/allison/Downloads/newyorkclean/11944_44k_01.wav'	80.749511719
'/home/allison/Downloads/newyorkclean/11944_44k_02.wav'	80.749511719
'/home/allison/Downloads/newyorkclean/11944_44k_03.wav'	80.749511719
'/home/allison/Downloads/newyorkclean/11944_44k_04.wav'	79.403686523
'/home/allison/Downloads/newyorkclean/11944_44k_05.wav'	82.095336914
'/home/allison/Downloads/newyorkclean/11944_44k_06.wav'	79.403686523
'/home/allison/Downloads/newyorkclean/11945_44k_01.wav'	84.786987305
'/home/allison/Downloads/newyorkclean/11945_44k_02.wav'	65.94543457
'/home/allison/Downloads/newyorkclean/11945_44k_03.wav'	67.291259766
'/home/allison/Downloads/newyorkclean/11946_44k_01.wav'	74.020385742
'/home/allison/Downloads/newyorkclean/11946_44k_02.wav'	75.366210938
'/home/allison/Downloads/newyorkclean/11946_44k_03.wav'	61.907958984
'/home/allison/Downloads/newyorkclean/11946_44k_04.wav'	74.020385742
'/home/allison/Downloads/newyorkclean/11946_44k_05.wav'	74.020385742
'/home/allison/Downloads/newyorkclean/11947_44k_01.wav'	74.020385742
'/home/allison/Downloads/newyorkclean/11947_44k_02.wav'	74.020385742
'/home/allison/Downloads/newyorkclean/11947_44k_03.wav'	75.366210938
'/home/allison/Downloads/newyorkclean/11947_44k_04.wav'	74.020385742
'/home/allison/Downloads/newyorkclean/11947_44k_05.wav'	72.674560547
'/home/allison/Downloads/newyorkclean/11947_44k_06.wav'	74.020385742
'/home/allison/Downloads/newyorkclean/11947_44k_07.wav'	74.020385742
'/home/allison/Downloads/newyorkclean/11947_44k_08.wav'	64.599609375
'/home/allison/Downloads/newyorkclean/11947_44k_09.wav'	68.637084961
'/home/allison/Downloads/newyorkclean/11947_44k_10.wav'	67.291259766
'/home/allison/Downloads/newyorkclean/11947_44k_11.wav'	67.291259766
'/home/allison/Downloads/newyorkclean/11947_44k_12.wav'	67.291259766
'/home/allison/Downloads/newyorkclean/11947_44k_13.wav'	59.216308594
'/home/allison/Downloads/newyorkclean/11947_44k_14.wav'	68.637084961
'/home/allison/Downloads/newyorkclean/11947_44k_15.wav'	65.94543457
'/home/allison/Downloads/newyorkclean/11948_44k_01.wav'	68.637084961
'/home/allison/Downloads/newyorkclean/11948_44k_02.wav'	67.291259766
'/home/allison/Downloads/newyorkclean/11948_44k_03.wav'	67.291259766
'/home/allison/Downloads/newyorkclean/11949_44k_01.wav'	47.103881836
'/home/allison/Downloads/newyorkclean/11949_44k_02.wav'	45.758056641
'/home/allison/Downloads/newyorkclean/11949_44k_03.wav'	58.543395996
'/home/allison/Downloads/newyorkclean/11949_44k_04.wav'	58.543395996
'/home/allison/Downloads/newyorkclean/11949_44k_05.wav'	44.412231445
'/home/allison/Downloads/newyorkclean/11949_44k_06.wav'	43.06640625
'/home/allison/Downloads/newyorkclean/11949_44k_07.wav'	45.758056641
'/home/allison/Downloads/newyorkclean/11949_44k_08.wav'	45.758056641
'/home/allison/Downloads/newyorkclean/11949_44k_09.wav'	57.870483398
'/home/allison/Downloads/newyorkclean/11953_44k_01.wav'	49.795532227
'/home/allison/Downloads/newyorkclean/11953_44k_02.wav'	49.795532227
'/home/allison/Downloads/newyorkclean/11953_44k_03.wav'	49.795532227
'/home/allison/Downloads/newyorkclean/11953_44k_04.wav'	49.795532227
'/home/allison/Downloads/newyorkclean/11953_44k_05.wav'	49.795532227
'/home/allison/Downloads/newyorkclean/11953_44k_06.wav'	49.795532227
'/home/allison/Downloads/newyorkclean/11953_44k_07.wav'	57.870483398

Figure 48: Results of Testing Frequencies of Red-winged Blackbirds from Ithaca, New York

These recordings show frequencies that are less consistent with each other than from the other regions, which could be due to the quality of the recording or interfering noise. These results also show that in the same time period and in the same location, the different grouping of frequencies vary through the 30 to 85 Hz range. Because one bird can sing at different frequencies, this is expected. Furthermore, Ithaca Red-winged

Blackbirds respond the same to songs of all the frequencies in their frequency range[3].

4.9.4 Test on Massachusetts Files

Sheet4

Filename	Frequency	Number of Bumps
'/home/allison/Downloads/massclean/11987_44k_01.wav'	68.637084961	58
'/home/allison/Downloads/massclean/11987_44k_02.wav'	91.516113281	42
'/home/allison/Downloads/massclean/11987_44k_03.wav'	37.683105469	21
'/home/allison/Downloads/massclean/11987_44k_04.wav'	37.683105469	23
'/home/allison/Downloads/massclean/11987_44k_05.wav'	36.337280273	20
'/home/allison/Downloads/massclean/11987_44k_06.wav'	37.683105469	23
'/home/allison/Downloads/massclean/11987_44k_07.wav'	36.337280273	15
'/home/allison/Downloads/massclean/11987_44k_08.wav'	36.337280273	18
'/home/allison/Downloads/massclean/11987_44k_09.wav'	36.337280273	15
'/home/allison/Downloads/massclean/11987_44k_10.wav'	36.337280273	23
'/home/allison/Downloads/massclean/11987_44k_11.wav'	37.683105469	15
'/home/allison/Downloads/massclean/11987_44k_12.wav'	36.337280273	20
'/home/allison/Downloads/massclean/11987_44k_13.wav'	36.337280273	18
'/home/allison/Downloads/massclean/11987_44k_14.wav'	36.337280273	22
'/home/allison/Downloads/massclean/11987_44k_15.wav'	36.337280273	21
'/home/allison/Downloads/massclean/11987_44k_16.wav'	41.720581055	32
'/home/allison/Downloads/massclean/11987_44k_17.wav'	65.94543457	34
'/home/allison/Downloads/massclean/11987_44k_18.wav'	43.06640625	31
'/home/allison/Downloads/massclean/11987_44k_19.wav'	36.337280273	20
'/home/allison/Downloads/massclean/11987_44k_20.wav'	37.683105469	23
'/home/allison/Downloads/massclean/11987_44k_21.wav'	41.720581055	30
'/home/allison/Downloads/massclean/11987_44k_22.wav'	44.412231445	32
'/home/allison/Downloads/massclean/11987_44k_23.wav'	71.328735352	27
'/home/allison/Downloads/massclean/11988_44k_01.wav'	52.487182617	34
'/home/allison/Downloads/massclean/11988_44k_02.wav'	53.833007813	31
'/home/allison/Downloads/massclean/11988_44k_03.wav'	53.833007813	27
'/home/allison/Downloads/massclean/11988_44k_04.wav'	53.833007813	30
'/home/allison/Downloads/massclean/11988_44k_05.wav'	53.833007813	29
'/home/allison/Downloads/massclean/11989_44k_01.wav'	86.1328125	59
'/home/allison/Downloads/massclean/11989_44k_02.wav'	44.412231445	31
'/home/allison/Downloads/massclean/11989_44k_03.wav'	51.81427002	40
'/home/allison/Downloads/massclean/11989_44k_04.wav'	51.141357422	27
'/home/allison/Downloads/massclean/11989_44k_05.wav'	52.487182617	36
'/home/allison/Downloads/massclean/11989_44k_06.wav'	44.412231445	30
'/home/allison/Downloads/massclean/11990_44k_01.wav'	51.141357422	32
'/home/allison/Downloads/massclean/11990_44k_02.wav'	49.795532227	28
'/home/allison/Downloads/massclean/11990_44k_03.wav'	51.141357422	28
'/home/allison/Downloads/massclean/11990_44k_04.wav'	51.141357422	32
'/home/allison/Downloads/massclean/11990_44k_05.wav'	51.141357422	28
'/home/allison/Downloads/massclean/11990_44k_06.wav'	53.833007813	24
'/home/allison/Downloads/massclean/11990_44k_07.wav'	49.795532227	30
'/home/allison/Downloads/massclean/11991_44k_01.wav'	67.291259766	30
'/home/allison/Downloads/massclean/11991_44k_02.wav'	41.720581055	24
'/home/allison/Downloads/massclean/11991_44k_03.wav'	44.412231445	32
'/home/allison/Downloads/massclean/11991_44k_04.wav'	69.982910156	19
'/home/allison/Downloads/massclean/11991_44k_05.wav'	41.720581055	27
'/home/allison/Downloads/massclean/11991_44k_06.wav'	41.720581055	27
'/home/allison/Downloads/massclean/11991_44k_07.wav'	43.06640625	30

Page 1

Figure 49: Results of Testing Frequencies of Red-winged Blackbirds from Massachusetts

Inspecting the number of bumps shows that it is inconsistent between frequency groups and therefore not be used as a parameter of similarity. Based on the results from *Sterotypy of Some Parameters of Red-winged Blackbirds*, trill length varies between the different song variations of one male Red-winged Blackbird and different Red-winged Blackbirds[11]. My results support this claim.

4.9.5 Regional Dialects

Based on the results from all the files tested from the Lab of Ornithology database, it seems that the regional dialects of Red-winged Blackbirds cannot be identified by the frequency of modulation. According to Temporal Cues and Species Recognition, the frequency extrema can. For instance, an Ithaca, New York Red-winged Blackbird will not sing at a modulation frequency of 20 to 40 Hz or 108 to 200 Hz, but a California Red-winged Blackbird will[3].

Instead, the dialect might lie more prominently in other factors. For example, in the spectrograms of the Red-winged Blackbirds from California, there was a set of straight lines surrounding the song, that was not present in the songs of other states.

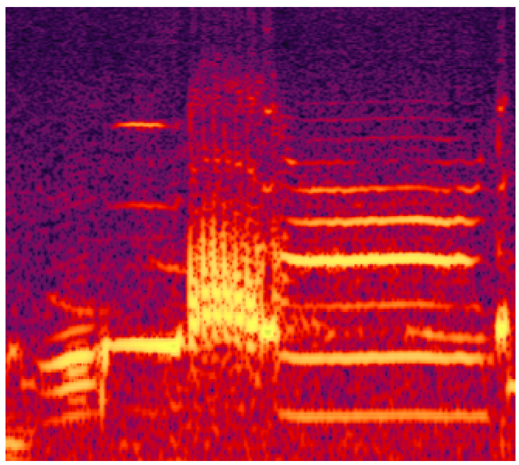


Figure 50: Spectrogram Displaying Nuance of California File

Or, for the case of Red-winged Blackbirds from Maryland, there was a set of points of power followed by a wave-like change in power where in the Bozeman calls there was a mesh.

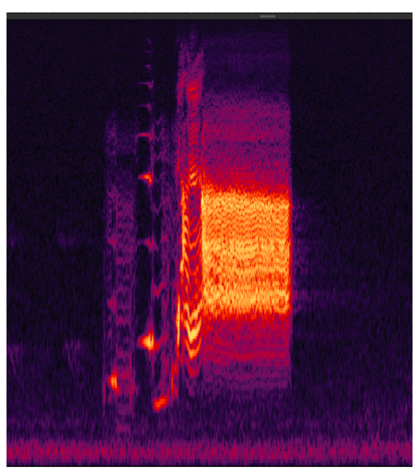


Figure 51: Spectrogram Displaying Nuance of Maryland File

For a Red-winged Blackbird, however, all that it needs to recognize another Red-winged Blackbird is to be in the right frequency and modulation frequency ranges [3].

4.10 Iterations on the Program

In this section I describe how I improve my working programs.

I took out the number of bumps as a parameter because, as I mentioned above, it was not constant enough for the different regions.

Because of the wide range of different frequencies that I found from the section above, I decided to create a program to would find all the instances of a reference signal that you put into it, instead of a constant 38 Hz, as I had before because of the results from Langhor Park. For optimal use, this signal should be from the recording, because of the different frequencies of modulation.

4.10.1 Creating a GUI

Not only were the frequencies of the songs of different regions variable, but also the frequency range at which they occurred and how much background noise was in the recording. To account for all these variations, I created a GUI using MATLAB's App Designer to go along with my program. When the user starts the program, the graph of the reference signal will show up, and there will be slider options to change the convolution size, center frequency and start and end times of the signal. When the user modifies the signal to her liking, she presses the 'Go' button and then the 'Continue' button from the pop-up window and the rest of the program will run to completion.

The function of the 'Continue' window is to make the program wait for the output of the GUI before running the hysteresis function. Otherwise, it will throw an error that all the input parameters are not present.

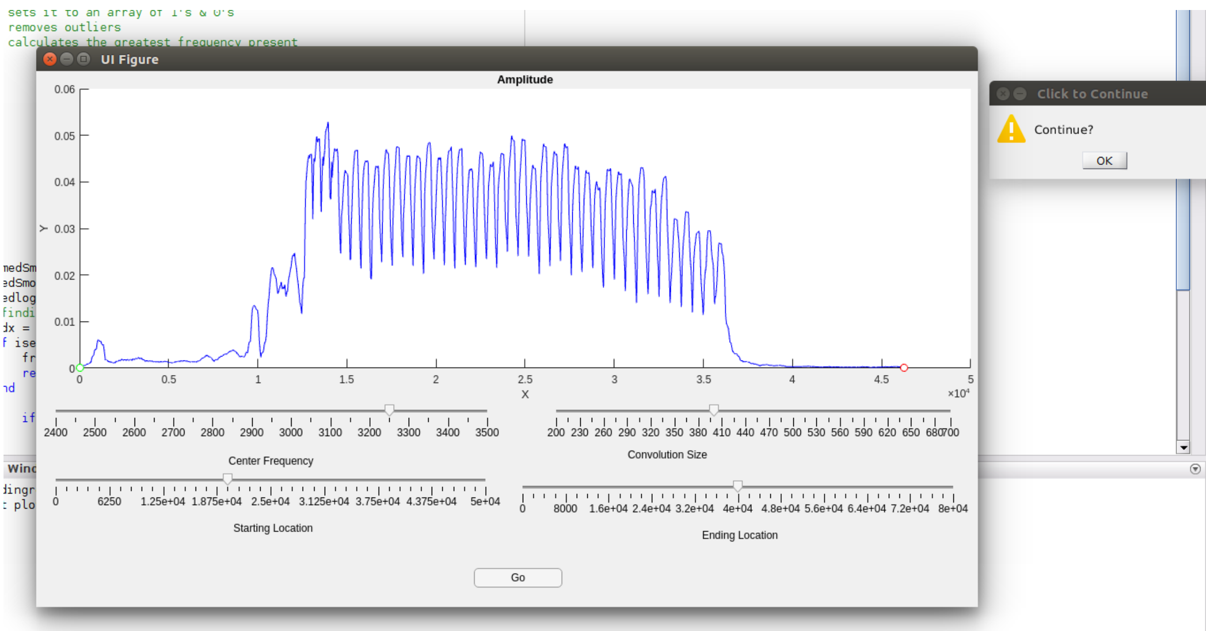


Figure 52: Final GUI System

4.10.2 Grouping Longer Songs Together

Some of the calls are longer than half a second but the program only checks for intervals of 20,000 samples or roughly 0.5 seconds. Therefore, when these longer signals were returned in the results, they were returned as two or more separate results, instead of one longer one. I solved this by adding a check at the end of the program that will group two results that have overlapping time intervals and the same frequencies together.

4.11 Testing New Program on Cornell Lab of Ornithology Files

In this section I describe testing my improved programs on files from the Cornell Lab of Ornithology.

4.11.1 File 11947 : New York

I tested the program just described on file 11947, from New York state, with a Red-winged Blackbird song, file 11947_04, as a reference signal. I checked the accuracy of the results by comparing the times of the songs with the spectrogram of the whole file on Audition. At first, the results corresponded with all the prominent Red-winged Blackbird songs. That is, with the exception of two, which it missed, because there were other bird calls going through the song, interfering with the data. Then, however, the times that the results produced began corresponding with Red-winged Blackbird songs that were not prominent. I extracted audio files for all the times that the program results output a match, as well as the prominent

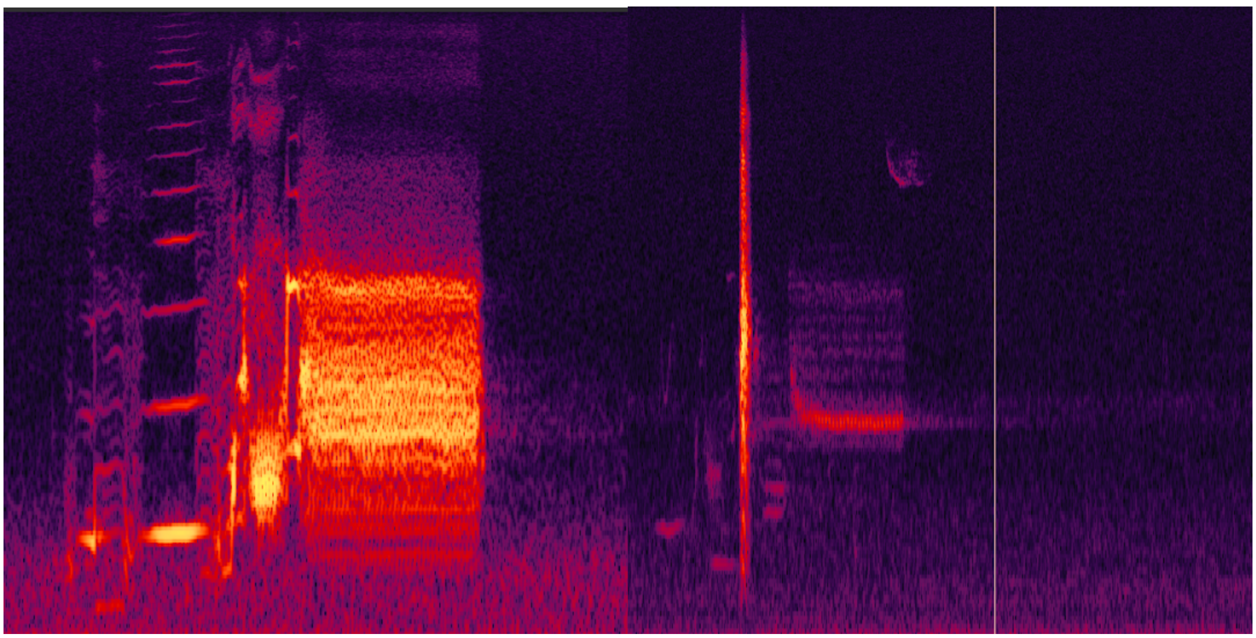


Figure 53: Prominent Song

Figure 54: Not Prominent Song of the Same Modulation Frequency

Red-winged Blackbird calls throughout the whole file and the two prominent calls that the program missed. I then used my frequency extraction program to find the frequencies of all those files.

Filename	Frequency
'/home/allison/Downloads/prominentandnonprominent/11947_44k_missed10prominent.wav'	68.6370849609
'/home/allison/Downloads/prominentandnonprominent/11947_44k_missed11prominent.wav'	65.9454345703
'/home/allison/Downloads/prominentandnonprominent/11947_44k_missed1butotherbirdcallthroughit.wav'	5.3833007813
'/home/allison/Downloads/prominentandnonprominent/11947_44k_missed2butbirdcallrunningthroughit.wav'	75.3662109375
'/home/allison/Downloads/prominentandnonprominent/11947_44k_missed3prominent.wav'	64.599609375
'/home/allison/Downloads/prominentandnonprominent/11947_44k_missed4prominent.wav'	68.6370849609
'/home/allison/Downloads/prominentandnonprominent/11947_44k_missed5prominent.wav'	67.2912597656
'/home/allison/Downloads/prominentandnonprominent/11947_44k_missed7prominent.wav'	67.2912597656
'/home/allison/Downloads/prominentandnonprominent/11947_44k_missed8prominent.wav'	67.2912597656
'/home/allison/Downloads/prominentandnonprominent/11947_44k_missed9prominent.wav'	59.2163085938
'/home/allison/Downloads/prominentandnonprominent/11947_44k_pickedup1.wav'	76.7120361328
'/home/allison/Downloads/prominentandnonprominent/11947_44k_pickedup11notprominent.wav'	71.3287353516
'/home/allison/Downloads/prominentandnonprominent/11947_44k_pickedup12notprominent.wav'	71.3287353516
'/home/allison/Downloads/prominentandnonprominent/11947_44k_pickedup13notprominent.wav'	72.6745605469
'/home/allison/Downloads/prominentandnonprominent/11947_44k_pickedup14notprominent.wav'	72.6745605469
'/home/allison/Downloads/prominentandnonprominent/11947_44k_pickedup15notprominent.wav'	72.6745605469
'/home/allison/Downloads/prominentandnonprominent/11947_44k_pickedup2.wav'	75.3662109375
'/home/allison/Downloads/prominentandnonprominent/11947_44k_pickedup3.wav'	74.0203857422
'/home/allison/Downloads/prominentandnonprominent/11947_44k_pickedup4butdiff.wav'	76.7120361328
'/home/allison/Downloads/prominentandnonprominent/11947_44k_pickedup5.wav'	75.3662109375
'/home/allison/Downloads/prominentandnonprominent/11947_44k_pickedup6.wav'	74.0203857422
'/home/allison/Downloads/prominentandnonprominent/11947_44k_pickedup7.wav'	72.6745605469
'/home/allison/Downloads/prominentandnonprominent/11947_44k_pickedup8.wav'	74.0203857422
'/home/allison/Downloads/prominentandnonprominent/11947_44k_pickedup9.wav'	74.0203857422

Figure 55: Results from Testing the Prominence of the Signals Detected in Program

Color: Type of Song/ Frequency
Dark Gray: Missed Prominent Songs, excluding the two mentioned from the beginning
White: The two missed calls from the beginning with other bird calls running through it
Light Gray: Songs picked up by the program that were prominent
Yellow: Songs picked up by the program that were non-prominent
Purple: Frequencies in the 71 - 76 Hz range
Green: Frequencies in the 67 - 68 Hz range, 59 Hz, 5 Hz and 75 Hz

These results show that the program did pick up the songs in the right frequency range, except for those two that were corrupted with the different bird calls.

5 Analysis

Frequency modulation is fundamental to producing the male Red-winged Blackbird song, and therefore, by looking for frequency modulation we are looking directly for Red-winged Blackbirds songs. Even though some of the spectrograms of Red-winged Blackbirds from different regions contain individual nuances, the results from this project show that Red-winged Blackbirds from California to Massachusetts sing with a frequency modulation between 20 and 80 Hz, disregarding a few exceptions. The fact that *Temporal Cues and Species Recognition* states that Red-winged Blackbirds sing at ranges from 20 to 200 Hz validates my range and provides incentive to continue analyzing songs to discover if I could reproduce the same extrema in their range.

Because of my previously mentioned result, the frequency modulation was not the determining factor of dialects for Red-winged Blackbirds from different regions, because all regions exhibited the same range of frequencies. Moreover, there were recordings from this project that exhibited Red-winged Blackbird songs that stayed within 2 Hz of a frequency, and other recording that exhibited groupings of frequencies, for which in each individual grouping, the frequency stayed almost constant, but the groupings as a whole were not close in value to each other, in respect to the 20 to 80 Hz range. I suspected that the groupings represent different male Red-winged Blackbirds in the same region. However, after reading the research papers I found on August 7th, the groupings could be the product of one bird singing different variations of his song. Moreover, the results of the two research papers showed different frequency ranges than I calculated. Without access to their data and method by which they calculated the modulation frequency, I can only guess as to where our calculations differed. However, in the case of *Temporal Cues and Species Recognition*, our ranges overlap, but their ranges are wider and differ for different geographical regions at the extrema. Possibly if I analyzed more recordings from different times of year and in different situations then I would discover song frequencies that expand my current range. Doing this and finding that the extrema of Red-winged Blackbirds in different regions differs, I could exploit that characteristic of modulation frequency to identify the potential region from where a Red-winged Blackbird song was sung automatically.

Compared to manually calculating modulation frequency, the analysis of the different extrema would be relatively short and easy to conduct, using the programs that I have created in the last ten weeks. My GUI program which is used for one specified Red-winged Blackbird song can reliably return the frequency of that song. Therefore, this program can be a useful tool in studying the frequency modulation of Red-winged Blackbirds and exploring the phenomenon of the differences in frequency modulation in a region.

Because of these varying frequencies of modulation in a region, my second program, which detects the instances of Red-winged Blackbird songs in a long audio file, will need to search for frequency through the whole modulation frequency range, whatever the user defines that to be, to find all instances of Red-winged Blackbirds. The trade-off for this approach is that it produces many more false positives. Alternatively, a user can input one Red-winged Blackbird song from the audio recording as a separate audio file, use the GUI to set the right parameters for the frequency detection of the reference signal, and let the program find all of the instances at which that specific Red-winged Blackbird from the reference signal sang at that specific modulation frequency. When I tested this on file 11947, I found that the power level of the songs detected from the frequency of the reference signal decreased throughout the recording. Why exactly this happened is unclear to me, but this shows that my program could potentially be a tool to systematically track the power levels of Red-winged Blackbird songs for a specific bird.

6 Conclusion

Red-winged Blackbirds have a pattern of modulation frequency . This can be abstracted so that it is detectable by a computer. The frequency at which the modulation occurs has insofar with my program been measured to be between 20 and 80 Hz. This includes Red-winged Blackbirds from New York, Bozeman, California, Maryland, Massachusetts, Ontario, Oregon, Manitoba, Indiana and Cuba. However, the frequency of a region was not constant. Instead, even in a single recording multiple groups of instances of different frequencies could be found.

All of this was able to be tested using a GUI program that works with the user to find the cleanest part of a Red-winged Blackbird signal and then uses the input as parameters to calculate the frequency of the modulation using a hysteresis system which converts the signal to an on-off system with 1's representing the peaks. It then finds the frequency of this system using MATLAB's *periodogram()* function and finding the frequency with the largest power.

This program, in turn, is used as a helper function for another program I created which can take a reference signal and a long audio file from which it came, implement the user interface to find the frequency of modulation of the reference signal, and then search for that signal in the long audio file.

The last program I created is similar in that it searches for all instances of Red-winged Blackbird songs in a long audio file. However, it does not take in a reference signal, instead searching using the parameters which you give to it : the center frequency, the convolution size, the reference frequency and how far from the reference frequency should the program look for signals in. This program works with 59.5 to 85.8 % accuracy, depending on the parameters set, and the amount of other noise present in the long audio file.

If I had more time to work on this study, I would search for more papers similar to the ones I found on August 7th. I would then use my program to further analyze the modulation frequency ranges that the authors describe to be specific to an area. I would like to see why my results insofar vary at the extrema with the results from Temporal Cues and Species Recognition. Furthermore, I would exploit the vast amount of data that I have and the facility by which I can analyze their modulation frequency. For example, in the case of the California Red-winged Blackbirds, I would conduct more experiments to discover if 43 Hz is actually a more common modulation frequency, or if my limited data set only portrayed it that way.

Also, now since I know that one Red-winged Blackbird has two to seven different modulation frequency variations, I would like to see if in combination with my programs, I would be able to exploit that fact and track one Red-winged Blackbird in its habitat. Possibly with the new information that I discovered at the end of the project, as well as my results with file 11947 showing change in power and more information about the animal behavior of Red-winged Blackbirds, a method of tracking individual birds in an audio file could be devised. Doing so would enable me to work towards facilitating that, which inspired me to pursue this project: accurately recording the number of bird, instead of bird songs, in a region.

In terms of modifying my program, I had debated during the program whether or not I should include the constant maximum and minimum thresholds of oscillations of the song as a parameter in my system. *Temporal Cues and Species Recognition* finds this phenomenon to be a distinguishing factor of mockingbird and Red-winged Blackbird songs, which both contain trills in a similar modulation frequency range. Therefore, with the chance to continue, I would certainly add this as a parameter to my test.

Nonetheless, I believe that the programs that I have created up to this point would be beneficial to researchers of animal behavior, like the authors of *Temporal Cues and Species Recognition* and *Stereotypy of Some Parameters of Red-winged Blackbird Song*. Using these programs could accelerate the process of analyzing bird song data and would enable the researchers to use more data and, thus, come to more accurate and more generalized conclusions about bird behaviors.

References

[1] *8: Correlation*. URL: http://www.ee.ic.ac.uk/hp/staff/dmb/courses/E1Fourier/00800_Correlation.pdf.

[2] *About Us*. URL: http://www.birds.cornell.edu/page.aspx?pid=1609.

[3] Eliot A.Brenowitz. "The contribution of temporal song cues to species recognition in the red-winged blackbird". In: *Animal Behaviour* 31.4 (), pp. 1116–1127.

[4] *Amplifier Clipping: How Do I Avoid It?* URL: https://www.mtx.com/library-clipping.

[5] N. Dwight Barnette. *Digital Audio*. URL: http://courses.cs.vt.edu/~cs1604/notes/chap1/sound.html.

[6] Enrico Bossart. *What is meant by the switching function "hysteresis" with electronic pressure switches?* URL: https://blog.wika.com/knowhow/switching-function-hysteresis-pressure-switch/.

[7] Erik Cheever. *The Convolution Integral*. URL: http://lpsa.swarthmore.edu/Convolution/Convolution.html#analytic.

[8] *conv*. URL: https://www.mathworks.com/help/matlab/ref/conv.html#buc311j-4.

[9] Christophe Couvreur. *octave*. URL: https://www.mathworks.com/matlabcentral/fileexchange/69-octave.

[10] *Digital Audio Basics: Sample Rate and Bit Depth*. URL: https://www.presonus.com/learn/technical-articles/Sample-Rate-and-Bit-Depth.

[11] Candance B. Breen Douglas G. Smith Fiona A. Reid. "Stereotypy of Some Parameters of Red-winged Blackbird SOng". In: *Condor* 82 (), pp. 256–266.

[12] Dan Ellis. *Audfprint*. URL: https://github.com/dpwe/audfprint.

[13] Michael Greenacre. *Chapter 4 :Measures of distance between samples: Euclidean*. URL: http://www.econ.upf.edu/~michael/stanford/maeb4.pdf.

[14] *High-Resolution Recording... the "R Series" Revolution*. URL: https://www.roland.com/us/products/r-09hr/.

[15] Julius O. Smith III. *Cross-Power Spectral Density*. URL: https://ccrma.stanford.edu/~jos/sasp/Cross_Power_Spectral_Density.html.

[16] Donald E.Kroodsma John R.Krebs. "Repertoires and Geographical Variation in Bird Song". In: *Advances in the Study of Behavior* 11 (1980), pp. 143–177.

[17] Ph.D. Joseph M. Szewczak Ph.D. Michael L. Morrison. "Bird Species Identification and Population Estimation by Computerized Sound Analysis". In: *STATE OF CALIFORNIA DEPARTMENT OF TRANSPORTATION* (2010).

[18] Rebecca A. Efroymson Latha M. Baskaran Virginia H. Dale and William Birkhead. “Habitat Modeling within a Regional Context: An Example Using Gopher Tortoise”. In: *The American Midland Naturalist* 155.2 (2006), pp. 335–351.

[19] Rober J. Lennox. “Conservation physiology of animal migration”. In: *Conservation Phsyiology* 4 (2016).

[20] Scott McKinney. *Fast root-mean-square (RMS) power*. URL: https://www.mathworks.com/matlabcentral/fileexchange/30085-fast-root-mean-square-rms-power.

[21] *mscohere*. URL: https://www.mathworks.com/help/signal/ref/mscohere.html#bvi4l0m-1.

[22] *Naming of musical notes and piano keys*. URL: http://www.sengpielaudio.com/calculator-notenames.htm.

[23] M. Nilsson. *The audio/mpeg Media Type*. URL: https://tools.ietf.org/html/rfc3003.

[24] Robyn Owens. *Fourier transform theory*. URL: http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OWENS/LECT4/node2.html.

[25] Jaroslav Picman. “Territory Establishment, Size, and Tenacity by Male Red-Winged Blackbirds”. In: *The Auk* 104.3 (), pp. 405–412.

[26] *Red-winged Blackbird Sounds*. URL: https://www.allaboutbirds.org/guide/Red-winged_Blackbird/sounds.

[27] Tamara Smyth. *Amplitude Detection in Matlab*. URL: https://www.cs.sfu.ca/~tamaras/sinusoids/Amplitude_Detection_in.html.

[28] *Sound intensity I and the inverse square law*. URL: http://www.sengpielaudio.com/calculator-squarelaw.htm.

[29] Petre Stoica and Randolph Moses. *SPECTRAL ANALYSIS OF SIGNALS*. URL: http://user.it.uu.se/~ps/SAS-new.pdf.

[30] James Tingay. *What are Octave and Third Octave Band Filters on a Sound Level Meter?* URL: https://www.cirrusresearch.co.uk/blog/2011/11/what-are-octave-and-third-octave-band-filters-on-a-sound-level-meter/.

[31] Avery Li-Chun Wang. *An Industrial-Strength Audio Search Algorithm*. URL: http://www.ee.columbia.edu/~dpwe/papers/Wang03-shazam.pdf.

[32] Ruye Wang. *Digital Convolution – E186 Handout*. URL: http://fourier.eng.hmc.edu/e161/lectures/convolution/convolution.html.

[33] Ruye Wang. *Median Filter*. URL: http://fourier.eng.hmc.edu/e161/lectures/smooth_sharpen/node2.html.

[34] *Welcome to Acoustic Atlas*. URL: http://acousticatlas.org/about.php.

[35] *What is xeno-canto?* URL: https://www.xeno-canto.org/.

[36] *Wildlife Monitoring and Wildlife Viewing Camera Systems : Frequently Asked Questions*. URL: https://www.nps.gov/pore/learn/nature/wildlife_monitoring.htm.

[37] Cheng Yang. *MACS: MUSIC AUDIO CHARACTERISTIC SEQUENCE INDEXING FOR SIMILARITY RETRIEVAL*. URL: infolab.stanford.edu/~yangc/pub/cy-waspaa01.pdf.

7 Attribution to Cornell Lab of Ornithology Files

The following recordings from the Macaulay Library at the Cornell Lab of Ornithology were referenced: 57173, 57174, 57175, 57176, 57194, 57195, 57196, 57198, 57199, 57500, 57503, 57504,57510, 57511, 57516, 63953, 68547, 11987, 11988, 11989, 11990, 11991, 11992, 11993, 11994, 11995, 11996, 11997, 11998, 11999, 12024, 12026, 12028, 12029, 12030, 12032, 12033, 12034, 12035, 12036, 12037, 12038, 12039, 12040, 12041, 12042, 12043, 12045, 12046, 12047, 12048, 12049, 12050, 12051, 12052, 12055, 12056, 12057, 12058, 12059, 12060, 12061, 12062, 12063, 12065, 12066, 12067, 12068, 12069, 12070, 12071, 12072, 12073, 12074, 12076, 12077, 12078, 12079, 12080, 12081, 12082, 12083, 12084, 12086, 12087, 12088, 12090, 12091, 12097, 12098, 12099, 12103, 12104, 12106, 12110, 12111, 12112, 12113, 12114, 12115, 12116, 12124, 12127, 12130, 12136, 03106, 12073, 12074, 11637, 11944, 11945, 11946, 11947, 11948, 11949, 11953, 11958, 11985, 12000, 12001, 12002, 12003, 12004, 12117, 12118, 12119, 12121, 12122, 12124, 12126, 12127, 12128, 12130, 12132, 12134, 12136, 12137, 12138, 12140, 12173, 12175, 12176, 12257, 12258, 12261, 12262, 12264, 12268, 12271, 12278, 12284, 12289, 12321, 12322, 12357, 12358, 12359, 12362, 12375, 71931, 72762, 72763, 93758 00739, 00740, 00761, 00777, 09008, 11971, 77271, 77272, 84698, 84701, 84702, 84708, 94214, 94215, 94229, 94230, 94233, 94234, 94240, 94241, 94251, 94252, 94266, 94447, 11050, 11073, 18647, 18649, 18660, 18669, 18852, 11976, 26478, 26479, 26550, 61708, 64409, 80005, 81509, 81510, 89395, 90982, 92406, 22990, 22992, 22993, 39884, 44372, 44386, 44568, 44570, 44848, 45384, 48803, 48811, 56692, 57651, 80364 , 06583, 06833, 11955, 11962, 29029, 29031, 29068, 11987, 11988, 11989, 11990, 11991, 11992, 11993, 11994, 11995, 11996, 11997, 11998, 11999, 12024, 12026, 12028, 12029, 12030, 12032, 12033, 12034, 12035, 12036, 12037, 12038, 12039, 12040, 12041, 12042, 12043, 12045, 12046, 12047, 12048, 12049, 12050, 12051, 12052, 12055, 12056, 12057, 12058, 12059, 12060, 12061, 12062, 12063, 12065, 12066, 12067, 12068, 12069, 12070, 12071, 12072, 12073, 12074, 12076, 12077, 12078, 12079, 12080, 12081, 12082, 12083, 12084, 12086, 12087, 12088, 12090, 12091, 12097, 12098, 12099, 12103, 12104, 12106, 12110, 12111, 12112, 12113, 12114, 12115, 12116, 12124, 12127, 12130, 12136, 34508, 34511, 34520, 105608, 105609, 105638, 176162, 176198.