What bird is singing?
Master Thesis

Andreas Qvist 890813-****
Pi-08

June 5, 2013

Abstract:

The goal of this work was to create a model for characterizing bird species through recordings of their song. A large set of twenty species with 10+ recordings in each species was considered and the algorithm used time-frequency models for the characterization. Three different approaches were used in an attempt to build the foundations for the final model in a bottom to top manner. The resulting model divided birdsong into syllables that were analyzed and compared in several time-frequency domains in order to characterize the species. The domains include the Spectrogram, the Doppler domain and the Ambiguity domain. The result varied depending on the complexity of the song and the quality of the recordings but for simple songs with recordings of good quality the results were very good.

Contents

1 Introduction

This project takes aim at determining species through studying the recording of the birds song. This will be attempted through modelling the bird song to determine the various structures that are characteristic for that specific species.

Some key points regarding the limitations and the aim of the entire project needs to be mentioned. The idea has been that the resulting algorithm of the project could be run on a portable device **without** connection to the internet. The reason for this is that such a connection allows for the identification-algorithm to be run on a server with great computational capacity. There is also no limitation on how large the data base with which to compare to signal can be. Another reason is that if we allow access to internet we might as well make use of *"mechanical turks*[1] *"*. Mechanical turk[2] is a concept that Amazon runs where you as a *requester* can register and pay small amounts to *workers* who execute *HITs*, an abbreviation for Human Intelligence Tasks. Using both a server and the mechanical turks we can first use a computational-wise heavier algorithm to come up with a couple of alternatives and then let human intelligence narrow down or decide which species is on the recording. Mechanical turks and a powerful server would of course be expensive but its not unrealistic since there already exists an App (or it is at least under development) called WeBIRD[3] which uses an external server for its identification.

We can conclude from this that it could be economically viable to have such an arrangement but I have chosen to limit the algorithm to such computational power that it could be run on a portable device. This sets limitation on recording resolution database size. I chose to set this limitation with time-frequency analysis as a field of study in mind, since even if these algorithms are heavy computational wise, they allow for large amounts of optimization. Also I have complete confidence that the algorithms that I have been using can be implemented to a smart-phone or similar and run without computational problems.

One first comment might be that the algorithms that I have provided are to resource demanding and would never be able to be run e.g. a smart-phone. While the algorithms are extremely demanding one has to remember that they only need to be run once for each recording which takes a couple of seconds. To create the database this has to be done hundreds of times, but this only has to be done once and can then be stored as a database that the program would have access to. So the program, which is run on a smart-phone in this example, records the song in question transforms it (which takes approximately 10 seconds) and then compares it to the database. This makes all the hard work with a database with keys worth it, cause now the algorithm (which does not need some sort of connection) can be run in seconds.

While this was a introduction to the aspirations of the project some sort of background is needed. There has been models trying to cover this area and bird characterization is a field relevant for both science and enthusiasts. Several articles are available on the subject with different premisses and approaches. One common premiss is to analyze one species and take a deeper look at the structure of this species song. The supervisor of the project *Maria Sandsten* has been involved in such a project before[2], which will be built upon in the third part of this paper. In one paper [3], some similarity can be

[1] `https://www.mturk.com/mturk/welcome`
[2] `http://en.wikipedia.org/wiki/Amazon_Mechanical_Turk`
[3] `http://10000birds.com/webird.htm`

found in the way of dividing the song into syllables, but again it only concerns one species. Attempts to model several species has also been made [4], but with a quite different approach.

2 Material and Methods

2.1 Limitation

Since there is an extreme amount of different bird species, a limitation was needed to get the project going. A discussion with *Maria Sandsten* (the master thesis supervisor) and a biologist *Bengt Hansson* from the Department of Biology at LTH, was held. The outcome was the decision to start with a set of birds where every species is coupled with another species due to similar song, appearance or another point of interest. The birds chosen were (pairwise):

Black-throated diver & Red-throated diver
Curlew (euroasian) & Whimbrel
Tawny owl & Tengmalm's owl
Chaffinch & Brambling
Thrush nightingale & Nightingale
Great reed warbler & Reed warbler
Grashopper warbler & River warbler
Crow & Rook
Great tit & Blue tit
House sparrow & Tree sparrow (euroasian)

2.2 Data Collection

Data was collected from the website Xeno-Canto[4], an enthusiast forum that is sharing birdsong recordings from all over the world. For most species a sample of 10-20 recordings were collected[5].

2.3 Theory

In this section general theory is presented with some background information like history and examples. Some theory, algorithms and methods will be presented in the different sections of the report and the reason for this is that they are best explained in their context.

2.3.1 Frequency Analysis

When analyzing sounds, studying the frequency content is of key interest. A very intuitive idea of frequency exists and this is the sounding *tone*. Fourier transform was invented in 1822[6] to solve a heat dissipation problem and had been around for a long time when Cooley and Tukey invented their famous FFT-algorithm [1] (*fast Fourier transform*) an algorithm for calculating the discrete Fourier transform. It was later discovered that they had actually reinvented an algorithm that the Princeps Mathematicorum (the Prince of Mathematics), Carl Friedrich Gauss (1777-1855), had invented and an unpublished manuscript for a very similar algorithm was found in his collected

[4] http://www.xeno-canto.org/

[5] In accordance with the copyright law on the site,Creative Commons SA-ND http://www.xeno-canto.org/about.php, http://creativecommons.org/licenses/

[6] http://en.wikipedia.org/wiki/Fourier_transform

works[7] [5].

In this report the discrete Fourier transform and the FFT will be extensively used and thus some of the key parameters and concepts need to be discussed.

2.3.2 Discrete Fourier Transform & FFT

The discrete version of the ordinary Fourier transform is the DFT which is adapted to discrete signals or data sets. For a signal x of length N where N is a length that can be written as $2^{integer}$ the complex sequence is formulated as follows

$$X(k) = \sum_{n=0}^{N-1} x_n \cdot e^{-i2\pi kn/N}.$$

The square of the absolute values of the Fourier transform gives the spectrum

$$S(k) = |X(k)|^2.$$

An FFT is usually a well optimized algorithm that computes the DFT. There are many different variations of it but no time was dedicated to finding the best one for this project.

These are both the discrete analogies to the continuous Fourier transform which is used for analytical/continuous signals. In the project the discrete versions are used throughout but for simplicity the continuous versions will be presented instead, throughout the rest of the theory section.

The continuous version of the Fourier transform: For a signal/function $x(t)$, the complex sequence is formulated as follows

$$X(f) = \int_{-\infty}^{\infty} x(t) \cdot e^{-i2\pi ft} dt.$$

The Fourier transform creates the complex function which in absolute gives the spectrum

$$S(f) = |X(f)|^2.$$

2.3.3 Time-Frequency Analysis - The Spectrogram

Since a lot of information in song and audio is contained in how the tones/frequencies are arranged in time, it is not sufficient to look only at the frequency content. For instance one long playing of a tone will look the same as a tone played several times no matter how they are arranged in time. In order to capture this information another approach is invented: time-frequency analysis and more specifically the spectrogram.

The spectrogram (aka Sonogram) is the most intuitive way of taking this approach through windowing the data and estimating the frequency content in each window. This gives a three dimensional picture with axis: time & frequency. The value obtain shows the quantity of that frequency at the specific time-point.

[7] `http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1162257`

There are many things to consider when creating the spectrogram. How to create the windows, should they be smoothed? Should they overlap? When the DFT is made how should this be done? Usually the FFT is adopted but this comes with many options. Most crucial is the length of the window and the corresponding length of the data sequence used in the FFT. These two control the resolution in the time and frequency domain, respectively.

While the FFT (or rather the implementation in MatLab) has been very well optimized it is still very demanding to calculate many of them and this limits the resolution of the spectrogram. For good resolution in the time-domain, short windows are needed and for good resolution in the frequency domain a long window is needed. Inevitably this creates a trade-off between computational speed and resolution and if the running time is fixed it is a trade-off between the resolution in the frequency-domain and in the time-domain.
In analogy with how the spectrum is created as the absolute value of the Fourier transform, the spectrogram is created as the absolute value of the Fourier transform at a certain time (timeframe/window). The Fourier transform of a signal in a "short" window is called the *Short-Time Fourier transform* and is defined as follows

$$X(t,f) = \int_{-\infty}^{\infty} x(t_1)h^*(t_1 - t)e^{-i2\pi f t_1} dt_1.$$

Here again $x(t)$ is an analytical signal or function. The spectrogram can then be designed using the short time Fourier transform as

$$S(t,f) = |X(t,f)|^2.$$

Below are some examples of spectrograms from owls singing.

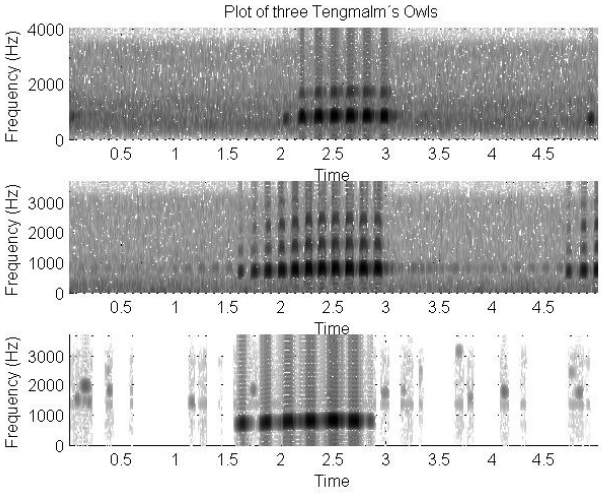

Figure 1: *Spectrogram of the song of three Tengelmans owls.*

If the original representation of the data is seen as one domain and the spectrogram (which is a transformation of this original domain) as a second, there exist two more

possible representations of the data i.e. two more domains. These are the Doppler domain and the Ambiguity domain. These domains are defined by their respective functions which will be described below.

2.3.4 Doppler domain

The Doppler domain will at some instances of this report be called the "freq-freq"' domain. This is due to the fact that it is in a way a frequency-frequency domain. It is calculated through a Fourier transformation along the time axis of the spectrogram creating this frequency-frequency domain.

$$D(\nu, f) = \int_{-\infty}^{\infty} S(t, f) e^{-i2\pi\nu t} dt \tag{1}$$

where $S(f, t)$ is the spectrogram of the signal to be transformed.

2.3.5 Ambiguity function

The Ambiguity domain is similar except that the axis/domain that we transform is changed. Even though the formulas are very similar, it is much harder to give an intuitive explanation of what this represents.

$$A(\nu, \tau) = \int_{-\infty}^{\infty} D(\nu, f) e^{i2\pi f\tau} df \tag{2}$$

where $D(\nu, f)$ is the Doppler function which is then inverse transformed but in the variable f instead t.

2.3.6 Singular Value Decomposition

Singular value decomposition or *SVD* is a matrix factorization and it will in this case be used to extract some characteristics from a matrix (with the purpose of reducing the information). The *SVD* is defined as follows:

$$X = USV' \tag{3}$$

where X is the matrix that is being decomposed, U is a unitary matrix (meaning that $UU' = I$), S is a rectangular diagonal matrix with non-negative values on the diagonal and V' is the transpose of V. In the case of a complex valued matrix X the transpose of V should be replaced with a conjugate. The columns of the U and the V matrices are called the *left singular vectors* and *the right singular vectors* of X. The key idea for using this in signal processing is that the singular vectors holds a lot of information about the matrix decomposed. Figure 2 shows an example of a matrix decomposition of a plot.

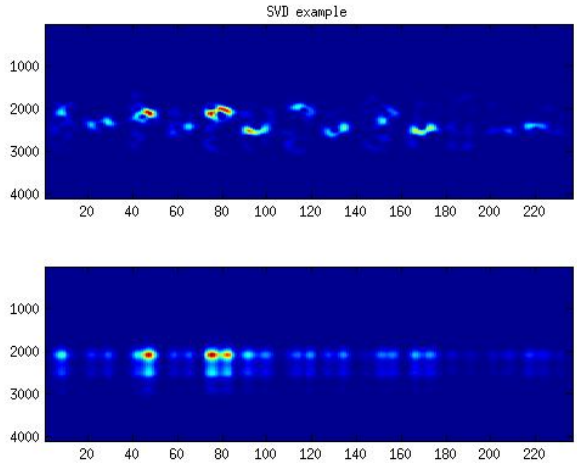

Figure 2: *A sequence of birdsong is broken decomposed into one left and one right singular vector. The upper plot is of the original signal and the lower is of the re-composed image, using only two vectors, one right singular and one left singular.*

3 Data Analysis

3.1 Song or Call

Most birds have several types of sounds. While the song is the most common characteristic for most species they also have calls of different types. These calls are generally used for getting the attention of other birds or serving as warnings etc[8]. For all birds except for the Crow and the Rook, the *song* was collected but for this pair the *call* was collected.

3.2 First analysis

The collected recordings are of varying length and sample rate. Most of them are of a sample rate of 44100 Hz but this information is kept along with all of the files in order to keep track of the frequencies. The amount of background noise is what varies the most but overall the recordings are quite good. In order to make the data more usable, the length of the recordings are all reduced to, at first, 5 seconds. In order to make sure that the data that was picked out of the recording was relevant some type sorting mechanism had to be invented. The solution was to search each file for the intensity maximum[9] and draw the conclusion that the birds song was in some sort of significant part around this time-point. After finding the maximum, 2.5 seconds of song was saved around this point resulting in a total of five seconds.

In order to limit the amount of data to use for the primary studies another limitation was made. This was to look at the first ten recordings of the Tengmalm's Owl. This was due to the fairly simple and regular nature of this species song[10]. The first initial idea and the most basic idea for categorizing the recordings is to look at the frequency content of the signal. After some reviewing of the frequency content it was considered to be safe to decimate the signals by a factor 4. This was done to all signals using the `decimate` function in `Matlab`. In order to make the analysis simpler, at first only a part of the signals are considered. The frequency content is analyzed through the Fourier transform and again the built in function in `Matlab` is used. This one is called `fft` and in order for it to work a number of FFT points needs to be chosen. Since the signal now (after the decimation/down-sampling) have a sampling frequency of 11025 Hz, in general, the number of FFT points (NFFT) is set to $2^{nextpow2(11025)}$ which is 16384. This was done for the ten owl recordings shown in figure refowl15 and figure 4.

[8] `http://en.wikipedia.org/wiki/Bird_vocalization`

[9] $I_{max} = indexofmax(rec^2)$

[10] This song consists of a repeated tonation of the same frequency, see spectrogram example 1

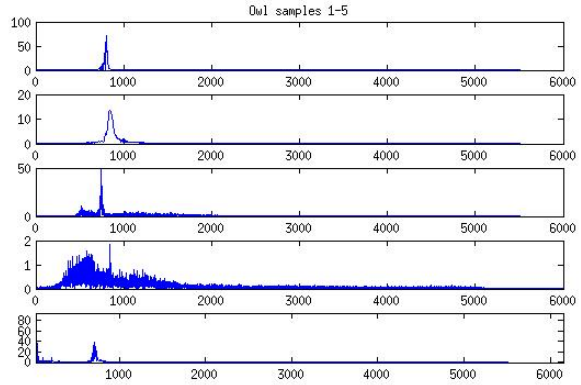

Figure 3: *Frequency content of the first five recordings. These first 5 recordings all share a peak around 800 Hz and they deviate with roughly 100 Hz. The fifth recording has been zoomed in around 800 Hz since there was a very dominant peak in very low frequency, probably due to noise-disturbance.*

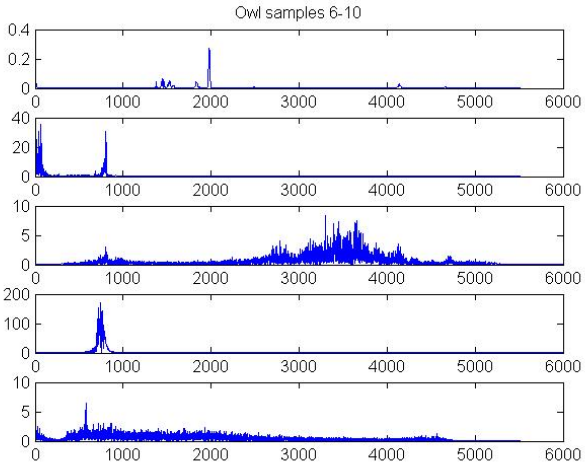

Figure 4: *Frequency content of the 6th to 10th recording. The first recording (6th) shows a deviation in the peak, and this recording indeed exhibits a higher tone in playback. However the really difficult signal in this set is the 8th which does not show the same structure at all as the others (one clear peak). When listening to this recording it is clear that there is another bird of a different species singing equally loud as the Owl and this would characterize this as a bad recording (since the objective is not in focus). This can also be used to test robustness.*

Seven out of ten signals showed very similar frequencies or characteristics. These all had a clear peak at around 800 Hz and this makes for a good first characterization. One signal showed the same one peak characteristics but both where at different frequen-

cies (one at $\approx$ 2000 Hz). The last deviation is a recording of an Tengmalm Owl and another unspecified bird singing at almost the same intensity. While in this last recording there is a peak at $\approx$ 800 Hz there is also a lot of other frequencies that are picked up.

If the recordings with deviations are removed, recording number 8 with two birds and the recordings 4,5 and 7 which had a lot of noise, the mean of the maximums for the owl recordings is 774.9 Hz. When plotted in the same figures it matches fairly well.

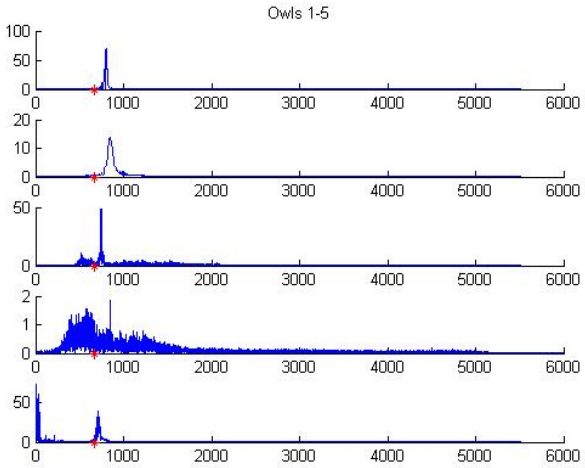

Figure 5: *Frequency content of the 1st to 5th recording with the maximum average for the six valid recordings displayed as a red dot.*

3.2.1 Results of the first approach

In order to get an idea of how viable this is as a classification method the strongest frequency is plotted for every species. Since this includes all the recordings it would be too time-demanding to go through each recording and correct/remove the outliers, so this is the crude average (no adjustments made) but it is still and indication of the potential of this method.

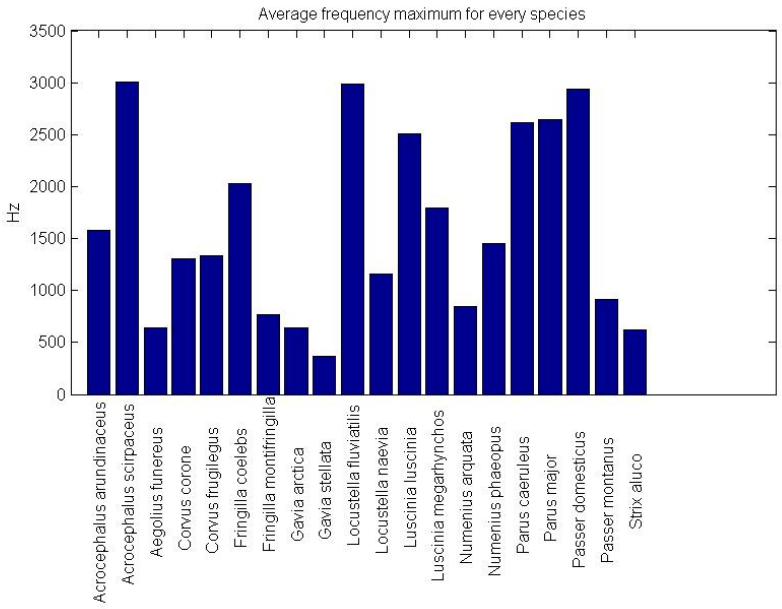

Figure 6: *For every recording the strongest frequency was found and it was weighted into the average for that species.*

From this table it can be concluded that this method is not sophisticated enough if only the strongest frequency is considered. It is a possibility that a combination of several frequencies, say the two or three strongest, could be used to determine the species.This possibility is left in an attempt to use more complex methods.

Comment on result

It is obvious that this coarse method could be greatly improved by high-pass filtering the signals, but since the different species are not distinctly separated (regarding the containing frequencies), this is left undone but applied for the coming methods.

3.3 Second approach - Spectrogram analysis

The spectrogram is the most natural approach to analyzing song or music. It holds[11] as much information as the original signal itself and it displays it through showing component frequencies and their appearance through the time of the recording. In order to analyze a recording it is crucial to know the nature of the recording in order to choose adequate parameters. What is most important? time or frequency resolution? For birdsong high resolution is required in both domains[12]. There is also the question of how long the windows should be and how they should be constructed. For the initial method, the number of FFT points (NFFT) is set to 8196, the window is set to be of length 100 and the overlap is set to 50. These settings will be used throughout the report when using spectrograms. A normalized Hanning window is used for windowing the signal.

When these settings where used it gave the following results:

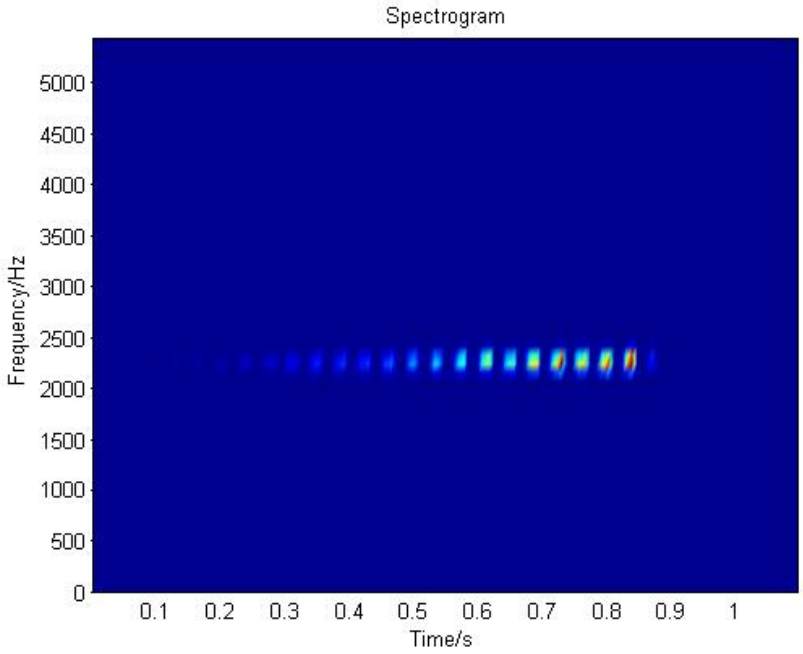

Figure 7: *Example of an owl singing, made with the following settings: Hanning window of length 100, windows overlapping with 50 steps, NFFT = 8196.*

The spectrogram itself is obviously quite complicated to use for any algorithm, since two different birds of the same species can sing very differently. So the first idea was to use the strongest frequency of the song but this proved to be insufficient. The second idea was to make a Fourier transformation along the time axis of the spectrogram. This transforms the spectrogram to a frequency-frequency domain (Doppler domain) and the idea behind this is to extract information about the birds song and how the different

[11] In the discrete it holds all information as long as the parameters are set to adequate values.

[12] This is an initial assumption

frequencies are produced in time to some sort of rhythmic representation, displaying in what repetitious matter the frequencies are produced and thus ignoring what frequency comes first and so on. The method applied is the Doppler function 1.

This creates an image of how the signal is in the so called Doppler-distribution [6] where the y-axis is kept as frequency and the x-axis is transformed into another frequency denoted as ν.

To demonstrate how this works a simple example is given below. A signal is constructed as repetitions of a sinusoid in a fixed frequency under a "Gaussian filter" (bell shaped window). First is a plot of the signal, then comes the spectrogram and finally the freq-freq plot (Doppler domain plot) of the signal.

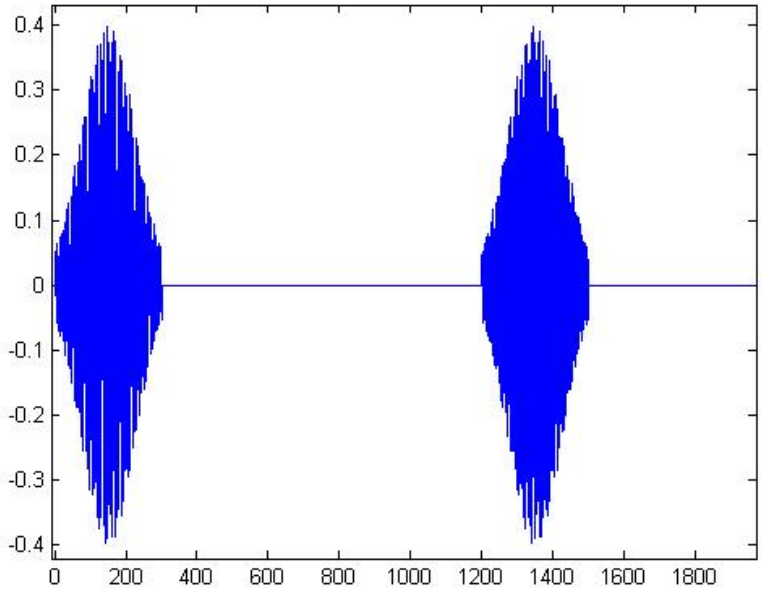

Figure 8: *Zoomed in portion of the constructed sinusoid.*

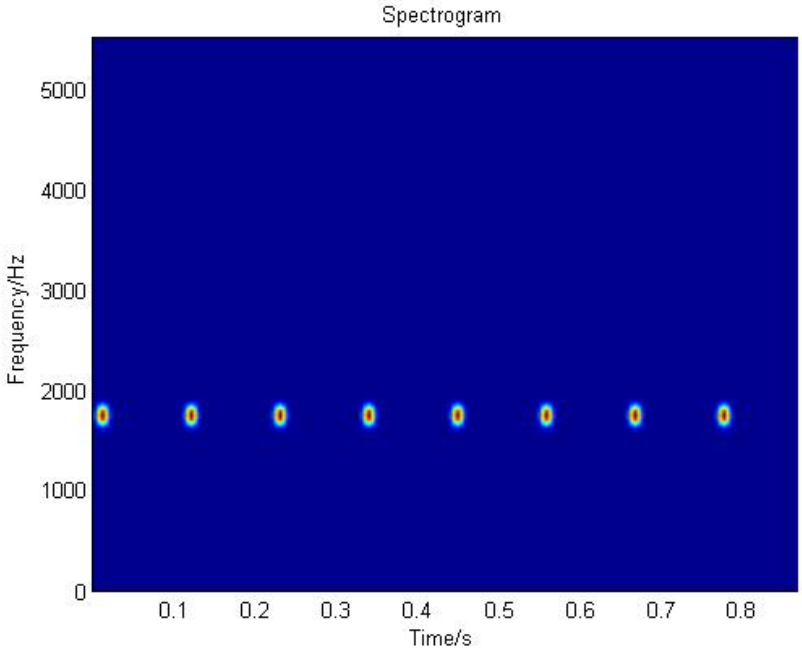

Figure 9: *Spectrogram of the constructed sinusoid.*

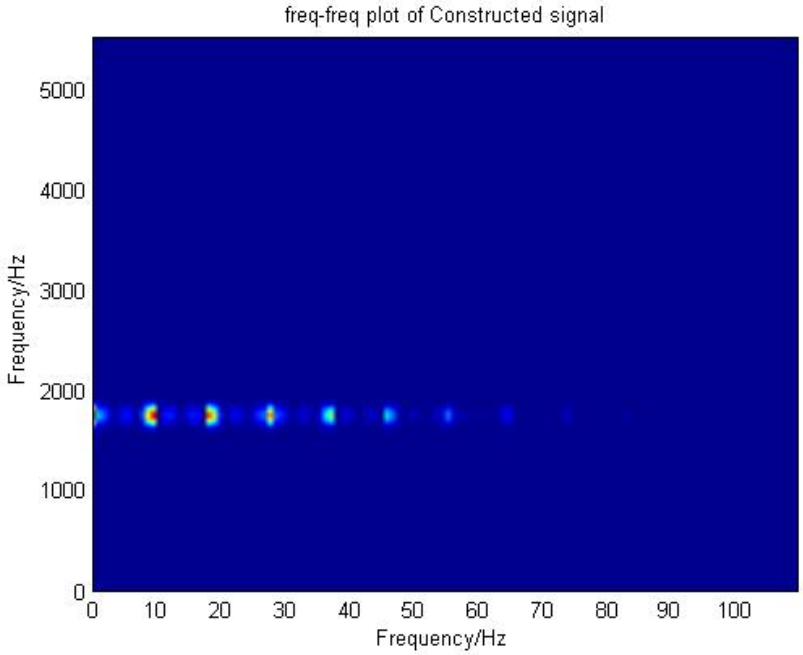

Figure 10: *Frequency transformation of the spectrogram.*

In order to demonstrate how this domain functions and how different signals relate to each-other a similar signal is constructed. This signal has twice as fast (and as many) repetitions corresponding to double the frequency of the first one. This will show how different Doppler plots relate to each-other.

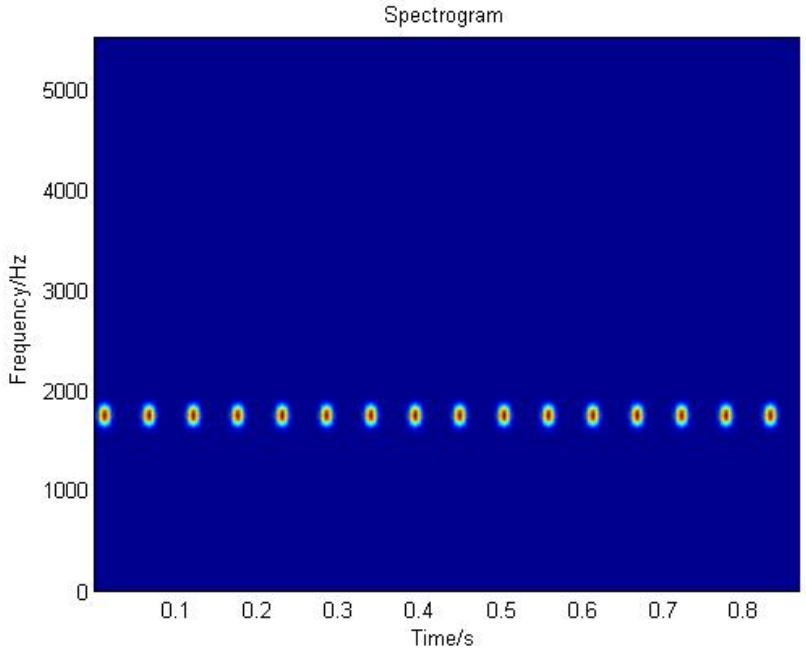

Figure 11: *Spectrogram of the constructed sinusoid. The same as the previous sinusoid but with doubly fast repetitions*

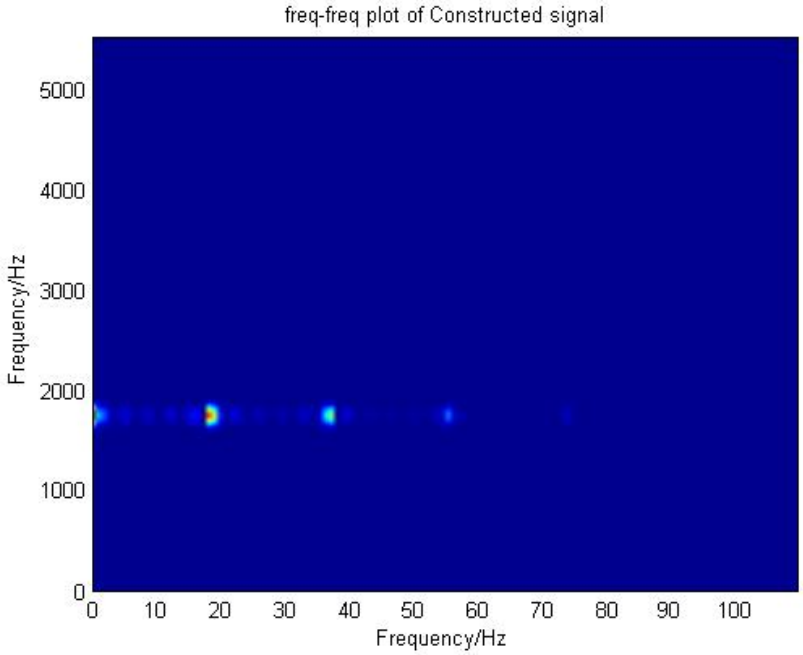

Figure 12: *Frequency transformation of the spectrogram. Same as the previous sinusoid but with doubly fast repetitions.*

These plots clearly illustrates the intention of the method and even though there are "overtones" and other things that could be filtered away, it indicates good possible usage. The reason for comparing figure 12 and figure 10 is to reveal to the relationship of the frequencies in the Doppler domain. The signals in comparison are equal except in one regard which is that in the second example the tone resonates more frequently. This gives a higher frequency along the x-axis in the Doppler domain. If two signals are similar in tone and structure they will have high values (dots) in similar places. One problem that arises during the testing of this method is that the output matrices from the "transformation" is of too high resolution to be able to work with them effectively, so the image is down-sampled three times[13] with a moving average filter with length two. This is described by the following formula.

$$F_{filt}(\nu, f) = \frac{F(\nu, f) + F(\nu + 1, f)}{2} \tag{4}$$

Maintaining the frequency on the x-axis (the newly transformed frequency axis) and halving the resolution on the true frequency (tone) axis. This is done three times which makes the files a lot easier to handle. Here is an example of how the file looks before and after this reduction.

[13] down-sampled three times after each-other. $y = filt(filt(filt(x)))$

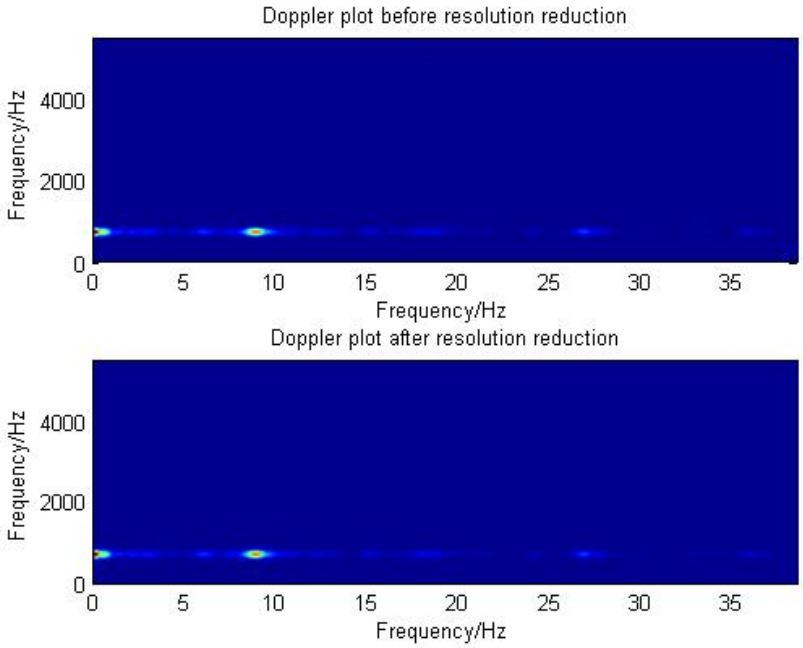

Figure 13: *Shows the decimation of the frequency-frequency plots. The recording used in this example is of a Tengmalm Owl (recording number 3).*

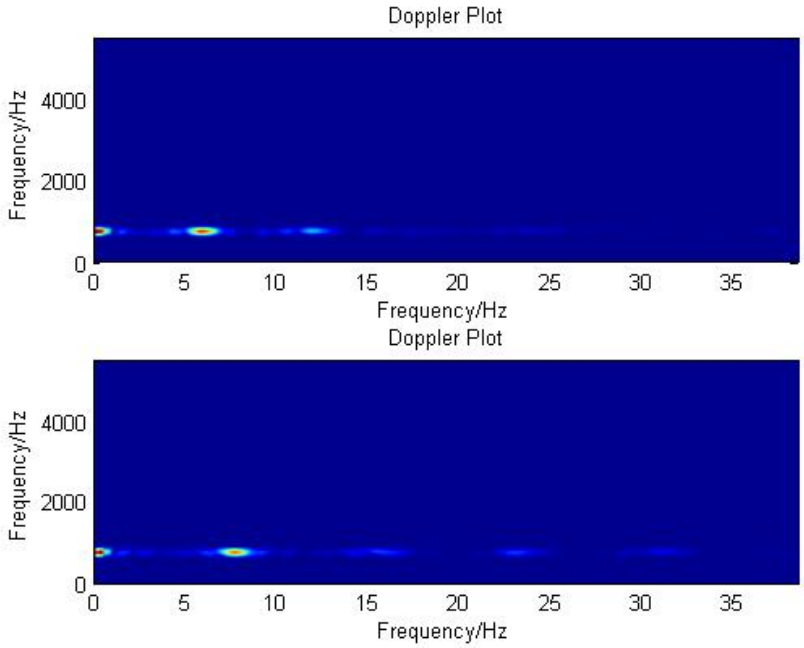

Figure 14: *Shows two other Tengmalm owls in the frequency-frequency domain.*

When analyzing the Doppler-domain plot, one thing comes to mind and it is the dot at $\nu = 0$ (the x-axis) for each frequency corresponding to either a frequency/tone that is not repeated throughout the song or something that has to do with an "error" in the transformation. If it is the latter it should be removed in order to improve the algorithm.

When considering the examples of the constructed signals this dot in the Doppler domain should not be there and is thus unwanted. It is created (correctly) since the input signal to the Fourier transform is a row in the spectrogram (the distribution of frequency/tone throughout time) and does only contain values above zero. So the output is correct but in order to see what frequency (ν) the tones (f) really have it is better to modify this so that the input signal to the Fourier transformation is zero-mean. With this slight modification the formula becomes:

$$F(\nu, f) = \sum_{t1}(S(t_1, f) - \frac{1}{n}\sum_{t} S(t, f))e^{-i2\pi\nu t_1}$$

where both summation sum over all elements along the time axis for that specific frequency which makes the second summation equivalent to removing the mean. Below is a plot of the signal that was used in a previous example. It is a sinusoid that was constructed and it has twice as fast repetitions as the other constructed sinusoid.

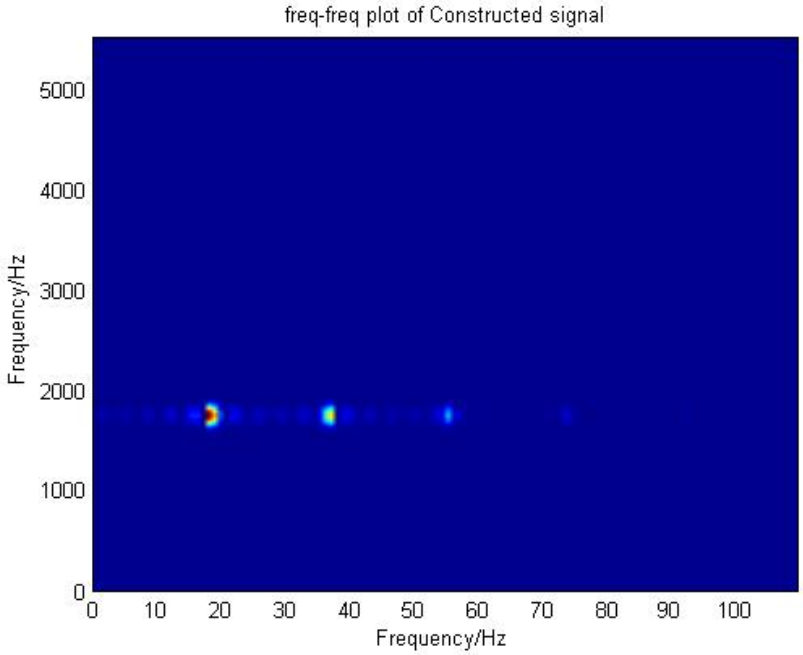

Figure 15: *The Doppler domain plot of the constructed signal (with doubly fast repetitions).*

Comparing this to the previous Doppler domain plot of the constructed signal (the one with quicker repetitions) we can see that the improvement is quite significant. Now the result is free of this artifact at $\nu = 0$.

In order to use this for determining species, the information of the Doppler plot has to be extract somehow. This is done through processing the image. The simplest way is to create some form of contrasted image.This contrasted image will be referred to as the *key*. The function that is created for this purpose, sets the value of a pixel to one if it is above a chosen threshold and to zero otherwise. In order to make this adjustable, the threshold is chosen as some value times the average of the entire pixel-map.

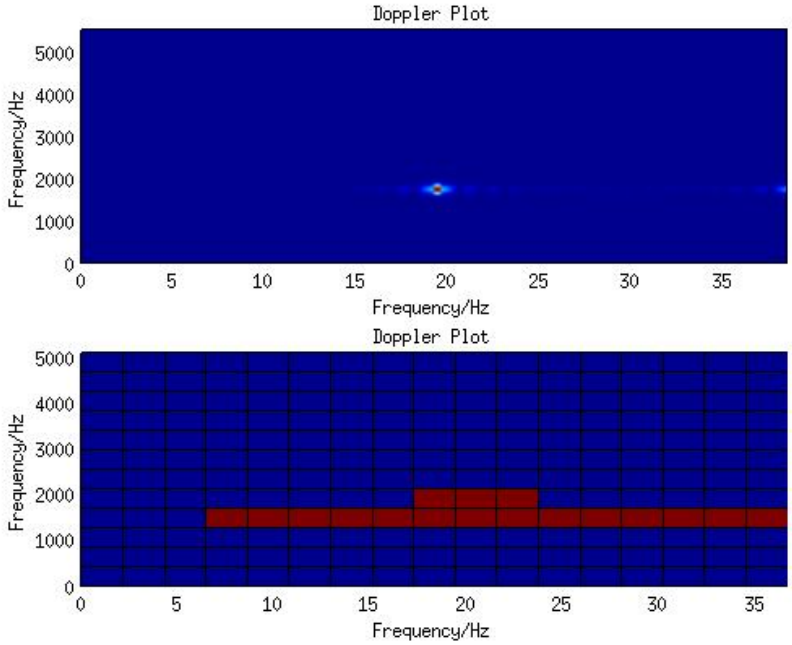

Figure 16: *The Doppler plot and the key that was generated to its image. The source recording is the constructed signal. The plot was cut off before processing in order to reduce the size of the matrices corresponding to the "keys". It was concluded that most of the relevant information in the Doppler plot was below* 40 *Hz along the x-axis, so that is where the image was cut.*

By studying the plot above it can be concluded that not much of the image remains. The resolution was chosen through testing and evaluating all possible combinations[14] of box (pixel) width and height. The factor that is used to choose the threshold is also tested and it was found that a somewhat higher threshold was preferred[15]. What does remain is a coarse image that resembles the Doppler plot. What remains is a clotted reconstruction of the main component of the signal and some of the ringing that can be seen around this component, as light waves along the x-axis at $y \approx 1600$. Even though we might suggest that this method should be improved through finding a better threshold (perhaps even higher through increasing the limit of the testing) or that we somehow manually should choose the block-size, it is a crude method that works for now.

3.3.1 Results of the second approach

Evaluating the settings is done through removing the last three recordings of each species and then creating an averaged key from each species (with the remaining recordings). Then a key is created for each of the test recordings (the ones removed

[14] Of course with some restrictions, Height:[2:2:40] , Width [1:10]
[15] Factor:[1:.1:2.5] and the result was 1.9

from the groups)[16]. These keys are tested against all the species keys and the one that has the highest match (measured through checking if their pixel-maps/matrices are the same) is selected as the best fit. When this is done the result is that the best fit is found correctly in 25/60 cases and that in 8 cases best fit was found for the species other pair[17]. Below is the table showing the results. The last row is the number of times the best fit was found in the other species in the pair, see list page 5.

First test score:
Acrocephalus arundinaceus: 1
Acrocephalus scirpaceus : 0
Aegolius funereus : 3
Corvus corone : 1
Corvus frugilegus : 0
Fringilla coelebs : 0
Fringilla montifringilla : 2
Gavia arctica : 2
Gavia stellata : 3
Locustella fluviatilis : 1
Locustella naevia : 2
Luscinia luscinia : 3
Luscinia megarhynchos : 2
Numenius arquata : 1
Numenius phaeopus : 2
Parus caeruleus : 1
Parus major : 1
Passer domesticus : 0
Passer montanus : 0
Strix aluco : 0
Pair : 8

If the species average keys are studied the reason for some of the results can be found.

[16]This is done to make sure that the key does not contain any information originating from the recording that is being analyzed.

[17]The database consist of pairs of species, two owls, two warblers etc

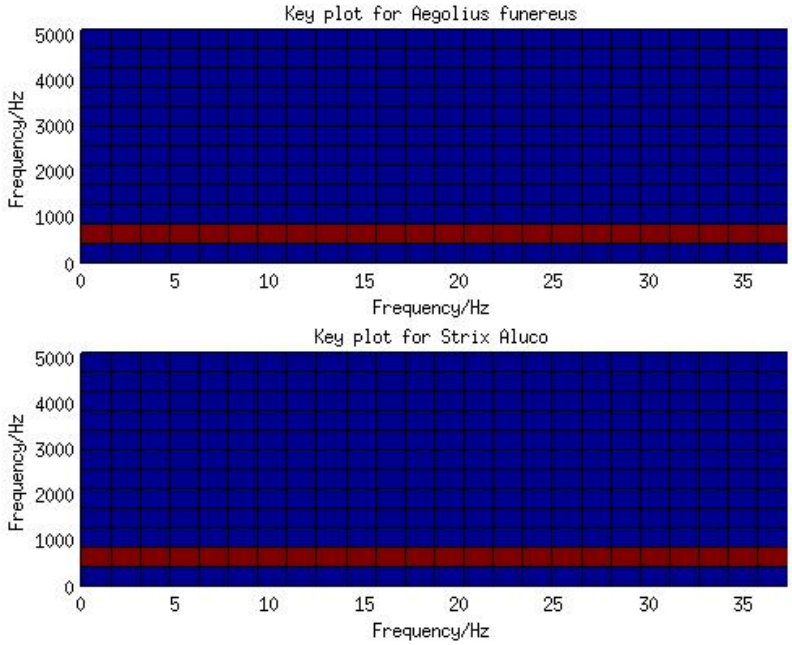

Figure 17: *The plot of the keys for the pair of Owl species.*

As we can see in the table above only the first species actually got any guesses right even though they have the same key. Examining the keys for the test recordings they are very much the same. The script that compares different keys in order to find the best match simply chooses the first one if two species are tied for a match. This gives the conclusion that when this algorithm is indifferent between the species in a pair it will always guess on the one appearing first (from the top) in the list. This gives a slightly more satisfying conclusion from the results above and that is: some of the pair guesses (8/60) could just as well have been the the right guesses (if some sort of other criteria would have been applied).

3.3.2 Improvements & results

While not making a method that can deal with the problem of two species having exactly the same Doppler images, other improvements can quite easily be implemented. One problem with the previous method is that for some signals the threshold that was used to contrast the images was too low and for others it was too high. Even though it was adaptive in the way that it was correlated to the mean of the pixel-map it just was not smart enough. In order to make the contrasting better a more adaptive method is developed. Now a much higher threshold is tried first, 30 times the mean) and if this results in that less than 5 boxes receive values above zero, the threshold is lowered by a factor .9 until at least 5 elements pass the threshold. Also instead of just setting the element to zero the value is kept if it passes the threshold. This allows for the possibility

to sort the elements and just keeping those that are more "important"[18] in that way. This is done and the Doppler plot that has to be identified is allowed to keep a maximum of ten elements. Then the score of the match[19] is calculated through the calculating the shortest distance between a point in our unknown (regarding the species) matrix and any nonzero element in the species-key matrix. More formally

$$score = 1 - sum(dist)/\sqrt{(m * n) * N};$$

where $dist$ is the shortest distance between elements, m is the number of rows, n is the number of columns and N is the number of elements compared.

The result of a familiar recording is displayed below.

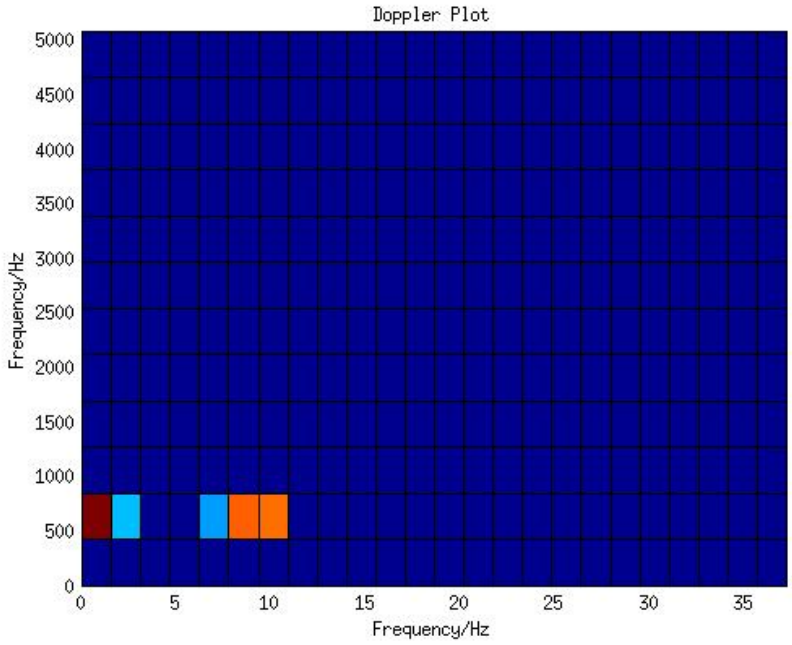

Figure 18: *The improved image of a Tengmalm owl recording number 3.*

If this method is applied and tried for all recordings the following results are achieved:

Acrocephalus arundinaceus: 0
Acrocephalus scirpaceus : 6
Aegolius funereus : 12
Corvus corone : 10
Corvus frugilegus : 2
Fringilla coelebs : 9

[18]There is no way of knowing if they are more important per se with this method but it's the only way of measuring which peaks are stronger than others.

[19]How well the two plots that are being compared, match each other

Fringilla montifringilla : 7
Gavia arctica : 4
Gavia stellata : 1
Locustella fluviatilis : 2
Locustella naevia : 6
Luscinia luscinia : 10
Luscinia megarhynchos : 0
Numenius arquata : 1
Numenius phaeopus : 2
Parus caeruleus : 6
Parus major : 1
Passer domesticus : 0
Passer montanus : 0
Strix aluco : 12
Pair : 32

Keep in mind that now there is a total of 308 recordings and that some of them are of bad quality and should possibly be discarded. However the result is a crude indication of the possibilities of this method and the average is 0.3994, so roughly 40 percent could be determined this way.

3.4 Third approach

In the search for better methods a recent paper is taken into consideration. In the paper [2], several methods for analyzing bird song are proposed and applied for characterization of a great reed warbler.

Syllable Analysis

The first part to investigate is syllable analysis which is in section 4 in the paper mentioned above. The idea behind this is that birdsong can be considered in a similar way to how human speech is considered meaning that speech and birdsong is constructed out of a set of syllables which put together gives different words or strophes. The method suggested, basically finds the peaks of the squared signal (where the signal has gone through some filtering mainly to reduce the noise level) through finding those parts that exceed a threshold. The filtering consist of the same decimation as before, where the signals are decimated three times with the decimation described in 4 and then a bandpass filter is applied with cutoffs at 150 Hz and 6000 Hz. The key to the method for finding the peaks, is that the threshold is varying and is a moving average of a number of samples around the point that is to be evaluated. A full description of this method can be found in the paper, [2]. It is implemented as a function called `syllablecut` which was received directly from one of the authors of the article.

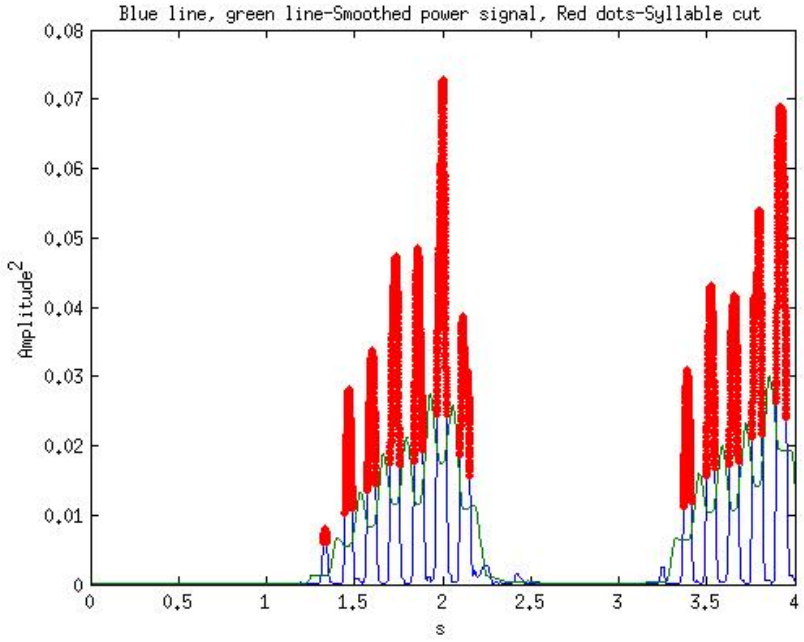

Figure 19: *This is the syllable cut function at work and we can see the blue line as the squared original signal and the green line is the smoothed signal giving a sort of moving average and the red dots are the peaks that were found.*

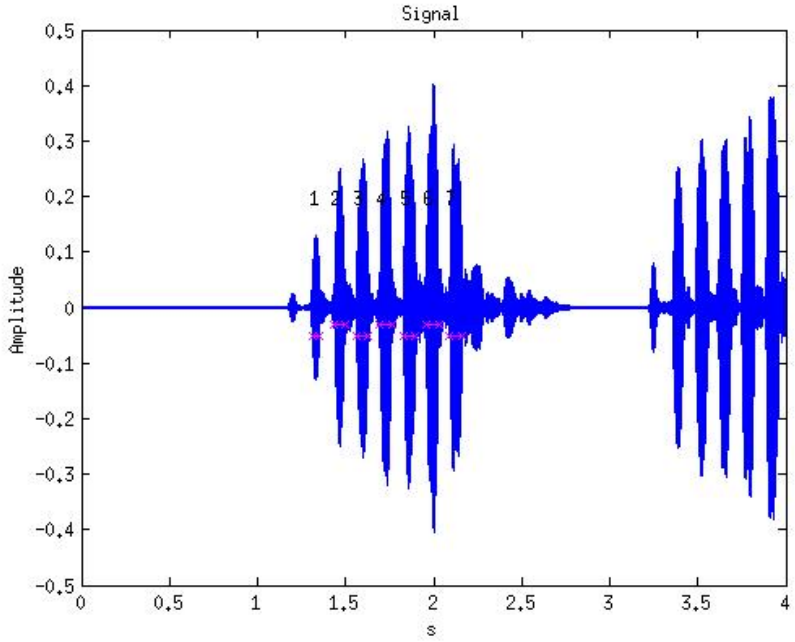

Figure 20: *An overview of the result of the syllable cut function*

As mentioned before this function operates under several parameters which creates a wide range of possibilities regarding its performance. Since the purpose of this project was to study a large and varying data set, these parameters where set to give as robust a function as possible[20].

Characterization

Once the syllables had been extracted the problem of characterization was attended to. Similar to the article, a wide range of time-series tools were applied in order to classify the recordings. The spectrum, spectrogram, Doppler domain and Ambiguity domain plots were evaluated for all the recordings. Just as in section 3.3 some method for extracting the data from these plots had to be considered. In [2], the SVDs are calculated and the first two left and right singular vectors are kept. The same method was applied in this approach.
In order to find a match the entire database of all syllables (except for the ones in the recording to be analyzed) were considered. For each such comparison a difference was calculated and it was this value that determined how well the syllables matched. The difference was calculated as

[20] `minsp=100, maxsp=200, extth=10, lev=15`

$$
\begin{aligned}
D_k = & \sum_{j=1}^{n} (L_{1a}(j) - L_{1r}(j))^2 + \sum_{j=1}^{n} (R_{1a}(j) - R_{1r}(j))^2 + \\
& \sum_{j=1}^{n} (L_{2a}(j) - L_{2r}(j))^2 + \sum_{j=1}^{n} (R_{2a}(j) - R_{2r}(j))^2
\end{aligned} \tag{5}
$$

where L is the left singular vector and R is the right singular vector. The notation R_{1r} notes that is the first right singular vector of the analysis syllable and j notes the index in the vector which is summarized. Simply this calculates the squared sum difference between the singular vectors and a smaller value indicates higher similarity.
These calculations were done for the spectrogram, the spectrum, the Ambiguity domain and the Doppler domain.

3.4.1 Results of the third approach

The time needed to calculate all of the SVDs for all syllables (close to 3000) and to then compare these, was around 14 hours[21]. This makes adjustments in methods etc. tedious so keep that in mind[22]. In order to illustrate the result a matrix is plotted. What the matrix represents is what species a recording is identified as. An identification is done as the species that gave the most matches for its syllables against the recording that is being analyzed. On the columns of the matrix are the guesses of each species. So in column 3 there is a dark red dot in row three and i lighter blue in the final row, number 20. This means that in most guesses for the recordings of species number three (Tengelmans Owl) the algorithm finds the correct species. In some cases it guessed on species number 20 (Tawny Owl). Except for the Owl pair the other pairs appear pairwise meaning that a guess in the neighboring column could be due to the similarity of the bird-pair. Before the matrix is presented each row is weighted so that the rows sum to one.

[21] Core 2 Duo @ 2.9 GHz with 6GB of RAM

[22] At first a smaller subset was used to verify the model, but what was true for the subset was not true for the entire set. This is always problematic and in this case it was very limiting regarding fine-tuning of the algorithm.

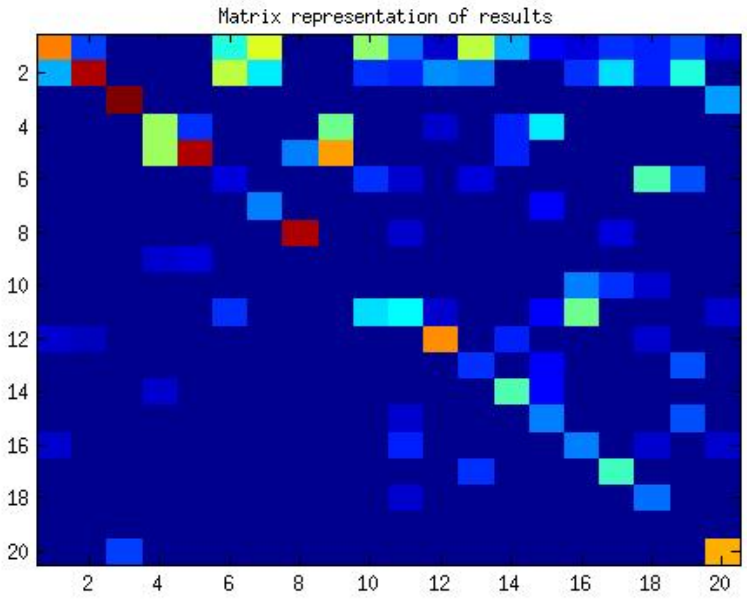

Figure 21: *The results of the third method. Each column represents the recordings of a species and the row position within each column shows what species gave the best match for each recording. The diagonal gives the correct answer.*

In the plot there appears to be a bias or clustering in the first rows. The reason for this is that sometimes the algorithm is not sure and finds two species equally probable. When this happens it will always go with the first choice creating a bias in the upper right half of the matrix.
A cheap solution is to remove these recordings from the results saying: if we are not sure, we are not guessing at all. This gives a slightly more satisfying result.

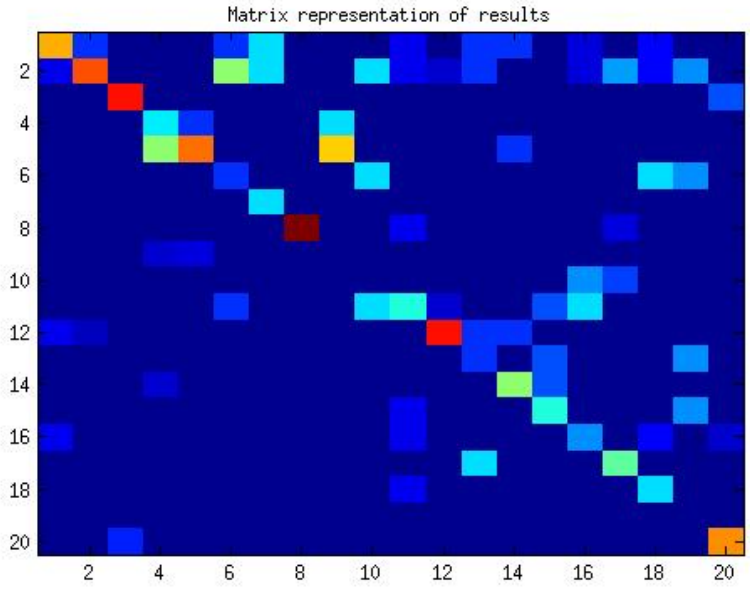

Figure 22: *The results of the third method after a slight modification. Each column represents the recordings of a species and the row position within each column shows what species gave the best match for each recording. The diagonal gives the correct answer.*

Regarding both the graphical representations 22 and the quantification of the results (in the table below) we need to keep some things in mind. For some recordings it is not possible to find any syllables[23] and for some species this was more of a problem resulting in few recordings/syllables to compare. Also for some species there were very few recordings. As an example species number nine, which gave no correct guesses at all, only had five recordings to use as reference and to analyze.

Since there was a varying number of recordings for each species it was considered best to compare the ratios of how often the algorithm guessed correctly instead of the actual number of times.

[23]Two possible causes for this directly comes to mind: either the recording is corrupt or bad in some way or the parameters used in the syllable cut function were not adequate for this type of song.

Table of results

0	0.7000	0.1579
0.1000	0.7895	0
0	0.8462	0
0	0.3571	0.1667
0.5000	0.7500	0
0	0.1667	0
0	0.3333	0
0	1.0000	0
0	0	0
0	0	0
0.3333	0.4000	0.0769
0	0.8462	0.1667
0	0.1667	0
0	0.5000	0.2000
0	0.4000	0
0	0.2500	0
0	0.4545	0
0	0.3333	0
0	0	0
0	0.7333	0

In the mid row is the ratio of how often the correct species was found. In the left column is the ratio of the species aligned to the left in the matrix and on the right is the one aligned to the right in the matrix.

4 Discussion

This section will serve as discussion for the entire project and hence it will be quite substantial. Because of this I have chose to divide it into subsections. There will be one subsection for each part of the project that I feel needs to be dealt with separately and also for other fields of interest.

4.1 Three Approaches

The First Approach

In the first attempt to characterize the recordings the simplest idea was applied which was to determine the species through analyzing only the frequency content of the recordings.

This simple approach obviously failed since the birds song is not consistent in frequency (inter-species). It is possible that one birds song is more consistent and that this could be used to track individuals but for determining species its not sophisticated enough.

The Second Approach

Spectrogram analysis which led to analysis of the Doppler domain proved a bit more resourceful and for quite an unsophisticated method I actually think it did quite good. The bird song is too varying for this method but it could be applicable in other fields where the signals that needs to be categorized are more consistent. One suggestion is for categorizing music. Much of the information about a song (at least a modern pop song) can be contained in the Doppler domain of the signal. An image of the upper right half-plane can contain information about the speed of the song. The BPM (beats per minute) will surely be shown as the major frequency of the lower register, while the frequency content in higher registers will give more information about the rest of the song.

The Third Approach

The final model was build upon cutting the song in a recording into syllables which then were analyzed in both the spectrogram, the Doppler domain and the Ambiguity domain.

After some heavy limitations and adjustments, the final method actually works satisfactorily. For some species it gives great results and for others it gives less good to bad results. This may be due to issues with the recordings which causes lack of samples for the reference database, as will be discussed later. This was definitely the most promising and sophisticated method. I believe that dividing signals such as song or speech into syllables is both logically sound and effective and that some of the theory for speech recognition might work well for determining bird song etc.

One such field of theory might be Cepstrum analysis. I considered it and looked into it but had to leave it behind. One fear that I had was that for it to be effective it would require high resolution and this would complicate my handling of the database.

4.2 Improvements

Data Selection and filtering

This is an area with obvious possibilities for improvements. I did no check at all on the data that was collected other than verifying that there was indeed a bird singing on the recording. During the project I noticed that in some recordings there were actually two different types of birds singing. In one case the algorithm had actually identified the bird singing in the background correctly (at least according to my layman ear). What should be done is of course to go through all the recordings that make the reference database in order to be sure that they are of the best possible quality and that they only contain song from the bird in question.
Also the signals could use a more adapted filter. All signals where filtered before looked upon[24] with a simple bandpass filter cutting of both lower and higher frequencies in order to remove noise.

One reason for not going through the data more thoroughly and removing bad or noisy signals was that I had the ambition to make an algorithm that could work through the bad recordings and hence making it more realistic and useful. In reality those ornithologists that use good recording devices probably already know what species it is that they are recording and so the algorithm would most likely be used by amateurs with microphones of cellphone quality.

Syllable cutting

While it is possible to find parameters for this method that fits many types of bird song it would be better to make a more thorough analysis of this field. If a technical definition of a syllable could be made it is possible that an adaptive method for finding the "true" syllables could be made.

Also I find that some of the sounds that the birds make that were characterized as syllables should possibly be left out in order to improve the performance of the characterization. Further it is a possibility that the order of some syllables could be unique to species or individuals and this could be another part to study.

Results illustration

One thing that could prove applicable in a program could be to show not only the species that was found to be the most probable for the observation, but the three or five most probable. Then the user could be provided with pictures and sound samples and the algorithm would be more of an aid than a device for determining. If the accuracy is consistent for all species (after some improvements) this might very well be an approach that could make the algorithm sufficient for an App or a pc-program.

4.3 Conclusion

Modelling bird song with the spectral analysis tools used in this report, is definitely possible. While I did not fully succeed in terms of my ambition, I feel that I have thoroughly narrowed done some of the possibilities in this field. In order to make this

[24] In the second and third approach.

algorithm[25] useful more time has to be put into some vital areas. First of all is the data selection and validation. Better and correct data for the reference recordings is the fundamental basis for this algorithm to work in practice. Second is the syllable cutting. This is a very interesting field and it is possible that another method could be better for this purpose, but the method that I used should at-least be improved and better applied[26] to this area. Third and last I believe that better implementations of the functions that I have written needs to be considered. Since this kind of algorithm needs to function very well in setting with loads of large vectors and matrices good organization is key. I solved this problem through organizing the material in structures and cells, but still struggled. For some scripts there was just too many loops, variables and indices. Therefore I encourage the next person that undertake such a program to be extremely organized and to think through a structure for these algorithms to work.

References

[1] Cooley J.W. & Turkey; J.W. (1965). *An algorithm for the machine calculation of complex Fourier series*. Maths Comput. **19**, 297.

[2] Sandsten M., Tarka M., Caissy-Martineau J., Hansson B., et al. (2011). *A SVD-based classification of bird singing in different time-frequency domains using multitapers*. European Signal Processing Conference, 2011,, 966 - 970. 19th European SIgnal Processing Conference (EUSIPCO-2011). Barcelona: European Association for Signal Processing (EURASIP) URL: http://www.eurasip.org/Proceedings/Eusipco/Eusipco2011/papers/1569422961.pdf.

[3] Chu W. & Blumstein D.T. (2011). *Noise robust bird song detection using syllable pattern-based hidden Markov models*. Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on, 345-348, Prague.

[4] Graciarena M., Delplanche M., Shriberg E. & Stolcke A. (2011). *Bird species recognition combining acoustic and sequence modeling*. Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on, 341 - 344, Prague.

[5] Heideman M.T., Johnson D.H. & Burrus C.S. (1985). *Gauss and the history of the fast Fourier transform*. Archive for History of Exact Sciences,1985,,265-277.

[6] Maria S. (2013). *Time-frequency analysis of non-stationary processes - An introduction*. Compendium.

[25] The third approach

[26] Meaning that the parameters of the function should be optimized and set to better fit birdsong when handling different types of birds. One suggestion is to make an adaptive method.