Audio Signal Analysis

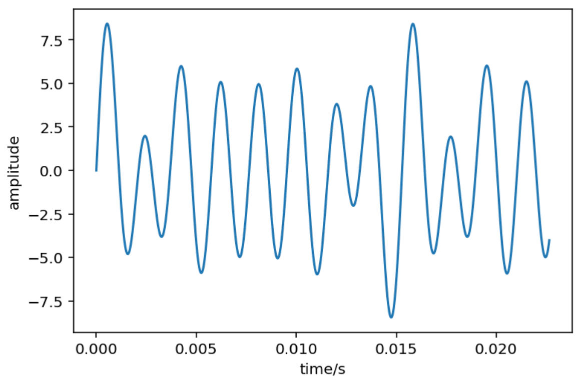
Introduction

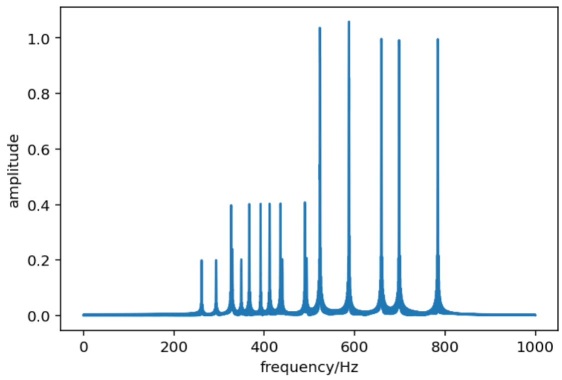
Animals often communicate through vocalisation, using their unique calls to warn others of danger, as a mating ritual or to convey information within their community such as about food sources. A particular area of interest is the study of the elaborate vocalisation songbirds, the pattern of sounds of which can be used to identify an individual species or even animal. In this project you will apply signal processing techniques to determine the frequencies in an audio signal, and generate a signature which may be used for bird song identification.

Background

Most of the sounds we hear are not associated with any single frequency, but comprise a superposition of many waves with a wide range of frequencies and amplitudes.

A Fourier transform is a technique used to decompose a signal waveform into its constituent frequencies. The diagram on the left shows an audio signal waveform and the diagram on the right is the same wave after transforming to the frequency domain, showing the constituent frequencies and the magnitude of their respective contribution to the overall wave.



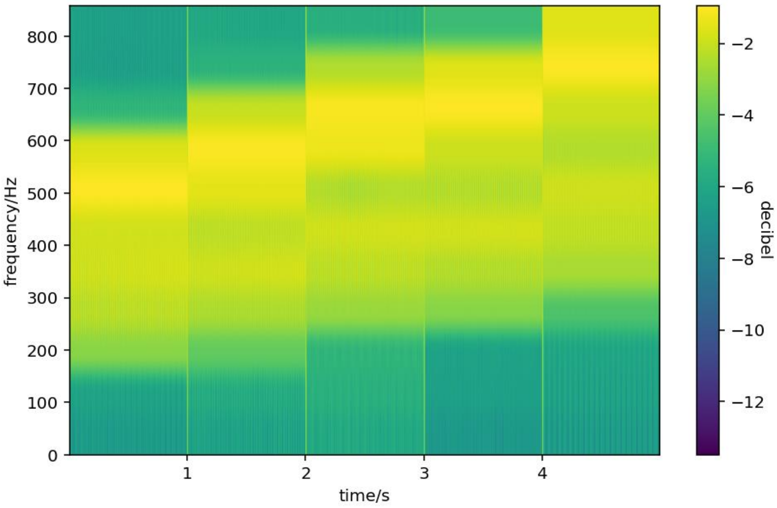


The common implementation of Fourier transform in computer program is the Fast Fourier Transform (FFT), which is an algorithm to calculate the Fourier transform of a discrete sequence. For a signal in time $x[t]$ with a set of N values, $\{x[0], x[1], \ldots, x[N-1]\}$, the corresponding frequency spectrum $\{X[0], X[1], \ldots, X[N-1]\}$ is calculated as follows:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-i\omega_0 kn}, \; k = 0, \, 1, \ldots, \, N-1$$

where the fundamental frequency, $\omega_0 = \frac{2\pi}{N}$.

Applying FFT to a signal extracts information on the frequency spectrum but loses information on the time order of these frequencies. This can be overcome by splitting the signal into discrete 'windows' and applying the FFT to each window separately. The figure below shows the frequency spectrum of a 5 second signal split into 1 second windows.



The Project

The goal is to develop an algorithm that, given a recording of a song bird, is able to identify the species. Example recordings of bird song can be found here:

[Bird Song Identifier | Audio of Garden Bird Calls - The RSPB]

First work out how to read audio data into Python using a library such as [Pydub], then apply a Fourier transform using [scipy.fft]. You might like to use `IPython.display.Audio` to render an audio player in your notebook.

Once you have the Fourier transform, you should determine the dominant frequencies. Comparing these frequencies to those of a set of 'reference' bird song recordings could be a way to identify the species.

You will need to use Numpy arrays to store and manipulate the data – see Tutorials 8 and 9 for background material.

Notes

- Implementing an algorithm that correctly identifies a bird species from a recording is a **very difficult** problem and you are unlikely to solve it in the duration of this project.
- Start with a much simpler problem, such as identifying the frequencies in a recording of a chord played on a piano, or the notes in the opening chord of [A Hard Day's Night].
- Move on to the problem of identifying the notes in a single-voice melody line.
- You can filter out some frequencies in your initial audio track to remove background noises or some sections which are not needed for analysis.

Extensions

- The techniques developed here could be used for identifying other types of audio signal – human voices, for example.