IMPLEMENTATION OF FREQUENCY SCALING ALGORITHIMS USING SPEECH ANALYSIS

Group Name: Matlads & Ro.

Tomás Smyth	(15187535) - tomas.smyth.2016@mumail.ie
Nathan Phipps O'Neill	(16304503) - nathan.phippsoneill.2017@mumail.ie
Mark Gaffney	(12721095) - mark.gaffney.2013@mumail.ie
Jordan O' Donnell	(16371101) - jordan.odonnell.2017@mumail.ie
Róisín Keating	(16412122) - roisin.nicheitinn.2017@mumail.ie
Aaron Nugent	(16460102) - aaron.nugent.2017@mumail.ie

Project Supervisor: Dr. Bob Lawlor

December 2019

Department of Electronic Engineering

Maynooth University

Abstract

The brief for this project is to research audio signals and apply signal processing techniques for a specific implementation. This report exhibits the development of the team's ability to understand and apply signal analysis tools to achieve a change in the tone of a person's voice.

Contents

Table of Figures

Table of Equations

1 Introduction

1.1 History of Speech Analysis

Signal processing refers to the field of study concerned with the analysis, synthesis and modification of signals. In the brief of this project, speech analysis refers to the process of analysing voice and speech patterns. Speech processing is comprised of a broad and varied set of techniques. These techniques include the acquisition, manipulation, storage, transfer and output of speech signals. (Priyanka A. Abhang, 2016)

The first recorded attempt at speech signal processing was conducted in 1952 by three researchers, Stephen Balashek, R.Biddulph and K.H Davis at Bell Labs. They developed a system which could recognise digits spoken by a single speaker, evident in *Figure 1*. Other attempts in the same era were more focused on understanding phonetic elements such as vowels and consonants. (Katz, 2018)

In 1992, Lawrence Rabiner created a technology at Bell Labs that was adopted by AT&T in their 'Voice Recognition Call Processing Service' that routed phone calls without a human operator. Since that date, speech analysis has transitioned towards complex neural networks and deep learning for improved performance. (Juang & Rabiner, 2006)

Figure 1 - Automatic Digit Recogniser developed at Bell Labs 1952 (Kincaid, 2018)

1.2 Project Aims & Objectives

The main objective of this project is to research the concept of speech processing and to implement our research by creating a pitch shifting algorithm. Overall, the group plans to implement new and existing knowledge on Digital Signal Processing to create a Voice Changer. Altering the tone, pitch and distortion are viable methods that can help to achieve this. The objective of this report is to record those methods used by the group, in detail.

1.3 Literature review

Throughout this report, the group referenced many scholar articles and websites. Most of the articles helped in researching the history and physics of the human voice, and Fourier Series, Fourier Transforms, Convolution and Filters. The group has some honourable literature mentions for each of the main bodies of this report;

- Application of time-scale and frequency-scale modification algorithms to voice-gender conversion (Jung, 2003)
- The TIMIT database
- The Origin of Speech (Holden, 2004)

1.4 Report Layout

The general layout of this report begins with a section on the history of speech analysis and how it relates to the project objective. This is followed by the crucial main body, which describes the implementation of various mathematical tools and signal analysis techniques to successfully alter the sound characteristics of an audio signal. The report concludes with a results section which investigates the findings of this report and future work that could aid towards the development of perhaps a more robust voice changing system.

2 Background & Theory

2.1 Fourier Series

The Fourier Series is an infinite series of sine and cosine functions. It deconstructs any periodic signal into the weighted sum of circular path functions. These functions create each time point of the signal. (Morrison, 1994) . The compilation and study of Fourier Series is known as harmonic analysis and is used to break up arbitrary periodic functions.

Once the signal is broken up into different frequencies, they can be passed through a filter that measures every possible part of the signal. Each filter must operate individually and independently of each other and must be combinable, time invariant and scalable.

The unit circle as seen in *Figure 2* - Unit Circle

has three properties, frequency (speed), amplitude (size) and phase angle (starting point). The combined position of all cycles is what defines a signal. (Anon., 2017).

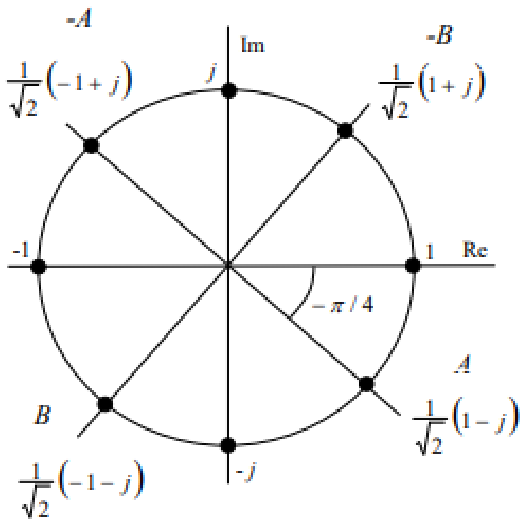

Figure 2 - Unit Circle (Anon., 2016)

2.2 Fourier Transform

The Fourier Transform used on non-periodic signals, works by decomposing a time signal into its component frequencies. These components are single sinusoidal oscillations at distinct frequencies, each with their own amplitude and phase. The output of a Fourier Transform is a complex number, the absolute value of which represents the amount of that frequency present in the input signal. The ensuing equation illustrates the Fourier Transform.

$$H(j\omega) = \int_{-\infty}^{\infty} h(t) e^{-j\omega t} dt$$

Equation 1 - Fourier Transform (Lawlor, 2018)

The amplitude and phase of a sinusoid can be combined into a single complex number called a Fourier coefficient. When the signal is discrete, the coefficients are periodic and are a finite set of complex numbers. The set is called the Discrete Fourier Transforms (DFT). (M.H., 1999).

$$S(f) = \int_{-\infty}^{\infty} s(t)e^{-j2\pi ft}\,dt$$

Equation 2 - Discrete Fourier Transform (Lawlor, 2018)

$$s(t) = \int_{-\infty}^{\infty} S(f)e^{j2\pi ft}\,df$$

Equation 3 - Discrete Inverse Fourier Transform (Lawlor, 2018)

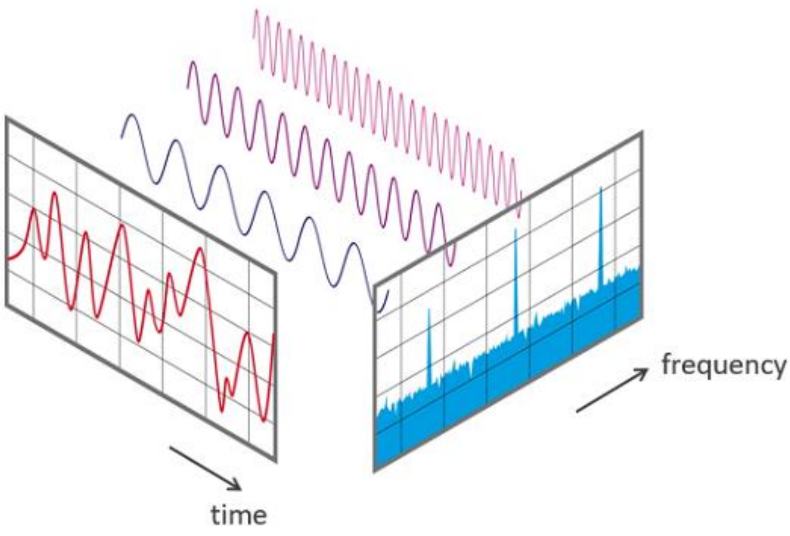

Figure 3 - Transformation from time to frequency domain

The Inverse Fourier Transform makes it possible to recover a function from its Fourier Transform. This allows the wave to be reconstructed, if the frequency and phase is known. (Parissis, 2011)

$$h(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} H(j\omega)e^{j\omega t}\,d\omega$$

Figure 4 - Inverse Fourier Transform (Lawlor, 2018)

2.3 Convolution

Convolution is when two separate signals are combined to form a third new signal, this shows how one function affects the other and can be seen in Figure 5 (Lawlor, 2018). Convolution can be used to filter a signal using another (filter) signal. To convolve two discrete signals, the first signal must be flipped so the first element of that signal is initially convolved with the first element of the second signal. One signal is then passed through the other which remains stationary. As each element of the first signal is convolved over each element of the second signal, the product of these is the new element for the third signal. The length of the third (convolved) signal should be the summation of the length of the first and second signals which were convolved together.

For continuous signals the convolution integral is used, and it is a similar process. It is also helpful that multiplication in the frequency domain is equivalent to convolution in the time domain. This allows the use of the Fourier transform to convert both signals to be convolved into the frequency domain, they can then be multiplied together, inverse Fourier transformed and then the convolved signal is produced as a result.

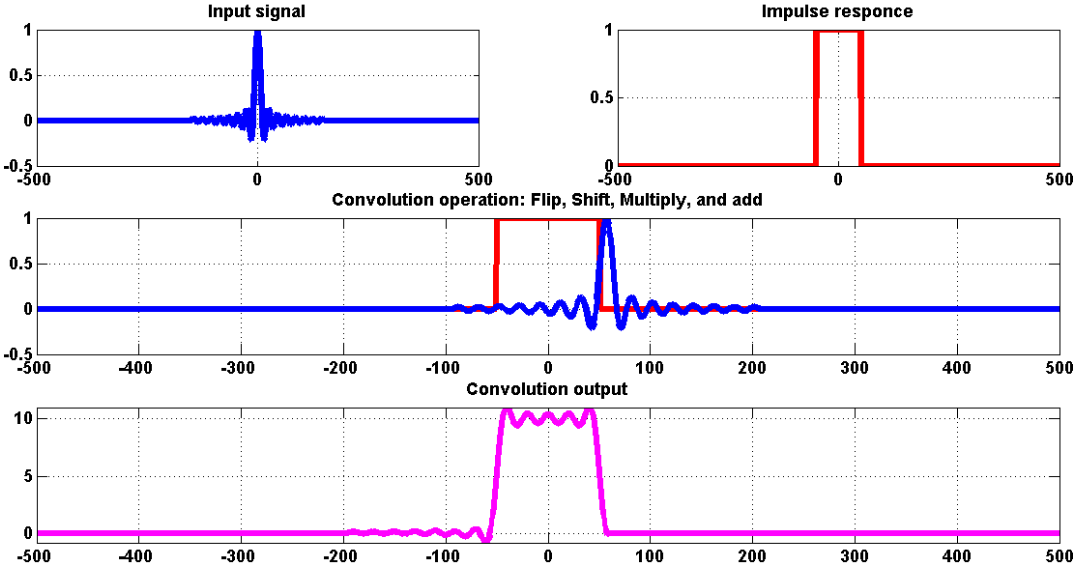

Figure 5 - Demonstration of convolution (Bhurane, 2018)

$$y(t)=x(t)\otimes h(t)=\int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau=\int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau$$

Figure 6 - Continuous time linear convolution integral (Lawlor, 2018)

$$y(n)=x(n)*h(n)=\sum_{m=0}^{n} x(m)h(n-m)=\sum_{m=0}^{n} h(m)x(n-m)$$

Figure 7 - Discrete time finite length convolution integral (Lawlor, 2018)

In this project for example the wave for the audio signal of a male saying the word cat and for a female saying cat, could be combined to create and show how the two signals affect each other.

A point on the discrete wave that passes through the origin is selected and all other time samples are set to zero. The impulse response of this single unit input allows for the understanding of how the signal will react and through this it can be seen how the signal will react to every other time sample.

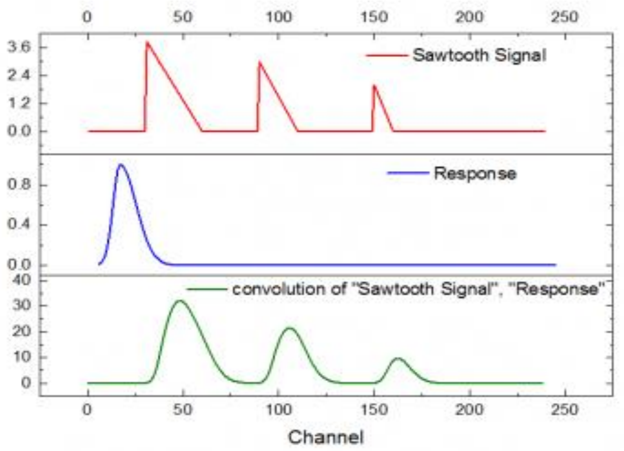

Figure 8 - Convolution of saw tooth signal with response signal (Anon., 2018)

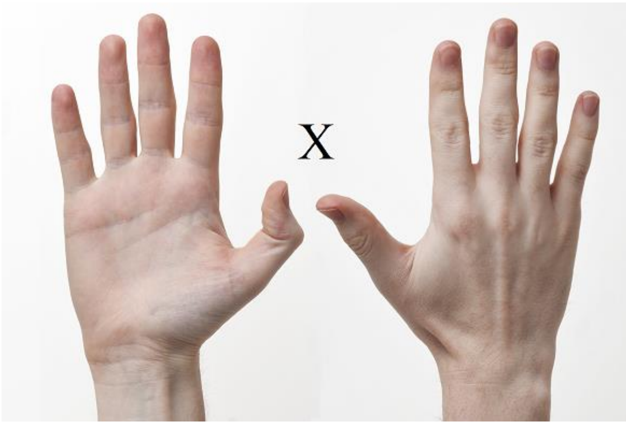

Figure 9 - Convolving Two Signals (Multiplication) (Anon., 2018)

Figure 10 - Result of Convolving Two Signals is One Signal (Santarossa, 2017)

Convolution is used to show the relationship between three signals of interest, the input signal, the impulse response, and the output signal. (Anon., 18)

2.4 Filters

A filter is a device or a process that is used to isolate a specific range of a signal for analysis. It aids to retrieve important parts of data and eliminate unwanted components such as noise. It is used in electronics, audio recording, and television, image processing etc. Filters are classified in several ways which include being: (Herod, 1997)

- Linear or Non-Linear
- Time-Variant or Time-Invariant
- Casual or Not-Casual

- Analog or Digital
- Discrete-Time or Continuous-Time
- Passive or Active type of continuous-time filter
- Infinite Impulsive Response (IIR) or Finite Impulsive Response (FIR) type of discrete-time or digital filter

There are a variety of filtering methods that can be implemented in this project. Some of the more common techniques include:

1. <u>High Pass Filters:</u>

A high pass filter (HPF) allows high frequency signals to pass through while attenuating the lower frequencies.

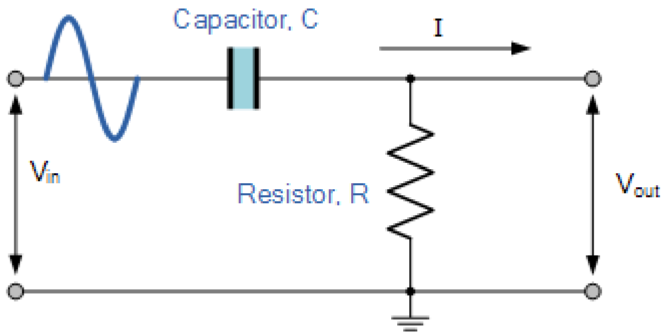

Figure 11 - High-Pass Filter (ElectronicsTutorials.ws, 2018)

The circuit in *Figure 11* implements a capacitor and resistor in series. Voltage is supplied to the circuit across the combination of the capacitor and resistor and the voltage across the resistor is the output. The time constant(τ) is the product of the resistance and capacitance. It is inversely proportional to the cut-off frequency fc, where fc is in hertz, τ is in seconds, R is ohms and C is in farads.

$$fc = \frac{1}{2\pi RC}$$

Equation 4 - Cut off frequency

2. <u>Low Pass Filters:</u>

Contrary, a low pass filter (LPF) is an electronic filter that allows low frequency components of a signal to pass through while attenuating the higher frequencies.

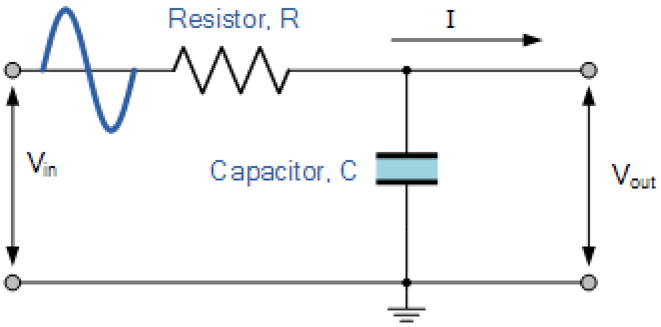

Figure 12 - Low-Pass Filter (ElectronicsTutorials.ws, 2018)

The circuit in *Figure 12* implements a resistor and capacitor in series as opposed to a high pass filter which features a capacitor and resistor in series. Voltage is supplied to the circuit across the combination of the resistor and capacitor in series and the voltage across the resistor is the output.

3. <u>Band Pass Filters:</u>

However, a band pass filter allows frequencies within a certain bandwidth to pass through while attenuating all frequencies outside that threshold.

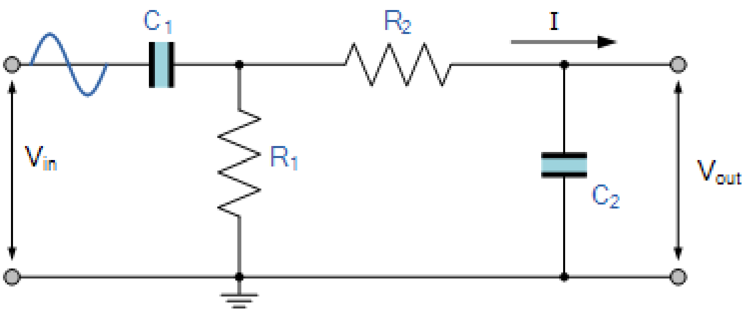

Figure 13 - Band-Pass filter (ElectronicsTutorials.ws, 2018)

This circuit in *Figure 13* implements two capacitors and two resistors unlike the last two circuits previously mentioned. This circuit can be created by simply placing a low pass filter and high pass filter in series. Voltage is supplied to the circuit across the combination of the resistors and capacitors and the voltage across the resistors is the output. (Anon., 2018)

2.5 History of the Human Voice

There is a gap of five million years between the evolution processes of human animal communication. There is much debate about how the human voice evolved so that it had the ability to produce speech. There are two main theories that incorporate the evolution of speech. Research shows that roughly two million years ago, the human brain underwent a rapid evolution process. During this evolution of the brain, the linguistic region grew in comparison with other primates. The Broca's located in the left frontal cortex is responsible for the ability to produce speech (Britannica, N/A). The Wernicke area that is found in the left temporal lobe is responsible for speech recognition. (Holden, 2004)

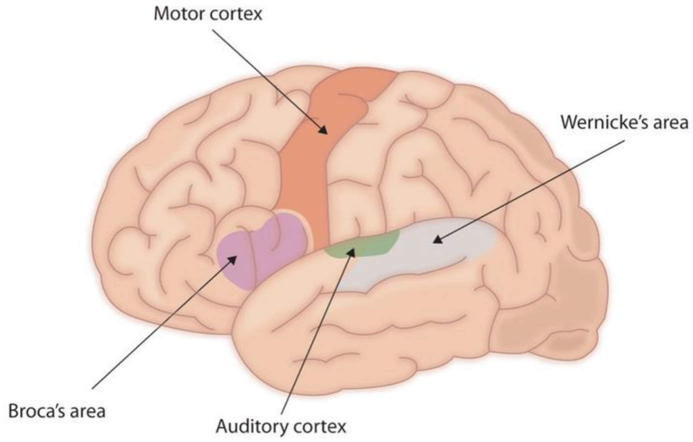

Figure 14 - Linguistic regions of the brain (Suri, 2014)

There is a gene that has been dubbed the speech gene. This gene is FOXP2. Scientists are unable to confirm when the gene finished its final mutation. However, the gene's final mutation is estimated to have occurred less than a hundred thousand years ago. This mutation granted the human species a new range of linguistic abilities. (Holden, 2004)

The other theory is that speech developed from human motor skills. Researchers believe that the discovery of stone tools approximately 2.4 million years ago could elude to the possibility that the forgers of these tools had linguistic capabilities. Undoubtedly, if this is the case, speech originates from the motor cortex of the brain. However, people argue that tool making capabilities does not equal speech capabilities. The strongest argument for speech being a motor function is that the motion of speaking requires moving multiple muscles in a precise way otherwise the correct sounds will not be made. The muscles that are needed for speech are the larynx, mouth, tongue and face, not to mention the control of the amount of air needed. All of these muscle movements need to happen in tandem so that the right sounds are made for communication.

Another argument for speech being a motor skill is that for people who lack the capability to speak instinctively revert to gesturing. This gesturing is more commonly known as sign language. It is believed that sign language is a method of communicating somewhere between speech and gesturing. Animals use gesturing as a method of communication. It is believed that humans no longer use gesturing as a method of communication due to flaws of only being able to communicate while there is enough lighting and the ability to vocalise allowed us to keep our hands free while working. However, supporters of the other theory argue that there is not enough evidence for humans transitioning from gestures to vocalising sound besides the ability to communicate in the dark. (Holden, 2004)

Speech is quite a unique aspect of the human body and it relates to the positioning of the larynx. The human larynx is located lower than the larynx of other primates. Primates possess the ability to lower their larynx to produce sounds in comparison with the human larynx. The male larynx in humans is lower than the female one making the male voice sound deeper (Holden, 2004). This makes the voice sound "bigger". There is a theory called the Size Exaggeration Hypothesis by W. Tecumseh Fitch from 1997. The theory behind this is that the lower larynx gives us the ability to "sound bigger". This would be beneficial for scaring off predators (Hannah, 2010). This evolution could only be beneficial for speech as the lower located larynx makes it easier to choke. Due to the low location of the larynx it makes it possible for humans to produce a broader range of sound. (Holden, 2004)

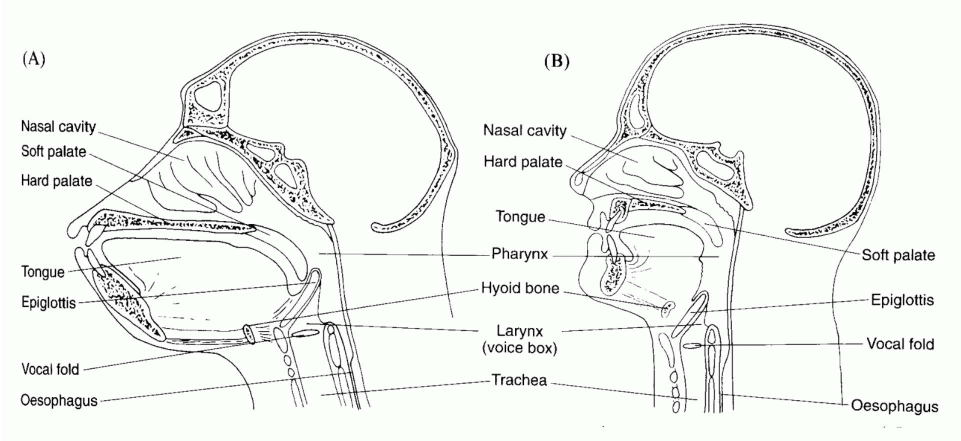

Figure 15 - Primate vs. Human larynx positioning (Kephart, 2010)

Language was fully developed approximately fifty thousand years ago. This would mean that the first language would have been in existence when a sudden uplift in cultural customs occurred. These customs included things like people making art and burying their dead. It is these customs that make scientists believe that there was a fluent form of communication between people. It is believed that this gene has had an impact on language and the ability to articulate. Scientists theorise that this gene appeared due to natural selection. The natural selection theory is based on the train of thought that only the strong survive. (Holden, 2004)

The end result is that nobody is exactly sure how speech became the main method of communicating between humans therefore researchers have turned to human's closest relatives in the animal kingdom. Scientists are now monitoring primates to discover more about the evolution of speech.

2.6 Physics of the Human Voice

Human speech comprises of three components:

1. The Lungs – The Power Source
2. Larynx – The Vibrator
3. Throat, Tongue, Nose and Mouth - The Articulators

1. <u>The Lungs – The Power Source:</u>

The lungs provide the function of supplying adequate air pressure and airflow to vibrate the vocal cord. As human's inhale, they draw air into their lungs. This air then travels through the windpipe until it reaches the larynx. The exhalation of this air allows the vocal cord to vibrate subsequently producing sound.

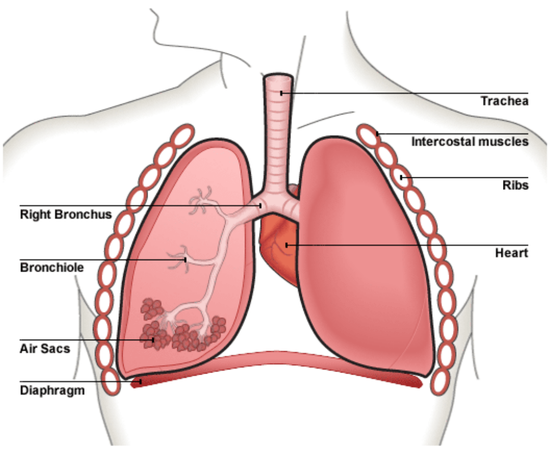

Figure 16 - Human Lungs (responsible for exhalation of air in speech) (Byjus.com, 2018)

2. Larynx – The Vibrator

The second phase in this process involves the larynx. Also referred to as the 'voice box'. It is located above the windpipe and consists of the vocal cord. The vocal cord is composed of two tight strings that provide functions such as opening and closing to allow for the passage air between the mouth and lungs necessary for human breathing. When a person exhales, the vocal cord vibrates which produces sound. They vibrate approximately 100-1000 times per second and are also referred to as the 'vocal folds'.

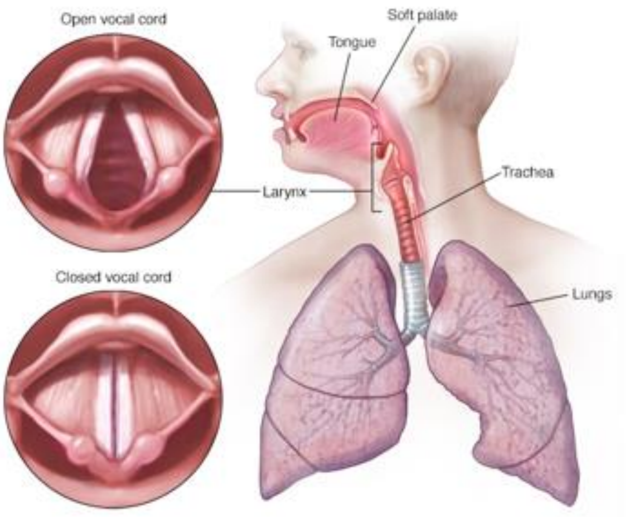

Figure 17 - Vocal Cord (responsible for creating sound in human speech) (Byjus.com, 2018)

3. Throat, Tongue, Nose and Mouth - The Articulators

The sound which the vocal cord produces is considered to be a buzzing-like tone, much like a bee. The vocal cord works in tandem with various articulators such as the tongue, throat, nose and mouth to modify the sound which is being produced. The interaction between the larynx and muscle/tissues responsible for the modification of sound give humans the ability to produce a wide range of content at diverse frequencies.

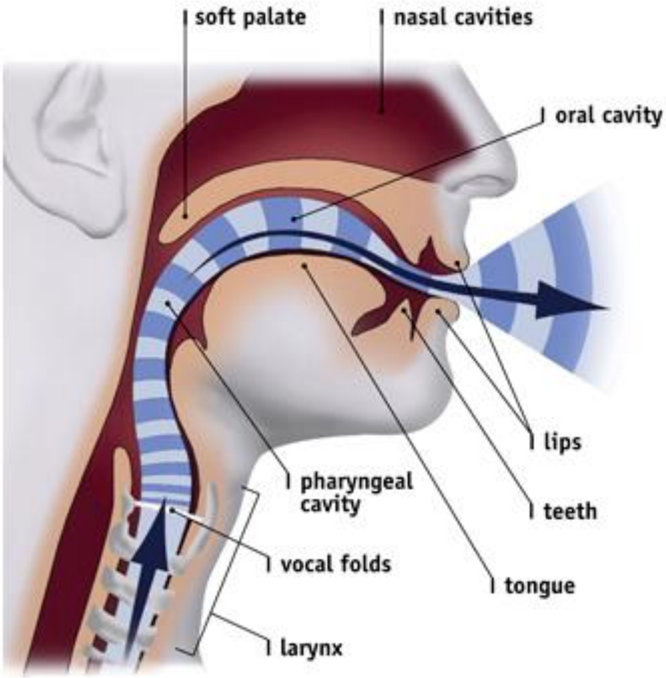

Figure 18 - Articulators (responsible for changing sound produced by larynx) (Byjus.com, 2018)

2.7 Analysis of the voice

The figures below show a spectrogram of a TIMIT audio sample ("*She had your dark suit in greasy wash water all year*").

Figure 19 shows how the voiced part of the speech signal is clearly identified in the corresponding spectrogram. It can also be observed that the voiced parts of the signal contain much more energy than the unvoiced parts. The fundamental frequency can be distinguished as a white horizontal line at approximately 1.5 and 2.5 kHz and they are visible in the spectrogram from approximately 0.60 seconds to 2.10 seconds. The harmonics are only present at the voiced sections of the audio sample. These are the white parts of the spectrogram which represent high amplitude at that specific frequency. The colours go from blue to pink to red to white which represents amplitude from low to high respectively.

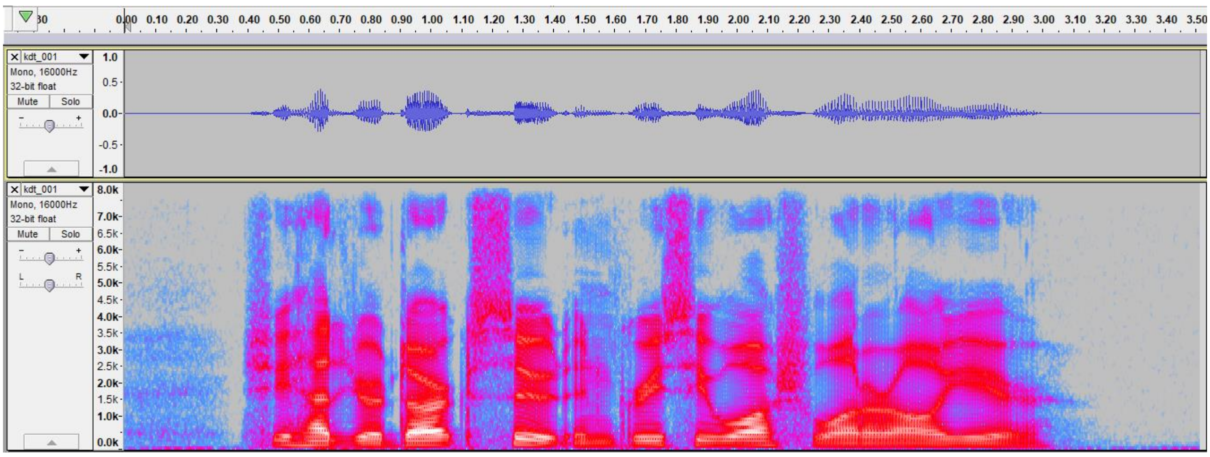

Figure 19 - Spectrogram of voice audio ("She had your dark suit in dirty wash water all year")

In *Figure 20*, the harmonics on the voiced parts can be more easily observed.

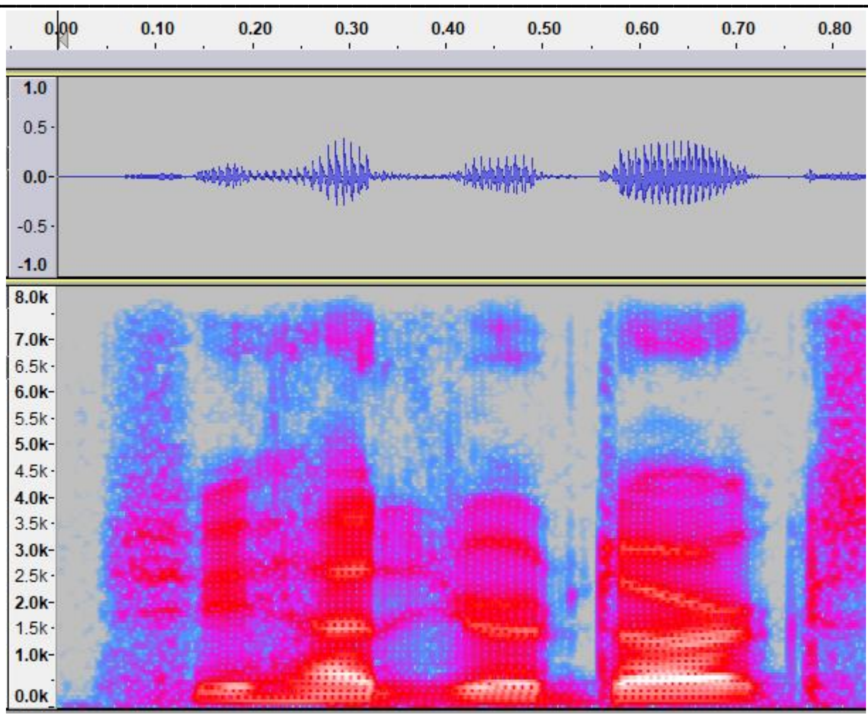

Figure 20 - Spectrogram of voice audio (Close up)

2.8 Speaker Normalisation

Speaker normalisation is a popular application of voice conversion. It is a method that converts a voice sample input to a normalised voice sample. The purpose behind this particular application is that it reduces the effective signal space. Subsequently, this will allow speech recognition and compression methods to perform better. An assumption is made that the phonological identifiers occurs when utterances of the same word occur. An example of this concept exhibits the following: the phrase 'cat' spoken by a male and female identify as the same word, although the spectrogram image of both audio signals will differ. The main difference between these two images relate to their vowel formant frequencies. (Johnson, 2018)

2.9 Male vs Female Voice Differences

The human voice is classified by two genders, male and female. Some of the main differences between both sound profiles are as follows:

Categories	Male	Female
Average Frequency/Pitch	110Hz	211Hz
Articulation *('the formation of clear and distinct sounds in speech')*	Rougher	Gentler
Intonation *('the rise and fall of the voice in speaking')*	Tend to use full pitch range during speech	Don't tend to use full pitch range during speech
Conversational Style	More likely to raise their volume for emphasis	More likely to raise their pitch for the same purpose
Larynx Size	Larger than female	Smaller than male

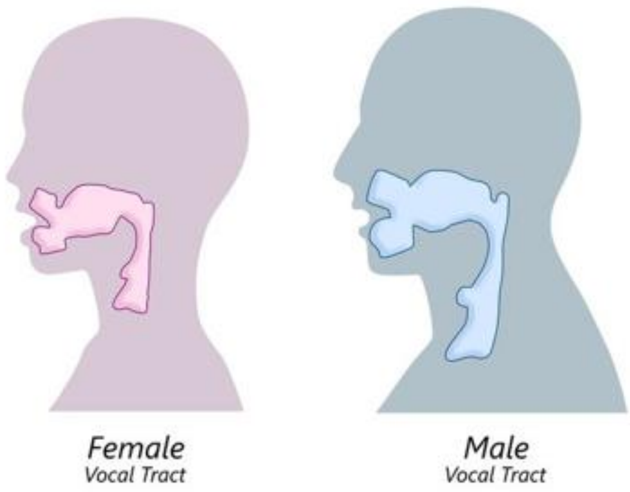

Figure 21 - Vocal tract differences between male and female (Pinterest.ie, 2018)

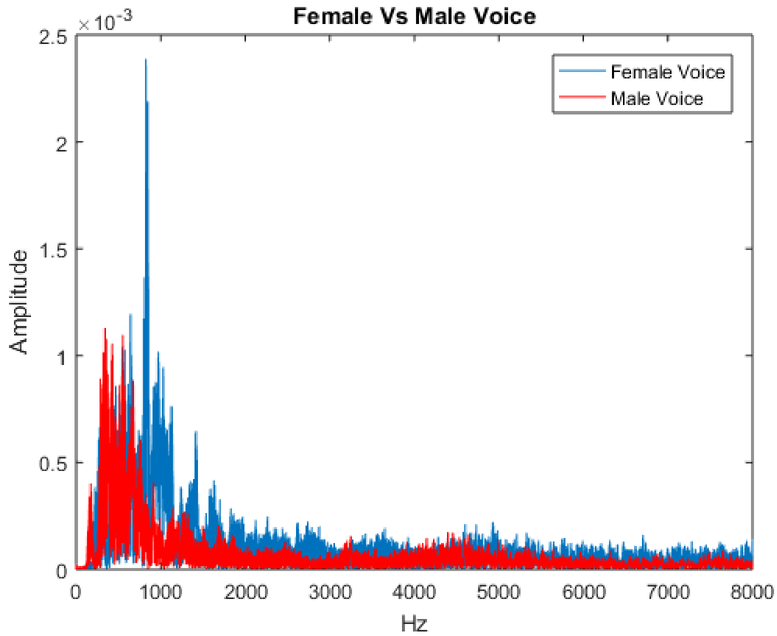

Figure 22 - FFT Female vs Male Voice frequency spectrum

The frequency of the resting male pitch is clearly lower. A spike in the female's frequency amplitude is due to the fact that the recording of the female consists of the person speaking very loudly at that frequency. The spectrograms of each signal can be seen below.

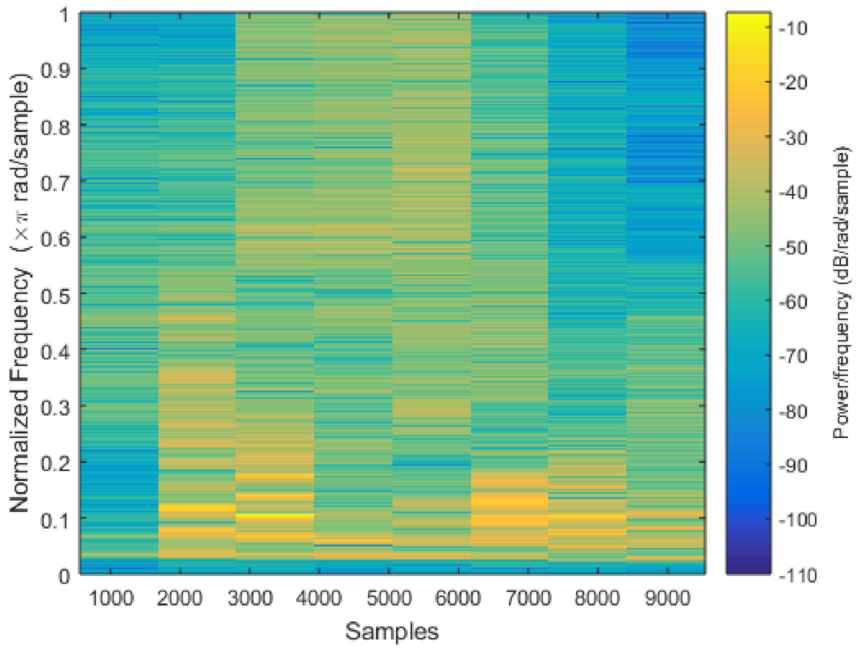

Figure 23 - Spectrogram of male voice

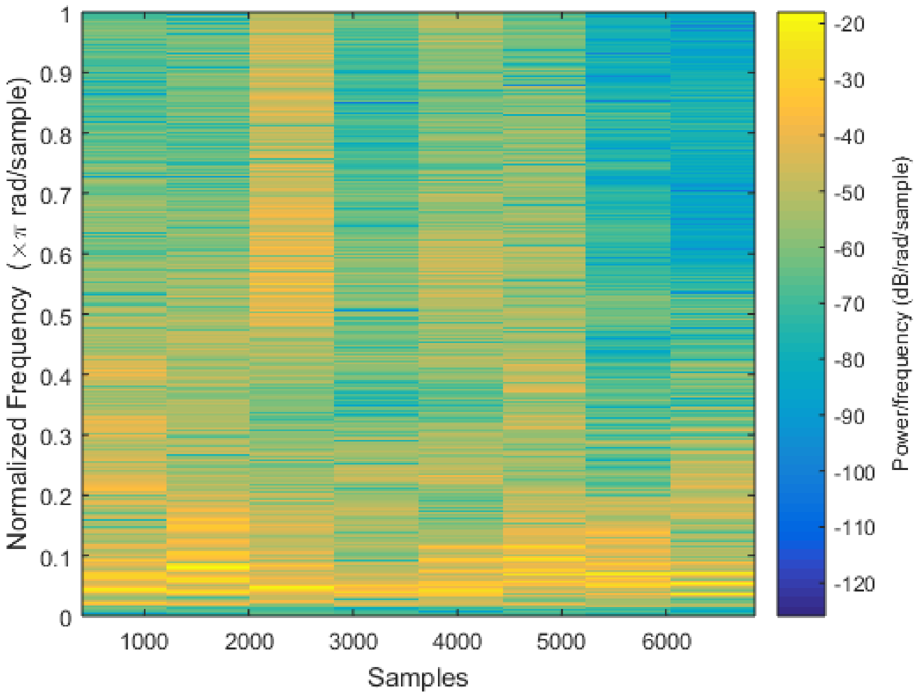

Figure 24 - Spectrogram of the female voice

Two TIMIT files were compared. One .wav file had a male voice and the other a female voice. They both said the same sentence and are shown in, *Figure 23* and *Figure 24.*

The fundamental frequency of the persons voice signal can be seen on the bottom line-shaped area of each graph. The female voice can be seen to have a slightly higher fundamental frequency. The sentences that TIMIT have recorded reads "*She had your dark suit in greasy wash water all year.*"

2.10 Phonetics and Phonemes

Humans have many different languages, accents and dialects between countries, counties and provinces. Specifically, one of the latest figures suggests that over 7’106 living languages exist in the present day (Translations, 2018). To list a few of the 56 main English accents that exist only in Ireland:

- Cork
- Dublin 4 (D4)
- Inner city Dublin
- Donegal
- Kerry
- Louth
- Limerick city
- Midlands
- Sligo town
- Waterford city
- West
- Wexford town

(Ríkharðsson, 2014)

Because of this large variety in human language and dialect, humans need to distinguish between different vowels and consonants. Phonetics are “*the system of speech sounds of a language or group of languages*” (Webster, 2018).

SIMPLE VOWELS /ː/ =long sound

/iː/ sleep	/æ/ cat	/ə/ China	/ɒ/ hot, coffee
/ə/ slip	/ʌ/ cut	/ɜː/ learn	/ʊ/ look
/e/ send	/aː/ father	/ɔː/ law	/uː/ food

DIPHTHONGS

/ʌɪ/ bay	/ɔɪ/ boy	/æʊ/ now	/eə, ɪə, eː/ where
/ɑɪ/ buy	/ʌʊ/ home	/iːə/ near	/uːə, ɔː/ tour

CONSONANTS

/p/ pay	/g/ go	/s/ soon	/dʒ/ judge	/w/ wine
/b/ bay	/f/ fine	/z/ zoo	/h/ high	/m/ my
/t/ too	/v/ vine	/ʃ/ ship	/j/ year	/n/ next
/d/ do	/θ/ three	/ʒ/ measure	/l/ load	/ŋ/ sing
/k/ coat	/ð/ this	/tʃ/ chair	/r/ road	

Figure 25 - Some important phonetics (Café, 2018)

The TIMIT database has recorded many phonetically rich sentences said by people of differing genders. This database is used in this project to experiment with speech because of TIMITs reduced noise recordings. The lack of noise in these recordings will help the group distinguish between implemented noise and non-implemented noise, because there will always be a noise-free signal provided by TIMIT.

2.11 Formants

While harmonics come from vibrations Figure 26 of the vocal chords themselves, **formants** come from the vibration of air inside the vocal tract. If the vocal tract did not exist, then the vocal chords themselves would create a quiet, harsh tone. This can be said because the vocal tract acts as a filter, which makes the voice softer and louder with the aid of formants. *"Formants filter the original sound source. After harmonics go through the vocal tract, some become louder and some become softer."* (Works, Unknown)

Figure 26 - Harmonics in the voice (Works, Unknown)

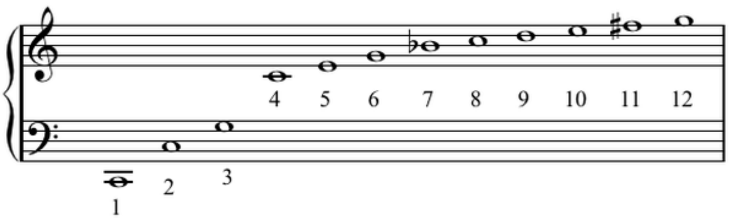

Figure 27 - Harmonics and formants combined on a musical staff (Works, Unknown)

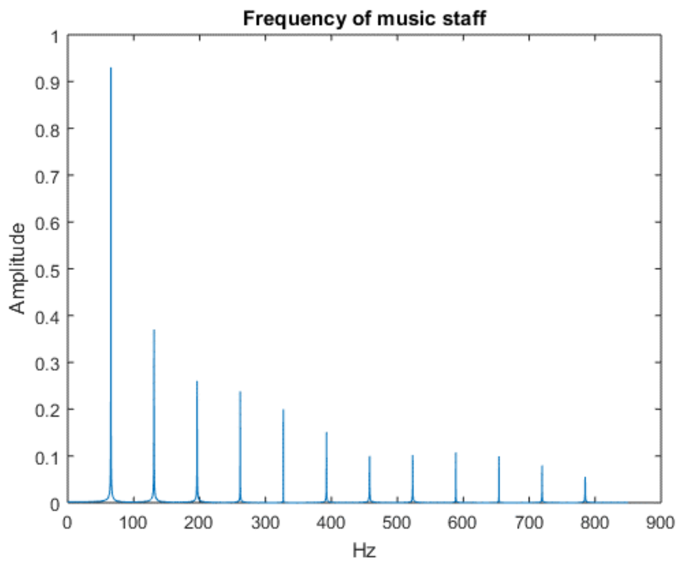

Figure 28 - Frequencies of music notes summed and Discrete Fourier Transformed

In *Figure 27* the differences in notes are not entirely correct, as the harmonics are only integer multiples of the fundamental;

$$F_0 = 65.406\ Hz$$

This frequency is approximately equivalent to a 'C2' in musical notation, which is the first note seen in *Figure 27* when reading left to right.

3 Technical Implementation

This section exhibits the development of both basic and complex signal processing techniques. The basic element involves conducting an experiment to test that Fourier Transforming a 2Hz and a 5Hz sine wave gives the intended output. The intended output of both signals was a spike at 2Hz and 5Hz respectively as depicted in Figure 34. Contrary, the complex element involves filtering out 50Hz of noise from the summation of 50Hz and 1kHz sine waves.

In relation to the overall goal of modifying the tone of a human voice, pitch shifting was performed on the audio sample ("She had your dark suit in greasy wash water all year"). Two different methods to pitch shifting were conducted, one was audio resampling while the other, circular shifting the frequency. Both approaches of pitch shifting successfully modified the frequency correctly although one application faced repercussions subsequently producing unnatural audio.

3.1 Fourier Transform of Pure Sine Waves

Below are figures generated in MATLAB proving that the Fourier Transform indeed works as expected. Two pure sine waves are generated at 2Hz and 5Hz respectively. These signals are then plotted in the time and frequency domains. The two signals are then added together. This new summation signal is then Fourier transformed. It can be seen in Figure 34 that the two pure sine waves are easily recognised.

The following figure are graphs of 2Hz and 5Hz sine waves.

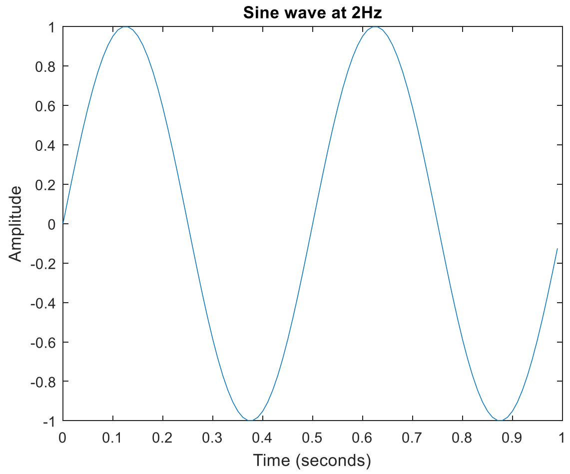

Figure 29 - Sine wave at 2 Hz

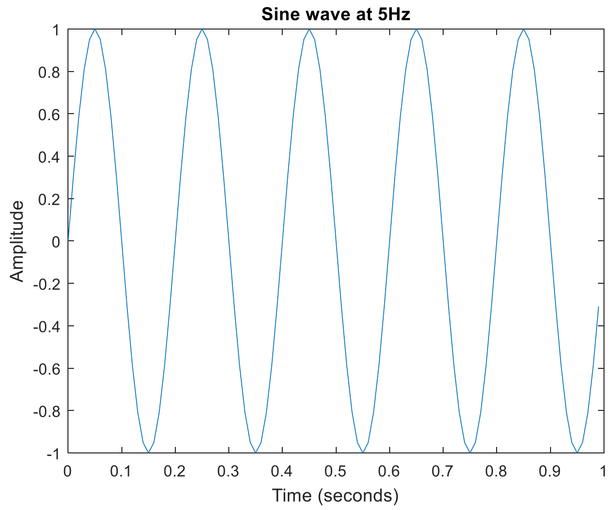

Figure 30 - Sine wave at 5Hz

The following figure is a Fourier Transform of 2Hz and 5Hz signals respectively.

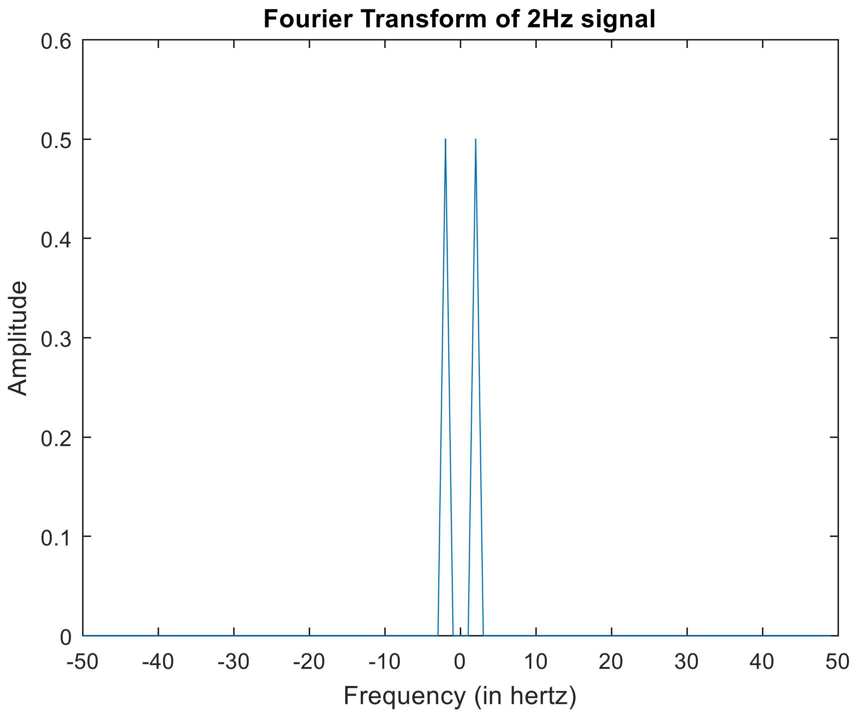

Figure 31 - Fourier Transform of sine wave at 2Hz

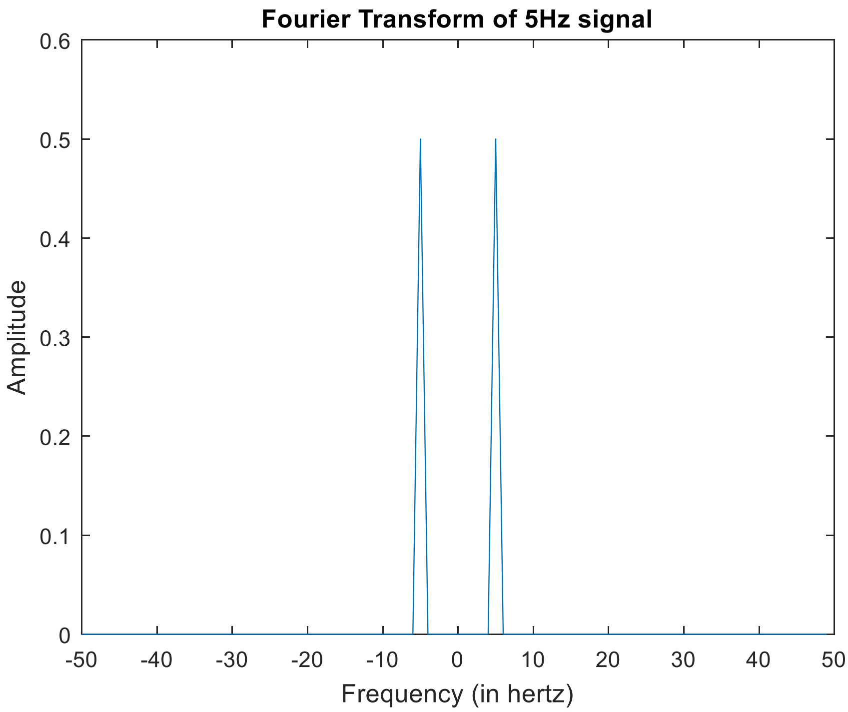

Figure 32 - Fourier Transform of sine wave at 5Hz

Both signals are then added together.

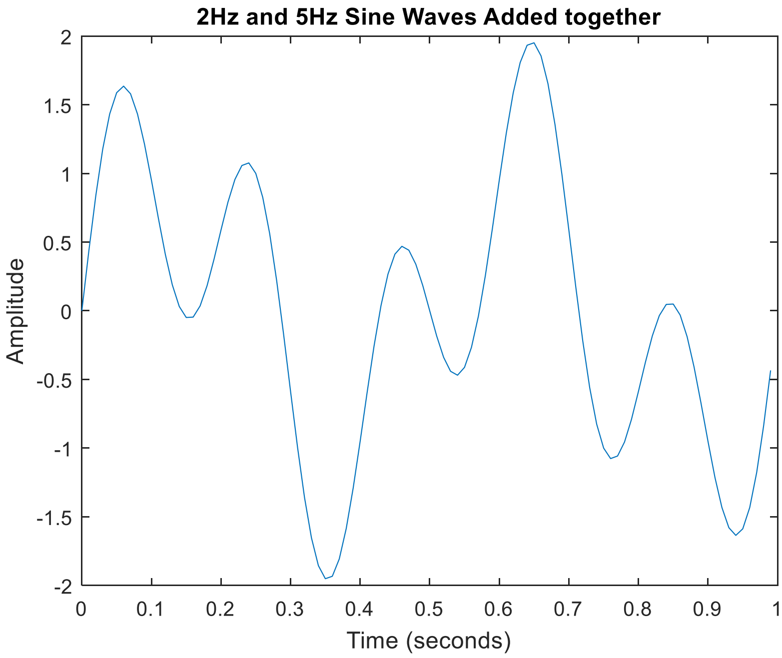

Figure 33 - 2Hz and 5Hz sine waves added together

The Fourier transform clearly shows the presence of the 2Hz and 5Hz signals.

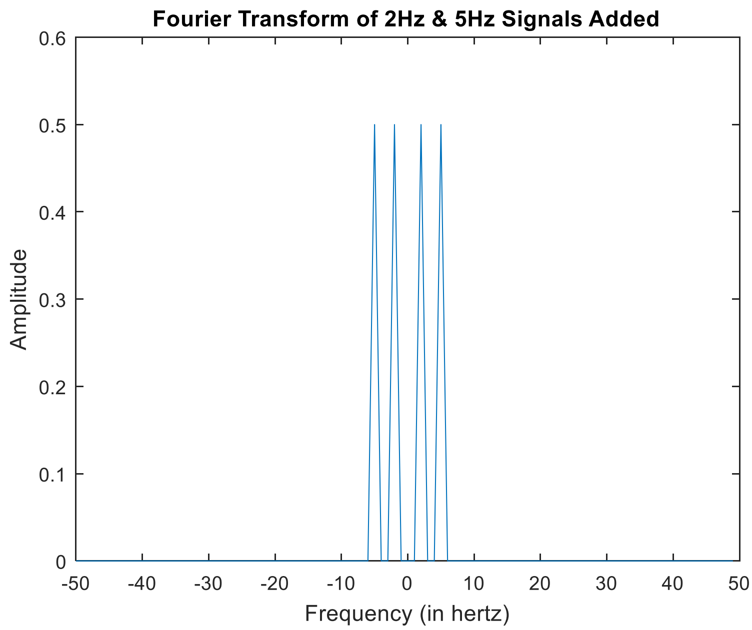

Figure 34 - Fourier Transform of summation of 2Hz and 5Hz sine waves

3.2 Filtering

Implementation of different types of filters.

3.2.1 High-pass filter

Below is a demonstration of filtering. A 50Hz signal is added to a 1 kHz signal. These values were chosen to simulate the 50Hz mains electricity signal interfering with a 1 kHz transmission signal. Both signals are plotted then added together.

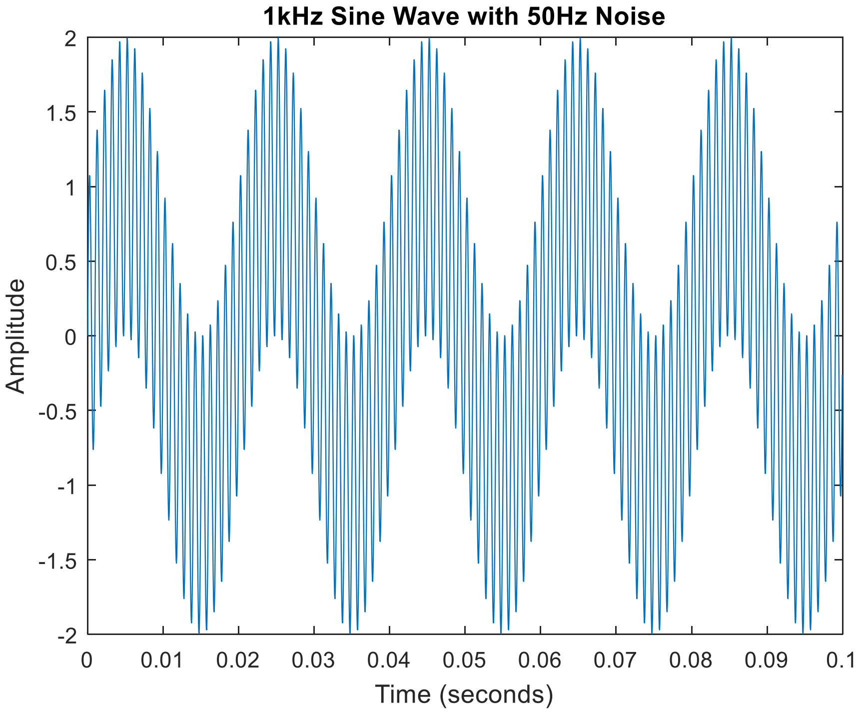

Figure 35 - Summation of 50Hz and 1kHz Sine Waves

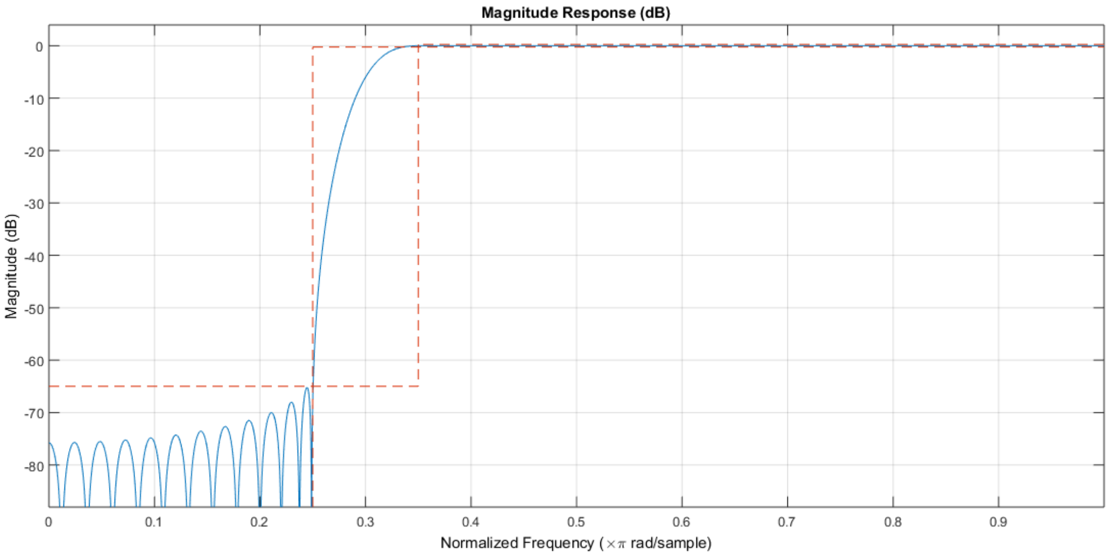

Figure 36 – High-pass filter response. Stopband 1 KHz.

The summation of the signals is then Fourier transformed. The graphs below clearly show the 50Hz and 1 kHz signals.

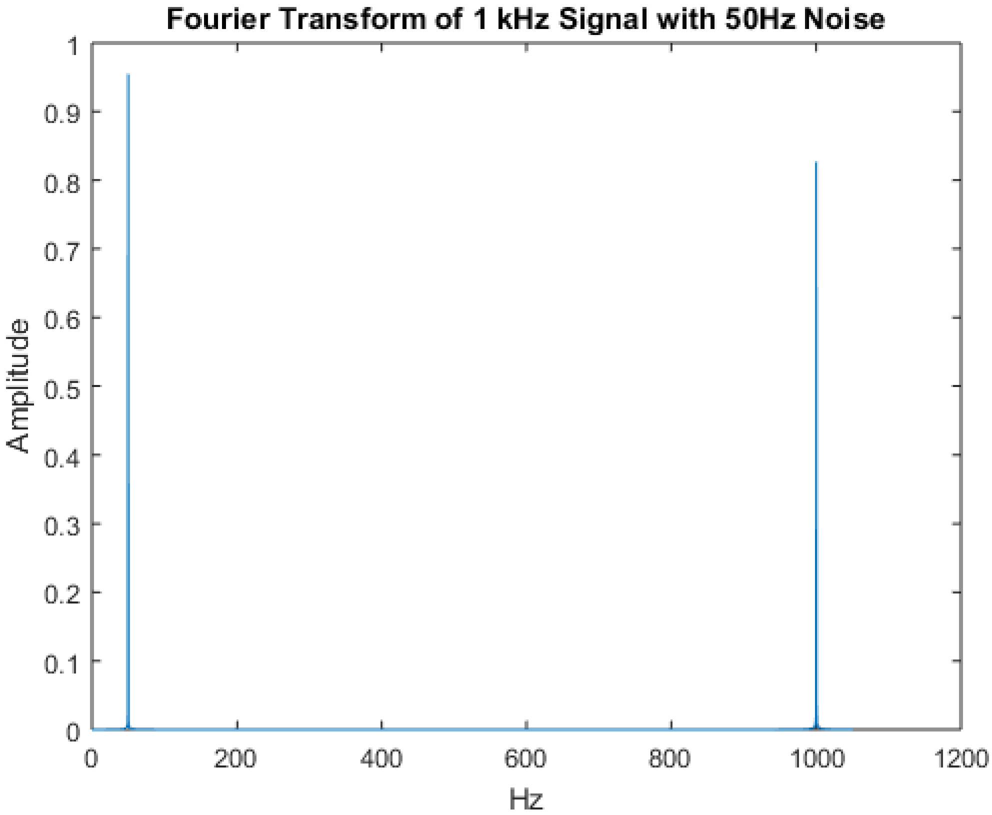

Figure 37 - FFT of input signal of 1kHz with 50Hz noise

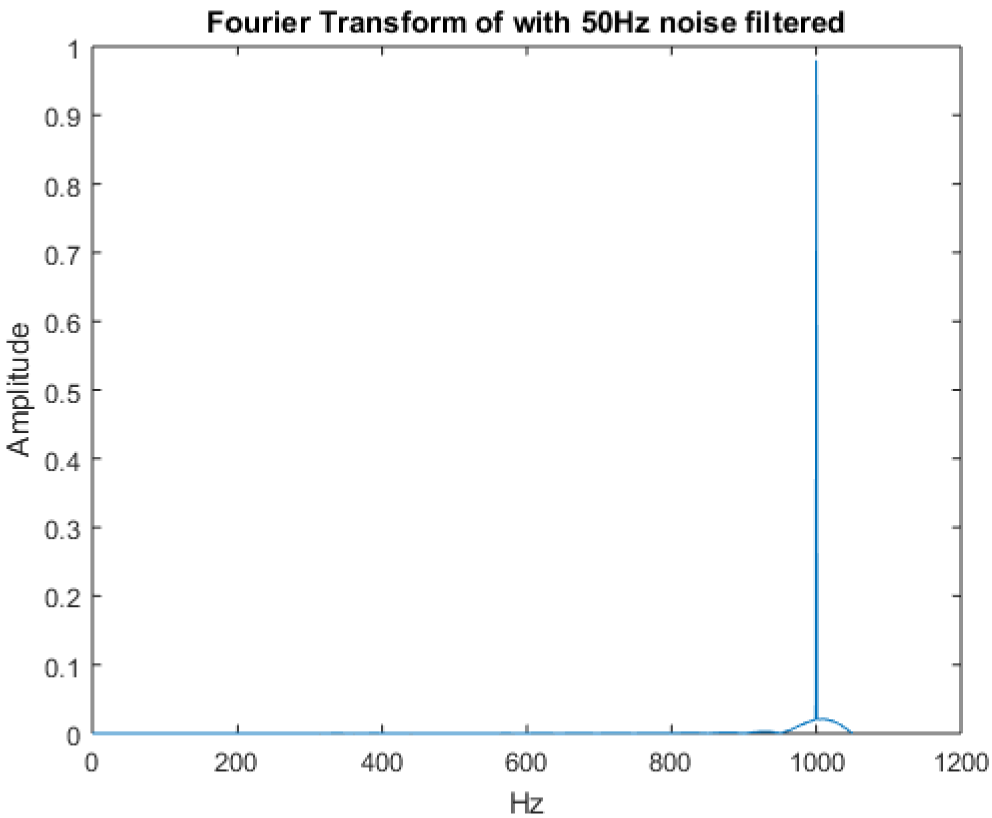

Figure 38 - Fourier Transform with 50Hz filtered out

The low frequencies of 50Hz will now be filtered out using a high pass filter. The following code shows the implementation of that filter on MATLAB;

```
FIRHighPass = dsp.HighpassFilter('PassbandFrequency',1000,...
    'StopbandFrequency',60);
fvtool(FIRHighPass)

for i = 1 : 1000
    x = Signal50HzPlus1kHz;
    y = FIRHighPass(x);
    FilteredSignal = conv(x,y,'same');
end

```

Figure 39 - High Pass Filter Code

3.2.2 Band-pass filter

A band pass filter is very useful in filtering voices as it allows only a specific band of frequencies through. The human voice typically has a normal speaking range of 80Hz to 8000Hz. A sum of synthetic sine waves was used to test the filter. The signal was comprised of 1, 2, 3 and 4 kHz.

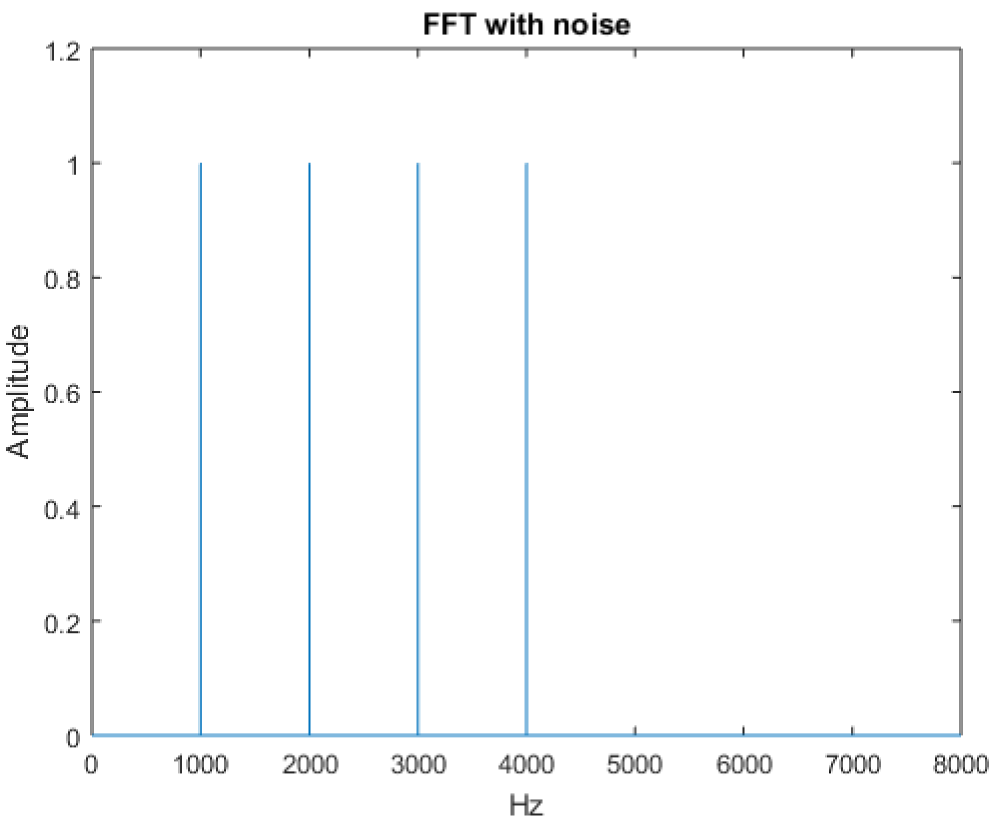

Figure 40 FFT of 4 sine waves at 1, 2, 3 and 4 kHz

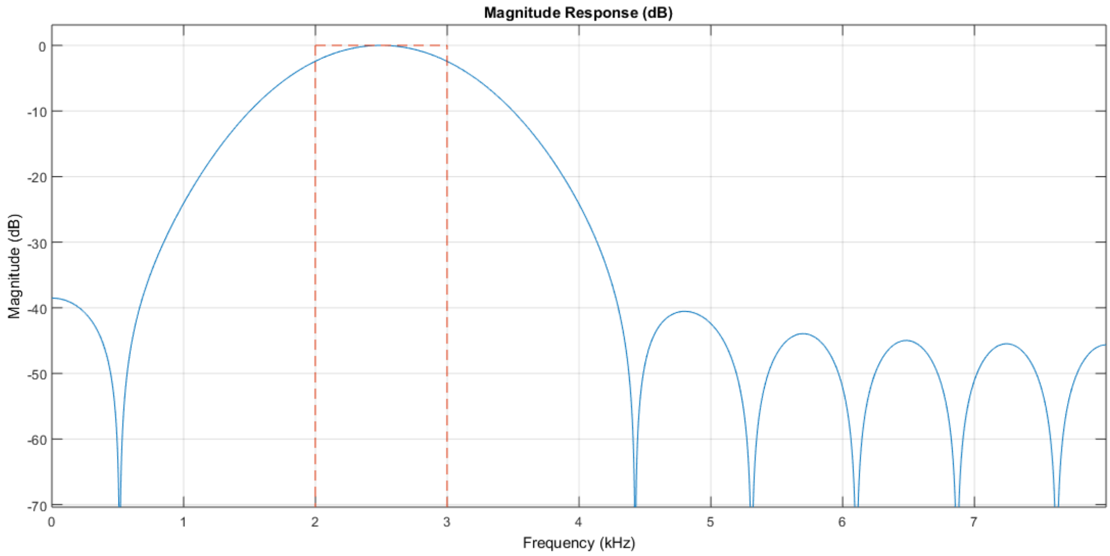

Figure 41 band-pass filter response. 20th Order.

The above filter response demonstrates that the band-pass filter, using hamming window, will attenuate frequencies outside 2000 Hz and 3000 Hz. The result of applying the filter to the synthetic signal can be seen below in Figure 42. The 1000Hz and 4000Hz sine waves are attenuated but still have a small amplitude. The order of the filter determines how accurate the response is, the objective being a perfect brick-wall on either side of the band. Raising the order of the filter would increase the attenuation outside the band.

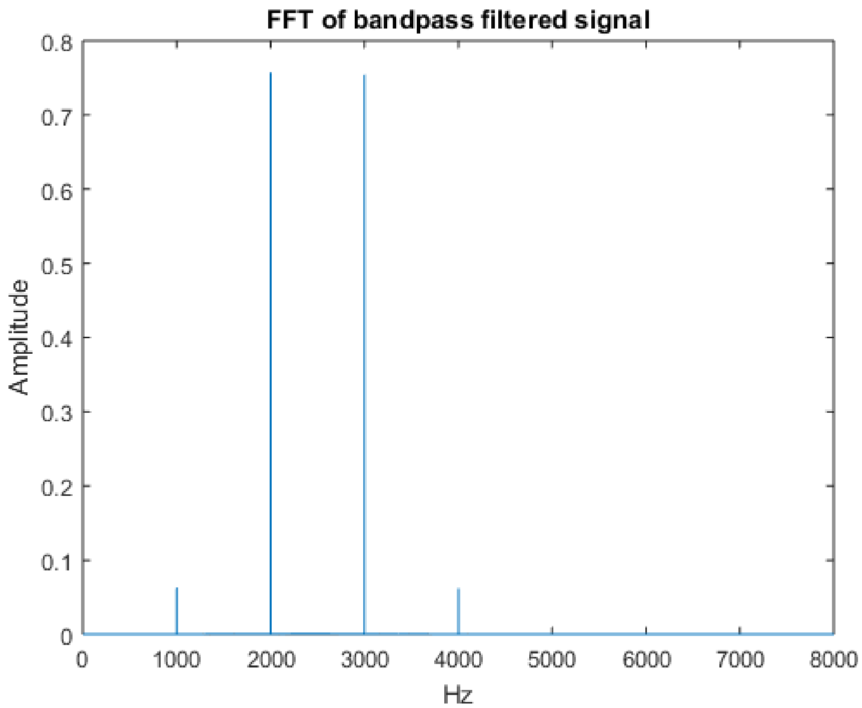

Figure 42 Band-pass filtered signal

3.3 Aliasing

Below is a demonstration of how aliasing occurs when the sample frequency is too low. The original 100Hz sine wave to be sampled can be seen in *Figure 43 - 100Hz pure sine wave*.

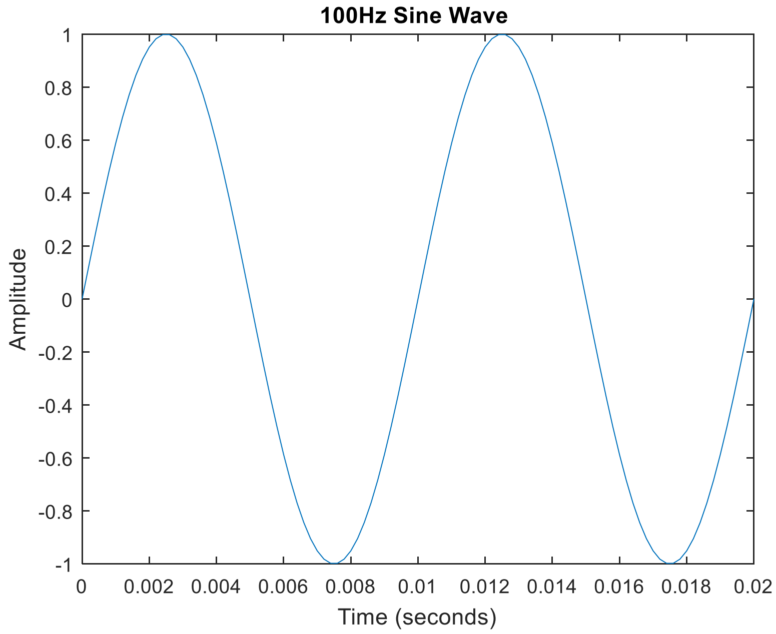

Figure 43 - 100Hz pure sine wave

In *Figure 44* the 100Hz pure sine wave is sampled at 5000 samples per second.

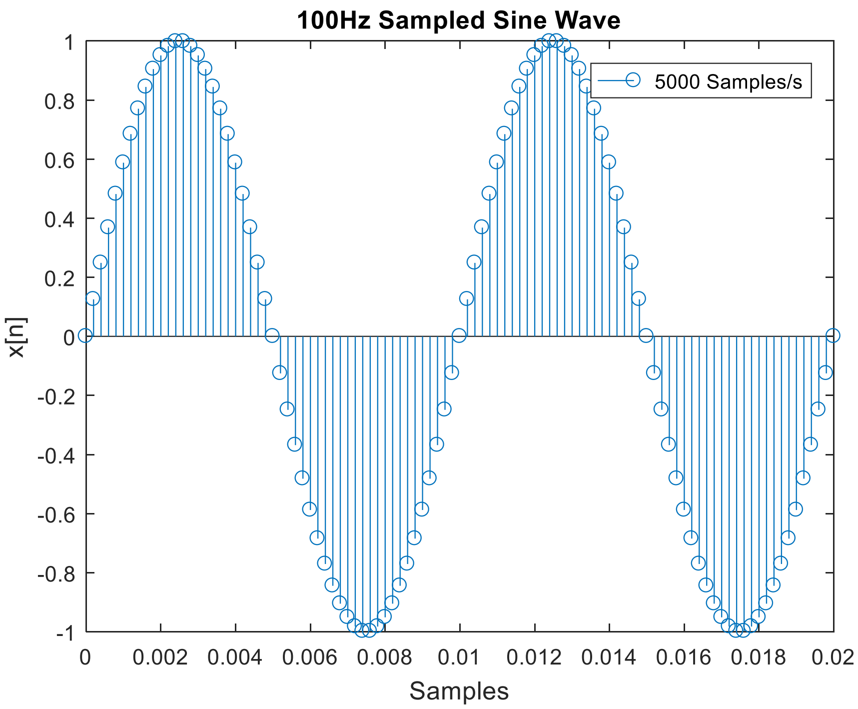

Figure 44 - 100Hz sampled sine wave

In *Figure 45* it can be seen that when the sample frequency gets too low, an accurate representation of the original signal is lost.

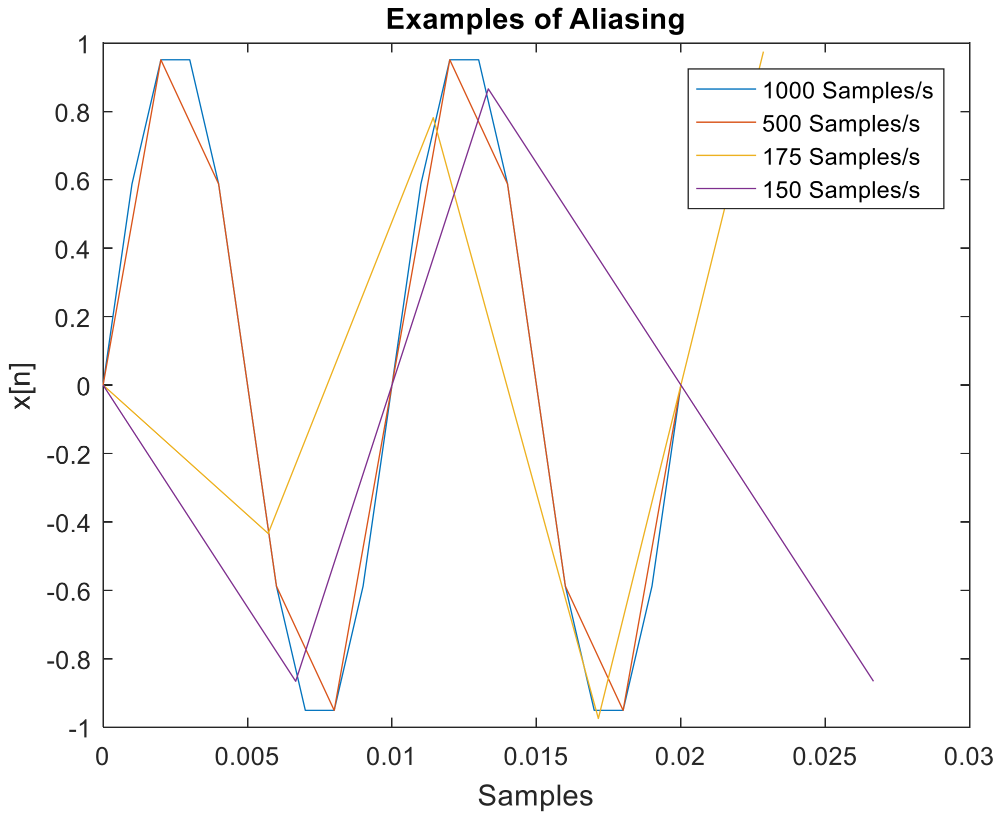

Figure 45 - Example of aliasing of pure sine wave

3.4 Pitch Shifting

Pitch shifting is a process by which the pitch or frequency of a signal is raised or lowered. Pitch shifting is implemented in this project by taking the Fourier transform of a signal. That signal is then circularly shifted it to increase or decrease the pitch by altering the elements of the array returned by the Fourier transform. The problem encountered showed that the Fourier transform returns a mirrored version of the frequency spectrum of a signal. To shift the signal effectively this mirrored portion was removed, the array was shifted, and then the mirrored portion was added back into the array. Finally, the signal was inverse Fourier transformed and the new audio signal was pitch shifted.

3.4.1 Audio Signal Resampling

Figure 46 shows a voice sampled at two different sample rates and the difference between them. It can be seen that as the sample rate is halved, the original signal is stretched to twice the duration of the original. This has the effect of reducing the pitch of the audio and making it sound lower in frequency.

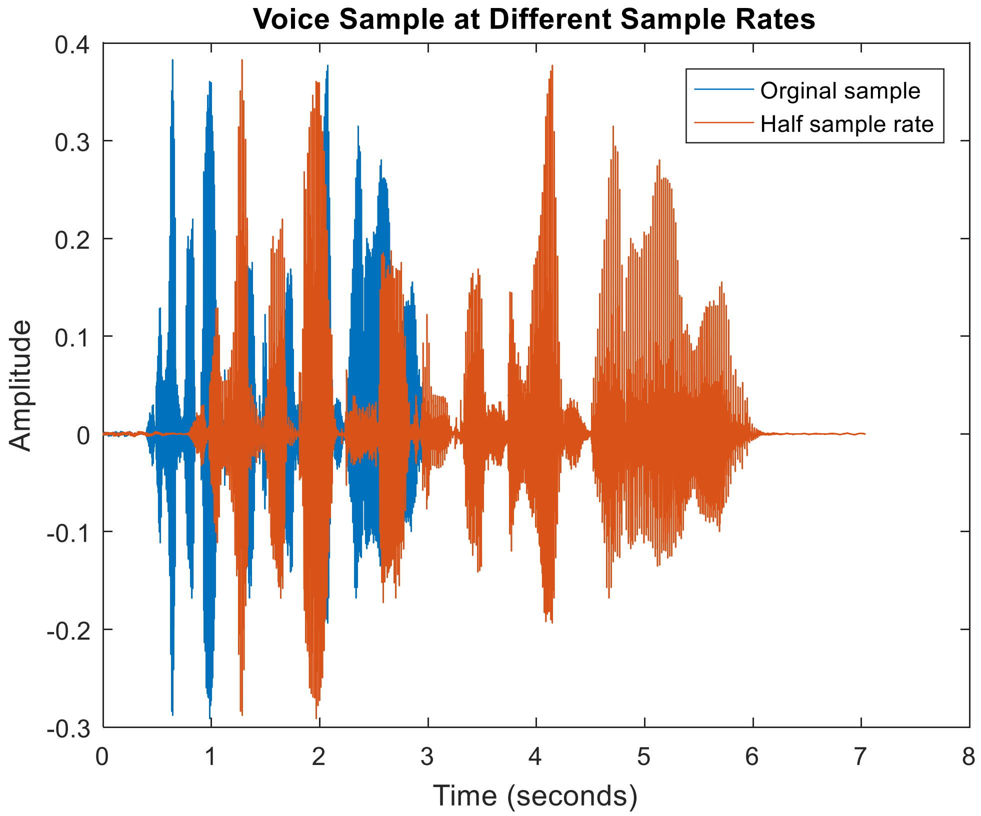

Figure 46 - Voice sample at different sample rates

Figure 47 shows that when the sample rate of a signal is halved, the frequencies in the sample are shifted. If the sample rate is reduced, then the sample will be stretched in reference to a constant playback speed. A halving of the sample frequency will result in a doubling of the length of the sample with respect to time. This means that the frequencies present in the sample must be lower which is proved in *Figure 47*. The same is true for increasing the sample rate, except it will sound higher in the frequency band and the length of the audio will decrease.

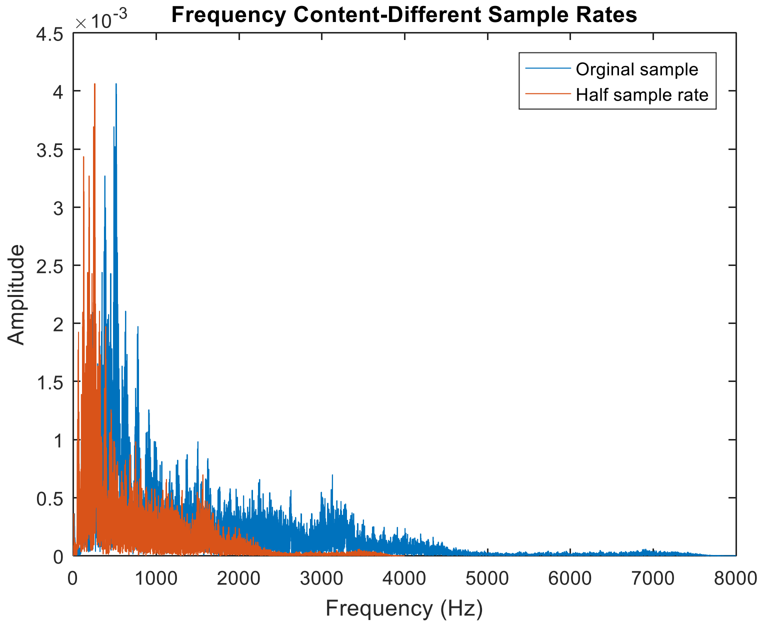

Figure 47 - Frequency content of voice sampled at different rates

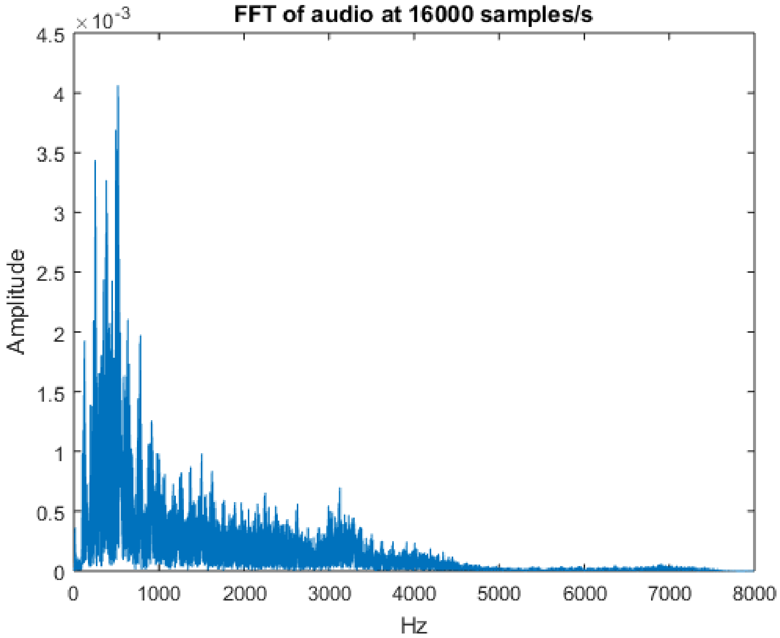

Figure 48 - Audio at original sample rate

The below *Figure 49* shows the discrete Fourier transform of the resampled signal, where the original sample rate of 16000 samples/s has been doubled to 32000 samples/s. It can be seen that each frequency has doubled.

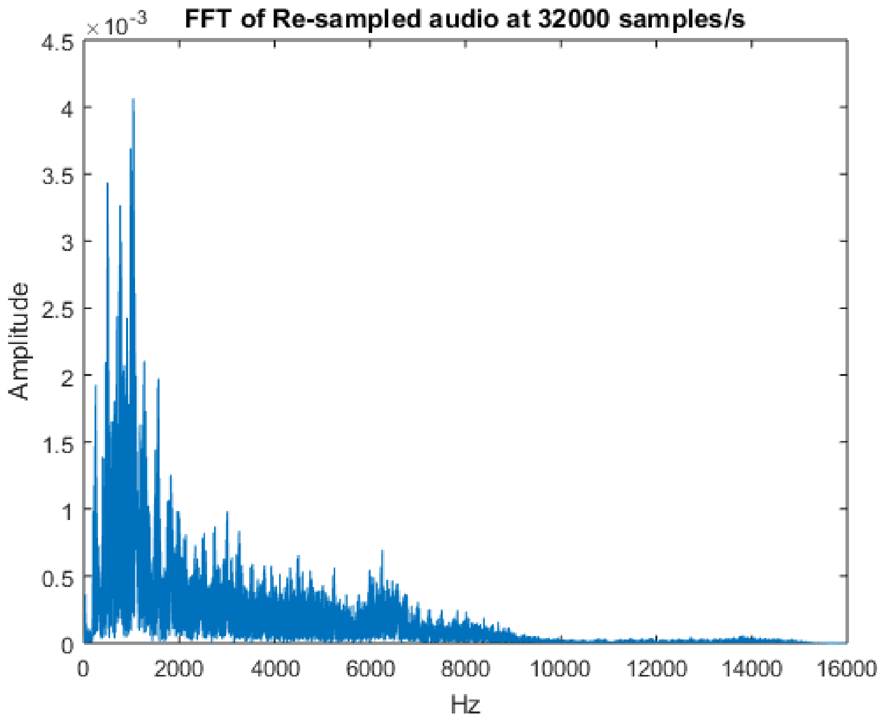

Figure 49 - Audio re-sampled at 32000 samples/s

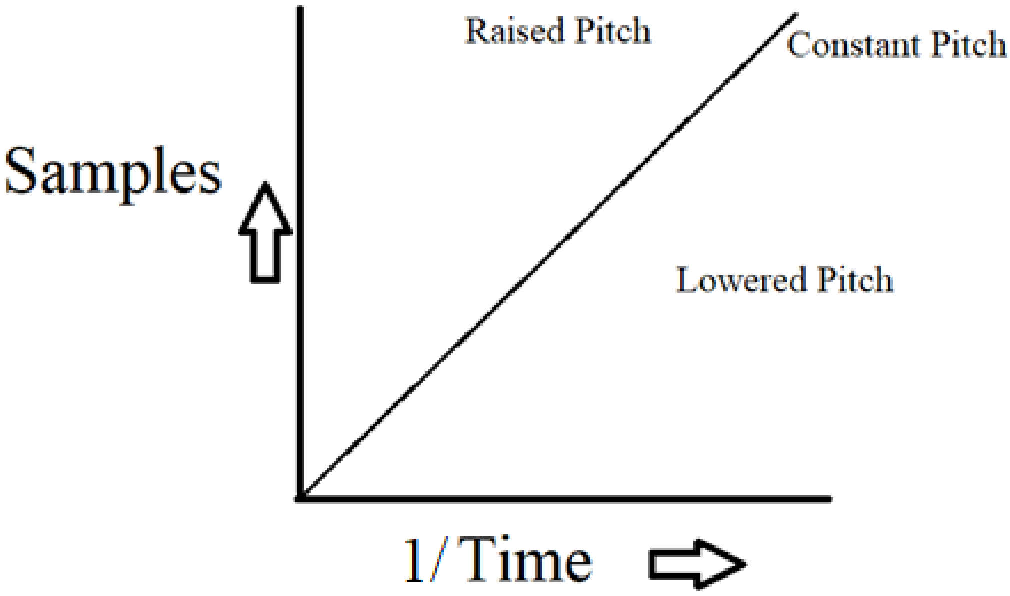

Figure 50 - Time vs. Sample for pitch shifting

3.4.2 Circular shifting frequency

The symmetrical properties of the discrete Fourier transform allow the circular shifting of the frequencies in the DFT array of complex values. This method of shifting the frequencies has the advantage of keeping the signal the same length while shifting the frequencies up or down.

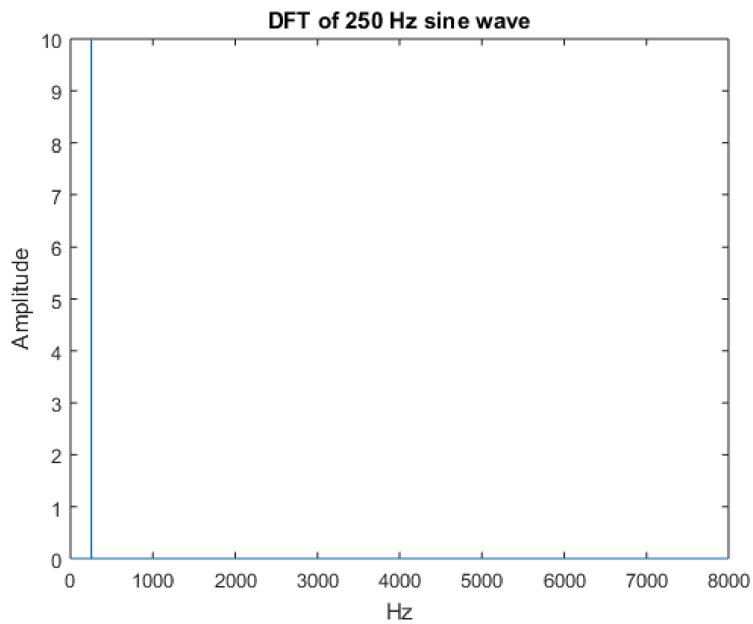

Figure 51 - Synthetic input used to test circular shift

A synthetic sine wave of amplitude 10 and frequency 250Hz was used as input to circular shift function.

```
xDFTwitoutDCgain= xdft(2:end); % Remove DC gain from begining of Array
frequencyShiftHz = 750;  %hz to shift. Positve or negative
frequencyShift = ceil((length(signaly)/Fs)*frequencyShiftHz) %calculate shift by size of DFT frequency bins
circShiftedSignalDFT = circshift(xDFTwitoutDCgain,frequencyShift);  %circular shift
flipped = conj(fliplr(circShiftedSignalDFT)); % Flip array and get conjugate of each component
newWhole = [xdft(1) circShiftedSignalDFT flipped ]; %Add back in DC gain and -/+ freq elements
```

Code snippet 1 - Circular shift

The above code snippet sets out the algorithm involved in shifting the DFT array by a given frequency (750Hz in this case) and adding back in the mirrored conjugates of the newly mirrored section resulting in *Figure 52.*

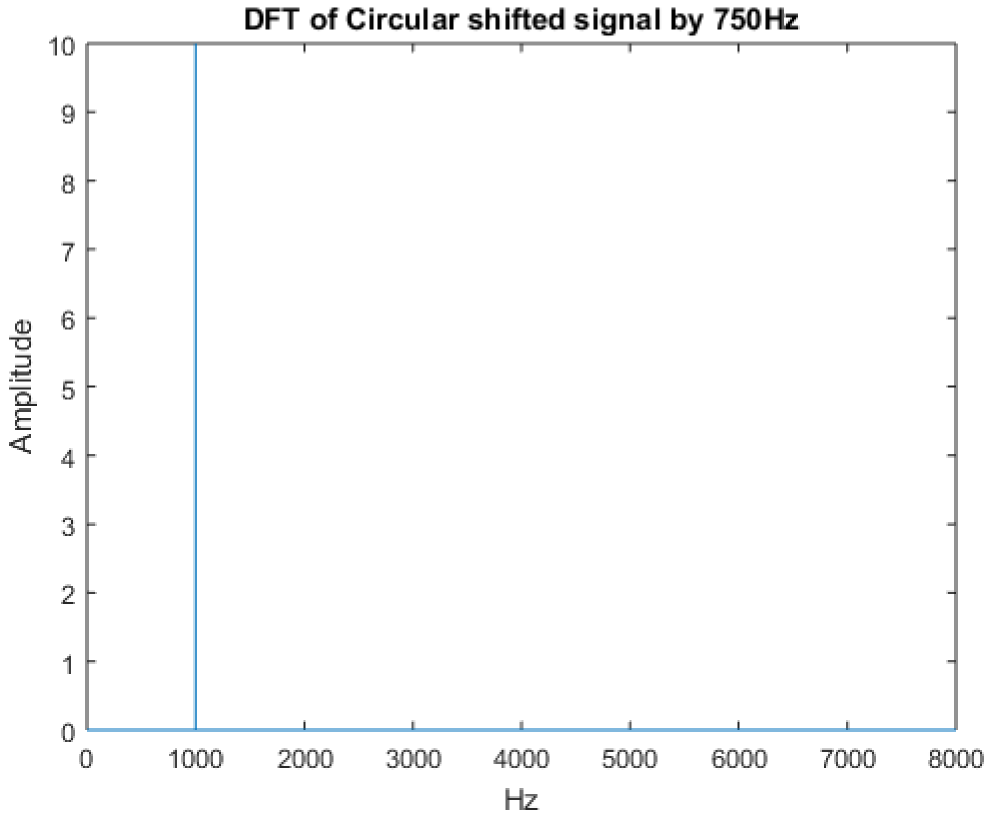

Figure 52 - Signal shifted 750Hz by circular shift

A real signal from the TIMIT database was then fed into the algorithm and can be seen in the time and frequency domains in the figures below respectively.

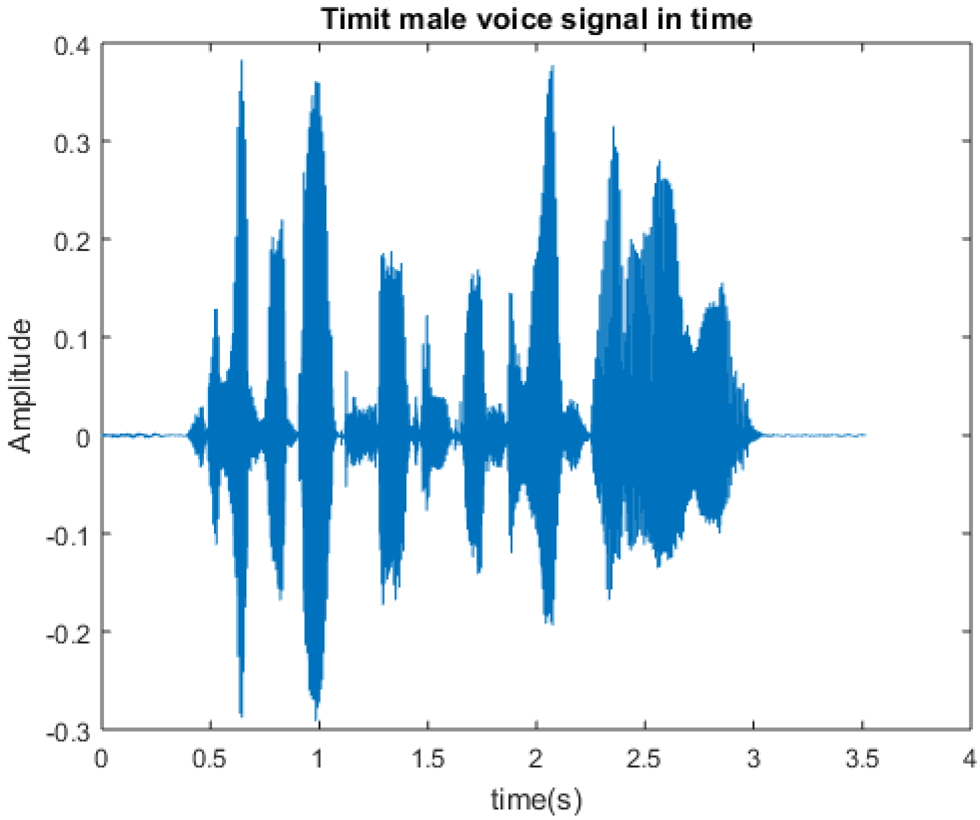

Figure 53- TIMIT input signal for pre-shifting

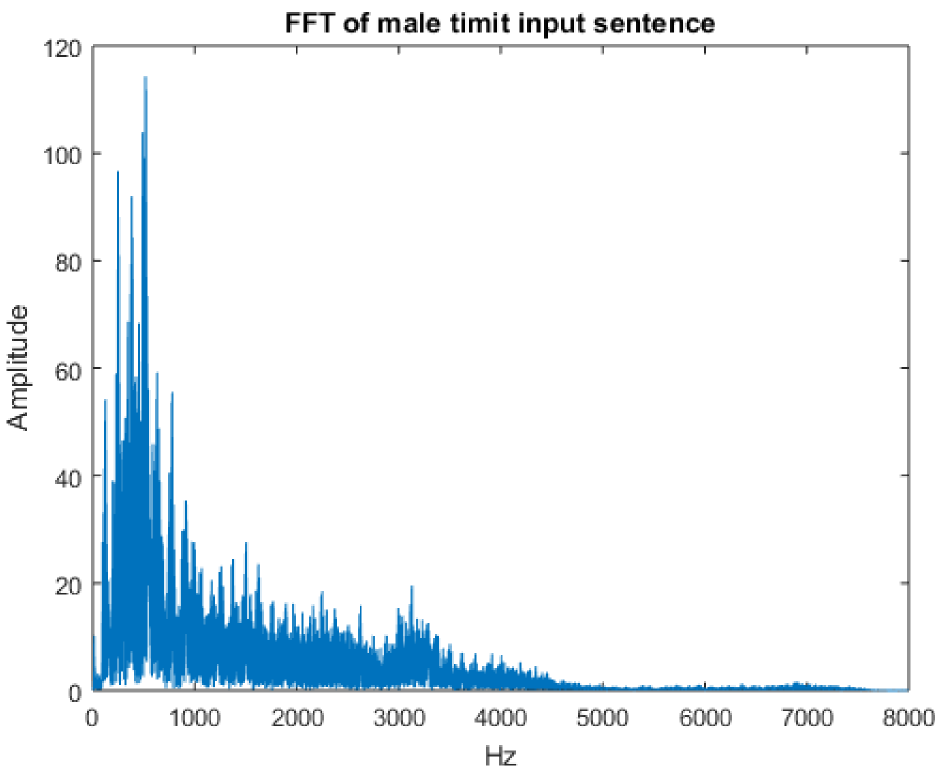

Figure 54 - FFT pre-shift of TIMIT sentence

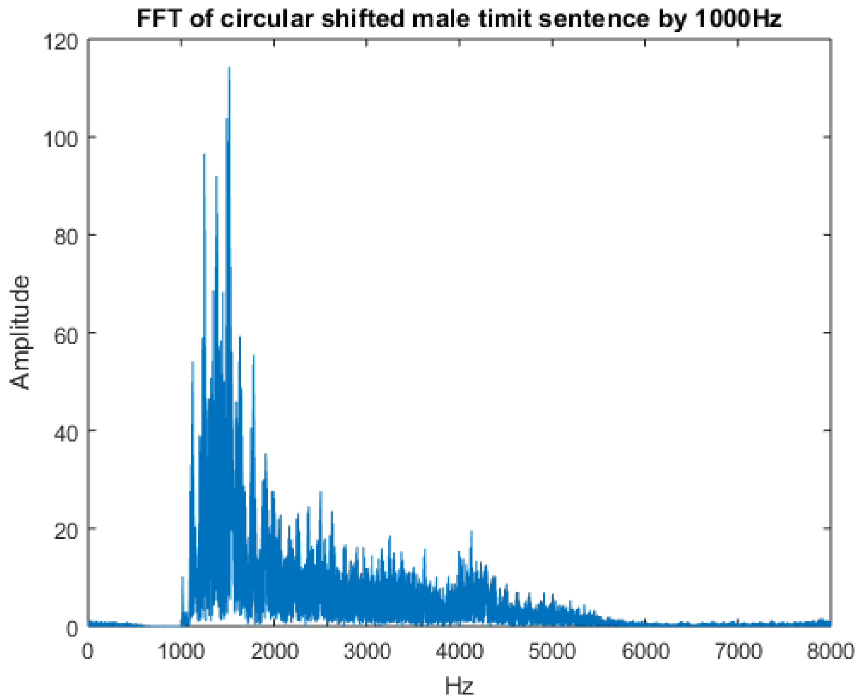

Figure 55 - Circular shifted real signal by 1000Hz

As can be seen in *Figure 55* the TIMIT database input sentence has had its frequencies successfully shifted by 1000Hz. The resulting amplitude has not been altered from the original input. The resulting time domain signal, resulting from the inverse DFT, seen below in *Figure 56* has the same length of ~3.5 seconds.

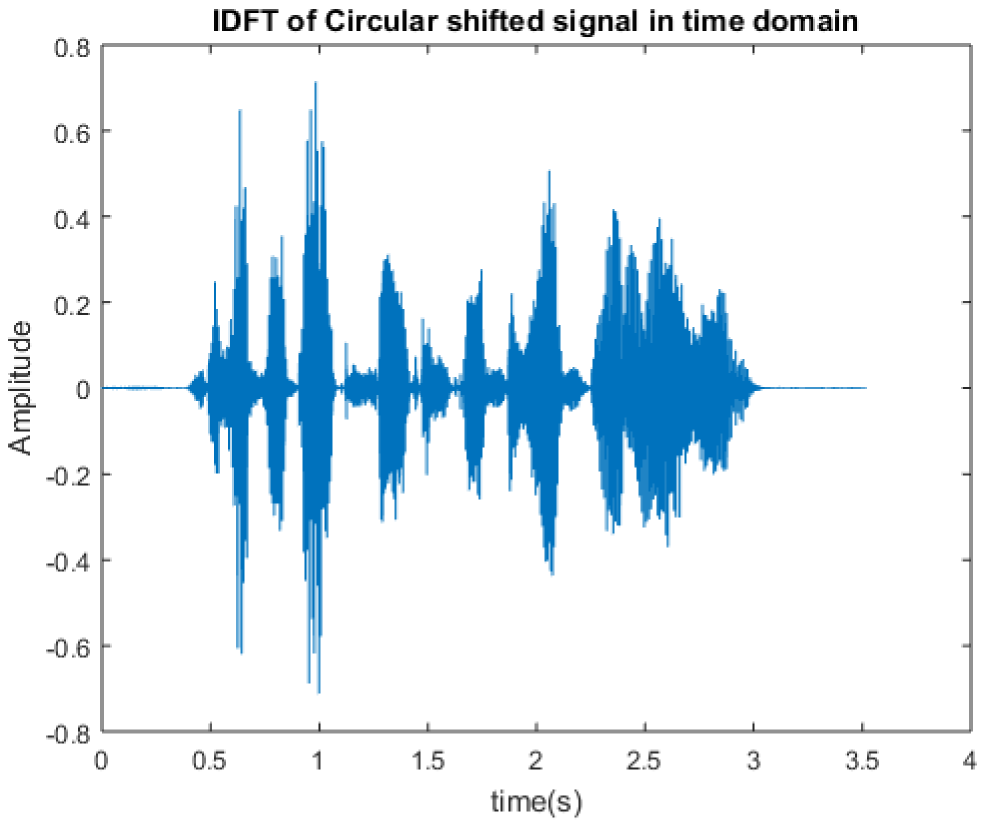

Figure 56 - Time domain shifted signal plot

The frequency of the sample input was shifted in frequency increments. 1Hz, 5Hz, 10Hz, 20Hz, 100Hz and 1000Hz. The observed inverse discrete Fourier transform was played back and each frequency was rated for quality by listeners.

Table 1- Circular shift for increasing frequency shift

Frequency shift(Hz)	Quality of playback
1	No change audibly heard in the signal
5	No change audibly heard in the signal
10	Small change in frequency observed
20	Detection of pitch change noticeable as well as slight metallic sound.
100	Pitch obviously changed and increased metallic echoing sound
1000	Pitch greatly changed with metallic sound and underwater affect.

The circular shift succeeded in shifting the pitch while maintaining the same audio length. However, it does not succeed in generating a natural human voice with metallic and underwater artefacts affecting the output.

4 Results & Discussion

The simple approach of re-sampling the audio succeeded in changing the frequency, but the resulting audio was sped up by the same factor as the resampling resulting in very unnatural audio. The voice produced by this shifting technique does not sound natural. The second approach of circular shifting the discrete time Fourier transform array succeeded in pitch shifting while maintaining the same audio length. The effect of the shift still however results in an abnormal sounding voice when the frequency is shifted by more than 20Hz.

4.1 Objectives Accomplished

The majority of this project is composed of work that has been achieved through the use of MATLAB. Some basic addition and Fourier transforms of pure sine waves was performed and the extraction of the individual signals was proved through the FFT and inverse FFT. Signals were circularly shifted in the frequency domain and then reproduced in the time domain to change the pitch. Voice analysis of TIMIT audio samples was also performed to identify harmonics and to visualise the frequency spectrum of speech.

4.2 Future Work

Linear predictive analysis (LPA) is a method to separate, source and filter components. It is used for encoding good quality speech at a low bit rate. This model is a close approximation of the speech production of the space between the vocal folds in human body. (Audiopedia, 2018)

$$\bar{s}(n) = \sum_{k=1}^{N} \alpha_k s(n-k)$$

Equation 5 - LPA formula

$\bar{s}(n)$ = predicted signal, N is the predicted order, $\propto_k$ = the predictor coefficients, $s(n)$ = the signal predicted. (Jung, 2003)

In this project it could be used to identify position formants in the human voice. Using deconvolution, the formants could be removed allowing the pitch to be changed and modified and to change the pitch to produce a ‘real’ sounding voice.

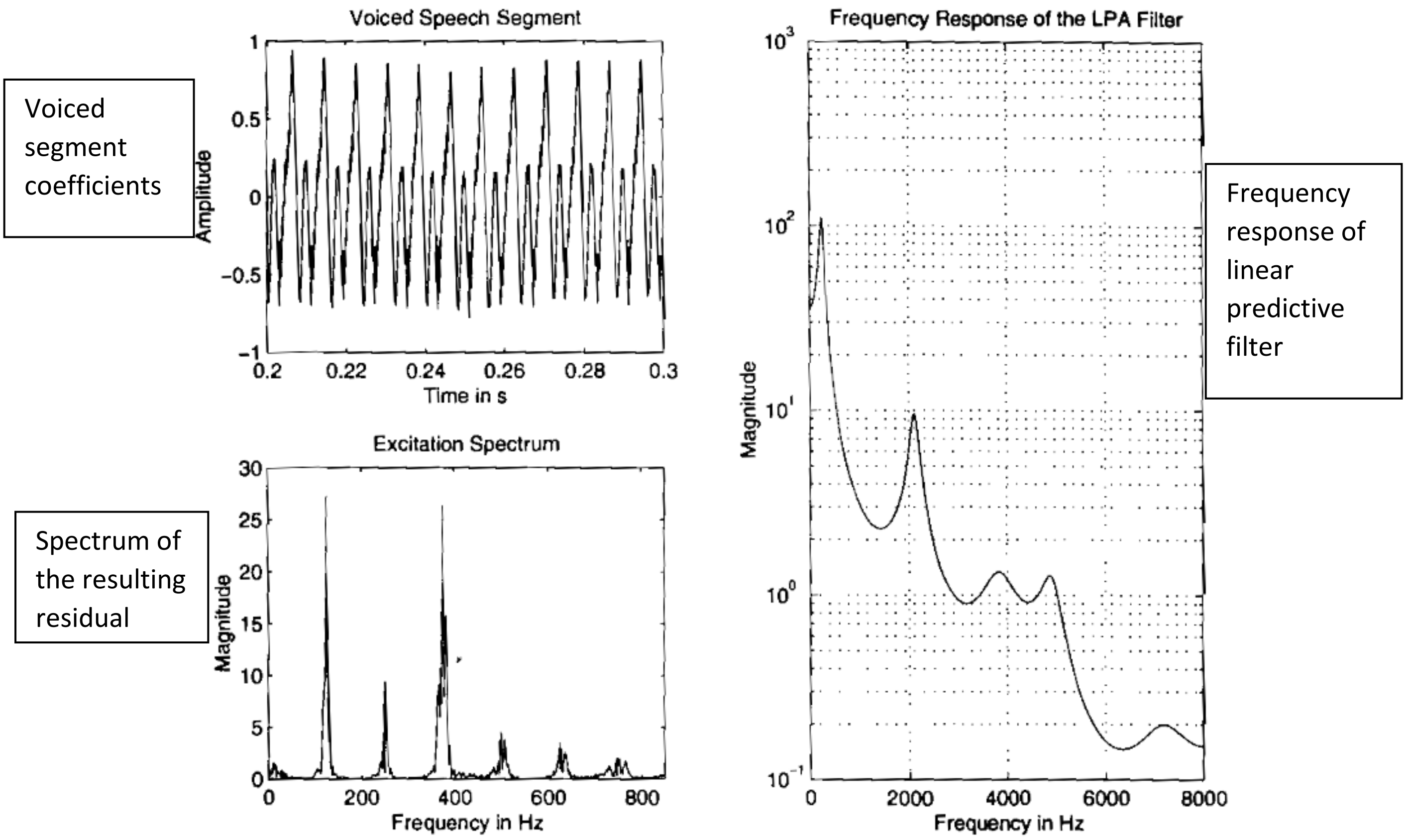

Figure 57 – LPA (Jung, 2003)

LPA deconvolution of an audio file by inverse filtering the segment with linear coefficients, the harmonic structure is clearly visible.

This harmonic structure can be utilised in speaker normalisation resulting in a more naturally sounding pitch shifted audio signal.

4.3 Ethics

Although the ethical issues of this project might not seem obvious, it should be said that the input and processing of audio data is a sensitive area when recording other people. If the program was to be used in a public domain, it would illegal not to observe the privacy of others upon releasing personal audio content without written consent.

Bibliography

Anon., 18. *www.analog.com.* [Online]

Available at: signals of interest: the input signal, the impulse response, and the output signal.

[Accessed 20 December 2018].

Anon., 2016. *moodle.nuim.ie.* [Online]

Available at:

https://2018.moodle.maynoothuniversity.ie/pluginfile.php/546407/mod_resource/content/0/Lec10_notes.pdf

[Accessed 2018 12 19].

Anon., 2017. [Online]

Available at: https://betterexplained.com/articles/an-interactive-guide-to-the-fourier-transform/

Anon., 2018. [Online]

Available at: https://en.wikipedia.org/wiki/Phonetics

Anon., 2018. [Online]

Available at: https://en.wikipedia.org/wiki/Fourier_series#cite_note-2

Anon., 2018. *aavos.eu.* [Online]

Available at: https://aavos.eu/glossary/fourier-transform/

Anon., 2018. *Electronic Tutorials.* [Online]

Available at: https://www.electronics-tutorials.ws/filter/filter_3.html

[Accessed 2018].

Anon., 2018. *OriginLab.* [Online]

Available at: https://www.originlab.com/doc/Origin-Help/Convolution

[Accessed 2018].

Anon., 2018. *Wikipedia.* [Online]

Available at: https://en.wikipedia.org/wiki/High-pass_filter

[Accessed 2018].

Anon., 2018. *Wikipedia.* [Online]

Available at: https://simple.wikipedia.org/wiki/Hand

[Accessed 15 September 2018].

Audio NTi, 2018. *FFT Fast Fourier Transform.* [Online]

Available at: http://www.nti-audio.com/en/functions/fast-fourier-transform-fft.aspx

Audiopedia, T., 2018. *Youtube.com.* [Online]
Available at: https://www.youtube.com/watch?v=TtIQAGHBKCY
[Accessed 20 December 2018].

Bhurane, A. A., 2018. *Matlab.* [Online]
Available at: https://uk.mathworks.com/matlabcentral/fileexchange/43642-demonstration-of-convolution-operation-graphically
[Accessed 12 November 2018].

Britannica, T. E. o. E., N/A. *Encyclopaedia Britannica.* [Online]
Available at: https://www.britannica.com/science/Broca-area
[Accessed 21 12 2018].

Byjus.com, 2018. *Byjus.com.* [Online]
Available at: https://byjus.com/physics/the-human-voice-how-do-humans-create-sound-with-their-vocal-cord/

Café, T. V., 2018. [Online]
Available at: https://www.google.com/url?sa=i&source=images&cd=&cad=rja&uact=8&ved=2ahUKEwibl-K2ja_fAhV9VBUIHdyTBpoQjRx6BAgBEAU&url=https%3A%2F%2Fwww.thevoicecafe.net%2FAccents_for_Actors%2FLearn-New-Zealand-Accent-Phonetics-Online.htm&psig=AOvVaw31pHOO8RTLCNBZpZ3Kue02
[Accessed 20 12 2018].

ElectronicsTutorials.ws, 2018. *ElectronicsTutorials.ws.* [Online]
Available at: https://www.electronics-tutorials.ws/filter/filter_3.html

ElectronicsTutorials.ws, 2018. *ElectronicsTutorials.ws.* [Online]
Available at: https://www.electronics-tutorials.ws/filter/filter_2.html

ElectronicsTutorials.ws, 2018. *ElectronicsTutorials.ws.* [Online]
Available at: https://www.electronics-tutorials.ws/filter/filter_4.html

Hannah, 2010. *Replicated Typo.* [Online]
Available at: http://www.replicatedtypo.com/why-a-lowered-larynx-is-good-for-human-speech-production/2697.html
[Accessed 17 12 2018].

Herod, E. M. H. I. a. J. V., 1997. [Online]
Available at: http://www.mathphysics.com/pde/filtering.html

Holden, C., 2004. *Science.* [Online]
Available at: http://science.sciencemag.org/content/303/5662/1316?fbclid=IwAR2IYF8ppn7uLwna555BaJ1-

T52jJHUmCrOD60rHDSjAxm8gnuOlg3y_rgk
[Accessed 17 12 2018].

Johnson, K., 2018. *www.linguistics.berkeley.edu.* [Online]
Available at: http://www.linguistics.berkeley.edu/~kjohnson/papers/revised_chapter.pdf
[Accessed 2018].

Juang, B.-H. & Rabiner, L., 2006. [Online]
Available at: https://en.wikipedia.org/wiki/Speech_processing
[Accessed 2006].

Jung, E., 2003. *Application of time-scale and frequency-scale modification algorithms to voice-gender conversion,* Dublin: Dublin Institute of Technology.

Jung, E., 2003. *arrow.dit.ie.* [Online]
Available at: https://arrow.dit.ie/cgi/viewcontent.cgi?article=1006&context=engmas
[Accessed 20 December 2018].

Jung, E., 2003. *Arrow.DIT.ie.* [Online]
Available at: https://arrow.dit.ie/cgi/viewcontent.cgi?article=1006&context=engmas
[Accessed 6 January 2003].

Katz, W. F., 2018. [Online]
Available at: https://www.dummies.com/education/language-arts/grammar/phonetics-for-dummies-cheat-sheet/

Kephart, R., 2010. *CrankyLinguist.Blogspot.com.* [Online]
Available at: http://crankylinguist.blogspot.com/2010/10/chimp-vs-human-vocal-tracts.html

Kincaid, J., 2018. *Medium.* [Online]
Available at: https://medium.com/descript/a-brief-history-of-asr-automatic-speech-recognition-b8f338d4c0e5
[Accessed 12 June 2018].

Lawlor, D. B., 2018. *Maynooth University Moodle.* [Online]
Available at:
https://2019.moodle.maynoothuniversity.ie/pluginfile.php/199567/mod_resource/content/0/1%20Fourier%20series.pdf
[Accessed 20 December 2018].

M.H., H., 1999. [Online]
Available at: https://en.wikipedia.org/wiki/Discrete_Fourier_series

Meiniar, W. et al., 2018. *ieeexplore.ieee.org.* [Online]

Available at: https://ieeexplore.ieee.org/document/8272563#full-text-section

[Accessed 21 November 2017].

Morrison, N., 1994. [Online]

Available at: http://mathworld.wolfram.com/FourierSeries.html

Parissis, Y., 2011. [Online]

Available at: https://yannisparissis.wordpress.com/2011/03/10/dmat0101-notes-3-the-fourier-transform-on-l1/

Pinterest.ie, 2018. *Pinterest.ie.* [Online]

Available at: https://www.pinterest.ie/pin/389631805252153022/

Priyanka A. Abhang, .. S. C. M., 2016. [Online]

Available at: https://www.sciencedirect.com/topics/neuroscience/speech-processing

Ríkharðsson, S., 2014. *How many accents are there in the British Isles and North America?.* [Online]

Available at: https://www.quora.com/How-many-accents-are-there-in-the-British-Isles-and-North-America

Santarossa, A., 2017. *News.com.au.* [Online]

Available at: https://www.news.com.au/sport/sports-life/why-john-cena-is-the-best-of-all-time/news-story/c9e166b82caa2ba63e01fc3e3bd40293

[Accessed 4 January 2017].

Shenoi, B. A., 2005. [Online]

Available at: https://en.wikipedia.org/wiki/Filter_(signal_processing)

Suri, S., 2014. *TheConversation.com.* [Online]

Available at: http://theconversation.com/what-sign-language-teaches-us-about-the-brain-29628

Translations, D., 2018. *World Languages.* [Online]

Available at: https://www.daytranslations.com/world-languages

Webster, M., 2018. *Phonetics.* [Online]

Available at: https://www.merriam-webster.com/dictionary/phonetics

Works, V. S., Unknown. *Voice Science Works.* [Online]

Available at: https://www.voicescienceworks.org/harmonics-vs-formants.html

[Accessed 19 12 2018].

Xiao, Y., 2011. *Projecthea.org.* [Online]

Available at:

https://www.projectrhea.org/rhea/index.php/Homework_7,_ECE438,_Fall_2011,_Yimin_Xiao%27s_solution

[Accessed 2018].

Appendix

MATLAB code

Resampling audio

```
[signaly,Fs] = audioread('cmu_us_ked_timit\wav\kdt_001.wav');

audiowrite('reSampledAudio.wav',signaly,Fs*2)

signaly = transpose(signaly)

T = 1/Fs;              % Sampling period

t = 0:T:(length(signaly)*T)-T; %the length of time in the file

plot(t,signaly)

title('Timit male voice signal in time')

xlabel('time(s)')

ylabel('Amplitude')

%sound(signaly) % PLay Original sound

figure

xdft = fft(signaly);

xdft = xdft(1:length(signaly)/2+1);

xdft = xdft/length(signaly);   %normalise power of signal

xdft(2:end-1) = 2*xdft(2:end-1);

freq = 0:Fs/length(signaly):Fs/2;  %Frequency spacing

plot(freq,abs(xdft))

title('FFT of audio at 16000 samples/s')

xlabel('Hz')

ylabel('Amplitude')

audiowrite('reSampledAudio.wav',signaly,Fs*2)

[reSampled,Fs_new] = audioread('reSampledAudio.wav');

sound(reSampled,Fs_new)
```

```
Fs_new
figure
xdft = fft(reSampled);
xdft = xdft(1:length(reSampled)/2+1);
xdft = xdft/length(reSampled);   %normalise power of signal
xdft(2:end-1) = 2*xdft(2:end-1);
freq = 0:Fs_new/length(reSampled):Fs_new/2;  %Frequency spacing
plot(freq,abs(xdft))
title('FFT of Re-sampled audio at 32000 samples/s')
xlabel('Hz')
ylabel('Amplitude')
```

4.3.1 Circular Pitch Shift

```
[signaly,Fs] = audioread('cmu_us_ked_timit\wav\kdt_001.wav');
signaly = transpose(signaly)
T = 1/Fs;             % Sampling period
t = 0:T:(length(signaly)*T)-T; %the length of time in the file
plot(t,signaly)
title('Timit male voice signal in time')
xlabel('time(s)')
ylabel('Amplitude')
%sound(signaly) % PLay Original sound
figure
xdft = fft(signaly);
xdft = xdft(1:length(signaly)/2+1);
%xdft = xdft/length(signaly);   %normalise power of signal
%xdft(2:end-1) = 2*xdft(2:end-1);
```

```
freq = 0:Fs/length(signaly):Fs/2;  %Frequency spacing

plot(freq,abs(xdft))

title('FFT of male timit input sentence')

xlabel('Hz')

ylabel('Amplitude')

%figure, plot(freq, angle(xdft)), title('Phase plot');

xdft = xdft*length(signaly);   %change power back to original

xdft(2:end-1) = xdft(2:end-1)/2;

xDFTwitoutDCgain= xdft(2:end); % Remove DC gain from begining of Array

frequencyShiftHz = 20;  %hz to shift. Positve or negative

percentToShift = 0.05;  % percent of Frequency bands to shift each componenet.

ratioShift = ceil((length(xdft)/100) * percentToShift)

frequencyShift = ceil((length(signaly)/Fs)*frequencyShiftHz) %calculate shift by
size of DFT frequency bins

circShiftedSignalDFT = circshift(xDFTwitoutDCgain,frequencyShift);  %circular shift

flipped = conj(fliplr(circShiftedSignalDFT)); % Flip array and get conjugate of each
component

newWhole = [xdft(1) circShiftedSignalDFT flipped ]; %Add back in DC gain and -/+
freq elements

freqShiftedSound = ifft(newWhole);

newWhole = newWhole(1:length(signaly)/2+1);  %Remove negative symmetrical part of
FFT for plotting

plot(freq,abs(newWhole))

title('FFT of Circular shifted male timit sentence by 1000Hz')

xlabel('Hz')

ylabel('Amplitude')

sound(freqShiftedSound,Fs)

plot(t,freqShiftedSound)

title('IDFT of Circular shifted signal in time domain')
```

```
xlabel('time(s)')

ylabel('Amplitude')
```

4.3.2 Band-pass

```
addpath('C:\Users\12721095\OneDrive - Maynooth University\EE301\')

[inputSignal,Fs] = audioread('cmu_us_ked_timit\wav\kdt_001.wav');

Fs = 16000;             % Sampling frequency

T = 1/Fs;              % Sampling period

%dt = 1/Fs; %change in time

%t = 0:dt:(length(inputSignal)*dt)-dt; %the length of time in the file

L = 16000;             % Length of signal

t = (0:L-1)*T;        % Time vector

noise = 1*sin(2*pi*7000*t);

%signaly = transpose(noise)+ inputSignal;

signaly = 1*sin(2*pi*1000*t) + 1*sin(2*pi*2000*t) + 1*sin(2*pi*3000*t)+
1*sin(2*pi*4000*t);

xdft = fft(signaly);

xdft = xdft(1:length(signaly)/2+1);

xdft = xdft/length(signaly);   %normalise power of signal

xdft(2:end-1) = 2*xdft(2:end-1);

freq = 0:Fs/length(signaly):Fs/2;  %Frequency spacing

plot(freq,abs(xdft))

title('FFT with noise')

xlabel('Hz')

ylabel('Amplitude')

xdft = fft(signaly);

xdft = xdft(1:length(signaly)/2+1);

xdft = xdft/length(signaly);   %normalise power of signal
```

```
xdft(2:end-1) = 2*xdft(2:end-1);

freq = 0:Fs/length(signaly):Fs/2;  %Frequency spacing

plot(freq,abs(xdft))

title('FFT with noise')

xlabel('Hz')

ylabel('Amplitude')

bpFilt = designfilt('bandpassfir','FilterOrder',20, ...

         'CutoffFrequency1',1999,'CutoffFrequency2',2999, ...

         'SampleRate',Fs, 'Window','hamming');

fvtool(bpFilt)

B = bpFilt.Coefficients

 signaly = filter(bpFilt,signaly)


xdft = fft(signaly);

xdft = xdft(1:length(signaly)/2+1);

xdft = xdft/length(signaly);   %normalise power of signal

xdft(2:end-1) = 2*xdft(2:end-1);

freq = 0:Fs/length(signaly):Fs/2;  %Frequency spacing

plot(freq,abs(xdft))

title('FFT of bandpass filtered signal')

xlabel('Hz')

ylabel('Amplitude')
```

4.3.3 High-pass filter

```
addpath('C:\Users\12721095\OneDrive - Maynooth University\EE301\')

[inputSignal,Fs] = audioread('cmu_us_ked_timit\wav\kdt_001.wav');

Fs = 2100;             % Sampling frequency

T = 1/Fs;              % Sampling period
```

```
dt = 1/Fs; %change in time

%t = 0:dt:(length(inputSignal)*dt)-dt; %the length of time in the file

L = 250;

t = (0:L-1)*T;        % Time vector

noise = 1*sin(2*pi*7000*t);

%signaly = transpose(noise)+ inputSignal;

signaly = 1*sin(2*pi*50*t) + 1*sin(2*pi*1000*t)

plot(t, signaly)

title('50Hz + 1Khz signal in time')

xlabel('time(s)')

ylabel('Amplitude')

xdft = fft(signaly);

xdft = xdft(1:length(signaly)/2+1);

xdft = xdft/length(signaly);   %normalise power of signal

xdft(2:end-1) = 2*xdft(2:end-1);

freq = 0:Fs/length(signaly):Fs/2;  %Frequency spacing

plot(freq,abs(xdft))

title('Fourier Transform of 1 kHz Signal with 50Hz Noise')

xlabel('Hz')

ylabel('Amplitude')

filterFreq = 1000;

stopbandW = filterFreq * ((2 * pi)/Fs);

hpFilt = designfilt('highpassfir','StopbandFrequency',0.25, ...

         'PassbandFrequency',0.32,'PassbandRipple',4, ...

         'StopbandAttenuation',65,'DesignMethod','kaiserwin');
```

```
fvtool(hpFilt)

signaly = filter(hpFilt,signaly);

xdft = fft(signaly);

xdft = xdft(1:length(signaly)/2+1);

xdft = xdft/length(signaly);   %normalise power of signal

xdft(2:end-1) = 2*xdft(2:end-1);

freq = 0:Fs/length(signaly):Fs/2;  %Frequency spacing

plot(freq,abs(xdft))

title('Fourier Transform of with 50Hz noise filtered')

xlabel('Hz')

ylabel('Amplitude')
```

4.3.4 Audio at different sample rates

```
[y,Fs] = audioread('kdt_001.wav') %extracts file data to y at sampling rate Fs
Fs_half = Fs/2;
audiowrite('halfSpeed.wav',y,Fs_half);
[y_half,Fs_half] = audioread('halfSpeed.wav');

dt = 1/Fs; %change in time
t = 0:dt:(length(y)*dt)-dt; %the length of time in the file

figure
plot(t,y);
hold on

dt_half = 1/Fs_half; %change in time
t_half = 0:dt_half:(length(y_half)*dt_half)-dt_half; %the length of time in
the file

plot(t_half,y_half);
xlabel('Time (seconds)');
ylabel('Amplitude') % plot in the time domain
title('Voice Sample at Different Sample Rates')
legend('Orginal sample','Half sample rate')
```

4.3.5 Frequency spectrum of audio at different sample rates

```
[y,Fs] = audioread('kdt_001.wav') %extracts file data to y at sampling rate Fs
Fs_half = Fs/2;
audiowrite('halfSpeed.wav',y,Fs_half);
[y_half,Fs_half] = audioread('halfSpeed.wav');

%Fourier Transform
xdft = fft(y);
xdft = xdft(1:length(y)/2+1);
xdft = xdft/length(y);   %normalise power of signal
xdft(2:end-1) = 2*xdft(2:end-1);
freq = 0:Fs/length(y):Fs/2;  %Frequency spacing

figure
plot(freq,abs(xdft))
hold on

%Fourier Transform
xdft = fft(y_half);
xdft = xdft(1:length(y_half)/2+1);
xdft = xdft/length(y_half);   %normalise power of signal
xdft(2:end-1) = 2*xdft(2:end-1);
freq = 0:Fs_half/length(y_half):Fs_half/2;  %Frequency spacing

plot(freq,abs(xdft))
xlabel('Frequency (Hz)')
ylabel('Amplitude')
title('Frequency Content-Different Sample Rates')
legend('Orginal sample','Half sample rate')
```

4.3.6 Sampling and aliasing code

```
n = 0:100;
Fs = 5000; % Sample frequency at 5000 samples/second
Ts = 1/Fs;
t = n*Ts;    % This generates a discrete time vector.

xd = sin(2*pi*100*t); % Pure sine wave at 100Hz

figure
plot(t,xd)
xlabel('Time (seconds)');
ylabel('Amplitude')
title('100Hz Sine Wave ');

figure
stem(t,xd)
title('100Hz Sampled Sine Wave');
xlabel('Samples');
```

```
ylabel('x[n]');
legend('5000 Samples/s')

n = 0:20;
Fs = 1000; % Sample frequency at 1000 samples/second
Ts = 1/Fs;
t = n*Ts;    % This generates a discrete time vector.

xd = sin(2*pi*100*t); % Pure sine wave at 100Hz

figure
plot(t,xd)
hold on

n = 0:10;
Fs = 500; % Sample frequency at 500 samples/second
Ts = 1/Fs;
t = n*Ts;    % This generates a discrete time vector.

xd = sin(2*pi*100*t); % Pure sine wave at 100Hz

plot(t,xd)
hold on
n = 0:4;
Fs = 175; % Sample frequency at 175 samples/second
Ts = 1/Fs;
t = n*Ts;    % This generates a discrete time vector.

xd = sin(2*pi*100*t); % Pure sine wave at 100Hz

plot(t,xd)
hold on
n = 0:4;
Fs = 150; % Sample frequency at 150 samples/second
Ts = 1/Fs;
t = n*Ts;    % This generates a discrete time vector.

xd = sin(2*pi*100*t); % Pure sine wave at 100Hz

plot(t,xd)
hold on
title('Examples of Aliasing');
xlabel('Samples');
ylabel('x[n]');
legend('1000 Samples/s','500 Samples/s','175 Samples/s','150 Samples/s')
hold off

%Fourier transform
xdft = fft(xd);
xdft = xdft(1:length(xd)/2+1);
xdft = xdft/length(xd);   %normalise power of signal
xdft(2:end-1) = 2*xdft(2:end-1);
freq = 0:Fs/length(xd):Fs/2;  %Frequency spacing
```

```
plot(freq,abs(xdft))
xlabel('Hz')
ylabel('Amplitude')
hold off
```