FINAL YEAR PROJECT REPORT
BACHELOR OF ENGINEERING

Noise Cancellation with Deep Networks

Student:	Elise ATKINSON
GUID:	2322854A
1^{st} Supervisor:	Bernd Porr
2^{nd} Supervisor:	Martin Lavery

2021-2022

Coursework Declaration and Feedback Form

This Student should complete and sign this part

Student Name: Elise ATKINSON	Student GUID: 2322854A
Course Code : ENG4110P	Course Name : INDIVIDUAL PROJECT 4
Name of 1st Supervisor: Bernd Porr	Name of 2nd Supervisor: Martin Lavery
Title of Project: **Noise Cancellation with Deep Networks**	
Declaration of Originality and Submission Information	
I affirm that this submission is all my own work in accordance with the University of Glasgow Regulations and the School of Engineering requirements *Signed (Student) : Elise Atkinson*	E N G 4 1 1 0 P
Date of Submission : 24/04/2022	
Feedback from Lecturer to Student – to be completed by Lecturer or Demonstrator	
Grade Awarded: Feedback (as appropriate to the coursework which was assessed):	
Lecturer Demonstrator:	Date returned to the Teaching Office:

Abstract

Audio signals are highly contaminated due to the nature of humans' bustling lifestyle and noise polluting actions. This report presents a deep learning algorithm which learns in real-time how to effectively cancel out frequencies regarded as noise from an input signal. This is demonstrated through filtration of a bass guitar acting as reference noise from a noise-polluted podcast signal containing the desired sound. Standardised microphones are used to obtain the recording and an 'off-the-shelf' DNN is used in conjunction with other electronic components to make up the deep neural filter (DNF). This was effective in improving the signal to noise ratio (SNR) of the audio signals by 96.97dB in comparison to other adaptive algorithms. The DNF improved noise reduction over a wide bandwidth of frequencies, and with some fine-tuning of the layer and learning rate relationship, this DNF has the potential to improve this SNR further for use across multiple disciplines with unpredictable noise interference.

Acknowledgements

Firstly, I would like to thank my project supervisor Bernd Porr for all his advice and help throughout my project. Also, a special thank you to Sama Daryanavard for her initial words of wisdom regarding the deep neural network. Last but not least, I'd like to thank all my friends and family for their continued support in my studies and helping me through a difficult year.

Contents

1 Introduction

Audio signals are contaminated with a multitude of background noise, enhanced by humans' technologically advancing lifestyle to enable growth of noise pollution in our environment (Liu et al., 2020). This excessive background noise can be damaging to health, especially within neuro-divergent communities or for individuals with hearing disabilities. Applying Deep Neuronal Networks (DNN) for noise cancellation in audio (such as noise-cancelling headphones) provides a wide range of benefits; stemming from listening to music, to health services such as helping autistic children (Ikuta et al., 2016) and use within hearing aids (Levitt, 2001).

Currently, to tackle noise within headphones, passive and active noise cancellation techniques are implemented with varying degrees of success (Shalool, Zainal, Beng Gan, & Umat, 2016). Active noise cancellation (ANC) produces a signal opposite to the noise present so destructive interference occurs, with specific improvement of low-frequency noise mitigation (S. Kuo & Morgan, 1999). Passive noise cancellation (PNC) relies on the physical properties of the headphone such as the shape of the headphones themselves as well as the material they are made from and whether they reflect or absorb sound. PNC improves mitigation of noise at higher frequencies but is inefficient at cancelling lower frequencies.

In using ANC, the signal to noise ratio (SNR) within the audio signal has been show to improve, with the addition of PNC improving this further (Shalool et al., 2016). However, superposition of sound for noise cancellation isn't fully effective as noise that is in phase with the desired signal causes constructive interference within ANC circuits, hence amplifying the unwanted noise. This amplification can also lead to a pressure build-up in the headphone cup causing discomfort within users. Currently this ANC dominates the market (S. M. Kuo, Chen, Chang, & Lai, 2018), but superior technologies such as least mean square (LMS) and other adaptive algorithms have been investigated for noise cancellation. The LMS adaptive algorithm works

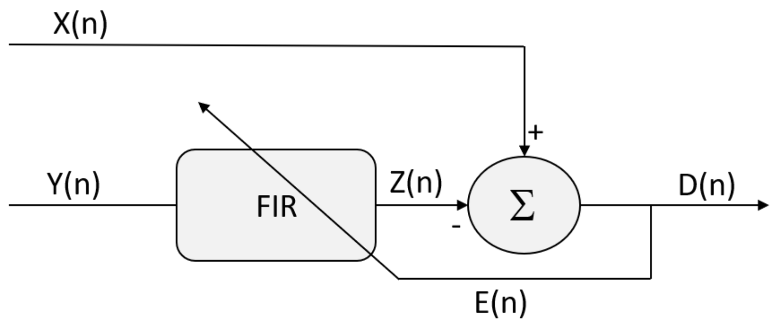


Figure 1: LMS Adaptive Filter

by using adjustable weighting coefficients which multiply the noise reference signal ($y[n]$) to reduce the mean-squared error via back-propagation, subtracting this generated canceller ($z[n]$) from the input signal ($x[n]$) to provide a desired output ($d[n]$). A delay line is implemented relating to the number of samples contained in the impulse response of a filter, with the finite response running for a number of 'taps'. These 'taps' represent the number of weighted coefficients within the filter, and this in conjunction with the LMS adaptive algorithm creates the LMS adaptive filter (Widrow et al., 1976). The LMS filter works in real-time to filter the signal sample by sample by using a reference noise to tune the weighted coefficients, which removes the unwanted frequencies.

Multiple approaches based on the LMS adaptive algorithm have been used to tackle noise present in audio applications (Douglas, 1997) (Elhossini, Areibi, & Dony, 2006), and whilst the use of LMS algorithms increase the signal-to-noise ratio (SNR) compared with active and passive noise cancellation techniques, issues still occur when power of the signal vastly fluctuates (Greenberg, 1998). This is due to the linear processing nature of the LMS algorithm and so distortions in the signal occur when non-linearity's are present - a common occurrence when dealing with audio signals. Other adaptive algorithms such as tangential hyperbolic function based LMS (FxLMS) have improved on managing these non-linearity's but are only successful for signals with minor non-linearity's (Ghasemi, Kamil, & Marhaban, 2016). Similarly to

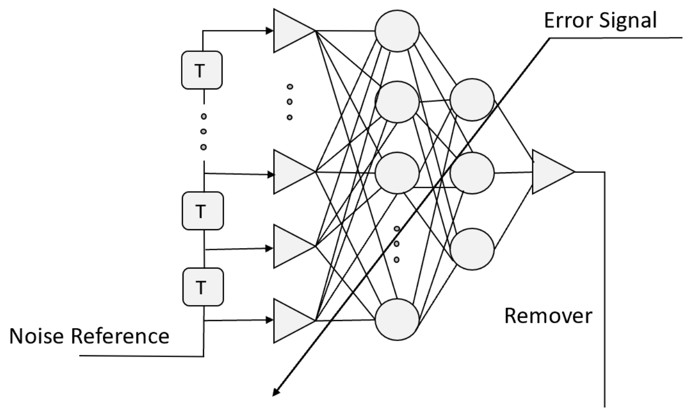


Figure 2: Deep Neuronal Network Diagram

the LMS algorithm, deep learning is driven by back-propagation of the error signal (otherwise known as the "teaching signal") through the network. The deep neuronal network (DNN) propagates its error signal through a non-linear network containing multiple layers to actively learn which frequencies to cancel out (Porr et al., 2020). This works via a canceller which alters the weights, minimising the overall mean-square error via a gradient descent algorithm. The canceller signal is generated within the neural network using non-linear activation functions, however this process can become heavily computer-intensive as demonstrated in function-link neural networks (Jafarifarmand & Badamchizadeh, 2013).

Adding a time delay to the input of the DNN (shown in figure 2) allows it to learn the characteristics of the noise signal and adjust the weights to react accordingly. The learning rate defines how quickly the weights adapt to minimise the error within the DNN (Porr, n.d.) and is related to the amount of samples within the signal, where in audio signals the most common sample rate is 44.1kHz. The learning rate should be always slower than the sample rate so that it can be viewed as a constant value for a real-time application.

Due to the non-linear model of multiple layers, the DNN is proven to cancel out signals featuring non-linearity's more effectively than its counterpart algorithms such as the FxLMS (Cichocki, Vorobyov, & Rutkowski, 2000). Subsequently, the SNR has been improved using the deep neuronal filter (DNF) compared to the LMS by up to 10dB according to previous literature (Porr et al., 2020). Where LMS filters are mainly effective in noise cancellation of lower frequency ranges (Flotte-Hernandez et al., 2008), attenuation of noise has been shown in using a Deep Neuronal Filter (DNF) where effective noise cancellation occurs through a wide range of frequencies (Daryanavard, Porr, & Dahiya, 2022). Suggesting that the application of a DNF for processing audio signals could improve the SNR, as the non-linearity's within the noise over a larger bandwidth of frequencies will be mitigated.

In this paper noise cancellation with deep networks is explored in relation to audio applications. The DNF implemented for my project is developed from 'Real-time noise cancellation of EEG signals' (Porr et al., 2020), where a standard neural network (Daryanavard et al., 2022) was used to improve the SNR of electroencephalogram (EEG) signals in the brain. This project is adapted for audio signals to mitigate noise across a wide range of frequencies in real-time and cancel out non-linearity's within the signal, hence improving the SNR.

Previous difficulties when implementing a DNF for noise cancellation include a highly computer-intensive design that isn't practical for compact applications such as noise cancelling headphones (Braun, Gamper, Reddy, & Tashev, 2021). My solution will be the implementation of a standard neural network (Daryanavard et al., 2022), which isn't as computer-intensive as previous models mentioned. The pre-training of neural networks with expected environmental noises has been explored to improve the accuracy at which the DNF recognises what signals are to be considered noise (Lee et al., 2018). In my experimentation the bandwidth at which the noise frequencies occur are specified, but in real-world situations, pre-training is beneficial as the noise source frequencies are unpredictable. This has some draw-backs as it is time-consuming and may be cost-intensive for initial commercialisation of noise cancelling headphones, but is necessary to deal with the inconsistent noise signals present in the real-world.

A stereo recording was made where one channel ideally contains the noise only ($y[n]$) (as the DNF exploits this assumption) and the other the raw input signal ($x[n]$) and its associated noise. The error signal ($e[n]$) for the deep network is equal to the output from the filter, and through back-propagation it removes any correlated frequencies between both input channels. Note we assume that the noise recorded in both channels is identical so that the DNN can learn this from the reference noise channel and remove from the raw input signal channel. Realistically, the noise recorded in each channel will be different (Zhang & Wang, 2021) due to physical parameters such as microphone placement, electrical noise associated with physical components and distance between the desired signal source and noise source (Vesely, Dolci, & Dolci, 2018). As the DNN exploits the fact that the noise within both channels are the same, physical distance could cause a weak commonality correlation between the signals. As neural networks learns the strongest correlation between the signals (Widrow & Lehr, 1990), it is important that the noise recorded in each channel is similar, therefore in my experimentation I will assume that the noise recorded is the same between both channels. On account of the potential weak commonality between the recorded noise, sections of the desired signal may also be interpreted as noise which will subsequently be cancelled out. This was illustrated previously with noise removal in an EEG experiment, as there is an amplitude reduction in the desired signal due to a weak correlation between the inner and outer electrodes, resulting in partial removal of the desired signal (Daryanavard et al., 2022).

2 Method and Experimental Setup

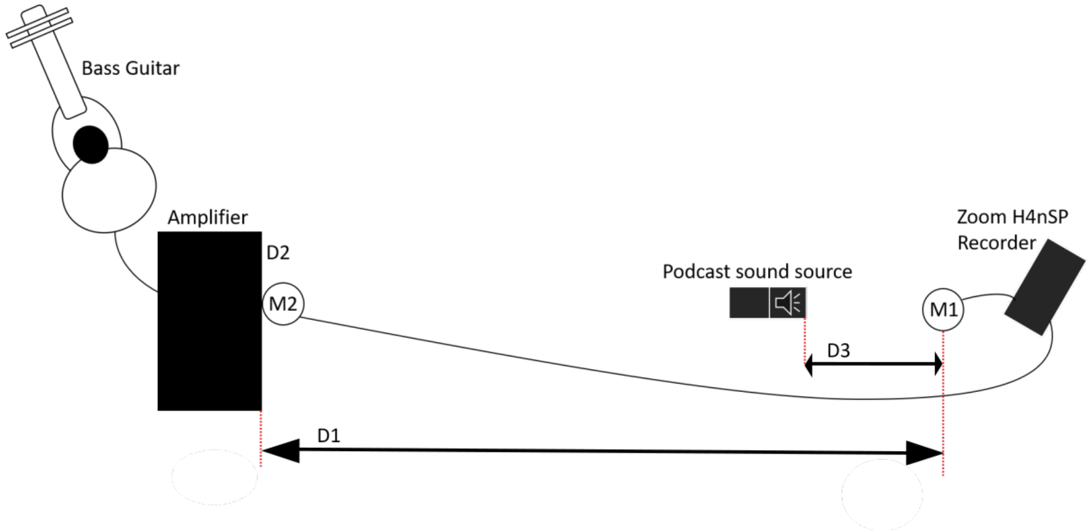


Figure 3: Experimental Setup

A stereo recording of 34 seconds was taken by identical microphones (*sdpc2*, 2022) into an analogue to digital converting device (*H4nsp Handy Recorder*, n.d.) of a bass guitar (*Milestone Series Peavy Bass*, n.d.) and a podcast (*The Artists of the Animal Kingdom*, 2021) being played from a mobile (*Pixel3a*, 2019), with the aim of removing the bass sound from the podcast. The bass acts as the reference noise to train the network that will be removed from the raw signal creating the desired output signal of only the podcast. The stereo track was split into two mono tracks at a sample rate of 44100 Hz with the bass recorded into channel 2 via microphone 2 and the podcast recorded into channel 1 via microphone 1, both with directional heads to localise the recording to the source.

The 'E' string - with a frequency of around 41Hz - was played on the bass guitar as this is easily distinguishable from the vocal range frequencies of the human voice in the podcast. This distinction between frequencies of the noise reference and raw signal is useful for initial experimentation, however for practical applications, this distinction is unrealistic. For example, when using noise cancelling headphones in the street, the background noise frequencies present are more unpredictable than the frequency of a bass guitar string. Applying this example to the DNF, the more complex the reference noise gets, an increase in the learning rate will be required for the DNF to learn characteristics of the noise signal.

Distances between the equipment were measured as:

- D1 = 0.15m : Noise source to Microphone 1.
- D2 = 0m : Bass amplifier to Microphone 2, negligible as microphone was placed directly onto amplifier.
- D3 = 0.015m : Podcast sound source to Microphone 1.

Increasing the distance between the noise source and the raw signal source would reduce the amount of noise present in the desired signal, however, other issues would occur such as a weak correlation between the noise recorded in each channel. The neural network will remove noise correlated between channels - so it is important to curb a weak correlation. Note it is assumed the signal is constant throughout filtering. As audio recording is executed with electrical components, the impedance of the microphones and any electrical noise associated with their physical components must be mitigated. To further reduce practical imperfections, identical microphones are chosen to record both channels and the experiment was undertaken in a quiet recording environment. In practical applications, the DNF must overcome external noisy environments so the idealistic conditions for obtaining the reference noise will not be possible.

2.1 Preliminary LMS Experiment

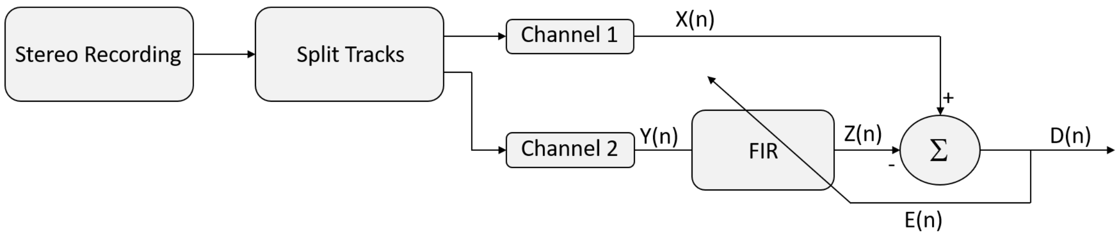


Figure 4: LMS Adaptive Filter with Mono Channel Inputs

As the DNF is based upon the architecture of the LMS algorithm, my initial experimentation tested the LMS filter for noise-cancellation in audio applications.

The mono tracks were converted via a python script into data files (each with approximately $1.5 \cdot 10^6$ samples) for digital processing. The data files contain both the timestamp and the data in separate columns, where in the code it is specified to use the data column for signal processing. To test that the code written for the LMS was functioning correctly, the mono recording of channel 1 ($x[n]$) was read in, and a dummy 50Hz sine wave was added to this signal. This 50 Hz will act as the noise to be removed from channel 1 and is input numerically into the code as the noise reference. Based on previous literature (Widrow et al., 1976), the LMS filter is expected to work effectively in using the reference noise as a canceller signal ($z[n]$) to remove the noise from the input signal, leaving only the desired podcast signal ($d[n] = e[n]$).

Once the testing of the filter with the dummy 50Hz is complete, the bass signal in channel 2 ($y[n]$) was then read in as the reference noise which is passed through the FIR filter, returning the canceller signal ($z[n]$) containing all elements common to both channels. From this similar results to the 50Hz test code are expected, where the bass noise should be removed from the podcast signal via the LMS algorithm.

An issue that may arise from the way the audio files are recorded, is that the sound that we want to hear ($d[n]$) could be partially removed as the canceller signal ($z[n]$) may have a weak commonality correlation with the noisy input signal ($x[n]$), i.e. sections of the desired signal can be interpreted as noise too. This was found to also be the case in previous experiments (Daryanavard et al., 2022) when using the LMS filter and is something that can't be expected to be omitted completely.

2.2 DNF Experiment

Once the LMS filter was working successfully, this was built upon by implementing the DNF in code for noise cancellation. Unlike the LMS filter, the DNF can deal with non-linearity's in the signal due to its model of multiple layers within the network.

The data was achieved with the same method of recording as described in section 2; the podcast and associated noise into channel 1 ($x[n]$), and the bass into channel 2 ($y[n]$). The DNF works

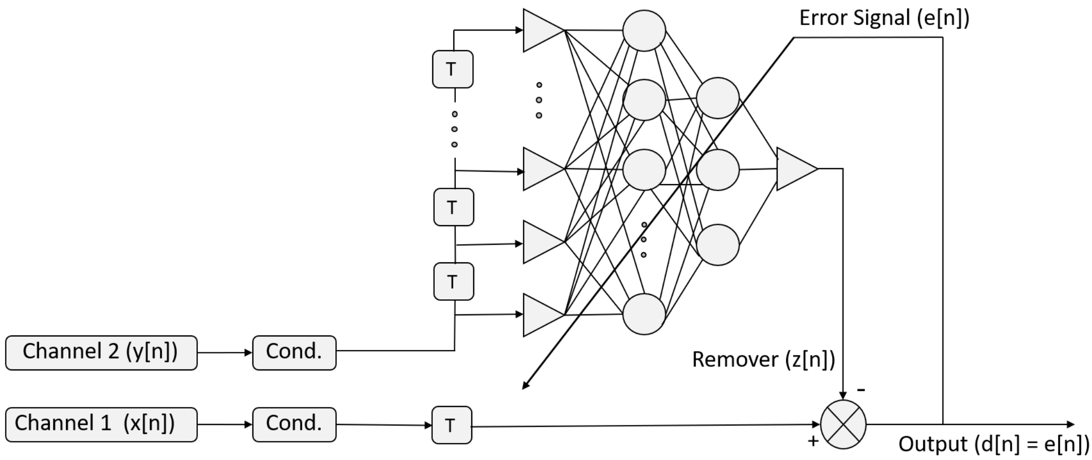


Figure 5: Deep Neuronal Filter Diagram

by removing any signal that is correlated between channel 1 and channel 2, assuming channel 2 contains only noise and none of the desired signal. This may not be the case due to the distance of the microphones being fairly close together (0.15m), so some of the desired signal may be interpreted as noise in the DNF's learning. Similarly to the LMS algorithm, the error signal is back-propagated through the DNN to act as the teaching signal so it learns to remove the correlated noise between the two channels. Note that the diagram in 2.2 shows for offline applications, where in live applications, an analogue to digital converter (ADC) would be placed at each channel output before the signal is processed digitally.

The signals are read in via .tsv files and are then filtered using separate high-pass Butterworth filters for channel 1 and channel 2. For channel 1 containing the raw signal, the cutoff frequency (f_{HP1}) was set to 20,000Hz which is the highest frequency the human ear can hear, so any frequency above this can be disregarded for the purposes of audio noise cancellation. For channel 2 containing the noise, the cutoff frequency (f_{HP2})was set to 20Hz which is at the lowest range of human hearing. Multiple sensible values were trialed for channel 2 cutoff frequency such as 41Hz; the frequency of the 'E' string on the bass, and 4000 Hz; the higher frequencies of overtones played by the bass guitar. However, cutoff frequency equal to 20Hz gave the result where the DNF output best matched the desired signal. In previous works (Daryanavard et al., 2022) DC mains removal at 50 Hz occurred, but since audio signals are being processed where some desired signals may contain tones equal to 50Hz, the DC mains removal was discarded as this would interfere with the desired signal output. The gains of channel1 ($x[n]$), channel2 ($y[n]$) and the canceller ($z[n]$) signals were all set equal to 1 as the input signal power provided adequate voltage per neuron within the network.

Comparable to the LMS filter, there is a delay line present prior to the DNN which channel

2 ($y[n]$) is sent through. The number of taps are related to the sampling rate of the signal (the audio sampling rate (f_s) is equal to 44.1kHz (*Digital Audio Basics: Audio Sample Rate and Bit Depth*, n.d.)) by equation 1:

$$Ntaps = f_s/f_{HP2} \tag{1}$$

Practically, this delay rate was too high and was causing the user interface of the DNF to glitch. After experimenting with different values, the best value found was $Ntaps = 100$. For channel 1 ($x[n]$), the signal is delayed by $Ntaps/2$ so that the DNF has time to react to the noise present in channel 2 ($y[n]$) and learn the characteristics of the signal.

The output signal from the DNN (the canceller, $z[n]$) ideally contains noise only and is subtracted from channel 1 ($x[n]$). This creates the error signal ($e[n]$) which is back-propagated in real-time through the DNN as the teaching signal for constant learning and adjustment of the neuron weights to minimise the mean-squared error signal ($e[n]$). Note that the error signal isn't equal to zero, but rather contains all of the non-correlated elements between channel 1 ($x[n]$) and channel 2 ($y[n]$) i.e. the noise-free desired signal ($d[n]$) (Widrow & Lehr, 1990).

The number of layers within the network was initially set to 6 with channel 1 ($x[n]$) gain equal to 10. This was better than LMS filtration but still wasn't an accurate representation of the channel 1 ($x[n]$) as the DNN began to interpret some of the signal as noise and subsequently cancelled out desired frequencies. By increasing the number of layers in the neuronal network, the DNF output was able accomplish a more accurate representation of the desired signal.

The number of layers within the neuronal network is equal to 20 ($L = 20$), so the number of neurons within each layer ($I(l)$) can be calculated as show in equation 2.

$$a = e^{\frac{ln \cdot Ntaps}{L-1}} \tag{2}$$

$$I(l) = \frac{Ntaps}{a^i} \tag{3}$$

The layer being output from the DNN contains only one neuron which generates the canceller signal ($z[n]$). Since Ntaps = 100, the number of neurons per layer results in: I=100,78,61,48,37, 29,23,18,14,11,8,6,5,4,3,2,2,1,1,1 neurons per layer with the first layer (containing 100 neurons) being fully attached to the delay line and subsequently funneled down in typical auto-encoder architecture so that the final layer contains only a single neuron.

As the audio input signals are AC signals, filtration of a DC free signal occurs. This, alongside randomly initialising the weights of the DNN in the range [0,1], produces unbiased weights within the DNN. Delving deeper into how weights propagate through the network layers, equation 4 shows channel 2 propagating through the first layer of the DNN.

$$b_j^0[n] = tanh\left(t_j^0[n]\right) = tanh(\sum_{k=0}^{Ntaps} w_{kj}^0 y(n-k)\right) \tag{4}$$

Where the activation function is tanh goes from linear to non-linear with growing signal strength allowing the DNN to process non-linearity's as the learning progresses. The filtered signal from the final (100^{th}) tap in the delay line for channel 2 ($y[n]$) is $y(n-k)$.

How propagation develops deeper through the network is shown in equation 5:

$$b_j^l[n] = tanh\left(t_j^l[n]\right) = tanh(\sum_{i=0}^{I(l)} w_{ij}^l b_i^{l-1}[n]\right) \tag{5}$$

The weights contained in the final output layer of the network are summed and this is equal to the canceller signal ($z[n]$) which is at the output of the DNN:

$$z[n] = tanh\left(t_0^{L-1}[n]) - tanh(\sum_{i=0}^{I(L-1)} w_i^{L-2} b_i^{L-2}[n]\right) \tag{6}$$

This canceller signal ($z[n]$) is then subtracted from channel 1 ($x[n]$) to remove the noise from the signal and this is equivalent to the output signal ($d[n]$) of the DNF which is also the error signal ($e[n]$).

$$e[n] = x[n] - z[n] \tag{7}$$

As previously explained, the error signal is equivalent to the output of the DNF, where the error in the output neuron is back-propagated through the DNN to alter the weights within the network (see equation 8). The back-propagation should minimise the overall squared error with a gradient descent (see equation 9) as positive and negative errors are equally unwanted, hence we care about the magnitude of the total error.

$$\delta_j^l = \sum_{k=1}^{K} w_{jk}^{l+1} \delta_k^{l+1} \cdot tanh'(t_j^l) \qquad where: l = L-2, ..., 0 \tag{8}$$

$$\Delta w_{ij}^l = -\mu b_i^{l-1} \cdot \delta_j^l \tag{9}$$

Δw_{ij}^l represents the change of the weights at each time step. Through experimenting with different values, the learning rate of the system (μ) was set equal to a 0.5. Due to the average amplitude of the signal in channel 1 ($x[n]$) constantly changing, adjusting the learning rate was performed manually in response to different inputs. The learning rate is set so that the DNF learns quickly as the audio sample being used is short in length (34 seconds). Learning within the system converges when noise frequencies in channel 2 ($y[n]$) are no longer detectable in the output signal of the DNF i.e.channel 1 ($x[n]$) and the error signal ($e[n]$) contain common frequencies.

The peak power of the signal occurs at around 125 seconds where the median power was calculated over the interval of samples 0 to 20000 to account for this peak alongside the lower value of noise powers found in the preceding samples.

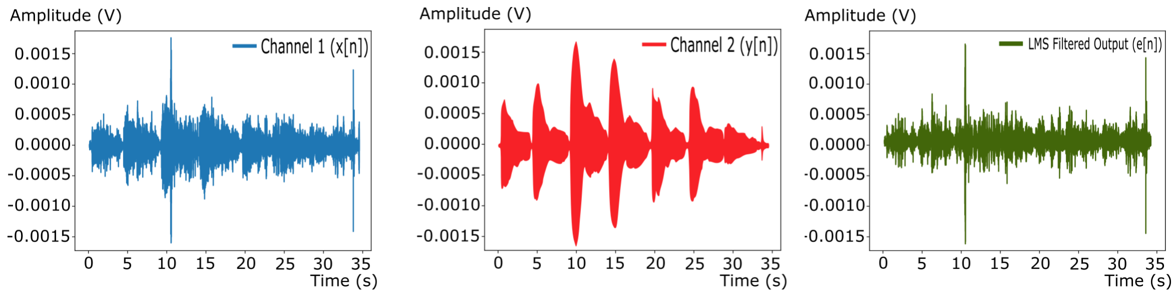
The noise generated by a bass guitar has the frequency range of 41Hz - 1000Hz (Up to 4000Hz including overtones). So, the noise power calculated is the sum of the power density spectrum between these frequencies using the Welch method (Barbe, Pintelon, & Schoukens, 2010).

Finally, the SNR's were calculated as in equation 10 for the unfiltered signal channel 1 ($x[n]$), LMS filtration and the DNF method. Where the different signals can be input to the equation as variable P.

$$SNR = \frac{median(P^2)}{\sum_{k=41Hz}^{1000Hz} Welch(P)[k]} \tag{10}$$

3 Results

3.1 LMS Results



Channel 1 (x[n]) Channel 2 (y[n]) Output Signal (e[n] = d[n])

Figure 6: LMS Filter Results

Figure 6 shows channel 1 ($x[n]$), channel 2 ($y[n]$) and the LMS filter output ($e[n]$), with the back-propagation of the error signal minimising the bass frequencies in the output signal ($e[n] = d[n]$) in comparison with the raw input signal ($x[n]$). Before filtering, the amplitude of the bass frequencies were equal to around $0.0005V$ at a maximum as shown in channel 1 ($x[n]$). After the signal is filtered, the bass peaks are visually indistinguishable from the rest of the signal. Upon listening to the filtered LMS output, there are slight murmurs of the bass audibly present - available to listen to at: (Atkinson, 2022), but overall the SNR has been improved.

Note the amplitude of the LMS output signal ($d[n] = e[n]$) is slightly reduced upon comparison with channel 1 ($x[n]$) - from a maximum of just above $0.0015V$ to a maximum of just under $0.0015V$. This is to be expected as compared with previous literature due to the weak commonality correlation between channel 1 ($x[n]$) and channel 2 ($y[n]$) (Zhang & Wang, 2021). For practical applications, the output signal of the filtered LMS could be multiplied by a constant gain to boost the overall signal amplitude. In the frequency domain from the filtered signal, the reduction in the bass noise frequencies (41Hz to 1000Hz) can be seen more clearly.

3.2 DNF Results

Figure 8 shows the user interface window of the DNF performing real-time filtration over the first 5680 samples of data for 'Recording 5'. The left column displays the data from the DNF and the right column shows data for an LMS filter for comparison.

Looking at the leftmost column, the first row indicates channel 2 ($y[n]$) containing the noise and the second row shows channel 1 ($x[n]$) containing the raw signal. Channel 2 is sent through the DNN, hence producing the internal canceller signal ($z[n]$) in row three which learns the characteristics of the noise, to ideally match the waveform shown in channel 2 ($y[n]$). Finally, the bottom row shows the output from the DNF which is equal to the error signal ($e[n]$) being back-propagated through the DNN.

As discussed in section 2.2, the initial number of layers was set to 6, with channel 1 requiring a gain of 10 for the DNF to cancel out noise. In the figure 9 below, we see that using 6 layers within the DNF doesn't produce accurate results for audio signal noise cancellation, as the canceller signal ($z[n]$) in row 3 fails to match channel 2 ($y[n]$) in row one. This demonstrates the

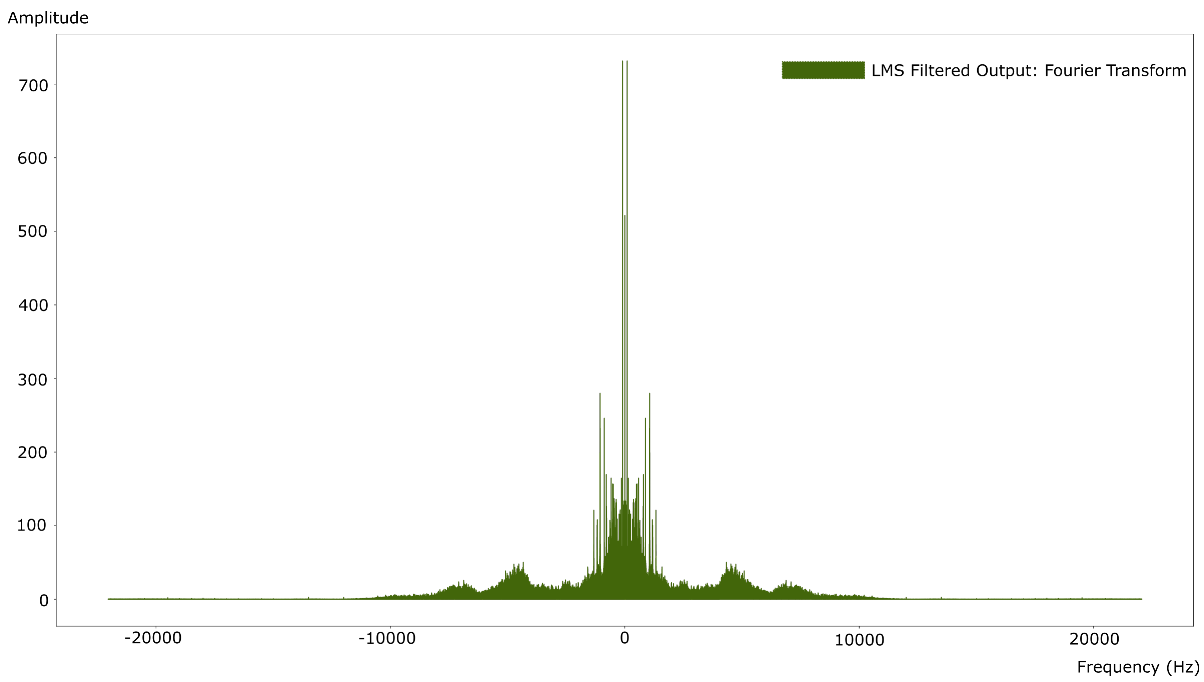


Figure 7: LMS Filter : Fast Fourier Transform of Filtered Output

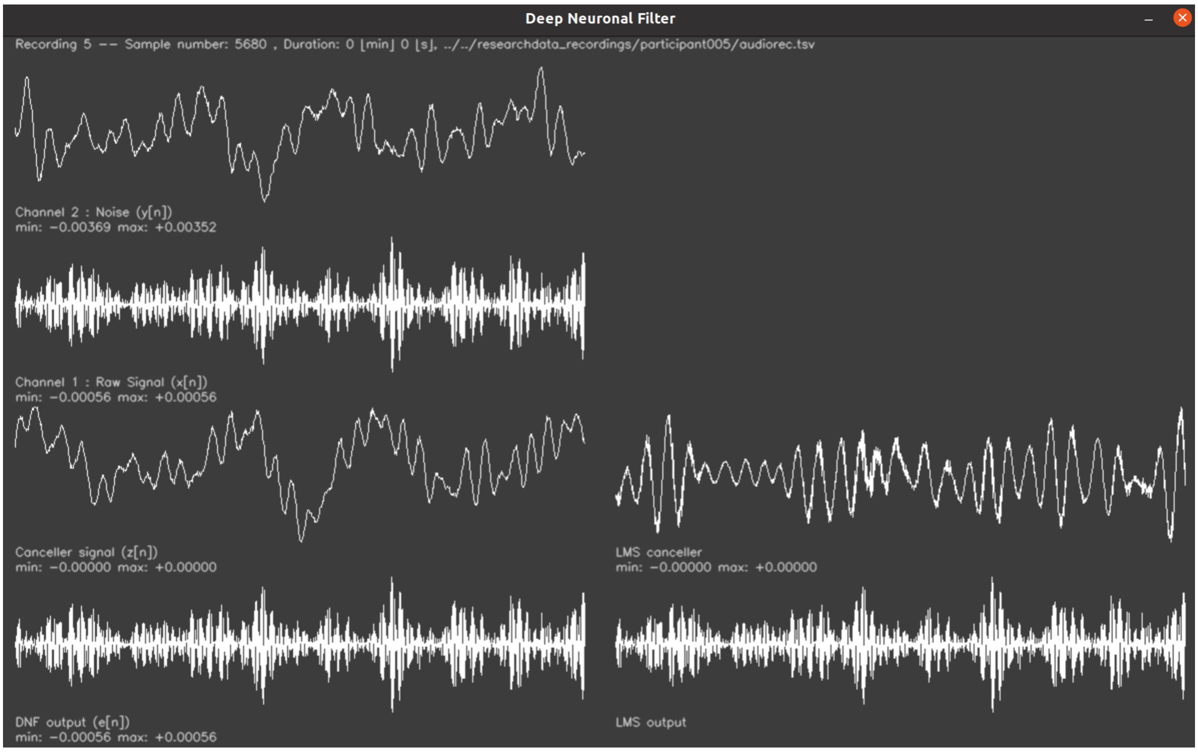


Figure 8: Deep Neuronal Filter User Interface

reasoning behind the increase of layers within the neural network. The DNF was then run with 20 layers for approximately 3 minutes (175 seconds) to show it working over a longer period of time. Note the subsequent results refer within this time-frame.

The canceller signal within the DNF deals with non-linearity's better than the LMS, which can be observed by comparing the canceller of the DNF with the LMS canceller in figure 10.

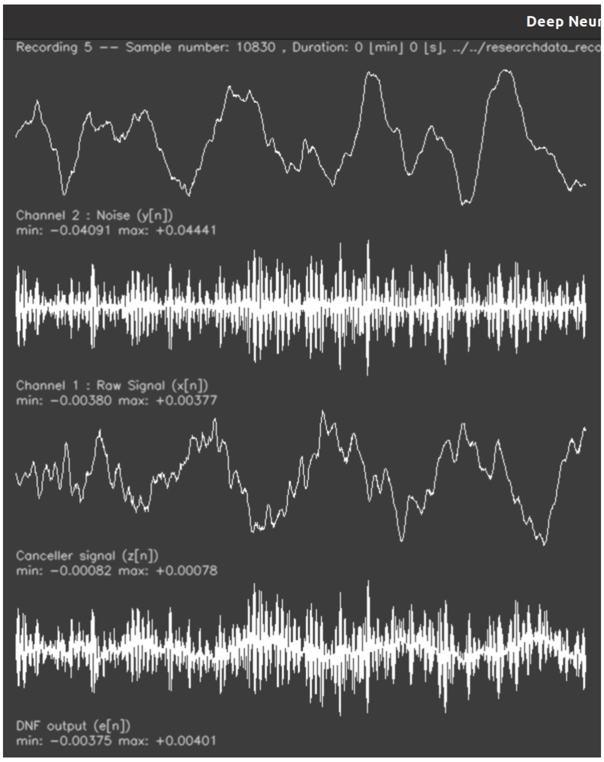
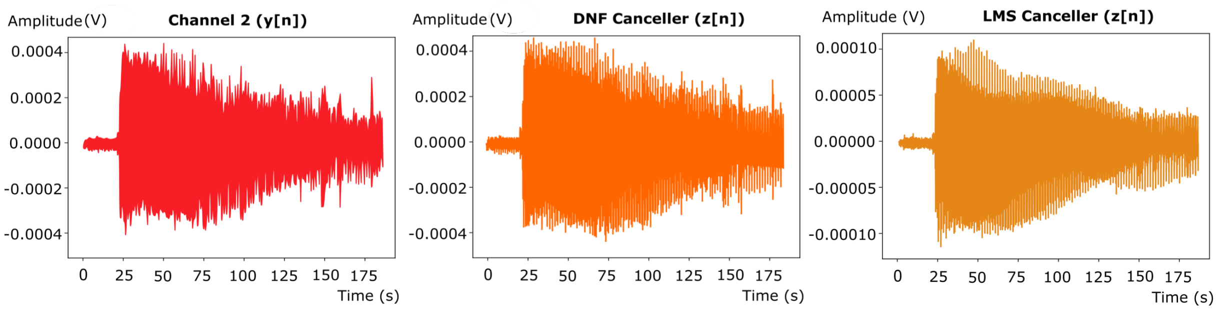


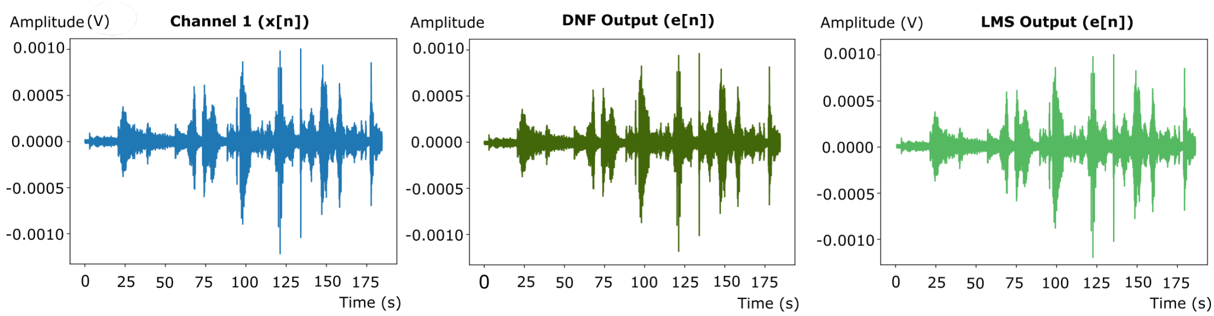
Figure 9: Deep Neuronal Filter User Interface (6 Layers)



Channel 2 y[n] DNF Canceller z[n] LMS Canceller z[n]

Figure 10: DNF Filter Results (1)

The LMS canceller generalises the shape of the waveform in channel 2 ($y[n]$), whilst the DNF canceller waveform matches reasonably accurately with only slight dissimilarities. Note the amplitude of the LMS waveform is reduced ($0.0004V$ to $0.0001V$)in comparison to channel 2 and the DNF canceller, this is due to the LMS filter being only capable of reducing the noise within the signal, failing to remove it completely. Through comparing Channel 1 with the DNF output in figure 11, the signal amplitude of the DNF is reduced slightly. This is due to the cross-talk occurring between the two channels as described in section 2. So, even further reduction of noise in the DNF will be required to improve the SNR in comparison to the LMS filter. Due to the closed loop corrective action (Daryanavard & Porr, 2020) of the DNN, the network will adjust the weights accordingly to minimise smaller dis-correlations, but cross-talk will still be a factor in reducing the DNF output amplitude. The weight development of the 20 layers within the neuronal network is shown in figure 12 where the weight distance is plotted from the initial random values for approximately 3 minutes (175 seconds).



Channel 1 x[n]   DNF Output e[n]   LMS Output e[n]

Figure 11: DNF Filter Results (2)

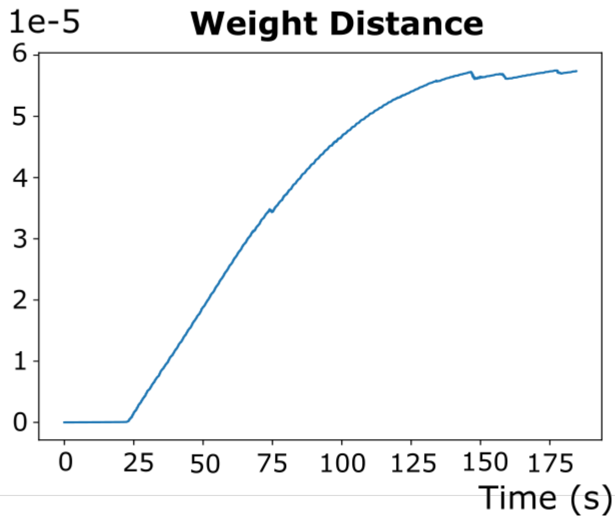


Figure 12: Weight Distance

Note that the learning in the initial 25 seconds is stationary due to the time delay at the input of the DNN. The weight values change most rapidly between 25 and 75 seconds as the DNF learns the characteristics of the signal. The DNF takes approximately 150 seconds before it stabilises and has learned the characteristics of the noise reference signal, with the weights adjusting throughout in response to the ever-changing audio signal, with smaller adjustments required after stabilisation occurs.

From the graph in figure 12, we can see that the weight distance from the initialised random weights jumps up at certain sections. This represents a large alteration in the noise generated, for instance when the bass string is being plucked. Figure 13 shows the power spectral density of Channel 1 ($x[n]$) compared with the LMS and DNF outputs ($d[n] = e[n]$). The LMS is effective in reducing the noise power slightly (from $10^{-}6$ to $10^{-}12$ at 41 Hz) but due to its linear nature, it reduces the spectral components without minimising the peaks within the signal, hence failing to deal with inconsistencies in the signal.

The DNF filter has a significant reduction in the noise power compared to the LMS (from $10^{-}12$ to just above $10^{-}15$ at 41 Hz) and Channel 1 (from $10^{-}6$ to just above $10^{-}15$ at 41 Hz). It is successful in reducing some frequencies more than others, with struggles particularly in the lower frequency range (0 Hz - 2000 Hz) where most of the bass noise frequencies occur. This could be due to the amplitude of the DNF output being reduced because of cross-talk (with

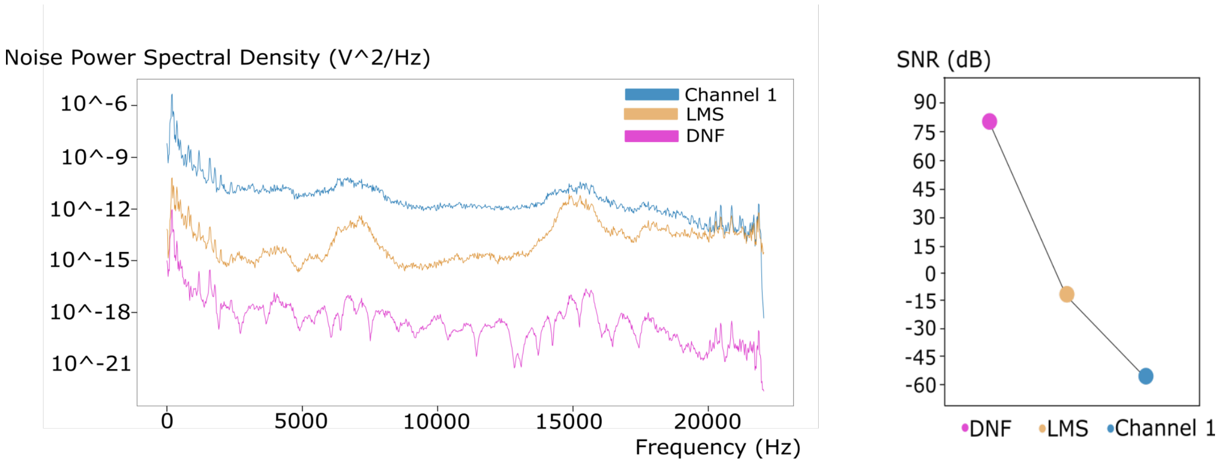


Figure 13: Noise Power Analysis

noise amplitude remaining the same), making it more difficult to improve the SNR within this frequency range.

The SNR for unfiltered Channel 1, LMS filtering and DNF filtering are shown in figure 13. Both the LMS and the DNF have improved on the original SNR of Channel 1 which sits at a relatively low value of -56.36dB. The SNR of the LMS is equal to -14.15dB and the DNF SNR is equal to 82.82dB. This demonstrates the LMS filter's improvement on the unfiltered SNR by 42.21dB and the DNF improving on the unfiltered signal by a high value of 135.18dB. The DNF improved on the LMS filtering by 96.97dB showing its effectiveness at dealing with the non-linearity's that the LMS struggles to cope with. A SNR of 80dB or greater is considered an ideal signal strength in audio signals, so the achieved SNR value of the DNF demonstrates its usefulness for noise cancellation in audio practices. The improvement of noise reduction can be listened to at: (Atkinson, 2022).

4 Discussion

LMS algorithms are currently used in regards to noise cancellation for audio and have been explored to work effectively in multiple other applications such as filtration of EEG signals (Porr et al., 2020) and tracking mobile communications (Lindbom, Sternad, & Ahlen, 2001). For audio applications, there have been various approaches to the use of LMS adaptive algorithms where multiple sensors have been used (Zangi, 1993), or LMS filtration in conjunction with micro-controllers (Chang & Li, 2011). However, due to the linear nature of the LMS filter, all of these efforts fail to reduce noise when inconsistencies occur within the signals. The deep neuronal network consisting of non-linear activation functions is crucial in dealing with these inconsistencies as explored in 3D Style Reconstruction (Friedrich, Wollstadt, & Menzel, 2020) and provide significant improvement for the removal of noise within this audio experiment (SNR increase when DNF compared to LMS of 96.97dB). The use of non-linear weight generation techniques have also been explored when modelling intelligent pressure sensors (Patra & van den Bos, 2000), but are notably more computer intensive than the use of the standardised DNN with additional components as demonstrated in this paper.

Using the noise as the reference signal for the DNN is efficient as this is readily available and can be obtained by placing a microphone on the outside of the headphones, as is standard practice in noise cancellation techniques used currently (Liang & Hu, 2016). It is unrealistic to train the DNN using a clean desired signal as all audio is effected in some way by noise; be it electrical noise generated by wires/microphones external environmental noise.

LMS filters have been proven to cancel noise mainly in the lower frequency region (Flotte-Hernandez et al., 2008), but use of the DNF cancels noise across the whole range of frequencies a human ear can perceive as demonstrated in figure 13. By using a DNF, this improved the SNR when compared to the LMS filter, alongside non-linearity's being omitted to largely create a noise-free audio signal.

It was assumed that the signal stayed constant throughout filtering so that maximum correlation between the noise in channel 1 ($x[n]$) and channel 2 ($y[n]$) occurs. As previously mentioned in section 2, this would not be the case, so this high improvement of SNR would be expected to be reduced slightly in real-world applications where the input signal is in-homogeneous. Multiplying the output of the DNF by a constant gain has potential to increase the SNR further, however additional amplifiers in the DNF design will add some electrical noise. Use of low noise signal amplifiers could be considered (Sathya, 2021), however the benefits of this signal amplification will need to outweigh the drawback of more electrical noise generated to further improve the SNR.

Increasing the number of layers within the neural network has been demonstrated to improve the accuracy at which the noise is cancelled out. By increasing the layers in the neural network from 6 to 20, this showed an improvement in the signal matching between the canceller and reference noise in channel 2. By further increasing the number of layers, it is expected that this would improve the SNR to a greater extent. However, since the number of neurons within the hidden layers are related to the time taken for the neural network to train, this increase in layers could cause a slowing down of the learning rate. Consequences of which include the amount of training time to become too large so the network will not learn which frequencies to cancel out within that time, resulting in in-adequate noise cancellation. The optimum value found from experimentation between the learning rate and the number of neurons in the DNN was 20 layers for this application and size of noise reference, though the optimal number of layers varies depending on the application.

A large training set of input signals has been proven to enhance the abilities of the DNN with noise cancellation in speech (Xu, Du, Dai, & Lee, 2015), the benefits of which are numerous for audio applications, as it allows the DNF to combat more complex noise present in real-world scenarios. The DNF currently improved the SNR by cancelling out noise in the bass frequencies as directed by the high-pass filtration of the noise reference to direct the learning towards the certain frequencies regarded as noise. Theoretically, it is assumed the recording conditions for the input signals are perfect and unchanging throughout processing as discussed previously. Practically, electrical recording equipment creates its own non-uniform noise and so the noise reference changes slightly over time (Ardila-Rey et al., 2021). In real-world situations, audio cancellation requires noise mitigation across a larger and unpredictable bandwidth of frequencies, suggesting that multiple training signals would be beneficial to the DNF for dealing with more complex signals. The DNF would also be required to learn faster than the rate at which the noise reference is changing for effective noise cancellation. Increasing or decreasing the learning rate of the system past a certain point relative to the desired signal input causes distortions within the output signal (Randelhoff, Shearer, & Levy, 1990), so a critical value will be present for optimum noise cancellation in real-world applications.

The distance between the microphones is fairly small (0.15m), leading to a strong correlation of noise between the channels, but this leads to cross-talk between the signals as mentioned previously. This cross-talk has been mitigated using filters, as well as choosing directional microphones to improve the input signal quality. For noise-cancelling headphones, multiple microphone types such as optical (Kida, Hirayama, Kajikawa, Tani, & Kurumi, 2009) and virtual sensing microphones (Cazzolato, 2002) have been implemented in precursory designs. In the real-world, distances between the noise sources will be constantly altering, hence affecting the DNF's ability to process the location of the sound, with cross-talk potentially becoming a common occurrence within noise signals present in the environment. Noise will also be of altering amplitudes as the source distance differs for multiple inputs. This will affect the SNR rate of the DNF as it will need to deal with multiple inputs of noise reference where increasing distance between microphones could cause a weaker correlation of noise resulting in less effective noise cancellation in real-world applications. So, the assumption of idealistic conditions for obtaining the noise reference are not possible for real-world applications.

5 Conclusions

This paper demonstrates the increased effectiveness of implementing a DNF for the filtration of noise from audio signals in real-time in comparison to preceding filtration techniques such as the LMS adaptive algorithm. Where other adaptive algorithms fall short, the DNF implemented deals efficiently with inconsistencies due to the multi-layered nature of the network, with an increase in the amount of layers showing improvement in the accuracy of the DNF output as described in section 2.2. There is a successful reduction of noise in a wider frequency range when compared to the LMS algorithm, and the SNR improved by 96.97dB when compared to the LMS algorithm and by 135.18dB when compared to an unfiltered signal.

The architecture, through its use of a standard deep network and easily accessible components, is cost-effective and not as computer intensive as other proposed solutions (Castillo & Melin, 2004). The DNF is successful at removing the noise within a short period of learning, and so can be applied to real-world situations for removal of noise in applications such as noise cancelling headphones. As the noise reference signal is generated, knowing the frequency spectrum of this for additional filtration through the high-pass filters was an important factor in increasing the SNR as this directed the DNF's learning to a specific frequency bandwidth. When the noise signal is less predictable than a bass guitar, as is the case in most real-world examples where multiple types of noise are present (Xu et al., 2015), we are unable to filter within certain frequency bandwidths, which could affect the DNF's ability to learn which signals to cancel out.

A solution to this could be to pre-train the neural network with different signals representing typical noises found in the real-world; a technique currently being investigated (Lu et al., 2021). This would allow the DNF to predetermine dissimilar characteristics between the noise and desired signals, However, this presents its own challenges as assumptions are made of what sounds are considered 'desired' signals, for example in ambient music a lot of typical 'noise' sounds are part of the composition, i.e. the desired signal.

5.1 Suggestions for further work

To further this project, an offline DNF could be created for use in noise cancelling headphones. Current technology for noise-cancelling headphones uses adaptive algorithms with 2 microphones placed on the headset (Priese, Bruhnken, Voss, Peissig, & Reithmeier, 2012) - one taking in the reference noise, and the other producing the desired signal to the user (which also acts as the error signal as discussed previously). Instead of using LMS adaptive algorithms as is currently available on the market, the DNF with multiple layers could be implemented for an improvement in the SNR, resulting in more effective noise cancellation.

Noise-cancelling headphones using the DNF will require an analogue to digital converter (ADC) as is used in current designs, alongside other standard components such as an amplifier at the output of the DNF before the signal reaches the users ear, to increase the amplitude of the signal. The architecture will be similar to noise-cancellation headsets that use adaptive algorithms, but instead of the single-layered LMS, the multi-layered DNN will be implemented. This may require more processing power than current headsets, but due to the standard deep net and simple components used in my experiment (as discussed in section 4), this shouldn't be detrimental to the design.

Researching into the relationship between the learning rate of the neural network, and the amount of layers within to optimise the speed at which the network can learn characteristics of a signal, would further DNF's audio cancellation abilities in real-world applications. By increasing the speed at which the DNF learns, whilst including a high number of neuron layers within the network, would allow accurate and fast processing of constantly-changing noise references (as is realistic in real-world applications) to further improve the SNR of the DNF compared to its current abilities.

References

(2022).

Ardila-Rey, J. A., Schurch, R., Poblete, N. M., Govindarajan, S., Muñoz, O., & de Castro, B. A. (2021). Separation of partial discharges sources and noise based on the temporal and spectral response of the signals. *IEEE Transactions on Instrumentation and Measurement*, *70*, 1-13. doi: 10.1109/TIM.2021.3121488

The artists of the animal kingdom. (2021). Retrieved from https://play.acast.com/s/bbcearthpodcast/theartistsoftheanimalkingdom

Atkinson, E. (2022). *Github - elise-atki/deepneuronalfilter: Deep neuronal filter (dnf): A closed-loop filter to remove noise from signals with the help of a noise reference signal.* Retrieved from https://github.com/elise-atki

Barbe, K., Pintelon, R., & Schoukens, J. (2010). Welch method revisited: Nonparametric power spectrum estimation via circular overlap. *IEEE Transactions on Signal Processing*, *58*(2), 553-565. doi: 10.1109/TSP.2009.2031724

Braun, S., Gamper, H., Reddy, C. K., & Tashev, I. (2021). Towards efficient models for real-time deep noise suppression. In *Icassp 2021 - 2021 ieee international conference on acoustics, speech and signal processing (icassp)* (p. 656-660). doi: 10.1109/ICASSP39728.2021.9413580

Castillo, O., & Melin, P. (2004). Adaptive noise cancellation using type-2 fuzzy logic and neural networks. In *2004 ieee international conference on fuzzy systems (ieee cat. no.04ch37542)* (Vol. 2, p. 1093-1098 vol.2). doi: 10.1109/FUZZY.2004.1375563

Cazzolato, B. (2002). An adaptive lms virtual microphone. In *In proceedings of active 02* (pp. 105–116).

Chang, C.-Y., & Li, S.-T. (2011). Active noise control in headsets by using a low-cost microcontroller. *IEEE Transactions on Industrial Electronics*, *58*(5), 1936-1942. doi: 10.1109/TIE.2010.2058071

Cichocki, A., Vorobyov, S., & Rutkowski, T. (2000, 08). Nonlinear interference cancellation using neural networks.

Daryanavard, S., & Porr, B. (2020, 11). Closed-Loop Deep Learning: Generating Forward Models With Backpropagation. *Neural Computation*, *32*(11), 2122-2144. Retrieved from https://doi.org/10.1162/neco_a_01317 doi: 10.1162/neco_a_01317

Daryanavard, S., Porr, B., & Dahiya, R. (2022, Mar). Deep neuronal filter: Eeg filter release. doi: 10.5281/zenodo.6360675

Digital audio basics: Audio sample rate and bit depth. (n.d.).

Douglas, S. (1997). An efficient implementation of the modified filtered-x lms algorithm. *IEEE Signal Processing Letters*, *4*(10), 286-288. doi: 10.1109/97.633770

Elhossini, A., Areibi, S., & Dony, R. (2006). An fpga implementation of the lms adaptive filter for audio processing. In *2006 ieee international conference on reconfigurable computing and fpga's (reconfig 2006)* (p. 1-8). doi: 10.1109/RECONF.2006.307767

Flotte-Hernandez, O., Pineda-Olivares, A., Dieck-Assad, G., Avila, A., Martinez-Chapa, S. O., & Bouchereau-Lara, F. (2008, 11). On the performance of active noise control fx-lms and fbfx-lms algorithms for duct network noise attenuation. In (p. 575 - 581). doi: 10.1109/CERMA.2008.30

Friedrich, T., Wollstadt, P., & Menzel, S. (2020). The effects of non-linear operators in voxel-based deep neural networks for 3d style reconstruction. In *2020 ieee symposium series on computational intelligence (ssci)* (p. 1460-1468). doi: 10.1109/SSCI47803.2020.9308308

Ghasemi, S., Kamil, R., & Marhaban, M. H. (2016, mar). Nonlinear thf-fxlms algorithm for active noise control with loudspeaker nonlinearity. *Asian J. Control*, *18*(2), 502–513. Retrieved from https://doi.org/10.1002/asjc.1140 doi: 10.1002/asjc.1140

Greenberg, J. (1998). Modified lms algorithms for speech processing with an adaptive noise canceller. *IEEE Transactions on Speech and Audio Processing*, *6*(4), 338-351. doi: 10.1109/89.701363

H4nsp handy recorder. (n.d.). Retrieved from https://www.zoom.co.jp/products/handy-recorder/h4nsp-handy-recorder

Ikuta, N., Iwanaga, R., Tokunaga, A., Nakane, H., Tanaka, K., & Tanaka, G. (2016, Dec). Effectiveness of earmuffs and noise-cancelling headphones for coping with hyper-reactivity to auditory stimuli in children with autism spectrum disorder: A preliminary study. *Hong Kong journal of occupational therapy : HKJOT*.

Jafarifarmand, A., & Badamchizadeh, M. A. (2013, mar). Artifacts removal in eeg signal using a new neural network enhanced adaptive filter. , *103*, 222–231. Retrieved from https://doi.org/10.1016/j.neucom.2012.09.024 doi: 10.1016/j.neucom.2012.09.024

Kida, M., Hirayama, R., Kajikawa, Y., Tani, T., & Kurumi, Y. (2009, 05). Head-mounted active noise control system for mr noise. In (p. 245 - 248). doi: 10.1109/ICASSP.2009.4959566

Kuo, S., & Morgan, D. (1999). Active noise control: a tutorial review. *Proceedings of the IEEE*, *87*(6), 943-973. doi: 10.1109/5.763310

Kuo, S. M., Chen, Y.-R., Chang, C.-Y., & Lai, C.-W. (2018). Development and evaluation of light-weight active noise cancellation earphones. *Applied Sciences*, *8*(7). Retrieved from https://www.mdpi.com/2076-3417/8/7/1178 doi: 10.3390/app8071178

Lee, S.-C., Chen, C.-W., Su, B.-H., Chen, Y.-S., Tseng, S.-P., & Wang, J.-F. (2018). Dnn regression model and microphone array for noise reduction system on real field. In *2018 international conference on orange technologies (icot)* (p. 1-4). doi: 10.1109/ICOT.2018.8705927

Levitt, H. (2001, 01). Noise reduction in hearing aids: A review. *Journal of rehabilitation research and development*, *38*, 111-21.

Liang, K.-W., & Hu, J.-S. (2016). An anc system with dual feedback control. In *2016 international automatic control conference (cacs)* (p. 95-99). doi: 10.1109/CACS.2016.7973890

Lindbom, L., Sternad, M., & Ahlen, A. (2001). Tracking of time-varying mobile radio channels .1. the wiener lms algorithm. *IEEE Transactions on Communications*, *49*(12), 2207-2217. doi: 10.1109/26.974267

Liu, Y., Ma, X., Shu, L., Yang, Q., Zhang, Y., Huo, Z., & Zhou, Z. (2020). Internet of things for noise mapping in smart cities: State of the art and future directions. *IEEE Network*, *34*(4), 112-118. doi: 10.1109/MNET.011.1900634

Lu, L., Yin, K.-L., de Lamare, R. C., Zheng, Z., Yu, Y., Yang, X., & Chen, B. (2021). *Active noise control techniques for nonlinear systems.* arXiv. Retrieved from https://arxiv.org/abs/2110.09672 doi: 10.48550/ARXIV.2110.09672

Milestone series peavy bass. (n.d.). Retrieved from https://peavey.com/Milestone-Series

Patra, J. C., & van den Bos, A. (2000). Modeling of an intelligent pressure sensor using functional link artificial neural networks. *ISA Transactions*, *39*(1), 15-27. Retrieved from https://www.sciencedirect.com/science/article/pii/S001905789900035X doi: https://doi.org/10.1016/S0019-0578(99)00035-X

Pixel3a. (2019). Retrieved from https://www.gsmarena.com/google_pixel_3a

-9408.php

Porr, B. (n.d.). *Adaptive fir least mean squares (lms) filters.* Retrieved from https://berndporr.github.io/digital_signal_processing/node66.html

Porr, B., Daryanavard, S., Bohollo, L. M., Cowan, H., Porr, B., & Dahiya, R. (2020). *Real-time noise cancellation with deep learning.* arXiv. Retrieved from https://arxiv.org/abs/2011.03466v3 doi: 10.48550/ARXIV.2011.03466

Priese, S., Bruhnken, C., Voss, D., Peissig, J., & Reithmeier, E. (2012, 09). Adaptive feedforward control for active noise cancellation in-ear headphones. *The Journal of the Acoustical Society of America*, *132*, 2014. doi: 10.1121/1.4755454

Randelhoff, M., Shearer, E., & Levy, D. (1990). Two microphone noise reduction for speech using vlsi lms adaptive filters. In *Ieee south african symposium on communications and signal processing* (p. 137-141). doi: 10.1109/COMSIG.1990.658543

Sathya, P. (2021). Design of a low noise signal amplifier for sensor signal amplification. In *2021 innovations in power and advanced computing technologies (i-pact)* (p. 1-6). doi: 10.1109/i-PACT52855.2021.9696986

Shalool, A., Zainal, N., Beng Gan, K., & Umat, C. (2016). An investigation of passive and active noise reduction using commercial and standard tdh-49 headphones. In *2016 international conference on advances in electrical, electronic and systems engineering (icaees)* (p. 606-609). doi: 10.1109/ICAEES.2016.7888118

Vesely, S. L., Dolci, C. A., & Dolci, S. R. (2018). The doppler effect in the perception of sound. *AIP Conference Proceedings*, *1990*(1), 020027. Retrieved from https://aip.scitation.org/doi/abs/10.1063/1.5047781 doi: 10.1063/1.5047781

Widrow, B., Glover, J., McCool, J., Kaunitz, J., Williams, C., Hearn, R., ... Goodlin, R. (1976, 01). Adaptive noise cancelling: Principles and applications. *Proceedings of the IEEE*, *63*, 1692 - 1716. doi: 10.1109/PROC.1975.10036

Widrow, B., & Lehr, M. A. (1990). 30 years of adaptive neural networks: perceptron, madaline, and backpropagation..

Xu, Y., Du, J., Dai, L.-R., & Lee, C.-H. (2015, jan). A regression approach to speech enhancement based on deep neural networks. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, *23*(1), 7–19. Retrieved from https://doi.org/10.1109/TASLP.2014.2364452 doi: 10.1109/TASLP.2014.2364452

Zangi, K. (1993). A new two-sensor active noise cancellation algorithm. In *1993 ieee international conference on acoustics, speech, and signal processing* (Vol. 2, p. 351-354 vol.2). doi: 10.1109/ICASSP.1993.319310

Zhang, H., & Wang, D. (2021). Deep anc: A deep learning approach to active noise control. *Neural Networks*, *141*, 1-10. Retrieved from https://www.sciencedirect.com/science/article/pii/S0893608021001258 doi: https://doi.org/10.1016/j.neunet.2021.03.037

6 Appendices

- Code for the Deep neuronal filter used within this project available at : *https://github.com/elise-atki*
- Interim report available at : *https://drive.google.com/file/d/1sabmncLyrilAOag63mRqsZ2YJhinC* FYY/view?usp=sharing
- Preliminary Report including Project Plan available at : *https://drive.google.com/file/d/1pkxAD-byhRcQ3LYiepbXAWrGZH3WBu0/view?usp=sharing*