Intelligent Background Sound Event Detection and Classification based on WOLA Spectral Analysis in Hearing Devices

Feifan Lai

A thesis
submitted in partial fulfillment of the
requirements for the degree of

Master of Science in Electrical Engineering

University of Washington

2019

Committee:
Kaibao Nie
Tadesse Ghirmai
Rania Hussein

Program Authorized to Offer Degree:
Electrical Engineering

University of Washington

Abstract

Intelligent Background Sound Event Detection and Classification based on WOLA Spectral Analysis in Hearing Devices

Feifan Lai

Chair of the Supervisory Committee:
Kaibao Nie
Electrical Engineering

Audio signals from real-life hearing devices typically contain background noises. The purpose of this thesis is to build a system model which can automatically separate background noise from noisy speech, and then classify background sound into predefined event categories. This thesis proposed to use weighted overlap-add algorithm (WOLA) for feature extraction and neural network (NN) for sound event detection. In this approach, an energy waveform 'trough detection' algorithm is used to separate out speech gaps which primarily contain background noise. To further analyze the noise signal's spectrum, the WOLA algorithm is used to extract spectral features by transforming a fraction of time domain signal into frequency domain data represented in 22 channels. Moreover, a feed-forward neural network with one hidden layer is used to recognize each event's diverse spectral feature pattern. Then it produces classification decisions based on confidence values. Recordings of 11 realistic background noise scenes, mixed with human speech at Signal to Noise Ratio (SNR) of 5 dB, are used for training. NN will learn the mapping between spectral feature characteristics and sound events categories. After training, the neural network classifier is evaluated by measuring the accuracy of event classification. The

overall detection accuracy has achieved 96%, while the event ‘hallway’ has the lowest detection rate at 85%. This detection algorithm has the ability to improve noise reduction in hearing devices by applying distinct compensation gains, which attenuate the noise dominated frequency bands for each particular predefined event. In our preliminary evaluation experiment, the application of gain patterns has been proven to be effective in reducing background noise. Its combinational usage with instant gain pattern would produce improved results with noticeably attenuated noise and smooth spectral cues in the processed audio output.

TABLE OF CONTENTS

Page

Chapter 1: Introduction 1

1.1 Background 1

1.1.1 General Overview and Motivation 1

1.1.2 Noise Reduction in Hearing Devices 2

1.1.3 WOLA in DSP 5

1.1.4 Intelligent System in DSP 7

1.1.5 Signal Processing in Cochlear Implants 10

1.2 Overview of Thesis 11

Chapter 2: Environmental Background Noise Extraction 13

2.1 Audio Preparation 13

2.2 Trough Detection Algorithm 16

Chapter 3: Feature Extraction 21

3.1 WOLA Spectral Analysis Introduction 21

3.2 Features for Classification 24

Chapter 4: Event Detection 30

4.1 Introduction to Feedforward Neural Network Training 30

4.2 Detection Results and Discussion 32

Chapter 5: Noise Reduction with Gain Patterns 38

5.1 Event Gain Pattern 38

5.2 Combined Usage of Event Gain Pattern with Instant Trough Gain Pattern 41

Chapter 6: Conclusions and Future Work 47

Bibliography 50

LIST OF FIGURES

Figure Number Page

1.1 Simplified flowchart of the working principle of spectral subtraction techniques …….. 3

1.2 A simple visual realization of neural network design with one hidden layer, 6 inputs and 2 outputs. ……. 8

1.3 A simple diagram to demonstrate the operations inside a neuron……... 8

1.4 Diagram of working principle of ACE. …….. 10

1.5 Diagram of working principle of CIS. …….. 11

1.6 Framework of the main signal processing procedures in this thesis…….. 12

2.1 Signal (a) is the event 'cafe' background noise. Signal (b) is the human speech audio signal. Signal (c) is the addition combination of the speech and noise at SNR = +5 dB. The arrow on signal(c) points to an ideal noise extraction segment. ...…….…….... 15

2.2 The flow chart of trough detection procedure. …… 16

2.3 Signal (a) is the original mixed audio signal. Signal (b) is the energy waveform derived from the audio signal. ……. 17

2.4 Demonstration of the decision rules of the trough detection algorithm. ……. 18

2.5 Demonstration of the application of the trough detection algorithm. The energy waveform is derived from a mixed audio signal with event 'station' at SNR = +5 dB. The x axis is the number of energy waveform samples. The green line is the trace of RC response line. The red dots are troughs detected, and they represent the background noise-only samples to be extracted. ……..... 19

2.6 Comparison of the mixed audio signal to the background noise only signal and extracted troughs samples. The noise event 'cafe' is used, with SNR = +5. It shows that the trough signal is a good representation of the background noise signal. Thus, it shows that the trough detection algorithm is working properly as designed to extract the background sound clips from the mixed audio signal. ……..... 20

3.1 Block diagram of WOLA filterbank analysis procedure. ……...……. 23

3.2 Block diagram of WOLA filterbank synthesis procedure. ……. 23

3.3 Workflow of how to obtain feature curve. ……..... 24

3.4 Block diagram to show how one energy trough is transformed into a feature curve through WOLA analysis. ……...…….. 26

3.5 A direct comparison of all 11 events' averaged feature curves, including events 'cafe', 'cafeteria', 'hallway', 'kitchen', 'living', 'meeting', 'office', 'resto', 'square', 'station', 'washing'. The horizontal axis stands for frequency channels, and the vertical axis is magnitude in dB. ……. 29

4.1 The neural network layer structure. ……...……. 30

4.2 Demonstration of the neural network structure used. ……. 31

4.3 Neural network detection process and output confidence levels for event 'station'..... 32

4.4 Plot (a) is the feature curve of 'hallway'. Plot (b) is the feature curve of 'meeting'. Two feature curves are found to be very similar to each other. …….……. 36

4.5 Plot (a) is the overlay of 100 feature curves of event 'hallway'. Plot (b) is the overlay of 100 feature curves of event 'kitchen'. The purpose is to compare the variance of the feature curves from two events. It's shown that 'hallway' feature curves have larger variance. ……...……. 37

5.1 Process of obtaining event gain curve to attenuate background noise. ……. 38

5.2 Plot (a) is the 'cafe' feature curve. Plot (b) is the cafe gain curve used for de-noising, which is derived from plot (a)'s feature curve. The gain curve is the flip-over of feature curve, with additional processes of normalization and Pre-emphasis. ……...…….. 39

5.3 Spectrograms of a piece of audio combining human speech and background 'cafe' noise with SNR = 0. Image (a) is the input audio spectrogram of the unprocessed sound. Image (b) is output audio after applying the de-noising gain curve. Image (b) shows a clearer image with less noise while the speech details are preserved in the low and mid frequency range. …… 40

5.4 Workflow of how the instant trough gain pattern is derived. ……. 41

5.5 A plot showing how the last four trough samples are selected to generate the instant gain curve. ……..…… 42

5.6 Spectrograms of a piece of audio combining human speech and background 'cafe'

noise with SNR = 0 dB. Image (a) is the input audio spectrogram without denoising modification, and image (b) is output audio after applying the instant gain curve. Image (b) shows a clearer image with less noise(more blue areas) while the speech details are reserved. But there are gaps in the speech signal in low and mid-range frequency. ……. 43

5.7 Spectrograms of a piece of audio combining human speech and background 'cafe' noise with SNR = 0. Image (a) is the input audio spectrogram without denoising modification, and image (b) is output audio after applying the de-noise instant gain curve. Image (b) shows that the usage of combined gain pattern has obtained improved noise reduction and speech smoothness. ……...…….. 45

5.8 These three images are zoom-in comparison between three types of gain patterns applied. Image (a) is the spectrogram result of using the averaged gain pattern. Image (b) is the spectrogram result of using the instant gain pattern. Image (c) is the spectrogram result of using the combined gain pattern. Image (a) preserves and enhances all the details in human speech. But it has horizontal spectral gaps, since its gain pattern remains the same over time. And it's less effective in reducing noise which doesn't match the averaged gain pattern. Image (b) could greatly reduce noise. It applies adaptive instant gain pattern from the background noise in real time. But it creates temporal (vertical) speech gaps, which causes the speech stuttering. Image (c) combines the detection results of the two gain patterns. It is more efficient in noise reduction, while retaining the smoothness of human speech. …… 46

6.1 Block diagram of the intelligent event detection and noise reduction systems of this thesis. ……. 47

List of Tables

Page

Table 1: Sound file list. ……. 13

Table 2: WOLA bins that are combined into 22 channels. ……. 25

Table 3 : Detection accuracy for 11 events. It shows the percentage rate of being correctly detected for each event. ……... 33

ACKNOWLEDGMENTS

I want to express my deepest sense of appreciation to my mentor Dr. Kaibao Nie, your expert advice and encouragement have been a strong support to the completion of my theiss. Thank you for your significant help to me with extended discussion and timely suggestions. Your invaluable guidance is the compass to my master thesis work as well as my life path. I would also like to thank Dr. Tadesse Ghirmai and Dr. Rania Hussein for your kind interest in becoming part of the committee and provide me with feedback for the improvement of this thesis.

Chapter 1
Introduction

1.1 Background

1.1.1 General Overview and Motivation

Hearing loss, also known as hearing impairment, is a type of disability where patients partially or totally can't hear sound. About 1.1 billion people suffered from hearing loss to some degree in 2013[4]. In USA, approximately 30 million people are affected by hearing loss[8]. The severity of hearing loss can be categorized as mild (25 to 40 dB), moderate (41 to 55 dB), moderate-severe (56 to 70 dB), severe (71 to 90 dB), or profound (> 90 dB)[2]. A person is considered to have hearing loss if he can't hear 25 decibels of sound in one or both ears[1]. Age and noise exposure are the main causes of hearing loss, accounting for half of all cases[5]. In the USA, 12.5% of children aged 6–19 years have permanent hearing damage from getting exposed in loud noise[6]. It is estimated that half of those between 12 and 35 are at risk of hearing loss from using loud personal audio devices[3]. WHO (World Health Organization) recommends that the highest permissible level of noise exposure in the workplace is 85 dB up to a maximum of eight hours per day[3]. However, only about one out of five people who could potentially benefit from hearing aid actually uses one[7]. There is a considerable market for hearing devices.

Hearing aid and cochlear implant are two main types of hearing devices. A hearing aid is like a sound amplifier which increases the sound volume for patients. On the other hand, cochlear implant sends electrical impulses directly to auditory nerves. Some of the more recent studies have introduced middle ear implants(MEI) which passes sound waves by vibrating the middle ear bones. The Bone-anchored hearing aid(BAHA) bypasses the middle ear and transmits sound directly to the inner ear.

In the past decades, hearing aids have been significantly developed mainly due to the advancement of DSP (Digital Signal Processing) technology[9]. DSP in hearing was first

introduced in 1996. In 2015, DSP technology can be found in 93% of all hearing aids sold in the United States [11].

Unlike analog signal processing, which passes sound signals in the form of electrical voltages and amplify them using electronic components, DSP stores and computes the sound signal digitally. Thus, DSP is more efficient, more functional, and requires smaller physical component size. Digital hearing aids also have the advantage of being flexible and programmable. Complicated mathematical computation is now achievable through the miniature DSP. This has enabled developers to utilize the system in a way that best suits a patient's needs. In recent years, the studies of applying intelligent system into hearing aids are emerging. Intelligent system is supposed to automatically adjust the hearing aid settings for optimal experience for the users. The limited computing power and energy consumption are some of the main reasons holding back the application of smart system implementation. Afterall, DSP has great potential to improve the quality of hearing devices in noise interference reduction, speech frequencies enhancement, feedback cancellation, directional processing, sound environment classification, etc.

The customer satisfaction data from MarkeTrak VII shows that hearing aids users are most satisfied with clearness tone/sound, and sound of voice, while the least satisfied aspects are wind noise and use in noisy situations[8]. It is obvious that there is a great need to improve the noise suppressing ability in hearing devices.

1.1.2 Noise reduction in Hearing Devices

Hearing impaired people often find it struggling to separate useful speech from a noisy background, which is also known as the 'Cocktail Party' problem. That's why improving the noise reduction ability is an urgent task in hearing device development. This thesis focuses on developing noise separation algorithms in hearing device to help deaf people hear speech clearer.

The noise reduction algorithms could be categorized into three groups: spectral subtraction techniques, harmonic extraction techniques, analysis and synthesis techniques[10].

Spectral subtraction techniques involve the use of FFT (Fast Fourier Transform) or filter banks. The signal would be analyzed in the power spectral density form. As shown in Figure 1.1, the noise only power spectral density would need to be obtained. Then the overall signal power spectrum is subtracted by the noise-only power spectrum. The strength of subtraction is set according to the estimated SNR (Signal-to-Noise Ratio). The subtraction result is an estimate of the power spectrum of noise-free signals.

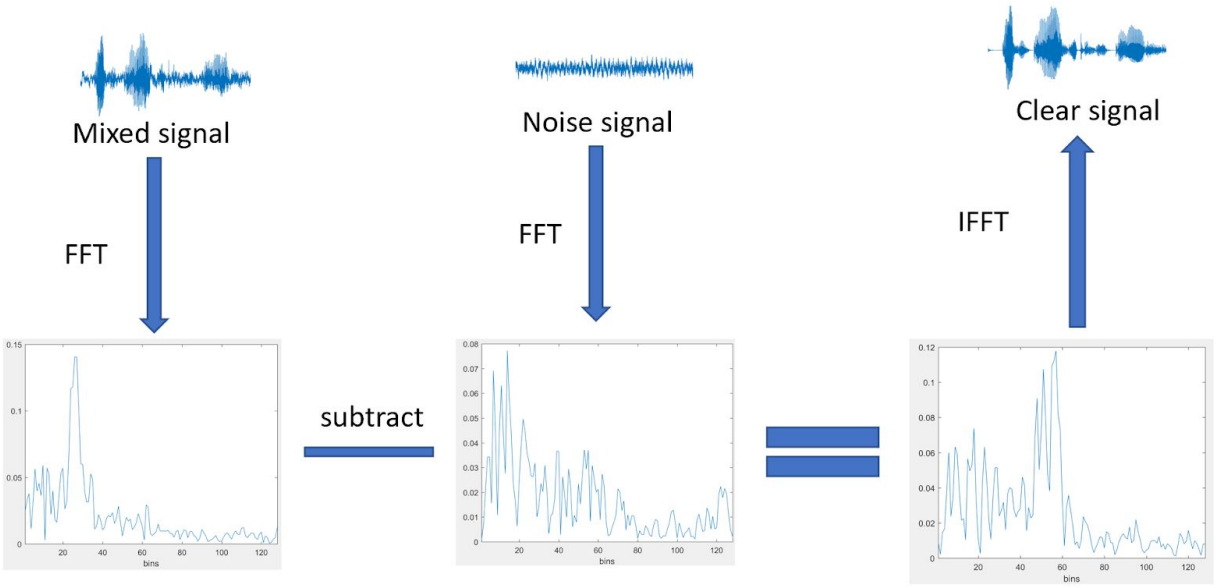


Figure 1.1: Simplified flowchart of the working principle of spectral subtraction techniques.

The implementation of spectral subtraction technique involves designing a non-adaptive noise reduction system. The non-adaptive type of noise reduction is simple but rigid. The term ‘non-adaptive’ indicates that the targeted attenuation template of noise power spectrum is non-changeable. Thus the noise reduction feature is only applicable to a certain fixed kind of noise. Typically this type of design is built for special purpose. For example, some old aviation headsets have this design to reduce the stationary noise of plane engine. While some of the latest aviation headsets already have been updated with adaptive noise cancelling technology.

Adaptive noise reduction is good for nonstationary types of noises. For adaptive noise reduction, it is crucial that information of the noise could be effectively extracted from a piece of mixed audio signals. Because the noise only power spectrum needs to be analyzed dynamically

before applying any sort of noise suppression algorithm. Some existing adaptive noise reduction systems have two microphones. The main microphone records the mixed audio with speech and noise. A secondary microphone is usually located in a distance to the main microphone. And the secondary microphone records only or mostly noise signal. This way, the noise only signal power spectrum could be estimated, analyzed, and subtracted from the mixed audio signal to acquire noise-free signals. This method has been proven successful and is widely used in many areas, especially in wireless communication. For example, most cell phones have two microphones. The main one is located in the lower area close to the mouth to record conversation during a call, while the secondary microphone is in the top of the cellphone to record environment noises. The DSP chip in the cell phones would process the two signals with a comparison algorithm to identify and separate conversation and environment elements. Then a compensation algorithm is applied to enhance conversation and attenuate noise in real time. However, the two microphone setup is not always available on a hearing device. So this thesis will only focus on adaptive noise reduction with single microphone.

Speech pause detection algorithm is one useful method for single microphone noise detection. This algorithm tracks the power envelops of the speech + noise signals and marks the minima point as ‘pause’. Marzinzik and Kollmeier[15] proposed a speech pause algorithm that maintains a low false-alarm rate compared to the standardized algorithm of ITU G.729, even in poor SNR. Their algorithm works well in detecting pauses between two words in a conversation. In our study, another ‘trough detection’ detection algorithm is developed on the basis of speech pause detection algorithm. Chapter 2 presents the detailed algorithm of estimating background noise by tracking power envelope troughs.

For noise reduction, Mwema and Mwangi[12] proposed an algorithm based on the spectral subtraction method to eliminate the Gaussian white noise. In their study, the speech signal is intentionally corrupted by white noise digitally. The performance of the spectral subtraction process is assessed by a comparison of the clean speech signal and the processed speech signal. An average Euclidean distance between the log-area ratios of the clean and processed speech is used to quantitate the performance simulated on computer. The results

showed that the processed speech is clearer after setting their weighted subtraction parameter to 0.5 ~ 1.

Although the conventional spectral subtraction method acquires clearer speech, it might leave an annoying residual musical noise. In another study by Verteletskaya et al. [14], an enhanced spectral subtraction method for noise reduction is proposed. To eliminate this residual musical noise, they introduced a reduced time-varying scaling factor of spectral subtraction, and a weighting function. By their observation, the higher the scaling factor is, the more likely speech distortion will occur. So the varying scaling factor needs to be set to a middle sweet point with highest SNR and minimum acceptable speech distortion. The weighting function is used to attenuate frequency spectrum components lying outside the identified formants regions. Their result shows that proposed method is more effective in reducing background noise in comparison with conventional spectral subtraction algorithm. The enhanced algorithm achieves a substantial reduction of the musical noise without significantly distorting the speech.

Another study by Lu et al. [13] also aimed to eliminate the musical noise from traditional power spectral subtraction algorithm and invalid assumptions about the cross terms being zero. They have proposed to take a new approach to spectral subtraction based on geometric principles, which estimates the cross terms involving phase differences between the noisy (and clean) signals and noise. Their experimental result shows that the proposed new method greatly outperforms the conventional subtract spectrum algorithm.

In this thesis, the spectral subtraction technique is used where the event feature gain pattern is applied to attenuate the noise frequency channels. More about the gain patterns is explained in Chapter 5.

1.1.3 WOLA in DSP

WOLA is one main method used in this thesis. WOLA is an efficient tool for analyzing a long piece of signal. So it is ideal for processing continuous audio recordings for low powered hearing device. It is one example of analysis and synthesis techniques. Its analysis procedure

breaks down a piece of continuous audio signal into separated small segments for feature extraction process or other tasks. When finished, each segment goes through the synthesis procedure, which is the opposite operation of analysis, to be combined into a continuous piece of signal.

WOLA is an efficient and powerful tool for signal processing. Brennan et al. [28] have built an adaptive filtering system using WOLA in an ultra-low-power DSP, and stated that's an extremely efficient and elegant solution. Frey et al. [33] implemented an efficient real time channelizer filter based on WOLA. According to their study, WOLA uses less hardware resource than a straight-forward implementation. Also, Sheikhzadeh et al. [34] proposed a delayless method for adaptive filtering through subband adaptive filters (SAF) systems. This method is based on WOLA. The delayless goal is achieved with the help of WOLA's block processing feature which doesn't involve long FFT or IFFT operations. Ahadi et al. [27] have used WOLA filterbank for feature extraction in front-end processing for their speech recognition system. Their result shows that WOLA usually have equally or slight better performance than other well-known feature extraction technique like MFCC (Mel-Frequency Cestrum Coefficients), especially in lower SNR. Besides the great accuracy, WOLA also provides benefits such as: efficient implementation; easy application of enhancement algorithms; and perfect or near-perfect reconstruction property. They concluded that WOLA has the potential to be further integrated into speech enhancement system in hearing aids.

Many studies on WOLA implementation in low power devices have been done. WOLA has been proven to be a robust and effective tool for noise reduction systems. Schuster et al. [36] has discussed about WOLA's noise cancelling performance using different schemes. They also explained how to apply WOLA structure to an optimal minimum mean square error (MSE) filtering, and how to find optimal gains for each WOLA frame. Brennan et al. [16] proposed two extremely low-power noise reduction systems suitable for industrial applications and hearing aids. They developed a robust spectral subtraction noise reduction algorithm and a low delay noise attenuation algorithm, both of which are tightly coupled with a highly optimized WOLA filterbank design. Their results showed that the low delay algorithm has a substantial noise attenuation without noticeable artifacts or degradations on audio signals. The spectral subtraction

routine also performed well except that at SNR lower than 5 dB the algorithm failed to obtain accurate noise models and resulted in artifacts in the reconstructed speech. In another study by Vicen-Bueno et al. [35], WOLA filterbank is used in a modified LMS-based (Least Minimum Square) feedback-reduction subsystems in digital hearing aids. The application of WOLA provides great flexibility to adapt the compensatory gain of each patient according to his/her hearing losses.

To study further into the combination usage of WOLA and spectral subtraction, Anil et al. [17] have experimented on the real time implementation using the two methods as well as Voice Activity Detection (VAD). Their result shows that the DSP processor performs equally well as MATLAB simulation in suppressing unwanted noise. In this study, WOLA is used to extract event noise features, which would be used in neural network. More details are discussed in Chapter 3.

1.1.4 Intelligent System in DSP

Intelligent system in hearing devices has been gaining attention in recent years, due to the increased processing power in low cost/low-power DSPs. Traditional DSP algorithms are made up of equations and parameters. On the contrary, neural network generates new algorithms for any new task all on its own. The training process is a procedure the computer tries and tests the weight values, until it finds a combination of weight values that will lead to a correct output. Learning rules could be categorized into: off-chip learning, chip-in-the-loop learning, and on-chip learning[21]. In this thesis, off-chip learning is used, since the experiment is carried out on Matlab simulation.

Neural network is the computer simulation architecture inspired from biological human brain. Neural network consists of components like neurons, weights, bias, and propagation function. A neural network model typically has one input layer, one or multiple middle hidden layers, and one output layer, as shown in Figure 1.2.

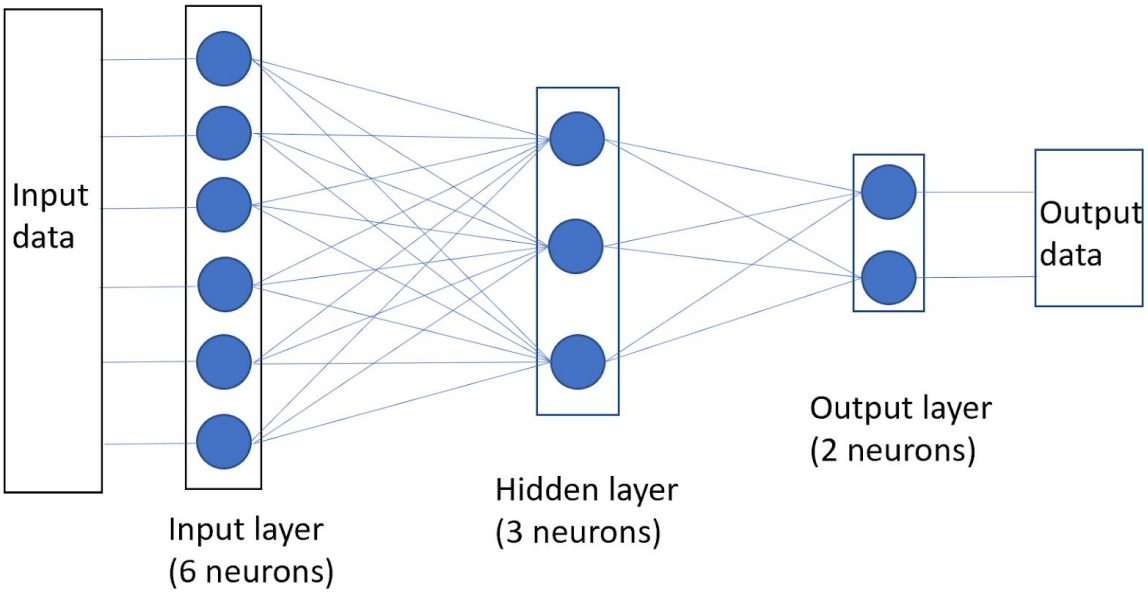


Figure 1.2: a simple visual realization of neural network design with one hidden layer, 6 inputs and 2 outputs.

Inside a neuron operation, each neuron input value is multiplied by a weight, and then they are summed together and processed by a non-linear sigmoid function, as in Figure 1.3.

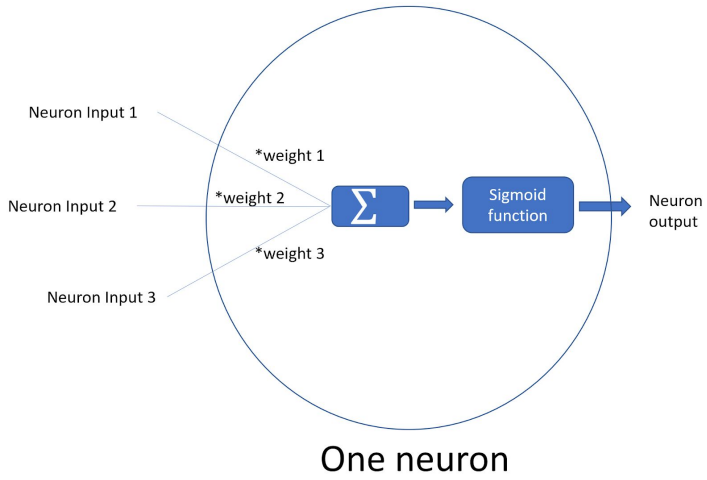


Figure 1.3: a simple diagram to demonstrate the operations inside a neuron.

Back in 1997, Mohamadian et al. [18] have implemented an artificial neural network on a Texas Instruments TMS320C30 digital signal processor (DSP) based system. Their neural network was trained to perform as an indirect field oriented controller.

The neural network running efficiency is exploited by Behan et al. [19]. They have performed ways to accelerate Integer Neural Networks (INN) on low cost DSPs by using DSP instructions. Mats [21] has also performed a practical way to implement artificial neural networks in hardware. Jung et al. [20] has given another example of implementing neural network in DSP to control a robotic hand in real time. They used a general feed forward structure that has an input layer, a hidden layer , and an output layer with 3 output units representing x-y-z axis movements. Their inverted pendulum experiment is more complicated, where a horizontal moving cart keeps an inverted pendulum standing straight. Compared to the PID (Proportional–Integral–Derivative) controller which partially fails and keeps moving toward one direction, their neural network controller controls the cart movement precisely and always keeps the pendulum perpendicular even when external force disruption is applied.

One popular application of neural network is classification, such as image recognition. Elaina [32] has implemented a Lane Detector Algorithm using a Deep Convolutional Neural Network for feature extraction, with the hardware is the 66AK2H Evaluation Model (EVM) by Texas Instruments (TI). The whole system works well unless when 16-bit fixed point is used which leads to data precision loss and algorithm failure. In another study by Namjin et al. [33], a DSP-Based Hierarchical Neural Network is used for classification of 11 modulation signals by analyzing on 31 statistical signal features. Hierarchical Neural Network is a more complex form consisted of stacked layers of neural networks. Compared to four classical classifier, their Hierarchical Neural Network classifier has achieved improvements in both processing time and classification rate. It is known neural network is useful for feature classification/detection. In this thesis, the neural network is used to classify 11 sound events based on 22-channel spectrum features.

1.1.5 Signal Processing in Cochlear Implants

Cochlear implant sends electrical impulses directly to auditory nerves. Cochlear implant also works for deaf or severely hearing impaired patients, since it bypasses the damaged hair cells. Advanced Combinational Encoder (ACE) and Continuous Interleaved Sampling (CIS) are two main types of signal processing strategies found in cochlear implants. Both strategies have up to 22 channels connected to 22 implanted electrodes.

ACE amplifies the audio signal and then divides it into 22 channels using band pass filters. The envelope of all 22 channels will be calculated, and then 8~10 channels with the largest envelope values are selected. The electrodes corresponding to the selected channels are activated simultaneously. The whole procedure is shown in Figure 1.4.

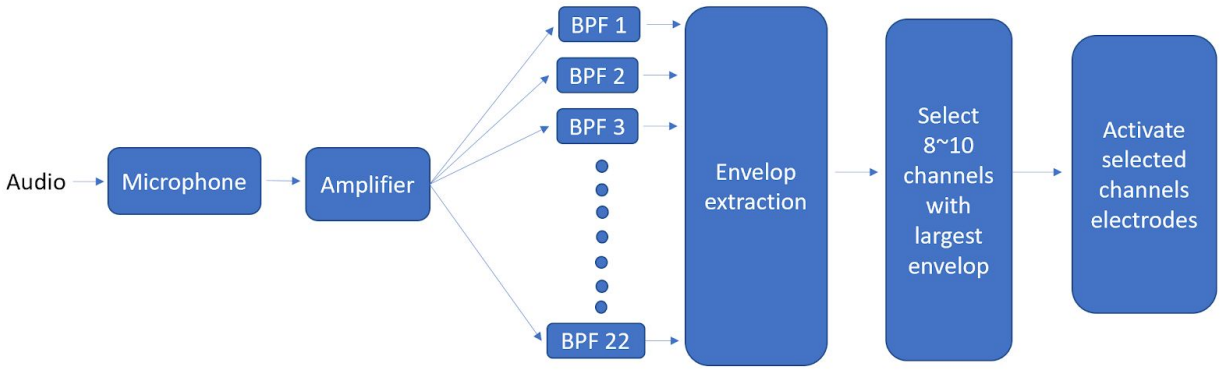


Figure 1.4: Diagram of working principle of ACE.

CIS also divides the audio signal into a number of channels using band pass filters. And then full wave rectifier and low pass filter are applied to each channel to calculate the envelope. Later, all channels are arranged in sequence, and corresponding electrodes are activated from one to one in a constant rate, as shown in Figure 1.5.

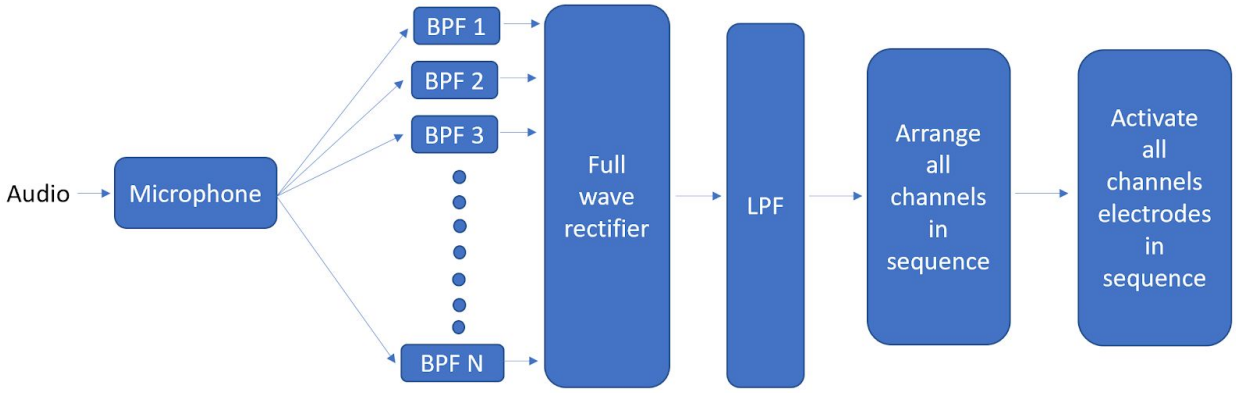


Figure 1.5: Diagram of working principle of CIS.

1.2 Overview of Thesis

Sound event is the audio segment that humans would label as a distinctive concept in an acoustic signal [22]. In this thesis, sound event refers to a set of environmental noises in different scenes such as ‘cafe’, ‘station’, ‘hallway’, etc. The acoustic signals used for analysis are mixtures of human vocal speech and real life background noise recordings under a variety of event categories. The proposed intelligent background sound event detection system aims to classify a given sound signal into one of the preset scenes. The scene recognition system could filter out the speech and focus on the background sound. The stages of sound event detection include trough detection, WOLA processing (frequency domain analysis), feature extraction, and neural network classification.

The first step of background event detection is background noise extraction. Given a piece of mixed audio, the noise signal can’t be separated from speech signal directly. Therefore, the overall sound signal is represented in energy envelop form. And then trough detection is used to extract only the background signal information. To apply band-specific gains to attenuate noise, WOLA is used to transform the time domain signals into frequency domain signals. Then a sound signal is split into 128 bins representing frequency from low to high. Each bin’s magnitude represents the signal’s energy within a certain frequency range. To make it more feasible for future implementation of the noise reduction system in embedded DSP systems, the

number of bins are reduced down to 22 in the present study. Also, the use of feedforward neural networks for sound event detection is proposed. The motivation of this work is that neural network can use different sets of its hidden units to model multiple simultaneous events in a short time frame. Spectral domain features are used to characterize the audio signals. And a neural network is trained using those spectral features to learn a mapping between feature sets and sound events. Generally, the input training data in the frequency domain of a number of sound files are prepared. Then the ability of this system in sound event detection is explored by evaluating the detection accuracy of event classification. Moreover, the event feature curves are transformed into gain patterns for noise reduction purpose. The results have been proven to be effective in reducing targeted background event noise.

The structure of the paper is as follows (see Figure 1.6). The trough detection algorithm used for separating background noise from speech audio signals is presented in Chapter 2. The feature extraction procedure, including the use of WOLA for sound analysis and synthesis, as well as the smoothing of estimated noise templates, are described in Chapter 3. In Chapter 4, the application of neural networks in sound event classification and its detailed structure design are discussed, as well as the evaluation of the detection accuracy derived from the testing dataset. The application of noise reduction gain patterns and results are discussed in Chapter 5. And lastly, the conclusion and future work are discussed in Chapter 6.

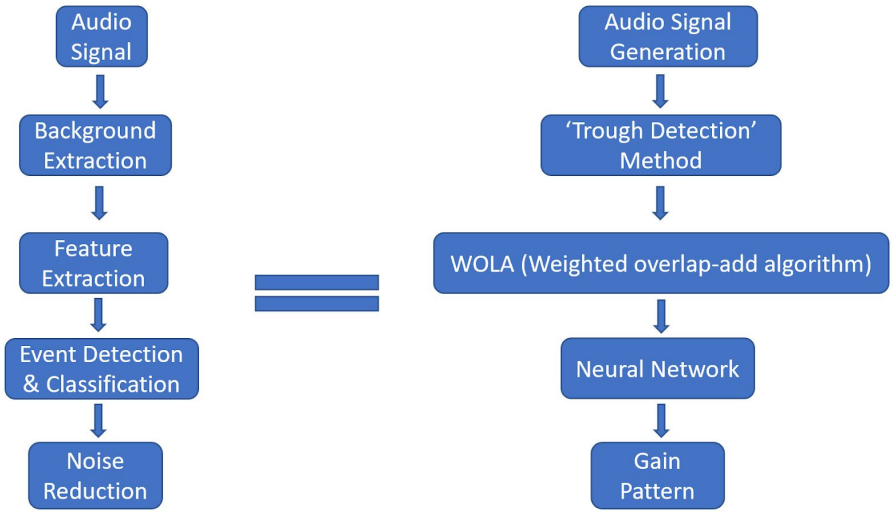


Figure 1.6: Framework of the main signal processing procedures in this thesis.

Chapter 2
Environmental Background Noise Extraction

2.1 Audio Preparation

The background noise audio files are downloaded from the Creative Commons website. They are a collection of 11 sound events, including 'cafe', 'cafeteria', 'hallway', 'kitchen', 'living', 'meeting', 'office', 'resto' , 'square', 'station', and 'washing'. Each event contains 16 files, and each file lasts 5 minutes long. In all, the sound templates last 80 minutes long for each event, or a total of 880 minutes of recordings for all 11 events.

Events	Number of files	Length of each file	Total length of audio	Major sound elements
café	16	5 min	80 min	chatting, tableware, footsteps
cafeteria	16	5 min	80 min	chatting, tableware, footsteps
hallway	16	5 min	80 min	footsteps, door
kitchen	16	5 min	80 min	water streaming, dishes
living	16	5 min	80 min	TV music, newspaper
meeting	16	5 min	80 min	consersation, desk
office	16	5 min	80 min	desk, typing
resto	16	5 min	80 min	chatting, tableware, footsteps
square	16	5 min	80 min	crowded people, chatting, birds
station	16	5 min	80 min	vehicles, white noise
washing	16	5 min	80 min	water streaming

Table 1: Sound file list.

The content of each sound file is a piece of continuous real life recording in that event location. And the major sound elements of each event are listed in Table 1.

The ideal type of background noise signals for detection should be stationary with a constant volume. Because the more uncertainty in noise, the more difficult it is for the system to classify the event. For example, if the training set of ‘park’ contains children laughing sounds, but the detection sample doesn’t. It might confuse the detection system and lead to less accurate detection results. Events of ‘hallway’, ‘kitchen’, ‘office’, ‘square’, ‘station’, and ‘washing’ are presumed to be relatively less challenging for detection, since they contain fewer sound elements. Events of ‘cafe’, ‘cafeteria’, ‘living’, ‘meeting’, and ‘resto’ are presumed to be

difficult for detection, since they contain varying sound elements. For example, events ‘living’ and ‘meeting’ contain long blocks of human conversation as distraction. And events ‘cafe’, ‘cafeteria’, and ‘resto’ sound very similar that they are even indistinguishable to normal human ears. All the events above are used in this thesis to test out the full ability of the intelligent detection system.

The speech file contains a piece of human speech consisting of 3 sentences and lasting about 6 seconds. The speech sentence is ‘A boy fell from the window’, ‘The wife helped her husband, and ‘Big dogs can be dangerous.’ If more than 6 seconds of speech signal is needed, this piece of speech audio would repeat itself to make up for the full length.

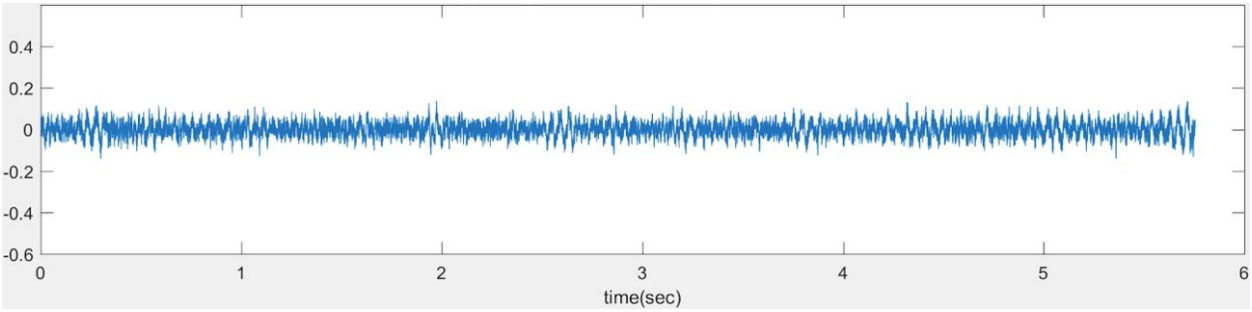
The ideal type of sound signal for this thesis would be the simulated real life auditory situation of speech under realistic environmental noise. The reason to generate simulated mixed audio is that recordings of speech under many different types of events are difficult to find. Therefore, the audio signals used in this thesis are mixed audios generated by mixing human speech signals with event noise signals. The Signal to Noise Ratio (SNR) at 0 dB and +5 dB are used for the sound event classification experiments. The sampling rate is 24 kHz. To generate a mixed audio at desired SNRs, the root mean square (RMS) of human speech and background noise are calculated, and then the formula below is used to create mixtures:

$$S_{mixed} = S_{speech} + S_{noise} \times \frac{RMS_{speech}}{RMS_{noise}} \times 10^{-1 \times \frac{SNR}{20}} \quad (1)$$

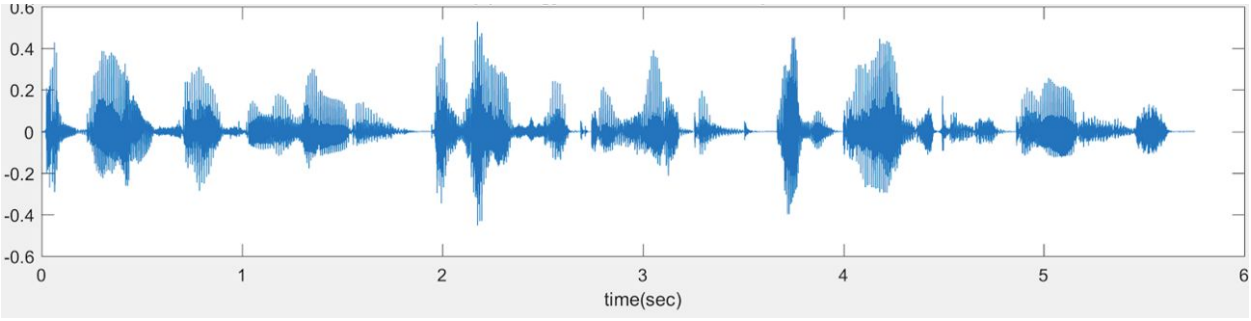
Figure 2.1 shows one example of the human speech, background noise and their mixture at SNR = +5 dB.

As mentioned previously, background extraction out of the mixed audio is the crucial part and also the first stage for adaptive noise reduction system. To extract the background noise data for further analysis, the sections where mixed audio signal has very small SNR (noise dominated) need to be found. For example, in image (c), the arrow points to an ideal noise extraction area. By comparing the signals (a) (b) and (c), it’s clear that the background noise is dominating (meaning small SNR) around 1.8 ~ 1.9 seconds for the mixed signal (c). However, if

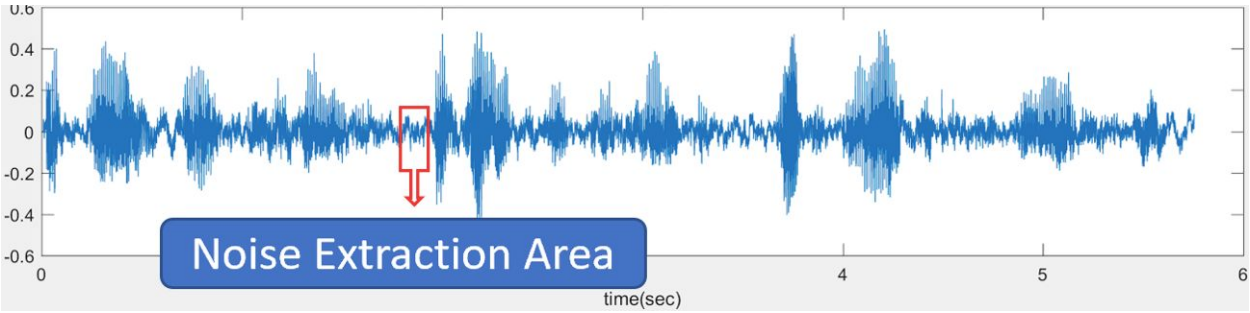
only signal (c) is provided, it would be difficult to visually find the noise extraction (noise dominating) area. Therefore, the 'trough detection' method is introduced in the next section.



(a)



(b)



(c)

Figure 2.1: Signal (a) is the event 'cafe' background noise. Signal (b) is the human speech audio signal. Signal (c) is the addition combination of the speech and noise at SNR = +5 dB. The arrow on signal(c) points to an ideal noise extraction segment.

2.2 Trough Detection Algorithm

The objective of the proposed 'trough detection' method is to temporarily locate the background noise data from the mixed signal. In this experiment, the SNR is +5 dB, meaning that vocal speech is 5 dB louder than background sound. The mixed audio signal is first converted to a sound energy waveform, and then the energy throughs are detected. Energy troughs are assumed to contain background noise-only data. Because human speech usually has periodic higher energy, while the background has relatively constant and lower energy. The last step is to extract these energy troughs.

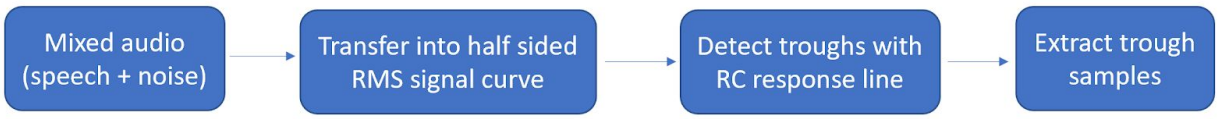
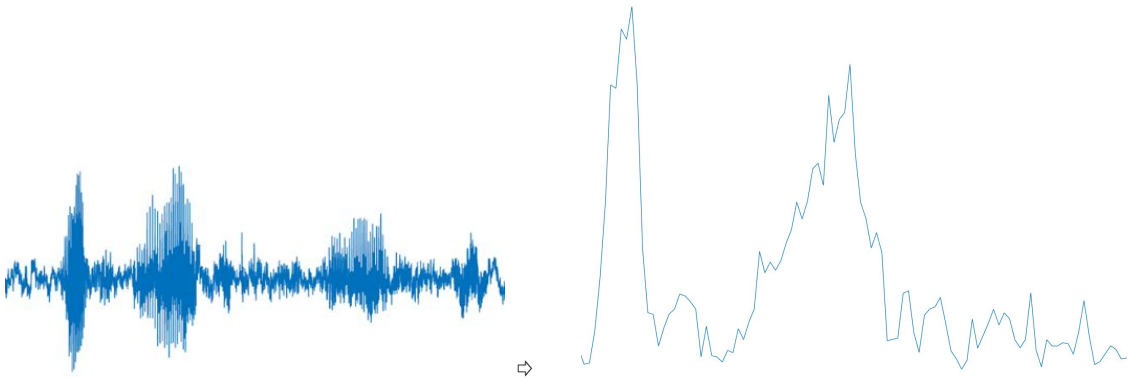


Figure 2.2: The flow chart of trough detection procedure.

Here the details about how to acquire the energy waveform are discussed, as shown in Figure 2.2. First of all, a high pass filter is applied to the mixed audio signal to enhance high frequency sound, since the low frequency parts usually have much higher volume than mid or high frequency signals. And then a half-wave rectifier is applied. Lastly, every 256 original audio data samples(equivalent to roughly 21.5 ms of signal) are transformed to one RMS (Root Mean Square) sample to generate a smooth RMS curve. This RMS curve (see Figure 2.3) is used to track energy troughs that contain primarily background event sound clips.



(a) (b)

Figure 2.3: Signal (a) is the original mixed audio signal. Signal (b) is the energy waveform derived from the audio signal.

Now the ‘trough detection’ algorithm could be applied to energy waveforms with the following procedure steps (also see Figure 2.4):

1. From the starting point, follow the curve of energy waveform.
2. When an energy trough is found, mark it as ‘energy trough’.
3. Draw a diagonal line starting from this ‘energy trough’.
4. Stop the diagonal line where it meets the audio energy waveform, and follow the curve of energy waveform again.
5. Again, when an energy trough is found, mark it as ‘energy trough’.
6. The same process repeats itself by looping step 3 to step 5, until the end of energy waveform is met.

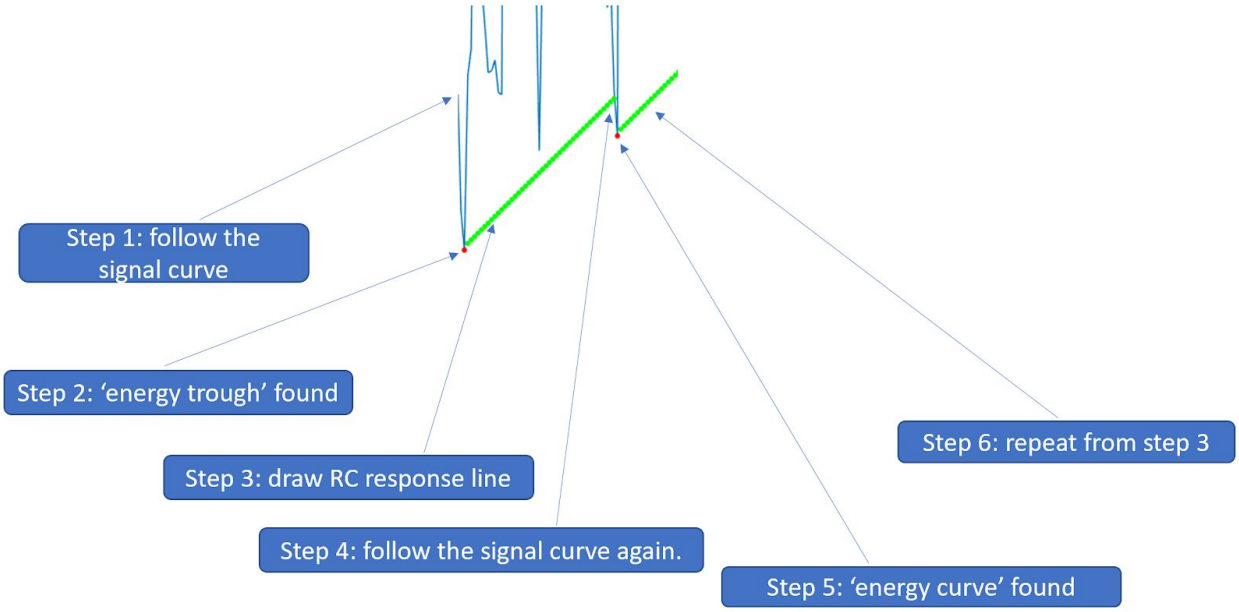


Figure 2.4: Demonstration of the decision rules of the trough detection algorithm.

As a pre-processing step for the feature extraction, the 'trough detection' method must be accurate enough to detect correct troughs which contain noise-only data samples. The diagonal line represents the RC (Resistor Capacitor) response time. In a Resistor Capacitor circuit, the RC response time is the time it takes to charge the capacitor from 0 to 63% of its max value. And the RC response time determines the time constant (tau-τ), defined by resistance R multiplies capacitance C. A similar idea is used in this trough detection algorithm. The slope angle of the line is determined by the tau value. The appropriate tau value to be used varies case to case, depending on the signal's energy magnitudes and sampling rate. To double check whether or not the proper tau value is being used, the trough detection plot should always be drawn to show that reliable troughs are being detected, like the example in Figure 2.5. A reliable trough is a 'global' trough which is considered a low energy spot through the entire timeline of energy waveform. The use of RC response line prevents extracting an unreliable 'local' trough. 'Local' troughs are low energy spots in small local segments on energy waveform. They generally have lower energy than adjacent troughs but still have higher energy than 'global' troughs. In Figure 2.5, a 'local' trough is marked by the red cross. A higher energy trough indicates that it probably

contains speech data samples. So ‘local’ troughs are not wanted, as they don’t represent noise-only samples. After ‘trough detection’, all the ‘global’ troughs are extracted.

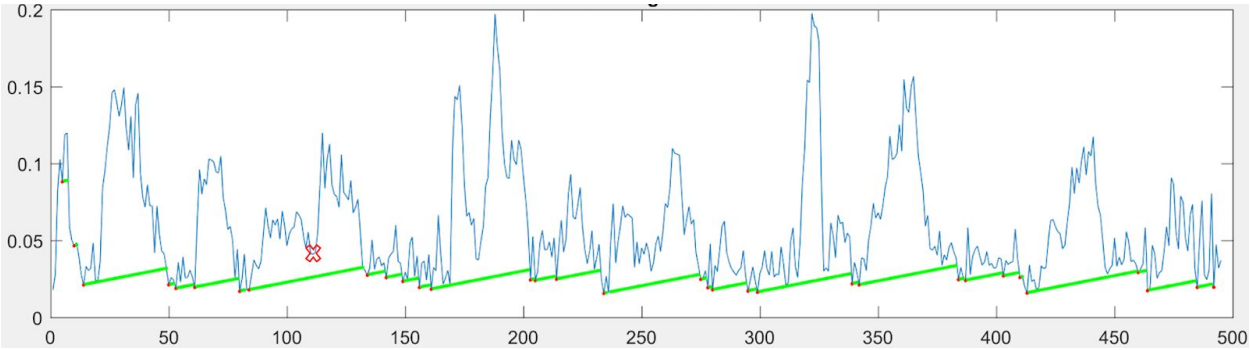


Figure 2.5: Demonstration of the application of the trough detection algorithm. The energy waveform is derived from a mixed audio signal with event ‘station’ at SNR = +5 dB. The x axis is the number of energy waveform samples. The green line is the trace of RC response line. The red dots are troughs detected, and they represent the background noise-only samples to be extracted.

It is important to make sure that trough detection algorithm is working properly. Ideally, the detected troughs extracted from the energy waveform should contain only the background noise samples. To double check, the sample data from the extracted troughs need to be drawn in comparison with the mixed audio signal and background noise signal. The example shown in Figure 2.6 demonstrates that the trough signal closely resemble the background signal. It can be concluded that the trough signal is a faithful representation of the actual background signal. In the ‘trough samples’ plot, the spikes are sound segments where the detection system failed to extract the noise-only data and picked up a small portion of speech signal instead. One of the spikes is highlighted by a red oval. Overall, the ‘trough detection’ method can successfully extract background noise information apart from the speech signal. Later on, the extracted troughs samples are processed by WOLA for feature extraction. More details are discussed in Chapter 3.

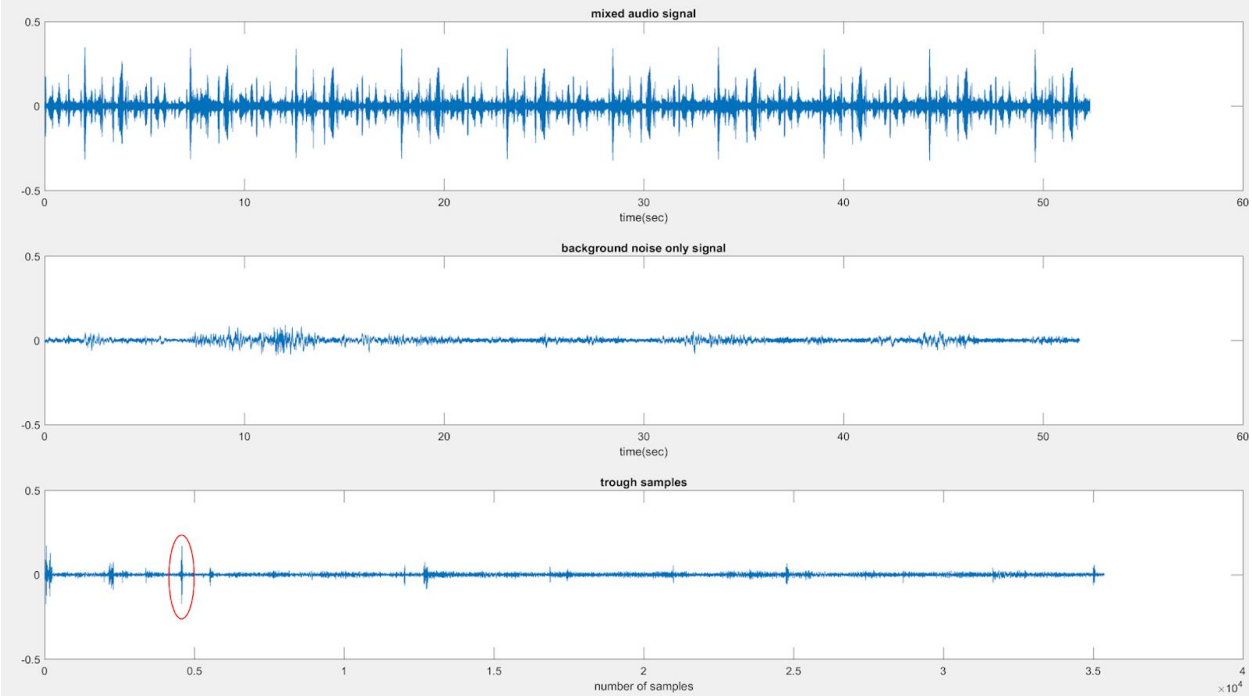


Figure 2.6: comparison of the mixed audio signal to the background noise only signal and extracted troughs samples. The noise event ‘cafe’ is used, with SNR = +5. It shows that the trough signal is a good representation of the background noise signal. Thus, it shows that the trough detection algorithm is working properly as designed to extract the background sound clips from the mixed audio signal.

Chapter 3
Feature Extraction

3.1 WOLA Spectral Analysis Introduction

Weighted overlap-add algorithm (WOLA-algorithm) is widely useful in analyzing audio signal. WOLA analyses the time domain signal block by block, making it an ideal method for real time processing devices. Compared to FFT method, WOLA is more suitable for hardware implementation with less distortions created between frames, especially for low power devices, such as hearing devices. WOLA is used in wireless communication devices such as cell phones for noise reduction. The offset quadrature amplitude modulation (OQAM)-filter bank based multicarrier (FBMC) is one of the candidates for 5G technology [29], and it could benefit from using a WOLA-based structure for its realization[30]. Moreover, Schuster and Ansorge [31] have experimented on WOLA's ability to filter out background noise in a noise canceling system. And their result shows that it is capable of delivering de-noised speech.

WOLA is an efficient algorithm to calculate the discrete convolution for a length of signal. In our study, a small section of time domain audio signal from each extracted trough is saved and processed by WOLA. WOLA has two processes: analysis and synthesis. WOLA analysis use overlapping window to transforms time domain signal into input blocks. And then those time domain input blocks are Fourier transformed to give a short-time Fourier spectrum. In short, its main function is to transform a finite number of overlapped block by block time domain signal into frequency domain. Its mathematical framework is defined as[24]

$$X_k(sR) = \sum_{m=-\infty}^{\infty} h(sR-m)x(m)W_M^{-mk} \qquad (2)$$

where $x(m)$ is a time domain signal sampled every R samples in time m, $W_M = e^{j2\pi/M}$, M is the number of frequency samples, k is the discrete frequency index, $h\ (m)$ is the analysis window, and s is the time index of the short-time transform at the decimated sampling rate (decimated by the integer factor R).

WOLA synthesis is like the reverse of WOLA analysis. It applies inverse Fourier transformation to segments of short-time Fourier spectrum and sum the results with overlap to obtain audio signal in time domain. Its main characteristic is to transform frequency spectrum data back to a finite length of time domain signal. Its mathematical framework is defined as[24]

$$\hat{x}(n) = \sum_{s=-\infty}^{\infty} f(n - sR')\frac{1}{M}\sum_{k=0}^{M-1}\hat{X}_k(sR')W_M^{nk} \tag{3}$$

where $\hat{X}_k(sR')$ is a discrete short-time spectrum, output $\hat{x}(n)$ is considered as sampled every R' samples in time n, and $W_M = e^{j2\pi/M}$, M is the number of frequency samples, k is the discrete frequency index, $f(n)$ is the synthesis window. and s is the time index of the short-time transform at the decimated sampling rate (decimated by the integer factor R').

The WOLA filterbank is a low-delay, highly efficient implementation of over-sampled generalized DFT (Discrete Fourier Transform) system[25, 26]. For WOLA analysis (see Figure 3.1), the first step is to shift R input samples at a time to the buffer with frame length L[27]. Note that when odd stacking is used, the input signal must be flipped in sign every N samples [28]. And then, this buffer will be processed by an analysis window with also length L. Lastly, this L-sample frame will be folded into N-sample blocks for Fast Fourier Transform(FFT) process to produce spectrum output in complex numbers. For WOLA filterbank synthesis (see Figure 3.2), the procedure is the inverse of WOLA filterbank analysis.

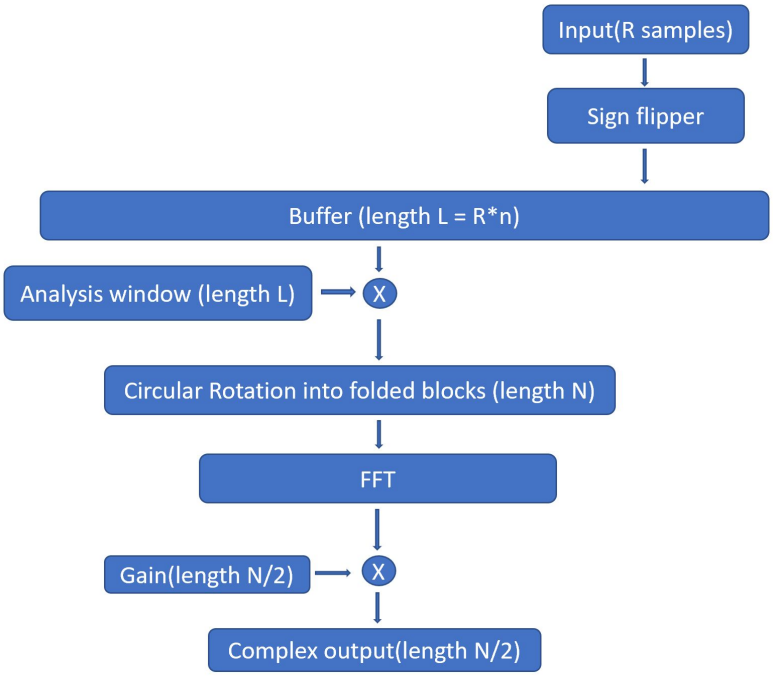


Figure 3.1: Block diagram of WOLA filterbank analysis procedure.

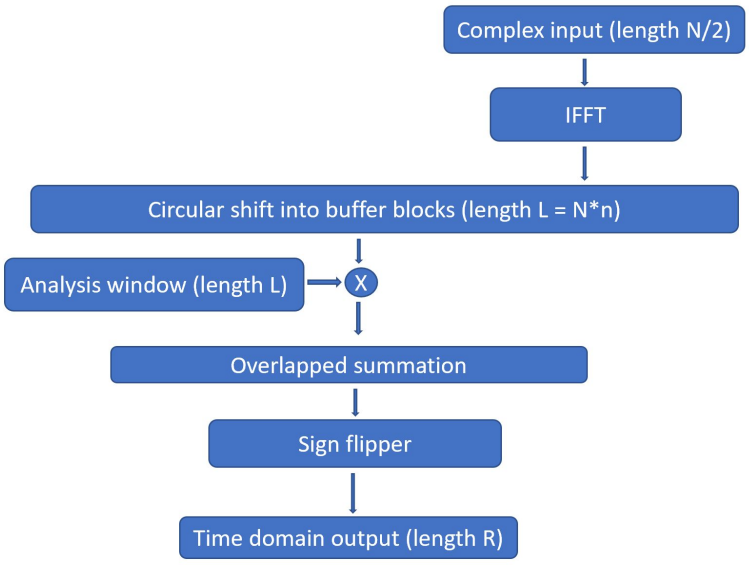


Figure 3.2: Block diagram of WOLA filterbank synthesis procedure.

The operation of WOLA is determined by a number of parameters: Analysis window size (La), Synthesis window size (Ls), Input block step size (R), and FFT size (N). WOLA filterbanks can do either EVEN or ODD stacking. ODD stacking has N/2 full-width bands; EVEN stacking has N/2-1 full-width bands, and two half-width bands at DC and the Nyquist frequency. In EVEN stacking the DC and Nyquist band outputs are real and are packed into the first element of the complex output data, with the real part containing DC and the imaginary part containing the Nyquist band output. All other bands have complex outputs. For ODD stacking, all bands have complex outputs. Note: The oversampling factor is OS = N/R. Oversampling is how much above critical sampling (i.e., at the Nyquist rate) the band outputs are sampled. The decimation factor is defined as DF = La/Ls. This name comes from the fact that the synthesis window is often constructed by decimating the analysis window (which is the case for the default windows created at the initialization of a WOLA object). In our experiments, La is set to 512, Ls is 256, N is 256, R is 64, and odd stacking is used. This parameter setting produces reasonable spectral and temporal resolutions for noise reduction.

3.2 Features for Classification

All the energy troughs extracted previously are now processed by WOLA analysis to generate spectral features to a neural network for sound event detection, as shown in Figure 3.3.

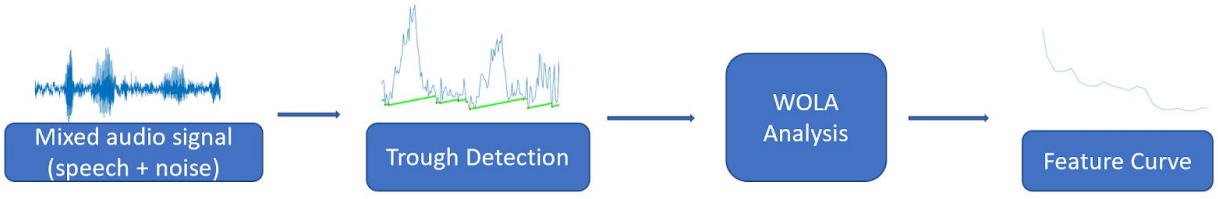


Figure 3.3: Workflow of how to obtain feature curve.

In our study, each energy trough contains 256 time domain audio samples. Then each energy trough could be processed through WOLA analysis and return 128 complex numbers

representing the value of 128 bins in frequency domain. The next stage is to calculate their magnitude value to convert them into real numbers. Now the characteristic of a background event could be visualized by drawing the feature curve with 128 real numbers. After a 128 bin feature curves is created, it is converted into a 22-channel feature curve by bin merging. The whole procedure is shown in Figure 3.4. In cochlear implants, only up to 22 channels of stimulating signals are generated to mimic normal hearing. Since the speech signal generally occupies the low-mid frequency, it is important to reserve a higher resolution of data in 0.1 kHz ~ 3 kHz frequency range. Thus, there are less bins (2 ~ 3 bins) merging into one channel in the low-mid frequency range. While there are more bins (4 ~ 10+ bins) merging into one channel in high frequency area. The bin merging pattern is as follows: [1 6 8 10 12 14 16 18 20 22 26 30 34 38 44 50 58 66 76 86 98 112], also as shown in Table 2. Each number represents the starting bin index of 128 bins that will be merged into one of the 22 channels. For example, the 1~5 bins of 128 bins data will be merged into the first channel. And 6~7 bins of 128 bins data will be merged into the second channel.

128 bin number	1~5	6~7	8~9	10~11	12~13	14~15	16~17	18~19	20~21	22~25	26~29
22 bin number	1	2	3	4	5	6	7	8	9	10	11

128 bin number	30~33	34~37	38~43	44~49	50~57	58~65	66~75	76~85	86~97	98~111	112~128
22 bin number	12	13	14	15	16	17	18	19	20	21	22

Table 2: WOLA bins that are combined into 22 channels.

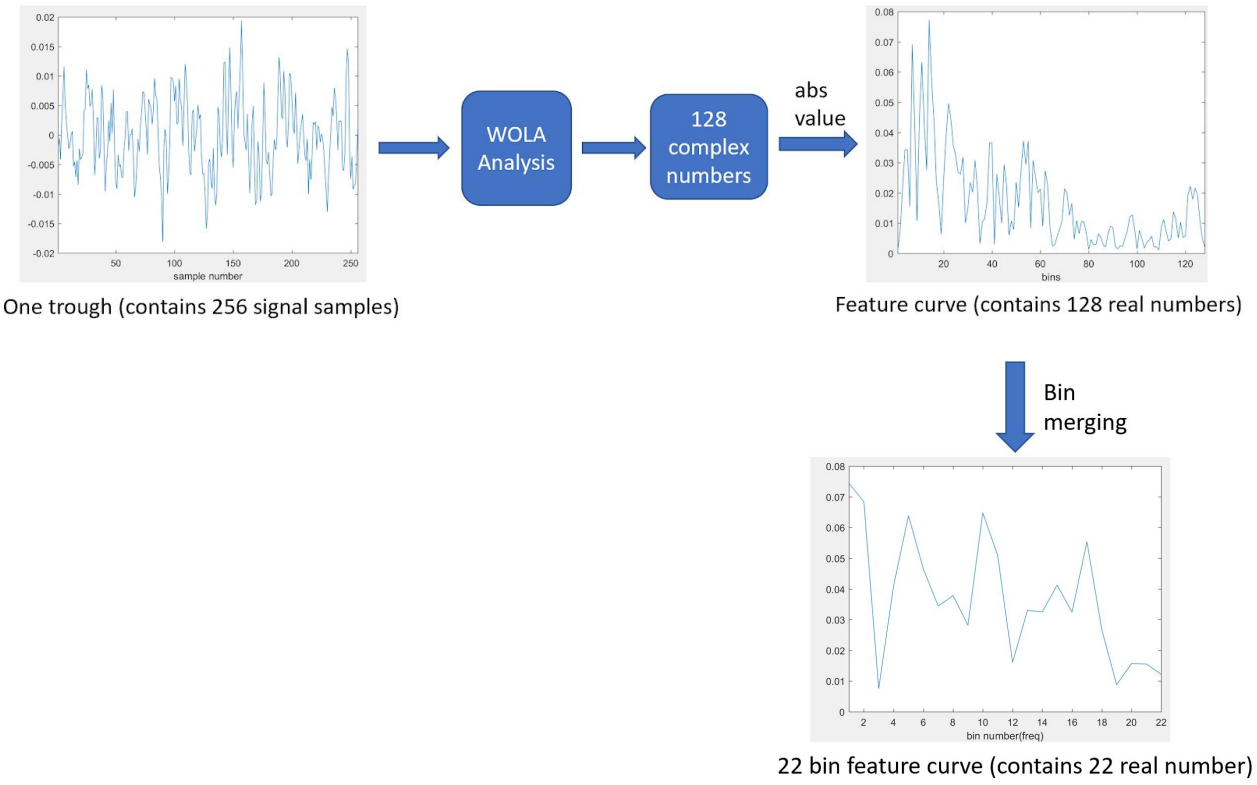
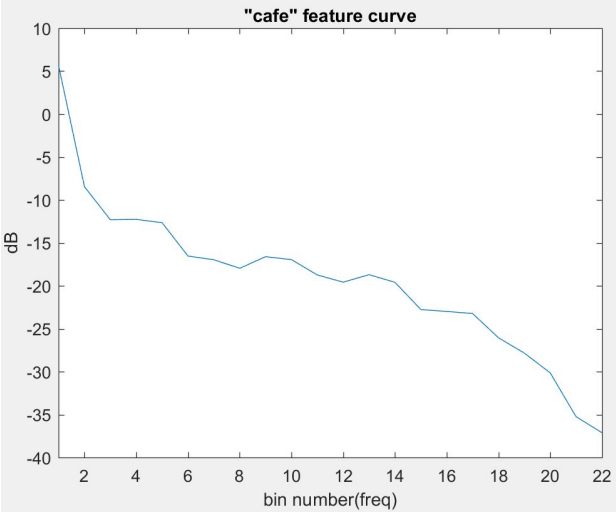
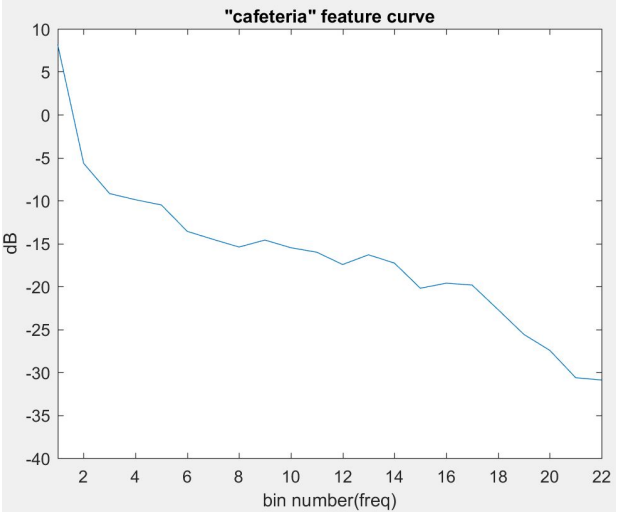
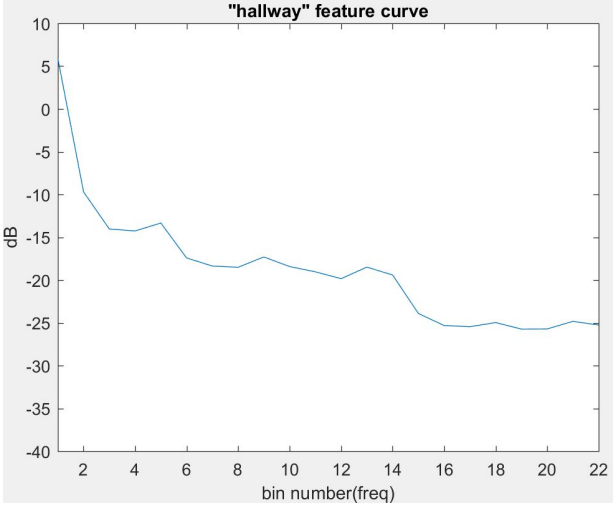


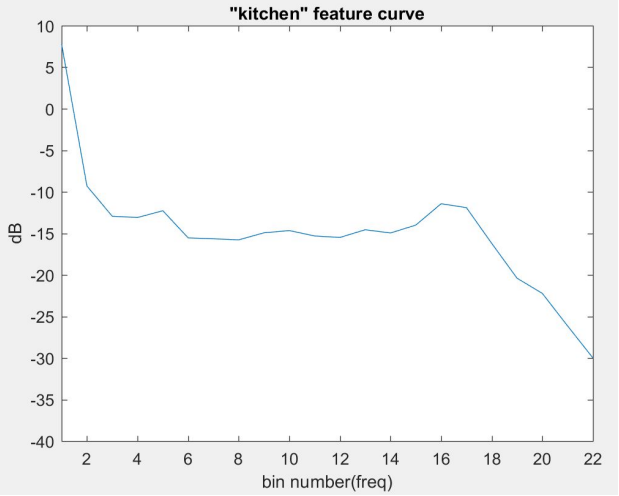
Figure 3.4: Block diagram to show how one energy trough is transformed into a feature curve through WOLA analysis.

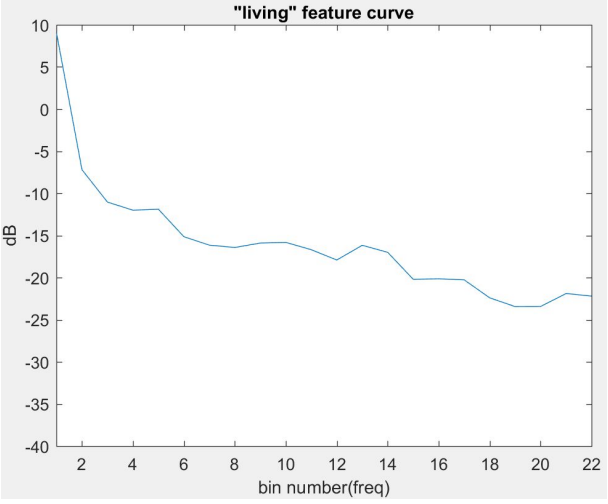
Each feature curve is different and unique. However, the feature curves from the same event is expected to share some similarities. Therefore, each event should have its own signature feature curve shape. For better observation, the plots of each events' averaged feature curve out of roughly 1500 feature curves are shown below. Those plots give valuable information about how this intelligent detection system 'sees' the events. In other words, the similarity of the feature curves between two events indicates the difficulty for machine classification decision.

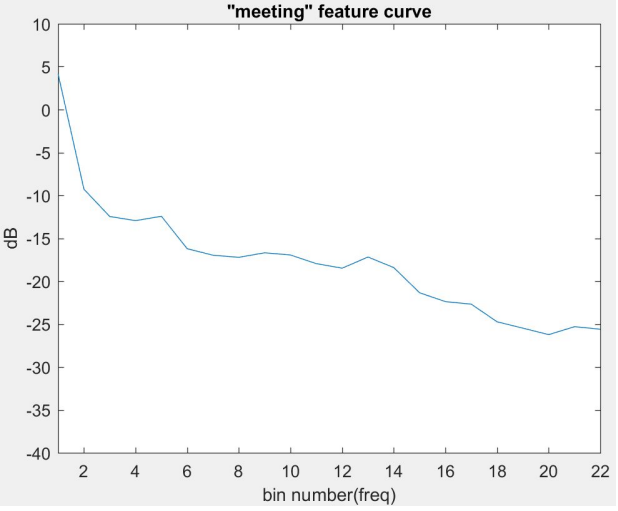

"cafe" feature curve
dB
bin number(freq)

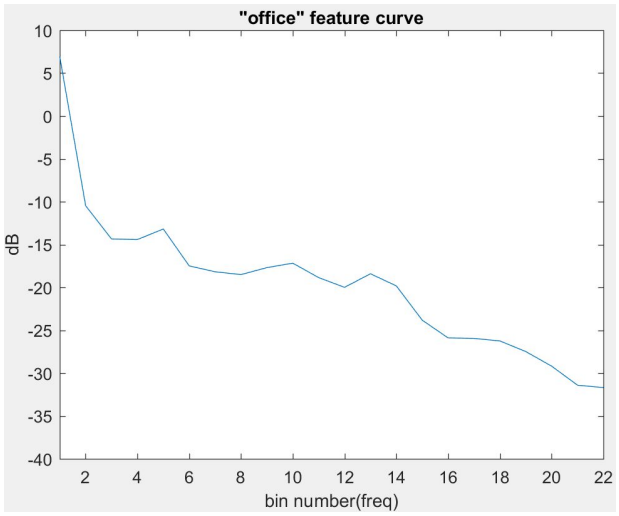

"cafeteria" feature curve
dB
bin number(freq)

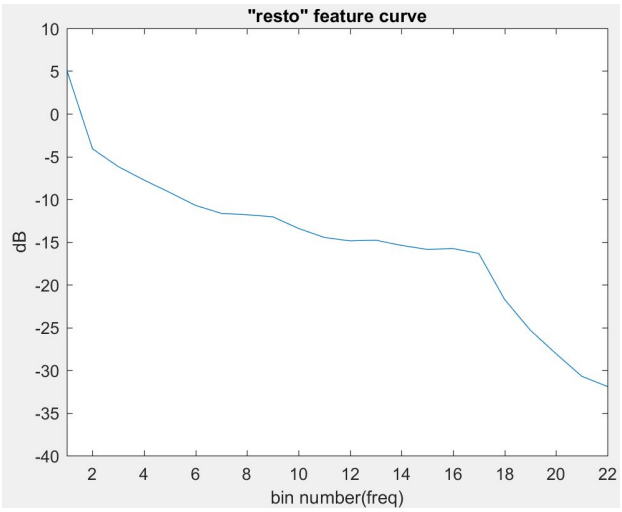

"hallway" feature curve
dB
bin number(freq)

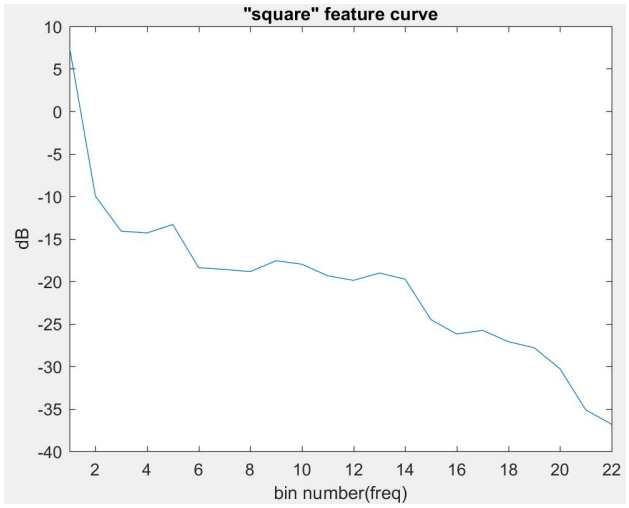

"kitchen" feature curve
dB
bin number(freq)

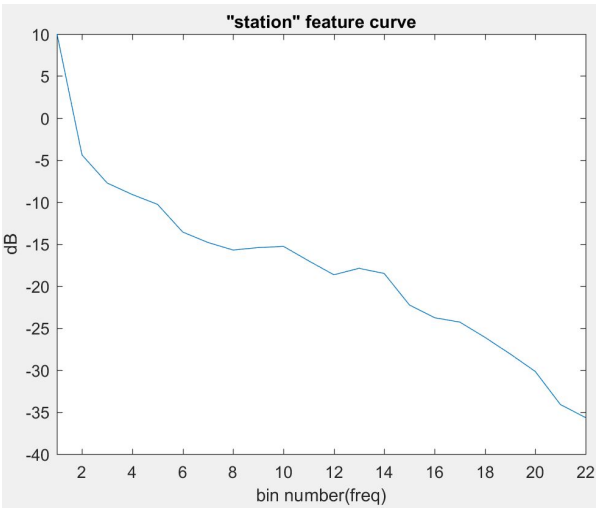

"living" feature curve
dB
bin number(freq)


"meeting" feature curve
dB
bin number(freq)


"office" feature curve
dB
bin number(freq)


"resto" feature curve
dB
bin number(freq)


"square" feature curve
dB
bin number(freq)


"station" feature curve
dB
bin number(freq)

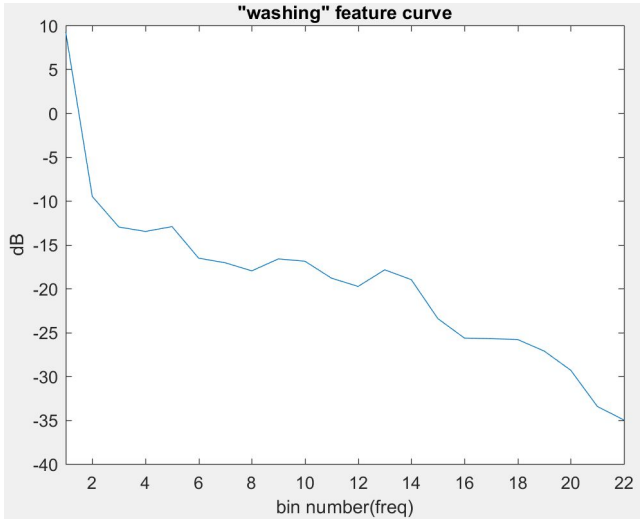

"washing" feature curve
dB
bin number(freq)

Figure 3.5: A direct comparison of all 11 events' averaged feature curves, including events 'cafe', 'cafeteria', 'hallway', 'kitchen', 'living', 'meeting', 'office', 'resto', 'square', 'station', 'washing'. The horizontal axis stands for frequency channels, and the vertical axis is magnitude in dB.

As shown in Figure 3.5, the distinguishable patterns of each event create the opportunity for machine recognition. Later on, all individual feature curves (not averaged feature curve shown above) derived from each trough and from all events will be passed on to neural network for training and detection procedures. Given a piece of 4 minutes mixed audio signal, about 1500 troughs are detected and thus the same number of feature curves are generated. As there are 11 events with 16 files for each, one whole training set contains about 300000 feature curves to be used for neural network training. More details about neural network training and detection are discussed in Chapter 4.

Chapter 4
Event Detection

4.1 Introduction to Feedforward Neural Network Training

Feed-forward neural networks with 1 hidden layer is used in this study. Deep architectures build a hierarchy among the features. In each layer, higher level features are extracted implicitly by the composition of lower level features. This automatic structure eases the work of learning highly nonlinear functions mapping the input to the output directly from data, therefore reducing the need to find human-crafted intermediate representations. Neural networks are composed of an input layer, one layer of 15 hidden neurons with nonlinear activation functions and an output layer, as shown in Figure 4.1. The training function of Levenberg-Marquardt is used. In our experiment, the neural network accepts 22 input values from 22 channels, and outputs 11 values representing the classification confidence for the 11 pre-defined events.

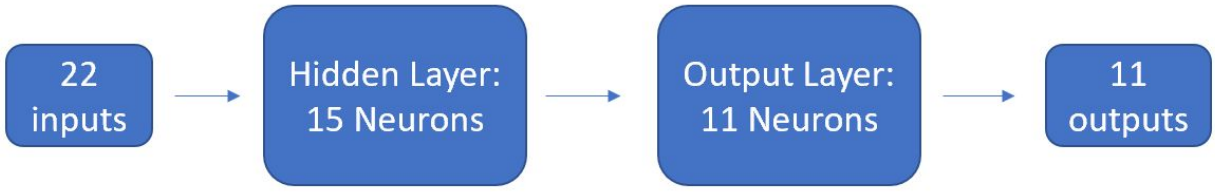


Figure 4.1: The neural network layer structure.

In neural network, each layer consists of a number of neurons. Each neuron forms a web structure, including the elements of weight, bias, transfer function, input and output. The neuron connection is represented with this formula:

$$o = f(i*w+b) \tag{4}$$

where ‘i’ denotes the input to the neuron, ‘o’ denotes the output of the neuron, ‘w’ is weight, ‘b’ is bias, and ‘f’ is the transfer function.

A neuron takes an input either from feature curves or from the output of previous layer. The input is multiplied by a constant value represented by weight. Then this result will be added with a bias. Like weight, a bias is a constant number assigned to this neuron. After that, a transfer function is applied to form the neuron output. The whole procedure is shown in Figure 4.2. There are linear and non-linear types of transfer functions to be chosen to use. In our case, the non-linear Log-Sigmoid transfer function is used.

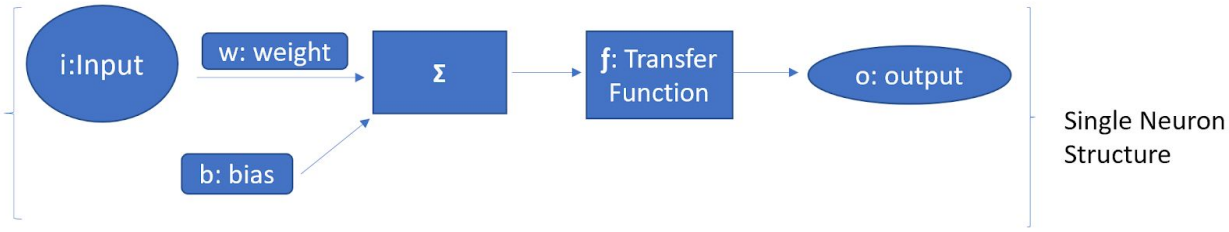


Figure 4.2: Demonstration of the neural network structure used.

After feature extraction, all the feature curves are stored in a matrix for training purpose. This feature matrix has a size of 22 rows and around 300,000 columns, which represents the 22 bins values of around 300,000 feature curves. Another matrix of targeted result is also stored, which has 11 rows and same number of columns. This target matrix provides information about the ideal output. It tells the neural network which feature curve belongs to which event.

During the training procedure, both the feature matrix and target matrix dataset are fed into the neural network. Then the self-learning neural network will run in loops trying out different values for weights and bias for all neurons. Overtime, the assigned values for weights and bias would approach to produce results similar to target matrix. When the neural network model has hit a satisfactory maximum correct hit rate and minimum false alarm rate, it will stop and store the neural network parameters. Those parameters are what would be used for the detection/testing procedure.

4.2 Detection Results and Discussion

This training and detection of sound events with the neural network is simulated digitally on Matlab. A set of audio recordings of 11 events from Creative Commons is used in this experiment. Each event has 16 sound files, and each file contains 5 minutes of background sound. For training, the first 4 minutes of each file is used. And the last 1 minute of each file is reserved for detection testing. During detection procedure, a 5 seconds time frame is sufficient to give a detection classification result. Because 5 seconds of input audio signal could have about 30 troughs extracted for detection analysis. The output from the neural network will give 11 values, representing the confidence level (likelihood) of the 11 events. The classification algorithm will select the event with the largest confident value out of 11 events.

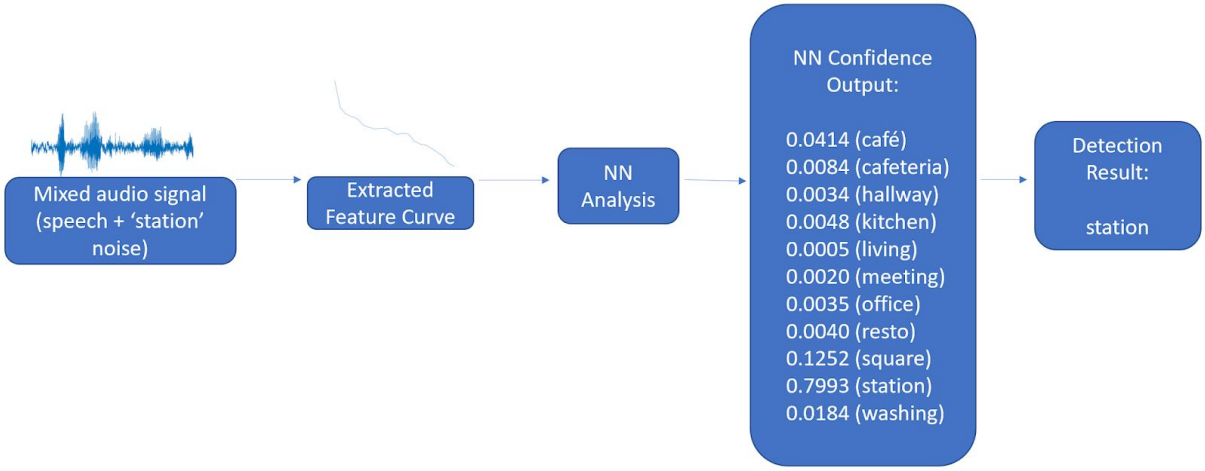


Figure 4.3: Neural network detection process and output confidence levels for event 'station'.

As shown in Figure 4.3, a piece of mixed audio signal is transformed into a feature curve. This feature curve would be fed into a trained neural network to determine the likelihood of each of the 11 events. After the confidence values are produced, the event classification decision would be made by selecting the event with the largest confidence value.

Afterall, a high detection accuracy has been achieved as shown in Table 3 below. The overall average accuracy is 96% for all events. 10 events has over 95% of chance to be correctly recognized, while the 'Hallway' event can only be identified 85% of the time.

Event	Café	Cafeteria	Hallway	Kitchen	Living	Meeting	Office	Resto	Square	Station	Washing
Accuracy	98%	96%	85%	99%	99%	98%	97%	96%	96%	99%	95%

Table 3 : Detection accuracy for 11 events. It shows the percentage rate of being correctly detected for each event.

Detection Averaged Confidence Matrix for all events

Detection Results	café	cafeteria	hallway	kitchen	living	meeting	office	resto	square	station	washing
café	0.9237	0.0003	0.0004	0.0027	0.0009	0	0.0056	0.0031	0.0447	0.0414	0.0206
cafeteria	0.0113	0.6966	0.0019	0.0364	0.0195	0.004	0.0499	0.1187	0.3614	0.0084	0.0762
hallway	0.0035	0.0017	0.6988	0.008	0.0442	0.3241	0.008	0.0047	0.0069	0.0034	0.0035
kitchen	0.0013	0.0588	0.0032	0.9886	0.0047	0.0035	0.0068	0.0368	0.0107	0.0048	0.0071
living	0.0004	0.001	0.1268	0.0079	0.9942	0.109	0.0036	0.0024	0.0009	0.0005	0.001
meeting	0.0002	0.002	0.1772	0.0033	0.0691	0.5676	0.0206	0.0021	0.0111	0.002	0.0041
office	0.0012	0.078	0.0021	0.006	0.0166	0	0.9354	0.0018	0.0718	0.0035	0.1074
resto	0.0083	0.1623	0.0026	0.0103	0.0154	0.0076	0.0035	0.8457	0.0454	0.004	0.0131
square	0.0293	0.0181	0.0031	0.0064	0.0031	0.0043	0.0279	0.0127	0.3692	0.1252	0.2265
station	0.0052	0.0156	0.001	0.0228	0.0015	0.0061	0.0036	0.0172	0.1307	0.7993	0.0177
washing	0.0439	0.0015	0.0003	0.0044	0.0033	0.0011	0.0013	0.0124	0.056	0.0184	0.762

Table 4 : Confidence Matrix shows the overall averaged confidence values for all 11 events. Each column is one event under detection, and the rows are the confidence level of 11 events.

Table 4 lists out the averaged confidence values for all 11 events. The confidence value is the output from neural network ranging from 0 to 1(0% to 100%). After each detection, one confidence level will be generated for each event, representing the likelihood to be recognized as that event. It gives information about the similarity of two events' feature curves from the detection system's perspective. For example, it could be found that 'meeting' is recognized as similar to 'hallway' by observing that 'meeting' has a 0.1772 confidence value to 'hallway', at the same time, 'hallway' has a 0.3241 confidence value to 'meeting'.

Detection Confusion Matrix for all events

Detection Results	café	cafeteria	hallway	kitchen	living	meeting	office	resto	square	station	washing
café	98%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
cafeteria	0%	96%	0%	1%	0%	0%	2%	4%	3%	0%	0%
hallway	0%	0%	85%	0%	0%	2%	0%	0%	0%	0%	0%
kitchen	0%	0%	0%	99%	0%	0%	0%	0%	0%	0%	0%
living	0%	0%	6%	0%	99%	0%	0%	0%	0%	0%	0%
meeting	0%	0%	9%	0%	1%	98%	0%	0%	0%	0%	0%
office	0%	0%	0%	0%	0%	0%	97%	0%	0%	0%	1%
resto	0%	4%	0%	0%	0%	0%	0%	96%	0%	0%	0%
square	0%	0%	0%	0%	0%	0%	1%	0%	96%	1%	4%
station	0%	0%	0%	0%	0%	0%	0%	0%	1%	99%	0%
washing	2%	0%	0%	0%	0%	0%	0%	0%	0%	0%	95%

Table 5: Confusion Matrix shows the overall detection accuracy for all 11 events. Each column is one event under detection, and the rows are the recognition rates in percentage for 11 events.

Table 5 lists out the overall detection accuracy for all 11 events. It provides a general view of the recognition correctness rate for each event. It also indicates which event is misclassified if error occurs. For example, event ‘square’ has a 3% chance to be mistakenly classified as ‘cafeteria’.

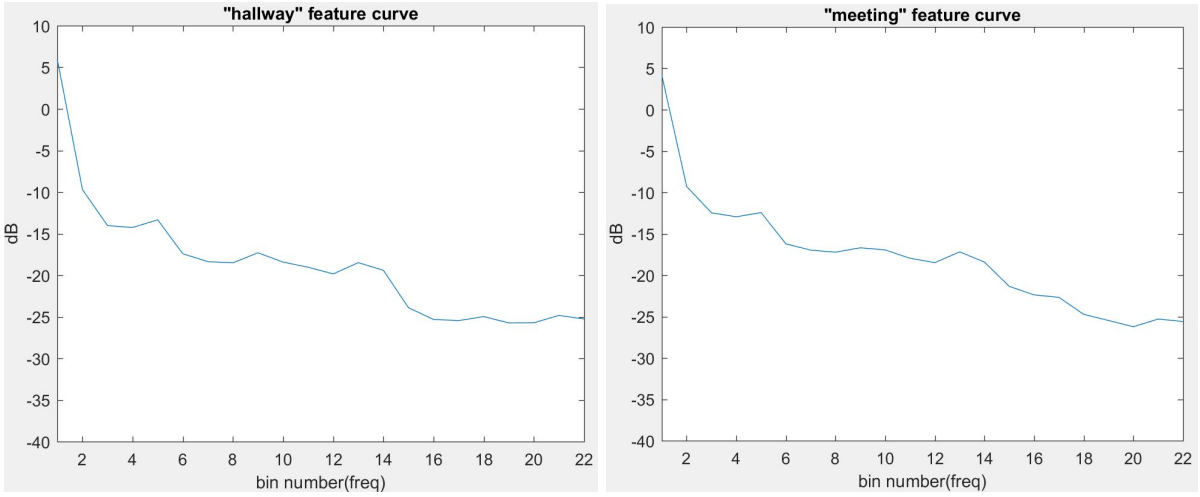
Table 4 and Table 5 are related but not the same. The averaged confidence levels could not be directly translated into accuracy. In other words, if one event has a very low averaged confidence value of itself, it couldn’t imply its detection accuracy is low. Because other events might have even lower confidence values, then this event will always have the highest confidence and be detected correctly. Vice versa, if one event has a high averaged confidence value, it might has a lower accuracy, because there could be another event jumping up with the highest confidence level at times, and this event would be mistakenly recognized as the other event. For example, event ‘square’ has a 0.3692 averaged confidence value to be detected correctly, and a 0.3614 averaged confidence value to be incorrectly classified as ‘cafeteria’. By reading the confidence values, it looks like event ‘square’ will often be recognized as ‘cafeteria’, as they have similar averaged confidence values. However, the confusion matrix shows that ‘square’ has a high 96% accuracy, and only a 3% chance to be wrongly detected as ‘cafeteria’.

The reason is that although the two events have similarly low confidence levels, the confidence values of ‘cafeteria’ is still smaller than that of ‘square’ in most of the time. On the other hand, for event ‘hallway’, the averaged confidence value of being correctly detected is as high as 0.6988, while the second largest averaged confidence value is only 0.1772 from event ‘meeting’. It seems that event ‘hallway’ should obtain a high detection accuracy. But, it only has a low 85% accuracy of being correctly detected, with a 9% chance to be recognized as ‘meeting’. The reason is that Table 4 only lists out averaged confidence levels. If event ‘meeting’ has confidence values with large variance, ‘meeting’ could have larger confidence values quite often when detecting event ‘hallway’. Reasons for obtaining a high detection accuracy are: 1. there are only 11 events for classification. 2. The background noise samples used for training and testing are recorded with the same microphone in the same physical locations.

The main two reasons for misclassification are:

1. Similar feature curves to other events

If two or more events have very similar feature curves, it could be challenging for the neural network to distinguish them. As mentioned before, the neural network output doesn’t perform very well with events ‘hallway’ and ‘meeting’. They are easily mistaken for each other. By observation, their feature curves are found to be very similar, as shown in Figure 4.4. This explains the reason why detection error rate is high.

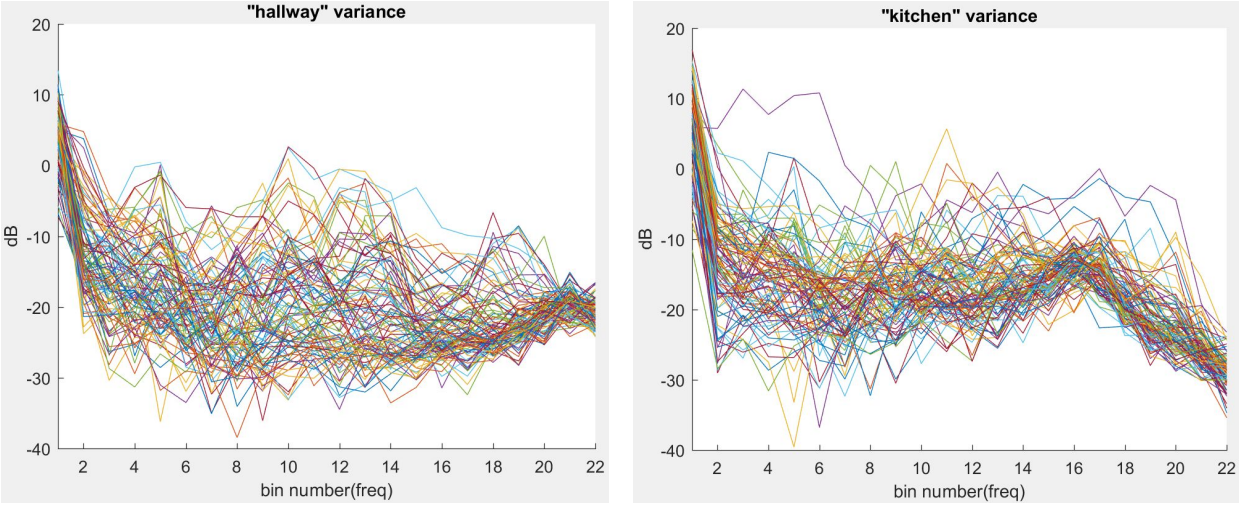


(a) (b)

Figure 4.4: Plot (a) is the feature curve of ‘hallway’. Plot (b) is the feature curve of ‘meeting’. Two feature curves are found to be very similar to each other.

2. Variance of the feature curves

Recall that each feature curve is derived from the samples in the energy troughs. If the data values from each energy trough varies significantly, the shape of each feature curve will also vary a lot. The variance of feature curves from one event will increase the difficulty for neural network to make classification decision. For example, event ‘hallway’ with 85% detection accuracy is compared to ‘kitchen’ with 99% detection accuracy. Figure 4.5 plots the overlay of 1000 feature curves for each event. It’s clearly shown that the variance of ‘hallway’ feature curves is noticeably larger, as it’s shade is wider. On the other hand, the ‘kitchen’ feature curves are more concentrated. An event might have large variance in feature curves if the sound elements changes rapidly overall time. For example, in event ‘hallway’ it is mostly quiet, and suddenly rapid footsteps and sharp door noise could occur in some short time frames. The negative effect caused by feature curve variance could be decreased by using a longer time frame for detection. This way the interruption of some sound elements suddenly jumping in could be significantly reduced.



(a) (b)

Figure 4.5: Plot (a) is the overlay of 100 feature curves of event ‘hallway’. Plot (b) is the overlay of 100 feature curves of event ‘kitchen’. The purpose is to compare the variance of the feature curves from two events. It’s shown that ‘hallway’ feature curves have larger variance.

Chapter 5
Noise Reduction with Gain Patterns

5.1 Event Gain Pattern

Noise reduction is one important real field application to take advantage of the automatic event detection results. With the event identified, there is potential of using its noise pattern in frequency domain to counteract the background noise. Since this thesis paper focuses on the DSP in hearing device, the de-noise algorithms should be minimized to the least calculation to comply with limited computing power. Afterall, a de-noised gain pattern could be easily applied to the frequency domain audio data after WOLA analysis but before WOLA synthesis. This de-noising gain pattern attenuates the frequency channels with noise being dominate. This way, the noise will be attenuated.

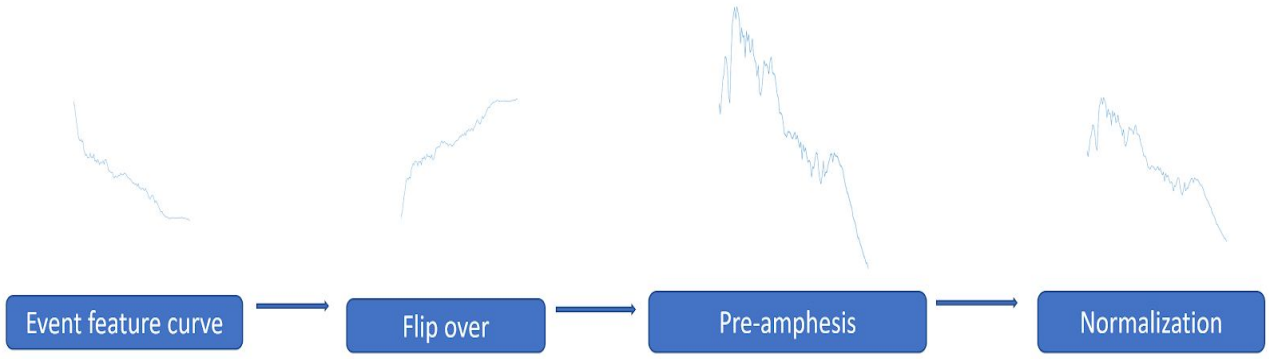
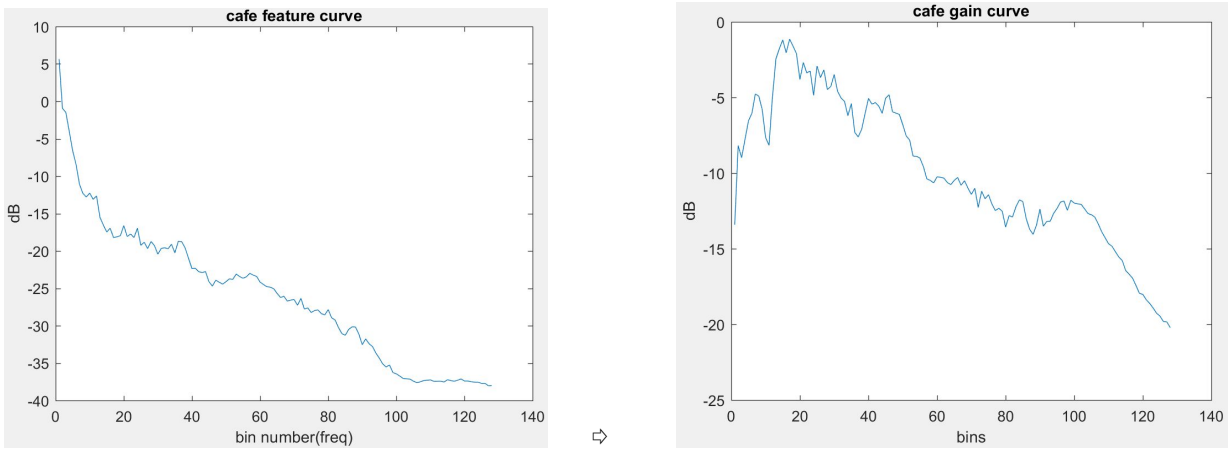


Figure 5.1: Process of obtaining event gain curve to attenuate background noise.

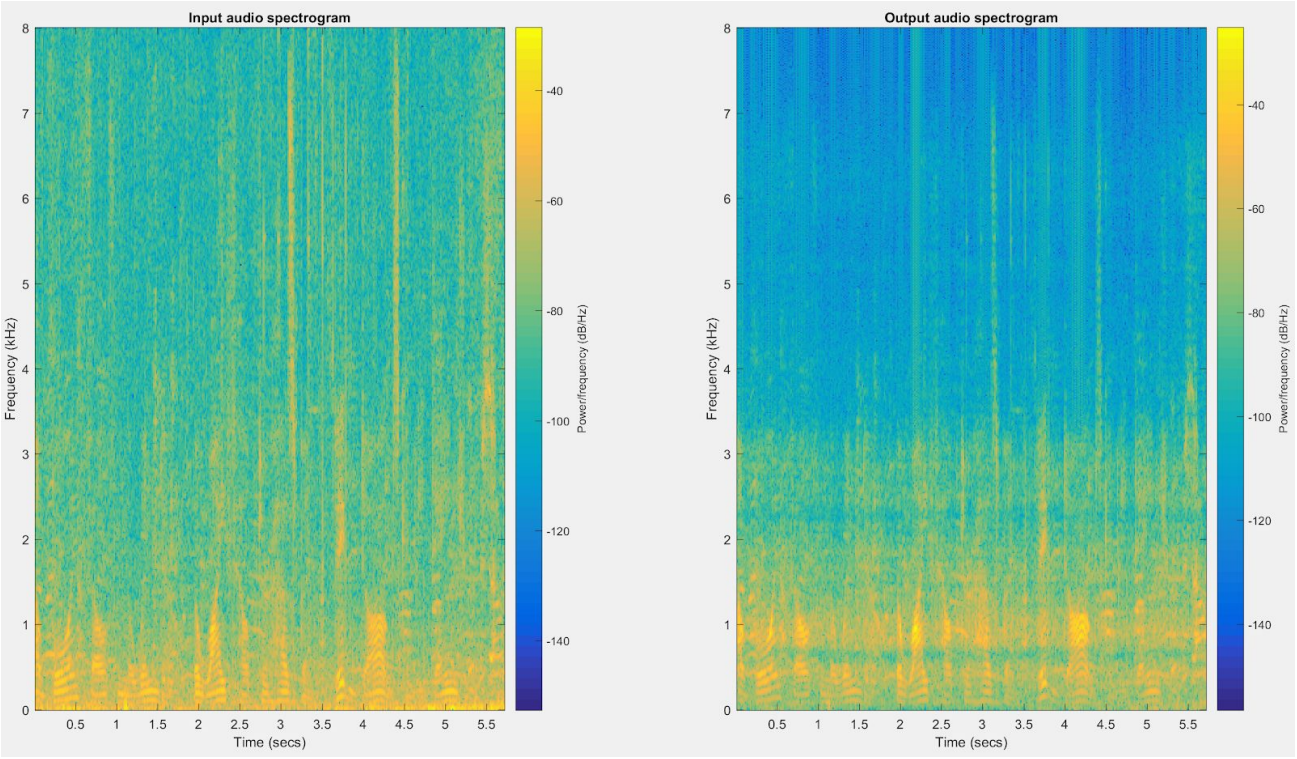
The gain patterns for each event could be derived from the events' feature curve patterns, as shown in Figure 5.2. The gain curves are the flip-over of the feature curve. And then normalization and pre-emphasis are applied to generate a meaningful gain curve with the attenuation range of 0 dB to -30 dB, as in Figure 5.1.



(a) (b)

Figure 5.2: Plot (a) is the 'cafe' feature curve. Plot (b) is the cafe gain curve used for de-noising, which is derived from plot (a)'s feature curve. The gain curve is the flip-over of feature curve, with additional processes of normalization and pre-emphasis.

During testing, gain curves are applied to human speech audio mixed with noisy background at SNR = 0 dB and +5 dB. Usually hearing aid patients aim to achieve a certain speech perception rate at SNR = +5dB. But since the evaluation of this noise reduction system is done by normal hearing people, lower SNR at 0 dB is used for better comparison. With the correct event detection result, each event background noise is processed by its corresponding event gain curve. The noise reduction performance is evaluated preliminarily by listening to the audio before/after de-noising, and visually comparing the spectrogram of input/output audio signals. The result shows that after applying the gain curve, the output audio sounds noticeably less noisy, with the speech more popped out. As for the spectrograms, the goal is to have lower energy(blue or cooler color) in the noisy areas, while the speech areas remain their energy strength(yellow or hotter color). By observation, the de-noised output spectrogram has eliminated the noise energy in the high frequency area and the very low frequency area (<100 Hz). While the mid-range frequency area (0.1 ~ 3.5 kHz) of speech signal has been preserved and only slightly modified.



(a) (b)

Figure 5.3: Spectrograms of a piece of audio combining human speech and background ‘cafe’ noise with SNR = 0. Image (a) is the input audio spectrogram of the unprocessed sound. Image (b) is output audio after applying the de-noising gain curve. Image (b) shows a clearer image with less noise while the speech details are preserved in the low and mid frequency range.

Although the event gain curve proposed has been proven to be effective in reducing the audio background noise, there is limitation to them. Gain curves are derived from environment noise feature curves, which could be seen as the overall characteristic representation of the event noises. In order words, these gain curves won’t react to sudden or periodic changes in background noises. It is most effective for elimination of stationary noise with constant volume level and similar characteristics.

5.2 Combined Usage of Event Gain Pattern with Instant Trough Gain Pattern

Another method to generate de-noise gain pattern is using the instant trough background noise data[23]. Recall that trough detection method was used to extract background noise-only data. In this approach, the last four troughs would be extracted and fed into a FIFO (First In First Out) to generate an averaged smooth feature curve (see Figure 5.5). Then this smooth feature curve would be applied to processes of flip-over, normalization, and pre-emphasize to derive the instant gain pattern for this moment, as shown in Figure 5.4.

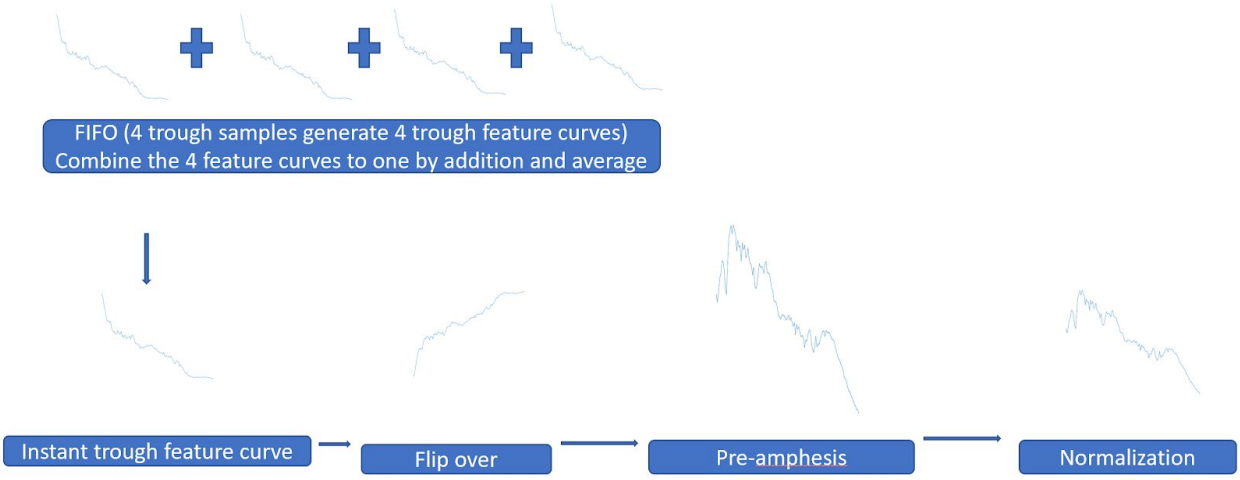


Figure 5.4: workflow of how the instant trough gain pattern is derived.

This instant gain pattern is updated every time a new trough is detected. By averaging the four trough samples, the instant gain curve is smoothed and become more accurate in matching the changing noise environment. The average process also prevents the error caused by a corrupted trough noise sample.

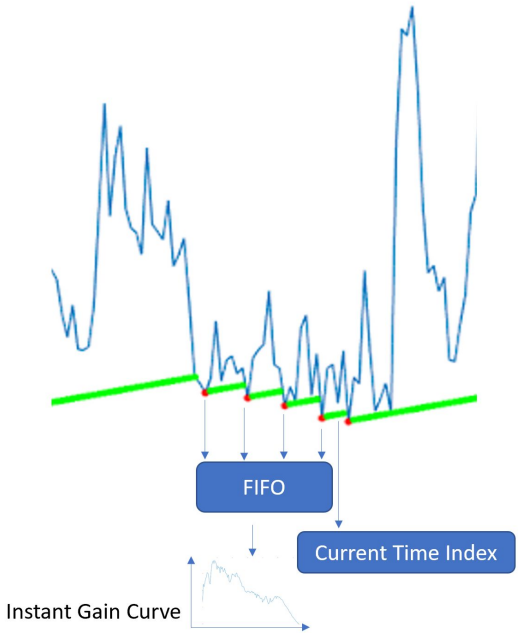
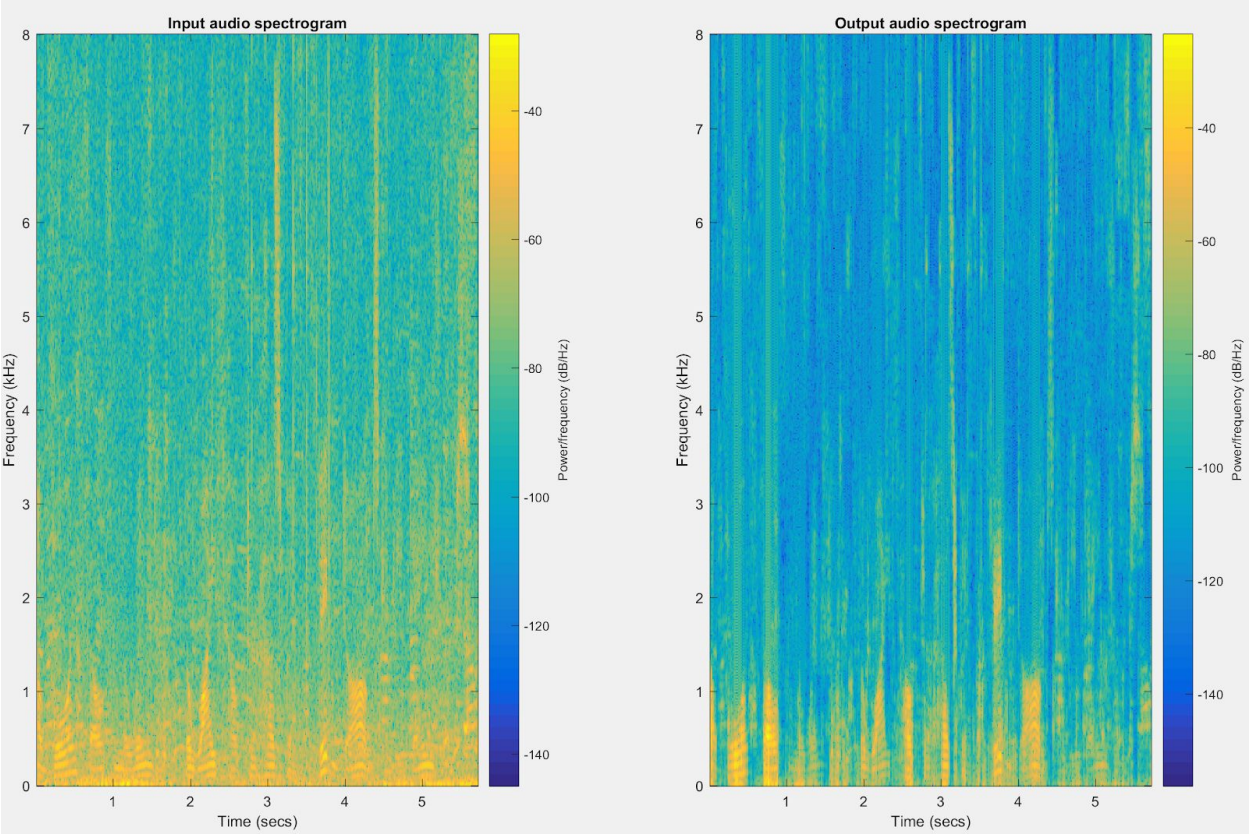


Figure 5.5: A plot showing how the last four trough samples are selected to generate the instant gain curve.

The instant gain curve is adaptive to the dynamically changing background noise, so the noise could be deeply suppressed. It is especially efficient to counteract abrupt noise, such as the bird chirping sounds in the 'field' background noise files, which only occurs randomly and lasts for merely one or two seconds.

However, the instant gain curve also comes with limitations. The instant gain doesn't update itself if no new trough is detected. This will lead to an instant gain pattern being used for a longer period of time and couldn't catch up with the new noise characteristic. When this happens, the instant gain curve whether has too little attenuation on the noise or mistakenly attenuates human speech.



(a) (b)

Figure 5.6: Spectrograms of a piece of audio combining human speech and background ‘cafe’ noise with SNR = 0 dB. Image (a) is the input audio spectrogram without denoising modification, and image (b) is output audio after applying the instant gain curve. Image (b) shows a clearer image with less noise(more blue areas) while the speech details are reserved. But there are gaps in the speech signal in low and mid-range frequency.

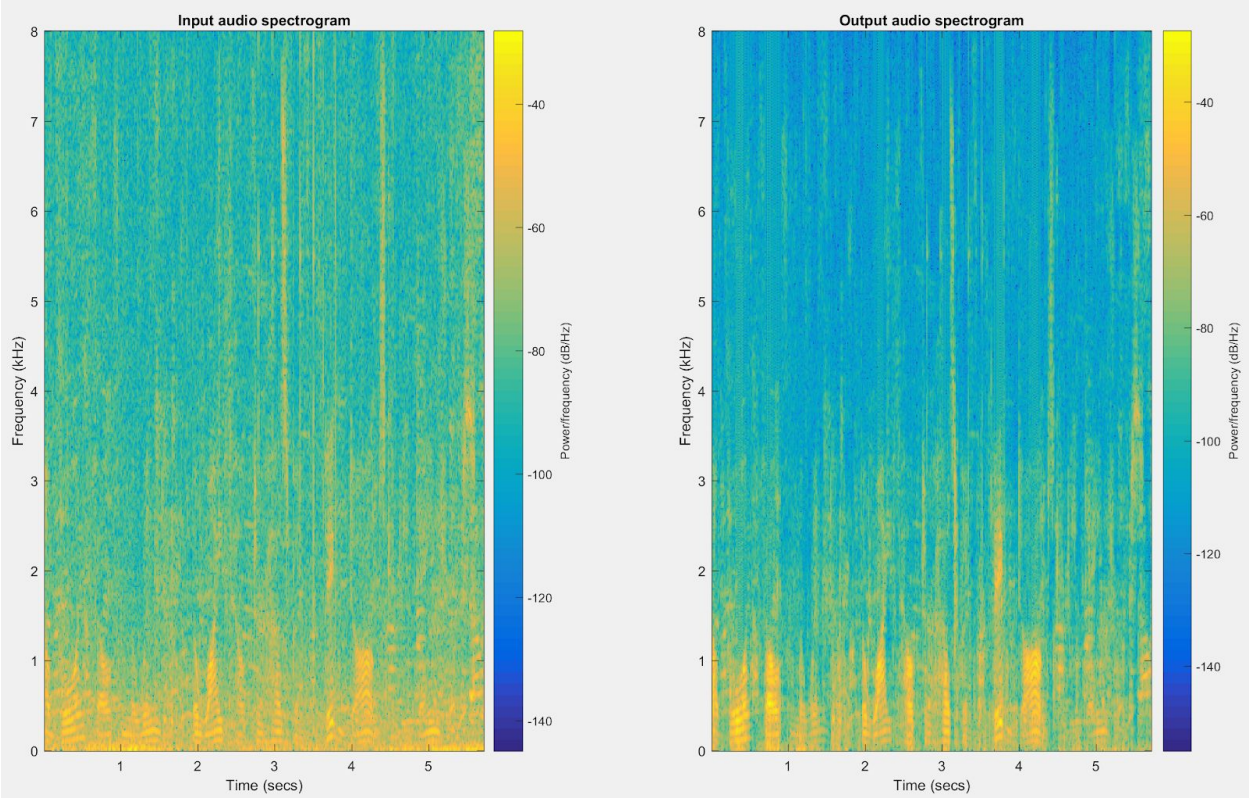
Through preliminary testing, it is found that the output audio’s speech volume goes up and down from time to time. This is caused by the abrupt changes of the instant gain curve. Also, the speech doesn’t sound smooth. Some audio information between two words are missing. It is difficult to hear a whole word’s pronunciation since the first and last syllables are usually cut off.

From observation of the spectrogram, it’s shown that the random noise has been effectively reduced, leaving large portions of blue areas. At the same time, there are wide gaps

between the yellow shades in the speech frequency (0 ~ 1 kHz). This explains the discontinuity of the speech during testing.

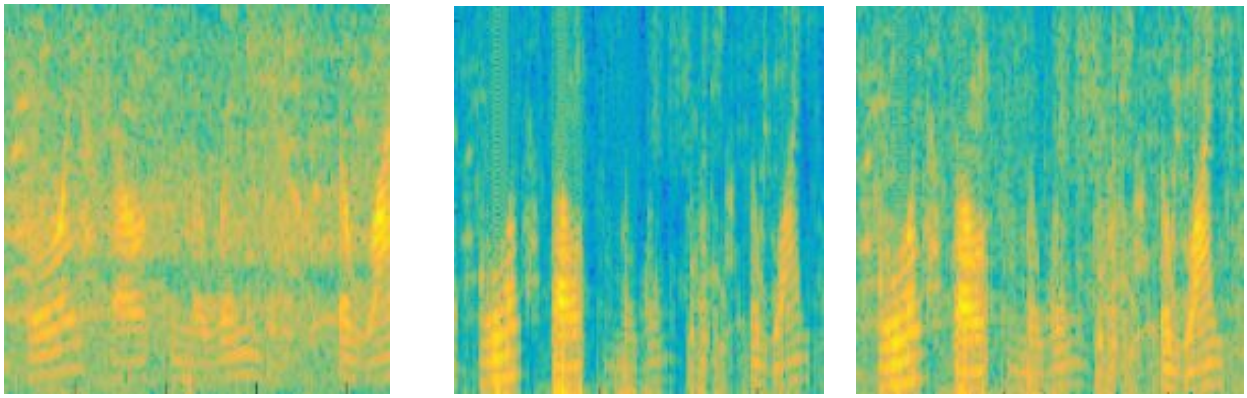
To overcome the limitations of both the averaged gain pattern and instant gain pattern, a combination of them with an adjustable weighting ratio is tested in the experiment. For example, the combined gain curve is the sum of averaged gain pattern and instant gain pattern curves. If the ratio is 1:1, we have ‘combined gain pattern’ = 0.5*averaged gain pattern + (1-0.5)*instant gain pattern.

The test results have proven that the combined gain pattern generates output audio with less noise over the course and smoother speech transitions between syllables. By observation of its spectrogram in Figure 5.7, it’s shown that output audio has improved noise reduction ability (larger blue area) compared to Figure 5.3 (b), and meanwhile it has smoother speech transitions (less gaps in low frequency yellow area) compared to Figure 5.6 (b). Therefore the combined gain pattern can potentially provide improved de-noising results, as shown in Figure 5.8.



(a) (b)

Figure 5.7: Spectrograms of a piece of audio combining human speech and background ‘cafe’ noise with SNR = 0. Image (a) is the input audio spectrogram without denoising modification, and image (b) is output audio after applying the de-noise instant gain curve. Image (b) shows that the usage of combined gain pattern has obtained improved noise reduction and speech smoothness.



(a) (b) (c)

Figure 5.8: These three images are zoom-in comparison between three types of gain patterns applied. Image (a) is the spectrogram result of using the averaged gain pattern. Image (b) is the spectrogram result of using the instant gain pattern. Image (c) is the spectrogram result of using the combined gain pattern. Image (a) preserves and enhances all the details in human speech. But it has horizontal spectral gaps, since its gain pattern remains the same over time. And it's less effective in reducing noise which doesn't match the averaged gain pattern. Image (b) could greatly reduce noise. It applies adaptive instant gain pattern from the background noise in real time. But it creates temporal (vertical) speech gaps, which causes the speech stuttering. Image (c) combines the detection results of the two gain patterns. It is more efficient in noise reduction, while retaining the smoothness of human speech.

Chapter 6

Conclusions and Future Work

In this study, a novel approach for sound event detection in realistic environments using WOLA and a feedforward neural network was proposed, as shown in Figure 6.1. The trough detection algorithm has been proven to be able to accurately extract background sections. WOLA is capable of transforming those background time domain sections into frequency spectrum (feature) curves for gain application. Then these feature curves are used as training data in neural network. After the neural network is built, any mixed audio signals can be fed into neural network for event classification. Moreover, this model contributes to background noise reduction by attenuating the targeted background noise frequency regions. A gain curve is derived by flipping the event feature curve. This gain curve is later applied back to the mixed audio signal in complex domain for noise reduction purpose. WOLA synthesis is used to transform the frequency domain data back to continuous time domain audio. Preliminary evaluations were carried out for a noise reduction efficiency test, and the noise suppression performance is discussed.

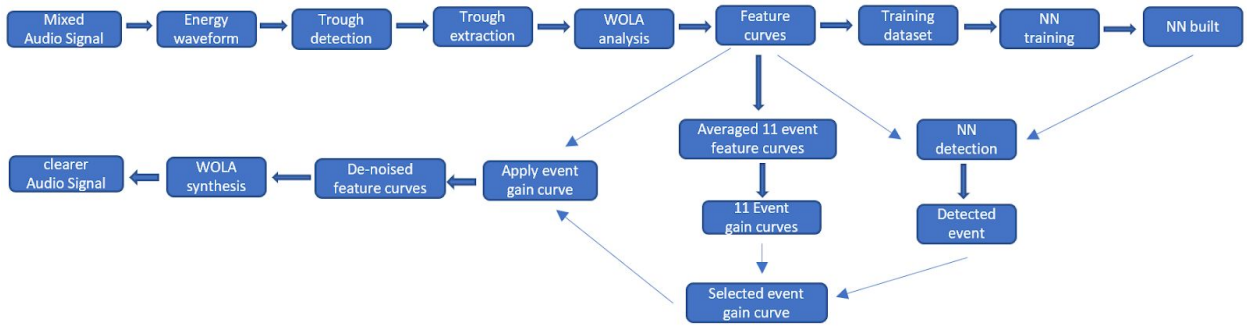


Figure 6.1: Block diagram of the intelligent event detection and noise reduction systems of this thesis.

For future work, more training data could be used to further understand the full capability of this system. In this thesis, 11 events with a total of 704 minutes were used for training, and 176 minutes of audio signal were used for testing. If needed, this system could be

trained and applied to many more other types of scenes for detection and noise reduction purposes. Also, the audio dataset used in this experiment is a set of recordings from the same microphone. And the recording environments are mostly stational, like the event ‘cafe’ is recorded in only one position in one cafe. If this system is to be migrated into some other platform, it would be necessary to train with the real life audio data for optimal results. For example, if the system is to be implemented to a certain model of hearing aid device, it would need to obtain training data from the real life recordings from that hearing device’s microphone. Moreover, if the event ‘bus’ is to be trained for detection, it is necessary to record sounds from different kinds of buses, and different times of travel. Because different types of bus engines, or road conditions could lead to varying sound signatures. It is desirable to include more types of sounds for ‘bus’ event to acquire the better detection result.

As this thesis was planned with the idea to be implemented into other hardware platforms, all the parameters are chosen to be easily compatible to hearing devices. The system is built up to be less computing power demanding, lighter, more efficient, and transformable. For example, during analysis the 128 bins are reduced to 22 channels so as to reduce the computational load. And the WOLA method is used to keep it real time efficient. However, the whole system is simulated on PC, but hasn’t been tested out in an actual hearing device implementation. The training process of the neural network requires a significant amount of storage space and computing power, so it has to be done on a PC. But the post-trained neural network could be migrated into hearing device for detection purpose. Future work could be carried on to test this system on an actual low powered, mobile hearing device. Hearing device could benefit from this system by automatically adjusting its fitting parameters according to the neural network detection results. For example, a hearing device could apply the corresponding de-noise event gain pattern after its reading of current background noise event.

Furthermore, this system could be applied to not only noise detection, but also smart sound recognition, such as male/female voice separation, or simple voice commands. Just like environmental noise, human speech of short key words could also be recognized. Thus, hearing devices could complete some simple tasks by listening to user voice input. For example, a hearing device could have a ‘volume up’ phrase trained and loaded into the system. And it can

also be potentially used to control the AGC (Automatic Gain Control) circuit with the detection result. Then a user could turn up the device output volume by saying ‘volume up’ to the microphone.

Afterall, the thesis has contributed to building up a virtual prototype intelligent sound event detection system for hearing device. This system has the ability to classify background sound environment automatically. And it has been proven effective in improving noise reduction performance for pre-defined noise events. In the future, there is potential to add more smart sound processing features to hearing devices.

Bibliography

[1] "Deafness and hearing loss Fact sheet N°300". March 2015

[2] Yoshinaga-Itano, Christine, et al. "Language of early-and later-identified children with hearing loss." *Pediatrics-English Edition* 102.5 (1998): 1161-1171.

[3] "1.1 billion people at risk of hearing loss WHO highlights serious threat posed by exposure to recreational noise" (PDF). *who.int*. 27 February 2015.

[4] Global Burden of Disease Study 2013 Collaborators (August 2015). "Global, regional, and national incidence, prevalence, and years lived with disability for 301 acute and chronic diseases and injuries in 188 countries, 1990-2013: a systematic analysis for the Global Burden of Disease Study 2013". *Lancet*. **386** (9995): 743–800. doi:10.1016/s0140-6736(15)60692-4.

[5] Oishi N, Schacht J (June 2011). "Emerging treatments for noise-induced hearing loss". *Expert Opinion on Emerging Drugs*. **16** (2): 235–45. doi:10.1517/14728214.2011.552427

[6] "Noise-Induced Hearing Loss: Promoting Hearing Health Among Youth". *CDC Healthy Youth!*. CDC. 2009-07-01.

[7] World Health Organization Deafness and hearing impairment. Fact sheet No. 300. 2006

[8] Haynes DS, Young JA, Wanna GB, Glasscock ME 3rd. Middle ear implantable hearing devices: an overview. *Trends Amplif*. 2009;13(3):206–214. doi:10.1177/1084713809346262

[9] Edwards, B. (2007). The Future of Hearing Aid Technology. *Trends in Amplification*, *11*(1), 31–45. https://doi.org/10.1177/1084713806298004

[10] Murray, Daniel J. C. and John V. Hanson. “Application of digital signal processing to hearing aids: a critical survey.” *Journal of the American Academy of Audiology* 3 2 (1992): 145-52.

[11] Strom KE. The HR 2006 Dispenser Survey. Hear Rev. 2006;13:16-39.

[12] W. N. Mwema and E. Mwangi, "A spectral subtraction method for noise reduction in speech signals," *Proceedings of IEEE. AFRICON '96*, Stellenbosch, South Africa, 1996, pp. 382-385 vol.1.
doi: 10.1109/AFRCON.1996.563142

[13] Yang Lu, Philipos C. Loizou, A geometric approach to spectral subtraction, Speech Communication, Volume 50, Issue 6, 2008, Pages 453-466, ISSN 0167-6393,

[14] Verteletskaya, Ekaterina & Simak, B. (2019). Enhanced spectral subtraction method for noise reduction with minimal speech distortion.

[15] M. Marzinzik and B. Kollmeier, "Speech pause detection for noise spectrum estimation by tracking power envelope dynamics," in *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 2, pp. 109-118, Feb. 2002. doi: 10.1109/89.985548

[16] Brennan, R & Schneider, T & Zhang, W. (2012). Ultra Low-Power Noise Reduction Strategies Using a Configurable Weighted Overlap-Add Coprocessor.

[17] Anil Chokkarapu, Sarath C. Uppalapati and Abhiram Chintakuntla. (2013). Implementation of Spectral Subtraction Noise Suppressor Using DSP Processor

[18] M. Mohamadian, E. P. Nowicki, A. Chu, F. Ashrafzadeh and J. C. Salmon, "DSP implementation of an artificial neural network for induction motor control," *CCECE '97.*

Canadian Conference on Electrical and Computer Engineering. Engineering Innovation: Voyage of Discovery. Conference Proceedings, Saint Johns, Newfoundland, Canada, 1997, pp. 435-437 vol.2. doi: 10.1109/CCECE.1997.608251

[19] Behan, Thomas & Liao, Zaiyi & Zhao, Lian & Yang, Chunting. (2008). 1 Accelerating Integer Neural Networks On Low Cost DSPs. International Journal of Intelligent Systems - IJIS.

[20] Jung, Seul & su Kim, Sung. (2007). Hardware Implementation of a Real-Time Neural Network Controller With a DSP and an FPGA for Nonlinear Systems. Industrial Electronics, IEEE Transactions on. 54. 265 - 271. 10.1109/TIE.2006.888791.

[21] Mats Forssell. Hardware Implementation of Artificial Neural Networks.

[22] T Heittola, A. Mesaros, T Virtanen, and M. Gabbouj, "Supervised model training for overlapping sound events based on unsupervised source separation," in Int. Corif. Acoustics, Speech, and Signal Processing (ICASSP), Vancouver, Canada, 2013, pp. 8677-8681.

[23] K. Nie *et al*., "A WOLA-Based Real-Time Noise Reduction Algorithm to Improve Speech Perception with Cochlear Implants," *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, Honolulu, HI, USA, 2018, pp. 951-954. doi: 10.23919/APSIPA.2018.8659579

[24] R. Crochiere, "A weighted overlap-add method of short-time Fourier analysis/Synthesis," in *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 1, pp. 99-102, February 1980. doi: 10.1109/TASSP.1980.1163353

[25] R.E. Crochiere and L. R. Rabiner, Multirate digital signal processing, Prentice-Hall, 1983.

[26] R. Brennan and T. Schneider, “A flexible filterbank structure for extensive signal manipulations in digital hearing aids”, Proc. IEEE Int. Symp. Circuits and Systems, pp.569–572, 1998.

[27] Ahadi, S.M., Sheykhzadeh, H., Brennan, R.L., & Freeman, G.H. (2004). A Weighted Overlap Add-based Front-end for Speech Recognition.

[28] R. L. Brennan, R. Abutalebi and H. Sheikhzadeh, "Adaptive filtering using a highly oversampled weighted overlap-add filterbank in an ultra low-power system," *Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers, 2002.*, Pacific Grove, CA, USA, 2002, pp. 806-810 vol.1. doi: 10.1109/ACSSC.2002.1197290

[29] R. Nissel, S. Schwarz, and M. Rupp, “Filter bank multicarrier modulation schemes for future mobile communications,” IEEE Journal on Selected Areas in Communications, vol. 35, issue 8, pp. 1768-1782, 2017

[30]J. H. Park and W. C. Lee, "An Efficient WOLA Structured OQAM-FBMC Transceiver," *2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN)*, Prague, 2018, pp. 782-784. doi: 10.1109/ICUFN.2018.8437042

[31] G. Schuster and R. Ansorge, "WOLA noise cancelling performance," *2008 16th European Signal Processing Conference*, Lausanne, 2008, pp. 1-5.

[32] Elaina Chai. (2014). Implementation of Deep Convolutional Neural Net on a Digital Signal Processor.

[33] F. Frey, R. Elschner, C. Kottke, C. Schubert and J. K. Fischer, "Efficient real-time implementation of a channelizer filter with a weighted overlap-add approach," 2014 The

European Conference on Optical Communication (ECOC), Cannes, 2014, pp. 1-3. doi: 10.1109/ECOC.2014.6964101

[34] H. Sheikhzadeh, R. L. Brennan, Z. Khan and K. R. L. Whyte, "Low-resource delayless subband adaptive filter using weighted overlap-add," *2005 13th European Signal Processing Conference*, Antalya, 2005, pp. 1-4.

[35] R. Vicen-Bueno, A. Martinez-Leira, R. Gil-Pita and M. Rosa-Zurera, "Modified LMS-Based Feedback-Reduction Subsystems in Digital Hearing Aids Based on WOLA Filter Bank," in *IEEE Transactions on Instrumentation and Measurement*, vol. 58, no. 9, pp. 3177-3190, Sept. 2009.
doi: 10.1109/TIM.2009.2017150

[36] G. Schuster and R. Ansorge, "WOLA noise cancelling performance," *2008 16th European Signal Processing Conference*, Lausanne, 2008, pp. 1-5.