An Investigation of a Wavelet-Based Approach for Automated Restoration of Vintage Audio

Matthew Wilde

22135229

BSc (Hons) Sound Engineering and Production

April 28, 2025

Supervisor: Dr Samuel Smith

Abstract

The restoration of vintage audio recordings is critical in the modern day for preserving cultural heritage, but remains a mainly manual and skilled task. This study investigates the feasibility of an automated restoration system based on wavelet analysis, capable of reducing unwanted vintage artefacts while maintaining the fidelity of the original recording. A MATLAB based prototype was developed, implementing noise reduction using discrete wavelet transforms (DWT) with a range of parameters, alongside three distinct transient detection and removal methods: local adaptive thresholding, modulus maxima verification, and time-domain interpolation.

An evaluation was completed, combining subjective listening tests and objective metrics, looking at both signal fidelity and noise reduction. The results identified symlet wavelets with adaptive or level-dependent thresholds as the most effective for preserving fidelity while reducing artefacts. Time-domain derivative interpolation emerged as the preferred method for crackle reduction. The system was especially effective on authentic vintage recordings and outperformed on them compared to artificial white noise.

This research validates the potential of wavelet-based processing for audio restoration and lays the groundwork for a real-time plugin implementation.

Acknowledgements

Firstly, I would like to thank my supervisor, Dr. Samuel Smith, for his guidance throughout this project.
I would also like to thank the listening test participants for their time and valuable feedback.

Contents

Word Count = 12000

List of Figures

List of Tables

Glossary

Wavelet A waveform with limited duration that is used to analyse time-varying frequency content in signals.

CWT (Continuous Wavelet Transform) A wavelet transform where the scale and translation are continuously varied, offering a detailed time-frequency representation.

DWT (Discrete Wavelet Transform) A computationally efficient wavelet method using downsampling.

IDWT (Inverse Discrete Wavelet Transform) The process of reconstructing the original signal from wavelet coefficients.

Modulus Maxima A technique used to detect signal peaks in wavelet coefficients.

Transient Artefact A short-duration, high-energy imperfection in audio, often caused by physical damage or digitisation errors.

SNR (Signal-to-Noise Ratio) An objective metric that compares the level of the desired signal to the level of background noise.

LSD (Log-Spectral Distance) A spectral distortion measure used to compare differences between the processed and original signal's frequency content.

ISD (Itakura–Saito Distance) A perceptual metric for comparing power spectra, sensitive to changes in spectral shape.

MCD (Mel Cepstral Distortion) A distance measure between MFCCs of two signals.

Chapter 1

Introduction

1.1 Problem Definition

The preservation and restoration of vintage audio recordings presents a significant challenge within audio engineering and archival practices. These recordings often contain undesirable artefacts such as crackles, pops, hums, and other noise components introduced by the limitations of historic recording technology and storage methods. These artefacts may reduce audio clarity and hinder the accurate preservation of culturally significant materials. Traditional restoration methods require substantial manual effort and specialised expertise, making large-scale preservation attempts costly and time-consuming. Due to this, an automated and efficient approach to this issue that preserves traditional vintage characteristics is desirable. This project seeks to address these issues by developing and evaluating an automated approach to digital restoration.

1.2 Scope

This project is focused on the development and evaluation of an audio restoration method, specifically aimed at vintage audio recordings. The project includes research of different audio analysis methods, identifying the strengths and weaknesses of different techniques,

which is most applicable and how it may be utilised.
The project does not encompass development of the automated tool itself, instead focusing on proof-of-concept, with the algorithm development undergoing robust evaluations and analysis.

1.3 Rationale

The importance of digital audio restoration development derives from the increasing importance of digital preservation and the need for more accessible restoration tools. Currently existing solutions range from sophisticated professional software, such as iZotope RX and Adobe Audition, to freely available yet less powerful options such as audacity, however many of these require a technical-proficiency or manual tuning to achieve effective results, often being inaccessible to archivists or casual audio enthusiasts.

The significance of the need for accessible tools is reinforced by the shift in contemporary audio production towards accessibility and ease of use, demonstrated by the ever-increasing number of bedroom producers and amateur enthusiasts. The intent is to bridge the gap between high-quality restoration tools and user accessibility, evaluating how an automated system may be developed to be used with minimal specialist knowledge.

1.4 Project Aims and Objectives

1.4.1 Project Aim

The aim of this project is to investigate, evaluate and propose an automated system for effective noise reduction and artefact removal in vintage audio recording whilst preserving the integrity of the original vintage recording.

1.4.2 Objectives

- Research current literature to identify existing noise reduction systems and issues commonly found with vintage audio.
- Develop audio processing systems and create a prototype denoising algorithm to address a range of artefacts with a range of configurations to explore optimising the systems.
- Quantify the performance of the prototype algorithms using comprehensive objective measurements, looking at how the audio is being affected and any compromises the system may be making.
- Organise and conduct subjective listening tests to compare against and further interpret objective measurements, determining user preferences and perceptual qualities.
- Analyse all test results to provide insights into the optimal restoration methods, and how these may further be used in the creation of an automated tool.

Chapter 2

Literature Review

2.1 Literature Search Methodology

The main method that will be used in the literature search will be searching through databases and libraries for relevant research papers and textbooks. This will include digital libraries such as Google Scholar, the Birmingham City University Library, IEEE Xplore and JSTOR. A range of keywords and search terms to be used to find appropriate sources and research can be seen in the following table

2.2 Themes

A thematic approach has been taken to identify the areas to explore to support the development stages of the project. Researching these topics should ensure a comprehensive understanding of the required subject areas. Keywords associated with each topic will be used to gather relevant information papers and research. Six main themes have been detailed below.

Themes	Keyword(s)	Reasoning
Vintage Audio	• “Vintage Audio Characteristics”	To gain a deeper understanding of the intrinsic qualities that make vintage audio be discernible from other sound.
	• “Magnetic Tape degradation” • “Vinyl degradation”	To explore the different ways that analogue files degrade over time, seeing how different artefacts are formed.
	• “Analogue file restoration” • “Vintage Audio Restoration”	To investigate methods of physical restoration, seeing if there are methods that make noise reduction redundant.
Historical Developments	• “Analogue noise reduction” • “Analogue signal processing”	To provide insight into older analogue reduction techniques, allowing for perspective on the evolution of the area.
	• “Artefact detection” • “Signal analysis” • “Audio analysis algorithms”	To provide research into more modern digital analysis systems, exploring different analysis methodologies currently being utilised.
	• “Fourier transform” • “Adaptive filtering” • “Wavelet analysis”	To research into specific audio analysis and reduction methods.
Plugin Development	• “MATLAB toolboxes” • “Python libraries” • “Signal processing prototype software”	This will allow for the best choice on what software will be used for the development of the prototype system.
	• “Plugin frameworks” • “Python plugin development” • “C++ Plugin development” • “VST3 Plugin development”	To explore different plugin methods and evaluate available plugin frameworks to find the most suitable development system for the project.
Preservation Standards	• “Archive standards” • “Audio metadata” • “Digital archiving”	To identify standards that the system may be required to meet to be an effective tool for archiving and preservation.
	• “Audio restoration ethical considerations”	To discuss the ethical considerations behind audio restoration in consideration to altering the original audio.
Listening Tests	• “Subjective Listening Tests” • “Paired comparison tests” • “Double blind tests”	To explore different methods that subjective listening tests may be carried out to evaluate the effectiveness of the system.
	• “Objective audio quality metrics” • “Objective listening tests”	To consider the scope of objective measurements that can be taken and how these can be incorporated with subjective feedback.

Table 2.1: Literature Review Search Terms

1. **Vintage Audio**

   This theme explores the defining characteristics of vintage audio, focusing on the artefacts that contribute to the unique vintage sound, such as crackles, pops, hisses and more. Research will be done into why and where these artefacts appear, and potential methods used to manage and restore them.

   Keywords – Vintage Audio Restoration, Vintage Audio Characteristics

2. **Historical Development of Noise Analysis & Reduction**

   The review will delve into the historical context of noise reduction, looking at the original analogue methods used up to more modern digital methods, such as wavelet analysis and adaptive filters. This will allow to see a broad scope of different methods for noise reduction. The strengths and weaknesses of current methods will be discussed, evaluating how they may be used within the project, and which is the most applicable.

   Keywords – Artefact Detection, Audio Analysis Algorithms, Fourier Transform, Wavelet Analysis, Noise Reduction

3. **Analysis & Reduction Techniques**

   Following on from the prior theme, this section takes the analysis and reduction techniques chosen and expands on the research into them. First looking at how the mathematics behind the techniques work to gain a fundamental understanding of how they can be stylised for vintage audio. The implementation will then be analysed, exploring how it may be executed in the prototype MATLAB system whilst specifically tailored to the artefacts identified in the vintage audio section.

4. **Plugin Development**

   This theme investigates the transition from a prototype system into a functional system. Considerations will include which processes should be fully automated, and whether they would operate in real-time or perform full analysis. Different development methods will be explored, such as JUCE, along with programming languages like C++ and Python.

   Keywords – VST Plugin Development, JUCE, C++ Plugin Development, Python Plugin Development

5. **Preservation Standards**

   As the project's aim is to assist in the preservation of vintage audio, research will be conducted into the standards the system will need to meet. This includes

archival standards such as audio quality and necessary metadata. Ethical aspects of restoration will also be explored-such as whether restoration risks altering the original intention of the recording, or whether that responsibility lies with the user.
Keywords – Archive Standards, Audio Metadata

6. **Listening Tests**
   A range of methods for listening tests will be examined, including both subjective and objective approaches. Objective measurements will be used to evaluate the system's performance and the changes it introduces. Subjective measurements will help fine-tune the system and determine how successfully it achieves the desired vintage audio qualities.
   Keywords – Subjective/Objective Listening Tests, Audio Quality Metrics

2.3 Review of Literature

2.3.1 Vintage Audio

Vintage audio commonly refers to audio stored on analogue mediums such as vinyl and magnetic tape. Vinyl records encode sound as grooves around a disc, read by a needle at a specific RPM (International Electrotechnical Commission 2020). Magnetic tape encodes sound by varying levels of magnetism along a pigment covering the tape, which is then played back via a playback head (Camras & Martin 1989). Both analogue formats are susceptible to quality degradation over time.

Vinyl

Degradation in vinyl is often due to improper storage, with temperature, moisture, and physical stress being key factors. Darrell et al. (1959) demonstrated that high moisture and temperature changes accelerate chemical reactions within the vinyl, deforming the resin grooves. Using techniques like infrared spectroscopy and chemical analysis they found that a 15°F increase in temperature could double the reaction rate, and that ultraviolet light was a major factor in vinyl degradation. St-Laurent (1992) confirmed these findings but argued that vinyl was minimally affected by high humidity. This discrepancy may arise from differences in methodology, or due to materials tested, as Darrell et al. excluded PVC discs, however it was the standard material during St-Laurent's research.

Tape

Magnetic instability in tape pigments can lead to the gradual loss of audio quality; This process can be sped up through nearby strong magnetic fields due to demagnetisation. CLIR (2017) found that metal particulate pigment experienced a 2 dB loss in quality over their lifespan. Cassidy (2015) identified hydrolysis as a major variable, where humidity causes "sticky-shed syndrome" (SSS), deteriorating the iron oxide coating and leaving residue that can disrupt playback. Symptoms of SSS commonly include higher noise floors, pops, clicks and full dropouts. Hobaica (2013) found with tapes from the 1970s-1990s SSS was often caused due to hydrolysis in the tape's binder layer, leading to increased residue buildup and loss of high frequency loss. Whilst Hobaica looked at a limited time range, Schüller & Häfner (2014) broadened the scope, agreeing with Hobaica on the time frame they explored, but further showcasing how tapes from the 1940s-1960s are more prone to brittleness and physical issues than chemical issues like SSS.

Physical Degradation & Playback Issues

Both vinyl and tapes face physical issues, such as dirt and residue buildup, warping and brittleness from incorrect handling, or environmental factors such as excess temperature. Dust on the vinyl disc creates static that could cause crackles and pops during playback. Tapes suffer from increased brittleness overtime due to repeated rewinding. Holmes (1966) identifies high-frequency distortion and mechanical interference as significant contributors to tape hiss and hum, however later advancements allowed Greenspun (1985) to find that this could be limited with companders to improve the signal-to-noise ratio. Greenspun did note however, that these systems could introduce new artefacts, and was not able to control all artefacts such as "breathing" effects tying tape hiss level to the signals level. Whilst there are physical restoration methods available, such as tape burning which involves heating up the tape so that the pigment warps back to its original state, Davis & Shetzline (2020) stated that the common consensus among audio technicians is that these treated tapes, eventually revert back to their original condition, so digital preservation would be more preferable for permanent storage.

Summary

Degradation in analogue vintage audio formats often produces unwanted artefacts, such as hiss, hums, crackles, pops, breathing, saturation, clicks, dropout and skipping. For further consideration, time specific artefacts will be referred to as transient artefacts. Whilst some of these artefacts are irreversible like skipping, due to loss of audio information, the prototype system developed will aim to address reversible artefacts.

2.3.2 Historical Development of Noise Analysis & Reduction

The history of analogue restoration and noise reduction reflects evolving efforts to mitigate unwanted artefacts. From rudimentary splicing and filtering methods to more advanced technologies like Dolby's noise reduction systems, these techniques paved the way for modern approaches while highlighting the inherent limitations of analogue methods.

Analogue Techniques

Before the advent of dedicated noise reduction systems, early methods were manual and imprecise. Tape splicing involved physically removing noisy sections, but was destructive and impractical for removing unwanted artefacts, whilst equalisation (EQ) used filters to target noisy frequencies. Pollack (1948) discovered through listening tests that while filtering improved intelligibility of speech mixed with white noise, it also often removed desired frequencies from the speech as well. This highlighted EQ's limitation due to being a broadband tool. These fundamental techniques proved inadequate for preserving audio fidelity, driving the development of more specific approaches. The Dolby A noise

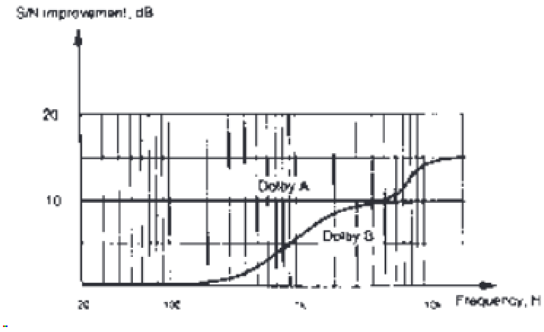

Figure 2.1: *Signal to noise ratio improvement found in Dolby a and b (Fisher 1998).*

reduction system introduced in 1965 represented a significant advancement. It employed pre-emphasis during recording and de-emphasis during playback, reducing hiss by 10-15 dB and hum by 10dB (Dolby 1968). It utilised a multiband approach, dividing frequency context into four bands, offering more precise noise reduction than a traditional analogue EQ technique. Subsequent Dolby products continue to innovate in the space, with Dolby B (Dolby 1971) and Dolby C (Dolby 1983) further enhancing noise reduction. Dolby C could achieve 15dB of noise reduction in the 2-8kHz range, whilst further incorporating a series of compressors and expanders for further capabilities. Whilst these advancements were impactful, Dolby's reporting of the systems raise concerns about bias with potentially false numbers for marketing due to originating from Dr Ray Dolby. Independent reports such as Fisher (1998) verified improvements in signal-to-noise ratio (SNR), as seen in Figure 2.1. Limitations still persisted with these systems however, as they were designed to only reduce consistent artefacts, such as hiss and hum, avoiding transient artefacts such as crackles and pops as this would require a much greater level of processing not available through analogue means. Furthermore, whilst analogue systems reduce noise during playback, they can't restore the original analogue audio file itself due to the inability to rewrite analogue media without degradation explored by Camras & Martin (1989).

Further analogue innovations such as the Dolby dbx and the Telefunken High-Com III improved noise reduction again, but still fell short in addressing transient artefacts. More modern analogue approaches such as the Vinyl NRS Box 3 address crackles, however this is achieved by digitising the analogue signal and applying processing before converting back to analogue. This reliance on digital processing suggests its importance within the area of artefact reduction.

Figure 2.2: *A representation of a pop located within an audio file and removed via a filter (Towne 2016).*

Digital Techniques

Digital noise reduction became more accessible with the rise of computers in the 1980's/ Early methods borrowed from analogue audio techniques, such as EQ, compressors and noise gates, however their non-destructive digital nature allowed for noise reduction without altering the original file. Transient artefacts like pops and crackles could be dealt with, however, as demonstrated by Towne (2016) they required manual intervention to do so. Towne demonstrates the removal of pops by applying filters to specific sections of the waveform seen in Figure 2.2. Whilst not automated, this approach laid the groundwork for how time-frequency specific techniques could be applied to handle transient artefacts.

The Fourier transform is a mathematical operation developed by Fourier (1822) which decomposes a signal, breaking it down into its fundamental frequencies (2.1).

$$f(x) = a_0 + \sum_{n=1}^{\infty} \left(a_n \cos\left(\frac{2\pi nx}{L}\right) + b_n \sin\left(\frac{2\pi nx}{L}\right) \right) \tag{2.1}$$

Whilst a great mathematical advancement, Fourier's series had limited applications as it was limited to periodic signals. Mathematicians such as Marcel Riesz and Henri Lebesgue would further develop this formula in the 20th century, expanding its use to include non-periodic signals. This is known as the Discrete Fourier Transform (DFT) (2.2).

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-\frac{2\pi i}{N} kn} \tag{2.2}$$

The is allowed for frequency analysis of sampled data, however it introduced other challenges, such as computational efficiency, requiring O(n2) operations, and no time domain representation to locate transient artefacts.

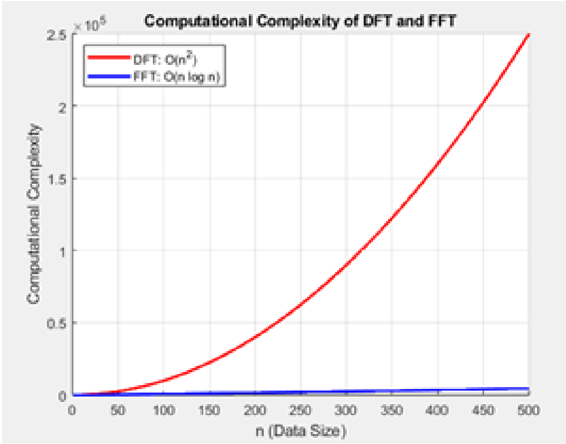

Figure 2.3: *Comparison of Complexity of DFT (O(n2)) vs FFT (O(nlogn))*

The Fast Fourier Transform (FFT), a breakthrough algorithm introduced by Cooley & Tukey (1965), reduced the complexity to O(nlog2n), allowing for large datasets to feasibly be processed and analysed in real-time application (figure 2.3). The FFT utilised a divide and conquer approach, recursively splitting the DFT into smaller sections to avoid redundant calculations being performed. This revolutionised signal processing for noise reduction by allowing for much more complex processing to occur than had previously been viable

Despite the FFT's efficiency, it was still held back by its inability to localise transient artefacts due to its lack of time domain representation. To improve upon this, the Short-Time Fourier Transform (STFT) was developed, utilising a sliding window approach to divide the signal into overlapping windows to allow for localised time-frequency analysis. An early iteration of the STFT (2.3) was utilised by Schafer & Rabiner (1973) to analyse and synthesise speech signals, however they noted that this iteration was computationally intensive and lacked efficiency. The formula was further refined, with one method including altering the global time index (2.3) used to a local time index (2.4) per window to improve efficiency by eliminating the need for time-reversing the windows.

$$X[k,l] = \sum_{m=lL}^{lL+N-1} w[l-m]x[m]e^{-i\omega_k m} \tag{2.3}$$

$$X[k,t] = \sum_{n=0}^{N-1} w[n]x[n+tL]e^{-i\omega_k n} \tag{2.4}$$

Despite the technical differences, Goodwin (2008) argued that the distinctions between these implementations are largely artificial, with the results differing only slightly. This

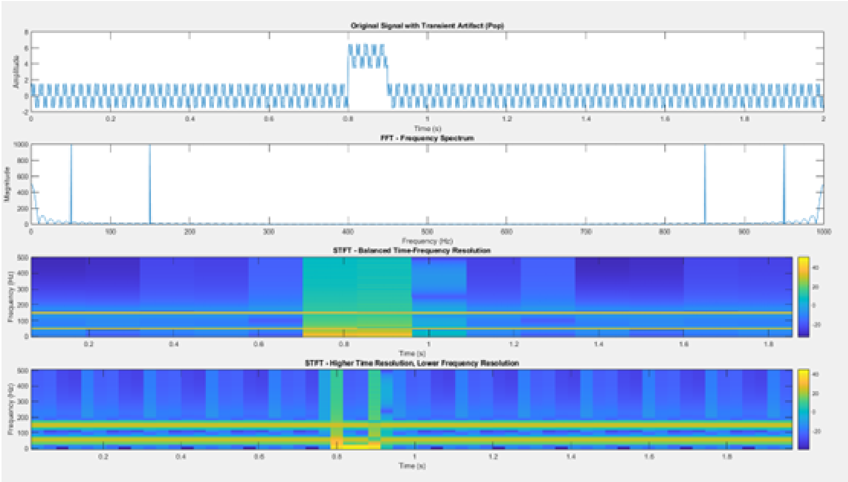

Figure 2.4: *Graph 1 = 2 sine waves with transient artifact added. Graph 2 = The FFT of this signal Graph 3 = STFT spectrogram, with a balanced time-frequency relationship. Graph 4 = STFT spectrogram with a higher time resolution and lower frequency resolution*

indicates that while the methods evolve to address efficiency and usability, their fundamental outputs remain closely aligned.

The effectiveness of the STFT is dependent on window size: larger windows improve frequency resolution but blur time detail, whilst smaller windows can better capture transient artefacts at the cost of frequency resolution (Figure 2.4). This ingrained trade-off demonstrates the limitations of the STFT. Despite its improvements over previous analysis techniques, both frequency and time domain information will be needed for the reduction of transient artefacts without affecting the original audio, so more advanced analysis techniques will be required.

Adaptive filtering adjusts its parameters in real-time based on incoming signal characteristics to remove noise or artifacts. It requires a reference signal that correlates with the noise, allowing the filter to adapt to the noise pattern (Beaufays 1995). This makes it effective in situations where the noise is predictable, such as periodic cardiac or respiratory signals in medical imaging. Deckers et al. (2006) demonstrated this technique in the context of reducing artifacts in medical data, where repetitive fluctuations, like those from heartbeats, could be identified and filtered out. The adaptive filter's performance in reducing these artifacts showed significant improvements in data quality by dynamically adjusting to the noise characteristics. Despite this, this methodology would not be suited for vintage audio restoration, since artefacts vary unpredictably between recordings, and there is often no known reference signal for the noise, making adaptive filtering impractical. Additionally, the reliance on known, predictable noise patterns limits its application for diverse and artefact reduction required for vintage audio.

Wavelet transforms offer a significant advantage over the STFT by allowing for variable resolution in time and frequency, which is especially useful for vintage audio restoration, as artefacts like hums or clicks can be targeted more precisely. Daubechies (1992) work on compactly supported wavelets has been foundational in practical applications, while Mallat (1998) introduced multiresolution analysis, crucial for time-frequency applications. These references are crucial for understanding wavelets' potential in audio restoration. Wavelets can be tailored to specific artefacts, such as those present in vintage recordings, allowing for more precise noise reduction (Flandrin 1999). Flandrin's work on wavelet time-frequency analysis is key to understanding their localised behaviour, making them ideal for isolating and removing artefacts from audio without affecting the signal quality. This body of literature highlights wavelet analysis as a crucial tool for improving audio restoration by offering flexibility and better localisation compared to STFT. In section 2.3.3 how wavelet analysis operates and demonstrate its effectiveness for vintage audio restoration is explored.

Machine learning and AI based approaches, such as convolution neural networks, have shown success in modern audio enhancement by learning to analyse complex patterns in signals (Purwins et al. 2019); However, for vintage audio restoration, AI faces significant challenges. It requires large, well-curated datasets for training, which are scarce for vintage audio. It also demands substantial computational resources, making it impractical for real-time applications, such as an audio plugin. Additionally, due to neural networks "black box" nature of not being able to see how a decision was made (Rudin & Radin 2019), it may cause issues if the system started to perform too much noise reduction, affecting the original intended audio, as well as the unwanted artefacts. Due to these issues, neural networks and AI is not used in this project. The more precise nature of wavelet analysis allows for a higher degree of accuracy.

Summary

The evolution of noise analysis and reduction techniques showcases the limitations of analogue methods for addressing transient artefacts, highlighting the need for advanced digital processing to achieve precise restoration. Wavelet analysis will be used to allow for the most tailored reductions to occur without affecting the original signal.

2.3.3 Wavelet Analysis

Wavelet analysis is grounded in established mathematical principles, widely acknowledged across relevant literature. This theoretical consistency has led to exploration in modern digital signal processing techniques, such as in applications requiring precise time-frequency analysis. By examining modern methodologies, this section aims to establish a clearer understanding of how wavelets can be effectively implemented in this project.

The main principle of wavelet analysis is the use of a fully scalable modulated window to solve the time-frequency domain issues found with Fourier transforms. This scalable window is shifted along the signal, calculating the similarities between the provided wavelet function and the original signal. It shifts along multiple times, changing the windows scaling each time. This results in a collection of time-frequency representations of the signal with differing resolutions, which allows for further multi resolution analysis (Valens n.d.). Daubechies (1992) states that for a wavelet function to be valid, it must fulfil two fundamental conditions. Firstly, the integral of the function over all time must equal zero (2.5).

$$\int_{-\infty}^{+\infty} \Psi(t)\, dt = 0 \tag{2.5}$$

Secondly, the function must possess finite energy, which is quantified by squaring the function and ensuring that the area under the resulting curve remains finite (2.6).

$$\int_{-\infty}^{+\infty} |\Psi(t)|^2 \, dt < \infty \tag{2.6}$$

Typically, the wavelet function is divided into two sections: the mother wavelet and the daughter wavelet. The mother wavelet, represented by Ψ(t), serves as the fundamental waveform. Daughter wavelets are the scaled version of the mother, allowing for the analysis of different frequencies and times, and is represented by Ψ(t), defined in (2.7).

$$\Psi_{a,b}(t) = \Psi\left(\frac{t-b}{a}\right) \tag{2.7}$$

The match between the wavelet and the signal is determined by taking the integral of the product of the original signal y(t) and the scaled and translated wavelet function (2.8).

$$T(a,b) = \int_{-\infty}^{+\infty} y(t) \cdot \Psi_{a,b}(t) \, dt \tag{2.8}$$

This equation calculates the total similarity between the wavelet function and the signal. When the signs match, a positive value is obtained; when they do not, a negative value is. The output signal T(a,b) provides the location within the original signal. As the wavelet slides from left to right along the signal, the initial matching of positive and negative values tends to cancel out around zero. However, as the match improves, the positive area becomes more significant, then decreases again as the match lessens. The amplitude of this function is calculated as the wavelet and the original signal move in and out of phase, with the peak amplitude being reached when they are perfectly in phase. The wavelet transform consists of both a real signal component and a complex wavelet function. The real signal is represented by y(t)y(t). The complex component is the wavelet function, which is composed of both real and imaginary parts. Wavelet functions are complex, combining a real and an imaginary component. A widely used example is the Morlet wavelet, as shown (2.9) and demonstrated by Morlet et al. (1982*a*,*b*)).

$$\Psi(t) = ke^{i\omega_0 t} \cdot e^{-\frac{t^2}{2}} \tag{2.9}$$

The complex exponential $e^{i\omega_0 t}$. can be decomposed using Euler's formula into its real (2.10) and imaginary (2.11) parts.

$$k\cos(\omega_0 t) \cdot e^{-\frac{t^2}{2}} \tag{2.10}$$

$$k\sin(\omega_0 t) \cdot e^{-\frac{t^2}{2}} \tag{2.11}$$

The cosine represents the real component and allows the function to analyse the phase of the signal to the wavelet. The sine represents the imaginary component, which is essential for capturing the signal's amplitude variations. Using these two components allows for the analysis of both time and frequency domains to provide a detailed representation of the signal's characteristics.

After the wavelet analysis has occurred, the restoration aspect then kicks in. Once artefacts have been located, T(a,b) has thresholding applied to it. In (2.12) and (2.13), methods of hard and soft thresholding are shown, as given by Ogden (1996). The threshold level is given by λ .

$$T(a,b) = \begin{cases} T(a,b) & \text{if } |T(a,b)| \geq \lambda \\ 0 & \text{if } |T(a,b)| < \lambda \end{cases} \tag{2.12}$$

$$T(a,b) = \begin{cases} \text{sign}(T(a,b)) \cdot (|T(a,b)| - \lambda) & \text{if } |T(a,b)| \geq \lambda \\ 0 & \text{if } |T(a,b)| < \lambda \end{cases} \tag{2.13}$$

Reconstruction of the signal is done by implementing the inverse wavelet transform function, as in (2.14).

$$f(t) = \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty T(a,b) \cdot \Psi_{a,b}(t)\, \frac{da\, db}{a^2} \tag{2.14}$$

In this, the constant CΨ is used to ensure that the signal is normalised correctly during reconstruction and is given by (2.15).

$$C_\psi = \int_0^\infty \frac{|\hat{\Psi}(\omega)|^2}{\omega}\, d\omega \tag{2.15}$$

Kularathne et al. (2023) utilised wavelets for the reduction of noise in vintage music originating from cassettes. It was found that the wavelet analysis was more effective than using the Fourier transform technique, as Fourier resulted in the removal of high frequency components of the music, whilst wavelets could avoid this. They found that the most practical wavelet function for vintage noise reduction was the Daubechies wavelet, and whilst others were considered, Daubechies gave the best results. The Haar wavelet was found to be unsuitable due to its requirement for 2n data points which is highly uncommon in audio recordings. Despite this, not all current literature agrees, as Brajevic (2011) found that the most effective methodology was for discrete artefacts such as clicks and crackles to be controlled using wavelet transform; However, with noise the most optimal solution was DFT processing. A difference in methodologies is that whilst Brajevic used wavelets for discrete artefacts in music, the noise reduction was used for image processing, which may have caused the difference. Brajevic also utilised Haar wavelet functions, which Kularathne, Fernando and Jayasinghe would later find to be unsuitable. Yang et al. (2021) focused on speech enhancement and found that when combining coiflet wavelet functions with traditional speech enhancement algorithms, the resulting denoised signal was much better than when only applying traditional methods. Patil (2015) also found that combining Singular Value Decomposition (SVD) with wavelet transform provided better noise reduction than either technique individually. Whilst these papers offer valuable insight into how different wavelet functions can be utilised for vintage audio, and how different de-noising methods can be implemented in addition to wavelet transform for a better outcome, there are few combining the two. This area may be worth further research.

2.3.4 Plugin Development

Prototype Development

A prototype algorithm is to be developed. Whilst a prototype may be developed in a range of programs, MATLAB will be developed in due to its ease of use, its variety of inbuilt toolboxes and its real-time processing abilities. Schooner (2023) stated that whilst python is an effective prototyping tool, MATLAB is generally preferred for data analysis due to built-in support for matrix operations. A useful tool provided using MATLAB is the built-in wavelet audio toolbox. Villanueva-Luna (2011) discusses this, providing algorithms to implement wavelet transforms for simplifying the programming involved to generate the wavelet function. For example, "wname = 'coif5';" is used to call a coiflet wavelet function. Mathworks (2017) implemented a wavelet denoising tool, allowing for more efficient prototyping for finding the appropriate wavelet functions to use.

Full Development

Following the prototype, further development of a full plugin will be proposed. Pirkle (2019) discusses three different plugin types: Avid Audio eXtension (AAX), Audio Units (AU) and Virtual Studio Technology 3 (VST3). The main difference between these is the DAW that they are used on, being Pro Tools, Logic Pro and other Mac OS DAW's, and the majority of DAW's cross platform respectively. Whilst there are development kits like VST SDK available for the development of VST3 plugins, it does not allow for the simultaneous creation. There are multiple frameworks capable of cross-platform development, such as iPlug2, JUCE and APSiK. JUCE will be proposed to be used due to its broad capabilities and detailed documentation, it can be seen as reliable for audio restoration and preservation as it is used by a range of companies involved in developing noise reduction tools, such as Izotope and Dolby (JUCE n.d.).

Summary

The prototype system evaluated here is to be developed on MATLAB, utilising the built-in wavelet functions and toolboxes. Further development into a full system, using the JUCE framework to build the plugin, will then be discussed, evaluating any potential limitations found within the prototype.

2.3.5 Preservation Standards

IASA (2009) establishes the Broadcast Wave Format (BWF) as the standard filetype for audio preservation, discussing the necessary inclusion of metadata for historical record. AES (2019) builds on this, agreeing with the same standard and further deliberating on how the BWF-E format offers additional metadata capabilities; For this project, BWF files would suffice as advanced metadata such as timecodes, as offered in BWF-E, would be unnecessary. Digital Preservation Coalition (2021) further expands on the role of BWF's metadata and how it is important to preserve not only the original audio quality, but the contextual information, reinforcing previous standards regarding file type.

IASA (2011) claims that restoration algorithms should only be applied to copies of vintage files, leaving the original digital file in its original state. Elaborating on the discussion of preserving original audio, Wallaszkovits (2018) raises ethical concerns around restoring vintage audio, stressing that any restoration techniques such as denoising should be sure to not change the original artistic intent; The choice of wavelets as the primary artefact reduction method adheres to this as they can be fine-tuned for specific changes.

As a plugin will be developed, the concerns surrounding file type and altering the original signal should be handled by the DAW it is being used on, rather than by the plugin itself. If standalone software were developed instead of a plugin, specific attention would need to be paid to file type compatibility and metadata preservation to meet established standards, such as those defined in IASA (2009) and AES (2019). However, since DAWs typically do not maintain or enforce metadata standards due to their focus on combining multiple audio files for mixing, this responsibility is ultimately placed on the user. Merging multiple files with differing metadata in a DAW would be impractical, therefore the ethical responsibility for complying with preservation standards falls on the user when using the plugin within their DAW environment.

2.3.6 Listening Tests

Evaluating the plugins performance and effectiveness will involve both subjective and objective measurements. These measurements will be analysed to see where the plugin can be improved.

Subjective Listening Tests

Zieliński (2006) discusses the importance of perceptual assessments, gathering listeners opinions through paired comparison tests. This is where participants evaluate two audio files - a before and after – and answer questions on their opinions. This provides useful insight into how the process is affecting the audio in a qualitative sense. Bech & Zacharov (2007) disagree with this concept, advocating for double-blind tests conducted on listeners knowledgeable on the topic. They criticise simpler methods, such as the paired comparison, as being vulnerable to bias, whilst Zielinski disagreed and stated that there is no evidence of bias between the two methods. Whilst Bech and Zacharovs more knowledgeable listeners provided more technical feedback, Zielinskis use of a range of listeners is better applicable to the target demographic of the plugin, who would have a range of technical skill. This is further discussed by Francombe et al. (2017) who argued that subjective evaluations should correlate with objective measurements, especially when optimising algorithms, as it would allow for the most in-depth analysis by finding the most important objective measurements to question subjective listeners on. Whilst this may overcomplicate the evaluation unnecessarily, if subjective and objective tests give widely different results this is worth investigating.

Objective Measurements

Torcoli et al. (2021) use objective metrics such as signal-to-noise ratio (SNR) and perceptual evaluation of audio quality (PEAQ) to quantify the impact that has been made on an audio quality. The PEAQ is a standardised algorithm to objectively measure perceived audio quality, grading it from 0 to 5. Liu & Fang (2023) expand on objective metrics, considering spectrograms and frequency specific analysis to provide a visual representation of noise measured within a classroom. These methods have been criticised due to lack of context meaning that how those objective changes take affect may not be clear, without contrasting those against subjective perceptions as suggested by Francombe et al.Bhattacharya et al. (2020) provides examples of wavelet transform personalised objective metrics, showcasing how traditional objective metrics such as the SNR and spectrograms can be utilised to visualise the effect of the wavelet transform on a signal.

Summary

By combining paired comparison tests demonstrated by Zielinski with wavelet specific objective measurements, a conclusion should be able to be made on the requirements fine-tuning or altering the system.

Chapter 3

System Design and Implementation

3.1 Overview of the System

Following the literature review, the prototype algorithm was developing in MATLAB due to its prototyping and signal processing capabilities. The system was split into different functions, seen in the flowchart in figure.3.1.

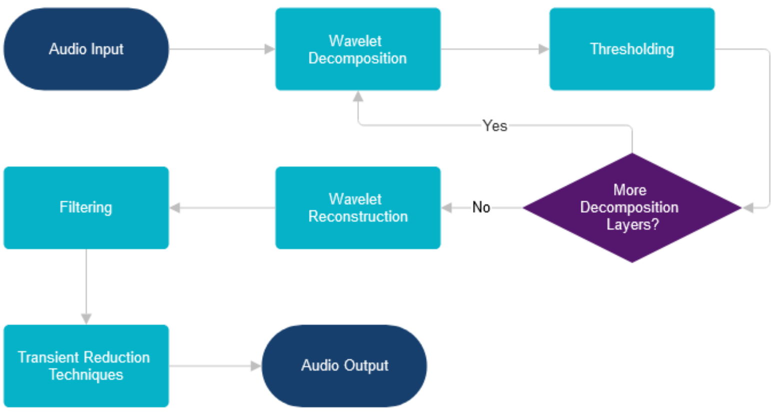

Figure 3.1: Flowchart showcasing the parts of the system.

3.2 Wavelet Transform Implementation

The initial function in the system is the wavelet decomposition, responsible for breaking down the signal into coefficients based on wavelet type and decomposition level provided. In the literature review the Continuous Wavelet Transform (CWT) was discussed 2.8, however due to improved computation speeds the Discrete Wavelet Transform (DWT) is used here instead. This efficiency arises from the DWT's use of repeated downsampling (Mallat 1999): at each decomposition level, the signal is downsampled by a factor of two. This reduces the number of calculations required at each level without compromising signal fidelity, as per the Nyquist theorem. Seen in equations 3.1 and 3.2, the DWT is comprised of two components, the detail coefficients (cD) and the approximation coefficients (cA); The cA holds the low frequency content whilst the cD coefficients contain the high frequency detailed components.

$$cA_j[k] = \sum_n x[n] \cdot h[2k - n] \tag{3.1}$$

$$cD_j[k] = \sum_n x[n] \cdot g[2k - n] \tag{3.2}$$

The DWT works by convolving the inputted signal with either a scaled filter (low-pass) or wavelet filter (high pass), repeating for each downsampled decomposition level. This allows for denoising thresholding to be applied exclusively to the cD coefficients, which then undergoes reconstruction using the Inverse Discrete Wavelet Transform (IDWT) (3.3) which does the opposite of the DWT, using the coefficients convolved with the inverses of the original filters to reconstruct the original signal.

$$x[n] = \sum_k cA_j[k] \cdot \tilde{h}[n - 2k] + \sum_k cD_j[k] \cdot \tilde{g}[n - 2k] \tag{3.3}$$

To handle edge effects during convolution, circular extension padding is applied before decomposition and removed after reconstruction. Padding ensures the wavelet filters have sufficient coverage at the signal boundaries. Circular padding was chosen as it preserves signal continuity by wrapping around the signal, preventing artificial discontinuities that could distort wavelet analysis, helping accurate reconstruction (Vetterli & Kovacevic 1995). Other common types of padding, such as zero padding, may have caused unwanted errors with sharp unwanted transients caused by the sharp signal drop to zero.

3.3 Thresholding Methods

Following the decomposition of the signal into its sets of coefficients, thresholding is applied to the detail coefficients to attenuate noise captured by the DWT. Three generalised thresholding methods were used within the system: universal, level-dependent, and adaptive. These were chosen based on MATLAB's Wavelet Toolbox but implemented manually from scratch.

The first method is the universal threshold, shown in Equation 3.4, which assumes the noise follows a Gaussian distribution. It uses the expression $\sqrt{2\log(N)}$, where N is the number of samples, as statistical theory suggests this provides an effective balance between noise reduction and minimal signal distortion (Donoho 1995).

$$T = \sigma\sqrt{2\log(N)} \tag{3.4}$$

The second approach is the level-dependent threshold, where the threshold value changes depending on the decomposition level i. As shown in Equation 3.5, this method reduces the threshold for deeper decomposition levels, which contain higher frequency content. This allows the algorithm to more aggressively suppress noise in higher frequencies while preserving low-frequency content that contributes to the fundamental structure of the signal (Donoho & Johnstone 1995).

$$T = \frac{\sigma\sqrt{2\log(N)}}{i+1} \tag{3.5}$$

The third method is the adaptive threshold, given in Equation 3.6, which introduces a dynamic scaling factor based on the square root of the decomposition level. While similar in concept to level-dependent thresholding, the adaptive method applies a more gradual reduction in threshold values across levels, providing a softer distinction between frequency bands. This can help avoid over-suppression of transients or subtle details in the mid-frequency range (Chang et al. 2000, Bhattacharya et al. 2020).

$$T = \frac{1}{\sqrt{i}} \sigma \sqrt{2 \log(N)} \tag{3.6}$$

Figure 3.2 illustrates how the threshold values vary across decomposition levels for each method. As shown, the universal method applies a constant threshold regardless of level, while the level-dependent and adaptive methods gradually reduce the threshold at higher levels. Notably, the level-dependent method decreases more sharply than the adaptive method, making it more aggressive in isolating high-frequency artefacts.

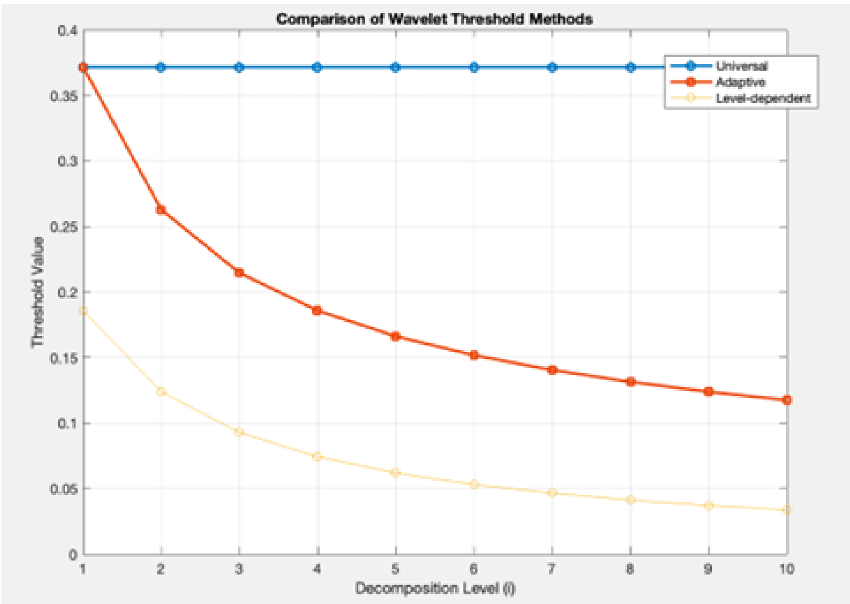

Figure 3.2: *Comparison of threshold values across decomposition levels for universal, level-dependent, and adaptive thresholding methods.*

3.4 Wavelet Selection and Design Rationale

A range of wavelets were used within the prototype to allow for testing and evaluation across different wavelet families and decomposition levels. These wavelets were selected to represent a variety of properties with respect to time and frequency localisation, symmetry, and reconstruction fidelity.

In the CWT, each wavelet is defined by a mother wavelet that is scaled and translated to form the complete transform. However, Mallat (2008) states that in the DWT, two filters are used: a high-pass filter (commonly referred to as the mother wavelet) and a low-pass filter (referred to as the father wavelet). Despite this difference, discrete wavelets must still satisfy the same conditions are continuous wavelet; They must have a mean of zero and a unit energy norm.

One of the most basic discrete wavelets is the Daubechies wavelet (db), known for its computational efficiency and strong time localisation (Daubechies 1992). The simplest of these, db1 -also known as the Haar wavelet- has coefficients given by:

$$h[n] = \left[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right] \tag{3.7}$$

$$g[n] = \left[\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right] \tag{3.8}$$

The number following a wavelet type (e.g., db4, coif5) corresponds to the number of vanishing moments it possesses. Daubechies (1992) found that a vanishing moment allows the wavelet to ignore polynomial trends of increasing order. For instance, a wavelet with four vanishing moments can remove up to cubic trends. As the number of vanishing moments increases, the wavelet provides more frequency resolution at the expense of time localisation.

Figure 3.3 shows the low-pass and high-pass filter coefficients for all wavelets used, clearly highlighting the increasing complexity with higher-order wavelets.

The wavelets included in the system are summarised below, along with the reasoning for their inclusion:

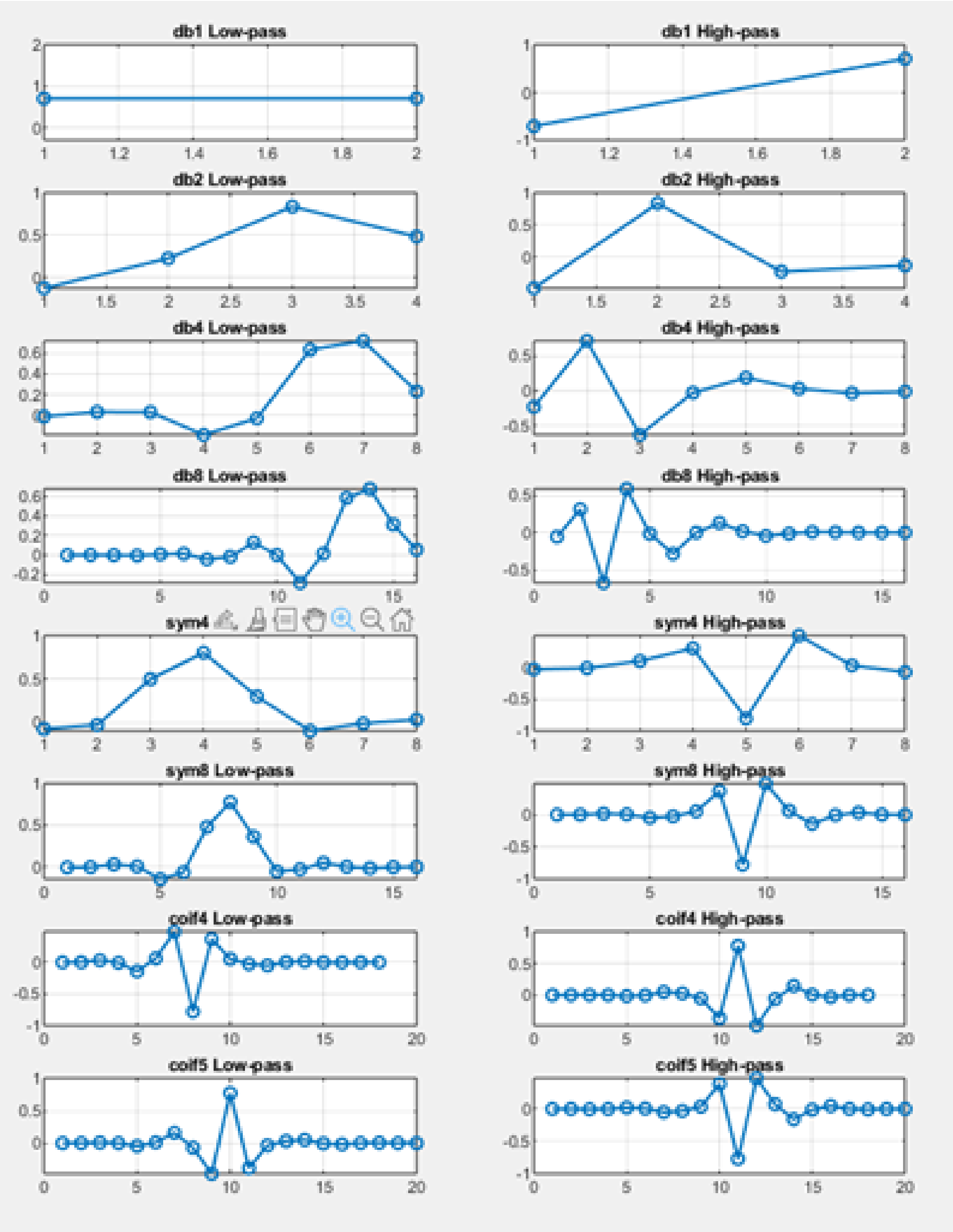

Figure 3.3: *Low-pass and high-pass filter coefficients for each wavelet used in the system.*

Wavelet Type	Reasoning
db1	Extremely simple and efficient, with strong time localisation. Ideal for detecting sharp transients, though it struggles with detailed frequency resolution. Often employed specifically for its time localisation (Vetterli & Kovacevic 1995).
db2 / **db4**	Intermediate options offering a more balanced trade-off between time and frequency localisation. Suitable for general-purpose denoising.
db8	Excellent frequency localisation, making it well-suited for analysing subtle signal variations. Computationally heavier and less responsive to rapid changes.
sym4 / **sym8**	Modified Daubechies wavelets designed for symmetry, which reduces phase distortion and improves reconstruction quality. (Mallat 2008, Vetterli & Kovacevic 1995)
coif4 / **coif5**	Coiflets have vanishing moments in both wavelet and scaling functions, allowing for effective preservation of both transients and smooth trends. They also improve reconstruction accuracy due to balanced analysis and synthesis filters. (Coifman & Wickerhauser 1992, Vetterli & Kovacevic 1995).

Table 3.1: Summary of wavelets used and their design rationale.

Figures 3.4 and 3.5 demonstrate the behaviour of different wavelets across four decomposition levels. In Figure 3.4, a fourth-order polynomial signal with an artificial pop is analysed using db1 through db8. It is clear that as the number of vanishing moments increases, frequency resolution improves, but time localisation diminishes. Simpler wavelets like db1 isolate the pop precisely, whereas more complex wavelets spread it over a broader window.

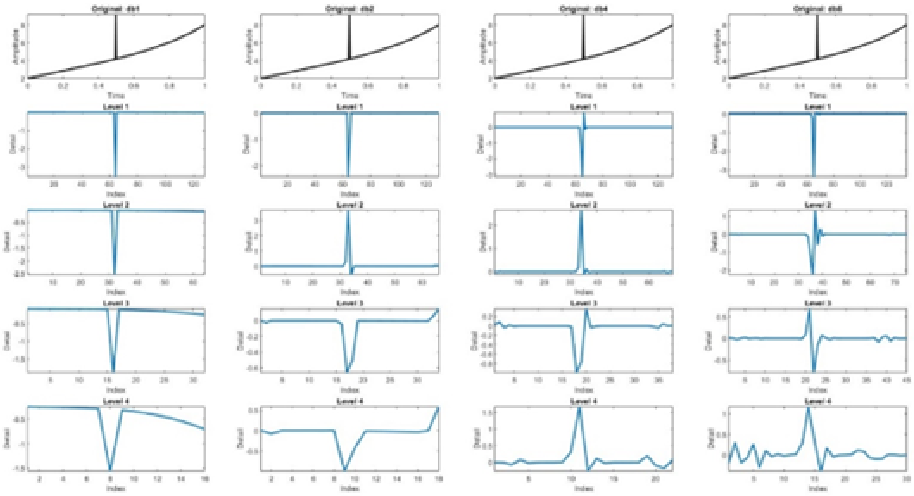

Figure 3.4: *Time-Amplitude plots showcasing wavelet detail coefficients (Levels 1–4) for a 4th-order polynomial signal with a synthetic pop. Columns represent different wavelets (db1, db2, db4, db8); rows show increasing decomposition levels (1–4). As the plots go right and down the wavelet coefficients can be seen having increasingly worse time localisation.*

Figure 3.5 repeats this analysis using white noise as the input. Again, the trade-off between time and frequency resolution is evident. Db1 captures transient peaks with high precision, while more complex wavelets blur these peaks but would provide more frequency information.

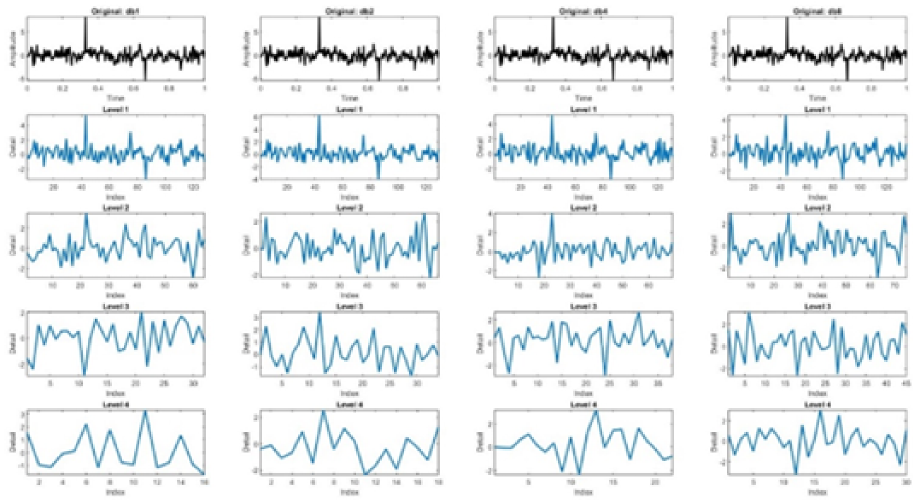

Figure 3.5: *Time-Amplitude plots showcasing wavelet detail coefficients (Levels 1–4) for white noise with a synthetic pop. Columns represent different wavelets (db1, db2, db4, db8); rows show increasing decomposition levels (1–4). As the plots go right and down the wavelet coefficients can be seen having increasingly worse time localisation.*

In implementation, decomposition filter coefficients are stored in dec_lo and dec_high arrays. These are flipped to create rec_lo and rec_high for use during the IDWT process.

During the development of the system, multiple methods were tested for applying the DWT across the frequency spectrum. Intially a single set of parameters was used, however this was unsuitable as the optimal decomposition level is different at different frequencies, upper bands being much more accepting of high decomposition levels. The signal was then tested with 3 methods: Three bands (Low, Mid, High), octave bands, and third-octave bands. Whilst an increasing number of bands would increase detail of the wavelet analysis, as a combination of filters (low,high,band pass) were used to separate the bands, when it came to third-octave bands the amount of filters would impact the audio quality due to filters interrupting each other. To compromise, the octave band splitting was selected. Additionally, to keep the number of parameters required to a minimum, the octave bands were further split into 3 categories:

- **Low band:** $<$500 Hz
- **Mid band:** 500–4000 Hz
- **High band:** $>$4000 Hz

This can be seen in figure 3.6.

3.5 Transient Artefact Detection and Removal

Transient artefacts such as crackles, clicks, and pops are a common problem found in vintage audio. To address these, three separate transient detection and removal methods were implemented and tested. Each was designed with unique strengths and trade-offs, suitable for different contexts.

```
% Define parameters for each band:
lowBandParams.wavelet = 'db4';
lowBandParams.level = 5;
lowBandParams.thresholdMethod = 'adaptive';

midBandParams.wavelet = 'sym8';
midBandParams.level = 4;
midBandParams.thresholdMethod = 'level-dependent';

highBandParams.wavelet = 'db2';
highBandParams.level = 17;
highBandParams.thresholdMethod = 'adaptive';


% wavelet multi-band denoising:
cleaned_audio = noise_clean_audio_signal_custom(noisySignal, Fs, lowBandParams, midBandParams, highBandParams);
```

ommand Window

```
Please enter the file input name (e.g., soundFile24.wav): lowNoise_Cheerios.mp3
Processing band 1 (0.0-28.3 Hz) center: 14.1 Hz...
Processing band 2 (28.3-56.6 Hz) center: 40.0 Hz...
Processing band 3 (56.6-113.1 Hz) center: 80.0 Hz...
Processing band 4 (113.1-226.3 Hz) center: 160.0 Hz...
Processing band 5 (226.3-452.5 Hz) center: 320.0 Hz...
Processing band 6 (452.5-905.1 Hz) center: 640.0 Hz...
Processing band 7 (905.1-1810.2 Hz) center: 1280.0 Hz...
Processing band 8 (1810.2-3620.4 Hz) center: 2560.0 Hz...
Processing band 9 (3620.4-7240.8 Hz) center: 5120.0 Hz...
Combining bands...
```

Figure 3.6: *MATLAB code showcasing a set of parameters and how an example sample is split into octave bands.*

3.5.1 Wavelet Coefficient Local Adaptive Threshold

The first method involved integrating transient detection directly into the wavelet denoising stage. A local adaptive threshold was calculated from the moving average of detail coefficients (cD) within a defined sliding window. Coefficients exceeding this threshold were attenuated cautiously, aiming to minimise false detections and preserve audio integrity. However, this careful approach means that it may not completely eliminate all transient artefacts, particularly those contained within in complex audio segments.

3.5.2 Modulus Maxima Method with Multi-Layer Verification

The second method uses a modulus maxima approach as discussed by Mallat & Hwang (1992), detecting transient peaks as local maxima (points higher in magnitude than their immediate neighbours) in wavelet detail coefficients. To ensure reliability, each detected peak underwent additional verification across multiple decomposition layers; a transient was confirmed only if a corresponding peak was detected at consecutive decomposition levels. This cross-layer verification significantly increased robustness against false positives.

However, this method has the inherent limitation of reduced temporal accuracy at deeper decomposition levels due to downsampling. Ideal transient detection generally occurs at the highest resolution decomposition level for optimal time localisation. Despite this, filtering simultaneously at multiple levels compensates somewhat, allowing for effective transient reduction.

Figure 3.7 illustrates transient peaks detected at multiple decomposition levels, highlighting this method's robustness.

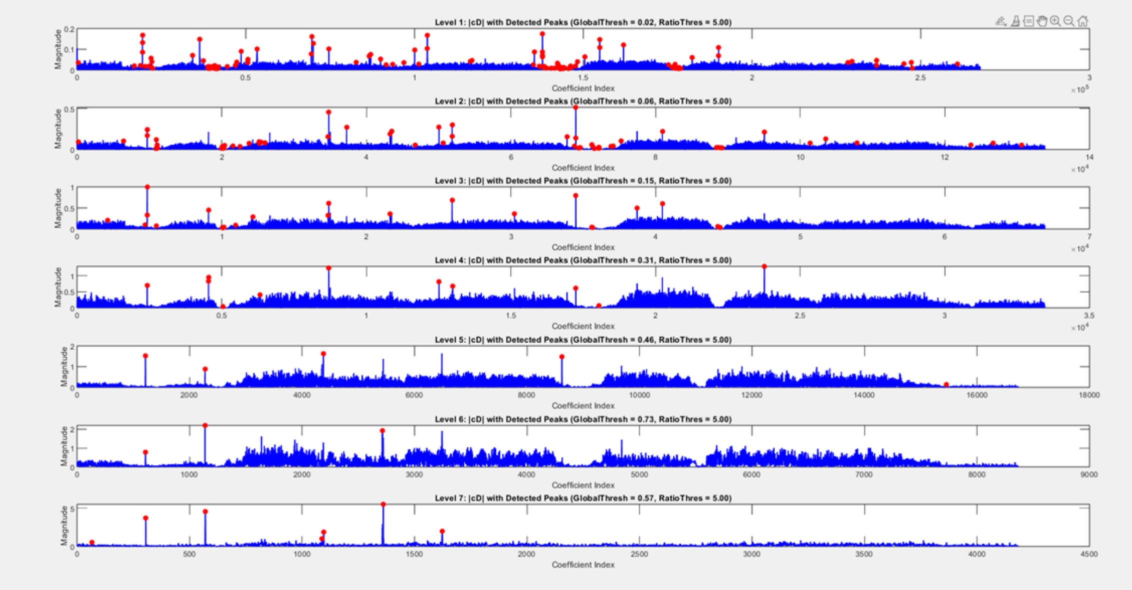

Figure 3.7: *Modulus maxima transient detection across multiple decomposition levels. Each graph represents 2 layers, for example the top layer checks decomp layers 1-2 together. Red markers indicate confirmed transient peaks.*

3.5.3 Time-Domain Amplitude and Derivative-Based Detection

The third method operated only in the time domain, simultaneously using amplitude and derivative-based criteria to robustly detect transients. This approach follows principles used in onset detection, where sharp changes in energy and slope often signify the start of transients (Bello et al. 2005). Specifically, the method:

- Calculated the local median amplitude within a sliding window.
- Identified sharp transient events by detecting derivative spikes exceeding eight

times the local median derivative.

- Confirmed transient events only if their amplitude exceeded 1.5 times the local window's average amplitude.
- Expanded each detected transient region by 4 milliseconds on either side of the peak to fully capture the artefact. 4ms on either side was chosen as many of the artefacts found within vintage audio, such as crackles, often span several ms as opposed to instantaneous pops that may be found with digital audio.
- Retained the transient identification only if the transient segment detected was shorter than 4 times the expansion window. If this condition is not met, it is assumed to be sustained audio and not an unwanted transient.
- Smoothed each confirmed transient region via linear interpolation between the start and end of the defined transient region.

The complete MATLAB implementation of this method is provided in the appendix in Listing C.1.

Figure 3.8 visualises the clearly identified transient detection segments in the time-domain waveform.

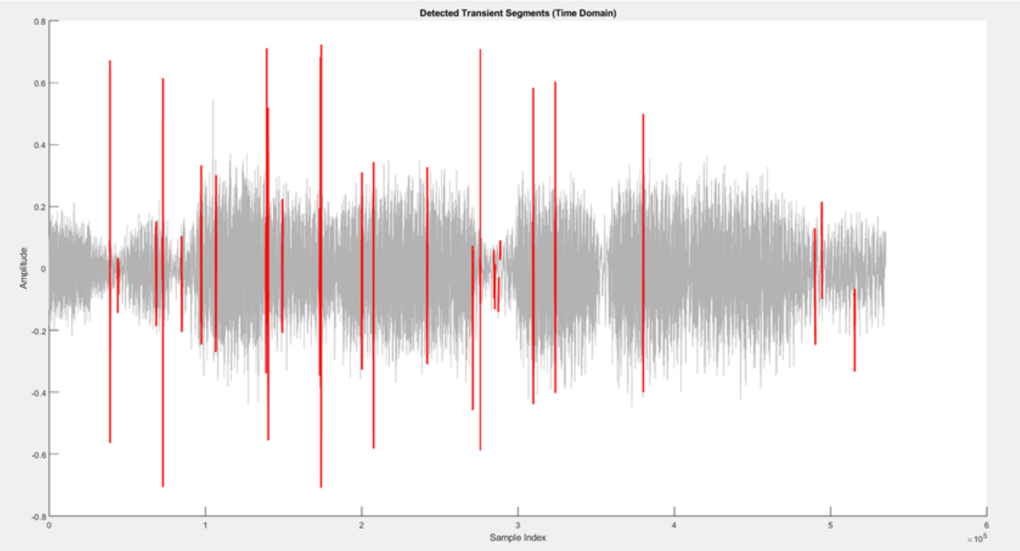

Figure 3.8: *Time-domain transient detection - segments detected as a transient marked in red.*

3.5.4 Comparative Evaluation of Methods

Comparative waveform analyses of all three methods in silent and active audio contexts demonstrate their relative effectiveness and limitations:

- During Silence (Figure 3.9): All three methods successfully detected and attenuated transient artefacts. Method 3 removed the transient entirely, though at the risk of introducing interpolation artefacts. Methods 1 and 2 moderately attenuated the transient, preserving audio integrity but not fully eliminating the artefact.
- During Audio (Figure 3.10): Method 1 only lightly affected the transient to avoid disturbing complex audio content. Method 2 significantly reduced transient amplitude but demonstrated slightly reduced interpolation accuracy. Method 3 fully removed the transient, yet posed a risk of introducing subtle interpolation artefacts into the audio.

Further subjective listening tests will quantify the perceptual impacts of each method.

Figure 3.9: *Transient interpolation comparison during silent audio segment. Purple line = Original Signal ; Orange = Method 1 ; Yellow = Method 2 ; Green = Method 3.*

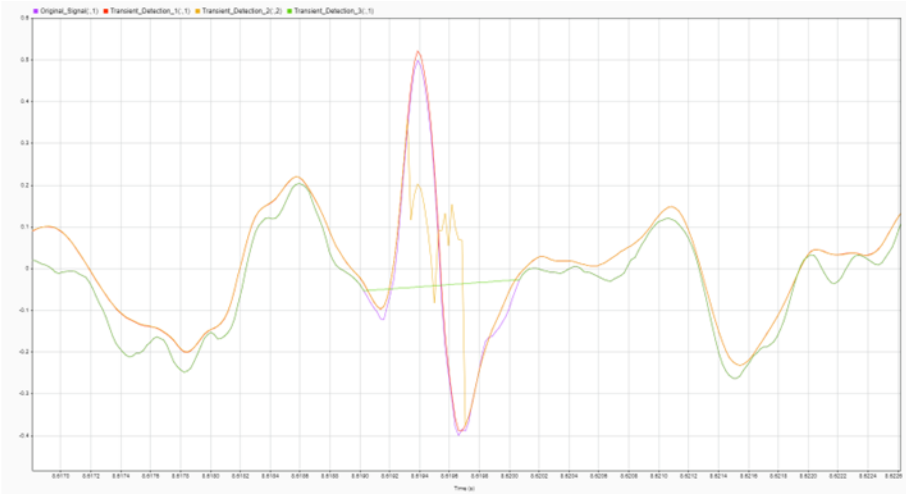

Figure 3.10: *Transient interpolation comparison during active audio segment. Purple line = Original Signal ; Orange = Method 1 ; Yellow = Method 2 ; Green = Method 3.*

3.5.5 System Design Philosophy

The systems developed were created for future transition to a plugin in mind. Although the system does not work in real-time, it was developed conscious of facilitating the future transition; The algorithms modular structure, separation of functions, and simple data handling were all created with this in mind. These considerations lead to a simple signal flow, and the modularity allows for single parameters changed in a plugin environment to not require the full system to be recomputed, just the applicable section. Additionally, the system design was approached using few internal MATLAB functions to facilitate rapid development when transitioning to another architecture.

Chapter 4

Testing Methodology

To assess the quality of the system and how varying settings may affect the outcome, both subjective listening tests and objective metrics were used for evaluation. A range of settings were applied to a variety of audio files and tested to find the accepted optimal settings, and any limitations that may be found automating the system.

4.1 Variant Selection and Parameter Justification

The three core parameters of the denoising stage of the system were important to ensure that they were chosen selectively for the tests. With the system capable of eight wavelet types, a reasonable range of 20 decomposition levels, and 3 threshold types, combined with the parameters being split into bands, left a possible 1440 parameters combinations. To narrow the selection down to 5 resulting audio files, assumptions were made to create 5 generalised sets of parameters, seen in table 4.1.

The parameters were chosen to represent a meaningful spread across a broad range, whilst also ensuring that the selected combinations would be sensible choices based on current literature and previous research in the area. Wavelet families like Daubechies, Symlets, and Coiflets each behave slightly differently, offering trade-offs in time and frequency resolution that affect how noise is removed and how the signal may be altered.

Table 4.1: Wavelet Denoising Parameters for Test Variants

Parameters	Variant 1	Variant 2	Variant 3	Variant 4 (S)	Variant 5 (S)	Variant 4 (M)	Variant 5 (M)
Low Band Wavelet	db4	sym4	coif5	db4	db4	db4	sym8
Low Band Level	7	8	8	4	7	5	5
Low Band Threshold	univ.	l-d.	l-d.	univ.	univ.	adapt.	adapt.
Mid Band Wavelet	db2	sym4	coif4	sym4	sym4	sym4	sym4
Mid Band Level	5	8	6	8	8	7	7
Mid Band Threshold	l-d.	l-d.	l-d.	l-d.	l-d.	l-d.	l-d.
High Band Wavelet	db4	sym8	coif5	db4	sym8	sym8	sym8
High Band Level	15	15	16	10	15	13	13
High Band Threshold	adapt.	adapt.	l-d.	adapt.	adapt.	adapt.	adapt.

Variant 1 used Daubechies wavelets across all bands, as these are the most commonly used and offer a simple starting point for noise reduction without being too aggressive, as suggested in earlier work. Variants 2 and 3 followed on from this, using Symlets and Coiflets respectively to see how alternative wavelet families perform under the same conditions.

Variants 4 and 5 were split up for speech and music signals to reflect their different characteristics. In general, they used a mix of Daubechies and Symlets, allowing for comparisons between how each wavelet family handles the same types of content. The music variants used adaptive thresholds in the low bands, while the speech variants used universal thresholds. This was based on the idea that speech tends to have more stable low-frequency content, with the most important frequencies for intelligibility sitting around 1–4 kHz (Zorila et al. 2012). Music, on the other hand, relies on detail across the entire spectrum, so adaptive thresholding helps preserve transients and texture in the low end from instruments like drums and bass.

Finally, Variants 4 and 5 used slightly lower decomposition levels than Variants 1 to 3, to give insight into how smaller changes in resolution might affect the denoising and overall sound quality.

4.2 Subjective Listening Tests

To assess perceptual quality, a listening test was carried out where participants were asked to rate the denoised signals based on preference and perceived quality. The test was split into two parts: the first focusing on general audio quality between the five wavelet-based denoising variants, and the second targeting specific artefacts related to transient crackles.

4.2.1 Test Design and Structure

The test was created using the Web Audio Evaluation Tool (Jillings et al. 2015), a browser-based tool dedicated to building accurate and repeatable audio listening tests. Auditory perceptual evaluation (APE) tests were used due to there widely accepted methods for accurate perceptual evaluation (Bech & Zacharov 2007, De Man et al. 2014) . Participants were presented with a reference signal and asked to rate five processed variants using a continuous sliding scale from 0 to 1, with 0 being the worst and 1 being the best.

Zieliński (2006) suggests that bias often occurs in perceptual listening tests due to visual factors, so to reduce this the variants were presented in random order for each participant and each question. Additionally, variants were labelled A–E, and these labels were shuffled for every question, so participants could not track which version was which.

The first six questions focused on overall denoising quality. Questions 1–3 presented speech signals, while Questions 4–6 used music excerpts. The final three questions (7–9) focused on transient artefact removal. For these, no explicit reference signal was provided. Instead, participants were asked to rate four signals - three processed using the transient methods described in Section 3.5, and one unprocessed original signal embedded within the group. For each section of 3 it was made up of 2 vintage audio files and 1 clean modern signal that had artificial white noise added; This allowed a comparison of how the system handled vintage, analogue artefacts differently than artificial digital noise. The reference material for each listening test question is summarised in Appendix B.1, providing context on the type and source of each signal used.

4.2.2 Evaluation Criteria

Participants were asked to judge the audio based on different criteria, depending on the type of material in the question:

Speech Tests

- **Speech intelligibility** - how clearly the speech is understood.
- **Absence of noise** - how effectively background noise has been reduced.

Music Tests

- **Music quality and fidelity** - how clearly the music is heard.
- **Absence of noise** - how effectively background noise has been reduced.
- **Naturalness of audio** - minimal distortion or unnatural artefacts.

Transient Artefact Test

- **Effectiveness of crackle reduction** - how well transient artefacts such as pops and clicks were removed.
- *Participants were asked to ignore other factors such as overall fidelity, tonal balance, or background noise.*

4.2.3 Test Setup and Equipment

To ensure consistency across responses, all tests were conducted using the same equipment: a MacBook, a Behringer UMC204HD audio interface, and Sennheiser HD599SE headphones. Tests were taken in quiet environments with no background noise.

4.2.4 Participant Background

The listening test was completed by 15 participants, a sample size consistent with previous perceptual audio evaluation studies conducted in controlled environments with participants from a range of skill levels (Cartwright et al. 2016, Goot et al. 2023). Before starting the test, participants were asked to rate their own experience with audio mixing and critical listening. This allowed the anonymous results to be viewed in the context of the listener's experience level. Since the system is intended to be usable by people with a lower skill level, agreement between low-skill and high-skill listeners would suggest that the system performs effectively for its intended audience.

4.3 Objective Metric Evaluation

To complement subjective testing and contribute to a more complete analysis, four objective metrics were used to evaluate the quality of the denoised signals. These metrics were selected based on their relevance to perceptual quality and their established use in previous denoising and audio enhancement research (Torcoli et al. 2021). The four metrics used were Log-Spectral Distance (LSD), Signal-to-Noise Ratio (SNR), Itakura–Saito Distance (ISD), and Mel Cepstral Distortion (MCD). Each metric was chosen to capture different aspects of the signal's fidelity and distortion, helping to provide a more rounded view of the performance of each processing variant.

Log-Spectral Distance (LSD)

LSD measures the difference between the log-magnitude spectra of the reference (noisy) signal and the denoised version. It quantifies how much the frequency content has changed due to processing. A lower LSD value generally means that the spectral envelope has been better preserved, which is particularly important for speech and music signals (Kubichek 1993), especially when used in an archival setting. On the other hand, a high LSD value suggests that significant spectral distortion has occurred, potentially altering important speech elements or the harmonic structure in music, which can negatively affect intelligibility or timbre.

Signal-to-Noise Ratio (SNR)

SNR is a widely used metric that in this context, compares the energy of the noisy input signal to the energy of the difference between the noisy and processed signals. While often associated with measuring clarity, here a high SNR suggests the processed signal remains close in energy to the original, which may mean that not much noise has been removed. Conversely, a lower SNR could suggest more aggressive noise reduction, but if it drops too far, it may also mean that important parts of the signal have been removed or degraded along with the noise (Martin 2001).

Itakura–Saito Distance (ISD)

ISD is a perceptually motivated distance measure that compares the power spectral densities (PSDs) of two signals. It is particularly sensitive to differences in spectral shape, making it useful for evaluating whether the natural distribution of energy across frequencies has been preserved. A low ISD value implies that the spectral characteristics of the signal are mostly intact, whereas a high ISD may indicate that the denoising process has introduced noticeable spectral changes, which can affect the naturalness of both speech and music (Gray & Markel 1976).

Mel Cepstral Distortion (MCD)

MCD measures the Euclidean distance between the Mel-frequency cepstral coefficients (MFCCs) of the reference and processed signals. Since MFCCs effectively capture the spectral envelope in a perceptually relevant way (Davis & Mermelstein 1980), MCD is often used to judge how well the tonal characteristics of speech and music have been preserved. Lower MCD values suggest that the timbre and spectral shape are intact, while higher values can point to noticeable distortions. In speech, small variations may still be acceptable, but for music, where subtle details are more apparent, a low MCD is especially important (Kominek & Black 2004).

Summary

Together, these objective metrics provide a quantitative basis to compare and interpret the subjective feedback obtained in listener tests.

Chapter 5

Evaluation

5.1 Objective Metrics and Results

All objective metrics were computed using custom MATLAB functions, ensuring consistent evaluation across all audio files and variants. Each metric was calculated following widely accepted formulas and standard definitions from the literature to ensure transparency and reproducibility.

5.1.1 Log-Spectral Distance (LSD)

The LSD results in Figure 5.1 indicate that for vintage recordings, Variant 3 generally performed poorly, with Variant 1 also showing relatively weaker results, though less severe than Variant 3. Variants 2, 4, and 5 provided the best outcomes on vintage material, with consistently lower LSD values suggesting reduced spectral distortion, likely due to the use of symlets. Conversely, Variant 3 performed better on artificial examples. However, this might suggest excessive removal of broadband noise rather than preservation of signal fidelity.

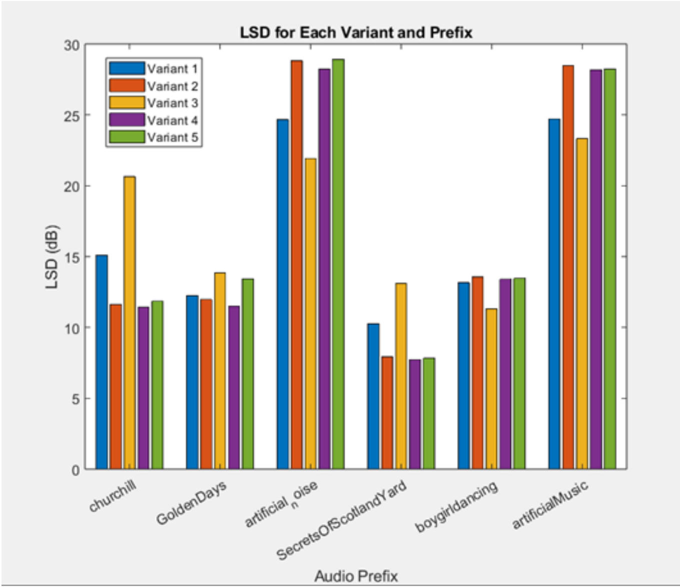

Figure 5.1: *Bar chart showcasing LSD across all variants for 6 denoised audio samples.*

5.1.2 Signal-to-Noise Ratio (SNR)

The SNR results shown in Figure 5.2 complement the LSD findings. Variants 2 and 4 consistently achieved higher SNR values, particularly for vintage recordings, reflecting effective noise removal without compromising audio quality. Variant 3 produced significantly lower SNR scores, highlighting its inadequacy in balancing noise removal with audio integrity. Variant 5 displayed a mixed performance, underperforming on speech but matching Variants 2 and 4 on music, indicating adaptive thresholding with symlets is more effective for complex audio signals.

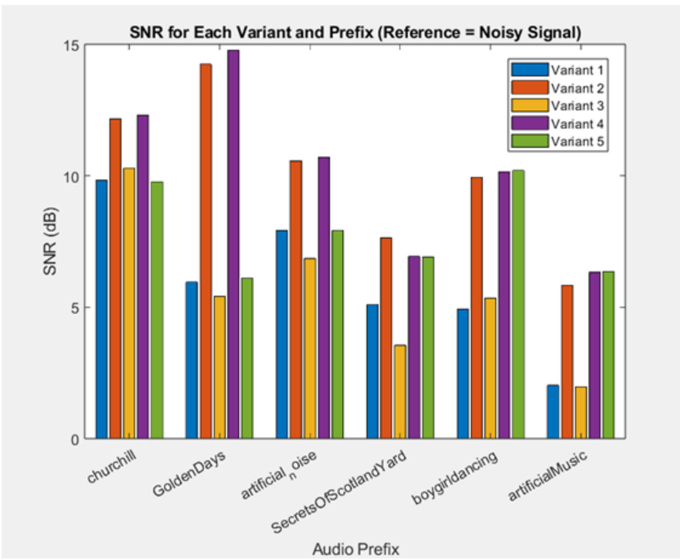

Figure 5.2: *Bar chart showcasing SNR across all variants for 6 denoised audio samples*

5.1.3 Itakura–Saito Distance (ISD)

ISD results in Figure 5.3 further illustrate the gap between authentic vintage and artificially noised recordings. Artificially noised samples consistently showed higher ISD values, signifying greater divergence from the original spectral characteristics. Variants 2 and 4 maintained relatively low ISD values, especially with vintage audio, suggesting good spectral fidelity (Bhattacharya et al. 2020). Variant 1 occasionally preserved spectral balance better despite weaker overall noise reduction. Variant 3 consistently showed higher ISD, reaffirming its suboptimal performance.

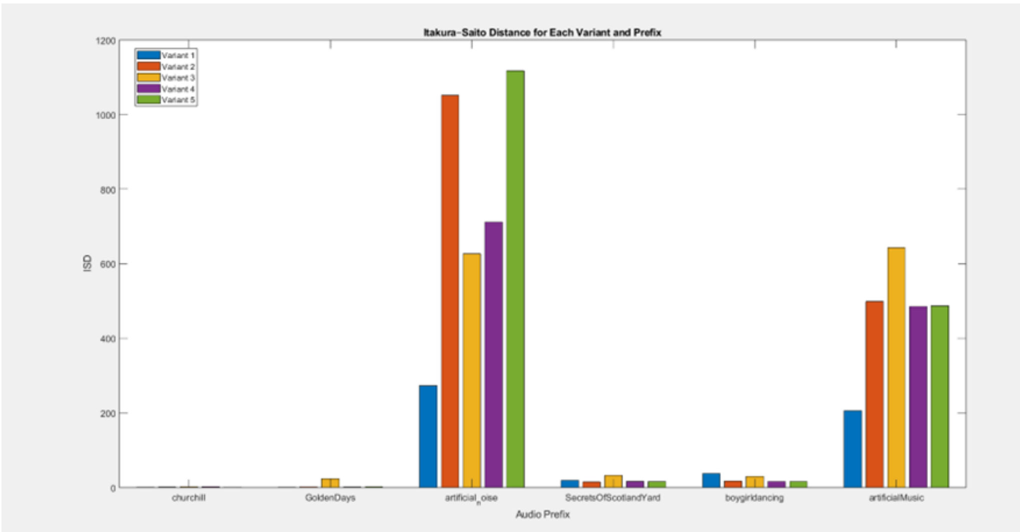

Figure 5.3: *Bar chart showcasing ISD across all variants for 6 denoised audio samples*

5.1.4 Mel Cepstral Distortion (MCD)

As illustrated in Figure 5.4, MCD values support trends seen in other metrics, with Variants 2 and 4 again showing the lowest distortion, particularly in speech. Variant 3 consistently showed the highest distortion across both speech and music. Variant 5 followed a similar pattern to its performance in SNR, underperforming in speech but performing comparatively well in music. This further supports adaptive thresholding as the better choice for more complex audio signals.

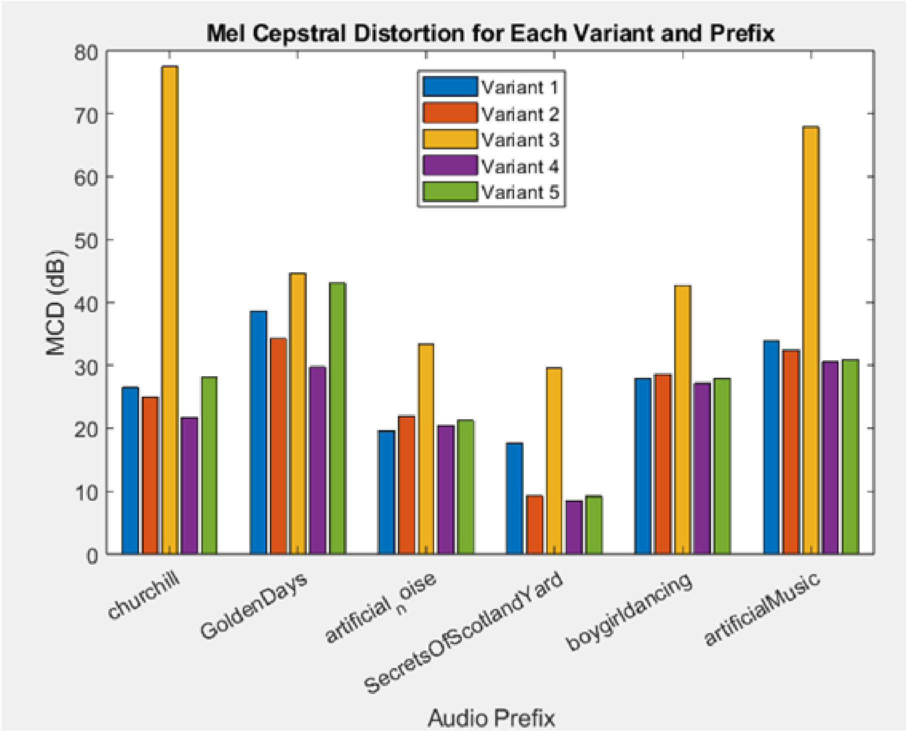

Figure 5.4: *Bar chart showcasing MCD across all variants for 6 denoised audio samples*

5.1.5 Objective Metric Summary

Across all four objective metrics, Variants 2 and 4 consistently delivered the best balance of effective noise removal and audio fidelity, clearly outperforming other variants. Variant 3 consistently showed poor performance, especially evident in artificial noise scenarios. Variant 5's performance varied with content type, highlighting adaptive thresholding and symlets as particularly beneficial for music.

The objective results clearly demonstrate that the system is particularly suited for vintage audio restoration, effectively addressing real-world audio artefacts compared to artificial noise scenarios. Figure 5.5 visually summarises these trends, reinforcing the consistently strong performance of Variants 2 and 4.

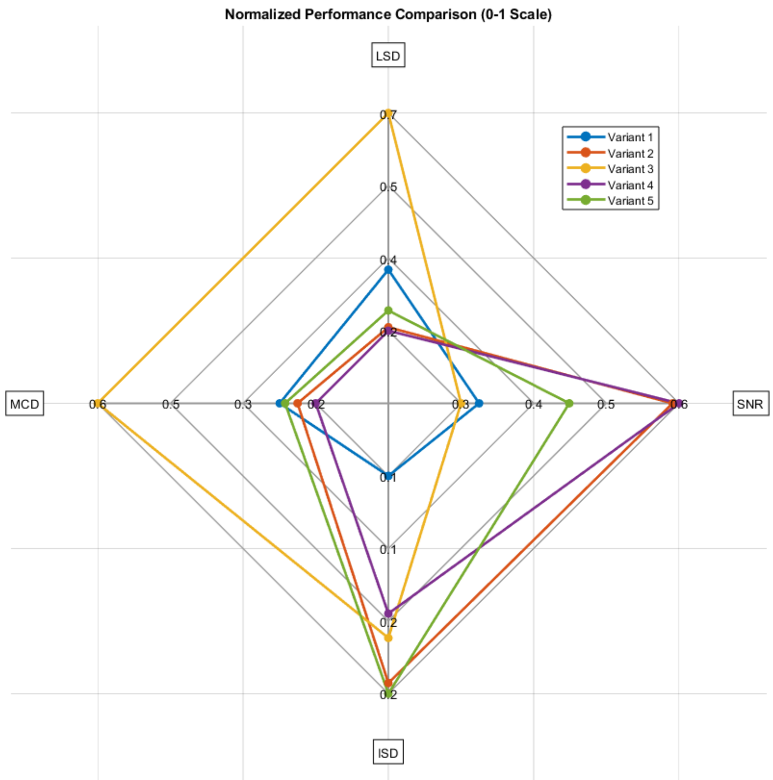

Figure 5.5: *Radar plot showcasing normalised average performance of each variant across all objective metrics.*

5.2 Integration with Subjective Evaluation

The subjective evaluations (Figures 5.6 and 5.7) closely align with objective findings. Variants 2 and 4 received the highest subjective ratings, consistent with their strong objective performance. Variant 3 was consistently rated lowest, mirroring its poor objective metrics. Variant 5's mixed subjective performance further validated objective findings, performing better on music than on speech seen in figure 5.8.

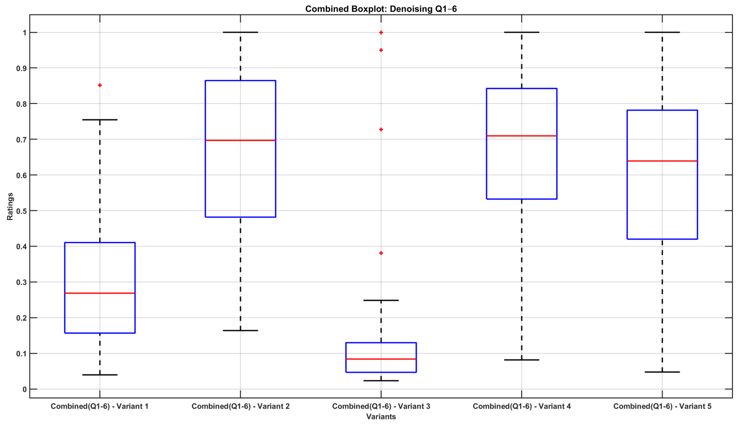

Figure 5.6: *Box plot showcasing listening test questions 1-6 results averaged.*

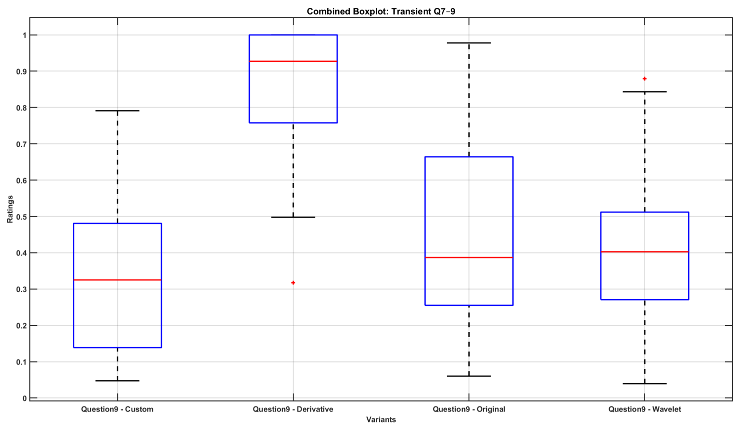

Figure 5.7: *Box plot showcasing listening test questions 7-9 results averaged.*

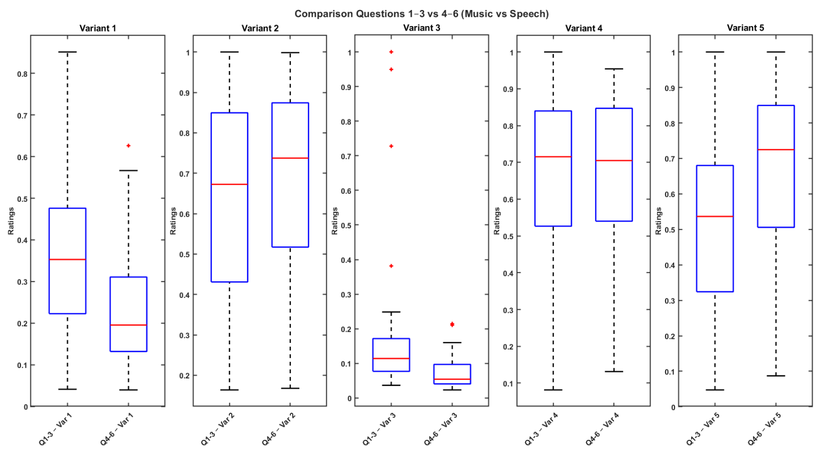

Figure 5.8: *Box plot showcasing listening test questions 1-3 averaged compared against questions 4-6 averaged.*

The transient-artefact removal evaluation (Figure 5.9) identified the derivative method as the most perceptually effective. Participants consistently rated it highest, particularly on artificial transient examples. The wavelet-based approach showed negligible subjective differences from the original, while the modulus-maxima method struggled on vintage samples but did better on artificial clicks.

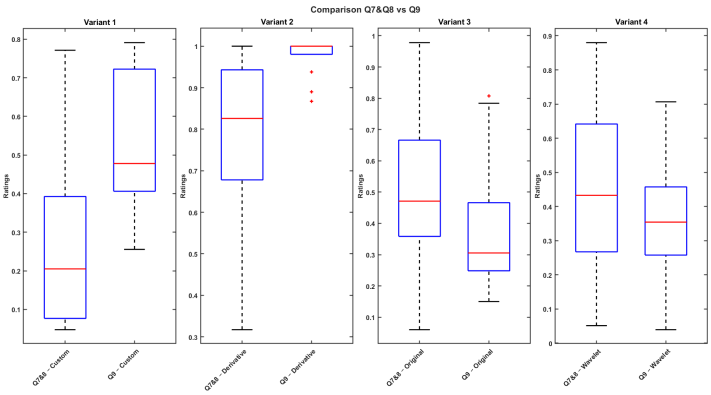

Figure 5.9: *Comparison of transient removal methods for the vintage samples (Q7-8) against the artificial sample (Q9). Plots: 1 = Method 2; 2 = Method 3; 3 = Original; 4 = Method 2*

All listening test result plots available in appendix section D.

5.3 Discussion and Limitations

Both subjective and objective results consistently identified Variants 2 and 4 as the best performing options for denoising vintage audio recordings. The strong performance of these variants can be attributed to their use of symlet wavelets combined with adaptive thresholding in high-frequency bands. Symlets offer near-symmetry and smoothness, traits which effectively preserve transient details and maintain signal integrity, essential characteristics when dealing with vintage audio content. Their balanced time-frequency localisation likely contributed to the preservation of important audio features, reflected in both listener preferences and objective metrics - The MCD of variants using symlets averaged 24dB, whilst Daubechies were at 28dB and coiflets 49dB showcasing symlets superior capability for preserving spectral qualities. This aligns with findings by Chavan (2010), who demonstrated the efficacy of Symlet wavelets in denoising applications for the preservation of speech signals.

Contrarily, the consistent underperformance of Variant 3 indicates potential issues inherent to their application in audio restoration contexts. Coiflets, though symmetrical and theoretically advantageous for signal reconstruction, appeared overly aggressive in noise removal. This aggressive removal often resulted in noticeable artefacts and significant distortion of the original audio signal. One possible reason for this could be the coiflets broader support width, causing excessive smoothing and inadvertently removing perceptually important features alongside noise.

Daubechies wavelets, as used in Variant 1, provided moderate results. Their simpler structure does not offer sufficient resolution for detailed audio signal preservation when compared to symlets. However, their performance in simpler contexts, such as speech, suggests that Daubechies wavelets still hold value, especially in scenarios where complexity and high resolution are less critical. Additionally, Daubechies wavelets provided valuable analysis for the transient detection algorithms, providing better time localisation than others; This reinforces the standard choice of the Haar wavelet (db1) for time specific analysis that is commonly seen within wavelet research (Daubechies 1992, Bello et al. 2005, Hartmann 2016)

Interestingly, some discrepancies emerged between subjective listener ratings and objective metrics. For instance, objective measures suggested Variant 1 would perform better on music, however, subjective tests indicated that Variant 1 performed notably better on speech, achieving an average listener rating of 0.4 for speech samples peaking at 0.85, compared to just 0.19 for music peaking at 0.58. This discrepancy arises because speech signals are inherently simpler, benefiting from minimal interference with spectral content due to the simpler structure of Daubechies wavelets. Listeners appeared to prefer the lighter denoising effect that preserved speech intelligibility and naturalness over more aggressive processing.
Similarly, while Variant 3 consistently ranked lowest overall with an average rating of just 0.08, there were notable exceptions. A minority of participants rated Variant 3 highly, particularly for vintage speech samples, giving it ratings as high as 0.95 and 1. This preference might result from Variant 3's aggressive approach, reducing audible noise significantly but substantially compromising spectral fidelity and signal quality, as evidenced by its higher MCD value of 78dB, compared to the other four variants which each fell within 20dB to 30dB. Nonetheless, simpler speech signals retained sufficient clarity for

certain listeners prioritising strong noise removal over fidelity. These observations highlight varying listener priorities, suggesting user preferences might differ based on individual tolerance for artefacts versus spectral fidelity.
These listening test results underline an important consideration: although most participants aligned closely with objective metrics (which clearly identified Variants 2 and 4 as the best with subjective ratings averaging around 0.71), individual user preferences can still broadly vary. Therefore, when further developing the system into an automated restoration tool, recognising and accommodating these listener preferences could be achieved through user-adjustable parameters.

Vintage artefacts often exhibit specific spectral characteristics which were correctly targeted by adaptive and level-dependent thresholding methods used in Variants 2 and 4. The results clearly demonstrate that the developed system is suited for addressing vintage artefacts, such as tape hiss, crackles, and hums, rather than artificially generated broadband white noise. This agrees with Li & Zhou (2008) who found adaptive thresholding was optimal for audio denoising, and was further expanded on by Nishan & Vijay (2014) who found that adaptive thresholds still struggle with Gaussian white noise. This also aligns with results found with the artificial noise audio samples resulting in lower scores than the vintage one, reinforcing the systems usage for vintage artefacts.

The findings in this research generally align with current research discussed in section 2.3.3, supporting the notion that wavelet-based approaches, specifically those employing symlets and adaptive thresholding, provide significant advantages in audio restoration tasks. Previous studies such as Yadav et al. (2015) similarly highlighted symlets capability in preserving fidelity and transient details, reinforcing their suitability in applications like vintage audio restoration. The observed limitations and advantages of coiflet and Daubechies wavelets also reflect broader academic consensus: coiflets tend to be overly aggressive and result in a consistently lower SNR than symlets (Zaeni et al. 2018); Daubechies wavelets, while effective in simpler scenarios, lack the precision and flexibility of symlets in more demanding contexts.

For transient artefact removal, the clear listener preference for the derivative method for interpolation confirms its practical value. However, the potential demonstrated by the custom modulus maxima method in accurately detecting transient artefacts suggests that a hybrid approach combining detailed wavelet modules maxima based detection with derivative-based interpolation could significantly enhance transient artefact removal effectiveness. Another common approach for transient detection is utilising onset detection and machine learning, however this often introduces unnecessary complication and computational time, with research such as Böck & Widmer (2013), Schlüter & Böck (2014) showcasing how despite the potential for machine learning in this scenario, simple CNN models with a lower computational complexity often still pick up false negatives. In contrast, the modulus maxima approach, particularly with its validation across multiple decomposition levels, offers more consistent detection of transients without the burden of extensive training data or model tuning, making it more suitable for an automated system.

The comprehensive analysis of these results showcase these two artefact removal approaches combined can be an effective strategy for vintage audio restoration. Integrating symlet-based denoising parameters with adaptive thresholding and a sophisticated transient detection and interpolation system achieves the most consistent results as a set of default parameters.

Chapter 6

Conclusions

This study successfully investigated and evaluated a wavelet based approach for automated vintage audio restoration. Comprehensive objective and subjective analysis confirmed that specific combinations of wavelet types and thresholding methods significantly influence the effectiveness of audio restoration. Variants 2 and 4, utilising symlets with adaptive and level-dependent thresholding, consistently showed superior performance across various metrics and listening tests, making them the ideal candidates for denoising vintage audio.

The findings revealed symlets effectiveness, due to their balanced time-frequency localisation and smoothness, providing robust preservation of important audio details. Conversely, coiflets demonstrated overly aggressive noise reduction, frequently causing noticeable audio distortions, making them unsuitable for vintage audio restoration. Daubechies wavelets offered moderate performance, adequate in simpler speech contexts but limited by their relatively basic structure when addressing more complex audio signals.

Transient artefact removal was effectively handled using a derivative-based interpolation method, which outperformed alternative methods in subjective evaluations. The custom modulus maxima approach demonstrated considerable potential in detecting transient artefacts, although interpolation accuracy remained a limitation. Integrating these two methods could achieve optimal results, balancing precise detection with effective artefact removal.

The study also confirmed that the system was specifically suited to handling vintage audio artefacts, effectively reducing tape hiss, crackles, and hum, while struggling with artificially generated broadband noise.

Overall, the project demonstrated the practical and theoretical benefits of tailored wavelet configurations and adaptive transient artefact removal methods, laying a solid foundation for the development of automated noise removal tools for vintage audio restoration.

Chapter 7

Recommendations for Further Work

Future work should focus on translating the successful prototype system into a fully automated audio restoration plugin.

First, the system needs an intelligent classification system able to identify the audio type (such as speech, music, or more). Machine learning techniques, particularly lightweight classification models trained on select appropriate audio samples, could efficiently categorise inputs in real-time, ensuring the appropriate wavelet parameters and thresholding methods are automatically applied. Given computational constraints in real-time plugins, these models should be optimised for minimal processing load.

As the MATLAB prototype was designed without relying on specific MATLAB toolboxes to ensure flexibility, plugin development should use the JUCE framework for cross-platform plugin creation. Using JUCE, the implementation of custom wavelet algorithms and thresholding approaches could be easily integrated into a real-time audio processing environment.

Additionally, further refinement of transient artefact detection and interpolation methods could significantly enhance restoration quality. A hybrid approach combining the detection capability of the modulus maxima method with the derivative methods interpolation should be explored thoroughly. Real-time constraints mean these algorithms would need efficient optimisation, possibly through selective multi-scale analysis, to maintain responsiveness without sacrificing quality. Additionally, transient detection may be the cause of limitations when translating the system to a real-time implementation, as an artificial delay at least as long as the transient detection window would be required for complete analysis using the current approach.

Latency optimisation is a key factor for future real-time implementation. The current transient detection algorithm employs a sliding adaptive window, which would introduce an artificial delay due to its reliance on future samples in a live processing scenario. To resolve this, additional development and testing would be necessary. Potential solutions include reducing the adaptive window length to an acceptable latency threshold or redesigning the algorithm to rely exclusively on past samples, removing the need for future context and reducing overall latency.

Lastly, extensive user testing within various DAWs should guide the final plugins user interface design, ensuring accessibility for users with differing skill levels. Features such as preset management, detailed help files, and generalised parameters could greatly enhance user experience and practicality for less skilled users.

Based on the findings of this project, future development of an automated vintage audio restoration plugin appears highly achievable.

Chapter 8

References

AES (2019), 'Aes standard on network and file transfer of audio - audio-file transfer and exchange - file format for transferring digital audio data between systems of different type and manufacture', https://www.aes.org/publications/standards/search.cfm?docID=53. [Accessed 21 Nov. 2024].

Beaufays, F. (1995), 'Transform-domain adaptive filters: an analytical approach', *IEEE Transactions on Signal Processing* **43**(2), 422–431.

Bech, S. & Zacharov, N. (2007), *Perceptual Audio Evaluation: Theory, Method and Application*, John Wiley & Sons, Ltd.

Bello, J. P., Daudet, L., Abdallah, S., Duxbury, C., Davies, M. & Sandler, M. B. (2005), 'A tutorial on onset detection in music signals', *IEEE Transactions on Speech and Audio Processing* **13**(5), 1035–1047.

Bhattacharya, P. C., Tangsangiumvisai, N. & Worapishet, A. (2020), 'Performance improvement of adaptive wavelet thresholding for speech enhancement using generalized gaussian priors and frame-wise context modeling', *IEEE Access* **8**, 168361–168380.

Böck, S. & Widmer, G. (2013), Maximum filter vibrato suppression for onset detection, *in* 'Proceedings of the 16th International Conference on Digital Audio Effects (DAFx-13)'.

Brajevic, Z. (2011), Elimination of unwanted signals in audio materials using wavelet transform, *in* 'Proceedings ELMAR-2011', IEEE, pp. 229–233.

Camras, M. & Martin, D. W. (1989), *Magnetic Recording Handbook*, Vol. 85, The Journal of the Acoustical Society of America.

Cartwright, M., Pardo, B. & Mysore, G. J. (2016), Fast and easy crowdsourced perceptual audio evaluation, *in* '2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)', IEEE, pp. 619–623.

Cassidy, B. M. (2015), Spectroscopy and Chemometrics for Cultural Heritage and Forensics: Conservation of Magnetic Tape by Identifying Degraded Polymer Signatures; and Characterizing the Chemiluminescent Reaction of Luminol with Blood, PhD thesis, University of South Carolina. [Accessed 19 Nov. 2024].

Chang, S. G., Yu, B. & Vetterli, M. (2000), 'Adaptive wavelet thresholding for image denoising and compression', *IEEE Transactions on Image Processing* **9**(9), 1532–1546.

CLIR (2017), 'What can go wrong with magnetic media?', `https://www.clir.org/pubs/reports/pub54/2what_wrong/`. [Accessed 12 Nov. 2024].

Coifman, R. R. & Wickerhauser, M. V. (1992), 'Entropy-based algorithms for best basis selection', *IEEE Transactions on Information Theory* **38**(2), 713–718.

Cooley, J. W. & Tukey, J. W. (1965), 'An algorithm for the machine calculation of complex fourier series', *Mathematics of Computation* **19**(90), 297–301.

Darrell, R. D., Pickett, A. G. & Lemcoe, M. M. (1959), 'Preservation and storage of sound recordings; a study supported by a grant from the rockefeller foundation', *Notes* **17**(1), 51.

Daubechies, I. (1992), *Ten Lectures on Wavelets*, Society For Industrial And Applied Mathematics, Philadelphia.

Davis, A. & Shetzline, J. (2020), Towards understanding the thermal remediation of degraded archival reel-to-reel audio tapes, *in* 'American Chemical Society Spring 2020 National Meeting, SciMeetings Virtual Platform'.

Davis, S. B. & Mermelstein, P. (1980), 'Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences', *IEEE Transactions on Acoustics, Speech, and Signal Processing* **28**(4), 357–366.

De Man, B., Reiss, J. D. & Moffat, D. (2014), Ape: Audio perceptual evaluation toolbox for matlab, *in* '136th Convention of the Audio Engineering Society'.

Deckers, R. H. R. et al. (2006), 'An adaptive filter for suppression of cardiac and respiratory noise in mri time series data', *NeuroImage* **33**(4), 1072–1081.

Digital Preservation Coalition (2021), ‘Preserving audio’, https://doi.org/10.7207/twgn21-11.

Dolby, R. (1968), ‘Dr. dolby on noise reduction’, *Audio* **52**(6), 19–22. [Accessed 2 Nov. 2024].

Dolby, R. (1971), ‘A noise reduction system for consumer tape recording’, https://secure.aes.org/forum/pubs/conventions/?elib=1905. [Accessed 2 Nov. 2024].

Dolby, R. (1983), ‘A 20 db audio noise reduction system for consumer applications’, *Journal of the Audio Engineering Society* **31**(3), 98–113. [Accessed 2 Nov. 2024].

Donoho, D. L. (1995), ‘De-noising by soft-thresholding’, *IEEE Transactions on Information Theory* **41**(3), 613–627.

Donoho, D. L. & Johnstone, I. M. (1995), ‘Adapting to unknown smoothness via wavelet shrinkage’, *Journal of the American Statistical Association* **90**(432), 1200–1224.

Fisher, D. (1998), *Noise Reduction Systems*, p. 139. [Accessed 20 Nov. 2024].

Flandrin, P. (1999), *Time-Frequency/Time-Scale Analysis*.

Fourier, J. B. J. (1822), *Théorie analytique de la chaleur*, Didot. [Accessed 2 Nov. 2024].

Francombe, J. et al. (2017), ‘Evaluation of spatial audio reproduction methods (part 2): Analysis of listener preference’, *Journal of the Audio Engineering Society* **65**(3), 212–225. [Accessed 21 Nov. 2024].

Goodwin, M. M. (2008), The stft, sinusoidal models, and speech modification, *in* J. Benesty, S. M. Mohan & Y. A. Huang, eds, ‘Springer Topics in Signal Processing’, Springer Berlin Heidelberg, pp. 229–258.

Goot, D. K., Siedenburg, K., Abeßer, J. & Plumbley, M. D. (2023), ‘Perceptual evaluation of music real-time communication applications’, *IEEE Access* **11**, 46860–46873.

Gray, A. H. & Markel, J. D. (1976), Distance measures for speech processing, *in* ‘IEEE Transactions on Acoustics, Speech, and Signal Processing’, Vol. 24, IEEE, pp. 380–391.

Greenspun, P. (1985), ‘Audio analysis iii: ‘hi-fi’ video recorders’, *Computer Music Journal* **9**, 68–79. [Accessed 16 Nov. 2024].

Hartmann, D. L. (2016), ‘Lecture 9: Wavelet analysis and time-frequency decomposition’.

Hobaica, S. (2013), 'Analysis of audio magnetic tapes with sticky shed syndrome by atr-ftir', *Journal of Applied Polymer Science* **128**(3), 1962–1973.

Holmes, R. B. (1966), 'What's wrong with our tape recordings?', *The Modern Language Journal* **50**, 463–468. [Accessed 12 Nov. 2024].

IASA (2009), 'Guidelines on the production and preservation of digital audio objects'.

IASA (2011), 'Ethical principles for sound and audiovisual archives'.

International Electrotechnical Commission (2020), 'Iec 60098:2020 - analogue audio disk records and reproducing equipment', `https://webstore.iec.ch/`. [Accessed 16 Nov. 2024].

Jillings, N., Moffat, D., De Man, B. & Reiss, J. D. (2015), Web Audio Evaluation Tool: A browser-based listening test environment, *in* '12th Sound and Music Computing Conference'.

JUCE (n.d.), 'Juce — juce', `https://juce.com/`. [Accessed 14 Nov. 2024].

Kominek, J. & Black, A. W. (2004), The cmu arctic speech databases, *in* 'Fifth ISCA Workshop on Speech Synthesis'.

Kubichek, R. (1993), 'Lpc distance measures for speech recognition', *IEEE Transactions on Speech and Audio Processing* **1**(4), 359–372.

Kularathne, N., Fernando, M. & Jayasinghe, J. (2023), 'High-frequency noise removal of audio files using daubechies wavelet transform', *Vidyodaya Journal of Science* **26**.

Li, N. & Zhou, M. (2008), Audio denoising algorithm based on adaptive wavelet soft-threshold of gain factor and teager energy operator, *in* '2008 International Conference on Computer Science and Software Engineering', Vol. 1, IEEE, pp. 1013–1016.

Liu, F. & Fang, J. (2023), 'Multi-scale audio spectrogram transformer for classroom teaching interaction recognition', *Future Internet* **15**.

Mallat, S. (1998), *A Wavelet Tour of Signal Processing*.

Mallat, S. (1999), *A Wavelet Tour of Signal Processing*, 2nd edn, Academic Press.

Mallat, S. (2008), *A Wavelet Tour of Signal Processing: The Sparse Way*, 3rd edn, Academic Press, Burlington, MA.

Mallat, S. & Hwang, W. (1992), 'Singularity detection and processing with wavelets', *IEEE Transactions on Information Theory* **38**(2), 617–643.

Martin, R. (2001), Noise power spectral density estimation based on optimal smoothing and minimum statistics, *in* 'IEEE Transactions on Speech and Audio Processing', Vol. 9, IEEE, pp. 504–512.

Mathworks (2017), 'Wavelet signal denoiser', `https://www.mathworks.com/help/releases/R2021b/wavelet/ref/waveletsignaldenoiser-app.html`. [Accessed 23 Nov. 2024].

Morlet, J., Arens, G., Fourgeau, E. & Giard, D. (1982*a*), 'Wave propagation and sampling theory – part i: Complex signal and scattering in multilayered media', *Geophysics* **47**, 203–221.

Morlet, J., Arens, G., Fourgeau, E. & Giard, D. (1982*b*), 'Wave propagation and sampling theory – part ii: Sampling theory and complex waves', *Geophysics* **47**, 222–236.

Nishan, S. & Vijay, L. (2014), 'Audio noise reduction from audio signals and speech signals', *International Journal of Computer Science Trends and Technology (IJCST)* **2**(5). Available at: `https://www.researchgate.net/publication/317686262_Audio_Noise_Reduction_from_Audio_Signals_and_Speech_Signals`.

Ogden, R. T. (1996), *Essential Wavelets for Statistical Applications and Data Analysis*, Birkhauser Boston Inc.

Patil, R. (2015), 'Noise reduction using wavelet transform and singular vector decomposition', *Procedia Computer Science* **54**, 849–853.

Pirkle, W. C. (2019), *Designing Audio Effect Plugins in C++: For AAX, AU, and VST3 with DSP Theory*, second edn, Routledge.

Pollack, I. (1948), 'Effects of high pass and low pass filtering on the intelligibility of speech in noise', *The Journal of the Acoustical Society of America* **20**(3), 259–266.

Purwins, H., Li, B., Virtanen, T., Schluter, J., Chang, S.-Y. & Sainath, T. (2019), 'Deep learning for audio signal processing', *IEEE Journal of Selected Topics in Signal Processing* **13**, 206–219.

Rudin, C. & Radin, J. (2019), 'Why are we using black box models in ai when we don't need to? a lesson from an explainable ai competition', *Harvard Data Science Review* **1**(2).

Schafer, R. & Rabiner, L. (1973), 'Design and simulation of a speech analysis-synthesis system based on short-time fourier analysis', *IEEE Transactions on Audio and Electroacoustics* **21**(3), 165–174.

Schlüter, J. & Böck, S. (2014), Improved musical onset detection with convolutional neural networks, *in* 'Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)', IEEE, pp. 6979–6983.

Schooner, N. (2023), 'Matlab vs python: 6 key differences [updated 2023]', `https://blog.boot.dev/python/matlab-vs-python/`. [Accessed 20 Nov. 2024].

Schüller, D. & Häfner, A., eds (2014), *Handling and Storage of Audio and Video Carriers (IASA-TC 05)*, 1st edn, IASA. [Accessed: 20 November 2024].

St-Laurent, G. (1992), 'Preservation of recorded sound materials', *ARSC Journal* **23**(2), 144–156.

Torcoli, M. et al. (2021), 'Objective measures of perceptual audio quality reviewed: An evaluation of their application domain dependence', *IEEE/ACM Transactions on Audio, Speech, and Language Processing* **29**, 1530–1541.

Towne, J. (2016), 'P-pops and other plosives', `https://transom.org/2016/p-pops-plosives/`. [Accessed 15 Nov. 2024].

Valens, C. (n.d.), 'A really friendly guide to wavelets'.

Vetterli, M. & Kovacevic, J. (1995), *Wavelets and Subband Coding*, Prentice Hall, Upper Saddle River, NJ.

Villanueva-Luna, A. E. (2011), *De-Noising Audio Signals Using MATLAB Wavelets Toolbox*, IntechOpen.

Wallaszkovits, N. (2018), 'Between standards and arts: Digitisation and restoration of audio material - a balancing act between authenticity and manipulation?', *Journal of New Music Research* **47**(4), 285–290.

Yadav, A. K., Roy, R., Kumar, A. P., Kumar, C. S. & Dhakad, S. K. (2015), De-noising of ultrasound image using discrete wavelet transform by symlet wavelet and filters, *in* '2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI)', pp. 1204–1208.

Yang, Y., Liu, P., Zhou, H. & Tian, Y. (2021), A speech enhancement algorithm combining spectral subtraction and wavelet transform, pp. 268–273.

Zaeni, A., Kasnalestari, T. & Khayam, U. (2018), Application of wavelet transformation symlet type and coiflet type for partial discharge signals denoising, *in* ‘2018 5th International Conference on Electric Vehicular Technology (ICEVT)’, pp. 78–82.

Zieliński, S. (2006), ‘On some biases encountered in modern listening tests’.

Zorila, T., Barker, J. & Cooke, M. (2012), ‘Speech intelligibility in adverse conditions: A review’, *Speech Communication* **54**(9), 1072–1088.

Appendix A

Gantt Chart

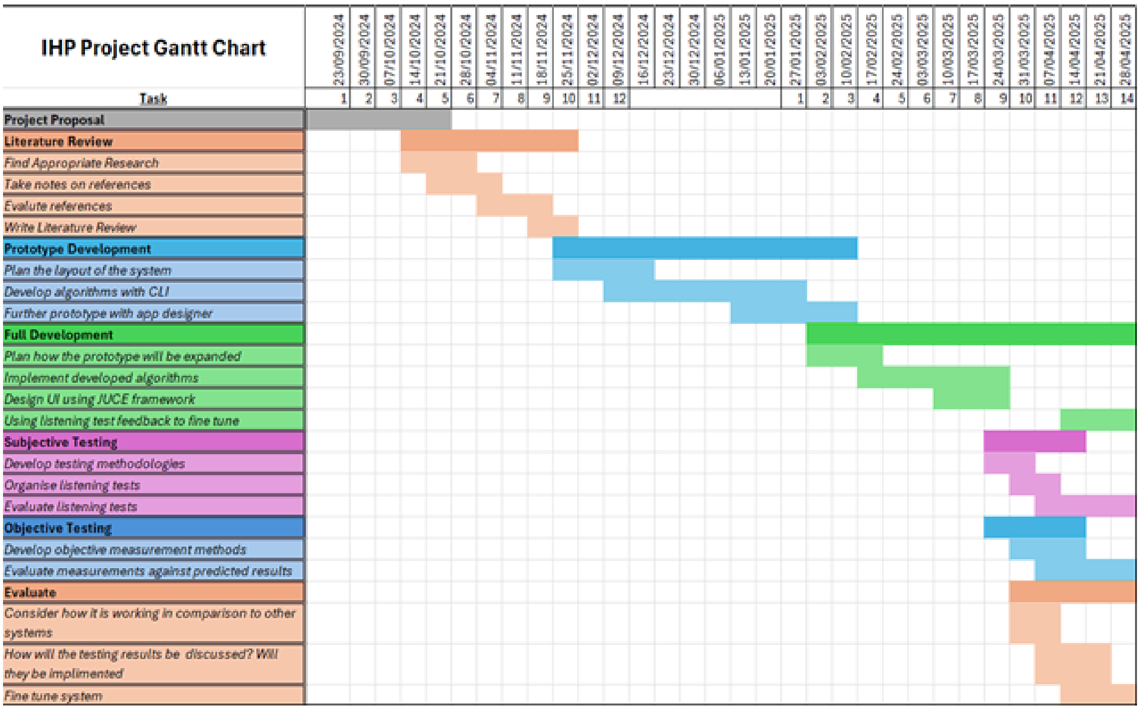

Figure A.1: *Gantt Chart showcase the Timeline of the project*

Appendix B

Value Tables

Reference Signals for Listening Tests

Table B.1: Reference Signals Used for Each Listening Test Question

Question	Reference Signal	Source Type	Original Material	Type
Question 1	Churchill Speech	Tape	Vintage speech	Vintage
Question 2	Golden Days	Tape	Radio drama	Vintage
Question 3	Artificial Speech Signal	Artificial	Synthesised speech	Clean
Question 4	Boy Girl Dancing	Tape	Vintage music	Vintage
Question 5	Secrets of Scotland Yard	Tape	Radio drama intro song	Vintage
Question 6	Self-Recorded Song	Artificial	Modern home-recorded song	Clean
Question 7	FatMan	Tape	Vintage music (FatMan)	Vintage
Question 8	Ten Little Fingers and Toes	Vinyl	Vintage children's song	Vintage
Question 9	Artificial Speech Signal (repeat)	Artificial	Synthesised speech	Clean

Appendix C

Code Listings

Time-Domain Transient Detection

Listing C.1: MATLAB function to detect and process transient artefacts in the time domain.

```
function out = processChannelTD_combined(sig, Fs)
    % --- Step 1: Derivative-based detection ---
    diffSig = abs(diff(sig));
    windowLength = round(0.25 * Fs);    % 0.25 sec window
    localMeanDiff = movmean(diffSig, windowLength);
    ratioThreshold = 8;
    candidateDeriv = find(diffSig > ratioThreshold * localMeanDiff);

    % --- Step 2: Amplitude-based detection ---
    ampSig = abs(sig);
    ampWindow = round(0.2 * Fs);        % 0.2 sec window
    localMeanAmp = movmean(ampSig, ampWindow);
    ampFactor = 1.5;
    candidateAmp = find(ampSig(2:end) > ampFactor * localMeanAmp(2:end));

    % --- Step 3: Intersection of criteria ---
    candidateIdx = intersect(candidateDeriv, candidateAmp);
    if isempty(candidateIdx)
```

```
        out = sig;
        return;
    end

    % --- Step 4: Expand candidate indices to mark regions ---
    windowSize = round(0.004 * Fs);  % 4 ms  total region
    mask = false(size(sig));
    for i = 1:length(candidateIdx)
        idx = candidateIdx(i);
        startIdx = max(1, idx - windowSize);
        endIdx = min(length(sig), idx + windowSize);
        mask(startIdx:endIdx) = true;
    end

    % --- Step 5: Identify contiguous segments ---
    dmask = diff([false; mask; false]);
    segStarts = find(dmask == 1);
    segEnds   = find(dmask == -1) - 1;
    maxSegmentLength = 5 * windowSize;
    %      further processing
end
```

Discrete Wavelet Transform (DWT) Step

Listing C.2: Single-level DWT via circular padding, convolution, and downsampling.

```
function [cA, cD] = noise_dwt_step(signal, dec_lo, dec_hi)
    signal = signal(:);
    % circular extension padding
    filter_len = length(dec_lo);
    pad_len = filter_len - 1;
    padded_signal = [signal(end-pad_len+1:end); signal; signal(1:pad_len)];

    % convolution
```

```
    cA_full = conv(padded_signal, dec_lo, 'valid');
    cD_full = conv(padded_signal, dec_hi, 'valid');

    % downsample by 2
    cA = cA_full(1:2:end);
    cD = cD_full(1:2:end);

    % equalise length
    min_len = min(length(cA), length(cD));
    cA = cA(1:min_len);
    cD = cD(1:min_len);
end
```

Inverse Discrete Wavelet Transform (IDWT) Step

Listing C.3: Single-level IDWT via upsampling, convolution, and circular cropping.

```
function reconstructed = noise_idwt_step(cA, cD, rec_lo, rec_hi)
    cA = cA(:); cD = cD(:); rec_lo = rec_lo(:); rec_hi = rec_hi(:);
    len = min(length(cA), length(cD));
    cA = cA(1:len); cD = cD(1:len);

    % upsample
    up_cA = zeros(2*len,1); up_cD = up_cA;
    up_cA(1:2:end) = cA;
    up_cD(1:2:end) = cD;

    % convolution with synthesis filters
    rec_A = conv(up_cA, rec_lo, 'full');
    rec_D = conv(up_cD, rec_hi, 'full');
    combined = rec_A + rec_D;

    % circular crop to original upsampled length
    startIdx = length(rec_lo);
```

```
    reconstructed = combined(startIdx : startIdx + length(up_cA) - 1);
end
```

Thresholding Methods

Listing C.4: Apply universal, level-dependent, or adaptive soft-threshold to each wavelet band.

```
% coeffs: cell array of detail bands {cD1, cD2,     }
for i = 1:level
    sigma = median(abs(coeffs{i})) / 0.6745;
    switch thresholdMethod
        case 'universal'
            threshold = sigma * sqrt(2*log(length(coeffs{i})));
        case 'adaptive'
            threshold = (1/sqrt(i)) * sigma * sqrt(2*log(length(coeffs{i})))
        case 'level-dependent'
            threshold = sigma * sqrt(2*log(length(coeffs{i}))) / (i+1);
        otherwise
            error('Unknown threshold method: %s', thresholdMethod);
    end
    % soft thresholding
    mask = abs(coeffs{i}) > threshold;
    temp = ones(size(coeffs{i}));
    temp(mask) = max(0, 1 - (threshold^2 ./ (coeffs{i}(mask).^2 + eps)));
    coeffs{i} = coeffs{i} .* mask .* temp;
end
```

Custom Modulus-Maxima Transient Detection

Listing C.5: Local z-score + clustering modulus-maxima detector.

```
function modmax_mask = custom_modmax_zscore(cD, windowLen, zThreshold, cluste
```

```
abs_cD = abs(cD);
% 1) candidate peaks by derivative sign
diff_f = [diff(abs_cD);0];
diff_b = [0;diff(abs_cD)];
cMask = (diff_b >0) & (diff_f <=0);
cand = find(cMask);

% 2) local z score
candZ = zeros(size(cand));
for k=1:numel(cand)
  idx = cand(k);
  w0 = max(1, idx-floor(windowLen/2));
  w1 = min(numel(abs_cD), idx+floor(windowLen/2));
  lv = abs_cD(w0:w1);
  med = median(lv);
  mad = median(abs(lv-med));
  sigma = max(mad/0.6745, eps);
  candZ(k) = (abs_cD(idx)-med)/sigma;
end

% 3) threshold & cluster suppress
keep = candZ>=zThreshold;
cand = cand(keep); candZ = candZ(keep);
final = [];
while ~isempty(cand)
  [~,iMax] = max(candZ);
  best = cand(iMax);
  final(end+1) = best;  %#ok<AGROW>
  d = abs(cand-best)<clusterWindow;
  cand(d)=[]; candZ(d)=[];
end

% 4) output mask
modmax_mask = false(size(cD));
```

```
    modmax_mask(final) = true;
end
```

Appendix D

Plots and Evaluation Results

Subjective Listening Test Results

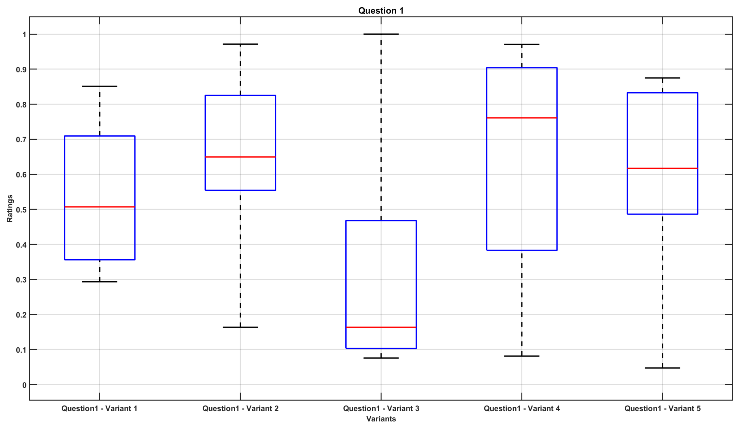

Figure D.1: *Box plot of listener ratings for Question 1.*

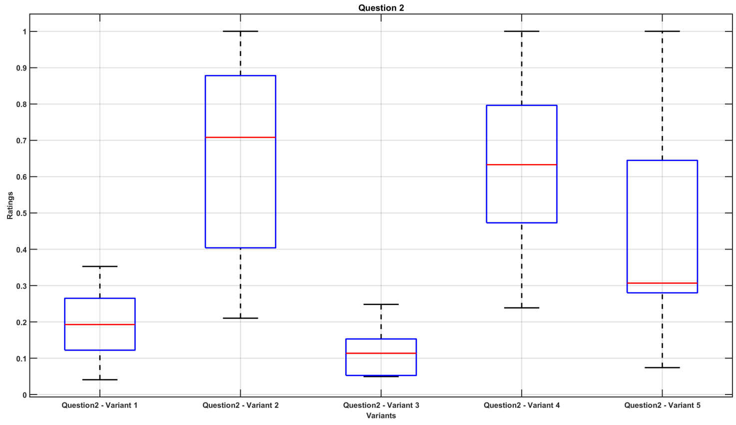

Figure D.2: *Box plot of listener ratings for Question 2.*

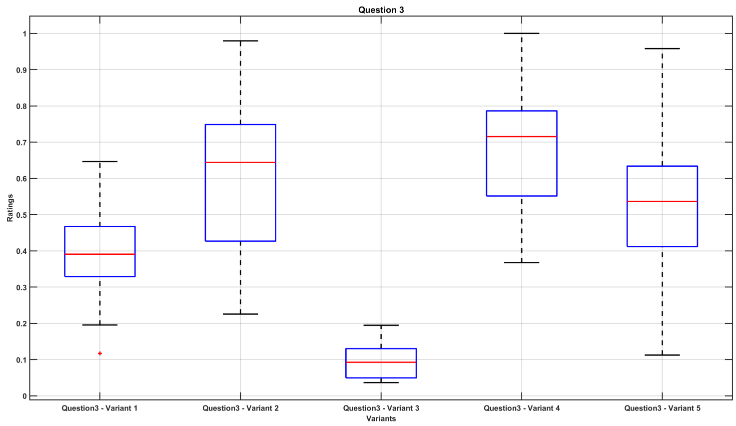

Figure D.3: *Box plot of listener ratings for Question 3.*

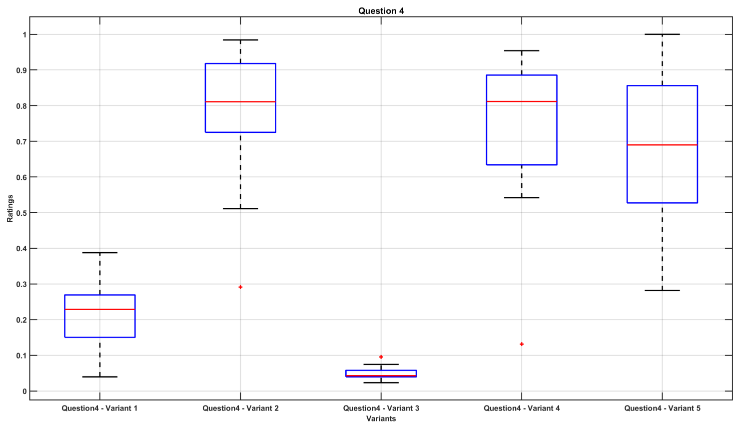

Figure D.4: *Box plot of listener ratings for Question 4.*

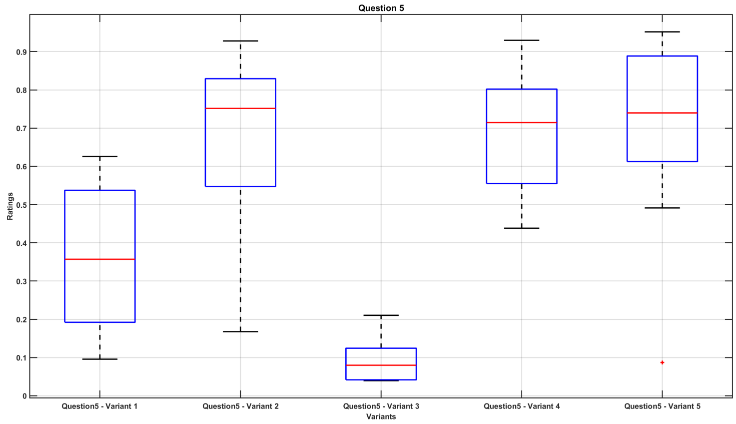

Figure D.5: *Box plot of listener ratings for Question 5.*

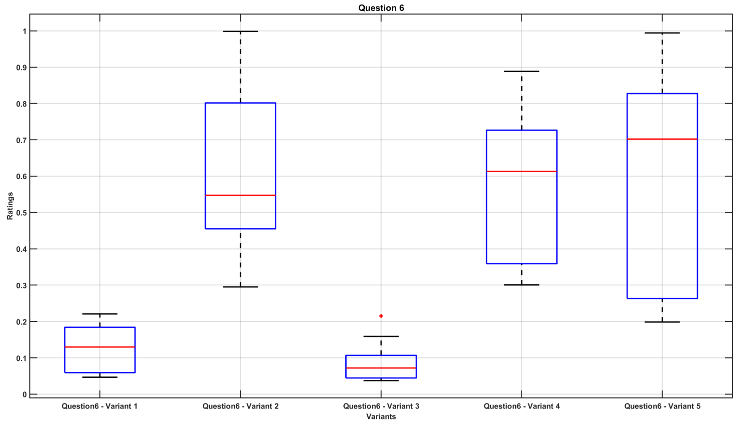

Figure D.6: *Box plot of listener ratings for Question 6.*

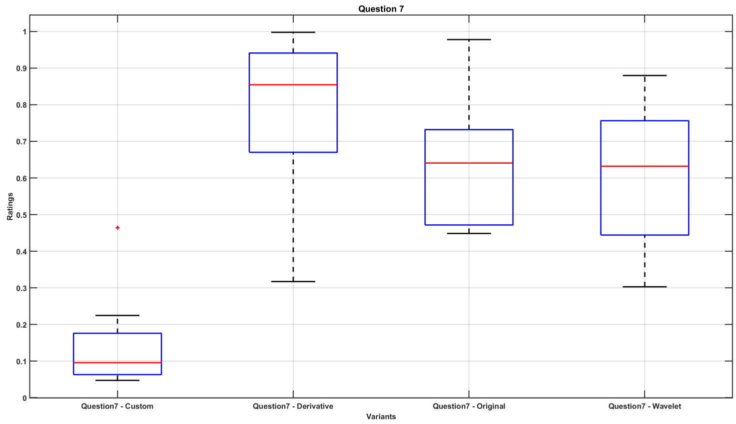

Figure D.7: Box plot of listener ratings for Question 7.

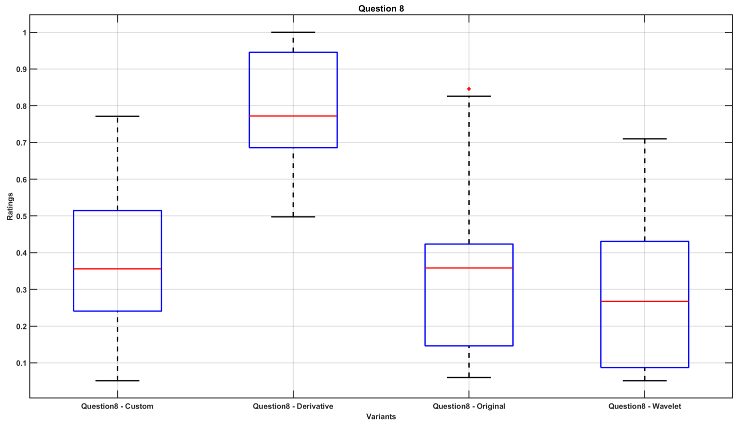

Figure D.8: Box plot of listener ratings for Question 8.

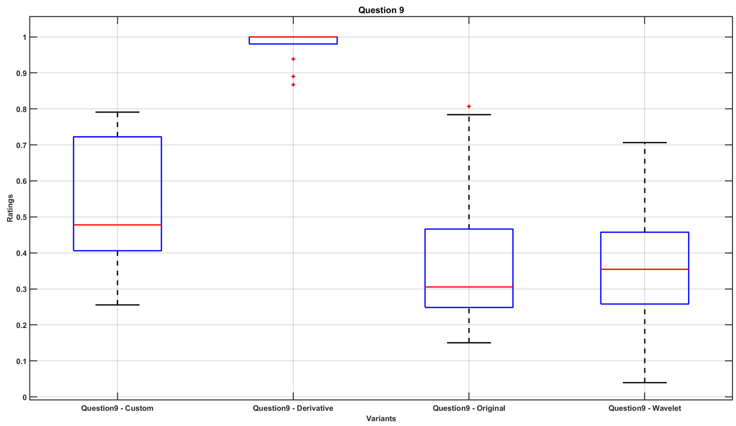

Figure D.9: Box plot of listener ratings for Question 9.