Griffith UNIVERSITY

GOLD COAST CAMPUS

School of Information Technology

Non-Speech Environmental Sound Classification System for Autonomous Surveillance

Michael Cowling,
Bachelor of Information Technology (Honours)

March, 2004

Submitted in fulfillment
of the requirements for the degree of

Doctor of Philosophy

in the

Faculty of Engineering and Information Technology,
Griffith University, Gold Coast Campus.

Principal Supervisor: Dr. Renate Sitte
Associate Supervisor: Dr. John Thornton

Abstract

Sound is one of a human beings most important senses. After vision, it is the sense most used to gather information about the environment. Despite this, comparatively little research has been done into the field of sound recognition. The research that has been done mainly centres around the recognition of speech and music.

Our auditory environment is made up of many sounds other than speech and music. This sound information can be taped into for the benefit of specific applications such as security systems. Currently, most researchers are ignoring this sound information.

This thesis investigates techniques to recognise environmental non-speech sounds and their direction, with the purpose of using these techniques in an autonomous mobile surveillance robot. It also presents advanced methods to improve the accuracy and efficiency of these techniques.

Initially, this report presents an extensive literature survey, looking at the few existing techniques for non-speech environmental sound recognition. This survey also, by necessity, investigates existing techniques used for sound recognition in speech and music. It also examines techniques used for direction detection of sounds.

The techniques that have been identified are then comprehensively compared to determine the most appropriate techniques for non-speech sound recognition. A comprehensive comparison is performed using non-speech sounds and several runs are performed to ensure accuracy. These techniques are then ranked based on their effectiveness. The best technique is found to be either Continuous Wavelet Transform feature extraction with Dynamic Time Warping or Mel-Frequency Cepstral Coefficients with Dynamic Time Warping. Both of these techniques achieve a 70% recognition rate.

Once the best of the existing classification techniques is identified, the problem of uncountable sounds in the environment can be addressed. Unlike speech recognition, non-speech sound recognition requires recognition from a much wider library of sounds. Due to this near-infinite set of example sounds, the characteristics and complexity of non-speech sound recognition techniques increases.

To address this problem, a systematic scheme needs to be developed for non-speech sound classification. Several different approaches are examined. Included is a new design for an environmental sound taxonomy based on an environmental sound alphabet. This taxonomy works over three levels and classifies sounds based on their physical characteristics. Its performance is compared with a technique that generates a structured tree automatically.

These structured techniques are compared for different data sets and results are analysed. Comparable results are achieved for these techniques with the same data set as previously used. In addition, the results and greater information from these experiments is used to infer some information about the structure of environmental sounds in general. Finally, conclusions are drawn on both sets of techniques and areas of future research stemming from this thesis are explored.

Table of Contents

List of Figures

List of Figures (cont.)

List of Tables

Statement of Originality

This work has not previously been submitted for a degree or diploma in any university. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made in the thesis itself.

Michael Cowling

Acronyms

ANN	Artificial Neural Network	k-NN	k-Nearest Neighbour
ASA	Auditory Scene Analysis	LMS	Least Mean Square
BPA	Back Propagation Algorithm	LPC	Linear Predictive Coding
CASA	Computational Auditory Scene Analysis	LTS	Long-Term Statistics
CSP	Crosspower-Spectrum Phase	LVQ	Learning Vector Quantization
CWT	Continuous Wavelet Transform	MFCC	Mel-Frequency Cepstral Coefficients
DCT	Discrete Cosine Transform	MLE	Maximum Likelihood Estimation
DTW	Dynamic Time Warping	MLP	Multi-Layer Perceptron
EM	Expectation-Maximization	MSE	Mean Square Error
FFT	Fast Fourier Transform	NN	Neural Network
FT	Fourier Transform	PCA	Principle Component Analysis
FWT	Fast Wavelet Transform	PLP	Perceptual Linear Prediction
GMM	Gaussian Mixture Model	SOM	Self-Organising Map
HCC	Homomorphic Cepstral Coefficients	STFT	Short-Time Fourier Transform
HMM	Hidden Markov Model	SVM	Support Vector Machine
HRTF	Head-Related Transfer Functions	TDE	Time Delay Estimation
IAD	Interaural Auditory Difference	VQ	Vector Quantization
ITD	Interaural Time Difference	WVD	Wigner-Ville Distribution

Chapter 1

Introduction

It has long been a goal of researchers around the world to build a computer that acts like a human being. The research of Brooks [Brooks98] is an example of developing human-like movement in robots. However, another subset of this research is to develop machines that have the same sensory perception as human beings. This work finds its practical application in the wearable computer domain (e.g. certain cases of deafness where a bionic ear (cochlea implant) cannot be used) as well as in the development of robots that can perform human-like sensory tasks (such as security).

Human beings use a variety of different senses in order to gather information about the world around them. If we were to list the five classic human senses in order of importance, it is generally accepted that we would come up with the sequence:

1. Vision
2. Hearing
3. Touch
4. Smell
5. Taste

Vision is undoubtedly the most important sense with hearing being the next important and so on. However, despite the fact that hearing is a human beings second most important sense, it is all but ignored when trying to build a computer that has human-like senses. The research that has been done into computer hearing revolves around the recognition of speech and music, with little research done into the recognition of non-speech environmental sounds.

This chapter will revise the relevant background and discuss the motivation of this thesis. It will also give an aim and a hypothesis related to that aim. Finally, it will discuss the domain of this thesis and the scope within that domain.

1.1 Problem Statement & Research Question

Research Question

How can I develop a computer system that can recognise non-speech environmental sounds, for the purpose of surveillance?

This research question develops from the problem explained in the introduction, that non-speech sound recognition research has been neglected in favour of the more fashionable speech recognition or speaker identification research. More formally, it can be stated as follows:

Problem Statement

Considering that the auditory component of the world is made up of not only speech, but also many other sounds, it is important that a computer can recognise and classify not only speech, but also the other common sounds in an environment. Areas such as hearing aid technology and security systems can benefit from research into a system that can identify non-speech sounds.

1.2 Aim

Aim

To develop an efficient system that can recognise non-speech environmental sounds, with the intent of installing this system in a self-contained sound sensing system for autonomous surveillance.

In order to facilitate the development of this system, several steps were carried out:

1. Investigate existing recognition techniques used in other domains (i.e. Speech recognition and Music recognition) and determine whether these techniques can be applied to non-speech sound recognition.
2. Comprehensively compare these techniques for the domain of non-speech in order to determine the most efficient algorithms for this domain.
3. Investigate advanced techniques to accelerate the identification of non-speech sounds. This will be done using several techniques, including a non-speech sound taxonomy based on the source of a sound and separated according to the physical state of the source (i.e. solid on liquid, liquid on liquid etc.) and its material composition (metal, wood etc.). This kind of taxonomy has not been developed previously. Other advanced techniques will also be presented.

I will elaborate on each of these points. Within previous research [Cowl00, CowlSit00, CowlSit01], I identified some techniques that could be used for non-speech sound identification and direction detection, and implemented them as an exploration case. However, no comparison was made between the selected techniques and other techniques in order to determine the most appropriate technique for the recognition of non-speech sounds.

Therefore, the first step in this research was to identify alternative techniques that could be used for recognition of non-speech sounds and compare them. I have done this by turning to two domains of research that have had greater focus over the years: speech recognition and music recognition. These techniques provide a starting point when looking for techniques that could be used for non-speech sound recognition. A comprehensive review of these techniques was carried out (in the literature review) and the techniques that can be used for non-speech sound recognition were identified.

After these techniques had been identified, each was implemented as a prototype and tested on specific data sets. This comparative surveying helps to determine the performance of each technique.

Based on the results from this comparison, additional advanced techniques (such as the implementation of a non-speech sound taxonomy and the development of an environmental sound alphabet) were investigated. These techniques were used in conjunction with results from the previous experiments to present a more generalized method for a larger corpus of sounds. This is because while speech is bounded to a typical range of frequencies, environmental sounds go beyond these boundaries. The frequency space is vast and as the number of sounds in a system increases, finding the matching pattern that identifies a sound is like finding the proverbial needle in a haystack. A sound taxonomy tree provides a systematic and efficient way to approach this problem.

1.3 Hypothesis

Premises:

1. Environmental sound identification is a classification problem, where an unknown sound is matched to known sound patterns for similarity and identification.
2. Environmental sounds are vast (uncountable), they can be produced by the interaction of media in *any* physical state, in any frequency range.
3. Human speech is a subset of environmental sounds always produced by the interaction of air and the vocal cords.

Hypothesis

If I can find a systematic way to identify environmental sounds, I could increase the efficiency of environmental sound identification for the purpose of security surveillance. A system can be developed that will recognise a large corpus of environmental sounds. This system will use a structured classification technique (sound taxonomy) to improve classification accuracy and speed.

Previous work has shown that it is possible to develop a system that will recognise non-speech environmental sounds [Cowl00, CowlSit00, CowlSit01]. However, it is surmised that as the amount of sounds trained in this system increases, the classification accuracy of the system will decrease. Also, as more sounds are trained in the system, the time taken for a classification will become much longer. This is because of the infinite amount of non-speech sounds in our environment.

Therefore, in order to develop an accurate and efficient system for non-speech sound recognition, advanced techniques must be used to overcome the problem of pattern matching against an almost infinite set of patterns. A technique that shows promise is the technique of an environmental sound taxonomy. This taxonomy looks at classifying sounds based on their physical properties (liquid to liquid, solid to liquid etc). Classification decisions are also made based on recognition of different materials involved in the sound (wood, stone, glass etc.). This method of classification is expected to improve accuracy and efficiency.

1.4 Domain & Scope

The aim of this research is to develop a self-contained sound sensing system. This system will consist of two parts: a non-speech sound recognition subsystem and a source localization subsystem. The purpose of the research is to develop the sound sensing (recognition) system for a security surveillance domain.

Using the results of this research, it is envisioned that an autonomous mobile robot can be built that should be able to navigate and explore a room, listening for sounds. When a sound is heard, the robot can identify the type of sound. Depending on the sound, it can then move towards or away from the sound and take the appropriate action (such as taking a picture or calling the police). For example, the system could detect a window being broken in the right hand side of the house and then detect footsteps in the same section of the house. It could then transmit this information to the authorities where it could be used to dispatch a police unit to investigate (although this intelligent transmission is not in the scope of this research).

The fundamental thinking behind this domain is as a limiting factor on the types of sounds that should be recognised by the sound sensing system. It is obvious that building a system that can recognise all non-speech sounds in our environment is not only impractical but also almost impossible using today's technology. Therefore, the security domain limits the sounds to be recognised by the system to a much smaller finite set (but a set still much larger than that used in speech recognition).

1.5 Research Contribution

The research in this thesis contributes to research into non-speech sound. It makes several contributions to the research community that have not been made previously.

Firstly, this thesis presents an *Environmental Sound Taxonomy*. While taxonomies exist for other domains, no taxonomy has been developed for the area of non-speech sound recognition. Also, the development of an environmental sound taxonomy required the development of a novel environmental sound alphabet as a means to build the hierarchy.

Secondly, this thesis contributes *a comprehensive comparison of existing speech and musical instrument recognition techniques* in the domain of non-speech sound recognition. This comparison is presented in a thorough fashion for all of the common recognition techniques in these domains. Comparison of different speech and musical instrument recognition techniques has been performed before, but not as comprehensively, and not in the domain of non-speech sound recognition.

Finally, this thesis presents an *several advanced algorithms for non-speech environmental sound recognition.* The area of non-speech is an area that has been all but neglected in the research community. These algorithms present a basis on which further research into refining the techniques can be built. Furthermore, results from this comparison show some of the general characteristics of environmental sounds, which will be invaluable for future research.

1.6 Organisation of this Document

The remainder of this thesis will be organised into six sections.

Chapter 2 will cover previous literature in the areas of non-speech sound recognition. It will also look at relevant work in the areas of speech and music recognition.

Chapter 3 will discuss the hypothesis and present possible solutions to the problems highlighted by the literature review.

Chapter 4 will cover the methodology proposed to begin to answer the research question and will include descriptions of the algorithms and techniques used.

Chapter 5 will present results of the research and draw conclusions from these results.

Chapter 6 will present the design for several advanced techniques that aim to reduce the pattern matching search space for non-speech environmental sounds.

Chapter 7 will present results from the application of these advanced techniques and draw conclusions from these results.

Chapter 8 will conclude this document and discuss future research directions that could be motivated by this work.

Chapter 2

Literature Review

This chapter is split into four sections. The first section will review the basic theory required in order to understand the research area. It includes details on sound digitization, transforms, the structure of environmental sound from a human perspective and the basics of neural networks.

The second section will cover the area of non-speech environmental sound recognition. In this section, existing non-speech sound recognition techniques will be analysed. A comparative study will also be discussed identifying those speech and music recognition techniques that can be used for non-speech sound recognition. This section will also discuss time-frequency techniques and their applicability to the domain. Finally, it will also discuss the related area of source separation, also known as computational auditory scene analysis.

The third section will cover the area of source localisation or direction detection. Again, existing techniques for source localization will be discussed.

Finally, the fourth section will discuss the results from the literature review and explain the problems that these techniques face when applied to environmental sounds. These problems will then be addressed in Chapter 3 and Chapter 4.

2.1 Basic Theory

Sound is generated when an object (such as a tuning fork) causes a disturbance in the density of the medium in which it resides (usually air) [Tipler91]. This disturbance propagates through the medium. When the disturbance reaches a human ear, it is converted into electrical signals that the brain interprets as sound. This disturbance takes the form of a wave (Figure 1), with the amplitude (or height) of the wave representing the amount of movement of molecules in the medium and the frequency (or length) of the wave representing the amount of time occurring before the waveform repeats.

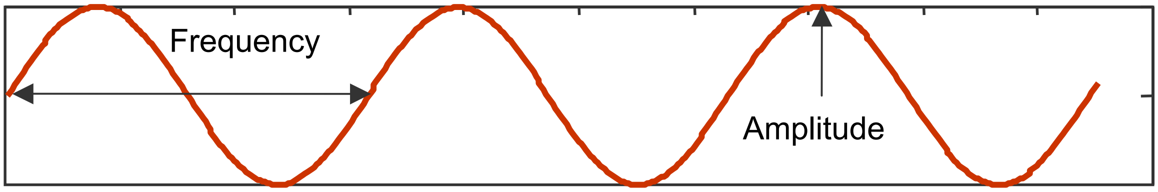


Figure 1 – A Simple Waveform

2.1.1 Sound Digitization

In order to investigate or analyse the processes involved in non-speech sound identification as performed by a computer, it is necessary to review the basics of how this analog sound information is stored in the computer. The process of transferring analog sound signals into digital bits to be stored on the computer is known as analog-to-digital conversion or the *digital encoding process* [Kefau99].

A true analog signal can be represented as a continuous waveform. However, to store this waveform in the computer, values of the wave are taken at regular intervals. This process is called sampling. It is the process by which a continuous-time variable is measured as distinct, separate instants of time. By sampling, the smooth curve from the measurements is replaced by a finite set of numbers. Each pulse amplitude is then rounded to one of a finite number of levels, in this case into an eight-bit number. This process is called quantization. In essence, the process of sampling and quantization is an encoding process that converts the analog waveform into a binary representation.

The sampling period depends on the nature of the signal (Figure 2). However, for a periodic signal it must be such that each lobe of the sinusoid (waveform) is sampled at least once [Palm00]. A Fourier series can then represent the signal in the frequency domain. The plot of the Fourier series versus the frequency is called the *spectrum* and is characteristic for a sound.

The specifics of this sampling process will be explained in more detail: Frequency is measured in Hertz, which represents the amount of times per second a sample point is taken. For instance, a sample rate of 4Hz means a sample point is being taken 4 times a second. Nyquist discovered that a minimum of two sample points per cycle is required in order to determine the correct shape of a waveform. Since the limit of human hearing is 22KHz, a waveform must be sampled at 44KHz in order to preserve all the information a human can hear [Steig96]. This means that, in applications such as the subject of this thesis (where frequencies can be within this entire range), a waveform will be sampled 44,000 times per second and each sample point will then be stored in the computer as a data point.

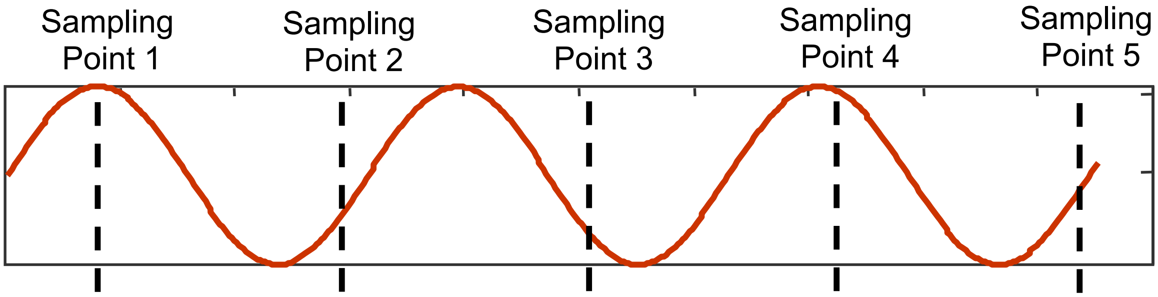


Figure 2 – Sampling a Wave

In the process of quantization that follows sampling, the amplitude of the signal at each sampling point is stored [Kefau99]. If we were to use a scale to measure amplitude with a +1 and –1 representing the maximum amplitude of the waveform, we could define the amplitude of the wave using a set of discrete values at each point. For instance, in Figure 3, sampling point 1 has a value of 0.000, sampling point 2 a value of 0.625 and so on. This is called the quantization interval.

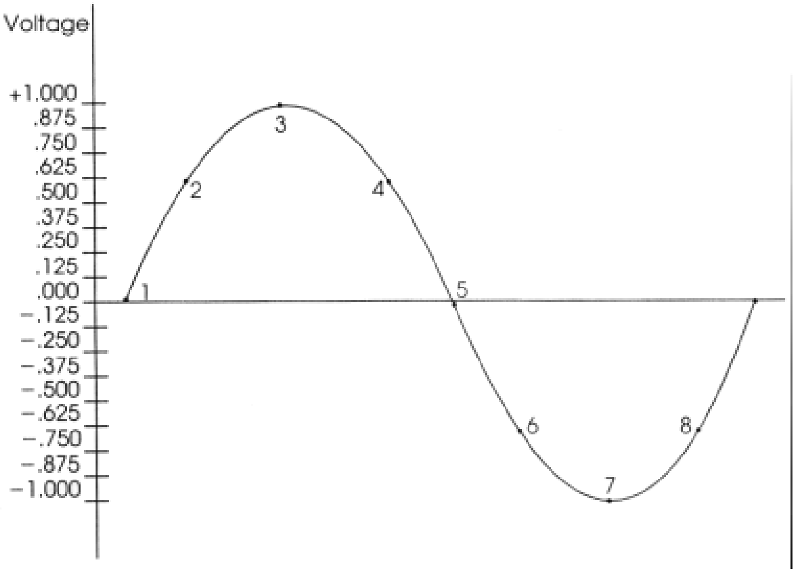


Figure 3 - Quantization in volts [Kefau99]

However, due to the fact that computers can only store information at the lowest level in binary, this decimal value is converted into a binary value (by assigning each value along the scale a binary equivalent). For instance, 0.25 becomes 01 binary, 0.5 becomes 10 binary and 1.0 becomes 11 binary. The amount of bits used in the scale is referred to as the resolution, and is measured in bits per word (amount of bits used to store the amplitude of each sampling point). In the example given, the resolution would be 2 bits per word, since 2 bits are used to represent the value of each sampling point. A problem quickly becomes apparent with this method. Using the scale above, how would the sampling point value of 0.625 be stored? Due to the amount of bits used in the scale, the value must be rounded up or down to the nearest number with a binary equivalent (in this case, rounded down to 0.5 or 10 binary). The process of rounding produces a *quantization error*. Quantization error is measured as the amount required to bring the value in line with one of the quantization intervals (in our example, the quantization error is 0.125, which is the amount rounded down from 0.625 to get 0.5).

2.1.2 Transforms & Windows

Once a sound wave has been sampled into a computer, it can be manipulated in various different ways (this reveals different types of information). One of the most important manipulations that can be performed on a sound wave is a Fourier transform [Shie99]. It has been studied and used for many decades, and its numerical methods are well developed. By representing the sound samples as a Fourier series, we simplify numerical manipulations of otherwise complex calculations. The Fourier transform process involves breaking a complex wave down into its characteristic sinusoidal components. This process allows the wave to be easily analysed. A Fourier transform performs this function by obtaining the amplitude for a given frequency. By applying different frequencies to a wave, it can be split into a set of discrete frequency components. These values can then be more easily analysed. Many other methods such as Laplace Transform, Fast Fourier Transform, Discreet Fourier Transform, Gabor expansion and Short-Time Fourier transform can also be used to analyse a given wave in both the frequency and time domains, but all work on a similar fundamental concept as the Fourier transform [Shie99].

For sound recognition tasks, a common alternative to the Fourier Transform is the Discrete Cosine Transform (DCT) [Acken99]. The DCT can be compared to an FFT on a one-dimensional sequence. Compared to an FFT, a DCT removes some erroneous high-frequency components that are introduced into the spectrum with an FFT. These components are introduced due to the method by which an FFT determines frequency. When an FFT is performed on a sequence, it is assumed that the sequence is repeated periodically. However, the joining of these sequences produces a glitch. This glitch introduces these erroneous components. In comparison, a DCT reflects a signal before extending it periodically. This creates a smoother transition between each sequence and prevents the production of the high-frequency components.

In addition to transforms, windowing techniques can also be applied to a signal in order to improve pattern recognition (that is, the ability to recognise patterns within the signal, which is important in areas such as the one covered in this thesis). A number of speech/speaker recognition techniques use overlapping windows on a signal in order to improve processing. When windowing a signal, the standard window forms a rectangle. However, this kind of window produces a signal with a series of ripples in the frequency response. Therefore, it is common to use a tapered windowing technique in order to remove these ripples. The most common technique used in speech recognition is the Hamming window technique [Carti00]. This technique uses a tapered window that causes the main lobe of the window spectrum to increase in width. In turn, this causes more rounding of the signal and therefore decreases ripples in the signal. However, this is at the expense of making the characteristics at the edge of the window less sharp. This can be compensated to some extent by increasing the length of the window, producing more acceptable results than a standard rectangular window.

2.1.3 The Sound Recognition Process

Most recognition and classification problems are implemented using a three-stage process.

1. Data Preprocessing
2. Feature Extraction
3. Classification

The sequence of these three stages is shown in Figure 4.

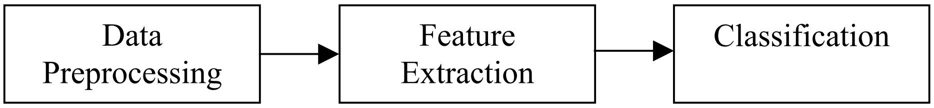


Figure 4 - Traditional Classification Sequence

Data Preprocessing is the first step in the process. This step differs depending on the classification task being performed. For instance, for handwriting recognition, this step involves splitting each sentence up into separate words and letters and performing other initial tasks (such as correcting the slant inherent in many individuals handwriting).

Data Preprocessing for sound recognition (including speech recognition), involves taking a sound from the environment and loading it into a computer. Typically, this is done using a microphone. In addition, a computer represents sounds in a digital format, which means that the analog signal produced by a microphone has to be converted into a digital format via sampling and quantization techniques [Kefau99].

Feature Extraction is then performed to reduce the huge data set produced in the previous step. Feature extraction involves selecting pieces of the input data that uniquely characterize that information [Fuku90]. The choice of features is up to the researcher and is based on their belief of which feature most accurately characterises a sound. Again, this step differs depending on the classification task being performed. For sound recognition, many techniques have been used for feature extraction, from the simple (identifying all of the frequencies in a sound), to the extravagant (modeling of the feature extraction of the human auditory system). However, no matter what the technique, sound researchers agree that feature extraction is the most difficult part of the recognition process.

Feature extraction can be performed at three levels of understanding, as shown in Table 1 [Faich00, Gonz97]. *Statistical feature extraction* works directly on the data from the environment. For instance, the colour of each pixel in a picture could be measured. This information could then be used as a feature for classification and testing. *Syntactical feature extraction* expands upon statistical feature extraction by understanding the structure of the object. For instance, a speech recognition system could use a syntactical technique to split the speech into separate words (requiring the system to understand the concept of a word, or at least the syntax of a speech signal (spaces between words)). Finally, *semantic feature extraction* requires prior knowledge of an object. For instance, a text recognition system may use a dictionary to process explanations of each word.

TABLE 1. LEVELS OF FEATURE EXTRACTION

Statistical	Data Based
Syntactical	Data with Structure
Semantic	Prior Knowledge of Environment

All of these levels of understanding can (and should) be combined together to produce a system that performs good feature extraction. For instance, a speech recognition system could use statistical techniques to identify when speech is being fed into a microphone (as opposed to silence). Syntactical techniques could then split the speech into separate words. Each word could then be recognised (e.g. using a statistical technique) and then a semantic technique could be used to interpret each word using a dictionary.

Classification is the third step in the recognition process. Classification involves taking the features generated in the previous step and linking each feature to a particular classification (a form of pattern recognition) [Schal90]. Again, this can be done in many ways. For sound recognition, many techniques have been used, including Hidden Markov Models, Neural Networks and Reference Model Databases (as used with Dynamic Time Warping). All of these techniques use a training/testing paradigm. *Training* gives the system a series of examples of a particular item, so it can learn the general characteristics of that item. Then, when *Testing* is performed, it can identify the class of the item being tested. As an optional step, classification can also involve fuzzy logic processing [Setnes99]. This is done prior to training in order to establish some correlation between the training set and the final classifications.

Classification does face one hurdle. It is important to ensure that the testing and training sets are recorded in the same conditions in order to get optimum results. In an analysis of training and testing techniques for speech recognition, Murthy explains how training data must be collected from within a variety of different environments to assure that a representative set of training data is stored in the database [Murthy99]. Murthy introduces the use of a filter bank to remove erroneous environmental sounds from the sound sample to ensure these do not effect classification. Speech recognition/Speaker identification researchers typically refer to techniques that attempt to correlate training and testing data as "robust" techniques [Lilly00, YuanX99]. Robust recognition techniques are most useful if noise and other factors affect the training data.

2.1.4 Aspects of Environmental Sounds

There have been few attempts to define the term "environmental sound". However, Vanderveer defines four general but important points that may help to identify an environmental sound [Vander79].

1. It is produced by real events.
2. It has meaning by virtue of the causal events.
3. It is more complicated than laboratory-generated sounds such as pure tones.
4. It is not part of a communication system such as speech.

This definition shows that Vanderveer believes that there is a clear distinction between speech and non-speech sounds. However, human hearing seems to make little distinction between speech and non-speech sounds.

Ballas and Howard [Ballas87] discuss how they believe that a human perceives environmental sounds as equivalent to a form of language. Of interest in the article, Ballas and Howard make reference to a lack of an equivalent to the phonetic alphabet for environmental sounds. They suggest that this is because speech is produced from a limited set of different actions by the vocal mechanisms of a human being whereas environmental sounds can be produced from a much wider range of sources.

Ballas and Howard detail how semantics (understanding of the context of a sound) are important in recognising the sound, regardless of whether that sound is speech or non-speech. They then explain how this equates to two forms of processing in the human auditory system: top-down and bottom-up. Top-down is related to the meaning (semantics) of the sound whereas bottom-up is related to the statistical features of the sound.

Ballas and Howard discuss how environmental sounds are typically described by their semantic meaning as opposed to their statistical properties. That is, a sound is described as “glass breaking” and not “a quick sound, with varying and yet constantly high-pitched elements”.

Ballas and Howard also show that a human beings confidence in correctly identifying a sound decreases as the amount of causes for that sound increase. They call sounds with a large amount of causes “sound homonyms”. They then suggest that just as the difference in speech homonyms such as “knight” and “night” cannot be determined without the context of a sentence, sound homonyms also need the context of other sounds in order to be identified by a human.

This theory is backed up by experiments related to the playback of several sounds to human beings. In these cases, the order of the sounds relates to a human being’s semantic interpretation of the sounds. For instance, if a clang sound is proceeded by a screeching sound, the semantic interpretation may be that the “auditory scene” just heard was that of a car crash (with the screech being interpreted as a tyre skid sound). On the other hand, if the same clang sound is combined with water dripping and a burst of air, the semantic interpretation is typically that of machine noise in a factory.

Finally, the theories of top-down processing suggest that the library of sounds that a human being has previously learnt directly relates to their ability to identify an environmental sound, just as it does for speech. If a human being has not previously heard a sound or series of sounds (such as the aforementioned car crash), it will not be recognised in such a way by that person. This idea of previously learnt ideas has applications in the area of computer sound identification and training of computers to recognise certain sounds.

2.1.5 Neural Networking Techniques

Neural Networks (NN) are a mathematical tool for finding best fit functions to given multivariate data. Although NN's were first created as a means for researchers to model the functioning human brain, they have gradually become a useful tool for analysis and classification of information [Haykin99]. The simplest NN consists of a single neuron with several inputs and a single output. A neuron is characterised by a series of numbers that are referred to as weights. As the neuron is given data (trained), weights are modified within that neuron according to a training algorithm. For the purpose of real world problems, a set of neurons is joined together into a network, which is then called a perceptron (Figure 5).

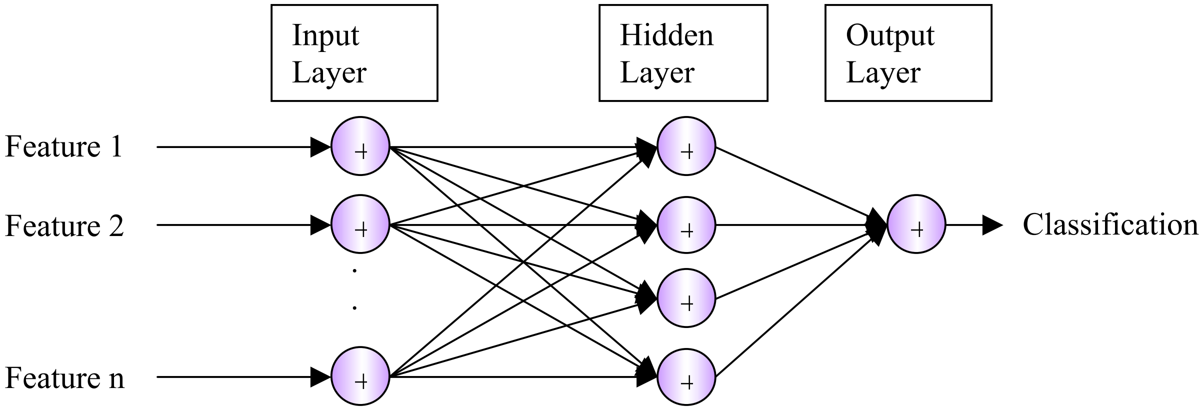


Figure 5 - An example Back Propagation Network.

A very common algorithm used within a perceptron is called the Back Propagation Algorithm (BPA) [Buhrke95]. The BPA is a supervised learning approach. This algorithm uses the same training and testing technique mentioned previously in order to classify results.

A NN can use a *supervised learning* approach to classify sound information [Haykin99]. First, Feature Extraction of the waveform is performed. The features are then given to the NN as inputs, with the user giving the correct output as well. The NN slowly adjusts itself to the data. New data can then be given to the NN and it will classify the data based on the previous information.

NN’s can also perform *unsupervised learning* by clustering data around expected features [Schal90]. One of the most common techniques for unsupervised neural networking is the Self-Organising Map (SOM), first developed by Kohonen [Kohon97]. It involves using a neural network to produce a map that shows how the different features cluster around each other. By applying some grouping rules to the map, it can be split into a set of classes that can be used for classification. One of the main advantages of this approach is that the classes need not be known prior to the training and testing of the neural network, due to the fact that the classes can be obtained from the self-organising map at the conclusion of the training.

The technique of using a NN for data storage and classification allows for the efficient storage of a large amount of information, due to the fact that once the information is given to the neural network for training, it can be forgotten. However, the problem with NN’s lies in the fact that the training process is sub-optimal [Buhrke95]. Since the neurons in a perceptron can only be adjusted by a slight variation in its weights, a NN does not give as exact a portrayal of the training data as if it was actually stored in a database and used directly. This means that the information in the NN can produce a wrong classification due to the way that the neurons have been weighted by the training data.

To understand this better, consider the difference between storing a textual explanation of a person’s facial features versus storing a picture of the person. No matter how well the textual description is written, the picture will always give a much clearer idea of what the person looks like. This is because features will always exist in the picture that are impossible to include in a textual description. This same logic can be applied to neural networks. A neural network stores the features of an object. While this will suffice in most cases, it does not equate to storing the actual object. This is because features will always exist in the object that are difficult to describe for the neural network.

2.2 Sound Recognition

Research into the field of non-speech sound recognition is sparse. As noted previously, the majority of auditory research is centred on the identification and recognition of speech signals. Those systems that do exist work on a very specialized domain or with only a few classes. Due to this, existing research into sound recognition is difficult to find. However, some research does exist. These results will be discussed in this section.

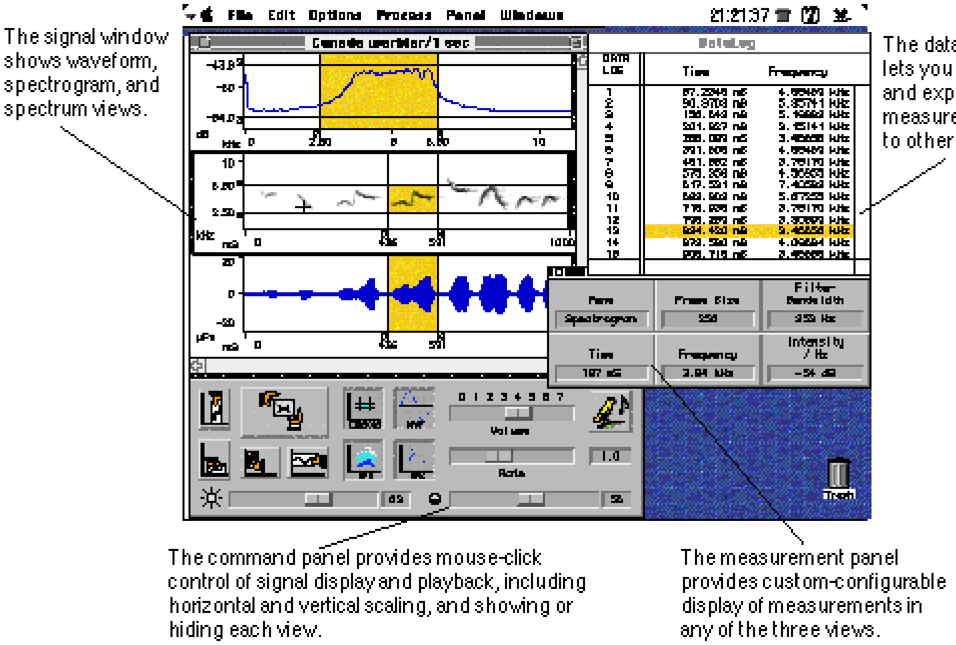
Due to this lack of non-speech sound recognition research, techniques for speech recognition and music recognition will also be discussed in the hope that these techniques could be adapted for use with non-speech sounds. Another recent research area, time-frequency feature extraction is also discussed with relation to its applicability to non-speech environmental sound recognition. Finally, a further research topic related to sound recognition called **Auditory Scene Analysis,** involving the separation of multiple sounds within an environment, will be discussed.

2.2.1 Existing Non-Speech Literature

This section will cover the literature related directly to the recognition of non-speech sounds.

Initial research into systems that perform non-speech sound recognition revealed a program that seemed suitable for environmental sound recognition. The Canary program was developed by the Cornell Lab of Ornithology and is designed to recognise bird song [Cornell01]. However, upon further scrutiny of the Canary program, it was found that it stopped just short of actually recognising the sound.

The Canary program (see Figure 6) analyses a signal given to it (of a bird song) and then allows the user to plot a spectrum of the signal. It also allows the user to perform a correlation analysis between two signals. However, the program lacks the ability to identify a signal as being the song of a specific type of bird. This has to be done either through visual inspection or by manual correlation with other frequency spectra of birds until a match is found. However, if the second method is used, the correlation still relies on visual inspection of a spectrogram representing the difference between the two signals.



DATA LOG	Time	Frequency
1	87.2349 mS	4.[illegible] kHz
2	90.9709 mS	5.35741 kHz
3	158.849 mS	5.[illegible] kHz
4	[illegible] mS	[illegible] kHz
5	[illegible] mS	3.[illegible] kHz
6	[illegible] mS	4.[illegible] kHz
7	[illegible] mS	3.75170 kHz
8	[illegible] mS	4.[illegible] kHz
9	617.531 mS	7.[illegible] kHz
10	[illegible] mS	5.[illegible] kHz
11	[illegible] mS	3.75170 kHz
12	[illegible] mS	3.[illegible] kHz
13	[illegible] mS	[illegible] kHz
14	[illegible] mS	4.[illegible] kHz
15	[illegible] mS	3.[illegible] kHz

Pane	Frame Size	Filter Bandwidth
Spectrogram	256	253 Hz

Time	Frequency	Intensity / Hz
197 mS	3.84 kHz	-54 dB

The signal window shows waveform, spectrogram, and spectrum views.

The data log lets you save and export measurements to other programs.

The command panel provides mouse-click control of signal display and playback, including horizontal and vertical scaling, and showing or hiding each view.

The measurement panel provides custom-configurable display of measurements in any of the three views.

Figure 6 - The Canary working environment.

From this example, it can be seen that non-speech sound identification is not a trivial exercise. It requires that a sound be recognised irrespective of differences in length and small changes in frequency.

Goldhor presents a technique that tries to tackle these problems. It uses a Mel frequency cepstral coefficient technique for feature extraction and a modified vector classification technique for classification [Goldh93]. However, as opposed to vector quantization, Goldhor uses the vectors in order to perform supervised clustering into classes. Goldhor also calculates mean and variance values for each sound class and uses a time warping technique to ensure all samples are a constant length.

Using this technique, Goldhor performed four tests on his database of 23 sounds. Throughout the four tests, Goldhor reports an accuracy of recognition close to 100%. Goldhor also notes that the problems with sound identification research may occur in sound separation and different environment issues. These issues are addressed in the section on Auditory Scene Analysis.

Hiyane also presents a signal processing based system for classification of five single impulsive sounds [Hiya00]. Single impulsive sounds are sounds created by the impact between objects. This system uses a training and testing technique based on the distinguishing features of a sound gathered at the peak and reverberation times. Unfortunately, Hiyane neglects to mention the specific technique used to classify the sound features. Hiyane notes that the recognition rate for this system is approximately 80%. He also notes similar problems to Goldhor with regards to multiple sound segregation and the fact that different sounds produce distinctly different waveforms.

Woodard presents a theoretical model for a system that uses a combination of linear predictive coding (LPC), vector quantization (VQ) and hidden Markov models (HMM) to classify three types of environmental sounds [Wood92]. Woodard uses a technique based on VQ product codes, where each index sequence in the VQ product code can be equated to a markov chain. However, the theoretical approach Woodard presents lacks comprehensive research data. An overall performance measure of 96% is given, but no information is available on the number of runs performed or the structure (and amount of overlap) of the test and training sets. In addition, classification of only three natural sounds could be considered to be statistically insignificant. Unfortunately, to this date, further review of the literature finds no further papers that would provide more detail in this area.

Interestingly, Dorken et. al present a uniquely different approach to the problem of environmental sound recognition [Dorken92]. Dorken et al use Knowledge-based signal processing methods in order to both recognise the sounds and separate them (a form of

Auditory Scene Analysis). This method works by comparison against a contour developed from a waveform. This contour is developed based on a short-time Fourier transform (STFT) of sections of the waveform and then selection of peaks within each STFT window. This approach is a novel approach that uses advanced signal processing and sound understanding approaches grouped together using knowledge-based techniques. However, because of this, the technique requires substantial effort in building up. It is computationally slow and unsuitable for a security system that requires fast response times.

Reyes-Gomez and Ellis [ReyesEl03] present a technique that uses cepstral coefficients for feature extraction combined with a clustering technique and HMM's. Reyes-Gomez and Ellis acknowledge the lack of natural basic units in HMM's. To combat this problem, they use a clustering technique or a GMM to generate clusters for different sounds. Sounds are split into "types" (such as "animal" sound, "machines" etc). The clusters are then used as states in the HMM, in the same way as phonemes are used as states. Using this technique, Reyes-Gomez and Ellis achieve a classification rate of 85% - 90% on their arbitrarily selected classes of sound (depending on clustering technique used). However, the applicability of the HMM is not fully explored (why not simply use the GMM for recognition?). In addition, their classification is performed using "types", so they have no defined way for their system to make a further, more refined classification, except by using traditional pattern recognition techniques.

Environmental Sound Recognition has also been performed in a much more limiting domain than those presented up to this point. A small amount of research has been performed (mainly for military applications) in identifying vehicles based on their acoustic emissions.

Liu presents an LVQ based technique for the recognition of ground vehicles (such as tanks) [Liu99]. Liu performs tests on the standard LVQ algorithm (as presented by Kohonen [Kohon97]) as well as two modified LVQ techniques: Tree Structure Vector Quantization (TSVQ) and Parallel TSVQ (PTSVQ). When using the PTSVQ technique on sounds already trained into the LVQ network, Liu obtains a 90% classification rate. However, when separate “unknown” test sounds are used, a recognition rate of 68% is achieved. In addition, Liu makes no mention of how many classes of vehicles were used or how the LVQ technique was selected.

Sampan also presents a ground vehicle recognition system [Sampan97]. In this system, Sampan tests several variations of two main techniques: multi-layer perceptron (MLP) neural networks and fuzzy algorithms. Again, Sampan performs tests using “ideal” data in the form of the Iris data set. When using the Iris data set, performance of all algorithms is close to 100%. Sampan also uses real data. For this data, Sampan takes five classes of ground vehicle and tries to classify test data into one of those five classes. For this test, each algorithm performs around 75% classification accuracy. However, it is unclear how the algorithms would perform if the amount of sounds were increased.

2.2.2 Speech Recognition

As mentioned previously, most pattern recognition techniques (including speech recognition) utilise three steps for recognition (Figure 4). These are:

1. Data Preprocessing
2. Feature Extraction
3. Classification

The first step, data preprocessing, has already been explained adequately in the **Basic Theory** section of this document. Data Processing for speech is performed in the same way as it is for non-speech. Therefore, this section will explain the different techniques used in speech recognition for each of the other steps (feature extraction and classification).

2.2.2.1 Feature Extraction

Feature Extraction techniques for speech recognition have evolved considerably over the years. Initially, features were extracted based on frequency. For instance, a researcher could identify the frequencies contained in a speech sample using some kind of frequency transform method. These frequencies then became the features of the sound [Rodman99].

In recent times, techniques have been developed that endeavor to filter the frequency information in order to isolate those frequencies that have a greater chance of characterising the speech sample [Juang00].

These filtering techniques can be broadly split into two areas:

- FFT-Based techniques
- Filter Bank Based techniques

In addition, some researchers have suggested that identification of multicomponent signals could also be used as a feature extraction technique [Fine91, Cohen92]. For instance, speech is naturally multicomponent and can be split into formants (resonant frequencies of the vocal tract). However, these techniques have not been well developed or benchmarked for speech and will therefore not be discussed in detail in this report.

FFT-Based techniques work with frequency information, specifically that produced by a Fast Fourier Transform (FFT). Techniques within this area include LPC (Linear Predictive Coding) and standard homomorphic Cepstral Coefficients [Lilly00]. The LPC technique included in this section works around the notion of a vocoder [Rodman99]. A vocoder is a device that can generate human speech when given the correct input. Coincidentally, the input for a vocoder also represents a set of features that can be used for speech recognition.

For *Cepstral Coefficients*, the signal is first split into frames using Hamming windows. A FFT is performed on each frame and the power spectrum of the frame is then calculated. A magnitude log is then performed and an inverse FFT is applied. This produces the Cepstral Coefficients.

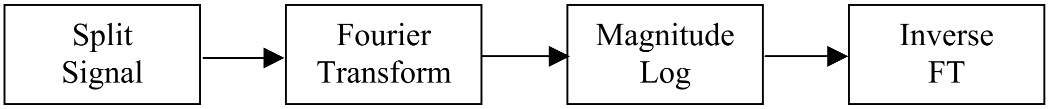


Figure 7 – Applying Cepstral Coefficients

For *Linear Predictive Coding*, the signal is also split into frames using Hamming windows. As with Cepstral Coefficients, an FFT is then calculated followed by calculation of the Power Spectrum. An inverse FFT is then applied to the frame. Finally, the Levison-Durbin Algorithm is applied to produce Linear Predictive (LP) features [Haykin99]. These LP features are then converted into LPC Cepstral Coefficients by the application of a formula that performs calculations based on the previous information in the signal.

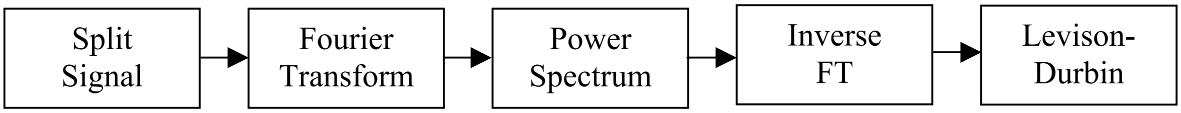


Figure 8 – Applying Linear Predictive Coding

Filter Bank Based techniques produce the same set of coefficients (either LPC Cepstral Coefficients or homomorphic cepstral coefficients). However, this set of techniques also includes a filter bank (based on a model of the human auditory system) which is designed to filter out those frequencies that are less likely to characterise a sound. Examples of filter bank based techniques include Bark Spectrum Cepstral Coefficients [Lilly00], Mel Spectrum Cepstral Coefficients and Perceptual Linear Prediction (PLP) [GoldM00].

Since the main difference in these techniques is in the contents of the filter bank, the method of implementation of these techniques is similar. First, Hamming windows are used to split the signal into frames. Each frame then has its Power Spectrum calculated. Next, the filter is applied to the frame. Signal Power is then calculated and the appropriate algorithm is applied to produce either Cepstral Coefficients or LPC Cepstral Coefficients.

For Cepstral Coefficients, a magnitude log is performed and then an inverse Discrete Cosine Transform (DCT) is performed. For LPC, an inverse DCT is performed and then the Levison-Durbin algorithm is applied.

Once features have been extracted from the speech sample, these features can be used for pattern recognition within the classification section of the procedure.

2.2.2.2 Classification

For classification, a plethora of different pattern recognition methods are used, depending on the originating source of the speech sample. If the original speech sample is an isolated word, standard statistical pattern recognition techniques such as Vector Quantization (VQ), Hidden Markov Models (HMM) [Jelin97], Acoustic Modeling (such as Dynamic Time Warping (DTW)) and Clustering Techniques (such as Self-Organising Maps (SOM) and Learning Vector Quantization (LVQ)) [Rabin90] can be applied to recognise the word. On the other hand, if the original speech sample is a series of connected words or a subset of continuous speech (such as dictation), more complicated methods have to be applied [Rabin93]. This is due to the difference between an isolated word, which is a single impulse sound, and continuous speech, which is a multiple impulse sound. A single impulse sound is one that has a short life in the time domain (e.g. Clicking fingers, door banging, single spoken word). A multiple impulse sound has a longer instance in the time domain [Melih98] (e.g. a musical composition, a sentence of words).

Multiple impulse sounds require more specialized variations on the HMM technique [Rabin93]. This involves first splitting the continuous speech signal into manageable sections (such as words, phonemes etc.). Each of these sections can then be recognised using the techniques listed above. The simplest way to do this is to use a whole-word technique, where each word is separated from the continuous speech sample and then techniques for isolated word recognition are applied [Rabin93]. However, this is time consuming because a system has to be trained for each word in all of its different phonetic variations. Also, the phonetic content of each individual word will inevitably overlap, causing redundancy in the system.

Due to this overlap, subword techniques are the preferred method of continuous speech recognition [Watro90]. This is because they do not need the above-mentioned redundancy. Subword techniques split the section into separate phonemes, syllables, demisyllables and acoustic units or, in the case of numbers, separate digits [Jiang00] with phonemes (typical length: 80ms) seeming to be the most popular choice [Kohon90, Lee90, Watro90]. The phonemes are then trained into a system as isolated words and can be recognised in the same way, with the additional requirement that the system must be able to form subwords back into the correct words.

2.2.3 Musical Instrument Recognition

To understand Music Instrument recognition, it is first important to understand the more general field of Recognition of Music. This field can be split into three distinct areas:

- Musical Pitch Recognition
- Music Instrument Recognition
- Computational Auditory Scene Analysis (CASA)

Musical Pitch Recognition concentrates on recognising the pitch of single tones or group of tones produced by musical instruments. This allows for structured searching of these instruments and, to a lesser extent, the generation of automatic musical score. In contrast, Music Instrument Recognition cares not for pitch, but concentrates on the recognition of the type of instrument being played (ie. Saxophone, Clarinet etc). Finally, the field of CASA is a major area of research in sound recognition work. CASA looks at splitting a polyphonic sound source (that may contain music, but also speech and environmental sounds) into separate sounds that can then be recognised by standard pattern recognition techniques. CASA is beyond the scope of this work, but in the general field of CASA, the thesis and subsequent work of Dan Ellis [Ellis96, Ellis98]; and the work in music CASA of Kashino & Murase [KashMur99] may be of some interest (CASA will also be discussed in more detail later in this chapter).

When looking at applying existing techniques from Music Recognition to Environmental Sound Recognition, it is readily apparent that the obvious choice between these areas is Music Instrument Recognition. Pitch does not play a role in our domain in the way it does for music (rather, simply as a feature for use in recognition of sounds). CASA is outside the scope of this work. However, Music Instrument Recognition could be considered similar to the recognition of environmental sounds.

Until the late 90's, little work had been done in the field of Music Instrument Recognition, and even less work had been done with realistic musical instrument recordings (and not single tones). The first major work that covered this area was the thesis of Keith Martin from the MIT Media Lab [Martin99].

Martin's thesis is of great interest because he proposes the use of a *taxonomic hierarchy* for the classification of musical instrument sounds. Martin suggests a multi-level hierarchy, where each level splits instruments into their natural categories (woodwind, brass, percussion; alto, tenor, bass etc). On each level, a selection of features is chosen based on the discriminatory properties of the sound under test. Typical features include items such as pitch, pitch variance, vibrato strength and onset duration. Classification is then performed using a simple maximum-likelihood estimation. Martin also suggests several enhancements to ensure a classification does not get "stuck" in the incorrect leaf node. Correct classification emerges from the lowest level of the hierarchy, with results (on realistic music samples, not isolated tones) of 90% for instrument families and 70% for individual instruments.

In addition to the work of Martin, several other seminal works also exist in the area of Music Instrument Recognition. The majority of these systems use standard feature extraction and pattern recognition techniques, oft-times taken from work previously done in the field of speech recognition (such as cepstral coefficients and k-nearest neighbour classification).

Marques & Moreno used several different techniques for the feature extraction and classification of musical instruments, with the best results coming from a combination of mel-frequency cepstral coefficients with either Gaussian Mixture Models (GMM) or Support Vector Machines (SVM). Martin [Martin99] reports initial results from Marques showing a classification rate of 72% for professional recordings and 45% for non-professional recordings. Since then, a further technical report from Cambridge Research Laboratory [MarqMor99] shows an error rate of 30% (70% classification) when using mixed data, reduced to only 2% (98% classification) when using data from a single source. This suggests the applicability of robustness techniques (also tested in speech recognition) to this domain, in order to combat this problem with variable training/test data.

Brown also presents a system using cepstral coefficients [Brown99]. In this case, the system uses Q-cepstral coefficients with a Gaussian Mixture Model for classification (one model for each instrument). On independent, noisy samples of music, the system achieves a classification rate of 94%, between oboe and saxophone recordings. However, this system achieves only 84% when extended to include four samples of instrument (oboe, saxophone, flute and clarinet).

Eronen & Klapuri apply a music instrument recognition system very similar to the system used by Martin, with the addition of some spectral features [EroKlap00]. They use Bark frequency cepstral coefficients to determine these spectral features as well as some temporal features. They then propose the use of taxonomy for classification, identical to that used by Martin, with the exception of the addition of a piano class. For this taxonomy, they use Gaussian classification at the higher levels and k-nearest neighbour (k-NN) classification on the lower levels of the hierarchy. This approach achieves results of 94.7% for classification of instrument families and 80.6% for classification of individual instruments.

In addition to these "realistic recording" systems, a number of recognition systems exist that classify musical instruments based on isolated tones. Systems such as these are limited in their applicability to environmental sound recognition (due to the tonal nature of signal tones). Nonetheless, they deserve some analysis.

Of these systems, the most recent (and arguably, the most successful) has been developed by Ichiro Fujinaga. Fujinaga (in cooperation with Karl MacMillan and Angela Fraser) [FujiMac00, FraserFuj99, Fujinaga98], proposes the use of pitch detection for feature extraction (based on the `fiddle` program by Puckette [PuckApZi98]) and the k-nearest neighbour (k-NN) technique for subsequent recognition of these features. The pitch detection technique used takes advantage of the fact that musical instruments are by their nature very tonal. This means that the pitch of a single note is an excellent feature to use. In addition, the database used specifically splits different tones with a short pause, allowing simple classification of a single tone. For these experiments, Fujinaga obtained a result of 95% - 100% when classifying 3 – 10 instrument groups, but this result fell to 68% on a 39-timbre group (made up of 23 orchestral instruments).

2.2.4 Time-Frequency Feature Extraction

Although the Fourier Transform (FT) is the most common technique for the transformation of amplitude-time signals into amplitude-frequency signals, it does not handle non-stationary signals well. This is because non-stationary signals do not contain the same frequencies in all parts of the signal. In order to understand this, an example is in order.

Given a vector *t*, the following formula will produce a stationary signal (a signal with frequencies contained equally throughout the whole signal).

$$x(t) = \sin(4\pi t) + \sin(40\pi t) \tag{1}$$

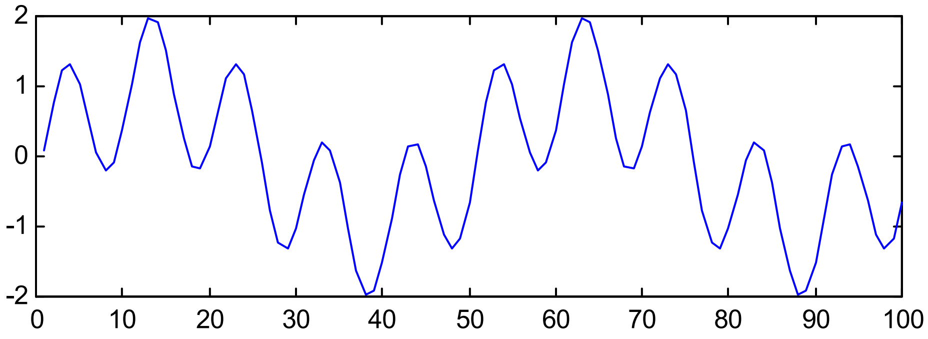


Figure 9 – A simple Waveform with Constant Frequencies

This signal is made up of the following two waveforms, with each waveform constant throughout the signal.

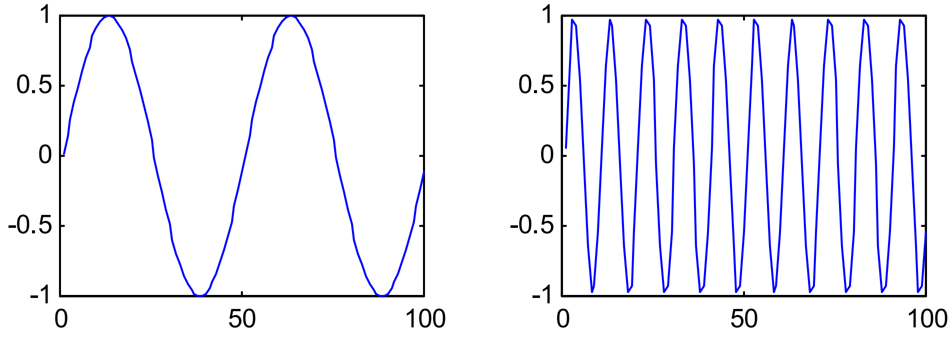


Figure 10 – Waveforms contained in Original Signal

The Fourier transform of this signal would look like this:

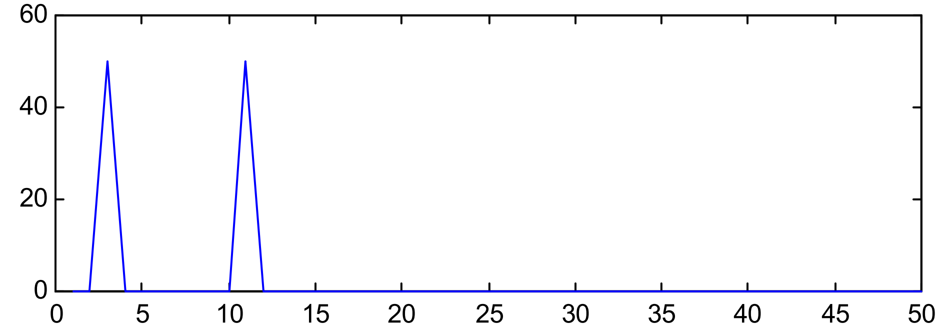


Figure 11 – Fourier Transform of Original Signal

In comparision, the following formula (using the same two signals) will produce a non-stationary signal (with frequencies spaced out unevenly in the signal). Most pratical signals (including sound signals) are of this type.

$$x(t) = \begin{cases} \sin(2*pi*2*t) & 1 \le t \le 100 \\ \sin(2*pi*10*t) & t > 100 \end{cases} \tag{2}$$

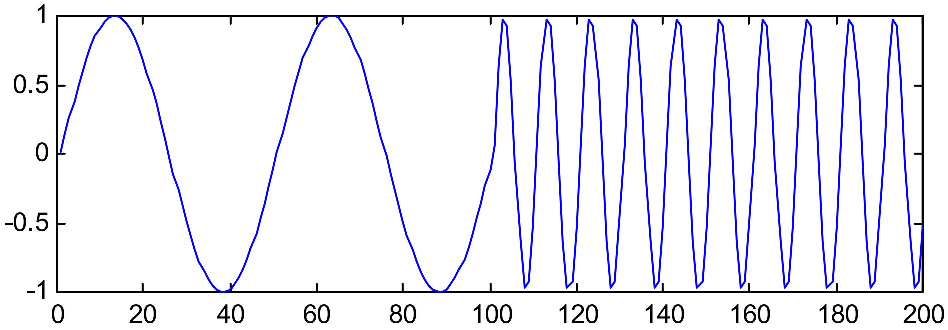


Figure 12 – Second Example Signal

The Fourier Transform of this signal would look like this:

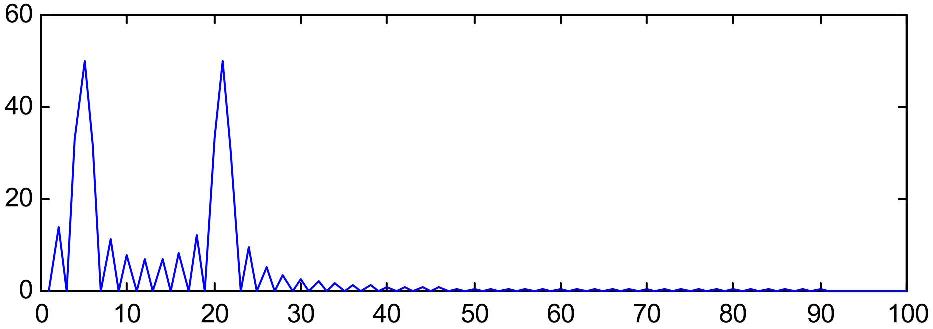


Figure 13 – Fourier Transform of Second Example Signal

It should be noticed that, apart from the jagged noise at the bottom (introduced due to the sharp transition from one frequency to the next), this signal is almost identical to the previous signal. This is because a Fourier transform does not take time information into account. It simply identifies *all* frequencies contained in a signal for the *entire* length of the signal. Therefore, in order to accurately represent the frequency information contained in a non-stationary signal, a technique needs to be used that preserves both time information and frequency information, therefore producing a signal spectrum in the time-frequency domain.

2.2.4.1 Time-Frequency Transforms

Several different techniques have been proposed to perform a time-frequency transform on a signal [Polikar03]. The most common of these are:

- Wavelets
- Short Time Fourier Transform (STFT)
- Wigner-Ville Distribution

Of these, the STFT is the easiest to implement. It simply applies a FT to successive windowed segments of the signal. This allows both frequency and time information to be presented for a signal. The STFT of the second signal is shown in Figure 14.

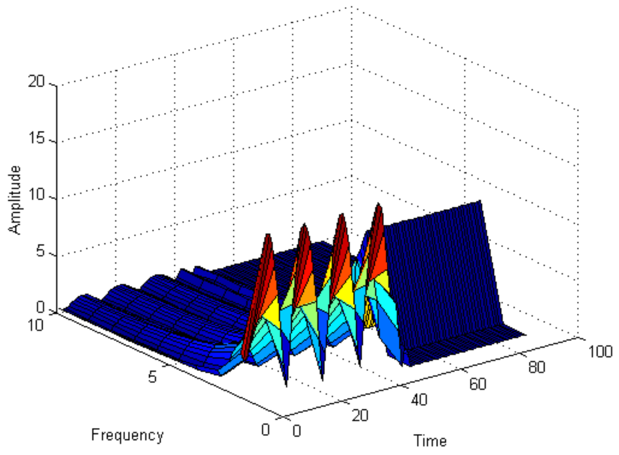


Figure 14 – An Example Short-Time Fourier Transform

Notice the graph is now three-dimensional, with the x axis representing time, y axis representing frequency and z axis representing the amplitude of the frequencies. Also notice that time on this plot is no longer represented in the original form, but in the number of overlapping windows that have been taken of the original signal to generate the STFT. This representation of time is called translation, as it no longer relates directly to time samples.

From this angle, the plot does not show much. However, the results are interesting if we turn the plot around to a different angle (as shown in Figure 15).

We can see that for the first 50 samples, the signal contains a lower frequency signal than for the last 50 samples. Using this method, we benefit from both time and frequency information and can gain a better representation of a non-stationary signal.

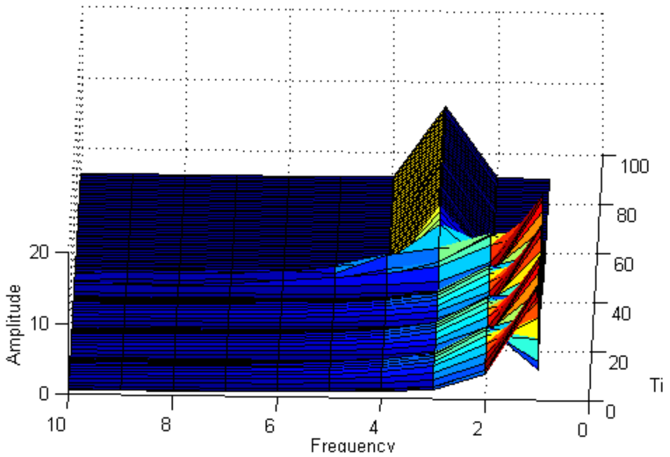


Figure 15 – An Example STFT from another angle

However, a problem is apparent from this plot. Namely, what frequencies are contained in this signal? Even taking into account the fact that the frequency axis is not normalised, each frequency is still not clear. The second frequency could be anything from 2 to 4 on the graph. Similarly, the first could be any frequency from 0 to 3.

This is due to the fact that the STFT function used to generate this plot takes a lot of time samples. Therefore, each frequency sample is quite small and frequencies are shown in wider bands. The algorithm can also be configured to take less time samples. This approach produces the following two plots:

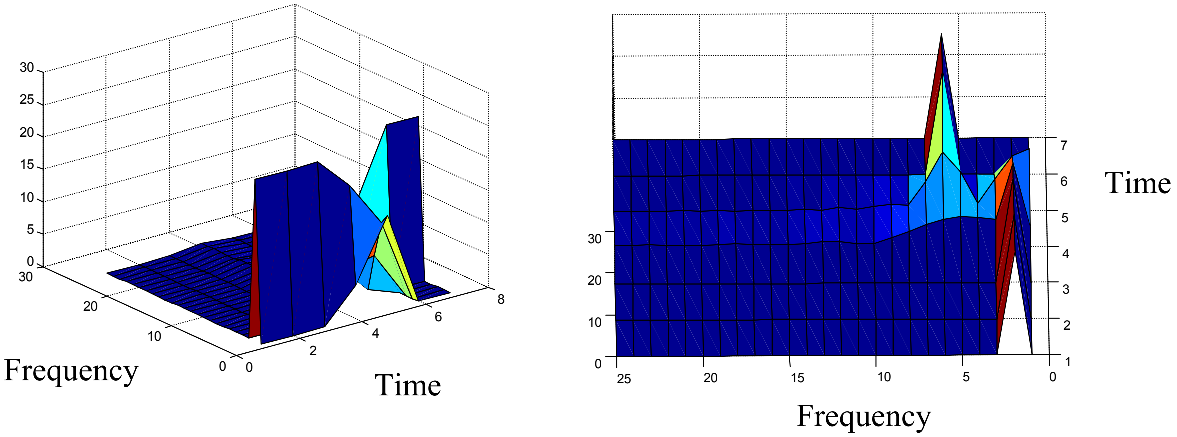


Figure 16 – Second Example STFT Plot

From the second plot it can be seen that frequency is now more easily determined. Each frequency takes a narrower range and it is therefore easier to determine each frequency. However, the first plot shows that now the times for each frequency overlap. It is not as clear when one frequency ends and another begins.

These two examples highlight the main problem with STFT: resolution. As the time resolution increases, the frequency resolution decreases. If we attempt to increase frequency resolution, the time resolution must decrease. This is unavoidable using a windowing technique (and is called the Heisenberg uncertainty principle).

So, how do we solve this problem? Application of a different technique can not solve the problem (the Heisenberg uncertainty principle is unavoidable), but it can allow different resolutions for frequency and time depending on which is more important. The technique that allows us to do this is the wavelet transform.

2.2.4.2 Wavelet Transform

In order to combat this resolution problem inherent in an STFT, wavelets are used. A wavelet applies good time resolution (and therefore poor frequency resolution) at high frequencies and good frequency resolution (and therefore poor time resolution) at low frequencies. Due to the fact that most signals contain high frequencies for only short amounts of time and low frequencies for longer amounts of time, this approach presents a good time-frequency representation.

To perform a wavelet transform, we must first select a *mother wavelet*. This wavelet is a typical representation that acts as a template and will be compressed or dilated and then compared with the signal selected. Several different mother wavelets have been proposed, including the Morlet wavelet and the mexican hat wavelet. An example of these wavelets (taken from the Mathworks Help files) is shown in Figure 17.

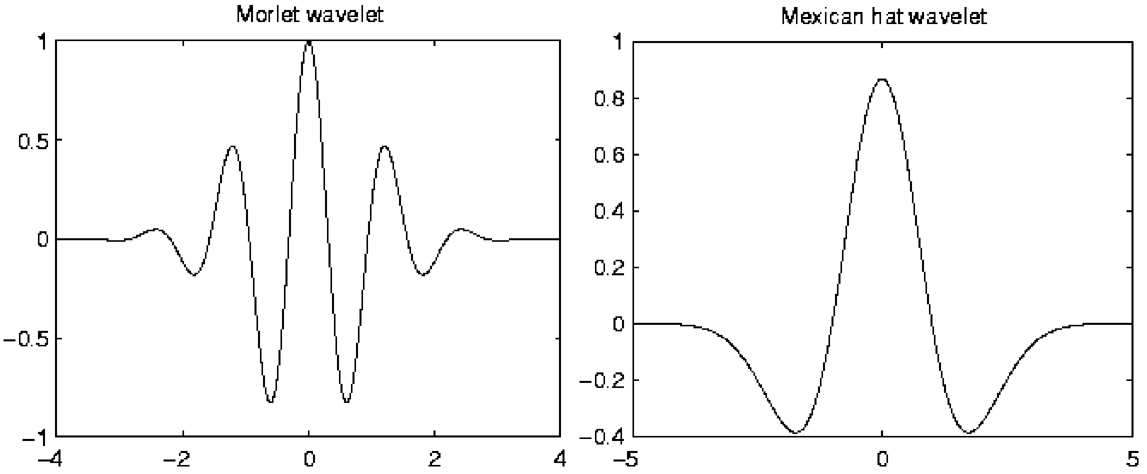


Figure 17 – Two Example Mother Wavelets

Once a mother wavelet has been selected, it is used to window the signal. First, it is compressed to a size equivalent to the highest frequencies in the signal. It is then windowed onto the signal (as with a STFT). If the frequencies match, a high number will be generated otherwise a low number will be generated. Once this windowing procedure is complete, the signal is dilated slightly and the process repeats. The product of this process is a series of values representing the frequencies contained in the signal. However, because of the way this technique is applied (with highly compressed (high frequency) waveforms applied first), frequency is reversed. This means that the first iteration of the technique generates high frequencies, while the last iteration detects lower frequencies.

In order to represent this inverse frequency (1/frequency) relationship, the term scale is used. Scale represents the inverse frequency of a waveform. This means that a CWT plot will show the higher frequencies at the lower end of the number scale and the lower frequencies at the higher end of the number scale. Since the CWT time-frequency transform is also based on overlapping window samples and not time samples, the term translation is also used. Figure 18 shows an example of a wavelet transform from two different viewpoints.

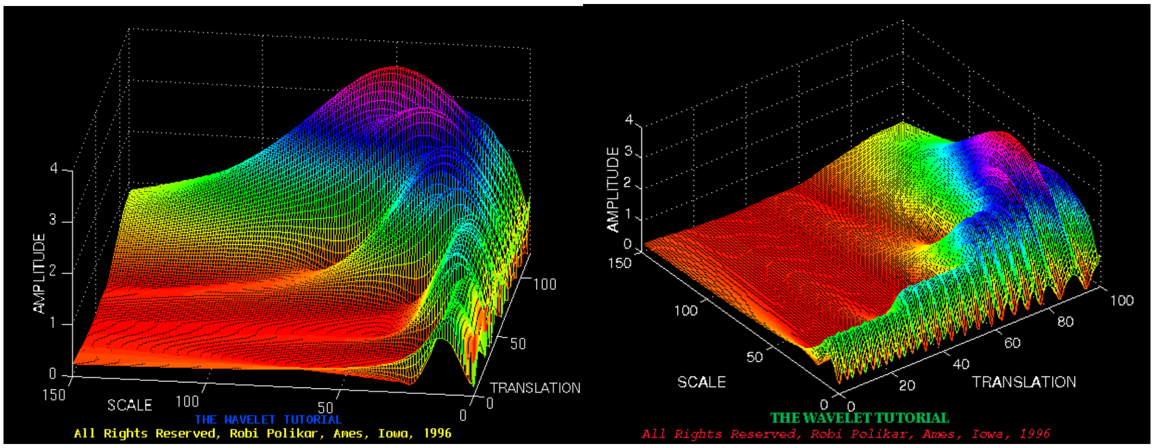


Figure 18 – An Example Continuous Wavelet Transform [Polikar03]

2.2.4.3 Discrete Wavelet Transform

The type of transform described in the previous section is known as a continuous wavelet transform (CWT). This transform is applied in a continuous fashion across the entire signal. If done in a computer, it is discretized by setting the scale values to some discrete set (ie. s = [1, 2, 3…]) but despite this, it is still a discretized CWT (also known as Wavelet Series due to the use of orthonormal wavelets).

However, just as with the Fast Fourier Transform (FFT), another algorithm has been developed to perform the Wavelet Transform process faster. This algorithm is called a Discrete (Fast) Wavelet Transform (FWT).

Essentially, a FWT produces the same results as a CWT when applied to a signal that is sampled discreetly (as is the case with almost ALL signals used for computer computation). A FWT does this by applying a series of high-pass and low-pass filters to the original signal. These filters split the signal into separate components. By continually breaking down the low-pass component of the signal, we obtain a set of data that represents the signal well in the time-frequency domain.

For example, say we had a signal containing a maximum frequency of 1000Hz that is 2048 samples long. We would first use a high pass and low pass filter on this signal. This would give us two signals, one containing only those frequencies from 500HZ – 1000Hz and one containing the frequencies from 1Hz to 500Hz. After the filters have been applied, we can then subsample these signals. This is because the low pass signal can only contain frequencies up to 500Hz, but the original length of the signal allows for double this frequency. Similarly, the high pass signal can only contain frequencies from 500Hz to 1000Hz, but the length allows for double this. We subsample by removing every other sample from the signal (producing two 1024 sample signals). This has the effect of halving the time resolution but doubling the frequency resolution (as with the low frequencies in the CWT). We then apply high and low pass filters to the new low passed signal and repeat the procedure. This is repeated until we have a signal with only one sample.

The result of this procedure is a series of signals, each half the length of the last. Working from the highest signal to the lowest signal, it is clear that the first signal generated contains 1024 samples of time resolution and frequency range from 500Hz to 1000Hz. The next signal contains 512 samples of time resolution and frequency range of 250Hz to 500Hz etc. The final signal contains 1 sample of time resolution and less than 1Hz of frequency range.

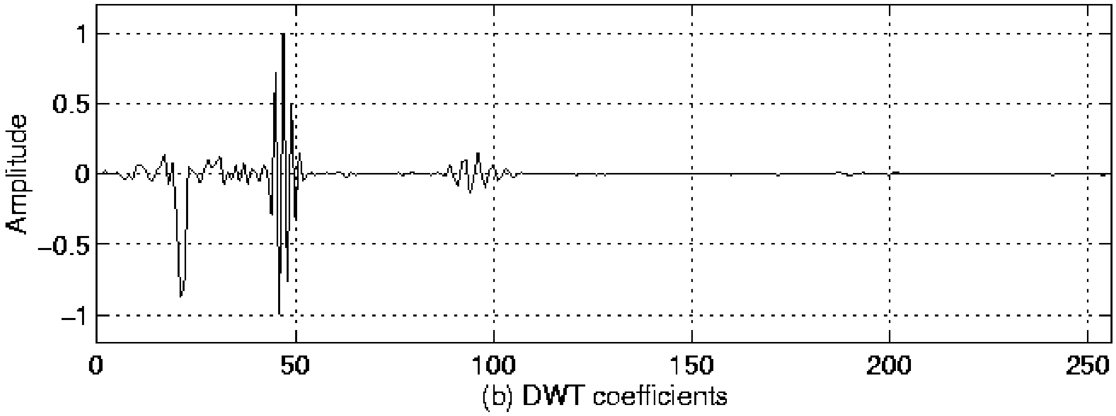


Figure 19 – A typical Discrete (Fast) Wavelet Transform [Polikar03]

This generated signal meets the criteria set out for the CWT. Although at first glance it would appear that lower signals contain less frequency information, one must bear in mind that more time samples are being used to represent less frequency information, therefore frequency resolution is increased. A plot of a typical FWT signal is shown in Figure 19.

In order to interpret this signal, we must mentally split it into sections based on the levels produced by the FWT process. The signal shown in Figure 19 is generated from a 256-point signal sampled at 10KHz. From our knowledge of the FWT, we know that the first 128 samples (from the right) correspond to those frequencies from 5000Hz to 10000Hz. We can see that this section contains no information. The second set of 64 samples then corresponds to frequencies from 2500Hz to 5000Hz. We can see that for this section, the sample produces some small frequencies in this range at around the middle of the sample. The next 32 samples represent frequencies from 1250Hz to 2500Hz. As can be seen, the majority of the signal energy is contained in this range, around the middle of the sample (in time). The process of analysis can continue for the remainder of the signal, but in this case we have identified that the majority of energy in the original signal was in the middle of the translation band between 1250Hz and 2500Hz. Considering the original signal shown in Figure 20, which has a length of 1sec, this is a reasonable conclusion.

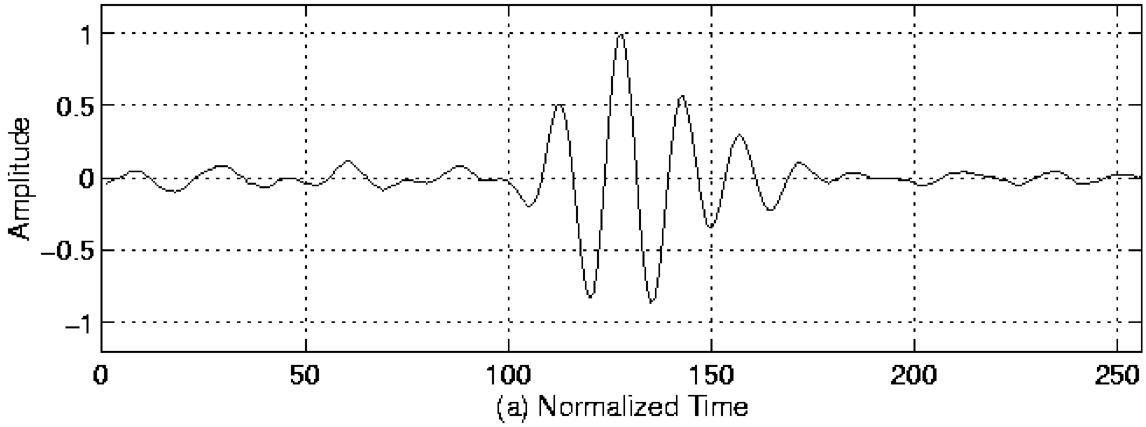


Figure 20 – The Original Waveform for the FWT [Polikar03]

2.2.4.4 Daubechies' Wavelets

A final point that should be noted when looking at the FWT transform is the issue of signal reconstruction. If we are using a FWT for compression, then it is imperative that the original signal can be reconstructed from the FWT information. Due to the extra information contained in a CWT, reconstruction is always possible with a CWT, but for a FWT, this information does not exist.

In order to be able to reconstruct a signal from FWT information, we require the application of ideal high pass and low pass filters. However, these type of filters do not exist. However, filters DO exist that allow for a perfect reconstruction. These are called the Daubechies filters (developed by Ingrid Daubechies [Daube92]) and a FWT performed using these filters produces Daubechies' wavelets, which are simply wavelets with the ability to be reconstructed into the original signal.

2.2.5 Auditory Scene Analysis

An extension of sound classification research is called Auditory Scene Analysis [Bregman90]. This process involves analysing a scene, such as the noise produced at a cocktail party, and then separating and classifying the sounds in the environment. When auditory scene analysis is implemented in a computer, it is called computational auditory scene analysis [Cooke01]. Computational auditory scene analysis requires understanding how sounds can be combined in an environment and then understanding the way the sounds are represented in a waveform. This information can then be used to split the waveform into separate sounds. It also involves using one of the identification techniques outlined above.

Although other approaches to source separation exist [HoytW94, Huang95, Liu99], the main research in computational auditory scene analysis can be broken into two areas: Data-Driven Source Separation and Prediction Driven Source Separation.

The approach discussed in [Bregman90] describes a data-driven approach to scene analysis. This involves using the statistical and semantic features of a waveform in order to understand the different sounds contained within the waveform.

The second approach to scene analysis is proposed in [Ellis96, Ellis98] and uses prediction techniques in addition to data-driven techniques. First, a small amount of the waveform is analysed. Sounds are identified using a data approach. A prediction is then made on how these sounds will change next, and how this is expected to affect the waveform. The next section of the waveform is then sampled and the prediction is modified accordingly. The process then continues. The advantage of this approach is that sounds that may have been muffled or covered by other sounds, i.e. overlapping (and therefore hard to detect), can still be recognised due to the previous prediction.

2.3 Direction Detection

To implement this environmental sound recognition system in an autonomous robot for surveillance, it must not only be able to identify a sound, but also have an idea of the direction from which it came. This entails the use of direction detection or *source localization* techniques. Unlike non-speech sound recognition, direction detection has had more attention in the research community. This section will discuss the two areas of source localisation research and what they involve.

Current research can be split into two broad areas:

- Simulation of Head Related Transfer Functions
- Time Delay Estimation

Each of these areas will be discussed in this section. In addition, this section will discuss the concept of **Microphone Arrays**, which is required in order to understand how source localization is implemented.

2.3.1 Microphone Arrays

Most source localization techniques rely on the concept of a microphone array [Brand97]. The simplest form of microphone array is modeled on the human auditory system and consists of two microphones. However, microphone arrays can contain many more microphones, depending on the way the technique is implemented [Brand97].

The majority of microphone array based techniques still rely on the use of the microphones in the array in pairs [Omolo97]. Omologo suggests that it is important to investigate the configuration of the microphone array [Omolo97]. This means that the way that the microphones in the microphone array are positioned relative to the sources of noise (distance) as well as relative to each other is important. The number of microphones (counted in pairs) is also relevant [Arslan00].

Another important consideration when building a microphone array is selection of microphones. Microphones are available in two types: omni-directional or directional. Omni-directional microphones hear sounds in all directions equally and are therefore suited to systems where sound arriving from say, the rear, is not an issue. Directional microphones receive sound from only a selected degree pattern. Figure 21 shows a microphone with a 90° field of "hearing". Sound A will be heard by this microphone while sound B will be ignored.

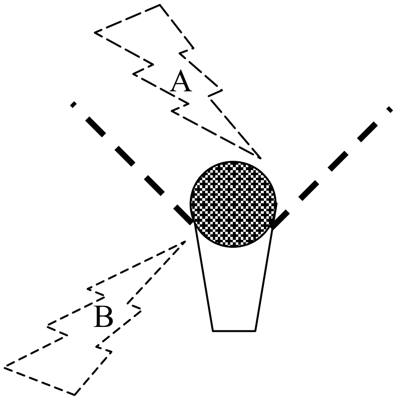


Figure 21 – Example of a directional microphone.

A benefit of using a directional microphone is the ability of a series of these microphones to form "beams". This type of microphone array is called a beamforming microphone array [Steele00, Brand00]. A beamforming microphone array uses directional microphones and selects a direction from which to gather a sound. Weights are assigned and these are used to emphasise the signal coming from the direction selected.

Finally, another consideration when using microphone arrays is near-field reflections and interference [Ryan00]. Most microphone arrays are designed with the intention that sounds will be far away from the microphones. This has the advantage of allowing researchers to assume that all waves approaching the microphones will be planar. However, in some applications, sounds are produced closer to the microphone. In these instances, wavefront curvature is detected by the array and errors can be recorded. Techniques have been developed to deal with this occurrence and will be investigated if the need arises.

2.3.2 Simulation of Head Related Transfer Functions

Head Related Transfer Functions (HRTF) relate to the way that a human being uses the information from their two ears to determine the direction of sound [Shaik99]. As well as being used to generate virtual 3D sounds (such as the techniques used in Dolby Headphone technology) [Georg00], techniques have also been developed that attempt to simulate HRTF in order to allow a computer to localise sound information. The majority of HRTF techniques work within the frequency domain (by first using some kind of Fourier Transform on the sound sample). However, techniques also exist that work in the time domain [Cheve98]. For instance, some **Auditory Scene Analysis** techniques present methods by which sounds are localized in the time domain.

Nandy discusses a frequency-based model that simulates the human ear's ability to recognise both interaural time differences (ITD) and interaural auditory differences (IAD) from each ear [Nandy95]. The model also attempts to simulate the different sections of the ear canal from the cochlea to the auditory cortex. A neural network is then trained using the data from both sources. (For background information on how ITD and IAD are used by human beings in order to localise sound, please refer to [KingC95, Konis95]. In addition, [Hart99] presents interesting information on ITD and IAD and their weaknesses.)

The experimental data presented in [Nandy95] suggests that simulation of HRTF allows for relatively good direction localisation within the horizontal plane. However, the system presented by Nandy does not localise within the vertical plane.

Liu presents a system based on a two microphone biological hearing system utilizing ITD [Liu99]. The system uses the Shamma's Stereausis neural network model to remove the neural delay from a system. This system also performs signal segregation (auditory scene analysis). Unfortunately, Liu presents only the theoretical underpinning of this technique and presents no experimental results, making the efficiency and accuracy of this technique difficult to determine.

Gill & Troyansky present a model for the determination of the elevation of a natural sound using monaural cues [GillT00]. This system is built around a neural network. Because this system is monaural, it does not use ITD or IAD, but uses multiple "snapshots" to determine direction. Results from this system are good, providing adequate snapshots are able to be gathered. The system produces little error for three snapshots, but jumps to 40% error if only two snapshots are used. In addition, the implementation of a HRTF model is complicated and cumbersome. Therefore, a simpler alternative using only ITD may be considered.

2.3.3 Time Delay Estimation

Similar to HRTF, Time Delay Estimation (TDE) relies on comparing the difference in time that a sound signal arrives at various microphones. Obviously, the sound signal arriving first is closer to the source of the sound. By carefully configuring the microphone array, the source of the sound can be determined with an error of less than 10cm [Omolo97]. However, unlike HRTF, the processing that occurs on the signals once they are encoded into the computer does not rely on inputting them into a model of the human auditory system. TDE techniques rely on signal processing techniques to compare the phase difference between the different sound waves [Brand97]. Techniques used include normalized cross correlation, least mean square (LMS) adaptive filter [Omolo97] and crosspower-spectrum phase (CSP) analysis techniques [Rabink96].

Hiyane uses a beamforming microphone array in order to track the direction of a sound source [Hiya00]. The microphone array used by Hiyane consists of 16 microphones arranged in a circular pattern with a diameter of 30cm (see Figure 1.5). In a single sound environment, Hiyane notes that the resolution of source direction estimation is less than 10 degrees. However, he also notes problems can occur when multiple sounds are introduced into the environment at the same time.

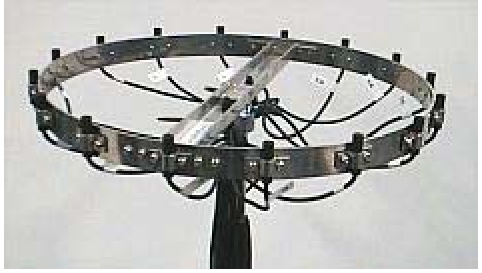


Figure 22 - Circular microphone array with 16 channels [Hiya00]

Interestingly, Huang presents a hybrid model that uses TDE and a precedence model (a simple HRTF model intended to cancel echoes and reverberation) for robot navigation [Huang99]. Huang presents this system on a small robot with three microphones. The time difference between the microphones allows the robot to determine direction. Interesting, the robot also contains a sonar device that allows it to avoid obstacles while moving towards the sound.

However, Huang only tests the abilities of this robot using two sounds sources: a sinusoidal tone produced by a speaker and the sound of hands clapping. This only represents one "real" environmental sound, according to the definitions put forth by Vanderveer [Vander79]. For the sinusoidal tone produced by the loudspeaker, error is only one degree. However, for the clapping, error is seven degrees.

This implementation seems to work to within an acceptable error and would be suitable as a base for this project. However, testing would have to be performed to ensure accuracy does not decrease when the system is exposed to a greater number of environmental sounds.

2.4 Problems with Existing Methods

The previous chapter has provided a comprehensive review of the existing research into source classification and source localization techniques. From this literature review, it becomes apparent that the area of source localization is quite well covered. It would be relatively easy to implement an existing source localization technique for a robot that needs to localize non-speech sounds, satisfactory to the purpose.

However, the area of source classification is not so clear-cut. Almost no research has been done in the area of non-speech sound recognition. Much more research has been conducted into speech and musical instrument recognition than has been applied to the area of general, non-speech environmental sounds. The research that has been done into non-speech sounds tends to focus on one technique to the exclusion of all others. This means that no comparison has been done to determine the most appropriate feature extraction and classification techniques for the recognition of non-speech environmental sounds.

In addition, while advanced techniques such as taxonomic hierarchies have been applied to the areas of speech recognition and musical instrument identification, these techniques have not been applied to environmental sounds. It is apparent that it is the application of these techniques that has allowed these domains to overcome the complexity of their pattern search space. However, because of the recognised lack of an environmental sound alphabet, no such structured classification methods for non-speech environmental sounds exist. It is possible that the implementation of a more structured technique based on an environmental alphabet would produce improved results for environmental sounds.

Chapter 3

Hypothesis & Proposed Solution[1]

From the literature review, it can be seen that the problem within this area of research can be broken into two components. First, no comprehensive comparative study has been done to determine the most appropriate technique for non-speech sound recognition. Secondly, advanced techniques for non-speech sound recognition have not been investigated.

This thesis aims to address these problems, as outlined in the hypothesis:

Hypothesis

If I can find a systematic way to identify environmental sounds, I could increase the efficiency of environmental sound identification for the purpose of security surveillance. A system can be developed that will recognise a large corpus of environmental sounds. This system will use a structured classification technique (sound taxonomy) to improve classification accuracy and speed.

[1] The work reported in this chapter and the following chapters resulted in the following publications:

- M. Cowling, R. Sitte, "Recognition of Environmental Sounds using Speech Recognition Techniques", Advanced Signal Processing for Communications Systems, 2002, Kluwer Academic Publishers.
- M. Cowling, R. Sitte, "Comparison of Techniques for Environmental Sound Recognition", *Pattern Recognition Letters*, Elsevier Science Inc.,Vol. 24, Issue 15, Nov. 2003, pp. 2895-2907.
- M. Cowling, R. Sitte, "Building an Environmental Sound Taxonomy for Autonomous Robot Surveillance", *Proc. of DSPCS'03*, Gold Coast, QLD, Australia, December, 2003.
- M. Cowling, R. Sitte, "Time-Frequency Environmental Sound Recognition for Autonomous Robot Surveillance", *Proc. of AMiRE 2003*, Brisbane, February, 2003.
- M. Cowling, R. Sitte, "Structured Classification of Environmental Sounds", *Proceedings of WoSPA 2002,* Brisbane, December, 2002.
- M. Cowling, R. Sitte, "Analysis of Speech Recognition Techniques for use in a Non-Speech Sound Recognition System", *Proceedings of DSPCS'02,* Manly, NSW, Australia, January, 2002.
- M. Cowling, R. Sitte, "Sound Identification and Direction Detection for Surveillance Applications", *Proceedings of ICICS 2001,* Singapore, October, 2001.

3.1 Proposed Solution

This section analyses a number of techniques for their suitability to environmental sound recognition. As mentioned in the literature review, sound recognition (be it speech or environmental) is generally done in two phases: first feature extraction, followed by classification (using artificial intelligence techniques). This section discusses techniques in both these phases. Feature extraction is where a sound is manipulated in order to produce a set of characteristic features for that sound. For instance, a sound could be considered a high-pitched sound, or a low-pitched sound. Classification is then used to recognize the sound by cataloging the features of existing sounds in some way (training) and then comparing the test sound to this database of features (testing).

Feature extraction can be split into two broad types: stationary (frequency-based) feature extraction and non-stationary (time-frequency based) feature extraction. Stationary feature extraction produces an overall result detailing the frequencies contained in the entire signal. With stationary feature extraction, no distinction is made on where these frequencies occurred in the signal. In contrast, non-stationary feature extraction splits the signal up into discrete time units. This allows frequency to be identified as occurring in a particular area of the signal, aiding understanding of the signal.

3.1.1 Feature Extraction (Stationary)

For stationary feature extraction, speech and musical instrument recognition rely on only a few different types of feature extraction technique (each with several different variations). Initially, I considered eight popular techniques (two of which are commonly used in musical instrument recognition and all of which are commonly used in speech recognition) as possible candidates for feature extraction of non-speech sounds. These were:

- Frequency Extraction (Music & Speech)
- Homomorphic Cepstral Coefficients
- Mel Frequency Cepstral Coefficients (Music & Speech)
- Linear Prediction Cepstral (LPC) Coefficients

- Mel Frequency Linear Prediction Cepstral (LPC) Coefficients
- Bark Frequency Cepstral Coefficients
- Bark Frequency Linear Prediction Cepstral (LPC) Coefficients
- Perceptual Linear Prediction (PLP) Features

It should be noted that while Frequency Extraction is a stationary technique, other techniques using Cepstral Coefficients could be considered "pseudo-stationary" techniques because they split the signal into time-slices. These are not true time-frequency techniques, because each time-slice has to be taken in context with other time-slices in order to produce relevant information.

Techniques based on LPC Coefficients were based on the idea of a *vocoder*, which is a simulation of the human vocal tract. Since the human vocal tract does not produce environmental sounds, these techniques typically seem to highlight non-unique features in the sound and are therefore not appropriate for recognition of non-speech sounds.

According to Lilly [Lilly00], the results of the Mel Frequency Based Filter and the Bark Frequency filter are similar, mainly due to the similar nature of these filters. Gold [GoldM00] also mentions that PLP and Mel Frequency are similar techniques. Based on these previous findings, I selected only the more popular Mel Frequency technique for testing.

3.1.2 Feature Extraction (Non-Stationary)

The main time-frequency techniques that are commonly mentioned in the general literature (eg. [Cohen95]; [Hubbard95]) are:

- Short-Time Fourier Transform (STFT)
- Fast (Discrete) Wavelet Transform (FWT)
- Continuous Wavelet Transform (CWT)
- Wigner-Ville Distribution (WVD)

All of these techniques use different algorithms to produce a time-frequency representation of a signal. While STFT uses a standard Fourier transform over several windows, Wavelet-based techniques apply a mother wavelet to a waveform to surmount the resolution issues inherent in STFT. WVD is a bilinear time-frequency distribution that also uses advanced techniques to try and combat these resolution difficulties. It has higher resolution than the STFT, but suffers from crossterm interference and produces results with coarser granularity than Wavelet techniques [Hubbard95]. Of the two wavelet techniques, FWT is usually used for encoding and decoding of signals, while CWT is used for recognition tasks.

Despite its common usage for speech/sound coding, the FWT could be used successfully for recognition tasks, so it should be included in our comparative study. However, early tests on the Wigner-Ville distribution showed a transformation duration in excess of five minutes for signals of the length typical for environmental sounds. Given the intention to develop my system into a real-time surveillance system, this excessive duration was deemed unacceptable.

Based on these findings, three techniques (STFT, FWT, CWT) should be tested for their ability to classify non-speech sounds.

3.1.3 Classification

After feature extraction, a classification technique is used to catalogue the features. Test features can then be compared to this database.

The following classification techniques are commonly used for speech/speaker recognition or have, in the past, been used for this application domain. They are:

- Dynamic Time Warping (DTW)
- Hidden Markov Models (HMM)
- Learning Vector Quantization (LVQ)
- Self-Organising Maps (SOM)
- Ergodic-HMM's

- Artificial Neural Networks (ANN)
- Long-Term Statistics (LTS)

In addition to these techniques, I also highlighted three techniques commonly used on realistic recordings in musical instrument recognition (not just isolated tones):

- Maximum Likelihood Estimation (MLE)
- Gaussian Mixture Models (GMM)
- Support Vector Machines (SVM)

To aid in selection of techniques, comparison tables were built (using [GoldM00]; [Lee96, Lee96b]; [Rodman99] as a base) to compare the different feature extraction and classification methods used by each of these techniques (Table 2, Table 3).

The comparison tables showed that some of these techniques, by their very nature, cannot be used for non-speech sound recognition. Any of the techniques that use subword features are not suitable for non-speech sound identification. This is because environmental sounds lack the phonetic structure that speech does. There is no set "alphabet" that certain slices of non-speech sound can be split into, and therefore subword features (and the related techniques) cannot be used (this is also noted in [ReyesEl03]).

Due to the lack of an environmental sound alphabet, the Hidden Markov Model (HMM) based techniques mentioned will be difficult to implement. However, this technique may be revisited in the future if necessary, and if a meaningful way of developing sound sub-units can be devised. However, this is beyond the purpose of this research.

The SOM and LVQ techniques are complementary to each other. Kohonen developed both techniques, with specific applications intended for each technique. For classification, Kohonen [Kohon90] suggests the use of the LVQ technique over the SOM technique. Therefore, LVQ will be the technique tested.

Long-term statistics cannot be applied in combination with non-stationary feature extraction techniques. Therefore, this classification technique will only be tested on its own feature extraction techniques.

Finally, all of the techniques used for musical instrument recognition work on a similar paradigm, that of unsupervised classification. For efficiency, I selected the most widely used of these techniques for testing, GMM's.

Table 2. Speech Recognition

Technique	Sub Technique	Relevant Variable(s) / Data Structures	Input	Output
Sound Sampling	ALL	Analog Sound Signal	Analog Sound Signal	Digital Sound Samples
Feature Extraction	Dynamic Time Warping (DTW)	Statistical Features (e.g. LPC coefficients)	Digital Sound Samples	Acoustic Sequence Templates
	Hidden Markov Models (HMM)	Subword Features (e.g. phonemes)	Digital Sound Samples	Subword Features (e.g. phonemes)
	Artificial Neural Networks (ANN)	Statistical Features (e.g. LPC coefficients)	Digital Sound Samples	Statistical Features (e.g. LPC coefficients)
Training and Testing	Dynamic Time Warping (DTW)	Reference Model Database	Acoustic Sequence Templates	Comparison Score
	Hidden Markov Models (HMM)	Markov Chain	Subword Features (e.g. phonemes)	Comparison Score
	Artificial Neural Networks (ANN)	Neural Network with Weights	Statistical Features (e.g. LPC coefficients)	Positive/Negative Output

TABLE 3. SPEAKER RECOGNITION

Technique	Sub Technique	Relevant Variable(s) / Data Structures	Input	Output
Sound Sampling	ALL	Analog Sound Signal	Analog Sound Signal	Digital Sound Samples
Feature Extraction	Dynamic Time Warping (DTW)	Statistical Features (e.g. LPC coefficients)	Digital Sound Samples	Acoustic Sequence Templates
	Hidden Markov Models (HMM)	Subword Features (e.g. phonemes)	Digital Sound Samples	Subword Features (e.g. phonemes)
	Vector Quantization (VQ)	Statistical Features (e.g. LPC coefficients)	Digital Sound Samples	Statistical Features (e.g. LPC coefficients)
	Ergodic-HMM's	Subword Features (e.g. phonemes)	Digital Sound Samples	Subword Features (e.g. phonemes)
	Artificial Neural Networks (ANN)	Statistical Features (e.g. LPC coefficients)	Digital Sound Samples	Statistical Features (e.g. LPC coefficients)
	Long-Term Statistics	Statistical Features (Mean and Variance)	Digital Sound Samples	Statistical Features (Mean and Variance)
Training and Testing	Dynamic Time Warping (DTW)	Reference Model Database	Acoustic Sequence Templates	Comparison Score
	Hidden Markov Models (HMM)	Markov Chain	Subword Features (e.g. phonemes)	Comparison Score
	Vector Quantization (VQ)	VQ Network & Codebooks	Statistical Features (e.g. LPC coefficients)	Distortion Value
	Ergodic-HMM's	Markov Chain	Subword Features (e.g. phonemes)	Comparison Score
	Artificial Neural Networks (ANN)	Neural Network with Weights	Statistical Features (e.g. LPC coefficients)	Positive/Negative Output
	Long-Term Statistics	Reference Model Database	Statistical Features (Mean and Variance)	Comparison Score

Chapter 4

Experiments with Existing Techniques from Speech and Music

This chapter will explain the method used to address the first task in the hypothesis (outlined in Chapter 1) for this work, the comparison of existing techniques to determine the best existing technique that can be used for classification. It discusses the equipment used in this project and the implementation details for each of the feature extraction and classification techniques selected.

4.1 Equipment Used

One of the goals of this project is to develop a sound identification and direction detection system that is viable for a commercial application such as a security system. To this end, equipment was chosen to be inexpensive while still producing output that would allow reasonable results. Table 4 shows a list of the equipment used for experimentation.

TABLE 4. EQUIPMENT USED

Equipment Used
2 x Optimus Electret Condenser Replacement Microphones Frequency Response 100 – 10,000Hz Impedance 1k +/- 30% at 1,000Hz Sensitivity -68 +/- 3dB Operating Voltage 1.5V DC Cable Length 1.8m
1 x Sony Minidisc Recorder / Optimus Stereo Cassette Recorder
1 x Go Multi-Voltage Plug Pack 240V AC to 3/4.5/6/7.5/9/12V DC – 500mA Cable Length 1.8M
1 x 1.2m stereo plug 3.5mm stereo plug to 3.5mm stereo plug

Due to this self-imposed budgetary constraint, selection of microphones was limited. Condenser microphones were required for this research (due to their longer range over Dynamic microphones (which have a maximum range of approximately 50cm)), so these were selected. Microphones were then chosen that balanced cost with frequency response, which was the next consideration. Based on these criteria, a pair of condenser microphones with 10,000Hz frequency response were selected.

The tape recorder was specifically chosen for its ability to combine two monaural channels from the two microphones into one stereo channel. While an adapter cable could be used for this, the tape recorder allows for the channels to be properly and easily combined together into one stereo channel. This is necessary to avoid confusion over which channel is the right channel and which channel is the left channel. The minidisc recorder was selected for its high sampling rate. This allowed me to ensure that the data capturing device was not losing important frequency information from the sounds.

4.2 Comparison Experiment

This section discusses the methodology used in my comparison of techniques. It includes the description of the experiment setup, the comparative study method and the implementation details. All calculations were done using Matlab 6 on a Pentium 4 1.6GHz Desktop machine with 528MB of RAM.

4.2.1 Experiment Setup

The experiment consists of tests on eight sounds, each with six different samples. The sounds used for this test are listed below and are some typical sounds that would be classified in a sound surveillance system.

Table 5. Set of Sounds used in the Experiments.

Sound Type			
Jangling Keys	Footsteps (Close)	Footsteps (Distant)	Wood Snapping
Coins Dropping	Footsteps on Leaves	Footsteps on Glass	Glass Breaking

The techniques are tested using a jackknife method, identical to the method used by Goldhor [Goldh93]. A jackknife testing procedure involves training a classification system with all data except the sound sample that will be tested. This sound is then tested against the classification system and the classification is recorded. In cases where the setting of initial weights may affect the classification result (as is the case with LVQ and ANN techniques), training is repeated 5 times, with different weight initializations each time. A correct classification is only recorded if more than three of the training runs are correct. This jackknife procedure is repeated with all six of the samples from each of the eight sounds.

For the experiment setup, sound recording was conducted under quiet conditions. Dual Condenser Microphones were used to record to Sony Minidisc using the maximum sampling rate of 44100Hz, with 16 bits per sample. It should be noted that Sony Minidisc uses the lossy AATRAC3 compression format, but I do not expect the application of the lossy compression used in AATRAC3 to unduly effect the recognition process.

4.2.2 Comparative Study Method

The feature extraction and classification techniques shown in the comparison are tested for their ability to classify non-speech sounds in two ways. First, testing is performed, using these techniques, on non-speech sounds. Data on the parameters, the resulting time taken and the final correct classification rate are recorded. Then, these results are compared with statistics and previous results for the performance of the classification techniques with speech recognition and with musical instrument recognition. This demonstrates how these techniques perform compared against each other and provide an evaluation to the results for non-speech.

Moreover, since feature extraction and classification are both required to recognise a sound, each classification technique must also be tested against each feature extraction technique to determine the best combination of these two techniques. The exception to this is the long-term statistics technique, which generates its own features and therefore requires no feature extraction techniques.

Based on the above and on the selections made in the previous chapter, this produces a set of experiments summarized in the following table:

TABLE 6. COMBINATION OF FEATURE EXTRACTION/CLASSIFICATION TECHNIQUES

	LTS	FE	MFCC	HCC	STFT	FWT	CWT
Learning Vector Quantization		✓	✓	✓	✓	✓	✓
Artificial Neural Networks		✓	✓	✓	✓	✓	✓
Dynamic Time Warping		✓	✓	✓	✓	✓	✓
Gaussian Mixture Models		✓	✓	✓	✓	✓	✓
Long-Term Statistics	✓						

4.2.3 Feature Extraction Implementation – Stationary

In this comparison, I tested three stationary feature extraction techniques, whose implementation is discussed in this section.

4.2.3.1 Frequency Extraction

Frequency Extraction was performed using the Fast Fourier Transform (FFT) routine, which uses the following equation for a DFT:

$$X(k) = \sum_{j=0}^{N-1} x(j)\omega_N^{\ jk} \qquad\qquad k = 0,\ldots,N-1 \tag{3}$$

where $\omega_N = e^{-i2\pi/N}$ and is the frequency we wish to check for, j counts all the samples in the signal and N is the length of the signal being tested. The results of the FFT were then windowed into a set number of bands, each with a constant length. The mean signal power of each band was then taken to produce a reduced FFT feature, with a single value for each band. This FFT feature was then used as input to train the classification system. Empirical testing for several bands (i.e. 64, 128, 256 and 512) revealed that splitting the frequency signal into 256 bands produced the most accurate results. Since non-speech sound covers a wider frequency range than speech (anywhere from 0Hz to 20,050Hz, the approximate limit of human hearing), a 44,100 point FFT (N = 44100) was performed, to allow a greater frequency resolution across all the frequencies required.

4.2.3.2 Mel-Frequency Cepstral Coefficients

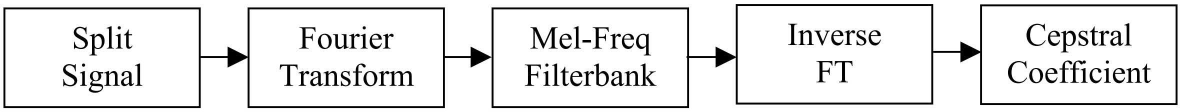


Figure 23 – Applying Mel-Frequency Cepstral Coefficients

I used the MFCC algorithm from the Auditory Toolbox by Malcolm Slaney of Interval Research Corporation [Slaney98]. This toolbox is in wide use in the research community.

The toolbox applies three steps to produce the MFCC. First, it splits the signal into sections (determined by the number of coefficients, which in this implementation is 13) and applies a Hamming window using the standard Hamming window equation:

$$h(k) = 0.54 - 0.46\cos\left(\frac{2\pi k}{N-1}\right) \qquad k = 1,\ldots,N \tag{4}$$

where *N* represents the length of the subset of the signal which is being windowed. A Mel-Frequency Filterbank is then applied to each windowed segment. The mel-frequency filter bank *m* is built using a logarithmic frequency mapping expressed by the following relation [Lilly00]:

$$m = \frac{1000\ln\left(1+\frac{f}{700}\right)}{\ln\left(1+\frac{1000}{700}\right)} \approx 1127\ln\left(1+\frac{f}{700}\right) \tag{5}$$

where *f* represents the range of frequencies in the signal. The application of this filterbank produces a series of magnitude values (one for each filter). A Cepstral Coefficient formula (shown in the next section) is then used to perform a frequency warping using these magnitude values to produce MFCC and these features are then collected into a single feature vector, which is more appropriate for training a network. Special attention was paid to removing the first scalar within the vector, which represents the total signal power and is therefore too sensitive to the amplitude of the signal (as suggested by Lilly [Lilly00] and Gold [GoldM00]).

4.2.3.3 Homomorphic Cepstral Coefficients

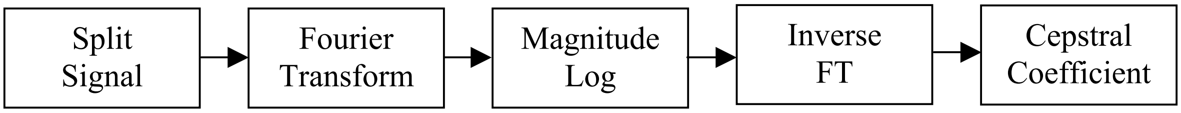


Figure 24 – Applying Homomorphic Cepstral Coefficients

My implementation of the Homomorphic Cepstral Coefficient (HCC) algorithm was based on the MFCC algorithm from the Auditory Toolbox by Malcolm Slaney of Interval

Research Corporation [Slaney98] (Figure 24). This algorithm was modified to produce HCC as opposed to MFCC by removing convolution with the Mel Frequency Filterbank.

To apply this method, we first split the signal using Hamming windows. We then calculate the cepstrum (X(n)) for each of the windowed segments. The cepstrum is the Fourier transform of the log magnitude spectrum. Once we have done this, we can calculate cepstral coefficients using the following relation [Lilly00]:

$$y(k) = w(k)\sum_{n=1}^{N} X(n)\cos\frac{\pi(2n-1)(k-1)}{2N} \quad k = 1,\ldots,N \tag{6}$$

where $$w(k) = \begin{cases} \sqrt{1/N}, & k = 1 \\ \sqrt{2/N}, & 2 \le k \le N \end{cases}$$

and *n* is the length of the windowed segment being manipulated. I selected the first 13 coefficients produced by this relation. These features were then used in a vector notation, which is more appropriate for training a network. As with the MFCC, special attention was paid to removing the first scalar within the vector, which represents the total signal power and is therefore too sensitive to the amplitude of the signal (as suggested by Lilly [Lilly00] and Gold [GoldM00]).

4.2.4 Feature Extraction Implementation - Non-Stationary

This section explains and discusses the implementation details of the three non-stationary feature extraction techniques that I tested in my comparative study. In the case of STFT and CWT, a principal component analysis (PCA) was used after feature extraction to reduce the dimensionality of the resulting signal. An adaptive algorithm was used to calculate the maximum number of principal components required for the training data used (based on the energy in each dimension and a variable threshold). In both cases, a threshold value of 1% was found to produce the most accurate results. This process reduced the size of the signal significantly. For the STFT, it reduced the size of the matrix from 128x67 (or 8643 features) to 18x67 (or 1206 features). For CWT, it reduced the size of the matrix from 8820x55 (or 485,100 features) to 12x55 (or 660 features).

4.2.4.1 Short-Time Fourier Transform

A Short-Time Fourier Transform (STFT) was implemented using Matlab's FFT routine and a rectangular windowing algorithm. This approach allowed finer control over the resultant resolution of the STFT by allowing me to systematically change the number of samples in both time and frequency. The signal was windowed and then a FFT was calculated for each windowed segment [Cohen95]. This produces the following relation for the calculation of a STFT:

$$S_t(\omega) = \frac{1}{\sqrt{2\pi}} \int e^{-j\omega t} s(\tau) h(\tau - t) \quad d\tau \tag{7}$$

where ω is the frequency, τ is the signal length, *s(t)* is the signal and *h(t)* is the windowing function. This algorithm was implemented in Matlab with a variable window size parameter (allowing the resolution of the STFT to focus more closely on either time data or frequency data). Empirical testing of several values for window size (ie. 50, 100, 150 and 200), over small repeated classification experiments, using various classification techniques, showed that a window size of the sample frequency scaled by 100 produced the most accurate results when tested.

4.2.4.2 Fast Wavelet Transform

For the fast wavelet transform (FWT), I used the periodized, orthogonal *FWT_PO* algorithm from the Matlab Wavelab toolbox by Stanford University [Dono02]. Like all FWT algorithms, this algorithm convolves the signal with a filter and then applies a subsampling relation:

$$y(n) = \sum_{k=-\infty}^{\infty} h(k) \cdot x(2n - k) \tag{8}$$

This subsampling equation is then repeated on the lower half of the signal (and optionally the high half of the signal), such that:

$$y_{high}(k) = \sum_{n} x(n) \cdot g(2k - n) \tag{9}$$

$$y_{low}(k) = \sum_{n} x(n) \cdot h(2k - n) \tag{10}$$

As a filter (ie. h(*t*) and g(*t*)) , I applied the popular Daubechies filters [Daube92] to the signal. Dabauchies filters allow for the perfect reconstruction of a signal from the FWT. A vanishing moment variable can be set for these filters upon generation; however, the value of this coefficient seemed to make little difference to the classification rate. Due to the nature of the FWT, the signal requires no PCA to reduce its dimensionality, meaning the result of the FWT can be used directly in the classification system.

4.2.4.3 Continuous Wavelet Transform

For the continuous wavelet transform (CWT), I used the discretized *CWT* algorithm from Stanford Unviersity's Matlab Wavelab toolbox [Dono02]:

$$CWT_x^{\psi}(\tau, s) = \Psi_x^{\psi}(\tau, s) = \frac{1}{\sqrt{|s|}} \sum_{t=1}^{N} x(t) \psi^{*}\left(\frac{t - \tau}{s}\right) \tag{11}$$

where τ represents translation, *s* represents scale and $\psi(t)$ is the mother wavelet, which was chosen to be the Morlet mother wavelet (Figure 25), defined as:

$$\psi(t) = e^{jat} e^{\frac{-t^2}{2s}} \tag{12}$$

where *a* is a modulation parameter and *s* again represents scale [Daube92]. This mother wavelet has been used for recognition tasks in the past and produced acceptable results [OrrPh01].

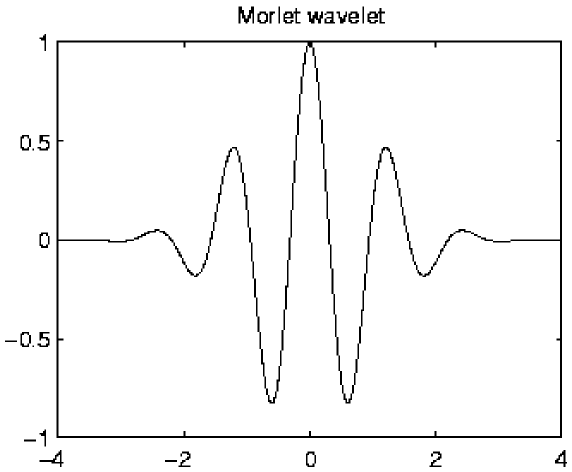


Figure 25 – The Morlet Mother Wavelet

4.2.5 Classification

Four classification techniques will be tested in this comparison. The implementation of each of these techniques will be discussed in this section.

4.2.5.1 Learning Vector Quantization

Learning vector quantization (LVQ) was implemented using the inbuilt LVQ routines in Matlab's neural network toolbox. The network was initialized with 50 competitive neurons and a learning rate of 0.05. Trials were performed using each set of sounds in the experiment suite and running until the network converged. Using this approach, it was found that ~ 50 iterations gave the most accurate classification rate.

4.2.5.2 Artificial Neural Networks

The Artificial Neural Network (ANN) was implemented using the fast back propagation algorithm (BPA) in the Matlab neural network toolbox. I used the Levenberg-Marquardt Back Propagation algorithm and tansig activation functions. The network was initialized with 50 hidden neurons and a learning rate of 0.05. The limit of the sum-squared error was set to 0.001 and the momentum constant was set to 0.95. These settings allowed the network to converge in ~ 500 iterations. Because ANN's work by random initialization of weights, several different runs were performed using the ANN and the results were averaged. This allowed a more consistent sampling of different random weight values.

4.2.5.3 Dynamic Time Warping

I implemented Dynamic time warping (DTW) using the *dtw* function in the Auditory Toolbox developed by Malcolm Slaney [Slaney98]. DTW uses a dynamic programming approach (as opposed to a linear approach) to align the length of the signal with the length of the reference signal. DTW minimizes a global error by using a sequential optimization strategy where the current estimate of the global optimization function is updated for each possible step. Enough information is retained on the set of plausible hypotheses to allow the set of choices for the minimal global error to be reconstructed at the end. This means that a signal warped using DTW more closely resembles the original signal than a signal warped using a linear time warp [GoldM00].

To use DTW, feature extraction was first applied to each signal and then the test signal was warped against each of the reference signals and the error between these two signals was recorded. The smallest error was taken to represent the closest class of sound.

4.2.5.4 Long-Term Statistics

The Long-Term Statistics (LTS) were implemented using the mean and covariance functions available in the standard Matlab distribution. Mean and covariance were calculated for each of the reference signals and stored in a matrix. The mean and covariance of the test signal was then compared to this matrix. The closest match was selected as the correct class. If the closest mean and covariance occurred in different classes, the test was deemed to be inconclusive.

4.2.5.5 Gaussian Mixture Models

I implemented Gaussian Mixture Models (GMM) using the Netlab toolbox developed by Ian Nabney [Nabney02]. GMM's use an unsupervised learning technique to determine the centres and variance of clusters within a search space. The GMM's in Netlab are initialized using the k-means classification technique and then trained with an Expectation-Maximization (EM) algorithm. I trained a GMM for each of the classes of sound in the domain. Once this was done, I worked out the probability of each of these models on the training data. Because each of the classes has equal priority, testing the system then simply involves finding the class C_i that produces the highest $p(\vec{x} \mid C_i)$, where $\vec{x}$ is the data under test.

Chapter 5

Results and Discussion (Existing Techniques)

In this chapter, I present the results of my comparison and discuss the validity of these techniques to the domain of environmental sound recognition. Results and their subsequent interpretation are presented for each classification system using all feature extraction techniques (Table 7 - Table 12 and Figure 26 - Figure 32). The result shown is the total classification rate over all sounds and all samples using the jackknife technique (which is explained in the previous chapter).

5.1 Non-Speech Suitability Testing

This subsection will present the results for the experiments on non-speech sounds, using various feature extraction and classification techniques. Results for the LVQ classifier are presented below (Table 7, Figure 26). These results show that the most accurate feature extraction technique is the CWT technique, which makes a correct classification 54% of the time. The worst technique is the STFT technique, which is unable to make any classifications. The success of the CWT technique can be expected, since this technique presents a large amount of information to the LVQ classifier. However, the poor performance of the STFT technique can be explained by the difficulty in finding a good resolution in both time and frequency using the STFT technique.

TABLE 7. LEARNING VECTOR QUANTIZATION (LVQ)

Method	% Correct
FT	50%
MFCC	37.5%
HCC	12.5%
STFT	0%
FWT	12.5%
CWT	54%

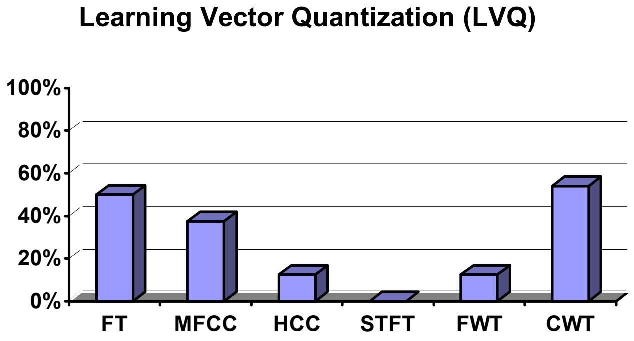


Figure 26 - Comparison of LVQ for Environmental Sound Recognition

Results for the ANN classifier are presented below (Table 8, Figure 27). Surprisingly, despite previous successful results using ANN's in speech, the results here are quite low. This can be explained due to the lack of ability for ANN's to linearly discriminate different results. The disparity between the ANN and LVQ results will be discussed more in the following section.

TABLE 8. ARTIFICIAL NEURAL NETWORK (ANN)

Method	% Correct
FT	0%
MFCC	4%
HCC	0%
STFT	0%
FWT	0%
CWT	41%

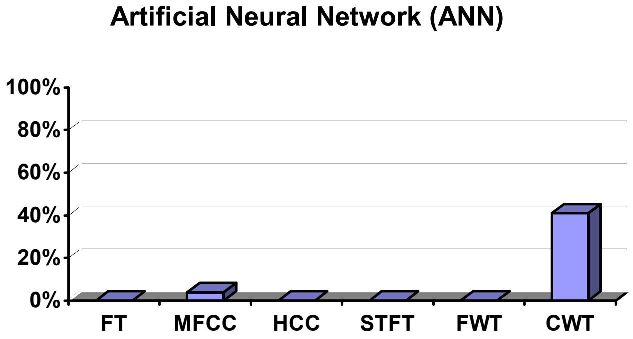


Figure 27 - Comparison of ANN for Environmental Sound Recognition

Results from the DTW classifier are presented below (Table 9, Figure 28). Of all of the results in this experiment, the results for the DTW classifier are by far the best, with a 70% classification rate using the MFCC or CWT feature extraction techniques. Once again, this is consistent as, before the introduction of HMM's in the speech recognition field, the DTW classifier as the common choice for recognizing speech signals. This is due to low computational complexity coupled with good results.

TABLE 9. DYNAMIC TIME WARPING (DTW)

Method	% Correct
FT	66%
MFCC	70%
HCC	29%
STFT	58%
FWT	12%
CWT	70%

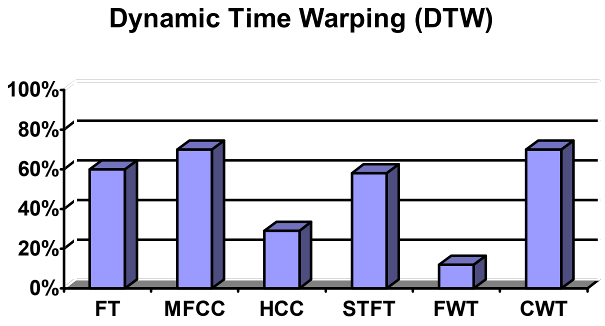


Figure 28 - Comparison of DTW for Environmental Sound Recognition

Results are presented below for the LTS technique (Table 10, Figure 29). These results are quite unremarkable (at 29%), but this is to be expected, as LTS is primarily used for speaker recognition and not speech/non-speech classification. Nonetheless, it is interesting to see which of these two types of techniques (speaker vs. speech) is more appropriate for this domain.

TABLE 10. LONG-TERM STATISTICS (LTS).

Method	% Correct
FT	29%
Power FT	29%

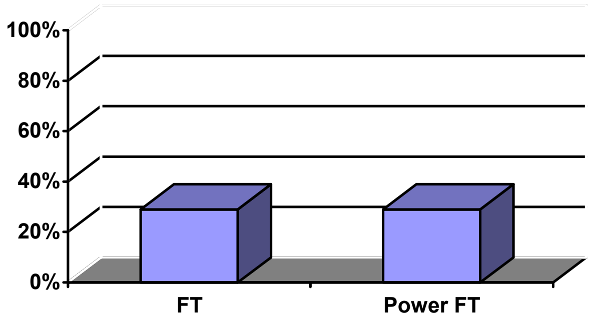


Figure 29 - Comparison of LTS for Environmental Sound Recognition

Finally, results are presented for the GMM classifier below (Table 11, Figure 30). GMM is the classifier most often used in music recognition (typically with the MFCC feature extraction technique). Despite this, it still performs quite well in the non-speech domain, with a result of 46% using either the MFCC or STFT technique. This shows the closeness between the music, speech and non-speech domains when it comes to choice of classifier. However, it also shows were some differences may occur (with typical scores for GMM in music being closer to 80%).

TABLE 11. GAUSSIAN MIXTURE MODELS (GMM)

Method	% Correct
FT	21%
MFCC	46%
HCC	12%
STFT	46%
FWT	25%
CWT	21%

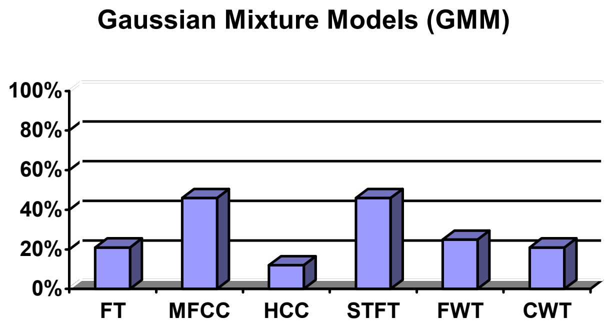


Figure 30 - Comparison of GMM for Environmental Sound Recognition

5.2 Interpretation of Non-Speech Results

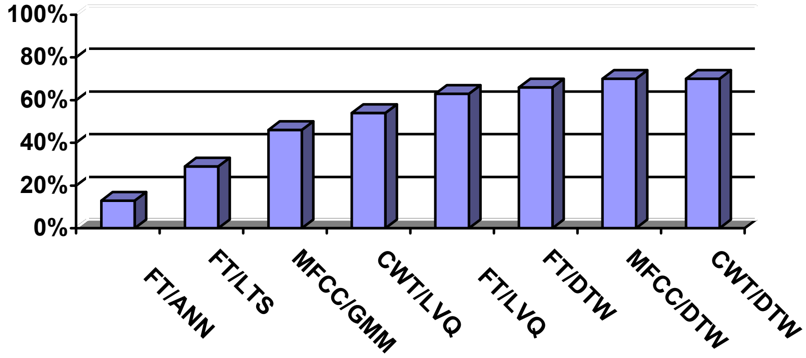


Figure 31 - Comparison of Best Results for Non-Speech Sound Recognition.

The results obtained for this set of experiments are somewhat surprising. Even though the results from speech recognition suggest that an ANN will outperform the LVQ technique, the opposite occurs for non-speech recognition. I propose that this is due to the closeness of the various environmental sounds presented to the two networks.

It is widely accepted that one of the main advantages of LVQ over ANN's is their ability to correctly classify results even where classes are similar. In this case, sounds such as footsteps (close) and footsteps (distant) appear the same but contain slightly lower or higher amplitudes. LVQ is able to classify these sounds properly where the ANN cannot distinguish them. Furthermore, the detailed results of each test show that the ANN was classifying *footsteps (close)* as *footsteps (distant)* and vice versa. To support this hypothesis, further tests were performed on the ANN using several different MSE values (to allow more training time). The results of these different experiments are presented in Table 12 and compared in Figure 32.

TABLE 12. FURTHER ANN RESULTS FOR ENVIRONMENTAL SOUND RECOGNITION.

Method	% Correct (MSE – 0.001)	% Correct (MSE – 0.0001)
FT	0%	0%
MFCC	4%	4%

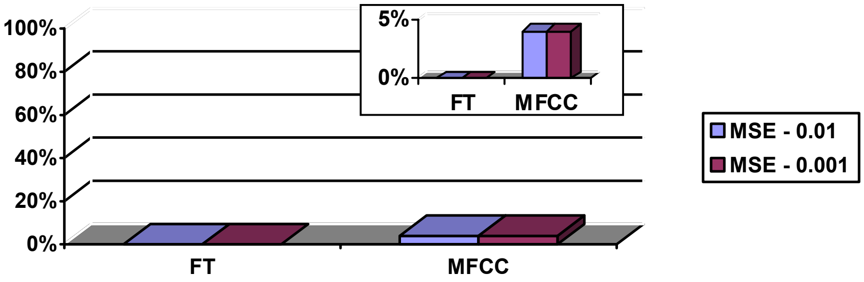


Figure 32 - Comparison of ANN Results with alternative MSE values

From these results, it can be seen that the ANN results remain the same regardless of the MSE value. This suggests that the ANN has problems training the sample sounds, most likely because these sounds are non-linearly separable.

The performance of the Mel Frequency Cepstral Coefficient (MFCC) feature extraction algorithm over the Fourier Transform (FT) Based Frequency Extraction algorithm is also interesting. Surprisingly, in all cases except when DTW or GMM is used as a classifier, the FT algorithm outperforms the MFCC algorithm. However, to achieve the same results with the FT algorithm, it has to spend almost 10 times as long training as the MFCC algorithm does.

For the LVQ tests, it seems that the MFCC algorithm can achieve a maximum classification rate of approximately 37.5%. In contrast, the FT algorithm can achieve a slightly higher rate, reaching its maximum at around 50%.

The DTW algorithm also produces surprising results. This algorithm shows only a small difference (equivalent to one classification) in performance between the MFCC algorithm and the FT algorithm. This is in contrast to the large difference between these algorithms in the LVQ tests. DTW also performs classification much quicker than the LVQ and ANN techniques. This is most likely due to the fact that DTW does not require any training and instead relies on a series of reference models. The downside to this approach is the extra storage space required for these templates.

In contrast to results presented by Lilly [Lilly00], my results show a substantial difference in classification between the HCC technique and the MFCC technique. Due to the fact that other researchers report similar classification rates using these two techniques (eg. [Lilly00], [GoldM00]), implementation of these techniques could conceivably be improved. However, since MFCC seems to be the more popular technique and produces the better results of the two techniques, at this stage I will continue to use it in its current form.

For time-frequency techniques, these results show that the combination of CWT with DTW produces the best results, with the CWT producing a top comparative study percentage of 70% with DTW. Results are also promising for the use of the CWT with LVQ and ANN, producing top results of 54% and 41% respectively.

The results from this comparative study reveal some interesting findings. It is interesting to note the poor performance of the STFT algorithm with both ANN and LVQ. Despite providing average performance using DTW (29%), a STFT combined with either an LVQ or ANN network fails to provide any correct classifications. Although results for stationary feature extraction techniques support this low classification rate for ANN’s, performance with LVQ is surprising. Further research may endeavour to manipulate the resolution of the STFT in order to improve LVQ classification rate (an issue that does not affect the Wavelet family of time-frequency techniques). Nonetheless, the performance of STFT with GMM produces quite good results, suggesting that maybe the problem lies with the learning algorithms in LVQ/ANN.

Overall, it is clear from these results that the DTW classification technique could be considered the most suitable for environmental sound recognition, especially in a surveillance context. Not only does the DTW technique perform the best of all the techniques that I investigated and compared, but it also produces the results quickly (under 1 second for a testing classification as compared to an average of 5 seconds for artificial intelligence techniques). However, this technique still needs to be investigated, as it uses a template matching method, which could turn out to be a weakness when the amount of sounds in the database increases. Nevertheless, there is ample opportunity to improve the technique in these circumstances with the use of difference measures to produce general representations of each class of sound.

For feature extraction, the results are not so clear-cut. They show that the pseudo-frequency technique of MFCC's produces a classification rate of 70% when used with the DTW technique. However, they also show that the same classification rate can also be achieved using the time-frequency technique of CWT. The relative effectiveness and classification efficiency of these two techniques will become apparent when they are applied to a larger database of sounds. Once again, opportunity exists to improve upon these techniques by systematic testing and refinement of these techniques over several iterations. The results presented in this work demonstrate the obvious superiority of these techniques over the other techniques that I investigated for environmental sound recognition.

It could be argued that these classification rates do not parallel with the accuracy that can be achieved in speech recognition using HMM's. However, as was explained earlier, they are not suitable for environmental sounds, because HMM uses a discrete model.

In general, due to the variability inherent in environmental sounds, accuracy with techniques such as DTW will probably always be lower than the classification rate that can be achieved in the more constrained area of speech recognition.

5.3 Comparison of Classification Results with Speech & Music

This section shows results of selected techniques from the results section (LVQ, ANN, GMM) in other related domains (speech recognition and musical instrument recognition). This allows a comparison of these techniques among the different domains.

5.3.1 Speech Recognition

For the sake of completeness, I compare my LVQ and ANN non-speech results with results reported for speech recognition systems. Due to the current popularity of HMM methods in speech recognition at the present time, results for DTW are difficult to find, therefore no DTW results are presented.

For ANN's, a selection of results from Castro and Perez [Castro93] are shown in Table **13**. Their results were taken on an isolated word recognition set with typically high classification error, the Spanish EE-set. Castro and Perez's Multi-Layer Perceptron (MLP) used the back propagation algorithm, contained 20 hidden neurons and was trained over 2000 iterations with various amounts of inputs. The figures given are the MLP's estimated error rate with a 95% confidence interval.

For LVQ, results from Van de Wouver e.a. [Vandew96] are shown in Table **14** for both female and male voices. These results present statistics for both a standard LVQ implementation for speech recognition and an implementation of LVQ that then has fuzzy logic performed on it (FI-LVQ). As can be seen from the results, the use of LVQ for speech recognition produces rather low recognition results.

TABLE 13. ANN FOR SPEECH RECOGNITION

Number of Inputs	% Correct
550 inputs	80.3%
220 inputs	83.7%

TABLE 14. LVQ FOR SPEECH RECOGNITION

Method	% Correct (Female)	% Correct (Male)
Standard LVQ	36%	29%
FILVQ	60%	64%

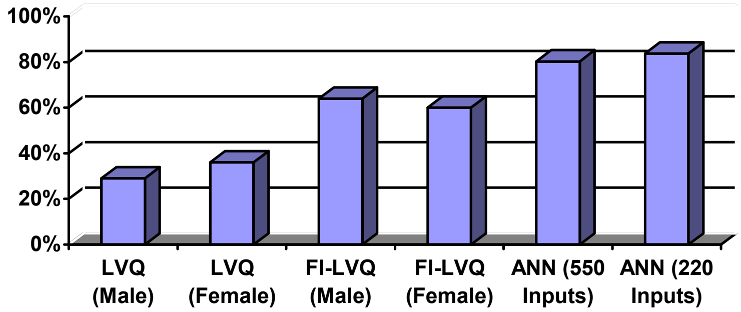


Figure 33 - Comparison of Speech Recognition Results.

Compared to the results from my comprehensive comparison, these results (Figure 33) are quite interesting. The comparison shows the best result for LVQ in non-speech is 54%, when combined with a CWT feature extraction technique. The results for LVQ in speech are only between 6 and 10% above this, even with the application of Fuzzy Logic. Without the application of fuzzy logic, the results for speech using LVQ are 18% worse than the non-speech results.

For ANN's, the results from speech show a much higher percentage rate than the results from non-speech. For non-speech, the best result is 41% using the CWT feature extraction technique. This is much lower than the 83.7% achieved using ANN's for speech. I believe this is due to the non-speech data being non-linearly separable, and will elaborate more on this in the discussion section.

5.3.2 Musical Instrument Recognition

I looked at techniques used in musical instrument recognition, considering that it might be closer to environmental sound recognition than to speech. In this field, two seminal works stand out as using GMM's. Marques & Moreno [MarqMor99] used several different techniques for the feature extraction and classification of musical instruments, with the best results coming from a combination of mel-frequency cepstral coefficients with either Gaussian Mixture Models (GMM) or Support Vector Machines (SVM). Martin [Martin99] reports initial results from Marques showing a classification rate of 72% for professional recordings and 45% for non-professional recordings. Since then, a further technical report from Cambridge Research Laboratory [MarqMor99] shows a classification rate of 70% when using mixed data, increased to 98% when using data from a single source. This suggests the applicability of robustness techniques (also tested in speech recognition) to this domain, in order to combat this problem with variable training/test data. These results are summarized in Table 15.

Brown also presents a system using cepstral coefficients [Brown99]. In this case, the system uses Q-cepstral coefficients with a Gaussian Mixture Model for classification (one model for each instrument). On independent, noisy samples of music, the system achieves a classification rate of 94%, between oboe and saxophone recordings. However, this system achieves only 84% when extended to include four samples of instrument (oboe, saxophone, flute and clarinet). These results are summarized in Table 16.

TABLE 15. MFCC/GMM FOR MUSICAL INSTRUMENT RECOGNITION

Mixed Data	Single Source Data
70%	98%

TABLE 16. Q-CEP/GMM FOR MUSICAL INSTRUMENT RECOGNITION

2 Instrument Types	4 Instrument Types
94%	84%

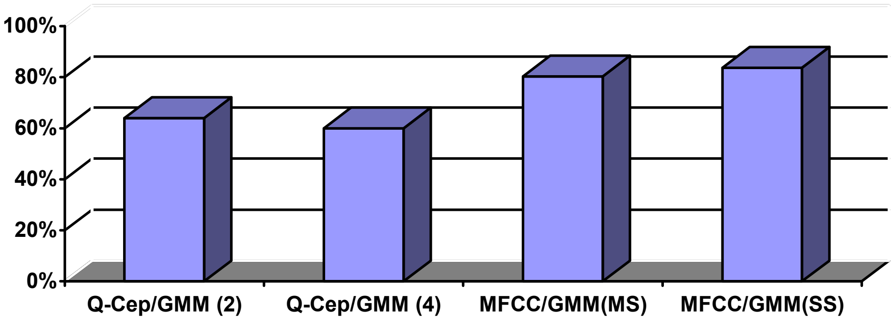


Figure 34 - Comparison of Musical Instrument Recognition Results.

If we compare these results (Figure 34) to the results for non-speech sounds, we see that GMM's can be much better applied in the musical instrument domain. The best result for GMM's in non-speech is 46%, while musical instrument recognition can achieve a 94% recognition rate. However, it must be considered that the results for non-speech are taken on 8 classes of sounds. The results for musical instrument recognition are taken on only 2 and 4 classes of sounds. This could account for the higher recognition rate. Brown also shows with her results that classification rate decreases quickly as the number of classes increase (down from 94% to 84% for 2 and 4 classes respectively). It is possible that, if Brown ran her tests on 8 classes of musical instrument, she would get similar results to those shown for non-speech sounds.

5.4 Conclusion of Comparative Study

This chapter presented the results of a comparative study of time-frequency and frequency-based (or pseudo-frequency) techniques for non-speech environmental sound recognition and showed the applicability of either of these representations to environmental sound recognition. However, classification rates do not parallel with the accuracy that can be achieved in speech recognition using Hidden Markov Model's. Due to the variability inherent in environmental sounds, accuracy is probably lower than with the more constrained area of speech recognition.

The results revealed that a combination of continuous wavelet transform with dynamic time warping produces a classification rate of 70%. Combination of Mel-Frequency Cepstral Coefficients with dynamic time warping also produced 70%. From this, it is clear that DTW is a superior technique for classification of environmental sounds. Now that this obvious superiority of techniques has been shown, further refinements can be performed on these techniques to possibly produce even better classification rates.

However, any refinements to the technique will still be constrained by the larger pattern search space that is underpins the nature of these techniques. If I can develop a technique that will reduce the size of this pattern search space, I should be able to produce a more efficient and accurate system. The following two chapters discuss the investigation of advanced techniques for classification of non-speech environmental sounds to facilitate this.

Chapter 6

Advanced Classification Techniques

The results presented in the previous section are good, but they do not parallel with results that can be obtained in the (albeit, much more constrained) area of speech using techniques such as Hidden Markov Models (HMM's). This is because the area of speech (and indeed, of music) requires a much smaller finite set of sounds to perform good classification. The human vocal system can only produce a certain number of phonemes and a well-trained speech recognition system only has to recognize each of these phonemes. In contrast, an environmental sound recognition system is required to classify a much larger set of sounds, in a near infinite set.

In order to combat this problem, I propose the use of advanced techniques, the purpose of which are to reduce the size of the pattern matching search space into smaller sets of classes, in much the same way as the splitting of spoken dialogue into words and phonemes achieves this for speech. Some sort of structured taxonomy would allow me to achieve this goal.

Two structured taxonomies are proposed in this section:

- Taxonomy Based on Source-Source Collisions & Physics
- Automatically Generated Taxonomy (using the C4.5 technique [Quinlan93]).

Each of these structured classification techniques will be discussed in this chapter. In addition, this section will also discuss the experimental method and equipment used.

6.1 Experimental Method

The taxonomy is implemented in three levels, with the first level using the features described for classification and the second and third levels using existing speech recognition techniques.

Recognition of a sound requires two independent stages: feature extraction and classification. Therefore, each classification technique must also be tested against each feature extraction technique to determine the best combination of these two techniques. This produces a set of experiments summarized in the following table (Table 17):

TABLE 17. COMBINATION OF FEATURE EXTRACTION/CLASSIFICATION TECHNIQUES

	MFCC	STFT	CWT
LVQ	✓	✓	✓
DTW	✓	✓	✓

6.2 Experiment Setup

The techniques are tested using a jackknife method, identical to the method used by Goldhor [Goldh93]. A jackknife testing procedure involves training a classification system with all data except the sound samples that will be tested. This sound is then tested against the classification system and the classification is recorded. In cases where the setting of initial weights may affect the classification result (as is the case with the LVQ technique), training is repeated 5 times, with different weight initializations each time. A correct classification is only recorded if more than three of the training runs are correct.

For the experiment setup, sound recording was conducted under quiet conditions. Dual Condenser Microphones were used to record to Sony Minidisc using the maximum sampling rate of 44100Hz, with 16 bits per sample. It should be noted that Sony Minidisc uses the lossy AATRAC3 compression format, but I do not expect the application of the lossy compression used in AATRAC3 to unduly affect the recognition process. In addition to the sounds recorded using Minidisc, additional sounds were used from the "Non-speech sound dry source database" developed by RWCP [Hiyane02]. Table 18 shows an example sound for each classification on the first level.

TABLE 18. EXAMPLES OF EACH TYPE OF SOUND USED.

Sound Type	***Example Sound***
Solid-Solid	*Soda can hit with a metal stick*
Solid-Liquid	*Water dripping into empty sink*
Solid-Gas	*Deodorant can spraying*
Liquid-Liquid	*Pouring water into a full glass*
Liquid-Gas	*Bubbles escaping from boiling water.*
Gas-Gas	*N/A*

6.3 Taxonomy Based on Physics

It seems evident that, as the amount of sounds that a system needs to recognise increases, the efficiency and accuracy of the system will decrease. This is because the system will take longer to train, the more sounds it needs to recognise. Also, as similar sounds are trained in the system, the difference between these sounds will be too fine to allow a distinction, meaning that accuracy will decrease.

In order to combat this decrease in usefulness as the amount of sounds increase, I propose the development of an environmental sound taxonomy, which classifies sounds on several levels before recognition. The advantage of this approach is that each classification level contains a smaller set of sounds, increasing the accuracy and efficiency of the system. Although taxonomies have been used in both speech [LiSet01] and music [MartYo98] in the past, no such taxonomy has been developed for environmental sounds.

However, there is a problem with the development of this taxonomy. What will be the higher level groups in the hierarchy on which the sounds are classified? In order to answer this question, I propose the development of a novel environmental sound alphabet (as suggested by Ballas & Howard). Due to the fact that human beings perceive sounds based on their semantic meaning and not on their characteristics [Ballas87], I propose that this is the most appropriate way for my system to classify sounds.

So, what will be the components of this environmental sound alphabet? Since the domain being researched is environmental sounds, it seems that the most logical selection of characters would be related to this domain. Within language, each character roughly represents a phonetic sound produced by the source of the sound (the human voice box and vocal tract). In environmental sounds, an alphabetic character could also represent the source of a sound. The lowest level of recognition relates to recognising the representative objects with which the sound was made and their physical states. As recognised in the field of physics, three physical states of objects exist [Tipler91]. These are:

1. Solid
2. Liquid
3. Gas

These states could therefore be used for classification of non-speech sounds. In addition, the specific properties of each physical object could also be recognised. For instance, solids produce different sounds depending on whether they are metal, glass or wood. These properties could also be used for classification.

Based on this information, my system will initially classify sounds using a three level hierarchy. On the first level, the system will identify the sound as belonging to one of six classes. Each class will look for the interaction of two objects, each related to one of the three states of matter in the environment (solid, liquid, gas). This will produce six classes that a sound could belong to:

1. Solid-Solid (S-S)
2. Solid-Liquid (S-L)
3. Solid-Gas (S-G)

4. Liquid-Liquid (L-L)
5. Liquid-Gas (L-G)
6. Gas-Gas (G-G)

A sound will be identified as belonging to one of these classes and then the appropriate second level classification will occur.

The next level of the system can classify sounds based on their material characteristics. For instance, a solid can be wood or glass or metal etc. This level of classification will classify each solid on these characteristics and will classify gases and liquids based on similar characteristics. Some examples could include:

- Metal on Metal (M-M)
- Metal on Wood (M-W)
- Water on Metal (W-M)
- Water on Wood (W-W)
- Steam through Plastic (S-P)
- Steam through Metal (S-M)
- Steam in Water (S-W)

Finally, the third level will contain the standard pattern recognition approach to recognition, but with one important difference. There will be several networks, each based on the different sounds produced by the previous classification. In this way, a sound can be tested against a much smaller network that has been trained with a smaller (but more specific) selection of sounds. This is hoped to improve the accuracy and efficiency of the system.

Figure 35 shows a hierarchy using these classification techniques.

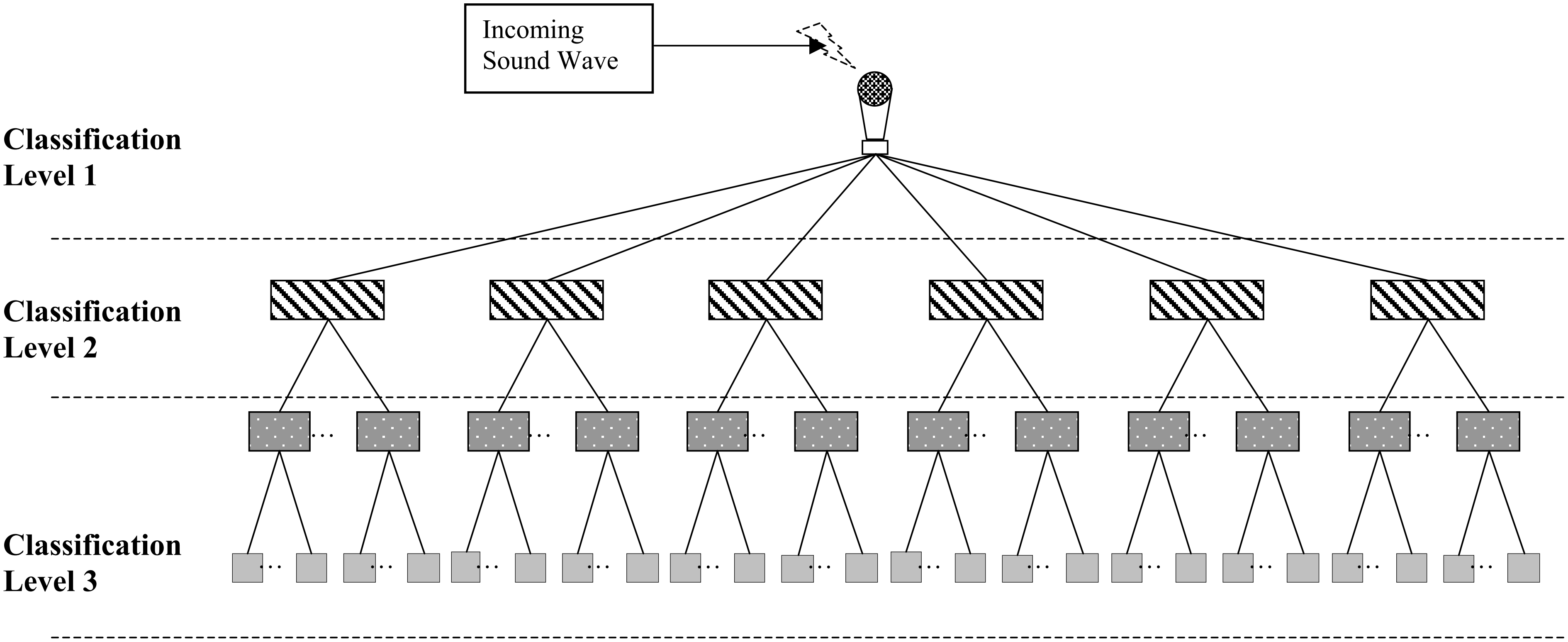


Figure 35 – Sound Taxonomy with Environmental Sound Alphabet Components

Techniques such as the structured taxonomy proposed have previously been used in domains other than environmental sound recognition. In independent work, Martin [Martin99] proposed the use of a similar taxonomic approach to the classification of musical instruments, which are suitable to this type of approach.

There is some parallelism between Martin's work and ours. He proposes the use of a *taxonomic hierarchy* for the classification of musical instrument sounds. Martin suggests a multi-level hierarchy, where each level splits instruments into their natural categories (woodwind, brass, percussion; alto, tenor, bass etc). On each level, a selection of features is chosen based on the discriminatory properties of the sound under test. Typical features include items such as pitch, pitch variance, vibrato strength and onset duration. Classification is then performed using a simple maximum-likelihood estimation. Martin also suggests several enhancements to ensure a classification does not get "stuck" in the incorrect leaf node. Correct classification emerges from the lowest level of the hierarchy, with results (on realistic music samples, not isolated tones) of 90% for instrument families and 70% for individual instruments.

In contrast to the work of Martin, environmental sounds do not have a natural set of classification indexes. In order to overcome this, I propose the development of an environmental sound taxonomy. This will be done in the three levels already proposed. The structure of each level will be described in the following sections.

6.3.1 Level One Filtering

For this first-level filtering, I have developed several features to show the distinction between the different types of sounds:

Impact Frequency (IF) Feature - Using a Short-Time Fourier Transform (STFT), determines the impact moment of the sound and then obtains the Fourier transform (and therefore the fundamental frequency) at that time.

Fundamental Frequency (FF) Feature – Performs a Fourier transform on the entire signal and determines the fundamental frequency for the entire signal.

Impact Tail Sum (ITS) Feature – Determines the activity of frequencies above a set frequency *f* at the impact point. Typically liquid-related sounds contained a much lower concentration of these frequencies (being generally much more tonal).

General Tail Sum (GTS) Feature – Calculates the activity of frequencies above a set frequency *f* for the entire waveform. Liquid-Gas sounds contain lower frequency values for this feature.

These features were then applied in a systematic fashion to determine the type of sound being dealt with. Figure 36 shows a decision tree outlining how these features were implemented (where *TH* is a empirically determined threshold value, f_1 is 10000Hz, f_2 is 2000Hz and f_3 is 4000Hz).

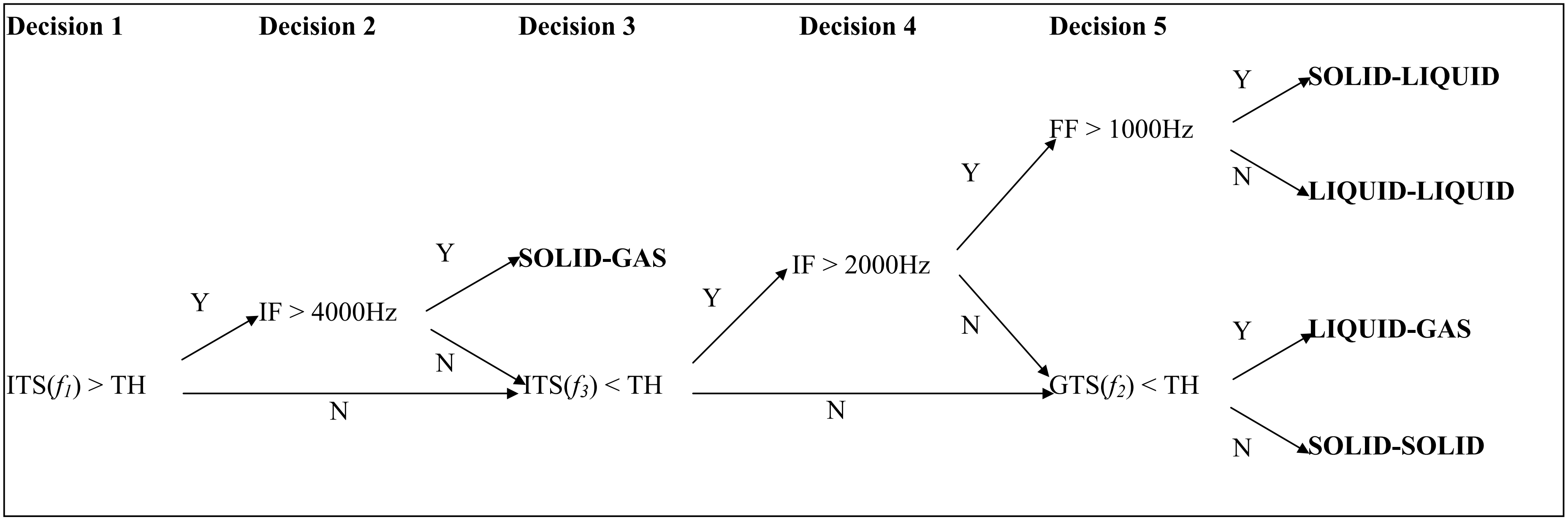


Figure 36 – Decision Tree showing the Taxonomies Classification on the 1st Level

6.3.2 Level Two & Three Classification

For the second and third level recognition in the taxonomy, I selected techniques previously used for speech recognition (see Chapter 2 & Chapter 3). The techniques I selected were:

Feature Extraction:

- Mel-Frequency Cepstral Coefficients
- Short-Time Fourier Transform
- Continuous Wavelet Transform

Classification:

- Dynamic Time Warping
- Learning Vector Quantization

6.3.2.1 Feature Extraction Techniques

In this comparison, I tested three feature extraction techniques, whose implementation is discussed in this section.

Mel-Frequency Cepstral Coefficients

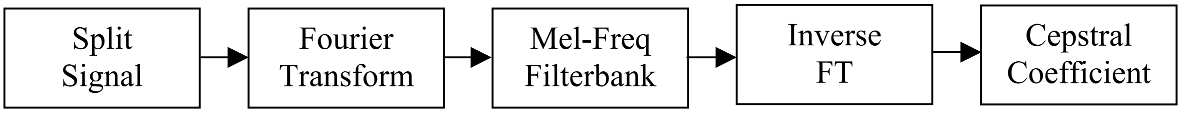


Figure 37 – Applying Mel-Frequency Cepstral Coefficients

I used the MFCC algorithm from the Auditory Toolbox by Malcolm Slaney of Interval Research Corporation (1998) [Slaney98], which is in wide use in the research community. The toolbox applies several steps to produce the MFCC (Figure 37). First, it splits the signal into sections (determined by the number of coefficients, which in this implementation is 13) and applies a Hamming window using the standard Hamming window equation:

$$h(k) = 0.54 - 0.46\cos\left(\frac{2\pi k}{N-1}\right) \qquad k = 1,\ldots, N \tag{13}$$

where *N* represents the length of the subset of the signal which is being windowed. A Mel-Frequency Filterbank is then applied to each windowed segment. The mel-frequency filter bank *m* is built using a logarithmic frequency mapping expressed by the following relation:

$$m = \frac{1000\ln\left(1+\frac{f}{700}\right)}{\ln\left(1+\frac{1000}{700}\right)} \approx 1127\ln\left(1+\frac{f}{700}\right) \tag{14}$$

where *f* represents the range of frequencies in the signal. The application of this filterbank produces a series of magnitude values (one for each filter). A Cepstral Coefficient equation:

$$y(k) = w(k)\sum_{n=1}^{N} X(n)\cos\frac{\pi(2n-1)(k-1)}{2N} \qquad k = 1,\ldots, N \tag{15}$$

where $w(k) = \begin{cases} \sqrt{1/N}, & k = 1 \\ \sqrt{2/N}, & 2 \le k \le N \end{cases}$

is then used to perform a frequency warping using these magnitude values to produce MFCC and these features are then collected into a single feature vector, which is more appropriate for training a network. Special attention was paid to removing the first scalar within the vector, which represents the total signal power and is therefore too sensitive to the amplitude of the signal [Lilly00, GoldM00].

Short-Time Fourier Transform

A Short-Time Fourier Transform (STFT) was implemented using Matlab's FFT routine and a rectangular windowing algorithm. This approach allowed finer control over the resultant resolution of the STFT by allowing me to systematically change the number of samples in both time and frequency. The signal was windowed and then a FFT was calculated for each windowed segment [Cohen95]. This produces the following relation for the calculation of a STFT:

$$S_t(\omega) = \frac{1}{\sqrt{2\pi}} \int e^{-j\omega t} s(\tau) h(\tau - t) \quad d\tau \tag{16}$$

where ω is the frequency, τ is the signal length, *s(t)* is the signal and *h(t)* is the windowing function. This algorithm was implemented in Matlab with a variable window size parameter (allowing the resolution of the STFT to focus more closely on either time data or frequency data). Empirical testing showed that a window size of the sample frequency scaled by 100 produced the most accurate results when tested.

A principal component analysis (PCA) was used after feature extraction to reduce the dimensionality of the resulting signal. An adaptive algorithm was used to calculate the maximum number of principal components required for the training data used (based on the energy in each dimension and a variable threshold). A threshold value of 1% was found to produce the most accurate results. This process reduced the size of the signal significantly, from 128x67 (or 8643 features) to 18x67 (or 1206 features).

Continuous Wavelet Transform

For the continuous wavelet transform (CWT), I used the discretized *CWT* algorithm from Stanford University's Matlab Wavelab toolbox [Slaney98]:

$$CWT_x^{\psi}(\tau, s) = \Psi_x^{\psi}(\tau, s) = \frac{1}{\sqrt{|s|}} \sum_{t=1}^{N} x(t) \psi^{*}\left(\frac{t-\tau}{s}\right) \tag{17}$$

where τ represents translation, *s* represents scale and $\psi(t)$ is the mother wavelet, which was chosen to be the Morlet mother wavelet [Daube92], defined as:

$$\psi(t) = e^{jat} e^{\frac{-t^2}{2s}} \qquad (18)$$

where *a* is a modulation parameter and *s* again represents scale. This mother wavelet has been used for recognition tasks in the past and produced acceptable results [OrrPh01].

6.3.2.2 Classification

Two classification techniques will be tested in this comparison. The implementation of each of these techniques will be discussed in this section.

Learning Vector Quantization

Learning vector quantization (LVQ) was implemented using the inbuilt LVQ routines in Matlab's neural network toolbox. The network was initialized with 50 competitive neurons and a learning rate of 0.05. These settings allowed the network to converge in ~ 50 iterations and were found to give the most accurate classification rate.

Dynamic Time Warping

I implemented Dynamic time warping (DTW) using the *dtw* function in the Auditory Toolbox developed by Malcolm Slaney [Slaney98]. DTW uses a dynamic programming approach (as opposed to a linear approach) to align the length of the test signal with the length of the reference signal. DTW minimizes a global error by using a sequential optimization strategy where the current estimate of the global optimization function is updated for each possible step. Enough information is retained on the set of plausible hypotheses to allow the set of choices for the minimal global error to be reconstructed at the end. This means that a signal warped using DTW more closely resembles the original signal than a signal warped using a linear time warp [GoldM00].

To use DTW, feature extraction was first applied to each signal and then the test signal was warped against each of the reference signals and the error between these two signals was recorded. The smallest error was taken to represent the closest class of sound.

6.4 Automatically Generated Tree

In addition to testing using a taxonomy based on physical states, other advanced techniques that would reduce the pattern matching search space were also investigated. The C4.5 technique developed by Ross Quinlan [Quinlan93] was selected for this comparison. The C4.5 technique was selected as a counterpoint to the physical tree. The C4.5 technique analyses the data and develops a decision tree from the data automatically. In this way, instead of having pre-determined levels and nodes within the tree, a tree is generated that will classify well based on the data.

In addition, a tree generated using the C4.5 technique is totally deterministic. This means that the decisions made by the tree can be analysed after the tree has been built and the ability of the tree on particular sets of data can be determined. This is in contrast to most artificial intelligence techniques, where the mechanics of *how* the classification technique made the choice can remain a mystery.

An extract of an example tree generated by C4.5 is shown in Figure 39. The determinants in this tree are taken from Fast Fourier Transform's (FFT's) of typical environmental sounds. They are labeled from lowest to highest frequency. The example uses 256 features from the FFT for classification. Looking at this tree, we can see the deterministic nature of the C4.5 technique. The example tree could be used with any sound to determine its class. For this comparison, C4.5 will be tested with the same feature extraction techniques used for the physical taxonomy (MFCC & FFT). It is also clear from the example that only a few of the features are used to make a determination. This is another important feature of C4.5. The technique makes it easy to assess which features carry the highest weight in uniquely identifying a sound.

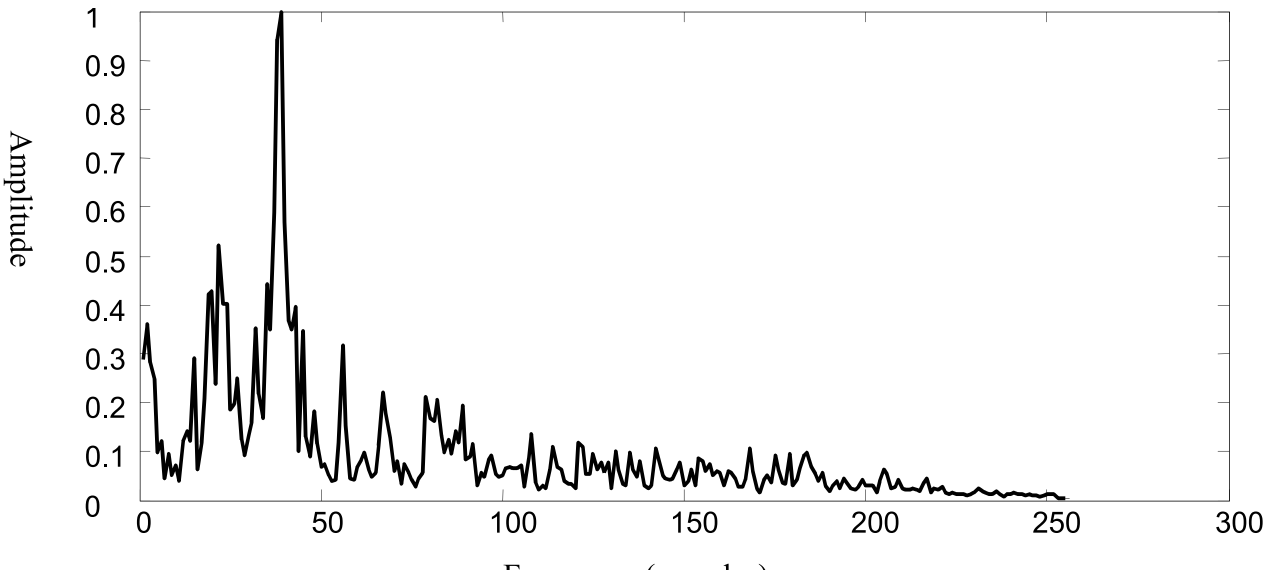


Figure 38 – Example of the features used for C4.5

An example may help to clarify how the C4.5 technique can be used. A sample sound is shown in Figure 38. This is the spectrum of coins dropping on a metal sheet. To aid in clarity, this graph is shown in sample and not Hertz, because this representation more closely parallels with the results from the C4.5 technique. We can use the tree generated by the C4.5 technique to classify this sound.

Working our way from the top of the tree to the bottom, we see that the first test uses feature 96. If the value of this feature is less than 0.47378 (normalized value) then the classification moves along the *Yes (Y)* branch of the tree, otherwise is moves along the *No (N)* branch of the tree. For our example, the value of feature 96 is obviously below 0.47378, so we move along the *Y* branch of the tree.

For the second test, the value of feature 36 is tested. In our example, the value of feature 36 is above the threshold shown (0.44536), so we follow the *N* branch of the tree. For the third test, the value of feature 105 is tested. In this case, the value of the feature in our example is below the threshold (0.2119), so the *Y* branch is followed.

Two tests are then performed on feature 1. The first checks whether the value of feature 1 is below 0.44353. In our example it is, so we follow the *Y* branch. Finally, the last test checks the value of feature 1 more closely. Once again, the value is below the threshold, meaning the *Y* branch is followed. This leads us to leaf node on the tree, which reports that the classification of the sound is as class 34, which is *Coins Dropping on Metal.*

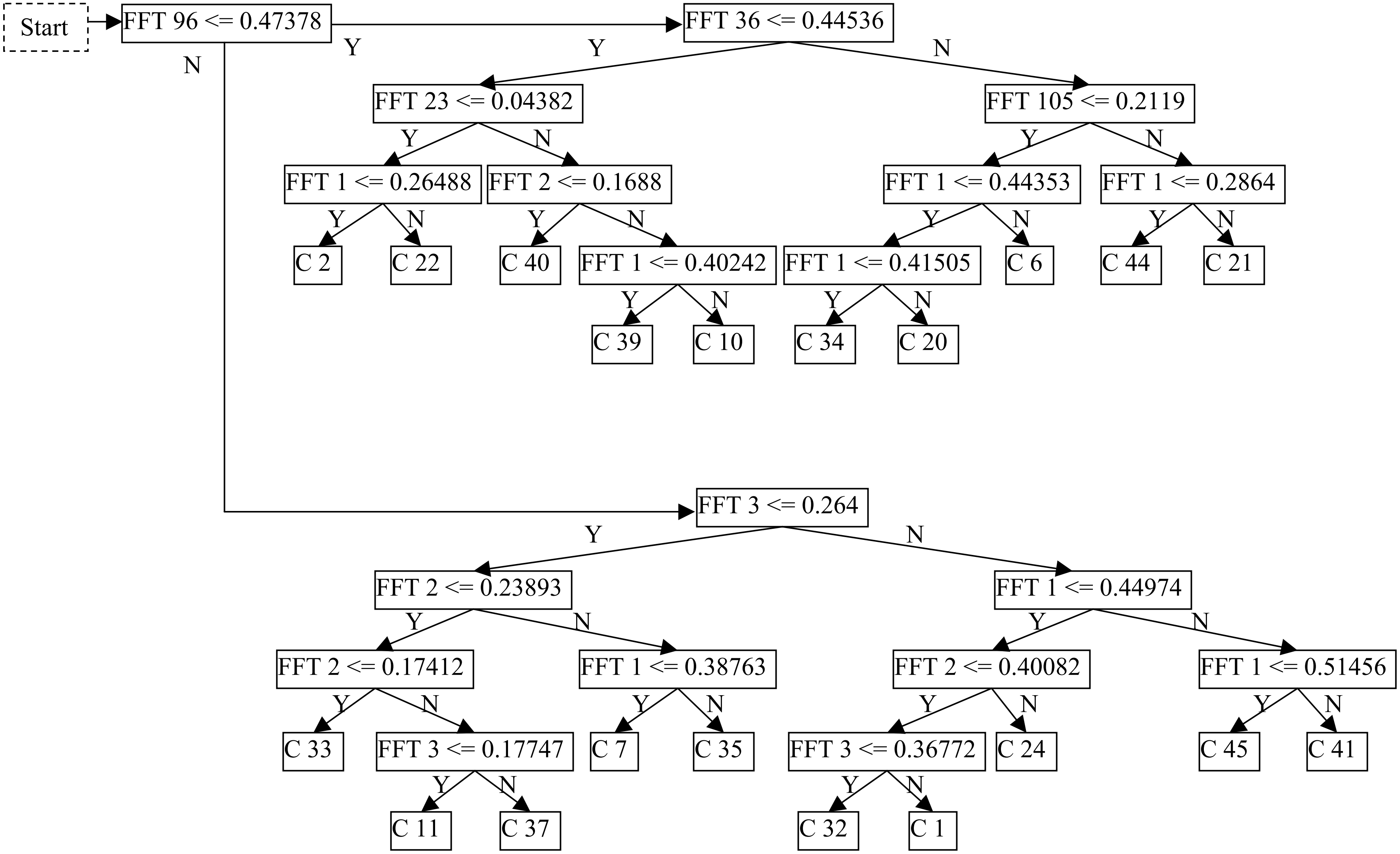


Figure 39 – An Example C4.5 Tree (Extract)

Chapter 7

Results & Discussion (Advanced Classification Techniques)

This chapter discusses the testing of the various different classifiers and feature extraction techniques within the taxonomy. Tests are performed with the previous set of sounds, and also with a larger corpus of sounds. Finally, some conclusions are also drawn on the nature of environmental sounds, based on the results of the C4.5 technique that are presented in this chapter.

7.1 Testing with the Jackknife Technique

This section shows the results from the experiments performed with the taxonomy. Each table (Table 19, Table 20) has two columns. The first column, *Level 2*, is the percentage correct classification into material type (Metal-Metal, Wood-Metal etc.). The second column, *Level 3*, is the percentage correct at the lowest level of the tree ("Coins Dropping, Wood Snapping" etc.). The two tables are also shown graphically in Figure 40 and Figure 41.

TABLE 19. LEARNING VECTOR QUANTIZATION (LVQ)

Method	Level 2	Level 3
MFCC	54%	16%
STFT	13%	4%
CWT	16%	16%

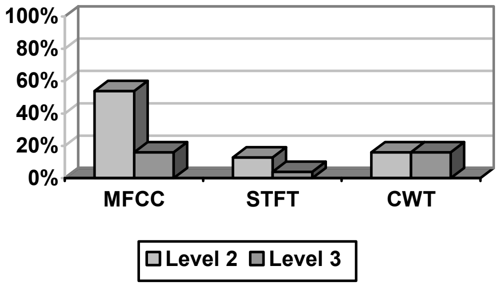


Figure 40 – Learning Vector Quantization

TABLE 20. DYNAMIC TIME WARPING (DTW)

Method	Level 2	Level 3
MFCC	70%	66%
STFT	16%	12%
CWT	20%	20%

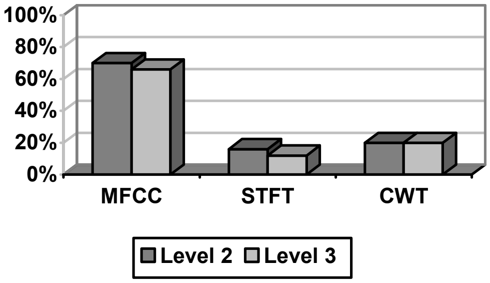


Figure 41 – Dynamic Time Warping

7.2 Testing With A Large Corpus Of Sounds

After performing this preliminary testing, testing was also performed with a larger set of sample sounds. The size of the set was increased in both variety (more classes) and volume (more sound samples). This large corpus of sounds was intended to test the hypothesis that the structured classification techniques would perform better with a larger set of different classes of sounds.

Each experiment was performed on the two most successful types of feature extraction technique (FFT & MFCC), using the DTW classification technique where required (in the physical taxonomy and on the flat classification). Two test sets were used (Table 21):

TABLE 21. TRAINING/TEST SETS USED FOR LARGER SET TESTS

Known	Training Set: 30 classes of sound Test Set: 15 classes of sound (all classes trained)
Unknown	Training Set: 20 classes of sound Test Set: 10 classes of sound (trained), 10 classes of sound (untrained)

Three experiments were conducted using the larger test sets. These were:

- **Experiment 1 -**Tree built with a physics model
- **Experiment 2 -** Automatic Tree using C4.5
- **Experiment 3 -** No Taxonomy, Flat Classification

Four runs were performed for each experiment. The results of these experiments are presented in Table 22 to Table 39 and the averages for each feature extraction technique (FFT & MFCC) are graphed in Figure 42 through Figure 47.

TABLE 22. FFT EXPERIMENT 1 (LARGER CORPUS)

Method	Known	Unknown
Run 1	32%	21%
Run 2	31%	22%
Run 3	43%	22%
Run 4	20%	20%
Average	32%	21%

TABLE 23. FFT EXPERIMENT 2 (LARGER CORPUS)

Method	Known	Unknown
Run 1	34%	17%
Run 2	23%	24%
Run 3	20%	12%
Run 4	30%	11%
Average	27%	16%

TABLE 24. FFT EXPERIMENT 3 (LARGER CORPUS)

Method	Known	Unknown
Run 1	80%	40%
Run 2	76%	40%
Run 3	73%	43%
Run 4	70%	33%
Average	75%	39%

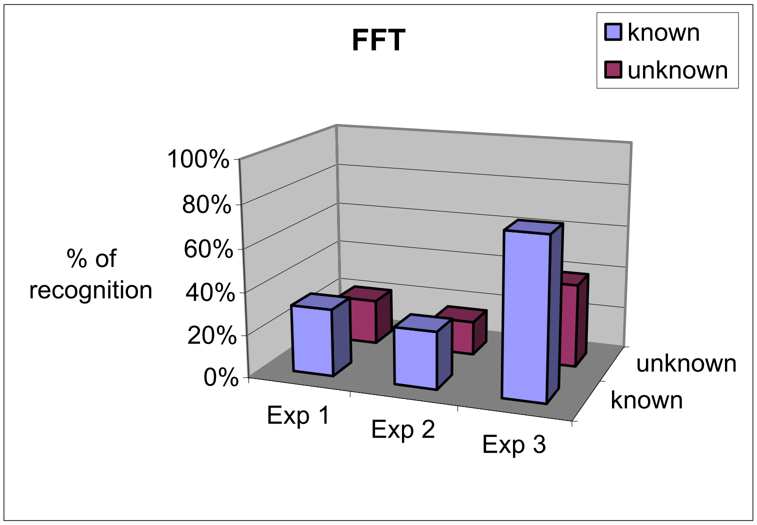


Figure 42 – Average Recognition Rates for FFT using a Larger Data Set

TABLE 25. MFCC EXPERIMENT 1 (LARGER CORPUS)

Method	Known	Unknown
Run 1	21%	17%
Run 2	29%	22%
Run 3	30%	12%
Run 4	20%	18%
Average	25%	17%

TABLE 26. MFCC EXPERIMENT 2 (LARGER CORPUS)

Method	Known	Unknown
Run 1	27%	5%
Run 2	7%	5%
Run 3	8%	10%
Run 4	13%	9%
Average	14%	7%

TABLE 27. MFCC EXPERIMENT 3 (LARGER CORPUS)

Method	Known	Unknown
Run 1	54%	33%
Run 2	59%	33%
Run 3	50%	29%
Run 4	58%	29%
Average	55%	31%

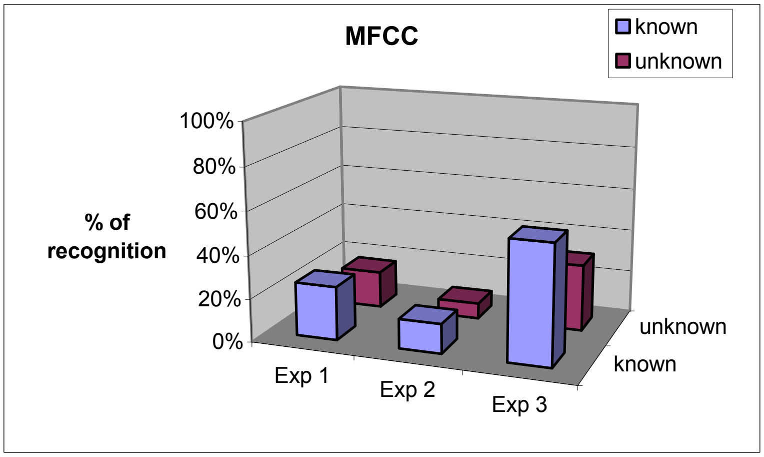


Figure 43 – Average Recognition Rates for MFCC using a Larger Data Set

TABLE 28. HCC EXPERIMENT 1 (LARGER CORPUS)

Method	Known	Unknown
Run 1	17%	5%
Run 2	15%	12%
Run 3	10%	10%
Run 4	3%	4%
Average	12%	8%

TABLE 29. HCC EXPERIMENT 2 (LARGER CORPUS)

Method	Known	Unknown
Run 1	7%	5%
Run 2	4%	4%
Run 3	10%	9%
Run 4	8%	2%
Average	7%	5%

TABLE 30. HCC EXPERIMENT 3 (LARGER CORPUS)

Method	Known	Unknown
Run 1	10%	3%
Run 2	15%	11%
Run 3	5%	7%
Run 4	10%	11%
Average	10%	8%

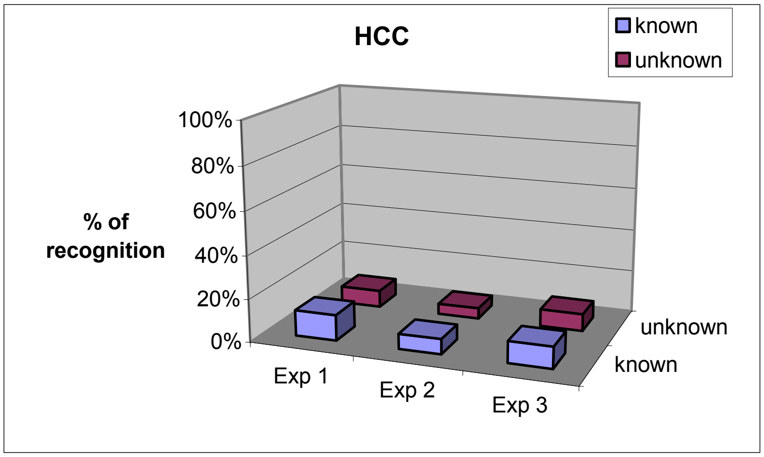


Figure 44 – Average Recognition Rates for HCC using a Larger Data Set

TABLE 31. CWT EXPERIMENT 1 (LARGER CORPUS)

Method	Known	Unknown
Run 1	17%	7%
Run 2	14%	11%
Run 3	25%	16%
Run 4	10%	9%
Average	17%	11%

TABLE 32. CWT EXPERIMENT 2 (LARGER CORPUS)

Method	Known	Unknown
Run 1	8%	7%
Run 2	12%	5%
Run 3	10%	3%
Run 4	10%	7%
Average	10%	6%

TABLE 33. CWT EXPERIMENT 3 (LARGER CORPUS)

Method	Known	Unknown
Run 1	26%	10%
Run 2	24%	22%
Run 3	33%	19%
Run 4	28%	13%
Average	28%	16%

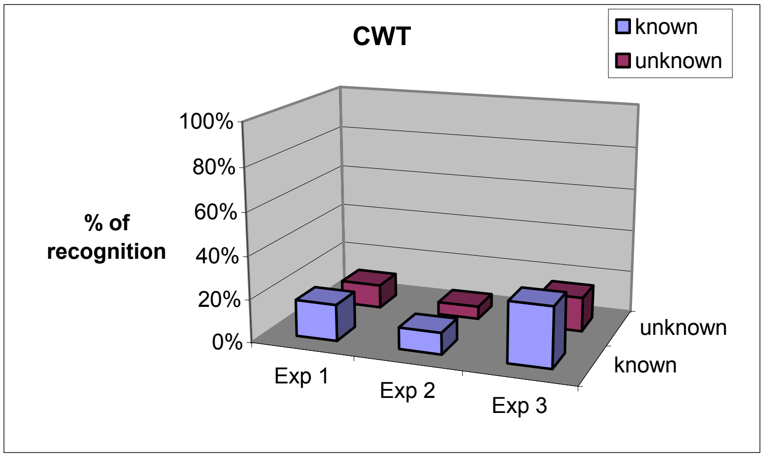


Figure 45 – Average Recognition Rates for CWT using a Larger Data Set

TABLE 34. FWT EXPERIMENT 1 (LARGER CORPUS)

Method	Known	Unknown
Run 1	14%	5%
Run 2	10%	7%
Run 3	15%	12%
Run 4	5%	3%
Average	11%	7%

TABLE 35. FWT EXPERIMENT 2 (LARGER CORPUS)

Method	Known	Unknown
Run 1	10%	5%
Run 2	5%	6%
Run 3	10%	10%
Run 4	10%	6%
Average	9%	7%

TABLE 36. FWT EXPERIMENT 3 (LARGER CORPUS)

Method	Known	Unknown
Run 1	7%	3%
Run 2	7%	5%
Run 3	10%	7%
Run 4	5%	2%
Average	7%	4%

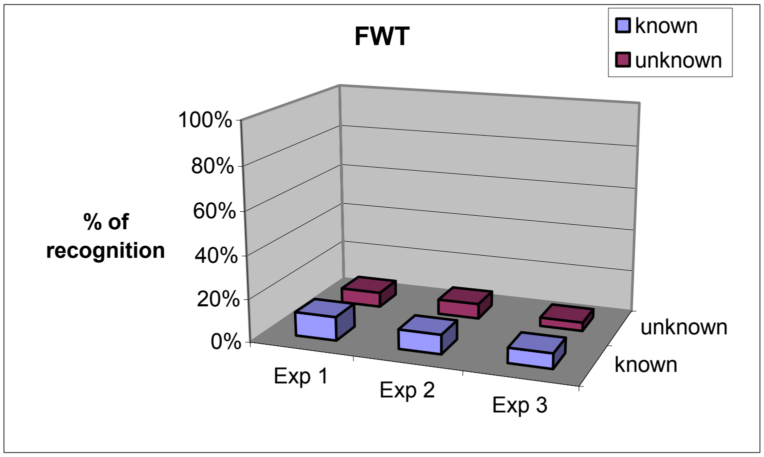


Figure 46 – Average Recognition Rates for FWT using a Larger Data Set

TABLE 37. STFT EXPERIMENT 1 (LARGER CORPUS)

Method	Known	Unknown
Run 1	17%	7%
Run 2	16%	7%
Run 3	18%	13%
Run 4	5%	4%
Average	14%	8%

TABLE 38. STFT EXPERIMENT 2 (LARGER CORPUS)

Method	Known	Unknown
Run 1	7%	5%
Run 2	7%	7%
Run 3	13%	7%
Run 4	8%	2%
Average	9%	5%

TABLE 39. STFT EXPERIMENT 3 (LARGER CORPUS)

Method	Known	Unknown
Run 1	10%	2%
Run 2	9%	5%
Run 3	10%	7%
Run 4	5%	2%
Average	9%	4%

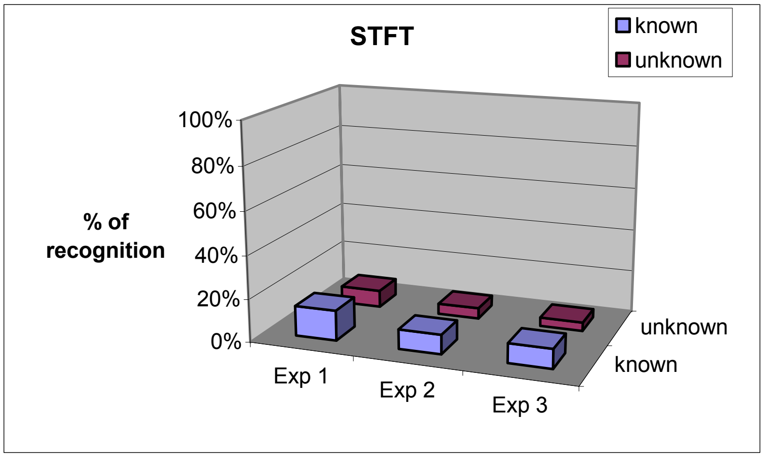


Figure 47 – Average Recognition Rates for STFT using a Larger Data Set

7.3 Discussion

These results are surprising, especially when compared to the results obtained previously using a non-taxonomic approach (see Chapter 5).

Previous results showed a classification of 70% with DTW, when it was used with either CWT or MFCC. LVQ showed a classification rate of 50% with frequency extraction and a classification rate of 54% with CWT.

Comparing the DTW results, we see some correlation between MFCC/DTW in the non-taxonomic approach and MFCC/DTW in the taxonomic approach. This shows the applicability of the taxonomic approach to classification of environmental sounds.

In comparison, the results presented for LVQ show a lesser classification rate for the final classification. Only on the second level are results comparable to those from the non-taxonomic approach. This is surprising, considering that the pattern search space has been significantly reduced. However, by analyzing the percentages for the higher levels and the results from the tests with a larger corpus of sounds, it quickly becomes apparent that the technique is getting “lost” in the lower levels of the taxonomy (this will be explained shortly).

The results from the tests with the larger corpus of sounds come as quite a surprise. Despite the physical taxonomy previously showing that it has the ability to perform as well as the non-taxonomy, this result does not scale well when the amount of sounds used in the comparison are increased. From the results it can be seen that while the non-taxonomy implementation maintains results similar to those obtained with the smaller corpus of sounds, the physical taxonomy obtains lower results than were obtained with the smaller corpus of sounds.

Also of surprise are the results from Experiment 2, which was the C4.5 technique. In these experiments, the C4.5 technique also performs below expectations, with results on par with the physical taxonomy.

Of the three techniques, the non-taxonomy approach shows the most promise for the classification of non-speech environmental sounds, despite the ability of the advanced techniques to split the pattern search space. I believe this is because the structured classification techniques are getting “lost” in the lower levels of the taxonomy.

As the taxonomy is being used for testing, classifications are occurring sequentially down the levels. On level one, classification is always correct, due to the deterministic approach taken. However, on level 2, the taxonomy can only be guaranteed to make a correct classification a certain percentage of the time (for instance, with MFCC/LVQ, 54% of the time). If it makes this correct classification, it then moves onto the appropriate third level node and performs pattern recognition on the third level. However, even if it does not make the correct classification (and at this stage the taxonomy has no idea if the classification it has just made is correct), it will still move to the third level and perform pattern recognition. Considering that it has incorrectly selected the third level node to move to, this will mean no chance of producing a correct classification.

Due to this, an incorrect classification on the second level will increase the chances greatly of an incorrect classification on the third level. This means that the percentage correct for the taxonomy on the third level is reduced proportionally to the amount of incorrect classifications on the second level. This also applies to the C4.5 technique.

The percentages of correct classification on the second level of the taxonomy support this and are shown for the MFCC and FFT techniques in Table 40. It can be seen that, as with the smaller set of sounds, the average result from the second level of the taxonomy is similar to the result with these techniques on the non-taxonomic approach. It is only on the third level that the result drops to that shown in the tables above.

TABLE 40. MFCC/FFT LEVEL 2 RESULTS (PHYSICAL TAXONOMY)

Method	MFCC	FFT
Run 1	56%	70%
Run 2	57%	68%
Run 3	57%	65%
Run 4	72%	70%
Average	61%	69%

In order to combat this problem, Martin proposes the use of beam searching techniques within the hierarchy [Martin99]. However, these beam searching techniques require the use of a confidence score to show how confident the node is that it has made a correct classification. If confidence on the third level is low, the taxonomy can move back to the second level and select the next best choice of third level node.

As of this time, techniques such as LVQ and DTW cannot produce this confidence score. Future work could examine the use of embedded hypothesis testing, confidence scores and beam searching as a way of improving the performance of the taxonomy.

7.4 Reverse Engineering the Search Environment

In addition to analysing the percentage recognition for these results, we can also use the results of the C4.5 technique to begin to make some determinations of the features that uniquely characterise each sound. As mentioned in the previous chapter, an advantage of the C4.5 technique is that it provides full information about how it makes its classifications (in a tree format). We can examine this tree to make determinations on which features are important for different sounds and classes of sounds.

By analysing the tree's produced for the FFT classification (which was the most successful C4.5 classification method), we can see some interesting observations emerging.

Firstly, on the lower level of the taxonomy generated, the features further up in the frequency range are used much more frequently to perform a classification. FFT features 1 through 4 are frequently used to make a final classification on the lower levels of the resultant taxonomy. Higher features (for examples, feature 103 and feature 209) are used on the first levels of the taxonomy, but the lower levels typically use features on the lower end of the frequency spectrum. This is not surprising, since *almost all* sounds contain lower frequencies, where as only *some* sounds contain higher frequencies. Therefore, the higher frequencies can be used to make the initial distinctions and then the lower frequencies can be used to differentiate between different sounds.

For example, for the first run of the FFT feature extraction technique with C4.5, the values of features 7, 45, 13, 17 and 156 are used to make a decision that a sound involves water. The value of feature 1 is then used to determine if this is a *bubbles in water* sound or a *pouring water* sound. Similarly, in the third run of the FFT feature extraction technique with C4.5, the values of features 6, 1 and 2 are used to make a decision on whether the sound involves metal (in this case, coins). The value of feature 4 is then used to determine whether the sound is *coins dropping on metal* or *coins dropping on wood*. To help illustrate this, the spectra of these four sounds are presented in Figure 48 to Figure 51. These spectra are presented with the horizontal axis measured in samples (and not the usual Hertz) to allow a direct comparison between the features in the C4.5 technique and the respective values on the frequency spectra.

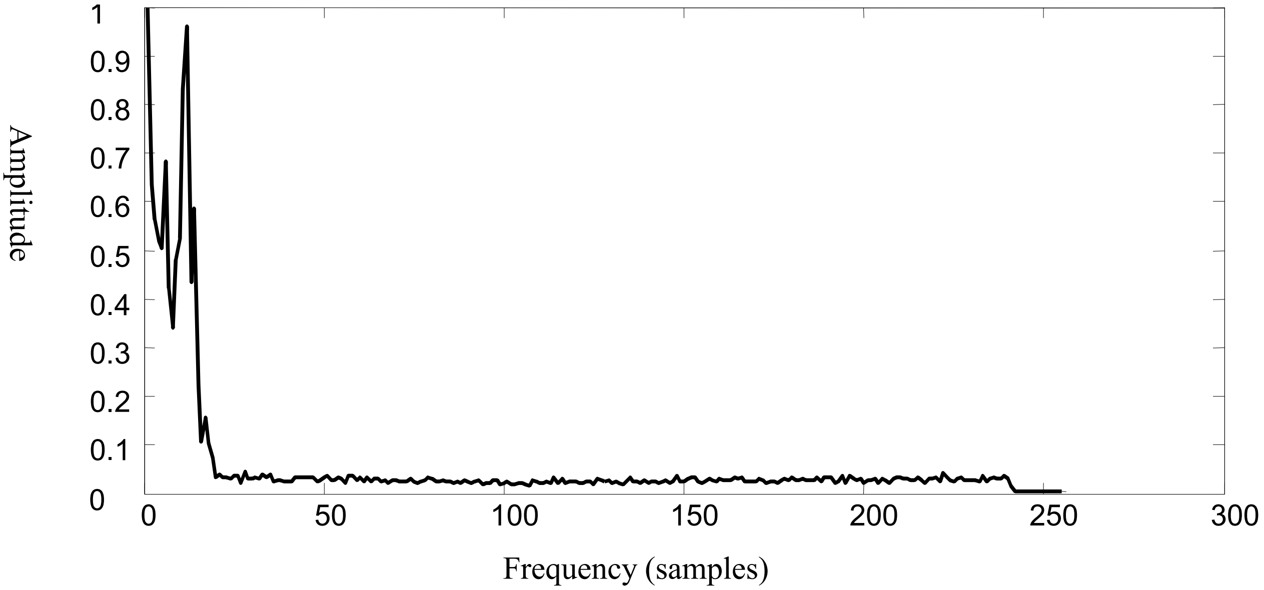


Figure 48 – Frequency Spectrum of Bubbles

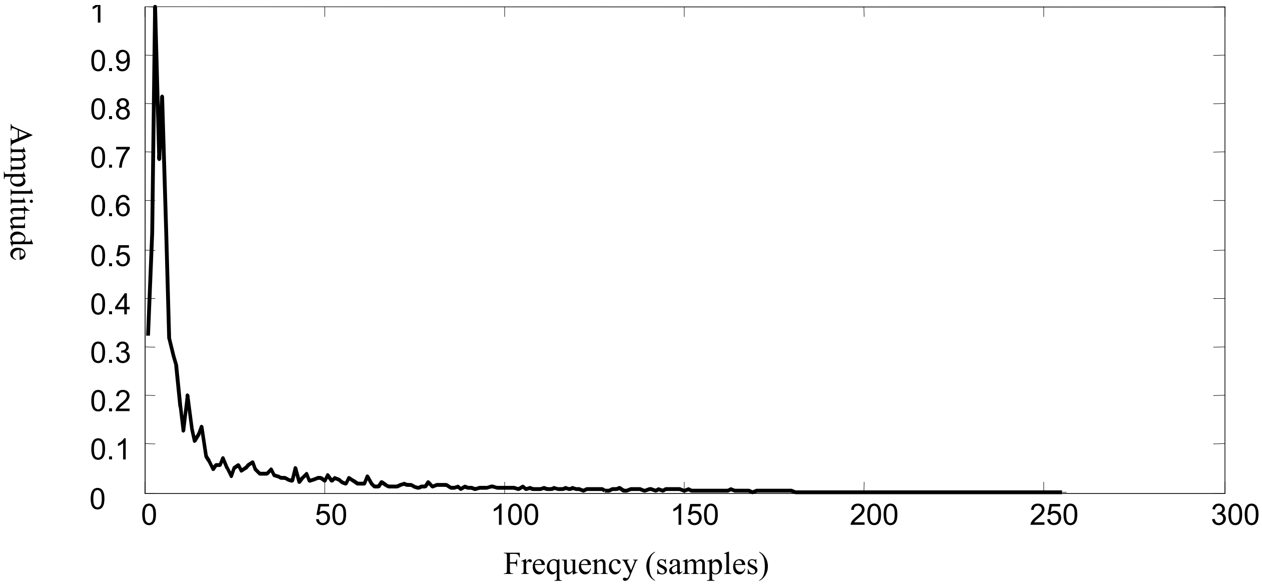


Figure 49 – Frequency Spectrum of Pouring Water

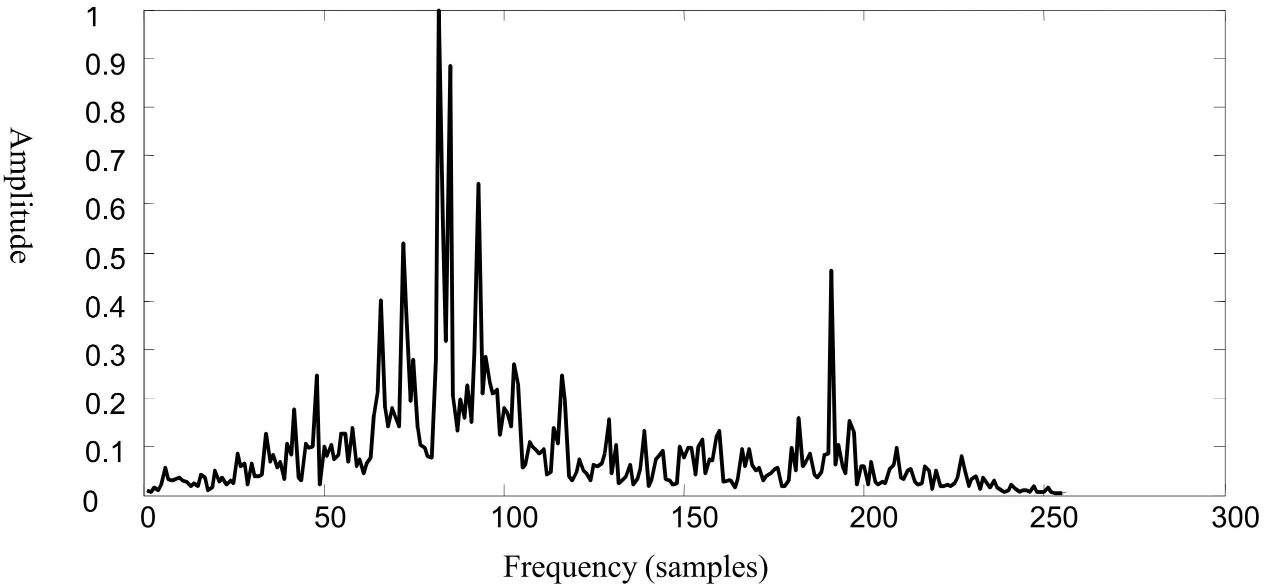


Figure 50 – Frequency Spectrum of Coins on Metal Sheet

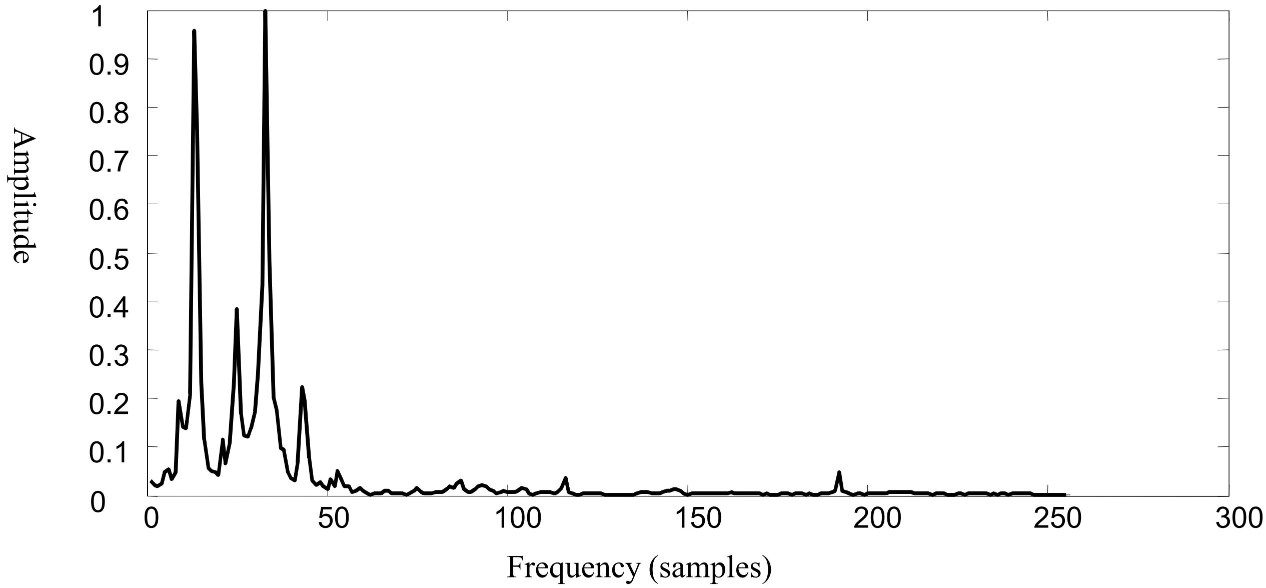


Figure 51 – Frequency Spectrum of Coins on Plywood Sheet

In addition, in some cases it can be seen that a particular feature is unique to a type of sound or set of sounds. For instance, in one of the runs, the value of feature 50 is used to determine the difference between a *singular clap of hands* and a *plurity of clapping*. Feature 50 is not used for any other purpose. It could be inferred that feature 50 is unique in determine the type of clapping sound that is being heard. Similarly, feature 5 is used to make a determination between the type of bell being heard (either a *small bell* or a *large bell*). This feature is also used in no other context. Figure 52 to Figure 55 show the spectrum of these sounds.

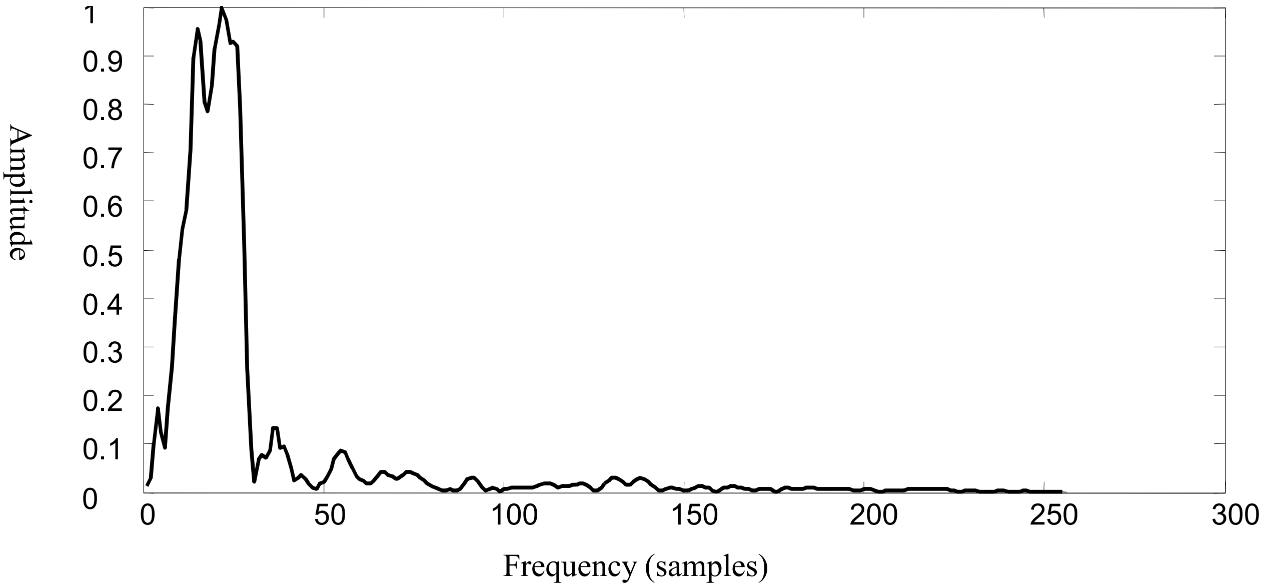


Figure 52 – Spectrum of a Single Clap of the Hands

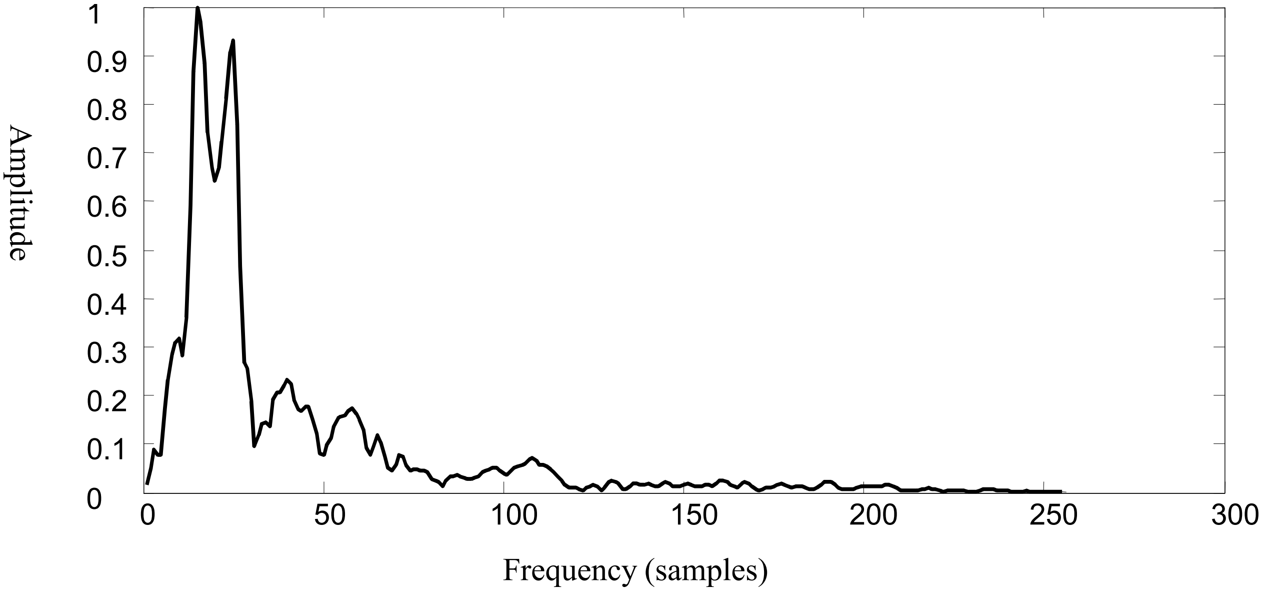


Figure 53 – Spectrum of a Plurity of Clapping

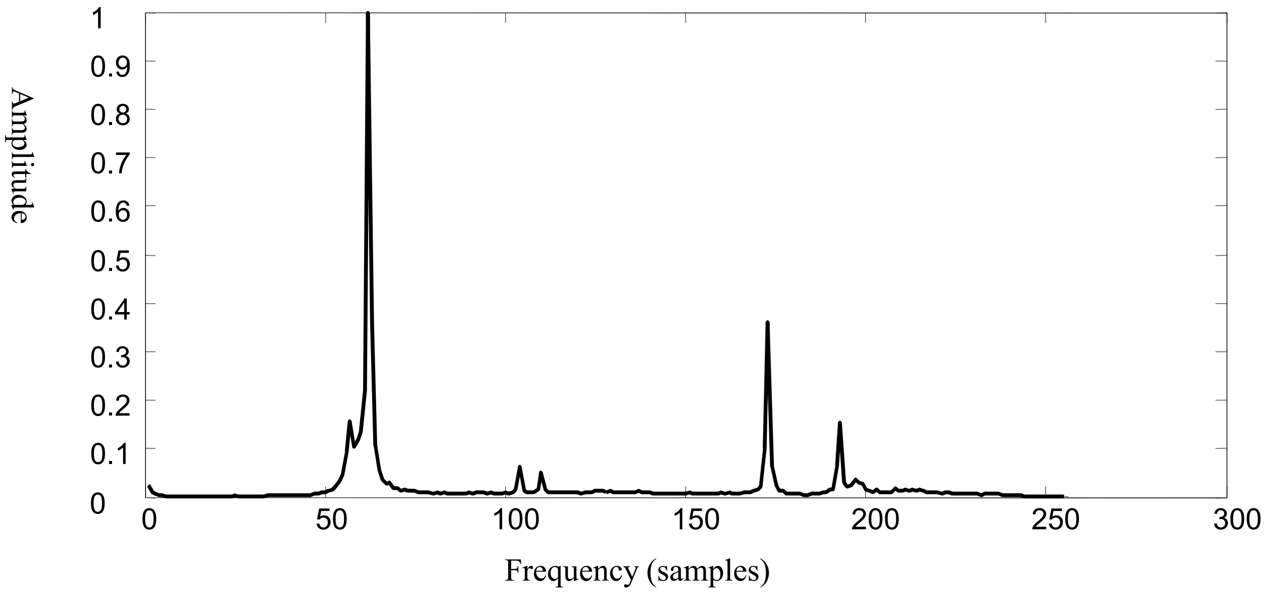


Figure 54 – Spectrum of a Large Bell Ringing

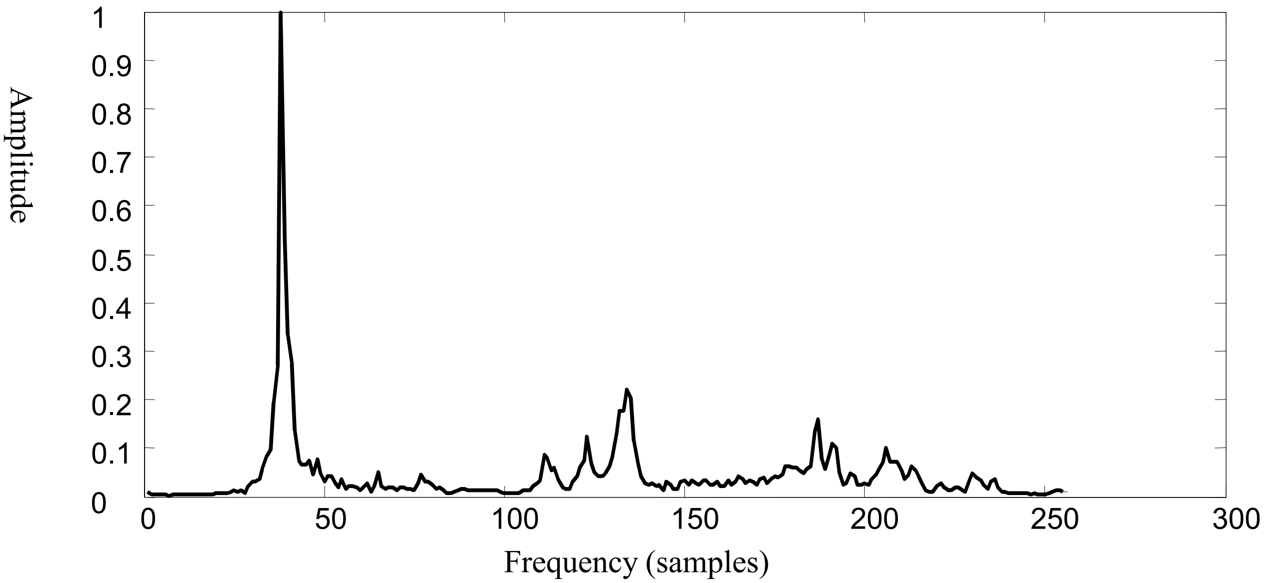


Figure 55 – Spectrum of a Small Bell Ringing

Finally, in some cases, a particular feature can be used to determine the type of sound being heard (in accordance with the physical taxonomy previously discussed). In more than one of the runs, the value of feature 209 is used to determine whether or not the sound being heard is a metal-metal type sound. This feature is used on the higher level of the taxonomy. On the lower levels, the values of features 1, 2, 5 and 13 are used to determine particular sounds, but all of which could be considered to be metal-metal sounds. Examples include *Soda Can / Metal Stick*, *Padlock Locking* and *Door Locking* (see Figure 56 to Figure 58). Feature 209 is not used anywhere else in the taxonomy. From this, it could be inferred that the value of feature 209 in a FFT (in addition with selected other features) can be used to uniquely work whether a sound involves a metal object in some way.

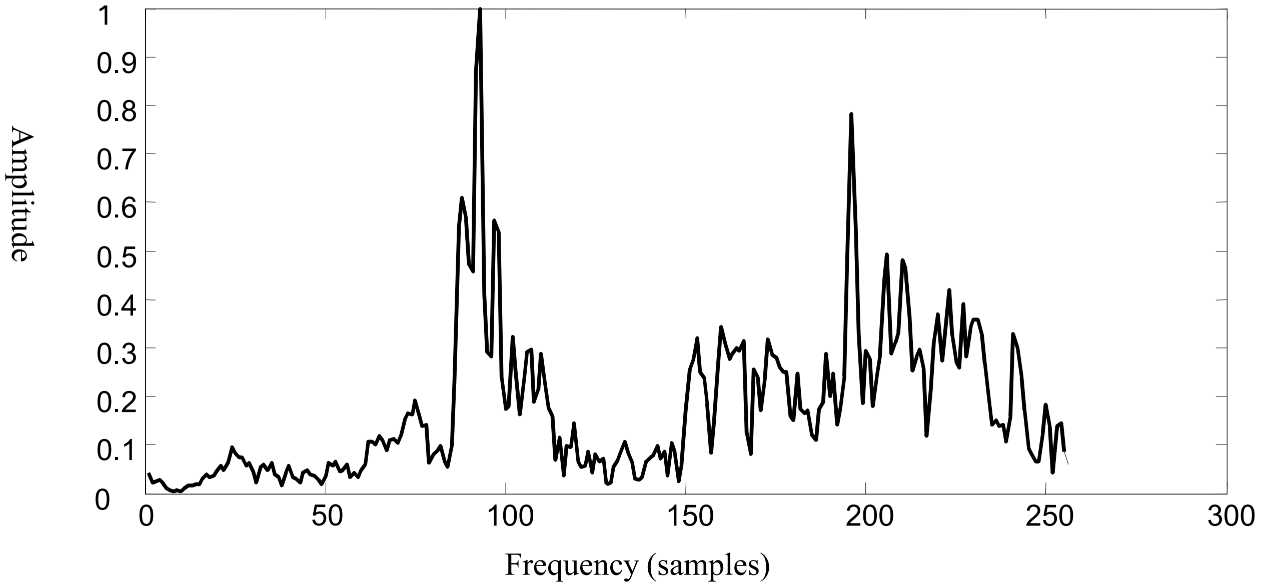


Figure 56 – Spectrum of Soda Can being hit with a Metal Stick

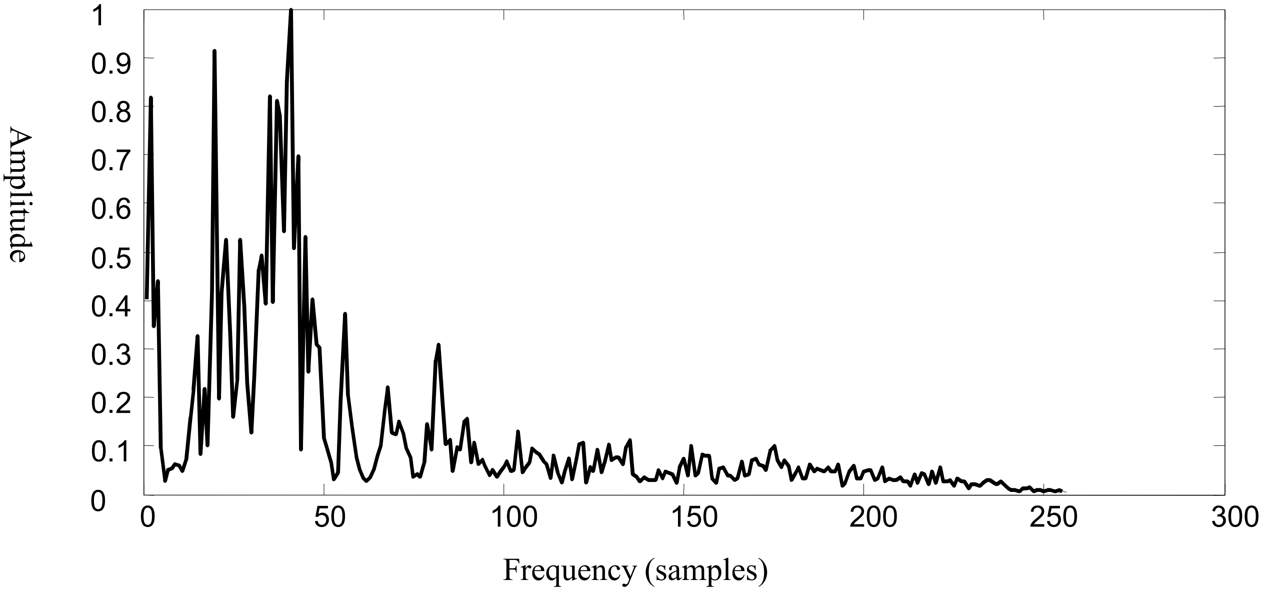


Figure 57 – Spectrum of Padlock Locking

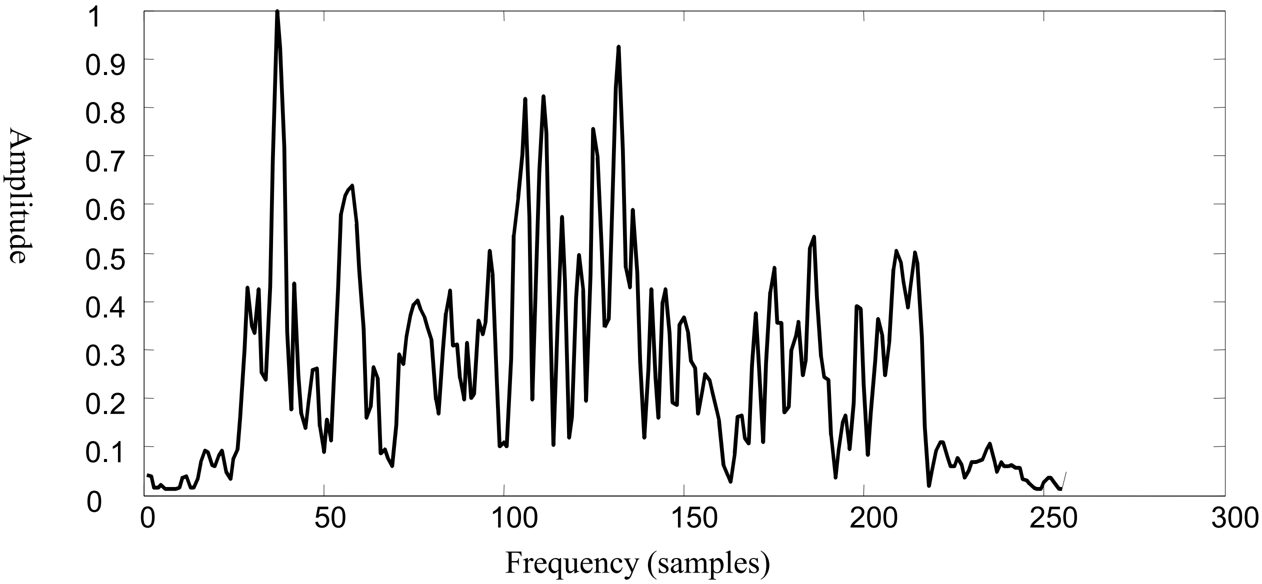


Figure 58 – Spectrum of Door Locking

Overall, although the results from this study do not present unique frequencies for *all* types of sounds (and indeed, no study could promise this), they do begin to give some ideas on the harmonic frequencies that could be used to classify different sounds and types of sounds. Combined with a novel environmental sound alphabet, these results allow us to begin to infer the general nature of environmental sounds. From results such as these, better classification systems can be built that take this advanced information into account, allowing more efficient and accurate classification of this extremely large pattern matching search space.

7.5 Lessons Learned

The following hypothesis was presented at the beginning of this work:

Hypothesis

If I can find a systematic way to identify environmental sounds, I could increase the efficiency of environmental sound identification for the purpose of security surveillance. A system can be developed that will recognise a large corpus of environmental sounds. This system will use a structured classification technique (sound taxonomy) to improve classification accuracy and speed.

The work in this thesis has made significant progress towards satisfying this hypothesis and the applicability of this work cannot be denied. Neither the work of Martin (which was in musical instrument recognition) or the work of researchers such as Reyes-Gomez & Ellis (in speech recognition) was able to perform acceptable second level classification for a set of sounds as large as has been demonstrated with this work. If further refinements can be done to improve the classification rate on the lower levels of the taxonomy, the high results on the second level of the taxonomy should allow a system that can classify with a great degree of accuracy.

In addition, and perhaps more importantly, this work begins to infer some information from the results on the general nature of environmental sounds. From the results of the C4.5 technique on such a large corpus of sounds, we can begin to determine the features that allow us to uniquely identify environmental sounds. Combined with a novel environmental sound alphabet, we can begin to identify the harmonic frequencies for different types of environmental sounds. This means that these inferences can then be used to further refine environmental sound recognition techniques, allowing greater accuracy and efficiency and a greater understanding of the nature of environmental sounds.

Despite this, these experiments confirm what has been shown in the initial literature review. Researchers such as Martin [Martin99] and Reyes-Gomez & Ellis [ReyesEl03] demonstrated the ability of techniques to classify sounds into broad groups with a reasonable accuracy. Both researchers also showed the difficulty in then classifying on a more detailed level once this initial classification had been made. The results from this survey correlate with these results from other researchers in the field.

Within this work, classification rates on the second level are consistently as good as those obtained using non-taxonomic approaches. It is only on the third level that classification rates fall below those using non-taxonomic techniques. The results from this work strengthen those already presented by researchers working in the similar but distinct fields of speech and music ([Martin99], [ReyesEl03]).

This work also presents many unique opportunities for future work. A novel taxonomy has been presented that has demonstrated its ability to classify sounds into broad groups. This taxonomy can be improved with alternative classification techniques to allow a greater classification on the lower levels. Techniques such as beam searching and fuzzy logic can be used to introduce a level of “intelligence” into the taxonomy, allowing for more structured classification on the third level and a subsequent increase in the classification for the technique.

Finally, it is surprising to note the remaining lack of competition in the field of environmental sound recognition. Despite the applicability of this research to fields such as Source Separation and Computational Auditory Scene Analysis (CASA), relatively little work is being performed. This makes the results of this work even more important, as they provide some of the only groundwork that can then be used by researchers in the field of CASA to greater understand environmental sounds and how they might be separated from other sounds in our environment.

Chapter 8

Conclusions

Considering that the auditory component of the world is made up of not only speech, but also many other kinds of sounds, it is important that a computer can recognise and classify not only speech, but also the other common sounds within an environment. As mentioned previously, areas such as hearing aid technology and security systems would benefit from research into a system that could identify non-speech sounds. In addition, a system such as this could also benefit from being able to identify the source of a sound (the direction it came from).

Based on this problem, the aim of this PhD thesis was to develop an efficient system that can recognise environmental sounds and their source, using a structured classification technique. The domain of this research is security, specifically with the intent of being embedded in an autonomous robot for surveillance.

This thesis has achieved this aim. It has presented a choice of techniques that can be used to classify environmental sounds with an accuracy of 70% in a robotic security system. It has also presented the design and testing of a more structured classification system that could be used to increase this classification rate. Finally, it has also shown some general features of environmental sounds that could be used in the future to develop more "intelligent" recognition systems.

8.1 Summary of Work

This work began with a comprehensive review of existing techniques for classification of environmental sounds. Due to a lack of literature on non-speech sound recognition techniques, speech and musical instrument recognition techniques were investigated for classification. A comprehensive analysis was performed on all speech recognition techniques in common use and those suitable for non-speech sound recognition were identified. Source localisation techniques were also reviewed and those suitable for non-speech direction detection were also selected. Finally, the few existing research papers on non-speech environmental sound recognition were studied and integrated into this work.

After a comprehensive literature review, it was decided that the area of non-speech sound recognition and source localisation would benefit from research into the following areas:

- Comprehensively compare techniques from Speech & Music to determine their applicability to Environmental Sounds. Implement these techniques and determine the best techniques for Environmental Sounds.
- Investigate advanced techniques for Environmental Sound Recognition that can reduce the pattern matching search space. Develop a new technique using ideas from physics.
- Compare and contrast the results from these two types techniques and draw conclusions

Chapters 3 and 4 cover the comprehensive comparison of existing techniques from speech and music as applied to environmental sounds. Existing techniques from speech and music were analysed and feature extraction/classification techniques that could be applied to non-speech environmental sounds were selected for testing. These techniques were:

Feature Extraction

- Frequency Extraction (FE)
- Mel-Frequency Cepstral Coefficients (MFCC)
- Homomorphic Cepstral Coefficients (HCC)
- Short-Time Fourier Transform (STFT)
- Continous Wavelet Transform (CWT)
- Discrete (Fast) Wavelet Transform (FWT)

Classification

- Learning Vector Quantization (LVQ)
- Artificial Neural Networks (ANN)
- Long-Term Statistics (LTS)
- Dynamic Time Warping (DTW)
- Gaussian Mixture Models (GMM)

In Chapter 5, these comparisons were performed and the quantitative results were extensively discussed. Without structured classification techniques, a combination of a MFCC based feature extraction technique and a DTW based classification technique is the most appropriate approach to developing an environmental sound recognition system (achieving 70% recognition). In addition, an FT/LVQ based approach or an MFCC/LVQ approach also shows promise.

Based on these results, Chapter 6 then discussed the use of structured classification techniques for classification of non-speech environmental sounds. A technique for classification using the physical properties of sounds was developed and other techniques were investigated for classification using a more advanced approach. Feature extraction techniques were then selected from those that performed best in the previous experiments. This produced the following list of techniques:

Feature Extraction

- Mel-Frequency Cepstral Coefficients (MFCC)
- Short-Time Fourier Transform (STFT)
- Continous Wavelet Transform (CWT)

Classification

- Physical Taxonomy
- C4.5 Tree Building Technique

Testing was performed using these techniques. Results were presented in Chapter 7, with a result of 66% using MFCC/DTW on a 3-level taxonomy based on the physical states of the originating objects. Slightly lower results were achieved using the structured C4.5 technique and the MFCC or FFT feature extraction techniques. Lower results from this technique are accepted in the research community when comparing to neural network and deterministic-based techniques.

Finally, Chapter 7 also discussed testing performed on a large corpus of sounds that contained greater variety and volume of environmental sounds. Testing on a larger corpus of sounds showed that, even with a smaller pattern search space, structured techniques do not work as well as unstructured techniques. Examining results from the second level classification of the physical taxonomy showed that this is because confusion on the first or second level means that the taxonomy can no longer make a correct classification. Implementation of a different classification technique that allows backtracking through the tree may help this result to improve, but this would require a further comparative study.

In addition, the results from this work on structured classification techniques was examined and interpreted. This interpretation revealed the beginning of some understanding of those features that uniquely identify environmental sounds. The results from the C4.5 technique were especially helpful in presenting this information, allowing a greater understanding of the weighting of different features for environmental sound recognition and how these different features could be used for classification in the future.

Overall, the main contribution of this thesis is to dispel the misconception that all speech recognition techniques can simply be used for non-speech sounds. It achieves this by giving a comprehensive comparison of speech recognition techniques that work with environmental sounds and presenting a novel advanced taxonomy that can be used to improve recognition (after it has been refined).

8.2 Future Research Directions

There are many interesting directions that could be taken from this work. I foresee three immediate directions for research stemming from this PhD work. These are listed below:

- Future research could investigate the implementation of alternative classification techniques within the physical taxonomy. If we can find a classification technique that allows us to perform a hypothesis test and measure the confidence of the classification that has been made, we can use this confidence rating to implement beam searching techniques that allow the taxonomy to move up to the previous level of the tree and make an alternative classification, thereby improving the classification rate.

- Fuzzy logic techniques could also be investigated for the classifications made on each level. This should allow those sounds that approach the boundaries of a particular classification to be classified successfully for the lower levels of the tree.

- The techniques outlined could be implemented in a robotic device. This would produce unique challenges in the areas of software design. For distributed control, communication methods would have to be investigated. For autonomous control, efficient implementations of the algorithms would have to be produced to work on the limited hardware available to autonomous robotic devices.

References

[Acken99] J. Ackenhusen, *Real-Time Signal Processing: Design and Implementation of Signal Processing Systems*, 1999, Prentice Hall, Inc., New Jersey, USA.

[Arslan00] G. Arslan, F. A. Sakarya, "A unified Neural-Network-Based Speaker Localization Technique", *IEEE Transactions on Neural Networks, Vol. 11, No. 4, July 2000.* IEEE, New York, USA.

[Ballas87] J. Ballas; J. Howard, "Interpreting the Language of Environmental Sounds" *Environment & Behaviour*, Vol. 19, No. 1, pp. 91-114, 1987 Sage Publications, Inc.

[Brand00] M. Brandstein; D. Ward, "Cell-Based Beamforming (CE-BABE) for Speech Acquisition with Microphone Arrays", *IEEE Transactions on Speech and Audio Processing, Vol. 8, No. 6, November 2000.* IEEE, New York, USA.

[Brand97] M. Brandstein; J. Adcock; H. Silverman, "A Closed-Form Location Estimator for Use with Room Environment Microphone Arrays", *IEEE Transactions on Speech and Audio Processing, Vol. 5, No. 1, January 1997.* IEEE, New York, USA.

[Bregman90] A. Bregman, *Auditory Scene Analysis,* 1990, MIT Press, Massachusetts Institute of Technology, Massachusetts, U.S.A.

[Brooks98] R. Brooks; C. Breazeal; M. Marjanovic; B. Scassellati; M. Williamson "The Cog Project: Building a Humanoid Robot." C. Nehaniv, ed., *Computation for Metaphors, Analogy and Agents*, Vol. 1562 of Springer Lecture Notes in Artificial Intelligence, Springer-Verlag, 1998

[Brown99] J. Brown, "Computer identification of musical instruments using pattern recognition with cepstral coefficients as features", *J. Acoust. Soc. Am. 105*, 1999, pg. 1933-1941.

[Buhrke95] E. Buhrke; J. LoCicero, "Multi-Layer Perceptron Neural Networks with Applications to Speech Recognition", *Signal Processing Methods for Audio, Images and Telecommunication. (Signal Processing and its Applications),* 1995, Academic Press Limited: Harcourt Brace and Company, Publishers. Printed in Great Britain by the University Press, Cambridge.

[Carti00] J. Cartinhour, *Digital Signal Processing: An overview of Basic Principles*, 2000, Prentice Hall, Inc., New Jersey, USA.

[Castro93] M. J. Castro; J. C. Perez, “Comparison of Geometric, Connectionist and Structural Techniques on a Difficult Isolated Word Recognition Task.”, *Proceedings of European Conference on Speech Comm. and Tech., ESCA*, Vol. 3, pp 1599-1602, Berlin, Germany, 1993.

[Cheve98] A. de Cheveigné, "Time domain processing in the auditory system", *Proc. ICONIP (International Conference on Neural Information Processing)*, 1998, pp 1327 – 1332.

[Cohen92] L. Cohen, “What is a Multicomponent Signal?”, *Proceedings of ICASSP,* Vol. V, pp. 113—116, 1992.

[Cohen95] L. Cohen, *Time-Frequency Analysis*, 1995, Prentice-Hall, Inc., New Jersey, USA.

[Cooke01] M. Cooke, D. Ellis, “The auditory organization of speech and other sources in listeners and computational models”, *Speech Communication, Vol. 35, Issues 3-4, Oct. 2001, pp. 141-177.*

[Cornell01] Cornell Bioacoustics Research Program, *Canary 1.2.4 Software Program*, Cornell Lab of Ornithology, Cornell University, 2001.

[Cowl00] M. Cowling, *Sound Identification and Direction Detection for Surveillance Applications*, Honours Thesis, Faculty of Enginerring and Information Technology, Griffith University, Gold Coast, October 2000.

[CowlSit00] M. Cowling, R. Sitte, “Sound Identification and Direction Detection in Matlab for Surveillance Applications”, *Proceedings of Matlab Users Conference*, Melbourne, Australia, November, 2000.

[CowlSit01] M. Cowling, R. Sitte, “Sound Identification and Direction Detection for Surveillance Applications”, *Proceedings of ICICS 2001, Singapore,* October, 2001.

[CowlSit02] M. Cowling, R. Sitte, “Analysis of Speech Recognition Techniques for use in a Non-Speech Sound Recognition System”, *Proceedings of DSPCS’02,* Manly, NSW, Australia, January, 2002.

[CowlSit02b] M. Cowling, R. Sitte, “Recognition of Environmental Sounds using Speech Recognition Techniques”, Advanced Signal Processing for Communications Systems, 2002, Kluwer Academic Publishers.

[CowlSit02c] M. Cowling, R. Sitte, “Structured Classification of Environmental Sounds”, *Proceedings of WoSPA 2002,* Brisbane, December, 2002.

[CowlSit03] M. Cowling, R. Sitte, “Time-Frequency Environmental Sound Recognition for Autonomous Robot Surveillance”, *Proc. of AMiRE 2003*, Brisbane, February, 2003.

[CowlSit03b] M. Cowling, R. Sitte, “Comparison of Techniques for Environmental Sound Recognition”, *Pattern Recognition Letters*, Elsevier Science Inc.,Vol. 24, Issue 15, Nov. 2003, pp. 2895-2907.

[CowlSit03c] M. Cowling, R. Sitte, “Building an Environmental Sound Taxonomy for Autonomous Robot Surveillance”, *Proc. of DSPCS’03*, Gold Coast, QLD, Australia, December, 2003.

[Daube92] I. Daubechies, *Ten Lectures on Wavelets,* Regional Conference Series in Applied Mathematics, Society for Industrial and Applied Mathematics, Vermont, USA, 1992.

[Dono02] D. Donoho, M. Duncan et. al., “WaveLab Toolbox™”, Stanford University, version 0.802, 2002.

[Dorken92] E. Dorken; S. Nawb; S. Milios, “Knowledge-Based Signal Processing Applications”, *Knowledge-Based Signal Processing,* Prentice Hall Publications, 1992.

[Ellis96] D. Ellis, *Prediction-driven computational auditory scene analysis*. PhD Thesis, Massachusetts Institute of Technology, Massachusetts, U.S.A, 1996.

[Ellis98] D. Ellis, *Using knowledge to organise sound: The prediction-driven approach to computational auditory scene analysis, and its application to speech/nonspeech mixtures.* International Computer Science Insitute, Berkley, CA, U.S.A, 1998.

[EroKlap00] A. Eronen, A. Klapuri, “Musical Instrument Recognition using Cepstral Coefficients and Temporal Features”, *Proc. of ICASSP 2000*.

[Faich00] J. Faichney, “Content-Based Retrieval of Digital Video”, PhD Confirmation Thesis, 2000, Griffith University, Qld, Australia.

[Fine91] A. B. Fineberg; R. J. Mammone, “Detection and Classification of Multicomponent Signals”, *Conf. Rec. of Twenty-Fifth Asilomar Conference on Signals, Systems and Computers,* 1991.

[FraserFuj99] A. Fraser, I. Fujinaga, “Toward real-time recognition of musical instruments”, *Proc. of the International Computer Music Conference,* 1999, San Fransisco, USA.

[FujiMac00] I. Fujinaga, K. MacMillan, “Realtime Recognition of Musical Instruments”, *Proc. of the International Computer Music Conference,* 2000.

[Fujinaga98] I. Fujinaga, “Machine recognition of timbre using steady-state tone of acoustic musical instruments”, *Proc. of the International Computer Music Conference,* 1998, Michigan, USA.

[Fuku90] K. Fukunaga, “Feature Extraction and Linear Mapping for Classification”, *Introduction to Statistical Pattern Recognition.* Academic Press, Inc. 1990, California, USA.

[Georg00] P. G. Georgiou; C. Kyriakakis, “A multiple input single output model for rendering virtual sound sources in real time”, *IEEE International Conference on Multimedia and Expo*, Vol. 1, pp 253 – 256, 30 July-2 Aug. 2000.

[GillT00] D. Gill; L. Troyansky; I. Nelken, “Auditory localization using direction-dependent spectral information”, *Neurocomputing*, Vol. 32 – 33, 2000, pp 767 – 773.

[Goldh93] R. Goldhor, “Recognition of Environmental Sounds”, *ICASSP-93 Vol. 1, 1993, p149 – p152.* IEEE, New York, NY, U.S.A.

[GoldM00] B. Gold; N. Morgan, *Speech and Audio Signal Processing*, John Wiley & Sons, Inc, 2000, New York, NY.

[Gonz97] R.Gonzalez, “Hypermedia Data Modelling, Coding and Semiotics”, Proceedings of the IEEE, July 1997.

[Hart99] W. M. Hartmann, “How we localize sound”, *Physics Today,* November, 2000, pp 23 - 29.

[Haykin99] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 1999, Prentice Hall, Upper Saddle River, N.J.

[Hiya00] K. Hiyane; J. Iio. “Non-speech sound recognition with microphone array.” *Presented at the RWC Symposium*, 2000.

[Hiyane02] K. Hiyane et. al, “Non-speech sound dry source database”, RWCP RG, *http://tosa.mri.co.jp/sounddb/*

[HoytW94] J. Hoyt; H. Wechsler, “Detection of Human Speech in Structured Noise”, *Proceedings of ICASSP,* Vol. 2, pp 237 – 240, 1994.

[Huang95] J. Huang; N. Ohnishi; N. Sugie, “A Biometric System for Localization and Seperation of Multiple Sound Sources”, *IEEE Trans. On Instrumentation and Measurement,* Vol. 44, No. 3, 1995, pp 733 – 738.

[Huang99] J. Huang; T. Supaongprapa et. al., “A model-based sound localization system and its application to robot navigation”, *IEEE Trans. on Robotics and Autonomous Systems,* Vol. 27, No. 4, 1999, pp 199 – 209.

[Hubbard95] B. Hubbard, *The World According to Wavelets: the story of a mathematical technique in the making*, 1995, Wellesley, Mass, USA.

[Jelin97] F. Jelinek, *Statistical Methods for Speech Recognition*. The MIT Press, 1997 Cambridge, Mass, USA

[Jiang00] H. Jiang, K. Hirose, Q. Huo, “A Minimax Search Algorithm for Robust Continuous Speech Recognition”, *IEEE Trans. on Speech & Audio Processing*, Vol. 8, No. 6, Nov. 2000 pp 688 – 694

[Juang00] B. Juang; S. Furui, “Automatic Recognition and Understanding of Spoken Language – A First Step Toward Natural Human-Machine Communication”, *Proceedings of the IEEE,* Vol. 88, No. 8, August, 2000.

[KashMur99] K. Kashino, H. Murase, “A sound source identification system for ensemble music based on template adaptation and music stream extraction”, *Speech Communication 27*, 1999, Elsevier Science.

[Kefau99] A. Kefauver, “The Digital Encoding Process”, *Fundamentals of Digital Audio. (Volume 14: The computer music and digital audio series)*. A-R Editions, Inc. Madison, Wisconsin, U.S.A. 1999.

[KingC95] A. King; S. Carlile, “Neural Coding for Auditory Space”, *The Cognitive Neurosciences, 1995,* The Mit Press, Massachusetts Institute of Technology, Londom, England.

[Kohon90] T. Kohonen, “The “Neural” Phonetic Typewriter” *Readings in Speech Recognition*, Edited by A.Waibel, K.F. Lee, 1990, Morgan Kaufmann Publishers, Inc. San Mateo, California, USA

[Kohon97] T. Kohonen, *Self-Organizing Maps.* 1997, Springer-Verlag Berlin, Germany. Printed in the U.S.A.

[Konis95] M. Konishi, “Neural Mechanisms of Auditory Image Formation”, *The Cognitive Neurosciences,* 1995, The Mit Press, Massachusetts Institute of Technology, Londom, England.

[Lee90] K. F. Lee, “Context-Dependant Phonetic Hidden Markov Models for Speaker-Independent Continuous Speech Recognition”, *Readings in Speech Recognition,* Edited by A. Waibel, K. F. Lee. Morgan Kaufmann Publishers, Inc.1990, San Mateo, Cal., USA

[Lee96] C. H. Lee; F. K. Soong; K. Paliwal "An Overview of Automatic Speech Recognition", *Automatic Speech and Speaker Recognition: Advanced Topics*. Kluwer Academic Publishers, 1996, Norwell, MA.

[Lee96b] C. H. Lee; F. K. Soong; K. Paliwal "An Overview of Speaker Recognition Technology", *Automatic Speech and Speaker Recognition: Advanced Topics*. Kluwer Academic Publishers, 1996, Norwell, MA.

[Lilly00] B. Lilly, *Robust Speech Recognition in Adverse Environments*, PhD Thesis, Faculty of Engineering, Griffith University, Nathan Campus, May 2000.

[LiSet01] D. Li; I. K. Sethi et. al, “Classification of general audio data for content-based retrieval”, *Pattern Recognition Letters,* Vol. 22, 2001, pp. 533 – 544.

[Liu99] L. Liu, “Ground Vehicle Acoustic Signal Processing Based on biological Hearing Models”, Masters Thesis, 1999, University of Maryland, College Park.

[MarqMor99] J. Marques, P. J. Moreno, “A Study of Musical Instrument Classification using Gaussian Mixture Models and Support Vector Machines”, *Cambridge Research Laboratory Tech. Report*, June, 1999.

[Martin99] K. Martin, *Sound-Source Recognition: A Theory and Computational Model,* PhD Thesis, 1999, MIT Media Lab, Massachusetts Institute of Technology, USA.

[MartYo98] K. Martin; K. Youngmoo, *Musical instrument identification: A pattern-recognition approach,* 1998, MIT Media Lab Machine Listening Group, Cambridge, MA, U.S.A.

[Melih98] K. Melih, *Content Based Audio Retrieval*, PhD Confirmation Thesis, 1998, Griffith University, Qld, Australia.

[Murthy99] H. Murthy; F. Beaufays et al. “Robust Text-Independent Speaker Identification over Telephone Channels”, *IEEE Transactions on Speech and Audio Processing, Vol 7, No. 5, September 1999*. IEEE, New York, NY, U.S.A.

[Nabney02] I. Nabney, *Netlab: Algorithms for Pattern Recognition*, Springer-Verlag, 2002, London, UK. Printed in Great Britain.

[Nandy95] D. Nandy; J. Ben-Arie, “Auditory Localization Using Spectral Information”, *Signal Processing Methods for Audio, Images and Telecommunication. (Signal Processing and its Applications),* 1995, Academic Press Limited: Harcourt Brace and Company, Publishers. Printed in Great Britain by the University Press, Cambridge.

[Omolo97] M. Omologo; P. Svaizer, “Use of the Crosspower-Spectrum Phase in Acoustic Event Location”, *IEEE Transactions on Speech and Audio Processing, Vol. 5, No.3, May 1997.* IEEE, New York, USA.

[OrrPh01] M. Orr, D. Pham, B. Lithgow, R. Mahony, “Speech perception based algorithm for the separation of overlapping speech signal”, Proceedings of The Seventh ANZIIS Conference, pp. 341 - 344 Perth, Western Australia, 2001.

[Palm00] W. J. Palm III. *Modelling, Analysis and Control of Dynamic Systems.* J. Wiley & Sons, Inc 2000, pp 662-664

[PuckApZi98] M. Puckette, T. Apel, D. Zicarelli“Real-Time Audio Analysis Tools for Pd and MSP”, *Proc. of the International Computer Music Conference,* 1998, Michigan, USA.

[Polikar03] R. Polikar, “The Wavelet Tutorial”*,* Rowan University, *http://engineering.rowan.edu/~polikar/WAVELETS/WTtutorial.html*

[Quinlan93] J. R. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufman Publisher, Inc. 1993, San Mateo, CA, USA.

[Rabin90] L. R. Rabine, S. E Levinson, A. E Rosenberg, J. G. Wilpon, “Speaker-Independent Recognition of Isolated Words using Clustering Techniques*”, Readings in Speech Recognition*, 1990, Ed. A.Waibel, K.F. Lee, Morgan Kaufmann Publ.Inc., San Mateo, Cal., USA

[Rabin93] L. R. Rabiner, B.H. Juang, *Fundamentals of Speech Recognition*. Prentice-Hall Signal Processing Series, 1993.Englewood Cliffs, NJ, USA

[Rabink96] D. Rabinkin, R. Renomeron, A. Dahl, J. French, J. Flanagan, and M. Bianchi, “A DSP Implementation of Source Location Using Microphone Arrays”, *Proceedings of the SPIE, Vol 2846*, pp. 88-99, August, 1996, Denver, Colorado.

[ReyesEl03] M. Reyes-Gomez, D. Ellis, “Selection, Parameter Estimation, And Discriminative Training Of Hidden Markov Models For General Audio Modeling”, *Proceedings of ICME-03,* Baltimore, USA, July, 2003.

[Rodman99] R. Rodman, *Computer Speech Technology.* Artech House, Inc. 1999, Norwood, MA 02062.

[Ryan00] J. G. Ryan; R. A. Goubran, “Array Optimization Applied in a the Near Field of a Microphone Array”, *IEEE Transactions on Speech and Audio Processing*, Vol. 8, No. 2, March 2000.

[Sampan97] S. Sampan, “Neural Fuzzy Techniques in Vehicle Acoustic Signal Classification”, Masters Thesis, 1997, Virginia Polytechnic Institute and State University, USA.

[Schal90] R. Schalkoff, *Pattern Recognition: Statistical, Structural and Neural Approaches.* 1990, John Wiley & Sons, Inc. New York, NY, U.S.A.

[Setnes99] M. Setnes; R. Babuska, “Fuzzy Relational Classifier Trained by Fuzzy Clustering”, *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics,* Vol. 29, No. 5, October, 1999.

[Shaik99] A. Shaik; C. Jin; S. Carlile, *Human Localisation of Band-Pass Filtered Noise,* 1999, University of Sydney, NSW, Australia.

[Shie99] Q. Shie; D. Chen, “Joint Time-Frequency Analysis”, *IEEE Signal Processing Magazine, Vol 16, No. 2, March 1999.* IEEE, New York, NY, U.S.A.

[Slaney98] M. Slaney, “Auditory Toolbox”™, Interval Research Coporation, version 2, 1998.

[Steele00] M. Steele, “A Direction Finding – Beam Forming Conference Microphone System”, *Proceedings of Texas Instruments - DSPS Fest,* August, 2000.

[Steig96] K. Steiglitz, “Sampling and Quantizing” *A Digital Signal Processing Primer. (with Applications to Digital Audio and Computer Music),* 1996 Addison-Wesley Publishing Company, Inc. Menlo Park, CA, U.S.A. (www.aw.com/cseng/authors/steiglitz/dspp/dspp.html)

[Tipler91] P. Tipler, “Sound”, *Physics for Scientists & Engineers*. Third Edition; Extended Edition. Worth Publishers Inc, New York, USA. 1991, Chap.14; p439 – p483.

[Vander79] N. VanDerveer, *Ecological acoustics : human perception of environmental sounds,* PhD Thesis, Cornell University, 1979.

[Vandew96] G. Van de Wouver; P. Scheunders; D. Van Dyck, “Wavelet-FILVQ Classifier for Speech Analysis”, *Proc. Int. Conference Pattern Recognition,* pp. 214-218, Vienna, 1996.

[Watro90] R.L Watrous, L. Shastri, A.H. Waibel, “Learned Phonetic Discrimination Using Connectionist Networks”, *Readings in Speech Recognition*, Edited by A.Waibel & K.F. Lee. 1990 Morgan Kaufmann Publ. Inc., San Mateo, Cal., USA

[Wood92] J. P. Woodard, “Modelling and Classification of Natural Sounds by Product Code Hidden Markov Models”, *IEEE Transactions on Signal Processing, Vol. 40, No. 7,* July, 1992.

[YuanX99] Z. Yuan; B Xu; C. Yu, “Binary Quantization of Feature Vectors for Robust Text-Independent Speaker Identification”, *IEEE Transactions on Speech and Audio Processing*, Vol. 7, No. 1, January, 1999. <http://tosa.mri.co.jp/nonspeech/RWC2000.pdf>