Audio Signal Classification:
History and Current Techniques

David Gerhard

Technical Report TR-CS 2003-07
November, 2003

Department of Computer Science
University of Regina
Regina, Saskatchewan, CANADA
S4S 0A2

ISSN 0828-3494
ISBN 0 7731 0456 9

Audio Signal Classification:
History and Current Techniques

David Gerhard

Abstract: Audio signal classification (ASC) consists of extracting relevant features from a sound, and of using these features to identify into which of a set of classes the sound is most likely to fit. The feature extraction and grouping algorithms used can be quite diverse depending on the classification domain of the application. This paper presents background necessary to understand the general research domain of ASC, including signal processing, spectral analysis, psychoacoustics and auditory scene analysis.

Also presented are the basic elements of classification systems. Perceptual and physical features are discussed, as well as clustering algorithms and analysis duration. Neural nets and hidden Markov models are discussed as they relate to ASC. These techniques are presented with an overview of the current state of the ASC research literature.

1 Introduction

This paper will present a review of the state of the current research literature pertaining to audio signal classifiation (ASC). I will begin by introducing the field of ASC and continue by discussing the more general research field of auditory scene analysis (ASA) and where ASC fits therein. I will then present the various classical and recent approaches to ASC.

1.1 Motivation for Audio Signal Classification Research

Humans classify audio signals all the time without conscious effort. Recognizing a voice on the telephone, telling the difference between a telephone ring and a doorbell ring, these are tasks that we don't consider very difficult. Problems do arise when the sound is weak or there is noise or it is similar to another sound. For example, I find it difficult to tell which of the doors in the hall outside has just closed.

There are three main areas of motivation for ASC research. First, it would be instructive to know how it is that humans do what they do. If we knew the general systems that we use to classify audio, we might be able to better diagnose and treat auditory ailments. The research that would answer these questions tends to be more psychological and physiological than computational, but the methods used in computer ASC systems might provide a starting point for human ASC research.

Second, it would be nice to have a machine that could do what a human could do with sound. For example, doctors listen to the way a patient breathes in order to diagnose respiratory ailments, and if a medical expert system could do the same, having been programmed with ASC knowledge, then remote areas could get diagnoses quickly without the expense of consulting a human expert who might be in a different country and need to be transported. In the same way, expert auto mechanics are able to diagnose car engine problems by listening to the sounds that the engine makes as it runs. There are many areas where human experts use their ears in their job. ASC expert systems could provide the opportunity for this work to be done in remote communities or in other situations where an expert would be inaccessible or expensive.

Finally, an ASC system has the potential to hear much better than a human. If computers could help us perceive sound in the same way that microscopes, television cameras and instant-replay have helped us to perceive the visual world, we could know much more about the world than we do now. Many tape recorders and voice mail systems provide variable speed playback, allowing the user to fast-forward across easily understood or low content sections, and to slow down and listen carefully to noisy or complicated sections. An ASC system could be constructed to automatically alter the playback speed to keep a constant rate of information. Further potential hearing augmentation applications include noise removal, sound separation and automatic transcription of music, text, Morse code or other sounds used for communication.

1.1.1 Classification Scope

As with all general classification domains, ASC as a research problem must be segmented before it can be solved. There are many smaller problems in ASC that are being worked on at present, and it is at least conceivable that some of the resulting systems might be connected at some time in the future to create a multi-dimensional system, useful for general sound research and classification. Pragmatically, it is clear that individual ASC problems are often solved with individual systems designed for a particular task.

One of the more common ASC problems to be tackled recently is that of the speech/music classifier. Given a piece of sound (often broadcast radio), is the sound source a human speaker or is it some form of music. This problem has interested many people lately, and systems exist that do fairly well at this task [65] [63]. Music can be classified by style, composer or instrumentation, but this type of classification is usually done by looking at musical symbols, not audio signals. Human speech can be classified into language or accent [40] and more more specifically, into which human is speaking, or which word or phoneme is being spoken. These problems are the classic speech and speaker recognition problems, and many researchers have been working on these problems for years [2] [57].

There are also researchers working with specific sound databases, and their data scope depends on what is in their databases, for example, compositional sound effects are investigated in [19]. Other researchers working with more general data use retrieval based on feature similarity and template matching [77] [79].

A further direction concentrates only on transcribeable data [71], sounds that may be written down in some way or other, which include speech, music and some sound effects. Transcribable data is often found in media sound, such as radio or television. The principal direction of researchers in this area is to create systems that will automatically generate closed captioning for television, or teletype for radio, or identify the subject or content of a media program.

1.2 Applications of Audio Signal Classification

ASC applications are potentially far reaching and relevant. Some applications already discussed include speech classification, database applications and automatic transcription. I have divided the applications into the following three areas. Some of these have been implemented, most have not.

1.2.1 Computing Tools

ASC can be used as a front-end for a number of currently existing computer applications to audio. Speech recognition is one of the more obvious applications of ASC, where signals are classified into phonemes and then assembled into words. ASC can be used to improve on current speech recognition technology in many ways. Speech contains much more information than just words, such as emotional content, idiom and emphasis. *Prosody* is the changes in pitch and loudness of speech that convey information, like rising pitch at the end of a question. An ASC system could be developed to identify and take advantage of prosody to create automatic text notations, such as italics, bolding, parentheses and punctuation.

In some cases, it would be good to know what the subject of a piece of speech was before trying to recognize the words. If a system were used to scan radio channels for a traffic report, it would improve performance if the speech could be classified into subject before recognizing the words. This could be done by investigating and identifying stress, accent and other prosodic characteristics of particular speech subjects.

Computers have many ways to look at sound, through speech recognition, music transcription, and command reception. Speech recognizers typically assume that all they will receive is speech, and music transcription systems tend to assume that all they will receive is music. It would be useful to have a general classification system as a front-end to a set of audio processing tools on a machine, and this system could identify incoming sound and route it to whichever sound processing application is appropriate.

Perhaps a more obvious application is toward the creation and use of audio and multimedia databases. ASC would speed up the creation of these databases, and could aid in accessing the database as well. Melody databases could be accessed by direct human input, in the form of humming or whistling or singing at the computer.

1.2.2 Consumer Electronics

Many applications of ASC can be developed into marketable products. This is important because in order to be able to do relevant research, one must have support. Government and industry granting programs are excellent sources of research support, but it is also useful to monetize current research as a source of funding for future research. Also, if research is to benefit humanity, it must be in a form consumable by humanity.

Monetizable ASC applications include embedded devices: microchips that are present in larger devices such as telephones, televisions and automobiles. An embedded ASC device in a telephone could be used to inform the user of the type of responding signal when making a call. The embedded device could be able to tell the difference between a fax, a modem, an answering machine, computer-generated speech or human speech. Depending on the application, different responses would generate different actions from the embedded device, which could instruct the telephone to hang up, redial, connect, wait, or navigate an automated telephone menu.

Devices embedded in televisions and radios could be used to detect advertising, and either mute the sound and blank the picture during the ad, or "channel surf" while the ad continues, returning to the original channel when the program resumes. Of course, this application is not attractive to the companies doing the advertising, who would prefer to have their ads seen and not muted or surfed over. These companies might then endeavor to produce advertisements that would "trick" the classification software into treating the advertisement as if it were scheduled programming. Embedded devices in radio could be designed to search for a particular desired signal, such as a traffic report or a weather report, or for a particular type of music. Many other applications can be found to fit this category, but the discovery of these will be left to the avid reader and to market research analysts.

1.2.3 Automatic Equalization

Equalization is applying, in parallel, a bank of filters to a sound signal. The signal is altered depending on the relative amplification or attenuation of the signal in each filter range, or channel. The intended result of this process is that the equalized signal is of higher perceptual quality than the original signal.

The problem with this process is that it is usually done by applying prefabricated settings, or by manually altering channel settings. Prefabricated settings are more likely not to be ideal, and manual settings require a person to be at the controls. Automatic equalization would analyze the signal and decide which settings would most improve the signal for any particular application. If the signal were identified as speech, a filter setting could be applied that would enhance the "speech-ness" of the signal. Similarly, if the signal were music, an appropriate setting could be applied. Devices exist today that identify feedback in public address systems, and use an equalizer to attenuate the channel or channels where the feedback is happening.

Equalization filters are currently used in hearing aids as well as in public address systems. Some modern hearing aids have a collection of possible filters for various situations. The user of the hearing aid must change the setting by hand, but an automatic equalization system could detect the current environment and apply an appropriate prefabricated filter setting or generate a setting specific to the environment.

1.3 Where Audio Signal Classification fits in the scheme of things

ASC is a subset of a more general research field called auditory scene analysis (ASA), which will be discussed in more detail in Section 2 on page 12. As the name suggests, ASA is the analysis of the auditory scene, which is the entire group of sounds that a listener hears at any one moment. The sounds mix together into a single signal, and yet the individual sources remain identifiable.

Computational ASA is the use of computers to analyze the auditory scene. The first part of the ASA process is often stream segmentation, where an auditory signal is decomposed into a group of signals representing the various sources that are be present in the signal. This is a very difficult problem being addressed by many researchers today. For more information on this topic, see [70] [5] [33].

Once the auditory scene has been divided into individual streams, each stream is analyzed depending on its contents. This is where ASC comes into the picture. In order for these streams to be analyzed, it is helpful to know what the streams contain. ASC can determine which streams contain music, which streams contain speech and which streams contain something else entirely.

Most ASC research concentrates on monophonic signals, which consist of a single auditory stream. This is in preparation for the time when powerful and robust stream separation techniques are available, but it is also for application to signals which by their nature are monophonic, for example a single human speaking in a quiet office or speaking through a localizing microphone such as those mounted on headsets.

Some ASC research is aiming toward polyphonic signals, which contain more than one audio stream. Polyphonic signals are more common and more naturally occurring - the auditory scene is usually a polyphonic signal. It is important for this research to continue regardless of the state of the art in stream separation, because it is likely that stream separation will be a computationally intense process, and it would be instructive to decide on the contents of an auditory scene before trying to separate it.

1.4 Auditory Classes

To be able to talk about ASC, it is important to develop a taxonomy of auditory signals. There is a wide variety of auditory signal classes and sub-classes, many of which overlap. The taxonomy I present here is my own, and is developed in order to study the domain of sound classes. In the process of developing a taxonomy, the developer becomes more familiar with the data domain and thus more able to discuss it and research within it. The taxonomy I present here is summarized in Figure 1 on page 8.

We can begin at the top with the most general class, the root class, which we will call *sound*. Even this is important to define - is sound simply a pattern of waves of air pressure, or does it become sound only when it is detected and interpreted, by human, animal or computer program? For our taxonomy, we can define sound as a pattern of air pressure that is detectable - it doesn't need to be "detected" to become sound, and so we avoid all of the nasty problems of trees falling in forests. We can then split this root class into sounds which a human can hear and sounds which a human cannot hear. It is important to note here that this division depends on which human we are talking about. The average human can hear frequencies between 20 Hz and 15,000 Hz, although young children can sometimes hear sounds with frequencies up to 27,000 Hz [9].

The intensity of the sound is another factor. Sound intensity is measured in decibels, or dB. The softest sound the average person can hear is defined as 0 dB, and louder sounds are measured by the log of the ratio between the intensity of the sound and the intensity of the "0 dB" sound. Most humans experience pain above 120 dB but can perceive sound above that level. It is important to note that the minimum audible pressure and the threshold of pain change depending on the frequency of the sound, with lower frequency sounds having a smaller perceptible intensity range. Both the frequency and intensity ranges decrease with age and exposure to excessively loud noise [50] [9].

There are two problems with using the average human to define hearable sound. First, people with above average hearing would be able to detect sound that we do not consider to be detectable by humans. Second, if we are using this taxonomy to design a system that will hear *better* than a human, it would be instructive to know how well the *best* human hearer hears, and design systems at that level or better. The extreme range of human hearing is theoretically 20 Hz to 27,000 Hz above 0 dB, so our first taxonomical sub-category will be sound containing any frequencies between 20 Hz and 27,000 Hz, with intensity greater than 0 dB. This we will call hearable sound. Non-hearable sound is everything else.

After describing hearable sound, the taxonomy gets more complicated. Hearable sound can be split into pleasant and unpleasant sound, except that this is a very subjective classification. We could classify hearable sound in terms of its components—harmonically simple sounds and harmonically complex sounds, but since most natural sound is harmonically rich, this division will not be very instructive. It would be good to classify sound using the classes of sound that humans naturally use. Some of the more general english nouns and phrases for kinds of sound are: music, noise, talking, natural sounds and artificial sounds. It is important to note here that these divisions are not disparate. Speech could be considered a natural sound, and many artificial sounds are considered noise.

The taxonomy I present is primarily one-dimensional, in that it attempts to divide the domain of sound into discrete chunks along a single axis. As the classification becomes more specific, the taxonomy becomes more multi-dimensional, in that classifications can be made on many scales at the same time. A taxonomy could be fully developed in multiple dimensions, using features of the sound at the top level. Section 4 on page 19 discusses the division of feature space, which is a multi-dimensional division of the sound data domain. Feature space division at higher classification levels leads to a much less intuitive taxonomy, so the present taxonomy will remain linear in the top levels.

1.4.1 Noise

From an engineering perspective, noise is a random or pseudorandom signal that can be classified by the distribution of energy in the spectrum of the signal. “White” noise contains a uniform energy distribution. Coloured noises contain non-uniform energy distribution. Pink noise has constant power per octave ($1/f$ frequency dependence) instead of constant power per hertz, thus being more suited to auditory research. While white noise is physically equal distribution of energy, pink noise *sounds like* it has equal distribution of energy, and this is because of the structure of the inner ear, which will be described later. Brown noise has a frequency distribution of $1/f^2$, and there are other coloured noises as well [66].

Noise can also be classified perceptually. Many people consider popular music of a particular era to be “noise.” From a perceptual perspective, noise is a sound that is unpleasant or undesirable to listen to. It is difficult to define this type of noise in an objective way, however some general comments can be made about perceptual noise. Sounds that contain a large percentage of energy in the higher frequencies are usually considered noise if the energy is not in harmonically related partials, and if the intensity of the signal is relatively high. Anharmonic sounds (sounds that do not contain harmonic series of partials) are often considered to be noisy, again with high intensity but it is interesting to note that some people consider white noise, which is anharmonic, relaxing to listen to, as well as natural anharmonic sounds such as ocean waves.

1.4.2 Natural Sounds

The general class of “natural sounds” is probably the least distinct of the classes at this level. In some sense, all sounds are natural, in the same sense that all matter in the universe is natural. Natural in this taxonomy can be defined as non-human and non-human-influenced, so natural sounds are sounds caused by nature and the natural world. Weather, water, animals and the Earth all make natural sounds.

Natural sounds can be classified by the object that makes the sound. Wind sounds, water sounds and animal sounds are three possible sub-categories, but it is also important to realize that many natural sounds are in fact due to the interaction of more than one object. There are rock sounds, there are water sounds, and there are sounds made by water crashing over rocks or pouring over rocks, or by rocks splashing into water. Wind sounds can be generated by wind through trees or wind interacting with the pinnae, or outer cartilage, of the ears of creatures listening to the wind.

1.4.3 Artificial Sounds

Artificial sounds can be considered the opposite of natural sounds - these are the sounds that *are* human or human-influenced in some way, excluding speech and music. Speech and music are not natural sounds by our definition, so we could put them in this category, but they contain so many sub-classes that we identify them as separate individual classes at this level. Artificial sounds include those made by machinery, cars, buildings and the like. The source of the sounds can be used as a classification feature, as well as the intent. A telephone ring is an artificial sound *intended* to indicate something, as is an ambulance siren, while a jack-hammer is an artificial sound which is unintended and would be removed or restricted if possible. There are some uncertainties here as well - is the sound made by a field of grain planted by a human natural or artificial? The grain is not constructed by humans, and it is not directly manipulated by humans when making the sound, so we consider that a natural sound.

As we move to the city, our environmental sounds shift from mostly natural sounds (with a few cars) to mostly artificial sounds (with a few birds). Again, in this class many sounds are in fact the result of objects interacting as opposed to the sound of objects by themselves. The tires of a truck interacting with the road is an artificial interactive sound.

1.4.4 Speech

Speech can be defined as sound made by the human vocal tract intended for communication. Recorded speech and computer-generated sound that approximates speech are also considered speech. There are many ways to split up the speech domain into sub-categories. An obvious way is to classify speech by language. One can also classify speech by who or what the speaker is, by the emotional content of the speaker, by the subject matter of the speech. Further down the hierarchy, we can classify speech into particular words used, and then particular phonemes.

1.4.5 Music

Music can be defined as sound made humans using instruments, including the human body, communicate particular emotions or feelings. Many people consider some natural sounds to be music, such as waterfalls or birdsongs, but in this taxonomy we will restrict music to be in some way human-made.

As with speech, there are many ways to categorize music. A first classification to make is whether the music is monophonic or polyphonic. That is, whether the music is made by one instrument or a group of instruments. Monophonic music can then be classified into the family of instrument being used (brass, string, percussion, voice, etc.) and then sub-classified into type of instrument (tuba, trombone, etc.), and then into individual instrument, for example the particular individual “D” Marine Band harmonica I have owned for 10 years. In the same way, polyphonic music can be classified by what set of instruments is being played. Both monophonic and polyphonic music can be put into sub-classes depending on the content of the music. These classes can be identified by culture of origin, genre, composer/lyricist, and performer(s).

As with speech, music classification can then be reduced to lower levels. Individual pieces of music can be classified by their chord progression, harmonization or melody, and at the base level, the music can be classified by individual notes.

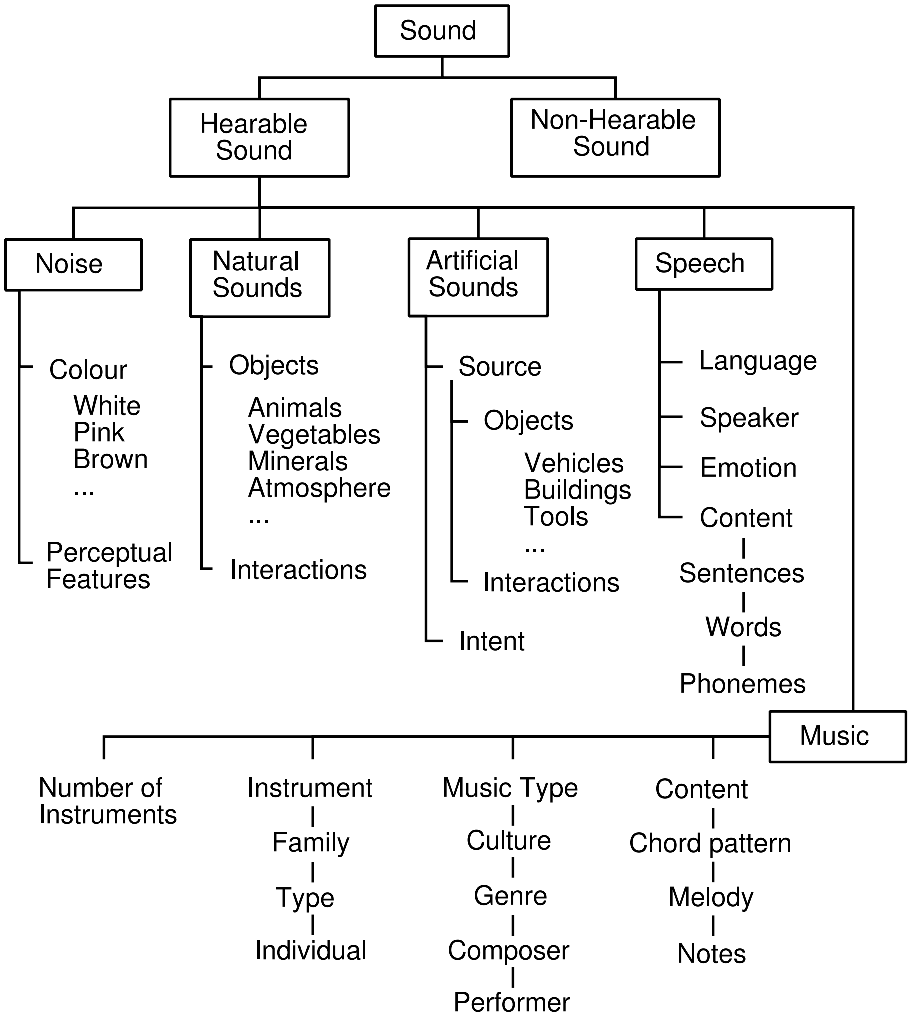

Figure 1: A taxonomy of sound.

1.5 Background: Fourier Analysis

A *sinusoid* is a mathematical function that traces out the simplest repetitive motion in nature. A ball on a rubber band will descend and slow as the band stretches, stop when the gravitational acceleration equals the restoring force of the rubber band, begin to ascend and stop again when the restoring force is zero and the gravitational acceleration equals the momentum. This system is called a simple harmonic oscillator. The repetitive up-and-down motion that it creates is called a sine wave or a sinusoid, and is found in many different forms in nature. In particular it is found in the varying air pressure of sound waves.

Any sound can be created by adding together an infinite number of these sine waves. This is the essence of Fourier synthesis. In the more general sense, any function can be generated from the summation of an infinite number of sinusoids of different *frequencies* and *amplitudes*. The frequency of a sinusoid is how many times it repeats in one second. The amplitude is how high the oscillation reaches. In our ball and rubber band example, the amplitude is the farthest up or down the ball travels from the resting state.

Humans and other vertebrates have an organ called the *cochlea* inside the ear that analyzes sound by spreading it out into its component sinusoids. One end of this organ is sensitive to low frequency sinusoids, and one end is sensitive to higher frequencies. When a sound arrives, different parts of the organ react to the different frequencies that are present in the sound, generating nerve impulses which are interpreted by the brain.

Fourier analysis is a mathematical way to perform this function. The opposite of Fourier synthesis, Fourier analysis consists of decomposing a function into its component sinusoids. The Fourier *transform* is a mathematical way to go between the functional representation of a signal and its Fourier representation. The Fourier representation of a signal shows the spectral composition of the signal. It contains a list of sinusoid functions, identified by frequency, and each sinusoid has an associated amplitude and *phase*. The phase of a signal is the start location of the sinusoid relative to some specific zero. Phase is measured as an angle, in degrees or radians, indicating some part of a complete oscillation. A sinusoid with a phase of 0 radians will be identical to a sinusoid with a phase of 2π radians. These signals are said to be "in phase". A sinusoid with a phase of π radians is the numerical opposite of a sinusoid with a phase of 0 radians. These signals are said to be "out of phase" and if combined, would cancel each other out.

It has been shown that the ear is "phase deaf"[17], which means that two sinusoids with different phases will be perceived as the same sound. In fact, two spectrally rich sounds with all frequency components having different phases, as in Figure 2, will sound the same. For this reason, the phase component of the Fourier representation is often discarded. However it has also been shown that while two steady state signals with the same amplitude spectrum sound the same regardless of their phase spectra, *changes* in the phase spectrum of a signal over time are perceivable. This change in phase is perceived as a shift in timbre, but not in pitch, so the phase information may be important depending on the application.

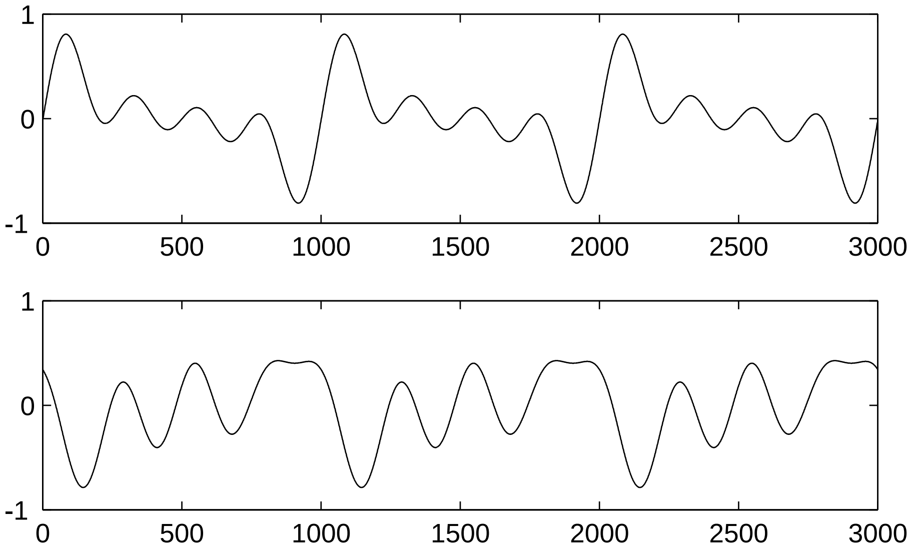

Figure 2: Two sound signals with sinusoidal components of the same frequency and amplitude, but with different phase.

1.5.1 The Short-Time Fourier Transform

The Fourier transform provides information about how much of each frequency is present in a signal. If the spectral content of the signal does not change much over time, then this works quite well, but if the signal changes over time, for example in a song where different notes are played one after another, the Fourier transform will not be able to distinguish between the different notes and the Fourier representation will show information about all of the notes together.

The short-time Fourier transform (STFT) is an attempt to fix the lack of time resolution in the classic Fourier transform. The input data is broken into many small sequential pieces, called frames or windows, and the Fourier transform is applied to each of these frames in succession. What is produced is a time-dependent representation, showing the changes in the harmonic spectrum as the signal progresses.

The original Fourier transform operates on a signal of theoretically infinite length, and so the STFT requires that each frame somehow be expanded to infinite length. This is done by repeating the frame an infinite number of times to produce a signal which is then transformed. As a consequence, there is often a discontinuity, or break in the signal, at the frame boundaries. This introduces spectral components into the transform that are not present in the original signal. The solution to this problem is to apply a windowing function to the frame, which gently scales the amplitude of the signal to zero at each end, reducing the discontinuity at frame boundaries. Using no windowing function is the same as using a windowing function shaped like a square. This is called a square window, or a boxcar window. The windowing functions do not completely remove the frame boundary effects, but they do reduce the effects substantially. Figure 3 shows a simple sine wave windowed with three different windowing functions, along with the corresponding Fourier representations. A single sine wave should have a Fourier representation of a singular component, and as can be seen in Figure 3, no STFT window completely removes the boundary effects, but some do better than others.

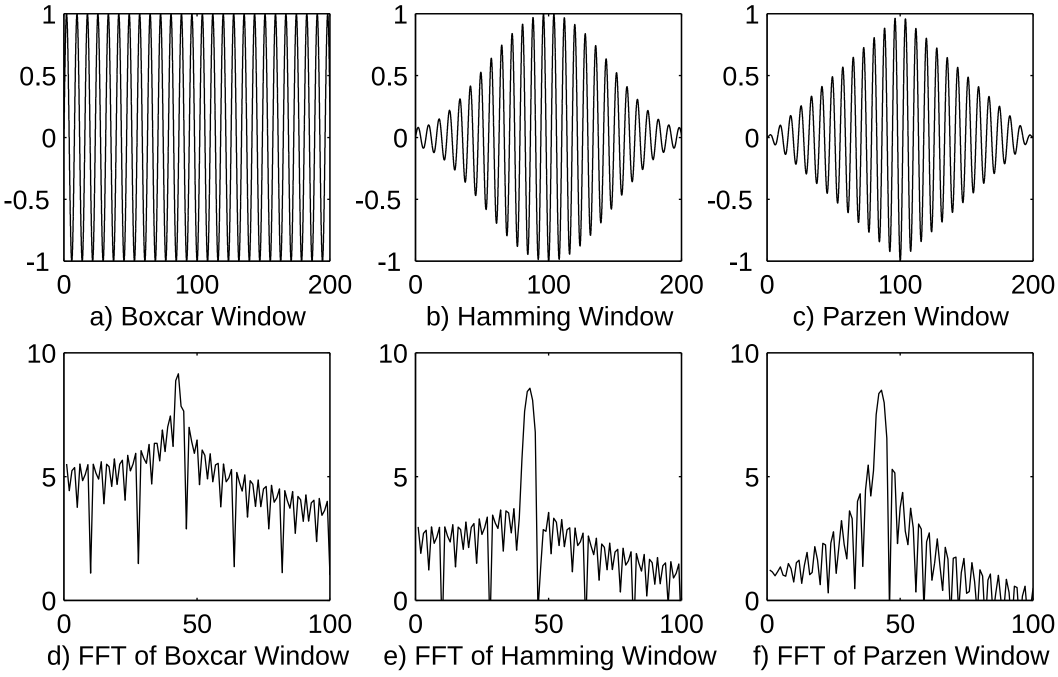

Figure 3: The effect of windowing functions on a sine wave.

Much work has been done to try to design a better windowing function, but as is made clear in [53], the improvements made by these more complicated windows are not worth the extra computation required to produce them. The Hamming window is very simple to implement, takes very little computation time, and yields good results. As long as the windowing function is not square, gradually reducing the amplitude of the input signal toward the edges of the frame will reduce the spectral blur.

When these windowing functions are applied to a signal, it is clear that some information near the frame boundaries is lost. For this reason, a further improvement to the STFT is to overlap the frames. When the each part of the signal is analyzed in more than one frame, information that is lost at a frame boundary is picked up between the boundaries of the next frame.

1.5.2 Other Spectral Techniques

The Fourier transform is not the only spectral transform, it is merely the most common. It was one of the original techniques, it is relatively easy to implement computationally, and it has some relevance to the real-world components of audio signals. It is useful for many applications, but there are things that the Fourier representation is not good at, such as time localization and accurate modeling of human frequency perception.

The constant-Q transform. In the discrete Fourier transform, each frequency band represents an equal amount of the spectrum. This is based on Fourier theory, and is easy to implement and comprehend. Spectrally rich signals that have harmonically related partials appear on the transform as a series of equally spaced peaks.

The human auditory system has long been understood to perform a kind of frequency analysis of the incoming sound. The analysis that is performed by the cochlea, however, is not equally spaced, but logarithmically spaced. Since all studies of sound are, to an extent, studies of the way humans and other vertebrates perceive sound, it makes sense to design a frequency analysis method that models the way the cochlea analyzes frequency.

Thus was born the Constant-Q transform [59]. In signal processing theory, Q is the ratio of the center frequency of a filter band to the bandwidth. The width of each frequency band in the constant-Q transform is related to its center frequency in the same way, and thus is a constant pitch interval wide, typically $\frac{1}{3}$ or $\frac{1}{4}$ of an octave. This allows for more resolution at the lower-frequency end of the representation and less resolution at the higher-frequency end of the representation, modeling the cochlear resolution pattern.

The difficulties with this implementation are that it is more difficult to program, it is more computationally intensive, and it is not necessarily invertible, that is, the result of analysis followed by synthesis might not be exactly the original signal. For non-real-time analysis-without-synthesis, these problems are tolerable.

The phase vocoder. The phase vocoder is, in its most basic state, a windowed short-time Fourier transform, implemented with a specific algorithm (the Fast Fourier Transform algorithm, or FFT). The improvement is to modify the frequency response such that the Fourier transform acts like a bank of bandpass filters at equally spaced frequencies.

The input signal is organized into overlapping frames, and the overlap, the frame size and window type are user-selectable. The output frequency versus amplitude representation is often formed into envelopes for the amplitude and frequency of the sinusoids over time, which can then be modified for later synthesis techniques.

The tracking phase vocoder is an improvement on the basic phase vocoder, where a peak detection algorithm is added, and the peaks in the spectrum are tracked through time according to certain rules. The phase vocoder is in essence a name given to a collection of recent improvements to the Fourier transform. Many Fourier transform systems use these improvements without taking the name "phase vocoder."[59]

Multi-resolution transforms. A major drawback of the Fourier transform is that it is a representation that is based completely in the frequency domain. Using the Fourier transform, one can have information about only the frequency behavior of the signal, without knowing when that behavior occurred, unless a technique like STFT is used.

Multi-resolution techniques look at the spectral makeup of the signal at many different time-resolutions, capturing the low-frequency information about the signal over a large window and the high-frequency information over a smaller window. In the *wavelet* transform, this is accomplished by using a basis function that is expanded and contracted in time [10] [69] [26]. The basis function, called a wavelet, can be thought of as a windowed sinusoid, although this description does not emphasize the mathematical nature of these functions. They are designed to be orthogonal, so that a transform using these wavelets would be reversible.

In the discrete wavelet transform, the wavelet is stretched to fill the entire time frame of the signal, analyzing how much low-frequency information is present in the frame. The wavelet is then scaled to fit half of the frame, and used twice to analyze the first half and the second half of the frame for slightly higher frequency information, localized to each half. Proceeding by halves, the entire frequency spectrum is covered. High-frequency information is highly localized in time, and low-frequency information is less localized.

Multi-resolution transforms, like the wavelet transform, attempt to cross the boundary between a purely time-domain representation and a purely frequency-domain representation. They do not correspond to "time" information *or* "frequency" information, rather the information that they extract from the signal is a kind of time-frequency hybrid. Methods can be employed to extract time or frequency information from a multi-resolution representation such as the wavelet transform.

2 Overview of Research Related to Audio Signal Classification

ASC is a broad research area in itself, but is also part of several much larger research fields. In order to create a system that will classify audio, it is important and instructive to analyze how we as humans perform this task, because many of the desired classifications in audio research are in fact subjective. What is noise to one person may be music to another's ears (literally) and because of this discrepancy, we must consider what is universal in human sound classification and what is not.

ASA is the larger research field into which the ASC fits. Traditionally, much sound research deals with individual sounds and not with combinations of sounds. ASA is the study of how we decompose an auditory scene into its component auditory events. When we are standing at a roadside in a city, we hear individual cars drive by, we hear people talking beside us, and we hear other environmental sounds, each distinctly and not as a single construct. For any classification scheme to work on a sound containing more than one auditory event, some ASA must be performed.

Other research areas that apply to or can be improved by ASC are pitch detection, automatic music transcription, speech and language applications, and multimedia databases. In the following sections, I will present some of the historical and recent work in these areas.

2.1 Auditory Scene Analysis

Albert Bregman's landmark book in 1990 [3] presented a new perspective in human sound perception. Until then, much work had been done in the organization of human visual perception, but little had been done on the auditory side of things, and what little there was concentrated on general concepts like loudness and pitch. Bregman realized that there must be processes going on in our brains that determine how we hear sounds, how we differentiate between sounds, and how we use sound to build a "picture" of the world around us. The term he used for this picture is the *auditory scene.*

The classic problem in auditory scene analysis is the "cocktail party" situation, where you are in a room with many conversations going on, some louder than the one you are engaged in, and there is background noise such as music, clinking glasses, and pouring drinks. Amid all this cacophony, you can readily filter out what is unimportant and pay attention to the conversation at hand. Humans can track a single auditory stream, such as a person speaking, through frequency changes and intensity changes. The noise around you may be louder than your conversation, and still you have little trouble understanding what your friend is saying.

An analogy that shows just how much processing is done in the auditory system is the lake analogy. Imagine digging two short trenches up from the shore of a lake, and then stretching handkerchiefs across the trenches. The human auditory system is then like determining how many boats are on the lake, how fast and where they are going, which one is closer, if any large objects have been recently thrown in the lake, and almost anything else, merely from observing the motion of the handkerchiefs. When we bring the problem out to our conscious awareness, it seems impossible, and yet vertebrates do this all the time every day without difficulty.

Bregman shows that there are many phenomena going on in the processing of auditory signals that are similar to those in visual perception. *Exclusive allocation* indicates that properties belong to only one event. When it is not clear which event a property applies to, the system breaks down and illusions are perceived. One of the more common visual examples of this is the "face-vase" illusion, Figure 4 where background and foreground are ambiguous, and it is not clear whether the boundary belongs to the vase or the two faces. This phenomenon occurs in audition as well. In certain circumstances, musical notes can be ambiguous. A pivot chord during a key change can be perceived as belonging to the old key or the new key, until the ambiguity is resolved by a chord which defines the new key.

Apparent motion occurs in audition as it does in vision. When a series of lights are flashed on and off in a particular sequence, it seems like there is a single light traveling along the line. If the lights are flashed too slow or they are too far apart, the illusion breaks down, and the individual lights are seen turning on and off. In audition, a similar kind of streaming occurs, in two dimensions. If a series of notes are of a similar frequency, they will tend to stream together, even if there are notes of

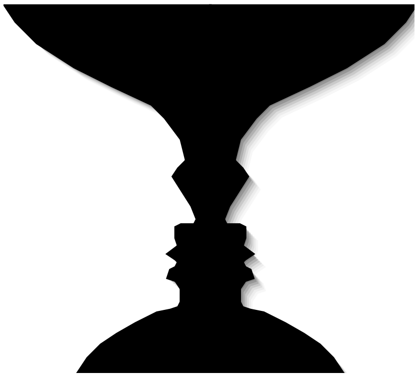

Figure 4: The Face-Vase Illusion.

dissimilar frequencies interspersed. A sequence that goes "Low-High-Low-High..." will be perceived as two streams, one high and one low, if the tempo is fast enough or if the difference between the "Low" and the "High" frequencies is large enough. If the tempo is slow and the frequencies do not differ by much, however, the sequence will be perceived as one stream going up and down in rapid succession.

These links to human visual perception are useful in two ways. First, it suggests concurrent processing for these systems in the brain, or at least similar brain structures. Second, it provides a recognizable frame of reference for research and experimentation.

2.2 Pitch Detection

Pitch detection has been a popular research topic for a number of years now. The basic problem is to extract from a sound signal the fundamental frequency (f_0), which is the lowest sinusoidal component, or *partial*, which relates well to most of the other partials. In a pitched signal, most partials are harmonically related, meaning that the frequency of most of the partials are related to the frequency of the lowest partial by a small whole-number ratio. The frequency of this lowest partial is f_0 of the signal.

Fundamental frequency and pitch are not strictly the same, although they are related. The first difference is that pitch is a perceptual quantity and f_0 is a physical property. Pitch values are related to the log of the f_0 values, with pitch increasing about an octave with every doubling in frequency. The relationship is not directly logarithmic, as frequency doubling above 1000 Hz corresponds to a pitch interval slightly less than an octave [17]. This relationship also changes with intensity. The perceived pitch of a sinusoid increases with intensity when the sinusoid is above 3000 Hz, and a sinusoid with frequency below 2000 Hz is perceived to drop in pitch as the intensity increases [9].

The perception of pitch changes with the harmonic content as well. A pure sinusoid at twice the frequency of a base sinusoid is perceived as slightly less than an octave. A richer spectrum seems to enforce the perception of the pitch, making the octave seem more "in-tune." The more sine-like a signal is, the more distinct the notion of frequency, but the less distinct the perception of pitch [74]. This sensation also varies with the relationship between the partials. The more harmonically related the partials of a tone are, the more distinct the perception of pitch and the more anharmonic partials a tone has, the less distinct the perception of pitch.

Most research into this area goes under the name of pitch detection, although what is being done is actually f_0 detection. Because the psychological relationship between f_0 and pitch is well known, it is not an important distinction to make, although a true pitch detector should really take the perceptual models into account and produce a result on a pitch scale instead of a frequency scale.

Pitch and f_0 detection are interesting to ASC research primarily because f_0 is often used as a feature in ASC systems. Classification systems based on melody, such as song recognition systems, rely heavily on f_0 as an indicator. Speech recognition systems use f_0 information for recognition, as well as to extract hidden content in the form of prosodic information. More information on pitch and f_0 detection is presented in Section 4.1.4 on page 22, and in Section 4.2.1 on page 25

2.3 Automatic Music Transcription

The purpose of an automatic music transcriber is to receive as input a musical signal, from a live concert or from recorded media, and produce as output a fully notated score. In this ASC problem, the signal is assumed to be music and is classified according to individual notes. The initial burst of research on this application occurred in the mid 1970's [48], when computers were first becoming able to deal with the immense amount of data used to represent "high-fidelity" recorded music. Parts of the problem were solved quickly, but other parts proved to be harder than first anticipated. For monophonic signals, the solution is trivial, and a thorough treatment is presented in [51]. Refinements to pitch detection make these monophonic music transcribers more accurate, but there has not been a significant change in the methodology since the early 1980's. The algorithmic breakdown of an automatic music transcription machine generally follows three stages:

1. Spectral Estimation
2. Pitch Detection
3. Symbol Formation

In each of these stages, information from the previous stage is used to make decisions. Spectral estimation is usually done with a Fourier-type analysis, although time-domain pitch detection methods are also used. The pitch information must then be organized into a format recognizable by human or computer. A common method is to present the data in MIDI (Music Instrument Digital Interface) format, leaving the responsibility on other researchers to develop tools to translate that data to a human-readable form.

The symbol formation part of music transcription is a very complicated and difficult task. A melody line represented by a series of pitches could be represented in any key signature and time signature. It takes intelligence and experience to decide what the most likely representation is. Grammar rules have been suggested for music [42] [28] which try to make this task less difficult, but an uncertainty principle is present here, in that the generality of any algorithm in this context is inversely related to the accuracy. A musical grammar set up to deal with all of the obscure and uncommon notational possibilities will find it more difficult to notate a simpler passage accurately, just because there are more notational possibilities to choose from.

The more difficult form of the automatic music transcription problem is the polyphonic case. Here, there are several instruments playing at one time, and many notes overlap, as in the case of a symphony orchestra. Even a piano or a guitar playing a chord is a case of polyphonic music, requiring the identification of several concurrent pitches. The difficulty of this problem becomes apparent when one considers the case of the pipe organ, where several pipes sounding concurrently, in particular harmonic relationships, perceptually "fuse" into a single note and cannot be identified as a set of individual notes. The question is complicated further by asking whether we would *want* such a note to be separated into its constituent parts or to be identified as a single note.

The problem of polyphonic pitch detection is not currently solved, but there is much work being done. Many researchers are separating the problem into two pieces: auditory stream separation [3] and monophonic pitch detection. If the polyphonic sound can be separated into a number of monophonic streams, then any monophonic music transcription system can be used on these individual streams. Spectral cues, like simultaneous vibrato and attack synchronicity are allowing researchers to split the sound signal into component streams.

2.4 Speech

Speech has been one of the fundamental audio research topics for many years now. Initially, much research was done on producing a machine that could synthesize speech, and as computers made this task easier, more research was directed toward producing a machine that could understand and analyze speech. Three standard topics in speech research as it applies to classification are speech recognition, speaker recognition, and voice detection.

2.4.1 Speech Recognition

This is the fundamental speech classification problem. The goal is to produce readable text from human speech. The fundamental problem of speech recognition is the analysis of ambiguous utterances. The phrases "Oh, my my!" and "Ohm, I'm I!" when spoken naturally are phonetically identical. This is where years of experience make humans good at the task. One of these phrases is clearly more likely than the other, even though we don't know the context. Other popular examples are "It's easy to recognize speech," which, when spoken, could be interpreted as "It's easy to wreck a nice beach," and "Euthanasia," which could just as accurately be interpreted as "Youth in Asia."

There are many comprehensive works on speech recognition that describe the history of the field [57] [31] [2] [27], and I refer the interested reader to these works for a history. The basic speech recognition algorithm is a classification algorithm. The system must decide which phoneme is being spoken at any point, and this is done with spectral analysis. Pattern matching and temporal shifting algorithms, such as hidden Markov models, are employed to see which of a set of templates is the closest match with the phoneme being classified. Many systems classify by diphones or triphones, two or three phonemes at the same time, working on the observation that human speech production often involves slurring phonemes together. Once the phonemes have been classified, the recognizer must assemble the phonemes into words. This is a large task in itself—many different word streams could be generated from an individual phoneme stream, and one of the jobs of the speech recognizer is to decide which word stream is most likely.

Speech recognition technology is commercially available now, and these application programs do a respectable job. The technology is nowhere near the ultimate goal of speech recognition research, which is fully automatic speaker independent recognition of natural speech in a natural environment. Presently, recognition applications can work on natural speech for a specific speaker on a headset microphone, or on separated-word speech independent of the speaker on a headset microphone, or on natural speech independent of the speaker on a headset microphone with a significant error rate, requiring human post-processing to clean up the text. As the technology and research continue to advance, speech recognition will improve, hopefully to the point where headset microphones are not necessary, or speech need not be word-separated, or speakers need not train the system before use.

2.4.2 Speaker Recognition

A commonly desired result of speech research is the speaker recognizer. The idealized application is a security device where the entry password is spoken, an analysis is done of the voice of the speaker, and if the voice is identified as someone who should be allowed to enter, the door is opened. Recently, Apple Computer has released "voice password" software in their operating system, where a user is given access to their file space only if their voice matches the expected voiceprint. This is a much smaller problem—the expected speaker is known, the expected phrase is known and the environment is likely to be low on noise. The more difficult problem is having a large number of authorized speakers, and making a decision on whether the speaker is a member of the list of authorized people without a keyphrase, without knowing which person the speaker claims to be, and with the potential of environmental noise.

Speaker Recognition is also called Speaker Verification [44] and current research consists of attempting to discover a set of speech parameters that has enough variability between speakers to be useful. Another problem in speaker verification is to discover an appropriate distance metric. If the metric is too precise, then small variations will lead to incorrect negative results. For example, if a person has a respiratory ailment like a cold, she might not be recognized as who she is if the system is too strict. On the other hand, the system must also be able to deal with linguistic impostors—people who try to sound like other people.

Many environments for speaker verification are not ideal. A building entry system would not likely have a headset input, but a microphone mounted on the wall next to the door. The environmental noise is a significant factor here, as is the psychological effect of humans unconsciously changing their voice pattern when trying to be heard and understood, for example by raising pitch or increasing the proportion of high-frequency partials.

2.4.3 Voice Detection

Neither of the above applications would work especially well if the algorithm were applied to something other than speech. Since the assumption is that the system will hear only speech, it is desirable to develop a system that will ensure that only speech is heard by these systems. This is one of the driving forces behind the field of voice detection. If a system were developed that would pass speech and block other signals, then used as a pre-processor it would ensure that the speech analysis systems receive only speech, and thus these systems can be optimized for this situation. Another type of pre-processor is the speech cleaner, which takes noisy speech and filters out the noise.

The first voice detectors, which are still in use today, are the so-called "voice-activated recorders" which are in fact set up to be sensitive to the intensity of the incoming signal. If the signal is loud enough, the recorder is activated, and if the signal drops below the threshold, the recording stops. In some of the more advanced models, the threshold is controllable by the user, but these are clearly not voice detectors in the true sense. Any noise of sufficient volume would trigger the recording device. A truly voice-activated recorder would engage only in the presence of speech.

Voice detectors have been developed for specific situations, in the form of speech/music discriminators [63] [65]. These systems have been developed for the specific goal of segmenting speech from music, but many of the tools used to make this classification can be extended to make a more general classification of speech versus all other sound.

3 Classification System Requirements

ASC, like other classification problems, requires an approach of many steps. The steps to be taken are common, but how the steps are performed, and which steps might be skipped, differs from problem to problem.

3.1 Features

The first step in a classification problem is typically data reduction. Most real-world data, and in particular sound data, is very large and contains much redundancy, and important features are lost in the cacophony of unreduced data. The data reduction stage is often called feature extraction, and consists of discovering a few important facts about each data item, or *case*. The *features* that are extracted from each case are the same, so that they can be compared. Feature extraction is rarely skipped as a step, unless the data in its original form is already in features, such as temperature read from a thermometer over time. ASC systems take as input a sound signal, in the form of a series of voltages representing sound pressure levels. The important information is usually in the form of quantities like frequency, spectral content, rhythm, formant location and such. These features can be *physical*, based on measurable characteristics, or *perceptual*, based on characteristics reported to be perceived by humans. Section 4 divides the feature domain discussion into these two categories.

3.2 Clustering

The next step in any classification problem is to find what feature values correspond to which categories. This can be done manually, by looking for features that might successfully separate a group of cases into desired classes. For example, if the goal was to separate birdcalls from other environmental sounds, an obvious feature would be whether the sound were periodic or not. If the goal were to separate starling calls from robin calls, this feature would not be as useful. In this situation, when the cases are more similar, the features to be used would be more difficult to identify by hand, and so automatic algorithms, such as *clustering* algorithms, are employed. Many clustering techniques are available depending on the type of data being clustered and how much pre-knowledge is available. A different clustering system would be employed if the desired categories are known a-priori, than one where the categories are discovered as the algorithm progresses. Section 5 presents a discussion of some of the clustering methods being used today.

3.3 Analysis Duration

When examining a feature of a sound, it is important to be aware of how much of the signal is being used to extract the feature. Some features can be extracted only from the entire sound, and some features are extracted from a short chunk of the sound. Many early classification systems used a single analysis duration frame, of a fixed size, swept across the signal. If the window is small enough, this method can be successful, as features requiring longer duration can be extracted from successive frames, but this often requires much more calculation than would be necessary by extracting features from frames of different sizes. Analysis duration is also used in the *template-matching* method of classification. Template comparison requires some form of duration matching, especially in situations like sound classification where sounds in the same class can have different durations. One method of duration matching is linear time stretching, where the time scale of the signal is stretched to match that of the template. This method is not very successful with ASC because it alters feature values such as frequency as it stretches. Hidden Markov models, or HMMs, are a method of duration matching that does not alter feature values. In the most basic description, they track the occurrence of expected events as a signal progresses. Section 6 discusses different methods of looking at the duration of analysis as well as methods of signal duration matching.

3.4 Classification Depth

Classification systems are usually used to assign an item to one of a *small* number of classes. As classification systems become more general, the number of classes to choose from becomes greater until the system performance degrades beyond acceptable levels. At this point, the designers must make a choice. Either limit the number of potential classes, or increase the *depth* of classification. Hierarchical classification systems contain a number of classification engines, each more specific than the last, and each limited to a small number of classes from which to choose. For example, an automobile classification system might begin by classifying by manufacturing company, and then classify as car, truck, van, station wagon or SUV. At this level, the choices are much reduced, because one particular company might not make trucks, and this knowledge makes the decision easier. As the classification becomes more precise, the process can be stopped at any point when the desired information has been made available.

Current ASC systems typically choose between a small number of potential classes, and as such few ASC systems use hierarchical classification. As the systems progress to become more general, hierarchical classification may well be used. The classification systems discussed in this paper are all single-level systems, so there will be no more discussion about hierarchical systems.

4 Feature Extraction

Feature extraction is typically the first stage in any classification system in general, and in ASC systems in particular. Some researchers have elected to apply a pre-processing module to their system which filters out unnecessary information for the particular application. For example, Kumpf and King [40] use a Hamming window and preemphasis in their accent classification system, because the data domain contains only speech. Researchers attempting more general classifiers typically have not used a pre-processing module, as it has the potential to remove information that would be useful for classification.

The features that are used in ASC systems are typically divided into two categories: *perceptual* and *physical*. Perceptual features are based on the way humans hear sound. Examples of perceptual features are pitch, timbre and rhythm. Physical features are based on statistical and mathematical properties of signals. Examples of physical features are f_0, Zero-Crossing Rate (ZCR), and Energy. Some perceptual features are related to physical features, as pitch is related to f_0, and timbre is related to the spectral content.

An interesting division of features is presented in [77], where the user enters information about a sound to be retrieved from a database. The user describes the sound in terms of features, and the authors divide these features into three categories: Acoustical/Perceptual features, Subjective features and Simile and Onomatopœia. Acoustical/Perceptual features take into account all of the features we have described so far. If a user is competent enough in engineering or signal processing, she can request a sound with ZCR in a certain range, or can request a sound with a given pitch track, typically input by singing or humming. Subjective features encompass what the authors call personal descriptive language, which can be more difficult for the system designers to deal with but can often be much more informative. An example of a subjective feature that the authors give is “shimmering”. Simile is requesting a sound by saying it is like another sound. This is often used to select a sub-category, like speech or noise. Onomatopœia is a way to request a sound by imitating the sound, for example making a buzzing noise to look for a sound of bees or electrical hum.

4.1 Physical Features

Physical features are typically easier to recognize and extract from a sound signal because they are directly related to physical properties of the signal itself. Perceptual features are related to the way humans consciously perceive the sound signals, and as such rely on a great deal of perceptual modeling. It is because of this that many researchers have elected to base their sound classification systems primarily on physical features. They are easier to define and measure, although they are not as directly relevant to human experience.

4.1.1 Energy

Perhaps one of the most straightforward of the physical features, energy is a measure of how much signal there is at any one time. Energy is used to discover silence in a signal, as well as dynamic range. The energy of a signal is typically calculated on a short-time basis, by windowing the signal at a particular time, squaring the samples and taking the average [79]. The square root of this result is the engineering quantity known as the root-mean square, which has been used by other researchers [77] [65]. Since most of these features are examined on a relative scale, as opposed to an absolute scale, The square root is not necessary, and may be used depending on the data and classification result desired.

Features *related* to the energy of the signal have also been used. Energy in specific frequency bands, and in particular, the variance of the low sub-band energy, is used in [49] to detect silence. Their argument is that strict energy thresholding would not detect the difference between frames which contained no signal and frames which contained signal with low energy, such as the beginning or end of a fade.

The distribution of energy over time has been used to distinguish between speech and music. Speech tends to consist of periods of high energy (voiced phonemes) followed by periods of low energy (unvoiced phonemes, inter-word pauses), while music tends to have a more consistent energy distribution. A measure of the energy distribution is used in [63], while a measure of the energy modulation rate is used in [65], where they claim that speech tends to have a modulation energy of around 4 Hz.

4.1.2 *ZCR* and related features

The zero-crossing rate (*ZCR*) is a method that has recently gained standing in the sound classification research literature. Since it was made popular in [36], its utility has often been in doubt, but lately it has been revived. Put simply, the *ZCR* is a measure of how often the signal crosses zero per unit time. The idea is that the *ZCR* gives information about the spectral content of the signal.

One of the first things that researchers used the *ZCR* for was f_0. The thought was that the *ZCR* should be directly related to the number of times the signal repeated per unit time, which is the frequency. It was soon made clear that there are problems with this measure of f_0 [59]. If the signal is spectrally deficient, like a sinusoid, then it will cross the zero line twice per cycle, as in Figure 5a. However, if it is spectrally rich as in Figure 5b, then it might cross the zero line many more times per cycle. A *ZCR* f_0 detector has been developed with initial filtering to remove the higher partials that contaminate the measurement, but the cutoff frequency needs to be chosen carefully to remove as much high-frequency information as possible without removing the f_0 partial of a higher-frequency signal. Another possibility for the *ZCR* f_0 detector would be to detect patterns in the zero-crossings, autocorrelating to find repetition.

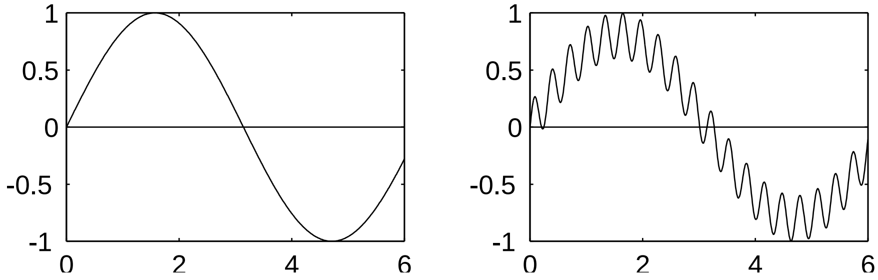

Figure 5: Influence of higher harmonics on zero crossing rate. On the left is $sin(x)$. On the right is $0.8sin(x) + 0.2sin(20x)$. Note multiple zero crossings in one cycle. (after [59])

It has since been shown that *ZCR* is an informative feature in and of itself, unrelated to how well it tracks f_0. Many researchers have taken to examining statistical features of the *ZCR*. For example, [65] uses the *ZCR* as a correlate of the spectral centroid of the signal,which indicates where most of the energy of the signal is. If the spectral centroid is of fairly high frequency, it could mean that the signal is an unvoiced human speech phoneme.

A purely statistical use of the *ZCR* is found in [63]. The author gathered data about how the *ZCR* changes over time, and called this a *ZCR* contour. He found that the *ZCR* contour of speech was significantly different than that of music, and used this feature to help discriminate between the two. A similar use of the *ZCR* is the Short-Time Average *ZCR* feature, used in [79]. Again, they use the *ZCR* as a measure of the spectral characteristics of the signal, to differentiate between speech and music. These unintuitive uses of the *ZCR* show an advantage of physical features over perceptual features - that some useful features of sound signals are not immediately evident from what we as humans perceive.

One of the most attractive properties of the *ZCR* and its related features is that these features are extremely fast to calculate. The *ZCR* is a time-domain feature, which means it is not necessary to calculate a spectrum before extracting information. It can be calculated in real time "on the fly," keeping a running total of the zero-crossings as the signal is received. A system which uses features entirely based on the *ZCR* would not even need analog-to-digital conversion. It would only need to sense when the input signal voltage is positive and negative, and send a pulse whenever the sign of the signal changes.

4.1.3 Spectral Features

The spectrum of a signal describes the distribution of frequencies in the signal. It has been shown that the human ear performs a kind of spectral analysis [47], and since humans extract information from the spectrum when hearing, it stands to reason that a system designed to process sound would benefit from examining the spectrum of the sound. Apart from this, spectral techniques have been used historically to analyze and classify sound.

The spectrogram is the time-varying spectrum of a signal. To generate a spectrogram, the signal is broken into frames, as for the STFT, the spectrum is calculated on each frame and these spectra are displayed as a time-varying spectrum. The result is a measure of the way the frequency content of the signal changes over time.

Many physical features of the spectrum of a signal can be used for classification, depending on the classification goal. One of the most fundamental spectral measures is *bandwidth*, which is a measure of what range of frequencies is present in the signal. This feature is used in [63] to discriminate between speech and music. In this case, music typically has a larger bandwidth than does speech, which has neither the low-frequency of the bass drum nor the high frequency of the cymbal. Bandwidth is also used in the system in [77], and in this case the bandwidth is calculated by taking the average of the difference between the frequency of each spectral component, and the spectral centroid of the signal. The authors of this paper also use the mean, variance and autocorrelation of the bandwidth as features.

A general feature called *Harmonicity* is used as a feature in several classification systems [77] [63]. Harmonicity refers to relationships between peaks in the spectrum. An object that vibrates in a resonant way, such as the human voice or a musical instrument, creates a sound that has strong frequency peaks at evenly spaced intervals across the spectrum. The harmonicity of a sound can be used to differentiate between voiced and unvoiced speech, or to identify music.

The speech/music classification system presented in [65] uses several features based on statistical measures of the spectrum and spectrogram. These include spectral rolloff point, spectral centroid and spectral flux. The spectral rolloff point is the frequency below which most of the spectral energy exists, and is used to distinguish between voiced and unvoiced speech. The spectral centroid is a measure of the average frequency of the signal. Music tends to have a higher spectral centroid than speech because of the percussive sounds. The spectral flux is a measure of the rate of change of spectral information, and music tends to have a higher rate of spectral flux than speech.

4.1.4 Fundamental Frequency

f_0 is only relevant for periodic or pseudoperiodic signals. Periodic signals are signals which repeat infinitely, and perceptually a periodic signal has a pitch. Pseudo-periodic signals are signals that *almost* repeat. There is a slight variation in the signal from period to period, but it can still be said to have a f_0, corresponding to the slowest rate at which the signal *appears* to repeat. It is clear that extracting the f_0 from a signal will only make sense if the signal is periodic. f_0 detectors often serve a dual purpose in this case—if the f_0 extracted makes sense for the rest of the signal, then the signal is considered to be periodic. If the f_0 appears to be randomly varying or is detected as zero, then the signal is considered to be non-periodic.

In a sound or multimedia database such as the one discussed in [77], f_0 is an important feature for distinguishing between pieces of music, or for retrieving pieces of music based on the melody. Here, they use the STFT with a peak extractor to identify the f_0 of the signal. In many of these systems, there is no differentiation made between pitch and f_0, and although the difference is well understood and easily modeled, it is important to remember that many of these systems do not include perceptual models of pitch detection. For more on multimedia databases, see [73] and [78].

f_0 is used to detect speech word boundaries in [58]. The idea here is that large variations in f_0 are unlikely to happen in the middle of a word, more likely they will happen at the end of the word. The authors discuss the utility of the method on various Indian languages (Hindi, Bengali, Marathi and Telgu) as well as German, however they do not discuss the f_0 extraction method used.

4.1.5 Formant Location

Voiced human speech is generated by a source (vocal cords) generating a periodic function (a glottal pulse) which is shaped by a filter (the vocal tract). The transfer function of the filter has peaks at specific frequencies, called *formants*, depending on the phoneme being articulated. In traditional speech recognition, the relative frequencies of the first two formants are typically used to identify the vowel being formed [57] [2]. While formants exist primarily in voiced speech, they also exist in some unvoiced sounds. Whispered speech is completely unvoiced, yet we can understand it as we understand normal speech. This is because whispered speech contains formants, as shown in Figure 6.

Formant location has been used for many years in traditional speech recognition, but it has also been used recently for more specific sound classification. A male/female classification algorithm has been proposed in [76] which uses the location of the first three formants of the sound signal to classify the gender of the speaker. The authors gathered data about the average formant frequencies for males and females, and found that there was sufficient difference to use this as a classification feature.

Since formants only exist (by definition) in human speech, this feature is useless for identification or classification of noise, non-vocal music, environmental sounds or artificial sounds. Within the domain of speech, it cannot be used on utterances that do not contain formants. It is possible that formants could be used to classify for emotion, prosody, content, language or accent. Accent classification is discussed in [40], where a foreign accent in a local language is identified by foreign phonemes in the local language.

4.1.6 Time-Domain Features and Modulation

Features based on time information are often used in cooperation with other features. Alone, time-based features are often not strong enough to make a classification, but as is the case with

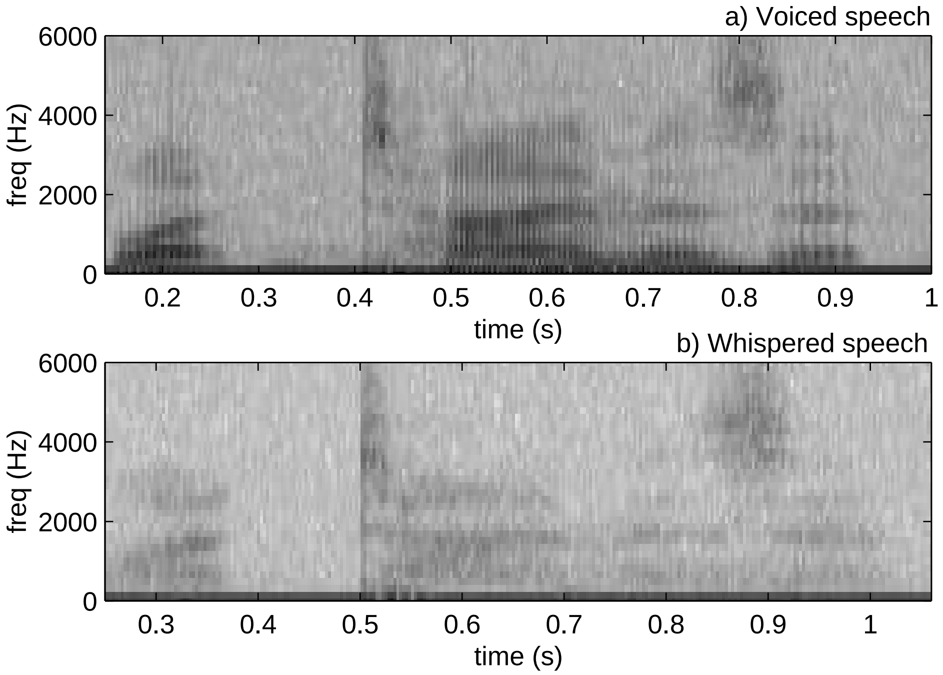

Figure 6: A comparison of normal speech and whispered speech using the spoken phrase "What time is it?"

most of the recent classification systems, features are used in combination to generate a reliable classification.

The simplest of the time-based features is the duration of the sample itself. In the multimedia database application described in [77], the duration of the sound is used as a feature. The sound of a finger snap is likely to be shorter than the sound of an explosion, which is again likely to be shorter than a sample of applause. Melody recognition, on the other hand, likely can not make use of duration as a recognition feature, since durations vary between versions of a song, and specifically between the version or versions stored in the database and the version being sung or hummed as input. Duration matching methods can be employed when the length of a sound is likely to be different than the length of the stored template. For more information on duration matching methods, see Section 6 on page 30.

The classification systems presented in [63] and [65] both use the duration of harmonic segments as a feature to identify speech. In speech, the duration and spacing of syllables tends to be fairly regular, while in other sounds, and specifically music, tone lengths tend to vary much more widely. This feature is measured in [65] as modulation energy, by filtering the signal at 4 Hz (the theoretical modulation rate of speech syllables) and using the energy in the 4 Hz band as the feature indicator. In [63], this feature is referred to as tonal duration, and is measured by first finding syllable onsets and offsets, using *ZCR* to identify fricative consonants, and then finding the time between syllables.

4.2 Perceptual Features

When extracting perceptual features from a sound, the goal is often to identify the features that we as humans seem to use to classify sound. Most perceptual features are related in some way to some physical feature, and in some cases, it is just as instructive to investigate the physical counterparts to these perceptual features. When physical features cannot be found that correspond to perceptual features, it is sometimes necessary to extract information by example, and classify based on templates of sounds which have been identified to contain a certain perceptual feature.

4.2.1 Pitch and Prosody

Pitch seems to be one of the more important perceptual features, as it conveys much information about the sound. It is closely related to the physical feature of f_0. While frequency is an absolute, numerical quantity, pitch is a relative, fluid quantity. A good example of this discrepancy is found in [48], where a system was developed to transcribe sound into a musical score. The system worked correctly, but provided an unexpected result—it placed the musical piece in the key of $C\sharp$ instead of the key of C, because the guitar was tuned slightly sharp.

Humans can perceive pitch in situations where current f_0 detectors fail. One of the most interesting examples of this is the phenomenon of the missing fundamental [60]. When presented with two simultaneous pure tones at a given interval, the human auditory system "hears" the fundamental frequency that would be common to both tones, if it were present. Thus, if two pure sinusoidal tones a fifth apart were played, a pure tone an octave below the lower of these tones would be perceived, as a fundamental to the perceived harmonic series. This implies that for pitch perception, the frequency spectrum of the signal is at least as important as f_0.

Prosody is a characteristic of speech corresponding to changes in pitch and phoneme duration, as well as significant pauses across a spoken phrase, indicating deeper meaning. For example, if the pitch of a phrase rises at the end, it is likely to be a question. Prosody is also used to emphasize certain words in a phrase. The sentence "I took her to the store," can mean different things depending on which part of the sentence has emphasis. "*I* took her to the store," conveys a different meaning than "I took *her* to the store," or "I took her to the *store*." This emphasis can be generated using higher pitch, energy, increased phoneme duration or a significant pause before the emphasized word.

The analysis of prosody for classification usually implies that speech recognition has already been performed, and the prosody of the speech conveys further meaning. As an example, in [46] the authors use prosody, among other tools, to identify dialog acts, or fundamental pieces of speech. Prosody could also be used when speech recognition has not been performed. A sentence with prosodic raising at the end could be classified as a question, and other characteristics of speech could be identified using prosodic features.

4.2.2 Voiced/Unvoiced Frames

One of the fundamental first steps in any speech recognition system is the classification of frames as voiced or unvoiced. On first inspection, this seems like a fairly physical feature: voiced frames tend to be harmonic, and have a lower spectral centroid than unvoiced frames, which tend to be anharmonic. This is where the difference between physical and perceptual features can be deeply understood. Classifications such as "voiced" and "unvoiced" are labels that we put on the phonemes that we hear, without consideration of the physical quantities that these labels represent.

The voiced-ness of a piece of sound can be used as a classification feature. The voiced-ness decision is often made with a pitch or f_0 detector, as in [75] and [30]. The assumption is that the input is speech, and thus when there is significant energy in a frame but no discernible pitch in normal speech range, the frame can reliably be classified as unvoiced. The system is really classifying on the basis of pitches in a certain range, but in the domain of speech recognition, this corresponds to a voiced/unvoiced classification.

4.2.3 Timbre

When humans discuss sound, they talk of pitch, intensity, and some other well-definable perceptual quantities, but some perceptible characteristics of a sound are more difficult to quantify. We clump these characteristics together, and call them collectively "timbre," which has been defined as that quality of sound which allows the distinction of different instruments or voices sounding the same pitch. Most of what we call timbre is due to the spectral distribution of the signal, specifically at the attack of the note. Many spectral characteristics, as discussed above, can be used as classification features, and many of these correspond to the timbre of the sound.

Zhang and Kuo provide a discussion of timbre in [79], and consider it the most important feature in differentiating between classes of environmental sounds, as well as speech and music. Acknowledging that spectral information contained in the attack of the note is important in timbre, they state that the temporal evolution of the spectrum of audio signals accounts largely for the timbral perception. Unfortunately, they do not discuss their method for extracting timbral information from a sound. The extraction of physical features that correspond to timbral features is a difficult problem that has been investigated in psychoacoustics and music analysis without definite answers.

The authors of the multimedia database system discussed in [77] describe subjective features as well as acoustic and perceptual features of sound. Words used to describe timbre include "shimmering", "bright" and "scratchy", and these ideas can be used in a template matching system, which would classify on the basis of timbre without identifying the corresponding physical features. The system collects examples of sounds that have been classified as "scratchy", clusters them according to the features they have in common, and uses these features to decide if a new sound belongs to this category or not.

4.2.4 Rhythm

When a piece of sound is considered *rhythmic*, it often means that there are individually perceivable events in the sound that repeat in a predictable manner. The *tempo* of a musical piece indicates the speed at which the most fundamental of these events occur. Researchers who are attempting to extract rhythmic information from a piece of sound often look at repetitive events in energy level, pitch or spectrum distribution, but musical rhythm is often not as simple as a pulse in energy every second or so. More likely, there is a complicated series of events, fitting into a rhythmic framework that repeats. The problem is that the tempo of this rhythmic framework often changes minutely throughout the piece, for example increasing during a particularly intensive part of the piece.

Another problem with rhythmicity as a feature is the question of what the feature will indicate. If a system is being designed to differentiate between speech and music, which class does rhythmicity indicate? Music is usually rhythmic, but there are forms of speech that are also rhythmic, such as poetry and chant. Are these music? Most other features would indicate speech. Will the rhythmicity detector override other features in this case? What if the speech is not intended to be poetic, but ends up being rhythmic because the speaker tends to speak in a rhythmic way? These are all problems to be considered with the feature of rhythm.

A rhythm detector was employed in [65], in the form of a "pulse metric," using autocorrelation to extract rhythmicity. The signal is filtered to permit various bands of energy, and each band is autocorrelated. The authors indicate that this method is useful to detect a strong driving beat in music, but fails when the rhythm deviates very much from a central time, as in *rubato* or "robbed-time" music. Rubato music has an underlying beat that is intentionally inconsistent in its duration, allowing for emotional expression in the music, but making the job of the rhythm detector very difficult. Some modern dance music uses machine-generated drumbeats, which are usually very rigid in the beat time, and because of this the pulse metric performs well on detecting the presence of rhythm in this type of music.

A more general classification system presented in [79] uses rhythm to detect sound effects such as footsteps, clock ticks and telephone rings. While they discuss the effects of rhythm, and why it is a useful feature, they do not discuss the extraction methods used in their rhythm detector.

5 Clustering Algorithms

Clustering is a very broad research topic of its own, and could easily constitute a complete depth paper. In this section I will give a brief overview of clustering as it applies to acoustic signal processing, and I will discuss (in fairly low detail) some of the issues involved in selecting an appropriate clustering technique, as well as some of the current techniques in use today. For a more complete discussion on clustering techniques, see [16] or [67]. For clustering as it applies specifically to sound classification and speech recognition, see [57], [2] or [27].

When a set of features has been extracted from a sound, the features are usually normalized to some specific numerical scale, for example, the amount of rhythm-ness on a scale from 0 to 10, and then the features are assembled into a feature vector. The next task is usually to decide to which of a set of classes this feature vector most closely belongs. This is known as classification. Clustering, on the other hand, is the automatic creation of a set of classes from a large set of example feature vectors. In the field of clustering, the features are usually referred to as *parameters*, and the feature vector representing a specific datum is called a *case*. In a typical clustering problem, there are a large number of cases to be clustered into a small number of categories.

Clustering algorithms usually make use of *representative cases*. These are cases which represent the clusters of which they are members, and are often chosen as the case closest to the centroid of the cluster. One of the simplest clustering algorithms starts with these representative cases, and when seeking to classify a new case, simply chooses the representative case that is closest to the new case, using some feature-space distance metric. An adaptive version of this algorithm would then choose a new representative case for the cluster, based on which case is now closest to the centroid of the cluster.

Clustering algorithms may have the representative cases pre-determined, or may determine the representative cases in the course of the algorithm. There may be a pre-determined number of clusters, or the algorithm may determine the number of clusters that best segments the parameter space. Also, the features may be chosen beforehand or may be discovered by the algorithm.

5.1 Neural Nets

It is possible to choose the classes beforehand, and allow the algorithm to choose parameters and map out the parameter space. This is the technique used in neural net clustering. The neural net is presented with a set of training cases, each with a corresponding class. The neural net then trains itself to select the correct class when presented with each case, and to be correct for as many of the training cases as possible. When this process is complete, the network is ready to classify new cases.

There are two dangers with this method. First, when a neural network has performed a classification, it is usually not possible to investigate the parameters used to make the classification. If the intent of the research is to discover and understand the parametric differences between a saxophone and a trumpet, this method will not be useful. The other danger is that the set of training vectors must be very carefully constructed. If all of the trumpet cases in the training set happened to be recorded using one microphone and all of the saxophone cases were recorded on a different microphone, it is possible that the network would classify based on parameters of the microphone instead of parameters of the instrument.

The neural net is a computational technique based on a model of a biological neuron, which receives as input a group of electrical impulses, and provides as output an electrical pulse if and only if the combined magnitude of the incoming impulses is above some threshold. Neural networks are groups of these modeled neurons which react to input and provide output. Usually there is an input layer of neurons which accept an incoming parameter vector, one or more hidden layers which do processing, and an output layer that provides a classification.

What makes the neural net powerful is not the neurons themselves, but the connections between them. Each connection has an associated weight, corresponding to how much of the signal from the source neuron is passed to the target neuron. If a neuron receives input pulses from four other neurons, but each connection has weight 0.1, the target neuron will not fire. If, however, the weights are all 0.2, the target neuron will fire, continuing the process.

Neural networks are usually set to a specific task by *training*. In this process, an input vector is presented along with a suggested result. The connection weights are adjusted using some algorithm to ensure that the network makes the proper classification. As more training vectors are used, the network more closely approximates a tool that can do the classification.

Neural networks have been used in classification systems for speech recognition [57], for multimedia databases [19] and, as multi-layer perceptrons, for dialog act classification [46]. They are sometimes passed over for other methods because the training of a neural network can be time-consuming, and it is difficult to tell what is going on inside of the network. There are ways to look at a trained network, analyze the connection weights and get information about the way the data has been clustered, using techniques such as *classification hyperplanes*, where information from the network connection weights are used to deduce the hyperplanes that separate the regions of classification that the network has come to use. Such techniques are often costly and the information produced is not always useful.

5.2 Successive Restriction

Neural networks and most other cluster-based classification techniques are *synchronous*, that is a choice is made between all possible classes in one step. A different way to classify is by *successive restriction*, a technique related to *process of elimination*. In both of these methods, the classification is made over several stages, at each stage one or more classes are removed from the list of possibilities. Successive restriction algorithms are usually designed heuristically for a specific task. In [79], the authors use successive restriction to classify several types of audio. As a first step, they separate silence using their energy and *ZCR* features. If the sound is not silence, they move to the next step which is to separate specific environmental noise sounds, which they then sub-classify as harmonic and changing or harmonic and stable, depending on how much variability is in the noise. If the sound is not silence or noise, they consider music next, looking at harmonicity, f_0, *ZCR* and variance of *ZCR* features. If the sound is not music, they consider speech next, and if it is none of these, the sound is classified as "other environmental sounds" and sub-classified as periodic, harmonic, or non-harmonic. This method is similar to the hierarchical classification discussed earlier.

5.3 *k*-Means Clustering

As a final section on clustering, I present a quick description of a specific clustering algorithm. *k*-means clustering, used in [8], is fairly straightforward, robust, quick to implement and easy to understand, but it will only classify a finite pre-determined number (k) of classes. To get around this, researchers will often set a *k*-means classifier to work with twice or three times as many classes as might be required, and then combine some of the clusters to form the final grouping. For example, if a researcher suspected that there were 3 classes in a case set, she might set a *k*-means classifier to find the best 9 clusters in the set, and then combine some of the clusters to generate the 3 best clusters.

The algorithm begins with the first k points in the case set as starting points for the representative cases for the k classes. As the algorithm considers each new case in sequence, it chooses the k points which are furthest apart, so that when all cases have been considered, the algorithm has the best approximation of the bounding n-dimensional k-gon of the case set, where n is the dimensionality of the parameter space. Figure 7 shows a set of cases in 2-dimensional parameter space. The left frame of the figure shows the k furthest points selected. In this case, there are three clusters to find, so $k = 3$.

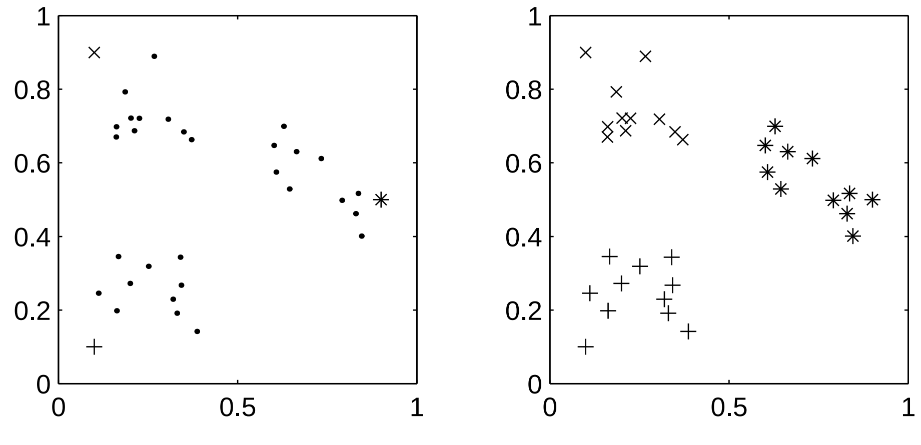

Figure 7: An example of a k-means classification for three classes. In the left frame, the three most disparate cases have been selected as representative and assigned to three separate classes, $\times$,$+$, and $*$. In the right frame, each remaining case has been assigned to the class whose representative case was closest.

Once these k representative classes have been chosen, the algorithm assigns all other cases to one of the k representative cases by a distance measure, to form k clusters, as seen in the right frame of Figure 7. The algorithm can conclude here, or the next step could be to re-assign the k representative classes to the centroid of these newly-formed clusters, and then cluster the remaining cases around *these* representative cases. The process can be repeated a couple of times, but usually after two or three iterations, there is very little improvement in accuracy.

6 Analysis Duration

As was discussed in Section 1.5 on page 8, it is often necessary to segment a signal into windowed frames in order to analyze features that change over time. The analysis of time-varying features is a difficult problem because different frame sizes generate different amounts of information, and some features need different frame sizes than others. This section will discuss the issues involved in analyzing a signal over time.

6.1 Fixed Analysis Frame

The simplest way to make use of the analysis frame is to choose a frame size that is useful for the application, and stick with it. The feature extractors can be optimized for the frame size, and as sequential frames are considered, the signal can be fully examined. The problem with this method is that some features in different time scales are less easily extracted in a single frame size. For example, features that exist over a very small time, like f_0, are measured well in smaller frames, and features that exist over longer periods of time, like rhythm, are better measured using longer frames. This problem is solved either by using multi-resolution analysis, as presented in the next section, or by observing how short-time features change from frame to frame. For example, rhythm can be observed using the same frame size as f_0, by observing the change in total signal energy from frame to frame.

6.2 Multi-resolution Analysis

Some classification algorithms attempt to make decisions about a signal from information contained within a single frame. This creates a classification that changes over time, with very high resolution, but does not take into account the dynamic nature of the perception of sound. The features of f_0 constancy or vibrato, for example, must be measured over several frames to be valid because each frame can have only a single f_0 measurement. One must observe how the f_0 changes from frame to frame to see if it is constant or varying in a vibrato-like way.

Other classification algorithms take into account all frames in a sound before making a decision. This is also not always a good idea, because if the sound contains more than one sequential event, the classification system will attempt to classify the sound as if it were a single event, and conflicting features will lead to incorrect classification.

Multi-resolution analysis is a term that includes all ways of investigating data at more than one frame size, as discussed in Section 1.5 on page 8. One multi-resolution analysis technique that has received much attention in the last five years is wavelet analysis. A wavelet, as the name implies, is a small wave. Wavelets have a frequency and an amplitude, just as waves (sinusoids) do, but they also have a location, which sinusoids do not. In the wavelet transform, the input signal is represented as a sum of wavelets of different frequencies, amplitudes and locations. A set of wavelets is generated from a single mother wavelet, and the different sizes of wavelets are generated by scaling the mother wavelet. It is for this reason that wavelet analysis is multi-resolution. The low frequency wavelets extract information from the signal at a low resolution, and the high frequency wavelets extract high resolution information. The success of this technique comes from the fact that many resolutions can be investigated at one time, using one transform.

6.3 Hidden Markov Models

In many audio classification problems, it is important to do some form of template matching. A set of templates is stored in the computer, the audio signal is compared to each template, and a distance measure is used to determine which template is closest to the input signal. For sounds such as bird calls, impact and percussive sounds, this technique works well because individual instances of the sound are very similar.

On the other hand, many sounds which humans would put into the same category have wide variation. A car honk could last a fraction of a second or go on for minutes. The human voice is the most obvious example of such a situation. When a person articulates a particular phoneme, it could be very brief or it could be drawn out for emphasis. In situations like this, simple template matching fails. Scaled template matching has been tried, where the template is stretched to fit the size of the input signal, but this also stretches the attack and decay portions of the sound, as well as changing the frequency, making it less likely to fit the template. A recording played back faster or slower sounds considerably different, and a human classifier would likely put these in the category of “speed-altered sound” instead of what was in the original recording.

The use of hidden Markov models, or HMMs in ASC is an attempt to rectify this problem [57]. A Markov model is a system with a set of states, and a set of transitions between states (or to the same state). Each transition has an associated probability, and the system proceeds from state to state based on the current state and the probability of transition to a new state. What makes a Markov model *hidden* is the observability of the states. In standard Markov models, the states are directly observable. HMMs have states which are not directly observable, rather there is a set of possible observations from each state, and like the state transitions, the observations from any one state depend on the probabilities of the possible observations.

HMMs are used in signal classification in a “backwards” way. Given a set of observations (the input signal), HMM signal classification attempts to decide which of a set of possible HMMs most likely could generate that set of observations. The classification system contains a number of HMMs, each corresponding to a category, and the class corresponding to the HMM that is most able to produce the incoming signal is the class into which the incoming signal fits.

HMMs work well with sounds that change in duration because the durational change can occur at a single state or across many states. An HMM can begin by following, state for state, the attack of the signal, then jump back and forth in a few middle states for the duration of the sustain portion of the sound, and then follow the decay portion down state for state. The sound is more closely modeled than if a template were stretched to fit the length of the input signal.

7 Conclusion

Audio signal classification (ASC) is a diverse research field encompassing many smaller research topics. Signal processing, feature extraction, statistics, clustering, pattern matching and psychoacoustics all play an important role in an ASC system. In designing any ASC system, one must first decide which specific ASC task will be performed by the system, and then which features will be relevant, how to extract the features, and how to cluster them into a classification. ASC is a vibrant and current research field and there is still much to be done.

References

[1] J. Murray Barbour. *Tuning and Temperament, A Historical Survey.* Michigan State College Press, East Lansing, 1953.

[2] Claudio Becchetti and Lucio Prina Ricotti. *Speech Recognition, Theory and C++ Implementation.* John Wiley & Sons, Toronto, 1999.

[3] Albert S. Bregman. *Auditory Scene Analysis.* MIT Press, Cambridge, 1990.

[4] Meribeth Bunch. *Dynamics of the Singing Voice.* Springer-Verlag, New York, 1982.

[5] Chris Chafe and David Jaffe. Source separation and note identification in polyphonic music. In *icassp*, pages 1289–1292. IEEE, 1996.

[6] Dan Chazan, Stettiner Yoram, and David Malah. Optimal multi-pitch estimation using EM algorithm for co-channel speech separation. In *icassp*, volume II, pages 728–731. IEEE, 1993.

[7] Michael Clarke. Composing at the intersection of time and frequency. *Organised Sound*, 1(2):107–117, 1996.

[8] Brian Clarkson and Alex Pentland. Unsupervised clustering of ambulatory audio and video. In *International Conference on Acoustics, Speech and Signal Processing*, volume VI, pages 3037–3040. IEEE, 1999.

[9] Stanley Coren, Lawrence M. Ward, and James T. Enns. *Sensation and Perception.* Harcourt Brace College Publishers, Toronto, 1994.

[10] Ingrid Daubechies. The wavelet transform, time-frequency localization and signal analysis. *IEEE Transactions on Information Theory*, 36(5):961–1005, 1990.

[11] Ph. Depalle, G. García, and X. Rodet. Tracking of partials for additive sound synthesis using Hidden Markov Models. In *icassp*, volume I, pages 225–228. IEEE, 1993.

[12] Erkan Dorken and S. Hamid Nawab. Improved musical pitch tracking using principal decomposition analysis. In *icassp*, volume II, pages 217–220. IEEE, 1994.

[13] Boris Doval and Xavier Rodet. Estimation of fundamental frequency of musical sound signals. In *icassp*, pages 3657–3660. IEEE, 1991.

[14] Boris Doval and Xavier Rodet. Fundamental frequency estimation and tracking using maximum likelihood harmonic matching and HMMs. In *icassp*, volume I, pages 221–224. IEEE, 1993.

[15] Shlomo Dubnov, Gerard Assayag, and Ran El-Yaniv. Universal classification applied to musical signals. In *ICMC*, pages 333–340, 1998.

[16] Richard O. Duda and Peter E. Hart. *Pattern Classification and Scene Analysis.* John Wiley and Sons, Toronto, 1973.

[17] John M. Eargle. *Music, Sound and Technology.* Van Nostrand Reinhold, Toronto, 1995.

[18] Alexander Ellis and Arthur Mendel. *Studies in the Histroy of Musical Pitch.* Frits Knuf, Amsterdam, 1982.

[19] Bernhard Feiten and Stefan Günzel. Automatic indexing of a sound database using self-organizing neural nets. *Computer Music Journal*, 18(3):53–65, Fall 1994.

[20] Neville H. Fletcher. *Physics and Music.* Hedges and Bell, Ltd., Austrialia, 1976.

[21] Edouard Geoffriois. The multi-lag-window method for robust extended-range f_0 determination. In *Fourth International Conference on Spoken Language Processing*, volume 4, pages 2239–2243, 1996.

[22] David Gerhard. Computer music analysis. Technical Report CMPT TR 97-13, Simon Fraser University, 1997.

[23] David Gerhard. Automatic interval naming using relative pitch. In *Bridges: Mathematical Connections in Art, Music and Science*, pages 37–48, August 1998.

[24] David Gerhard. Audio visualization in phase space. In *Bridges: Mathematical Connections in Art, Music and Science*, pages 137–144, August 1999.

[25] David Gerhard. Audio signal classification: an overview. *Canadian Artificial Intelligence*, 45:4–6, Winter 2000.

[26] David Gerhard and Wiltold Kinsner. Lossy compression of head-and-shoulder images using zerotrees of wavelet coefficients. In *Canadian Conference on Electrical and Computer Engineering*, volume I, pages 433–437, 1996.

[27] Ben Gold and Nelson Morgan. *Speech and Audio Signal Processing.* John Wiley & Sons, Toronto, 2000.

[28] S. Handel. *Listening.* MIT Press, Cambridge, 1989.

[29] Dik J. Hermes. Timing of pitch movements and accentuation of syllables. In *Fourth International Conference on Spoken Language Processing*, volume 3, pages 1197–1200, 1996.

[30] Léonard Janer, Juan José Bonet, and Eduardo Lleida-Solano. Pitch detection and voiced/unvoiced decision algorithm based on wavelet transforms. In *Fourth International Conference on Spoken Language Processing*, volume 2, pages 1209–1212, 1996.

[31] Jean-Claude Junqua and Jean-Paul Haton. *Robustness in Automatic Speech Recognition.* Kluwer Academic Publishers, Boston, 1996.

[32] Christian Kaernbach and Laurent Dernay. Psychophysical evidence against the autocorrelation theory of auditory temporal processing. *jasa*, 104(4):2298–2305, October 1998.

[33] Anssi Kalpuri. Sound onset detection by applying psychoacoustic knowledge. In *icassp*, volume VI, pages 3089–3092. IEEE, 1999.

[34] Matti Karjalainen and Tero Tolonen. Multi-pitch and periodicity analysis model for sound separation and auditory scene analysis. In *icassp*, volume II, pages 929–932. IEEE, 1999.

[35] Haruhiro Katayose. Automatic music transcription. *Denshi Joho Tsushin Gakkai Shi*, 79(3):287–289, 1997.

[36] Benjamin Kedem. Spectral analysis and discrimination by zero-crossings. *Proceedings of the IEEE*, 74(11):1477–1493, November 1986.

[37] Damián Keller and Barry Truax. Ecologically-based granular synthesis. In *ICMC*, pages 117–120, 1998.

[38] Francis Kubala, Tasos Anastasakos, Hubert Jin, Long Mguyen, and Richard Schwartz. Transcribing radio news. In *Fourth Internalional Conference on Spoken Language Processing*, volume 2, pages 598–601, 1996.

[39] William B. Kuhn. A real-time pitch recognition algorithm for music applications. *cmj*, 14(3):60–71, Fall 1990.

[40] Karsten Kumpf and Robin W. King. Automatic accent classification of foreign accented austrialian english speech. In *Fourth Internalional Conference on Spoken Language Processing*, volume 3, pages 1740–1743, 1996.

[41] John E. Lane. Pitch detection using a tunable IIR filter. *cmj*, 14(3):46–57, Fall 1990.

[42] Fred Lerdahl and Ray Jackendoff. *A Generative Theory of Tonal Music.* MIT Press, Cambridge, 1983.

[43] Llewelyn S. Lloyd and Hugh Boyle. *Intervals, Scales and Temperaments.* MacDonald, London, 1963.

[44] Michael A. Lund and Lee. C. C. A robust sequential test for text-independent speaker verification. *Journal of the Acoustical Society of America*, 99(1):609–620, January 1996.

[45] Gerald M. Mackay. *Global Scales.* Mackay, 1996.

[46] M. Mast, R. Kompe, S. Harbeck, A. Kießling, H. Niemann, Nöth, E.G. Schukat-Talamazzini, and V. Warnke. Dialog act classification with the help of prosody. In *Fourth Internalional Conference on Spoken Language Processing*, volume 3, pages 1732–1735, 1996.

[47] Brian C. M. Moore, editor. *Hearing.* Academic Press, Toronto, 1995.

[48] James A. Moorer. On the transcription of musical sound by computer. *Computer Music Journal*, pages 32–38, November 1977.

[49] Yasuyuki Nakajima, Yang Lu, Masaru Sugano, Akio Yoneyama, Hiromasa Yanagihara, and Akira Kurematsu. A fast audio classification from MPEG coded data. In *International Conference on Acoustics, Speech and Signal Processing*, volume VI, pages 3005–3008. IEEE, 1999.

[50] Harry F. Olson. *Music, Physics and Engineering.* Dover Publications, New York, 1967.

[51] Martin Piszczalski. *A Computational Model for Music Transcription.* PhD thesis, University of Stanford, 1986.

[52] Martin Piszczalski and Bernard A. Galler. Predicting musical pitch from component frequency ratios. *jasa*, 66(3):710–720, September 1979.

[53] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipies in C.* Cambridge University Press, Cambridge, 1992.

[54] Donald F. Proctor. *Breathing, Speech and Song.* Springer-Verlag, New York, 1980.

[55] Francisco J. Quirós and Pablo F-C Enríquez. Real-time, loose-harmonic matching fundamental frequency estimation for musical signals. In *icassp*, volume I, pages 221–224. IEEE, 1994.

[56] Lawerence R. Rabiner, Michael J. Cheng, Aaron E. Rosenberg, and Carol A. McGonegal. A comparative performance study of several pitch detection algorithms. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 24(5):399–418, October 1976.

[57] Lawrence Rabiner and Biing-Hwang Juang. *Fundamentals of Speech Recognition.* Prentice Hall, New Jersey, 1993.

[58] Rao G. V. Ramanaz and J. Srichand. Word boundary detection using pitch variations. In *Fourth International Conference on Spoken Language Processing*, volume 2, pages 813–816, 1996.

[59] Curtis Roads. *The Computer Music Tutorial.* MIT Press, Cambridge, 1996.

[60] Juan G. Roeder. *The Physics and Psychophysics of Music.* Springer-Verlag, New York, 1995.

[61] S. Rossingnol, X. Rodet, J. Soumagne, J.-L. Collette, and P. Depalle. Features extraction and temporal segmentation of acoustic signals. In *ICMC*, pages 199–202, 1998.

[62] Hajime Sano and B. Keith Jenkins. A neural network model for pitch perception. *cmj*, 13(3):41–48, Fall 1989.

[63] John Saunders. Real-time discrimination of broadcast speech/music. In *International Conference on Acoustics, Speech and Signal Processing*, pages 993–996. IEEE, 1996.

[64] Eric Scheirer. *Music Perception Systems (Proposal).* PhD thesis, Massachusetts Institute of Technology, 1998.

[65] Eric Scheirer and Malcolm Slaney. Construction and evaluation of a robust multifeature speech/music discriminator. In *International Conference on Acoustics, Speech and Signal Processing*, volume II, pages 1331–1334. IEEE, 1997.

[66] Manfred Schroeder. *Fractals, Choas, Power Laws.* W. H. Freeman and Company, New York, 1991.

[67] Jürgen Schürmann. *Pattern Classification.* John Wiley and Sons, Toronto, 1996.

[68] Carl E. Seashore. *Psychology of the Vibrato in Voice and Instrument.* The University Press, Iowa, 1936.

[69] J. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *IEEE Transactions on Signal Processing*, 41(12):3445–3462, 1993.

[70] Leslie S. Smith. Sound segmentation using onsets and offsets. *Journal of New Music Research*, 23:11–23, 1994.

[71] Michelle S. Spina and Victor W. Zue. Automatic transcription of general audio data: Preliminary analyses. In *Fourth Internalional Conference on Spoken Language Processing*, volume 2, pages 594–597, 1996.

[72] Mark Steedman. The well-tempered computer. *Phil. Trans. R. Soc. Lond. A.*, 349:115–131, 1994.

[73] V. S. Subrahmanian. *Multimedia Database Systems.* Morgan Kaufmann Publishers, Inc., San Francisco, 1998.

[74] Barry Truax, editor. *Handbook for Acoustic Ecology.* A.R.C. Publications, Vancouver, 1978.

[75] Luc M. Van Immerseel and Jean-Pierre Martens. Pitch and voiced/unvoiced determination with auditory model. *jasa*, 91(6):3511–3526, June 1992.

[76] Rivarol Vergin, Azarshid Farhat, and Douglas O'Shaughnessy. Robust gender-dependant acoustic-phonetic modelling in continuous speech recogntion based on a new automatic male/female classification. In *Fourth Internalional Conference on Spoken Language Processing*, volume 2, pages 1081–1084, 1996.

[77] Erling Wold, Thom Blum, Douglas Keislar, and James Wheaton. Content-based classification, search and retrieval of audio. *IEEE MultiMedia*, pages 27–37, Fall 1996.

[78] Bin Yang. Content based search in multimedia databases. Master's thesis, Simon Fraser University, June 1995.

[79] Tong Zhang and Jay C.-C. Kuo. Hierarchical classification of audio data for acrhiving and retrieving. In *International Conference on Acoustics, Speech and Signal Processing*, volume VI, pages 3001–3004. IEEE, 1999.