Singing Phoneme Class Detection In Polyphonic Music Recordings

Vagia Ourania

Supervisor:Perfecto Herrera

Universitat Pompeu Fabra

A thesis submitted for the degree of

Master in Information Technologies, Communication and Audiovisual Media
Department of Information and Communication Technologies
Music Technology Group
Barcelona, September 2008

in memory of my father,

Acknowledgements

I would like to give special thanks to my supervisor, Perfecto Herrera, for his help and guidance on doing research. Other persons of great importance during this Master have been Xavier Serra that accepted me to join the Music Technology Group of the UPF and all the members of the group. Especially, I would like to thank Ferdi, Cyril and my office mates Elena, Paqui, Martin, Vassilis, Uri, Marcelo, Justin and Gerard for their help.

Abstract

Automatic singing detection and singing phoneme recognition are two MIR research topics that have gained a lot of attention the last years. The first approaches borrowed successful techniques widely used in Automatic Speech Recognition (ASR) as speech and singing share similar acoustical features since they are produced by the same apparatus. Moving from monophonic to polyphonic audio signals the problem becomes more complex as the background instrumental accompaniment is regarded as a noise source that has to be attenuated.

This thesis presents research into the problem of singing phoneme detection in polyphonic audio, in which the lyrics are in English. Specifically, we are interested in building statistical classification models that are able to automatically distinguish sung consonants and vowels from pure instrumental music in polyphonic music recordings.

The approach begins with a database creation to be used for training, testing and evaluating the models. Several sets of extracted low-level features are used in the classification process. Different classification functions are compared like SVM, MLP and logistic as well as different classification schemes (3-class classifiers, binary classifiers in series and in parallel). The best classification model found reaches an overall accuracy of 78% in distinguishing between the 3 different classes.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

General audio consists of a wide range of sounds such as speech, music, singing and environmental sounds produced by very different sources. Sound classification for humans seems to be an easy task. We can easily distinguish sounds as a door knocking, a woman singing or rain falling, even if we don't often consider of how we can do it and we don't put any conscious effort.

Automatic sound recognition, using computers, is a popular research task for more than fifty years. Several topics, ranging from Automatic Speech Recognition (ASR), pitch detection, speech and music discrimination, instrument and genre classification, speech and singing voice discrimination, singing voice detection in polyphonic audio and many more have been under research. Several techniques and tools have been implemented and applied in order to analyze and recognize such diverse audio content, especially threshold methods and statistical classifiers.

In this thesis, we are interested in the singing voice in polyphonic recordings of popular music. We address the problem of processing a music audio file and segmenting it into fragments containing vocals (with instrumental background) and pure instrumental music. We extend this classification scheme and segment further the singing voice segments into sung vowel and sung consonant sounds. The procedure is based on extracting audio features from short frames of the audio files and then classifying each frame using statistical classifiers.

The remainder of this chapter presents motivation for the research presented in this thesis, application areas of the work done, the goals of this Master thesis as well as the outline of the following chapters.

1.1 Motivation

Although ASR has reached a great level of robustness, singing phoneme recognition is a new research topic with a great deal of challenges and possible applications. Taking inspiration from techniques widely used in ASR, researchers try to adapt them to the case of the singing voice despite their wide range of differences (2.3). Many approaches integrate higher knowledge of available lyrics, song structure, mixing techniques etc. in order to reduce the complexity of the task.

For the case of polyphonic popular music recordings, which is examined in this thesis, singing phoneme recognition is not a trivial task. The background music is in this case considered as a noise source that can cause bad performance to models trained with pure singing voice signals. A lot of research attention is given currently in order to solve Source Separation problems, like the classical example of "the cocktail party problem". In this thesis, the objective is not to perform source separation in order to separate the singing voice from the instrumental background. The intention is to examine if classification models are capable of distinguishing singing vowel and singing consonant sounds from pure instrumental music, without performing any kind of prior processing to the audio files and without using any higher-level knowledge. Machine Learning techniques will be used in order to handle the topic addressed.

1.2 Application Areas

During the last decade, with the release of MP3 encoder and the peer-to-peer file sharing tools, people store in their PCs a great amount of audio files. Many applications are being built nowadays for integrating all possible representations for audio files, like lyrics, midi representation, score and video in order to give to the listener more possible applications and flexibility to the use of them. Listeners enjoy while listening to a song to have the corresponding lyrics displayed. This is still done by searching the Web and locating the lyrics of the song in question. Then, the user scrolls down the lyrics while listening to the song.

Furthermore, in applications such as Karaoke, which are based on lyrics to audio alignment, the annotation is still done manually. A great deal of the approaches in Singing Voice Detection and Singing Voice Recognition is oriented towards the development of automatic lyrics transcription or alignment systems for Karaoke applications (Gruhne *et al.*, 2007; Iskandar *et al.*, 2006; Kan *et al.*, 2008; Shenoy *et al.*, 2005; Wong *et al.*, 2007; Zhu *et al.*, 2005). Starting using techniques from Automatic Speech Recognition, singing voice transcription was proved to be a great challenge, since singing voice is different from normal speech in many ways and instrumental accompaniment makes the problem even harder. The problem soon turned to lyrics to singing voice alignment, as almost all songs' lyrics are available on the Web. Automatic alignment between singing voice and text refers to the temporal relationship between audio signals and the corresponding lyrics. The goal of the aligner is to precise automatically the starting and ending points of each phoneme, word or sentence (entity) in real-world polyphonic music audio signals, given the lyrics.

As lyrics and singing voice are different representations of an audio file in many music genres, binding efficiently these representations can help in many research topics in the area of Music Information Retrieval. Some of them can be, for example, a lyrics-based artist/song similarity system (Li and Ogihara, 2006; Li *et al.*, 2006; Logan *et al.*, 2004; Mahedero *et al.*, 2005) a query by lyrics music search engine (Muller *et al.*, 2007; Suzuki *et al.*, 2007), an automatic language identification system (Mahedero *et al.*, 2005) or a system that performs semantic analysis of the words sung and finds emotions and mood of the listeners based on the lyrics (Laurier *et al.*, 2008).

Another possible application of these systems could be the search for certain words within a song. If the starting and ending points of every word were known, as well as the word sung (from the lyrics), a database could be built containing annotated segments of audio. A user could search for a specific word and get all the segments of the songs in the database containing that word.

In addition, lyrics-based audio retrieval and navigation in music collections is another possible application. Semantic analysis of song lyrics, artist similarity based on lyrics, emotion detection and language identification can also be helped by the aligner. One of the most significant applications of such a system is that it will make available a big database of singing phonemes and words. These fragments can be used as synthesis units for systems such as Vocaloid [1]. These units can be used for further transformations and morphing, using signal processing techniques. Within the same concept, an Audio Mosaicing (Janer and de Boer, 2008; Lazier and Cook, 2003) system, could take advantage of these singing voice segments and add singing voice to the audio created by the Audio Mosaicing system.

1.3 Hypotheses, Goals of this Thesis

When the research on this thesis started, the topic was balancing between voiced vs. unvoiced and vowel vs. consonant singing phoneme discrimination. While experimenting and reviewing available literature, we found that the vowel-consonant case is more general. The reason for this is that phonemes are clearly separable in these categories, while for the case of voiced/unvoiced the separation depends on the language. For example, the /h/ phoneme, in all the languages it exists is a consonant, but its assignment to the voiced or unvoiced phoneme class depends on the language. Thus, this thesis is focused on the discrimination between sung vowel and sung consonant phonemes. Finally, for sake of completeness of the system a class for pure instrumental music -without any vocals- was added turning the problem to a three-class classification one. Thus, all the parts of the audio can be assigned to one of the three classes, without leaving out parts of the sound that don't contain vocals. Also, having this classification scheme further refinements and extensions are feasible in a second level, i.e. classification between different vowels and consonants. One last reason for researching this topic is that many approaches to the singing voice to lyrics alignment, trying to enhance the singing voice over the instrumental background (accompaniment sound reduction), omit the consonants that carry most of the information in speech (Owens *et al.*, 1968). This study had the following objectives in mind:

- Investigate if suitable segmentation, feature extraction and statistical modelling are capable of distinguishing sung vowels from sung consonants and

[1] http://www.vocaloid.com

pure instrumental music, without performing any kind of pre-processing of the audio signals or using any high-level or song-specific knowledge i.e. song structure, lyrics, recording techniques etc. Such an approach, apart from being more general, and thus widely applicable, is also more computationally efficient compared to other approaches.

- Use Western music with lyrics in English to test our hypothesis
- Not interested in distinguishing between different consonants or different vowels. Focus on vowel/ consonant/ music discrimination, as a more general and generic case.
- Investigate the accuracy of a statistical model to distinguish consonants from music segments with percussion. Percussion sounds resemble a lot consonant phonemes and even the human ear is difficult to recognize an -alone- consonant with musical background.
- Develop a more detailed discrimination of music and singing voice.

1.4 Outline

Concerning the structure of this document, it is divided in 6 chapters.

- In chapter 2 we present some necessary terms and procedures in order to introduce the reader to the subject and describe the theoretical framework of this study.
- In chapter 3 we review state of the art techniques used in singing voice detection and singing phoneme recognition and alignment.
- In chapter 4 we describe the methodology used in this study.
- In chapter 5 we present the experiments performed during this thesis along with the results and motivation behind them.
- Chapter 6 finishes off with a discussion on the conclusions and suggestions for future work.

Chapter 2

Scientific Background

2.1 MIR

Music Information Retrieval (MIR) is an emerging multidisciplinary and interdisciplinary research area which appeared in the late nineties. It encompasses Musicology and Music Theory, Computer Science, Information Retrieval, Engineering, Signal Processing, Cognitive Science and Psychology. The term MIR encompasses a number of different research and development activities that have the common denominator of being related to music access. Despite its name, MIR is not only about retrieving information from music but to fulfil users' music information, amusement or training needs. And as these needs are more aimed at music retrieval that music information retrieval, so are the consequent approaches. Also, the term "retrieval" has a broader sense since it encompasses tasks such as filtering, classification, identification, indexing and visualization that become increasingly useful for the final users (Orio, 2006).

Most of the research works on MIR, of the proposed techniques, and of the developed systems are content-based. The main idea underlying content-based approaches is that a document can be described by a set of features that are directly computed from its content (Orio, 2006). In the case of MIR, content it is the implicit and explicit information that is related to a sound or a piece of music and that is embedded in the signal itself. The methodologies of MIR are based on Information Retrieval, thus techniques of statistics and probability theory are used to describe underlying models.

The first step towards music retrieval involves the automatic processing of the audio signals in order to extract meaningful content descriptors. There are different approaches to music processing, depending on the form and format in which musical documents are instantiated and on the dimensions of interest. As it can be expected, great part of the research on feature extraction has been devoted to the audio form, from which most of the music dimensions are particularly challenging to extract (Orio, 2006). In MIR these descriptors are usually divided in high-level or user-centered (rhythmic patterns, tonality, key etc.), mid-level or object-centered (spectral envelop, beat, dynamic range, etc.) and low-level or signal-centered(fundamental frequency, onset, note duration, spectral centroid, spectral flux, etc.) (Serra, 2008). More about these descriptors will be presented in section 2.7. Some of the topics MIR includes are [1]:

- musical feature extraction for monophonic and polyphonic audio,
- computational methods for classification, clustering, and modelling, similarity and pattern matching,
- music identification and recognition, such as score following, automatic accompaniment,
- filtering for music and music queries, query languages, standards and other metadata or protocols for music information handling and retrieval, multiagent systems, distributed search,
- software for music information retrieval, human-computer interaction and interfaces, mobile applications, user behavior,
- music perception, cognition, affect and emotions,
- music similarity metrics, syntactical parameters, semantic parameters, musical forms, structures, styles and genres,
- music annotation methodologies,

[1] http://en.wikipedia.org/wiki/Music_information_retrieval

- music analysis and knowledge representation, automatic summarization, citing, excerpting, downgrading, transformation, formal models of music, digital scores and representations,
- music indexing and metadata,
- music archives, libraries, and digital collections,
- intellectual property rights, national and international intellectual property right issues, digital rights management, identification and traceability,
- sociology and economy of music,
- user profiling, validation, user needs and expectations, evaluation of music IR systems, building test collections, experimental design and metrics.

2.2 The Human Voice

The human vocal organ can produce several types of sounds like speech, laughing, whispering, singing etc. When we speak, we create a disturbance in the air around us, a small but rapid variation in air pressure. During this sound, the air pressure at the speaker's lips fluctuates and a corresponding wave is generated. This sound wave reaches the listener's ear drum and causes small movements which are sensed by the brain and interpreted as a specific sound, with particular pitch and loudness. The pitch of the sound depends on the rate of repetition of the sound wave. The loudness depends on the size of the variations in air pressure. Differences in amplitude are measured in dB. The third way in which sounds can differ is in quality, often called timbre.

In English, like in most European languages the meaning of a word remains the same irrespectively of the pitch. However, pitch changes in other languages like Mandarin can change the meaning of a word. In English, pitch changes are used in a different way. In that case, the meaning of a group of words can change and this difference in pitch is then called a difference in intonation. Singing differs from speaking in that when you sing the pitch of the voice has to remain constant, usually for one or two syllables and then jump to the next note. In speech the pitch is always changing, even within a single syllable. Pitch changes can also convey different kinds of information. For example, we could say if the speaker is angry or happy just by listening to the tune without even listening to the words. The pitch of the voice carries much of the emotional content of the speech.

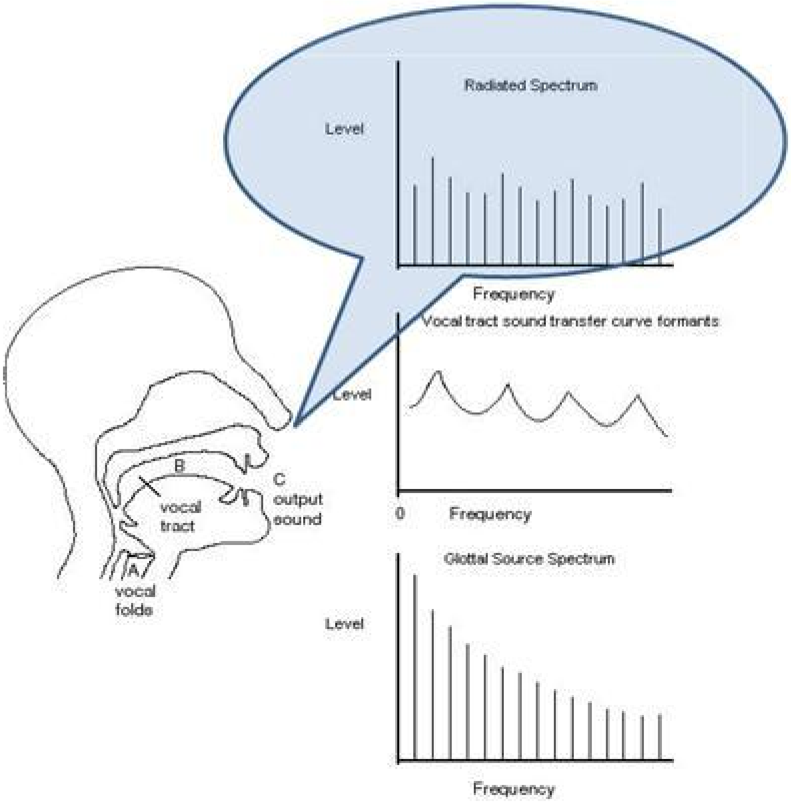
The voice organ is composed of three systems, the breathing apparatus, the vocal folds and the vocal tract. The breathing system compresses the air in the lungs and the generated airstream passes through the glottis (the slit between the vocal folds) and the vocal tract. In terms of activity the vibration of the vocal folds is responsible for the phonation, the generation of a primary sound, as the airstream passes through them. This voice source is then shaped acoustically when passes through the vocal tract. This shaping depends on the vocal tract configuration, which is controlled by articulation. In terms of functioning the breathing system acts like a compressor, the vocal folds as an oscillator and the vocal tract as a resonator (Sundberg, 1987).

Since the vocal folds open and shut the glottis at identical time intervals, a tone is generated which poses a certain frequency, the vibration frequency of the vocal folds. The vocal folds are not the only oscillator in the voice organ. Other parts of the voice organ can work as oscillators to produce unvoiced sounds. When the airstream from the lungs is forced to pass through a narrow slit with reasonable rigid walls, noise is generated, a signal with non-periodic or irregular variations.

The air enclosed in the vocal tract acts as a resonator. The main characteristics of a resonator is that sound within it decays slowly and that it allows sounds with specific frequency to pass through it. In the human vocal tract these especially transmitted frequencies that fit the resonator optimally are called formant frequencies. Thus the ability of the vocal tract to transmit sounds is greatest at the formant frequencies and tones in other than these frequencies are transmitted with a reduced amplitude. The vocal tract possesses four or five resonance frequencies. The two lowest formants determine most of the vowel colour and all of them are significant for the voice timbre and for distinguishing between vowels.

During phonation, the vibrating vocal folds do not produce a single tone, but an entire spectrum of tones. The lowest tone in the spectrum is called fundamental frequency and the other tones are called overtones. The fundamental plus the overtones are called partials. Their frequencies form an harmonic series, which means that the frequencies of the partials are multiple integers of the fundamental frequency. As these partials pass through the vocal tract, they are treated differently as they have different frequencies. The partials that are closer to a formant frequency are radiated with a higher amplitude than other partials. These formant frequencies are determined by the shape of the vocal tract and determine the vowel quality and voice colour.

Figure 2.1: Illustration of the generation of voice sounds (reproduction of Figure 2.10, (Sundberg, 1987, p.20))



2.3 Singing Voice vs Speech

Although speech and singing voice sounds have many properties in common because they originate from the same apparatus, there are several differences to bear in mind (Gerhard, 2003; Kim, 2001; Loscos *et al.*, 1999; Michael W. Macon, 1997; Sundberg, 1987):

- Duration of voiced sounds: In English, speech consists of approximately 60% voiced sounds and 40% unvoiced sounds, while in singing, the percentage of phonation time can increase up to 95% in the case of opera music . In the most common classical singing technique, known as bel canto, singers are taught that vowel sounds should be held as long as possible between consonants because they carry the melody. Also sonorant consonants may have very large durations.

- Loudness: The dynamic range as well as the average loudness is greater in singing than in speech. The spectral characteristics of a voiced sound change with the loudness. Especially, as loudness is increased, the higher spectrum overtones gain more amplitude than the lower ones.

- Pitch: Speech utterances sometimes have a monotone pitch, but often have a pitch track varying across syllables and indicating speaker's intent or other prosodic characteristics, expresses the emotional state of the speaker or add intelligibility to the spoken words. This frequency range of f0 is very small compared to singing where it can be up to three octaves. Sung utterances have a noticeable melody, and target pitches adhere to some form of musical scale. The melody is followed in precise and discrete steps over customary musical intervals, which commonly are not smaller than semitones in Western music, though quarter and eighth tones are frequently used in Oriental and African music.

- Vibrato: Two types of vibrato exist in singing. The classical vibrato in opera music corresponds to periodic modulation of the phonation frequency, and in popular music the vibrato implies an added amplitude modulation (tremolo) of the voice source. In speech, no vibrato exists.

- Formants: As in singing, the musical quality of the voice is more critical than the intelligibility of the lyrics, in cases like high pitch singing, wide excursion vibratos, hoarse and aggressive attacks or very loud singing, there is an alteration of the formants position, and therefore "syllable identification is more or less a guesswork" (Sundberg, 1987, p. 176).

- Rhythm: Is a feature that many listeners indicate as evidence for a sound being a song. The rhythm of the fixed tonal steps follows the pattern prescribed by the composer and long notes may be sustained for special effects.
- Rhyme: Usually indicative of poetry or singing, is another difference with normal speech. Rhyme is a higher-level repetition than rhythm, taking phonetic information into account.

2.4 Phones, Phonemes and Allophones

Phonemes are the smallest units of speech that distinguish meaning of a language and most languages have 50 or fewer phonemes. Phonemes are not the physical segments themselves, but, in theoretical terms, cognitive abstractions or categorizations of them. A phoneme may encompass several recognizably different speech sounds, called phones. Phones that belong to the same phoneme are called allophones and they are considered equivalent for a given language. Allophones are the linguistically non-significant variants of each phoneme or the multifarious "physical" realizations of a given phoneme abstract category. In English for example, [th] and [t] are allophones of the phoneme /t/, as in tip and stand. The first consonant of tip is aspirated, while in stand is not. Switching allophones of the same phoneme doesn't change the meaning of the word. Their difference may not even be audible to native speakers. However, allophones of a phoneme are not the same for every language. For example, [p] and [ph] belong to the same phoneme in English, but to different phonemes in Chinese, thus the meaning of the word changes.

A broad transcription uses only one symbol for all allophones of the same phoneme. This is enough information to distinguish a word from other words of the language. What details you have to include in a broad transcription will depend on what language or dialect you are transcribing. In English, there are about 42 phonemes.

English phonemes can be distinguished in two main categories: voiced and unvoiced. Most languages have only voiced vowels, e.g. /a/, /e/, /i/, /o/, /u/. The voiced consonants are /l/, /r/, /j/, /w/, /m/, /n/, /N/, /b/, /d/, /g/, /v/, /D/, /z/, /Z/ and /dz/ while the unvoiced consonants are /p/, /t/, /k/, /T/, /f/, /S/, /s/ and /ts/, as described in detail in the following section. Because of their periodic nature, voiced sounds are often easier to detect and analyze.

As phonemes are language-dependent, their identification in different languages requires different phoneme detectors. In today's speech recognition systems usually higher-level knowledge is used since they require word and syntax-level knowledge to identify a word from the sound. Robust phoneme recognition can contribute to systems, which need only low-level acoustic features for the task.

2.5 Sounds of Consonants and Vowels

This section is a summary of the main characteristics of English vowels and consonants as presented in Ladefoged (2005). This book has been one of the main sources for our manual annotations (4.2).

There are about 200 different vowels in the world's languages and more than 600 different consonants. In General American English there are 14 or 15 different vowels, while in the form of British English used by national newscasters there are 20 different vowels.

Vowels are sounds produced without any kind of obstruction of the outgoing breath. Different vowels are like different instruments. Although a clarinet, a piano or a violin can play the same tone-same fundamental frequency, we can distinguish them from the smaller variations within each repetition of the sound wave -their overtones. Within the same concept, vowels will retain their quality irrespectively of the pitch produced by the vocal folds. In human voice these overtones are the formants, the resonances of the vocal tract. Trying to listen to the separate formants of a vowel is difficult because we are very used in considering each vowel as a single meaningful entity.

The vowels of a speaker can be described by the frequencies of their formants. These frequencies may differ between speakers, as they depend on the resonating cavities of each person. In order to represent the vowels of a language a relative value of the formants' values is needed. Vowels can be described sufficiently by the values of their first three formants.

Sounds that have some obstruction to the breath stream, such as the bringing of the lips together, are called consonants. The differences in consonants between British and American English are only minor. The voiced and voiceless consonants in English are divided into stops (or plosives), approximants, nasals, fricatives and affricatives.

The phonemes /b/, /d/, /g/ are stops that constitute just ways of beginning or ending vowels. They are called stops because the air in the vocal track is completely stopped either by the lips or the tongue. As the stop closure is being formed or is opening, the shape of the vocal tract is changing and the formant frequencies are moving. For all these three sounds the frequency of the first formant increases when they are at the beginning of a syllable and falls if they are at the end. The movements of the other two formants distinguish these sounds from one another. Another set of stop consonants in English is /p/, /t/, /k/ that are made with the same gestures as /b/, /d/, /g/. The difference between these two sets is in the vibration of the vocal folds. In words beginning with /b/,/d/,/g/ the vocal folds are vibrating while the lips or tongue are moving apart, while for /p/, /t/, /k/ the vocal folds are apart at the beginning of the movement and there is a burst of air that produces a different kind of sound. As these sounds are not produced by the action of the vocal folds, these sounds are noisier with a less well-defined pitch. These voiceless stops may have similar formant movements to their voiced correspondent after a vowel, but at the beginning of a word they are distinguished by the frequencies of the bursts of noise, produced as the stop closure is released.

Another set of consonants constitutes of /w/, /j/, /l/, /r/ like in the words wet, yet, let, rest and are called approximants. They are opposite of stop consonants in that they do not involve any kind of closure of the vocal tract. Instead there is simply a narrowing at some point caused by the lips, the teeth or the tongue. The approximants have their own formant patterns.

The next set of sounds is called nasals because they involve sound radiated while air comes out of the nose. The nasal sounds are /m/,/n/, /N/ like in the words ram, ran, and rang. They are like vowels and approximants in that they can be characterized largely in terms of their formant frequencies, but they differ in that the formants are not as loud as they are in vowels. The nasals are made by blocking the sound from coming out of the mouth while allowing it to come out through the nose, and this affects the relative amplitude of the formants. They have first formants with a very low frequency, around 200 Hz, another visible formant around 2500Hz and little energy in the region normally occupied by the second formant. They have similar formant movements to the corresponding stops (/b/, /d/, /g/).

Another class of consonants comprises the fricatives. The voiceless fricatives are /f/, /T/, /s/ and /S/ as in words fie, thigh, sigh and shy respectively. In these sounds the vocal folds are held apart so that they do not vibrate. The noise is made by air being forced through a narrow gap. Instead of formants, their most prominent acoustic features are high-pitched more random frequencies. They are called fricatives to indicate that the noise is produced by the friction, the resistance to the air, as it rushes through a narrow gap. Another fricative that is special from the view that its source is not air being forced through a narrow gap is /h/ as in high. The origin of the sound is the turbulence -the random variations in air pressure- caused by the movement of the air across the edges of the open vocal folds and other surfaces of the vocal tract. Each of the voiceless fricatives /f/, /T/, /s/ and /S/ has their voice counterpart /v/, /D/, /z/, /Z/ like in voice, this, zoo and pleasure respectively. Voiced fricatives have formants produced by pulses from the vocal folds as well as more random energy produced by forcing air through a narrow gap.

The last class of sounds to be considered is /ts/ and /dz/like in the words church and judge. Affricatives are not really single sounds but two, a /t/ followed by an /S/ and a /d/ followed by a /Z/ respectively. This combination of a stop followed by a fricative is called an affricative. Both the voiceless /S/ and the voiced /Z/ are considerably shorter than when they occur on their own.

2.6 Speech Recognition Techniques

Signal recognition practice has a set of techniques that are applied commonly, independently of the type of the signal (speech, singing voice, biological, economic etc). Here we are going to refer to common practice in speech recognition even if our target signals are not speaking voice signals but singing voice ones. These techniques are also widely used also in the case we are examining. There are four levels of attempts to increase robustness in speech-based pattern recognizers, as presented in (Pool, 2002).

- Noise removal in order to enhance the speech signal and improve its quality. The background interference causes degradation of the intelligibility of speech, which leads to bad recognition performance. The goal of speech enhancement algorithms is to reduce the interfering background noise, which is added to the speech signals. Several measures, categorized in objective and perceptual are used to evaluate speech enhancement algorithms, like signal-to-noise ratio (SNR) and intelligibility respectively. The choice of the suitable measure depends on the application. Speech enhancement techniques can be divided in those based on stochastic processing of speech models and those based on perceptual aspects of speech.

- Feature Extraction, using the correct feature representation, which usually varies in the several situations. Feature extraction or selection refers to the representation of the speech waveform with a suitable set of characteristics, derived from its waveform. This is based on the fact that features of speech change slowly within frames of a few milliseconds, in contrast with its waveform. Thus available signal information is compressed and represented in a more effective space. The goal is to select the more representative features, so as to keep the more relevant information for the classification task, depending on the application and the available signal.

- Models and classifiers, used for training and testing. One of the most important parts of the speech recognizer is the model it uses to capture and represent the characteristics of the signal. Probabilistic models are widely used in speech recognition, which use probability distributions to represent the speakers and output probabilities that are used for the classification. They are divided in parametric and non-parametric based on the parameters of the probability density function. Non-parametric models are Nearest Neighbor and Vector Quantization (VQ) models, while parametric models are for example Gaussian and Gaussian Mixture Models (GMMs). The last is one of the most popular speaker models and has been widely used for speaker identification.

- Process of the classifier likelihood scores in such a way so as to prune errors and improve the performance of the recognition systems (widely known as error-pruning techniques). The recognizer's model outputs for each frame likelihoods for each class. According to these probabilities one frame is assigned to one of the classes. As there is often the case to get outliers in the classification task, error-pruning techniques and rules are applied so as to eliminate some of the errors.

These steps of speech recognition systems are adapted and implemented for the case of singing phoneme recognition with background music. In polyphonic music recordings, the instrumental interference is treated as the noise source that causes degradation to the intelligibility of the singing voice signal. Singing voice enhancement techniques are implemented in many approaches in order either to extract the singing voice signal or to attenuate instrumental sounds. The following steps are the same for both speech recognition and singing voice recognition. Feature extraction and statistical modelling are performed in every singing voice recognition system. Last, error-pruning techniques are implemented so as to improve the models' performance. In Chapter 3 we review several approaches implemented in every of these steps in different singing voice recognition systems.

2.7 Audio Signal Analysis

An audio signal can be represented in both time-domain (waveform) and frequency-domain (spectrogram). These representations are complementary and their use depends on the type of information one wants to extract. Some features can be estimated in both domains, while others only in one. Feature extraction is performed in blocks of samples, commonly referred as frames. The length of the frame, as well as the type of the window and their overlapping factor are important configuration parameters that depend usually on the type of the signal and the application. Below we are presenting briefly some of the most common audio features. A complete list of audio features and their description is listed in (Peeters, 2003).

2.7.1 Time-domain Features

Autocorrelation is used for finding repeating patterns in a signal. It is defined as the cross-correlation of a signal with a time-shifted version of itself and expresses the amount of their similarity.

$$\hat{r}_x(l) = \frac{1}{N}\sum_{n=0}^{N-1} \bar{x}(n)x(n+l) \tag{2.1}$$

where N denotes the number of samples in the window.

Root-Mean Square (RMS) describes basically the global energy of the signal by taking the root average of the square of the amplitude.

$$x_{rms} = \sqrt{\frac{1}{N}\sum_{n=0}^{N-1} x_n^2} \tag{2.2}$$

Zero-crossing rate (ZCR) is the rate of sign-changes along a signal. It is used to measure the noisiness or brightness of a signal since noisy sounds tend to have high ZCR. For monophonic tonal signals, the zero-crossing rate can be used as a primitive pitch detection algorithm.

Onset detection This technique is a way of determining the tempo and is basically the computation of a curve, showing the successive bursts of energy. Then, a pick-peaking algorithm detects the estimated positions.

2.7.2 Frequency-domain Features

Spectral centroid is the barycenter of the spectrum. Perceptually, spectral centroid corresponds to the degree of brightness of a sound. It is computed considering the spectrum as a distribution which values are the frequencies and the probabilities to observe these frequencies are the normalized amplitude:

$$\mu = \int xp(x)\delta x \tag{2.3}$$

where:

$$x = freq(v(x)) \tag{2.4}$$

$$p(x) = \frac{ampl(v(x))}{\sum_x ampl(v(x))} \tag{2.5}$$

Spectral spread is the variance of the above defined distribution, or else the spread of the spectrum around its mean value:

$$\sigma^2 = \int (x-\mu)^2 p(x)\delta x \tag{2.6}$$

The **spectral skewness** gives a measure of the asymmetry of a distribution around its mean value. It is computed from the 3rd order momentum divided by the standard deviation raised to the third power:

$$m_3 = \int (x-\mu)^3 p(x)\delta x, \gamma_1 = \frac{m_3}{\sigma^3} \tag{2.7}$$

The **spectral kurtosis** gives a measure of the flatness of a distribution around its mean value. It is computed form the 4th order momentum divided by the square of the variance:

$$m_4 = \int (x-\mu)^4 p(x)\delta x, \gamma_2 = \frac{m_4}{\sigma^4} \tag{2.8}$$

The **spectral roll-off** is defined as the frequency below which a certain fraction of the total energy is concentrated. Usually this fraction is fixed to 0.95. It is a way to estimate the amount of high frequency in the signal.

$$\sum_0^{f_c} \alpha^2(f) = 0.95 \sum_0^{sr/2} \alpha^2(f) \tag{2.9}$$

where α the amplitude, f_c the spectral roll-off frequency and $sr/2$ the Nyquist frequency.

The **spectral brightness** is another method for measuring high-frequency energy in fixing the cut-off energy and measuring the fraction of energy above that.

The **spectral entropy** is a measure of the periodicity of a signal, since periodic sounds tend to have significantly lower entropy value than noise. Entropy is computed as:

$$H(x) = -\sum_{i=1}^{N} p(x_i) log_{10} p(x_i) \tag{2.10}$$

The **spectral flatness** indicates whether the distribution is smooth or not and is computed as the ratio between the geometric mean and the arithmetic mean of the energy spectrum value:

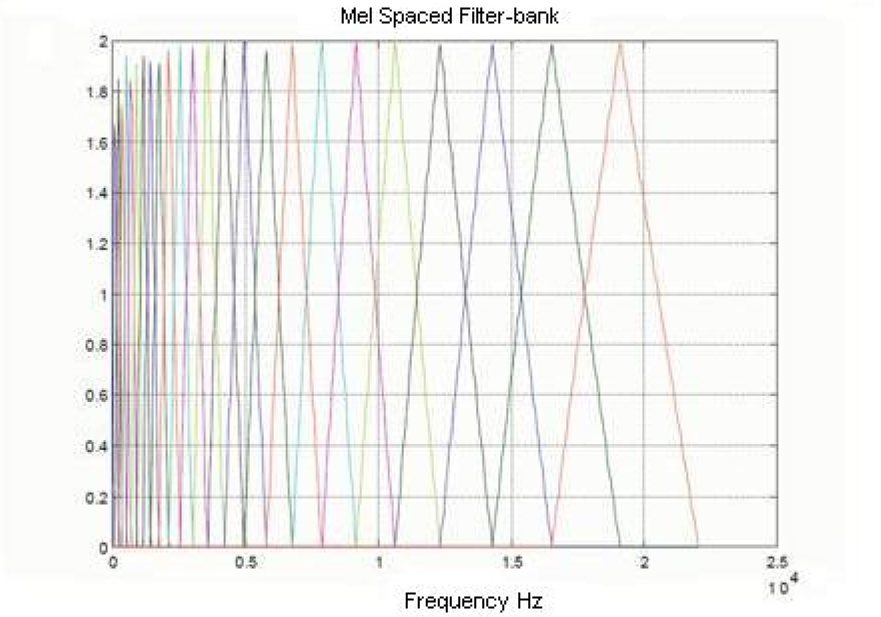
$$SF(band) = \frac{\left(\prod_{k \epsilon band} \alpha(k)\right)^{1/k}}{\frac{1}{k} \sum_{k \epsilon band} \alpha(k)} \tag{2.11}$$

where, $\alpha(k)$is the amplitude in frequency band number k.

Mel-frequency cepstral coefficients (MFCCs) are very important descriptors for the field of speech processing. Also, they have shown to work well in monophonic audio signals, as they capture effectively the shape of the spectrum. In polyphonic recordings they are not such effective, as they capture the shape of the spectrum calculated from several sources. However, if only one instrument is playing or is relatively more salient, they have been proven to be particularly useful. The Mel scale models how the human auditory system perceives different frequencies, by being linear at low frequencies and logarithmic at high frequencies (above 1000Hz) as depicted in Figure 2.2.

For the computation of the MFCCs the signal is segmented in windows and the STFTs are computed. Then amplitudes are mapped to a filter-bank on the Mel-scale and their logarithm is computed. Finally, the MFCCs are taken from the Discrete Cosine Transform of these amplitudes. Usually, the number of coefficients are 12 or 13. In his study we extracted 13 MFCCs in order to build our classification models.

Figure 2.2: Mel Filter-bank



2.8 Classification Basics

Classification is the task of assigning objects to one of several predefined categories. Especially, a classifier takes as input data a collection of records (or instances) that are characterized by a tuple (**x**,y), where **x** is the attribute set and y is the class label. The attribute set includes several features or properties of the instance and can be either discrete or continuous. On the other hand, the class label must be a discrete attribute and this distinguishes classification from regression. Thus, classification is the task of learning a target function f that maps each attribute set $\mathbf{x}$ to one of the predefined class labels y. This target function is also known as classification model. A classification technique (or classifier) is a systematic approach to build classification models from a given data set.

Examples of classifiers are decision trees, neural networks, support vector machines, logistic models etc. Each technique employs a learning algorithm in order to build a model that best fits the relationship between the attribute set and the class label of the data. This model should apart from fit well input data, correctly predict the class labels of instances it has never seen before. The input data consist the training set, while the unknown records consist the testing set.

In order to measure the performance of a model, the number of correctly and incorrectly predicted test records is measured. These measures are usually presented in a tabular form, known as a confusion matrix:

Table 2.1: Confusion matrix

	Predicted Class	
Actual Class	class1	class2
class1	True Positive	False Negative
class2	False Positive	True Negative

In order to compare the performance of different models metrics such as accuracy and error rate are widely used:

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \tag{2.12}$$

$$\text{Error rate} = \frac{\text{Number of wrong predictions}}{\text{Total number of predictions}} \tag{2.13}$$

Other measures that are used in Information Retrieval are precision, recall and f-measure, computed as:

$$\text{Precision} = \frac{\text{Number of documents retrieved that are relevant}}{\text{total number of documents that are retrieved}} \tag{2.14}$$

$$\text{Recall} = \frac{\text{Number of documents retrieved that are relevant}}{\text{total number of documents that are relevant}} \tag{2.15}$$

$$\text{F-measure} = \frac{2 * \text{precision} * recall}{\text{precision} + \text{recall}} \tag{2.16}$$

2.9 Tools

In this project we have used several software tools for audio processing and classification. Especially, we performed feature extraction from the audio files using Essentia[1], an audio processing library developed at the Music Technology Group and also using the MIRToolbox[2] for MATLAB, developed by Olivier Lartillot. The Machine Learning tool we used to perform our experiments was Weka[3],a collection of machine learning algorithms for data mining tasks written in Java. Apart from these main tools, several programming languages were used, mainly Python, for writing scripts to process the features files and also for conversions between the several outputs in order to be compatible, Java for Weka commands and MATLAB scripts to perform feature extraction using the MIRToolbox and plot classification results.

[1] http://mtg.upf.edu/technologies/essentia

[2] http://www.jyu.fi/hum/laitokset/musiikki/en/research/coe/materials/mirtoolbox

[3] http://www.cs.waikato.ac.nz/ml/weka/

Chapter 3

Literature Review

As the main goal of this thesis is to develop classification models that are capable of distinguishing between sung vowels and consonants and pure instrumental music, we are interested in reviewing techniques both on singing voice detection and singing voice recognition. As it becomes obvious from the following sections, these two topics are highly related and they share common techniques and procedures.

3.1 Singing Voice Detection

Singing voice detection addresses the problem of segmenting an audio file into fragments containing singing voice (with or without instrumental background) and purely instrumental (without singing). Apart from the approaches that address specifically this task, there are approaches in similar problems like singer identification (Berenzweig *et al.*, 2002; Kim, 2002; Tsai and Wang, 2006; Zhang, 2003) or singing voice separation (Suzuki *et al.*, 2007) that include a singing voice discrimination step. The approaches follow some similar basic steps. Firstly, some pre-processing of the audio files may be implemented in order to attenuate the background instrumental sounds and enhance the singing voice signal. Then, feature extraction is performed from audio signal frames that are considered nearly stationary. The descriptors extracted and also the frame size may differ in the several approaches. The following step is to classify the frames to one of the classes using statistical classifiers, or threshold methods, using as input the feature vectors extracted. Last, some approaches in order to reduce classification errors include bootstrapping or smoothing techniques, usually based in heuristics.

As long as the step of pre-processing the audio files is concerned, the main goal is to reduce the influences from the accompaniment sounds. (Nwe *et al.*, 2004) and (Shenoy *et al.*, 2005) used a filterbank of triangular filters spaced on a linear-logarithm scale and a series of inverse comb filters respectively, in order to attenuate the signal at the frequencies (and the corresponding harmonics) in the key of the song. This technique is based on the fact that instrumental sounds have a harmonic structure, while singing voice is not perfectly harmonic, due to two important aspects of singers' F0 control, vibrato and intonation. Thus, it will remain after the filtering. In (Kim, 2002) the audio signal is firstly filtered with a band-pass filter, which allows the vocal regions to pass through while attenuating other frequency regions. In order to further cut out other high energy sounds that belong in this region of 200Hz-2000Hz, an inverse filterbank is used to find the fundamental frequency where the signal is most attenuated.

In the literature there is a great set of features extracted from the audio and used for the classification. In (Shenoy *et al.*, 2005) the audio feature extracted is the amplitude variation over time in each subband, because the vocal frames are normally reflected by a rise in the energy of the audio signal. In (Nwe *et al.*, 2004) Harmonic Attenuated LFPCs were used, while in (Tsai *et al.*, 2004) and (Suzuki *et al.*, 2007) MFCCs were extracted from the audio files. In (Berenzweig and Ellis, 2001) a vector of posterior probabilities features, their derived statistics and averages of these values were implemented as feature vectors. In (Chou and Gu, 2001) the features extracted were 4 Hz modulation energy, harmonic coefficient, 4Hz harmonic coefficient, delta MFCC and delta log energy in order to detect singing voice. (Berenzweig *et al.*, 2002) used 13 PLPCs, their deltas and double deltas. (Kim, 2002) used a harmonicity measure, defined as the ratio of the total signal energy to the maximally harmonically attenuated signal. (Zhang, 2003) extracted energy features, ZCR, harmonic coefficient and Spectral flux. (Maddage *et al.*, 2003) used LPC, LPC derived cepstrums (LPCC), MFCC, spectral power (SP), short time energy (STE), and ZCR. In (Maddage *et al.*, 2004) the twice-iterated Fourier Transform (TICFT) is computed over each frame, where the magnitude spectrum of a first FT of the audio frame is input to a second FFT.

(Tzanetakis, 2004) used spectral shape features (Rolloff, Flux, Relative Subband Energy), MFCCs, mean and standard deviation of pitch and centroid and LPCs for the classification. In (Rocamora and Herrera, 2007) different sets of features were tested such as MFCCs and their deltas, LFPC their deltas and double deltas, PLPCs and their deltas, HC and pitch, with MFCCs and deltas found to perform better.

The classification method used in (Shenoy *et al.*, 2005) was a threshold method on the energy function, such as the proportion of frames classified as vocals to be equivalent to the proportion of the singing in the entire song, as estimated by a vocal duration processor. In (Nwe *et al.*, 2004) the classification was done with multiple HMM models based on three parameters, the section type (intro, verse, chorus, bridge and outro), the tempo and the loudness. In (Berenzweig and Ellis, 2001) an HMM framework with two states, “singing” and “not singing” was used for the task. In (Chou and Gu, 2001) and (Tsai *et al.*, 2004) GMM models were implemented to distinguish vocal from non-vocal signals. In (Berenzweig *et al.*, 2002) a two class (voice/music) MLP has been used. (Kim, 2002) used a threshold method on the harmonicity measure to classify the segments. In (Zhang, 2003) the audio features extracted were compared with a set of predetermined thresholds. In (Maddage *et al.*, 2003) a Multi-layer Neural Network, an SVM and a GMM were compared for their performance and the SVM was found to outperform the other classifiers. In (Maddage *et al.*, 2004) singing voice frames were separated from instrumental frames based on a linear threshold on the energy of the second FFT spectrum. (Tzanetakis, 2004) used a naive bayes network, nearest neighbour algorithms, back propagation ANN, a decision tree classifier based on the C4.5 algorithm , a support vector machine trained using the Sequential Minimal Optimization (SMO) and logistic regression as classifiers. In (Tsai and Wang, 2006)the vocal/non-vocal classifier consists of a front-end signal processor that converts digital waveforms into spectrum-based feature vectors, and a back-end statistical processor that performs modelling, matching and decision making, based on log-likelihoods. In (Rocamora and Herrera, 2007) different classifiers were compared an SVM, a back propagation NN, a decision tree classifier and two different K-Nearest Neighbors, with the SVM found to outperform the other classifiers.

As long as classification error-pruning is concerned, a bootstrapping method is implemented in (Nwe *et al.*, 2004), using the frames with high confidence score to build song-specific vocal and non-vocal models. In (Chou and Gu, 2001) the segmentation results were further smoothed into homogeneous segments, using a rule-based post-filtering method. (Maddage *et al.*, 2004) used heuristic rules which are based on knowledge of chord pattern changes of popular music in order to improve accuracy. In (Tsai and Wang, 2006) segment-based decision was improved by merging adjacent segments into longer homogeneous ones, if those adjacent segments do not cross a vocal/non-vocal boundary. (Rocamora and Herrera, 2007) considered some post processing strategies to improve the classification performance, based on heuristics.

In the several approaches we encounter different segmentation techniques for the audio files, which in most of the cases are acting also as an error-pruning process (2.6), when the decision of the classifier is made for whole segments. In (Nwe *et al.*, 2004), (Maddage *et al.*, 2004) and (Shenoy *et al.*, 2005) beat length segments were used. In (Berenzweig and Ellis, 2001) an HMM was used to make the segmentation. In (Tzanetakis, 2004) the segmentation is performed specifically for each individual song using a bootstrapping process. In (Li and Wang, 2007) the segmentation was done into portions by detecting instances when significant spectral changes occur.

3.2 Singing Phoneme Recognition

Singing phoneme recognition is very different from ASR because of the differences we already saw between speech and singing voice. Here we are going to review approaches to singing phoneme recognition and singing voice to lyrics alignment either in pure singing voice signals or in polyphonic music signals.

The first approaches to the singing voice transcription problem were considering pure monophonic singing voices, without accompaniment and adapted speech recognizers for lyrics transcription.

(Loscos *et al.*, 1999) presented some ideas on how to move from speech phoneme recognition to singing voice to text alignment with a real-time application. The authors' interest is mainly in the architecture of the HMM and the adaptation of its parameters in order to fit the singing voice case. Several low-level and high-level features are extracted from the audio signal and are given as input to the HMMs. As the task is the alignment of the audio signal with the corresponding lyrics, a Viterbi text to speech algorithm is implemented to find the most probable path and give the time positions of each phoneme.

(Suzuki *et al.*, 2007) use both the melody and the lyrics of the user's singing voice in order to retrieve a song from a database. The authors used a large vocabulary speech recognition system, with an HMM as the acoustic model, adapted to the singing voice using the speaker adaptation technology. As the task of recognition here exploits the constraints posed by the assumption that the sung lyrics are a part of the database, the approach uses a FSA that accepts only lyrics from the database.

(Sasou *et al.*, 2005) tested an Auto Regressive HMM with pure singing voice signals from the RWC database[1]. (Yaguchi and Oka, 2005) developed a system that retrieves songs from a database using sung and spoken clauses. For the phonemic recognition of both query and database audio signals, the authors used discriminate functions using the Bayes estimation. 26 different discriminate functions were made for 26 different kinds of phonemes, using speech data. Although the phoneme label error was relatively high, the task was considered as a simple conversion method, since both query and database audio signals were transformed into phonemic sequences with the same method and thus, with a "same type" of error. After this conversion a Continuous Dynamic Programming algorithm was implemented to match the input query with a song from the database.

(Gruhne *et al.*, 2007) implemented a system that performs automatic classification of 15 voiced sung phonemes in polyphonic audio. Their procedure was based on harmonics extraction and resynthesis of a number of partials as a preprocessing step, in order to reduce influences from accompanying sounds. Then low-level features were extracted from the audio and classified using different classification techniques like SVM, GMM and MLP.

LyricAlly by (Kan *et al.*, 2008) is probably the first English lyrics sentence-level alignment system for aligning the lyrics to the music signals for a specific

[1] http://staff.aist.go.jp/m.goto/RWC-MDB/

structure of songs. The problem of alignment is divided into a high-level alignment of the song's structural elements and a second round of low-level line alignment. Beat, measure and chorus detection are used for the high-level alignment. Also, repeated sections in the lyrics are detected and assigned to the chorus. Approximate duration of each section are estimated based on the previous features using a singing phoneme duration database. For the low-level alignment a vocal detector is implemented like in (Nwe *et al.*, 2004) to locate parts in the audio which contain vocals. Durations are estimated for each line exploiting the rhythmic structure of the song. In later work (Iskandar *et al.*, 2006) the line-level alignment is refined to syllabic-level. In this approach MFCCs are used to represent each frame and a triphone is implemented with a three-state left-to-right HMM as the recognition model. Then, the two sequences are aligned using a Dynamic Programming algorithm that takes into consideration constrains posed by music.

(Fujihara *et al.*, 2006) perform automatic synchronization between lyrics and polyphonic music signals for Japan CD recordings. Their proposed system includes detection of vocal segments, segregation of vocals and adaptation of a speech recognizer to the segregated vocal signals. During the first step, harmonics extraction and resynthesis is performed as in (Gruhne *et al.*, 2007). Then a simple HMM is used in order to keep only the vocal regions and remove the non-vocal sections. Last, features are extracted from the audio (MFCCs, delta MFCCs and delta power) and the Viterbi algorithm is used to align the segmented vocal parts with the corresponding lyrics. In this step, only the vowel phonemes are considered as the consonants have been removed at the accompaniment sound reduction step. Thus, the sequence of lyrics is transformed to a sequence of vowels, with short pauses introduced between the words.

(Wong *et al.*, 2007) propose a system for real-time alignment of Cantonese music, which is a particular tone-language. As mentioned before, in tone languages the meaning of a word changes when pronounced with a different pitch. The authors are interested in sentence-level and not phoneme-level alignment. The interesting technique used here is the enhancement and extraction technique of the singing voice. This innovative technique is based on the common mixing practice, where the different tracks of the singing voice and the musical instruments are mixed together to give the final track. Based on the fact that the singing voice and the drums are in the center position of the stereo channel, the technique performs spectral subtractions to obtain the enhanced vocal signal. Then an MLP is used to segregate the vocal from the non-vocal segments taking as input the spectral flux, the HC, the ZCR, the MFCCs, the amplitude level and the 4Hz modulation energy. Last, the DTW algorithm is used to align the two sequences.

Chapter 4

Methodology

4.1 Database Creation- Corpus Selection

As in every automatic classification problem, the first task to be accomplished is the selection or creation of a database to be used for training, testing and evaluating the statistical models. In this study our interest is in polyphonic recordings with English lyrics. A set of songs has been selected that belong to a wide spectrum of genres and that are performed both by female and male singers. As this procedure was very time-consuming (around 8 hours for each a 15 sec snippet), the number of the annotated songs were cut down to 15. Specifically, 3 jazz, 3 pop, 2 funk, 1 trip-hop, 1 hip-hop, 1 country, 1 soul, 1 blues, 1 electronica and 1 rock song were annotated, 8 performed by male singers and 7 performed by female singers. After the decision about the genres to be included in the database, representative singers and songs were selected for each of the genres.

From these songs, 15 sec snippets containing mostly vocals were extracted and further used in the procedure. Since our next task was to annotate phonemically the singing voice signals, the segments we selected didn't contain multiple voices, clapping, laughing, audience screaming etc. as in these cases it would be difficult to find the correct time boundaries. Other considerations regarding this selection, were to take snippets from different parts (intro, chorus, verse, bridge, outro) and avoid very strange pronunciation and accent. As many of these songs contain slung words that are not included in a dictionary (so that their phonemic transcription is available), we selected parts with "clear" lyrics. Finally, we added a set of 166 of pure instrumental snippets. All the audio files used in the database were sampled at 44100Hz. In Appendix A there is a list of the songs we used in our database, along with the starting and ending time of the durations used. Following the same procedure we annotated 5 more 5sec snippets in order to evaluate our models (see Section 5.2). Especially:

- one audio file containing pure instrumental music without any percussion sounds (instance no1, Norah Jones - Seven Years)
- one file of pure instrumental music with percussion (instance no2, Brad Mehldau - Knives Out)
- one file containing pure singing voice without any instrumental background, even if the models were not trained with pure singing voice signals (instance no3, Carol Sloane - My one and only love)
- one file with singing voice and instrumental accompaniment without percussion (instance no4, Duffy - Syrup and Honey)
- one file with singing voice and instrumental background with percussion (instance no5, Oi Va Voi - Refugee)

4.2 Data Annotation Method

Our next step was to manually align these songs. Alignment between singing voice and text refers to the temporal relationship between audio signals and the corresponding lyrics. The goal in this step was to precise the starting and ending points of each phoneme. These manual annotations were also used as our ground-truth for the system implemented. Before the annotation task, the lyrics of the songs were collected from several web pages[1] and a listening task was performed in order to select the right version for each song. As some lyrics contained abbreviations like “waitin”, “told ya”, “darlin”, “gonna”, “n”, etc. they had to be corrected, so that the words were contained in the dictionary.

[1] www.lyrics.com, www.azlyrics.com, www.lyricsbay.com

The lyrics were then turned to their phonemic representation using a SAMPA[1] transcription module (see next paragraph for details). Silences were produced from the transcription module between the several words.

Several phonetic transcriptions have been adopted from scientists over time. The most commonly used are the IPA[2] and the SAMPA transcription. The sounds represented by the symbols of these transcription methods are typical sounds in many different languages (Ladefoged, 2005). Here we are interested in the English language and the transcription we have used is the SAMPA transcription method. Figure 4.1 shows the SAMPA chart for English (allophones are inside blue frames). The symbols that are used throughout this thesis to represent phoneme sounds are according to the SAMPA transcription method.

The main annotation task was about putting time borders on the signal for the beginning and ending time of each sung phoneme. The software used in this step was WaveSurfer[3] (Sjlander and Beskow, 2000) that is used widely for signal transcription tasks. It has an option of loading text labels, with which the phonetic transcription in SAMPA was inserted. The software outputs the annotations in a .lab file that includes the starting times, the ending times and the corresponding phonemes. This software gives several options in representing the waveform and the spectrogram and has also an interactive playback option. Based on the signals representations in both time and frequency domain, and continuous listening tests we assigned time boundaries for each of the phonemes. Figure 4.2 visualizes the annotation software.

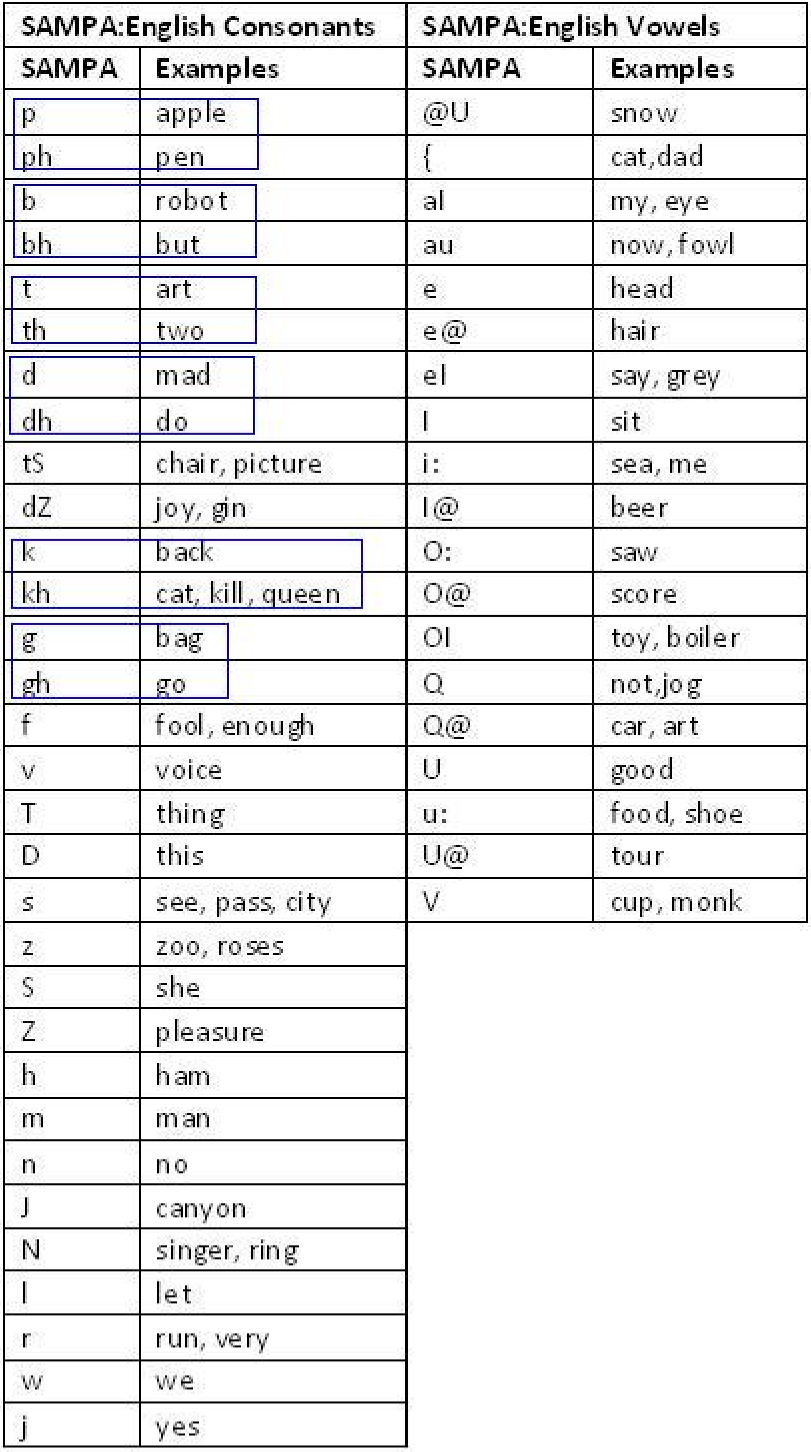
As the task of annotating signals is not trivial, a literature review had to be done on this task. Since most of this work was done for the case of speech we turned to annotated speech databases and manuals. The TIMIT database (Garofolo *et al.*, 1993) and its documentation provided a very useful guide. Several practice tests had to be made using speech signals from the TIMIT database and their correspondent annotations. Also several guides were used that describe spectrogram properties for each phoneme, their average durations and other prop-

[1] http://www.phon.ucl.ac.uk/home/sampa/

[2] http://www.arts.gla.ac.uk/IPA/

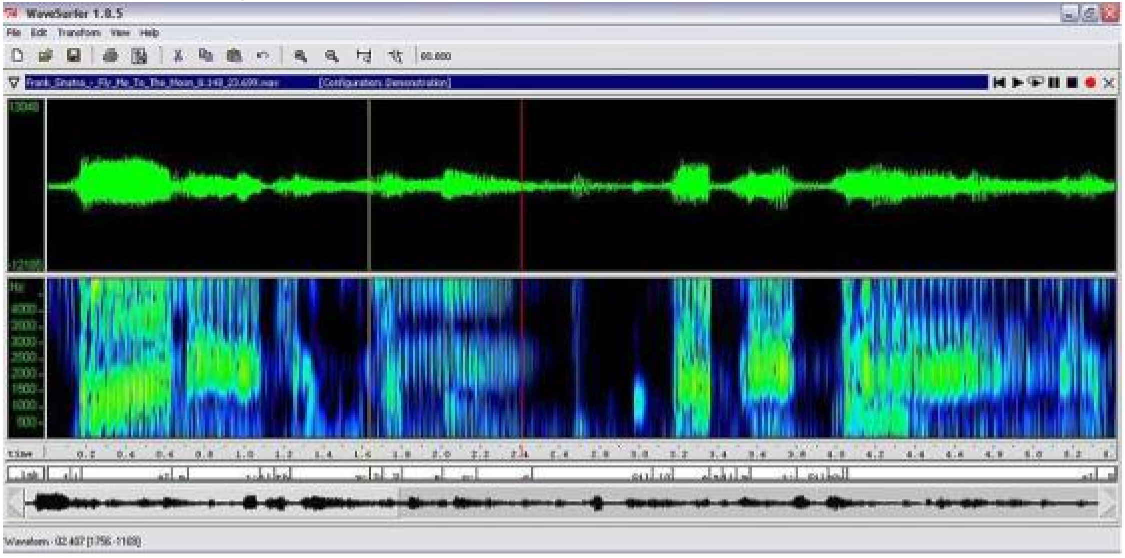
[3] http://www.speech.kth.se/wavesurfer/

Figure 4.1: SAMPA chart for English



SAMPA:English Consonants		SAMPA:English Vowels	
SAMPA	Examples	SAMPA	Examples
p	apple	@U	snow
ph	pen	{	cat,dad
b	robot	aI	my, eye
bh	but	au	now,fowl
t	art	e	head
th	two	e@	hair
d	mad	eI	say, grey
dh	do	I	sit
tS	chair, picture	i:	sea, me
dZ	joy, gin	I@	beer
k	back	O:	saw
kh	cat, kill, queen	O@	score
g	bag	OI	toy, boiler
gh	go	Q	not,jog
f	fool, enough	Q@	car, art
v	voice	U	good
T	thing	u:	food, shoe
D	this	U@	tour
s	see, pass, city	V	cup, monk
z	zoo, roses		
S	she		
Z	pleasure		
h	ham		
m	man		
n	no		
J	canyon		
N	singer, ring		
l	let		
r	run, very		
w	we		
j	yes		

Figure 4.2: Annotation with Wave Surfer Software



erties of each phoneme (Ladefoged, 2005)), based on which phonemic annotations are done for speech signals.

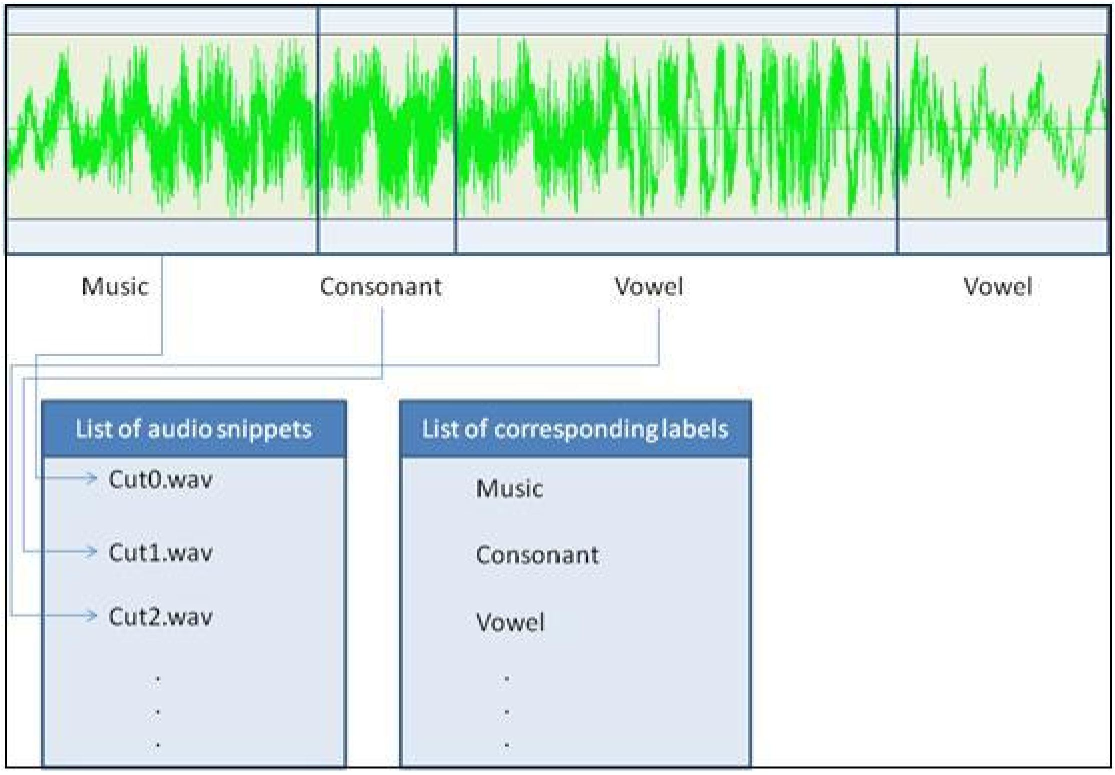
Of course, in our case the background noise made the task not so clear as several properties of the phonemes are "hidden" under the instrumental accompaniment. Especially the percussive sounds were causing deterioration to the spectrograms and the annotation was difficult in cases were consonant phonemes coincided with percussion. In cases like that even the human listener is not able to perceive the consonant phonemes. It was a challenge if a classification model would be able to overcome this.

4.3 Splitting Methods- Segmentation

The fundamental step for audio content analysis is the signal segmentation. Information within segments has to be nearly stationary. After the segmentation step, feature extraction is performed and other kinds of processing, like statistical information modelling. In this thesis we implemented 3 segmentation techniques that are described below.

The first approach followed was to segment the audio signal according to the time boundaries we have annotated, hereinafter called manual annotation segmentation. This procedure was done with a python script using as input a set of audio signals in .wav format along with their corresponding annotations. The output was a set of smaller audio files, each one containing only one phoneme, or pure instrumental music according to the annotations and a set of corresponding annotation files. Figure 4.3 visualizes this segmentation procedure.

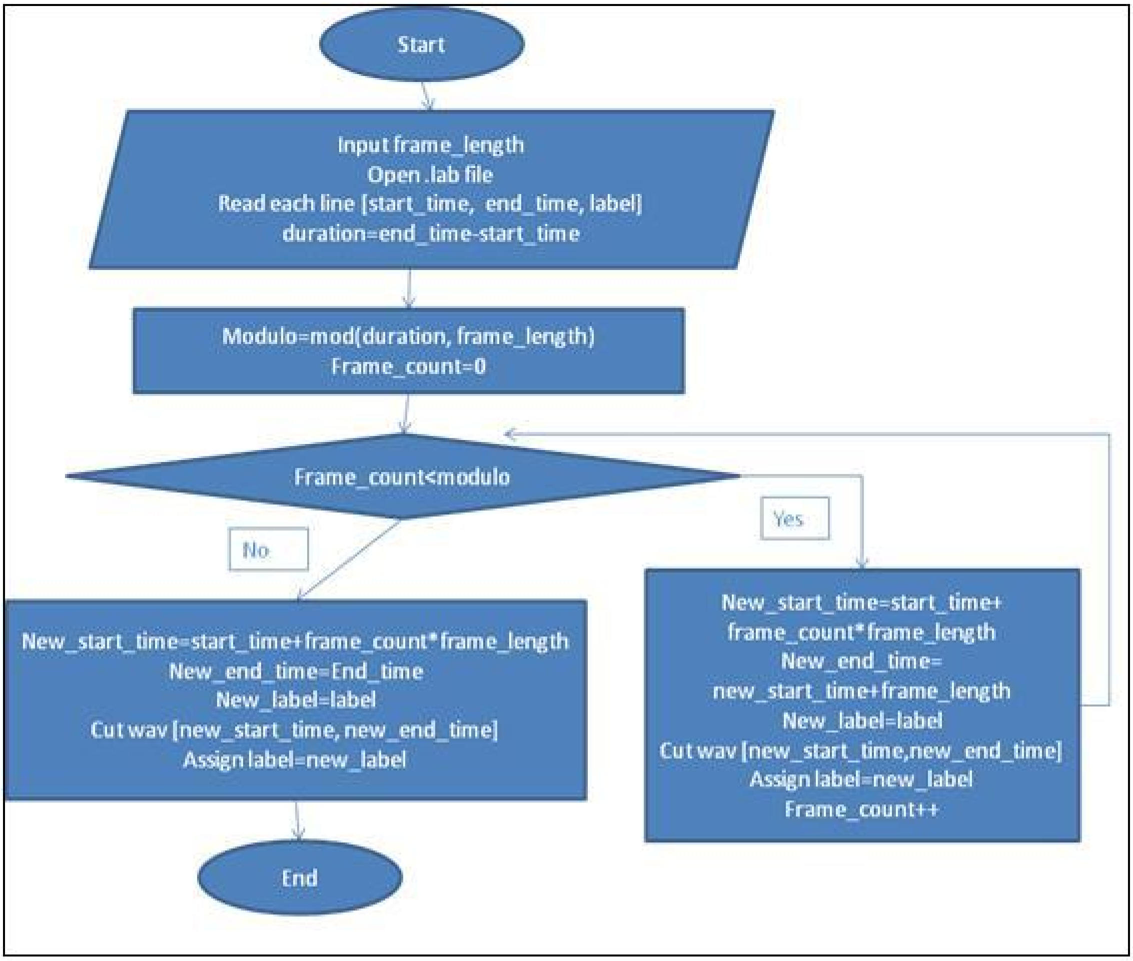
Figure 4.3: Manual annotation segmentation procedure



As our final goal was not to build a classifier that would decide on the class based on the manual annotation segmentation but to try to segment automatically the audio files in an "optimal" way, further segmentation techniques were implemented. The manual annotation segmentation method would give us an upper limit for the accuracy of any automatic segmentation method. Also, the information it conveys, like the average length of vowel and consonant phonemes can be used in the final system.

Our next approach was to automatically segment the audio files using a fixed frame length of a few milliseconds. In order not to lose any information from the manual annotations in the training phase this algorithm was implemented so as to use the annotated time stamps. In figure 4.4 we present the flow chart diagram of the python script we implemented for this segmentation, hereinafter called annotation-dependent fixed-length segmentation.

Figure 4.4: Annotation-dependent fixed-length segmentation algorithm flowchart


Start
Input frame_length
Open .lab file
Read each line [start_time, end_time, label]
duration=end_time-start_time
Modulo=mod(duration, frame_length)
Frame_count=0
Frame_count<modulo
No
Yes
New_start_time=start_time+frame_count*frame_length
New_end_time=End_time
New_label=label
Cut wav [new_start_time, new_end_time]
Assign label=new_label
New_start_time=start_time+
frame_count*frame_length
New_end_time=
new_start_time+frame_length
New_label=label
Cut wav [new_start_time,new_end_time]
Assign label=new_label
Frame_count++
End

Different segmentation lengths (from 30 ms to 150 ms) were applied in order to find peaks in precision and recall for each class and for each frame length and incorporate this information in the final system.

The last step was to find the optimal segmentation length for any unknown to the classification system audio file for the evaluation phase. This segmentation technique need not to use any information from the manual annotations. The audio signals were segmented in fixed frame length snippets of different lengths, overlapping and non-overlapping in order to observe the accuracy in each case. In order to evaluate the classification models in this case labels had to be assigned for each frame of the evaluation set. As each frame could belong to more than one class, decisions about the real class of each frame are decided according to class that the middle sample of each frame belongs. Using this technique, the accuracy was measured as the percentage of correctly classified frames. Another possible technique in order to measure the accuracy of the models, is to find the percentage of correctly assigned overall durations. This segmentation technique will be called hereinafter annotation independent fixed length segmentation.

4.4 Extracted Features

Feature extraction is a major stage in any classification system in general, and in audio signal classification systems in particular. After the segmentation of the audio, several representative features are extracted from the audio files.

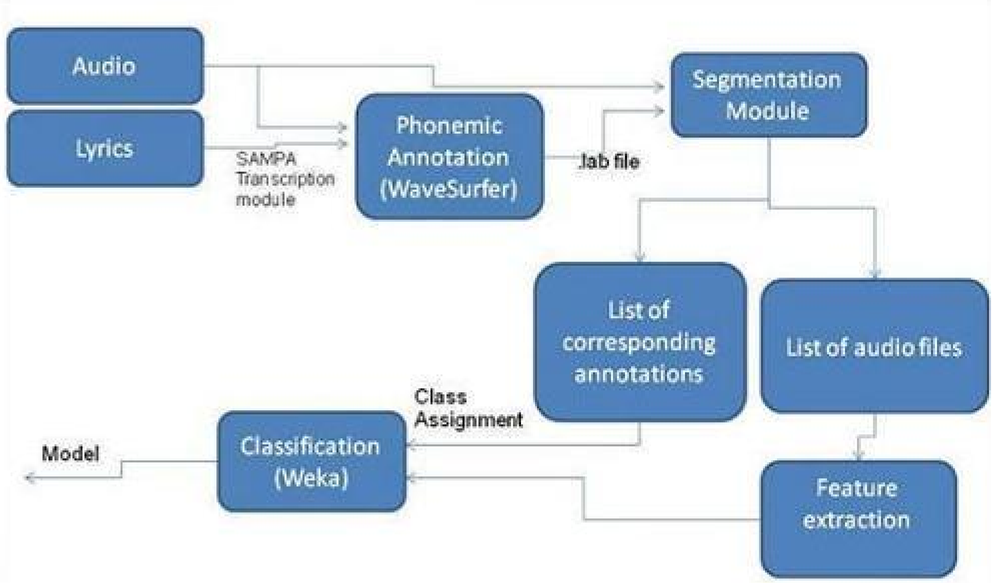
In this research we used two main tools for feature extraction. The first tool, was the MIRToolbox for MATLAB and the second was Essentia, which outputs a large set of features. Especially, using MIRToolbox, we extracted a set of 13 MFCCs, the spectral centroid, brightness, skewness, spread, kurtosis, entropy, flatness, irregularity and roll-off, while using Essentia we extracted more than 300 available features. In the second case the feature vector had to be cut down in order to avoid overfitting, and a feature selection process has been carried out, using the Weka software. The final number of features selected for each experiment is cited along with the results in Chapter 5.

4.5 Classifiers

For the classification process we used the Weka software. Input data were converted to a CSV or an ARFF file, so as to be compatible with Weka. They were firstly filtered so as to have the same number of instances in each class. Also, feature selection was performed using Weka algorithms, in order to reduce the feature vector dimension and avoid overfitting. Weka gives many options for the test set. Apart from supplying a different set as a test set, a 10-fold cross validation, a percentage split of the training set or an evaluation set with no labels can be loaded. For the testing phase a 10-fold cross validation was performed. Our evaluation phase was performed on a holdout independent validation set using Weka's option for outputting predictions for an unknown set with no labels assigned to each frame. Several classifiers have been tested in this research like Weka's SMO algorithm for SVM, j48 trees, the logistic function and Multi-layer Perceptrons. Weka outputs several performance measures, like the confusion matrix precision, recall, f-measure etc.

Figure 4.5 presents the block diagram of the system.

Figure 4.5: Block Diagram of the Methodology



Chapter 5

Experiments and Results

During this project several experiments have been performed in order to optimize the performance of the classification scheme. These experiments can be categorized in attempts to:

- locate the optimal segmentation length. A fundamental step in audio content analysis is the segmentation technique used. Within a segment, signal parameters are considered fairly stationary. As this step is then followed by feature extraction and statistical information modelling, it is important for the overall accuracy of the system. In speech, fixed length segmentation has been used commonly. In audio content analysis, fixed length segmentation has also been employed. Other segmentation techniques that are being used for music signals are based usually on the rhythm (Maddage *et al.*, 2004, 2008; Nwe *et al.*, 2004; Shenoy *et al.*, 2005) or in detection of spectral changes (Li and Wang, 2007). The segmentation techniques used in this study are described in section 4.3.

- extract a representative set of descriptors. As we already described, feature extraction is also an important step. The goal is to extract a representative set of features to train the classification model. In this thesis several relevant features have been tested. Feature selection has been performed mainly using Weka's algorithms, that gives the importance of each descriptor, so as to filter out features that can cause the model either to over-fit or to use a large set of non-important features.

- find the optimal classification scheme and classification function. As in this study we had to deal with a 3-class (multi-class) classification problem, there were several schemes of classification models. The first and simplest scheme is a 3-class classifier to assign each frame into one of the 3 classes (vowels, consonants, music). Another scheme implemented consisted of 3 binary (one-against-rest) classifiers in parallel, a music-vs-other (vowels or consonants), a vowel-vs-other (consonants or music) and a consonant-vs-other (vowels or music) classifier. The last scheme implemented consists of 2 binary classifiers in series (chained), a music-vs-singing voice binary classifier and a vowel-vs-consonant binary classifier that takes as input only the frames classified as containing singing voice.

5.1 Experiments

In this chapter we present the main categories of experiments performed in this study. All the results obtained during the testing phase are acquired using a 10-fold cross-validation in Weka. Also, filters are applied in all the experiments so as to get the same number of instances in each class. In this section the most representative experiments are presented along with the procedure followed and the corresponding results.

5.1.1 Experiment no1: Annotation-based Segmentation Models

One of the first experiments performed was the classification of features extracted from phoneme-length segments. Following the manual annotation segmentation we described in section 4.3, we segmented the audio files into smaller snippets. We extracted several features that are considered relevant for discriminating vowels from consonants and music. Using the MIRToolbox, 13 MFCCs as well as the spectral centroid, brightness, skewness, spread, kurtosis, entropy, flatness, irregularity and roll off were extracted for each audio snippet. We, then, performed feature selection to this initial feature set in order to find a smaller set of important descriptors. 12 descriptors were considered relevant (MFCC mean 1, 3, 6, 8,

Table 5.1: f-measure of different classification functions in Experiment no1

class	logistic	SMO	MLP
consonant	0.811	0.793	0.779
music	0.9	0.924	0.887
vowel	0.753	0.77	0.694

10, 11, Spectral skewness mean, Spectral spread mean, Spectral kurtosis mean, Entropy of Audio waveform mean, Spectral flatness mean, Spectral irregularity mean). Performing this experiment using different classification functions we got the results presented in Table 5.1.

Evaluation of the logistic model on the evaluation set described in Section 5.2 gave 75 % of accuracy (as computed by formula 2.12). From these first evaluation results it is quite safe to conclude that since the performance of the model in the two sets is close (6.92% decrease in the evaluation set), the training set is good enough in capturing the variability of the data of the different classes. Table 5.2 shows the confusion matrix of this classification results.

Table 5.2: Evaluation of classifier in Experiment no1

consonants	music	vowel	classified as
12	0	6	consonant
1	3	0	music
3	0	15	vowel

From the confusion matrix in Table 5.2 we can conclude that vowels achieve high precision, music high precision and recall and consonants high recall. Furthermore, even if the number of instances used in the evaluation set is small, we can observe that the classifier achieves quite high accuracy if the segmentation of the audio snippets is that of their duration.

From these first results we could see that extracting a small set of features (12 as defined from feature selection) the f-measure of the classifier reached an average of 81.9209 %. Of course, in any unknown audio file to be classified, annotation based segmentation cannot be applied, so we had to implement an automatic segmentation technique. These results could be used as an upper bound of the feasible system accuracy. The experiments that followed focused on finding the optimal segmentation length.

5.1.2 Experiment no2: Fixed Length Segmentation

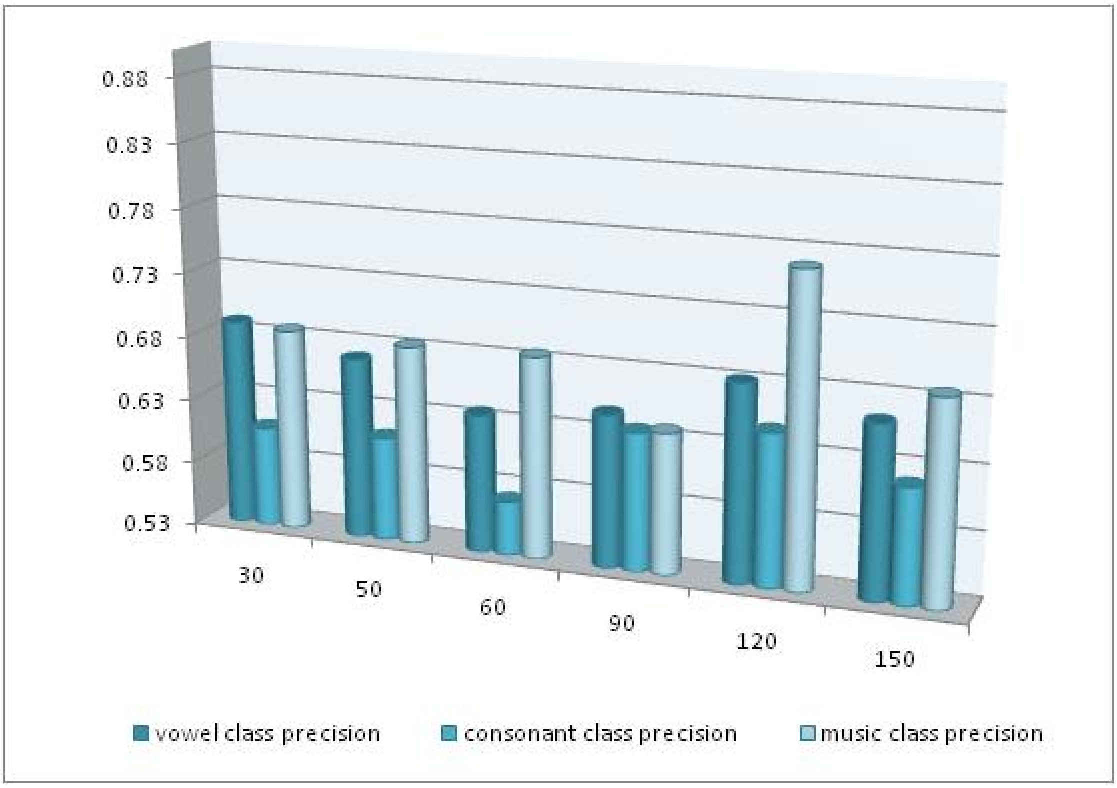
The next set of experiments performed were about finding the optimal segmentation length (fixed) for the audio files. Each audio file was segmented in fixed length non-overlapping frames, according to the annotation dependent fixed length segmentation technique described in Chapter 4. From each of these frames the same feature set as in Experiment no1 was extracted and used for the classification. We searched in the area of 30 ms to 150 ms. Table 5.3 shows the performance of an SMO classifier (that was found to out-perform other classification functions) built using different segmentation lengths.

Table 5.3: Performance of 3-class classifiers at different segmentation lengths

Segmentation length	Precision	Recall	F-measure	Class
30 ms	0.69	0.637	0.663	vowel
	0.608	0.686	0.645	consonant
	0.687	0.651	0.669	music
60 ms	0.637	0.618	0.628	vowel
	0.572	0.626	0.598	consonant
	0.686	0.643	0.664	music
90 ms	0.648	0.63	0.639	vowel
	0.637	0.602	0.619	consonant
	0.639	0.692	0.664	music
120 ms	0.682	0.641	0.661	vowel
	0.648	0.722	0.683	consonant
	0.769	0.726	0.747	music
150 ms	0.663	0.61	0.635	vowel
	0.619	0.657	0.637	consonant
	0.687	0.7	0.693	music

From these results we could see that generally the 120 ms segmentation was the optimal one for a 3-class classifier. Nevertheless, from the accuracy of each class at each segmentation length, we saw that different classes had different "optimal" segmentation lengths. Figure 5.1 shows the precision obtained for each class at each segmentation length. This information could be used in building three one-class-versus-other (binary) classifiers for each class using these optimal segmentation lengths found and then combining their results.

Figure 5.1: Precision per class per segmentation frame length



5.1.3 Experiment no3: Binary Classifiers based on Optimized Segmentation Length

As in the previous experiment we found that each class had a different optimal segmentation length, we used this information and built three one-vs-all (binary) classifiers. For the classes of music and consonants the optimal segmentation length found was 120 ms, while for the vowels was 30 ms. Thus, we segmented our audio files in 30 ms and classified our instances as belonging to the vowel class or the other classes (music or consonants), in 120 ms and classified our instances as belonging to the music class or the others (vowels or consonants) and finally to the consonant class or the others (vowels or music). In this experiment the segments were obtained using the annotation-dependent fixed length segmentation technique. The same set of features as before was extracted from the audio segments and used for the classification. Table 5.4 shows the precision and recall obtained for each class by the three optimized binary classifiers, using the SMO algorithm for SVM in Weka that was found to out-perform MLP and logistic classification functions.

Table 5.4: Performance of binary classifiers

Classifier	Precision	Recall	Class
Vowels vs other	0.789	0.826	vowel
	0.817	0.779	other
Consonants vs other	0.764	0.803	consonants
	0.793	0.752	other
Music vs other	0.817	0.79	music
	0.797	0.823	other

These binary classifiers can be used in parallel in order to assign each segment into one of the three classes. A short evaluation of this scheme showed that further heuristic rules have to be implemented in order to handle overlaps between the three classes. Some of these heuristics can be based in the average duration of each class. Usually, classes do not change very quickly between music and singing voice (vowels and consonants). Music segments between singing voice segments last for several seconds, so it is quite safe to apply heuristic rules based on class durations to the classification output in order to improve accuracy. Specifically, we applied the following smoothing heuristic rule to the classification output:

$$\text{if (frame(i-1)=music and frame(i+1)=music) then frame(i)=music} \tag{5.1}$$

The result was an additional 5% increase in the overall accuracy of the models. Some comments that can be made on this rule is that it doesn't handle overlaps between the vowel and consonant classes and further post-processing of the output is needed. Furthermore, this evaluation procedure showed that it is probably more efficient to have two classifiers in series, one binary classifier to detect music segments and a second one to assign singing voice segments into vowels and consonants.

5.1.4 Experiment no4: Extraction of a Large Feature Set

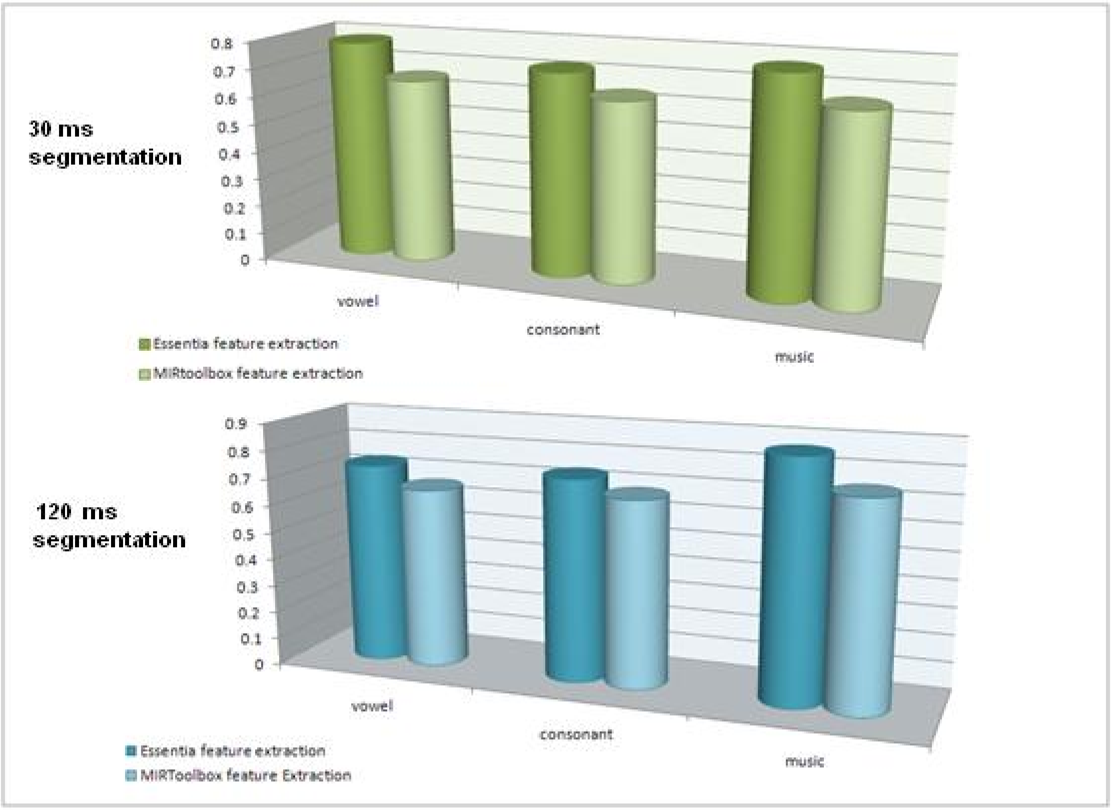
After experimenting with different segmentation lengths and techniques, we used the optimal ones found to build more robust classification models using a larger set of audio features. Essentia was used to get an initial set of 389 audio features. After that, feature selection was performed using Weka's feature selection algorithms. As explained in Experiment no3 the optimal segmentation lengths found were 30 ms and 120 ms. Those lengths were used for the segmentation in this experiment and again the segmentation method was the annotation dependent fixed length segmentation. In Table 5.5 the results of the 3-class classifiers are presented. For this classification a set of 60 features was selected and Weka's algorithm SMO for SVM was used as the classifier.

Table 5.5: Performance of 3-class classifiers with Essentia's feature extraction

Segmentation length	Precision	Recall	F-measure	Class
30 ms	0.795 0.745 0.75	0.779 0.709 0.801	0.787 0.727 0.775	vowel consonant music
120 ms	0.739 0.739 0.862	0.732 0.739 0.871	0.736 0.739 0.867	vowel consonant music

If we compare these results with those of Table 5.3 for the correspondent segmentation lengths we can observe a 9.38% increase in performance as we used a bigger set of descriptors to train the models (Figure 5.2).

Figure 5.2: Comparison of 3-class models' f-measure using different feature sets



Another classification scheme that was considered as rational and was implemented consists of two classifiers in series. The first one to distinguishes music from singing voice and the second takes as input the singing voice segments and classifies them as vowels or consonants. From the results we got by the 3-class classification scheme in Table 5.5 we expected the 120 ms segmentation to perform better for the singing voice detection and the 30 ms segmentation to be the optimal one for distinguishing vowels from consonants. This hypothesis was consistent with the results. In tables 5.6, 5.7 the performance measures of these binary classifiers are presented. Again here the SMO algorithm was used for the classification.

Table 5.6: Performance of binary classifiers using 120 ms segments

Classifier	Precision	Recall	F-measure	Class
Music vs Singing Voice	0.874 0.904	0.907 0.869	0.89 0.886	music Singing voice
Vowels vs Consonants	0.786 0.792	0.794 0.784	0.79 0.788	vowels consonants

Table 5.7: Performance of binary classifiers using 30 ms segments

Classifier	Precision	Recall	F-measure	Class
Music vs Singing Voice	0.841 0.877	0.883 0.833	0.861 0.854	music Singing voice
Vowels vs Consonants	0.845 0.814	0.805 0.853	0.825 0.833	vowels consonants

5.2 Evaluation

As in every classification problem, evaluation of the models is needed in order to verify the accuracy of the obtained models. In this study, we evaluated the final models built in Experiment no4, as they are the optimal models that gave us the best results in the testing phase. For the evaluation we used a set of unknown to the system audio files (see Section 4.1).

From the selection of the audio files, it becomes obvious that special attention is given to percussion sounds. As we stated in sections 1.3 and 4.2 we assume that percussion sounds are highly possible to be confused with consonants, as they resemble a lot and even the human ear is difficult to distinguish them, especially when they coincide. As already explained in section 4.3, the audio files used for evaluation were segmented according to the annotation-independent fixed-length segmentation technique. Using Essentia, a set of 389 features were extracted from the audio snippets and finally a smaller set was used according to the feature selection of the training sets. Using the testing mode that outputs predictions in Weka, we got a class assignment for every frame of the audio signals in the evaluation set. Table 5.8 presents the correctly classified instances obtained by the several classification models of Experiment no4, for the 5 different instances used in the set. As every instance represents a different case of audio music signal we will comment each one and its results separately.

5.2.1 Instance no1, pure instrumental audio signal,no percussion

As we can see from the evaluation results in Table 5.8, instance no1 achieves high accuracy with every classification scheme. The 3-class classifier correctly classifies most music frames, as well as the binary music-vs-singing voice classifier. Table 5.9 presents the confusion matrices of this instance for the 3-class classifiers and Table 5.10 for the binary classifiers using different segmentation lengths.

Also, applying simple heuristics in this case, so as not to allow quick changes between the classes (see equation 5.1), the accuracy of the classifier reaches 100 % for the 3-class classifier and for the binary music-vs-singing classifier applying 120ms non-overlapping segmentation. From these results we can see that the models are able to distinguish pure instrumental music segments without percussion from sung vowels and consonants with high accuracy.

Table 5.8: Percentage of correctly classified instances in the evaluation set

Classifier	Segmentation Length	inst1	inst2	inst3	inst4	inst5	N.A [1]	N.A [1](no inst2,3)
3-class	120ms (non-overlap)	91.66%	32.56%	29.41%	73.91%	59.38%	57.38%	74.98%
3-class	120ms (overlap)	92.47%	34.34%	30.30%	70.06%	64.29%	58.29%	75.61%
3-class	30ms (non-overlap)	90.48%	43%	6.46%	56.11%	64.62%	52.13%	70.40%
music/singing	120ms (non-overlap)	97.92%	35%	7.84%	86.96%	77.14%	60.97%	87.34%
music/singing	120ms (overlap)	98.39%	28%	7.58%	86.44%	75%	59.082%	86.61%
vowels/cons.	30ms (non-overlap)	-	-	58.21%	70.72%	84.21%	71.05%	77.47%

[1] Normalized Average

Table 5.9: Confusion matrices of instance no1 for 3-class classifiers, music should be detected in all frames

segmentation	pred.as:music	pred.as:consonant	pred.as:vowel
120ms (non-overlap)	44	2	2
120ms (overlap)	172	9	5
30ms (non-overlap)	171	13	5

Table 5.10: Confusion matrices of instance no1 for binary classifiers, music should be detected in all frames

segmentation	pred.as:music	pred.as:singing voice
120ms (non-overlap)	183	3
120ms (overlap)	47	1

5.2.2 Instance no2, pure instrumental audio signal with percussion

Instance no2 from the other hand gets very low accuracy, random for the 3-class classifiers and worse than random for the binary classifiers. Table 5.11 presents the confusion matrices of this instance for the 3-class classifiers and Table 5.12 for the binary classifiers using different segmentation lengths.

Table 5.11: Confusion matrices of instance no2 for 3-class classifiers, music should be detected in all frames

segmentation	pred.as:music	pred.as:consonant	pred.as:vowel
120ms (non-overlap)	57	51	58
120ms (overlap)	14	11	18
30ms (non-overlap)	73	64	32

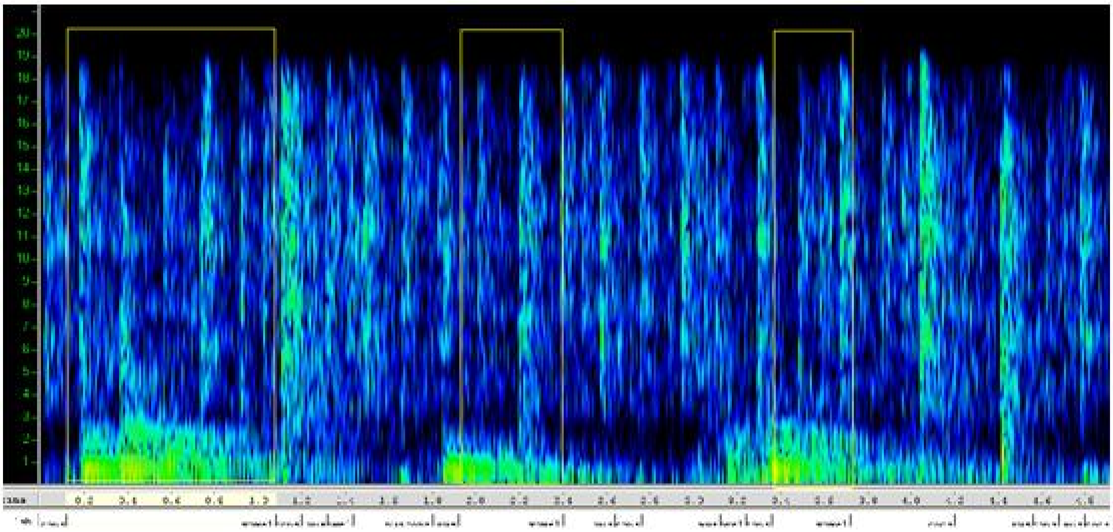
A closer look at the frames assigned by the model to the vowel class shows that contained a dominant piano. Figure 5.3 depicts the spectrogram of this instance and the frames assigned by the model to the vowel class (according to the 3-class model with 120ms non overlapping segmentation) are inside yellow frames. From the spectrogram and the waveform of these frames we could observe high

Table 5.12: Confusion matrices of instance no2 for binary classifiers, music should be detected in all frames

segmentation	pred.as:music	pred.as:singing voice
120ms (non-overlap)	47	119
120ms (overlap)	15	28

harmonicity and higher energy in low bands. The same durations were assigned by the binary music-vs-singing classifier to the singing voice class. The frames that were recognized as consonants presented low amplitude and energy spread in a wide range of bands. Frames that had dominant percussion sounds were correctly assigned to the music class.

Figure 5.3: Spectrogram of Instance no2



5.2.3 Instance no3, pure singing voice

Instance no3 is pure singing voice without any instrumental background. Although the models are not trained with pure singing voice signals, one such file was included in the evaluation set in order to observe the reaction of the model in this case and learn from his mis-classifications. Furthermore it is useful to observe its performance for the binary vowels-vs-consonants classifier. Tables 5.13, 5.15, 5.14 present the confusion matrices of the 3-class classifier, the binary classifiers vowels-vs-consonant and music-vs-singing voice respectively for instance no3.

Table 5.13: Confusion matrices of instance no3 for 3-class classifiers, singing voice should be detected in all frames

Segmentation	real class	pred.as:music	pred.as:cons.	pred.as:vowel
120ms, overlapping	consonant	10	38	2
	vowel	12	114	22
120ms, non-overlapping	consonant	3	11	0
	vowel	4	29	4
30ms, non-overlapping	consonant	38	12	2
	vowel	138	10	1

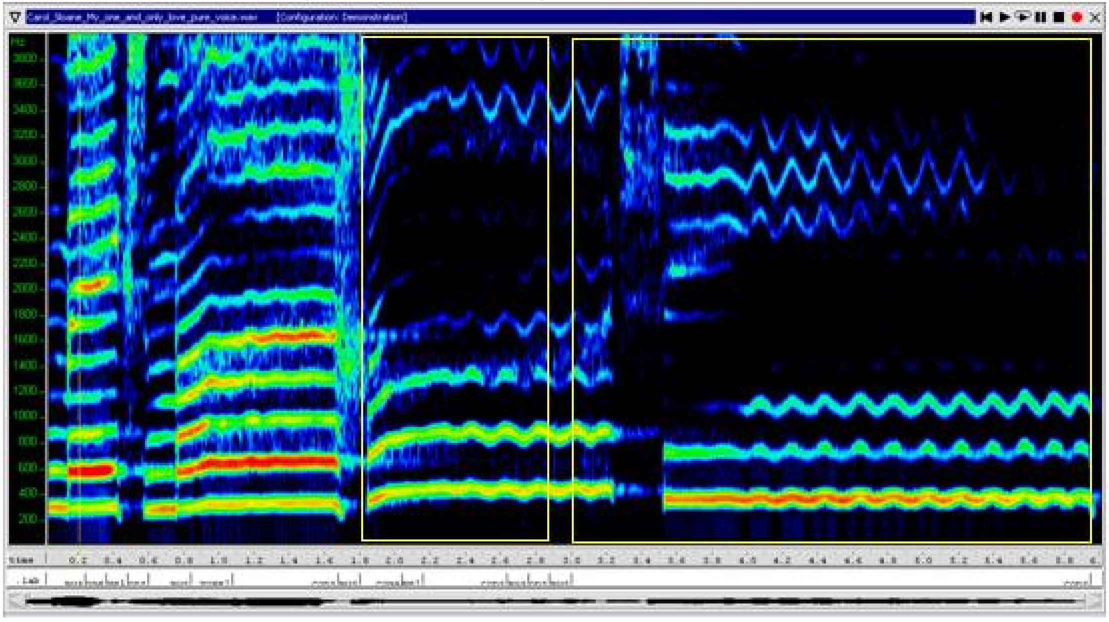
As we can see in Table 5.13 apart from the consonants that are correctly assigned as ones, the majority of the vowel instances are also assigned to the consonant class using 120ms segmentation. In Figure 5.4 we can see the spectrogram of this instance. The areas in yellow frames are two vowels with vibrato that the recognizer mis-classified as consonants using the 120 ms segmentation, probably because of the quick frequency modulation and thus quick spectral change it detects in these frames. Furthermore, for this evaluation we can see that different characteristics of the singing voice are captured by the models when instrumental background is present, that are not able to do so in the case of pure singing voice.

On the contrary, using the 30ms 3-class model, the same vowel frames are assigned to music. This also happens with the binary music-vs-singing voice models. This is caused probably because of the high harmonicity of the voice in this case.

Table 5.14: Confusion matrices of instance no3 for music/singing classifiers, singing voice should be detected in all frames

Segmentation	pred.as:music	pred.as:singing voice
120ms (non-overlap)	47	4
120ms (overlap)	183	15

Figure 5.4: Spectrogram of Instance no3



If we observe the class predictions for the binary vowels-vs-consonants classifier, we can see that most of the consonants are correctly identified, but many vowel frames are assigned to consonants. Those frames are again the vowels with high vibrato.

Table 5.15: Confusion matrix of instance no3 for vowels/consonants classifier

Segmentation	real class	pred.as:cons.	pred.as:vowel
30 ms	consonant	44	7
	vowel	77	73

5.2.4 Instance no4, singing voice with instrumental background, no percussion

This instance contains a female singing voice along with instrumental background and has no frames that contains only music, so all of the frames are either sung vowels or sung consonants. Tables 5.16, 5.18, 5.17 present the confusion matrices of the 3-class classifier, the binary classifiers vowels-vs-consonant and music-vs-singing voice respectively for instance no4. For the 3-class classifiers using 120 ms segmentation we can see that vowel frames are well recognized and the overall percentage of correctly classified instances is high enough. The 3-class model with the 30 ms segmentation mis-classifies some vowel and consonant instances to the music class, probably because frames are short.

Table 5.16: Confusion matrices of instance no4 for 3-class classifiers, singing voice should be detected in all frames

Segmentation	real class	pred.as:music	pred.as:cons.	pred.as:vowel
120ms, overlapping	consonant	4	13	30
	vowel	0	19	111
120ms, non-overlapping	consonant	2	4	5
	vowel	0	5	30
30ms, non-overlapping	consonant	10	19	15
	vowel	39	15	82

Table 5.17: Confusion matrix of instance no4 for vowels/consonants classifier

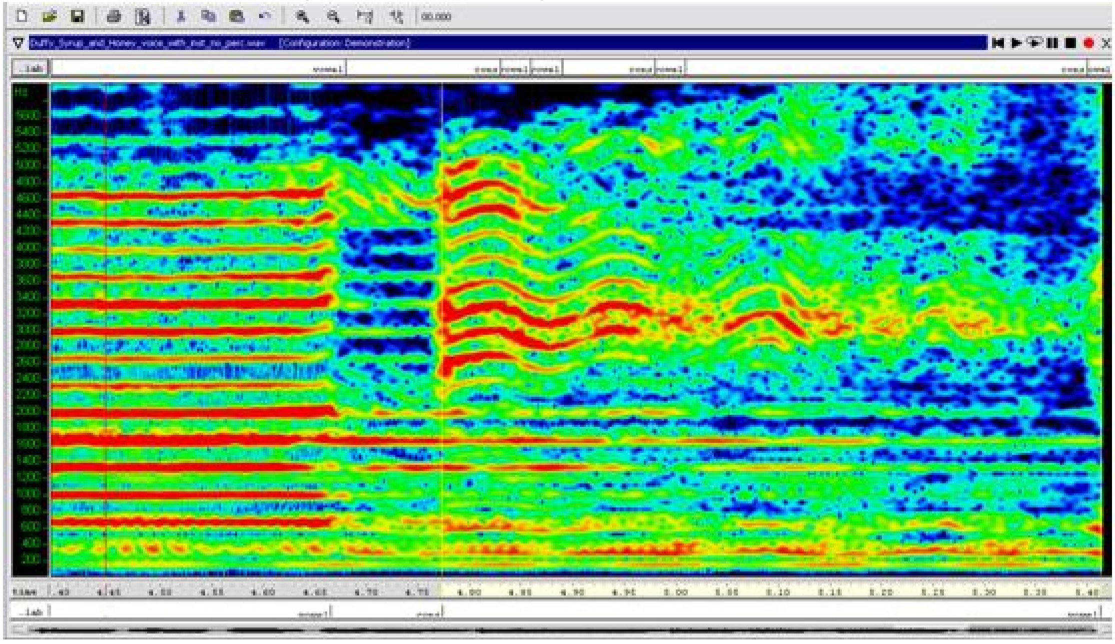
Segmentation	real class	pred.as:cons.	pred.as:vowel
30 ms	consonant	28	17
	vowel	36	100

Table 5.18: Confusion matrices of instance no4 for music/singing classifiers, singing voice should be detected in all frames

Segmentation	pred.as:music	pred.as:singing voice
120ms (non-overlap)	6	40
120ms (overlap)	24	153

The binary music-vs-singing voice classifiers achieves high accuracy. For the binary vowel-vs-consonant classifier, we can see from the confusion matrix that some vowel instances are assigned to the consonant class. Like in instance no3,

Figure 5.5: Spectrogram of Instance no4



vibrato exists in these frames. Figure 5.5 shows the spectrogram at these frames with the real class at the bottom and the predicted ones on the top.

5.2.5 Instance no5, singing voice with instrumental background and percussion

Instance no5 contains a female singing voice along with instrumental background with percussion. Also, contains frames only with music and no singing voice. This is the most common case in popular music. Tables 5.19, 5.20 and 5.21 present the confusion matrices of the 3-class classifier, the binary classifiers vowels-vs-consonant and music-vs-singing voice respectively for instance no5.

We can see that for the 3-class classification most of the vowels are correctly predicted, but some music instances are assigned to the consonant class. Figure 5.6 shows the spectrogram of a music snippet from Instance no5. The 3-class models mis-classified some of these frames as consonants, as it is shown in the transcription pane on the top of the figure. These frames contain high frequency

Table 5.19: Confusion matrices of instance no5 for 3-class classifiers

Segmentation	real class	pred.as:music	pred.as:cons.	pred.as:vowel
120ms, overlapping	music	24	19	10
	consonant	2	4	10
	vowel	0	4	53
120ms, non-overlapping	music	5	6	2
	consonant	1	0	3
	vowel	0	1	14
30ms, non-overlapping	music	21	20	12
	consonant	6	5	5
	vowel	3	0	58

Table 5.20: Confusion matrix of instance no5 for vowels/consonants classifier

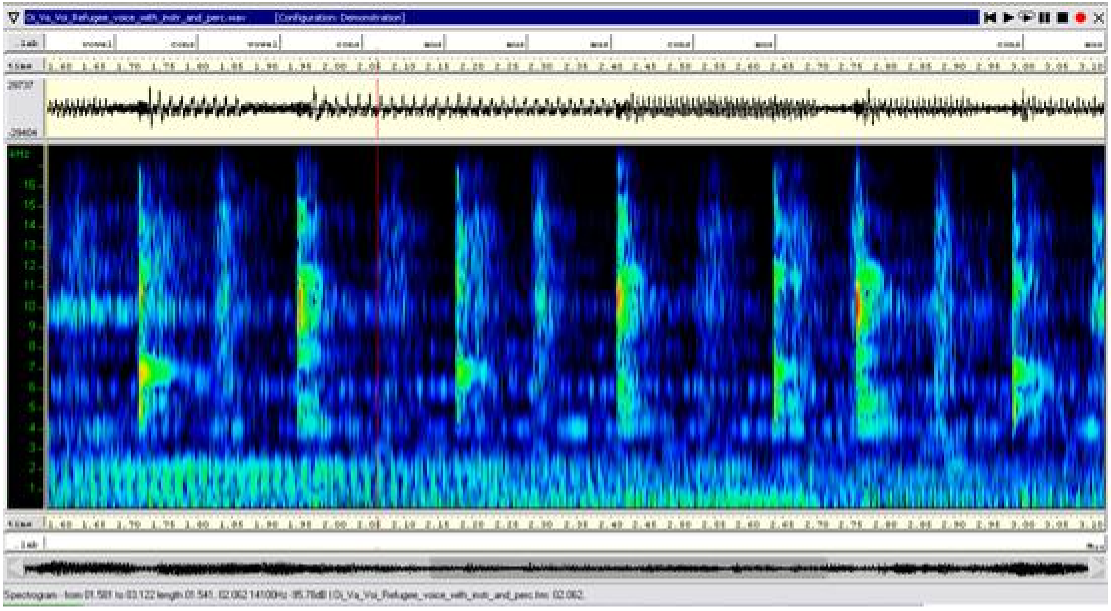
Segmentation	real class	pred.as:cons.	pred.as:vowel
30 ms	consonant	12	5
	vowel	7	52

Table 5.21: Confusion matrices of instance no5 for music/singing classifiers

Segmentation	real class	pred.as:music.	pred.as:singing voice
120ms, overlapping	music	21	32
	singing voice	2	81
120ms, non-overlapping	music	5	8
	singing voice	0	22

bell chime synth-effect sounds that are recognized by the models as consonants. The same instances are assigned to voice using the music-vs-singing voice models.

Figure 5.6: Spectrogram of Instance no5



Spectrogram - from 01.581 to 03.122 length 01.541, 02.062 14100Hz -95.78dB | Di_Va_Vai_Refugee_voice_with_instr_and_perc.hm: 02.062

5.3 Conclusions

In this Chapter we elaborated the experiments held during this thesis and the classification models obtained. The models that show the highest accuracy were evaluated with a set of different music instances in order to observe better the weaknesses found in each one. Especially, we found that models mis-classify as consonants music instances that have vibrato in the singing voice and also music frames that contain high pitched chimes. Also, we found that fricative consonants are very robustly detected by the models.

Chapter 6

Conclusion

6.1 Discussion

In this thesis we have developed statistical models that distinguish pure instrumental music from sung vowels and consonants. Our approach was based on fixed-length segmentation of the raw audio, followed by low-level feature extraction and classification. A manually annotated phoneme-based database was created and used as ground-truth for our experiments and also for training and testing our models.

From our experiments, we have found that if the segmentation is based on the annotations, feature extraction and segment-based classification achieves an average f-measure of 82.13% for the 3-class classification model. Further exploration of suitable fixed-length segmentation showed that performance drops and that different classes have different optimal segmentation length. Binary classifiers were built based on this optimal segmentation length for each class and gave an average f-measure of 76.88%. These models presented overlaps between the assignments in different classes that were difficult to overcome using simple heuristic rules. At that point, other classification schemes were considered as possibly more efficient, especially two chained binary classifiers, one to distinguish between music and singing voice and a second one to assign the singing voice segments into sung vowels or consonants. Afterwards, a larger set of features was extracted and used to train 3-class classification models and the binary music-vs-singing voice and vowels-vs-consonants classifiers. A 9.38% increase in accuracy was achieved in the 3-class classification models (77.18% average f-measure). For the binary classifiers we found that a 120ms segmentation is optimal for distinguishing between music and singing voice (88.8% average f-measure) while the 30ms segmentation was optimal for distinguishing between sung vowels and consonants (82.9% average f-measure). From these experiments we could observe that suitable fixed-length segmentation can yield accuracy close to the one achieved based on annotation-based segmentation.

From the evaluation procedure performed, the results showed close performance achieved with that of a 10-fold cross validation of the testing set. Also, some confusions of the system came up, like in the case of vibrato in the singing voice or in the case of music containing high-pitched chimes, when frames are assigned to the consonant class. Furthermore, the models were not able to distinguish robustly pure singing voice segments (without any instrumental background) as they were not trained to do so. It was also found that fricatives and pure instrumental music without percussion are very robustly detected by the models.

With the current configurations, the models developed could be used in any polyphonic audio signal with English lyrics in order to distinguish between the 3 classes in question but also the binary models could be used to distinguish singing voice from pure instrumental music and also to distinguish sung vowels from consonants in singing voice segments.

6.2 Future Work

Several improvements could extend this study. First, a larger training set could yield better results, in order to overcome confusions like i.e. in the case of vibrato, which was probably observed because not many instances of vibrato voice where included in the training set. In addition, different segmentation algorithms could be applied and tested for their effectiveness in this task. For example, for the singing voice discrimination beat-length segmentation could probably yield better results.

Furthermore, pre-processing techniques could be applied to the raw audio in order to enhance the singing voice signal, like for example a band-pass filter, which allows the vocal regions to pass through while attenuating other frequency regions.

Also, the output of the models could be aligned with the corresponding lyrics in order to improve the accuracy and reduce classification errors. As shown from the evaluation of the models, fricatives are more reliable than other consonants in their classification and several post-processing rules could take advantage of this fact if the corresponding text is given. Such rules for example, taking advantage of fricatives reliability were developed in (Loscos *et al.*, 1999).

Furthermore, different sets of extracted features can be tested for their ability to distinguish between the different classes along with different classification algorithms and their parametrization.

It would be interesting to test the robustness of the models in other languages apart from English, that share a common set of phonemes. One interesting topic for research, given a robust classifier to detect vowels, consonants and music is presented in (Patel *et al.*, 2006). The authors compare the rhythm and melody between British and French for both speech and music. Their hypothesis is that music reflects prosodic patterns in the composer's native language and although they are interested in the case of pure instrumental music they state that "It might not be surprising if vocal music reflected speech prosody; after all, such music must adapt itself to the rhythmic and melodic properties of a text". For the case of speech, authors propose the use of an index that measures the durational contrast between successive vowels. Such an index could also be applied for the case of singing voice, given a robust vowel detector.

6.3 Contributions

In this thesis we developed several classification schemes in order to distinguish singing vowel sounds from singing consonant sounds and pure instrumental music. The models were trained using the boundaries assigned by the annotation process and using several segmentation lengths. Evaluation on a set of unknown to the system files, using simple fixed-frame segmentation showed that the models achieve good performance.

Especially, this study showed that even without any preprocessing of the audio signals and extracting only low-level features, 3-class models are capable to distinguish between the different classes with an average error-rate of 23%. Also, the several binary models give greater flexibility, as they can be tested and applied replacing for instance the music vs singing voice model with another one developed or applying the vowel vs consonant classifier to singing voice segments.

As we saw in the literature review, many approaches, in order to perform singing voice enhancement, pre-process the signals using harmonic analysis and omit unvoiced sounds or consonants. Also, many approaches perform singing phoneme recognition in segments that are a priori manually classified as containing singing voice. In contrast, in this study music is considered as a separate class in the models and no a priori segmentation of the audio files is needed.

To our knowledge, for the singing voice case in polyphonic recordings, no relevant research exists that considers consonants as a class. This classification can be used as a first classification step in a singing phoneme recognition system in order to first classify sounds in groups. In a relevant study in speech recognition by (K.Driaunys *et al.*, 2005) the authors begin from recognizing groups of phonemes and then try to recognize each phoneme itself. They state "Our experience showed that direct phoneme classification can't provide good enough recognition results. As were observed in some of our later investigations it could be meaningful prior to direct phoneme classification perform phoneme classification into some of phoneme groups or some super-classes using group features of phonemes." Thus, such a classification scheme could be further extended, building one-phoneme recognizers to recognize each phoneme exactly inside this group using additional features or additional classification methods.

Appendix A

List of Songs

Air- Playground Love (146-161 sec)
Amy Winehouse- You know that I am no good (198.8-213.8 sec)
Ella Fitzerald- Sunshine (13.43-28.62 sec)
Faith Hill- Red Umbrella (49.58-64.58 sec)
Frank Sinatra- Fly me to the moon (8-13 sec)
James Brown- I feel good (21-36 sec)
Madonna- Frozen (17.518-32.385 sec)
Nina Simone- My baby just cares for me (155-170 sec)
2Pac- California Love (160-175 sec)
Robbie Williams- Rock DJ (49-64 sec)
Aretha Franklin- Rescue Me (15-30 sec)
Lamb- Gabriel (9-24 sec)
Gary Moore- Still got the blues (75-90 sec)
Green Day- She (27-42 sec)
Michael Bolton- How can we be lovers (56-71 sec)

Appendix B

Glossary

ASR	Automatic Speech Recognition
MFCC	Mel-Frequency Cepstral Coefficient
MLP	Multi-Layer Perceptron
SVM	Support Vector Machine
GMM	Gaussian Mixture Model
SMO	Sequential Minimal Optimization
KNN	K-Nearest Neighbors
HMM	Hidden Markov Model
LPC	Linear Predictive Coding
LFPC	Log-Frequency Power Coefficients
PLPCs	Perceptual Linear Prediction Coefficients
HC	Harmonic Coefficient
DTW	Dynamic Time Warping
ZCR	Zero-Crossing Rate
FT	Fourier Transform
FFT	Fast Fourier Transform
DFT	Discrete Fourier Transform
STFT	Short-time Fourier Transform
DCT	Discrete Cosine Transform
DP	Dynamic Programming
FSA	Finite State Automaton

References

Berenzweig, A. and Ellis, D. (2001). Locating singing voice segments within music signals. *Applications of Signal Processing to Audio and Acoustics, 2001 IEEE Workshop on the*, pages 119–123. 25, 26, 27

Berenzweig, A., Ellis, D., and Lawrence, S. (2002). Using voice segments to improve artist classification of music. In *AES 22nd International Conference on Virtual, Synthetic and Entertainment Audio*. 24, 25, 26

Chou, W. and Gu, L. (2001). Robust singing detection in speech/music discriminator design. 25, 26, 27

Fujihara, H., Goto, M., Ogata, J., Komatani, K., Ogata, T., and Okuno, H. (Dec. 2006). Automatic synchronization between lyrics and music cd recordings based on viterbi alignment of segregated vocal signals. *Multimedia, 2006.ISM'06.Eighth IEEE International Symposium on*, pages 257–264. 29

Garofolo, J., Lamel, L., Fisher, W., Fiscus, J., Pallett, D., Dahlgren, N., , and Zue, V. (1993). Timit acoustic-phonetic continuous speech corpus. Linguistic Data Consortium, Philadelphia. 33

Gerhard, D. B. (2003). *Computationally measurable differences between speech and song*. Ph.D. thesis, Simon Fraser University, Burnaby, BC, Canada, Canada. Adviser-Fred Popowich. 11

Gruhne, M., Schmidt, K., and Dittmar, C. (2007). Phoneme recognition in popular music. *Fraunhofer IDMT*. 3, 28, 29

Hnsler, E. and Schmidt, G. (2006). *Topics in Acoustic Echo and Noise Control Selected Methods for the Cancellation of Acoustical Echoes, the Reduction of Background Noise, and Speech Processing*, chapter 12. Springer.

Hu, G. and Wang, D. (Feb. 2007). Auditory segmentation based on onset and offset analysis. *Audio, Speech, and Language Processing, IEEE Transactions on [see also Speech and Audio Processing, IEEE Transactions on]*, **15**(2), 396–405.

Hu, G. and Wang, D. (Sept. 2004). Monaural speech segregation based on pitch tracking and amplitude modulation. *Neural Networks, IEEE Transactions on*, **15**(5), 1135–1150.

Hua, X., Lu, L., and Zhang, H. (2004). P-karaoke: personalized karaoke system. In *MULTIMEDIA '04: Proceedings of the 12th annual ACM international conference on Multimedia*, pages 172–173, New York, NY, USA. ACM.

Iskandar, D., Wang, Y., Kan, M., and Li, H. (2006). Syllabic level automatic synchronization of music signals and text lyrics. In *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia*, pages 659–662, New York, NY, USA. ACM. 3, 29

Janer, J. and de Boer, M. (2008). Extending voice-driven synthesis to audio mosaicing. In *5th Sound and Music Computing Conference*, Berlin. 4

Kan, M.-Y., Wang, Y., Iskandar, D., New, T. L., and Shenoy, A. (2008). Lyrically: Automatic synchronization of textual lyrics to acoustic music signals. *Audio, Speech, and Language Processing, IEEE Transactions on*, **16**(2), 338–349. 3, 28

K.Driaunys, V.Rudionis, and P.vinys (2005). Analysis of vocal phonemes and fricative consonant discrimination based on phonetic acoustics features. Information Technology And Control, Kaunas, Technologija. 62

Kim, Y. (2001). Excitation codebook design for coding of the singing voice. *Applications of Signal Processing to Audio and Acoustics, 2001 IEEE Workshop on the*, pages 155–158. 11

Kim, Y. (2002). Singer identification in popular music recordings using voice coding features. In *In Proceedings of the 3rd International Conference on Music Information Retrieval*, pages 13–172. 24, 25, 26

Knees, P., Schedl, M., and Widmer, G. (2005). Multiple lyrics alignment: Automatic retrieval of song lyrics.

Korst, J. and Geleijnse, G. (2006). Efficient lyrics retrieval and alignment. *Philips Research Laboratories*, page 205.

Kothmeier, A. (2006). *Automatic Audio and Lyrics Alignment*. Master's thesis, Johannes Kepler University Linz.

Ladefoged, P. (2005). *Vowels and Consonants*. Blackwell Publishing, second edition. 13, 33, 35

Laurier, C., Grivolla, J., and Herrera, P. (2008). Multimodal music mood classification using audio and lyricse. *International Conference on Machine Learning and Applications*. 3

Lazier, A. and Cook, P. (2003). Mosevius: Feature driven interactive audio mosaicing. In *DAFX-03*. 4

Li, T. and Ogihara, M. (June 2006). Toward intelligent music information retrieval. *Multimedia, IEEE Transactions on*, **8**(3), 564–574. 3

Li, T., Ogihara, M., and Zhu, S. (Dec. 2006). Integrating features from different sources for music information retrieval. *Data Mining, 2006.ICDM '06.Sixth International Conference on*, pages 372–381. 3

Li, Y. and Wang, D. (May 2007). Separation of singing voice from music accompaniment for monaural recordings. *Audio, Speech, and Language Processing, IEEE Transactions on [see also Speech and Audio Processing, IEEE Transactions on]*, **15**(4), 1475–1487. 27, 40

Logan, B., Kositsky, A., and Moreno, P. (2004). Semantic analysis of song lyrics. *Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference on*, **2**, 827–830 Vol.2. 3

Loscos, A., Cano, P., and Bonada, J. (1999). Low-delay singing voice alignment to text. *ICMC*. 11, 27, 61

Lu, L. and Hanjalic, A. (Jan. 2008). Audio keywords discovery for text-like audio content analysis and retrieval. *Multimedia, IEEE Transactions on*, **10**(1), 74–85.

Maddage, N. C., Xu, C., and Wang, Y. (2003). An svm-based classification approach to musical audio. In *ISMIR*. 25, 26

Maddage, N. C., Wan, K., Xu, C., and Wang, Y. (2004). Singing voice detection using twice-iterated composite fourier transform. *Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference on*, **2**, 1347–1350 Vol2. 25, 26, 27, 40

Maddage, N. C., Kankanhalli, M. S., and Li, H. Z. (2008). Effectiveness of signal segmentation for music content representation. *International Conference on Multimedia Modeling*. 40

Mahedero, J., Martinez, A., Cano, P., Koppenberger, M., and Gouyon, F. (2005). Natural language processing of lyrics. In *MULTIMEDIA '05: Proceedings of the 13th annual ACM international conference on Multimedia*, pages 475–478, New York, NY, USA. ACM. 3

Michael W. Macon, Leslie Jensen-link, J. O. M. A. C. E. B. G. (1997). Concatenation-based midi-to-singing voice synthesis. 103rd AES Meeting. 11

Muller, M., Kurth, F., Damm, D., Fremerey, C., and Clausen, M. (2007). Lyrics-based audio retrieval and multimodal navigation in music collections. *Research and Advanced Technology for Digital Libraries*, pages 112–123. 3

Nwe, T. L., Shenoy, A., and Wang, Y. (2004). Singing voice detection in popular music. In *MULTIMEDIA '04: Proceedings of the 12th annual ACM international conference on Multimedia*, pages 324–327, New York, NY, USA. ACM. 25, 26, 27, 29, 40

Orio, N. (2006). Music retrieval: a tutorial and review. Foundations and Trends in Information Retrieval. 6, 7

Owens, E., Talbott, C., and Schubert, E. (1968). Vowel discrimination of hearing-impaired listeners. *Journal of speech and hearing research*, pages 648–655. 4

Patel, A. D., Iversen, J. R., and Rosenberg, J. C. (2006). Comparing the rhythm and melody of speech and music: The case of british english and french. *The Journal of the Acoustical Society of America*, **119**(5), 3034–3047. 61

Peeters, G. (2003). A large set of audio features for sound description (similarity and classification) in the cuidado project. 18

Pellegrino, F. (1999). An unsupervised approach to language identification. In *Acoustics, Speech, and Signal Processing, 1999. ICASSP '99. Proceedings., 1999 IEEE International Conference on*, volume 2, pages 833–836 vol.2.

Pool, I. (2002). *Investigation of the impact of High Frequency transmitted speech on Speaker Recognition*. Master's thesis, University of Stellenbosch. 16

Rocamora, M. and Herrera, P. (2007). Comparing audio descriptors for singing voice detection in music audio files. In *Brazilian Symposium on Computer Music, 11th. San Pablo, Brazil*. 26, 27

Sasou, A., Goto, M., Hayamizu, S., and Tanaka, K. (2005). An auto-regressive, non-stationary excited signal parameter estimation method and an evaluation of a singing-voice recognition. *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, **1**, 237–240. 28

Serra, X. (2008). Sound and music computing research: Trends and challenges. SMC Network. 7

Shenoy, A., Wu, Y., and Wang, Y. (2005). Singing voice detection for karaoke application. In *Visual Communications and Image Processing 2005. Edited by Li, Shipeng; Pereira, Fernando; Shum, Heung-Yeung; Tescher, Andrew G. Proceedings of the SPIE, Volume 5960, pp. 752-762 (2005).*, volume 5960 of *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference*, pages 752–762. 3, 25, 26, 27, 40

Sjlander, K. and Beskow, J. (2000). Wavesurfer-an open source speech tool. In *ICSLP-2000*. 33

Smit, C. (2007). Solo voice detection via optimal cancellation. In *Applications of Signal Processing to Audio and Acoustics, 2007 IEEE Workshop on*, pages 207–210.

Sundberg, J. (1987). *The Science of the Singing Voice*. Northern Illinois University Press. vii, 9, 10, 11, 12

Suzuki, M., Hosoya, T., Ito, A., and Makino, S. (2007). Music information retrieval from a singing voice using lyrics and melody information. *EURASIP J.Appl.Signal Process.*, **2007**(1), 151–151. 3, 24, 25, 28

Swaminathan, K. and Doddihal, V. (10-14 Jan. 2007). Audio segmentation assisted synchronized lyrics editing for ce devices. *Consumer Electronics, 2007.ICCE 2007.Digest of Technical Papers.International Conference on*, pages 1–2.

Taniguchi, T., Tohyama, M., and Shirai, K. (2008). Detection of speech and music based on spectral tracking. *Speech Communication,*, **50**(7), 547–563.

Tsai, W., Rodgers, D., and Wang, H. (2004). Blind clustering of popular music recordings based on singer voice characteristics. *Comput. Music J.*, **28**(3), 68–78. 25, 26

Tsai, W.-H. and Wang, H.-M. (2006). Automatic singer recognition of popular music recordings via estimation and modeling of solo vocal signals. *IEEE Transactions on Audio, Speech, and Language Processing*, **14**(1), 330–341. 24, 26, 27

Tzanetakis, G. (2004). Song-specific bootstrapping of singing voice structure. *Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference on*, **3**, 2027–2030 Vol.3. 26, 27

Wong, C. H., Szeto, W. M., and ., K. H. W. (2007). Automatic lyrics alignment for cantonese popular music. *Multimedia Systems*, **12**(4-5), 307–323. 3, 29

Yaguchi, Y. and Oka, R. (2005). Song wave retrieval based on frame-wise phoneme recognition. *Information Retrieval Technology*, pages 503–509. 28

Zhang, T. (2003). System and method for automatic singer identification. 24, 25, 26

Zhu, Y., Chen, K., and Sun, Q. (2005). Multimodal content-based structure analysis of karaoke music. In *MULTIMEDIA '05: Proceedings of the 13th annual ACM international conference on Multimedia*, pages 638–647, New York, NY, USA. ACM. 3