New approaches for Rhythmic Description of Audio Signals

Enric Guaus i Termens

Research work

PhD in Computer Science and Digital Communication

Director: Dr. Xavier Serra

Pompeu Fabra University

September 2004

Abstract

The subject of this work is the study of how rhythmical description of music can be included in Music Information Retrieval works, but from a more musical point of view. The goal is to provide a suitable representation of rhythm and explore how useful it can be for many different applications. This new representation is based on the so called *Rhythm Transform* that transforms data from the time domain to a so called *rhythm domain*. This transformation has proved to give good results for automatic classification and similarity applications. The theoretical and computational issues of the Rhythm Transform are explained, and some results for specific applications are also shown.

Acknowledgements

I would like to thank all the people of the Music Technology Group and specially Xavier Serra, Eloi Batlle, Vadim Tarasov, Jaume Masip, Jose Pedro Garcia, Rosamerica Urtasun, Teresa Carrasco, Karin Dressler, Perfecto Herrera, Emilia Gomez, Pedro Cano, Fabien Gouyon, Joana, Cristina, Ramon, Marteen and many others.

In addition, i would like to thank all my friends.

Finally, I would like to thank all my family, specially my grandparents Magda, Jaume, Manel and above all Dolors.

And Paola.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Music Content Processing

Let us imagine that we are in a CD store. Our decision to buy a specific CD will depend on many different aspects like genre, danceability, instrumentation, etc. Basically, the information we have is limited to the genre, artist and album, but sometimes this information is not enough to take the correct decision. Then, it would be useful to retrieve music according to different aspects on its content but, what is the *content*?

The word *content* is defined as "*the ideas that are contained in a piece of writing, a speech or a film*" [1]. This concept applied to a piece of music can be seen as the implicit information that is related to this piece and that is represented in the piece itself. Aspects to be included inside this concept are, for example, structural aspects, rhythmic, instrumental, and melodic characteristics of the piece.

The concept of *content-analysis* is defined as the "*analysis of the manifest and latent content of a body of communicated material (as a book or film) through a classification, tabulation, and evaluation of its key symbols and themes in order to ascertain its meaning and probable effect*" [2]. Several techniques are included under the concept of *"Music-content analysis"*, as techniques for automatic transcription, rhythm and melodic characterization, instrument recognition and genre classification; that is, the techniques intended to describe any aspect related to the content of music.

Music Content Processing is a topic of research that has become very relevant in the last few years. The main reason for this is that a great amount of audio material has been made accessible to the home user through networks and other storage supports. This fact makes it necessary to develop tools intended to interact with this audio material in an easy and meaningful way. Many researchers are currently studying and developing techniques aimed at automatically describe and deal with audio data in a meaningful way. There are many disciplines involved in this issue, as signal processing, musicology, psychoacoustics, computer music, statistics and information retrieval.

1.2 Rhythmic Description

'Music is organization'. This is a debatable sentence, we know, but this is the main idea that justifies lots of studies about music during the last centuries. It is well known that a sound can be defined by using four features: frequency, timbre, duration and intensity. Then, the '*Music is organization'* sentence could be extended into these four dimensions.

For instance, if we look at the frequency organization through time, we can study melodies, while if we look at the frequency distribution in a specific time, we can study harmony (if the frequencies belong to different notes) or timbre (if frequencies belong to a unique note). This simple example shows that multiple relationships can be established between those four basic features.

Most of the musical elements can be defined in this way. But the main problem we found is that these relationships are not so simple. For instance, what is Rhythm? In a first approach for defining rhythm, one can think that *Rhythm is the time evolution of intensity*. It is true, but it is not all the truth. What is about harmonic rhythm? What is about rhythm defined by different textures of a sound? Therefore, this is an incomplete definition.

Roughly speaking, rhythm can be defined as the time organization of any aspect of music. But there is no universally accepted definition of rhythm [36]. To avoid confusion, we will define rhythm as the *temporal and accentual patterning of sound*, and *accentual* means that sound is perceptually salient in some way (melody, harmony, intensity accent, etc.).

According to Seashore in [11], there are two fundamental factors in the perception of rhythm: an instinctive tendency to group impressions in hearing and a capacity for doing this with precision in time.

From a perceptual point of view, we can assume that there exists a relationship between sound and image cognition processes. If we analyze the Kanizsa's triangle shown in Fig.1.1, we will see a group of three black circles, with a black contour triangle at the middle of them, and another white triangle in the top of the picture. But it is not objectively true: the figure is only formed by three incomplete circles and six lines. The cognitive process of audio can be seen in a similar way, that is grouping elements (incomplete circles and lines) and arranging them (circles and triangles). If we listen to a typical clock's sound, we will group the impulsive noise as a *tic-tac* sound. This grouping is perfectly defined in time in such a way that we can predict when the next *tic-tac* will sound.

This is just a simple example that shows us how complex a perceptual process can be. Neither musical analysis nor rhythm analysis are a simple tasks. Neither for rhythm. If this kind of analysis is not evident for humans, neither for computers. One of the main goals of this study deals with the rhythmical description of music, a rhythmical description that should reflect all these perceptual aspects of music. And this is not an easy task.

1.3 Motivation

Computer music community is a relative small group in the big computer science field. Most of the people in this small group is a great enthusiast of music. The problem arises when computers meet music. Sometimes, this world of num-

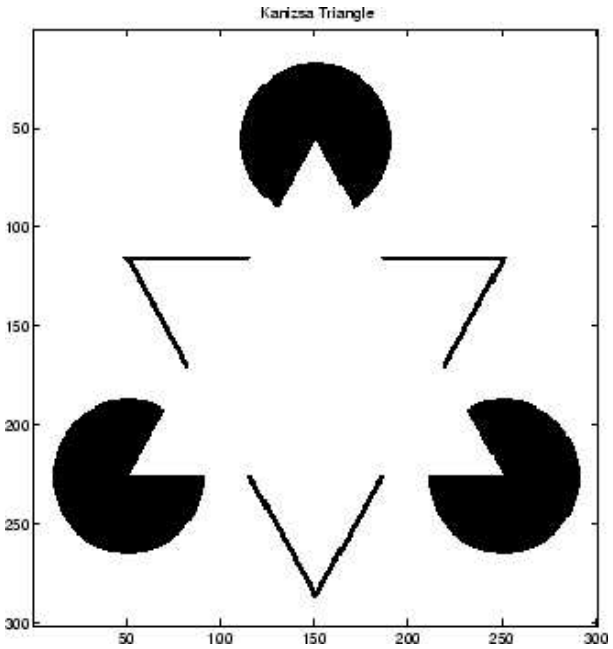


Figure 1.1: Illusory Contours

bers, probabilities and sinusoids, everything about music can be forgotten: final applications are really far from musicians or music enthusiasts requirements.

Our research into the Music Information Retrieval field tries to join these two worlds, but sometimes it becomes a difficult task. More research is needed, and this research should focus on different aspects of music:

- Objective description of music, where BPM detection or melody extraction algorithms have to be developed and improved.
- Musicology description of music, where formal studies in a high level structure allow to distinguish between a Jazz solo and a Opera overture.
- Psychological aspects of music, that is, how the different musical stimulus affect to the human behavior

All these (and probably more) aspects of music have to be taken into account when managing music. For instance, lots of successful studies about BPM detection have been done, but this information could be completely useless if it is not completed with other information, i.e. whether the song has *swing* or not.

1.4 Objectives

The goal of this research work is to include the rhythmic aspects of music in the Music Information Retrieval field providing it a more human (or musical) point of view.

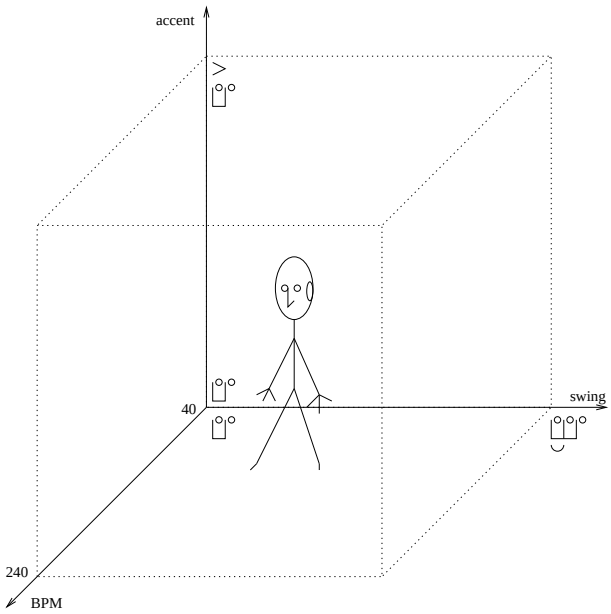


Figure 1.2: Symbolic representation of a possible rhythm browser

On a first approach, *Query by tapping* (QBT) systems show to be good rhythmical interfaces for music information retrieval. A QBT system is a multimedia database containing rhythmical information of a set of audio files, with a set of input/output interfaces that allows the user to browse and find the correct audio file, under a specific criterion. Some successful implementations of QBH systems are described in [38][21][37], but all these kind of systems are subject to many kinds of restrictions (monophonic input audio, little databases, etc.).

By using the rhythmical properties of music described in this research, the QBH system should be able to find the correct audio file without those limitations, that is, from a simple audio file, from a commercial CD or from real-time radio stream. Furthermore, rhythm browsing will be possible (symbolic representation of a possible rhythm browser is shown in Fig. 1.2).

The research work can be divided into different steps:

- General Overview
  - Study how rhythm is represented in the literature.
  - Review the different techniques and algorithms that have been used to extract this representation.
  - Review the application contexts in which rhythm description is needed.
- Review of the different existing tools that can help us in our objective
  - different high-level and low-level descriptors
  - different statistical techniques for dealing with descriptors
- Test of the proposed rhythmical description in different applications
  - Speech-Music discriminator

- Automatic Genre Classification
- Measure of rhythmic similarity

1.5 Scope

This research work is organized as follows. In Chapter 2 a general description of rhythm is shown. Different kinds of rhythm and different levels of the rhythm concepts are discussed. In Chapter 3, diffenent points of view about Genre are introduced, and some reflexions about the different taxonomies are also presented. In chapter 4, a review of theoretical concepts and some previous work in MIR field are commented. All these theoretical concepts will be applied in Chapter 5, where my first contributions to the community are presented. Finally, in Chapter 6, conclusions of our research and future work are shown.

Chapter 2

Rhythm Description

2.1 Historical Review

In the beginning, it was rhythm. Most of the musicians (composers, performers, musicologists, sonologists ...) will agree with this sentence. It is well known that rhythm is one of the most complex elements referred to the music. There are lots of books about Harmony, Counterpoint and so on, but what's about rhythm? There are several courses of Harmony and Counterpoint in the conservatories, but what's about rhythm? A correct knowledge of rhythm is crucial for composers and performers, but it is not until the 20th century that quite detailed studies about rhythm appear [14].

In Table 2.1 some important studies about rhythm are shown. Note that most of them have been developed in the 20th. Century [58]

Partly, this lack of studies about rhythm have been produced due to a non-effective definition of all the concepts involved in rhythm, and it produces ambiguities in the terminology.

But in the last few years, lots of studies about rhythm have emerged from all the universities and research centers all over the world. Depending on the point of view, rhythm studies can be divided into three main groups:

Theoretical analysis: They are involved in the theories about rhythmic structures of music from all the centuries and countries. Some important contributions can be found in [14][58][76].

Psychological studies: They are based on the concept that rhythm does not refer to the rigid properties of the different notation systems, but on their perceptual aspects. Some successful works can be found in [35][25].

Computational analysis and modeling: These studies involves all the automatic rhythm identification systems (i.e. meter detection, BPM detection...) and such different modeling tools developed for many different applications. Some important studies in both fields can be found in [24][22].

Our rhythm-related work belongs to the third group commented above. In the next sections, some definitions and structural aspects of rhythm will be explained for a further understanding of the applied techniques.

Context	Author	Title	Year
General	A. Ruckmich	A Bibliography of Rhythm	1913
General	T. Wiehmayer	Musikalische Rhythmik und Metrik	1917
General	O. Bie	Rhythm	1925
General	G. Becking	Der musikalische Rhythmus als Erkenntnisquelle	1928
Antiquity & M. Ages	A. Quintilianus	Peri mousik ês [On music]	1983
Antiquity & M. Ages	J. de Garlandia	Ars rithmica, De mensurabili musica	1974
Antiquity & M. Ages	R. Westphal	Griechische Rhythmik und Harmonik nebst de Geschichtr der drei musicalischen Disziplinen	1867
Antiquity & M. Ages	R. Westphal	Aristoxenos von Tarent: Melik und Rhythmik des classischen Hellenentums	1883
Classical & Romantic	J. Riepel	Anfangsgründe zur musikalischen Setzkunst	1752
Classical & Romantic	H. Koch	Versuch einer Anleitung zur Composition	1983
Classical & Romantic	J.Momigny	Cours complet d'harmonie et de composition	1803
Tonal rhythm	L. Klages	Vom Wesen des Rhythmus	1934
Post-Tonal rhythm	O. Messiaen	Technique de mon langage musical	1944
Psychological	W. James	The Principles of Psychology	1890
Metre and Accent	E. Levy	Von der Synkope	1933

Table 2.1: Some important studies about rhythm

2.2 Subjective Rhythm vs. Objective Rhythm

In Chapter 1, the concept of grouping equal sounds is introduced. This process is part of the *subjective rhythm* because it is an intrinsic behavior in humans[11]. On the other hand, the *objective rhythm* appears when this grouping process is made by sounds, i. e. one sound in a group of four is louder, or one sound in a group of three has a higher fundamental frequency.

In opposition to what we might think, subjective rhythm is more fundamental than objective rhythm. Since subjective rhythm is strongly related to perception, different interpretations of the same rhythmic sequence can be obtained from different tests. Referring to the clock example mentioned in Chapter 1, a clock noise can be percieved as a *tic tac* sequence or as a *tic tac, tec tac* sequence depending on the perception of the listener. Furthermore, the perception of equal quarter notes played by different instruments could produce different sensations, that is fast, slow, soft ... These differences depend on many biological, social and cultural environments of individuals. In this case, the objective rhythm can only focus the perceived rhythm into a specific way. This could be the case if the clock were spoilt or if some of those quarter notes were accentuated.

Figure 2.1: Human Perception of Beats

2.3 Architectural levels

The understanding of the architecture of rhythm is fundamental for a full comprehension of it. How is Rhythm organized?

Taking literature as an example, letters gather forming words, words gather creating phrases, phrases gather forming paragraphs, and so on until a whole book is written. In music, rhythm follows a similar structure: It is organized into different levels, from the lowest level to the highest level. This does not mean that low levels are less important than high ones. Is it possible to write a book without a good organization of letters? Of course, it is not.

In music, this kind of structural organization is not exclusive for rhythm: harmony and melody follow the same pattern. It is important to note how these different levels are not independent: the individual parts at the low levels of the structure are joined in a specific way in order to create the high levels.

Which are these rhythmical levels? As shown in Sec. 2.5, many rhythmical concepts can be studied and classified at different structural levels. For simplicity, three architectural levels structure will be defined, and most of those concepts can be fitted into one of those three levels.

2.3.1 Beat

Beat is the at lowest level of the structure. It is also called the *rhythmic primary level*. *Beat* is the basic unit, and it can be defined as each equivalent perceived stimuli. As mentioned in Sec. 2.2, the perceived stimuli is the most important one. Let us imagine a sequence of n identical beats. If a gap occurs at one of its supposed position, human tends to perceive that *silence* or *beat lack* as another *normal* beat (see Fig. 2.1).

The brain tends to follow a periodic structure although the physical stimuli is missing[11]. In fact, composers and performers search different ways to brake this perceived periodicity by using anticipations, delays, syncopation.... These aperiodicities are opposite to the brain operation. How does the brain respond to that confrontation?

When beats are not exactly periodic and the brain is cheated, this level of the rhythmical architecture is divided into two different sub-layers:

lower primary layer: The non-periodic beats are not interpreted as beats. In a limit situation, these beats can be so impredictible than they can be interpreted as noise.

higher primary layer: The repeated aperiodicity is taken as a new beat frequency with all the beat characteristics described above, that is, that they can not build a higher structural level. More information (i.e. the accents) is needed.

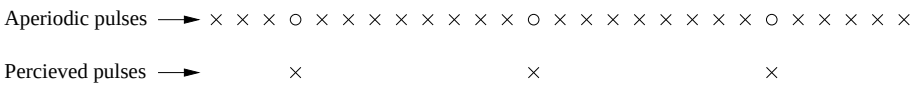


Figure 2.2: Human Perception of aperiodic pulses

	Simple	Compound
Duple	$\frac{2}{4}$ or $\frac{4}{4}$	$\frac{6}{8}$ or $\frac{12}{8}$
Triple	$\frac{3}{4}$	$\frac{6}{8}$

Table 2.2: Typical architectural subdivisions of meter

A train noise or a periodic bass drum over a continuous aperiodic snare can be good examples for this situation (See Fig. 2.2).

2.3.2 Meter

At the mid-level of the rhythmical architecture, *Meter* is defined. It is defined as the measure of the pulses between musical accents. Note how these accents are not exclusively *loudness accents*. There are different kinds of accents (see Sec. 2.5 for details):

Intensity accent is the most evident for defining meter and it can be defined as the beat that is louder than the mean.

Melodic accent is produced when a single note (or a little group of notes) is so far than the others.

Timbre accent can be defined when new instruments appear periodically.

Most of the times, the perceived accent is a combination of all of them. For instance, fingered bass in a Jazz piece, loudness (the intensity grows up fastly) melodic (really low frequencies) and timbre (quite different than the brass section) accents are combined.

At his point, the perceived beat and the perceived accent are combined to define the meter. The basic unit in the Meter is the bar where strong and weak beats are strictly defined. According to the standard definitions, meter can be *simple* or *compound*, *duple* or *triple*. Then, meter can also be classified into different sub-layers. The most common subdivision is shown in Tab. 2.2.

Simple meters imply a beat grouping by twos at the low sub-level of the meter architectural layer, while compound meters imply a beat grouping by threes at the same level. On the other hand, duple meters also imply a beat grouping by twos but at the highest sub-level of the meter architectural level. Finally, triple meters imply a beat grouping by threes at this level. For instance, simple-triple meter $\frac{3}{4}$ can be seen as three units (that is triple) of a (simple) $\frac{2}{8}$ meter. In opposition of that, a compound-triple meter $\frac{6}{8}$ can be seen as two units (that is duple) of a (triple) $\frac{3}{8}$ meter. Note how the length of the defined bar is the same for both cases (3 quarter notes), but doesn't for the accents.

2.3.3 Rhythm

In Chapter 1, a little introduction about rhythm is presented. After some explanations, a non exhaustive definition of rhythm is given as the *temporal and accentual patterning of sound*, where *accentual* means that sound is perceptually salient in some way (melody, harmony, intensity accent ...).

But rhythm is also the highest level in the architectural structure (this is why it is not so easy to define rhythm: it can be seen from many different points of view). Then, rhythm can be defined by grouping different bars or by any other structure defined in any lower level.

As in the previous levels, rhythm can also be divided into different sub-layers. At first glad, groups of bars can define rhythm. But bars can be grouped into phrases and this phrase grouping can define rhythm, phrases can be grouped defining a *Chorus*, and so on. Where is the limit? The limit is imposed by the human memory, the short-term memory and the long-term memory: An opera recitative can be specially boring if the long-term memory is unable to percieve the whole concept of the composition. The same for some Jazz solos. See [63] and [65] for details about the memory behavior.

But as shown in previous sections, accents that define the meter (and the rhythm), are not only loudness accents. Then, why not to define different kinds of rhythm?

Loudness Rhythm: This is the most known type of rhythm. Basically, strong beats (sometimes, the long ones) have to be placed in strong parts of the bar while weak beats (usually the short ones) have to be placed in the weak parts of the bar.

Harmonic Rhythm: The harmonic rhythm is defined by the organization of chords in one or two bars. Each chord (and its tonality function) in a weak part of a bar must resolve to an equal or more important tonality function in the stronger part of the bar. This kind of rhythm is really important for composers due to the importance of tonality in most of the popular, classical and jazz music (See [55] and [61] for details).

Melodic Rhythm: Roughly speaking, melodic rhythm is defined by the presence of non-diatonic notes in a bar (apoggiatures, tensions...). Diatonic notes have to be placed in the strong beats in a bar while the non-diatonic ones have to be placed in the weak beats.

Of course, it is not an strict receipt. In fact, rhythm in music is just a combination of all the imaginable exceptions of these rules.

2.4 Managing rhythm

It is clear that the basic unit at the lowest architecture level is the beat, and the basic unit for meter is the bar. But, which one is the basic unit for rhythm? It is not so easy to define.

Most of the studies about rhythm deals with tempo, ticks, and other rigid features of rhythm. Many different algorithms try to find, by applying different techniques, some of these descriptors. But from a musical point of view, all these descriptors are not so important and at least they are not unique. Items

	Musicians View	Both	Technicians View
Rhythm level	allegro walking ...	chorus movement ...	rhythmicity periodogram ...
Meter level		$\frac{3}{4}$, $\frac{6}{8}$ ...	
Beat Level	foot tapping ...	metronome beat ...	tic onset ...

Table 2.3: Classification of some descriptors of rhythm according to different points of view

like *Allegro*, *slow* are most commonly used by musicians for describing rhythm. Furthermore, there are other phrasal-words for describing rhythm: "This solo doesn't walk" or "You have no swing".

All these items are good for describing rhythm at any of its architectural level. However, these descriptors only represents one specific part of rhythm which is fitted into one specific level of the whole architecture. On the other hand, these descriptors can be more technical (onsets...) or more *musical* (andante...). All of them are complementary. In Tab. 2.3 some examples of classification of rhythm-related words are shown.

Finally, doesn't the *Tempo* take part into these definitions? Not really: Tempo does not affect to the structure of rhythm, only to the speed that this structure must be performed.

2.5 Definitions

A lot of different concepts have been used in previous sections, and more concepts will be introduced during the next chapters. A review about the rhythm-related terminology and some definitions[1] can be useful for the reader. Here are the most important [30]:

Rhythm: Many things about rhythm have been discussed Sec. 2.3. Here, five basic grouping structures are enumerated [14]:

iamb: $\cup -$

anapest: $\cup \cup -$

trochee: $- \cup$

dactyl: $- \cup \cup$

amphibrach: $\cup - \cup$

For this definition of rhythm, the beat concept is used in a broad meaning. Here, Rhythm is independent of meter for two reasons: it can exist without

[1] Thanks to Fabien Gouyon for this clever classification

a specific meter (i.e. Gregorian Chant) and the anapest rhythm can exists in a simple duple meter.

Accent: The accent means differentiating events and thus giving a sense of shape or organization [58]. As discussed in Sec. 2.3.2, different kinds of accents can be defined. The perceptual accent is a combination of them.

Pulse - Beat: Used indistinctly, it is the periodic perceived stimuli. In western music notation, it is usually defined by the time signature. See Sec. 2.3.1 for details.

Tempo - Tactus: Tempo can refer to a musical concept, like the number of beats per minute, as well as a psicological concept, as the perceived rate of events. BPM (beats per minute) is the unit for tempo when it refers to the musical concept. Otherwise, tempo can be confused with *tactus*, and the used units are expressions like "fast", "slow" and so on.

Beat "phase": Beats are characterized by both period and phase. Period is the time distance between peaks (inversely proportional to the tempo) and phase is the temporal location of one beat.

Metric structure - Meter: Meter can be interpreted as an abstraction of the regularity of the different musical accents in one period[13]. Different accents can be grouped into a higher abstraction level and the Metric Structure tries to organize it. See Sec. 2.3.2 for details.

Time signature: It is the usual way to show the meter in western music. It is restricted to only two levels: the lower level defines the elements for beats and the upper level defines how to group them[76]. The bar lines defines the period where the same structure will be repeated. See Sec. 2.3.2 for details.

Downbeat: The "downbeat" corresponds to the first beat in a measure. Any other beat in a measure is called "offbeat". The beat that immediately precedes the downbeat is the "upbeat"[58].

Tick - Tatum: The term *tick* is proposed by Gouyon in [30] and it can be defined as "the regular time division that most highly coincides with all note onsets" [9]. Some other terms can be assigned to this concept: Bilmes uses the term "tatum" in [9], Schloss uses the term "attack-point" in [3] and Hoffman-Engl refers to the "cronota" for such a similar concept in [34]

Quantized duration - Metrical point: The GTTM (Generative Theory of Tonal Music) [46] propose a good structure for metrics in Western music. Pulses of a metrical level must be equally spaced, and there must be a pulse of the metric structure for every note. Then, all notes in a musical excerpt will be moved and fitted into one of the pulses defined in the structure. This operation is widely known as the *quantization* process.

Swing: This term originates in jazz music. It is difficult to define swing due to the personal interpretation of the concept depending on the performer, the musical piece and so on. Friberg and Sundtröm define this term as

"consecutive notes that are performed as long-short patterns" in [43] and Laroche defines it as a "slight delay of the second and fourth quarter beats".

Groove: This is the most difficult term to define. It can be interpreted as a "How to play", related to the "feeling" of the musical piece. It is defined basically by rhythm modifications of the rigid structure proposed in GTTM, but also melody, harmony and other aspects of music affect to the groove. Swing is just a particular case of groove.

IOI (Inter-onset interval): It is defined as the time difference between two successive onsets[6], but it can also be defined as the difference between any two onsets [17].

Chapter 3

Genre Description

3.1 Introduction

Genre classification is one of the basic tasks when managing with recorded music. Genre is the most important parameter and widely used by both the music industry and the consumers. It is crucial when searching a specific CD in a store or in internet. Furthermore, radio-stations and musical TV channels usually focus its playlist on some specific genres. Then, Automatic Genre Classification Systems could help the user in this amazing and difficult task.

Nowadays, with the increasing number of Web Sites for selling and sharing music, there is a huge amount of musical data on the net. Traditional techniques, like the typical "Ask Google", are becoming obsolete because all this data is not perfectly (manually) labeled. New musical browsing systems are needed and, for usability issues, Automatic Genre Classification Systems must be included.

In this chapter, a brief explanation about genre is given, discussions about different taxonomies are discussed, some used classifications are compared and, finally, automatic genre classification systems are also discussed.

3.2 Definition

This section is supposed to start with a precise and exhaustive definition of the term "Genre". Unfortunately, such a definition does not exist yet, and different authors differ in the attempts of defining this term. The basic reason could be found in the emotional, personal, cultural and social aspects of music.

The term *genre* comes from the Latin word *genus*, which means *kind* or *class*. Then, genres should be described as a musical category defined by some specific criteria. Due to the inherent personal comprehension of music, this criteria can not be universally established. Then, genres will be different for different people, groups, countries, and so on.

Genres are supposed to be characterized by the instrumentation and rhythmic structure, but there are many other factors that influence the classification. The major challenge in Automatic Genre Classification System is to define and fix as more factors as possible.

On the other hand, it is obvious that this task is not easy for humans either. Many taxonomies from different known libraries or web sites can differ a lot.

All these taxonomies are hand-made by musicologists and expert musicians from all different genres in music. Does it means that all of them are in a mistake? Of course, not. The only problem is that different points of view of music are applied in all those classifications. This aspect will be discussed later in Sec.3.4.

For simplicity, we will define the term *Genre* as *that kind of music that has similar properties, in those aspects of music that differ from the others.* What does it means? Some music can be clearly identified by the instruments used, i.e. the *Sardana* (in which the *Tenora* is a fundamental instrument of the *Cobla*), while other genres can be identified by the rhythm, i.e. *tecno*. Of course, both examples can be discussed because the instrument and the rhythm are not the only factors that define these genres respectively. In the proposed definition, the *"those aspects of music"* words refer to the possible combinations of them.

3.3 Evolution

Is not the purpose of this section to give a complete explanation from the evolution of music from all over the centuries. But one can realize that in the last twenty years, due to the technology improvements and the increasing computational power of computers, much more music is being produced. Of course, there is not any kind of problem on that fact, but the classification issues for all this produced music is becoming unapproachable.

But what is really important is that most of the computer music produced differs a lot from the traditional concepts of music. Since the Gregorian Chant up to the seventies, music can be characterized by its rhythm, harmony and melody. But nowadays, a new concept have been introduced: textures[75]. There exist several so called *chill-out* or *relaxation* CDs which are based on textures of soft sounds, and the textures change producing the musical sensations. There is neither melody nor harmony. How do we classify this kind of music? All Automatic Genre Classification Systems need to take all these aspects into account too.

3.4 Taxonomies

Depending on the application, typical taxonomies can be divided into two different groups[26]:

- Taxonomies of the music industry
- Internet Taxonomies
- Specific Taxonomies

Taxonomies of the music industry: These taxonomies are made by importan CD stores (i.e. Fnac, Virgin...). The goal of these taxonomies is to guide the consumer to a specific CD or track in the shop. They are usually built with four different levels:

1. Global music categories
2. Sub-categories
3. Artists (usually in alphabetical order)

4. Album (if available)

In Tab. 3.1 two examples of albums by using this taxonomy are shown (we will not discuss whether they are right or not). Although this taxonomy has shown its usability in exceed, some inconscistences can be found:

- Most of the stores have other *sections* with promotions, collections...
- Some authors have different recordings which should be classified in another Global Category.
- Some companies manage the labels according to the copyright management.

even so, it is a good taxonomy for music retailers.

Internet Taxonomies: The main benefit of the internet databases is that multiple relationships between authors, albums... can be established. They are not in a specific physical place. Then, with these multiple relationships, the consumer can browse according to his more personal point of view about genres. In Tab.3.2, two different paths for a search of the same album are shown (www.amazon.com). Some inconsistencies are also found here, specially from the semantic point of view:

- Hierarchical links are usually genealogical. But sometimes, more than one father is necessary. i.e. both *Pop* and *Soul* are the "fathers" of *Disco*
- In most of the taxonomies, geographical inclusions can be found. It is really debatable if this kind of classification is correct. Some sites propose the genre *World Music* (www.ebay.com) in which, if one is strict, one should be able to find some *Folk* music from China, *Pop* music of Youssou N'Dour and *Rock* music of Bruce Springsteen.
- Aggregation is commonly used to join different styles: *Reggae-Ska* → *Reggae* and *Reggae-Ska* → *Ska* (www.ebay.com).
- Repetitions can also be found: *Dance* → *Dance* (AllMusicGuide).
- Historical Period labels may overlap, specially in classical music: *Baroque* or *Classical* and *French Impressionist* may overlap.
- Specific random-like dimensions of the sub-genre can create confusion.

Specific Taxonomies: Sometimes, some quite specific taxonomies are needed, even if they are not really exhaustive or semantically correct. A good example can be found in Ballroom taxonomies in which *Tango* category can include classical titles from "Piazzolla" as well as electronic titles from "Gotan Project".

At this point, one could wonder which the right taxonomy is. In fact, including my personal experience as a member of the Library Committee at the ESMUC[1], one can conclude that the *perfect* taxonomy does not exist. The perfect taxonomy is the one that best helps us in our work.

[1]ESMUC: Escola Superior de Música de Catalunya; www.esmuc.net

Levels	Example 1	Example 2
Global category	Pop	Jazz
Sub-category	General	Live Albums
Artist	Avril Lavigne	Keith Jarrett
Album	Under my skin	Koln Concert

Table 3.1: Two examples of the industrial taxonomy

#	Path
1	Styles → International → Caribbean&Cuba → Cuba → Buena Vista Social Club
2	Styles → Jazz → Latin Jazz → Buena Vista Social Club
3	Music for Travelers → Latin Music → Latin Jazz → Buena Vista Social Club

Table 3.2: Two different virtual paths for the same album, for the Internet taxonomy

3.5 Genres and Music Information Retrieval

Over the last years, many systems for Automatic Genre Classification have been implemented. All these systems take different techniques from other fields like speech, statistics and musical analysis: extraction of timbre features, onset detection and beat tracking or Hidden Markov Models are some of them. The structure of these systems can be divided in three different parts: feature extraction, pattern recognition and post-processing[49]

Feature extraction: This is the most important part of the process. Depending on the extracted features, the system will classify genres according to this description (See Sec. 3.3). If no rhythmical features are extracted, the system will not classify Genres according to a rhythmic criteria. In this case, a Ballroom music database could not be classified. In a general way, timbre related descriptors (MFCC and derivates) and rhythm related descriptors (onset detection, automatic time-signature detection and beat tracking) are used. Some melodic and harmonic features should also be included.

Pattern Recognition: There are several general-purposed machine-learning and heuristic-based techniques that can be adapted to this task. Hidden Markov Models or Neural Networks are usually used. Here, performing a supervised training or an unsupervised training can be chosen. With supervised training, manually labeled data is needed and it must be consistent with the chosen taxonomy. The most precise data is chosen, the better training will be performed. On the other hand, unsupervised techniques allow more (unlabeled) general training data, but the output will not be consistent with any taxonomy. Results are made only under a mathematical approaches and they are not supposed to have any musical meaning. Nevertheless, they can be useful for specific applications[73].

Post Processing: Once the system is trained, different data can be extracted from the system. Genre labels can be frame-to-frame or unique for the

Figure 3.1: An usable visualization for clustered concepts (www.allmusic.com)

whole song; a black-or-white output or a probability density function.... It depends on the application. In Fig. 3.1 an example of an usable visualization for clustered concepts is shown.

3.6 Previous work

Some successful works have been made in Automatic Genre Classification Systems. Most of them have been designed for genre classification with very strong boundaries, i.e. the number of available genres, or the audio database for testing.

According to [57], humans can predict musical genres on 250[ms] of audio. This process is a real-time process, then, the conclusion is that no other higher layers are needed for genre classification: this task could be easier than might be thought.

From now on, one of the most important works in this field is made by Tzanetakis in [69]. His work is divided in two parts. The first is called GenreGram and it is developed for real time radio broadcasts and displays cylinders bouncing up and down that represent each genre. The second work is called GenreSpace and it is a 3-D representation of genre space. It is used for representing huge databases. These ideas are finally presented with full implementation in [68]. In this system, timbre and rhythmical features are combined with pitch information for classification. This system is able to distinguish between 10 different genres with a 61% of successful classifications. This results are quite comparable with human behavior.

Another successful approach is presented by Kosina in [41], with the inclusion of the MUGRAT system as a part of her thesis.This work provides an excellent overview about genre recognition and is able to distinguish between *metal*, *dance* and *classical* music with a success rate up to 82%. The WEKA software framework is used for this purpose [74].

A most recent work is presented by Grimaldi in [31]. In it, the system uses the discrete wavelet transform (DWT) and extracts many features in the (1) time-domain and (2) scale-domain. In this work, the classification system is more complicated than the Tzanetakis' work, but results are also quite encouraging. Lambrou has also a classification system in [42].

Finally, some good reviews of the state of the art in genre classification can be found in [49] and [7]. Nevertheless, we are a little bit far for a full featured automatic Genre classification system and, as Tzanetakis cite in [51]:

"An important contribution of this work is the comparison of the automatic results with human genre classifications on the same dataset. The results show that, although there is room for improvement, genre classification is inherently subjective and therefore perfect results can not be expected neither from automatic nor human classification."

perhaps this effort has been done in vain.

Chapter 4

Technical Review

4.1 General Descriptors

In this section, a short overview of the most used general-purpose descriptors is sown. This list is not exhaustive, and we suggest further readings for details.

4.1.1 Energy

The Energy is not a representative descriptor at all. The energy of the input audio depends on many not fixed parameters of the experiment, as the mic/line-in amplifier level while recording, or the used codification.

From a mathematical point of view, the time-domain energy of the input signal can be defined as:

$$E = \sum_{n=0}^{N} x[n]^2 \tag{4.1}$$

where $x[n]$ is the input time-domain data and N is the length of $x[n]$ (in samples).

4.1.2 Zero Crossing Rate

As defined in [40][59], the Zero Crossing Rate (ZCR) of the time domain waveform provides a measure of the weighted average of the spectral energy distribution. This measure is similar to the spectral center of mass or Spectral Centroid of the input signal (see Sec.4.1.3). It also can be interpreted as the "noisiness" of the input signal.

From a mathematical point of view, it can be calculated as:

$$ZCR = \frac{1}{2} \sum_{n=1}^{N} |sign(x[n]) - sign(x[n-1])| \tag{4.2}$$

where *sign* function is 1 for positive arguments and 0 for negative arguments, $x[n]$ is the input time-domain data and N is the length of $x[n]$ (in samples) [68].

4.1.3 Spectral Centroid

As defined in [60], the Spectral Centroid is the *balancing point* of the spectral power distribution. The Spectral Centroid value rises up, specially for percussive sounds, due to the high density of harmonics in the upper bands of the spectrum.

This concept has been introduced by psychoacoustic and music cognition fields. It can be interpreted as a measure of the average frequency, weighted by amplitude, of a spectrum, that is, a measure related with the brightness of the signal.

Be careful by confusing the Spectral Centroid and the Fundamental Frequency: while the Spectral Centroid can be higher for a trumpet sound than for a flute sound, both instruments can play exactly the same note.

From a mathematical point of view, the Spectral Centroid can be calculated as:

$$SC = \frac{\sum f_i a_i}{\sum a_i} \tag{4.3}$$

where f_i is the frequency value of each bin of the FFT and a_i is its amplitude.

In many applications, it is averaged over time. This $\bar{SC}$ value can be averaged into different time-domain frames as shown in next equation:

$$\bar{SC} = \frac{1}{N} \sum SC_i \tag{4.4}$$

where N is the number of frames and SC_i is the Spectral Centroid value for each frame.

Finally, the spectral centroid is sometimes normalized with the fundamental frequency, making this value adimensional:

$$SC = \frac{\sum f_i a_i}{f_1 \sum a_i} \tag{4.5}$$

4.1.4 Spectral Flatness

The Spectral Flatness can be defined, according to Ozgur Izmirli in [52], as the ratio of the geometric mean to the arithmetic mean of the power spectral density components in each critical band for the input signal.

The process is reccomended to follow next steps: the signal should be sampled at $f_s = 22050[Hz]$ and the 2048-points FFT should be performed after the Hanning windowing. The windows should be 30% overlapped. A Pre-emphasis filter should be applied in order to compensate the behavior of the human ear. The bark-band filter output should be calculated from the FFT and the power spectral density should be computed for each critical bands. All these values are used to compute the arithmetical and geometrical means.

$$SFM = \frac{G_m}{A_m} \tag{4.6}$$

where G_m and A_m are the arithmetical and geometrical means of the spectral power density function respectively.

Sometimes, the Spectral Flatness Measure is converted to decibels as follows [39]:

$$SFM_{dB} = 10 log_{10} \frac{G_m}{A_m} \tag{4.7}$$

and, furthermore, it can be used to generate a coefficient of tonality α as follows:

$$\alpha = min\left(\frac{SFM_{dB}}{SFM_{dB_{max}}}, 1\right) \tag{4.8}$$

i.e.., an SFM of $SFM_{dB_{max}} = -60dB$ is used to estimate that the signal is entirely tonelike, and an SFM of $0dB$ to indicate a signal that is completely noiselike.

4.1.5 4Hz Modulation

The 4Hz Modulation Energy Peak is a characteristic feature of speech signals due to a near 4Hz syllabic rate. It is calculated by decomposing the original waveform into 20 [64] or 40 [60] (depends on accuracy) mel-frequency bands. The energy of each band is extracted and a second band pass filter centered at 4 Hz is applied to each one of the bands.

Of course, this 4Hz value depends on the language. It is well known that in Catalan or Spanish languages, this value is near the $6[Hz]$ instead of the $4[Hz]$ value for English.

4.1.6 Mel-Cepstrum

The Cepstrum of an input signal is defined as the Inverse Fourier Transform of the logarithm of the spectrum of the signal [54]:

$$c[n] = \frac{1}{N}\sum_{k=0}^{N-1} \log_{10} |X[k]|^{j\frac{2\pi}{N}kn}, \quad 0 < n < N-1 \tag{4.9}$$

where $X[k]$ is the spectrum of the input signal $x[n]$ and N is the length of $x[n]$ (in samples).

The process is an Homomorphic Deconvolution because it is able to separate the excitation part of the input signal for further manipulations.

But for the Mel-Cepstrum calculations, some little modifications have to be done. The *mel* scale tries to map the perceived frequency of a tone onto a linear scale, as shown in Eq. 4.10:

$$mel\ frequency = 2595 \cdot \log_1 0\left[1 + \frac{f}{700}\right] \tag{4.10}$$

The *mel* scale can be interpreted as an approximation of the perceptual frequency scale (referred as *critical bands* or *barks*)[77]:

$$bark = 13 \cdot \arctan\left(\frac{0.76 \cdot f}{1000}\right) + 3.5 \cdot \arctan\left(\frac{f^2}{7500^2}\right) \tag{4.11}$$

4.1.7 Delta descriptors

For some specific cases, the descriptor we use has not the whole information we need. This case is specially critical when this data is supposed to be the input of a Hidden Markov Model system. Then, the first-order or the second-order differentiation of the original parameter is used. The first order differentiation

can be calculated in many different ways, but we use the Causal FIR filter implementation, as shown in Eq. 4.12 [5]:

$$\dot{p}[n] = \frac{\partial p[n]}{\partial t} = \sum_{m=n}^{m=N} p[n-m] \tag{4.12}$$

where N is the depth of the differentiation. The descriptor we get by applying differentiation is denoted as *delta* descriptor (∇). Finally, if we have to apply the second-order differentiation, we will compute the Eq. 4.12 recursively. Then, we have the *delta-delta* descriptor (∇^2).

4.2 Rhythm-related Descriptors

This section will show a short review of the most used descriptors and techniques for managing with rhythm.

4.2.1 Inter Onset Interval

As defined in 2.5, the Inter Onset Interval (IOI) is the time difference between two successive onsets[6], but it can also be defined as the difference between any two onsets [18]. Many algorithms can be found for IOI computations, but only one of them will be explained here. According to Gouyon in [30], the IOI histogram can be computed as:

1. Onset detection: First of all, the energy of each non-overlapping frames is calculated. The onset will be detected when the energy of the current frame is superior to a specific percentage (i.e. 200%) of a fixed number (i.e. 8) of the previous frame energy average. It is assumed that there is a gap of 60$[ms]$ between onsets, and a weighting factor is applied to each onset according to the number of consecutive onsets whose energy satisfies the threshold condition mentioned above.

2. IOI computations: In this algorithm proposed by Gouyon, the time differences between any two onsets is taken. Each IOI has an associated weight according to the smallest weight among the two onsets used for this IOI computation

3. IOI histogram computation: With all these computed IOI, a histogram is created. This histogram is smoothed by the convolution of a Gaussian function. The parameters of this Gaussian function are fixed empirically.

At this point, the histogram of IOI is available. This data can be used for tick induction computations, rhythm classification, automatic BPM detection, and so on.

4.2.2 Beat Tracking

The main goal for Beat Tracking Systems (BTS) is to construct an algorithm for "beat" or "pulse" symbolic representation of rhythm. But some difficulties can be found in this process. They can be resumed as[48]:

- Commercial music has usually many different instruments. Then, the onsets become difficult to detect.

- The classical technique of peak-finding is not useful due to the high content of peaks in the audio signal that are not directly related with rhythm.
- As mentioned in Sec. 2.2, beats, pulses and accents are not directly related with a change in the audio signal due all of them are perceptual concepts.
- In MIDI data, it is difficult to determine which note-value a beat corresponds to, and whether a beat is a strong beat or a weak beat.

Taking into account all these challenges, some important works have been done:

Beat Tracking System proposed by Scheirer

According to Scheirer in [22], pulses can be described by its frequency and phase components. As mentioned in Sec. 2.5, the frequency determines the BPM of the rhythm while the phase determines where the *downbeat* is located. The process for Beat Tracking can be divided in different steps:

1. The input signal is decomposed into six bands.
2. The envelope and the derivative of the envelope is calculated for each one of these six bands.
3. Each of the envelope derivatives is filtered with a set of tuned resonators.
4. One of the resonators is selected to be the reference for phase computations.
5. The output of each resonator is analyzed in order to fix periodicities (a phase-locked behavior). This information is saved for each of the subbands
6. All this data is mixed for the tempo estimation (BPM), and phase-locked resonators determines the phase of the rhythm (structure).

In Fig. 4.1 a screenshot of the block diagram for this beat tracking system is shown.

Beat Tracking System proposed by Goto

Goto proposes a BTS that examines multiple possibilities of positions of beats in parallel[48]. Linear prediction techniques are also used for to predict when the new onset should be located. Fig. 4.2 shows a screenshot of the block diagram for this beat tracking system. It can be resumed as:

1. Frequency Analysis: This block searches the notes' onsets from the A/D converted signal. It also detects the Bass drums (BD) and the Snare drums (SD) onsets, related with strong and weak beats respectively.
2. Beat Prediction: Multiple agents interpret the onsets times previously found and construct parallel hypotheses: each agent calculates the IBI(inter-beat-interval), predicts the next beat time, proposes which kind of beat it should be, and evaluates its reliability.

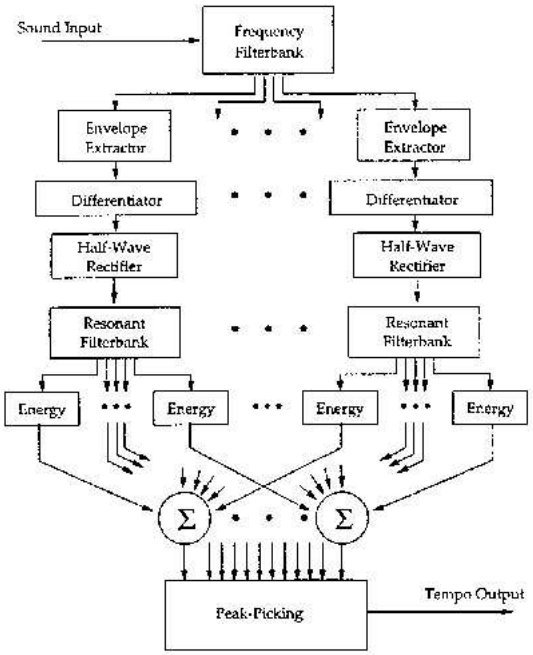


Figure 4.1: Screenshot of the block diagram for Beat Tracking calculations proposed by Eric Scheirer

3. BI Generation: This part manage with all the hypotheses and selects the most reliable one.

4. BI Transmission: Transmits the beat-information (BI) to other applications

Just a comment about this system: Due to the predictions that have to be done for successful results when real input analog data is entered to the system (the system is basically tested with popular music), it does not work on real time.

4.2.3 Beat Histogram

The *Beat Histogram* concept was proposed by Tzanetakis in [69]. It is a part of his Automatic Genre Classification system. Some techniques like the Wavelet Transform are used[71, 16]. After applying toe Multi-resolution Analysis techniques, the Wavelet decomposition of a signal can be interpreted as a successive high-pass and low-pass filtering of the time domain signal. This decomposition is defined by:

$$y_{high}[k] = \sum_n x[n]g[2k-n] \tag{4.13}$$

$$y_{low}[k] = \sum_n x[n]h[2k-n]$$

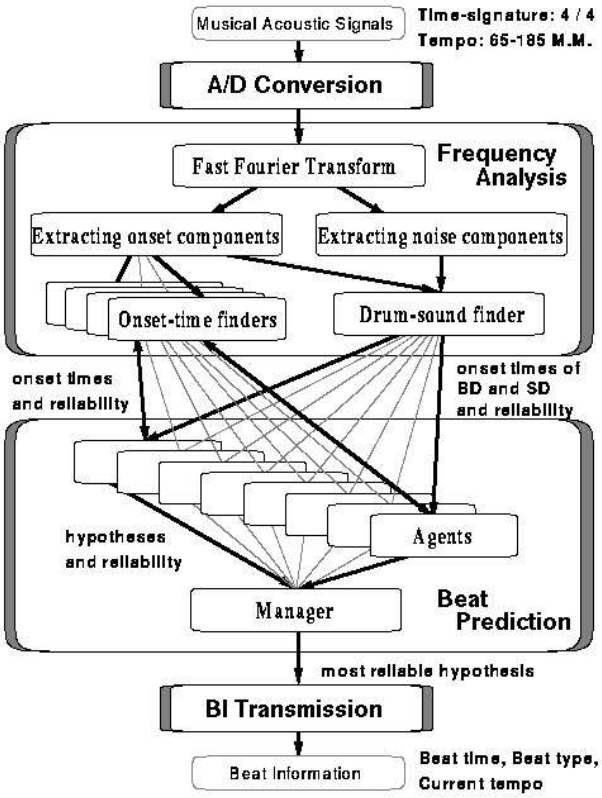


Figure 4.2: Screenshot of the block diagram for Beat Tracking calculations proposed by Masataka Goto.

where $y_{high}[k]$ and $y_{low}[k]$ are the output of high-pass and low-pass filters respectively, and $g[n]$ and $h[n]$ are the filter coefficients for the high-pass and low-pass filters associated to the scalar and wavelet functions for 4th. order Daubechies Wavelets. The main advantage by using the Wavelet Transform deals with the similarity of the decomposed signal to a 1/1 octave filter bank in a similar way than the human ear does.

When the signal is decomposed, some additional signal processing (in parallel for each band) is needed:

1. Full Wave Rectification (FRW):

$$z[n] = abs(y[n]) \tag{4.14}$$

   where $y[n]$ is the output of the Wavelet decomposition at that specific scale (or octave)

2. Low-pass filtering (LPF): One pole filter with $\alpha = 0.99$:

$$a[n] = (1 - \alpha)z[n] - \alpha \cdot a[n] \tag{4.15}$$

3. Downsampling($\downarrow$) by k=16:

$$b[n] = a[kn] \tag{4.16}$$

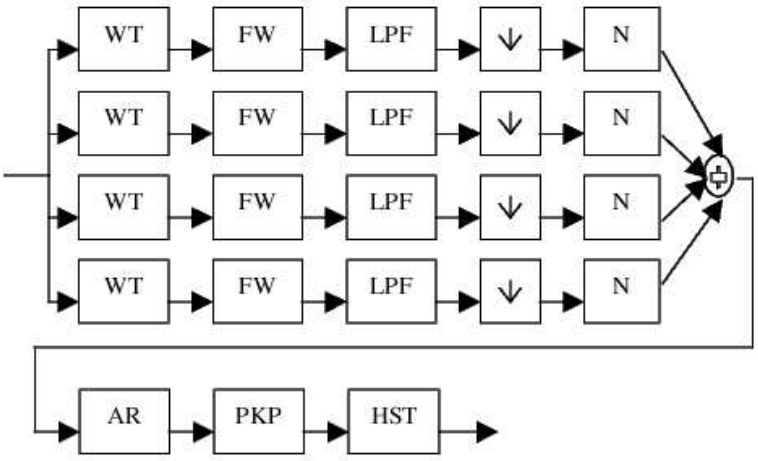


Figure 4.3: Screenshot of the block diagram for Beat Histogram calculations proposed by Tzanetakis

4. Normalization (Noise removal NR):

$$c[n] = b[n] - E\left[b[n]\right] \tag{4.17}$$

5. Autocorrelation (AR):

$$d[n] = \frac{1}{N} \sum_{n} c[n]c[n+k] \tag{4.18}$$

This autocorrelation is computed by using the FFT for efficiency.

In Fig. 4.3 a screenshot of the block diagram for beat histogram calculations proposed by Tzanetakis is shown.

At this point, the first five peaks of the autocorrelation function are detected and their corresponding periodicities in beats per minute(BPM) are calculated and added to the beat histogram.

Finally, when the beat histogram is computed, some features can be used:

1. Period0: Periodicity in BPM of the first peak
2. Amplitude0: Relative amplitude of the first peak
3. Ratio Period1: Ratio of the periodicity of the second peak to the first one
4. Amplitude1: Relative amplitude of the second peak
5. Ratio Period2, Amplitude2...

Other authors use the number of peaks, their distribution, max and min operations over the peaks... of the beat histogram as input features to their classification system[31].

4.2.4 Beat Spectrum

The concept of *Beat Spectrum* was introduced by Foote et. al. in [23]. It is a measure if the acoustic self-similarity as a function of time lag. The goal of this method is that it not depends on fixed thresholds. Hence, it can be applied to any kind of music and Genre, and furthermore, it can distinguish between different rhythms at the same tempo. The *Beat Spectrogram* concept is also introduced in this work as the time evolution of the rhythm representation. It can be computed by the following steps:

1. Audio parameterization: The FFT of the windowed input data is computed. Then, by using any known filtering technique (i.e MFCC), the vector of the log energy for each band is obtained.

2. Calculating frame similarity: Data derived from previous parameterization is embedded in a 2D representation. A dis-similarity measure between two vectors i and j is computed as:

$$D_C(i,j) = \frac{v_i \cdot v_j}{\|v_i\|\|v_j\|} \tag{4.19}$$

3. Distance Matrix Embedding: The similarity matrix S contains all the measures for all the i and j as shown in Eq. 4.19. In this matrix, audio similarities can easily be observed.

4. The Beat spectrum: Periodicities and rhythmic structure can be derived from this similarity matrix. An estimation of the Beat Spectrum can be found by summing S along the diagonal as follows:

$$B(l) = \sum_{k \subset R} S(k, k+l) \tag{4.20}$$

   where $B(0)$ is the sum for all the elements of the main diagonal over some continuous range R, $B(1)$ is the sum of all the elements along the first super-diagonal, and so on. A more robust estimation of the Beat Spectrum can be computed as:

$$B(k,l) = \sum_{i,j} S(i+k, j+l) \tag{4.21}$$

   where the autocorrelation of S is computed. Some applications like onset detections can be computed by using this Beat Histogram.

4.2.5 Swing Ratio

As mentioned in Sec. 2.5, Friberg and Sundtröm define swing as "consecutive notes that are performed as long-short patterns" in [27], and Laroche defines it as a "slight delay of the second and fourth quarter beats" in [43].

The swing ratio can be computed, from a mathematical point of view, as the relationship between the duration of the first eighth-note and the second eighth-note:

$$Swing\ Ratio = \frac{t_{1rst.\ eighth\ note}}{t_{2nd.\ eight\ note}} \tag{4.22}$$

The Swing Ratio coefficient is adimensional.

4.3 Statistics

4.3.1 Mean

The *mean* or *expected value* of a discrete random variable X, can be computed as [50]:

$$\mu = E(X) = \sum_x xf(f) \tag{4.23}$$

where x are the obtained values of the experiment and $f(x)$ is the weight of each for these values. In a typical coin experiment, $f(x) = 0.5$ for all of the cases.

4.3.2 Variance

The *variance* of X is a measure of the dispersion of the samples around the mean value, and it can be computed as [50]:

$$\sigma^2 = V(X) = E(X-\mu)^2 = \sum_x (x-\mu)^2 f(x) = \sum_x x^2 f(x) - \mu^2 \tag{4.24}$$

or, by using in MATLAB nomenclature [1] [67]:

$$\sigma^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 \tag{4.25}$$

Finally, the *Standard Deviation* of X can be computed as:

$$\sigma = [V(X)]^{\frac{1}{2}} = \big(\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2\big)^{\frac{1}{2}} \tag{4.26}$$

4.3.3 Skewness

The skewness of a distribution is defined as [67]:

$$y = \frac{E(x-\mu)^3}{\sigma^3} \tag{4.27}$$

where μ is the mean of x, σ is the standard deviation of x and $E(t)$ is the expected value of t.

Skewness is a measure of the asymmetry of the data around the sample mean. If skewness is negative, the data are spread out more to the left of the mean than to the right. If skewness is positive, the data are spread out more to the right. The skewness of the normal distribution (or any perfectly symmetric distribution) is zero.

[1] MATLAB calculates the variance with $\frac{1}{n-1}$ instead of $\frac{1}{n}$

4.3.4 Kurtosis

The kurtosis of a distribution is defined as [67]:

$$k = \frac{E(x-\mu)^4}{\sigma^4} \tag{4.28}$$

where μ is the mean of x, σ is the standard deviation of x and $E(t)$ is the expected value of t.

Kurtosis is a measure of how outlier-prone a distribution is. The kurtosis of the normal distribution is 3. Distributions that are more outlier-prone than the normal distribution have kurtosis greater than 3; distributions that are less outlier-prone have kurtosis less than 3.

4.4 Periodogram

In this section, we will present an overview of the Periodogram concept and computations which will be used in next chapters. The Periodogran was introduced by Schuster in 1898 to study periodicity of sunspots. As shown in Sec. 4.3, the sample mean and the sample variance are unbiased and asymptotically unbiased estimators respectively. Furthermore, they are both consistent estimators[5].

Sometimes, in digital signal processing field, the estimation of the power density spectrum $P_{ss}(\Omega)$ of a continuous stationary random signal $s_c(t)$ is needed. After the anti-aliasing filtering, another discrete-time stationary random signal $x[n]$ will be created, and its power density spectrum $P_{xx}(\omega)$ will be proportional to $P_{ss}(\Omega)$ over the whole new bandwidth of $x[n]$:

$$P_{xx}(\omega) = \frac{1}{T} P_{ss}\left(\frac{\Omega}{T}\right) \qquad |\omega| < \pi \tag{4.29}$$

where T is the sampling period. Then, a good estimation of $P_{xx}(\omega)$ will provide a reasonable estimation of $P_{ss}(\Omega)$.

Let $v[n]$ be the windowed input signal:

$$v[n] = x[n] \cdot w[n] \tag{4.30}$$

where $w[n]$ is the windowing function. Then, the Fourier Transform of $v[n]$ can be computed as:

$$V(e^{j\omega}) = \sum_{n=0}^{L-1} w[n]x[n]e^{-j\omega n} \tag{4.31}$$

where L is the length (in samples) of the windowing function.

Now, let $I(\omega)$ be the estimation of the power density spectrum:

$$I(\omega) = \frac{1}{LU}|V(e^{j\omega})|^2 \tag{4.32}$$

where U is the normalization factor for removing the bias in the spectral estimate. Depending on the windowing function, this estimator can be:

- If $w[n]$ is the rectangular window $\rightarrow$ $I(\omega)$ is the *periodogram*

- If $w[n]$ is NOT the rectangular window $\rightarrow$ $I(\omega)$ is the *modified periodogram*

Furthermore, note that the periodogram can also be computed as:

$$I(\omega) = \frac{1}{LU} \sum_{m=-(L-1)}^{L-1} c_{vv}[m] e^{-j\omega m} \tag{4.33}$$

where

$$c_{vv}[m] = \sum_{n=0}^{L-1} x[n]w[n]x[n+m]w[n+m] \tag{4.34}$$

As $c_v v[m]$ is the aperiodic correlation sequence for the finite-length sequence $v[n]$, the periodogram can be interpreted as the Fourier Transform of the aperiodic correlation of the windowed input data.

Finally, as we are in discrete domain, the periodogram can only be obtained at discrete frequencies. Then, discrete periodigram can be computed as:

$$I(\omega_k) = \frac{1}{LU} |V[k]|^2 \tag{4.35}$$

where $V[k]$ is the N-point DFT of $w[n]x[n]$.

4.5 Linear Discriminant Analysis

4.5.1 Introduction

One of the goals in Music Information Retrieval systems is data classification. This classification is usually made under some specific conditions according to the final purpose of our system. Content-based processing is needed for this task, and many different descriptors extracted from the original signal (audio, MIDI, XML files...) are also needed.

This is the context where the *LDA* (Linear Discriminant Analysis) becomes imprescindible to get the objectives. LDA is a set of statistical techniques used for analyze high dimensional sets of data. By using LDA, all the parameterized input data is divided into different sub-spaces. Then, the classifier (implemented in a HMM system, Neural Networks...) is not so complicated and, furthermore, results are improved.

Another classification technique commonly used is *PCA* (Principal Component Analysis). The main difference between PCA and LDA that one can found is on the previous assumptions about the distribution of the new spaces: PCA does not assume anything about these new spaces while LDA does. In Fig. 4.4 there are two sets of data that perfectly define two different classes: class 1 and class 2.

Each one of the ellipses represents a different class. If PCA analysis is applied in this specific situation, we get that the maximum variation of data is along the x axe. Otherwise, if LDA analysis is applied, we get that the best discrimination is made by the projection of data to the y axe.

Some important properties about LDA are:

- Capacity for dimensionality reduction

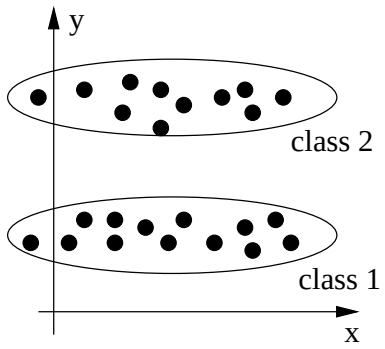


Figure 4.4: Graphical interpretation of LDA

- Decorrelation between new coefficients
- High discrimination power between classes

4.5.2 Different approaches in LDA

Data sets can be transformed and test vectors can be classified in the transformed space by two different approaches[28]:

Class-dependent transformation: This type of approach involves maximizing the ratio of between class variance to within class variance.

Class-independent transformation: This approach involves maximizing the ratio of overall variance to within class variance.

The first approach maximizes the ratio so that adequate class separability is obtained. The second approach uses only one criterion to transform the data sets and hence all data points irrespective for their class identity are transformed using this transform. That means that all classes are considered separate classes against the other ones.

4.5.3 Calculations

From a mathematical point of view, LDA is just a linear transform:

$$o = A'v \tag{4.36}$$

where v is the input vector of the parameterization block, A' is the transformation matrix (that's what exactly we are looking for) and o is the new set of parameters.

At this point, the study will be shown, for simplicity, for only two different classes. It will be easy to expand the results to a higher dimensional set of data.

Let the data be:

$$v_1 = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \\ \dots & \dots \\ a_{m1} & a_{m2} \end{bmatrix} \qquad v_2 = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \\ \dots & \dots \\ b_{m1} & b_{m2} \end{bmatrix} \tag{4.37}$$

where, in a_{ij}, i is the index of the parameter and j is the index of the measure.

First of all, it is necessary to define a kind of distance between classes. This distance can be calculated as the distance of the means of each class. The mean value for each class can be defined as:

$$\mu_i = \frac{1}{n_i} \sum_{v \in \nu_i} v \qquad i = 1, 2 \tag{4.38}$$

and the mean of each class after applying the transformation shown in Eq. 4.36 can be defined as:

$$\tilde{\mu_i} = \frac{1}{n_i} \sum_{o \in O_i} o = \frac{1}{n_i} \sum_{v \in \nu_i} A'v = A'\mu_i \tag{4.39}$$

The distance between the two classes, after the projection, is:

$$d = |\tilde{\mu_1} - \tilde{\mu_2}| = |A'(\mu_1 - \mu_2)| \tag{4.40}$$

Note that this distance can be as high as one wants by applying a scale factor to the matrix A'. The chosen scale factor is the within-class scatter:

$$S_0 = \sum_{o \in O_1} (o - \tilde{\mu_1}) (o - \tilde{\mu_1})' + \sum_{o \in O_2} (o - \tilde{\mu_2}) (o - \tilde{\mu_2})' \tag{4.41}$$

Then, the new defined distance is:

$$J(A) = \frac{|\tilde{\mu_1} - \tilde{\mu_2}|}{S_0} \tag{4.42}$$

The Fisher Linear Discriminant is defined as the linear projection $o = A'v'$ in such a way that the distance criterion $J(A)$ is maximized [56]. Then the Eq. 4.41 can be written as:

$$S_0 = A'S_vA \tag{4.43}$$

where S_v is the between-class scatter matrix, represented by S_W. For a general application with K different classes, S_W is defined as:

$$S_W = \sum_{k=1}^{K} S_k = \sum_{k=1}^{K} \sum_{v \in \nu_k} (v - \mu_k) (v - \mu_k)' \tag{4.44}$$

Furthermore, the concept of scatter matrix can be expanded to a general scatter matrix S_T defined as:

$$S_T = \sum_{k=1}^{K} \sum_{v \in \nu_k} (v - \mu)(v - \mu)' \tag{4.45}$$

where μ is the mean for all de data in the original space:

$$\mu = \frac{1}{\sum_{k=1}^{K} n_k} \sum_{k=1}^{K} \sum_{v \in \nu_k} v \tag{4.46}$$

The Eq. 4.45 can be recalculated:

$$\begin{aligned} S_T &= \sum_{k=1}^{K} \sum_{v \in \nu_k} (v - \mu)(v - \mu)' \\ &= \sum_{k=1}^{K} \sum_{v \in \nu_k} (v - \mu_k + \mu_k - \mu)(v - \mu_k + \mu_k - \mu)' \\ &= \sum_{k=1}^{K} \sum_{v \in \nu_k} (v - \mu_k)(v - \mu_k)' + \sum_{k=1}^{K} \sum_{v \in \nu_k} (\mu_k - \mu)(\mu_k - \mu)' \\ &= \sum_{k=1}^{K} \sum_{v \in \nu_k} (v - \mu_k)(v - \mu_k)' + \sum_{k=1}^{K} n_k (\mu_k - \mu)(\mu_k - \mu)' \\ &= S_W + S_B \end{aligned} \tag{4.47}$$

where S_B is the general between-classes scatter matrix:

$$S_B = \sum_{k=1}^{K} n_k (\mu_k - \mu)(\mu_k - \mu)' \tag{4.48}$$

At this point, the Eq. 4.42 can be recalculated as:

$$J(A) = \frac{|A' S_B A|}{|A' S_W A|} \tag{4.49}$$

and the A matrix that maximizes this expression is generated by the eigenvectors of:

$$S_W^{-1} S_B a_i = \lambda_i a_i \tag{4.50}$$

4.5.4 Dimensionality reduction

About the capacity for dimensionality reduction, in Fig. 4.5 the percentage of the global variance for all the data set is represented as a function of the number of the chosen coefficients. That means that, if the system has originally 60 coefficients, this number can be reduced to 50 with few looses in his classification powerful, about 2.5%. This allows to reduce the complexity and the number of computations to the next blocks of the system.

4.5.5 Conclusions

The LDA has been presented here as a classification technique, and some comparisons between LDA and PCA have been made. Throughout most of the explanation has been focused to a 2-class problem, generalization to a more dimensional space has been shown. Two different approaches to LDA (class-independent and class-dependent) have been shown too: If the LDA implementation is focused on a generalization problem, independent transformation is preferred. Otherwise, if the LDA implementation is focused on discrimination, dependent transformation will be used.

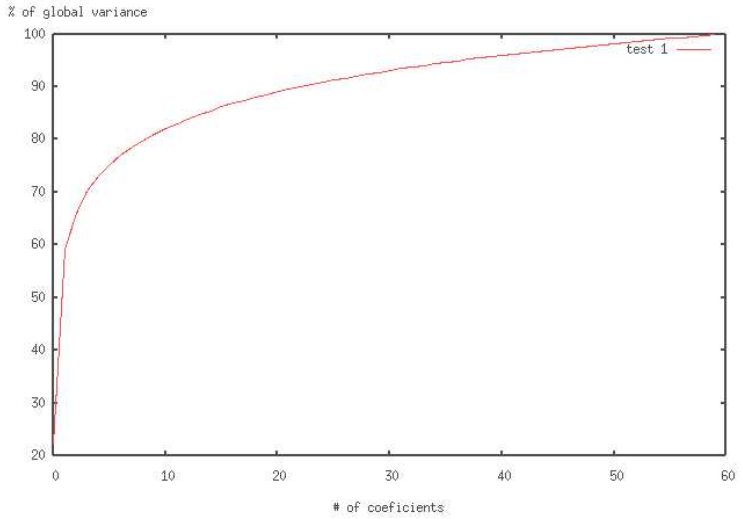


Figure 4.5: % of the global variance of the projected data as a function of the number of parameters

4.6 Hidden Markov Models

4.6.1 Introduction

Hidden Markov Models (HMM) have shown to be a powerful statistical tool in speech processing. The theoretical aspects about HMM are quite old but it is not until the sixties (when new parameter estimation techniques appear [8]) that they become really important in mathematical developments. The inclusion of HMM in speech is not until the seventies. Nowadays, HMM are included in many different research areas, and they have shown his robustness and elegance in exceed[45].

4.6.2 Main Idea

The idea of HMM is quite simple. Nature is full of processes that are still unknown for humans. The unique knowledge one has about these processes is via observation. Observation is the way for discover the real behavior of the system. At this point, one can choose between a deterministic or statistical description of the system. Deterministic descriptions are based on a perfectly knowledge of the behavior of the system, i.e. a sine generator. In the other hand, statistical descriptions are used when this behavior is not fully controlled and only statistical estimations can be done. This would be the case for Gaussian models, Markov Models and Hidden Markov Models.

Hidden Markov Markov models are an extension of the Markov Models. The Markov Models are useful when each observation corresponds to a physical event, but this case is too restrictive to many problems of interest. Hidden Markov Models are used when the observations are also probabilistic functions. Sometimes, Hidden Markov Models are referred as a double embedded stochastic process with an underlying stochastic process that is not observable (hidden) but

can only be observed through another set of stochastic processes that produce the sequence of observations[45].

4.6.3 Elements of a HMM

Each nature fenomena can be represented as a sequence of vectors or observations O, defined as[66]:

$$O = o_1, o_1 \dots o_T \tag{4.51}$$

where o_t is the vector observed at time t. Then, the phenomena recognition problem can be resumed, from a mathematical point of view, as:

$$argmax_i \left\{ P\left(w_i|O\right) \right\} \tag{4.52}$$

where w_i is the ith. model previously defined in our *vocabulary*. Now, by using the Bayes' Rule, we get:

$$P\left(w_i|O\right) = \frac{P\left(O|w_i\right) P\left(w_i\right)}{P\left(O\right)} \tag{4.53}$$

Given a set of prior probabilities $P(w_i)$, the most probable model (previously defined in our vocabulary) depends only on the likelihood $P(O|w_i)$.

Now, the unique problem is the estimation of the observations. Due to the high dimensionality of observation vectors, this process can't be done in a deterministic way. We will use a Markov model for that purpose.

Markov Model

A Markov model is a finite-state machine which changes state once every time unit. Each time t, the machine enters to a new state j and an observation vector o_t is generated by its probability density function $b_j(o_t)$. The transition between the previous state i and the actual state j is defined by the discrete probability value a_{ij}.

Fig 4.6 shows a 6 state Markov model example. Here, the state sequence $X = X1, X2 \dots X_8$ moves through the Markov model in order to generate the observation vector $O = o1, o_2, o_3, o_4, o_5, o_6, o_7, o_8$. The joint probability that O is generated by this specific model is:

$$P\left(O, X|M\right) = a_{12} b_2(o_1) a_{22} b_2(o_2) a_{23} b_3(o_3) \dots \tag{4.54}$$

Hidden Markov Models

Note that, in fact, the underlying X sequence is unknown. That is what it is called Hidden Markov Models.

The full HMM system is a set of Markov models (like shown in Fig. 4.6), and the purpose of Eq. 4.52 is try to find the model which best generates (that is, with the major probability) the observed sequence:

$$P\left(O|M\right) = \sum_X a_{x(0)x(1)} \prod_{t=1}^{T} b_{x(t)}\left(o_t\right) a_{x(t)x(t+1)} \tag{4.55}$$

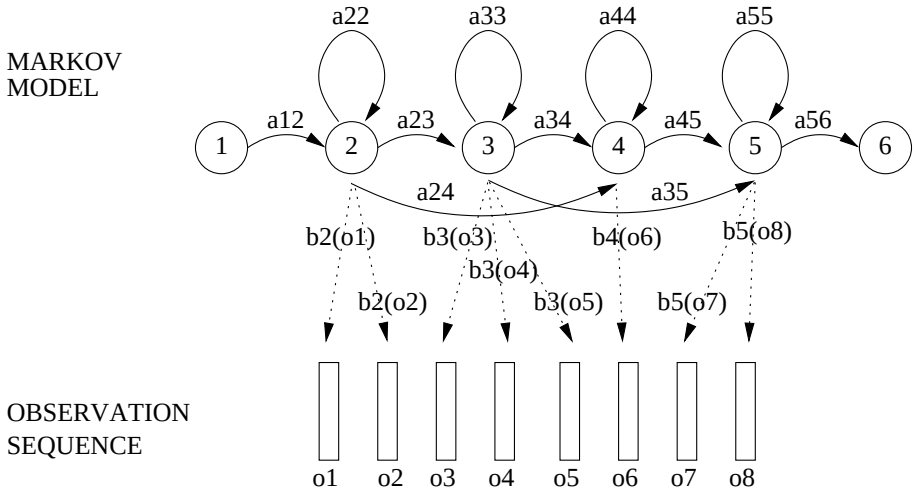


Figure 4.6: Markov Generation Model

where $x(0)$ is the model entry state and $x(t+1)$ is the model exit state. Here, M is the set of Markov models ($M = M_1, M_2 \ldots$), in the same sense that in Eq. 4.52, w_i pas presented as an specific model in our vocabulary.

It is obvious that Eq. 4.55 can not be computed directly. Some recursive techniques are needed. But whatever the recursive technique is, it supposes that $\{a_{ij}\}$ and $\{b_j(o_t)\}$ are known. We need some kind of training process for that.

4.6.4 Training Process

Given a set of examples for each model, all these parameters ($\{a_{ij}\}$ and $\{b_j(o_t)\}$) can be automatically estimated by using some re-estimation techniques. Then, the global system can be summarized as follows:

- Each model is trained with a sufficient number of examples of that specific model.
- When a new unknown input sequence has to be recognized, the likelihood of each model generating that input sequence is calculated.
- The recognized sequence belongs to the model that best generates the input sequence.

Gaussian Mixtures

Before entering in detail with the re-estimation process, note that, sometimes, the output probabilities $b_j(o_t)$ are represented as an addition of S independent data streams. Gaussian Mixture Densities are commonly used for that purpose.

From a mathematical point of view, let $b_j(o_t)$ be:

$$b_j(o_t) = \prod_{s=1}^{S} \left[\sum_{m=1}^{M_s} c_{jsm} N(o_{st}; \mu_{jsm}, \Sigma_{jsm}) \right]^{\gamma_s} \tag{4.56}$$

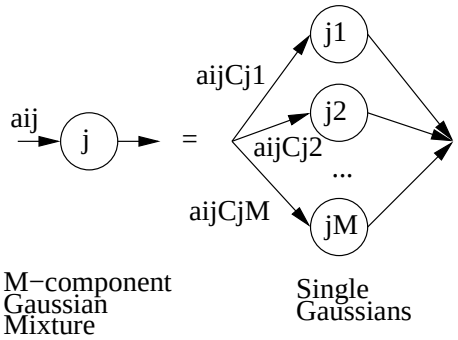


Figure 4.7: Decomposition of Gaussian Mixtures

where M_s is the number of mixture components in the stream s, c_{jsm} is the weight of the m'th. component, γ_s is the stream weight (it is usually a manual setting) and $N(\cdot; \mu, \Sigma)$ is a multivariate Gaussian with mean μ and covariance matrix Σ:

$$N(\cdot; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} e^{-\frac{1}{2}(o-\mu)' \Sigma^{-1} (o-\mu)} \tag{4.57}$$

where n is the dimensionality of o.

4.6.5 Baum-Welch Re-Estimation

In the training process, a good estimation for each model parameters is required. The Baum-Welch Re-Estimation algorithm improves these estimations providing more accurate results in our HMM system.

As seen in in Sec. 4.6.4, the output probabilities are usually represented as an addition of S mixture Gaussian components. For the study of the Baum-Welch Re-Estimation algorithm, only the case for one single Gaussian stream is considered. Note how multiple Gaussian mixtures can be interpreted as substates in each state in which the transition probabilities are exactly the mixture weights C_{jsm}. See Fig. 4.7 for details.

Then, the problem can be reduced to a mean and variance estimation of the output probabilities of a single-Gaussian HMM system:

$$b_j(o_t) = \frac{1}{\sqrt{(2\pi)^n |\Sigma_j|}} e^{-\frac{1}{2}(o_t-\mu_j)' \Sigma_j^{-1} (o_t-\mu_j)} \tag{4.58}$$

In the case of a single state HMM, the computation of these parameters is quite easy:

$$\hat{\mu}_j = \frac{1}{T} \sum_{t=1}^{T} o_t \tag{4.59}$$

and

$$\hat{\Sigma}_j = \frac{1}{T} \sum_{t=1}^{T} (o_t - \mu_j)(o_t - \mu_j)' \tag{4.60}$$

But in real case, the HMM is not usually single state and, furthermore, it is not possible to assign each observation sequence to a specific state. Then, the expressions 4.59 and 4.59 can be interpreted only as a non-random initialization, but they have to be redefined.

In order to solve this problem, let each sequence be assigned to each of the states in proportion to the probability of the model being in a state when the vector was observed. Then:

$$\hat{\mu}_j = \frac{\sum_{t=1}^{T} L_j(t) o_t}{\sum_{t=1}^{T} L_j(t)} \tag{4.61}$$

and

$$\hat{\Sigma}_j = \frac{\sum_{t=1}^{T} (o_t - \mu_j)(o_t - \mu_j)'}{\sum_{t=1}^{T} L_j(t)} \tag{4.62}$$

where $L_j(t)$ is the probability of being in state j at time t and the denominators in both expressions represent the normalization factors.

At this point, the also called probability of state occupation $L_j(t)$ must be computed, and the so called *forward-backward algorithm* will be used for this task.

Forward-Backward algorithm

Let $\alpha_j(t)$ be the *forward probability*:

$$\alpha_j(t) = P(o1, \ldots, o_t, x(t) = j|M) \tag{4.63}$$

where M is one of the models and N is the number of states for that model.

The meaning of this forward probability is the joint probability of observing the first t vectors and being at state j at time t. From a mathematical point of view, it can be calculated recursively as:

$$\alpha_j(t) = \left[\sum_{i=2}^{N-1} \alpha_i(t-1) \alpha_{ij} \right] b_j(o_t) \tag{4.64}$$

Note that states 1 and N are non-emitting states, that is, no output probability can be calculated from them. They are useful for state-transition problems.

The initial conditions for recursion are:

$$\alpha_1(1) = 1 \tag{4.65}$$

and

$$\alpha_j(1) = a_{1j} b_j(o_t) \qquad 1 > j > N \tag{4.66}$$

and the final condition is:

$$\alpha_N(T) = \sum_{n=2}^{N-1} \alpha_i(T) \alpha_{iN} \tag{4.67}$$

Then,

$$P(O|M) = \alpha_N(T) \tag{4.68}$$

In a similar way, let $\beta_j(t)$ be the *backward probability*:

$$\beta_j(t) = P(o_{t+1}, \ldots, o_t | x(t) = j, M) \tag{4.69}$$

This probability can be calculated as:

$$\beta_i(t) = \sum_{j=2}^{N-1} a_{ij} b_j(o_{t+1}) \beta_j(t+1) \tag{4.70}$$

The initial condition is:

$$\beta_i(T) = a_{iN} \qquad 1 < i < N \tag{4.71}$$

and the final condition is:

$$\beta_1(1) = \sum_{j=2}^{N-1} a_{ij} b_j(o_1) \beta_j(1) \tag{4.72}$$

Then, we get the state occupation probability by multiplying the forward probability (joint probability) and the backward probability (conditional probability):

$$\alpha_j(t)\beta_j(t) = P(O, x(t) = j|M) \tag{4.73}$$

Finally,

$$\begin{array}{rcl} L_j(t) & = & P(x(t) = j|O, M) \\ & = & \frac{P(O, x(t)=j|M)}{P(O|M)} \\ & = & \frac{1}{P}\alpha_j(t)\beta_j(t) \end{array} \tag{4.74}$$

where $P = P(O|M)$. Note that these operations involves probability multiplications, and resolution can affects the final results. Hence, these values are usually computed in log arithmetic.

4.6.6 Viterbi Decoding

In Sec. 4.6.5, an efficient method for computing the forward probability is shown. When new data comes to the system and the observation vectors are extracted, the same algorithm could be applied successfully for recognition, that is, find the model which yields the maximum value of likelihood $P(O|M_i)$. But this method has a disadvantage: if one transition's probability is zero ($a_{i,j} = 0$), it is possible that the most probable state in a given time unit t belongs to an unexisting path.

This problem can be solved by computing the algorithm but calculating the maximum likelihood state sequence instead of the maximum probability state sequence. It is computed by using the same algorithm, but the summation is replaced by a maximum operation.

Let $\phi_j(t)$ be the maximum likelihood of observing vectors o_1 to o_t, and being at state j at time t. It can be computed, in a similar way than Eq. 4.64 does, as:

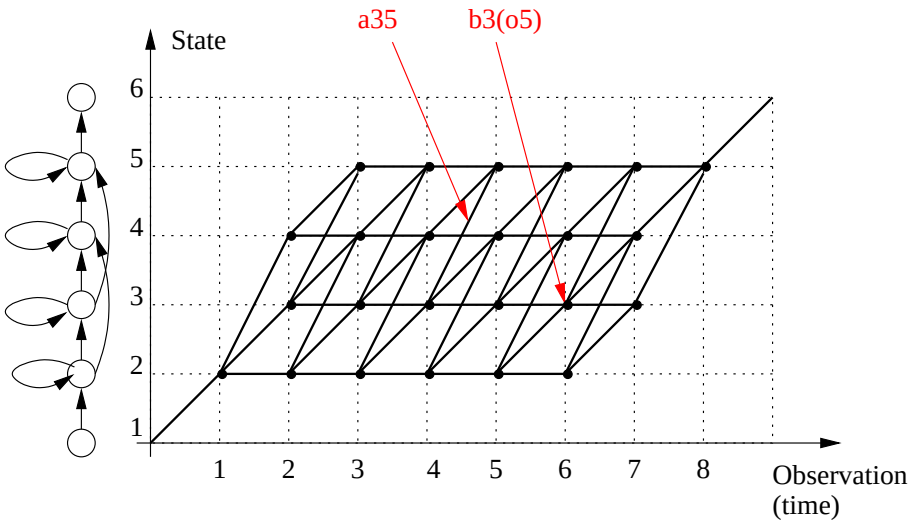


Figure 4.8: Viterbi Algorithm

$$\phi_j(t) = \max_i \{\phi_i(t-1)a_{ij}\}\, b_j(o_t) \tag{4.75}$$

The initial conditions are:

$$\phi_1(1) = 1 \tag{4.76}$$

and:

$$\phi_j(1) = a_{1j} b_j(o_1) \qquad i < j < N \tag{4.77}$$

The final condition is:

$$\phi_N(T) = \max_i \{\phi_i(T)a_{iN}\} \tag{4.78}$$

Then, the maximum likelihood is:

$$\hat{P}(O|M) = \phi_N(T) \tag{4.79}$$

As for the Baum-Welch algorithm, these calculations leads to underflow. Log arithmetics are also used in this case:

$$\psi_j(t) = \max_i \{\psi_i(t-1) + \log(a_{ij})\} + \log(b_j(o_t)) \tag{4.80}$$

In Fig. 4.8 the Graphical interpretation of the Viterbi algorithm is shown.

4.7 Mathematical Morphology

4.7.1 Introduction

Mathematical Morphology is a set of non-linear signal processing techniques proposed by Serra in [62]. They are based on the maximum and minimum operations.

Morphology has several advantages over other techniques especially when applied to image processing. Here are some of them:

- Preserves edge information
- Works by using shape-based processing
- Can be designed to be idempotent
- Computationally efficient

Morphology has been used in a wide range of applications. Some possible applications are:

- Image enhancement
- Image restoration (i.e. removing scratches from digital film)
- Edge detection
- Texture analysis
- Noise reduction

Although Mathematical morphology techniques are specially designed for image processing, they can also be applied to audio processing. In the next sections, a brief description of Mathematical Morphology is shown.

4.7.2 Basic Structures

The basic structure is the *complete reticulum*. The complete reticulum are those sets P that:

$$\begin{aligned} A \geq A, \quad A \in P \\ (A > B) \bigcup (B > A) \implies A = B \\ (A > B) \bigcup (B > C) \implies A > C \end{aligned} \tag{4.81}$$

Then,

1. There exists a partial relationship between the elements
2. For all the sets of elements $\{X_i\} \in P$ there exists a *Supremum* and a *Infimum* defined as:
   (a) Supremum: $\vee\{X_i\}$
   (b) Infimum: $\wedge\{X_i\}$

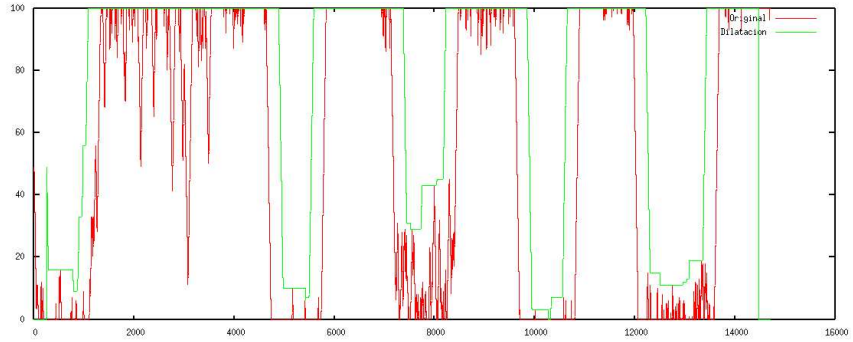


Figure 4.9: Comparison between an original signal and the dilated signal

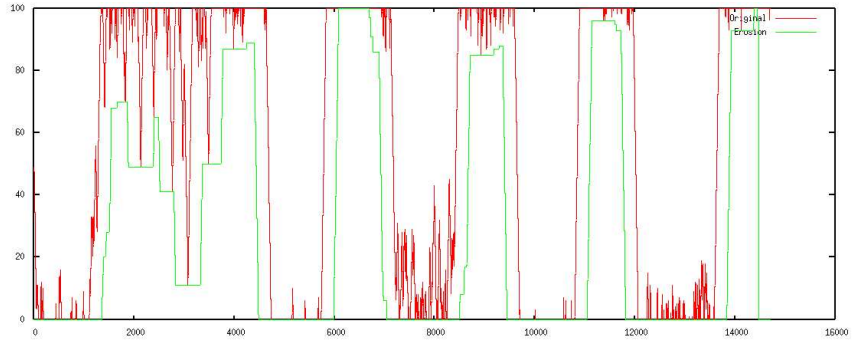


Figure 4.10: Comparison between an original signal and the erosioned signal

4.7.3 Dilation and Erosion

Two basic operations can be defined over the Supremum and Infimum laws:

1. Dilation: Operation over the supremum:

$$\varphi\left(\vee\{X_i\}\right) = \vee\{\varphi\left(X - i\right)\} \tag{4.82}$$

   The resulting effect of this operation is shown in Fig. 4.9.

2. Erosion: Operation over the Infimum:

$$\varphi\left(\wedge\{X_i\}\right) = \wedge\{\varphi\left(X - i\right)\} \tag{4.83}$$

   The resulting effect of this operation is shown in Fig. 4.10.

4.7.4 Opening and Closing

Opening and Closing operations are defined by the successive application of the dilation and erosion operations defined above:

1. Opening: Successive application of the erosion and dilation operations.
   (a) Removes all the positive peaks of the original signal that are smaller than the structural element.

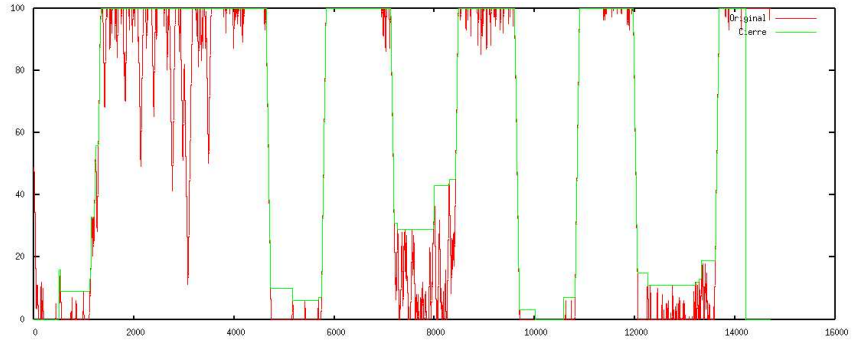


Figure 4.11: Comparison between an original signal and the *opened* signal

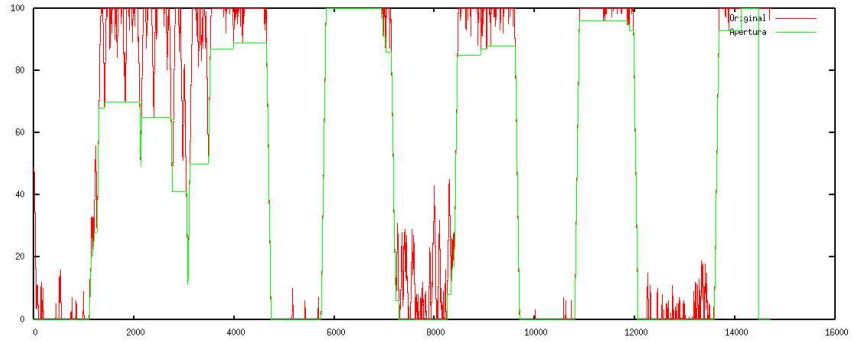


Figure 4.12: Comparison between an original signal and the *closed* signal

(b) The resulting function is always below the original one.

The effect of this operation is shown in Fig. 4.11.

2. Closing: Successive application of the dilation and erosion operations.

   (a) Removes all the negative peaks of the original signal that are smaller than the structural element.

   (b) The resulting function is always above the original one.

   The effect of this operation is shown in Fig. 4.12.

It is obvious that both process are in opposition. This characteristic will be used in the Speech-Music discrimination system described in Sec. 5.3

Chapter 5

First Contributions

In this chapter, firsts contributions to the Music Information Retrieval field are shown. The chapter is divided in five sections. In the first section, the context for all this work and the used framework will be reviewed. In the second section, some new descriptors are introduced. The third section deals with the Speech-Music discrimination system based on the descriptors just described. This Speech-Music discrimination system is presented as the first step to build up a more complex Automatic Genre Classification System that will be shown in the fourth section. Finally, a new Rhythm Similarity System is also presented in the fifth section.

5.1 Environment

Most of this research has been made in the context of the AIDA project[1], and some little applications have been tested in the MTG-DB system, described by Cano et. al. in [10]

5.1.1 AIDA Project

The major goal for the AIDA project is the automatic recognition of broadcast audio. This process may appear quite simple but it becomes more and more complicated when huge audio databases are managed. Some techniques are applied for reducing the large amount of data although they increase the complexity of the system [53].

Hidden Markov Model techniques are used for this purpose. By using HMM, the system is not a TRUE/FALSE identification process, but a non-linear similarity measure. In this context, we find the *identified song* as the *most similar* song. When this *most similar song* is found under some other constrains, it is considered as the *identified song*. This technique is quite useful for other similarity applications such as rhythmical similarity.

In this framework, the Similarity or Identification process will depend on the used descriptors. If only timbrical descriptors are used, the identification process will be performed from a timbrical point of view. Then, it is a good

[1]The AIDA project has been founded by the Sociedad Digital de Autores y Editores - SDAE

idea to include many different kinds of descriptors: identification or similarity process will be more accurate and robust.

Concerning the robustness, the system must be robust to multiple distortions from the input signal as well. Is widely known that almost all the radio-stations apply different distortions to the audio signal in order to increase the listener's attention. The most common radio distortions are Compressor/Limiter, Stereo Base-width, Exciter/Enhancer and Pitching. Furthermore, the system must also be robust to GSM codification. This problem is difficult to solve and a lot of considerations must be taken into account.

On the other hand, the system must be source-independent. Different sampling frequencies, bit depth or codification must not affect the robustness of the system. Signals from cellular phones must be identified as well as MP3 files or direct real-time streaming.

5.1.2 AMADEUS technology

All this work have been implemented using the AMADEUS technology. AMADEUS is a set of C++ classes that provides all the needed functions for real-time input data, parameterization, and HMM operations. AMADEUS has been implemented at the Music Technology Group (MTG) and it is available on the web. See [20] for a more detailed explanation about the possibilities of AMADEUS.

5.2 New descriptors

5.2.1 Voice2White descriptor

It is well known that speech data has a limited frequency band, from 300$[Hz]$ to 4000$[Hz]$ approximately. The Voice2White descriptor is a measure of the energy inside this limited frequency band in relation to the whole audible margin. This will give us an idea about how *speechy* is the input audio data (See Fig. 5.1).

From a mathematical point of view, it is a measure of the energy of the audio inside the typical speech frequency band ($300Hz..4KHz$) respect the energy of the whole audible margin (in case of $sr = 44100Hz$) or the global band (in case of $sr < 44100Hz$). The process is similar to the Spectral Flatness described in Sec. 4.1.4, but the final calculations are slightly different:

$$v2w = 10log_{10}\frac{\sum_{f_i=300}^{4000} B_{f_i}}{\sum_i B_i} \tag{5.1}$$

where B_i is the energy value of the corresponding critical band, and f_i is the critical band containing that frequency value.

5.2.2 The rhythmical transformation

Many rhythmical descriptors from input audio data can be computed. As shown in Sec. 4.2, most of them depend on some manually fixed parameters or experimental thresholds and they only give a partial point of view about the whole rhythm, as explained in Chapter 2. The so called *Rhythm Transform* pretends to be the solution to all these problems: a complete rhythmical representation of the input signal without using thresholds based on experimental values.

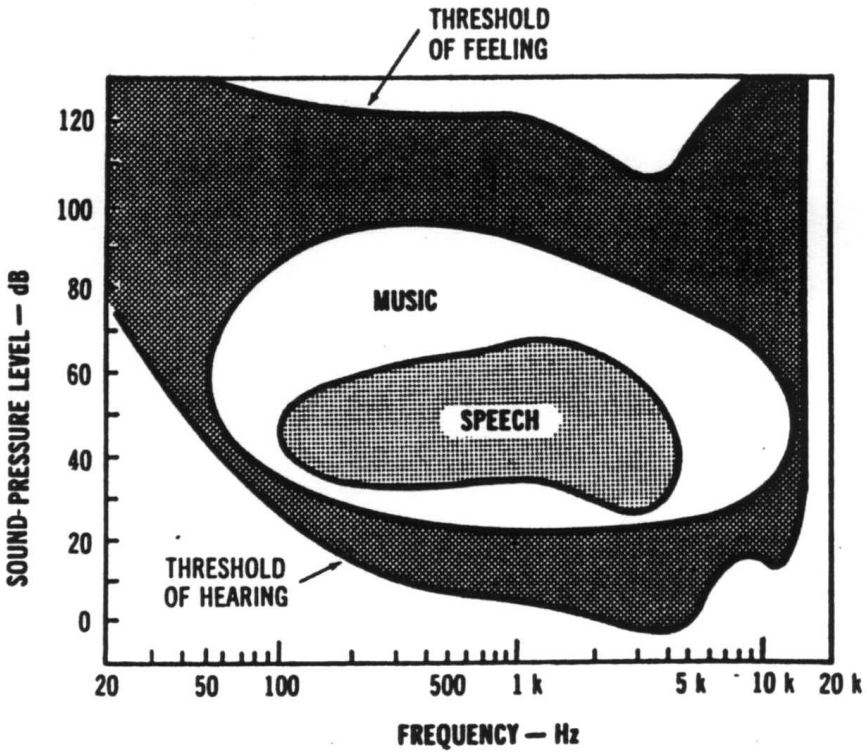


Figure 5.1: Frequency and Loudness limits for speech

Note that we call *rhythm transform* from a conceptual point of view. It is not a real transform from a mathematical point of view since the inverse transform can not be defined. But the obtained data could be interpreted as data in the so called *rhythm domain*.

Rhythm transform

Most of the beat tracking systems compute the frequency analysis of the input signal and search for the common energy periodicities through different (linear o mel-frequency based) sub-bands. The energy's periodicity search is usually implemented as a bank of resonators and represented as a Beat Spectrum or as a Beat Histogram. The Rhythm Transform is slightly different: the periodogram is calculated for the energy derivative of each sub-band of the input data and, finally, a weighted sum is implemented for a global rhythm representation:

Frequency decomposition: The input data $x(t)$ is filtered with the anti-alias filtering and sampled with $f_s = 22050[Hz]$. The length of the frames is $l = 300[ms]$ according to perceptual behavior of the ear[44], the hop-size is $h = 30[ms]$ and Hamming windowing is applied. Digital windowed data $x_w[n]$ is decomposed into different sub-bands with a 1/3 octave filter bank according to perceptual behavior of the human ear. At this point, different

digital signals are obtained:

$$\begin{array}{rcl} x_{f_c=20[Hz]}[n] & = & \frac{1}{N_1}\sum_{f=17.8[Hz]}^{f=22.4[Hz]}|x_w[n]| \\ x_{f_c=25[Hz]}[n] & = & \frac{1}{N_2}\sum_{f=22.4[Hz]}^{f=28.2[Hz]}|x_w[n]| \\ & \cdots & \\ x_{f_c=10000[Hz]}[n] & = & \frac{1}{N_{28}}\sum_{f=8913[Hz]}^{f=11220[Hz]}|x_w[n]| \end{array} \tag{5.2}$$

where N_i is the number of points of the FFT inside each 1/3 octave band.

Energy Extraction: The log of the energy is obtained for each band:

$$\begin{array}{rcl} e_{f_c=20[Hz]}[n] & = & log_{10}\left(x_{f_c=20[Hz]}[n]\right) \\ e_{f_c=25[Hz]}[n] & = & log_{10}\left(x_{f_c=25[Hz]}[n]\right) \\ & \cdots & \\ e_{f_c=10000[Hz]}[n] & = & log_{10}\left(x_{f_c=10000[Hz]}[n]\right) \end{array} \tag{5.3}$$

Derivative of the Energy: The derivative of the energy is computed:

$$\begin{array}{rcl} d_{i,f_c=20[Hz]}[n] & = & e_{i,f_c=20[Hz]}[n] - e_{i-1,f_c=20[Hz]}[n] \\ d_{i,f_c=25[Hz]}[n] & = & e_{i,f_c=25[Hz]}[n] - e_{i-1,f_c=25[Hz]}[n] \\ & \cdots & \\ d_{i,f_c=10000[Hz]}[n] & = & e_{i,f_c=10000[Hz]}[n] - e_{i-1,f_c=10000[Hz]}[n] \end{array} \tag{5.4}$$

and all these values have a length of $L = 6[s]$, which is the worst case for a full 4/4 bar at 40 BPM.

Periodogram calculations: The periodogram is computed for each buffer, as explained in Sec. 4.4. Then,

$$\begin{array}{rcl} I_{f_c=20[Hz]}[\omega] & = & \frac{1}{LU}\sum_{m=-(L-1)}^{L-1} c_{vv,f_c=20[Hz]}[m]e^{-j\omega m} \\ I_{f_c=25[Hz]}[\omega] & = & \frac{1}{LU}\sum_{m=-(L-1)}^{L-1} c_{vv,f_c=25[Hz]}[m]e^{-j\omega m} \\ & \cdots & \\ I_{f_c=10000[Hz]}[\omega] & = & \frac{1}{LU}\sum_{m=-(L-1)}^{L-1} c_{vv,f_c=10000[Hz]}[m]e^{-j\omega m} \end{array} \tag{5.5}$$

where c_{vv,f_i} is the aperiodic correlation sequence of each d_{f_i} sequence.

Weighted sum: Finally, the weighted sum for all the periodograms for each band is computed. The weighting vector is:

$$r[1..nBands] = \left\{\frac{1}{nBands}, \frac{1}{nBands}, \ldots\right\} \tag{5.6}$$

but it can be manually modified in order to emphasize some frequency bands. For general pop music, where the rhythm is basically played by Bass and Bass drums, it can be any descendent sequence, i.e.:

$$r[1..nBands] = \left\{\frac{1}{1}, \frac{1}{2}, \ldots\right\} \tag{5.7}$$

and for some kind of Latin music, where some high-frequency instruments are usually played, the weighting vector could be:

$$r[1..nBands] = \left\{\frac{1}{1}, \frac{1}{2}, \ldots, \frac{1}{9}, \frac{1}{10}, \frac{1}{9}, \ldots, \frac{1}{2}, \frac{1}{2}\right\} \tag{5.8}$$

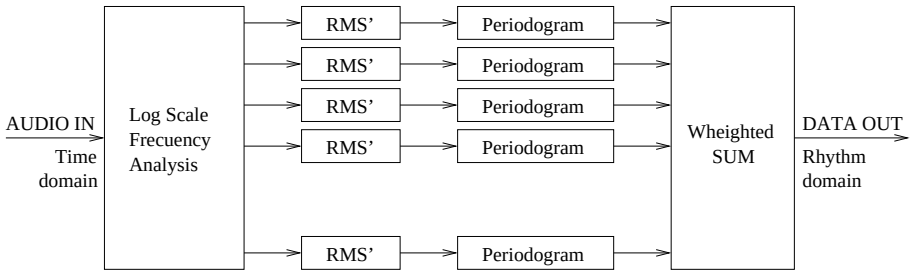


Figure 5.2: Block diagram for *Rhythm Transform* calculation

(a) Strong beats → low per. (b) Weak beats → higher per.

Figure 5.3: Periodicities of a musical signal

But in a general way, the weighting vector described in Eq. 5.6 is good enough.

By performing the weighted sum of the squared values of all the periodograms, data in the *rhythm domain* is obtained:

$$T(\omega) = \sum_{j=1}^{nBands} r(j) I_{f_j}[\omega] \tag{5.9}$$

In Fig. 5.2 the global block diagram for Rhythm Transform computations is shown.

Interpretation of data in *Rhythm Domain*

Which information is available from data in the *Rhythm Domain*? The BPM information can be found as the greatest common divisor for all the representative peaks since the beat can be defined as the common periodicity of the energy peaks for all the instruments in a song. For BPM detection, any peak detection algorithm across data in rhythm domain data can be used.

But the major advantage of this representation is that it gives some time domain information too. Let's see this duality from a conceptual point of view: It is well known that music is structured in bars, in a given meter. It is well known that the strongest beat in a bar is usually the first one. This means that this strongest beat in a bar appears less frequently: it has the minor periodicity. On the other hand, a weak beat will appear more frequently since it has a higher periodicity (see Fig. 5.3)

In rhythm domain, weak beats appear at higher BPM than strong beats. Furthermore, in time domain, weak beats appear later than strong beats too.

This correspondence allows to interpret data in rhythm domain *as* data in time domain. This is what we call *duality* of data in rhythm domain and time domain.

Assuming this duality, the time signature from audio data can easily be deduced. Data between two higher peaks can be seen as the distribution of the beats in a bar. If data between two maximum peaks is divided by twos, a simple meter is assumed as it is shown in Fig.5.4(a) and Fig. 5.4(b). If data between two maximum peaks is divided by threes, a compound meter is assumed as it is shown in Fig. 5.4(c) and Fig. 5.4(d). On the other hand, if data in a simple or compound bar is sub-divided by twos, a duple meter is assumed as it is shown in Fig. 5.4(a) and 5.4(c), and if this data is sub-divided by threes, a triple meter is assumed as it is shown in Fig. 5.4(b) and 5.4(d). Finally, in Fig. 5.4(e) if a simple duple meter is sub-divided by twos, the presence of swing is assumed (Let the swing structure as a dotted quarter-note and a eight-note).

All these examples (available in `http://www.iua.upf.es/~eguaus`) are based on polyphonic MIDI generated audio data, except for the last example which is directly extracted from the CD.

In conclusion, the main advantage of this method is that we have much more information than the information available at the output of a set of resonators tuned at the different BPM typical values and a unique frequency value is related to a unique BPM value. The BPM resolution is higher than other methods, and furthermore, we have not only the BPMs, but all the existing periodicities as well according to different human aspects of music. Different rhythms with the same meter and structure, but played with different *feeling* can be distinguished by using this Rhythm Transform.

Limitations

Sometimes, tempo is only defined by pitch variations. This descriptor fails for those cases with non-attack instruments. Strings, choirs, synthetic pads, etc. are not good friends of the Rhythm Transform.

Furthermore, the Rhythm Transform is limited by FFT resolution. For low BPM values, the periodicity is low, then the subdivisions by twos or threes will be much closer than the distance between two bins.

5.2.3 Beatedness descriptor

The *beatedness* calculation is an application on the use of data in *Rhythm domain*. This concept was introduced by Foote et al in [24] and evaluated by Tzanetakis et al in [70]. The Beatedness is a measure of how strong are the beats in a musical piece. The beatedness is computed as the Spectral Flatness of the sequence but in the rhythm domain. Spectral Flatness is a measure of the tonality components in a given spectrum, and it is defined as:

$$SF_{dB} = 10 \cdot \log \frac{G_m}{A_m} \tag{5.10}$$

where G_m and A_m are the geometric an arithmetic mean values from all the bins of the Fourier Transform of the signal, respectively. In the case of the Beatedness computation, G_m and A_m are the geometric an arithmetic mean values from all the bins of data in rhythm domain.

Genre	BPM_{mean}	BPM_{var}	Beatedness
Dance	140	0.49	3.92
Pop	108	0.71	5.23
Soul	96	0.43	3.48
Jazz	132	1.26	3.54
Classic	90	32	0.95
Voice	66	46.9	0.36

Table 5.1: BPM and Beatedness for different musical genres

High beatedness values are due to very rhythmic compositions, this is Dance, Pop ... Low beatedness values are due to non rhythmical compositions, this is some Jazz Solo, classical music, speech ...

Some BPM & Beatedness measures for different musical genres are shown in Table 5.1. All these measures belong to one frame of "No Gravity" by DJ Session One for Dance music, "Whenever,Wherever" by Shakira for Pop music, "Falling" by Alicia Keys for Soul music, "Summertime" by Gershwin for Jazz music, "Canon" by Pachelbel for Classic music and one minute of radio recording for voice. Note that in *Dance*, *Pop* and *Soul* music, the the system shows us a quite low BPM_{var}. That means that the BPM measure is successful. Not the same for *Classic* and *Voice*, but it's not an error: Do some excerpts of classic music or speech have any tempo?

Numerically, high values are due to rhythmic music and low values are due to *Classic* music or *Voice*.

5.3 Speech-Music discrimination

5.3.1 Introduction

The Speech-Music discrimination problem is not new, and some of the developed systems have been quite successful. But all these systems are designed under some rigid constrains (see Sec. 5.3.2). This is the main goal of this system: our environment has not any kind of constrains. Broadcast audio signals (from radio stations, TV, GSM or Internet) are the input of our system and, as one can imagine, the content of this data is out of control. Mainly, we have two different kind of problems:

Channel distortions: Since the sources of audio can be very different, the system is not focused on clean audio files. The used descriptors are strategically chosen to be, for instance, independent of the spectrum of the input signal. Then, MP3 or GSM codifications of audio input will not produce a failure to the system

Audio content Since the content of the system is unknown, the system has to discriminate between speech and music in many different situations: interviews, films, commercials, sports...

How do we design this system? The main idea is to create a *genre classification system* in which the musical genres are conceptually different with regard to

the typical ones. For instance, a "cellular male voice" could be a genre for our system. Furthermore, the representative features of the input signals are not only based on the spectrum of the signal, but on the rhythm.

5.3.2 State of the art

The Speech-Music discrimination problem has become quite important for last years due to the automatic indexing or classification problem. Multimedia data identification and indexation has become more and more important due to the fast growth of electronic databases through Internet. Large amount of data must be automatically analyzed, and speech-music discrimination is only one step for the whole process.

Main characteristics

A lot of studies have been done in Speech-Music Discrimination. The previous work can be summarized in three different categories:

- Time domain based systems, such as zero-crossing or energy-evolution.
- Frequency domain based systems, such as cepstral coefficients.
- Mixed time-frequency domain based systems, such as 4Hz Modulation or harmonic coefficients.

This classification can be done just taking into account which kind of parameters are extracted from the input signal. But one could think in other classification scheme just taking into account the signal processing method performed to that data. Then, the Speech-Music discrimination systems can be divided in:

- Decision trees
- Neural Networks
- Gaussian Mixture Models
- Hidden Markov Models

All of the systems we will talk about in the next section should be enclosed into one of the categories of the first classification scheme as well as into one of the categories of the second classification scheme, independently.

Related work

The first successful approach on the speech-music discrimination was made by John Saunders in 1996 [59]. In this study, Saunders compare different features like the *Tonality*, *bandwidth*, *excitation patterns*, *tonal duration* and *energy sequences*. The Zero-crossing rate is introduced as a significant parameter in the speech-music discrimination, and some experiments and results are presented.

Another important approach was made by Eric Sheirer and Malcom Slaney in [60]. This work presents a study of thirteen different features, derived from 8 original ones (4HzModulation, Spectral Centroid, Cepstrum, Pulse Metric,

style	description	main group
mal	Male voice	speech
fem	Female voice	speech
cma	Cellular male voice	speech
cfe	Cellular female voice	speech
cla	Classical music	music
cop	Copla & Author music	music
ele	Electronic soft music	music
jaz	Jazz music	music
pop	Pop music	music
roc	Hard rock music	music
tec	Tecno & Dance music	music
sil	Silence	speech

Table 5.2: Definition of different styles for Speech-Music discrimination

etc.). Each one of them is supposed to be a good discriminator at once. Different sets of features have been trained and tested by using Single Gaussian Mixture Models, but the results are not spectacular. The conclusion of this work deals with more research: no significant good results are found.

At this point, the basic features and procedures for discrimination are presented, and future works will only introduce new features or little deviations of these main ideas. Wu Chow and Liang Gu present us a set of features derived from Harmonic Coefficient and its 4Hz Modulation in [12]. This approach is based on a two-level processing structure, one for singing/non singing musical signals detection and the other for the typical speech-music discrimination. After a rule-based post-filtering smoothing algorithm, significant enhancements are obtained for complex audio streams. Karnebäck presents an exhaustive study on Low Frequency Modulations in [64], and Berenzweig and Ellis present some new statistical features (defined in [29]) embedded in a simple HMM for distinguishing between singing and instrumental music in [4].

Some comparisons on the methods mentioned above have been made. M. J. Carey [47] has tested most of those different features. The cepstral coefficients and delta cepstral coefficients seem to be the most successful parameters, while the zero-crossing and the energy (mean and variance values across the time) are not so important. Cepstra and delta cepstra can give us an equal error rate of about 1.2%, slightly far of the 6% of equal error rate by using the zero-crossing coefficients.

Finally, a study of the State of the Art has been made by the *Audio Research Group* in *Tampere University of Technology.*

5.3.3 Description

In few words, the systems is designed as a rough musical genre classification. But the genres have not any special musical meaning. A genre is, from our point of view, a group of audio signals with some common (spectral, timbrical or rhythmical) features. The selected genres are grouped into two main groups, *Speech* or *Music*, as shown in Table 5.2.

Why do we need so many different styles? As we will discuss later, the system is based on different features. It is obvious that these features are quite different between Hard Rock and Classical music. Then, we apply the "divide and conquer" principle. Different models will be defined for different music styles, so the models will hold more accurate descriptions of the music. The speech-music discrimination system is, in fact, a genre classification system in which only two of the genres are interesting: *mal* and *fem*.

The developing process is clearly divided in five steps:

- Data acquisition
- Parameterization of both training and test audio data.
- Training Process
- Real-time Recognition and post-processing.
- Graphical user interface.

The training process is made by using the HTK software and the whole process is oriented to the HTK philosophy. The Wavesurfer software is also used for creating labels.

Data Acquisition

The system works on real-time. The AMADEUS technology described in Sec. 5.1 is used for this purpose.

But a lot of audio data must be recorded and manually labeled for a successful training process. As the system will work in a broadcast audio environment (from many different radio stations), many excerpts of radio broadcast audio have been recorded. Finally, this data have been edited and many different audio files have been produced. All these audio files are 1 minute long with $f_s = 22050Hz$, 16 bits and mono. But the main characteristic of these audio files is that they belong into one specific musical genre, that is, each file belongs exclusively to an specific genre from the beginning to the end. Finally, Different HMM models will be defined for different genres, so the models will hold more accurate descriptions of the music.

Parameterization

From now on, we have a lot of audio files, and each one of them can be associated with a specific musical genre. The parameterization process should transform all these audio files into a set of description files. Table 5.3 shows us the available descriptors we can use, and we describe the right selection in Sec. 5.3.4.

The parameterization is made by an *AMADEUS* application which generates an HTK-format file `*.htk` for each audio file.

At this point, all this parameterizated data have to be labeled according to the styles defined in Table 5.2. For each one of the `*.wav` files there will exist, a part of the `*.htk` file mentioned above, a `*.lab` in which the style is specified to the system. The `*.lab` file follows the next structure:

$$\underbrace{0}_{begin} \quad \underbrace{135007423}_{end} \quad \underbrace{pop}_{style}$$

Descriptor	L	Value	∇	∇^2
MFCC	12	*	*	*
Energy	1	*	*	*
4Hz Modulation	1	*	*	*
Zero Crossing Rate	1	*	*	*
Spectral Centroid	1	*	*	*
Spectral Flatness	1	*	*	*
Voice2White	1	*	*	*

Table 5.3: List of available descriptors for Speech-Music Discrimination

where *begin* and *end* are the beginning and the end positions where the label is valid, the whole file in our case. These values are written in a sample units:

$$1323000 = 60[sec] * 22050 \frac{[samp]}{[sec]} \tag{5.11}$$

These files are hand-made by using the `WaveSurfer` software.

Training Process

From now on, we have a set of `*.wav`, `*.htk` and `*.lab` for each file (audio, parameterization and labels respectively). After some experimental tests, the best results are obtained by using models as follows:

Number of means and variances per state: This value is fixed by the descriptors' selection.

Number of States: Our system will have 3 states, that is, only one state plus the input state plus the output state.

Number of Gaussian Mixtures: The model will be created, initially, with only 1 Gaussian mixture. After the initialization, we will increase the number of Gaussian Mixtures up to 16.

Left to right model: The Transition matrix will not allow backward paths.

Finally, the training process is started and the trained models are saved for a future use in our real-time application.

Real-time recognition and post-processing

As we have discussed before, the use of HMM does not give us a digital output, that is, a *Speech* or *Music* label. Then, some post-processing techniques are needed. After some tests and discussions, Mathematical Morphology techniques are used. Mathematical Morphology[62] is a set of non-linear techniques based on *maximum* and *minimum* operators (See Sec. 4.7 for details).

In Fig. 5.5 there is a comparison between the *opening* and *closing* operations applied to an input signal. Low values are assigned to *Speech* and high values are assigned to *Music*. In the context of the AIDA project, we have to assume that not false positive values are allowed. Let's define a "false positive hit"

when the label *Speech* is the output of the system for a musical signal at the input. On the other hand, as we can see in Fig. 5.5, the opening operation gives results below the original points, while the closing operation gives results above the original points. Then, the input signal can be cataloged as *Speech* under two conditions:

1. The result of the opening operation is exactly 0%.
2. The result of the closing operation is under a threshold, manually selected (5% in our case).

Graphical User Interface

The system is implemented by using the AMADEUS technology. Some graphical results can be shown (see Fig. 5.6). This monitoring tool is a real-time implementation in a Pentium IV, 2.5MHz, 512Mb RAM and Red-Hat 9 operating system.

5.3.4 Results

First approach

The first tests we made were based on all the descriptors shown in Table 5.3. This is a really large amount of data and, of course, the process could not be executed in real-time. Then, some tests were made in order to select only the representative descriptors and get an optimized version. The parameters used in each test are shown in Table 5.4, and the results of the tests for all the configurations and all the audio files are shown in Table 5.5. The tests are made against a set of 11 real audio recordings. Each audio file is 10 minutes long and manually labeled for this purpose.

All these preliminary tests (except for the last one) have been made with the next properties: $f_s = 22050[Hz]$, $Frame = 200[ms]$, $hopsize = 50[ms]$, 3 state model (1 state + input state + output state) and 16 mixture models.

The best results are obtained with the AB descriptors combination, getting an accuracy of 83.0% (The AC set of descriptors combination has been discarded for computational problems).

These are not really good results. Some short-time false positive hits makes the accuracy to fall down. This problem will be arranged with the post-processing techniques.

Inclusion Rhythm Transform

The inclusion of the *rhythm transform* descriptor in our experiment is to make both the system robust against frequency manipulations and increase the accuracy for Classical Music. Previous audio test files have no excerpts of Classical Music. In fact, the system labels as *Speech* the classical music files. The test is configured with the descriptors shown in Table. 5.6.

The used files are the same than those defined for previous experiments, but with some excerpts of classical music included. Then, the length of the files is now about 15 minutes. Results are shown in Table 5.7.

#Id	Parameters used
A	MFCC
B	Energy
C	Δ Energy
D	Δ^2 Energy
E	4Hz Modulation
F	Δ4Hz Modulation
G	$\Delta^2$4Hz Modulation
H	Spectral Centroid
I	ΔSpectral Centroid
J	Δ^2Spectral Centroid
K	Spectral Flatness
L	ΔSpectral Flatness
M	Δ^2Spectral Flatness
N	Zero Crossing
O	ΔZero Crossing
P	Δ^2Zero Crossing
Q	Voice to White
R	ΔVoice to White
S	Δ^2Voice to White
T	$A + B + C + \ldots + S$
U	$A, E, H, K, N, Q, \Delta MFCC, \Delta^2 MFCC$
V	$N + O$
W	$U + O$
X	$H + I$
Y	$K + L$
Z	$Q + R$
AA	$E + F$
AB	$U + C + F + I + L + O + R$
AC	AB with $frame = 1000[ms]$

Table 5.4: Descriptors used for initial tests in the Speech-Music discrimination system

Although results are less impressive than the previous ones, classical music can be included in our system. On the other hand, the error is basically introduced for the short-time false positive hits.

Inclusion of Mathematical Morphology

As mentioned before, non-linear mathematical morphology techniques are applied in the post-processing part of the system. With the inclusion of these techniques, we can avoid the system fails for short-time false positive hits. The "short-time" period is selected according to the length of the structural element for the *opening* and *closing* operators. Furthermore, with mathematical morphology techniques applied, we can exactly define the point in which the system labels the input audio as *Speech*. We won't consider as an error all those very little audio excerpts with speech, music or both speech and music (news,

#Id	1	2	3	4	5	6	7	8	9	10	11	$\bar{x}$	τ
A	72.3	68.5	69.1	81.6	65.7	66.5	60.3	71.9	70.7	74.6	76.4	70.69	5.46
B	46.8	54.2	55.2	51.0	60.6	53.6	63.8	53.0	54.3	65.0	54.8	55.66	5.17
C	40.7	49.0	52.2	45.7	51.9	44.7	58.3	50.5	50.3	60.7	48.7	50.06	5.17
D	40.7	49.0	52.2	45.7	51.9	44.7	58.3	50.5	50.3	60.7	48.7	50.20	5.38
E	40.7	48.9	50.7	45.7	51.9	44.7	58.1	50.5	50.3	60.7	48.6	50.08	5.39
F	40.8	49.1	52.2	45.8	52.0	44.8	58.2	50.6	50.1	60.6	48.6	50.20	5.39
G	42.3	48.7	49.5	47.5	49.5	45.7	54.8	47.9	49.4	56.2	48.4	49.08	3.64
H	50.3	55.2	74.7	58.2	72.2	57.8	65.0	65.1	51.7	75.5	66.3	62.90	8.51
I	40.7	49.0	49.1	45.7	51.7	44.7	58.3	50.5	50.3	60.7	48.7	49.94	5.42
J	40.7	49.0	44.8	45.7	45.7	44.7	58.3	50.5	49.1	60.7	48.7	48.90	5.66
K	63.7	50.8	66.3	83.3	67.4	67.7	70.4	77.5	55.9	80.8	76.9	69.10	9.67
L	53.2	49.2	52.1	52.9	51.7	52.0	48.5	49.3	51.5	52.4	52.3	51.37	1.53
M	44.7	50.5	52.0	49.1	51.8	50.8	57.4	54.2	54.0	56.8	52.7	52.18	3.39
N	60.4	56.0	63.4	52.6	66.6	43.3	68.9	66.9	51.9	76.9	61.6	60.77	8.93
O	56.4	60.4	54.0	61.9	56.2	57.7	63.8	54.5	62.3	59.9	57.8	58.62	3.10
P	41.3	49.0	52.1	46.2	51.9	44.8	58.3	50.9	50.3	60.7	48.7	50.38	5.31
Q	67.4	42.7	56.5	52.6	56.0	56.0	53.6	67.9	43.8	63.1	61.4	56.45	7.93
R	54.6	42.9	56.2	50.7	47.9	42.5	50.3	48.0	53.1	55.0	47.0	49.80	4.45
S	43.0	50.9	52.9	46.3	52.6	45.3	59.6	49.3	50.1	59.4	48.2	50.68	5.04
T	89.0	82.5	83.9	94.6	75.5	74.3	79.7	88.3	88.2	93.2	91.4	84.63	6.80
U	87.7	80.4	82.5	94.3	72.4	71.4	71.1	83.5	78.6	89.3	92.6	82.16	7.91
V	73.3	71.0	76.0	66.6	81.6	45.6	79.2	74.6	66.9	81.7	69.9	71.49	9.61
W	89.4	81.6	81.4	94.8	73.0	71.3	72.9	84.9	78.0	89.9	92.7	82.70	7.93
X	57.7	61.0	77.4	61.0	74.9	61.2	69.0	67.2	50.5	81.7	68.5	66.37	8.80
Y	63.8	50.8	66.3	83.6	67.4	70.4	77.6	55.9	80.8	76.9	69.1	69.10	9.60
Z	67.6	42.5	55.9	52.6	53.7	56.2	44.0	68.3	48.9	62.8	61.6	55.82	8.29
AA	40.7	48.8	50.7	45.7	51.9	44.7	58.1	50.5	50.3	60.7	48.7	50.07	5.39
AB	87.1	81.6	82.0	93.7	75.2	70.5	76.9	84.3	78.0	91.8	92.0	83.00	7.22
AC	86.9	86.3	83.2	93.2	84.3	74.3	81.4	91.0	78.1	95.1	88.9	85.69	6.03

Table 5.5: Evaluation results for all the combinations of parameters for the Speech-Music discriminator system

commercials, films, etc.) labeled as *Music*.

Taking into account all these considerations, the accuracy of the system, with post-processing techniques applied, can be up to 94.3%.

Tests for GSM data

Finally, some GSM audio files have been tested in our system. It is really difficult to give an exact number for the accuracy in this case. As the input audio files are obtained just recording audio with a cellular phone near a loudspeaker, the quality of the GSM codification is unknown. We have seen that better results are obtained when we use the Rhythm Transform descriptors: the accuracy is near 85%. Results are right, but more efforts have to be made in that sense.

Descriptor	L	Value	∇	∇^2
4Hz Modulation	1	*	*	
Spectral Centroid	1	*	*	
Spectral Flatness	1	*	*	
Zero Crossing	1	*	*	
Voice to White	1	*	*	
MFCC	12	*	*	*
Rhythm Transform	12	*		

Table 5.6: Descriptors used for rhythm tests in the Speech-Music discrimination system

#Id	1	2	3	4	5	6	7	8	9	10	11	$\bar{x}$	τ
A	82.7	75.9	81.3	86.4	80.1	71.2	81.6	82.9	78.1	87.9	81.9	80.9	4.43

Table 5.7: Evaluation results for rhythm tests in the Speech-Music discrimination system

5.4 Genre Classification System

5.4.1 Overview

The Genre Classification system has been designed as an extension of the Speech-Music Discrimination system. It is still in a very preliminary phase, but the main characteristics will be discussed here.

As commented in Sec. 3.4, the selection of the right taxonomy is crucial for this kind of systems. In our case, due to the system has to be used with broadcast radio audio files, the next taxonomy will be used:

- Classic: Mozart, Bartok, Back, Savall...
- Dance: Orbital, Dj Jean, Eiffel 65, Moby...
- Pop: David Bisbal, Madonna, Dido...
- Rap Hip-Hop: Eminem, Digital Underground, Lauren Hill...
- Rock: Blur, Sound Garden, Metallica...
- Soul: Alicia Keys, Aretha Franklin, Natalie Cole...
- Speech: interviews, magazines, news...

The system is trained with about 50 different representative songs for each one of these predefined genres. After some tests, the parameterization used is composed by next list of descriptors:

- MFCC (Mel Frequency Cepstrum Coefficients)
- Spectral Flatness
- Spectral Centroid

- Zero Crossing Rate
- Voice 2 White
- Beatedness
- Spectral Centroid (of the Rhythmical description)
- MFCC (Mel Frequency Cepstrum Coefficients of the Rhythmical description)

The number of descriptors is 30 (12+1+1+1+1+1+1+12 respectively). The first sixteen descriptors deals with the spectrum (or timbre) and the other ones deal with rhythm. Note how the system will classify according to both aspects of music.

The length of the frames is 300[ms], the hop-size is 10[ms] and the *hamming* window is used. The Markov models are single-state with eiight Gaussian mixtures per model.

But the new feature of this system is the inclusion of the LDA analysis in order to reduce the number of descriptors(See Sec. 4.5). Furthermore, LDA performs a pre-classification of the input signal, improving the discrimination power of the traditional HMM system located behind the LDA analysis.

5.4.2 Results

From now on, results are not much spectacular, as shown in Table. 5.8.

In Table 5.8, the first four rows belong to four song examples of *classic* music, the second group of four rows belong to four examples of *dance* music, and so on. The output of the automatic classification is computed for each frame, and shown (in percentage) for each one of the examples. Note how results are quite successful for *classical* and *speech* (the first and the last group of rows), where the maximum percentages are given for the right genre. Nevertheless, results for *Soul* music are really bad. Of course, more efforts have to be made in this system.

Finally, a real-time Graphical user interface has also been developed for this Automatic Genre Classification System (See Fig. 5.7)

5.5 Rhythm Similarity System

5.5.1 Overview

As mentioned in Sec. 5.1.1, AIDA is a system for Automatic Recognition of audio. AIDA is the base of this similarity system because any identified song can be interpreted as “the most similar song”. Furthermore, if the used descriptors are a representation of the rhythm of the input audio, AIDA is transformed to a Rhythmic Similarity System.

To identify an unknown piece of audio, we use the property of a Hidden Markov Model from what an HMM can be seen as a double stochastic process. Therefore, HMM could be used to generate observations and it is possible to calculate the probability that some observations are generated by a given HMM.

Song	Classic	Dance	Pop	Rap Hip-Hop	Rock	Soul	Speech
# 1	91.9	8.1	0.0	0.0	0.0	0.0	0.0
# 2	94.4	4.0	0.0	0.0	1.6	0.0	0.0
# 3	96.2	3.3	0.0	0.0	0.5	0.0	0.0
# 4	60.5	31.9	0.0	1.2	6.5	0.0	0.0
# 1	0.0	53.6	5.4	0.0	0.0	0.0	41.0
# 2	0.0	82.9	0.4	9.3	1.3	0.0	6.0
# 3	0.5	29.2	9.1	34.7	1.1	0.0	22.6
# 4	0.0	38.6	9.0	31.3	0.1	0.0	21.0
# 1	2.1	51.4	16.6	24.5	0.6	0.1	4.7
# 2	6.6	3.9	59.9	15.3	8.4	0.0	5.9
# 3	2.9	36.5	14.8	11.1	11.9	0.0	22.8
# 4	2.3	23.9	2.6	12.1	38.1	3.2	17.8
# 1	0.0	0.2	12.4	58.1	1.4	0.0	27.8
# 2	1.6	6.5	3.4	78.6	0.5	0.0	9.4
# 3	9.5	0.7	0.0	0.4	0.0	0.0	89.3
# 4	0.7	2.4	3.3	86.5	0.6	0.0	6.6
# 1	1.4	35.2	1.3	33.5	27.5	0.0	1.2
# 2	0.0	66.4	2.0	4.9	3.6	0.0	23.0
# 3	4.1	89.5	0.0	0.2	4.7	0.0	1.4
# 4	0.0	14.9	7.3	6.3	57.2	0.0	14.3
# 1	2.2	0.79	0.0	1.4	0.3	1.4	93.8
# 2	0.7	6.17	7.22	35.4	4.1	0.0	46.3
# 3	24.3	3.13	0.3	7.5	5.2	1.9	57.6
# 4	0.0	0.37	0.0	11.4	1.1	0.0	87.1
# 1	0.59	0.0	0.0	0.0	0.0	0.0	99.4
# 2	0.0	0.0	0.0	0.0	0.0	0.0	100
# 3	0.0	0.0	0.0	0.0	0.0	0.0	100
# 4	0.0	0.0	0.0	0.0	0.0	0.0	100

Table 5.8: Results of the Automatic Genre Classification System

Figure 5.8 represents a sequence of HMM that models a song. Each HMM is a part of the temporal structure belonging to music. Since music can be seen as a sequence of events sorted in time, this music can be modeled with a sequence of HMMs. The evolution in time of the song is represented with the jumps from one state to the next one.

Let's suppose we have an unknown fragment of audio, where O are all the vectors of parameters (i.e. Mel-cepstrum, rhythm features, harmonic structure description, etc.). If a known song in our database is modeled using an HMM λ, the probability that the generation of this song was the same than the generation of the unknown fragment is [45]:

$$P(O \mid \lambda) = \sum_{q_1, q_2, \ldots, q_T} \pi_{q_1} b_{q_1}(o_1)\, a_{q_1 q_2} b_{q_2}(o_2) \cdots a_{q_{T-1} q_T} b_{q_T}(o_T) \tag{5.12}$$

With this equation in mind, the identification process can be seen in the fol-

lowing way. We have a database of HMMs sequences that model our repository. When we are given an unknown audio fragment from which we have derived some observations (melody parameters, mel-cepstrum, etc.), Eq. 5.12 answers the question whether the HMMs of a song λ in the repository would be able to generate that unknown music. This approach has several advantages over a more classical matching approach. The first advantage is that it is more robust to noise because two songs can be modeled in a very different way because one is noisy and the other one is clean, and however the identification process will give the right result. Another advantage is that the same process can be used to retrieve audio similarity as explained in [19].

The identification algorithm matches an input streaming audio against all the fingerprints to determine whenever a song section has been detected. The Viterbi algorithm is used again with the purpose of exploiting the observation capabilities of the HMM models contained in the fingerprint sequences. Nevertheless, this time the model is not a complete graph but the HMM ring shown in Fig. 5.9. In this structure, each HMM only has two links, one to itself and one toward its immediate neighbor. The identification algorithm scales linearly with the number of songs in the database because no backtracking is required for single path models.

5.5.2 Features

The used features are obviously related with the rhythm. In our case, the first 150 bins of data in rhythm domain (See Sec. 5.2.2) are used. This is an experimental value, but after some tests we found that, inside these 150 bins, there is enough information for a successful rhythmic parameterization. The Beatedness information is not used at this moment.

5.5.3 Training of the system

The biggest problem that arises with a music recognition scheme based on HMMs is how to find the more suitable HMM set that will lead to a good recognition even in bad noisy environments. The training of a speech recognition system has still some issues but it is a well studied problem. In speech, the target for each HMM is a phoneme (or other phonetic related characteristic), but in music there are not such "phonemes". In [19], the author presents a way to define some properties for the units that can suit the music identification problem as well as music similarity.

To automatically derive some good units to represent the music, we follow an iterative approach based on the EM algorithm [15].

The algorithm is composed by several steps:

1. Number of different HMM: The first decision has to be taken and it is the number of different HMM that will be used. In other identification tasks, this decision is easy and usually an HMM for each phoneme is used. In the music identification problem, since there are no "phonemes", the number of HMM has to be carefully chosen. The higher the number of HMMs used, the higher the accuracy of the models for each song, but this would mean also more complexity for both the training and identification. If the numbers of HMM is very low, the accuracy of the representation will be

Type	Author	Song
Original Song	Alicia Keys	A woman's worth
Similarity #1	Alicia Keys	A woman's worth
Similarity #2	Enigma	Morphing Thru Time
Similarity #3	Staind	Epiphany
Original Song	Albert Pla	Joaquin el Necio
Similarity #1	Albert Pla	Joaquin el Necio
Similarity #2	Argelis	Amiga mia
Similarity #3	M. McCain	Nana

Table 5.9: Results of the Rhythmical Similarity System

very poor and the system will need a longer fragment of unknown audio to identify it. In our case the number of HMM is set to 1024.

2. Initialization: Originally the bootstrap models we used were pure random and all the means and variances were chose at random and the transition probabilities were set to 0.5 for both stay and jump. Unfortunately, this methods lead often to a local maximum that is not good enough for identification because the discrimination capacity for each HMM was very poor. A second method of k-means to create the bootstrap parameters shows a very good performance.

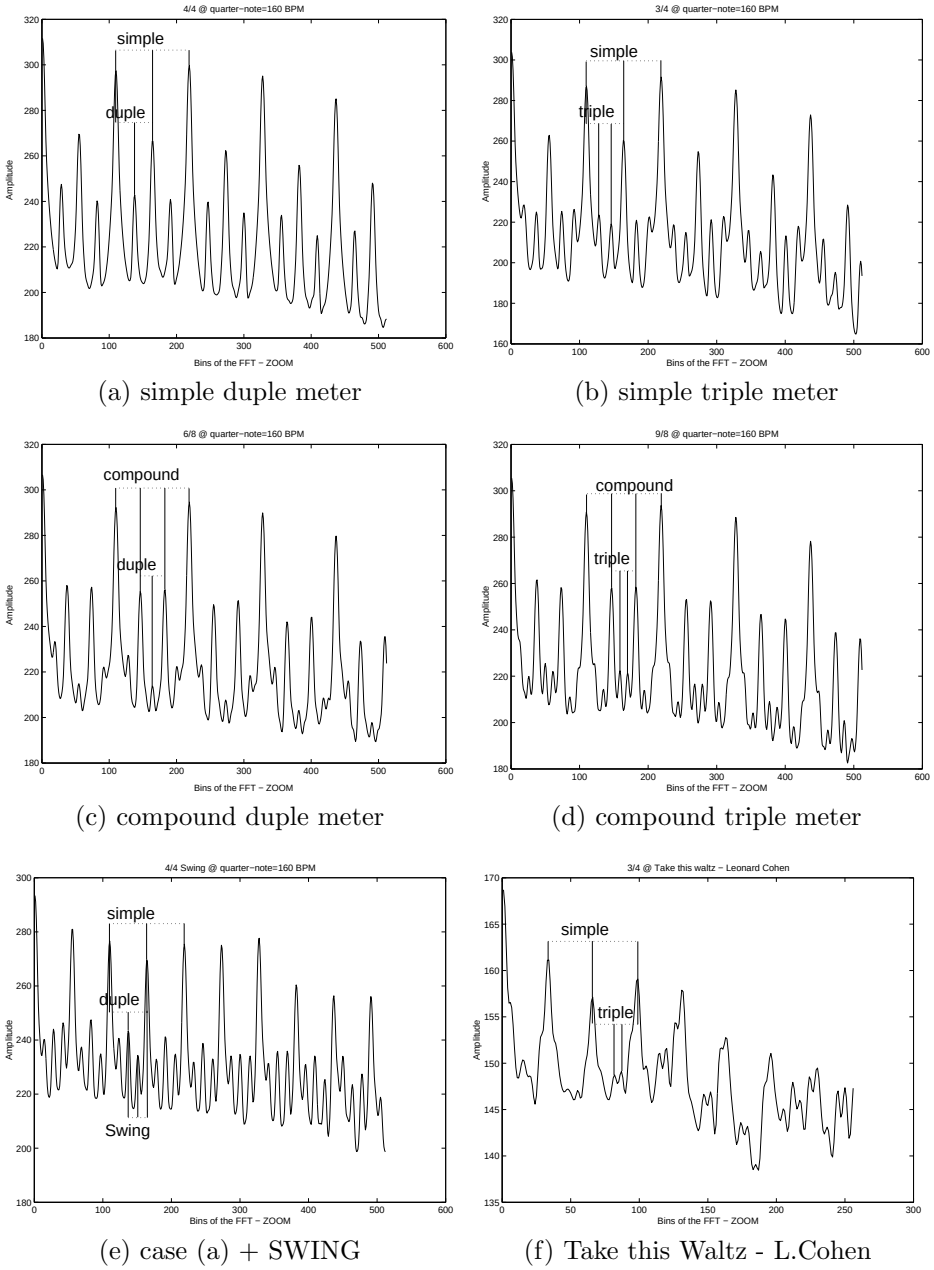
3. Realignment With the current parameters, the system calculates a new sequence of HMM in order to increase the observation probability. This is done with the Viterbi algorithm [72].

4. Update: With the alignment calculated in the previous step, we use the Baum-Welch algorithm to update the means, variances and transition probabilities.

5. Loop: Steps 3 and 5 are repeated until the global probability of generation is not growing from one iteration to the next one.

5.5.4 Results

Few experiments have been made. The problem of analyzing results with this system is that, with a database of about 5800 songs, how possible is to confirm that the found excerpts of audio are really the most similar ones to the original, from a rhythmical point of view? Similarity is an ambiguous concept. Then, numerical results are also ambiguous. Nevertheless, some results are shown in Table 5.9.



(a) simple duple meter (b) simple triple meter

(c) compound duple meter (d) compound triple meter

(e) case (a) + SWING (f) Take this Waltz - L.Cohen

Figure 5.4: Examples of data in Rhythm Domain for different cases

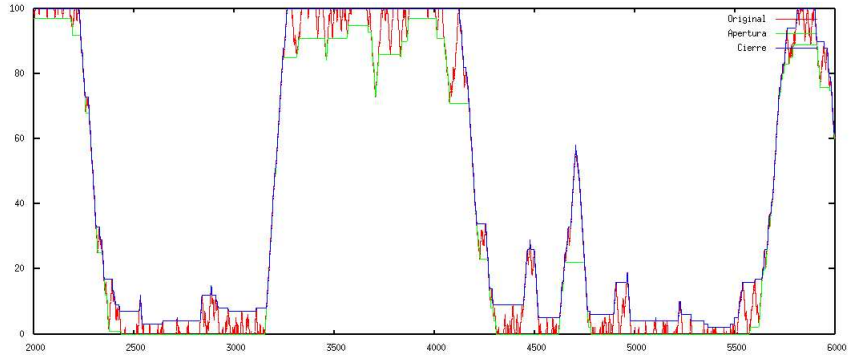


Figure 5.5: *opening* and *closing* operations on the Speech-Music discrimination system

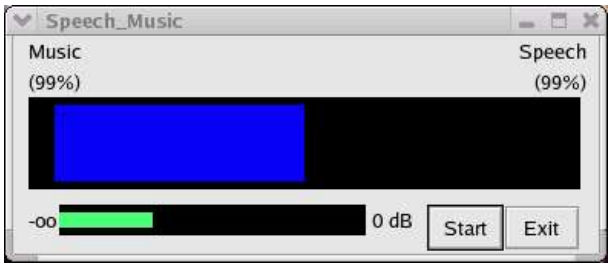


Figure 5.6: Graphical User Interface for the Speech-Music discrimination system

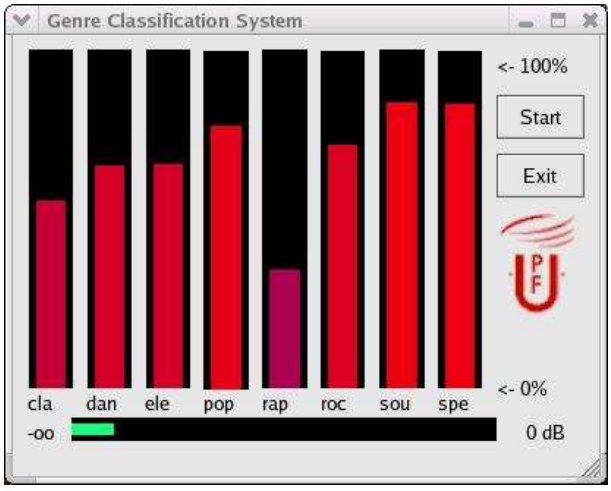


Figure 5.7: Graphical User Interface for the Automatic Genre Classification system

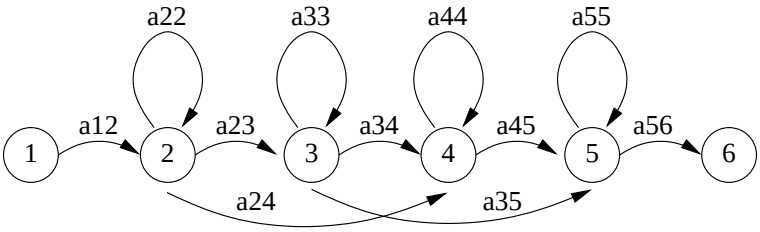


Figure 5.8: Song representation with an HMM sequence

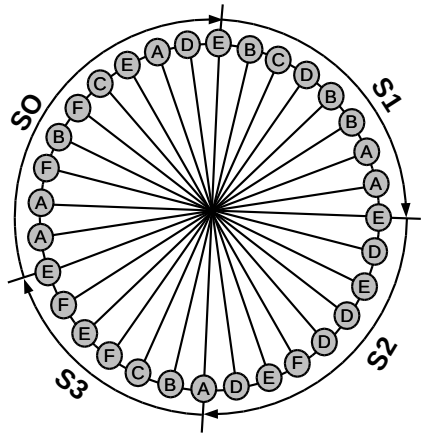


Figure 5.9: Song Model for the Rhythmic Similarity system

Chapter 6

Conclusions and future work

6.1 Summary

Some initial works have been presented in Chapter 5. These works are basically focused on the design of new (both high level and low level) descriptors or tools and on the application of these descriptors in the Music Information Retrieval field. The goals of all this preliminary research are not the tools or applications themselves, but they are the basis for future research (See Sec. 6.2). The initial works can be summarized as:

Voice2White: This descriptor has been specially developed for the Speech-Music discriminator. It is well known that speech data has a limited frequency band, from 300$[Hz]$ to 4000$[Hz]$ approximately. The Voice2White descriptor is a measure of the energy inside this limited frequency band respect to the whole audible margin. It will give an idea about how *speechy* is the input audio data.

Rhythm Transform: This tool has been developed for rhythmical representation of audio data. The main goal of *Rhythm Transform* is that the input data can be transformed to the so called *rhythm domain*. The Rhythm Transform provides information in the *rhythm domain* in the same sense that the Fourier Transform provides information in the frequency domain. Therefore, many descriptors can be extracted from data in the rhythm domain and some other rhythmical properties of the input signal can be obtained (BPM, time-signature...)

Beatedness: The Beatedness is a measure of the rhythmicity of the input data. It is based on the rhythm transform and it is computed as the Spectral Flatness of data in the rhythm domain. As the Spectral Flatness is a measure of the brightness of data in spectral domain, the Beatedness is a measure of the brightness of data in the rhythm domain, that is, the rhythmicity. The Beatedness is a good example of how data in rhythm domain can be used.

Speech-Music Discrimination: This is the first global system we have developed. By using the new descriptors mentioned above and using Hidden Markov Models for classification, a real time Speech-Music discriminator has been developed. Some post-processing techniques like mathematical morphology have been used in order to adapt it to the requirements of the AIDA system. Results for this system are quite successful (94.5%). The errors are biased to classic music: the system tends to recognize classic music as speech.

Automatic Genre Classification: This system is a generalization of the Speech-Music discrimination system. Similar descriptors (including those descriptors that can be obtained from data in rhythm domain) and Hidden Markov Models are also used here. The main difference can be found in the number of predefined models: instead of only two models (speech - music), eight different models are used in the classification process (classic - dance - electronic - pop - rap - rock - soul - speech). Since the classification is more complicated, Linear Discriminant Analysis has been included. Although results are quite good, more efforts have to be done in this sense.

Rhythmical Similarity: In the context of the AIDA system (Automatic Identification of Audio), the identification process can be seen as a similarity process. When similarities are found under some specific constrains, the similar song is labeled as the identified song. The similarity system can be implemented with the help of the Rhythm Transform and the Hidden Markov Models. The main idea is slightly different from classical approaches: instead of the identified song, an ordered list of all the (rhythmically) similar songs is shown. The obtained results are quite satisfactory, but it is really difficult to quantify them because the similarity concept is intrinsically confusing.

6.2 Future Work

6.2.1 Main Idea

The common objective for all the contributions described above is to study automatic genre classification but including the musicological point of view. Most of the people in Music Information Retrieval community are great enthusiast of music. The problem arises when music meets computers. Sometimes, the work made by MIR community has been done without taking into account the point of view of the musicians. Why not to ask them for their impressions? In this context, new ways for music description have to be found, and musicians must understand them.

The research should be be focused in both musical genre definition and its computational description. Successful results will appear when the computational description of a genre will be able to (musically speaking) identify it. This description has to be human comprehensible.

For this purpose, musical and technological studies have to be done and some tools will be used, as shown in the next section.

Planning

The future work will be organized as follows:

Musicological studies: Lots of musicological studies about genre have been done, but the main problem is that some genres are fully documented while others are not. Musicians and music enthusiasts have to be asked for the specific features that can define a Genre. This information has to be collected and a general (but complete) overview for all the selected genres has to be carried out. The selection of the right taxonomy will be another important task.

Recordings: In order to study the detailed characteristics for all the proposed genres, a complete audio database is needed. This database can be divided into two main groups, depending on the specific genre description features:

- Common recordings for those cases in which the specific features are a general property of the music, that is, a musical form or a *how to play*.
- Specific recordings for those cases in which a concrete feature of music is crucial for genre description, that is, a particular rhythm pattern or the sound of a specific instrument.

Descriptors: All this audio data have to be analyzed. The first step in the analysis process is the audio description. There are several available audio descriptors but some of them are really far to provide a musical meaning. These descriptors have to be combined and reinterpreted for providing them this musical meaning. The descriptors that will be used for this purpose are divided as shown:

- Perfectly known descriptors: Some descriptors like MFCC or Spectral Centroid will be used. All of them are widely used in the MIR community.
- Adapted descriptors: Some descriptors can be adapted to emphasize a specific feature of a specific genre, according to the conclusions of the musicological studies to be obtained by the first step of this research.. For instance, song tonality descriptors can be adapted, if they are short enough, to an harmonic descriptor based on two chords in a bar [32][33].
- New descriptors: With the same purpose than the adapted descriptors, some new and very specific descriptors may be implemented. For instance, a syncopation description could be useful to detect dance music.

Building models: With the right selection of descriptors, a different model for each specific genre has to be built. This model must show all the musical aspects that contribute to define a genre, and must be able to be directly compared with other models. Since fusion is a general characteristic of music and most of the composers search for new formulas in their compositions, they are never perfectly located in a specific style but in a distance between two or three.

Interpretation: Each genre should be perfectly defined by these meaningful models. It is obvious that an objective description of music has to be included in this research, but the interpretation of the resulting models is much more important. If they prove to be good enough, this interpretation can be associated to the musical cognition process, and a first approximation of this will be available. Then, several musicological, psychological and technological studies can be derived from that.

Note that we have not defined how the models are supposed to be. They will be defined according to the results of the musicological studies, and created according to the results of different tests performed during the research.

Tools

Some of the descriptors commented in Chapter 4 are low-level descriptors while the used descriptors to generate the models should be high-level descriptors. Note that this system does not need low-level descriptors like score detection, BPM detection or instrument detection algorithms. This is because the scores, BPM, etc. are representations of a little part of the music. For our purpose, we need high-level descriptors: it is more important to know whether the rhythm of a song is *walking*, whether a melody is *happy* or whether a timbre is *brilliant*. These descriptors will be computed and tested by using the AMADEUS technology.

On the other hand, the models will be generated by using Hidden Markov Models (HMM). Since HMM can be seen as a double embedded stochastic process, they can be understood as a set of generators that describe a specific property of each specific genre. Gaussian Mixtures will help for that purpose. Distances between models can be inferred because HMM are statistical tools, but different strategies will be taken into account..

Some other tools like Linear Discriminant Analysis or Mathematical Morphology will be used in some specific parts of the whole process. Finally, Neural Networks or other classification techniques will also be considered.

Bibliography

[1] *Cambridge international dictionary of english online; 2002; http://dictionary.cambridge.org.*

[2] *Merrian-webster's collegiate dictionary; 2002; fttp://www.m-n.com.*

[3] Schloss A. On the automatic transcription of percussive music – from acoustic signals to high-level analysis. Technical report, CCRMA, 1985.

[4] Adam L. Berenzweig and Daniel P.W. Ellis. Locating Singing Voice Segments within Music Signals. *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, October 2001.

[5] Alan V. Oppenheim and Ronald W. Schaffer. *Discrete-Time Signal Processing.* Prentice Hall International Inc, 1989.

[6] P. Allen and R. Dannenberg. Tracking musical beats in real time. In *Oric. ICMC*, 1990.

[7] J.J. Aucouturier and F. Pachet. Musical genre: A survey. 32(1), 2003.

[8] L. E. Baum and J. A. Eagon. An Inequality with Applications to Satistical Estimation for Probabilistic Functions of Markov Processes and to a Model for Ecology. *BAMS*, pages 360–363, 1967.

[9] Jeff A. Bilmes. *Timing is of the Essence: Perceptual and Computational Techniques for Representing, Learning, and Reproducing Expressive Timing in Percussive Rhythm.* PhD thesis, MIT, 1993.

[10] P. Cano, M. Koppenberger, S. Ferradans, A. Martinez, F. Gouyon, V. Sandvold, V. Tarasov, and N. Wack. Mtg-db: A repository for music audio processing. In *Proceedings of 4th International Conference on Web Delivering of Music*, Barcelona, Spain, 2004.

[11] E. Seashore Carl. *Psychology of Music.* Dover Publications INC, 1967.

[12] W Chow and L.Gu. Robust Singing Detection in Speech/Music Discriminator Design. *Proc. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, 2:865–868, 2001.

[13] E. Clarke. *Rhythm and Timing in Music.* The Psychology of Music, series in Cognition and Perception. Academic Press, 1999.

[14] Grosvenor Cooper and Leonard B. Meyer. *The Rhythmic Structure of Music.* The University of Chicago PressD, 1960.

[15] A. P. Dempster and et altri. Maximum Likelihood from Incomplete Data via the EM Algorithm. In *Journal of the Royal Statistical Society*, volume 39, pages 1–38, 1977.

[16] Ronald A. DeVore and Bradley J. Lucier. *Wavelets*.

[17] S. Dixon. A beat tracking system for audio signals. *Conference on Mathematical and Computational Methods in Music*, 1999.

[18] S. Dixon. Automatic extraction of tempo and beat from expressive performances. *Journal of New Music Research*, 30(1), 2001.

[19] E. Guaus E. Batlle and J. Masip. Open position: Multilingual orchestra conductor. lifetime opportunity. In *Proceedings of 26th ACM/SIGIR International Symposium on Information Retrieval*, Toronto, Canada, 2003.

[20] E. Guaus E. Batlle, J. Masip. Amadeus: A scalable hmm-based audio information retrieval system. In *Proceedings of First International Symposium on Control, Communications and Signal Processing*, Hammamet, Tunisia, 2004.

[21] Eisenberg, Gunnar; Batke, Jan-Mark; Sikora, Thomas. Beatbank - an mpeg-7 compliant query by tapping system. In *Proceedings of the 116th. AES Convention*, 2004.

[22] Eric D. Scheirer. Tempo and Beat Analysis of Acoustical Musical Signals. *J. Acoust. Soc. Am.*, 103(1):558–601, Jan 1998.

[23] J. Foote and S. Uchihashi. The Beat Spectrum: A New Approach to Rhythm Analisis. In *Proc. International Conference on Multimedia and Expo*, 2001.

[24] J. Foote; M. Cooper and U. Nam. Audio Retrieval by Rhythmic Similarity. In *Proc. ISMIR*, 2002.

[25] P. Fraisse. *Rhythm and Tempo*, chapter 6. The Psychology of Music, series in Cognition and Perception. Academic Press, 1982.

[26] François Pachet and Daniel Cazaly. A Taxonomy of Musical Genres. April 2000.

[27] A. Friberg and J. Sundström. Swing ratios and ensemble timing in jazz performances: Evidence for a common rhythmic pattern. *Music Perception*, 19(3), 2002.

[28] S. Balakrishna A. Ganapathiraju. *Linear Discriminant Analysis - A Brief Tutorial*. Mississippi State University.

[29] Gethin Williams and Daniel P.W.Ellis. Speech/Music Discrimination based on Posterior Probability Features. In *Proceedings of Eurospeech*, pages 687–690, September 1999.

[30] Fabien Gouyon. *Towards Automatic Rhythm Description of Musical Audio Signals. Representations, Computational Models and Applications*. PhD thesis, Music Technology Group. Pompeu Fabra University, 2003.

[31] M. et al. Grimalidi. Classifying music by genre using a discrete wavelet transform and a round-robin ensemble. Technical report, Trinity College, University of Dublin, 2003.

[32] E. Gómez. *Melodic Description of Audio Signals for Music Content Processing*. PhD thesis, 2002.

[33] E. Gómez and P. Herrera. Automatic extraction of tonal metadata from polyphonic audio recordings. In *Proceedings of 25th International AES Conference*, London, UK, 2004.

[34] L. Hofmann-Engl. Rhythmic similarity - a theoretical and empirical approach. In *ICMPC7*. Keele University, 2002.

[35] Henkjan Honing. From time to time: The representation of timing and tempo. *Computer Music Journal*, 25(3), 2001.

[36] John Sloboda Irène Deliège, editor. *Perception and Cognition of Music*. Psychology Press, 1997.

[37] Jang, Jyh-Shing Roger; Lee, Hong-Ru; Chen, Jiang-Chun. Super mbox: An efficient/effective content-based music retrieval system. In *The 9th. ACM Multimedia Conference*, 2001.

[38] Jang, Jyh-Shing Roger; Lee, Hong-Ru; Yeh, Chia-Hui. Query by tapping: A new paradigmfor content-based music retrieval from acoustic input. In *Proceedings of the 2nd. IEEE Pacific-Rim Conference on Multimedia*, 2001.

[39] Johnston J.D. Transform coding of audio signals using perceptual noise criteria. *IEEE Journal on Selected Areas in Communications*, 6(2):314–323, Feb. 1998.

[40] B Kedem. Spectral Analysis and Discrimination by Zero-Crossings. In *Proceedings of the IEEE*, volume 74, November 1986.

[41] Karin Kosina. Music genre recognition. Master's thesis, Medientechnik und - design, 2002.

[42] Lambrou T.; Kudumakis P.; Speller R.; Sandler M.; Linney A. Classification of audio signals using statistical features on time and wavelet transform domains. In *Proc. ICASSP, IEEE*, volume 6, pages 3621–3624, 1998.

[43] J. Laroche. Estimating Tempo, Swing and Beat Locations in Audio Signals. In *Proc. WASPAA*, pages 135–139, 2001.

[44] Lawrence E. Kinsler et. Al, editor. *Fundamentals of Acoustics*. John Wiley and Sons, 4 edition, 1999.

[45] Lawrence R. Rabiner. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. In *Proceedings of the IEEE*, volume 77, pages 257–286. February 1989.

[46] F. Lerdahl and R. Jackendoff. *A generative theory of tonal music*. The MIT Press, 1983.

[47] E. S. Paris M. J. Carey and H. Lloyd Thomas. A comparison of features for speech,music discrimination. In *Proc. ICASSP*, volume 1, pages 149–152, 1999.

[48] Masataka Goto and Yoichi Muraoka. A Beat Tracking System for Acoustic Signals of Music. In *Proc. ACM Multimedia*. ACM, 1994.

[49] Cory McKay. Automatic classification of musical genre: A review of the current state of the art. Technical report, McGill University, 2003.

[50] Montgomery y Runger. *Probabilidad y Estadística aplicadas a la ingenieria*. Limusa Wiley, 2002.

[51] Stefaan Lippens; George Tzanetakis; Jean-Pierre Martens; Tom De Mulder. A comparison of human and automatic musical genre classification (abstract). *ICASSP*, 2004.

[52] Ozgur Izmirli. Using Spectral Flatness Based Feature for Audio Segmentation and Retrieval. Technical report, Center for Arts and Technology, Department of Mathematics and Computer Science, Connectucut College, 1999.

[53] Pedro Cano; Eloi Batlle; Harald Mayer and Helmut Neuschmied. Robust Sound Modeling for Song Detection in Broadcast Audio. In *112th AES Convention*, May 2002.

[54] J.W. Picone. Signal Modeling Techniques in Speech Recognition. *Proceedings of the IEEE*, 81(9), September 1993.

[55] Walter Piston. *Harmony*. W. W. Norton and Company, 5 edition, 1987.

[56] P.E. Hart R.O. Duda. *Pattern Classification and Scene Analysis*. John Wiley & Sons, 1973.

[57] D. Perrot; R.O.Gjerdigen. An exploration of factors in the identification of musical style. In *Proc. Society for Music Perception and Cognition*, 1999.

[58] Stanley Sadie and John Tyrrell, editors. *The New Grove Dictionary of Music and Musicians*. 2001.

[59] J. Saunders. Real-Time Discrimination of Broadcast Speech/Music. In *Proc. ICASSP*, pages 993–996, 1996.

[60] E. Scheirer and M. Slaney. Construction and Evaluation of a Robust Multifeature Speech/Music Discriminator. In *Proc. ICASSP*, pages 1331–1334, 1997.

[61] Arnold Schonberg. *Theory of Harmony*. University of California Press, 1983.

[62] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, 1982.

[63] Bob Snyder. *Music and Memory: An Introduction*. The MIT Press, 2001.

[64] Stefan Karnebäck. Discrimination between Speech and Music based on a Low Frequency Modulation feature. In *Proceedings of Eurospeech*, 2001.

[65] Stephen Handel. *Listening: An Introduction to the Perception of Auditory Events*. The MIT Press, 2 edition, 1991.

[66] Steve Young et. al. The HTK Book (for HTK Version 3), July 2000.

[67] The MathWorks, inc. *Statistics Toolbox*, 1994 - 2001.

[68] G. Tzanetakis and P. Cook. Musical Genre Classification of Audio Signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, July 2002.

[69] G. Tzanetakis; G. Essl and P. Cook. Automatic Musical Genre Classification of Audio Signals. In *Proceedings ISMIR*, 2001.

[70] G. Tzanetakis; G. Essl and P. Cook. Human perception and computer extraction of musical beat strength. In *Proc. DAFx-02*, September 2002.

[71] Brani Vidakovic and Peter Müller. *Wavelets for Kids: a Tutorial Introduction*. Duke university.

[72] A. J. Viterbi. Speaker Recognition: A Tutorial. In *IEEE Transactions on Information Theory*, volume 13, 1967.

[73] P. Cano; M. Koppenberger; P. Herrera; O. Celma; V.Tarasov. Sound effect taxonomy management in production environments. In *Proceedings of 25th International AES Conference London, UK*, 2004.

[74] www.cs.waikato.ac.nz/ ml/weka/index.html.

[75] Javier Blanquez y Omar Morera, editor. *Loops, una historia de la música electrónica*. Reservoir Books, 2002.

[76] M. Yeston. *The stratification of musical rhythm*. Yale University Press, 1976.

[77] E. Zwicker and E. Terhardt. Analytical expressions for critical-band rate and critical bandwidth as a function of frecuency. *J. Acoust. Soc. Am.*, 68(5):1523–1525, 1980.

Appendix A

Related publications

- Batlle, E. Masip, J. Guaus, E. 'Amadeus: A Scalable HMM-based Audio Information Retrieval System'. Proceedings of First International Symposium on Control, Communications and Signal Processing. 2004. Hammamet, Tunisia.
- Guaus, E. Batlle, E. 'A non-linear rhythm-based style classifcation for Broadcast Speech-Music Discrimination'. Proceedings of AES 116th Convention. 2004. Berlin, Germany.
- Guaus, E. Batlle, E. 'Visualization of metre and other rhythm features'. Proceedings of IEEE Symposium on Signal Processing and Information Technology. 2003. Darmstadt, Germany.
- Batlle, E. Guaus, E. Masip, J. 'Open Position: Multilingual Orchestra Conductor. Lifetime Opportunity'. Proceedings of 26th ACM/SIGIR International Symposium on Information Retrieval. 2003. Toronto, Canada.
- Batlle, E. Masip, J. Guaus, E. 2002. 'Automatic Song Identification in Noisy Broadcast Audio'. Proceedings of IASTED International Conference on Signal and Image Processing. 2002. Kauai, Hawaii, USA.